ANAFLASH: AI-DRIVEN SIZING OF ANALOG CIRCUIT USING FAST ANALYTICAL CIRCUIT MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Analog circuit sizing requires jointly satisfying coupled performance specifications whose response to device dimensions depends on the circuit topology, operating point, and technology process. Recent AI-based workflows either require substantial circuit-specific training data or ask large language models (LLMs) to directly propose numerical device updates from simulator feedback while the absense of an analytical solver of the equation of the circuit creates trial and error feedback loop approaches. This paper AnaFlash, a hybrid AI-analytical sizing framework in which a fast circuit model, rather than the LLM, drives numerical exploration. Starting from a physically valid seed, AnaFlash derives a SPICE-based quality-metric sensitivity model and computes sizing updates through constrained optimization. SPICE validates each candidate, while a lightweight LLM is invoked only when the analytical engine detects a locally unproductive sizing update. The LLM then revises metric priorities rather than transistor dimensions. Across six OTA topologies and five GMC filter topologies, three technology processes, and thirty specification configurations per circuit–process pair, AnaFlash achieves training-free transfer with 100% OTA success and $94.7 - 100\%$ filter success. AnaFlash requires only between 17 and 236 SPICE simulations on average for the circuit sizing to converge and achieves up to $56\times$ lower runtime than recent sizing workflows on shared benchmarks.

1 INTRODUCTION

Given the circuit topology, analog circuit sizing selects transistor dimensions, bias conditions, and passive-component values so that the sized circuit matches with the values specified for different quality metrics. The task is difficult because such quality metrics depend non-linearly on the same design parameters, with trade-offs determined by the topology, operating point, and process design kit (PDK). Although effective in selected small settings, current black-box sizing methods, e.g., reinforcement learning and surrogate approaches, require substantial simulation data or training, and are often limited when the circuit topology, PDK, or target-specification distribution changes Xu et al. (2025).

LLM-based agents have recently emerged as a training-free alternative. They interpret a netlist, specifications, and simulator feedback, to directly propose numerical sizing actions Ahmadzadeh et al. (2025a). However, the method at the core of LLM-based solutions transfers the parameter space exploration effort to the LLM itself. While an LLM can recall qualitative circuit-design heuristics, it does not explicitly derive nor solve the operating-point-dependent mathematical representation of the circuit that is necessary to quantify how each device dimension affects every target quality metric. LLM-only flows, therefore, must solely rely on SPICE-level simulations to evaluate numerical guesses, making their efficiency dependent on trial-and-error simulation, prompt design, and large frontier models. Notably, recent benchmarks also show that current language and multimodal models remain unreliable at symbolic and analytical circuit reasoning Akbari et al. (2026).

Our paper addresses this methodological gap with *AnaFlash*, a hybrid AI-analytical framework in which an explicit circuit model drives the exploration of the sizing space. AnaFlash quickly derives an operating-point-dependent analytical model from SPICE that supports the device-level metric sensitivities analysis to propose an analytical driven sizing update methodology. Unlike LLM-only approaches, AnaFlash does not ask the LLM to infer how transistor dimensions should be changed.

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.

Table 1: Comparison of *AnaFlash* with state-of-the-art works targeting analog sizing, performance estimation, and automated circuit understanding.

Method	Approach	No Training Dataset	Analytical Model	Infeasibility Proof	Verified Explainability	Performance Validation
MMM Lin et al. (2024)	ML Macro-Modeling	✗	✗	✗	✗	✗
AnaCraft Ahmadzadeh et al. (2025b)	Duel-Play Prob. RL	✗	✗	✗	✗	✓
ZeroSim Yang et al. (2025)	Neural simulator + RL	✗	✗	✗	✗	✗
Insight Poddar et al. (2024)	Neural simulator + RL	✗	✗	✗	✗	✗
FALCON Mehradfar et al. (2025)	Neural sim. + gradient	✗	✗	✗	✗	✗
DiffCKT Liu et al. (2025)	Diffusion generation	✗	✗	✗	✗	✗
AnaFlow Ahmadzadeh et al. (2025a)	LLM agents	✓	✗	✗	✗	✗
CircuitSense Akbari et al. (2026)	MLLM Benchmark	✓	✗	✗	✓	✗
MOEA Chen et al. (2026)	ML + Evol. Algorithm	✗	✗	✗	✗	✓
AnaFlash	**LLM+Derived Model**	✓	✓	✓	✓	✓

The analytical model determines the numerical update and SPICE simulations are used to validates its physical effect, while the LLM orchestrates the hybrid model preventing local optima. In particular, AnaFlash advances the state of the art with two main contributions:

- *Hybrid sizing methodology* - AnaFlash combines an analytical circuit model with a lightweight LLM. A novel quality-metric sensitivity analysis drives fast sizing convergence, while the LLM redirects the analytical search by revising metric priorities when it reaches a locally unproductive trade-off region.
- *Extensive and realistic experimental campaign* - AnaFlash presents an extensive experimental campaign spanning two circuit families, each of them with at least five circuit topologies, three technology nodes, and thirty specification configurations, demonstrating robust training-free sizing and up to $56\times$ lower runtime than prior recent state-of-the-art workflows.

The rest of this paper is organized in four parts. Section 2 discusses related works. AnaFlash and the related experimental campaign are presented in Section 3 and Section 4, respectively, while conclusions and drawn in Section 5.

2 Related Works

Table 1 compares representative work on analog sizing, performance estimation, and circuit understanding across six dimensions: computational approach, reliance on circuit-specific training data, use of an analytical circuit model, infeasibility analysis, explainability validation, and end-to-end performance evaluation.

Traditional automated sizing relies on black-box optimization. Recent work employs multitask evolutionary algorithms, such as MOEA Chen et al. (2026), to manage competing objectives, or reinforcement learning, such as AnaCraft Ahmadzadeh et al. (2025b), to improve sizing robustness. To reduce the cost of repeated SPICE evaluations, other approaches learn performance surrogates. MMM Lin et al. (2024) uses ML macro-models, whereas INSIGHT Poddar et al. (2024) and ZeroSim Yang et al. (2025) use transformer-based neural simulators for circuit evaluation. FALCON Mehradfar et al. (2025) and DiffCKT Liu et al. (2025) further employ neural and generative models for circuit design. These approaches shift part of the optimization cost to offline data collection and training, often requiring hundreds of thousands to millions of simulated example, e.g., 400K for DiffCKT, 1M for FALCON, and 3.6M for ZeroSim, and may require adaptation when the topology or technology changes. Moreover, their learned models approximate circuit behavior and must be validated empirically for the operating regions considered during sizing.

More recently, LLMs have enabled agentic sizing workflows that avoid circuit-specific training. However, CircuitSense Akbari et al. (2026) shows that state-of-the-art multimodal LLMs remain unreliable at symbolic circuit derivation, achieving below 19% accuracy on its analytical-reasoning tasks. Nevertheless, AnaFlow Ahmadzadeh et al. (2025a) uses LLM agents to propose numerical

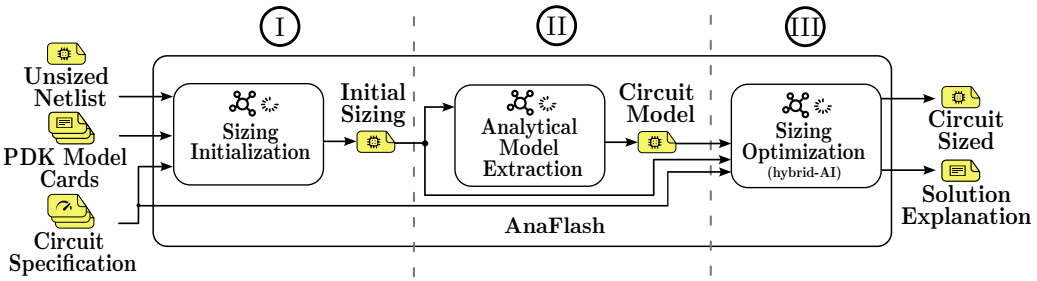


Figure 1: The three main stages of AnaFlash workflow

device sizes from circuit descriptions and simulator feedback. Such a flow relies on the LLM to infer the numerical effect of parameter updates without explicitly constructing the operating-point-dependent circuit model. SPICE must therefore evaluate the proposed updates, and the resulting search efficiency depends on heuristic LLM decisions, prompting, and the capability of the underlying model. Its explanations are likewise derived from model reasoning rather than from an explicit ranking of device-level metric sensitivities.

The gap is therefore not simply the absence of another sizing optimizer, but the absence of a hybrid exploration strategy in which an analytical model directly generates the numerical sizing updates and an LLM is restricted to high-level search adaptation. AnaFlash addresses this gap by deriving a local circuit model from the SPICE operating point and using its quality-metric sensitivities to steer sizing exploration. SPICE validates every proposed update; when the local analytical search no longer yields the predicted improvement, a lightweight LLM revises the relative priorities of the target metrics and redirects the next analytical step. This combination avoids circuit-specific training while retaining explicit, circuit-dependent guidance for numerical exploration.

3 METHODOLOGY

AnaFlash defines a hybrid AI-analytical framework that sizes analog circuits from an unsized netlist, PDK, and target quality metrics. As shown in Figure 1, the flow comprises three stages: *Sizing Initialization*, *Analytical Model Extraction*, and *Hybrid Sizing Optimization*. The first two stages construct a physically valid circuit state and its operating-point-dependent analytical model. The final stage uses the derived analytical model to drive numerical sizing exploration, while a lightweight LLM adapts the relative priority of conflicting metrics only when the analytical engine identifies a locally unproductive search direction.

The rest of this section is organized in three parts. Section 3.1 and Section 3.2 discuss *Sizing Initialization* and *Analytical Model Extraction* stages, respectively, while Section 3.3 presents the *Hybrid Sizing Optimization* stage.

3.1 STAGE 1 - SIZING INITIALIZATION

The *Sizing Initialization* stage generates a initial sizing seed $x_0 \in \mathbb{R}^d$ from the circuit specifications. This seed provides a valid operating point from which the subsequent analytical model can be extracted. AnaFlash parses the unsized netlist into a bipartite graph and identifies functional transistor groups, including differential pairs, current mirrors, cascodes, and bias branches. First-order design relations translate the target specifications into transconductance and current requirements, which are propagated through the graph according to structural multiplicities and mirror ratios to assign a drain current $I_{D,i}$ to each device i.

Given the assigned current, channel length L_i, and inversion coefficient $(g_m/I_D)_i$, the framework queries pre-characterized multi-dimensional Lookup Table (LUT) to obtain the normalized current density $J_{D,i}$. The initial transistor width is then computed as defined in Equation (1).

$$W_i = \frac{I_{D,i}}{J_{D,i}\left(L_i, (g_m/I_D)_i\right)} \tag{1}$$

The initialization enforces symmetry constraints for matched devices and assigns initial bias values from the same LUT characterization. A SPICE operating-point analysis verifies that the seed satisfies the required operating-region margins, including $M_D > 0$, before it is passed to the analytical stage.

3.2 Stage 2 - Analytical Model Extraction

Starting from the validated sizing seed, AnaFlash constructs an operating-point-dependent analytical model of the circuit. A SPICE operating-point analysis provides the small-signal parameters of each active device, including g_m, g_{mb}, g_{ds}, and parasitic capacitances. These parameters are stamped into a sparse Modified Nodal Analysis (MNA) system defined by Equation (2),

$$Y(s)z(s) = b(s), \qquad Y(s) = G + sC, \tag{2}$$

where $z(s)$ is the electrical-state vector, and G and C are the conductive and capacitive admittance matrices, respectively. Notably, $z(s)$ is used to distinguish the electrical state from the sizing seed x_0.

For a circuit response y, the corresponding adjoint state λ is obtained from the dual MNA system. The sensitivity of y to an internal electrical parameter p is defined by Equation (3).

$$\frac{\partial y}{\partial p} = -\lambda^T \left(\frac{\partial G}{\partial p} + s\frac{\partial C}{\partial p} \right) z. \tag{3}$$

Notably, the forward and adjoint solutions are independent of the individual parameter p, thus the engine evaluates the sensitivities of multiple device parameters without exhaustive perturbation sweeps. The engine subsequently propagates electrical sensitivities to physical design knobs. For metric m_i and knob k_j, the resulting quality-metric sensitivity is defined by Equation (4).

$$J_{ij} = \frac{d \ln m_i}{d \ln k_j}. \tag{4}$$

The corresponding chain rule accounts for both the direct device-level variation and the DC-bias shifts induced across the circuit. The resulting design Jacobian $\mathbf{J}$ ranks the influence of every physical knob on every monitored quality metric and directly drives the numerical exploration in the final stage. Before iterative sizing, AnaFlash also evaluates topology- and PDK-dependent reachability bounds. For example, intrinsic-gain and transconductance-efficiency limits provide conservative stage-gain ceilings for the considered topology. When a target exceeds such a bound, the engine reports the limiting physical condition rather than spending the full optimization budget on the associated local search.

3.3 Stage 3 - Hybrid Sizing Optimization

The Hybrid Sizing Optimization stage combines analytical-model-driven numerical exploration with LLM-guided priority adaptation. This division of responsibilities is central to AnaFlash. From the one hand, the analytical model determines *how* circuit parameters should change. From the other hand the LLM determines *which* unresolved trade-off should receive greater priority after the current local search direction has failed. LLM outputs are schema-validated and restricted to the weight interface. In particular, they cannot directly alter device dimensions, PDK constraints, or SPICE results.

AnaFlash defines $\mathbf{k}$ as the vector of design knobs, including transistor geometries, bias voltages, and passive-component values, and $\mathbf{u} = \ln \mathbf{k}$ denotes the corresponding log-domain variables. At a circuit state $\mathbf{u}$, the SPICE simulation returns a vector of quality metrics $\mathbf{m}(\mathbf{u}) = [m_1(\mathbf{u}), \ldots, m_q(\mathbf{u})]^T$. Each metric is mapped to a normalized violation v_i, where $v_i > 0$ indicates that the corresponding target is unmet and $v_i \leq 0$ that it is satisfied. The analytical engine uses a priority vector $\mathbf{w}$ to express the relative importance of the violations during sizing optimizations.

At iteration t, the analytical engine uses the Jacobian extracted at the current state $\mathbf{u}_t$ to predict the response to a candidate update $\Delta\mathbf{u}$. Since $\mathbf{J}$ is defined in logarithmic coordinates, the local model is defined according to Equation (5)

$$\ln \widehat{m}_i(\mathbf{u}_t + \Delta\mathbf{u}) = \ln m_i(\mathbf{u}_t) + \mathbf{J}_i \Delta\mathbf{u}, \tag{5}$$

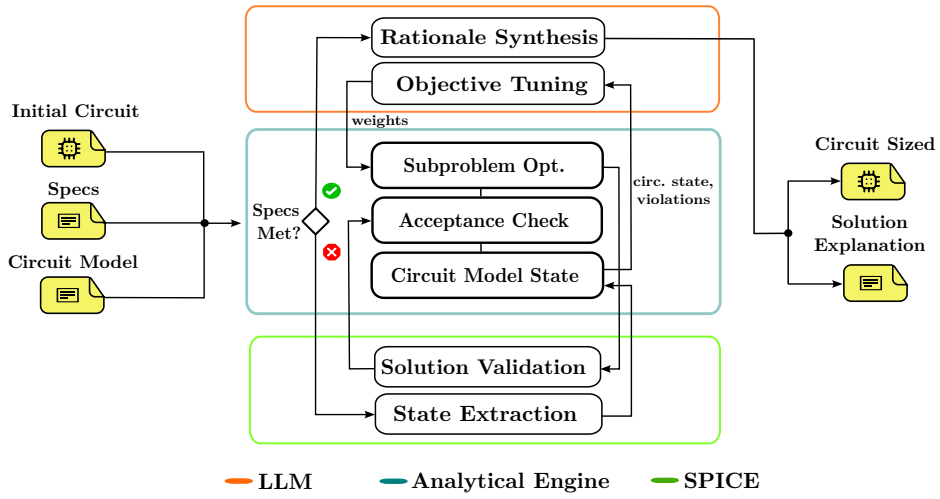


Figure 2: Overview of the Hybrid Sizing Optimization stage. The Analytical Engine computes numerical parameter updates (sizing), which are then validated by SPICE. If the numerical search stalls, the LLM adjusts priority weights to resolve competing trade-offs. Once all specifications are met, the LLM synthesizes a human-readable explanation of the final design.

where $\mathbf{J}_i$ is the i-th row of the design Jacobian. The predicted metrics are converted into normalized predicted violations $\widehat{v}_i$.

The analytical engine computes the candidate sizing update by solving a weighted Chebyshev linear program,

$$\begin{aligned} \min_{\Delta \mathbf{u}} \quad & \max_i \left(w_i \widehat{v}_i\right) \\ \text{s.t.} \quad & |\Delta u_j| \le r_j, \quad \forall j \end{aligned} \tag{6}$$

where r_j is the trust-region radius for knob j. Notably, the optimization problem assumes extra constraints in terms of the linearized saturation margins and the feasibility limits encoded into the PDK. The solution minimizes the largest weighted predicted violation and thus provides a circuit-specific numerical search direction. Once all specifications are satisfied, the same formulation can optimize a secondary objective, such as area or static power, while retaining the satisfied specifications as hard constraints.

The candidate $\mathbf{u}_t + \Delta\mathbf{u}$ is evaluated in SPICE. SPICE acts solely as the physical oracle by returning the measured quality metrics, while the analytical engine decides how to continue the search. In particular, the engine compares the observed reduction in the weighted violation objective as defined by Equation (8).

$$V_{\mathbf{w}}(\mathbf{u}) = \max_i \left(w_i v_i(\mathbf{u})\right) \tag{7}$$

with the reduction predicted by the local model through the trust-region gain ratio

$$\rho = \frac{V_{\mathbf{w}}(\mathbf{u}_t) - V_{\mathbf{w}}(\mathbf{u}_t + \Delta\mathbf{u})}{V_{\mathbf{w}}(\mathbf{u}_t) - \widehat{V}_{\mathbf{w}}(\mathbf{u}_t + \Delta\mathbf{u})}. \tag{8}$$

When the predicted and observed improvements agree sufficiently, the engine accepts the update, refreshes the circuit model at the new operating point, and adapts the trust-region radius. Otherwise, it performs directional backtracking by evaluating smaller fractions of the same analytically derived direction.

If neither the candidate nor its backtracked variants yield a productive reduction in $V_{\mathbf{w}}$, the analytical engine diagnoses the current metric prioritization as locally unproductive. Only at this point does it invoke the LLM. The LLM receives a structured summary of the current violations, active weights, predicted and observed metric changes, and device-level sensitivity rankings. It returns bounded

revised weights $\mathbf{w}$, thereby changing the relative emphasis assigned to competing metrics in (6). The subsequent numerical update is again generated solely by the analytical engine.

After the final sizing is obtained, AnaFlash re-evaluates the circuit and provides the LLM with the final netlist, sizing values, specification margins, and device-level sensitivity rankings. The LLM uses this evidence to generate a human-readable explanation of the final design and its dominant trade-offs.

4 Experimental Results

This section discusses the experimental campaign to assess the efficiency and robustness of AnaFlash considering different circuit classes, technologies, and specification set. AnaFlash is also compared with state of the art solutions. The rest of this section is organized in four parts. Section 4.1 presents the experimental setup, Section 4.2 discusses AnaFlash results across a large variety of examples while Section 4.3 and Section 4.4 presents the comparison with state-of-the-art, and the ablation results, respectively.

4.1 Experimental setup

AnaFlash is evaluated on two analog circuit families with distinct sizing objectives: six operational transconductance amplifier (OTA) topologies and five continuous-time G_m–C filter topologies. The OTA set includes five-transistor, class-AB five-transistor, current-mirror, two-stage Miller, telescopic-cascode, and fully differential folded-cascode amplifiers. The filter set includes low-pass, band-pass, and high-pass biquads, together with third- and fourth-order low-pass filters.

Each circuit is evaluated in three technology processes, i.e., PTM 45 nm, SkyWater 130 nm, and GF 180 nm. To ensure statistic significance, for every circuit-PDK pair, the evaluation considered thirty fixed specification configurations. These configurations span simultaneous and potentially conflicting quality requirements rather than isolated single-metric targets. The primary evaluation therefore comprises

$$30 \times (6 + 5) \times 3 = 990$$

sizing tasks. No circuit-specific training, fine-tuning, or PDK-specific retraining is performed.

Table 2 defines the circuit quality metrics and the target ranges explored by the thirty configurations. OTA tasks require lower bounds on gain, unity-gain bandwidth (GBW), phase margin (PM), and slew rate, as well as an upper bound on input-referred noise. Filter tasks require lower bounds on corner frequency, stopband rejection, passband gain, and quality factor, as well as upper bounds on output noise and power. A run is counted as *met* only when the final SPICE evaluation satisfies all constraints of the corresponding configuration simultaneously.

The *specification success rate* is defined as the fraction of configurations that meet all requested circuit metrics. The *SPICE simulations* defines the number of simulator calls from initialization to final verification. The *wall-clock time* measures the end-to-end optimization latency. Notably, SPICE simulations are used as primary efficiency quality metric because they are independent of the host platform and directly measure the quality metrics of the circuit. Detailed hardware, software, and circuit specifications are reported in the Appendix A.

4.2 Training-Free Quality Results Across Circuits and Technologies

Tables 3 and 4 report the results for every circuit-PDK pair. Notably, reported results are the averaged value over the 30 considered configurations for every circuit-PDK pair. AnaFlash meets all thirty configurations for every OTA topology across all three processes, yielding 540/540 successful OTA tasks. This result extends to the filter family almost verbatim. in particular, all SkyWater and GF tasks are met, while the PTM set achieves 94.7% mean success. The only unsuccessful cases occur for the most constrained fourth-order low-pass filter in PTM technology.

Notably, the same analytical engine and lightweight LLM are used across all circuit families and technologies. The analytical engine computes numerical sizing updates from SPICE-derived local sensitivities. the LLM changes only the relative metric priorities after the engine identifies a locally

Table 2: **Benchmark definition and specification configurations.** Thirty fixed configurations are evaluated for every circuit–process pair. Arrows indicate lower ($\geq$) and upper ($\leq$) bounds. Ranges report the minimum and maximum target values across the thirty configurations.

Family	Quality metric	Constraint	Target range	SPICE quantity
OTA	DC gain	$\geq$	20.09–34.15 dB	Low-frequency voltage gain
	GBW	$\geq$	14.85–30.65 MHz	Unity-gain bandwidth
	Phase margin	$\geq$	45.02–57.52°	Closed-loop stability margin
	Slew rate	$\geq$	13.05–26.36 MV/s	Large-signal transient response
	Input-referred noise	$\leq$	75.9–166.8 μV_{rms}	Integrated input-referred noise
GMC filters	Corner frequency	$\geq$	4.15–4.59 MHz	Characteristic frequency
	Stopband rejection	$\geq$	15.39–17.05 dB	Stopband attenuation
	Passband gain	$\geq$	-4.61–-3.63 dB	In-band transfer magnitude
	Quality factor	$\geq$	0.641–0.685	Selectivity / peaking factor
	Output noise	$\leq$	40.0–43.9 μV_{rms}	Integrated output noise
	Power	$\leq$	872–955 μW	DC power consumption

Table 3: **OTA transfer results across circuit topologies and processes.** *Met* is the number of configurations satisfying all five OTA specifications, out of thirty. *Sims* and *Time* are means over the thirty configurations.

	PTM 45 nm			SkyWater 130 nm			GF 180 nm		
OTA topology	Met	Sims	Time	Met	Sims	Time	Met	Sims	Time
`ota_5t`	30/30	17.5	20 s	30/30	18.7	31 s	30/30	28.4	41 s
`ota_5t_classab`	30/30	15.4	19 s	30/30	18.5	41 s	30/30	23.0	44 s
`ota_cm`	30/30	15.0	22 s	30/30	39.7	86 s	30/30	21.4	59 s
`ota_twostage`	30/30	17.0	23 s	30/30	30.2	62 s	30/30	48.4	75 s
`ota_telescopic`	30/30	16.8	24 s	30/30	23.6	34 s	30/30	33.8	34 s
`ota_folded_fd`	30/30	21.5	85 s	30/30	34.0	130 s	30/30	29.9	122 s
Mean	**100%**	**17.2**	**32 s**	**100%**	**27.5**	**64 s**	**100%**	**30.8**	**63 s**

unproductive trade-off. Thus, the method transfers without retraining across circuit structures, metric sets, and technology processes.

AnaFlash requires 17.2-30.8 SPICE simulations on average for OTA sizing and 22.1-35.9 for filter sizing. The corresponding end-to-end time is between 32 s and 64 s for OTAs and 56 s to 1.9 min for filters. The longer runtime of selected PTM filter tasks is largely due to simulator cost and circuit complexity. Even the 44-device fourth-order low-pass filter completes in at most 3.7 min on average.

4.3 Comparison with Prior Sizing Workflows

This section compares AnaFlash with AnaCraft Ahmadzadeh et al. (2025b) and AnaFlow Ahmadzadeh et al. (2025a) on the two-stage Miller and fully differential folded-cascode OTAs considered by such state-of-the-art works. Table 5 reports published results alongside an agent-only reimplementation of AnaFlow using open-weight Qwen2.5 models.

AnaFlash reaches feasible designs in 21 SPICE simulations for the two-stage OTA and 45 simulations for the folded-cascode OTA. This is substantially lower than the 1,035-1,808 simulations reported by AnaCraft and the 205-360 simulations reported by AnaFlow. The resulting runtimes are 17 s and 1.8 min, respectively, corresponding to up to $56\times$ lower runtime than the published workflows. Notably, the agent-only AnaFlow fail to meet the specifications within a 400-SPICE simulation budget for every circuit thus resorting to the Baysian optimization safepath. This does not imply that LLMs cannot assist analog design, while it shows that language-model reasoning alone is not a reliable numerical sizing mechanism for these coupled trade-offs. AnaFlash instead assigns device-level updates to the analytical engine and reserves the LLM for high-level priority adaptation.

Table 4: G_m–C **filter transfer results across topologies and processes.** *Met* is the number of configurations satisfying all six filter specifications, out of thirty. *Sims* and *Time* are means over the thirty configurations.

	PTM 45 nm			SkyWater 130 nm			GF 180 nm		
Filter topology	Met	Sims	Time	Met	Sims	Time	Met	Sims	Time
`gmc_biquad_lp`	30/30	30.1	1.9 min	30/30	37.0	63 s	30/30	27.6	1.7 min
`gmc_biquad_bp`	30/30	25.6	34 s	30/30	21.5	16 s	30/30	17.1	49 s
`gmc_biquad_hp`	30/30	16.9	59 s	30/30	25.6	81 s	30/30	25.9	67 s
`gmc_lp3`	30/30	37.7	2.2 min	30/30	10.0	10 s	30/30	42.4	2.0 min
`gmc_lp4`	22/30	43.9	3.7 min	30/30	16.5	1.9 min	30/30	66.5	2.7 min
Mean	**94.7%**	**30.8**	**1.9 min**	**100%**	**22.1**	**56 s**	**100%**	**35.9**	**1.7 min**

Table 5: **Comparison with prior analog sizing workflows on the AnaFlow benchmarks.** Published values are reproduced from the respective papers. Agent-only AnaFlow uses the AnaFlow interaction protocol without AnaFlash's analytical sizing engine. "–" denotes that no specification-satisfying design was found within the 400-simulation budget.

Workflow	**Model**	**Two-stage Miller OTA**		**Folded-cascode OTA**	
		Sims	Time	Sims	Time
AnaCraft Ahmadzadeh et al. (2025b)	RL	1,035	14 min	1,808	31 min
AnaFlow Ahmadzadeh et al. (2025a)	Gemini-2.5-Flash	282	16 min	318	17 min
AnaFlow Ahmadzadeh et al. (2025a)	GPT-o3	205	46 min	360	82 min
AnaFlow agent-only	Qwen2.5-14B	>400	8 min	>400	26 min
AnaFlow agent-only	Qwen2.5-32B	>400	12 min	>400	38 min
AnaFlash	**Qwen2.5-7B**	**21**	**17 s**	**45**	**1.8 min**

4.4 Ablation of Hybrid Priority Adaptation

This section explores the contribution of LLM-based priority adaptation on the 540 OTA tasks. Table 6 compares analytical sizing without LLM adaptation, analytical sizing with random priority updates, and the proposed Qwen2.5-7B policy. The analytical engine alone meets specifications in 97.0% of tasks. In contrast, random reprioritization increases this to 98.5% while the use of AnaFlash equipped with a Qwen2.5-7B-based LLM support allows to reach 540/540 successful tasks. This ablation clarifies the division of responsibilities in AnaFlash. The analytical model supplies the main sizing capability and ensures that the numerical exploration remains circuit-aware. The LLM improves robustness by revising the relative importance of the unresolved metrics only when the analytical engine identifies that the current priorities lead to an unproductive local trade-off.

To quantify design quality after feasibility is achieved, Table 7 compares Qwen2.5-7B against the median outcome of three independent random-priority runs for each task. The 7B controller reduces mean power relative to the random baseline in PTM and GF, achieving $0.828\times$ and $0.885\times$ of the baseline power, respectively. It also reduces area in most GF tasks.

Finally, Table 8 evaluates whether a larger LLM improves the hybrid controller. Results demonstrate that increasing the model from 7B to 14B or 32B parameters does not improve success rate nor reduce the number of SPICE simulations for the sizing to converge. All in all, the 7B model therefore offers the best accuracy-cost trade-off. Indeed circuit-aware numerical exploration remains in the analytical engine, allowing a lightweight LLM to suffice for search redirection.

5 Conclusions

This paper presented AnaFlash, a hybrid AI-analytical framework for analog circuit sizing. Unlike LLM-based sizing flows that directly rely on language models to propose device dimensions from simulator feedback, AnaFlash uses a SPICE-based analytical sensitivity model to drive the numerical exploration of the sizing space. The analytical engine computes and validates device-level updates,

Table 6: **Success-rate ablation of priority adaptation on OTA tasks.** Each process contains 180 tasks. Random adaptation aggregates three independent random-priority runs per task.

Priority policy	PTM 45 nm	SkyWater 130 nm	GF 180 nm	Total
No LLM adaptation	174/180	178/180	172/180	524/540 (97.0%)
Random adaptation	537/540	534/540	526/540	1,597/1,620 (98.5%)
Qwen2.5-7B	**180/180**	**180/180**	**180/180**	**540/540 (100%)**

Table 7: **Final design quality of Qwen2.5-7B relative to random priority adaptation.** The random baseline is the median of three runs per task. A ratio below one is favorable; win rate is the fraction of tasks with strictly lower cost.

Process	Power ratio	Power win rate	Area ratio	Area win rate
PTM 45 nm	0.828×	81%	0.967×	53%
SkyWater 130 nm	1.000×	50%	0.969×	53%
GF 180 nm	0.885×	69%	0.796×	67%

Table 8: **Effect of LLM scale on the 540 OTA sizing tasks.** Calls and tokens are averages over all tasks.

LLM	Met	Sims	Calls/run	Tokens/run
Qwen2.5-7B	**540/540**	25.1	4.7	3.13 k
Qwen2.5-14B	534/540	26.8	3.4	2.26 k
Qwen2.5-32B	532/540	**24.9**	3.4	2.30 k

whereas the lightweight LLM is restricted to revising the priorities of unresolved quality metrics when the current analytical search direction becomes locally unproductive. This separation combines circuit-dependent numerical guidance with adaptive handling of conflicting design objectives.

The experimental campaign demonstrates training-free transfer across six OTA topologies, five GMC filter topologies, three technology processes, and thirty specification configurations per circuit–process pair. AnaFlash achieves 100% success on all OTA tasks and $94.7 - 100\%$ success on filter tasks, while requiring a limited number of SPICE evaluations to converge. On shared benchmarks, it achieves up to $56\times$ lower runtime than recent analog sizing workflows. These results show that lightweight LLMs can effectively support analog design when their role is combined with, rather than substituted for, an explicit analytical circuit model.

6 AI USE STATEMENT

We used gnerative AI tools during this work as general-purpose assistants. They helped revise the text and grammar of the manuscript, and helped identify and fix bugs in the code. We have not used generative AI tools to gather, process, or analyze data, to develop the methodology, to conduct experiments, or to generate research claims. All AI-assisted text and suggestions were reviewed, verified, and revised by the authors for technical accuracy, fidelity to the implemented methodology, and consistency with the experimental results. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

7 ETHICS STATEMENT

This work does not involve human subjects or personal data. ThunderGen targets analog integrated-circuit design automation. The methodology is intended to reduce manual design effort and does not make decisions affecting individuals or protected groups. We report the experimental setup and evaluation criteria to support reproducibility and responsible interpretation of the results. The main contributions of the paper clearly stated both in the Abstract and the Introduction (Section 1) and study conducted in Related Work (Section 2) demonstrate how our work can foster the research toward new frontiers of analog circuit design and optimization.

8 REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our framework, we provide comprehensive descriptions of all algorithmic steps, analytical models, and mathematical formulations within the Methodology 3 section and Appendix A. The detailed equations and workflow definitions are designed to allow independent reimplementation. Furthermore, to facilitate the replication of our experimental setup, we deliberately utilize publicly available, open-weight language models (the Qwen2.5 family) and conduct evaluations on openly accessible technology nodes (e.g., SkyWater 130 nm).

REFERENCES

Mohsen Ahmadzadeh, Kaichang Chen, and Georges Gielen. Anaflow: Agentic llm-based workflow for reasoning-driven explainable and sample-efficient analog circuit sizing. In *2025 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, pp. 1–7. IEEE, October 2025a. doi: 10.1109/iccad66269.2025.11240818. URL http://dx.doi.org/10.1109/ICCAD66269.2025.11240818.

Mohsen Ahmadzadeh, Jan Lappas, Norbert Wehn, and Georges Gielen. Anacraft: Duel-play probabilistic-model-based reinforcement learning for sample-efficient pvt-robust analog circuit sizing optimization. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 45(2):901–914, 2025b.

Arman Akbari, Jian Gao, Yifei Zou, Mei Yang, Jinru Duan, Dmitrii Torbunov, Yanzhi Wang, Yihui Ren, and Xuan Zhang. Circuitsense: A hierarchical mllm benchmark bridging visual comprehension and symbolic reasoning in engineering design process, 2026. URL https://arxiv.org/abs/2509.22339.

Minyang Chen, Yijia Hao, Miguel Gandara, Muhammad Imran, and Bo Liu. Trade-off-aware analog circuit sizing based on a multitask surrogate model-assisted evolutionary algorithm. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2026.

Yishuang Lin, Yaguang Li, Meghna Madhusudan, Sachin S. Sapatnekar, Ramesh Harjani, and Jiang Hu. Mmm: Machine learning-based macro-modeling for linear analog ics and adc/dacs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(12):4740–4752, 2024. doi: 10.1109/TCAD.2024.3416894.

Chengjie Liu, Jiajia Li, Yabing Feng, Wenhao Huang, Weiyu Chen, Yuan Du, Jun Yang, and Li Du. Diffckt: A diffusion model-based hybrid neural network framework for automatic transistor-level generation of analog circuits, 2025. URL https://arxiv.org/abs/2507.00444.

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, 2025. URL https://arxiv.org/abs/2505.21923.

Souradip Poddar, Youngmin Oh, Yao Lai, Hanqing Zhu, Bosun Hwang, and David Z. Pan. Insight: Universal neural simulator for analog circuits harnessing autoregressive transformers, 2024. URL https://arxiv.org/abs/2407.07346.

Kangwei Xu, Denis Schwachhofer, Jason Blocklove, Ilia Polian, Peter Domanski, Dirk Pflüger, Siddharth Garg, Ramesh Karri, Ozgur Sinanoglu, Johann Knechtel, et al. Large language models (llms) for electronic design automation (eda). *arXiv preprint arXiv:2508.20030*, 2025.

Xiaomeng Yang, Jian Gao, Yanzhi Wang, and Xuan Zhang. Zerosim: Zero-shot analog circuit evaluation with unified transformer embeddings, 2025. URL https://arxiv.org/abs/2511.07658.

A APPENDIX

This appendix summarizes the software libraries, circuit simulator, device models, and language-model configuration used throughout the AnaFlash experimental campaign as described in table 9. The complete pipeline was implemented in a unified Python environment, with circuit evaluation and validation performed through external ngspice processes. Language-model inference was executed locally rather than through a hosted API.

A.1 CIRCUIT METRICS

To rigorously evaluate the scalability and versatility of the proposed framework, we construct a comprehensive benchmark suite spanning two distinct analog circuit paradigms: Operational Transconductance Amplifiers (OTAs) and G_m-C continuous-time filters. Table 10 outlines the structural complexity of each test topology, detailing primitive device counts (NMOS, PMOS,

Table 9: Hardware, software, simulator, and model configuration used in the AnaFlash experiments.

Category	Version or configuration
Operating system	Ubuntu 24.04.5 LTS
CPU	Intel Xeon E5-2667 v4, 32 logical CPUs
System memory	125 GB RAM
GPUs	4 × NVIDIA Tesla V100-SXM2, 16 GB each
Python	3.12.3
Circuit simulator	ngspice 42
NumPy	2.5.3
SciPy	1.18.1
SymPy	1.13.1
NetworkX	installed Python package
h5py	installed Python package
PyTorch	2.6.0+cu124
CUDA runtime	CUDA 12.4 build
Transformers	Hugging Face Transformers
Accelerate	Hugging Face Accelerate
SentencePiece	SentencePiece tokenizer support
BitsAndBytes	8-bit/quantized model-loading support
LLM	Qwen/Qwen2.5-7B-Instruct
LLM execution	Local inference; fixed seed; temperature 0
Device models	PTM45, SkyWater 130 nm, gf180x

passive components) alongside the resulting design search space dimensionality, represented by the total derived design knobs. Furthermore, Table 11 summarizes the key performance metrics and specifications evaluated across both circuit families, ranging from fundamental small-signal specs (such as DC gain and gain-bandwidth product) to dynamic and filter-specific characteristics (such as slew rate, stopband rejection, and quality factor).

Table 10: Device counts and complexity metrics for the tested OTA and GMC filter circuits.

Circuit	Devices	NMOS	PMOS	Resistors	Capacitors	Derived knobs
Operational Transconductance Amplifiers (OTAs)						
`ota_5t`	5	3	2	0	0	7
`ota_5t_classab`	9	4	3	1	1	11
`ota_current_mirror`	9	5	4	0	0	15
`ota_two_stage`	9	4	3	1	1	11
`ota_telescopic`	9	5	4	0	0	15
`ota_folded_fd`	28	13	9	4	2	36
G_m-C (GMC) Continuous-Time Filters						
`gmc_biquad_lp`	22	12	8	0	2	35
`gmc_biquad_bp`	22	12	8	0	2	35
`gmc_biquad_hp`	18	9	6	0	3	28
`gmc_lp3`	33	18	12	0	3	53
`gmc_lp4`	44	24	16	0	4	71

Table 11: **Circuit specification metrics.** A summary of the performance targets evaluated for the OTA and GMC filter families.

Metric	Simplified Description
Operational Transconductance Amplifiers (OTAs)	
Gain (`A0_db`, dB)	How much the circuit amplifies a steady (DC) input signal.
Gain-bandwidth (`gbw`, Hz)	The maximum frequency where the circuit can still amplify a signal (gain > 1).
Phase margin (`pm`, deg)	A measure of stability; indicates how much phase shift can occur before the circuit starts oscillating or ringing.
Slew rate (`slew`, V/s)	The maximum speed at which the output voltage can physically change (rise or fall).
Input noise (`noise`, Vrms)	The total unwanted electrical noise added by the circuit, measured at the input.
G_m-C (GMC) Continuous-Time Filters	
Corner freq. (`cutoff_hz`, Hz)	The specific frequency where the filter begins to significantly block or pass the signal.
Stopband rej. (`stopband_rej_db`, dB)	How effectively the filter suppresses unwanted frequencies outside of its operating band.
Passband gain (`passband_gain_db`, dB)	The amount of amplification (or attenuation) applied to the signals the filter is designed to let through.
Q-factor (`q`, —)	The sharpness or resonance of the filter's frequency response near its corner frequency.
Output noise (`onoise`, Vrms)	The total unwanted electrical noise measured at the filter's output.
Power (`power`, W)	The total steady-state electrical power consumed by the circuit from the supply.