TRACE: LLM-ORCHESTRATED BAYESIAN OPTIMIZATION FOR ANALOG CIRCUIT SIZING

Anonymous authors
Paper under double-blind review

ABSTRACT

An analog circuit sizing task must reconcile competing specifications across testbenches and process, voltage, and temperature corners through costly circuit simulations. Bayesian optimization (BO) reuses these evaluations to guide sampling, but a stalled loss curve alone does not identify the circuit-level cause or the intervention it calls for. We introduce TRACE, an LLM-orchestrated framework that composes the sizing workflow online as simulation evidence accumulates. At each decision point, TRACE refreshes six diagnostic views of the committed results, reads them alongside device roles and circuit topology, and decides *whether, when, and at what scope* to act: continue BO, refine a local candidate, explore an unsampled region, or revise the sizing task. Each decision stays bound to the evidence available when it was made, so an episode can be replayed and its design checked against simulator records. On four circuits in a TSMC 28 nm process with Cadence Spectre, TRACE returns a fully verified design in every run and achieves the lowest mean simulation count among the compared methods. Relative to TRACE, tSS-BO requires 2.09–$4.96\times$ as many simulations and 1.62–$3.96\times$ as much wall time. Action ablations show that neither single intervention is uniformly best across the tested circuits.

1 INTRODUCTION

Analog circuit sizing selects device parameters for a fixed topology so that a design meets every specification across testbenches and process, voltage, and temperature (PVT) corners, and each candidate is evaluated by circuit simulation. Bayesian optimization (BO) reuses these costly evaluations to guide further sampling (Lyu et al., 2018; Touloupas et al., 2021). Here, BO sees the circuit through an aggregate loss. The loss can indicate stalled progress without identifying its cause. A parameter bound that is too tight, a trade-off between specifications, and an error in the task definition all flatten the same curve, and each calls for a different response. A designer separates them using device roles and the simulation record, and a large language model (LLM) agent can reason over the same context (Figure 1).

Recent methods attach LLM reasoning to specific stages of the search. ADO-LLM proposes candidates alongside BO within a fixed search space (Yin et al., 2024). LEDRO and ASTRA refine parameter ranges or focus the search on influential variables (Kochar et al., 2025; Xing et al., 2025), AnaFlow and TopoSizing interpret results to guide sampling and parameter changes (Ahmadzadeh et al., 2025; Wei et al., 2025), and AutoSizer revises the search space and coordinates optimizers in an outer loop (Yu et al., 2026). These studies demonstrate several ways for LLM reasoning to guide numerical search. The benefit of an intervention, however, depends on when it is applied and what it costs. An action that helps resolve one stall may offer little benefit at another. We therefore ask how an agent should choose the timing and type of intervention while accounting for its cost within a single sizing episode. Making these decisions requires a compact view of the evolving search state. Preserving the evidence available at each decision keeps its basis traceable as new results arrive.

We introduce **TRACE**, *LLM-Orchestrated Bayesian Optimization for Analog Circuit Sizing*, in which an LLM decides whether, when, and at what scope to intervene during a sizing episode. The LLM first configures the sizing task from circuit knowledge and authors an initial design to warm-start BO. It then runs a decision loop alongside BO, reading six diagnostic views of the shared simulation history to select one of four operations, which continue BO, refine a local candidate,

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.

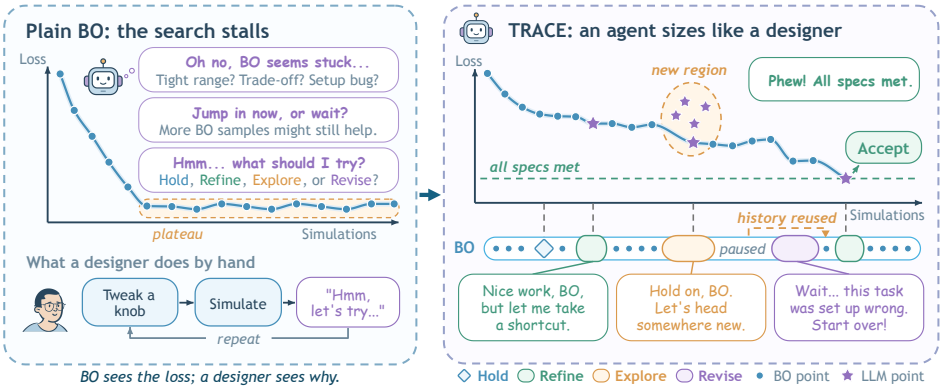

Figure 1: Conceptual illustration of TRACE (curves are illustrative). *Left:* a stalled loss curve shows that BO is stuck but not why. *Right:* like a designer, an LLM agent reads the simulation evidence and decides when and how to intervene, while BO keeps sampling from the shared history.

explore an unsampled region, or revise the task configuration. The LLM specifies each operation's circuit-specific arguments, and each decision is recorded with the evidence available at the time. The executed workflow therefore takes shape within the episode instead of being fixed in advance.

We evaluate TRACE on four TSMC 28 nm circuits using Cadence Spectre, with five runs per method and circuit. Among the compared methods, including adapted ADO-LLM and LEDRO, TRACE has the lowest mean simulation count on all four circuits and the lowest mean wall time on three. tSS-BO (Gu et al., 2024) requires 2.09–$4.96\times$ as many simulations and 1.62–$3.96\times$ as much wall time as TRACE. Ablations show no single intervention is uniformly best across circuits, while diagnostic views reduce decision overhead. Task revision recovers verified designs in all four restricted configurations, while BO alone finds none within the tested duration.

Our contributions are as follows.

1. **A closed-loop, LLM-orchestrated sizing system.** TRACE couples an LLM agent with a BO backend in an autonomous closed loop that intervenes at whatever scope and moment the search calls for, from a single candidate to an unsampled region to the task definition itself, and holds back when the evidence does not yet warrant action.
2. **Connecting search and agent through diagnostic views.** Optimizer statistics are distilled into bounded evidence an LLM can cite, so each decision receives a fixed-size diagnostic summary of the accumulated search history, without reconstructing it from raw records however long the episode runs. The interface could also support other LLM-overseen, simulation-driven searches.
3. **Traceable records of long-horizon orchestration.** Long, asynchronous orchestration rests on an infrastructure that pairs every decision with the evidence it saw and attributes delayed results to the operations that requested them, so an episode can be replayed decision by decision without new simulations and its design verified against simulator records.

2 Related Work

LLM-guided analog circuit sizing. ADO-LLM pairs GP-based BO with LLM proposals conditioned on selected designs from their shared history (Yin et al., 2024). LEDRO alternates LLM-proposed ranges with numerical optimization and feedback (Kochar et al., 2025). AutoSizer combines algorithm selection in an inner loop with search-space revision in an outer loop (Yu et al., 2026). TRACE retains one BO backend and adaptively chooses intervention timing and scope: no intervention, a single local candidate, regional exploration, or, when warranted, task revision.

Online workflow selection. ADAS searches agent programs, and AFlow searches code-represented workflows using execution feedback (Hu et al., 2025; Zhang et al., 2025). Dynamic algorithm configuration adapts algorithm parameters during execution (Biedenkapp et al., 2020), while BO portfolios select acquisition rules (Hoffman et al., 2011). TRACE selects among circuit-sizing operations with different scopes and costs, including regional probes and task revision. Revising the task configuration goes beyond choosing an acquisition rule.

Evidence and memory. AnalogAgent distills execution feedback into an evolving memory and retrieves guidance for subsequent design attempts (Bao et al., 2026). Experience reuse and evidence summarization serve different purposes here. Memory retains lessons from past actions, a diagnostic interface organizes observations into a current search state, and the agent selects operations from that state. In TRACE, these summaries are computed deterministically from committed simulation results, providing a consistent view of the search history at each decision.

Long-horizon and asynchronous agents. ReAct interleaves reasoning and tool use (Yao et al., 2023), while Reflexion retains reflective feedback for subsequent trials (Shinn et al., 2023). Long-task evaluations examine sustained execution (Kwa et al., 2025). Asynchronous BO addresses parallel evaluations with different completion times (Kandasamy et al., 2018), and delayed-feedback learning studies decisions whose outcomes arrive later (Joulani et al., 2013). In TRACE, BO keeps committing results while the LLM deliberates, and the task configuration itself can change mid-episode. TRACE therefore associates each result with the operation that requested it and with the revision under which it was evaluated, which keeps a long episode reconstructable step by step.

3 METHOD

3.1 PROBLEM SETTING

Sizing seeks device parameters $\mathbf{x}$ for a fixed topology within a PDK-defined domain $\mathcal{X}$ that satisfy all declared constraints,

$$\text{find} \quad \mathbf{x} \in \mathcal{X} \quad \text{s.t.} \quad m_i(\mathbf{x}) \geq 0, \quad i = 1, \ldots, N_c,$$

where each margin m_i is normalized to a fixed scale. BO uses the simulator-reported scalar loss to guide the search, which terminates when a feasible design passes final verification. A design is accepted only when simulator records satisfy every declared bound and device operating region on every testbench and PVT corner, so failed or unexecuted analyses leave requirements unverified. Each candidate is simulated in phases, from nominal conditions to progressively harsher PVT corners, and evaluation stops at the first phase that fails its screening criterion. A rejected candidate therefore costs less than a full evaluation, and candidate count is not proportional to simulation costs.

A *task* is the complete configuration under which sizing runs, and it has two parts. The *working surface* is the part BO searches. It holds the independent physical parameters of each transistor, such as width, length, and finger count, together with their ranges and the matching and ratio constraints among them. The remaining part fixes the operating region required of every device and the evaluation and optimizer settings. The working surface is drawn from a *reference surface* that holds the parameter bounds declared at configuration, and every configuration of the task is a *task revision*. A *candidate* is one complete assignment of the working-surface parameters submitted for simulation, and its result is *committed* once it is read back into the shared history of BO and LLM evaluations.

3.2 OVERVIEW

TRACE runs the LLM as a decision loop concurrent with BO (Figure 2; Algorithm 1). Because BO may commit many simulations while the LLM deliberates once, the two loops advance at different rates, and each decision begins from every result that BO and the LLM have committed by that moment. The pace of decisions is thus set by LLM deliberation and by the results each operation awaits, with no optimization cycle required to finish first. Writing $\mathcal{H}_{\tau(t)}$ for the results committed when decision t begins, the state s_t is the six diagnostic views computed from $\mathcal{H}_{\tau(t)}$, the action a_t is one of four operations, and the LLM acts in context as the policy mapping s_t and the circuit to a_t and its arguments.

Algorithm 1 A TRACE sizing episode

Require: topology, knowledge, contract
$c \leftarrow$ CONFIGURE(topology, knowledge, contract)
$\mathcal{H} \leftarrow$ INITIALDESIGN(c) ▷ $\leq 4\times5$ candidates
start BO on c from $\mathcal{H}$ ▷ BO commits to $\mathcal{H}$
repeat
  $\tau \leftarrow |\mathcal{H}|$ ▷ all BO and LLM results so far
  $(a, \theta) \leftarrow$ LLM(VIEWS($\mathcal{H}_\tau$), circuit)
  if $a =$ REFINE **then**
    submit a candidate near a parent ▷ BO runs
  else if $a =$ EXPLORE **then**
    pause BO; probe 5 candidates; resume BO
  else if $a =$ REVISE **then**
    $c \leftarrow$ revised task; restart BO with all of $\mathcal{H}$
  end if ▷ HOLD: wait for k new BO results
  wait for the candidates of a
until a zero-loss candidate passes final review

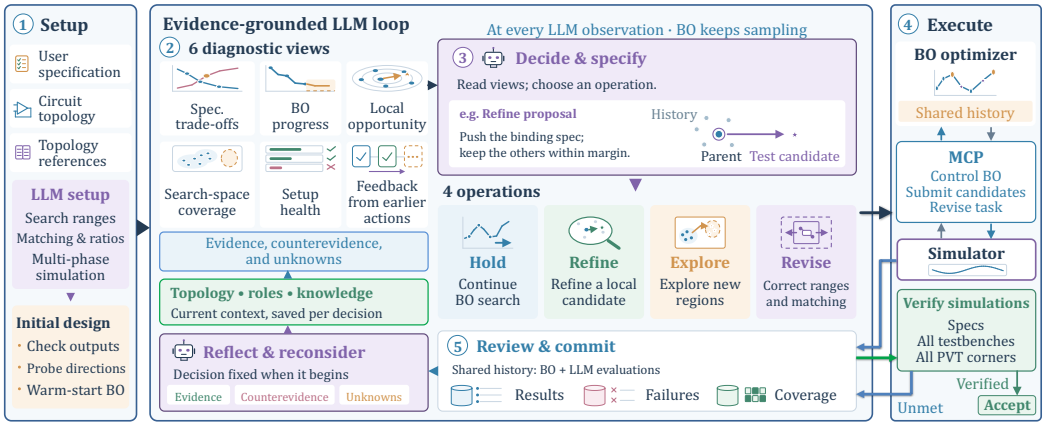

Figure 2: TRACE composes the sizing workflow in a decision loop that runs alongside BO. The LLM reads six diagnostic views, selects one operation, and authors its circuit-specific arguments.

3.3 Task setup and initial design

Beyond the specification that the designer provides, the LLM configures the task on its own. It reads the circuit netlist and gathers published material on the topology, from which it identifies device roles and confirms the matching and ratio constraints that leave the independent parameters. Working ranges are set around nominal values that the LLM estimates from first-order design equations collected for the topology, such as those of the g_m/I_D methodology (Silveira et al., 1996), and the evaluation is organized into phases that screen critical specifications first. The *acceptance contract*, comprising the specification bounds and the required testbench, corner, and operating-region coverage, comes from the user specification and stays fixed for the whole episode.

In place of a space-filling design, the LLM then probes the configuration adaptively with at most four batches of five candidates, stating what each batch tests and reading its results before issuing the next. Together the batches serve three ends. They verify that the configuration runs to completion and returns every required output, give the first diagnostic views evidence along distinct parameter directions, and provide the warm start for BO. Probing ends once all three are met or the four batches are used up, and from then on only evidence of a specific defect returns the episode to configuration.

3.4 Six diagnostic views

Before each decision, the results committed since the previous one enter the history, and the views are recomputed from the verified results it holds, those admitted under a fixed intake rule. Each view returns bounded evidence, counterevidence, and unknowns, indexed by parameter or device group, so the incumbent and the specification that binds it change as results arrive. Evaluation failures stay apart from poor measured performance (Appendix A). No view restricts the action set.

Two views establish where the design stands. **Specification trade-offs** name the lowest-loss verified candidate with full output coverage as the incumbent, and add up to eight non-dominated candidates that show which specifications compete and offer parents other than the incumbent. Each specification carries a signed margin to its bound, measured at the worst corner. **BO progress** reads how much the loss has improved and how steeply it still falls over the last twelve verified candidates. To tell a stall that still reaches new ground from one that resamples near earlier points, it counts how many recent candidates $\mathcal{K}_t$ lie close to something already tried,

$$\nu_t = \frac{1}{|\mathcal{K}_t|} \sum_{i \in \mathcal{K}_t} \mathbf{1}\left[\min_{i'<i} d(i,i') \le \delta\right], \qquad \delta = 0.02, \tag{1}$$

where d is the mean absolute distance over shared parameters on the normalized working surface.

Three views match the scopes at which a remedy can act, and a fourth looks back. **Local opportunity** fits distance-weighted ridge models (Hoerl & Kennard, 1970) around the best comparable candidate, using up to twelve neighbours evaluated under the same configuration and coverage and

preferring complete results. The models relate each specification to a change in parameters, and a hypothesis qualifies only if the fit predicts held-out neighbours, the direction repeats, and the protected margins survive. On thinner evidence the view reports unknown. **Search-space coverage** asks how often the best candidates sit against a bound, comparing how often the lowest-loss quarter hits a bound with how often all comparable candidates do, for each parameter and each side of its range. It also reports where sampling has concentrated inside the working surface, which regions are newly visited, and which freedoms of the reference surface the working surface leaves out. **Setup health** groups failed candidates by how they terminated, why they failed, and which metric errored, and weighs them against recent verified candidates, which separates a faulty configuration from a region that is merely hard to satisfy. It covers only the current revision, so failures from a configuration that has since been corrected no longer count as evidence. **Feedback from earlier actions** pairs each resolved outcome with the decision that produced it. This is the only view that looks back across the episode. When the situation resembles an earlier one, the agent may reuse what succeeded before, and it can also see which attempts in a region yielded no gain.

> The views are an interface suited to both sides of the loop, computed with the statistics of numerical optimization and delivered as bounded evidence an LLM can reason over. They add no information beyond the history, and a capable model could derive each of them from raw records. What the views change is how much a decision must read. A sizing episode accumulates hundreds of candidates, and an LLM's use of relevant evidence can depend on its position in a long context (Liu et al., 2024). Because the views summarize the whole history at a fixed size and from a fixed set of vantage points, every decision starts from the same view of the episode, however long it has run. The set provides distinct evidence relevant to the different operation scopes, and we do not claim it is minimal. Without the views, decision time rises in both mean and variance, and the agent retrieves more simulation-history records per decision (Section 4).

3.5 Four orchestration operations

Local BO can exploit structure poorly modeled by global surrogates (Eriksson et al., 2019), while acquisition portfolios adapt sampling strategies (Hoffman et al., 2011). TRACE offers four operations with different scopes, costs, and feedback delays. The agent combines the views with circuit topology and device roles to choose an operation and its arguments: a parent and parameter changes for Refine, a region and candidates for Explore, or configuration changes for Revise. The views do not prescribe these choices. For the stalls in Section 1, boundary clustering can motivate Explore within the working surface before considering a wider range through Revise. Competing specifications can motivate Refine from alternative parents in the trade-off view, while setup-health evidence can justify Revise. Hold lets BO continue when intervention is not yet justified.

Hold allows BO at least $k = 3$ further results before the next decision, either because BO is still improving and an intervention would disturb a productive search, or because a probe would not settle the open question. The LLM may hold again once those results arrive.

Refine claims that one direction is better and spends a single candidate to test it, submitted near a verified parent while BO keeps running. The result enters the shared history even when it contradicts the hypothesis that prompted it.

Explore claims only that a region deserves sampling. TRACE pauses BO, fixes the best verified historical loss as the baseline, and submits five candidates that are not required to improve on the incumbent. Before simulation, a fixed fraction of them must lie on untried ground, and if too few do, the round is redesigned once at no cost in simulations. After evaluation, at least one verified candidate from the batch must come within a fixed tolerance of the baseline, and if none does, one further round is allowed. Neither the task nor the working surface changes.

> Exploration breaks prolonged BO stagnation. Once BO has stalled, further proposals return little, and the next simulations are better spent on the circuit reasoning BO lacks, so BO pauses while the LLM samples a promising region. A round submits several candidates, since one sample cannot support a claim about a region, and one good point rarely shifts a surrogate fitted on the whole history. Since it cannot know which will prove good, the round is judged on novelty and competitiveness, with improvement left to later refinement. The working surface is unchanged, so a failed probe costs only simulations.

Revise repairs a configuration defect that the evidence has established, such as a range that excludes good designs or a matching constraint that ties the wrong devices. The LLM names the task to move

to, configures it, and reads it back before BO restarts. BO restarts from the entire committed history, which the revision leaves intact, and each observation stays attributed to the revision that produced it. The acceptance contract is preserved, so success cannot be obtained by loosening the target. Poor performance alone does not justify a revision.

> Hold and Revise express the same restraint from opposite ends. Hold gives the LLM the right to keep a hypothesis without acting on it. The agent is under no obligation to propose candidates, and while the facts remain unclear, leaving BO undisturbed is the better use of the interval. A hold therefore spans several BO results, since one more result would leave the views nearly unchanged and the next decision would face the same evidence. BO samples where it expects improvement, so low loss in a region partly reflects that choice. A cut on this evidence can remove designs that were never tried, and the excluded region returns no further observations, so nothing later in the episode can reveal the mistake. Reshaping the space repeatedly also has a price, since a surrogate needs enough observations under one definition of the space, and every revision resets the search state BO has built up.

3.6 Decision records and acceptance

Each decision uses a frozen brief, so later results may inform its evaluation but are not treated as evidence available when it was made. The chosen operation, rationale, and cited evidence are recorded before execution. An operation is credited only with the candidates it submitted. Those candidates also enter the surrogate, so BO proposes differently afterwards, and any improvement measured later therefore remains observational. These records support decision-level replay. An exact zero-loss candidate triggers a final review, the only stage at which the LLM inspects the simulated waveforms, confirming that simulator records meet the acceptance contract. Given previously reported limitations in LLM reasoning self-correction without external feedback (Huang et al., 2024), acceptance rests on simulator evidence and never on an unverified LLM judgment or the scalar loss alone. Record formats and replay are detailed in Appendix A.

4 Experiments

We test whether online intervention composition lowers sizing cost relative to fixed optimizers and single-action variants (Section 4.2), diagnostic views reduce decision overhead (Section 4.3), and task revision recovers from restrictive configurations (Section 4.4).

4.1 Tasks and setup

Tasks. We evaluate an operational transconductance amplifier (OTA), comparator (COMP), low-dropout regulator (LDO), and bandgap reference (BGR). Table 1 summarizes their scale; Appendix D.1 gives counting conventions, schematics, specifications, and corners. Acceptance requires verified specifications and device operating constraints on every frozen testbench and corner.

Environment. We use GPT-6 astra at medium reasoning effort, connected to Cadence Virtuoso and Spectre through custom-built MCP interfaces, with the TSMC 28 nm PDK and four parallel Spectre simulator instances. The view ablation additionally uses DeepSeek V4.1 Flash Max (Section 4.3).

Baselines and ablations. We compare tSS-BO (Gu et al., 2024), TuRBO (Eriksson et al., 2019), and adapted ADO-LLM (Yin et al., 2024) and LEDRO (Kochar et al., 2025). TRACE and its ablations use tSS-BO as the BO backend. Both BO baselines and all ablations use TRACE's phased evaluator. Initial design + BO disables online interventions; Refine only and Explore only enable their respective operation. Appendix D.3 gives setup details.

Table 1: Circuit scale and full-evaluation cost. Sims/eval counts testbench–corner runs.

Circuit	Difficulty	#Devices	#Indep. params.	Space size	#Specs	#Sims/eval
OTA	Easy	9	15	$\sim 10^{29}$	10	21
COMP	Medium	17	25	$\sim 10^{24}$	5	12
LDO	Hard	19	26	$\sim 10^{48}$	10	21
BGR	Hard	43	31	$\sim 10^{38}$	5	12

Table 2: TRACE has the lowest mean simulation count on all four circuits. $\bar{S}$ counts testbench–corner simulations and $\bar{T}$ measures wall time (min), reported as mean $\pm$ standard deviation over five complete episodes. Arrows indicate the preferred direction.

Method	OTA (Easy)		COMP (Medium)		LDO (Hard)		BGR (Hard)	
	$\bar{S}\downarrow$	$\bar{T}\downarrow$	$\bar{S}\downarrow$	$\bar{T}\downarrow$	$\bar{S}\downarrow$	$\bar{T}\downarrow$	$\bar{S}\downarrow$	$\bar{T}\downarrow$
Bayesian optimization								
tSS-BO	832 $\pm$ 41	70.9 $\pm$ 6.5	912 $\pm$ 93	71.7 $\pm$ 8.2	1446 $\pm$ 76	133.5 $\pm$ 5.6	1364 $\pm$ 82	106.2 $\pm$ 6.7
TuRBO	1043 $\pm$ 52	92.3 $\pm$ 7.9	936 $\pm$ 89	73.6 $\pm$ 9.6	1226 $\pm$ 70	123.5 $\pm$ 5.6	1248 $\pm$ 68	94.1 $\pm$ 4.7
LLM-based baselines								
ADO-LLM	462 $\pm$ 51	43.0 $\pm$ 5.2	336 $\pm$ 29	45.3 $\pm$ 3.3	861 $\pm$ 79	72.5 $\pm$ 8.6	554 $\pm$ 45	81.8 $\pm$ 5.5
LEDRO	693 $\pm$ 51	60.9 $\pm$ 5.5	504 $\pm$ 29	47.2 $\pm$ 4.2	1155 $\pm$ 66	99.4 $\pm$ 7.2	720 $\pm$ 38	74.9 $\pm$ 4.3
TRACE ablations								
Initial design + BO	674 $\pm$ 65	60.2 $\pm$ 5.9	669 $\pm$ 84	53.1 $\pm$ 5.8	1011 $\pm$ 87	101.8 $\pm$ 6.6	894 $\pm$ 76	82.2 $\pm$ 6.3
Refine only	392 $\pm$ 28	33.3 $\pm$ 3.2	198 $\pm$ 26	**18.0** $\pm$ 1.5	923 $\pm$ 79	76.3 $\pm$ 8.3	582 $\pm$ 52	72.1 $\pm$ 5.6
Explore only	522 $\pm$ 56	45.0 $\pm$ 7.2	231 $\pm$ 34	23.0 $\pm$ 2.2	767 $\pm$ 68	69.1 $\pm$ 6.3	737 $\pm$ 62	87.8 $\pm$ 8.4
TRACE	**386** $\pm$ 23	**32.7** $\pm$ 2.9	**184** $\pm$ 24	18.1 $\pm$ 1.6	**692** $\pm$ 56	**60.4** $\pm$ 8.6	**493** $\pm$ 35	**65.7** $\pm$ 4.7
Speedup vs. tSS-BO	2.2$\times$	2.2$\times$	5.0$\times$	4.0$\times$	2.1$\times$	2.2$\times$	2.8$\times$	1.6$\times$

Bold: lowest mean per circuit; shading identifies TRACE. Speedup is tSS-BO/TRACE (larger is better).

Protocol and metrics. We report mean testbench–corner simulation count $\bar{S}$ and episode wall time $\bar{T}$ (minutes), covering initialization through verified feasibility. Simulation counts capture differences in per-candidate evaluation cost, while candidate counts alone do not account for phased evaluation or different intervention cadences. External comparisons measure complete sizing-system cost; Appendix D.5 details this scope, repeats, seeds, and accounting.

4.2 Main results

Table 2 averages complete-episode costs over five runs. All methods reach verified designs, and TRACE has the lowest mean simulation count on all four circuits. Mean wall time is also lowest except on COMP, where Refine only is marginally faster (18.04 versus 18.12 minutes). Relative to TRACE, tSS-BO requires 2.09–4.96$\times$ as many simulations and 1.62–3.96$\times$ as much wall time. TuRBO uses fewer simulations and less wall time than tSS-BO on LDO and BGR, but more on OTA and COMP; TRACE has lower costs on both measures than either BO baseline across all four circuits. ADO-LLM and LEDRO take 1.14–2.61$\times$ as long as TRACE. On BGR, TRACE uses 11.1% fewer simulations than ADO-LLM (493 versus 554) and 19.6% less wall time.

Initial design + BO reduces both costs relative to the BO baselines on every circuit, but remains more expensive than full TRACE. On OTA and COMP, full TRACE offers little additional wall-time benefit over Refine only. In one OTA run, the first batch produced a promising candidate sufficient to warm-start BO. Judging the target within reach, the agent instead chose three further batches and reached a verified design within 20 candidates without starting BO. This episode illustrates adaptive initialization; the aggregate ablation results support additional savings from online interventions.

Full TRACE has lower mean costs than either single-action variant on LDO and BGR. The stronger single-action variant is Explore only on LDO and Refine only on BGR. Relative to these variants, full TRACE uses 9.8% and 15.3% fewer simulations and takes 12.6% and 8.8% less wall time, respectively. Both single-action variants succeed in every run. This reversal may reflect different bottlenecks, with LDO benefiting from coordinated changes across blocks and BGR from sustained local refinement (Appendix D.4). Explore's five-point batches can also improve previously visited regions, so the comparison does not isolate search breadth from batch size.

Figure 3 traces two BGR episodes. In both, a compensation-network probe moves the design onto a better branch, followed by local refinements that close the remaining gap. The final refinement in each episode cites its predecessor's improvement to justify continuing in the same direction. TuRBO has not yet reached the PSRR target after the same number of candidate evaluations.

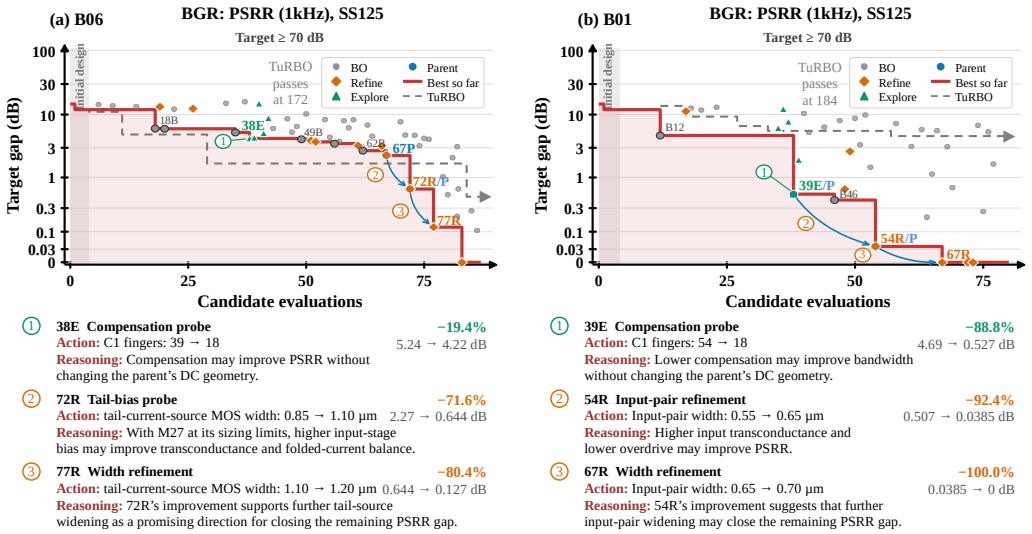

Figure 3: Two BGR episodes, tracked on PSRR at 1 kHz in the SS125 corner against a target of $\geq$ 70 dB. The target gap is $\max(70 - \mathrm{PSRR}, 0)$ in dB on a nonlinear scale, and the horizontal axis indexes candidate evaluations. Red steps track the best observed gap, markers distinguish BO candidates, Refine, Explore, and selected parents, and blue arrows connect a parent to the proposal it motivated, annotated with the change and the stated reason. Dashed curves are TuRBO trajectories. A zero gap means PSRR meets its SS125 target, not full acceptance.

4.3 Diagnostic views and reasoning overhead

We evaluate whether diagnostic views reduce the cost of choosing an action on LDO and BGR, using GPT-6 astra medium and DeepSeek V4.1 Flash Max. For each model and circuit, we compare **Full views**, which provides the six diagnostic views, with **Raw records only**, which provides the underlying simulation records without these summaries. The summaries also remain unavailable during subsequent retrievals. Within each circuit, the initial design, BO seeds, observation intervals, and stopping rules are matched across conditions. Each model uses the same version and inference settings with and without the views.

Figure 4 reports LLM inference time and record queries per decision after initialization, together with total episode wall time. Inference time sums the latency of all LLM requests needed to choose an action and specify its arguments, excluding retrieval execution, simulation, and optimizer waiting. Record queries count entries retrieved from simulation history or stored records. Total wall time covers the complete episode, including reasoning, simulation, and waiting.

Full views reduce inference time per decision by 33–45% and record queries by 20–56% across the four circuit–model pairs, with smaller standard deviations in both metrics. Total episode wall time falls by 13% on LDO and 12% on BGR for GPT-6 astra, and by 9% on both circuits for DeepSeek.

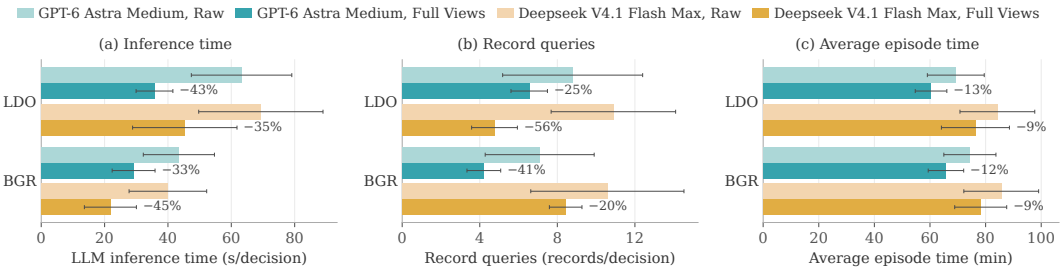

Figure 4: Decision and episode costs with and without diagnostic views on LDO and BGR. Panels (a) and (b) report mean LLM inference time and record queries per decision, with standard deviations over decisions pooled from three runs per condition. Panel (c) reports average episode time across three runs, including reasoning, simulation, and waiting, with error bars showing the standard deviation across runs. Percentages show reductions under Full Views relative to Raw records.

More retrieval under Raw records is consistent with the agent reconstructing information supplied by the views. The measured cost reductions across both models support diagnostic views as an efficient interface for intervention decisions. Both Full views and Raw records only reached verified feasible designs in all three runs for each model and circuit.

4.4 Case studies: recovery from restrictive setups

We present four case studies of TRACE recovering from restrictive initial sizing configurations through task revision. These configurations use the parameter ranges listed in Appendix D.6, which differ from the main-comparison settings in Appendix D.2. The agent is not informed which parameter ranges are restrictive. BO alone, with Refine, Explore, and Revise disabled, found no verified design on any restricted configuration within five hours (Appendix D.6).

Table 3: Recovery from restricted search spaces, including all revision costs. Each cost ratio divides the recovery cost by the unrestricted TRACE mean for the same circuit; values above 1 indicate higher cost. Parameter restrictions are listed in Appendix D.6.

Circuit	Revisions	Candidates	Candidate ratio	Time (min)	Time ratio
OTA	1	212	3.37	73.52	2.25
COMP	1	137	3.19	33.76	1.86
LDO	2	109	1.30	51.24	0.85
BGR	2	199	2.62	78.10	1.19

TRACE recovers a verified design on all four circuits, requiring one revision on OTA and COMP and two on LDO and BGR. The revisions respond to circuit-specific evidence. On COMP, delay decreases as the input-pair length is reduced but remains above the specification at the lower bound, motivating a further range extension. On LDO, severe PSRR degradation under heavy load motivates expanding the pass-device sizing range. These cases illustrate how simulation feedback guides targeted configuration changes while the original acceptance contract remains fixed. Appendix D.6 traces the evidence behind each revision, including OTA's unsuccessful current-increase probes and BGR's range extensions and metric-label correction.

5 Discussion and conclusion

TRACE decides the sizing workflow online, and the experiments show where this matters. On the easy circuits a few local refinements suffice; the gap opens on LDO and BGR, where the stronger single-action variant is Explore only on one circuit and Refine only on the other. Neither intervention is uniformly better across the tested circuits, motivating orchestration that adapts when and at what scope to intervene as evidence accumulates, rather than committing to one intervention pattern in advance. The diagnostic views lower inference time and record queries per decision. By linking each decision to the evidence it used and attributing delayed results to their originating operations, TRACE supports decision-level replay and final design verification against simulator records.

TRACE shows substantial cost reductions across circuits of varying complexity, although the main comparison covers four circuits and five runs per method and circuit. Broader coverage and more repeats would strengthen the generality of these findings. Task revision succeeds in all tested cases, with useful diagnostic evidence from the views, but still requires additional candidate evaluations. Future work could develop diagnostic algorithms tailored to circuit sizing to identify configuration problems earlier and reduce recovery cost.

Beyond individual episodes, records gathered across circuits and topologies could help identify when particular interventions are useful and which lessons transfer to new sizing tasks. An agent that reviews these records could use earlier outcomes to inform later decisions, accumulating experience across runs. Evaluating such reuse offers a path toward orchestration that improves with experience beyond the capabilities of the base model.

AI USE STATEMENT

We used AI tools to aid and polish writing and to support literature retrieval and the discovery of related work. We also used AI tools to assist with implementing the LEDRO baseline and the BO progress diagnostic view.

REFERENCES

Mohsen Ahmadzadeh, Kaichang Chen, and Georges Gielen. (invited paper) AnaFlow: Agentic LLM-based workflow for reasoning-driven explainable and sample-efficient analog circuit sizing. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 1–7, 2025. doi: 10.1109/ICCAD66269.2025.11240818.

Zhixuan Bao, Zhuoyi Lin, Jiageng Wang, Jinhai Hu, Yuan Gao, Yaoxin Wu, Xiaoli Li, and Xun Xu. AnalogAgent: Self-improving analog circuit design automation with LLM agents, 2026. URL https://arxiv.org/abs/2603.23910.

André Biedenkapp, H. Furkan Bozkurt, Theresa Eimer, Frank Hutter, and Marius Lindauer. Dynamic algorithm configuration: Foundation of a new meta-algorithmic framework. In *Proceedings of the 24th European Conference on Artificial Intelligence*, 2020. doi: 10.3233/FAIA200122.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. doi: 10.1017/CBO9780511804441. URL https://web.stanford.edu/~boyd/cvxbook/.

William S. Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836, 1979. doi: 10.1080/01621459.1979.10481038.

David Eriksson, Michael Pearce, Jacob R. Gardner, Ryan Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/6c990b7aca7bc7058f5e98ea909e924b-Abstract.html.

J. C. Gower. A general coefficient of similarity and some of its properties. *Biometrics*, 27(4): 857–871, 1971. doi: 10.2307/2528823.

Tianchen Gu, Jiaqi Wang, Zhaori Bi, Changhao Yan, Fan Yang, Yajie Qin, Tao Cui, and Xuan Zeng. tSS-BO: Scalable bayesian optimization for analog circuit sizing via truncated subspace sampling. In *Proceedings of the Design, Automation & Test in Europe Conference & Exhibition*, pp. 1–6, 2024. doi: 10.23919/DATE58400.2024.10546770. URL https://past.date-conference.com/proceedings-archive/2024/DATA/31_pdf_upload.pdf.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. doi: 10.1080/00401706.1970.10488634.

Matthew Hoffman, Eric Brochu, and Nando de Freitas. Portfolio allocation for bayesian optimization. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pp. 327–336. AUAI Press, 2011. URL https://mwhoffman.com/papers/hoffman_2011.pdf.

Shengran Hu, Cong Lu, and Jeff Clune. Automated design of agentic systems. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/36b7acf6f6010652b3f2a433774a66fe-Abstract-Conference.html.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=IkmD3fKBPQ.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492, 1998. doi: 10.1023/A:1008306431147.

Pooria Joulani, András György, and Csaba Szepesvári. Online learning under delayed feedback. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28(3) of *Proceedings of Machine Learning Research*, pp. 1453–1461, 2013. URL https://proceedings.mlr.press/v28/joulani13.html.

Kirthevasan Kandasamy, Akshay Krishnamurthy, Jeff Schneider, and Barnabas Poczos. Parallelised bayesian optimisation via thompson sampling. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 133–142, 2018. URL https://proceedings.mlr.press/v84/kandasamy18a.html.

Dimple Vijay Kochar, Hanrui Wang, Anantha P. Chandrakasan, and Xin Zhang. LEDRO: LLM-enhanced design space reduction and optimization for analog circuits. In *Proceedings of the IEEE International Conference on LLM-Aided Design*, pp. 141–148, 2025. doi: 10.1109/ICLAD65226.2025.00011.

Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Garcia, Max Hasin, Sami Jawhar, Megan Kinniment, Nate Rush, Sydney Von Arx, Ryan Bloom, Thomas Broadley, Haoxing Du, Brian Goodrich, Nikola Jurkovic, Luke Miles, Seraphina Nix, Tao Lin, Neev Parikh, David Rein, Lucas Jun Koba Sato, Hjalmar Wijk, Daniel Ziegler, Elizabeth Barnes, and Lawrence Chan. Measuring AI ability to complete long software tasks. In *Advances in Neural Information Processing Systems*, volume 38, pp. 92213–92266, 2025. doi: 10.52202/085713-3086. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/85069585133c4c168c865e65d72e9775-Abstract-Conference.html.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Wenlong Lyu, Fan Yang, Changhao Yan, Dian Zhou, and Xuan Zeng. Batch bayesian optimization via multi-objective acquisition ensemble for automated analog circuit design. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3306–3314, 2018. URL https://proceedings.mlr.press/v80/lyu18a.html.

M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979. doi: 10.1080/00401706.1979.10489755.

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, pp. 8634–8652, 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Fernando Silveira, Denis Flandre, and P. G. A. Jespers. A g_m/I_D based methodology for the design of CMOS analog circuits and its application to the synthesis of a silicon-on-insulator micropower OTA. *IEEE Journal of Solid-State Circuits*, 31(9):1314–1319, 1996. doi: 10.1109/4.535416.

M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(2):111–133, 1974. doi: 10.1111/j.2517-6161.1974.tb00994.x.

Konstantinos Touloupas, Nikos Chouridis, and Paul P. Sotiriadis. Local bayesian optimization for analog circuit sizing. In *Proceedings of the 58th ACM/IEEE Design Automation Conference*, pp. 1237–1242, 2021. doi: 10.1109/DAC18074.2021.9586172.

Ziming Wei, Zichen Kong, Yuan Wang, David Z. Pan, and Xiyuan Tang. TopoSizing: An LLM-aided framework of topology-based understanding and sizing for AMS circuits, 2025. URL https://arxiv.org/abs/2509.14169.

Wei W. Xing, Baowen Ou, Yuxuan Zhang, Zhuohua Liu, and Yuanqi Hu. ASTRA: Automatic sizing of transistors with reasoning agents. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 1–9, 2025. doi: 10.1109/ICCAD66269.2025.11240675.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Yuxuan Yin, Yu Wang, Boxun Xu, and Peng Li. ADO-LLM: Analog design bayesian optimization with in-context learning of large language models. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 1–9, 2024. doi: 10.1145/3676536.3676816.

Xi Yu, Dmitrii Torbunov, Soumyajit Mandal, and Yihui Ren. AutoSizer: Automatic sizing of analog and mixed-signal circuits via large language model (LLM) agents, 2026. URL https://arxiv.org/abs/2602.02849.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xionghui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, Bingnan Zheng, Bang Liu, Yuyu Luo, and Chenglin Wu. AFlow: Automating agentic workflow generation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/5492ecbce4439401798dcd2c90be94cd-Abstract-Conference.html.

A EVIDENCE STORE AND DECISION RECORDS

Verified store. A simulation that returns a value outside a specification bound still provides information about circuit performance; an evaluation error instead leaves the affected measurement unavailable. TRACE retains observations for diagnosis under a fixed intake rule, recording any metric errors and which required analyses were completed. This collection is called the verified store, but inclusion does not mean that a design meets all specifications. Results obtained before an evaluation stops early can still guide subsequent decisions, although they cannot establish feasibility without the remaining required analyses. The underlying simulation records remain available for inspection when a diagnostic summary is insufficient.

History used by the views. Each decision uses views computed from the simulation records already committed when it begins. New results update the views for later decisions; they do not alter the evidence recorded for an earlier decision. Different views use different parts of the history. For example, setup health considers only the current task revision, so failures from a corrected configuration are not treated as current setup problems. Feedback from earlier actions can span the episode, exposing earlier decisions and their outcomes. The purposes and computations of all six views are described in Section 3.4 and Appendix B.

Decision brief. At the start of each decision, TRACE saves the current task and revision, the current best candidate and the status of its specifications, and the six diagnostic views used for the decision. A separate action record states the chosen operation, the reasons for choosing it, the supporting simulation records, and any remaining uncertainties.

Delayed feedback. Each Refine or Explore action records the candidates it submits. Once all of those simulations have finished or failed, their outcomes are added to the history and linked to the action that submitted them.

Final review. Final acceptance requires simulator records confirming every specification and device operating-region constraint on all required testbenches and PVT corners. A design is not accepted if required results are missing, some required analyses are incomplete, or the records cannot be matched to the task being reviewed.

Example decision record. The following excerpt comes from a BGR Full episode. Field names are retained, evidence references use readable labels, and the rationale is shortened. The unresolved questions specify what the proposed simulation must test.

```
{
  "decision_seq": 1,
  "context_ref": "decision 1 evidence",
  "action": "refiner",
  "evidence_refs": ["candidate 7", "initial-design batch 2"],
  "rationale": "Refine current full-coverage idx7 with matched
     input-pair gm increase.",
  "counterevidence_refs": [],
  "unresolved_unknowns": [
    "Input-pair size effect on SS TC and Region margins requires full
       simulation."
  ],
  "execution_status": "not_executed"
}
```

Here SS denotes the slow process corner and TC the temperature coefficient; Region margins refer to device operating-region constraints. The empty counterevidence list records that none was cited, not that none could exist. The record was saved before the action was executed, so `not_executed` describes its status at that time.

B COMPUTING THE DIAGNOSTIC VIEWS

The six views use the committed records available when a decision begins (Appendix A). The following steps specify their inputs, calculations, and outputs. Numerical settings are fixed across circuits, seeds, and runs; missing evidence is reported as unknown. Performance summaries use verified observations, while setup health also uses evaluation failures. Each view retains supporting evidence, counterevidence, and unknowns where applicable; none determines which operation the LLM must choose. The interface uses a fixed summary structure and bounded representative lists within a fixed input-size limit. As observations accumulate, counts and summary values are updated; the underlying records remain separately available through evidence references. The first five views describe the current search state. The sixth links earlier decisions to their outcomes across the episode.

Observation intake. Simulations that terminate unsuccessfully enter the failure store. For a completed evaluation, divide the number of explicitly errored outputs by the number of outputs associated with the phases evaluated for that candidate. Outputs from later, unexecuted phases are excluded from this denominator. A ratio above 50% places the result in the failure store; otherwise it enters the verified store with any errors preserved. Early-stopped partial results may inform diagnostics but cannot establish full-coverage feasibility.

Shared distance calculation. For each parameter, subtract its working lower bound, divide by its range width, and clip the result to $[0, 1]$. The distance between two candidates is the mean absolute difference over their shared observed parameters (Gower, 1971); it is unknown if they share none. Local-response and boundary analyses additionally require complete parameter vectors inside the working surface.

B.1 SPECIFICATION TRADE-OFFS

Select the verified candidate with the lowest simulator-reported loss among those with full output coverage, breaking ties by earlier simulation index. If none has full coverage, use the best partial result provisionally. From the same full-coverage pool, or the partial pool when necessary, add up to eight Pareto non-dominated alternatives (Boyd & Vandenberghe, 2004, Section 4.7), comparing each specification's worst returned loss and selecting by total loss, then earlier index. Partial results cannot establish dominance on unobserved specifications.

For each specification, calculate a signed margin to its zero-loss interval. For a lower bound, subtract the bound from the measured value; for an upper bound, subtract the measured value from the bound. Divide by the span between the endpoints of the configured loss curve. For logarithmic specifications, take logarithms of the measurement, bounds, and curve endpoints before these subtractions. If both bounds exist, keep the smaller margin. Positive margins indicate slack, zero indicates a boundary, and negative margins indicate a violation. Report the smallest margin across required corners only when every corner margin is known. This calculation requires a contiguous zero-loss interval with monotone penalty tails, and positive values in logarithmic coordinates. Unsupported specifications, missing outputs, and evaluation errors leave margins unknown.

B.2 BO PROGRESS

Order verified candidates by simulation index and take the most recent twelve, or all candidates if fewer are available. Use the best loss before this window as the baseline, or the first observed loss if there is no earlier candidate. Subtract the window's best loss from the baseline, clamp negative improvement to zero, and divide by the larger of the baseline's absolute value and 10^{-12} to obtain relative improvement.

Fit an ordinary least-squares line with an intercept to loss against position in the window and report its slope; fewer than two usable observations leave the slope unknown. For each recent candidate, find its nearest comparable predecessor in the history. Report the fraction whose distance is at most 0.02, excluding candidates without a comparable predecessor. These summaries include both BO and LLM candidates.

B.3 LOCAL OPPORTUNITY

Select comparable neighbours. Group observations by task configuration and successfully evaluated specifications and corners, preferring full coverage. Among eligible groups, choose the one with the most evaluated outputs, then the most observations, breaking remaining ties by a fixed ordering of configuration and output identifiers. Boundary comparisons use this same group. Within the selected group, centre the analysis on the candidate with the lowest summed specification loss. Select the nearest twelve distinct neighbours, or all if fewer are available, whose normalized parameters each differ from the centre by at most 0.20. Fewer than five neighbours leave local opportunity unknown. Exclude the centre itself and repeated parameter vectors, and break distance ties by earlier simulation index.

Fit local responses. For up to six violated or protected specifications, predict the change in response from the changes in normalized parameters relative to the centre. Rank violated outputs by descending loss and satisfied outputs by increasing margin, placing unknown margins first. When both exist, select up to three violated outputs, fill remaining slots with satisfied outputs, then use any spare slots for further violated outputs; break ties by output identifier. Use the negative signed margin as the response when available throughout the neighbourhood, and the returned weighted loss otherwise. Let $\Delta\mathbf{z}_i$ and Δy_i denote neighbour i's parameter and response changes from the centre. Fit a linear model without an intercept using weighted ridge regression (Cleveland, 1979; Hoerl & Kennard, 1970):

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left[\sum_i w_i \left(\Delta y_i - \boldsymbol{\beta}^\top \Delta\mathbf{z}_i \right)^2 + 0.10 \left\| \boldsymbol{\beta} \right\|_2^2 \right]. \tag{2}$$

The first term fits the observed changes; the second limits large coefficients. Set each neighbour's weight to one divided by the sum of its distance from the centre and 0.05. A negative coefficient suggests increasing that parameter; a positive one suggests decreasing it. Limit the diagnostic step to the smallest of the remaining range in that direction, 0.05 in normalized coordinates, and the largest observed local displacement along that parameter. Flag a direction without observations on its proposed side as extrapolation.

Check the direction and protected specifications. Refit the model with each neighbour withheld in turn (Stone, 1974). Compute a predictive score as one minus the total squared held-out error divided by the total squared response changes from the centre. Directional support requires a positive score, at least 80% agreement in coefficient sign across these fits, and a nonzero admissible step. Flat responses leave predictive quality unknown.

For a protected specification with a stable, predictive margin model, subtract the fitted coefficient times the proposed parameter step from the centre's margin. Flag a possible violation if this predicted margin minus the root mean squared held-out error is negative. This error is an empirical scale, not a confidence bound. Report protection risks separately from directional support. A hypothesis has support for both improvement and protection only when its direction passes the predictive checks and every required protection is assessed without a flagged violation. Unmodelled or unstable protections remain unknown. These checks support a hypothesis for simulation; they do not certify that the proposed circuit satisfies its specifications.

B.4 SEARCH-SPACE COVERAGE

Boundary concentration. Within a comparable group, select the lowest-loss quarter, rounding its size up to at least one candidate. For each parameter and each bound, calculate the fraction of this subset within 0.05 normalized units of the bound. Subtract the corresponding fraction over the whole group. A positive difference indicates excess boundary concentration among good candidates. Report both fractions and their counts; this does not establish that widening the range would improve the design.

Region occupancy and novelty. Group normalized parameters by circuit block, splitting groups into at most eight coordinates in the fixed parameter order; partition ungrouped parameters the same way. Divide each parameter's weight equally among the groups that contain it. Represent a region by the mean of its assigned vectors; regions with fewer than three observations remain provisional.

Algorithm 2 compares a candidate with these regions, including provisional ones. It uses the split parameter weights both within groups and when combining their votes.

Algorithm 2 Check whether a candidate is in a new region

Require: Normalized candidate, region centres, parameter groups and split weights
if comparable parameter values are unavailable **then**
  return unknown
end if
for each existing or provisional region **do**
  Compute the weighted root mean squared difference from its centre in each group
  Mark groups whose distance is at most 0.20 as matching
  Support $\leftarrow$ weight in matching groups divided by weight in all groups
  if support is at least 0.50 **then**
    return previously visited
  end if
end for
return new

Maintain regions by processing newly committed verified candidates in simulation index order. First compare established regions, selecting the largest support and breaking ties by region creation order. Support of at least 0.75, exceeding the runner-up by more than 0.10, gives an interior assignment; other matches with support at least 0.50 are boundary assignments. If none matches, use the best provisional region with support at least 0.50, or create a new provisional region. Update the assigned region's count and mean after interior or provisional assignments; boundary assignments leave its centre unchanged. A provisional region becomes established after three contributing observations.

Report region occupancy and newly visited regions, including partial evaluations with known parameters and assignments. Compare the reference and working surfaces to identify excluded parameter freedoms. Occupancy does not estimate unexplored volume.

Explore uses this novelty calculation with the pre-action regions fixed. At least three of its five proposed candidates must be new, counting candidates rather than distinct regions. After simulation, the lowest verified batch loss must be at most 1.10 times the best verified historical loss fixed before the action. This candidate may belong to a new or visited region; a batch without a verified result fails. A proposal that fails the pre-simulation novelty check may be redesigned once without running simulations. Failure of the post-simulation loss check has already incurred the batch's simulation cost and permits another round only if the two-round action limit has not been reached. Collected results update regions for later decisions without reassessing the action's fixed novelty fraction.

B.5 Setup health

Use failures from the current task revision, grouped by termination status, failure reason, and affected metric. Mark a label as recurrent when its count reaches both two occurrences and 40% of the failure records, rounding the latter threshold up. Also report the failure rate among terminal evaluations, unresolved device-configuration issues, and recent verified observations as counterevidence. Failures from earlier revisions do not contribute to current setup health. A recurrent failure is a reason to investigate, not proof of a configuration defect; poor measured performance alone does not justify Revise. Metric errors retained in verified observations remain visible, so an empty failure group does not establish error-free evaluation.

B.6 Feedback from earlier actions

Group completed decisions by operation and link their outcomes to the frozen decision briefs. For Refine and Explore, use the results of the candidates each action submitted, as described in Appendix A. For Hold, the feedback is BO progress during the subsequent waiting interval. For Revise, it is whether the identified setup defect was corrected and how search proceeds under the revised configuration. This view spans the episode, so the LLM can inspect earlier attempts and their outcomes. Completion alone does not establish success. Improvements from subsequent BO search

remain observational; they are not treated as improvements from candidates directly submitted by Refine or Explore.

C Implementation details

C.1 Device geometry and effective width

For a fingered MOS device, total effective width is $W_{\mathrm{eff}} = W_{\mathrm{fg}}\, N_{\mathrm{f}}\, M$, where W_{fg} is the width of each finger, N_{f} is the number of fingers per device, and M is the number of parallel copies. Channel length L is shared by all fingers. The PDK constrains which combinations of width, length, and finger count are legal, so the tabulated intervals do not imply that every combination is realizable. Spectre may adjust BO-submitted device dimensions to comply with PDK geometry constraints. Evaluations therefore use the dimensions reported by the simulator, which may differ from the submitted values.

C.2 LLM and optimizer settings

The reusable skills provide the orchestration instructions: inspect the current evidence, choose one allowed operation, explain its target and protected specifications, and save the rationale before execution. Refine inherits a verified parent's complete parameter vector and specifies only the changes; Explore proposes five complete candidates within the current ranges. These instructions leave the circuit-specific hypotheses and candidate values to the LLM. Table 4 lists the principal numerical settings.

Table 4: Principal settings for TRACE. BO settings refer to the tSS-BO backend used within TRACE; adapted baselines are described in Appendix D.3.

Component	Setting
LLM	GPT-6 astra, medium reasoning effort; DeepSeek V4.1 Flash Max additionally used in the view ablation
BO proposal rule	Expected improvement (Jones et al., 1998); candidate pool of 256
GP numerical settings	Jitter 10^{-5}; length-scale bounds $[0.02, 2]$
Concurrent evaluation	Four BO trials and four Spectre simulator instances
Hold	At least three further BO results before another decision
Refine	One candidate per action
Explore	Five per round; $\geq 60\%$ new before simulation; at most two rounds
Repeat seeds	$1, 2, 3, 4, 5$ for the main comparison

D Supplementary experimental details

D.1 Circuit topology and PVT corners

Counting conventions. Devices are transistor, resistor, and capacitor instances in the DUT schematic; ports are excluded, and fingers and parallel multipliers are not expanded into separate instances. BGR includes its three bipolar transistors. Independent parameters are counted after matching and ratio constraints. Search-space size is the product of the independent discrete grid sizes before joint geometry constraints, so it does not count only legal or feasible designs.

A simulation in Table 1 is one distinct testbench–corner job. Multiple output metrics and device-region checks obtained from that job share its result; different corners require separate simulations. Internal sweep samples are not counted as separate jobs, and area calculations are excluded. The full-candidate counts are $7 \times 3 = 21$ for OTA and LDO and $4 \times 3 = 12$ for COMP and BGR. They describe complete evaluation coverage, not the average cost of candidates stopped in an early phase.

Specifications and units. Table 5 restores the complete target list summarized in Table 1. Device operating-region requirements remain part of acceptance in addition to these scalar specifications. LDO PSRR is displayed as positive rejection in dB, whereas its raw supply-to-output gain has the opposite sign. OTA noise is the input-referred spectral density at 100 kHz, reported in nV/$\sqrt{\mathrm{Hz}}$.

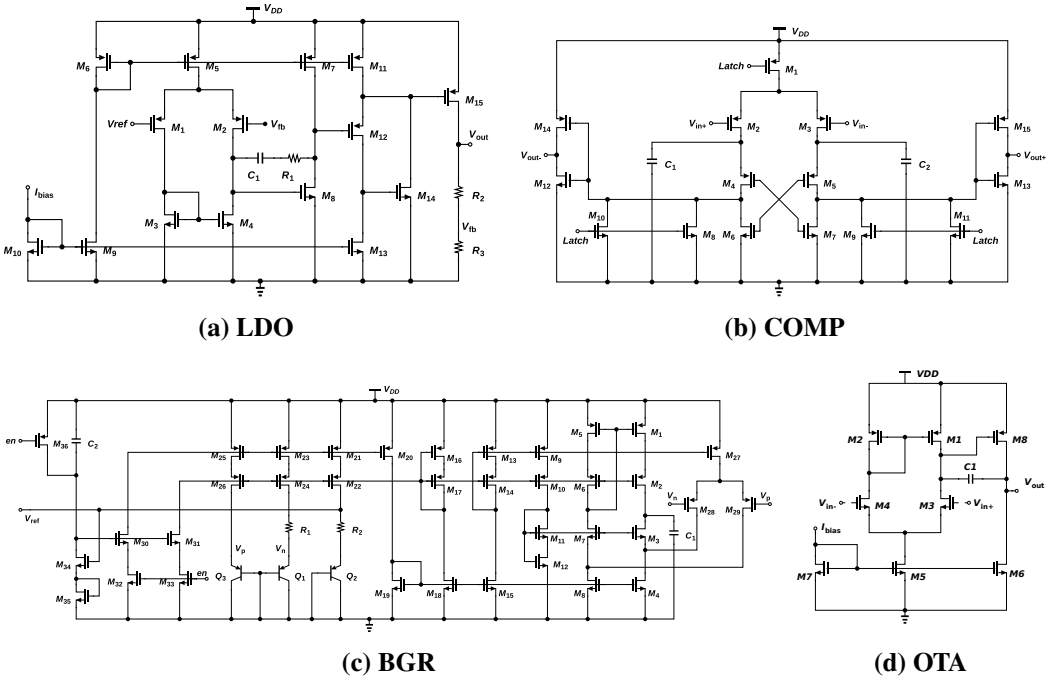

Figure 5: Circuit topologies. Top row: (a) LDO and (b) COMP. Bottom row: (c) BGR and (d) OTA.

Table 5: Complete design targets. I_L is load current.

Circuit	Design requirements	
OTA	DC gain ≥ 80 dB	Power ≤ 100 μW
	Phase margin $\in [45, 80]^\circ$	Output swing ≥ 1.5 V
	GBW ≥ 60 MHz	Slew rate ≥ 60 V/μs
	CMRR ≥ 70 dB	Settling time ≤ 100 ns
	PSRR ≥ 70 dB	Noise at 100 kHz ≤ 100 nV/$\sqrt{\text{Hz}}$
COMP	Reset time ≤ 20 ps	Energy per evaluation ≤ 100 fJ
	Decision time ≤ 80 ps	Kickback noise ≤ 80 mV
	Set time ≤ 100 ps	
LDO	PM at $I_L = 1,100$ mA $\in [45, 80]^\circ$	PSRR at 1/100 kHz:
	Load regulation ≤ 0.003 %/mA	$I_L = 1$ mA: $\geq 70/50$ dB
	$I_q \leq 100$ μA	$I_L = 100$ mA: $\geq 65/45$ dB
		$I_L = 300$ mA: $\geq 60/40$ dB
BGR	$\lvert\text{LNR}\rvert \leq 400$ μV/V	PSRR at 1/10 kHz $\geq 70/50$ dB
	Temperature coefficient ≤ 18 ppm/°C	PSRR at 100 kHz ≥ 30 dB

Process, voltage, and temperature corners. All four circuits use the same three corners: TT25, SS125, and FF-40, denoting typical, slow, and fast process conditions at 25, 125, and $-40\,^\circ$C, respectively. TT25 uses the nominal supply voltage, while SS125 and FF-40 use supplies 5% below and above nominal, respectively. For COMP, the supply voltages at TT25, SS125, and FF-40 are 0.900, 0.855, and 0.945 V. OTA and BGR both use 1.800, 1.710, and 1.890 V in the same corner order. LDO uses input supply voltages of 3.300, 3.135, and 3.465 V, respectively. Thus, each corner combines the specified process condition, supply voltage, and temperature.

Fixed bipolar devices. The BGR uses three vertical PNP transistors with identical $5\ \mu\text{m} \times 5\ \mu\text{m}$ unit emitters. Their parallel multipliers are fixed at $M_{Q_1} : M_{Q_2} : M_{Q_3} = 1 : 8 : 1$, giving the same ratio of total emitter areas. These settings are not optimized.

Evaluation phases. Table 6 lists the specifications evaluated in each phase. A phase passes when the sum of its metric losses is strictly below its screening threshold. In phase order, the thresholds are $(10, 10, 30)$ for OTA, $(10, 5, 30)$ for COMP, $(20, 12, 8, 24)$ for LDO, and $(20, 24, 16)$ for BGR. Failing a phase skips the remaining analyses for that candidate. These positive thresholds allow limited specification violations during screening; final acceptance still requires every declared specification and device check to pass. Operating-region checks accompany the OTA stability and gain analyses, LDO stability and PSRR analyses, and BGR DC, PSRR, and temperature analyses. The BGR configuration also collects reference voltage, power, and quiescent current as diagnostic outputs; these are not additional scalar acceptance targets.

Table 6: Ordered evaluation phases. All preceding phases must pass before the next is evaluated. PM denotes phase margin and TC temperature coefficient.

Circuit	Phase	Corners	Specifications evaluated
OTA	1	TT25	PM, GBW, DC gain, power, output swing
	2	TT25	Noise, slew rate, settling time, CMRR, PSRR
	3	SS125, FF−40	All OTA specifications
COMP	1	TT25	Reset time, decision time, set time
	2	TT25	Kickback noise, energy per evaluation
	3	SS125, FF−40	All COMP specifications
LDO	1	TT25	PM at $I_L = 1, 100$ mA
	2	TT25	PSRR at 1/100 kHz and $I_L = 1, 100, 300$ mA
	3	TT25	Load regulation, quiescent current
	4	SS125, FF−40	All LDO specifications
BGR	1	TT25	Line regulation
	2	TT25	TC; PSRR at 1/10/100 kHz
	3	SS125, FF−40	All BGR specifications

D.2 Device parameter ranges and matching constraints

Tables 7–10 give the MOS, capacitor, and resistor sizing ranges and fixed values used in the main comparison. Device labels match the schematics in Figure 5.

The notation $a : s : b$ denotes an inclusive linear grid from a to b with step s; a single number is fixed. W_{fg} is per-finger width, L is channel length, N_f is finger count, and M is the parallel multiplier. MOS devices grouped in the same variable row share W_{fg}, L, and N_f, including the OTA and LDO pairs. Additional links between rows are specified in the table notes. Parameter combinations must also satisfy the PDK's joint geometry constraints. Fixed passive components introduce no optimization variable; the total counts match Table 1.

Table 7: OTA device sizing ranges and fixed values: 15 independent parameters.

Figure devices	W_{fg} (μm)	L (nm)	N_f	M
M_1, M_2	$0.4 : 0.005 : 3$	$270 : 5 : 1200$	$1 : 1 : 16$	1
M_3, M_4	$0.4 : 0.005 : 3$	$270 : 5 : 1200$	$1 : 1 : 16$	1
M_5	$0.4 : 0.005 : 3$	$270 : 5 : 1200$	$1 : 1 : 16$	1
M_6	$0.4 : 0.005 : 3$	$270 : 5 : 1200$	$1 : 1 : 32$	1
M_7	$0.4 : 0.005 : 3$	$270 : 5 : 1200$	$1 : 1 : 12$	1
M_8	$0.4 : 0.005 : 5$	$270 : 5 : 1200$	$1 : 1 : 32$	1

Figure devices	Passive parameter	Range or fixed value
C_1	Capacitance (fF)	$150 : 25 : 450$

M_6, M_5, and M_7 share W_{fg} and L, but their finger counts vary independently.

Table 8: COMP device sizing ranges and fixed values: 25 independent parameters.

Figure devices	W_{fg} (μm)	L (nm)	N_f	M
M_1	1.6 : 0.05 : 2.4	40 : 5 : 100	2 : 1 : 6	1
M_2, M_3	1.6 : 0.05 : 2.4	40 : 5 : 100	4 : 1 : 10	1
M_4, M_5	1.6 : 0.05 : 2.4	40 : 5 : 100	2 : 1 : 6	1
M_6, M_7	1.2 : 0.05 : 1.8	40 : 5 : 100	2 : 1 : 6	1
M_8, M_9	0.8 : 0.05 : 1.2	40 : 5 : 100	2 : 1 : 6	1
M_{10}, M_{11}	0.8 : 0.05 : 1.2	40 : 5 : 100	2 : 1 : 6	1
M_{12}, M_{13}	0.8 : 0.05 : 1.2	40 : 5 : 100	2 : 1 : 4	1
M_{14}, M_{15}	1.2 : 0.05 : 1.8	40 : 5 : 100	2 : 1 : 4	1
Figure devices	Passive parameter		Range or fixed value	
C_1, C_2	Matched capacitance (fF)		3 : 1 : 15	

MOS devices in the same row share W_{fg}, L, and N_f. The two capacitors share one capacitance variable.

Table 9: LDO device sizing ranges and fixed values: 26 independent parameters.

Figure devices	W_{fg} (μm)	L (nm)	N_f	M
M_1, M_2	0.4 : 0.025 : 2	270 : 5 : 1200	1 : 1 : 64	1
M_3, M_4	0.4 : 0.025 : 2	270 : 5 : 1200	1 : 1 : 30	1
M_5, M_{11}	0.4 : 0.025 : 1.2	270 : 5 : 1500	25 : 25 : 150	1
M_6	0.4 : 0.025 : 1.2	270 : 5 : 1500	2 : 2 : 12	1
M_7	0.4 : 0.025 : 1.2	270 : 5 : 1500	32 : 32 : 192	1
M_8	0.4 : 0.025 : 4	270 : 5 : 700	1 : 1 : 30	1
M_9, M_{10}	0.4 : 0.025 : 1.2	270 : 5 : 1500	1 : 1 : 8	1
M_{12}	0.4 : 0.025 : 4	270 : 5 : 700	1 : 1 : 30	1
M_{13}	0.4 : 0.025 : 1.2	270 : 5 : 1500	6 : 6 : 48	1
M_{14}	0.4 : 0.025 : 4	270 : 5 : 700	1 : 1 : 30	1
M_{15}	2 : 0.025 : 5	270 : 5 : 550	16 : 1 : 48	300
Figure devices	Passive parameter		Range or fixed value	
C_1	Capacitance (fF)		80 : 5 : 500	
R_1	Resistance (kΩ)		30 : 0.1 : 250	
R_2	Resistance (kΩ)		150 (fixed)	
R_3	Resistance (kΩ)		300 (fixed)	

M_{13} shares width and length with M_{10}, with $N_{f,13} = 6N_{f,10}$. M_5, M_7, and M_{11} share width and length with M_6, with $N_{f,5} = N_{f,11} = 12.5N_{f,6}$ and $N_{f,7} = 16N_{f,6}$. Thus $N_{f,6}$ uses even values. For the pass device M_{15}, the parallel multiplier is fixed at 300. C_1 and R_1 set compensation; R_2 and R_3 form the fixed feedback divider.

D.3 Baselines and ablations

tSS-BO (Gu et al., 2024) and TuRBO (Eriksson et al., 2019) use their native initialization without LLM involvement; in our evaluation, both use the same phased evaluator as TRACE. ADO-LLM (Yin et al., 2024) combines GP/qEI proposals with LLM-authored candidates; LEDRO (Kochar et al., 2025) uses TuRBO within LLM-proposed parameter ranges, informed by candidate metrics and device operating-point feedback. Our adapters map absolute parameter vectors, simulation submissions, and returned metrics to our custom-built MCP interfaces, using the same Virtuoso/Spectre backend and PDK. Candidates must respect the accepted parameter grid and linked geometry. Both methods evaluate all required testbenches and corners for every candidate without phased early stopping; they do not reuse TRACE's partial evaluations.

ADO-LLM starts with five independently authored candidates. Each subsequent round combines four GP/qEI candidates and one LLM candidate. LEDRO independently initializes its local TuRBO search. All LLM calls in the main comparison, including those in ADO-LLM and LEDRO, use GPT-6 astra at the same medium reasoning effort. These are adapted implementations with independent initialization, not exact reproductions of the original execution environments. The scalar objective is the negative sum of returned weighted losses; informational operating-point outputs are kept separate from the objective.

Table 10: BGR device sizing ranges and fixed values: 31 independent parameters.

Figure devices	W_{fg} (μm)	L (nm)	N_f	M
M_1, M_5	$0.55 : 0.025 : 1.1$	$750 : 25 : 1500$	$1 : 1 : 4$	1
M_2, M_6	$0.8 : 0.025 : 1.3$	$750 : 25 : 1500$	$1 : 1 : 4$	1
M_3, M_7	$0.45 : 0.025 : 0.95$	$1000 : 25 : 2000$	$1 : 1 : 4$	1
M_4, M_8	$0.6 : 0.025 : 1.3$	$1000 : 25 : 2000$	$1 : 1 : 4$	1
M_9, M_{16}	$0.5 : 0.025 : 2$	$1400 : 25 : 2000$	$1 : 1 : 4$	1
M_{10}, M_{17}	$0.5 : 0.025 : 2$	$1400 : 25 : 2000$	$1 : 1 : 4$	1
M_{11}	0.5	1000	1	1
M_{12}, M_{13}	0.27	2000	1	1
M_{14}	1	2000	2	1
M_{15}, M_{18}, M_{19}	1	2000	1	1
$M_{20}, M_{21}, M_{23}, M_{25}$	$0.5 : 0.025 : 1.1$	$1000 : 25 : 2000$	$1 : 1 : 4$	1
M_{22}, M_{24}, M_{26}	$0.8 : 0.025 : 1.3$	$1000 : 25 : 2000$	$1 : 1 : 4$	1
M_{27}	$0.6 : 0.025 : 1.2$	$1000 : 25 : 2000$	$1 : 1 : 4$	1
M_{28}, M_{29}	$0.45 : 0.025 : 1$	$1000 : 25 : 2000$	$6 : 1 : 14$	1
M_{30}, M_{31}	1	550	12	1
M_{32}, M_{33}	1	550	1	1
M_{34}	1	1000	1	1
M_{35}	1	550	10	1
M_{36}	0.4	1000	1	1

Figure devices	Passive parameter	Range or fixed value
C_1	MOM capacitor finger count	$18 : 1 : 72$
C_2	Capacitance (fF)	10 (fixed)
R_1	Poly-resistor series segments	34 (fixed)
R_2	Poly-resistor series segments	357 (fixed)

MOS rows containing only single values are fixed schematic geometry; grouping identical fixed values does not introduce a matching constraint. Other multi-device rows share all three sizing variables. All MOS parallel multipliers are fixed at one. For C_1, finger length is 1 μm, width 50 nm, spacing 70 nm, and multiplier 5; the PDK computes capacitance from geometry. C_2 is the startup capacitor. Both resistors use 5 μm-long, 1 μm-wide segments with one parallel branch; nominal resistances are approximately 100.91 kΩ and 1.05956 MΩ, respectively.

LEDRO's original protocol uses 200 full-space evaluations before the first LLM range proposal, followed by 100 evaluations per proposed region for ten rounds. On every circuit, our adapter uses 25 initial search candidates, including 20 Latin-hypercube points (McKay et al., 1979) (Table 11). Each subsequent region receives 15 evaluations. Subsequent regions use five Latin-hypercube points and ten TuRBO points in the absence of a trust-region restart, with batches of five throughout.

The original 200/100-evaluation schedule is impractical under our simulator cost, so we reduce the per-stage allocations to make complete baseline runs tractable. This adaptation invokes LLM-guided range adaptation earlier and reduces the number of full-space evaluations required before the first intervention. All adapted LEDRO runs nevertheless reach verified feasible designs.

Table 11: LEDRO candidate allocations. Every circuit uses 20 initial LHS points and 15 candidates per subsequent region.

Circuit	Initial search	Initial LHS	Per subsequent region
OTA	25	20	15
COMP	25	20	15
LDO	25	20	15
BGR	25	20	15

Three ablations examine the contributions within TRACE. **Initial design + BO** runs BO from the LLM-authored initial design without interventions. **Refine only** and **Explore only** retain the rest of TRACE and enable only the named intervention.

D.4 Interpreting the intervention ablations

In Table 2, the lower-cost single-action variant is Explore only on LDO and Refine only on BGR. Relative to these variants, TRACE reduces mean simulation count by 9.8% and 15.3%, and mean wall time by 12.6% and 8.8%, respectively.

This reversal is consistent with different local search conditions. In one LDO episode, two Refine proposals failed to resolve the tradeoff between PSRR and heavy-load phase margin. The agent then used Explore to propose five coordinated changes spanning the amplifier, buffer, pass device, and compensation network. Its third candidate passed the final specification and device checks. That candidate was assigned to an already visited region, illustrating that an Explore batch can produce a useful design without every successful candidate being region-novel. The batch still met its overall novelty requirement.

In one BGR episode, three successive Refine proposals adjusted the input pair and tail-current device while preserving the reference structure. PSRR at 1 kHz in SS125 rose from 67.11 to 68.46, 69.65, and 70.63 dB, and the last proposal passed final verification. These traces illustrate coordinated search and sustained local improvement, respectively; they do not identify the cause of the five-run ablation difference. The BGR examples in Figure 3 also benefit from Explore before local refinement, so the observations do not imply that exploration is generally unnecessary for BGR. This comparison does not separate Explore's search policy from its five-point batch size.

D.5 Evaluation protocol and cost accounting

For every circuit and method, Table 2 reports mean $\pm$ standard deviation over five runs using seeds 1, 2, 3, 4, and 5. Each run continues until a feasible design is verified against all required specifications and device operating-region constraints over every declared testbench and corner; all runs reach this termination condition. TRACE and its ablations begin from the initial design of Section 3.3, while the BO baselines use their native initialization. Initial observations therefore need not match across methods, and a shared seed does not imply identical LLM-authored candidates. Simulation counts are displayed as integers and wall times to one decimal place; ratios and percentage reductions are computed before display rounding.

We report the mean number of executed testbench–corner simulations $\bar{S}$ and the mean end-to-end episode wall time $\bar{T}$. A candidate is not a common unit of cost across methods. TRACE, its ablations, tSS-BO, and TuRBO evaluate candidates in phases and may stop an unsuccessful candidate before the remaining analyses are executed, whereas adapted ADO-LLM and LEDRO execute all required analyses for each candidate. The orchestration cadences also differ: adapted ADO-LLM inserts one LLM-authored candidate after every four GP/qEI candidates, while TRACE has no fixed BO-to-LLM ratio and allows BO to continue asynchronously between evidence-conditioned decisions. Candidate count therefore mixes differences in evaluation coverage and orchestration schedule; $\bar{S}$ instead records the testbench–corner jobs actually executed, while $\bar{T}$ measures their resulting end-to-end cost.

We retain the complete-observation evaluation interface of baselines that do not define partial candidate observations rather than introducing TRACE's phased screening into them. Early termination would leave part of the objective, constraint, or metric set unobserved and would require an additional rule for constructing the observation returned to the optimizer. Such a rule would alter the baseline rather than merely its simulation schedule. In our adapted ADO-LLM and LEDRO implementations, each candidate therefore evaluates all required testbenches and corners, while TRACE may retain partial evaluations for diagnosis without treating them as evidence of feasibility.

The external comparisons in Table 2 should consequently be interpreted as comparisons of **complete sizing-system cost**, including each method's search, intervention, and evaluation behavior, rather than proposal efficiency in isolation. The TRACE ablations serve a different purpose: Initial design + BO, Refine only, Explore only, and full TRACE share the same phased evaluator and accounting rules, so their comparison quantifies the cost differences associated with online interventions and their orchestration.

Both cost metrics cover the complete episode, from initialization through final verification, including interventions, failed evaluations, and work across task revisions. Simulation count includes only the testbench–corner jobs actually executed. Wall time additionally includes LLM reasoning and waiting through acceptance. The OTA run that reaches acceptance during the initial-design stage without starting BO is included in all reported means.

D.6 RESTRICTED SETUPS AND RECOVERY EVIDENCE

We report one recovery run per circuit, using initial configurations distinct from the main-comparison settings in Appendix D.2. Table 12 reports their initial and repaired parameter ranges. Grid reductions are relative to the repaired ranges, not the main-comparison ranges. For each repaired parameter, the excluded fraction is $1-n_{\text{initial}}/n_{\text{repaired}}$, counting discrete grid values, including retained defaults and explicit samples. These percentages concern individual parameter grids, not the joint search space or its feasible subset. The control uses BO alone, with Refine, Explore, and Revise disabled. It found no verified feasible design on any of the four restricted configurations within five hours. This does not establish that the restricted spaces are infeasible.

Table 12: Restricted and repaired parameter ranges. Counts describe the saved discrete grids; percentages are rounded to the nearest integer.

Circuit	Parameter	Restricted	Repaired	Grid counts	Removed
OTA	Compensation capacitance (pF)	1.5–4.5	0.15–4.5	121/175	31%
COMP	Input-pair length (nm)	100–200	40–200	21/33	36%
LDO	Pass-device finger width (μm)	0.4–0.6	0.4–5	9/185	95%
	Parallel finger count	1–5	1–12	5/12	58%
BGR	Capacitor finger count	18–72	8–72	55/65	15%
	$M_{20}, M_{21}, M_{23}, M_{25}$ finger width (nm)	270–400	270–800	8/25	68%

OTA. Reducing compensation capacitance from 3 to 1.5 pF raises nominal GBW from 12.15 to 24.20 MHz and reduces settling time from 476 to 235 ns. Increasing current instead violates the phase-margin and power specifications. These probes support extending the capacitance range downward, although they do not establish feasibility before simulation.

COMP. The comparator consistently exceeds the delay specification with the restricted input-pair lengths. Probes with shorter lengths progressively reduce delay, but it remains above the limit even at the 100 nm lower bound. This trend motivates extending the input-pair length range down to the PDK minimum of 40 nm to search for a feasible design. The revision retains the 130 nm default and symmetry constraints; the evidence does not justify shortening other device groups. The final candidate passes all 15 specification checks across the three corners.

LDO. PSRR is broadly satisfactory at 1 mA but deteriorates severely at 300 mA. This load-dependent failure suggests that the main PMOS pass device may have insufficient current-driving capability, limiting supply rejection under heavy load. Recovery proceeds through two revisions, first expanding the pass-device finger-width range and then its finger-count range. With the parallel multiplier fixed at 300, increasing the allowed finger count provides additional effective device width to test whether greater drive capability can meet the heavy-load PSRR requirement. After both revisions, the final design satisfies all required specifications and device operating-region constraints.

BGR. The width restriction applies to the matched current-mirror group $M_{20}, M_{21}, M_{23}, M_{25}$ in Figure 5. Controlled probes improve PSRR at the compensation lower bound and line regulation at the mirror-width upper bound, motivating expansion of both ranges. The second revision also corrects interchanged display labels for line regulation and quiescent current, without changing the measurements or acceptance criteria. Quiescent current is recorded for diagnosis only. BO resumes under the revised circuit settings using the previous observations. Its final candidate passes all 15 scalar specification checks and nine device operating-region checks.

Record interpretation and cost accounting. The completed BGR recovery episode uses 199 candidates and 78.10 minutes, including both revisions. The ratios in Table 3 compare each recovery episode with the unrestricted TRACE mean for the same circuit and are descriptive comparisons across unpaired runs. In particular, LDO's time ratio of 0.85 compares one recovery episode with the unrestricted mean; it does not establish that restrictive settings or task revision accelerate sizing. The unrestricted TRACE candidate means used as denominators are 63, 43, 84, and 76 for OTA, COMP, LDO, and BGR, respectively. The wall-time denominators from Table 2 are 32.67, 18.12, 60.38, and 65.74 minutes in the same circuit order. Candidate ratios count parameter assignments, whereas the simulation counts in Table 2 count testbench–corner jobs.