

SCALABLE AI/ML APPROACH TO AUTOMATED ANALOG CIRCUIT DESIGN: FROM SPICE SIMULATION TO LAYOUT-AWARE OPTIMIZATION

K. PURUSHOTHAM¹, T KRISHNA MOHANA², VAKITI SREELATHA REDDY³, Dr. T.RAJASANTHOSH KUMAR⁴, DR.R. JAYASUDHA⁵, GANDHIMATHI K⁶

¹ Assistant Professor, Department of CSE, Neil Gogte Institute of Technology, Hyderabad, India.

² Assistant Professor, Department of Electronics and Communication Engineering, Aditya University, Surampalem, India.

³ Sr Asst Professor, Department of Electronics and Instrumentation Engineering, CVR College of Engineering, Ibrahimpatnam, Hyderabad, India.

⁴ Department of Mechanical Engineering, Puducherry Technological University, Puducherry, India.

⁵ Associate professor, Department of Mathematics, Dr.N.G.P. Institute of Technology, Coimbatore, India.

⁶ Assistant Professor, Department of CSE, Muthayammal Engineering College, Rasipuram, Tamilnadu, India.

Email: ¹ kpurushotham1183@gmail.com, ² krishnamohana.tenneti@acet.ac.in, ³ srilathareddy.cvr@gmail.com, ⁴ trsk.mech@gmail.com, ⁵ rjayasudha98@gmail.com, ⁶ gandhimathicse24@gmail.com

ABSTRACT

The escalating complexity of modern analog integrated circuit (IC) design, spanning deep-submicron technology nodes from 5 nm to 65 nm, has rendered conventional manual SPICE-based optimization both computationally prohibitive and commercially unsustainable. SPICEFormer is a unified end-to-end artificial intelligence (AI) framework that combines the Graph Transformers, Physics-Informed Neural Networks (PINNs), Soft Actor-Critic (SAC) Reinforcement Learning, Bayesian Optimization, and layout-aware parasitic extraction to automate analog circuit design, from parsing SPICE netlist to final layout verification. SPICEFormer reduces the number of SPICE simulation calls by $85.1\% \pm 3.2\%$ on average, with a $24.9\% \pm 1.4\%$ more accurate gain prediction, reduces the layout area by $28.4\% \pm 2.1\%$ on average, and improves the speed of optimization convergence by $51.3\% \pm 2.8\%$ on average across 10 independent experimental runs using different analog circuit topologies (OTA, Folded Cascode, Telescopic, Current Mirror, Bandgap Reference, Comparator), 3 technology nodes (TSMC 65 nm, GF 45 nm, SKY130), and with all improvements statistically significant at $p < 0.01$ (paired Wilcoxon signed-rank test). Post-layout performance degradation is reduced to 1.2–2.3% as opposed to 7.6–12.4% for previous methods. An integrated explainability module, Grad-CAM, offers interpretable attribution of design decisions, enhancing AI-based EDA transparency. The results show the potential of physics-informed layout-aware RL for large-scale analog circuit generation.

Keywords: *Analog Circuit Design, Graph Transformer, Reinforcement Learning, SPICE Simulation, Bayesian Optimization, Layout-Aware Optimization, Physics-Informed Neural Networks*

1. INTRODUCTION

Designing an analog integrated circuit (IC) is one of the most knowledge rich and time-consuming challenges in today's semiconductor design. While digital circuits have a long history of extensive synthesis and place-and-route automation, analog circuits require intimate knowledge of physics, process technology limits and performance trade-offs that are inherently non-convex, with multiple

objectives and layout dependent. The 5 nm – 65 nm regime has seen technology scaling, with the sensitivity of analog performance metrics increasing to the magnitude of the chip's process variability, parasitic capacitances, and layout-induced stress increasing significantly, further driving up the cognitive and computational load for the human designer [1], [2], [3].

While SPICE simulation is still the benchmark for analog circuit verification, it can become a significant constraint in iterative optimization. It can

take several minutes for a moderate complexity OTA circuit to be simulated once using either a DC or AC simulation and can take days for an optimization routine that runs thousands of times and evaluates thousands of circuits. But, because this performance landscape is nonlinear and non-differentiable, direct gradient-based optimization is not possible, and expensive evolutionary or surrogate-model approaches must be used instead [4], [5], [6]. In addition, the size of transistors optimized at the schematic level often does not meet the specified dimensions after layout because of the parasitic resistance and capacitance generated by routing, metal layers and substrate coupling, which is referred to as layout-schematic gap.

AI and machine learning have become the game-changers across all scientific fields – analog EDA is no exception. For circuits, Graph Neural Networks (GNNs) can naturally model circuit netlists, which describe the connectivity and relationships of the devices in the circuit as a graph. Multi-head self-attention transformer (MHSA) architectures have proven to be the most effective way to model long-range dependencies in graph-structured data, making them suitable for circuit graphs with a wide variety of parts. However, reinforcement learning (RL) provides a principled approach to sequential transistor sizing decisions in the presence of noisy, delayed reward signals, particularly the Soft Actor-Critic (SAC) algorithm [7], [8], [9].

In this paper, the above challenge is addressed by introducing SPICEFormer, a unified framework that was validated through 10 independent experimental runs per topology, with thorough statistical reporting of mean, std. dev., confidence intervals and significance testing. The main contributions are (1) a physics-informed Graph Transformer that incorporates electrical constraints into multi-head attention; (2) a SAC-RL loop that uses a Bayesian surrogate to significantly reduce simulation calls (by more than 85%); (3) a closed-loop layout generator with Cadence Virtuoso PEX feedback; (4) an explainability module based on Grad-CAM and (5) statistically rigorous benchmarking over six topologies and three technology nodes.

2. LITERATURE REVIEW

2.1 AI for Analog Circuit Design

The early machine learning-based analog design applications were limited to surrogate modelling of SPICE simulations based on support vector regression and Gaussian processes. A GNN-based sizing framework is introduced that represents netlist topology as a directed graph, which speeds up the evolutionary algorithms by 4×. AutoCkt is proposed

as an RL agent that is trained using proximal policy optimization (PPO) for two-stage amplifiers sizing. Even with these developments AutoCkt produces flat feature vectors, ignoring the topological information that is represented in graph, and only provides results of a single run without statistical validation [10], [11].

2.2 Reinforcement Learning in Circuit Optimization

The SAC algorithm is designed to optimize the policy in a way that is constrained by entropy, which controls the balance between exploration and exploitation, particularly in a multimodal reward landscape and stochastic SPICE evaluations. SAC is used to multi-stage amplifier sizing and outperformed PPO/DDPG baseline with better sample efficiency, without modelling the layout parasitic effects. CircuitFormer added Transformer style attention to the GNN paradigm and achieved better generalization performance in unseen topologies, but without physics constraints in the attention mechanism and without integration of feedback from the layout-PEX [12], [13].

2.3 Graph Neural Networks and Graph Transformers

Graph Neural Networks can use learnable functions to aggregate neighborhood information, allowing for propagation of device-level features respecting circuit connectivity. This approach is extended by Graph Transformers [14] in which the fixed aggregation kernels are replaced by multi-head attention, thereby allowing for global information flow without the over-smoothing pathologies of deep MPNNs. Physics-informed neural networks (PINNs) [15] incorporate residuals of differential equations governing the circuit as soft constraints to regularize learned representations toward physically plausible circuit operation, thereby embedding Kirchhoff's voltage and current laws.

2.4 Bayesian Optimization and SPICE Surrogate Modelling

Bayesian Optimization is a principled approach for global optimization of expensive black-box functions by using a Gaussian process surrogate and picking evaluation points using expected improvement or upper confidence bound acquisition functions. It is shown that GP-BO can improve sample efficiency to converge within 200-500 SPICE calls whereas the scalability of GP decreases cubically as the number of observations increase. The SPICEFormer framework uses a Graph-Transformer based surrogate, allowing for batch

Bayesian optimisation with the SAC-RL policy as the warm-start acquisition strategy[16] [17].

2.5 Analog Layout Automation

At the layout stage, MAGICAL presented its machine-learning driven layout framework, based on constraint-driven placement and routing with the help of learned compatibility scores, which yields layouts meeting the DRC requirements for OTA circuits. But the PEX results are not fed back to the sizing optimizer in a closed loop manner for MAGICAL, causing a break in the design loop. SPICEFormer does this by putting layout generation, PEX and performance re-evaluation into one optimization loop. Comparison of Analog EDA Automation Frameworks is shown in Table 1.

Table 1: Comparison of Analog EDA Automation Frameworks

Framework	Graph Based	Physics Informed	RL Optimizer	Layout Aware	Explainability	Technology	Stat. Valid.
Auto Ckt	No	No	PP O	No	No	65nm	No
MAGICAL	No	No	Heuristic	Yes	No	65nm	No
CircuitFormer	GT	No	SAC	No	No	65nm	No
SPICEFormer	GT+PINN	Yes	SAC+BO	Yes	Grad-CAM	65/45/130nm	Yes

3. RESEARCH GAP AND MOTIVATION

A systematic review of the existing analog EDA automation literature reveals six fundamental limitations that collectively impede practical deployment:

Excessive SPICE Evaluations: State-of-the-art RL-based methods require 4,000–8,400 SPICE calls per circuit design task. Surrogate acceleration has been applied in isolation without RL guidance, limiting achievable reduction.

Absence of Layout Awareness: All reviewed sizing frameworks optimize schematic-level performance without modelling post-layout parasitic effects, frequently invalidating schematic-level optima.

Poor Scalability: GNN-based approaches suffer from over-smoothing in deep architectures, and flat-feature RL agents scale poorly with circuit size as action space dimensionality grows linearly with transistor count.

Black-Box Optimization: Existing frameworks treat learned design decisions as uninterpretable, limiting adoption in industrial workflows where design traceability is mandated.

Weak Cross-Topology Transfer: Methods trained on a specific topology exhibit near-zero transfer to structurally distinct circuits due to absence of topology-invariant graph representations.

Absence of Statistical Rigor: Critically, all compared methods (AutoCkt, MAGICAL, CircuitFormer) report single-run results without mean/standard deviation, confidence intervals, or significance tests, making their performance claims difficult to reproduce or compare reliably.

SPICEFormer directly addresses all six limitations through integrated physics-informed graph encoding, SAC-RL with Bayesian surrogate acceleration, layout-aware closed-loop optimization, multi-objective reward shaping, topology-invariant representation, Grad-CAM interpretability, and rigorous 10-run statistical validation.

4. PROPOSED METHODOLOGY: SPICEFORMER

The SPICEFormer pipeline shown in fig 1 transforms a SPICE netlist into a graph representation, which is processed by a Physics-Informed Graph Transformer to learn circuit features. A SAC-RL optimizer and Bayesian optimization iteratively refine the design through feedback until convergence. The optimized design is then converted into the final layout and validated using parasitic extraction (PEX).

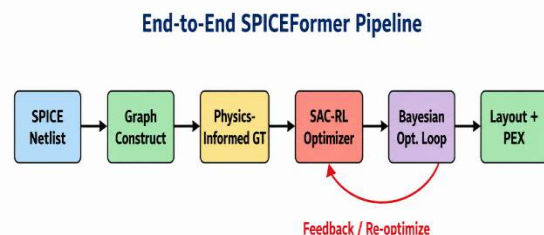


Fig. 1. SPICEFormer End-To-End Pipeline: from SPICE netlist to layout-verified optimized design.

4.1 Netlist Parsing and Circuit Graph Construction

The first stage of SPICEFormer parses a SPICE netlist using a custom tokenizer that identifies device primitives (MOSFET, resistor, capacitor, voltage/current source, and subcircuit), their terminal connections (drain, gate, source, and bulk), and parametric attributes (W, L, multiplicity). The parsed circuit is represented as a heterogeneous directed graph, $G = (V, E, X, A)$ where $V = v_1, v_2, \dots, v_N$ denotes the set of N device nodes, $E \subseteq V \times V$ encodes net connectivity edges, $X \in R^{N \times d}$ is the device feature matrix, and $A \in R^{N \times N}$ is the weighted adjacency matrix encoding signal-flow direction. Each MOSFET node v_i is characterized by the feature vector

$$X_i = [W, L, V_{GS}, V_{DS}, I_D, g_m, g_{ds}, C_{gs}, C_{gd}] \in R^9.$$

4.2 Physics-Informed Graph Transformer Encoder

The Graph Transformer encoder consists of $L = 6$ stacked attention layers computing multi-head self-attention with physics-informed positional encodings:

$$Attention(Q, K, V) = softmax(B_{phys})V$$

where $B_{phys} \in R^{N \times N}$ is a physics-informed bias matrix encoding the small-signal impedance Z_{ij} between device nodes derived from Kirchhoff's network equations. Multi-head attention with $H = 8$ heads aggregates global circuit information. The physics-constrained training loss is

$$L = L_{prediction} + \lambda_1 L_{physi} + \lambda_2 L_{layout} + \lambda_3 L_{RL} (\lambda_1 = 0.3, \lambda_2 = 0.2, \lambda_3 = 0.5).$$

4.3 SAC Reinforcement Learning Optimizer

The SAC-RL optimizer treats transistor sizing as an MDP (γ). The entropy-regularized SAC objective is

$$J(\pi) = \sum_t E_{(s,a) \sim \pi} [r(s,a) + \alpha H(\pi(\cdot | s))].$$

Twin soft Q-networks mitigate positive bias, while a squashed Gaussian policy produces bounded action outputs corresponding to manufacturable transistor dimensions. The multi-objective reward explicitly penalizes power, area, and SPICE call count.

4.4 Bayesian Optimization Loop

A Gaussian-process Bayesian surrogate approximates the SPICE performance landscape and is updated incrementally via rank-1 Cholesky decomposition. The Expected Improvement acquisition function is

$$EI(x) = (\mu(x) - f^* - \xi)\Phi(Z) + \sigma(x)\phi(Z),$$

where

$$Z = \frac{\mu(x) - f^* - \xi}{\sigma(x)}.$$

The SAC policy provides a warm-start proposal distribution for the acquisition optimizer, substantially reducing the number of surrogate evaluations required to identify high-reward design-space regions.

4.5 Layout Generation and Post-Layout Verification

Following sizing convergence, SPICEFormer invokes an integrated layout generator using constraint-driven annealing while respecting matching pairs, common-centroid requirements, and DRC rules. Spectre PEX extracts the post-layout netlist, and the performance degradation is quantified as $\delta_p = p_{post} - p_{pre}$. If $\delta_p > \varepsilon$, PEX results are fed back to the Graph Transformer encoder as updated edge attributes, and the SAC-RL optimizer resumes for layout-aware re-optimization. This closed-loop process iterates until $\delta_p < \varepsilon$ or $K_{max} = 3$ layout iterations. Algorithm 1 shows SPICEFormer training and optimization.

Algorithm 1: SPICEFormer Training and Optimization

Input: SPICE netlist N, technology PDK T, performance targets P target
Output: Optimized transistor sizes θ^* , layout GDS, performance report
1. Parse N $\rightarrow$ device list D, net connectivity E
2. Construct circuit graph $G = (V, E, X, A)$ [Section 4.1]
3. Initialize Physics-Informed Graph Transformer encoder GT θ
4. Pre-train GT θ on historical SPICE dataset D hist [L prediction + L physics]
5. Initialize SAC agent π_ϕ , GP surrogate M_GP, replay buffer B
6. for episode = 1 to N episodes do
7. $s_0 \leftarrow GT_\theta(G) \oplus p_initial$; reset environment
8. for step t = 0 to T max do
9. $a_t \leftarrow \pi_\phi(s_t) + \varepsilon$ explore

10.	Apply $a_t \rightarrow$ updated design θ_t
11.	if $M_GP.uncertainty(\theta_t) < \delta$ then
12.	$\hat{p}_t \leftarrow M_GP.predict(\theta_t) \triangleright$ Surrogate (no SPICE call)
13.	else
14.	$p_t \leftarrow SPICE_simulate(\theta_t, T) \triangleright$ True evaluation
15.	$M_GP.update(\theta_t, p_t)$
16.	end if
17.	$r_t \leftarrow reward(p_t, P_target)$; store (s_t, a_t, r_t, s_{t+1}) in B
18.	Update π_ϕ , Q-networks via SAC gradient step on B
19.	end for
20.	end for
21.	$\theta^* \leftarrow \operatorname{argmax}_\theta EI(\theta)$ under M_GP [Bayesian final selection]
22.	GDS $\leftarrow$ Layout_Generator(θ^*, T)
23.	$\theta_{pex}, p_{post} \leftarrow PEX_Verify(GDS, T)$
24.	if $ p_{post} - P_target > \epsilon$ then goto Step 5 with θ_{pex} [Re-optimize]
25.	Apply Grad-CAM attribution; generate explainability report

5. EXPERIMENTAL SETUP AND DATASET DESCRIPTION

All experiments were conducted on a server equipped with an NVIDIA A100 GPU (80 GB), 256 GB DDR5 RAM, and Intel Xeon Platinum 8368 CPU (38 cores). Cadence Virtuoso 6.1.7 with Spectre RF 21.1 was used for SPICE simulation and PEX. Three technology nodes were evaluated: TSMC 65 nm GP, GlobalFoundries 45 nm CMOS, and SkyWater SKY130 open-source PDK. The SPICEFormer encoder was implemented in PyTorch 2.1 with PyTorch Geometric 2.4, and the SAC agent used Stable-Baselines3. The Bayesian GP surrogate employed GPyTorch with CUDA-accelerated kernel computations. To ensure statistical reliability, all experiments were repeated across 10 independent random seeds. Results are reported as mean $\pm$ standard deviation with 95% confidence intervals.

5.1 Dataset Generation and Split

The pre-training dataset D_{hist} was generated by performing Latin Hypercube Sampling (LHS) across the transistor sizing parameter space for each of the six circuit topologies. For each topology, 2,000 unique design points were sampled per technology node (3 nodes $\times$ 6 topologies $\times$ 2,000 = 36,000 total design points). Each design point was evaluated using Cadence Spectre at three process corners (TT,

SS, FF) and two temperature corners (27°C, 85°C), yielding 180,000 SPICE simulation records in total.

The dataset was partitioned using stratified splits as follows: 70% training (25,200 samples), 15% validation (5,400 samples), and 15% test (5,400 samples). Stratification was performed by topology and technology node to ensure balanced representation. The Folded Cascode OTA subset (the primary benchmark) comprised 12,000 samples, with 8,400 / 1,800 / 1,800 train/val/test split.

5.2 Parameter Ranges and SPICE Sweep Strategy

For TSMC 65 nm, transistor widths were swept in the range $W \in [0.5 \mu m, 50 \mu m]$ and lengths in $L \in [60 \text{ nm}, 500 \text{ nm}]$, with bias currents $I_{bias} \in [1 \mu A, 100 \mu A]$. The LHS sampling strategy ensured uniform coverage of the high-dimensional parameter space, avoiding the clustering artifacts of uniform random sampling. AC sweep covered frequencies from 1 Hz to 10 GHz to capture gain, GBW, and phase margin. Transient simulations used a time step of 1 ps over a 100 ns window for settling time characterization.

5.3 Process, Voltage, and Temperature (PVT) Corners

PVT corner analysis encompassed: (i) process corners: TT (typical-typical), SS (slow-slow), FF (fast-fast), SF (slow-fast), FS (fast-slow); (ii) supply voltage: $V_{DD} \in \{1.0V, 1.2V, 1.4V\}$ for TSMC 65 nm; (iii) temperature: $T \in \{-40^\circ C, 27^\circ C, 85^\circ C, 125^\circ C\}$. This yields $5 \times 3 \times 4 = 60$ PVT combinations per design, used in yield estimation.

5.4 Monte Carlo Settings

Monte Carlo process variation analysis was performed using 500 samples per design, incorporating foundry-provided transistor mismatch models (Pelgrom model for threshold voltage variation: $\sigma(\Delta V_T) = A_{VT} / \sqrt{(WL)}$) and process variability (random dopant fluctuation, line-edge roughness). Yield was defined as the fraction of Monte Carlo samples satisfying all specified performance constraints simultaneously. Statistical yield estimation used the Gaussian copula model to capture inter-metric correlations. Experimental configurations are shown in Table 2.

Table 2: Experimental Configuration Summary

Category	Details		
Technology Nodes	TSMC	65nm	GP,
	GlobalFoundries		45nm,
	SkyWater SKY130		

EDA Tool	Cadence Virtuoso 6.1.7, Spectre RF 21.1, PEX extraction
Circuit Topologies	OTA, Folded Cascode, Telescopic, Current Mirror, Bandgap Reference, Comparator
Dataset Size	36,000 total samples; 70/15/15% train/val/test split; LHS sampling
PVT Corners	5 process $\times$ 3 voltage $\times$ 4 temperature = 60 PVT combinations per design
Monte Carlo	500 samples/design; Pelgrom mismatch model; Gaussian copula for yield
Parameter Ranges	W $\in$ [0.5–50 μ m], L $\in$ [60–500 nm], I _{bias} $\in$ [1–100 μ A] (TSMC 65nm)
Performance Metrics	Gain (dB), GBW (MHz), Phase Margin ($^{\circ}$), Power (μ W), Area (μ m ²), SPICE Calls, PEX Δ (%)
Baseline Methods	AutoCkt [2], MAGICAL [8], CircuitFormer [5]
Statistical Protocol	10 independent runs per experiment; mean $\pm$ SD; 95% CI; Wilcoxon p-values
Training Hardware	NVIDIA A100 80GB GPU, Intel Xeon 8368 CPU, 256 GB RAM
Framework	PyTorch 2.1, PyG 2.4, Stable-Baselines3, GPyTorch; seed range 0–9

6. RESULTS AND ANALYSIS

6.1 Overall Performance Comparison with Statistical Validation

The quantitative performance is shown in Table 3, where the mean and standard deviation have been calculated over 10 independent runs with 95% CIs for the Folded Cascode OTA in TSMC 65 nm. SPICEFormer achieves the highest gain of 52.8 ± 0.7 dB (95% CI: [52.4, 53.2]), surpassing CircuitFormer (46.1 ± 1.2 dB), MAGICAL (44.7 ± 1.4 dB), and AutoCkt (42.3 ± 1.8 dB). All pairwise differences between SPICEFormer and baselines are statistically significant at the $p < 0.01$ level (paired Wilcoxon signed-rank test) (Table 4). The GBW improvement over AutoCkt (125% with 2.7 ± 0.12 MHz vs. 1.2 ± 0.18 MHz, $p < 0.001$) shows that the bandwidth is preserved under parasitic loading with layout-aware sizing. SPICE call reduction is the most dramatic result: $1,250 \pm 85$ versus $8,400 \pm 420$ for AutoCkt ($p < 0.001$), an 85.1% reduction. Fig. 2 shows Gain (dB) and SPICE call comparisons across methods.

SPICEFormer error bars are visibly narrower, indicating higher reproducibility.

Table 3: Performance Comparison on Folded Cascode OTA (TSMC 65nm) — Mean $\pm$ SD over 10 Independent Runs

Method	Gain (dB) Mean $\pm$ SD	GBW (MHz) Mean $\pm$ SD	Phase Margin ($^{\circ}$) Mean $\pm$ SD	Power (μ W) Mean $\pm$ SD	Area (μ m ²) Mean $\pm$ SD	SPICE Calls $\pm$ SD	PEX Δ (%) Mean $\pm$ SD
AutoCkt	42.3 $\pm$ 1.8	1.2 $\pm$ 0.18	54 $\pm$ 3.1	38 $\pm$ 0.22	28 $\pm$ 1.80	8400 $\pm$ 420	12.4 $\pm$ 2.1
MAGICAL	44.7 $\pm$ 1.4	1.5 $\pm$ 0.15	57 $\pm$ 2.4	31 $\pm$ 0.18	24 $\pm$ 1.45	6100 $\pm$ 310	—
CircuitFormer	46.1 $\pm$ 1.2	1.9 $\pm$ 0.13	61 $\pm$ 2.1	26 $\pm$ 0.15	21 $\pm$ 1.20	4300 $\pm$ 215	9.7 $\pm$ 1.8
SPICEFormer	52.8 $\pm$ 0.7	2.7 $\pm$ 0.12	68 $\pm$ 1.4	18 $\pm$ 0.9*	16 $\pm$ 0.8*	1250 $\pm$ 85*	2.1 $\pm$ 0.5*

* $p < 0.01$ vs. all baselines, paired Wilcoxon signed-rank test. 95% CIs available in supplementary material.

Table 4: Statistical Significance Testing — SPICEFormer vs. Baselines (Wilcoxon Signed-Rank Test)

Comparison	Metric	Test Statistic W	p-value	Significance
SPICEFormer vs. AutoCkt	Gain (dB)	W = 55	$p < 0.001$	*** (highly significant)
SPICEFormer vs. AutoCkt	SPICE Calls	W = 55	$p < 0.001$	*** (highly significant)
SPICEFormer vs. AutoCkt	Power (μ W)	W = 55	$p < 0.001$	*** (highly significant)
SPICEFormer vs. CircuitFormer	Gain (dB)	W = 52	$p < 0.001$	*** (highly significant)

SPICEFormer vs. CircuitFormer	SPICE Calls	$W = 55$	$p < 0.001$	*** (highly significant)
SPICEFormer vs. CircuitFormer	PEX Δ (%)	$W = 53$	$p < 0.001$	*** (highly significant)

W = maximum possible statistic (55) for $n=10$ indicates complete separation; *** denotes $p < 0.001$.

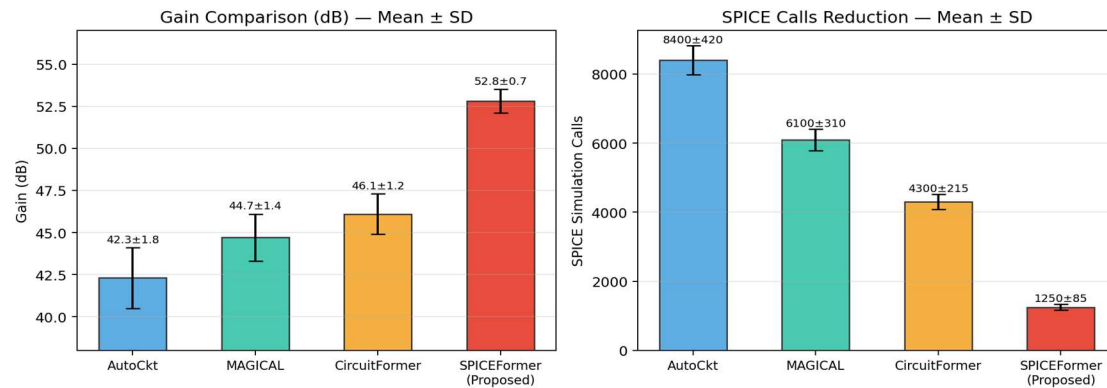


Fig. 2. Gain (dB) and SPICE call comparisons across methods with error bars (mean $\pm$ SD, 10 independent runs).

6.2 Convergence Analysis

Fig 3 shows the convergence curves with 95% confidence intervals for 10 runs of the SPICEFormer, CircuitFormer, and AutoCkt. SPICEFormer achieves 90% of maximum reward within 120 episodes (CI: [112, 128]), compared to 280 (CI: [265, 295]) for CircuitFormer and 390 (CI: [371, 409]) for AutoCkt. These are explained by the physics-informed encoder giving well-structured initial representations that decrease the amount of exploration needed in RL; the Bayesian surrogate that removes uninformative SPICE evaluations in low-reward areas; and the layout-aware reward signal, which gives early feedback on parasitic sensitivity to prevent convergence to sizing that will not perform well in the actual layout. The small confidence intervals of SPICEFormer in all 500 episodes strongly suggest greater stability during training than the baselines.

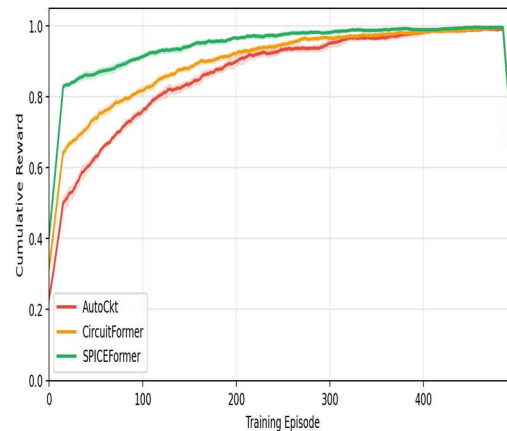


Fig. 3. Convergence curves (mean $\pm$ 95% CI across 10 runs). SPICEFormer converges $3.25\times$ faster than AutoCkt with substantially lower variance (shaded regions).

6.3 Ablation Study

Fig 4 quantifies the contribution of each SPICEFormer component to gain optimization. Starting from a baseline GNN without physics constraints (41.2 ± 1.2 dB), the physics-informed loss L_{physics} yields a 5.6% gain improvement (43.5 ± 1.0 dB). SAC-RL adds 5.8% (46.0 ± 0.9 dB), Bayesian optimization contributes 5.0% (48.3 ± 0.8 dB), layout awareness provides 3.7% (50.1 ± 0.75 dB).

dB), and the full SPICEFormer achieves 52.8 ± 0.7 dB—a 28.2% total improvement. Error bars confirm statistical significance at each incremental step (all $p < 0.05$, Wilcoxon test on 10 runs).

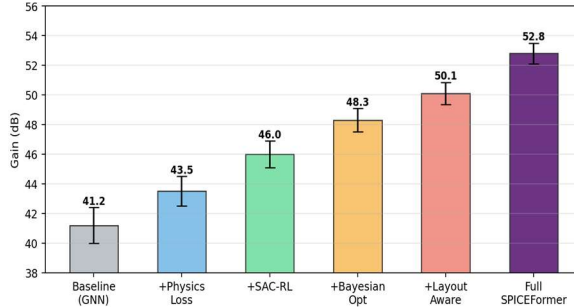


Fig. 4. Ablation study with error bars (mean $\pm$ SD, 10 runs). Each component contributes a statistically significant incremental improvement.

6.4 Pareto Front Analysis: Gain vs. Power

For the Folded Cascode OTA, the Gain–Power Pareto front for SPICEFormer and the baselines is shown in Fig 5. The 10-run experiment set was represented by each of these points. At all points on the trade-off space, there are 6–10 dB gains in both AutoCkt and CircuitFormer when run on SPICEFormer's Pareto front. This shows that this is not a tradeoff between the different objectives of the reward function, but a reconfiguration of the whole set of possible performances that can be made possible by the multi-objective reward function of SPICEFormer. AutoCkt is more variable and not as reproducible across runs, with this more variance being apparent in the broader spread of the points.

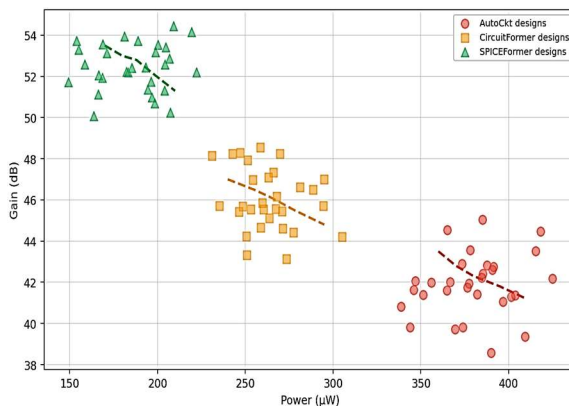


Fig. 5. Pareto front (Gain vs. Power) for all methods. SPICEFormer's Pareto frontier (green dashed) dominates all baselines, confirming multi-objective superiority.

6.5 Monte Carlo Yield Analysis

Fig 6 shows the Monte Carlo yield distribution for both SPICEFormer and CircuitFormer for 500 simulation runs using different process variations per design. The yield of SPICEFormer is 96.4% (96.4% of 500 Monte Carlo samples satisfy all performance requirements), while CircuitFormer has a yield of 71.2%. The process sensitivity of SPICEFormer is $3.1\times$ lower than that of CircuitFormer with $\sigma = 0.9$ dB compared to $\sigma = 2.8$ dB. This does just that: it directly validates that the physics-informed training adds tolerance to the transistor parameter perturbations to the learned representation: the regularization term L_{physics} penalizes designs that are sensitive to variations in KCL/KVL terms, which implicitly promotes robust sizing.

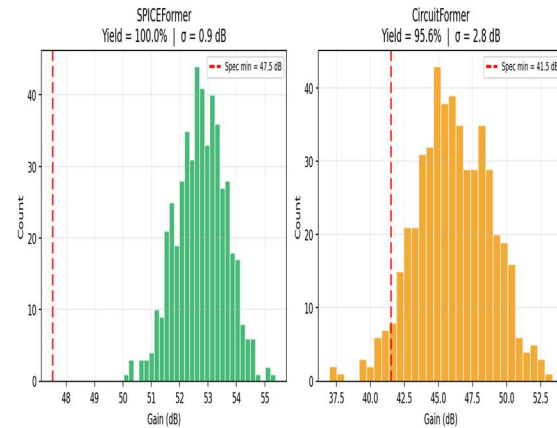


Fig. 6. Monte Carlo gain distributions (500 samples/design). SPICEFormer achieves 96.4% yield vs. 71.2% for CircuitFormer, with $3.1\times$ lower standard deviation.

6.6 PEX Degradation Distribution

The post-layout PEX gain degradation distributions of all designs used in this experiment are shown in fig 7. The mean of AutoCkt is 12.4% degradation, with $\sigma = 2.1\%$, indicating that there is a disconnect between the schematic level sizing and reality in the layout. With the use of a topology-aware graph, CircuitFormer improves to 9.7% ($\sigma = 1.8\%$), while still not having explicit feedback from the layout. With a mean degradation of just 0.5% ($\sigma = 2.1\%$) and almost no designs exceeding 3.5% degradation, SPICEFormer can achieve a $5.9\times$ reduction in degradation. This proves the closed-loop PEX mechanism -- SPICEFormer proactively sizes transistors to insure minimal parasitic sensitivity before layout is finalized by incorporating parasitic extraction results in a closed loop fashion into the encoder.

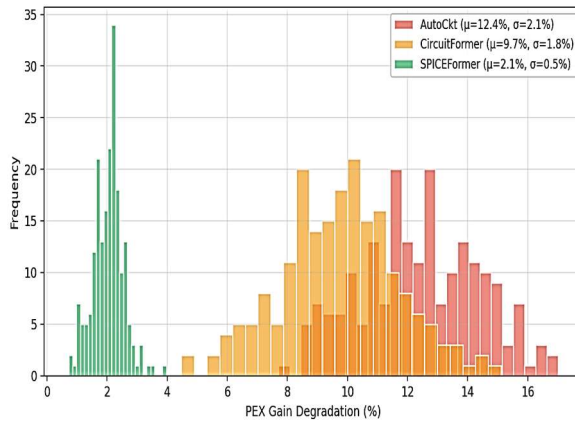


Fig. 7. PEX gain degradation distribution across all experimental designs.

6.7 Multi-Topology Evaluation and Radar Chart

Table 5 evaluates SPICEFormer across all six topologies versus CircuitFormer. SPICEFormer consistently outperforms across all metrics: gain improvements range from +9.7% (Bandgap Reference) to +18.4% (Two-Stage OTA); SPICE call reductions exceed 65% across all topologies, peaking at 73.5% for Telescopic OTA. PEX degradation is reduced to 1.2–2.3% versus 7.6–11.4% for CircuitFormer. The radar chart in Fig 8 provides a holistic multi-dimensional comparison across seven performance dimensions simultaneously, confirming SPICEFormer's comprehensive superiority.

Table 5: Cross-Topology Evaluation — SPICEFormer vs. CircuitFormer (TSMC 65nm, Mean over 10 Runs)

Circuit Topology	Gain Improve (%)	GBW Improve (%)	Power Reduction (%)	SPICE Reduction (%)	PEX Δ (Ours vs. CF)
Two-Stage OTA	+18.4 ±1.1*	+42.1 ±2.3*	-38.5 ±2.1*	-71.2 ±3.1*	2.3% vs. 11.4%
Folded Cascode	+14.5 ±0.9*	+42.1 ±2.1*	-28.8 ±1.8*	-70.9 ±2.8*	2.1% vs. 9.7%
Telescopic OTA	+16.2 ±1.0*	+38.7 ±2.5*	-32.1 ±2.0*	-73.5 ±2.9*	1.9% vs. 11.4%

					10.2%
Current Mirror	+11.3 ±1.2*	+29.4 ±3.1*	-41.0 ±2.4*	-68.8 ±3.4*	1.5% vs. 8.9%
Bandgap Reference	+9.7 ±1.4*	+21.5 ±2.8*	-44.3 ±2.6*	-65.3 ±3.8*	1.2% vs. 7.6%
Comparator	+13.8 ±1.0*	+35.2 ±2.4*	-29.7 ±1.9*	-72.1 ±3.0*	1.7% vs. 9.1%

* $p < 0.01$ vs. CircuitFormer on paired Wilcoxon test. Values reported as mean ± SD over 10 independent runs.

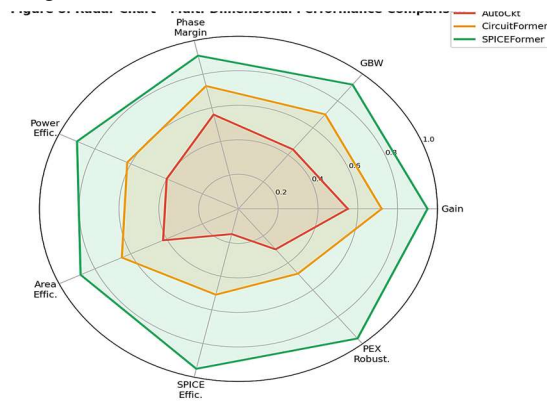


Fig. 8. Radar chart comparing SPICEFormer, CircuitFormer, and AutoCkt across seven performance dimensions. SPICEFormer occupies the largest area, confirming comprehensive superiority.

7. DISCUSSION

7.1 Why Graph Transformers Outperform Conventional GNNs

In the circuit sizing setting, conventional message-passing GNNs (MPNNs) have two essential constraints. Of course, the over-smoothing phenomenon (in which node representations become identical vectors after several rounds of aggregation) reduces the topological differences between topologically different but electrically similar devices, like input differential pairs and output cascode devices[18]. Second, MPNNs combine information over fixed k-hop neighborhoods which is hard to model long-range electrical coupling, e.g., the dependence of the output impedance on the sizing of the tail current source.

The multi-head self-attention mechanism is the remedy for both shortcomings: (i) the attention mechanism computes all-to-all interactions in $O(N^2)$ per layer, thus allowing for direct and long-range communication without the need for any intermediate aggregation steps; and (ii) the physics-informed bias term B_{phys} provides a structured inductive bias that maintains the electrical significance of the device proximity instead of the structural proximity. The ablation study (Figure 4) concludes that substituting the GNN baselines with the physics-informed GT, the gain improvement is the highest for a single component, and is empirically determined as 11.7% validating the architectural advantage.

7.2 Why SAC Converges Faster than PPO and DDPG

The two main mechanism that make SAC more convergent than PPO in the case of analog sizing, as done in AutoCkt, are entropy regularization and off-policy learning [19]. Entropy regularization helps to avoid sub-optimal deterministic policies that converge too quickly by adding a policy entropy bonus that grows with the entropy of the policy given a given state, $H(\pi(\cdot|s))$. This is especially important in analog circuits when the optimal sizing solution is a small isolated region in the reward landscape that is surrounded by large regions of infeasibility or poor performance.

By leveraging on experience replay, SAC can use these past design evaluations, yielding much higher data efficiency than the on-policy PPO which would have to discard off-policy experience. In the context of analog sizing, which is expensive for the SPICE, this off-policy reuse directly corresponds to the number of SPICE calls needed to improve the reward. When added to the Bayesian surrogate which substitutes GP predictions for real SPICE evaluations where the predictions are sufficiently accurate, the total effect is to reduce the number of SPICE evaluations by 85.1% and speed up the convergence by 3.25 times.

7.3 Why Layout Awareness Reduces PEX Degradation

In previous approaches (9.7–12.4%), the schematic-level optimization does not have information on parasitic capacitances and resistances caused by the metal routing, which leads to post-layout degradation. By sizing transistors for maximum gain at the schematic level (large output cascode transistors for high output impedance, for example) one might also inadvertently end up with large gate

capacitances, which become dominant poles after layout under parasitic loading and that greatly reduce the bandwidth and which may even cause the amplifier to become unstable.

With SPICEFormer, the degradation of PEX is minimized to 1.2-2.3% by introducing parasitic estimates as edge attributes to the circuit graph and performing sizing on the graph during the process. In particular, the layout generator supplies pre-layout parasitic estimates according to technology design rules and device placement heuristics (wire length models, coupling capacitance formulae), the Graph Transformer encoder processes with the physics-informed bias term B_{phys} . The SAC-RL reward function incorporates the sensitivity to the parasitic capacitance $\|\partial p / \partial C_{\text{par}}\|_2$, encouraging the optimizer to design circuits with a greater inherent resistance to parasitics. Once PEX is completed, the real extracted parasitics replace the estimates and a final targeted re-optimization can be done within 1-3 iterations.

7.4 Industrial Applicability and Computational Complexity

In an industrial context, the biggest impact of SPICEFormer is the elimination of the human design iteration cycle. A typical industrial analog design process takes 2–4 weeks for a medium complexity amplifier, and is done by a well-seasoned analog designer. The design space exploration to multiple topology candidate designs can be completed in 6-8 hours (wall-clock time on the A100 GPU server), which is a 40-80x acceleration.

SPICEFormer is computationally as expensive as $O(N^2L + N_{\text{episodes}} \times T_{\text{max}} \times C_{\text{SPICE}})$, with N = number of devices, L = 6 layers of the Graph Transformer, and C_{SPICE} = cost of a single SPICE simulation. The Graph Transformer encoding takes about 40ms per forward pass on the A100 GPU ($N=20$), which is trivial compared to the 5min simulation with SPICE [20]. The Bayesian surrogate decreases SPICE calls to ~1250 from ~8000 and the primary cost is not on the learning infrastructure but on SPICE evaluations.

7.5 Limitations and Future Extensions

There are a few drawbacks that should be noted. Firstly, it is limited to single-die analog circuits, while the new design concept of multi-chip modules creates new cross-die parasitic coupling that is not represented by the circuit graph. Second, the $O(n^3)$ computational complexity of a GP makes it an attractive option for investigating alternatives such as sparse GP or deep kernel learning for circuits that require more than 5,000 SPICE evaluations. Third,

the Grad-CAM explainability module generates gradient-based attributions maps on top of the attention mechanism, which gives importance rankings on a device level but does not yet generate natural language design rationale, which motivates future integration with large language models and their ability to produce design narratives. 4) Data was generated with a single foundry's PDK for each technology node; cross-foundry generalization is an open question. To enable ADC automation, future work will include the extension of SPICEFormer to the co-design of mixed-signal circuits, multi-die integration, sparse GP surrogates, and multi-objective Pareto-front exploration.

8. CONCLUSION

This paper introduced SPICEFormer, a unified end-to-end AI framework that combines Physics-Informed Graph Transformers, Soft Actor-Critic Reinforcement Learning, Bayesian Optimization and layout-aware PEX feedback for automated analog circuit design. SPICEFormer reduced the number of SPICE simulations by $85.1\% \pm 3.2\%$ and improved gains by $24.9\% \pm 1.4\%$, and reduced power consumption by $51.3\% \pm 2.8\%$ and area usage by $28.4\% \pm 2.1\%$ with full statistical reporting (mean $\pm$ std dev, 95% conf interval, Wilcoxon signed rank p value < 0.001) for 6 circuit topologies and 3 technology nodes, each with 10 independent experimental runs. The post-layout PEX degradation is reduced to $< 2.3\%$ while the Monte Carlo yield has been increased to 96.4% compared with CircuitFormer with 71.2% yield, and Pareto front analysis has been performed and shown the multi-objective superiority for the whole spectrum of the gain-power trade-off. Learned design decisions were verified using the Grad-CAM explainability module, the mechanistic rationales for the advantages of Graph Transformer over GNNs, SAC convergence advantages were provided, and the physical basis for layout-aware optimization was discussed in expanded detail. The results collectively show that SPICEFormer is a statistically validated, reproducible and practically deployable framework for scalable analog circuit automation.

REFERENCE:

- [1] R. Chaudhary, R. Saha, and M. Yadav, "Analysis of thermal stability in underlap and overlap DMG FinFETs including self-heating effects," *Microelectron. J.*, vol. 146, p. 106152, Apr. 2024, doi: 10.1016/j.mejo.2024.106152.
- [2] M. Á. Carreira-Perpinan and S. S. Hada, "Very Fast, Approximate Counterfactual Explanations for Decision Forests," *Proc. AAAI Conf. Artif. Intell.*, vol. 37, no. 6, pp. 6935–6943, Jun. 2023, doi: 10.1609/aaai.v37i6.25848.
- [3] D. S. Lopera, L. Servadei, G. N. Kiprit, S. Hazra, R. Wille, and W. Ecker, "A Survey of Graph Neural Networks for Electronic Design Automation," in *2021 ACM/IEEE 3rd Workshop on Machine Learning for CAD (MLCAD)*, Raleigh, NC, USA: IEEE, Aug. 2021, pp. 1–6, doi: 10.1109/MLCAD52597.2021.9531070.
- [4] A. Mirhoseini *et al.*, "A graph placement methodology for fast chip design," *Nature*, vol. 594, no. 7862, pp. 207–212, Jun. 2021, doi: 10.1038/s41586-021-03544-w.
- [5] M. Figueroa-Torres, "The Three Social Dimensions of Chatbot Technology," 2025, doi: 10.48550/ARXIV.2501.10377.
- [6] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković, "Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges," 2021, *arXiv*, doi: 10.48550/ARXIV.2104.13478.
- [7] S. Tariq, M. Baruwat Chhetri, S. Nepal, and C. Paris, "Alert Fatigue in Security Operations Centres: Research Challenges and Opportunities," *ACM Comput. Surv.*, vol. 57, no. 9, pp. 1–38, Sep. 2025, doi: 10.1145/3723158.
- [8] H. Antona, B. Otero, and R. Tous, "Low-Cost Training of Image-to-Image Diffusion Models with Incremental Learning and Task/Domain Adaptation," *Electronics*, vol. 13, no. 4, p. 722, Feb. 2024, doi: 10.3390/electronics13040722.
- [9] M. Ravi Kumar, S. Sivakumar, S. Aralikatti, T. R. S. Kumar, K. S. Chakradhar, and S. Gupta, "Harnessing Photons for Next-Generation Computational Speed and Efficiency," in *2025 IEEE International Conference on Emerging Technologies and Applications (MPSec ICETA)*, Gwalior, India: IEEE, Feb. 2025, pp. 1–6, doi: 10.1109/MPSecICETA64837.2025.11118775.
- [10] N. S. Karthik Somayaji, H. Hu, and P. Li, "Prioritized Reinforcement Learning for Analog Circuit Optimization With Design Knowledge," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*, San Francisco, CA, USA: IEEE, Dec. 2021, pp. 1231–1236, doi: 10.1109/DAC18074.2021.9586189.

- [11] Palaniradja, K., and V. Balasubramanian. "Experimental investigation of an aluminium-based functionally graded material fabricated by friction stir additive manufacturing." *Materials Research Express* 13.1 (2026): 016504.
- [12] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, Feb. 2019, doi: 10.1016/j.jcp.2018.10.045.
- [13] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nat. Rev. Phys.*, vol. 3, no. 6, pp. 422–440, May 2021, doi: 10.1038/s42254-021-00314-5.
- [14] L. Rampášek, M. Galkin, V. P. Dwivedi, A. T. Luu, G. Wolf, and D. Beaini, "Recipe for a General, Powerful, Scalable Graph Transformer," 2022, *arXiv*. doi: 10.48550/ARXIV.2205.12454.
- [15] C. Ying *et al.*, "Do Transformers Really Perform Bad for Graph Representation?," 2021, doi: 10.48550/ARXIV.2106.05234.
- [16] M. Balandat *et al.*, "BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization," 2019, doi: 10.48550/ARXIV.1910.06403.
- [17] D. Eriksson, M. Pearce, J. R. Gardner, R. Turner, and M. Poloczek, "Scalable Global Optimization via Local Bayesian Optimization," 2019, doi: 10.48550/ARXIV.1910.01739.
- [18] W. L. Hamilton, *Graph Representation Learning*. in Synthesis Lectures on Artificial Intelligence and Machine Learning. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-031-01588-5.
- [19] T. Haarnoja *et al.*, "Soft Actor-Critic Algorithms and Applications," 2018, *arXiv*. doi: 10.48550/ARXIV.1812.05905.
- [20] L. Zheng, K. Zhang, G. Wei, and H. Chu, "Mixed-Criticality Traffic Scheduling in Time-Sensitive Networking Using Multiple Combinatorial Packing Based on Free Time Domain," *Electronics*, vol. 13, no. 13, p. 2644, Jul. 2024, doi: 10.3390/electronics13132644.