Inverse Sizing and Exploration Based on Diffusion Models and Structural Knowledge

***Abstract*—Quick and effortless exploration of design spaces is of major importance in the face of rapid development cycles, varying specifications and increasingly complex circuits. It is also well known, that the exploitation of slack in specifications by making adjustments to transistor device sizes can lead to improved yield. We address those challenges by combining two state of the art machine learning approaches to achieve accurate generation of optimal performance points, while also allowing for the possibility to explore nearby sizings that might lead to more robust designs. To this end, a diffusion model is combined with an algorithm, that analyzes the circuit and splits the problem into simpler subproblems. The models are tested on a set of typical operational amplifiers.**

I. Introduction

Since the development of SPICE, the industry accepted method for designing analog circuits has not changed much. While the SPICE models keep becoming more complex [1], the approach of manual device tuning persists [2]. Key challenges of making analog CAD tools useful in practice are robustness, ease of use and applicability to a high range of use cases. All those aspects are not easy to grasp on a fundamental level. On the other hand, machine learning approaches excel in modeling nonlinear black box phenomena. Their combination with already existing conventional techniques has a high potential to bridge remaining gaps, that prevent the adoption of tools in industry.

Analog design is inherently an inverse problem. While the calculation of performances yield etc., is satisfactorily possible via simulation, the inverse, i.e. getting a circuit with optimal parameterization requires experience, trial and error and complex numerical routines. As shown in previous works [3]–[5]

With increasing MOSFET model complexity [1], analog design, particularly OpAmp design, remains manual in industrial practice [2], despite the existence of automatic tools. Various procedural methods for automated design have been discussed in [6]–[8]. The idea of using smaller building blocks and hierarchical strategies is successfully demonstrated, e.g., in [9]–[13]. Early on, modeling performance characteristics of circuits has been popular, e.g. by symbolic analysis [14], [15]. Some prior works in the machine learning realm follow the idea of skipping or replacing the optimization algorithm by other means such as neural networks (NNs). In [16], artificial neural networks (ANNs) are used to guess circuit sizings; due to modeling the entire netlist with one ANN and including non-optimal results in the former case, the complexity of the learning task is higher and thus large sample sizes (in [16] 16600 or roughly $16 - 160\,x$ our sample sizes) are required to learn the inverse problem within a smaller performance range. Similarly to our work, [5] uses optimal samples for training but without exploiting structural knowledge. In [17], [18], reinforcement algorithms are presented, substituting the optimizer with an NN. Here, the simulation effort is increased as multiple rounds are required for every sizing, similar to traditional optimization. In [19], the complexity of the NNs is reduced by having small networks represent single device parameters and cascading them in different ways, feeding outputs of earlier networks to the input of later networks in the tree. Unfortunately, with a circuit of e.g. 15 devices/networks, optimally ordering them in a chain can lead to a complexity of 15! in the worst case. Other works, such as [20]–[23], follow the traditional approach of treating sizing as an optimization problem or concentrate on performance modeling. While achieving good accuracy in their findings, they contend with the costly iteration process inherent in optimization.

In this paper, utilizing sampled sizings from the open-source tool FUBOCO [24], we introduce a machine learning algorithm for sizing operational amplifiers **without the need for optimization and simulation** during the deployment phase. Our approach stands out by **capitalizing on automatically extracted structural** and **functional information**, a departure from related works. This brings about benefits in terms of speed, flexibility, and accuracy. The required sample size, setup- and evaluation-time, as well as necessary expert knowledge are kept at a minimum. Additional contributions include the creation of a comprehensive database of netlists, the development of a custom loss function, and the utilization of a quasi-Newton optimizer for training purposes.

In section II, we provide an overview of the proposed method and its connection to structural analysis. In sec. **??**, we describe the sizing process in detail. We conclude with a presentation of results for two OpAmps.

II. From Structural Analysis to Circuit Sizing via Machine Learning

A. Basics of Structural and Functional Blocks

Our method heavily relies on decomposing circuits into subcircuits, which we call structural building blocks. Most building blocks are known to the analog designer by names such as *current mirror* or *differential pair*.

We further distinguish different functional variants of one and the same structural block. To this end, we assign to

Parts of this work have been done within the collaborative project HoLoDEC funded by the Federal Ministry of Education and Research Germany (BMBF) under the funding code 16ME0705.

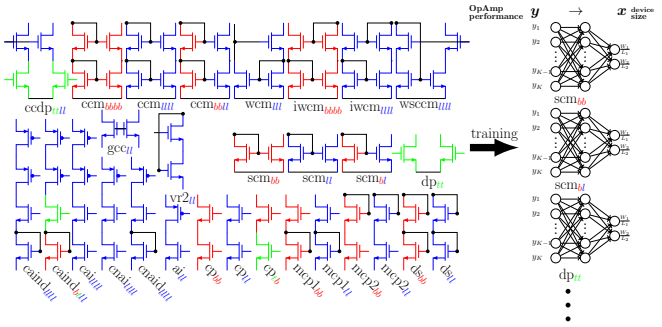


Fig. 1: Splitting various OpAmps into canonical building blocks with different functional characteristics and accumulating intrinsic design know-how by training each building block with random optimal samples from all available OpAmps. Indices b, l and t represent *bias*, *load* and *transconductance* functionality of devices. Every building block maps the overall OpAmp performance to its sizing.

every transistor in a circuit its functional property, out of three different possibilities: *bias, load* or *transconductance*. For example, a *simple current mirror* (scm) can act as *load*, but in other parts of a circuit, the same building block can act as *bias*. And sometimes, some part of a current mirror can act as *load*, while another part acts as *bias*. Figure 1 shows all structural blocks and their functional variants from our dataset. An example of decomposing a Folded-Cascode OpAmp into blocks can be seen in figure 2a. Its structural building blocks are shaded, the names annotated close by. Structural and functional block recognition is fully automated. The time required for this is much less than one second.

B. Overview of the Method

III. Conclusion

References

[1] C. Gatermann and R. Sommer, "Teaching the mosfet: A circuit designer's view," in *2022 18th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 2022, pp. 1–4.

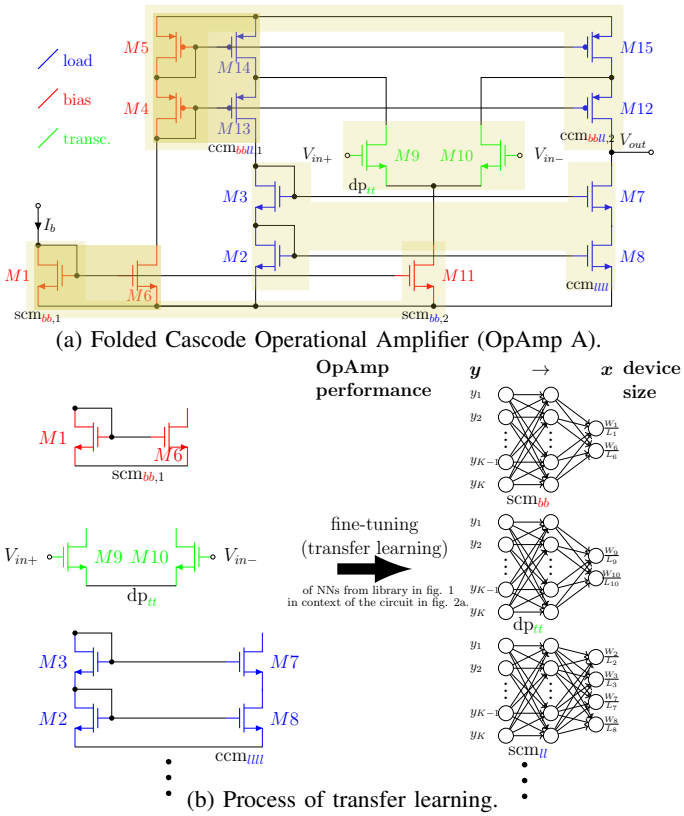
[2] G. Gielen, "Analog synthesis 3.0: Ai/ml to synthesize and test analog ics: hope or hype ?" in *2023 ACM/IEEE 5th Workshop on Machine Learning for CAD (MLCAD)*, 2023, pp. 1–1.

[3] P. Eid, F. Azevedo, R. Martins, and N. Lourenço, "Solving the inverse problem of analog integrated circuit sizing with diffusion models," in *2024 20th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 2024, pp. 1–4.

[4] M. Leibl and H. Graeb, "Optimizer-free sizing of opamps leveraging structural and functional properties," in *2024 20th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 2024, pp. 1–4.

[5] N. Lourenço, E. Afacan, R. Martins, F. Passos, A. Canelas, R. Póvoa, N. Horta, and G. Dundar, "Using polynomial regression and artificial neural networks for reusable analog ic sizing," in *2019 16th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 2019, pp. 13–16.

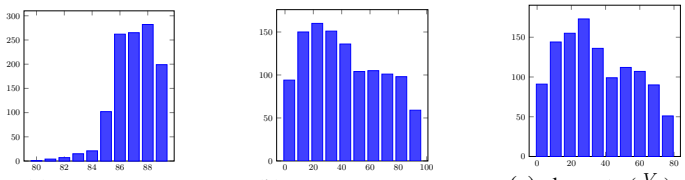
[6] B. Prautsch, U. Eichler, S. Rao, B. Zeugmann, A. Puppala, T. Reich, and J. Lienig, "Iip framework: A tool for reuse-centric analog circuit design," in *2016 13th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 2016, pp. 1–4.



(a) Folded Cascode Operational Amplifier (OpAmp A).

(b) Process of transfer learning.

Fig. 2: Instead of training one huge network per OpAmp, we can split an Opamp, e.g. OpAmp A, into blocks (shading and coloring in fig. 2a) and use weights from our network-database as initial training points to merely fine-tune our networks (2b).



(a) gain (dB) (b) trns. fr. (Mhz) (c) slew rate ($\frac{V}{\mu s}$)

(d) power (mW) (e) phase margin (f) CMRR

Fig. 3: Histograms of some performance features for the Folded-Cascode OpAmp in Fig. 2a.

[7] J. Crossley, A. Puggelli, H.-P. Le, B. Yang, R. Nancollas, K. Jung, L. Kong, N. Narevsky, Y. Lu, N. Sutardja, E. J. An, A. L. Sangiovanni-Vincentelli, and E. Alon, "Bag: A designer-oriented integrated framework for the development of ams circuit generators," in *2013 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2013, pp. 74–81.

[8] M. Schweikardt and J. Scheible, "Expert design plan: A toolbox for procedural automation of analog integrated circuit design," in *2022 18th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 2022, pp. 1–4.

[9] M. Meissner and L. Hedrich, "Feats: Framework for explorative analog topology synthesis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 2, pp. 213–226, 2015.

[10] I. Abel and H. Graeb, "Fuboco: Structure synthesis of basic op-amps by functional block composition," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 27, no. 6, jun 2022. [Online]. Available: https://doi.org/10.1145/3522738

[11] K. Mendhurwar, H. Sundani, P. Aggarwal, R. Raut, and V. Devabhaktuni, "A new approach to sizing analog cmos building blocks using pre-compiled neural network models," *Analog Integrated Circuits and Signal Processing*, vol. 70, pp. 265–281, 2012.

[12] Z. Zhao, J. Luo, J. Liu, and L. Zhang, "Signal-division-aware analog circuit topology synthesis aided by transfer learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 42, no. 11, pp. 3481–3490, 2023.

[13] A. Canelas, F. Passos, N. Lourenço, R. Martins, E. Roca, R. Castro-López, N. Horta, and F. V. Fernández, "Hierarchical yield-aware synthesis methodology covering device-, circuit-, and system-level for radiofrequency ics," *IEEE Access*, vol. 9, pp. 124 152–124 164, 2021.

[14] E. Tlelo-Cuautle, E. Martínez-Romero, C. Sánchez-López, and S. X.D.-Tan, "Symbolic behavioral modeling of low voltage amplifiers," in *2010 7th International Conference on Electrical Engineering Computing Science and Automatic Control*, 2010, pp. 510–514.

[15] P. Wambacq, F. Fernandez, G. Gielen, W. Sansen, and A. Rodriguez-Vazquez, "Efficient symbolic computation of approximated small-signal characteristics of analog integrated circuits," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 3, pp. 327–330, 1995.

[16] N. Lourenço, J. Rosa, R. Martins, H. Aidos, A. Canelas, R. Póvoa, and N. Horta, "On the exploration of promising analog ic designs via artificial neural networks," in *2018 15th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, 2018, pp. 133–136.

[17] J. Gao, W. Cao, and X. Zhang, "Rose: Robust analog circuit parameter optimization with sampling-efficient reinforcement learning," in *2023 60th ACM/IEEE Design Automation Conference (DAC)*, 2023, pp. 1–6.

[18] K. Settaluri, Z. Liu, R. Khurana, A. Mirhaj, R. Jain, and B. Nikolic, "Automated design of analog circuits using reinforcement learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 9, pp. 2794–2807, 2022.

[19] P.-O. Beaulieu, E. Dumesnil, F. Nabki, and M. Boukadoum, "Analog rf circuit sizing by a cascade of shallow neural networks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1–1, 2023.

[20] 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 Transactions on Circuits and Systems I: Regular Papers*, pp. 1–14, 2023.

[21] A. F. Budak, D. Smart, B. Swahn, and D. Z. Pan, "Apostle: Asynchronously parallel optimization for sizing analog transistors using dnn learning," in *2023 28th Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2023, pp. 70–75.

[22] G. Wolfe and R. Vemuri, "Extraction and use of neural network models in automated synthesis of operational amplifiers," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 2, pp. 198–212, 2003.

[23] K. Hakhamaneshi, M. Nassar, M. Phielipp, P. Abbeel, and V. Stojanovic, "Pretraining graph neural networks for few-shot analog circuit modeling and design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 42, no. 7, pp. 2163–2173, 2023.

[24] Abel, I. and Neuner, M. and Eick M. Fuboco: Github repository. Technical University of Munich. [Online]. Available: https://github.com/inga000/acst