

ADO-LLM: Analog Design Bayesian Optimization with In-Context Learning of Large Language Models

Yuxuan Yin[‡], Yu Wang[‡], Boxun Xu, Peng Li*

{y_yin,yu95,boxunxu,li}@ucsb.edu

Department of Electronic and Computer Engineering, University of California, Santa Barbara
California, USA

ABSTRACT

Analog circuit design requires substantial human expertise and involvement, which is a significant roadblock to design productivity. Bayesian Optimization (BO), a popular machine-learning-based optimization strategy, has been leveraged to automate analog design given its applicability across various circuit topologies and technologies. Traditional BO methods employ black-box Gaussian Process surrogate models and optimized labeled data queries to find optimization solutions by trading off between exploration and exploitation. However, the search for the optimal design solution in BO can be expensive from both a computational and data usage point of view, particularly for high-dimensional optimization problems. This paper presents ADO-LLM, the first work integrating large language models (LLMs) with Bayesian Optimization for analog design optimization. ADO-LLM leverages the LLM’s ability to infuse domain knowledge to rapidly generate viable design points to remedy BO’s inefficiency in finding high-value design areas specifically under the limited design space coverage of the BO’s probabilistic surrogate model. In the meantime, sampling of design points evaluated in the iterative BO process provides quality demonstrations for the LLM to generate high-quality design points while leveraging infused broad design knowledge. Furthermore, the diversity brought by BO’s exploration enriches the contextual understanding of the LLM and allows it to more broadly search in the design space and prevent repetitive and redundant suggestions. We evaluate the proposed framework on two different types of analog circuits and demonstrate notable improvements in design efficiency and effectiveness.

CCS CONCEPTS

• **Hardware** → **Electronic design automation.**

ACM Reference Format:

Yuxuan Yin[‡], Yu Wang[‡], Boxun Xu, Peng Li*. 2024. ADO-LLM: Analog Design Bayesian Optimization with In-Context Learning of Large Language Models. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD ’24)*, October 27–31, 2024, New York, NY, USA. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3676536.3676816>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ICCAD ’24, October 27–31, 2024, New York, NY, USA

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-1077-3/24/10

<https://doi.org/10.1145/3676536.3676816>

1 INTRODUCTION

Analog circuit sizing is a critical yet challenging task in electronic design automation, characterized by a vast and intricate design space that requires extensive human expertise and involvement. The multi-objective nature of the design process introduces further complications. Designers need to find a delicate balance among competing objectives such as power efficiency, area minimization, and performance maximization. Moreover, each shift in circuit topology or technology requires a reevaluation of established design principles, adding to the complexity of achieving optimal designs.

Bayesian Optimization (BO) has emerged as a powerful machine learning tool for tackling analog circuit sizing due to its ability to efficiently navigate large and complex search spaces. BO employs a principled probabilistic surrogate model, typically a Gaussian process (GP), to estimate the performance of various circuit design points and relies on the minimization of a well-defined acquisition function to propose new design points, which balances the exploration and exploitation in the search space. The new design points are then simulated to obtain performance metrics that provide additional labeled training data to refine the surrogate model in an active learning manner. This method has been notably advanced in recent lines of work such as [14], [7], and [15].

Despite these advancements, BO is not without its limitations. Firstly, as a black-box optimization method, it intrinsically lacks domain-specific analog design knowledge and merely searches for the best mathematically defined figure-of-merit (FOM). It does so without exploring other informative feedback provided by the circuit simulator such as transistor regions of operation that can shed light on key circuit structural properties underlying optimized FOM and robustness of the circuit. As such, it does not exploit all available opportunities in the iterative design process for fast design convergence. Secondly, BO is well suited for single-objective optimization tasks but can struggle with the multi-objective nature of analog circuit sizing, often failing to capture the nuanced trade-offs between competing design objectives required for optimal solutions. While these limitations are discussed within the framework of Bayesian optimization, they are indicative of a broader challenge faced by many machine learning models. Specifically, they highlight the inherent difficulty in effectively integrating a broad range of design knowledge when the models are trained solely on a limited amount of domain-specific data.

Large Language Models (LLMs) present great promise in addressing the above challenge and may be well-positioned to augment BO’s capabilities in analog circuit sizing tasks. We argue that the

[‡]Equal contribution.

*Corresponding author.

power of LLMs in circuit design originates from two key capabilities: the extensive prior knowledge embedded in their pre-training data and their ability to perform in-context learning that can be utilized to incorporate domain expertise. This allows LLMs to suggest design modifications and innovations with a high degree of relevance. Typically, the application of LLMs in this context follows an iterative loop, where previous design examples are demonstrated to the LLM, prompting it to generate new, potentially optimal design points. These points are then simulated, and the outcomes are used to further enrich the LLM's training examples [11]. However, this few-shot generation process has limitations. The quality of optimization heavily depends on both the intrinsic capabilities of the LLM and the quality of the input demonstrations. Moreover, LLMs tend to produce solutions that closely mimic the provided examples, showing a hesitance to extrapolate beyond the demonstrated examples and explore new design areas.

In this paper, we introduce a novel unified framework named ADO-LLM, which combines Large Language Models (LLMs) with Bayesian Optimization (BO) to leverage the strengths of both methods while overcoming their respective limitations. ADO-LLM comprises three primary components: a standard Gaussian Process-based Bayesian Optimization (GP-BO) proposer, an LLM agent, and a high-quality data sampler. The GP-BO proposer operates on the entirety of a collected dataset consisting of all design points evaluated so far, proposing new points for additional evaluation. In the meantime, the LLM agent focuses on a subset of high-quality demonstrations sampled from this shared dataset to generate new design suggestions. These proposed design points are then evaluated using a circuit simulator and added to the dataset, following the standard iterative process of BO.

Within the ADO-LLM framework, BO benefits from the LLM's ability in infusing domain knowledge to rapidly generate viable design points that improve the quality of the shared dataset. This remedies BO's inefficiency in finding high-value design areas during exploration, specifically when the scarcity of labeled training data limits the design space coverage of the BO's probabilistic GP surrogate model. With the aid of the LLM, BO continues to explore promising yet under-explored areas using the GP surrogate, proposing diverse new design points to be evaluated. Concurrently, the BO helps provide quality demonstrations sampled from the shared common dataset to the LLM; this quality assurance in demonstrations provides a basis for the LLM to generate high-quality design points while leveraging infused broad design knowledge. Furthermore, the diversity of the common dataset contributed by BO enriches the contextual understanding of the LLM and allows it to more broadly search in the design space and prevent repetitive and redundant suggestions.

We evaluate the proposed framework on two different types of analog circuits and demonstrate notable improvements in design efficiency and effectiveness. By combining the exploratory capabilities of BO with the rich contextual knowledge of LLMs, our approach sets a new benchmark for automated analog circuit design, promising significant reductions in design time and improvements in outcome predictability.

2 BACKGROUND

2.1 Bayesian Optimization in Circuit Sizing

Bayesian optimization (BO) [2] is a powerful optimization algorithm to search the global optimum of a black-box function. For analog circuit sizing tasks, BO maximizes a figure of merit (FOM) function:

$$x^* = \underset{x}{\operatorname{argmax}} FOM(x) \quad (1)$$

where $x \in \mathcal{X} \subset \mathbb{R}^d$ is a d -dimensional vector in a design space $\mathcal{X}$, $FOM : \mathbb{R}^d \rightarrow \mathbb{R}$ is a single value function that balances all circuit performance metrics, such as gain, unit gain frequency, etc.

BO leverages a probabilistic surrogate model to provide uncertainty quantification for the design space, and a corresponding acquisition function to trade off exploitation and exploration. In each iteration, BO determines the next query parameter set that maximizes the acquisition function, and then updates the surrogate model with the new simulation. Typically the surrogate is a Gaussian process (GP) [16]. The circuit design flow of BO is summarized in Algorithm 1.

Algorithm 1 Bayesian Optimization for Analog Circuit Design

- 1: **Input:** Initial data size N_{init} , number of iterations N_{iter}
 - 2: $x_{init} \leftarrow$ Randomly sample N_{init} points from the design space $\mathcal{X}$
 - 3: $FOM_{init} \leftarrow$ Simulate the circuit with parameters x_{init}
 - 4: Fit an initial surrogate model $\mathcal{GP}$ on (x_{init}, FOM_{init})
 - 5: **for** $t = 1$ **to** N_{iter} **do**
 - 6: Select the next point for simulation via maximizing the acquisition function $\alpha: x_t \leftarrow \arg \max_x \alpha(x, \mathcal{GP})$
 - 7: $FOM_t \leftarrow$ Simulate the circuit with parameters x_t
 - 8: Update the surrogate model $\mathcal{GP}$ with (x_t, FOM_t)
 - 9: **end for**
 - 10: **return** The best observed point and value (x^*, FOM^*)
-

2.2 Practice of Large Language Model in Analog Circuit Design

Trained by the next token generation, Large language models (LLMs) have emergent abilities in many downstream tasks [21]. Recent research [11] shows that LLMs have prior knowledge about analog circuits, and the text box below demonstrates that LLMs understand codes of circuit netlists.

It is promising to leverage LLMs' prior knowledge about analog circuits for transistor sizing. The authors of [11] design the first LLM agent LADAC for analog parameter design. They integrate two techniques: 1) in-context learning [4] for extracting information from design examples and a local design knowledge library, and 2) chain-of-thought [22] for decision-making. Based on GPT-4, LADAC successfully discovers good circuit parameter sets of 3 analog circuits satisfying design specifications.

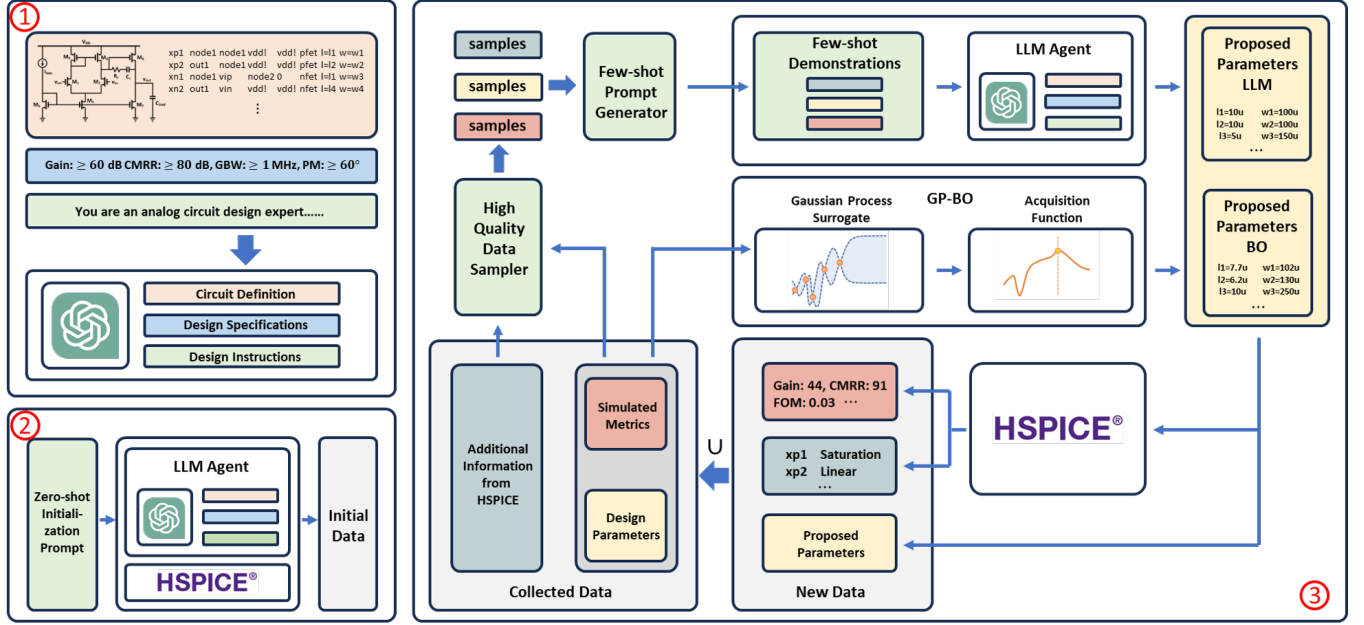


Figure 1: Overview of the ADO-LLM framework. Top-left: the LLM agent is initialized with the circuit definition, the design specifications, and the design instructions. Bottom-left: the LLM leverages the inherent domain knowledge to generate viable initial design points. Right: the optimization flow of ADO-LLM in each iteration.

LLM's Understanding with Circuit Definition

User: Analyze the structure of the following netlist of a two-stage differential amplifier:

...

```

xn1  n1  v1  n2  0    nfet  l=l1  w=w1
xn2  o1  v2  n2  0    nfet  l=l1  w=w1
xp1  n1  n1  vdd! vdd! pfet  l=l2  w=w2
xp2  o1  n1  vdd! vdd! pfet  l=l2  w=w2

```

...

ChatGPT:

...

Stage One: The first stage is a differential amplifier with a current mirror as the load. Here's how it's structured:

1. *Differential Pair:* The first stage consists of transistors xn1 and xn2, which are N-channel MOSFETs forming a differential pair...

2. *Current Mirror Load:* xp1 and xp2 are P-channel MOSFETs forming a current mirror load....

...

2.3 BO and LLM for Circuit Sizing: Pros and Cons

BO has been extensively utilized for its ability to efficiently navigate complex design spaces, employing probabilistic GP surrogate

models to balance exploration and exploitation. This balance is achieved through sophisticated acquisition functions, which prioritize areas of uncertainty that might yield high returns. Despite these strengths, BO's application in circuit sizing often faces challenges due to its generic approach, which lacks the incorporation of domain-specific knowledge. This deficiency can lead to a prevalence of non-viable solutions, i.e., parameters failed for success simulation. This phenomenon exacerbates in larger or more complex design landscapes, where the absence of tailored guidance becomes markedly detrimental.

Concurrently, LLMs have started to play a pivotal role in EDA by leveraging their vast reservoirs of encoded prior knowledge. These models apply advanced language understanding and generation capabilities to mimic expert-level decision-making processes. Through the use of few-shot learning and in-context learning techniques [4], LLMs can quickly generate initial design points that are both innovative and feasible. However, their dependency on the quality and scope of the training data, as well as the few shot demonstration, often restricts the models to optimizing within a narrow region around provided examples, leading to potential sub-optimality and poor exploration of broader design spaces.

To summarize, while BO excels in systematic exploration across a wide range of potential solutions, it lacks the intuitive, knowledge-driven insights provided by LLMs. Conversely, LLMs offer rapid generation of viable design points based on learned data and expert patterns but are constrained by their limited ability to generalize beyond familiar scenarios. This motivates us to propose a novel cooperative approach that synergies the predictive power of BO with the knowledge-driven capabilities of LLMs, aiming to harness

the strengths of each while mitigating their individual limitations, as detailed in the next section.

3 METHODS

3.1 Overview of the ADO-LLM Framework

The ADO-LLM framework integrates Large Language Models (LLMs) with Bayesian Optimization (BO) to address the complex multi-objective optimization challenges inherent in analog circuit sizing. This framework consists of four primary components: a Gaussian Process-based Bayesian Optimization (GP-BO) proposer, an LLM agent, a high-quality data sampler, and an HSPICE simulator, which evaluates the proposed design points.

Initially, the LLM agent is configured with circuit definition files and design specifications (see Figure 1 top-left). Then it utilizes a zero-shot approach to leverage its relevant analog design knowledge obtained from the vast amount of model pre-training data for generating initial design points (see Figure 1 bottom-left). These points, enriched with embedded domain insights, initiate an iterative loop where design points are proposed, evaluated via the simulator, and used to update the dataset, thereby enhancing the learning context for subsequent iterations. The overall architecture and optimization flow is illustrated in Figure 1.

3.2 The GP-BO Proposer

The Gaussian Process-based Bayesian Optimization (GP-BO) proposer is an integral part of the ADO-LLM framework, ensuring the diversity of newly generated design points. It employs a Gaussian Process (GP) with a Radial Basis Function (RBF) kernel as its surrogate model, fitted on the entire collected dataset to predict the Figure of Merit (FOM) across the extensive design space.

An acquisition function that maximizes expected improvement guides the GP-BO's search strategy, which balances between exploring new potential areas and exploiting regions with promising performance, following standard optimization practices [14]. This uncertainty-guided search ensures the diversity of the collected data and prevents the optimization process from becoming stuck at local optima.

3.3 The LLM Agent

The LLM agent is pivotal within the ADO-LLM framework and serves two key roles. Initially, it uses extensive pre-trained knowledge during the zero-shot initialization phase to suggest viable design points. In each subsequent optimization iteration, the agent utilizes a selectively chosen set of few-shot demonstrations—identified for their high Figure of Merit (FOM) from the diversely collected dataset—as well as domain-specific knowledge, to generate innovative design solutions.

The effectiveness of the LLM agent hinges on three critical components: the inherent domain expertise of the selected model, the quality of in-context learning designed to incorporate human expertise, and the quality of the demonstration data. The rationale behind the design choices for each of these components is further elaborated below:

3.3.1 Model Choice: Currently, there is no suitable LLM specifically tailored for analog circuit design. Adapting or pretraining

an LLM for this specialized field requires substantial data, and additional fine-tuning would be necessary to refine its ability to follow complex instructions. Given these challenges, we selected ChatGPT-3.5 Turbo as the backbone of the LLM agent, following the successful application of the ChatGPT API in other LLM-based EDA research [5].

3.3.2 In-context Learning: Circuit sizing is inherently a complex task that typically requires multiple steps of reasoning. In ADO-LLM, we follow the chain of thoughts [22] to prompt the LLM to decompose the problem into multiple steps:

a. Interpreting the Circuit Definition: Initially, the LLM is prompted to read and explain the given circuit definition, detailing the role of each transistor. Since ChatGPT-3.5 lacks specific domain knowledge, we incorporate human annotations in the prompt to minimize misinterpretations and potential error accumulation.

b. Balancing Trade-offs in the Design Specifications: We prompt the LLM to describe how to meet all design specifications for the interpreted circuit and explain the trade-offs between each objective. This step helps articulate the design knowledge pre-embedded within the large language dataset used for training of LLMs.

c. Providing Few-shot Examples with Diverse Simulation Results: Few-shot examples are provided to guide the LLM toward generating relevant and focused responses. In ADO-LLM, these examples include not only design points and simulated metrics but also higher-level feedback from simulators, such as transistor operational regions. This approach enables the LLM to base decisions on comprehensive data rather than just numerical outputs.

d. Injecting Human Expertise with Design Principles: We also utilize additional instructions termed “design principles” to improve the generation quality of the LLM agent. The adopted design principles instruct the LLM to ensure that all transistors operate within specific desired regions when proposing design points. These principles help the LLM utilize high-level simulator feedback effectively, while also preventing it from merely regurgitating trivial solutions from its training data.

Finally, the LLM is prompted to generate a new design point following the thought of the previous steps with the given format and parameter range constraints. We adopt a parser that ensures the correct formatting of the response and requests the LLM to regenerate if the requirements are not satisfied.

3.3.3 High-Quality Data Sampler: The diversity of design points has been enhanced by the GP-BO proposer. Within ADO-LLM, however, we selectively use only a subset of the full dataset as few-shot examples for the LLM agent. This selection is facilitated by a high-quality data sampler that identifies and samples the top-performing demonstrations based on the top-k Figure of Merit (FOM). This method ensures the chosen examples are not only diverse—thereby preventing the LLM from converging on local optima—but also of high value, providing potent models for the LLM to emulate. Consequently, this strategic sampling empowers the LLM to generate relevant and innovative responses, leveraging the most effective designs to propose new design points for future evaluation.

4 EXPERIMENTS

Table 1: Hyper-parameters of the proposed ADO-LLM

Sub-model	Hyper-parameter	Value
LLM Agent	# Queries per Step	1
	# Top Examples in Context	5
	LLM Version	GPT 3.5
	Temperature	0.5
	Context Window Length	16k
	Max Token Generation Length	1,000
GP-BO	# Queries per Step	4
	Kernel Function	RBF
	Acquisition	qEI [3]
	Acquisition Optimizer	L-BFGS

We benchmark ADO-LLM on the sizing of two distinct and fundamental analog circuits: (1) a two-stage differential amplifier and (2) a hysteresis comparator. Initially, we demonstrate that the cooperative interaction between the LLM and Bayesian Optimization (BO) significantly enhances search efficiency compared to using either method alone. Subsequently, we conduct several ablation studies to analyze the effectiveness of different components within the system.

ADO-LLM Model Settings. ADO-LLM has two parallel optimization models: the GP-BO model and the LLM agent. The configurations of both models are presented in Table 1. ADO-LLM starts with 5 initial examples predicted by the LLM agent. In each data query iteration, we let the LLM agent propose 1 candidate, based on the designed prompt and top 5 data points; and let GP-BO provide 4 candidates from a qEI acquisition function [3], constructed over the posterior prediction of a GP surrogate.

Baseline Models. We compare ADO-LLM with two baselines: the GP-BO model with random initialization and a single LLM proposer that uses zero-shot initialization to generate starting points. The number of initial design data is 5 for both methods. To ensure a fair comparison, the total number of evaluations of each method is set to be the same as that of ADO-LLM.

4.1 Two-Stage Differential Amplifier

Experimental Settings. The two-stage differential amplifier is depicted in Figure 2. It has 18 parameters: the length and the width of 8 transistors (M_1 to M_7 , and M_b), the resistance of a feedback resistor (R_z), and the capacitance of a compensation capacitor (C_c). The number of free parameters is 14, where some transistors share the same size (M_1 and M_2 ; M_3 and M_4). The design space of circuit parameters is illustrated in Table 2. We use Synopsys HSPICE to simulate the performance of designed circuits, under a commercial 90 nm CMOS technology.

We aim to design a circuit that satisfies the specifications of several performance metrics, including gain-bandwidth product (GBW), gain, common-mode rejection ratio (CMRR), phase margin (PM), and power consumption. These specifications are listed in

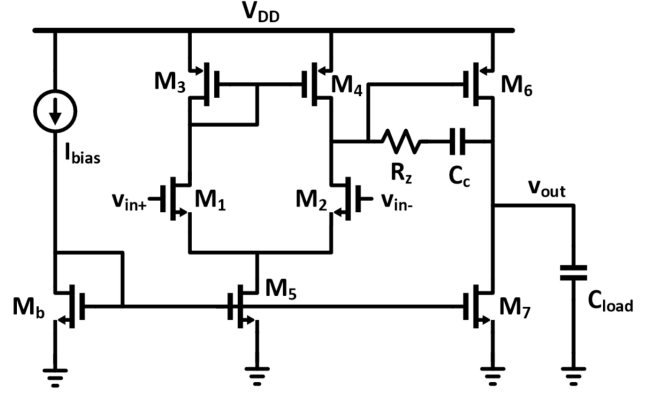


Figure 2: The circuit schematic of the two-stage differential amplifier

Table 2: Design configuration of a two-stage differential amplifier under a commercial 90nm CMOS technology

Attribute	Value
Transistor Width Range	[120nm, 50μm]
Transistor Length Range	[80nm, 1μm]
Resistance Range	[10Ω, 100kΩ]
Capacitance Range	[10fF, 100pF]
Gain-Bandwidth Product (Spec)	≥ 1MHz
Gain (Spec)	≥ 60dB
Common-Mode Rejection Ratio (Spec)	≥ 75dB
Phase Margin (Spec)	≥ 60°
Power Consumption (Spec)	≤ 30μW
Gain-Bandwidth Product (Norm)	[0MHz, 10MHz]
Gain (Norm)	[0dB, 60dB]
Common-Mode Rejection Ratio (Norm)	[0dB, 80dB]
Phase Margin (Norm)	[0°, 45°]
Power Consumption (Norm)	[0μW, 30μW]
Gain-Bandwidth Product (Failed)	-10MHz
Gain (Failed)	-60dB
Common-Mode Rejection Ratio (Failed)	-80dB
Phase Margin (Failed)	-180°
Power Consumption (Failed)	80μW

Table 2. To facilitate GP-BO for this multi-objective optimization task, we construct FOM as a linear combination of the considered metrics:

$$FOM = \widetilde{GBW}^b + \widetilde{Gain}^b + \widetilde{CMRR}^b + \widetilde{PM}^b - \widetilde{Power}^b \quad (2)$$

The operator $\widetilde{\cdot}$ is a function to check whether the specification is satisfied, and then to normalize the metric. It is defined as

$$\widetilde{m} := \begin{cases} \frac{m - m_{\min}}{m_{\max} - m_{\min}} & \text{if } m \text{ hits } m_{\text{spec}} \\ \frac{m_{\text{failed}} - m_{\min}}{m_{\max} - m_{\min}} & \text{otherwise} \end{cases} \quad (3)$$

Table 3: The performance of the best parameter set found by each method for the two-stage differential amplifier

Method	# Simulation Init. + Batch × Iter	Gain (dB) ≥ 60	CMRR (dB) ≥ 75	GBW (MHz) ≥ 1	PM (°) ≥ 60	Power (μW) ≤ 30	FOM	# Missed Spec.
GP-BO	5 + 5 × 20	27.12 (X)	106.08	4.63	97.76	27.02	1.89	1
GP-BO	5 + 5 × 80	63.09	79.57	7.22	94.28	65.76 (X)	2.10	1
LLM agent	5 + 1 × 100	60.35	73.04	93.09	87.02	39.00 (X)	0.20	1
ADO-LLM	5 + 5 × 20	60.83	78.38	1.35	92.29	19.79	3.52	0

for a metric m where the normalization range and the failed value of each metric are listed in Table 2.

The superscript b means the bounded value, i.e., a metric does not need to exceed far beyond its design specification. This bound is set to 2 for the normalized values of GBW, Gain, CMRR, and PM.

Results. For each method, we present its performance metrics, FOM, and number of missed specifications in Table 3. The best metric across all methods is bold, and a cross mark is appended to a metric when it fails to meet the corresponding design specification. Compared with GP-BO and LLM agent, ADO-LLM achieves the largest FOM, which is even better than that acquired by GP-BO after 4x iterations of simulation. More importantly, only ADO-LLM satisfies all design specifications. While the baselines GP-BO and the LLM agent excel in optimizing several circuit performance metrics, they fail to balance these metrics with a limited simulation budget. This result demonstrates the data efficiency of ADO-LLM in analog circuit design tasks.

4.2 Hysteresis Comparator

Experimental Settings. We aim to optimize a hysteresis Comparator in Figure 3 via adjusting the size of 12 transistors (M_1 to M_{11} and M_b). It has 12 free parameters to be determined, including 6 values of length and width of transistors. We use Synopsys HSPICE to simulate the performance of designed circuits, under a commercial 90 nm CMOS technology.

The metrics to be optimized are gain, unit gain frequency (UGF), Hysteresis Error (V_{hys_err}), voltage offset (V_{offset}), and power. We construct *FOM* as follows:

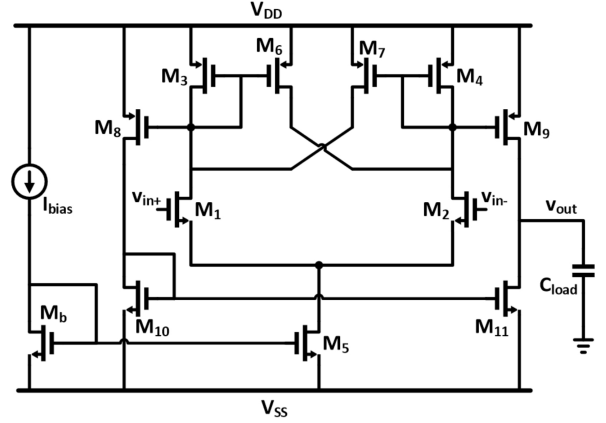
$$FOM = \widetilde{Gain}^b + \widetilde{UGF}^b - \widetilde{V_{hys_err}} - |\widetilde{V_{offset}}| - \widetilde{Power} \quad (4)$$

Table 4 presents our design specifications and configurations. The upper bound is set to 2 for the normalized values of Gain and UGF in our experiments.

Results. Table 5 shows the performance metrics, the FOM, and the number of missed design specifications of each optimization approach. The result of ADO-LLM in the comparator is consistent with that in the amplifier: it achieves the highest FOM, and only our method fulfills the specification demands.

4.3 The significance of Zero-Shot Initialization

We conduct an ablation study to demonstrate that the zero-shot initialization with the LLM proposer plays a vital role in generating a good starting point and hence accelerating the optimization process, compared with random initialization.

**Figure 3: The circuit schematic of a hysteresis comparator****Table 4: Design configuration of a hysteresis comparator under a commercial 90nm CMOS technology**

Attribute	Value
Transistor Width Range	[90nm, 200μm]
Transistor Length Range	[90nm, 1μm]
Gain (Spec)	≥ 60dB
Unit Gain Frequency (Spec)	≥ 10MHz
Absolute Hysteresis Error (Spec)	≤ 300mV
Absolute Voltage Offset (Spec)	≤ 20mV
Power Consumption (Spec)	≤ 150μW
Gain (Norm)	[0dB, 60dB]
Unit Gain Frequency (Norm)	[0mHz, 10MHz]
Hysteresis Error (Norm)	[0mV, 300mV]
Absolute Voltage Offset (Norm)	[0mV, 20mV]
Power Consumption (Norm)	[0μW, 150μW]
Gain (Failed)	-60dB
Unit Gain Frequency (Failed)	-10MHz
Hysteresis Error (Failed)	600mV
Absolute Voltage Offset (Failed)	40mV
Power Consumption (Failed)	300μW

Table 5: The performance of the best parameter set found by each method for a hysteresis comparator

Method	# Simulation Init. + Batch $\times$ Iter	Gain (dB) ≥ 60	UGF (MHz) ≥ 10	$V_{\text{hys_err}}$ (mV) ≤ 300	V_{offset} (mV) $ \cdot \leq 20$	Power (μW) ≤ 150	FOM	# Missed Spec.
GP-BO	5 + 5 $\times$ 20	55.47 (X)	8.42 (X)	199.42	-3.95	77.70	-3.38	2
GP-BO	5 + 5 $\times$ 80	30.18 (X)	10.10	186.33	3.55	94.31	-1.42	1
LLM agent	5 + 1 $\times$ 100	40.52 (X)	13.66	161.14	7.36	121.37	-1.35	1
ADO-LLM	5 + 5 $\times$ 20	60.83	12.04	159.83	-1.00	109.89	0.90	0

Table 6: The best design of GP-BO with different initializations

Method	Circuit	FOM	# Missed Spec.
GP-BO with random init.	Amp2	1.89	1
GP-BO with LLM's init.	Amp2	2.20	1
GP-BO with random init.	Comp	-3.38	2
GP-BO with LLM's init.	Comp	-1.64	1

Experimental Settings. We employed Gaussian Process-based Bayesian Optimization (GP-BO) for the design of the specified amplifier and comparator circuits. We compared two initialization strategies: uniform sampling from the entire search space and zero-shot initialization provided by the LLM proposer. We configured the experiments with 5 initial points, a batch size of 5 queries, and a total of 20 iterations.

Results. Table 6 shows the best FOM acquired by GP-BO, where the abbreviation Amp2 stands for the two-stage amplifier, and Comp stands for hysteresis comparator. The initial data from the LLM contributes to a better FOM in either circuit, which is even comparable with that acquired after 80 iterations, as listed in Table 3 and Table 5. This result demonstrates that the domain-specific prior knowledge of the LLM is able to provide good initialization parameter sets in potential high-value regions for analog circuit design tasks.

4.4 The Effectiveness of In-Context Learning and Proposed Sampler

The quality of few-shot demonstrations during in-context learning plays a critical role in eliciting high-quality responses from the LLM agent. We conducted experiments to validate the necessity and impact of these demonstrations.

Experimental Settings. We design two variants of the LLM agent: the first LLM agent is not provided with any demonstration. The second agent learns from few-shot demonstrations that are uniformly sampled from the collected dataset. We compare these two variants with the proposed LLM agent which is provided with few-shot demonstrations with the highest top-k FOM. For all three variants, the number of sampled demonstrations if there is any in each iteration is 5, and the total simulation evaluation budget is 105.

Results. Table 7 reports the best design example of the LLM agent variants in optimizing the amplifier and the comparator. Here the

Table 7: The best design of the LLM agent with different in-context examples

Method	Circuit	Sampler	FOM	# Missed Spec.
LLM agent w/o ICL	Amp2	NA	-1.78	3
LLM agent with ICL	Amp2	Rand 5	-0.63	2
LLM agent with ICL	Amp2	Top 5	0.20	1
LLM agent w/o ICL	Comp	NA	-2.29	2
LLM agent with ICL	Comp	Rand 5	-2.53	3
LLM agent with ICL	Comp	Top 5	-1.35	1

abbreviation ICL stands for in-context learning. While using random sampling to provide demonstrations improves the performance of the amplifier circuit, this strategy degrades the zero-shot capacity of the LLM agent when optimizing the comparator. With the high-quality design demonstrations provided by the proposed sampler, the LLM agent achieves superior performance in both the FOM and the number of specifications met, demonstrating the effectiveness of the in-context learning and our sampling strategy for analog circuit design.

5 RELATED WORK

5.1 ML-based Automated Analog Circuit Design

Machine learning has been a popular tool for analog circuit design for decades [17]. Past ML approaches primarily utilize either reinforcement learning (RL) [8] or Bayesian optimization (BO) [2, 18].

RL-based algorithms focus on training a design agent that outputs circuit parameters aimed at maximizing the ultimate reward. For instance, GCN-RL [20] employs a graph convolutional neural network (GCN) [10] to interpret circuit topology information, while Prioritized RL [9] introduces a non-uniform sampler to prioritize design parameters from potentially high-value areas.

On the other hand, BO-based methods balance the exploration and exploitation of the design space by optimizing an acquisition function over the Gaussian process [16] posterior predictions. Techniques like MACE [14] and pHCB0 [7] use ensembles of multiple acquisition functions. MACE selects query batches from the Pareto front computed by DEMO [15], and pHCB0 constructs batches from the optimal points of each acquisition function.

However, both BO and RL approaches face challenges in meeting multi-objective design specifications because they typically reduce the analog circuit design task to a single-objective global

optimization problem, focused on maximizing a Figure of Merit (FOM), a weighted linear combination of performance metrics. This reduction creates a dilemma: a simplistic FOM cannot adequately cover all design specifications, yet a complex FOM becomes too cumbersome to optimize effectively. Thus, the design of the FOM critically influences the overall performance of circuit sizing.

5.2 Large Language Models for Electronic Design Automation

The integration of large language models (LLMs) into the field of electronic design automation (EDA) has been a subject of considerable interest, exploring various methodologies and applications. Broadly, the utilization of LLMs in EDA can be classified into two distinct approaches: the use of externally hosted APIs and the development of domain-adapted LLMs specifically tailored for EDA tasks.

In terms of application diversity, significant contributions have been made across several key areas. First, the completion of Verilog code, where LLMs assist in generating and suggesting code segments to streamline the hardware design process, shows promise in enhancing productivity and reducing error rates [13, 19]. Second, in the design of accelerators, LLMs have been employed to optimize the architectural decisions, potentially leading to more efficient processing for specific tasks [1, 5]. Another noteworthy application is in the completion of EDA software code, which not only aids in software development but also ensures that tools are more adaptable to the needs of hardware engineers [6].

Further, the use of LLMs in debugging represents a significant shift towards more intelligent troubleshooting methods in circuit design, enabling quicker identification and rectification of errors in both hardware and software components [12]. Lastly, in the domain of analog circuit sizing, LLMs have demonstrated their utility by automating the adjustment of component sizes for optimal performance, thus facilitating more efficient design cycles [11].

6 LIMITATION AND FUTURE WORKS

In this paper, we proposed ADO-LLM, which integrates Bayesian Optimization (BO) with Large Language Models (LLMs) to enhance the efficiency of analog circuit sizing and has demonstrated promising results on two different analog circuit sizing problems. However, like all methodologies, it is not without limitations, which open avenues for further research and refinement.

1. Dependency on External LLM APIs: Currently, our system utilizes LLMs accessed via closed-source ChatGPT APIs, which can be costly and lack transparency. Furthermore, these models are not specifically tailored to the nuances of circuit design, potentially limiting their effectiveness and applicability. Future work could explore the development and integration of domain-adapted LLMs [12] that are specifically trained on electronic design automation tasks. Additionally, incorporating techniques like Retrieval-Augmented Generation (RAG) could enable the system to leverage more precise, domain-specific knowledge, enhancing the accuracy and relevance of the solutions generated.

2. Optimality and Scalability of GP in BO: The Gaussian Processes (GP) used in our current Bayesian Optimization (BO) setup effectively manage the design spaces of the circuits examined

in this study. However, they may not scale well for more complex designs. In practical applications, as design spaces expand, the computational overhead and the inefficacy of optimizing all parameters simultaneously become significant constraints. A promising direction for future research is to implement hierarchical optimization strategies, where only a subset of parameters is optimized in each iteration, mimicking the approach of human experts. This could potentially increase both the efficiency and reliability of the design process. We leave these interesting directions to be explored in the future.

7 ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grants No. 1956313 and No. 2334380.

REFERENCES

- [1] Jason Blocklove, Siddharth Garg, Ramesh Karri, and Hammond Pearce. 2023. Chip-Chat: Challenges and Opportunities in Conversational Hardware Design. In *2023 ACM/IEEE 5th Workshop on Machine Learning for CAD (MLCAD)*. IEEE. <https://doi.org/10.1109/mlcad58807.2023.10299874>
- [2] Eric Brochu, Vlad M Cora, and Nando De Freitas. 2010. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599* (2010).
- [3] Clément Chevalier and David Ginsbourger. 2013. Fast Computation of the Multi-Points Expected Improvement with Applications in Batch Selection. In *Learning and Intelligent Optimization*, Giuseppe Nicosia and Panos Pardalos (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 59–69.
- [4] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234* (2022).
- [5] Yonggan Fu, Yonggan Zhang, Zhongzhi Yu, Sixu Li, Zhifan Ye, Chaojian Li, Cheng Wan, and Yingyan Lin. 2023. GPT4AIGChip: Towards Next-Generation AI Accelerator Design Automation via Large Language Models. *arXiv:2309.10730* [cs.LG]
- [6] Zhuolun He, Haoyuan Wu, Xinyun Zhang, Xufeng Yao, Su Zheng, Haisheng Zheng, and Bei Yu. 2024. ChatEDA: A Large Language Model Powered Autonomous Agent for EDA. *arXiv:2308.10204* [cs.AR]
- [7] Hanbin Hu, Peng Li, and Jianhua Z Huang. 2018. Parallelizable Bayesian optimization for analog and mixed-signal rare failure detection with high coverage. In *2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. IEEE, 1–8.
- [8] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. 1996. Reinforcement learning: A survey. *Journal of artificial intelligence research* 4 (1996), 237–285.
- [9] N.S. Karthik Somayaji, Hanbin Hu, and Peng Li. 2021. Prioritized Reinforcement Learning for Analog Circuit Optimization With Design Knowledge. In *2021 58th ACM/IEEE Design Automation Conference (DAC)*. 1231–1236. <https://doi.org/10.1109/DAC18074.2021.9586189>
- [10] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [11] Chengjie Liu, Yijiang Liu, Yuan Du, and Li Du. 2024. LADAC: Large Language Model-driven Auto-Designer for Analog Circuits. *Authorea Preprints* (2024).
- [12] Mingjie Liu, Teodor-Dumitru Ene, Robert Kirby, Chris Cheng, Nathaniel Pinckney, Rongjian Liang, Jonah Alben, Himyanshu Anand, Sanmitra Banerjee, Ismet Bayraktaroglu, Bonita Bhaskaran, Bryan Catanzaro, Arjun Chaudhuri, Sharon Clay, Bill Dally, Laura Dang, Parikshit Deshpande, Siddhanth Dhodhi, Sameer Halepete, Eric Hill, Jia Shang Hu, Sumit Jain, Ankit Jindal, Bruce Khailany, George Kokai, Kishor Kunal, Xiaowei Li, Charley Lind, Hao Liu, Stuart Oberman, Sujeet Omar, Ghasem Pasandi, Sreedhar Pratty, Jonathan Raiman, Ambar Sarkar, Zhengjiang Shao, Hanfei Sun, Pratik P Suthar, Varun Tej, Walker Turner, Kaizhe Xu, and Haoxing Ren. 2024. ChipNeMo: Domain-Adapted LLMs for Chip Design. *arXiv:2311.00176* [cs.CL]
- [13] Mingjie Liu, Nathaniel Pinckney, Bruce Khailany, and Haoxing Ren. 2023. VerilogEval: Evaluating Large Language Models for Verilog Code Generation. *arXiv:2309.07544* [cs.LG]
- [14] Wenlong Lyu, Fan Yang, Changhao Yan, Dian Zhou, and Xuan Zeng. 2018. Batch Bayesian optimization via multi-objective acquisition ensemble for automated analog circuit design. In *International conference on machine learning*. PMLR, 3306–3314.
- [15] Miha Mlakar, Dejan Petelin, Tea Tušar, and Bogdan Filipič. 2015. GP-DEMO: differential evolution for multiobjective optimization based on Gaussian process models. *European Journal of Operational Research* 243, 2 (2015), 347–361.

- [16] Carl Edward Rasmussen. 2004. Gaussian Processes in Machine Learning. In Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures, Olivier Bousquet, Ulrike von Luxburg, and Gunnar Rätsch (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 63–71. https://doi.org/10.1007/978-3-540-28650-9_4
- [17] Rob A. Rutenbar, Georges G. E. Gielen, and Jaijeet Roychowdhury. 2007. Hierarchical Modeling, Optimization, and Synthesis for System-Level Analog and RF Designs. *Proc. IEEE* 95, 3 (2007), 640–669. <https://doi.org/10.1109/JPROC.2006.889371>
- [18] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando de Freitas. 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* 104, 1 (2016), 148–175. <https://doi.org/10.1109/JPROC.2015.2494218>
- [19] Shailja Thakur, Baleegh Ahmad, Zhenxing Fan, Hammond Pearce, Benjamin Tan, Ramesh Karri, Brendan Dolan-Gavitt, and Siddharth Garg. 2022. Benchmarking Large Language Models for Automated Verilog RTL Code Generation. *arXiv:2212.11140* [cs.PL]
- [20] Hanrui Wang, Kuan Wang, Jiacheng Yang, Linxiao Shen, Nan Sun, Hae-Seung Lee, and Song Han. 2020. GCN-RL Circuit Designer: Transferable Transistor Sizing with Graph Neural Networks and Reinforcement Learning. In 2020 57th ACM/IEEE Design Automation Conference (DAC). 1–6. <https://doi.org/10.1109/DAC18072.2020.9218757>
- [21] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682* (2022).
- [22] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *arXiv:2201.11903* [cs.CL]