TOWARDS EXPERT-LEVEL ANALOG PHYSICAL DESIGN: LARGE LANGUAGE MODEL AS OPTIMIZER

Anonymous authors
Paper under double-blind review

ABSTRACT

Analog physical design requires coordinated decisions about device placement, routing, and the parasitics that determine circuit performance. Yet much of its automation optimizes individual stages, leaving engineers to reconcile downstream electrical failures with earlier layout choices. We introduce LAMBDA, a large language model based analog physical design framework that combines global reasoning with fast automated execution. To connect reasoning with execution, a customized B* Graph unifies analog constraints, placement, and routing in an executable search space, enabling a multimodal LLM to optimize coupled design decisions using circuit knowledge, layout observations, and accumulated experience. Multi-fidelity optimization couples low-cost physical and parasitic models with detailed electrical evaluation feedback, balancing exploration efficiency with post-layout performance. On nine commercial 28-nm circuits, LAMBDA improves geometric mean full layout area and wirelength and the composite analog electrical score over SOTA baselines.

1 INTRODUCTION

Analog and mixed-signal (AMS) circuits connect continuous physical signals to digital computation, making them indispensable to sensing, communication, and data conversion. Analog circuitry may occupy only a limited share of an AMS chip, yet its physical design can determine whether the entire chip meets its specifications. An illustrative industry estimate suggests that the analog portion accounts for approximately 75% of overall chip-design risk (Gielen, 2024). This disproportionate burden arises in part from long design cycles and labor-intensive layout: placement and routing, as shown in Figure 1(a), affect parasitics, mismatch, and coupling that can undermine an otherwise correct schematic. Achieving design closure therefore requires engineers to repeatedly reconcile circuit performance, analog constraints, and technology rules, as shown in Figure 1(b).

To reduce this manual effort, extensive research has sought to automate analog physical design. Due to different optimization objectives and objects among different physical design stages, current design automation flows separate it into constraints definition, placement and routing (Chen et al., 2021; Kunal et al., 2019). Given the design constraints, the placement stage models each device as a bounding box and determines its geometric arrangement by minimizing total area and estimated wirelength, without constraint violations (Mangalagiri et al., 2024; Xu et al., 2026). At the routing stage, the layout is modeled as a routing graph, and the physical realization of interconnects is determined by minimizing routing area, wirelength, and parasitics, subject to the design constraints (Xu et al., 2025; Chen et al., 2021; Kunal et al., 2019). These stages are performed sequentially, as shown in Figure 1(c). Only after routing is completed are parasitic extraction (PEX) and post-layout simulation performed to evaluate the quality of the analog physical design. If the post-layout performance fails to meet the required specifications, the flow must return to earlier stages for adjustment. Consequently, this iterative process suffers from low design efficiency.

To improve design efficiency and quality, recent studies have employed artificial intelligence (AI) models to predict circuit performance (Shahane et al., 2023) and incorporate layout effects into early-stage optimization (Mehradfar et al., 2025). However, reliable generalization across circuit topologies and process technologies remains a challenge. Besides, Large language model (LLM)-based agents, which integrate reasoning with tool use (Yao et al., 2023; Schick et al., 2023), have also been applied to identify post-layout performance bottlenecks and automatically plan corrective

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

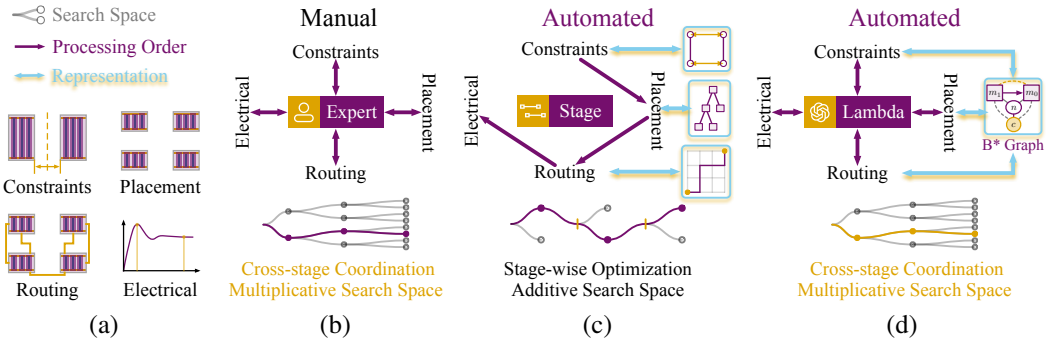


Figure 1: Analog physical design paradigms. (a) Design stage and object. (b) Cross-stage coordination in slow manual iteration to produce high-quality layouts. (c) Ordered stage execution in automated design, but locally optimal early-stage decision. (d) LAMBDA uses LLM-guided cross-stage coordination to produce high-quality layouts at moderate design cost.

actions in a decoupled, sequential framework (Liu et al., 2025; Zhang et al., 2026). However, such automated flows fix decisions made at earlier stages and optimize subsequent stages in sequence. Although this approach reduces the number of combinations explored and accelerates execution, a locally optimal early-stage decision may rule out superior downstream solutions.

To recover design options excluded by sequential optimization, early work explored simultaneous analog placement and routing using simulated annealing (SA) (Ou et al., 2013). Exploring this larger space can be costly when candidate evaluation requires detailed routing, PEX, and post-layout simulation. Related work studies LLMs as optimizers (Yang et al., 2024), iterative refinement from feedback (Shinn et al., 2023; Madaan et al., 2023), graph editing (Zheng et al., 2025; Li & Cao, 2026), and structured geometric reasoning (Zhang et al., 2025; Gong et al., 2026). Multi-fidelity optimization further uses inexpensive approximations to guide costly evaluations (Kandasamy et al., 2017; Takeno et al., 2020). Applying these ideas to joint analog physical design requires a unified, executable representation that exposes the coupling among analog constraints, device placement, and routing and connects design revisions to physical and electrical feedback.

To address this challenge, we propose LAMBDA, an LLM-guided analog physical design framework (Figure 1(d)). First, we introduce a B* Graph that unifies analog constraints, device placement, routing spaces, and net connectivity in an executable representation. Second, we use a multimodal LLM to optimize this graph through coupled revisions of constraints, placement, and routing, guided by circuit knowledge, layout observations, and accumulated evaluation feedback. Third, we develop a multi-fidelity evaluation loop that combines low-cost physical metrics and parasitic estimates with detailed physical verification, PEX, and post-layout simulation. Detailed electrical feedback guides graph refinement, while PEX measurements calibrate subsequent estimates. Experiments on nine circuits in a commercial 28-nm process against five baselines demonstrate improvements in both layout compactness and post-layout performance: LAMBDA reduces geometric mean full layout area by 19.6%–80.1% and wirelength by 9.8%–87.9%, and improves the composite post-layout electrical score for analog circuits by 25.5%–55.7%.

2 PRELIMINARIES

B* Tree. A B* Tree compactly encodes relative rectangular-block placement as a rooted, ordered binary tree (Chang et al., 2000). A left child lies immediately to its parent's right, while a right child lies above with aligned left edges. Contour-based traversal determines vertical positions without overlap, allowing tree edits to adjust placement without explicit absolute coordinates. However, the tree alone cannot jointly represent design constraints, placement, and routing, thereby limiting global layout understanding and optimization. To address the challenge, Section 3.2 introduces a B* Graph that represents devices, nets, and routing space; its decoding rules are given in Appendix B.

Problem formulation. To enable global analog-layout understanding and optimization toward expert-level analog physical design, we formally formulate the problem: given a sized analog circuit design D, a testbench $\mathcal{T}$, and expert-defined quality thresholds $\mathbf{q}$, automatically and jointly deter-

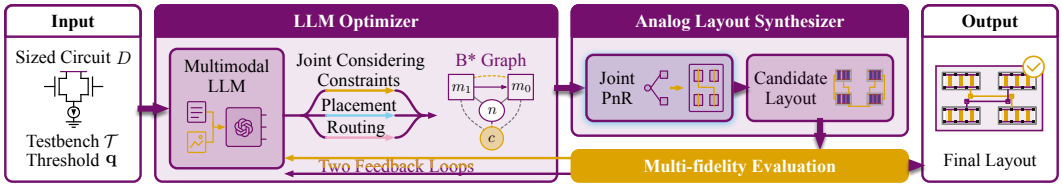


Figure 2: Overall flow of LAMBDA.

mine *placement* (device positions and orientations) and *routing* (physical connections within each net, paths, layers, widths, and vias) to produce a complete layout L. The objective is to minimize layout area and routed wirelength and maximize post-layout electrical performance, while satisfying physical legality, circuit functionality, and the prescribed quality thresholds $\mathbf{q}$.

3 METHOD

3.1 OVERVIEW

As shown in Figure 2, LAMBDA takes a sized circuit D, a fixed testbench $\mathcal{T}$ defining operating conditions (e.g., supply voltage, stimulus, and load), and fixed physical and electrical quality thresholds $\mathbf{q}$. Using circuit context and analog knowledge, a multimodal LLM jointly reason about analog constraints, placement, and routing, encoding these coupled decisions in a B* Graph: a unified, semantically meaningful representation links device placement to routing resources and embeds analog constraints, enabling coordinated physical edits. The analog layout synthesizer then converts the edited B* Graph into a candidate layout through joint placement and routing (PnR). Physical-aware multi-fidelity evaluation guides the LLM in refining the B* Graph through two feedback loops, connecting physical metrics, verification diagnostics, and post-layout electrical responses to coordinated physical refinement. The final layout is returned once it meets all thresholds $\mathbf{q}$.

3.2 B* GRAPH REPRESENTATION

Graph structure. B* Graph extends the B* Tree placement skeleton with nets, rectangular routing spaces, and space-allocation and analog-constraint edges. Block–space boundaries and space–space adjacency link placement to routing passages, allowing routing demands to reshape device spacing. In the B* Graph $G = (V, E)$, nodes represent blocks b (devices or symmetry groups), nets n, or routing spaces c; edges encode relative placement, pin connectivity, routing-space relationships, or analog constraints (Table 1(a)).

PnR representation. Block nodes and parent–child edges form the B* Tree placement skeleton of B* Graph, while nets record pins, allowed layers, and wire widths. Allocation and adjacency edges specify routing spaces and legal transitions. For example, in Figure 3, block b_0 represents device m_0; its right child is the symmetry-group block u, containing devices m_1, m_2 above m_0, and its left child is block b_1, representing device m_3 to the right. Sky blue highlights nodes m_0, m_1 and m_2 from B* Graph and their corresponding layout placement. Similarly, pink highlights the pin edges connecting the shared-source net n_s to pins p_0^d, p_1^s, p_2^s (d, s, g denote drain, source, and gate) in the B* Graph and the corresponding routing in the layout, while allocation edges assign routing-space nodes c_1, c_4, c_5 to this net. Nets may share routing spaces while remaining electrically separate.

Analog constraint representation.

B* Graph encodes analog constraints including pair, self-, and common symmetry, signal-flow ordering, minimum spacing, and route matching (Table 1(b)). In Figure 3, devices m_1, m_2 form a mirror pair within block u, while m_0 is self-symmetric within block b_0. The right-child edge from b_0 to u aligns the two blocks' symmetry axes, with the mirrored pair positioned on opposite sides of m_0's axis. Gold highlights boundary edges connecting space node c_8 to block b_1 at m_3's top side and to block u at m_2's bottom side. Together, these edges encode the local minimum-spacing requirement g_{c_8}, a lower bound on the vertical separation between the two facing sides. Increasing g_{c_8} from 2 to 3 shifts block u upward and lengthens the mirrored branches of net n_s while preserv-

Table 1: B* Graph representation.

(a) Nodes and edges.

Symbol	Type	Meaning	Symbol	Type	Meaning
□	**Block node** b	A device or symmetry group.	---	**Boundary edge**	A block side's contact with a space and its local minimum-spacing requirements.
○	**Net node** n	Connected pins and allowed routing layers and wire widths.	—	**Adjacency edge**	A shared free boundary or legal via connection between spaces.
◯	**Space node** c	A free rectangle on a routing layer.	---	**Allocation edge**	A space available to a net.
→	**Tree edge**	An L/R parent–child relation between blocks.	--→	**Constraint edge**	An analog constraint between blocks or nets.
—	**Pin edge**	A device pin's connection to a net.			

(b) Analog constraints.

Analog constraint	Graph encoding	Physical condition
Pair symmetry	Paired devices and a block axis	Mirrored device geometry, including pins
Self-symmetry	A device paired with itself	Device geometry invariant under reflection
Common symmetry	Constraint edge between symmetry groups u, v	Coincident symmetry axes
Signal-flow ordering	Directed constraint edge ordering u, v along a specified direction	$y_v \geq y_u + h_u$ for upward order
Minimum spacing	Boundary edges identifying facing sides and their local gap requirement g_c	$y_v - y_u - h_u \geq g_c$ for vertical gaps
Route matching	Paired pins and specified matching properties	Mirrored paths or matched lengths, layers, widths, and vias as required

ing the encoded constraints. Gray dashed lines mark the initial positions. Appendices B.6 and B.7 provide geometric definitions and a differential-net matching example.

Analog layout synthesizer. The synthesizer traverses the B* Graph, packing blocks along the occupied-space contour under encoded constraints and reserved gaps. An A* router (Malavasi & Sangiovanni-Vincentelli, 1993) finds legal paths within each net's allocated spaces, which the synthesizer realizes as wires and vias. Routing-driven block displacements or gap enlargements trigger updates to the space partition and net allocations, producing the routed layout L.

3.3 LLM-Guided Joint PnR Optimization

Analog constraints, placement, and routing are tightly coupled: changing a device grouping or spacing can alter pin access, interconnect geometry, and parasitic loading. Evaluating the resulting electrical behavior requires layout synthesis, PEX, and simulation, making the search a costly black-box problem over discrete graph structures and geometric attributes. LAMBDA uses a multimodal LLM to combine analog knowledge, circuit context, and accumulated layout feedback when choosing coordinated edits to the B* Graph. This executable representation gives the optimizer a common decision space for constraints, placement, and routing, leading to the following formulation of the problem in Section 2.

Problem 1 (B* Graph based joint analog PnR) Given a sized circuit D, a fixed testbench $\mathcal{T}$, and quality thresholds $\mathbf{q}$, choose an attributed graph $G = (V, E)$ from the admissible B* Graph space $\mathcal{G}(D)$, preserving the devices, parameters, and connectivity of D. The decisions comprise device grouping into block nodes in V; tree, net–space allocation, and analog-constraint relations in E; and node and edge attributes.

Let $\mathcal{F}_D$ denote the analog layout synthesis mapping. We seek G and its realized layout L that solve

$$\begin{aligned} \underset{G,L}{\text{minimize}} \quad & \mathbf{f}(L;\mathcal{T}) \\ \text{subject to} \quad & G \in \mathcal{G}(D), \quad L = \mathcal{F}_D(G), \quad L \in \mathcal{V}(D,\mathcal{T}), \\ & \mathbf{c}_{\text{ana}}(G,L) \preceq \mathbf{0}, \quad \mathbf{c}_{\text{q}}(L;\mathcal{T},\mathbf{q}) \preceq \mathbf{0}. \end{aligned} \tag{1}$$

Here $\mathbf{f}(L;\mathcal{T})$ collects the circuit's physical and post-layout electrical objectives, all expressed in minimization form. Physical objectives include full layout area, routed wirelength, whitespace, and congestion; electrical objectives include gain, bandwidth, swing, and mismatch. The set $\mathcal{V}(D,\mathcal{T})$ contains layouts passing DRC, LVS against D, and post-layout functional tests under $\mathcal{T}$. The vectors $\mathbf{c}_{\text{ana}}$ and $\mathbf{c}_{\text{q}}$ are residuals for graph-encoded analog constraints and fixed quality requirements, respectively; $\preceq$ denotes componentwise inequality, with nonpositive residuals indicating satisfaction.

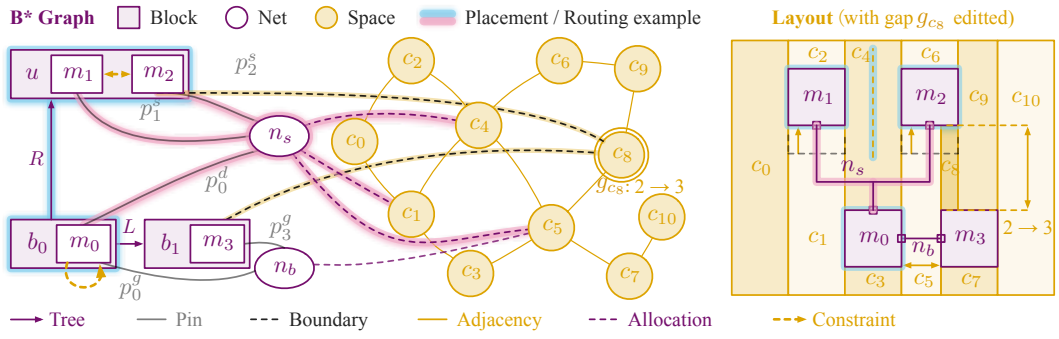


Figure 3: B* Graph example for joint PnR: (a) graph representation; (b) corresponding layout.

To solve Problem 1, LAMBDA performs iterative graph refinement, using layout synthesis and evaluation to assess the effects of coupled design choices. Available knowledge and retrieved layout precedents guide candidate construction, while feedback from earlier candidates provides experience for subsequent refinement. We organize this process as a sequence of decision steps, each informed by a growing history H_t that combines the initial knowledge, retrieved precedents, and available physical tools with previous decisions, candidate graphs, layouts, images, and evaluation results.

At decision step t, LLM conditions on D, $\mathcal{T}$, $\mathbf{q}$, and H_t to choose the next action. It first constructs an editable graph G^{init} and subsequently revises it using this context. Once the current layout is eligible for detailed evaluation, LLM can either continue physical refinement or request verification/simulation according to the procedure in Section 3.4.

Within G^{init}, LLM controls relative placement through device groupings and tree-edge connections, and assigns routing spaces to nets through allocation edges. Pairings, constraint edges, and node and edge attributes express analog relationships, orientations, gaps, routing layers, and wire widths. These choices can be revised together: when a narrow inter-device passage forces a critical net to detour, LLM can increase the minimum-spacing bounds on the corresponding boundary edges and allocate the relevant space nodes to that net, while preserving its symmetry and route-matching requirements. The synthesizer then adjusts device positions to widen the passage and searches the allocated spaces for a more direct connection.

To execute these decisions, fixed postprocessing converts the editable graph into a candidate B* Graph G. Initially, G^{init} contains four directional space nodes per device. After each edit, postprocessing fuses spaces and removes redundant nodes to form a rectangular routing-space partition, updates net–space associations, and validates the graph and its encoded constraints (Appendix B.3). When these checks pass, the analog layout synthesizer produces $L = \mathcal{F}_D(G)$ through joint PnR. A retained space-reference mapping links layout feedback to editable objects in G^{init}. Available objective components and constraint evaluations for $\mathbf{f}$, $\mathbf{c}_{\text{ana}}$, and $\mathbf{c}_{\text{q}}$, together with parasitic estimates, layout images, and failure diagnostics, extend the history to H_{t+1} and guide the next decision.

3.4 PHYSICAL-AWARE MULTI-FIDELITY EVALUATION

LAMBDA uses two evaluation fidelities to connect physical layout refinement with circuit behavior (Figure 4). Basic evaluation checks routing and analog constraints and provides low-cost physical metrics and parasitic estimates for frequent graph refinement. For eligible candidates selected by LLM, detailed evaluation performs DRC and LVS, followed by PEX and post-layout simulation once both checks pass. The resulting diagnostics and electrical responses guide subsequent physical edits, while paired parasitic estimates and PEX measurements calibrate the low-cost estimator. Final acceptance requires physical and functional verification, satisfaction of active analog constraints, and compliance with all fixed physical and electrical quality thresholds.

For each candidate graph G and synthesized layout L, basic evaluation measures the physical components of $\mathbf{f}(L; \mathcal{T})$, including area, wirelength, whitespace, and congestion. It checks routing completion and graph-encoded analog constraints $\mathbf{c}_{\text{ana}}(G, L) \preceq \mathbf{0}$, such as symmetry, and returns layout

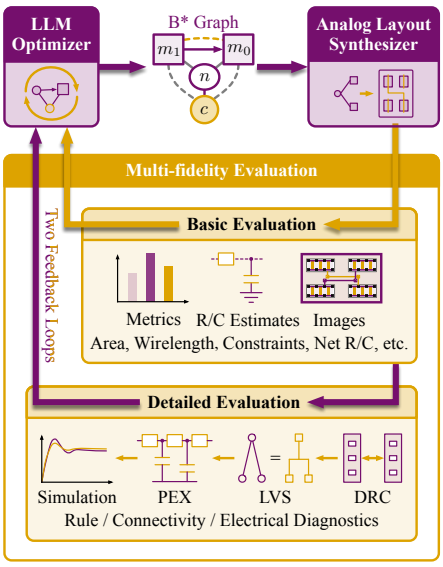


Figure 4: Physical-aware multi-fidelity evaluation.

images and estimates of critical-net resistance, capacitance, and load. These observations guide edits to G^{init}, which joint PnR realizes for reevaluation. A candidate becomes eligible for detailed evaluation only when routing is complete and its active analog constraints and basic physical limits in $\mathbf{q}$ are satisfied. LLM then chooses whether to continue refinement or obtain detailed feedback.

If detailed evaluation is selected, design-rule checking (DRC) verifies compliance with technology rules, and layout-versus-schematic (LVS) checking verifies device, parameter, and connectivity consistency with D. Only after both checks pass do we perform PEX and post-layout simulation under the fixed testbench $\mathcal{T}$.

The functional-test outcomes and electrical responses characterize circuit behavior, with the responses determining the electrical components of $\mathbf{f}(L;\mathcal{T})$ and $\mathbf{c}_{\text{q}}(L;\mathcal{T},\mathbf{q})$. Available verification diagnostics, extracted parasitics, and simulation outcomes return to the LLM as feedback. The graph's block, net, and space identities help relate this feedback to editable physical objects: verification failures guide corrective edits, while measured electrical responses and deviations from pre-layout references guide performance refinement. For example, excessive capacitive loading on a sensitive net can prompt coordinated changes to device positions, routing-space allocation, and wire geometry.

Detailed results also calibrate the low-cost parasitic estimator: discrepancies between basic estimates and PEX measurements from the same layout, net, quantity, and extraction boundary update its parameters for subsequent basic evaluations. The search returns the first candidate that passes DRC, LVS, and functional tests, completes PEX and post-layout simulation, satisfies its active analog constraints, and meets all physical and electrical thresholds in $\mathbf{q}$.

4 Experiments

We evaluate complete layout quality across nine routed and extracted circuits, the contributions of multi-fidelity feedback and joint placement and routing revisions to LLM-guided search, and the evolution of reusable resources over repeated design episodes. LAMBDA achieves the best automated geometric mean for full layout area and wirelength and the highest composite post-layout electrical score for analog circuits; all nine designs are DRC- and LVS-clean and pass their circuit-specific post-layout functional criteria. LAMBDA uses `gpt-5.6-sol` (OpenAI, 2026a) by default; the ablation study also evaluates `gpt-6-astra` (OpenAI, 2026b) as an alternative backbone.

4.1 Experimental Setup

We use nine industrial circuits in a commercial TSMC 28-nm process: six analog circuits (Mixer, LPF, Ring-4, IF amp, Peak detector, and Z amp) and three mixed-signal circuit modules (Div-2, TSPC-2, and TSPC-3), spanning oscillation, frequency division, amplification, filtering, mixing, sequential logic, and detection. All nine benchmark circuits are unseen design tasks for LAMBDA and are absent from its pre-existing layout library and other knowledge repositories. Sized netlists, technology, and circuit-specific testbenches are fixed across methods.

We compare against SA (Ou et al., 2013), CDLS (Mangalagiri et al., 2024), PANDA (Zhang et al., 2026), AlphaPlacer (Li et al., 2026), and PAPnR, which combines PAPlace (Xu et al., 2026) with PARoute2 (Xu et al., 2025). PANDA uses the same LLM backbone as LAMBDA. Baselines receive expert constraints where required, while LAMBDA generates and revises constraints autonomously. Appendix F.1 details the circuits and baseline configurations.

Table 2: Physical design results.

(a) Computational cost and success rate

Cost	SA		CDLS		PANDA			AlphaPlacer		PAPnR		**LAMBDA (Ours)**		
	Time	Pass@1	Time	Pass@1	Time	Tokens	Pass@1	Time	Pass@1	Time	Pass@1	Time	Tokens	Pass@1
AVG	[min/ckt] ↓	[%] ↑	[min/ckt] ↓	[%] ↑	[min/ckt] ↓	[k/round] ↓	[%] ↑	[min/ckt] ↓	[%] ↑	[min/ckt] ↓	[%] ↑	[min/ckt] ↓	[k/round] ↓	[%] ↑
	0.75	69	1.05	39	26.84	37.07	76	4.24	76	14.45	86	20.75	55.11	100

(b) Layout area [μm^2] ↓

Circuit	SA	CDLS	PANDA	AlphaPlacer	PAPnR	**LAMBDA (Ours)**
Mixer	24,692.5 (6.65×)	18,768.6 (5.05×)	8,272.5 (2.23×)	4,798.7 (1.29×)	4,737.0 (1.28×)	**3,715.1 (1.00×)**
LPF	9,489.4 (3.34×)	5,538.5 (1.95×)	8,596.5 (3.02×)	4,355.8 (1.53×)	6,532.9 (2.30×)	**2,842.7 (1.00×)**
Ring-4	21,942.1 (7.38×)	12,960.5 (4.36×)	7,285.6 (2.45×)	6,144.0 (2.07×)	3,383.0 (1.14×)	**2,972.4 (1.00×)**
IF amp	30,627.5 (3.85×)	15,858.4 (1.99×)	17,725.9 (2.23×)	9,307.9 (1.17×)	9,748.2 (1.23×)	**7,956.2 (1.00×)**
Peak det.	43,008.9 (4.00×)	17,063.9 (1.59×)	16,735.1 (1.56×)	14,516.3 (1.35×)	17,519.8 (1.63×)	**10,756.9 (1.00×)**
Z amp	2,003.0 (1.61×)	2,946.5 (2.37×)	2,070.6 (1.66×)	**1,187.7 (0.95×)**	1,895.0 (1.52×)	1,244.6 (1.00×)
Div-2	1,338.2 (8.66×)	1,167.3 (7.55×)	469.6 (3.04×)	**141.2 (0.91×)**	218.3 (1.41×)	154.6 (1.00×)
TSPC-2	603.3 (6.96×)	596.4 (6.88×)	403.7 (4.65×)	95.9 (1.11×)	138.3 (1.59×)	**86.7 (1.00×)**
TSPC-3	1,278.1 (8.37×)	1,284.8 (8.41×)	804.9 (5.27×)	174.5 (1.14×)	370.8 (2.43×)	**152.8 (1.00×)**
AVG	6,223.1 (5.03×)	4,621.0 (3.73×)	3,316.6 (2.68×)	1,539.9 (1.24×)	1,934.4 (1.56×)	**1,238.4 (1.00×)**

(c) Routed wirelength [μm] ↓

Circuit	SA	CDLS	PANDA	AlphaPlacer	PAPnR	**LAMBDA (Ours)**
Mixer	31,006.8 (14.78×)	20,920.3 (9.97×)	4,622.2 (2.20×)	2,639.6 (1.26×)	2,448.7 (1.17×)	**2,098.6 (1.00×)**
LPF	29,625.4 (13.54×)	2,420.0 (1.11×)	14,972.7 (6.84×)	2,425.7 (1.11×)	2,448.5 (1.12×)	**2,187.7 (1.00×)**
Ring-4	16,691.7 (10.21×)	10,695.4 (6.55×)	2,903.1 (1.78×)	1,729.6 (1.06×)	**1,487.0 (0.91×)**	1,634.1 (1.00×)
IF amp	23,959.0 (10.92×)	15,248.4 (6.95×)	4,202.7 (1.92×)	**2,178.6 (0.99×)**	2,856.2 (1.30×)	2,193.2 (1.00×)
Peak det.	29,797.3 (17.61×)	13,884.2 (8.20×)	5,485.0 (3.24×)	2,113.0 (1.25×)	2,349.3 (1.39×)	**1,692.3 (1.00×)**
Z amp	491.0 (1.92×)	1,796.5 (7.04×)	404.7 (1.59×)	285.5 (1.12×)	396.6 (1.55×)	**255.1 (1.00×)**
Div-2	3,331.2 (6.41×)	2,668.6 (5.13×)	1,236.5 (2.38×)	561.6 (1.08×)	584.1 (1.12×)	**519.9 (1.00×)**
TSPC-2	1,392.3 (5.12×)	1,296.5 (4.77×)	1,078.0 (3.97×)	303.0 (1.11×)	285.5 (1.05×)	**271.7 (1.00×)**
TSPC-3	4,168.4 (7.27×)	3,437.5 (6.00×)	2,858.8 (4.99×)	587.3 (1.02×)	793.4 (1.38×)	**573.4 (1.00×)**
AVG	7,775.7 (8.27×)	5,127.4 (5.45×)	2,680.4 (2.85×)	1,042.5 (1.11×)	1,135.5 (1.21×)	**940.4 (1.00×)**

The best values in each row are bold and gold; ↓/↑ indicate smaller/larger is better.

Layout quality is measured by full layout area, routed wirelength, and circuit-specific post-layout electrical responses. We report geometric means for area and wirelength and an equally weighted composite score over the six analog circuits (Analog AVG), together with runtime, token cost, and Pass@1.

For layout-quality comparisons, each method performs one continuous optimization task of at most six rounds per circuit; all reported quality metrics refer to the same selected layout. Pass@1 is measured separately as the fraction of independently initialized rounds that pass physical and post-layout functional checks. Appendix F.2 specifies candidate selection, passing criteria, and metric and cost aggregation.

4.2 COMPLETE PHYSICAL DESIGN RESULTS

Table 2(a) shows that LAMBDA achieves 100% Pass@1, compared with 39–86% for the baselines. SA, CDLS, AlphaPlacer, and PAPnR are faster but have lower Pass@1 values. Compared with PANDA, the other LLM-based method, LAMBDA takes 20.75 minutes per circuit on average, 22.7% less than PANDA's 26.84 minutes, while keeping token cost at 55.11k per round versus 37.07k. It thus combines higher reliability and shorter runtime with a higher token cost per round. Table 2(b)–(c) shows that LAMBDA achieves the lowest geometric means across the nine circuits for full layout area ($1{,}238.4\,\mu m^2$) and routed wirelength ($940.4\,\mu m$). The closest baseline, AlphaPlacer, uses $1.24\times$ the area and $1.11\times$ the wirelength. These ratios rise to $1.56\times$ and $1.21\times$ for PAPnR and $2.68\times$ and $2.85\times$ for PANDA. These results show that LAMBDA produces compact layouts with low routed wirelength.

Figure 5 shows five complete layouts generated by LAMBDA, spanning oscillation, amplification, mixing, detection, and filtering. In these layouts, symmetric device pairs, closely packed passive arrays, and signal-flow-oriented device groups preserve routing access between neighboring regions. Their compact, regular placement resembles that in expert-designed layouts. Appendix G provides the complete geometry measurements for each circuit.

Table 3: Post-layout electrical results.

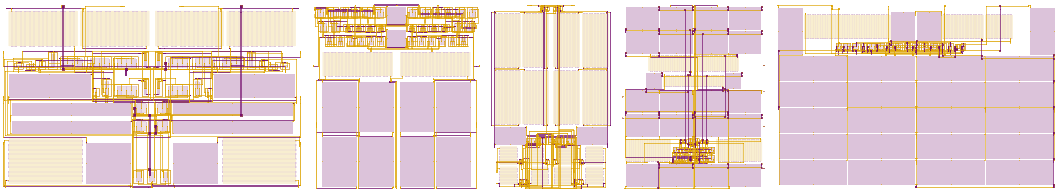
(a) Analog circuits

Circuit and response	SA	CDLS	PANDA	AlphaPlacer	PAPnR	LAMBDA (Ours)
Mixer mean IF p–p [mV] ↑	56.133 (0.82×)	53.366 (0.78×)	54.207 (0.80×)	43.043 (0.63×)	42.664 (0.63×)	**68.137 (1.00×)**
Mixer mismatch [%] ↓	0.693718 (418.41×)	5.831896 (3517.43×)	1.111221 (670.22×)	1.319683 (795.95×)	0.617138 (372.22×)	**0.001658 (1.00×)**
LPF mean OUT p–p [mV] ↑	5.151 (0.99×)	5.183 (1.00×)	4.968 (0.96×)	5.190 (1.00×)	5.175 (1.00×)	**5.198 (1.00×)**
LPF mismatch [%] ↓	0.141712 (17.62×)	0.032794 (4.08×)	0.275947 (34.31×)	0.030821 (3.83×)	0.027185 (3.38×)	**0.008042 (1.00×)**
Ring-4 freq. [MHz] ↑	68.937 (0.17×)	46.814 (0.12×)	161.302 (0.40×)	216.083 (0.54×)	338.719 (0.84×)	**400.896 (1.00×)**
Ring-4 mean OUT p–p [mV] ↑	154.637 (0.52×)	204.652 (0.69×)	65.362 (0.22×)	157.756 (0.53×)	242.889 (0.82×)	**297.424 (1.00×)**
Ring-4 mismatch [%] ↓	6.276574 (6.97×)	7.066549 (7.85×)	7.413437 (8.24×)	2.401709 (2.67×)	1.023517 (1.14×)	**0.899996 (1.00×)**
IF amp gain ↑	0.3123 (0.95×)	0.3098 (0.94×)	0.2612 (0.79×)	0.3268 (0.99×)	0.3265 (0.99×)	**0.3294 (1.00×)**
Peak det. OUT p–p [mV] ↑	2.585 (0.04×)	0.331 (0.00×)	3.607 (0.05×)	48.074 (0.65×)	58.779 (0.80×)	**73.484 (1.00×)**
Peak det. OUT@30 ns [V] ↑	0.963922 (0.94×)	0.961301 (0.94×)	0.956262 (0.93×)	1.010331 (0.99×)	1.015375 (0.99×)	**1.024231 (1.00×)**
Peak det. IQ [μA] ↓	32.773 (1.29×)	32.774 (1.29×)	25.363 (1.00×)	32.742 (1.29×)	32.776 (1.29×)	**25.310 (1.00×)**
Peak det. UGB [MHz] ↑	2.886 (0.46×)	2.928 (0.46×)	**12.951 (2.05×)**	2.782 (0.44×)	2.883 (0.46×)	6.312 (1.00×)
Peak det. PM [°] ↑	91.707 (0.95×)	91.571 (0.95×)	80.559 (0.83×)	95.301 (0.99×)	92.680 (0.96×)	**96.504 (1.00×)**
Z amp gain ↑	37.828 (0.99×)	37.858 (0.99×)	38.114 (1.00×)	37.834 (0.99×)	37.824 (0.99×)	**38.128 (1.00×)**
Analog AVG ↑	0.647	0.643	0.631	0.701	0.783	**0.983**

(b) Mixed-signal circuit modules

Circuit and response	SA	CDLS	PANDA	AlphaPlacer	PAPnR	LAMBDA (Ours)
Div-2 Q p–p [mV] ↑	798.977 (0.98×)	799.705 (0.98×)	803.202 (0.99×)	815.197 (1.00×)	814.246 (1.00×)	814.126 (1.00×)
Div-2 $\lvert r-2\rvert$ [10^{-6}] ↓	0	-	0	0	4	0
TSPC-2 Q p–p [mV] ↑	801.612 (0.98×)	801.228 (0.98×)	804.122 (0.99×)	815.533 (1.00×)	808.638 (0.99×)	814.402 (1.00×)
TSPC-2 QN p–p [mV] ↑	808.295 (0.98×)	803.521 (0.98×)	809.506 (0.99×)	822.605 (1.00×)	822.178 (1.00×)	821.512 (1.00×)
TSPC-2 mismatch [%] ↓	0.827 (0.96×)	0.285 (0.33×)	0.665 (0.77×)	0.860 (0.99×)	1.647 (1.90×)	0.865 (1.00×)
TSPC-2 $\lvert r-2\rvert$ [10^{-6}] ↓	0	5	10	1	9	0
TSPC-3 OUT p–p [mV] ↑	802.650 (0.99×)	801.365 (0.98×)	802.407 (0.99×)	814.767 (1.00×)	809.763 (0.99×)	814.212 (1.00×)
TSPC-3 $\lvert r-3\rvert$ [10^{-6}] ↓	0	1	2	0	0	0

The best values in each row are bold and gold; ↓/↑ indicate smaller/larger is better; – denotes a measurement that is not applicable or unavailable.



(a) Mixer. (b) LPF. (c) Ring-4. (d) IF amp. (e) Peak det.

Figure 5: Layouts generated by LAMBDA.

4.3 POST-LAYOUT ELECTRICAL PERFORMANCE

All nine LAMBDA layouts meet their circuit-specific functional criteria. LAMBDA achieves the highest Analog AVG (0.983), exceeding the five baselines by 25.5%–55.7% (Table 3(a)). It leads in mean output swing and normalized mismatch on Mixer and LPF, and gain on IF amp and Z amp. On Ring-4, it reaches 400.896 MHz (18.4% above PAPnR), with the highest mean swing (297.424 mV) and lowest normalized mismatch (0.900%), though PAPnR has lower absolute swing mismatch (2.499 versus 2.689 mV). On Peak det., its full-window OUT p–p (73.484 mV) exceeds PAPnR by 25.0%; this and its output at 30 ns are closest to schematic references (Appendix F.2). It also achieves the lowest IQ and a PM above its schematic reference, while PANDA leads in UGB. Mixed-signal output swings vary little across methods (Table 3(b)), while LAMBDA retains substantial physical-quality advantages (Table 2). Appendix Figure 15 gives the each circuit comparison, and Appendix I provides the full post-layout metric matrix.

Overall, LAMBDA outperforms all five baselines in geometric-mean area and wirelength and composite analog electrical score, while achieving 100% Pass@1. All nine final layouts meet their functional criteria. Despite some baseline advantages on individual metrics, these aggregate gains support joint optimization of constraints, placement, and routing with multi-fidelity feedback.

4.4 ABLATION STUDIES

Setting and interventions. We study evaluation feedback, joint PnR, and LLM decisions on LPF under the same circuit, testbenches, and experimental pass criteria. Following Section 3.4, *w.o. basic visual feedback* removes layout images; *w.o. basic metric evaluation feedback* withholds area, wirelength, whitespace, and routing-quality metrics, together with Dconstraint-check feedback; and *w.o. detailed feedback* withholds PEX results, simulation responses, and electrical retention from

Table 4: Ablation study results.

Variant	Time [min/round] ↓	Tokens [k/round] ↓	Pass@1 [%] ↑	Layout area [μm^2] ↓	Wirelength [μm] ↓	mean OUT p–p [mV] ↑	mismatch [%] ↓	abs. mismatch [μV] ↓
LAMBDA (Sol)	9.99 (1.00×)	55.11 (1.00×)	**100**	**2842.7 (1.00×)**	2187.7 (1.00×)	5.198 (1.00×)	**0.008 (1.00×)**	**0.418 (1.00×)**
w.o. basic visual feedback	10.25 (1.03×)	**47.63 (0.86×)**	43	4075.0 (1.43×)	2271.8 (1.04×)	5.181 (1.00×)	0.009 (1.06×)	0.441 (1.06×)
w.o. metric eval. feedback	5.25 (0.53×)	61.79 (1.12×)	0	-	-	-	-	-
w.o. detailed feedback	16.61 (1.66×)	49.46 (0.90×)	71	2968.3 (1.04×)	2083.7 (0.95×)	5.200 (1.00×)	0.062 (7.65×)	3.199 (7.65×)
w.o. joint PnR	10.64 (1.07×)	117.33 (2.13×)	57	3091.6 (1.09×)	2124.5 (0.97×)	5.197 (1.00×)	0.015 (1.86×)	0.778 (1.86×)
w.o. LLM	**5.85 (0.59×)**	-	33	5999.3 (2.11×)	2754.3 (1.26×)	5.175 (1.00×)	0.018 (2.26×)	0.942 (2.25×)
LAMBDA (Astra)	36.74 (3.68×)	144.37 (2.62×)	**100**	2960.0 (1.04×)	**1983.5 (0.91×)**	**5.211 (1.00×)**	0.010 (1.20×)	0.501 (1.20×)
Tools-only (Astra)	53.20 (5.33×)	1938.76 (35.18×)	51	3843.0 (1.35×)	2341.8 (1.07×)	4.891 (0.94×)	0.031 (3.81×)	1.499 (3.59×)

The best values in each column are bold and gold; ↓/↑ indicate smaller/larger is better; – denotes a measurement that is not applicable or unavailable.

B* Graph editing. The *w.o. joint PnR* condition fixes placement before routing and prevents routing feedback from revising it. The *w.o. LLM* condition replaces the LLM with simulated annealing to perturb the B* Graph. Table 4 reports physical and post-layout quality, Pass@1, and mean tokens and time per executed round, including failures and retries. Parenthesized ratios use LAMBDA (Sol) as the reference. Failed conditions report their last available geometry and are excluded from best-value comparisons. Appendices F.3 and J provide protocol details.

Backbone configurations. LLM-based component ablations use `gpt-5.6-sol` (OpenAI, 2026a) at `max` reasoning effort, referencing full LAMBDA (Sol). LAMBDA (Astra) uses `gpt-6-astra` (OpenAI, 2026b) at `ultra` reasoning effort with identical initial code, skills, layout library, and tools. Tools-only (Astra) retains the Astra configuration, circuit, PDK, testbenches, and basic design and verification tools, but omits the LAMBDA framework.

Results and analysis. Relative to LAMBDA (Sol), *w.o. basic visual feedback* reduces reliability and compactness, while *w.o. metric eval. feedback* fails to complete routing, highlighting their complementary roles in geometric refinement and routability. The *w.o. detailed feedback* condition yields $7.65\times$ the absolute swing mismatch despite similar mean swing and shorter wirelength, demonstrating the importance of detailed electrical feedback for matching quality. The *w.o. joint PnR* condition reduces reliability and matching quality while increasing token usage. Although *w.o. LLM* runs faster without LLM calls, it produces less reliable designs with larger area, longer wirelength, and greater mismatch despite measured electrical feedback and expert-provided constraints. These comparisons support joint PnR and LLM-guided search for coordinating physical and electrical objectives. Comparing backbones, LAMBDA (Sol) and LAMBDA (Astra) both achieve 100% Pass@1 with comparable layout quality, but Sol requires substantially fewer tokens and less runtime, suggesting limited benefit from the stronger backbone in this setting. Tools-only (Astra) yields lower reliability, poorer compactness and matching, and lower mean swing at higher computational cost than LAMBDA (Astra). This same-backbone comparison demonstrates that the framework improves both design quality and computational efficiency beyond providing tool access alone (Appendix J.2).

4.5 TOWARDS EXPERT-LEVEL ANALOG PHYSICAL DESIGN

Comparisons with expert layouts show progress towards expert-level design (Appendix A). Evolution of code, skills, and experience further improves layout quality (Appendix K).

5 CONCLUSION

We presented LAMBDA, an LLM framework for analog physical design. B* Graph connects circuit-derived constraints, placement, and routing in same search space for joint PnR. LLM acts as a history-dependent optimizer, combining circuit knowledge with accumulated feedback to coordinate graph edits and evaluation. Basic evaluation supports frequent placement and routing exploration and low-cost parasitic estimation; LLM decides whether an eligible current candidate proceeds to detailed verification, extraction, and post-layout simulation. Electrical responses and aligned parasitic discrepancies guide further graph edits and calibration until a verified layout meets all fixed quality thresholds. Across nine commercial 28-nm circuits, every LAMBDA layout is completely routed, DRC- and LVS-clean, extracted, and functionally verified. LAMBDA achieves the best automated geometric mean for full layout area and wirelength, as well as the highest composite post-layout electrical score for analog circuits.

AI USE STATEMENT

The authors led the research and verified all AI-assisted work, including manuscript text, figures, code, mathematical arguments, and experimental analyses. Generative AI tools supported method formalization, mathematical analysis, implementation and data processing, interpretation of results, manuscript drafting, polishing and translation, and figure and table preparation. The authors take full responsibility for the scientific claims, reported results, and final manuscript. The use of LLMs as components of LAMBDA and the comparison systems is documented in Sections 3 and 4 and Appendix J.2.

REPRODUCIBILITY STATEMENT

Section 3 and Appendices B–D specify the graph representation, optimization procedure, theoretical assumptions, and proofs. Appendix F documents the circuit suite, toolchain, fixed testbenches, functional criteria, metric definitions, and run accounting. Model configurations and ablation protocols appear in Section 4.4 and Appendix J, with the evolution protocol in Appendix K. Complete physical and electrical measurements are reported in Appendices G and I.

ETHICS STATEMENT

This work develops an LLM-based framework for analog physical design automation. It does not involve human subjects, personal data, or sensitive user information.

REFERENCES

Yun-Chih Chang, Yao-Wen Chang, Guang-Ming Wu, and Shu-Wei Wu. B*-trees: A new representation for non-slicing floorplans. In *ACM/IEEE Design Automation Conference (DAC)*, pp. 458–463, 2000.

Hao Chen, Mingjie Liu, Xiyuan Tang, Keren Zhu, Abhishek Mukherjee, Nan Sun, and David Z. Pan. MAGICAL 1.0: An open-source fully-automated ams layout synthesis framework verified with a 40-nm 1GS/s $\Delta\sum$ ADC. In *IEEE Custom Integrated Circuits Conference (CICC)*, pp. 1–2, 2021.

Georges Gielen. Analog designers, synthesis or generative AI? Keynote presentation at SMACD 2024, July 2024. URL `https://smacd2024.e-ce.uth.gr/Keynote_Gielen_SMACD2024_distribution.pdf`. July 4, 2024, slide 20. Illustrative design-risk estimate attributed to Mentor Graphics.

Shengjie Gong, Wenjie Peng, Hongyuan Chen, Gangyu Zhang, Yunqing Hu, Huiyuan Zhang, Shuangping Huang, and Tianshui Chen. Learning hierarchical and geometry-aware graph representations for Text-to-CAD. In *The Fourteenth International Conference on Learning Representations*, 2026.

Kirthevasan Kandasamy, Gautam Dasarathy, Jeff Schneider, and Barnabás Póczos. Multi-fidelity Bayesian optimisation with continuous approximations. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1799–1808. PMLR, 2017.

Kishor Kunal, Meghna Madhusudan, Arvind K. Sharma, Wenbin Xu, Steven M. Burns, Ramesh Harjani, Jiang Hu, Desmond A. Kirkpatrick, and Sachin S. Sapatnekar. ALIGN: Open-source analog layout automation from the ground up. In *Proceedings of the Design Automation Conference*, 2019. doi: 10.1145/3316781.3323471.

Rui Li and Shuang Cao. From trajectories to graphs: Contract-checked editing for verifier-guided LLM reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 43259–43306. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.2004.

Yapeng Li, Peng Xu, Mingzi Wang, and Tinghuan Chen. AlphaPlacer: Analog placement enhanced by Monte Carlo tree search. In *Proceedings of the Design Automation Conference*, 2026. doi: 10.1145/3770743.3804024.

Bingyang Liu, Haoyi Zhang, Xiaohan Gao, Zichen Kong, Xiyuan Tang, Yibo Lin, Runsheng Wang, and Ru Huang. LayoutCopilot: An LLM-powered multiagent collaborative framework for interactive analog layout design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2025. doi: 10.1109/TCAD.2025.3529805.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Enrico Malavasi and Alberto Sangiovanni-Vincentelli. Area routing for analog layout. *IEEE transactions on computer-aided design of integrated circuits and systems*, 12(8):1186–1197, 1993.

Prasanth Mangalagiri, Praveen Mosalikanti, Farrukh Zafar, Lynn Qian, Phoebe Chang, Arun J. Kurian, and Vinay Saripalli. CDLS: Constraint driven generative AI framework for analog layout synthesis. In *Proceedings of the Design Automation Conference*, 2024. doi: 10.1145/3649329.3656232.

Asal Mehradfar, Xuzhe Zhao, Yilun Huang, Emir Ceyani, Yankai Yang, Shihao Han, Hamidreza Aghasi, and Salman Avestimehr. FALCON: An ML framework for fully automated layout-constrained analog circuit design. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

OpenAI. GPT-4o System Card. `https://cdn.openai.com/gpt-4o-system-card.pdf`, August 2024.

OpenAI. Gpt-5.6 system card. `https://deploymentsafety.openai.com/gpt-5-6/gpt-5-6.pdf`, jul 2026a.

OpenAI. GPT-6 Astra System Card. `https://deploymentsafety.openai.com/gpt-6-astra/gpt-6-astra.pdf`, September 2026b.

Hung-Chih Ou, Hsing-Chih Chang Chien, and Yao-Wen Chang. Simultaneous analog placement and routing with current flow and current density considerations. In *ACM/IEEE Design Automation Conference (DAC)*, pp. 1–6, 2013.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Aditya Shahane, Saripilli Swapna Manjiri, Ankesh Jain, and Sandeep Kumar. GCX: Graph of circuits with GNN for exploring the optimal design space. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Shion Takeno, Hitoshi Fukuoka, Yuhki Tsukada, Toshiyuki Koyama, Motoki Shiga, Ichiro Takeuchi, and Masayuki Karasuyama. Multi-fidelity Bayesian optimization with max-value entropy search and its parallelization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9334–9345. PMLR, 2020.

Peng Xu, Jindong Tu, Guojin Chen, Keren Zhu, Tinghuan Chen, Tsung-Yi Ho, and Bei Yu. Paroute2: Enhanced analog routing via performance-drive guidance generation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 44(10):3654–3667, 2025. doi: 10.1109/TCAD.2025.3550445.

Peng Xu, Yuan Pu, Keren Zhu, Tinghuan Chen, Tsung-Yi Ho, and Bei Yu. PAPlace: Performance-driven differentiable analog placement. *ACM Transactions on Design Automation of Electronic Systems*, 2026. doi: 10.1145/3799979.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *The Twelfth International Conference on Learning Representations*, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023.

Haoyi Zhang, Weijian Fan, Xiaohan Gao, Bingyang Liu, Runsheng Wang, and Yibo Lin. PANDA: An LLM-enhanced performance-driven analog design framework bridging design intent and layout generation. In *Proceedings of the Design Automation Conference*, 2026.

Shuyuan Zhang, ChenHan Jiang, Zuoou Li, and Jiankang Deng. ShapeCraft: LLM agents for structured, textured and interactive 3D modeling. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Kaizhi Zheng, Xiaotong Chen, Xuehai He, Jing Gu, Linjie Li, Zhengyuan Yang, Kevin Lin, Jianfeng Wang, Lijuan Wang, and Xin Wang. EditRoom: LLM-parameterized graph diffusion for composable 3D room layout editing. In *The Thirteenth International Conference on Learning Representations*, 2025.

A TOWARDS EXPERT-LEVEL ANALOG PHYSICAL DESIGN

We compare the layouts generated by LAMBDA with expert-designed layouts of the analog circuits. Expert and LAMBDA layouts are evaluated using the same circuit-specific testbench $\mathcal{T}$. The comparison covers the electrical metrics in Table 3(a) and full layout area. Matching expert performance is judged by each metric's direction and is distinct from passing the circuit-specific functional tests.

A.1 SCHEMATIC, EXPERT POST-LAYOUT, AND LAMBDA RESPONSES

Table 5 reports schematic and post-layout responses. We set the electrical quality thresholds in $\mathbf{q}$ to allow at most 3% degradation relative to expert performance: a metric with $\uparrow$ must reach at least 0.97 times the expert value, and a metric with $\downarrow$ must remain at or below 1.03 times that value.

Table 5: Electrical results relative to expert layouts.

Circuit and response	Schematic	Expert	LAMBDA (Ours)	Change vs. expert
Mixer mean IF p–p [mV] ↑	127.321	43.6651	68.1367	**Improvement 56.04%**
Mixer mismatch [%] ↓	0.187225	0.455481	0.001658	**Improvement 99.64%**
LPF mean OUT p–p [mV] ↑	9.76305	5.222789	5.197528	**Degradation 0.48%**
LPF mismatch [%] ↓	0.027651	0.498373	0.008042	**Improvement 98.39%**
Ring-4 freq. [MHz] ↑	410.024	302.3638	400.8964	**Improvement 32.59%**
Ring-4 mean OUT p–p [mV] ↑	727.0551	38.4955	297.4236	**Improvement 672.62%**
Ring-4 mismatch [%] ↓	0.666639	1.603082	0.899996	**Improvement 43.86%**
IF amp gain ↑	1.577872	0.333627	0.329376	**Degradation 1.27%**
Peak det. OUT p–p [mV] ↑	114.8054	94.11569	73.48419	Degradation 21.92%
Peak det. OUT@30 ns [V] ↑	1.074614	1.052426	1.024231	**Degradation 2.68%**
Peak det. IQ [μA] ↓	31.72477	33.05798	25.31023	**Improvement 23.44%**
Peak det. UGB [MHz] ↑	6.578116	2.986106	6.311568	**Improvement 111.36%**
Peak det. PM [$^\circ$] ↑	91.83218	92.33874	96.50421	**Improvement 4.51%**
Z amp gain ↑	39.03431	38.07595	38.12842	**Improvement 0.14%**

Note. Bold marks changes within the electrical quality thresholds $\mathbf{q}$; gold marks improvements over the expert layouts; ↓/↑ indicate smaller/larger is better.

Of the 14 electrical metrics, LAMBDA matches or exceeds expert performance on ten; three are only slightly below the expert result(within 3%). Overall, 13 of the 14 metrics meet the expert-relative electrical thresholds $\mathbf{q}$, indicating post-layout electrical quality competitive with the expert-designed layouts. Peak det. passes its functional test, but its OUT p–p is 21.92% below the expert value and misses the 3% quality tolerance.

A.2 FULL LAYOUT AREA

For area, we set the quality criteria in $\mathbf{q}$ to a target of $1.25\times$ the expert layout area and an acceptable upper limit of $1.35\times$. Table 6 shows that all six layouts remain within the acceptable limit, five meet the target, and two are smaller than the expert layouts. Mixer and Ring-4 reduce area by 4.27% and 26.28%, respectively. IF amp uses $1.312\times$ the expert area, above the target but within the acceptable limit.

Table 6: Layout area relative to expert layouts.

Circuit	Expert [$\mu\mathrm{m}^2$]	LAMBDA [$\mu\mathrm{m}^2$]	LAMBDA/Expert	Assessment
Mixer	3,880.77	3,715.10	$0.957\times$	**Smaller than expert**
LPF	2,485.10	2,842.70	$1.144\times$	**Target met**
Ring-4	4,031.93	2,972.40	$0.737\times$	**Smaller than expert**
IF amp	6,063.07	7,956.20	$1.312\times$	Acceptable
Peak det.	8,969.80	10,756.90	$1.199\times$	**Target met**
Z amp	1,028.12	1,244.60	$1.211\times$	**Target met**

Note. Bold marks changes within the electrical physical thresholds $\mathbf{q}$; gold marks improvements over the expert layouts.

Figure 6 compares five expert layouts with the corresponding LAMBDA layouts, with displayed areas proportional to the full layout areas in Table 6. The expert layouts embody substantial manual design and refinement. Although LAMBDA still trails these expert references on some electrical responses and produces larger layouts for several circuits, most post-layout electrical metrics are better, most layouts meet the area target, and all remain within the acceptable area limit. Mixer and Ring-4 achieve smaller areas while improving even every reported electrical metric relative to their expert

layouts. These results provide concrete evidence of progress towards expert-level analog physical design.

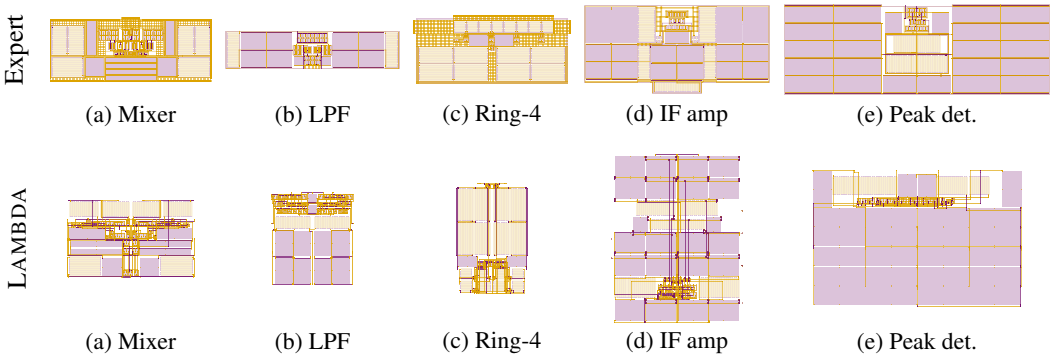


Figure 6: Layouts designed by expert and layouts generated by LAMBDA, scaled in proportion to full layout area.

B B* GRAPH DETAILS

LLM initializes or perturbs an init B* Graph that expresses analog constraints, placement, and routing together. Fixed postprocessing deterministically absorbs space into existing directional space nodes and removes redundant nodes before the graph enters joint PnR. This appendix defines the editable objects, the fusion rule, and the resulting rectangular space partition.

B.1 INIT B* GRAPH AND JOINT EDITING

The init graph G^{init} contains blocks, nets, and four space symbols for every device m_i: $c_i^{\uparrow}$, $c_i^{\downarrow}$, $c_i^{\leftarrow}$, and $c_i^{\rightarrow}$. The symbols are part of the device's representation and refer to space accessed from its four sides. Their geometric regions are determined by postprocessing. A block is a device or symmetry group; devices inside a group retain their individual side references. Table 7 summarizes the objects and their associations.

Table 7: B* Graph objects and attributes.

Object	Meaning	Principal attributes
Block b	Device or symmetry group	Device identities, local tree, mirror pairs, orientations, gaps, and dimensions
Net n	Pins and routing requirements	Pin identities, layers, and wire widths
Initial space symbol	Reference from one device side	Device, side, layer references, and routing allocations
Space node c	Free rectangle after fusion	Layer, rectangular bounds, boundary passages, and routing allocations
Tree edge	Block $\rightarrow$ block: relative placement	Left or right child
Pin edge	Block–net: pin membership	Device, pin, and circuit role
Boundary edge	Block–space: side association	Device side, contact interval, and local minimum-spacing requirements
Adjacency edge	Space–space: routing passage	Shared boundary or legal via connection
Allocation edge	Net–space: space allocation	Allocated spaces and conductor footprints
Constraint edge	Block–block or net–net: analog constraint	Symmetry, signal-flow ordering constraints, paired pins, matching properties, and tolerances

LLM infers analog constraints and directly initializes or perturbs the complete init graph, jointly choosing device placement, space reservations, net–space allocations, routing layers, and wire

widths while preserving the circuit's pin connectivity. Sharing a space does not connect different nets. Pin edges retain individual pin identities even when several pins belong to the same block and net. Conductor footprints are recorded from the realized layout. The inferred constraints guide these choices and are encoded as pairings, attributes, and constraint edges.

Fixed postprocessing takes this complete graph, determines rectangular space ownership among existing nodes, removes redundant nodes, and checks the resulting associations. Invalid tree relations, inconsistent constraints, unavailable passages, or inconsistent net–space associations return diagnostics to the LLM. The resulting G is the input to joint PnR. Candidate records retain G^{init}, G, their reference mapping, and the realized layout L, so evaluation results can guide further edits to the init graph.

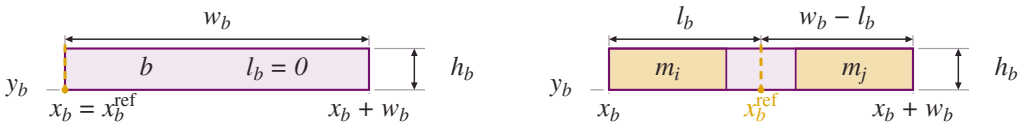
B.2 Device Geometry

The tree supplies relative placement rules within joint PnR. A device or symmetry group is treated as a rectangular block whose position depends on both tree relations and routing-space requirements. In Figure 7, y_b is the rectangle's bottom coordinate, h_b its height, and x_b^{ref} its horizontal reference. An ordinary block uses its left boundary as the reference; a symmetry group uses its symmetry axis. A block consisting of one self-symmetric device, such as b_0, is a single-device symmetry group and uses the device's symmetry axis as its reference.

Let l_b be the distance from the reference line to the left boundary and w_b the block width. The left boundary is

$$x_b = x_b^{\text{ref}} - l_b. \tag{2}$$

The occupied horizontal interval is $[x_b, x_b + w_b]$, and the distance from the reference to the right boundary is $w_b - l_b$. For an ordinary block, $l_b = 0$, so $x_b^{\text{ref}} = x_b$. For symmetry groups, equating their reference coordinates aligns their symmetry axes even when their widths differ.



(a) Ordinary block: left-edge reference. (b) Symmetry group: axis reference.

Figure 7: Block geometry and reference coordinates. The purple rectangle bounds block b; the gold line marks its horizontal reference x_b^{ref}. The left offset l_b and width w_b locate its two boundaries.

The labels left and right refer to branches of the tree: a left child is placed to the right of its parent, and a right child is placed above it. Let g_u^x and g_u^y be the required gaps to the right of and above parent u, computed from channel reservations, routing demand, and technology spacing. The child's horizontal reference is

$$x_v^{\text{ref}} = \begin{cases} x_u + w_u + g_u^x + l_v, & \text{left child,} \\ x_u^{\text{ref}}, & \text{right child.} \end{cases} \tag{3}$$

For a left child, the first rule places its left boundary g_u^x beyond the parent's right boundary. For a right child, the reference coordinates coincide, and $y_v \geq y_u + h_u + g_u^y$ leaves the required space above the parent. During joint PnR, the packing contour accounts for placed blocks and required gaps. The synthesizer adjusts device positions to routing demand and updates the space partition, net–space allocations, and wires in the same realization.

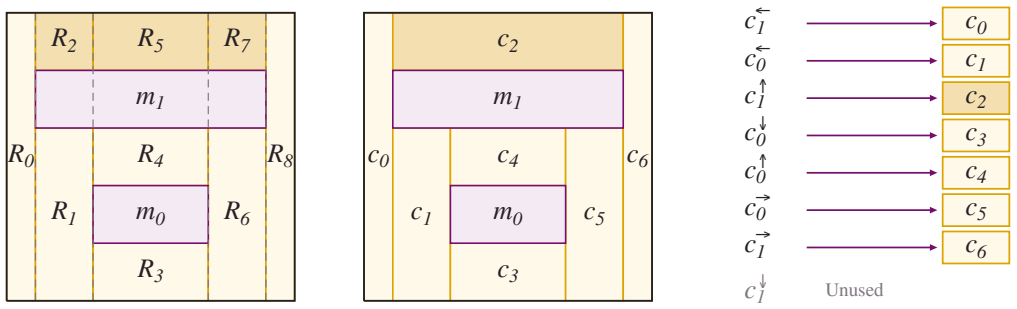
B.3 Space Fusion

The construction uses the evaluated geometry encoded in the init graph. Consider a routing layer with a rectangular outline and $N \geq 1$ axis-aligned rectangular device obstacles, with positive clearances between devices and from the outline. A symmetry group's bounding box does not obstruct its internal free space. Let F_k denote the free space on layer k. Device identifiers and the side order left, right, bottom, top provide a fixed total order for the existing directional nodes. All guarantees below concern this fixed geometry and ordering.

Internal geometry. The x-coordinates of all vertical sides of the outline and devices define full-height vertical strips. Within each vertical strip, subtract the blocked vertical intervals; each remaining maximal interval defines an internal free rectangle R_j. Horizontally adjacent rectangles with identical lower and upper boundaries are coalesced into their maximal continuous runs. These runs uniquely define the final rectangular partition; they are geometric records and do not create graph nodes. Shared free boundaries use a fixed half-open ownership convention, while device boundaries remain obstacles. Corner contact alone does not provide a passage.

Canonical ownership. Each initial side symbol refers to all R_j sharing a positive-length boundary with that device side. This contact region may be nonrectangular. Construct a bipartite graph between the coalesced rectangles and the existing side symbols, connecting a rectangle to each symbol whose device side it touches over a positive length. A complete injective matching assigns each rectangle to a distinct existing symbol. Sort rectangles by their left, bottom, right, and top coordinates, and choose the lexicographically smallest complete assignment under the fixed symbol order. This choice considers complete matching feasibility rather than independently choosing a recipient for each rectangle. The next subsection establishes that such a matching exists.

Figure 8 follows these steps for two devices: nine internal rectangles become seven final rectangles, each assigned to a different existing side symbol.



(a) Vertical strips. (b) Coalesced rectangles. (c) Existing-node ownership.

Figure 8: From geometric rectangles to existing nodes. (a) Device vertical boundaries define nine internal free rectangles. (b) The highlighted equal-height rectangles R_2, R_5, R_7 coalesce into c_2. (c) Canonical matching assigns each final rectangle to a distinct existing side symbol; $c_1^{\downarrow}$ is unused.

The selected symbols retain their node identities. A retained node can relinquish parts of its original contact region and absorb other parts; unused symbols are removed. Freeze the entire assignment before updating geometry, and write $\mathrm{owner}(j)$ for the selected existing node that receives R_j. Its final region is

$$S_c = \bigcup_{j:\,\mathrm{owner}(j)=c} R_j. \tag{4}$$

All members of this union belong to one coalesced run, so S_c is a rectangle. Geometry and association updates are committed together after validation; intermediate geometric fragments never acquire node identities. Rectangularity applies to the committed nodes. The mapping from an original side symbol records all final nodes receiving its contact region and may therefore contain more than one node.

Association updates. Fusion updates boundary and allocation edges and derives same-layer adjacency from positive-length shared free boundaries. The original-symbol reference mapping transfers net–space associations to the corresponding final rectangles. Device and pin identities, net identities, and local resource requirements are retained. Duplicate geometric references are removed without duplicating resource demands or merging independent electrical connections. Missing associations, conflicting attributes, or unavailable passages produce diagnostics.

Require: complete init graph G^{init} with its encoded layout information

1: evaluate the geometry and form internal free rectangles R_j on each layer

2: coalesce horizontally adjacent rectangles with identical vertical bounds into maximal runs
3: form side-contact associations and find the lexicographically smallest complete injective matching
4: freeze ownership among existing nodes; prepare geometry and reference updates
5: validate coverage, matching, net–space associations, and encoded constraints
6: commit all updates and remove unselected nodes if checks pass; otherwise return diagnostics
7: return the fused graph G and the original-symbol reference mapping

B.4 Spatial Completeness

Under the rectangular-device assumptions above, deterministic space fusion yields a unique complete partition into free rectangles, assigns every rectangle to a distinct existing node, and never increases the node count.

Coverage and unique ownership. No device vertical boundary lies inside a vertical strip, so its blocked vertical intervals are constant across the vertical strip. Their complement uniquely defines the internal rectangles R_j. Every free point belongs to one such rectangle under the boundary convention. Coalescing equal vertical bounds preserves coverage and disjoint interiors, and each maximal horizontal run remains a rectangle. A complete matching gives every run one owner. Consequently,

$$F_k = \biguplus_{c \text{ on layer } k} S_c, \tag{5}$$

where $\biguplus$ denotes disjoint union under the fixed boundary convention.

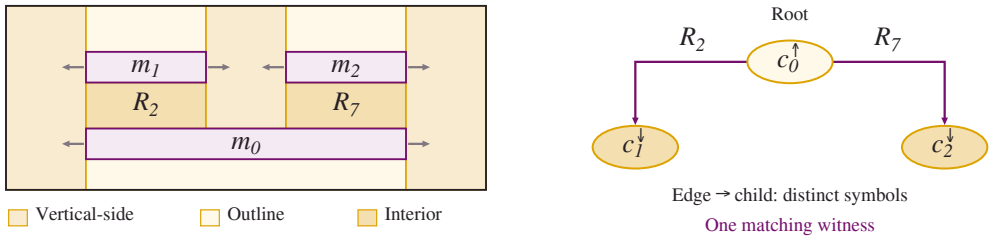
Existence of a complete matching. Each device's vertical side touches exactly one R_j and hence one coalesced rectangle. Indeed, a device obstructing part of that side in the immediately adjacent vertical strip would either introduce a vertical boundary inside that vertical strip or violate positive clearance. Assign every rectangle touching a device vertical side to one such side symbol. These assignments cannot conflict because each vertical side touches only one rectangle.

Consider a remaining rectangle, which touches no device vertical side. Its upper and lower boundaries are supported by fixed device horizontal sides or the outline: switching between different devices at the same height would require the devices to meet without positive clearance. The rectangle spans the full overlap of the supporting devices' horizontal projections, or the entire projection of its sole supporting device when the other boundary is the outline. Otherwise an intervening obstacle would terminate the rectangle within that overlap and contribute a vertical-side contact, a contradiction. A full-height rectangle necessarily touches a device vertical side because $N \geq 1$, so it was already assigned. A remaining rectangle adjacent to the outline's top or bottom can be assigned to its sole device side; it occupies that side's full projection, leaving no competing rectangle incident to the same symbol.

The remaining interior rectangles define edges between their lower device's top symbol and upper device's bottom symbol. This graph is a forest. To see this, choose in any subgraph a symbol whose device has the smallest right coordinate. If it had two neighbors, all three device projections would contain a common horizontal coordinate immediately left of that right coordinate. Both neighbors are above a top-side vertex or below a bottom-side vertex, so the nearer neighbor would obstruct visibility to the farther one, contradicting the full-overlap property. Every subgraph therefore has a vertex of degree at most one; parallel edges are also excluded by the unique projection overlap. Root each tree and assign every edge rectangle to its child vertex. These assignments use distinct horizontal symbols and do not conflict with the earlier assignments. A complete injective matching thus exists.

Figure 9 illustrates this existence argument. Once rectangles touching vertical device sides or supported by the outline have owners, the two remaining rectangles form a tree on horizontal side symbols and can be assigned to distinct child vertices.

Determinism and node count. The evaluated geometry uniquely determines the vertical strips, maximal runs with identical vertical bounds, and side-contact graph. The finite nonempty set of



(a) Rectangle contact classes.

(b) Matching the remaining rectangles.

Figure 9: A complete matching for the existence proof. (a) Short arrows mark vertical-side contacts. Three rectangles have such contacts, three remaining rectangles adjoin the outline, and two lie between horizontal device sides. (b) The last two rectangles, R_2 and R_7, form edges in a tree rooted at $c_0^{\uparrow}$. Assigning each edge to its child gives distinct owners $c_1^{\downarrow}$ and $c_2^{\downarrow}$. Appendix B.3 selects the canonical assignment among complete matchings.

complete matchings has one lexicographically smallest element, so ownership and retained identities are unique. Updates use this frozen assignment; changing the order of processing internal rectangles cannot change the committed geometry. Adjacency, attribute aggregation, and output ordering follow fixed rules as well. This is a guarantee for space fusion at fixed geometry, not for unresolved coordinate choices or the subsequent PnR outcome. Every final rectangle uses a different one of the $4N$ existing side identities. Geometric calculations create no nodes, and the commit only updates retained nodes and deletes the rest, so the node count never increases and is at most $4N$ per layer.

Routing connectivity. Each final space node is a free rectangle. Joint PnR uses the allocated spaces, legal adjacency, and routing constraints to connect each net's pins. It determines the traversal order, crossing locations, wire geometry, and vias. Finite wire widths, spacing, via enclosures, and competing nets still require physical checks; rectangular coverage alone does not establish physical routability.

B.5 Layers, Local Resources, and Feedback

Layer references are attributes of the four side symbols. Each layer is processed with its own obstacles and free rectangles, and fusion acts on existing references within that layer. A final node is identified together with its layer; legal vias connect regions across layers. The matching proof and node-count bound apply separately to layers satisfying the rectangular-device assumptions above.

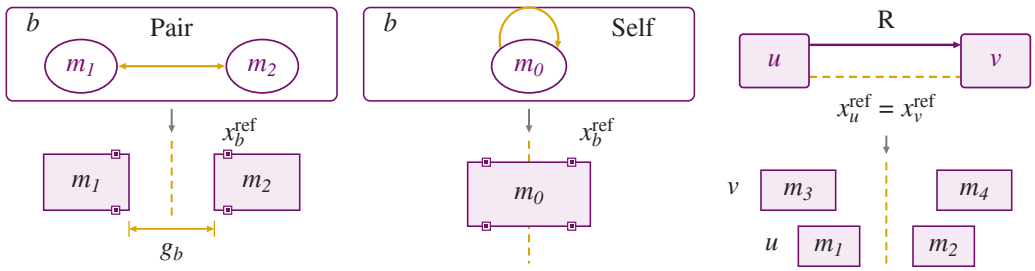
Additional fixed obstacles, devices that do not obstruct a layer, and nonrectangular outlines require explicit validation of the side associations, rectangular coverage, and a complete matching to existing layer references; they are not covered automatically by the existence proof. Missing associations or an incomplete matching return diagnostics without creating nodes or accepting nonrectangular regions. A via requires a legal location and metal enclosure on both layers, and its allocation includes every rectangle touched by its footprint.

Routing capacity is evaluated at the actual passages and device boundaries. For a passage in space c between lower block u and upper block v, the local vertical gap must satisfy

$$y_v - y_u - h_u \geq \max\{g_c, s_{uv}, d_c\}, \tag{6}$$

where y_u, y_v are the blocks' bottom coordinates and h_u is the lower block's height. The minimum-spacing requirement g_c, technology spacing s_{uv}, and routing demand d_c from wires, pin access, and vias apply to the specified facing sides. Horizontal gap requirements use the corresponding left and right sides. If several independent demands share a passage, the physical checks consider them together.

Joint PnR realizes device placement and routing under these requirements. When geometry or the outline changes, the fixed rule recomputes the affected layer's partition, side contacts, canonical ownership, and net–space allocations. Pin connectivity and active analog constraints are checked



(a) Device pair symmetry. (b) Self-symmetry. (c) Common symmetry.

Figure 10: Device constraints in the graph (top) and layout (bottom). Pair symmetry mirrors two devices, self-symmetry centers one device on the axis, and a common symmetry constraint aligns multiple groups.

on the resulting layout. The original-symbol reference mapping remains tied to device identities, allowing feedback on a final rectangle to identify the relevant objects in G^{init}. The LLM then perturbs that init graph for the next candidate.

B.6 Analog Constraints

The analog constraints in Table 1(b) specify requirements on device placement and routing. Each candidate's realized layout must satisfy its active analog constraints, some of which may be enforced directly by that candidate's tree relations. Figures 10 and 11 connect each graph-encoded analog constraint to the property checked on the realized layout.

Device pair symmetry. Pair symmetry places two devices as mirror images about a common symmetry axis. The graph records their pairing $m_i \leftrightarrow m_j$ within block b. Let S_i be the placed shape of device m_i, including its labeled pins. Their geometry must satisfy

$$S_j = \text{mirror}_{x_b^{\text{ref}}}(S_i), \tag{7}$$

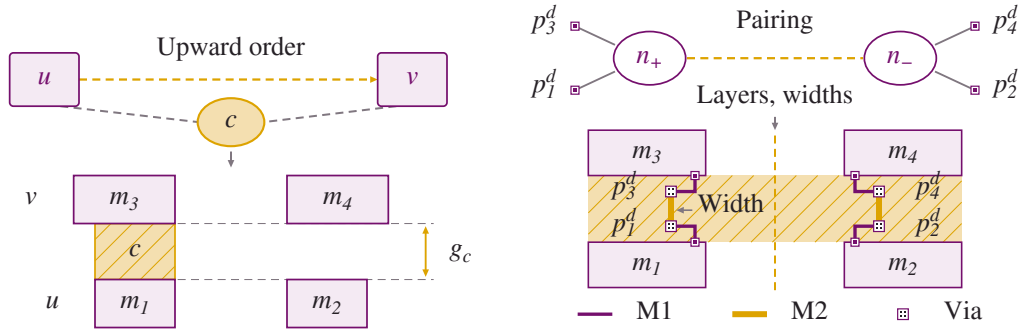
where $\text{mirror}_{x_b^{\text{ref}}}$ reflects every point (x, y) to $(2x_b^{\text{ref}} - x, y)$. It reflects the device orientation and pins together with the shape. In Figure 10(a), m_1 and m_2 form a differential pair with an internal gap g_b. Increasing this gap moves the devices apart about the reference axis while preserving their mirrored shapes, pins, and orientations.

Self-symmetry. A self-symmetric device lies on the axis and coincides with itself under reflection. The graph pairs the device with itself, requiring $S_i = \text{mirror}_{x_b^{\text{ref}}}(S_i)$. In Figure 10(b), the self-loop on tail device m_0 expresses this condition. A device of width w_i is centered on the axis by placing its left boundary at $x_i = x_b^{\text{ref}} - w_i/2$.

Common symmetry. Several symmetry groups may need to share one axis. A constraint edge records $x_u^{\text{ref}} = x_v^{\text{ref}}$ for blocks u and v. Placing the groups along a right-child chain preserves this equality during tree decoding. Figure 10(c) shows groups of different widths aligned at their symmetry axes; their left boundaries consequently differ.

Signal-flow ordering constraint. When required by circuit intent, signal flow can induce a signal-flow ordering constraint, such as placing an input pair below its load. A directed constraint edge records which block precedes the other and in which coordinate direction. In Figure 11(a), block v follows u upward, requiring $y_v \geq y_u + h_u$.

Minimum-spacing constraint. Boundary edges identify the facing device sides and their contact with a local passage, and jointly encode its minimum-spacing requirement $g_c \geq 0$. For facing horizontal sides of a lower block u and an upper block v, the condition is $y_v - y_u - h_u \geq g_c$; horizontal separation is constrained analogously. The highlighted passage in space c in Figure 11(a)



(a) Signal-flow ordering and minimum spacing.

(b) Differential route matching.

Figure 11: Signal-flow ordering, minimum spacing, and route matching in the graph (top) and layout (bottom). The directed constraint edge places the load above the input pair, while boundary edges specify the minimum gap g_c between their facing sides. The highlighted rectangle is their local passage in space c. Paired pins and matching properties specify the correspondence between differential routes.

illustrates this gap requirement. Joint PnR combines this bound with technology spacing and routing demand (Equation 6).

Route matching. Route matching specifies which properties of paired nets must match, such as paths, lengths, layers, wire widths, or via types. A constraint edge records the required properties and the pin pairings. For mirror routing, corresponding wires, layers, widths, and vias reflect about the specified axis, yielding equal route lengths. A length-only requirement instead bounds their relative difference. For two matched nets n_u, n_v with positive routed lengths, it requires

$$\frac{|\mathrm{WL}_{n_u} - \mathrm{WL}_{n_v}|}{\max\{\mathrm{WL}_{n_u}, \mathrm{WL}_{n_v}\}} \leq \text{matching tolerance}. \tag{8}$$

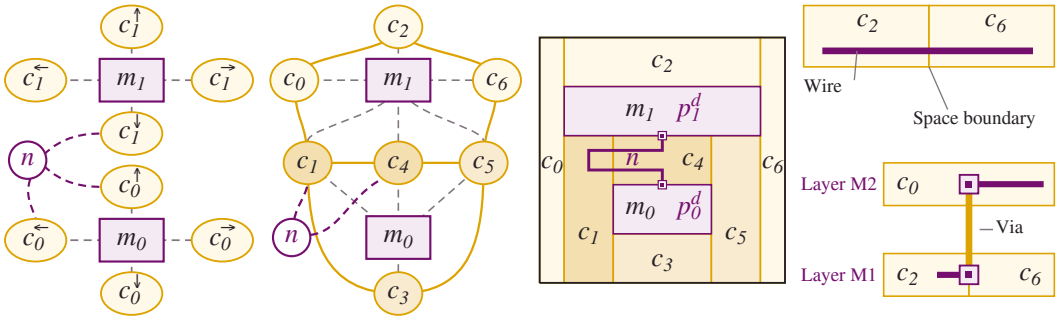
Figure 11(b) shows an example of mirror routing for differential nets. Drain pin p_i^d belongs to device m_i and faces the channel: n_+ connects p_1^d to p_3^d, and n_- connects p_2^d to p_4^d. The pin map pairs p_1^d with p_2^d and p_3^d with p_4^d. The two routes mirror each other's paths, layer choices, widths, and vias. Increasing the local gap reservations can lengthen both routes while preserving this correspondence (Figure 13 in Appendix B.7).

B.7 Fusion and Joint PnR Example

Figure 12 shows two separated devices with nested horizontal projections. Their eight initial side symbols refer to nine internal free rectangles and yield seven final rectangular nodes. The three equal-height rectangles above m_1 coalesce into c_2, retaining $c_1^{\uparrow}$. The three parts below m_1 are assigned to the existing left, top, and right symbols of m_0, producing c_1, c_4, and c_5; the redundant symbol $c_1^{\downarrow}$ is removed. The outer left and right rectangles are c_0 and c_6, and the rectangle below m_0 is c_3. The labels $c_0, \ldots, c_6$ follow geometric order and denote retained identities rather than newly created nodes.

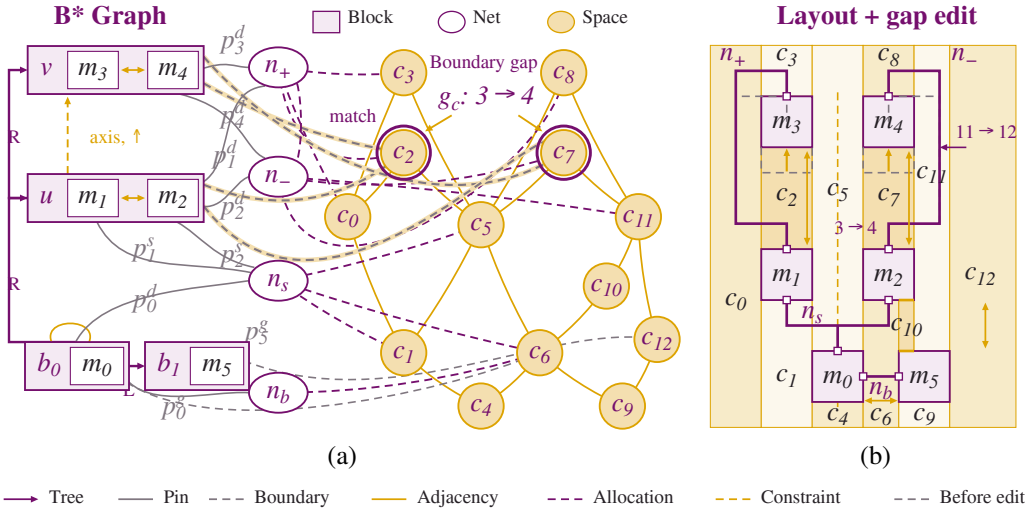
The illustrated layout connects p_0^d and p_1^d through allocated spaces c_4 and c_1. Joint PnR determines the detailed wire geometry and vias within these spaces. Colors distinguish the seven rectangular nodes. Wire and via footprints may cross artificial partition boundaries, while clearance is checked against physical obstacles.

Figure 13 combines the analog constraints in a six-device example. The tree specifies relative block placement, space adjacency describes routing passages, and net allocations identify the spaces used by each net. Increasing the minimum-spacing bounds recorded by the highlighted boundary edges moves the upper group and lengthens both differential routes. Device pairing, the shared symmetry axis, the active signal-flow ordering constraint, and route matching remain unchanged. Corresponding labels connect graph objects to their physical effects; the space nodes retain their identities,



(a) Space references. (b) Fused graph. (c) Routed layout. (d) Boundary and vias.

Figure 12: Space fusion and routing. (a) Each device carries four initial side symbols. (b) Canonical ownership assigns each coalesced rectangle to a distinct existing node. (c) Eight initial nodes become seven rectangles: the three internal rectangles above m_1 form c_2, and the bottom symbol of m_1 is removed. The illustrated route uses c_4 and c_1. (d) Wire and via footprints can cross rectangle boundaries.



(a) (b)

Figure 13: Six-device joint placement and routing. (a) Highlighted boundary edges encode local minimum-spacing requirements g_c in c_2 and c_7; tree branches R and L place children above and to the right, respectively. (b) Increasing the reservations from 3 to 4 moves the upper pair and increases both matched route lengths from 11 to 12, preserving symmetry. The horizontal projections of m_5 and m_2 overlap, enclosing passage c_{10}. Gray dashed lines show the initial positions and routes; values use illustrative units. Pin superscripts d, s, g denote drain, source, and gate.

adjacency, and net allocations as joint PnR realizes the changed geometry. Appendix B.6 defines the corresponding geometric constraints.

C Decision Procedure

Algorithm 1 implements the history-conditioned search in Section 3.3 for Problem 1. A decision initializes or perturbs the init graph to adjust device placement, net–space allocations, routing layers and wire widths, or analog constraints, or initiates detailed evaluation of the current candidate once eligible. Fixed postprocessing converts an init B* Graph into a fused B* Graph as input of joint

Algorithm 1 LLM optimization of analog physical design

Require: sized circuit D; fixed testbench $\mathcal{T}$; fixed quality thresholds $\mathbf{q}$; analog knowledge and physical tools

initialize H_1 with the circuit context, knowledge, and tools; set $t = 1$
while no candidate has passed all verification and quality requirements **do**
  choose $a_t \sim \mathrm{LLM}(\cdot \mid D, \mathcal{T}, \mathbf{q}, H_t)$
  if a_t specifies a graph edit **then**
    infer analog constraints; initialize or edit the init B* Graph G^{init}
    apply deterministic space fusion to obtain rectangular space nodes in G; validate associations and encoded constraints
    if the graph checks pass **then**
      jointly realize G as the current layout L; obtain basic evaluation and check active constraints
    else
      mark the edited candidate ineligible and record graph diagnostics
    end if
    retain any execution failure diagnostics
  else if the current candidate is eligible for detailed evaluation **then**
    run DRC, LVS, PEX, and simulation on L, advancing only after prerequisites succeed
    record verification diagnostics, extracted parasitics, and available electrical responses
    if paired parasitic estimates and PEX measurements are available **then**
      update parasitic-estimator parameters using paired estimates and PEX measurements
    end if
    if verification and functional tests pass and all quality thresholds $\mathbf{q}$ are met **then**
      return the current layout L and its verification and evaluation results
    end if
  end if
  retain the init graph, space-reference mapping, and available results in H_{t+1}; set $t \leftarrow t + 1$
end while

PnR. The objective in Equation 1 guides graph edits, and the acceptance requirements in Section 3.4 determine completion.

Detailed evaluation is available only when the current candidate is eligible as defined in Section 3.4. Even then, the LLM can choose to continue improving physical metrics. Each decision adds its results to the design history, including diagnostics from failed verification. Any change to a candidate's realized geometry creates a new candidate whose quality must be evaluated. A completed post-layout simulation provides design feedback; satisfaction of all fixed quality thresholds determines the final output. When enabled, the extensions in Appendix E supply evolving code and skills, accepted precedents, or engineer guidance to this search. Library updates admit only layouts that satisfy the full acceptance criteria.

Algorithm 1 specifies quality-based termination. Under a finite design budget, search can stop before any candidate is accepted. The experimental protocol in Appendix F.2 reports selected-candidate quality and Pass@1 separately from this verified-acceptance event.

D Quality Attainment in Adaptive Search

Fix a circuit D, testbench $\mathcal{T}$, and quality thresholds $\mathbf{q}$. At decision step t, the history H_t records the current and previous candidates, including their init graphs, fused graphs, available layouts and evaluation results, and the current design resources. The conditional decision rule in Section 3.3 induces an adaptive search process: both the proposed graph edits and the scheduling of detailed evaluation depend on this history. Probabilities below are taken over model sampling and any randomness in physical execution or evaluation.

Let T_{accept} be the first decision step at which detailed evaluation confirms that the current candidate satisfies its graph-encoded analog constraints, passes physical and functional verification, and meets the quality thresholds $\mathbf{q}$. Set $T_{\mathrm{accept}} = \infty$ if this event never occurs. Thus the event includes actual

verification of a realized layout, with its graph, geometry, and electrical measurements referring to the same candidate.

Proposition 1 (Quality attainment) *Suppose the search continues while no candidate has been fully verified to satisfy* $\mathbf{q}$*, each execution or evaluation returns an outcome, and there are deterministic lower bounds* $P_t \in [0,1]$ *such that, for each* $t \geq 1$*,*

$$\Pr(T_{\text{accept}} = t \mid H_t) \geq P_t \quad \textit{almost surely on } \{T_{\text{accept}} \geq t\}. \tag{9}$$

Then, for every observation horizon $T \geq 1$*,*

$$\Pr(T_{\text{accept}} > T) \leq \prod_{t=1}^{T}(1 - P_t) \leq \exp\left(-\sum_{t=1}^{T} P_t\right). \tag{10}$$

If $\sum_{t=1}^{\infty} P_t = \infty$*, the process reaches a fully verified layout satisfying* $\mathbf{q}$ *with probability one.*

Proof. Whether a candidate has already been accepted is determined by H_t. Conditioning on this history and applying Equation 9 gives

$$\begin{aligned} \Pr(T_{\text{accept}} > t) &= \mathbb{E}\left[\mathbf{1}_{\{T_{\text{accept}} \geq t\}} \Pr(T_{\text{accept}} > t \mid H_t)\right] \\ &\leq (1 - P_t) \Pr(T_{\text{accept}} > t - 1). \end{aligned} \tag{11}$$

Starting from $\Pr(T_{\text{accept}} > 0) = 1$ and iterating proves the product bound. Applying $1 - P_t \leq \exp(-P_t)$ proves the exponential bound. If the sum diverges, the bound tends to zero; continuity of probability yields $\Pr(T_{\text{accept}} = \infty) = 0$. The conditioning retains the full adaptive history, so the argument does not require independent decisions. □

The lower bound concerns the complete search and verification process: the current candidate must be eligible, undergo detailed evaluation, and be confirmed to satisfy the thresholds. A step devoted solely to a graph edit can have $P_t = 0$; persistent quality attainment depends on subsequent opportunities for successful verification. This conditional property connects history-dependent proposal and evaluation decisions to the framework's quality-based stopping rule. The observation horizon T measures search progress rather than prescribing when the design process terminates. The bound depends on the stated probabilities and continued search.

E Framework Extensions

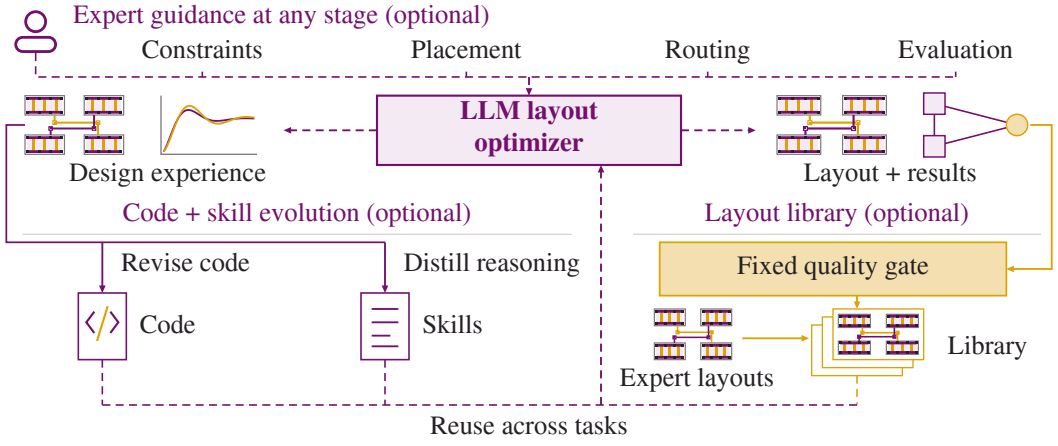


Figure 14: Optional extensions of the LLM layout optimizer. Engineers contribute design experience through natural-language guidance at any stage, while code and skills capture experience from design practice for reuse; generated layouts pass a fixed quality gate before entering the reusable library. Dashed connections denote optional paths.

Code and skill evolution. Figure 14 presents three optional extensions to the autonomous optimizer. Code and skill evolution turns design experience into reusable execution capabilities and reasoning. When evaluation reveals inadequate physical operations or unreliable estimates, the LLM can improve auxiliary design tools or parasitic estimators and evaluate the resulting candidates. The B* Graph representation of joint PnR and its construction and validation rules remain fixed. Skills summarize effective design choices, their applicability, and useful evaluation strategies. Both successful and unsuccessful attempts can inform these updates: code preserves how to execute an improvement, while skills preserve when and why to use it.

Layout library updates. A high-quality layout library starts from engineer-designed examples. None of the nine benchmark circuits appears in the pre-existing layout library or other knowledge repositories used by LAMBDA. Each record retains the circuit context, B* Graph representation, realized geometry, and paired evaluations. Retrieved precedents guide constraint generation, graph construction, and evaluation decisions for subsequent circuits, using verified design experience to improve their starting points. Generated layouts join the library after verification confirms that they satisfy all fixed quality thresholds, expanding the precedents available for reuse.

Engineer interaction. An engineer can guide any stage of design in natural language, including analog constraint formulation, device placement, routing, and evaluation. For example, the engineer may specify a matching constraint before layout generation, or request wider access around a sensitive node while preserving a differential pair's symmetry. The LLM turns this guidance into graph edits, physical operations, or evaluation actions, and assesses the resulting candidates through the same feedback loops under the fixed quality thresholds.

F BENCHMARK AND EVALUATION DETAILS

F.1 CIRCUITS AND COMPARED METHODS

Table 8 summarizes nine industrial circuits in a commercial TSMC 28-nm process: six analog circuits (Mixer, LPF, Ring-4, IF amp, Peak detector, and Z amp) and three mixed-signal modules (Div-2, TSPC-2, and TSPC-3), spanning oscillation, frequency division, amplification, filtering, mixing, sequential logic, and detection. All nine benchmark circuits are unseen design tasks for LAMBDA and are absent from its pre-existing layout library and other knowledge repositories. Their different functions require circuit-specific measurements and functional tests, covering frequency, gain, output swing, matching, current, bandwidth, stability, and division accuracy.

Table 8: Benchmark circuits.

Case	Circuit class	Reported responses
`Mixer_First_FrontEnd`	mixer front end	IF swing, mismatch
`LPF_simple`	low-pass filter	differential swing, mismatch
`4ring_2`	ring oscillator	frequency, differential swing
`IFAmpliflier`	amplifier	voltage gain
`peak_detection`	peak detector	sampled voltage, OUT p–p
`Z_AMP_INV`	inverting amplifier	voltage gain
`DIVIDER2`	frequency divider	division ratio, Q swing
`TSPC_DIVIDER2`	TSPC divider	division ratio, Q/QN matching
`TSPC_DIVIDER3`	TSPC divider	division ratio, output swing

All methods receive the same sized circuit and use identical circuit-specific testbenches and functional criteria in the same TSMC 28-nm HPC, 1P8M environment. Physical layouts are created in Cadence Virtuoso IC6.1.8, connectivity is checked and parasitics are extracted with Calibre 2020.4, and every extracted circuit is simulated with Spectre at the `top_tt` corner and 27 °C. A layout passes only if it clears DRC and LVS, completes PEX and post-layout simulation, and satisfies every circuit-specific functional criterion in Table 9 under the stated stimulus and evaluation window. This pass criterion does not require meeting all physical and electrical quality thresholds $\mathbf{q}$.

Table 9: Post-layout functional test criteria for Pass@1.

Case	Evaluation protocol and reported measures
Mixer	*Stimulus/window:* `mixer_0p9V_10M_9M`; full transient. *Pass:* IFI+ and IFI− p–p each > 0.001 mV. *Reported:* IF swing, mismatch.
LPF	*Stimulus/window:* `lpf_0p9V_1M`; full transient. *Pass:* OUTP and OUTN p–p each > 0.100 mV. *Reported:* swing, mismatch.
Ring-4	*Stimulus/window:* `ring_0p8V_0p50V`; 0.2–1 μs. *Pass:* LOI+/LOI− nonconstant; second-to-third rising-crossing period > 0. *Reported:* frequency, swing, mismatch.
IF amp	*Stimulus/window:* `ifamp_0p9V_10k`; full transient. *Pass:* finite OUT; OUT p–p > 0.100 mV. *Reported:* gain, output swing.
Peak det.	*Stimulus/window:* standard transient (VIN 0.3–0.7 V); 0–40 ns. *Pass:* 0 <OUT at 30 ns< 1.5 V; OUT p–p > 1.000 mV. *Reported:* sampled voltage, OUT p–p; extended tests report IQ, UGB, and phase margin (PM).
Z amp	*Stimulus/window:* `zamp_0p8V_2M`; full transient. *Pass:* finite nonconstant OUT; gain > 0.1. *Reported:* gain, output swing.
Div-2	*Stimulus/window:* `divider_0p8V_1ns`; 5–20 ns. *Pass:* sufficient CP/Q crossings; $1.8 < r < 2.2$. *Reported:* ratio, Q swing.
TSPC-2	*Stimulus/window:* `divider_0p8V_1ns`; 5–20 ns. *Pass:* sufficient CLK/Q crossings; $1.8 < r < 2.2$. *Reported:* ratio, swing, mismatch.
TSPC-3	*Stimulus/window:* `divider_0p8V_1ns`; 5–20 ns. *Pass:* sufficient CLK/OUT crossings; $2.7 < r < 3.3$. *Reported:* ratio, output swing.

Baseline configurations. SA (Ou et al., 2013) performs simultaneous placement and routing with current-flow and current-density considerations. CDLS (Mangalagiri et al., 2024) learns design constraints and synthesizes layouts through placement based on mixed-integer linear programming (MILP), MILP-based global routing, and detailed routing based on Boolean satisfiability (SAT). PANDA (Zhang et al., 2026) uses LLMs to coordinate topology, sizing, and layout generation, incorporating post-layout feedback into performance-driven refinement. Its official implementation uses `gpt-4o` (OpenAI, 2024); for a comparison with the same backbone, we evaluate PANDA with `gpt-5.6-sol`, as in LAMBDA. AlphaPlacer (Li et al., 2026) uses expert layout priors and similarity rewards to guide Monte Carlo tree search (MCTS) over placements represented by sequence pairs. Following the setup in its paper, the resulting placement is automatically routed using MAGICAL (Chen et al., 2021). PAPnR combines PAPlace (Xu et al., 2026), a differentiable placer guided by a learned performance surrogate, with the performance-driven routing guidance of PARoute2 (Xu et al., 2025). All methods produce routed, extracted layouts for evaluation. For baselines that require analog constraints as input, expert engineers provide these constraints in advance, whereas LAMBDA generates and adjusts its constraints autonomously.

F.2 METRICS AND RUN ACCOUNTING

For the complete physical design comparison in Section 4.2 and the post-layout electrical performance comparison in Section 4.3, each method performs one continuous closed-loop optimization task per circuit with at most six design rounds, retaining design state, history, and feedback across rounds. Tables 2(b)–(c) and 3 report the candidate with the highest figure of merit (FoM) among those rounds. All physical and electrical measurements for a method on a circuit refer to that same selected layout. Quality thresholds, testbenches, and functional criteria remain fixed throughout the comparison.

In the main comparison, ablations, and controls, runtime and token costs are measured only in the continuous layout-quality optimization tasks. Independently initialized rounds used to measure Pass@1 contribute only to Pass@1, not to cost averages. In Table 2(a), runtime is averaged arithmetically over the nine circuits, and token usage is averaged over executed design rounds.

A selected candidate is an accepted output of Algorithm 1 only if all verification and quality requirements are met. Budget-limited candidate reporting also includes layouts that pass functional tests but fall short of a quality threshold; Appendix A evaluates the reported analog layouts against the expert-relative thresholds.

For Pass@1, each method is evaluated over ten independently initialized design rounds on each of the nine circuits, separately from the continuous layout-quality optimization tasks. Pass@1 is the number of passing rounds divided by 90; all measured rounds remain in the denominator, including those that end before post-layout simulation. Stochastic components use paired random seeds across methods, with the same seed assigned to corresponding rounds on the same circuit.

Before each round used to measure Pass@1, the same initial resources are restored for the respective condition, including the initial graph, code, skills, and layout library where applicable, and all prior design history is cleared. No experience or resource updates carry over between these rounds.

Full layout area is the bounding box area of the complete routed layout in $\mu\mathrm{m}^2$. Wirelength is the sum of routed wire centerline lengths in $\mu\mathrm{m}$. In Table 2(b)–(c), AVG reports the geometric mean of the nine absolute measurements and, in parentheses, the geometric mean of the nine method/LAMBDA ratios. All methods and physical metrics use the same nine circuits.

Analog AVG is the equally weighted arithmetic mean of the analog-circuit scores. Each circuit score is the arithmetic mean of all its metric scores in Table 3(a), including PM. For method m, circuit c, and metric j, the score is $s_{mcj} = x_{mcj}/\max_k x_{kcj}$ for $\uparrow$ metrics and $s_{mcj} = \min_k x_{kcj}/x_{mcj}$ for $\downarrow$ metrics, with extrema taken across the compared methods and schematic-reference saturation for Mixer mismatch and Peak det. PM as defined below. Thus, scores lie in $[0, 1]$, and

$$\text{Analog AVG}_m = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \frac{1}{|\mathcal{J}_c|} \sum_{j \in \mathcal{J}_c} s_{mcj},$$

where $\mathcal{C}$ contains the analog circuits and $\mathcal{J}_c$ contains the metrics for circuit c. The parenthesized method/LAMBDA ratios in the table describe raw measurements, rather than these metric scores.

Analog AVG summarizes relative electrical quality across the compared methods. It is neither a fraction of schematic performance retained nor a functional pass rate: available responses are scored even when a circuit fails its functional criterion, with those failures reported in Appendix I.

Mean output swing averages the two output swings; absolute output swing mismatch is their absolute difference, and normalized mismatch divides that difference by the larger swing.

For Ring-4, the paired outputs are LOI+ and LOI−, and schematic-level simulation gives a reference frequency of 410 MHz. All post-layout frequencies in Table 3 are below this reference, so the $\uparrow$ direction indicates closer agreement with the schematic response over the reported range.

For Peak det., OUT p–p is the maximum-minus-minimum output voltage over the full 0–40 ns transient, including startup, under the fixed stimulus in Table 9; OUT@30 ns is the output voltage sampled at 30 ns. Schematic-level simulation gives 114.805 mV and 1.074614 V, respectively. All post-layout values in Table 3 are below their respective references, so $\uparrow$ indicates closer agreement with the schematic responses over the reported range. Steady-state ripple and peak-detection error require separate measurements. Schematic-level stability analysis gives a phase margin of 91.83218° under the same bias and load as the post-layout stability tests. Its score is $s_{\mathrm{PM}} = \min(1, \mathrm{PM}/91.83218^\circ)$, assigning full credit once the schematic phase margin is reached.

For Mixer, schematic-level simulation under the same stimulus and full 0–$5\,\mu$s transient window gives a normalized output swing mismatch of $d_0 = 0.187225\%$. Scoring uses the effective mismatch $\widetilde{d}_m = \max(d_m, d_0)$ and $s_{m,\mathrm{mismatch}} = \min_k \widetilde{d}_k/\widetilde{d}_m$. Matching at or below the schematic mismatch receives full credit; further reductions receive no additional reward. The table reports the measured mismatch d_m.

Division error is $|r - r^\star|$, where r is the measured ratio and $r^\star = 2, 2, 3$ for Div-2, TSPC-2, and TSPC-3. Errors are expressed in units of 10^{-6}; their method/LAMBDA ratios are omitted because LAMBDA has zero error. Matched net length mismatch is the largest absolute length difference divided by the longer path across matched net pairs.

A design round comprises iterative candidate refinement on a single circuit and at most one detailed-evaluation attempt. Graph edits, layout synthesis, and basic evaluation may repeat within the round without increasing the round count. Detailed evaluation comprises DRC, LVS, PEX, and post-layout simulation; the round ends when this attempt completes or is discontinued. Further refinement or resubmission after the round ends starts a new round. Resuming interrupted tools or measuring multiple responses within an ongoing evaluation does not. A round abandoned before detailed

evaluation counts as failed. All internal computation contributes to the round's cost. The table below lists each method's internal operations and the candidate submitted for detailed evaluation.

Method	Internal operations within a round	Candidate submitted for detailed evaluation
SA	Annealing moves, temperature steps, and unselected candidates.	One joint PnR candidate.
CDLS	Constraint prediction, MILP solver iterations, and internal routing steps.	One candidate from constraint-driven placement and routing.
PANDA	Internal tool or search steps.	One candidate from an LLM-coordinated design cycle.
AlphaPlacer	Tree expansion, rollouts, and unselected nodes.	One MCTS-selected placement routed by MAGICAL.
PAPnR	Placement gradient updates and internal routing iterations.	One PAPlace/PARoute2 candidate.
LAMBDA	Repeated graph edits, layout synthesis, and basic evaluation.	One routed layout synthesized from a B* Graph.

F.3 ABLATION STUDY AND EVOLUTION PROTOCOLS

Table 10 summarizes the evaluation protocols.

The LLM-based LPF component ablations use `gpt-5.6-sol` at `max` reasoning effort; *w.o. LLM* uses simulated annealing for candidate decisions. For layout-quality evaluation, each ablation and backbone configuration performs one continuous closed-loop optimization task on LPF with at most six design rounds, retaining design history and feedback across rounds. The reported layout is selected by the highest FoM. A candidate can undergo further refinement in a subsequent round after passing the functional test. Routing failures may be addressed within the ongoing round; a round abandoned because routing cannot be completed counts as failed, even without detailed evaluation. Model-call retries within an ongoing round do not create extra rounds.

Geometric and electrical measurements refer to the same candidate. If no valid post-layout results are obtained during the six-round layout-quality optimization task, we report the available geometry of the candidate layout from the last round. Table 13(c) reports mean token usage and runtime over all executed rounds of the continuous layout-quality optimization task, including failed rounds and the cost of retries within each round.

Separately, each ablation and backbone configuration is evaluated over 100 independently initialized design rounds on LPF. Pass@1 is the number of passing rounds divided by 100, with all measured rounds included in the denominator. Before each round, the condition's initial resources are restored and the design history is cleared as specified in Appendix F.2.

The detailed feedback ablation withholds the results returned by detailed evaluation from the design policy, including PEX results, pre- and post-layout simulation metrics, and electrical retention. The same final post-layout test measures every condition's outcome.

The evolution study combines fixed-input routing comparisons, code and experience statistics, and an LPF trajectory of six evolution steps. Routing comparisons on Ring-4, Mixer, and LPF hold placement, design intent, device geometry, and connectivity fixed; the TSPC-3 comparison evaluates independent and joint selection of mirrored pin exits on the same candidate. The LPF trajectory continues from its first functional layout toward a full layout area of at most $3000\,\mu\mathrm{m}^2$, carrying code, skills, and design feedback across steps. Appendix K reports the measurements, resource updates, and design outcomes.

G COMPLETE GEOMETRY RESULTS

Table 11 reports the absolute geometry of every LAMBDA candidate used in the main comparison. Full layout area ranges from $86.718\,\mu\mathrm{m}^2$ for TSPC-2 to $10{,}756.934\,\mu\mathrm{m}^2$ for Peak det., motivating per-circuit ratios and geometric means across circuits.

Table 10: Evaluation protocols.

Study	Controlled protocol
Policy, feedback, and joint PnR ablations	LPF; six conditions; common circuit and experimental pass criteria; Pass@1; mean cost per executed round.
Code, skills, and experience	Fixed-input routing on three circuits; code and distinct experience record counts; six LPF evolution steps with two layouts receiving detailed evaluation.
Backbone and framework	LPF; LAMBDA (Sol), LAMBDA (Astra), and Tools-only (Astra); at most six closed-loop rounds per layout-quality task; Pass@1 measures the fraction of independently initialized design rounds passing the corresponding functional test in Table 9; failed rounds consume budget.

Table 11: LAMBDA geometry.

Case	Full layout area (μm^2)	Wirelength (μm)
Mixer_First_FrontEnd	3,715.101	2,098.589
LPF_simple	2,842.740	2,187.670
4ring_2	2,972.372	1,634.123
IFAmpliflier	7,956.231	2,193.230
peak_detection	10,756.934	1,692.266
Z_AMP_INV	1,244.622	255.095
DIVIDER2	154.605	519.937
TSPC_DIVIDER2	86.718	271.734
TSPC_DIVIDER3	152.776	573.372

H CIRCUIT-LEVEL PERFORMANCE COMPARISONS

Figure 15 compares full layout area, routed wirelength, and the designated post-layout primary response for every circuit. Values are method/LAMBDA ratios. Values above one favor LAMBDA for area and wirelength, while values below one favor LAMBDA for the primary response. AVG denotes the geometric mean over all nine circuits. The primary responses are drawn from Table 3; its Analog AVG also includes matching and operating characteristics, assigning equal weight to each analog circuit and averaging bounded metric scores equally within each circuit (Appendix F.2). Table 12 provides the full electrical measurements.

I COMPLETE POST-LAYOUT COMPARISON

Table 12 reports the complete electrical measurements for all six methods, including the frequency, gain, output swing, matching, division accuracy, and operating characteristics compared in the main text. Frequency is in MHz and voltage amplitudes are in mV unless stated otherwise. Metric definitions and aggregation follow Appendix F.2.

All methods except CDLS satisfy every circuit-specific functional criterion. CDLS has insufficient crossings to measure the Div-2 periods and ratio and fails the Peak det. activity check despite producing finite extracted responses. Across valid responses, different methods lead on different metrics for individual circuits, motivating the vector-valued comparison.

Analog-circuit performance. LAMBDA achieves the highest Analog AVG (0.983) in Table 3(a), exceeding the five baselines by 25.5%–55.7%. On Mixer and LPF, it achieves the highest mean output swings and lowest normalized mismatches. On Ring-4, it reaches 400.896 MHz (18.4% above PAPnR), with the highest mean output swing (297.424 mV) and lowest normalized mismatch (0.900%), although PAPnR has a slightly lower absolute output swing mismatch (2.499 versus 2.689 mV). LAMBDA also obtains the highest gains on IF amp and Z amp.

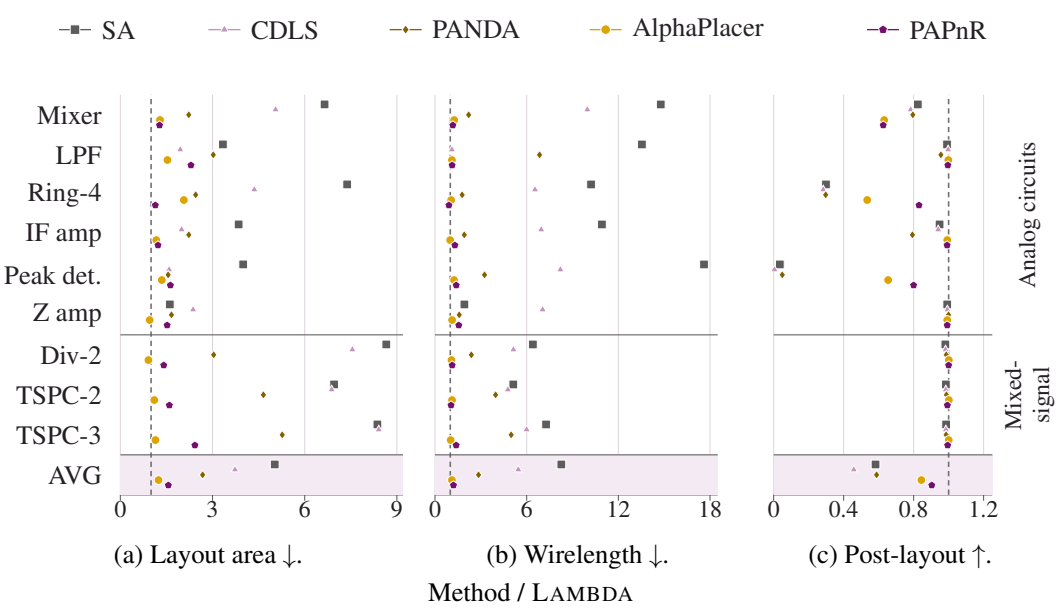


Figure 15: Physical and post-layout performance by circuit. The Ring-4 response combines normalized frequency and mean output swing geometrically.

Mixed-signal performance. The mixed-signal modules exhibit stable post-layout output swings with little variation across methods (Table 3(b)), while LAMBDA retains substantial advantages in physical quality (Table 2). Figure 15 provides the circuit-level physical and electrical comparisons.

Table 12: Post-PEX electrical details.

Circuit	Metric [unit]	SA	CDLS	PANDA	AlphaPlacer	PAPnR	LAMBDA (Ours)
Analog circuits							
Mixer	ifn_max [V]	0.2316962	0.2285686	0.2522888	0.223716	0.2252829	0.2370046
	ifn_min [V]	0.1753678	0.1768059	0.1977785	0.1803874	0.182487	0.1688685
	ifn_pp [mVpp] ↑	56.32836	51.7627	54.51029	43.3286	42.79596	68.13617
	ifp_max [V]	0.2305294	0.2311005	0.2493623	0.2270587	0.2249831	0.2363061
	ifp_min [V]	0.1745918	0.1761321	0.1954577	0.1843019	0.1824513	0.1681688
	ifp_pp [mVpp] ↑	55.9376	54.9684	53.90456	42.7568	42.53185	68.1373
	pp_abs_mismatch [μV] ↓	390.76	3205.7	605.73	571.8	264.11	1.13
	pp_normalized_mismatch [%] ↓	0.693718	5.831896	1.111221	1.319683	0.617138	0.001658
LPF	outn_max [V]	0.2058764	0.2060416	0.2074034	0.2057959	0.2059741	0.206085
	outn_min [V]	0.2007216	0.2008577	0.2024286	0.2006063	0.2007998	0.2008873
	outn_pp [mVpp] ↑	5.154808	5.1839	4.974868	5.1896	5.174306	5.197737
	outp_max [V]	0.2057144	0.2058518	0.2071923	0.2055051	0.2058325	0.2058951
	outp_min [V]	0.2005669	0.2006696	0.2022311	0.2003139	0.2006567	0.2006978
	outp_pp [mVpp] ↑	5.147503	5.1822	4.96114	5.1912	5.175713	5.197319
	pp_abs_mismatch [μV] ↓	7.305	1.7	13.728	1.6	1.407	0.418
	pp_normalized_mismatch [%] ↓	0.141712	0.032794	0.275947	0.030821	0.027185	0.008042
Ring-4	frequency [MHz] ↑	68.93685	46.81364	161.3017	216.083	338.7193	400.8964
	loi_minus_max [V]	0.4702974	0.4901495	0.3791203	0.517228	0.5952831	0.5422493
	loi_minus_min [V]	0.3206702	0.2929936	0.3112422	0.3575542	0.3511446	0.2461702
	loi_minus_pp [mVpp] ↑	149.6272	197.1559	67.8781	159.6738	244.1385	296.0791
	loi_plus_max [V]	0.4746879	0.4975216	0.3772721	0.4344984	0.5927132	0.5383532
	loi_plus_min [V]	0.3150403	0.2853742	0.3144261	0.2786595	0.3510735	0.2395852
	loi_plus_pp [mVpp] ↑	159.6476	212.1474	62.846	155.8389	241.6397	298.768
	period [ns] ↓	14.50603	21.3613	6.199561	4.627851	2.952297	2.49441
	pp_normalized_mismatch [%] ↓	6.276574	7.066549	7.413437	2.401709	1.023517	0.899996
IF amp	gain [1] ↑	0.312346	0.30975	0.261213	0.326842	0.326542	0.329376
	out_end [V]	0.0995491	0.0995418	0.0996176	0.0995451	0.0995332	0.0995243
	out_max [V]	0.1050208	0.1050142	0.104305	0.1052685	0.1052842	0.1053554
	out_min [V]	0.0987739	0.0988192	0.0990808	0.0987317	0.0987534	0.0987679
	out_pp [mVpp] ↑	6.246921	6.195	5.224263	6.536822	6.530842	6.587513
	out_start [V]	0.1	0.1000008	0.1	0.0999999	0.0999998	0.0999996
Peak det.	DC: iq [μA] ↓	32.77334	32.77417	25.363346	32.74191	32.77585	25.31023

Continued on next page

Table 12 continued

Circuit	Metric [unit]	SA	CDLS	PANDA	AlphaPlacer	PAPnR	LAMBDA (Ours)
	out_at_30ns [V]	0.9639215	0.9613013	0.9562621	1.010331	1.015375	1.024231
	out_max [V]	0.9643753	0.9615489	0.9589862	1.014208	1.019058	1.028944
	out_min [V]	0.96179	0.9612175	0.9553793	0.966134	0.9602792	0.9554601
	out_pp [mVpp] ↑	2.585215	0.331335	3.6068	48.07405	58.77863	73.48419
	vin_max [V]	0.699993	0.6999786	0.6999971	0.6999997	0.6999999	0.7
	vin_min [V]	0.3000058	0.3000001	0.3000054	0.3000015	0.3	0.3000001
	STB: pm [°] ↑	91.70679	91.5707	80.558704	95.3009	92.67983	96.50421
	STB: ugb [MHz] ↑	2.886471	2.928134	12.951468	2.781841	2.883235	6.311568
	100-mVpp: out_at_30ns_100mvpp [V]	0.9625976	0.9615672	0.9571185	0.9713125	0.9667265	0.9674368
	200-mVpp: out_at_30ns_200mvpp [V]	0.9637194	0.961763	0.9568426	0.9846428	0.9832106	0.9875478
	50-mVpp: out_at_30ns_50mvpp [V]	0.9620341	0.9613864	0.9578193	0.9673065	0.961545	0.9590781
Z amp	gain [1] ↑	37.82795	37.858205	38.11429	37.83387	37.82439	38.12842
	in_max [V]	0.01	0.01	0.0099999	0.0099988	0.0099999	0.009999939
	in_min [V]	-0.0099996	-0.01	-0.0099998	-0.0099998	-0.01	-0.009999874
	input_pp [mVpp]	20	20	20	20	20	20
	out_max [V]	0.7803711	0.7813585	0.781	0.7827486	0.7820714	0.7824398
	out_min [V]	0.023812	0.0241945	0.0187143	0.0260712	0.0255837	0.01987142
	out_pp [mVpp] ↑	756.5591	757.1641	762.2857	756.6774	756.4877	762.5684
Mixed-signal circuit modules							
Div-2	cp_max [V]	0.8	0.8	0.8	0.8	0.8	0.8
	cp_min [V]	0	0	0	0	0	0
	cp_period [ns]	1	-	1	1	1	1
	q_10n [V]	0.7999676	0.0047581	0.7999941	0.7999994	0.7999994	0.7999994
	q_20n [V]	0.7999677	0.0047514	0.7999941	0.7999994	0.7999995	0.7999994
	q_max [V]	0.8011138	0.8006499	0.8019995	0.8085409	0.80742	0.8075642
	q_min [V]	0.0021363	0.0009445	-0.001203	-0.0066557	-0.0068256	-0.0065622
	q_period [ns]	2	-	2	2	1.999996	2
	q_pp [mVpp] ↑	798.977457	799.705384	803.202461	815.19662	814.245583	814.126402
	ratio [1]	2	-	2	2	1.999996	2
	$\lvert r-2\rvert$ [10^{-6}] ↓	0	-	0	0	4	0
TSPC-2	clk_max [V]	0.8	0.8	0.8	0.8	0.8	0.8
	clk_min [V]	0	0	0	0	0	0
	clk_period [ns]	1	1	1	1	1	1
	pp_normalized_mismatch [%] ↓	0.82671	0.2853	0.665087	0.859605	1.646755	0.86544
	q_10n [V]	0.7999998	0.7999998	0.7999995	0.7999999	0.8	0.7999998
	q_20n [V]	0.7999998	0.7999998	0.7999995	0.7999999	0.8	0.7999998
	q_max [V]	0.8016448	0.8014158	0.801651	0.8085088	0.8075422	0.8074525
	q_min [V]	0.0000324	0.0001876	-0.0024711	-0.0070246	-0.0010961	-0.0069498
	q_period [ns]	2	2.000005	2.00001	2.000001	2.000009	2
	q_pp [mVpp] ↑	801.612364	801.228191	804.122127	815.533393	808.638333	814.402297
	qn_10n [V]	-0.0042318	-0.0005761	-0.0051749	-0.0125562	-0.0127799	-0.0118007
	qn_20n [V]	-0.0042318	-0.0005741	-0.0051754	-0.0125633	-0.0127794	-0.0118007
	qn_max [V]	0.8011311	0.8005352	0.8012497	0.8059563	0.8057803	0.8052315
	qn_min [V]	-0.0071635	-0.0029854	-0.0082563	-0.0166482	-0.0163973	-0.0162805
	qn_pp [mVpp] ↑	808.294613	803.520637	809.506043	822.60454	822.17758	821.51199
	ratio [1]	2	2.000005	2.00001	2.000001	2.000009	2
	$\lvert r-2\rvert$ [10^{-6}] ↓	0	5	10	1	9	0
TSPC-3	clk_max [V]	0.8	0.8	0.8	0.8	0.8	0.8
	clk_min [V]	0	0	0	0	0	0
	clk_period [ns]	1	1	1	1	1	1
	out_max [V]	0.8017859	0.800887	0.80106	0.8075056	0.8070381	0.8078616
	out_min [V]	-0.0008641	-0.0004783	-0.0013471	-0.0072609	-0.002725	-0.0063508
	out_period [ns]	3	3.000001	3.000002	3	3	3
	out_pp [mVpp] ↑	802.650028	801.365322	802.407123	814.766547	809.763114	814.212409
	ratio [1]	3	3.000001	3.000002	3	3	3
	$\lvert r-3\rvert$ [10^{-6}] ↓	0	1	2	0	0	0

↓/↑ indicate smaller/larger is better; – denotes an unavailable measurement.

I.1 ADDITIONAL PEAK DET. RESPONSES

The Peak det. evaluator supplements the standard transient response with DC, stability, and extended-amplitude measurements. Table 12 reports these axes separately to show the current, bandwidth, stability, and input-amplitude dependence.

For the standard transient response, LAMBDA's full-window OUT p–p of 73.484 mV exceeds PAPnR by 25.0%; both this response and its sampled output at 30 ns are closest to their schematic

references (Appendix F.2). It also achieves the lowest quiescent current (IQ) and a phase margin (PM) above its schematic reference, while PANDA attains the highest unity-gain bandwidth (UGB).

J MECHANISM ABLATIONS

J.1 DECISION POLICY, EVALUATION FEEDBACK, AND JOINT PNR

We study evaluation feedback, joint PnR, and LLM decisions on LPF under common circuit inputs, testbenches, and experimental pass criteria. The LLM-based component ablations use `gpt-5.6-sol` at `max` reasoning effort, with the complete LAMBDA (Sol) as the reference. Each feedback ablation removes one channel supplied to the LLM while retaining this configuration. The final verification and post-layout test remain common to all conditions; withheld results are unavailable to the LLM for B* Graph editing. Table 13 reports the selected layouts. Its Pass@1 values come from 100 independently initialized design rounds per condition, with initial resources restored and design history cleared between rounds (Appendix F.3).

w.o. basic visual feedback removes the layout images returned by basic evaluation. Relative to LAMBDA (Sol), Pass@1 falls from 100% to 43% and layout area increases by 43.3%, supporting the role of visual feedback in placement and routing-space refinement.

w.o. metric eval. feedback withholds area, wirelength, whitespace, and routing-quality metrics, together with symmetry constraint-check feedback, from basic evaluation. Pass@1 falls to 0%, and the layout-quality optimization task fails during routing and never reaches electrical evaluation. Its last candidate has 40.30% whitespace; no routing geometry is saved. This failure highlights the importance of basic metric feedback for routability.

w.o. detailed feedback removes the results returned by detailed evaluation, including PEX results, pre- and post-layout simulation metrics, and electrical retention. Mean OUT p–p barely changes relative to LAMBDA (Sol), but absolute swing mismatch reaches $7.65\times$ the reference despite 4.8% shorter wirelength. Detailed feedback therefore supports output matching beyond what mean swing and wirelength alone can indicate.

w.o. joint PnR fixes placement before routing and prevents subsequent routing feedback from changing it. The LPF joint/sequential comparison uses the same circuit and experimental pass criteria. Relative to LAMBDA (Sol), Pass@1 falls to 57%, full layout area increases by 9%, absolute swing mismatch increases by 86%, and token usage more than doubles (Table 13). Wirelength is slightly shorter, at $0.97\times$ the reference, while the mean post-layout response ratio is 0.9998. These results support joint PnR for improving reliability and matching quality while reducing token usage. Both reported layouts have 100% completion and one post-layout attempt each. Post-layout attempts count detailed evaluations within the cost-measured design tasks.

w.o. LLM uses the same circuit input, testbench, and analog layout synthesizer as LAMBDA, with simulated annealing (SA) perturbing the same init B* Graph. Both optimizers have the same placement and routing perturbation operations and scope for joint perturbations, covering device grouping, B* Tree parent–child relations, orientations, reserved gaps, net–space allocations, routing layers, and wire widths, with coordinated updates to symmetric devices and nets subject to route-matching constraints. Expert engineers provide the analog constraints for SA, whereas LAMBDA generates and adjusts its constraints autonomously. Basic evaluation settings, eligibility criteria for detailed evaluation, and parasitic-estimate calibration are identical to those in LAMBDA. SA prioritizes physical feasibility and scores candidates using area, whitespace, aspect ratio, and measured electrical responses, including mismatch. When detailed evaluation provides electrical measurements, they enter the SA score and guide candidate acceptance and subsequent perturbations. SA explores candidates through geometric cooling and the Metropolis acceptance criterion. Both conditions use the same DRC, LVS, PEX, post-layout functional tests, and fixed quality thresholds.

The selected layout of the *w.o. LLM* control passes DRC, LVS, PEX, and the post-layout functional test, while the condition achieves 33% Pass@1. Relative to LAMBDA (Sol), its area, wirelength, and absolute swing mismatch are $2.11\times$, $1.26\times$, and $2.25\times$, respectively. Mean runtime falls from 9.99 to 5.85 minutes per round, and no LLM calls are required. This LPF comparison favors LLM-guided search over SA with expert-provided constraints in reliability, compactness, and matching, even when both use measured electrical feedback. Internal SA proposals do not count as separate

Table 13: Ablation study details.

(a) Physical design

Variant	Layout area [μm^2]	Wirelength [μm]	Whitespace [%]	Matched net length mismatch [%]	Vias	Bends
LAMBDA (Sol)	2842.7	2187.7	31.38	0.71	706	208
w.o. basic visual feedback	4075.0	2271.8	52.95	0.37	701	218
w.o. metric eval. feedback	-	-	40.30	-	-	-
w.o. detailed feedback	2968.3	2083.7	34.84	0.51	738	198
w.o. joint PnR	3091.6	2124.5	36.18	0.27	689	185
w.o. LLM	5999.3	2754.3	68.52	0.40	849	194
LAMBDA (Astra)	2960.0	1983.5	34.75	-	-	-
Tools-only (Astra)	3843.0	2341.8	48.29	3.59	554	254

(b) Post-layout electrical responses and verification

Variant	OUTP p–p [mV]	OUTN p–p [mV]	mean OUT p–p [mV]	mismatch [%] ↓	abs. mismatch [μV]	DRC	LVS	PEX	Functional test
LAMBDA (Sol)	5.197319	5.197737	5.197528	0.008042	0.418	Pass	Pass	Pass	Pass
w.o. basic visual feedback	5.181093	5.181534	5.181314	0.008511	0.441	Pass	Pass	Pass	Pass
w.o. metric eval. feedback	-	-	-	-	-	-	-	-	-
w.o. detailed feedback	5.198377	5.201576	5.199977	0.061501	3.199	Pass	Pass	Pass	Pass
w.o. joint PnR	5.196358	5.197136	5.196747	0.014970	0.778	Pass	Pass	Pass	Pass
w.o. LLM	5.174295	5.175237	5.174766	0.018202	0.942	Pass	Pass	Pass	Pass
LAMBDA (Astra)	-	-	5.210533	0.009615	0.501	Pass	Pass	Pass	Pass
Tools-only (Astra)	4.890153	4.891652	4.890903	0.030646	1.499	Pass	Pass	Pass	Pass

(c) Pass@1 and mean cost per executed round

Variant	Pass@1 ↑	Input [k/round]	Output [k/round]	Total [k/round]	LLM time [min/round]	Total time [min/round]
LAMBDA (Sol)	100%	45.84	9.27	55.11	5.11	9.99
w.o. basic visual feedback	43%	41.58	6.05	47.63	5.16	10.25
w.o. metric eval. feedback	0%	55.52	6.27	61.79	3.63	5.25
w.o. detailed feedback	71%	40.23	9.23	49.46	5.18	16.61
w.o. joint PnR	57%	106.29	11.03	117.33	-	10.64
w.o. LLM	33%	-	-	-	-	5.85
LAMBDA (Astra)	100%	132.09	12.28	144.37	18.76	36.74
Tools-only (Astra)	51%	1924.42	14.34	1938.76	50.15	53.20

– denotes a measurement that is not applicable, unrecorded, or unreached.

design rounds. The round limit bounds closed-loop design attempts, while internal search operations and tool calls can differ; runtime accounts for the executed computation. Token costs and LLM time are not applicable.

LAMBDA (Sol), the basic visual feedback ablation, the detailed feedback ablation, and the joint PnR ablation all produce layouts that pass DRC, LVS, PEX, and the post-layout functional test. Among the routed variants, matched net length mismatch remains nonzero, showing that exact device symmetry does not by itself ensure matched routed net lengths.

Electrical and matching metrics follow Appendix F.2. Parenthesized ratios in Table 4 use LAMBDA (Sol) as the reference. Conditions without valid post-layout results retain their last available geometry and are excluded from best-value comparisons. Table 13(c) gives mean token usage and runtime over all executed rounds of the layout-quality optimization task, including failed rounds and the cost of retries within each round. Total time includes detailed evaluation when performed. The joint PnR ablation lacks separate LLM timing but has complete total runtime and token records. Appendix J.2 describes the backbone and framework controls.

J.2 BACKBONE AND FRAMEWORK CONTROLS

We use three configurations to separate two questions. Comparing LAMBDA with `gpt-5.6-sol` and `gpt-6-astra` tests a stronger model configuration with the framework held fixed. Comparing LAMBDA (Astra) with Tools-only (Astra) tests the contribution of the framework with the same

model and reasoning effort. Each configuration uses the same sized LPF netlist, PDK, technology files, testbench, and quality thresholds. The two LAMBDA configurations additionally share their initial code, skills, layout library, and tool interfaces; resources do not transfer between configurations.

LAMBDA (Sol) uses `gpt-5.6-sol` at `max` reasoning effort to keep repeated design iterations computationally affordable.

LAMBDA (Astra) uses `gpt-6-astra` at `ultra` reasoning effort to test whether a stronger model configuration substantially improves results within the same framework.

Tools-only (Astra) also uses `gpt-6-astra` at `ultra`, allowing the framework's contribution to be assessed under that same model configuration.

Tools-only (Astra) provides generic code execution, file and layout database access, and the necessary physical design, DRC, LVS, PEX, and simulation tools. Tool documentation specifies interfaces and argument formats. This condition supplies no prior knowledge such as the LAMBDA controller, analog layout synthesizer, learned layout library, prepared placement and routing skills, or expert simulation procedures. Wrappers perform tool I/O without supplying design decisions, repair policies, or evaluation scheduling. The model may develop its own code and workflow during the design task and inspect the resulting raw diagnostics. An independent harness accounts for budget and measures quality without proposing design actions.

For layout-quality evaluation, each design task has at most six consecutive closed-loop rounds and follows the design-round accounting in Appendix F.2. Failed rounds count toward the limit; model-call retries within the same candidate evaluation add cost without creating extra rounds. Token accounting includes provider-reported input and output over all calls, including failures, retries, reflection, and code revision. Cached input and reasoning output are already included. Runtime includes model inference and all tool execution. Token usage and runtime are averaged over all executed rounds of the layout-quality optimization task.

On LPF, LAMBDA (Sol) and LAMBDA (Astra) achieve comparable layout quality and both attain 100% Pass@1 (Table 4). Compared with the Astra configuration, Sol reduces mean runtime and token usage per round by 72.8% and 61.8%, respectively. These results show that, with LAMBDA, a previous-generation model can deliver design quality close to that of the latest flagship model configuration at lower computational cost.

Tools-only (Astra) achieves 51% Pass@1. Its selected layout has $3843.0\,\mu\text{m}^2$ full layout area, with $2341.8\,\mu$m wirelength, 48.29% placement whitespace, 554 vias, and 254 bends. Its maximum matched net length mismatch is 3.59%. Mean OUT p–p is 4.890903 mV, with $1.499\,\mu$V absolute output swing mismatch and 0.030646% normalized mismatch. Relative to LAMBDA (Astra), area increases by 29.8%, wirelength by 18.1%, and absolute swing mismatch reaches $2.99\times$ the reference, while mean output swing is lower.

Tools-only (Astra) averages 1924.42k input tokens and 14.34k output tokens per round, totaling 1938.76k tokens. Its mean LLM time is 50.15 minutes and total runtime is 53.20 minutes per round. In comparison, LAMBDA (Astra) uses 132.09k input tokens and 12.28k output tokens, with 18.76 minutes of LLM time and 36.74 minutes of total runtime. With the backbone held fixed, these LPF results show that the framework improves reliability, layout compactness, and electrical matching while reducing token usage and runtime.

K EVOLUTION OF CODE, SKILLS, AND LAYOUT EXPERIENCE

All experiments outside this appendix use code evolved using a separate layout library that excludes the nine benchmark circuits. Here, we use LPF as a running example to illustrate how code and skills evolve in response to design feedback.

K.1 REUSABLE ROUTING CAPABILITIES ON FIXED INPUTS

Table 14 compares three reusable routing capabilities on Ring-4, Mixer, and LPF, with placement, routing intent, devices, and connectivity fixed within each circuit. Mirrored-tree routing constructs symmetric routing trees; shared collectors consolidate local pin connections; paired collectors coor-

dinate corresponding connections across matched nets. These examples evaluate how refinements to a common execution capability affect multiple circuit types.

Table 14: Fixed-input routing across three circuits and three implementations.

Circuit	Routing	Route bbox [μm^2]	Whitespace [%]	Wirelength [μm]	Vias	Matched net length mismatch [%]	Equal-via pairs
Ring-4	Mirrored	3526.194	33.86	1084.475	696	1.464	9/9
	Shared	3515.931	33.86	1118.660	461	1.615	9/9
	Paired	3515.931	33.86	1125.170	427	1.623	9/9
Mixer	Mirrored	3715.101	43.34	2098.589	1098	9.575	3/5
	Shared	3730.299	43.34	1947.126	604	1.425	5/5
	Paired	3730.299	43.34	2082.806	619	1.425	5/5
LPF	Mirrored	3260.927	34.88	1922.505	1317	2.394	3/6
	Shared	3256.250	34.88	1788.078	1029	1.900	6/6
	Paired	3246.351	34.88	1877.262	907	1.425	6/6

Mirrored, Shared, and Paired denote mirrored-tree routing, shared collectors, and paired collectors. Route bbox is the wire-envelope bounding box; whitespace uses physical device envelopes. Matched net length mismatch is the maximum over matched nets. Equal-via pairs count matched nets with the same number of vias.

Relative to mirrored-tree routing, paired collectors reduce via counts by 38.65%, 43.62%, and 31.13% on Ring-4, Mixer, and LPF, respectively. Equal-via pairs increase from $15/20$ to $20/20$, and the maximum Mixer matched net length mismatch falls from 9.575% to 1.425%. Ring-4 uses 3.75% more wire, while Mixer and LPF use 0.75% and 2.35% less. Shared collectors achieve shorter routes than paired collectors on all three circuits, illustrating the tradeoff between routed wirelength and via count and matching. A further pin access search modification fails to complete routing on any of the three circuits. This comparison measures routing geometry; paired runtime and post-layout measurements are unavailable. The mirrored-tree Ring-4 layout also fails its oscillation-frequency measurement.

K.2 Code growth and pin access capability

Table 15 measures the placement and routing modules at three representative capability stages. Physical-envelope checks and mirrored routing occupy 4680 lines; adding collectors, passive-device protection, layer-specific via envelopes, bounded pin access, and supply-width handling increases the total to 11,278. Joint selection of paired pin exits adds a further 30 lines. A separate 145-line geometric evaluator measures occupied-area unions, outline fill, and the largest empty rectangle, using device, wire, and port extents to determine the full envelope. It supplies physical feedback in the IF amp, Mixer, and Peak det. refinements.

Table 15: Static size of representative execution and feedback code.

Capability and scope	Modules	Physical lines	Nonblank, noncomment lines	Functions
Physical envelopes and mirrored routes	2	4680	4299	89
Collectors and bounded access	2	11,278	10,409	163
Joint paired exits	2	11,308	10,432	164
Exact floorplan feedback	1	145	127	5

Physical lines include comments, blank lines, and documentation strings. Nonblank, noncomment lines retain documentation strings. Function counts include methods and nested functions.

Paired pin access illustrates how a local code change expands the set of realizable designs. A TSPC-3 candidate cannot be routed when the exits of symmetric pins are chosen independently. Selecting both exits together and checking their complete metal and via geometry enables routing of the same candidate under unchanged physical constraints. The change adds 84 and removes 54 lines, with one additional function. The completed route connects all 129 pins, 23 body contacts, and four ports, with zero geometric contacts between distinct nets and 0.231% matched net length mismatch. The unsuccessful attempt takes 10.442 s and the completed route 54.752 s. This comparison evaluates pin access feasibility and routing completion.

K.3 SKILL FORMATION AND EXPERIENCE REUSE

Table 16 follows three stages of experience accumulation. The memory grows from 13 to 56 distinct records, while semantic and procedural rules increase from three to twelve. Nine rules qualify for reuse after validation. The collection also retains unsuccessful experiences, allowing guidance to include both useful actions and conditions under which those actions fail.

Table 16: Growth of experience memory and reusable rules.

Stage	Records	Semantic/ procedural rules	Qualified rules	Rejected	Rule text characters
Initial experience	13	3	0	0	1519
Cross-circuit reuse	41	10	7	10	4455
Extended validation	56	12	9	13	5211

Rejected counts rejected experiences. Text size includes rule content and applicability conditions.

Conditional orientation guidance. Two 180° rotations of LPF symmetric device pairs both satisfy the physical checks but have opposite routing effects: one reduces wirelength by $5.076\,\mu$m and removes 16 vias, while the other adds $12.400\,\mu$m and four vias. The resulting skill treats orientation as a pin access decision whose value must be established by rerouting. The unfavorable example makes its applicability conditional on the local pin arrangement.

A routing-space operation reused across circuits. A skill for routing-space allocation is learned on LPF and reused on Z amp. It checks that the selected space exists and that the proposed assignment is legal before applying the change. On Z amp, the reused operation reduces wirelength by $7.52\,\mu$m and removes four vias relative to the starting layout. On LPF, retrieval instead trades a $71.894\,\mu$m wirelength increase for 23 fewer vias relative to refinement without retrieval. The Z amp candidate without retrieval also attains a lower routing cost. These outcomes illustrate reuse of an operation with circuit-dependent tradeoffs.

Visual observations and extraction failures become checks. A Z amp layout with 49.7% placement whitespace motivates an absolute compactness rule covering area, utilization, and empty space, which is then incorporated into the physical evaluator. A separate extraction failure caused by M1 crossing a resistor body motivates a layer-dependent access rule: leave the device body before general routing, exclude M1 crossings through that body, and permit crossings on higher metal layers. The corresponding geometric check makes the learned restriction executable on later candidates.

K.4 ONLINE LPF REFINEMENT WITH EVOLVING GUIDANCE

The LPF study uses `gpt-6-astra`. Table 17 reports all six evolution steps from the initial proposal to a verified layoutwithin the $3000\,\mu\text{m}^2$ full layout area target. E3 passes functional verification but exceeds that target; subsequent steps continue refinement, and E6 meets the target. Each of E1–E5 contributes one executable guidance update and one skill update for subsequent proposals. The evolving code converts observed feedback into decision guidance, while the analog layout synthesizer supplies the physical operations. Its size changes from 41 lines initially to 29, 37, 42, 43, and 44 lines. The skills address pin access, preservation of verified connections, geometric clearance, and repeated proposals, with 5, 7, 9, 8, and 9 routing principles after the respective updates. Detailed evaluation checks layout save/reopen, DRC, LVS, extraction, and post-layout function.

Between E3 and E6, layout area falls by 17.8% and placement whitespace drops by 12.07 percentage points. Wirelength increases by 1.8%, while vias decrease from 620 to 574. Both endpoint layouts pass DRC, LVS, extraction, and the functional test. Their common pre-layout mean output swing is 9.76305 mV; post-layout mean swing changes from 5.203421 to 5.210533 mV, corresponding to retention of 53.297% and 53.370%. Absolute output swing mismatch increases from 0.159 to $0.501\,\mu$V, despite matched net length mismatch decreasing from 0.222% to 0.098%. The more compact layout therefore retains functional operation with greater routed wirelength and greater output swing mismatch.

Table 17: LPF refinement with evolving executable guidance and skills.

Step	Outcome	Layout area [μm^2]	Wirelength [μm]	Whitespace [%]	Vias	Detailed evaluation
E1	Proposal fail	-	-	-	-	-
E2	Routing fail	-	-	42.02	-	-
E3	Routed	3602.5	1947.9	46.82	620	Pass
E4	Routed	3057.9	1990.0	36.91	563	-
E5	Routed	3057.9	1990.0	36.91	563	-
E6	Routed	2960.0	1983.5	34.75	574	Pass

Geometry follows the saved placement and routing representation. Pass denotes passing physical verification and the functional test; − denotes an unreached or unscheduled measurement.

The candidates in Table 17 undergo three detailed evaluation attempts, with two successful evaluations covering two distinct layouts. The trajectory measures the combined effects of evolving guidance, skills, and design decisions.