Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-26)
Special Track on AI4Tech: AI Enabling Critical Technologies

ACCFormer: Predicting Analog Circuit Performance Metrics via Topology-Aware Transformers

Bowen Liao[1], Yutong Feng[2], Jianhua Lin[1], Zhaohui Wu[1], Yuxuan Liang[2†], Bin Li[1†]
[1]School of Microelectronics, South China University of Technology, China
[2]The Hong Kong University of Science and Technology (Guangzhou), China
{phzhwu, phlibin}@scut.edu.cn; yuxliang@outlook.com;
{ytfeng.caspian,jian_hua_lin}@163.com; bwliao.hathaway@gmail.com

Abstract

Reusing and migrating analog circuit intellectual property (IP) across process nodes poses a significant challenge in modern chip design. Efficient and generalizable circuit performance prediction methods for analog circuits are crucial to achieving this goal. Current data-driven approaches typically rely on manually designed features, which perform poorly on unseen circuit architectures and struggle to model the inherent structural relationships within analog designs. To address these challenges, we propose **ACCFormer**, a novel topology-aware Transformer framework for predicting performance metrics of analog circuit. Our model combines device parameters with connectivity data to learn topology-aware representations, followed by a performance-oriented cross-attention mechanism where trainable metric queries adaptively focus on the most critical devices for each target parameter. Validated across different process nodes, our model achieves state-of-the-art prediction accuracy and demonstrates strong cross-process adaptability, highlighting its potential to accelerate IP reuse and reduce design cycles.

1 Introduction

With the development of electronic design automation (EDA) [Reis, 2022] and advanced manufacturing technologies [Ji *et al.*, 2019], efforts have been made to integrate AI into the design of analog integrated circuits (ICs) [Zadeh and Elamien, 2025]. This aims to shorten chip design cycles, enabling rapid product iteration to meet evolving market demands. In this work, we focus on the following task: predicting key performance metrics (e.g., DC gain, phase margin) of analog circuits manufactured at a specified process node (e.g., SMIC 65nm), based on circuit design parameters (e.g., device sizes, bias conditions) and interconnect relationships (netlists) between devices, as illustrated in Figure 1.

Automating this process is crucial for addressing a major bottleneck in contemporary analog design: reusing intellectual properties (IPs) across process nodes[1] [Brunoli *et al.*, 2002]. By avoiding "reinventing the wheel", chip design companies can reduce design costs and accelerate product iteration. Currently, migrating existing IPs to new process nodes requires extensive manual recalibration by specialized analog engineers to compensate for shifts in device characteristics due to process variations, which consumes significant time and human resources. Compounding the issue, most existing EDA software relies on solving massive nonlinear equations based on underlying physical laws to calculate performance changes in analog circuits when altering process nodes and design parameters [Scheffer *et al.*, 2018]. While accurate, a single simulation typically takes several hours for large scale analog ICs, hindering automated parameter searches and rapid design iterations. With the accumulation of data from simulation and production, coupled with advancements in AI, the industry has shifted toward data-driven approaches [Shin, 2023] rather than solving massive equation systems to achieve rapid prediction of metrics.

Previous work primarily employed simple machine learning methods [Huang *et al.*, 2021] combined with feature engineering for performance prediction. While effective in limited scenarios, this approach suffers from significant shortcomings: artificially designed features inherently lack flexibility and are tightly coupled to specific circuit architectures. Analog circuit design exhibits diverse topological structures, and the mapping between design parameters and performance metrics is not only highly nonlinear but also fundamentally dependent on the specific interconnections between devices. For example, while the *Telescopic Cascode Amplifier* (Figure 1 (a)) and *Folded Cascode Amplifier* (Figure 1 (b)) share identical device quantities and types, their differing topologies enable the former to deliver higher voltage gain and the latter to provide greater output swing under matching device parameters [Razavi, 2005]. These manually designed features often fail to transfer to new circuit topologies, necessitating expert-driven feature redesign for new circuits.

Inspired by the *small signal analysis* [Razavi, 2021], we approach this problem from the perspective of *graph regression*. We model circuit as a graph, where devices serve as

†Corresponding authors. Primary contact: Bin Li.

[1]IPs refer to reusable, modular circuit designs. Process nodes denote the fabrication technology provided by a specific manufacturer. These concepts will be elaborated upon in Section 2.1.

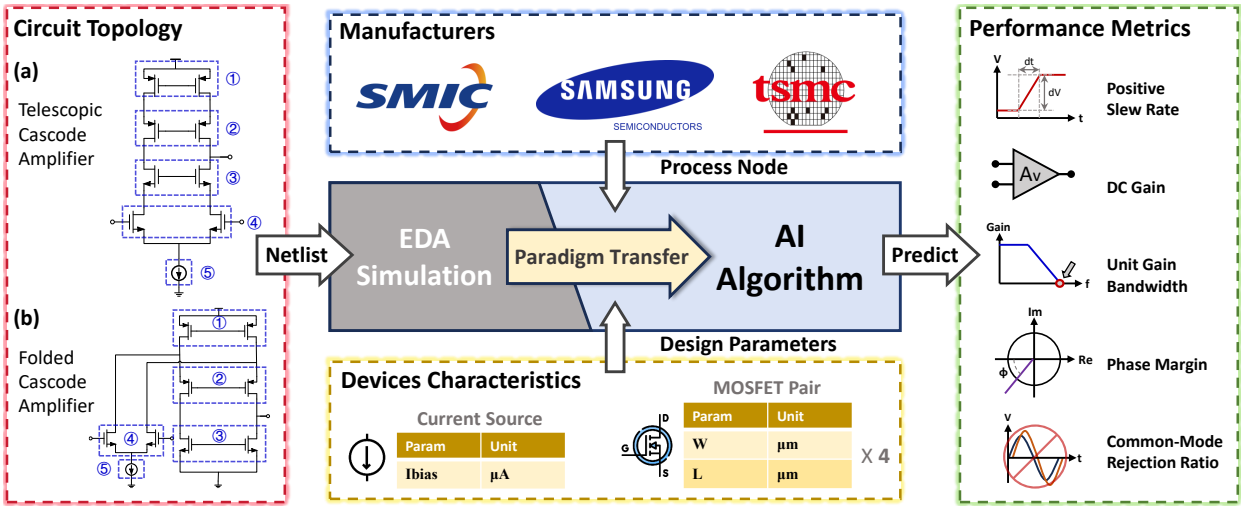

Figure 1: Pipeline of performance metrics prediction.

nodes and their design parameters as node attributes. The netlist is transformed into an adjacency list of the graph to characterize the connections between nodes. By identifying key subgraphs in the circuit, we address this problem from a data-driven perspective rather than manually designed feature engineering. Based on this, we propose **ACCFormer**, a novel topology-aware model for **A**nalog **C**ir**C**uit performance prediction based on the Trans**Former** architecture [Vaswani *et al.*, 2017]. To leverage the device and topology information of circuits, we propose a **Topology-Aware Attention** layer. It converts the circuit netlist into connectivity masks, guiding the multi-head self-attention mechanism to focus on actually connected device pairs. To identify devices critical to circuit performance, we introduce a **Performance-Oriented Filtering** layer. Each performance metric to be predicted is encoded as a query vector. Through a cross-attention mechanism, a subset of key devices that most significantly impact specific performance metrics is dynamically selected from all device representations. Experiments on commercial process nodes demonstrate the accuracy and robustness of our proposed model, highlighting its potential to accelerate design cycles. Our contributions can be summarized as follows:

- **A Novel Perspective.** We adopt a data-driven approach rather than physical simulation, addressing the performance metric prediction problem of analog circuits from the perspective of graph machine learning. By replacing the solution of complex equation systems with an end-to-end black-box model, this paradigm can significantly improve computational speed while ensuring accuracy.
- **ACCFormer.** We propose ACCFormer, a novel Transformer framework for analog circuit performance prediction. By incorporating the Topology-Aware Attention and Performance-Oriented Filtering layers, our model captures both topological influences and identifies the devices that dominate specific circuit performances. Its hierarchical pipeline explicitly encodes circuit connectivity, overcoming the limitations of previous approaches to enable accurate prediction and device-level interpretability.
- **Empirical Evidence.** We conduct extensive experiments to evaluate the effectiveness of ACCFormer. Empirical evaluation demonstrates that our model achieves SOTA accuracy in cross-process migration. To the best of our knowledge, this is the first work to present a large-scale experimental validation on commercial process nodes for the performance metrics prediction task.

2 Preliminary

2.1 Basic Concepts

In IC design, a circuit *Intellectual Property (IP)* refers to a reusable, modular design block that implements a specific function (e.g., an amplifier, a data converter). Migrating a proven IP to a new manufacturing technology is essential for design reuse. A *process node* (e.g., SMIC 65nm) refers to a specific semiconductor fabrication technology, characterized by its physical design rules, manufacturer, and transistor properties. Different nodes exhibit distinct device behaviors, making a design that works perfectly in one node likely non-functional in another without adjustment. Thus, process migration is a fundamental yet costly challenge in IC design.

2.2 Problem Definition

We formalize the core task as **Performance Metric Prediction (PMP)** for analog circuits. Given a circuit's **netlist** (a *graph* description listing devices like transistors and their interconnections) and their associated **design parameters** (e.g., device sizes, bias conditions), the goal is to predict a vector of key analog **performance metrics** (e.g., gain, bandwidth, power). The mapping $f : (\mathcal{G}, \mathcal{P}) \rightarrow \mathcal{Y}$ is highly nonlinear and topology-dependent, where $\mathcal{G}$ is the circuit graph describing the connection between devices, $\mathcal{P}$ is the set of device parameters, and $\mathcal{Y}$ is the performance vector. The task extends to a cross-process PMP scenario, where a model trained on data from a source process node must be adapted to accurately predict performance for the same circuit topology in a target process node, given only very limited data from the target.

3 Methodology

3.1 Motivation

Previous research treats circuits as a flat collection of devices and directly mapping all design parameters to performance metrics, discarding the connectivity between components. This unstructured approach suffers from two fundamental limitations that fail to align with the intrinsic characteristics of analog circuits:

- **Inadequate modeling of operating state dependence.** Transistor performance is highly dependent on its bias-determined operating region. The same set of design parameters can drive a transistor into entirely different operating regions under varying conditions or peripheral circuitry, leading to sharply distinct physical behaviour.
- **Ignoring circuit topology.** In analog design, performance metrics emerge from the flow of signals and the interactions among devices within a specific topology. Ignoring topology equates to overlooking the physical essence of circuit functionality, making it difficult for the model to establish accurate correlations between performance indicators and underlying device parameters.

In modern analog ICs, MOSFETs are a crucial component, introducing the most nonlinearity into circuits. Under different bias conditions (part of the design parameters), MOSFETs fall into different operating regions, exhibiting vastly different characteristics. As illustrated in Figure 2, in the cutoff, ohmic, and saturation regions, the electrical behavior of a MOSFET can be approximated by an ideal switch, a variable resistor, and a voltage-controlled current source, respectively. Since bias voltages are typically much larger than input signals, we can locally linearize their behavior within a specific operating region using simple circuit elements. In classical circuit analysis, analog engineers first (1) determine the operating region of each MOSFET to (2) define its behavior, then (3) identify key signal paths and components through simplified circuits to (4) calculate final performance metrics. This 4-step progressive analytical approach, termed "*small-signal analysis*" [Razavi, 2021; Neamen, 2007], as shown in Figure 3, inspired our design.

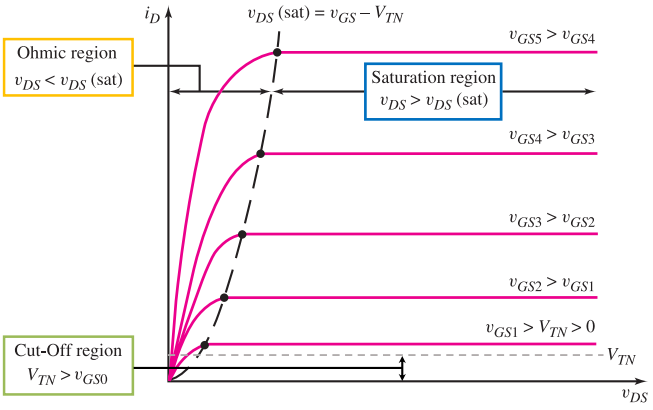

Figure 2: Different operating regions of MOSFET.

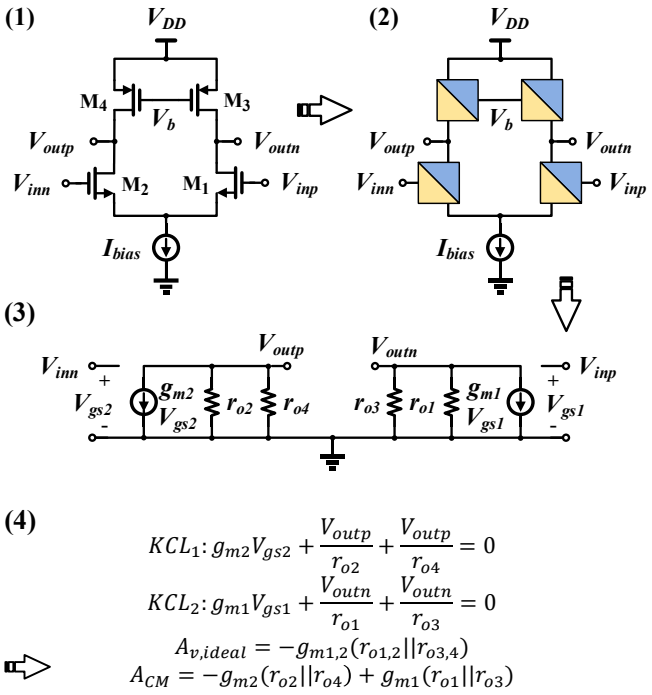

Figure 3: The process of small-signal analysis.

3.2 Overall Architecture

We propose ACCFormer, a hierarchical reasoning framework that follows the principle of "from *device characteristics*, to *topological interactions*, and finally to *performance correlations*" [Razavi, 2021]. ACCFormer decomposes performance prediction into four stages based on small-signal analysis:

- **Process-Parameter Fusion Embedding.** By integrating design parameters with process knowledge, we generate an initial embedded tensor for each device to characterize its behavioral characteristics and performance potential at a specific process node, corresponding to the Step (1).
- **Topology-Aware Attention.** We utilize the connectivity masks extracted from netlists to guide the self-attention mechanism, strictly limiting information exchange to physically interconnected devices. This enables learning dependencies determined by circuit architecture. This module corresponds to the Step (2).
- **Performance-Oriented Filtering.** We introduce a set of learnable performance query vectors. Through a cross-attention mechanisms, each query automatically focuses on the most critical subset of devices affecting its performance metrics. This module mimics the Step (3).
- **Multi-Dimensional Prediction Head.** In the model's output, filtered performance-related contextual information is mapped to specific performance metric predictions. This module aligns with the Step (4).

Our proposed model not only systematically addresses the shortcomings of previous flat models but also makes the decision-making process more interpretable, as it mimics the modular analytical logic of a design engineer.

3.3 Process-Parameter Fusion Embedding

This module aims to generate a comprehensive embedding for each device by fusing information from three aspects: its design parameters, the inherent characteristics of the process node, and its functional role within the circuit. The module receives the circuit netlist and device design parameters as its two primary inputs. The circuit netlist is used to define the device types and their interconnections, while the design parameters mainly include the width (W) and length (L) of each transistor, as well as the values of passive components such as resistors and capacitors. From these, the module constructs a unified representation through the following steps:

- **Process-Characteristic Encoding.** We construct a feature embedding $\mathbf{T}_i$ that encapsulates its inherent properties under the target process node, which includes fundamental physical parameters, parasitic characteristics, and device-type attributes (e.g. transistors, capacitors). We use $\mathbf{T} \in \mathbb{R}^{D \times d_T}$ to represent the matrix of these process-characteristic embeddings for all $\mathbf{D}$ device types and $\mathbf{d}_T$ represents the dimension of process embedding.
- **Functional-block Annotation.** We extracted the sub-functional blocks formed by each device and its adjacent devices (e.g., differential pair, current mirror). Each block type is associated with a learnable embedding, yielding a functional vector $\mathbf{F}_i$ for device $\mathbf{i}$. These embeddings forms the matrix $\mathbf{F} \in \mathbb{R}^{M \times d_F}$, where M represents the total number of all sub-functional blocks in the circuit and $\mathbf{d}_F$ represents the dimension of function embedding.
- **Design-Parameter Preprocessing.** We apply a feature engineering that transforms the design parameters of transistors into compact indicators to measure its performance potential (e.g., drive capability). We denote the parameter matrix as $\mathbf{P} \in \mathbb{R}^{\mathbf{N} \times \mathbf{d_P}}$, where N represents the device number in the circuit and $\mathbf{d}_P$ represents the feature dimension.

For the $\boldsymbol{i}_{\text{th}}$ device, its preliminary representation is composed of the three types of tensors mentioned above. While $\mathbf{T}_i$ and $\mathbf{P}_i$ are unique to the device, $\mathbf{F}_i$ must accommodate the case where a single device belongs to multiple sub-functional blocks. Therefore, its functional embedding $\mathbf{F}_i$ is defined as the the sum of the learnable embeddings of all sub-functional blocks assigned to it, which can be written as:

$$\mathbf{F}_i = \sum_{k \in \mathcal{F}(i)} \mathbf{e}_k$$

where $\mathcal{F}(i)$ represents the set of block types assigned to device i and $\mathbf{e}_k$ represents the embedding vector of sub-functional block k. Finally the three tensors are concatenated to form the device's preliminary representation:

$$\mathbf{D}_i = [\mathbf{P}_i || \mathbf{T}_i || \mathbf{F}_i] \in \mathbb{R}^{d_P + d_T + d_F}$$

The composite matrix $\mathbf{D} = [\mathbf{D_1^T}, ..., \mathbf{D_n^T}]^T$ will be processed through a shared MLP to obtain the final device embedding:

$$\mathbf{H}^{(0)} = MLP_{merge}(\mathbf{D}) \in \mathbb{R}^{N \times d}$$

where d is the hidden dimension of MLP. Each row of $\mathbf{H}^{(0)}$ represents the characterization embedding of a device.

3.4 Topology-Aware Attention

After completing the fusion of device information, we need to analyze the electrical characteristics exhibited by each device based on the circuit topology. To this end, we introduce the Topology-Aware Attention module. By stacking multiple-head self-attention layers (Multi-Head Self-Attention, MHSA) and incorporating attention masks derived from the netlist, this module restricts correlation coefficients to exist only between topologically connected devices.

The circuit netlist is quantified as a connection mask matrix $\mathbf{CM} \in \{0, -\infty\}^{N \times N}$, which is used to explicitly constrain the attention range and prevent the devices from focusing on unconnected devices, thereby avoiding the influence of noise: only when there is a direct or indirect signal path (i.e., an electrical connection) between devices i and j, $\mathbf{CM}_{ij} = 0$, can information be exchanged between them. Otherwise $\mathbf{CM}_{ij} = -\infty$, it will completely block the attention between unrelated devices. A crucial design consideration is that although Vdd (the power supply) and GND (ground) are topologically connected to many devices, they belong to the global reference potential nodes and do not constitute specific signal transmission paths. Consequently, if device i and device j are connected only through VDD or GND, we define that there is no functional connection between device i and device j, so we set $\mathbf{CM_{ij}} = -\infty$.

Within this module, $H^{(0)}$ serves as the input sequence from which the Query (Q), Key (K), and Value (V) are linearly projected. The connectivity attention mask CM is directly incorporated into the attention computation to restrict information exchange to topologically connected devices. For the $I-$th attention block(where $l = 1, 2, ..., L$) the output of the block as follows:

$$\mathbf{Q_h^l}, \mathbf{K_h^l}, \mathbf{V_h^l} = \mathbf{H^{(l-1)}} W_h^{Q,(l)}, \mathbf{H^{(l-1)}} W_h^{K,(l)}, \mathbf{H^{(l-1)}} W_h^{V,(l)}$$

$$\mathbf{head_h^{(l)}} = Softmax(\frac{\mathbf{Q_h^l}(\mathbf{K_h^{(l)}})^T}{\sqrt{d_k}} + \mathbf{CM}) V_h^{(l)}$$

$$\mathbf{H_{attn}^{(l)}} = Concat(\mathbf{h_1^{(l)}}, ..., \mathbf{h_h^{(l)}}) W^{O,(l)}$$

$$\mathbf{H^{(l)}} = LayerNorm(\mathbf{H^{(l-1)}} + \mathbf{H_{attn}^{(l)}})$$

where $\mathbf{H^{(l-1)}} \in \mathbb{R}^{N \times d}$ represents the output of the previous block($\mathbf{H^{(0)}}$ is the output of Process-Characteristic Encoding module), $W_h^{Q,(l)}, W_h^{K,(l)}, W_h^{V,(l)} \in \mathbb{R}^{d \times d_k}$ and $W^{O,(l)} \in \mathbb{R}^{h \cdot d_k \times d}$ are projection matrices for the $l-$th block, d_k is the feature dimension per head, h is the number of attention heads, and the attention mask CM is shared across all blocks.

3.5 Performance-Oriented Filtering

We introduce a set of learnable performance embeddings $\mathbf{Q}_{\text{perf}} \in \mathbb{R}^{M \times d}$, where M represents the number of performance metrics to be predicted. Each parameter metric embedding $\mathbf{q}_m$ is designed to implicitly encapsulate the types of analysis (e.g., DC bias or AC sensitivity) that its corresponding performance metric relies on.

We take the performance embeddings $\mathbf{Q_{perf}} \in \mathbb{R}^{M \times d}$ and the device tensors $\mathbf{H}^{(L)}$ from the previous module incorporating global structural information as both Keys and Values,

which are then fed into a Multi-Head Cross-Attention layer. This process enables each performance encoding to attend to all devices and highlight those that exert the most significant impact on it through attention weights. The calculation for cross-attention can be referenced from the steps listed in the previous subsection, and we will not elaborate further here.

The output of the cross-attention layer $\mathbf{C}$ is then refined through a feed-forward network (FFN) followed by layer normalization (LayerNorm). The computation is as follows:

$$\mathbf{C}^{'} = LayerNorm(\mathbf{C} + FFN(\mathbf{C}))$$

The final embedding $\mathbf{C}^{'} \in \mathbb{R}^{M \times d}$ contains the final contextual representations for each performance metric after the crucial device screening and feature refinement, and serves as the direct input to the multi-dimensional prediction head.

3.6 Multi-Dimensional Prediction Head

We now have an embedding $\mathbf{C}'$ that encapsulates the key information, and to map it back to the corresponding performance metrics, we can decode it using some simple decoders. We employ a shared MLP to serve as the decoder. It integrates the critical device information contained within each performance metric representation to generate the final prediction results $\hat{y}$, where $\hat{\mathbf{y}} \in \mathbb{R}^M$ for all M target metrics.

3.7 Discussion: From First Principle to Data

Traditional methods are based on first principles, where linear equations are formulated using Kirchhoff's voltage and current laws and then solved to obtain circuit responses. This *white-box* approach relies on complete knowledge of the circuit's structure and component parameters. While accurate, solving large systems of equations is time-consuming, especially as circuit sizes increase and various high-order effects of MOSFETs are considered, which makes the process even more cumbersome. In contrast, our approach encodes circuit components as embeddings and uses a decoder to predict performance metrics. This constitutes a *black-box* model. Through a data-driven method, we train the model to learn the mapping from circuit descriptions to performance metrics.

4 Experiments

4.1 Settings

Dataset Description. We constructed our dataset using circuit simulation data from two commercial process nodes: SMIC 180nm as the source node and SMIC 65nm as the target node. Simulations were performed in Cadence Virtuoso for two classic analog building blocks considered in this study: 5T-OTA and two-stage OP-AMP.

To ensure data diversity and mitigate the bias introduced by the highly nonlinear design space (where many random parameter sets lead to non-functional circuits), we adopted an active sampling strategy. For the source node, we initially performed 50,000 simulations per topology and then retained 10,000 balanced samples that adequately represent both functional and non-functional design regions. For the target node, we performed 5,000 simulations per topology and retained 1,000 balanced samples, reflecting the practical scenario of limited data availability in a new process.

Baselines for Comparison. We compare our ACCFormer with the following baselines:

- MLP: MLP was the model adopted in previous research on this problem [Wu and Savidis, 2022].
- $\text{MLP}_{\text{embedding}}$: Since embeddings of device types and subfunctions are introduced in ACCFormer, we also incorporate these embedding into the input for fairness.
- ResMLP: ResMLP introduces a residual structure on the basis of the conventional MLP.
- $\text{ResMLP}_{\text{embedding}}$: Consistent with $MLP_{embedding}$, we also introduce the embedding into $ResMLP$ to ensure the fairness of input comparison.

Implementation Details. Our model is implemented using PyTorch 2.5.1 and trained on an NVIDIA A800 GPU. We employ the AdamW optimizer with a batch size of 128. The initial learning rate is set to 1×10^{-4} for the 5T-OTA and 3×10^{-4} for the Two-Stage Op-Amp, and is halved every 20 epochs. The hidden dimension of all models is fixed at 512. For ACCFormer, the number of topology-aware attention and attention heads are selected as 3 and 8, correspondingly.

Evaluation Metrics. We leverage Loss and Coefficient of Determination(R^2) for evaluation, where Loss is the average of the Mean Squared Error (MSE) and the Mean Absolute Error (MAE). A smaller value of Loss and a larger R^2 indicate better performance of the model.

4.2 Model Comparison

According to the results in Table 1, our proposed ACCFormer demonstrates remarkable improvements over the baseline models across all datasets and metrics. While MLP and ResMLP achieve competitive results on simpler circuits like the 5T-OTA, MLP generalizes better for complex metrics such as CMRR, especially in structurally intricate designs like the two-stage op-amp, due to its lower overfitting tendency. Incorporating embeddings substantially boosts performance by encoding topological cues, but MLP-based models remain limited by their flat architecture. In contrast, ACCFormer excels at leveraging topological information, showing narrow margins in simple cases but significantly wider advantages in complex, highly nonlinear scenarios.

4.3 Ablation Study

Effects of Topology-Aware Attention. We construct an additional variant termed Topology-Aware Attention-Free, recorded as TAAF, in which this module is entirely removed while keeping other components unchanged. The results demonstrate that the standard model outperforms TAAF, which verifies that the Topology-Aware Attention module can effectively capture topological structural information.

Effects of Performance-Oriented Filtering. To evaluate the contribution of the Performance-Oriented Filtering module, we construct an additional variant termed Performance-Oriented Filtering-Free, recorded as POFF, in which this module is entirely removed while keeping all other components of ACCFormer unchanged. The results show that our proposed module can effectively identify and highlight the most influential devices for each target performance metric.

Model	Positive Slew Rate		DC Gain		UGF		Phase Margin		CMRR	
	Loss	$R^2(\%)$	Loss	$R^2(\%)$	Loss	$R^2(\%)$	Loss	$R^2(\%)$	Loss	$R^2(\%)$
5T-OTA (SOURCE)										
MLP	5.22	98.52	5.70	98.02	5.01	98.66	5.22	98.45	20.83	80.29
$\text{MLP}_{\text{embedding}}$	2.38	99.22	2.68	98.90	1.77	99.73	1.80	99.57	16.21	85.96
ResMLP	4.36	98.71	4.42	98.45	4.29	98.87	3.82	98.97	24.40	76.53
$\text{ResMLP}_{\text{embedding}}$	2.32	99.55	2.59	99.20	2.10	99.68	2.04	99.53	1.60	85.11
ACCFormer(ours)	**1.36**	**99.80**	**1.99**	**99.42**	**1.17**	**99.89**	**1.75**	**99.60**	**11.22**	**90.29**
5T-OTA (TARGET)										
MLP	6.37	97.83	7.74	96.33	6.56	97.93	6.73	97.87	11.50	92.34
$\text{MLP}_{\text{embedding}}$	2.92	98.86	5.01	97.46	2.11	99.68	2.83	99.36	10.57	91.18
ResMLP	4.94	98.25	5.86	97.42	4.89	98.55	5.14	98.30	13.34	89.63
$\text{ResMLP}_{\text{embedding}}$	3.11	99.03	3.95	98.20	2.72	99.47	2.64	99.53	7.16	94.94
ACCFormer(ours)	**1.83**	**99.34**	**3.62**	**98.89**	**1.32**	**99.84**	**1.96**	**99.69**	**5.36**	**96.16**
Two-Stage OP-AMP (SOURCE)										
MLP	18.21	88.07	31.07	65.47	19.92	85.30	25.03	81.69	24.78	77.49
$\text{MLP}_{\text{embedding}}$	14.39	90.86	21.27	77.96	15.37	89.47	21.40	85.04	17.97	84.35
ResMLP	21.41	84.75	43.80	50.20	25.91	78.64	31.04	75.64	36.85	62.84
$\text{ResMLP}_{\text{embedding}}$	15.16	90.23	23.86	74.57	15.46	89.27	21.77	84.67	19.07	83.37
ACCFormer(ours)	**12.91**	**91.65**	**18.46**	**80.07**	**13.10**	**90.93**	**20.84**	**85.62**	**14.49**	**87.90**
Two-Stage OP-AMP (TARGET)										
MLP	17.61	88.69	32.76	64.74	19.85	86.70	32.16	73.81	31.12	68.80
$\text{MLP}_{\text{embedding}}$	11.69	93.09	24.59	74.83	13.81	92.22	22.36	79.78	22.35	78.97
ResMLP	18.20	87.50	41.36	53.42	24.19	81.84	39.89	63.89	37.35	61.58
$\text{ResMLP}_{\text{embedding}}$	12.38	92.84	27.74	69.48	14.29	91.56	27.91	78.07	24.72	75.92
ACCFormer(ours)	**7.89**	**95.59**	**17.82**	**81.97**	**9.36**	**95.22**	**22.18**	**82.51**	**18.82**	**81.54**

Table 1: Model performance comparison. The unit of $Loss$ is $1E-2$.

Variant	5T (SRC)		5T (TGT)		Two (SRC)		Two (TGT)	
	Loss	R^2	Loss	R^2	Loss	R^2	Loss	R^2
DC Gain								
TAAF	2.01	99.37	4.39	96.70	40.50	52.75	33.04	62.94
POFF	2.81	99.03	4.70	97.58	27.34	69.76	34.90	59.51
Standard	**1.99**	**99.42**	**3.62**	**98.89**	**18.46**	**80.07**	**17.82**	**81.97**
CMRR								
TAAF	12.99	88.40	6.50	94.39	32.49	67.37	34.31	62.88
POFF	17.13	83.85	9.20	91.43	21.71	80.13	30.95	67.13
Standard	**11.22**	**90.29**	**5.36**	**96.16**	**14.49**	**87.90**	**18.82**	**81.54**

Table 2: Ablation study on the key components of ACCFormer.

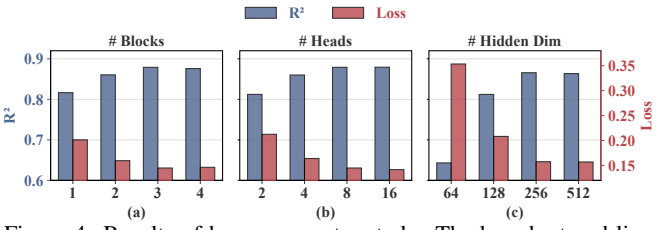

Figure 4: Results of hyperparameter study. The bar chart and line chart represent R^2 and loss, respectively.

4.4 Hyperparameter Study

We investigate the effects of the number of Topology-Aware Attention blocks L, the number of heads N_h and the hidden layer dim C. The results are shown in Fig 4(a) illustrates that the model performance improves as L increases, reaches its best at $L = 3$, and then deteriorates when $L = 4$. Fig 4(b) shows that the model performance improves with the increase of N_h, but the improvement between N_h=8 and $N_h = 16$ is marginal. Fig 4(c) demonstrates the variation of model performance with the hidden layer dimension, where the best performance is achieved at $C = 256$.

4.5 Visualization Study

To further elucidate the internal behavior of ACCFormer, we visualize how the attention distribution in the Performance-Oriented Filtering (POF) module evolves as the model depth increases. Figure 5 presents the attention heatmaps for a given set of valid design parameters of the two-stage op-amp, comparing the POF module's outputs after $L = 1$ and $L = 3$ Topology-Aware Attention blocks. Given that metrics like UGF and Phase Margin involve complex pole-zero analyses, we focus our interpretation on Positive Slew Rate, DC Gain, and CMRR, whose first-order analytical expressions are:

$$Pos\_Slewrate \approx \frac{I_{ss}}{C_c}$$

$$DC\_Gain \approx g_{m2}(r_{o2}||r_{o4})\frac{2g_{m4}r_{o4}+1}{2(g_{m4}r_{o4}+1)}g_{m6}(r_{o6}||r_{o7})$$

$$CMRR \approx (1+2g_{m1,2}R_{ss})g_{m3,4}(r_{o1,2}||r_{o3,4})$$

where I_{ss} is the bias current, C_c is the compensation capacitor, r_{oi} is the output resistance of transistor i and R_{ss} is the output resistance of the tail current source. The heatmaps align remarkably well with these theoretical insights:

- **Positive Slew Rate.** For $L = 1$, the POF module disperses attention to several non-current-mirror transistors (X1–X8, except X2, since when testing the swing rate, the output is directly connected to the gate terminal of X2.) and pays little attention to I_{bias} For $L = 3$, it shifts focus primarily to the devices most relevant to the slew rate formula, demonstrating a more physics-aware concentration.
- **DC Gain.** The $L = 1$ model shows non-zero attention to the capacitor, whereas the $L = 3$ model correctly ignores it. This is consistent with circuit theory, as capacitor has negligible effect at the low frequencies relevant to the gain.
- **CMRR.** With $L = 3$, attention is almost exclusively placed on the tail current source transistor (X5), the key component in the CMRR expression. The channel length of the tail current source transistor directly determines the performance of the CMRR. In contrast, the $L = 1$ model also attributes some importance to X6, a transistor whose parameters do not directly influence CMRR.

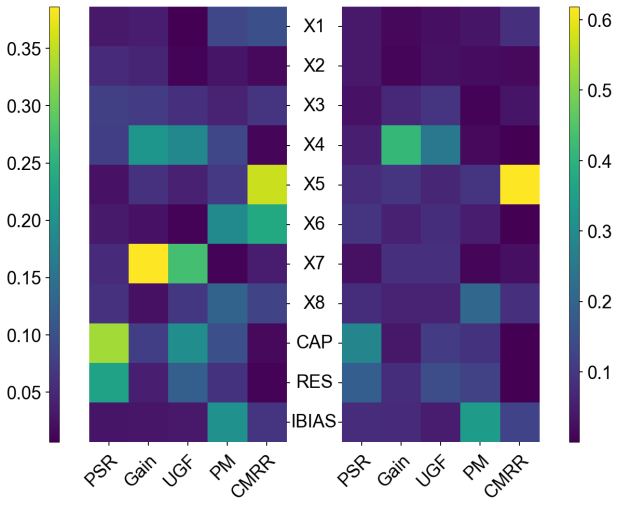

Figure 5: The visualization of Topology-Aware Attention blocks. Left subfigure: $L = 1$; right subfigure: $L = 3$.

This visualization study confirms that deeper Topology-Aware Attention blocks enable ACCFormer to progressively refine its structural understanding, guiding the Performance-Oriented Filtering module to allocate attention in a manner that is both focused and consistent with circuit principles.

5 Related Works

AI4EDA. The application of machine learning and artificial intelligence in electronic design automation (AI4EDA) [Yu, 2023; Shin, 2023; Huang *et al.*, 2021] has evolved significantly, transitioning from computer-aided design (CAD) [Kahng and Koushanfar, 2015; Catanzaro *et al.*, 2008] to AI-assisted EDA [Wu *et al.*, 2022], and more recently, toward AI-native and agentic design paradigms [Zang *et al.*, 2025; Wu *et al.*, 2024]. This evolution is driven by the increasing complexity of ICs and the need to manage vast design spaces beyond human capability. A prominent application of AI in EDA is design space exploration and optimization. Tools like Synopsys's DSO.ai [Verma, 2024; Venkatachar, 2021] leverage reinforcement learning to autonomously search for optimal PPA (Power, Performance, Area) solutions within the enormous solution space of chip design. Similarly, Cadence's Cerebrus [Metcalfe, 2021] employs generative AI to automate and optimize the IC design flow, achieving significant improvements in throughput. For verification, which consumes a substantial portion of the design cycle, AI-driven platforms such as the Verisium Platform [Nguyen *et al.*, 2024] and Synopsys's VSO.ai [Lnu, ; Reddy, 2024] utilize machine learning to accelerate coverage convergence, automate root cause analysis.

Performance Metric Prediction. Early and impactful research includes Google's work on chip placement [Goldie and Mirhoseini, 2024] using graph neural networks (GNNs) [Mirhoseini *et al.*, 2021; El Sayed *et al.*, 2025] and reinforcement learning [Markov, 2023; Mirhoseini *et al.*, 2020], which demonstrated that AI could produce superhuman results in optimizing physical design objectives. For analog and mixed-signal circuits, research has focused on overcoming the "black art" nature of design. Projects like BAG [Crossley *et al.*, 2013; Hakhamaneshi *et al.*, 2019] and more recent frameworks such as ALIGN [Kunal *et al.*, 2019; Dhar *et al.*, 2020], MAGICAL[Xu *et al.*, 2019; Chen *et al.*, 2020], and FALCON [Mehradfar *et al.*, 2025] explore automated circuit sizing and layout generation . Commercial tools like Synopsys's ASO.ai [Synopsys Inc., 2026] apply reinforcement learning to accelerate simulation and characterization across PVT (Process, Voltage, Temperature) corners.

6 Conclusion and Future Work

In this work, we addressed the critical challenge of predicting the performance metrics of analog circuits across different semiconductor process nodes. Moving beyond simulation-heavy physical solvers and inflexible feature-based approaches, we reformulated the problem from a graph machine learning perspective. To this end, we introduced ACCFormer, a topology-aware Transformer model for accurate and interpretable prediction of analog circuit performance across process nodes. By framing the problem as graph regression and integrating circuit connectivity into the attention mechanism, our method outperforms traditional approaches, enabling faster design cycles without costly simulations. In the future work, we will refine the graph representation to the port level for more granular circuit modeling.

Appendix

To facilitate understanding of the performance prediction task discussed in this paper, this appendix summarizes the main analog circuit performance metrics involved in our experiments. These metrics describe different aspects of circuit behavior, including low-frequency amplification, frequency response, closed-loop stability, common-mode noise rejection, and large-signal transient speed.

- **DC Gain (A_v).** The DC gain measures the small-signal voltage amplification capability of an analog circuit at low frequency. It is usually defined as the ratio between the output voltage variation and the input voltage variation, i.e., $A_v = v_{\mathrm{out}}/v_{\mathrm{in}}$. A higher DC gain generally indicates stronger amplification capability and better accuracy in feedback-based analog systems.
- **Unity-Gain Frequency (UGF).** The unity-gain frequency denotes the frequency at which the magnitude of the open-loop voltage gain drops to one, or equivalently $0\,\mathrm{dB}$. It characterizes the effective bandwidth of an amplifier and reflects how fast the circuit can respond to input signal variations. A larger UGF usually indicates a wider operating bandwidth and faster settling in practice.
- **Phase Margin (PM).** The phase margin is a stability metric measured at the unity-gain frequency. It is defined as the difference between the circuit phase and -180° when the loop gain magnitude reaches unity. A sufficient phase margin indicates that the amplifier is less likely to oscillate and has better closed-loop stability and reduced ringing.
- **Common-Mode Rejection Ratio (CMRR).** The common-mode rejection ratio measures the ability of a differential circuit to suppress common-mode signals while amplifying differential-mode signals. It is commonly defined as the ratio between the differential-mode gain and the common-mode gain. A higher CMRR indicates stronger immunity to common-mode noise and mismatch-induced disturbances.
- **Positive Slew Rate (SR).** The positive slew rate describes the maximum rising rate of the output voltage under a large-signal transient response. It is typically expressed as $\mathrm{SR}^+ = \max(dv_{\mathrm{out}}/dt)$ during the rising transition. A higher positive slew rate indicates that the circuit can charge the load or compensation capacitor more rapidly, enabling faster large-signal response in practice.

Contribution Statement

Bowen Liao and Yutong Feng contributed equally to this work.

Acknowledgments

This work was partially supported by the Guangdong S&T Programme, China (2022B0101180001), as well as partially by the Engineering Research Center of Design and Technology Co-Optimization of Integrated Circuits, Ministry of Education, China.

References

[Brunoli *et al.*, 2002] Mike Brunoli, Masao Hotta, Felicia James, Rudy Koch, Roy McGuffin, and Andrew J Moore. Analog intellectual property: now? or never? In *Proceedings of the 39th annual Design Automation Conference*, pages 181–182, 2002.

[Catanzaro *et al.*, 2008] Bryan Catanzaro, Kurt Keutzer, and Bor-Yiing Su. Parallelizing cad: A timely research agenda for eda. In *Proceedings of the 45th annual Design Automation Conference*, pages 12–17, 2008.

[Chen *et al.*, 2020] Hao Chen, Mingjie Liu, Biying Xu, Keren Zhu, Xiyuan Tang, Shaolan Li, Yibo Lin, Nan Sun, and David Z Pan. Magical: An open-source fully automated analog ic layout system from netlist to gdsii. *IEEE Design & Test*, 38(2):19–26, 2020.

[Crossley *et al.*, 2013] John Crossley, Alberto Puggelli, H-P Le, B Yang, R Nancollas, Kwangmo Jung, Lingkai Kong, Nathan Narevsky, Yue Lu, Nicholas Sutardja, et al. Bag: A designer-oriented integrated framework for the development of ams circuit generators. In *2013 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 74–81. IEEE, 2013.

[Dhar *et al.*, 2020] Tonmoy Dhar, Kishor Kunal, Yaguang Li, Meghna Madhusudan, Jitesh Poojary, Arvind K Sharma, Wenbin Xu, Steven M Burns, Ramesh Harjani, Jiang Hu, et al. Align: A system for automating analog layout. *IEEE Design & Test*, 38(2):8–18, 2020.

[El Sayed *et al.*, 2025] Ziad El Sayed, Zeng Wang, Hana Selmani, Johann Knechtel, Ozgur Sinanoglu, and Lilas Alrahis. Graph neural networks for integrated circuit design, reliability, and security: Survey and tool. *ACM Computing Surveys*, 58(4):1–44, 2025.

[Goldie and Mirhoseini, 2024] Anna Goldie and Azalia Mirhoseini. How AlphaChip transformed computer chip design, sep 2024.

[Hakhamaneshi *et al.*, 2019] Kourosh Hakhamaneshi, Nick Werblun, Pieter Abbeel, and Vladimir Stojanović. Bagnet: Berkeley analog generator with layout optimizer boosted with deep neural networks. In *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 1–8. IEEE, 2019.

[Huang *et al.*, 2021] Guyue Huang, Jingbo Hu, Yifan He, Jialong Liu, Mingyuan Ma, Zhaoyang Shen, Juejian Wu, Yuanfan Xu, Hengrui Zhang, Kai Zhong, et al. Machine learning for electronic design automation: A survey. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 26(5):1–46, 2021.

[Ji *et al.*, 2019] Zhigang Ji, Haibao Chen, and Xiuyan Li. Design for reliability with the advanced integrated circuit (ic) technology: challenges and opportunities. *Science China. Information Sciences*, 62(12):226401, 2019.

[Kahng and Koushanfar, 2015] Andrew B Kahng and Farinaz Koushanfar. Evolving eda beyond its e-roots: an overview. In *2015 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 247–254. IEEE, 2015.

[Kunal *et al.*, 2019] Kishor Kunal, Meghna Madhusudan, Arvind K Sharma, Wenbin Xu, Steven M Burns, Ramesh Harjani, Jiang Hu, Desmond A Kirkpatrick, and Sachin S Sapatnekar. Align: Open-source analog layout automation from the ground up. In *Proceedings of the 56th Annual Design Automation Conference 2019*, pages 1–4, 2019.

[Lnu,] Deepak Kumar Lnu. Pcie verification at high speeds: Challenges, solutions, and future trends.

[Markov, 2023] Igor L Markov. The false dawn: Reevaluating google's reinforcement learning for chip macro placement. *arXiv preprint arXiv:2306.09633*, 2023.

[Mehradfar *et al.*, 2025] Asal Mehradfar, Xuzhe Zhao, Yilun Huang, Emir Ceyani, Yankai Yang, Shihao Han, Hamidreza Aghasi, and Salman Avestimehr. Falcon: An ml framework for fully automated layout-constrained analog circuit design. *arXiv preprint arXiv:2505.21923*, 2025.

[Metcalfe, 2021] Rod Metcalfe. Machine learning-driven full-flow chip design automation. *Accessed: Mar*, 1:2024, 2021.

[Mirhoseini *et al.*, 2020] Azalia Mirhoseini, Anna Goldie, Mustafa Yazgan, Joe Jiang, Ebrahim Songhori, Shen Wang, Young-Joon Lee, Eric Johnson, Omkar Pathak, Sungmin Bae, et al. Chip placement with deep reinforcement learning. *arXiv preprint arXiv:2004.10746*, 2020.

[Mirhoseini *et al.*, 2021] Azalia Mirhoseini, Anna Goldie, Mustafa Yazgan, Joe Wenjie Jiang, Ebrahim Songhori, Shen Wang, Young-Joon Lee, Eric Johnson, Omkar Pathak, Azade Nova, et al. A graph placement methodology for fast chip design. *Nature*, 594(7862):207–212, 2021.

[Neamen, 2007] Donald A Neamen. *Microelectronics: circuit analysis and design*, volume 43. McGraw-Hill New York, 2007.

[Nguyen *et al.*, 2024] Chi Lan Phuong Nguyen, Quyet Van Hoang, An Hai Lam Tran, Khoa Dac Tran, Nghia Trong Tran, Takafumi Noguchi, James David, So Katogi, Yukie Endo, and Koji Hirakimoto. Enhancing functional verification productivity through an automated workflow with machine learning based tools. In *2024 9th International Conference on Integrated Circuits, Design, and Verification (ICDV)*, pages 166–170. IEEE, 2024.

[Razavi, 2005] Behzad Razavi. *Design of analog CMOS integrated circuits*. 2005.

[Razavi, 2021] Behzad Razavi. *Fundamentals of microelectronics*. John Wiley & Sons, 2021.

[Reddy, 2024] Taruna Reddy. Reducing manual effort and achieving better chip verification coverage with ai and formal techniques, sep 2024.

[Reis, 2022] Ricardo Augusto Da Luz Reis. Eda: Overview and some trends. *Journal of Integrated Circuits and Systems*, 17(3):1–10, 2022.

[Scheffer *et al.*, 2018] Louis Scheffer, Luciano Lavagno, and Grant Martin. *EDA for IC system design, verification, and testing*. CRC press, 2018.

[Shin, 2023] Youngsoo Shin. Ai-eda: Toward a holistic approach to ai-powered eda. In *2023 ACM/IEEE 5th Workshop on Machine Learning for CAD (MLCAD)*, pages 1–3. IEEE, 2023.

[Synopsys Inc., 2026] Synopsys Inc. Synopsys ASO.ai: Unleashing the power of AI to analog design, 2026. Accessed: 2026-01-17.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Venkatachar, 2021] Arun Venkatachar. Confluence of ai/ml with eda and software engineering. In *2021 22nd International Symposium on Quality Electronic Design (ISQED)*, pages 13–15. IEEE, 2021.

[Verma, 2024] Piyush Verma. Dso. ai-a distributed system to optimize physical design flows. In *Proceedings of the 2024 International Symposium on Physical Design*, pages 115–116, 2024.

[Wu and Savidis, 2022] Zhengfeng Wu and Ioannis Savidis. Transfer learning for reuse of analog circuit sizing models across technology nodes. In *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1033–1037. IEEE, 2022.

[Wu *et al.*, 2022] Nan Wu, Yuan Xie, and Cong Hao. Ai-assisted synthesis in next generation eda: Promises, challenges, and prospects. In *2022 IEEE 40th International Conference on Computer Design (ICCD)*, pages 207–214. IEEE, 2022.

[Wu *et al.*, 2024] Haoyuan Wu, Zhuolun He, Xinyun Zhang, Xufeng Yao, Su Zheng, Haisheng Zheng, and Bei Yu. Chateda: A large language model powered autonomous agent for eda. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(10):3184–3197, 2024.

[Xu *et al.*, 2019] Biying Xu, Keren Zhu, Mingjie Liu, Yibo Lin, Shaolan Li, Xiyuan Tang, Nan Sun, and David Z Pan. Magical: Toward fully automated analog ic layout leveraging human and machine intelligence. In *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 1–8. IEEE, 2019.

[Yu, 2023] Bei Yu. Machine learning in eda: When and how. In *2023 ACM/IEEE 5th Workshop on Machine Learning for CAD (MLCAD)*, pages 1–6. IEEE, 2023.

[Zadeh and Elamien, 2025] Danial Noori Zadeh and Mohamed B Elamien. Generative ai for analog integrated circuit design: Methodologies and applications. *IEEE Access*, 2025.

[Zang *et al.*, 2025] Zelin Zang, Yuhang Song, Bingo Wing-Kuen Ling, Aili Wang, and Fuji Yang. The dawn of agentic eda: A survey of autonomous digital chip design. *arXiv preprint arXiv:2512.23189*, 2025.