A Data-Driven Analog Circuit Synthesizer with Automatic Topology Selection and Sizing

Souradip Poddar, Ahmet Budak, Linran Zhao, Chen-Hao Hsu, Supriyo Maji, Keren Zhu,
Yaoyao Jia, and David Z. Pan
ECE Department, The University of Texas at Austin, Austin, TX, USA
{souradippddr1, ahmetfarukbudak, lrzhao, chenhaoh}@utexas.edu, supriyo.maji@austin.utexas.edu, keren.zhu@utexas.edu,
yjia@austin.utexas.edu, dpan@ece.utexas.edu

***Abstract*—Despite significant recent advancements in analog design automation, analog front-end design remains a challenge characterized by its heavy reliance on human designer expertise together with extensive trial-and-error simulations. In this paper, we present a novel data-driven analog circuit synthesizer with automatic topology selection and sizing. We propose a modular approach to build a comprehensive, parameterized circuit topology library. Instead of starting from an exhaustive dataset, which is often not available or too expensive to build, we build an adaptive topology dataset, which can later be enhanced with synthetic data generated using variational autoencoders (VAE), a generative machine learning technique. This integration bolsters our methodology's predictive capabilities, minimizing the risk of inadvertent oversight of viable topologies. To ensure accuracy and robustness, the predicted topology is re-sized for verification and further performance optimization. Our experiments, which involve over 360 OPAMP topologies and over 540K data points demonstrate our framework's capability to identify optimal topology and its sizing within minutes, achieving design quality comparable to that of experienced designers.**

I. Introduction

The analog front-end circuit design comprises two fundamental stages: circuit topology selection and device sizing. While contemporary algorithms like [1]–[4] excel at device sizing within a specified topology, these methods still rely on expensive simulations and they only work for a given circuit topology. The quest for practical and effective circuit topology searches remains an ongoing challenge. Their real-world application requires further improvement due to various challenges, like excessive runtime caused by in-the-loop simulations [5], [6], need for substantial design knowledge to formulate equations [7]–[9], availability of a limited set of circuit topologies [10], computationally expensive and time-consuming approaches [11], and the possibility of generating invalid topologies [12]. While some studies prioritize novel circuit topology generation over topology selection [13], [14], industrial practice favors selecting from a known, tested set, especially for simpler specifications. This inclination is driven by the aim to achieve robust reliability in post-fabrication performance, a characteristic further reinforced in designs supported by validated silicon-proven observations. As a result, the broader potential of analog circuit synthesis and sized netlist generation for previously unseen performance specifications remains limited.

In this paper, we propose a novel data-driven strategy for swift analog circuit synthesis, encompassing both automated topology selection and device sizing. In contrast to methods like [5], [6], which may take days to converge and impose constraints on subblock complexity for improved convergence rates, our data-centric approach can generate a complete sized schematic netlist in minutes. Our approach entails an initial one-time effort to collect simulation-based performance data, which is subsequently leveraged in a data-centric flow to facilitate efficient topology search and device sizing. While comprehensive, diverse performance datasets for each topology are essential to capture design capabilities, the compute-intensive nature of SPICE simulations is a severe limitation. This data-driven approach is scalable and has an edge over traditional optimization schemes [15] regarding time and computational effort. In this paper, we have validated our concept for the Operational Amplifiers (OPAMPs) case.

The key contributions of this work are as follows.

- We present a novel data-driven methodology for analog circuit synthesis, encompassing automatic topology selection and sizing. We establish a topology library and corresponding data repository with a one-time data collection effort, incorporating both simulation-based and synthetic performance data with ML assistance. When generating a new netlist for an unseen set of performance specifications, our algorithm leverages this library and repository to select a topology and optimize its device sizing automatically.
- We propose a modular approach for building a topology library from scratch. We use an analog automatic device sizer for simulation-based performance data collection for each OTA topology. Additionally, we leverage VAE to enhance the density of the performance data using the collected dataset.
- We propose an efficient method for netlist generation leveraging the established data repository. We select a ranked list of topologies based on the target user specification and the data repository. These topologies undergo verification and automatic device sizing optimization to meet the target specification.
- We have validated our methodology for OPAMPs, encompassing 4000+ topologies, of which a representative subset of around 360 based on performance spectra was curated by our expert designers for this paper to showcase our method's robustness and interpretability. Our

979-8-3503-4859-0/DATE24/© 2024 EDAA

approach can generate optimal-sized topology netlists in approximately 10 minutes and is easily scalable to even more extensive libraries.

II. Preliminaries

In this section, we discuss established automatic analog sizing techniques such as DNN-Opt (Section II-A) and highlight the definition of a key design quality metric, Figure of Merit (FOM), that will guide our approach. An introduction to the generative machine learning model, VAE, is also presented in Section II-B

A. Automatic Analog Sizing

After generating a library comprising a diverse array of OPAMP topologies, we use an automatic analog sizer to size the devices and generate performance data. Compared to conventional approaches, the transferability of Deep Reinforcement Learning (DRL) based techniques improves the efficiency of device sizing in different technology nodes [16].

In this work, we employed DNN-Opt [3], a state-of-the-art multi-threaded sizing optimization tool to size the devices in each OTA topology and generate the simulation-based dataset. DNN-Opt is an online optimization method where the simulation response is obtained during the optimization iterations. DNN-Opt is engineered to harness the strengths of actor-critic algorithms, a key feature of reinforcement learning (RL), and population-based optimization techniques. The learning-based optimization engine globally searches the optimal device parameters that maximize the circuit performance.

Given an objective performance metric $f_0(\mathbf{x})$, and m constraint metrics $\{f_i(\mathbf{x}) \leq 0 \text{ for } i = 1, \ldots, m\}$, DNN-Opt evaluates the quality of a design by defining a Figure of Merit (FoM) in the following form:

$$\text{FoM}(\mathbf{x}) = w_0 \times f_0(\mathbf{x}) + \sum_{i=1}^{m} \min\left(1, \max(0, w_i \times f_i(\mathbf{x}))\right) \quad (1)$$

where w_i is the weighting factor, a $\max(\cdot)$ clipping is used for equating designs after constraints are met, and $\min(\cdot)$ is used to prevent single constraint violation from dominating FoM value. We will utilize this formulation in our work.

B. Variational Autoencoders

Variational autoencoders (VAEs) [17] [18] constitute a powerful class of generative models in machine learning that operate within a probabilistic framework $p(x)$, where x represents the observed data points. VAEs consist of two neural networks: encoder (E_ϕ), which converts x to lower-dimensional latent variable z space (prior distribution $p(z)$) and decoder (D_θ), which reconstructs them back to the data space $(p(x|z))$, resulting in a likelihood of $p(x) = \int p(x|z)p(z)dz$. Fig. 1 shows a VAE architecture.

VAEs employ the Evidence Lower Bound (ELBO) as a surrogate objective for computational ease to maximize log-likelihood $\log p(x)$. ELBO can be defined as:

$$ELBO = \mathbb{E}_{q(z|x)}[\log p(x|z)] - \text{KL}[q(z|x)||p(z)],$$

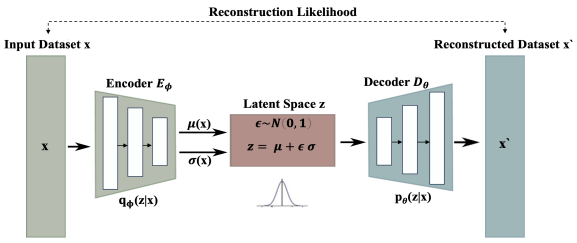


Fig. 1: Architecture of a VAE

where $\mathbb{E}_{q(z|x)}[\log p(x|z)]$ represents the reconstruction loss and $\text{KL}[q(z|x)||p(z)]$ is the Kullback-Leibler (KL) divergence between the approximate posterior distribution $q(z|x)$ assuming prior distribution $p(z)$ to be a simple Gaussian distribution.

We capture crucial tradeoffs among analog specifications using VAE to generate synthetic performance data to enrich the relatively small simulation-based dataset to explore the innate design potential embedded within each topology, saving time and resources.

III. Algorithms

Fig. 2 illustrates our overall proposed framework for generating sized analog schematic netlists. Our overall flow can be broadly divided into two main parts: performance dataset generation;topology selection, verification and sizing. Section III-A provides a detailed explanation of the performance dataset generation phase. This phase involves a one-time data collection effort that includes establishing a parameterized topology library and a subsequent data repository generation for each topology, incorporating simulation-based data from DNN-Opt along with VAE-generated synthetic performance data. Once the data repository is established, users can submit queries with diverse, unseen performance specifications, and our algorithm will leverage this library and repository to select a topology and optimize its device sizing automatically. Section III-B delves into the topology selection, verification and sizing phase in detail.

In this paper, our specific area of focus will be Operational Amplifiers (OPAMPs), often known as Operational Transconductance Amplifiers (OTAs), and on the following 14 performance metrics: DC gain (Gain), quiescent current (IQ), 3dB-bandwidth (BW), common-mode rejection ratio (CMRR), power supply rejection ratio (PSRR), minimum common-mode input voltage (VCM Min), maximum common-mode input voltage (VCM Max), minimum output voltage (Vout Min), maximum output voltage (Vout Max), slew rate (SR), rise time, fall time, input-referred noise density at 1KHz (IRN Density), and phase margin (PM). These metrics can be categorized into two groups: those where higher values are better, and those where lower values are more desirable as demonstrated in Table I.

TABLE I: Performance Metrics

Higher is better	DC Gain, BW, CMRR, PSRR, VCM Max, Vout Max, SR, PM
Lower is better	IQ, VCM Min, Vout Min, Rise time, Fall time, IRN Density

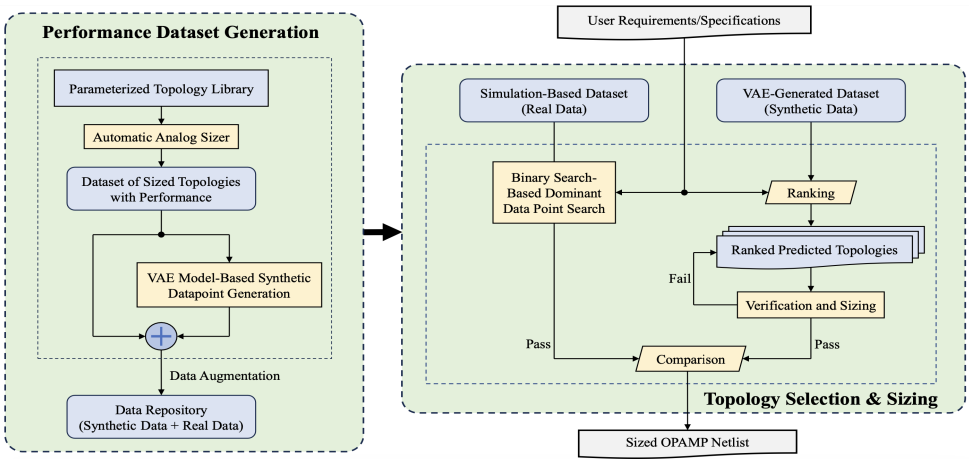


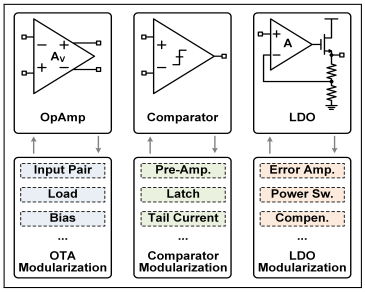
Fig. 2: The overall framework for Analog Circuit Synthesis

A. *Performance Dataset Generation*

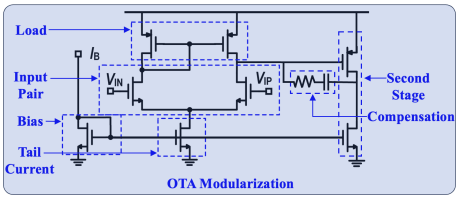
Fig. 2 highlights the overall process of generating simulation-based performance data and subsequently enriching it with a synthetic dataset. It comprises of two main sections: Topology Library Generation (Section III-A1, and Real and Synthetic Performance Dataset Generation(Section III-A2).

1) Topology Library Generation: Conducting a comprehensive topology search necessitates a sufficiently extensive search space. However, constructing an extensive topology library from scratch presents a challenge. We noticed that different categories of analog circuits can be normalized to a standard format customized for their specific circuit type. For instance, comparators can be divided into modules like the pre-amplifier, latch, tail current, and more. Similarly, low-dropout regulators (LDOs) consist of the error amplifier, power switch, compensation, and more. Likewise, OPAMPs can be deconstructed into different fundamental building blocks, such as differential input pairs, tail current, bias generator, compensation network, and load components, among others. The overall idea of this modular concept for some analog circuit types is showcased in Fig. 3a. In order to facilitate the sizing tool optimizer in generating performance data for each topology in the subsequent stage, each module is parameterized, employing the sizes of the contained MOSFETs as parameters. Varied selections within each module type, with appropriate constraints, then create different parameterized functional topologies for a given circuit type. We have developed SKILL codes to automate this entire process. This modular strategy helps expedite the swift construction of an initial foundational library comprising various parameterized standard topologies for the specified circuit type within minutes.

Fig. 3b illustrates an OTA modularization diagram. Multiple options are available within each module element of an OTA. For instance, the input pair elements can vary from a standard P/NMOS differential pair, with or without source degeneration, to an inverter-based configuration, among other options. The load can be implemented using a resistor, an active load, a current source, a cascode current source, or an inverter-based load, among other options. Furthermore, each OPAMP can



(a) Modular breakdown of analog circuits



(b) Modular Approach for building OPAMPs

Fig. 3: A Modular Approach for Topology Library Generation

TABLE II: Representative examples of some key OTA modules

Module Name	Examples
Input pairs	Differential pair [with/without source degeneration], Inverter-based
Load	Resistors, Active load, Current source, Folded-cascode, Telescopic cascode
Tail current	Current mirror, Cascode current mirror, Resistor
Bias generation	Diode connected mosfet, Magic battery current mirror, Resistor
Compensation	Miller cap, Nulling resistor, Indirect compensation
Second stage	Common-source, Common drain, Differential OTA

serve as an individual building block, allowing us to increase the diversity of topologies by cascading different OPAMPs. Table II provides representative examples of some fundamental building blocks of an OTA that apply to both NMOS and PMOS configurations.

We use a valid topology filter incorporating designer expertise-derived compatibility constraints to guide the module combination process. This constraint-based method is essential

because certain building blocks have inherent compatibility requirements. For instance, merging an NMOS-input pair with a PMOS tail current would generate an invalid topology. Similarly, an NMOS-input pair has to be combined with a PMOS-based load. Applying these constraints removes the invalid topologies, narrowing our search space to encompass only those OPAMP architectures that demonstrate functional behavior. Following the above approach, we swiftly generated an initial foundational library comprising approximately 4,000 valid standard parameterized OPAMP topologies. From this pool, our expert designers curated a representative subset of around 360 topologies based on performance spectra for this paper to showcase our method's robustness and interpretability.

2) Real and Synthetic Performance Dataset Generation: After selecting a library of 360 parameterized OPAMP topologies (as discussed in Section III-A1), we employ an automatic analog sizer to size the devices within each OPAMP topology and obtain the corresponding spice simulation-based performance metrics. For this study, we utilized the DNN-Opt sizing tool [3] for this purpose. To enhance the diversity of the DNN-Opt-generated performance dataset for each topology, we implement six distinct configurations of objectives and constraints in the sizing tool. This approach ensures we encompass a broad range of performance behaviors and variations within our dataset. Due to the computational intensity of spice simulations, we limited our simulation-based performance dataset to around 1,000 data points per topology, balancing resource usage with valuable performance insights.

We further enrich the depth of our simulation-based performance dataset by leveraging the generative capabilities of variational autoencoders to generate new synthetic data points that closely resemble the inherent characteristics of the performance distribution. VAEs excel at learning continuous and smooth data representations in a latent space modeled as a Gaussian distribution, ensuring that small changes in the latent variables correspond to small, meaningful changes in the data space. This characteristic facilitates easy random sampling and interpolation between data points within this latent realm, facilitating a systematic exploration of gradual shifts and transitions across dimensions. This capability is crucial for understanding tradeoffs among analog specifications and comprehensively exploring a topology's performance capabilities. We individually train VAE models based on the simulation-based performance dataset corresponding to each topology and generate synthetic performance data specific to each. In the experiments showcased in this paper, we've generated synthetic data approximately 10X the size of the real dataset per topology. Using this data repository we can swiftly obtain the predicted topology and its sizing within minutes for a given user specification, showcasing the effectiveness of our data-driven approach.

B. Topology Selection, Verification and Sizing

Fig. 2 illustrates the comprehensive process that follows when a user inputs objective and constraints. As depicted, we adopt a two-fold strategy that combines a binary search based topology selection method (Section III-B1) applied to the actual simulation-based performance dataset for identifying dominant data points, complemented by a parallel synthetic data based topology ranking (Section III-B2), to glean insights beyond the scope of the binary search approach alone. The ranked topologies are automatically sized in the verification and sizing flow (Section III-B3). Ultimately, the final sized topology from the synthetic dataset is compared with that from the binary search method based on performance.

1) Binary Search based Topology Selection: We leverage the real simulation-based performance data repository in this stage. To facilitate constraint-based searches, we sort the dataset of topologies into 14 lists, based on each metric. When a user queries with objectives and constraints, we perform a binary search on sorted lists for each metric and extract a set of feasible data points for each. The intersection of these points across all sets provides a list of candidate topologies. Binary search, with a time complexity of $O(\log n)$, ensures swift operations. This set of candidate topologies is prioritized based on their best objective number. The top-ranking schematic netlist is designated as the output. This output will be subsequently compared with the VAE-based verified prediction based on their respective objective numbers.

2) Synthetic Data based Topology Ranking: We utilize the synthetic data repository to systematically rank topologies based on user specifications (objective and constraints). We will adopt the Figure of Merit (FOM) definition as described in Equation 1 in Section II-A, to evaluate the design quality and rank the topologies. According to this definition, for a given set of user specifications and two sets of performance metrics $\mathbf{x}$ and $\mathbf{y}$, if $FoM(\mathbf{x}) \leq FoM(\mathbf{y})$, it signifies that $\mathbf{x}$ aligns more closely with the user's constraints than $\mathbf{y}$. We sort the topologies in ascending order based on their best synthetic dataset-derived FOMs. In cases where multiple topologies yield a zero value, we prioritize those with superior synthetic objective numbers. We proceed with the top three from this ranked list for subsequent verification.

3) Verification and Sizing: We initiate a verification process after obtaining the ranked list of synthetic data-based candidate topologies (Section III-B2). DNN-Opt [3] is invoked on this sorted list using constraints and objectives tailored to the user's specifications. We iterate till we achieve a Figure of Merit (FOM) 0. For efficiency, we strategically initialize the elite population sample of the sizing tool with points from our simulation-based dataset closest to the user query. Some points with poor FOM are also included for robust convergence. This flow validates our predictions and pushes each topology's FOM to its lowest possible value, optimizing performance to match user-defined criteria closely, even in challenging scenarios, swiftly generating simulation-verified sized schematic netlists.

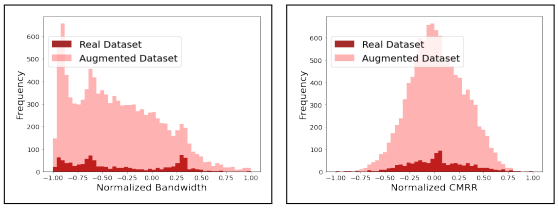
In the final step, we compare the sized topology from the synthetic dataset with that from the real dataset, if available, based on their performance. If both topologies meet the user-defined constraints, we give precedence to the one with the

superior objective for netlisting. However, if one or both fail to meet the constraints fully, we prioritize the one that most closely aligns with or fulfills the user specifications.

IV. Experimental Results

Our schematic simulation results are based on freePDK45nm [19] technology node and obtained using SPICE (Simulation Program with Integrated Circuit Emphasis) simulator in Analog Design Environment (ADE). The experiments were conducted in a Linux environment on a 4-core 3.3GHz CPU.

Fig. 4 demonstrates the enriched performance data through histograms comparing the real dataset with its augmented counterpart for some of the normalized specification dimensions for a randomly picked topology: Single-stage OTA with telescopic cascode (NMOS).



(a) Normalized BW (b) Normalized CMRR

(c) Normalized VCM Min (d) Normalized Slew Rate

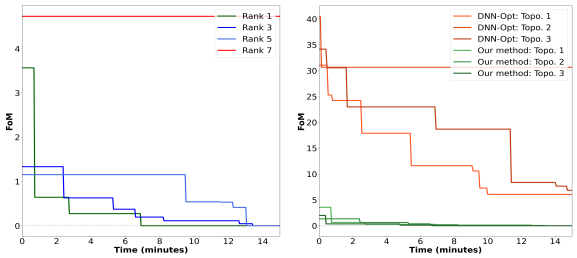
Fig. 4: Histogram for Real Dataset v/s Augmented Dataset for Single-stage OTA with telescopic cascode (NMOS) topology

To demonstrate the efficiency of our algorithm, we present the test results for our algorithm for six distinct test cases, each with varying objectives and constraints, in Table III, juxtaposed with designer expectations and DNN-Opt performance numbers for a comprehensive comparison. Fig. 5a highlights the effectiveness of our ranking approach. It showcases the performance trends for topologies ranked 1, 3, 5, and 7 for a test case through an FOM plot over time. Our algorithm can swiftly generate a fully-sized circuit netlist in just 6.87 minutes to meet the requirements of this test case. In Table III, we observe that for test case 1, both binary search and VAE-generated synthetic data yield similar predictions. However, in test cases 2, 3, and 5, binary search falters due to a dearth of data points in proximity to the user query space. In contrast, the synthetic data-based approach adeptly identifies, ranks and sizes the topologies. Test case 4 depicts a case where binary search-predicted result outperforms the synthetic data-based result, highlighting how binary search and VAE go hand in hand in providing sized schematic netlists for a given user specification. Table III affirms that our predictions meet designer expectations, noting a maximum run time of 12 minutes. With a cap on the maximum number of samples produced by the sizing tool, in the worst case where our algorithm runs for all three top-ranked topologies, the total time taken would be approximately $3 \times 12 = 36$ minutes. If the user specification is very complex, our algorithm reports the sized topology closest to the user specification (has the lowest FOM), as demonstrated in test case 6.

A. Comparison with existing works

We compare our algorithm against two other methods: a) MOJITO, an established topology selection approach, and b) DNN-Opt for sizing after topology selection. Existing topology selection methods like MOJITO [20] involve extensive in-the-loop simulations making them computationally demanding. In addition, they impose subblock complexity constraints to enhance convergence. If we scale the reported runtime of 7 days for 101904 topologies in the library [20] to our setup, achieving comparable results based on a user specification would take approximately 7 days $\times$ (5$\times$2$\times$4 cores $\times$ 2 GHz) / (4 cores $\times$ 3.3 GHz), i.e., 42.4 days of computational effort for 101,904 topologies, or about 3.6 hours for 360 topologies. Our methodology, being data-centric, inherently surpasses these conventional approaches by 5-10X times, with a worst-case time of approximately 36 minutes even for a very challenging specification.

We also conduct an extensive comparison between our approach and DNN-Opt, assuming that the topology is given for DNN-Opt, which serves as our baseline. We present the Figure of Merit (FOM) plotted against time for our approach and DNN-Opt, showcasing the top-ranked and verified topologies corresponding to two distinct test cases in Fig. 5b. The start and end FOM values in Fig. 5b, along with the corresponding FOM @500 samples and run-time column values in Table III, highlight the efficiency of our data-centric approach. It enables the generation of a fully-sized schematic netlist within minutes, surpassing DNN-Opt by 5–10X times.



(a) Ranking efficiency (b) Comparison with DNN-Opt

Fig. 5: FOM v/s time plots

V. Conclusion

This paper introduces a novel data-driven approach for rapid analog circuit synthesis, including automated topology selection and device sizing. A constraint-based modular approach was also introduced to swiftly establish foundational parameterized analog topology libraries from scratch, which is

TABLE III: Comparative Analysis of the test results: Designer Expectations v/s Our Method v/s DNN-Opt

Test case	Designer Expectation	Binary Search-Based Prediction	VAE-Based Prediction	Final Predicted Sized Topology	Our Method: FOM @500 samples	Our Method: Run-time (mins)	DNN-Opt: FOM @500 samples	DNN-Opt: Run-time (mins)
1	Two-stage OTA with cascode load	Two-stage OTA with cascode load (NMOS)	Two-stage OTA with cascode load (NMOS)	Two-stage OTA with cascode load (NMOS)	0	10.95	7.03	73
2	Standard Two-stage OTA	Failed	Standard Two-stage OTA (PMOS)	Standard Two-stage OTA (PMOS)	0	6.875	6.05	63
3	Single-stage inverter-based OTA with cascode	Failed	Single-stage inverter-based OTA with cascode	Single-stage inverter-based OTA with cascode	0	6.593	1.66	48.7
4	Two-stage OTA with cascode load, Single-stage OTA with folded-cascode load	Two-stage OTA with cascode load (NMOS)	Single-stage OTA with folded cascode load and cascode tail current (NMOS)	Two-stage OTA with cascode load (NMOS)	0	10.7	7.03	73
5	Standard Two-stage OTA	Failed	Standard Two-stage OTA (NMOS)	Standard Two-stage OTA (NMOS)	0	10.959	6.5	67
6	Single-stage inverter-based OTA with cascode	Failed	Single-stage inverter-based OTA with cascode	Single-stage inverter-based OTA with cascode	0.15	10.9	3.04	48.5

beneficial when a pre-existing topology repository is not readily available. Simulation-based performance data and VAE-generated synthetic data were incorporated through a one-time data collection effort to capture each topology's design capabilities, notably streamlining search and sizing processes. By combining binary search with synthetic data-driven ranking, verification, and an initial data-driven sizing strategy, we efficiently produced optimal-sized topology netlists in minutes. We rigorously validated our methodology on 360+ diverse OPAMP topologies. This work represents an advancement towards data-driven analog circuit synthesis. We aim to extend its application to intricate analog circuits, advancing data collection techniques to bolster VAE model robustness for future studies.

VI. Acknowledgement

This work was supported in part by NSF under grants CCF-1704758 and CCF-2112665, SRC under task 3160.007, and an equipment donation from Nvidia.

References

[1] H. Wang, K. Wang, J. Yang, L. Shen, N. Sun, H.-S. Lee, and S. Han, "GCN-RL circuit designer: Transferable transistor sizing with graph neural networks and reinforcement learning," in *Proc. DAC*. IEEE, 2020, pp. 1–6.

[2] K. Settaluri, A. Haj-Ali, Q. Huang, K. Hakhamaneshi, and B. Nikolic, "Autockt: Deep reinforcement learning of analog circuit designs," in *Proc. DATE*. IEEE, 2020, pp. 490–495.

[3] A. F. Budak, P. Bhansali, B. Liu, N. Sun, D. Z. Pan, and V. K. Chandramouli, "DNN-Opt: An RL inspired optimization for analog circuit sizing using deep neural networks," in *Proc. DAC*, 2021, pp. 1219–1224.

[4] A. F. Budak, D. Smart, B. Swahn, and D. Z. Pan, "APOSTLE: Asynchronously parallel optimization for sizing analog transistors using DNN learning," in *Proc. ASPDAC*, 2023, pp. 70–75.

[5] T. McConaghy, P. Palmers, G. Gielen, and M. Steyaert, "Simultaneous multi-topology multi-objective sizing across thousands of analog circuit topologies," in *Proc. DAC*, 2007, pp. 944–947.

[6] P. Palmers, T. McConnaghy, M. Steyaert, and G. Gielen, "Massively multi-topology sizing of analog integrated circuits," in *Proc. DATE*, 2009, pp. 706–711.

[7] P. Veselinovic, D. Leenaerts, W. Van Bokhoven, F. Leyn, F. Proesmans, G. Gielen, and S. W., "A Flexible Topology Selection Program as Part of an Analog Synthesis System," in *Proc. EDTC*, 1995, pp. 119–123.

[8] T. McConaghy, P. Palmers, G. Gielen, and M. Steyaert, "Automated Extraction of Expert Knowledge in Analog Topology Selection and Sizing," in *Proc. ICCAD*, 2008, pp. 392–395.

[9] Z. Zhao and L. Zhang, "An Automated Topology Synthesis Framework for Analog Integrated Circuits," *IEEE TCAD*, vol. 39, no. 12, pp. 4325–4337, 2020.

[10] P. Maulik, L. Carley, and R. Rutenbar, "Integer programming based topology selection of cell-level analog circuits," *IEEE TCAD*, vol. 14, no. 4, pp. 401–412, 1995.

[11] Ž. Rojec, Á. Bűrmen, and I. Fajfar, "Analog circuit topology synthesis by means of evolutionary computation," *Engineering Applications of Artificial Intelligence*, vol. 80, pp. 48–65, 2019.

[12] J. Lu, Y. Li, F. Yang, L. Shang, and X. Zeng, "High-Level Topology Synthesis Method for Δ-Σ Modulators via Bi-level Bayesian Optimization," *IEEE TCAS II*, 2023.

[13] T. R. Dastidar, P. P. Chakrabarti, and P. Ray, "A synthesis system for analog circuits based on evolutionary search and topological reuse," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 2, pp. 211–224, 2005.

[14] M. Fayazi, M. T. Taba, E. Afshari, and R. Dreslinski, "AnGeL: Fully-Automated Analog Circuit Generator Using a Neural Network Assisted Semi-Supervised Learning Approach," *IEEE TCAS I*, pp. 1–14, 2023.

[15] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," 2012.

[16] A. F. Budak, D. Z. Pan, H. Chen, K. Zhu, M. Liu, M. B. Alawieh, S. Zhang, W. Shi, and X. Tang, "CAD for analog/mixed-signal integrated circuits," *Advances in Semiconductor Technologies: Selected Topics Beyond Conventional CMOS*, pp. 43–60, 2022.

[17] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[18] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[19] North Carolina State University, "FreePDK: Open-Source Process Design Kit for IC Design," https://eda.ncsu.edu/freepdk/freepdk45/, accessed: 2023.

[20] T. McConnaghy, P. Palmers, M. Steyaert, and G. Gielen, "Trustworthy Genetic Programming-Based Synthesis of Analog Circuit Topologies Using Hierarchical Domain-Specific Building Blocks," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 4, pp. 557–570, 2011.