Learning from Experience: Applying ML to Analog Circuit Design

Kishor Kunal[1], Tonmoy Dhar[1], Yaguang Li[2], Meghna Madhusudan[1], Jitesh Poojary[1], Arvind K. Sharma[1], Wenbin Xu[2], Steven M. Burns[3], Ramesh Harjani[1], Jiang Hu[2], Parijat Mukherjee[3], Sachin S. Sapatnekar[1]

[1] University of Minnesota, Minneapolis, MN, USA

[2] Texas A&M University, College Station, TX, USA

[3] Intel Labs, Hillsboro, OR, USA

ABSTRACT

The problem of analog design automation has vexed several generations of researchers in electronic design automation. At its core, the difficulty of the problem is related to the fact that machine-generated designs have been unable to match the quality of the human designer. The human designer typically recognizes blocks from a netlist and draws upon her/his experience to translate these blocks into a circuit that is laid out in silicon. The ability to annotate blocks in a schematic or netlist-level description of a circuit is key to this entire process, but it is a process fraught with complexity due to the large number of variants of each circuit type. For example, the number of topologies of operational transconductance amplifiers (OTAs) easily numbers in the hundreds. A designer manages this complexity by dividing this large set of variants into classes (e.g., OTAs may be telescopic, folded cascode, etc.). Even so, the number of minor variations within each class is large. Early approaches to analog design automation attempted to use rule-based methods to capture these variations, but this database of rules required tender care: each new variant might require a new rule. As machine learning (ML) based alternatives have become more viable, alternative forms of solving this problem have begun to be explored.

Our effort is part of the ALIGN (Analog Layout, Intelligently Generated from Netlists) project [2, 3], which is developing open-source software for analog/mixed-signal circuit layout [1]. Our specific goal is to translate a netlist into a physical layout, with 24-hour turnaround and no human in the loop. The ALIGN flow inputs a netlist whose topology and transistor sizes have already been chosen, a set of performance specifications, and a process design kit (PDK) that defines the process technology. The output of ALIGN is a layout in GDSII format.

The philosophy of ALIGN is to use a mix of algorithmic techniques, template-driven design, and ML. ALIGN mimics the expert designer by identifying hierarchies within the netlist. These hierarchies correspond to building blocks of the design that can be assembled during layout, in the style of the human expert. At the lowest level of this hierarchy is an individual transistor; these transistors are then combined into larger fundamental primitives (e.g., differential pairs, current mirrors), then modules (e.g., OTAs), up through several levels of hierarchy to the system level.

The ALIGN methodology is particularly useful in deeply scaled technologies, where the complexity of design rules implies a smaller design space for layout (e.g., due to unidirectional routing rules), combined with increased difficulty for the layout engineer in comprehending these rules, and has been applied to generate circuit layouts both in FinFET and in bulk technologies.

This presentation overviews our experience in the use of machine learning based methods for analog design. Specifically, we touch upon (a) results from our efforts so far, (b) appropriate methods for mixing ML methods with traditional algorithmic techniques for solving the larger problem of analog layout, (c) limitations of ML methods, particularly in relation to the construction and availability of training sets, and (d) techniques for overcoming these limitations to deliver workable solutions for analog layout automation.

CCS CONCEPTS

• **Hardware → Electronic design automation**; **Physical design (EDA)**; **Analog and mixed-signal circuit optimization**; • **Computing methodologies → Machine learning**; **Neural networks**.

KEYWORDS

Analog layout automation, machine learning

ACM Reference Format:

Kishor Kunal[1], Tonmoy Dhar[1], Yaguang Li[2], Meghna Madhusudan[1], Jitesh Poojary[1], Arvind K. Sharma[1], Wenbin Xu[2], Steven M. Burns[3], Ramesh Harjani[1], Jiang Hu[2], Parijat Mukherjee[3], Sachin S. Sapatnekar[1]. 2020. Learning from Experience: Applying ML to Analog Circuit Design. In *Proceedings of the 2020 International Symposium on Physical Design (ISPD '20), March 29-April 1, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3372780.3378172

ACKNOWLEDGMENTS

This work was supported in part by DARPA IDEA program under SPAWAR contract N660011824048.

REFERENCES

[1] Accessed February 1, 2020. ALIGN: Analog Layout, Intelligently Generated from Netlists. https://github.com/ALIGN-analoglayout/ALIGN-public.

[2] K. Kunal, T. Dhar, M. Madhusudan, J. Poojary, A. K. Sharma, W. Xu, S. M. Burns, J. Hu, R. Harjani, and S. S. Sapatnekar. 2020. GANA: Graph Convolutional Network Based Automated Netlist Annotation for Analog Circuits. In *Proc. DATE.* ACM, New York, NY.

[3] K. Kunal, M. Madhusudan, A. K. Sharma, W. Xu, S. M. Burns, R. Harjani, J. Hu, D. A. Kirkpatrick, and S. S. Sapatnekar. 2019. ALIGN: Open-source analog layout automation from the ground up. In *Proc. DAC.* ACM, New York, NY, 77–80.

ISPD '20, March 29-April 1, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7091-2/20/03.
https://doi.org/10.1145/3372780.3378172