EDALearn: A Comprehensive RTL-to-Signoff EDA Benchmark for Democratized and Reproducible ML for EDA Research

(Invited Paper)

Jingyu Pan
jingyu.pan@duke.edu
Duke University
Durham, NC, USA

Chen-Chia Chang
chenchia.chang@duke.edu
Duke University
Durham, NC, USA

Zhiyao Xie
eezhiyao@ust.hk
Hong Kong University of Science and Technology
Hong Kong

Yiran Chen
yiran.chen@duke.edu
Duke University
Durham, NC, USA

Hai (Helen) Li
hai.li@duke.edu
Duke University
Durham, NC, USA

Abstract

The application of Machine Learning (ML) in Electronic Design Automation (EDA) for Very Large-Scale Integration (VLSI) design has garnered significant research attention. Despite the requirement for extensive datasets to build effective ML models, most studies are limited to smaller, internally generated datasets due to the lack of comprehensive public resources. In response, we introduce EDALearn, the first holistic, open-source benchmark suite specifically for ML tasks in EDA. This benchmark suite presents an end-to-end flow from synthesis to physical implementation, enriching data collection across various stages. It fosters reproducibility and promotes research into ML transferability across different technology nodes. Accommodating a wide range of VLSI design instances and sizes, our benchmark aptly represents the complexity of contemporary VLSI designs. Additionally, we provide an in-depth data analysis, enabling users to fully comprehend the attributes and distribution of our data, which is essential for creating efficient ML models. Our contributions aim to encourage further advances in the ML-EDA domain.

CCS Concepts

• **Hardware** → **Electronic design automation**; **Logic synthesis**; **Physical design (EDA)**; • **Computing methodologies** → **Machine learning**.

Keywords

Electronic Design Automation, Machine Learning, Dataset

ACM Reference Format:
Jingyu Pan, Chen-Chia Chang, Zhiyao Xie, Yiran Chen, and Hai (Helen) Li. 2024. EDALearn: A Comprehensive RTL-to-Signoff EDA Benchmark for Democratized and Reproducible ML for EDA Research: (Invited Paper). In *Proceedings of 2024 International Conference on Computer-Aided Design (ICCAD'24).* ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

ICCAD '24, October 27–31, 2024, New York, NY, USA

ACM ISBN 979-8-4007-1077-3/24/10.
https://doi.org/10.1145/3676536.3697116

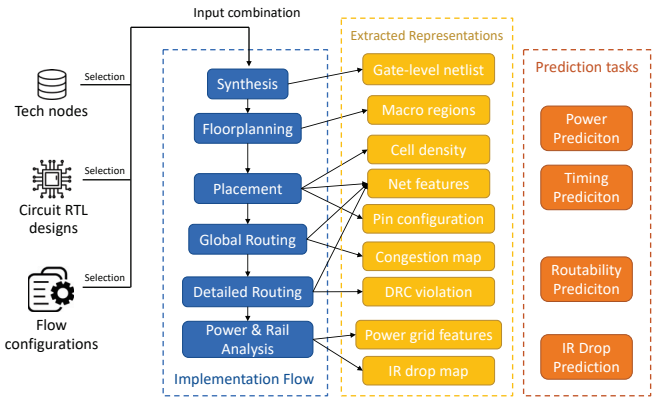


Figure 1: Overview of the EDALearn benchmark.

1 Introduction

The process of VLSI circuit design is typically bifurcated into two stages: front-end and back-end design. While the former focuses on the circuit's functionality, the latter is responsible for translating the circuit into manufacturable geometries or layouts. However, as technology continues to advance, back-end design often proves to be a time-intensive process. This is largely due to the iterative information exchange between the various design stages during the optimization phase.

To mitigate these delays and streamline the process, cross-stage prediction techniques have been introduced. By replacing long feedback loops between design stages with more localized loops within each stage, these techniques have significantly accelerated the back-end design process. Machine learning (ML), known for its predictive capabilities, has been widely employed to expedite

Table 1: Contribution and Limitations of Previous Benchmarks

Benchmark	Contribution	Limitations
CircuitNet [4]	First open-source dataset for ML applications in EDA.	Insufficient variety of circuits (merely 6 RISC-V designs) and non-reproducible dataset due to use of closed-sourced industrial technology node.
TITAN [16]	Focuses on FPGA placement and routing with 23 benchmark circuits.	Limited coverage of possible designs in the VLSI domain.
Hutton et al. [14]	Discussed parameterized generation of synthetic combinational circuits.	Oversimplified circuit with limited sizes and incomplete methods for sequential circuits.
OpenABC [6]	Large labeled dataset focusing on the synthesis flow of open-source hardware IPs.	Only focuses on synthesis stage and lacks data collection during backend stages.

various early-stage prediction tasks in the design flow, such as routability [20] and IR drop [13].

In the face of increasing complexity in modern VLSI circuits and the evolving demands of advanced technology nodes, EDA tools are experiencing mounting challenges. In response to this, there is a growing urgency for more intelligent and efficient methods to enhance the design process. The objective is to optimize the performance, power, and area (PPA) metrics of VLSI circuits. In this respect, machine learning stands out as a promising solution with its data-driven approach and prediction capabilities, providing invaluable assistance in overcoming challenges in the EDA domain.

This paper presents EDALearn[1], a comprehensive benchmark suite for evaluating and comparing ML-based approaches in the context of EDA. Figure 1 presents the overview of the EDALearn benchmark flow. The benchmark covers a wide range of critical EDA tasks, such as placement, routing, power analysis, timing analysis, and IR drop prediction. The suite includes a diverse set of real-world VLSI design instances, along with a standardized evaluation framework that facilitates the fair assessment of different ML models and techniques.

By providing a systematic and unified platform for performance evaluation, the proposed EDALearn benchmark aims to foster collaboration and knowledge sharing within the EDA community, encourage the development of novel ML-based algorithms and methodologies, and ultimately contribute to the advancement of EDA tools and processes. Our contribution is listed as follows:

- We introduce a holistic benchmark suite called EDALearn which provides an end-to-end flow from synthesis to physical implementation, and present comprehensive and detailed data collection across multiple stages of our flow, thus addressing the limitations of existing benchmarks that only focus on certain stages.
- We provide an open-source reference flow for our EDALearn benchmark, fostering reproducibility and enabling research on ML transferability across different technology nodes, an area overlooked by other benchmarks [4].
- Unlike benchmarks limited in their coverage of possible designs and circuit sizes [4, 14, 16], our EDALearn benchmark accommodates a wide variety of VLSI design instances and sizes, making it more representative of modern VLSI design complexity.
- We provide an in-depth data analysis, enabling users to better understand the attributes and distribution of our data, which is critical for developing effective ML models.

The remainder of this paper is organized as follows: Section 2 summarizes previous ML for EDA benchmarks, highlighting their limitations for comprehensive ML evaluation. Section 3 describes our EDALearn benchmark suite in detail, including the used design instances, technology nodes, flow variations, extracted feature representations, and supported EDA tasks. Section 4 presents a data analysis of the EDALearn benchmark. Finally, Section 5 concludes the paper and outlines future ML for EDA research directions enabled by our EDALearn benchmark.

2 Related Works

In this seciton, we review previous ML for EDA benchmarks and discuss their limitations for comprehensive ML evaluation.

CircuitNet [4] is the first open-source dataset for machine learning applications in electronic design automation (EDA). It provides holistic support for cross-stage prediction tasks in back-end design with diverse samples. However, it only adopted 6 RISC-V designs in total, which is very limited for practical ML for EDA tasks. CircuitNet2 [15], an updated version of CircuitNet, included some GPU/AI chip designs to increase the span of the designs and covered a 14nm FinFET tech node besides the 28nm CMOS tech node in the earlier version. Both are based on closed-sourced industrial technology nodes, making the generated dataset not reproducible.

TITAN [16] is a benchmark that focuses on FPGA placement and routing. It uses Altera's Quartus II FPGA CAD software to perform HDL synthesis and creates the Titan23 benchmark set, which consists of 23 benchmark circuits. However, the coverage of these circuits is limited to a narrow range of possible designs or use cases in the general VLSI domain. [9] used the TITAN benchmark to demonstrate a reinforcement learning (RL)-based optimization for FPGA placement.

[14] discussed parameterized generation of synthetic combinational circuits, providing a convenient way to generate large numbers of circuits with a wide range of characteristics. However, the sizes of the generated circuits are limited to a few thousand gates, which is insufficient for modern VLSI design. Besides, the paper

[1]The dataset and a reference flow of our EDALearn benchmark suite is open-sourced at https://github.com/panjingyu/EDALearn.

Table 2: Synthesis variations.

Parameters		Values
Frequency (MHz)		100/500/1000
DRC	Max Fanout	10/40
	Max Transition	0.5
	Max Capacitance	5
Command		compile/compile_ultra
Options	Dynamic Power Optimization	on/off
	Leakage Power Optimization	on/off

Table 3: Physical design variations (global options).

Parameters	Values
Design Flow Effort	express/standard
Design Power Effort	none/high
Target Utilization (%)	50/60/70/80/90

Table 4: Physical design variations (placement options).

Parameters	Values
Global Timing Effort	medium/high
Global Congestion Effort	medium/high
Detail Wire Length Opt Effort	medium/high
Global Max Density	0.75/0.9
Activity Power Driven	on/off

Table 5: Physical design variations (CTS options).

Parameters	Values
Pre-CTS Opt Max Density	0.8
Pre-CTS Opt Power Effort	none/low/high
Pre-CTS Opt Reclaim Area	on/off
Pre-CTS Fix Fanout Load	on/off
Cell Density	0.5
Clock Gate Buffering Location	below
Clone Clock Gates	on
Post-CTS Opt Power Effort	low/high
Post-CTS Opt Reclaim Area	on/off
Post-CTS Fix Fanout Load	on/off

does not extend its methodology to the generation of sequential circuits, limiting its applicability in comprehensively simulating and testing the full spectrum of modern VLSI designs.

OpenABC [6] is also a large labeled dataset whose focus is the synthesis flow of open-source hardware IPs. The hardware IP designs are collected from various sources, including MIT_CEP, IWLS, OpenROAD, OpenPiton, etc.. For each hardware IP, it generates synthesis results with around 1500 random variations, or so-called synthesis recipes. [22] explored RL-based area optimization of logic synthesis on FPGAs based on the data collected from OpenABC.

While contributions such as CircuitNet, TITAN, synthetic combinational circuits, and OpenABC have set the stage for the integration of machine learning applications in EDA, they also exhibit certain shortcomings. These limitations range from restricted design diversity, reliance on closed-sourced technology nodes, limited size of generated circuits, to a narrowed focus on specific stages of the design process. Most importantly, current benchmarks lack comprehensive ML evaluation capabilities, which is instrumental in addressing practical EDA challenges. Despite these challenges, each benchmark has added value to the broader pursuit of refining ML for EDA, paving the way for future research. Moving forward, the integration of a more comprehensive, diverse, and reproducible benchmark suite would be instrumental in mitigating these limitations, advancing the scope and precision of machine learning applications in EDA. This paper introduces such a benchmark, designed to harness the strengths of the earlier works while addressing their limitations, offering an advanced framework for evaluating ML models in the EDA domain.

3 The EDALearn Benchmark

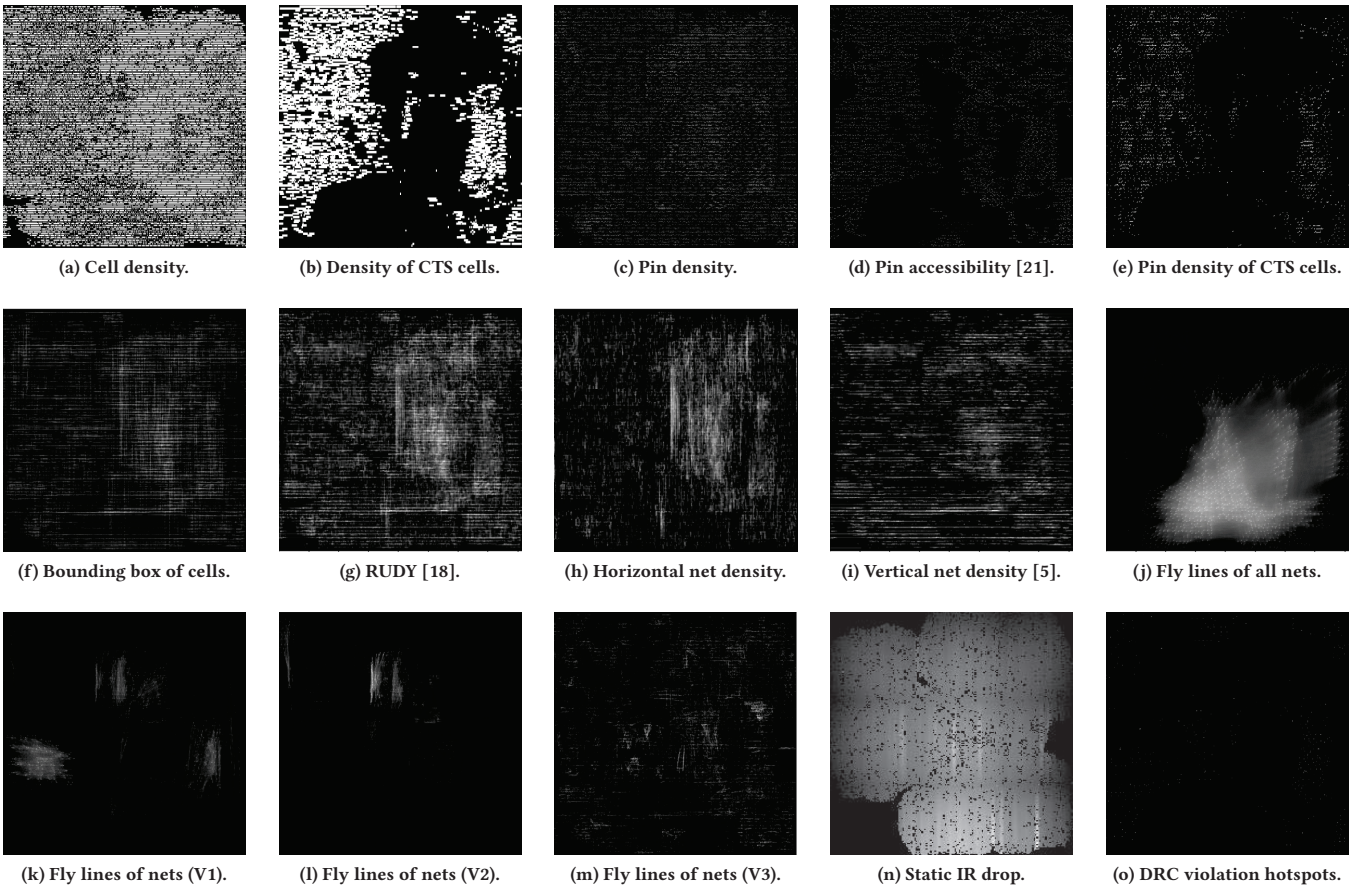
This section presents the details of our comprehensive benchmark suite, designed to evaluate and compare ML-based approaches in the context of EDA tasks. The benchmark suite is based on the FreePDK 45nm [19], the Synopsys Armenia Educational Department (SAED) 32nm [12], and ASAP 7nm [7] technology nodes, and it supports routability prediction, IR drop prediction, and cross-stage data analysis. The suite includes design instances from the IWLS benchmark and BOOM CPU designs from Chipyard, facilitating a diverse and practical evaluation of ML models and techniques.

3.1 Technology Nodes

Our benchmark suite covers 3 distinct technology nodes: FreePDK 45nm, SAED 32nm and ASAP 7nm. All of them are open-sourced tech nodes, making our data fully-reproducible. Each node brings its unique attributes and characteristics to the fore, providing a diverse and challenging landscape for the assessment of machine learning techniques within the sphere of EDA tasks.

The FreePDK 45nm and SAED 32 nodes represents an earlier phase in semiconductor manufacturing technology, characterized by larger feature sizes and lower transistor density. Consequently, circuits designed at this node tend to be more straightforward with respect to layout and thermal management, yet pose challenges in terms of power efficiency and high-speed performance.

On the other hand, the ASAP 7nm node exemplifies the cutting-edge in semiconductor technology, characterized by extremely small feature sizes and high transistor density. Designs at this node face significant challenges related to power, performance, and area trade-offs, as well as manufacturing yield and reliability. Furthermore, the high complexity and cost associated with this advanced node necessitate efficient and accurate EDA tools to ensure optimal design outcomes.



(a) Cell density. (b) Density of CTS cells. (c) Pin density. (d) Pin accessibility [21]. (e) Pin density of CTS cells.

(f) Bounding box of cells. (g) RUDY [18]. (h) Horizontal net density. (i) Vertical net density [5]. (j) Fly lines of all nets.

(k) Fly lines of nets (V1). (l) Fly lines of nets (V2). (m) Fly lines of nets (V3). (n) Static IR drop. (o) DRC violation hotspots.

Figure 2: Examples of image-like feature and label representations. "CTS cells" refers to cells directly connected to clock tree nets. Sub-figures (k)-(n) depict varying net selections, denoted as V1-V3, each based on a different bounding box area threshold: 4k μm for V1, 1k μm for V2 and 0.1kμm for V3. Two examples of post-routing stage labels are IR drop maps and DRC violation hotspots.

Through the juxtaposition of these distinct technology nodes, our benchmark suite offers a broad perspective, enabling a comprehensive and insightful evaluation of ML techniques across diverse technological contexts in EDA tasks.

3.2 Flow Variations

Table 2 lists the synthesis variations used in the benchmark suite. Table 3, Table 4, and Table 5 shows the global option variations, placement-stage variations and CTS-stage variations used for the physical design, respectively. The inclusion of these variations affords a broad spectrum of potential physical implementations. This diversification in data distribution significantly enhances the scope for development in machine learning for EDA by accommodating an array of possible real-world scenarios.

3.3 Design Instances

Our EDALearn benchmark suite comprises a varied selection of practical VLSI design instances, promoting a thorough and dependable assessment of machine learning models and methodologies. Table 6 shows the statistics of the design instances, each sourced from an openly accessible benchmark, offers a diverse range of circuit statistics. We proceed to detail the benchmarks that serve as the origin of these designs.

- **OpenCore [17]:** A set of open-source hardware designs from the OpenCore project. These designs cover a broad spectrum of complexity and are ideal for evaluating the effectiveness of ML techniques in various open-source hardware scenarios.
- **ISCAS [2, 3]:** A collection of benchmark circuits from the International Symposium on Circuits and Systems (ISCAS). These designs, which include both combinational and sequential circuits, provide a diverse set of challenges for assessing ML techniques in the context of circuit analysis and optimization.
- **Gaisler [11]:** A series of designs from Gaisler Research, known for their work in digital hardware design for commercial and aerospace applications. These designs, including the LEON2 and LEON3 processors, offer a unique opportunity

Table 6: Circuit statistics of representative designs.

Designs	Circuit Statistics			Source
	#Cells	#Nets	#Macros	
Small BOOM	686k	898k	39	Chipyard
Medium BOOM	998k	1247k	39	Chipyard
simple_spi	500	800	0	OpenCores
pci	25565	33838	0	OpenCores
ac97_ctrl	11317	14032	0	OpenCores
b14	19676	21983	0	ITC'99
b14_1	19622	21912	0	ITC'99
b15	10019	11460	0	ITC'99
b15_1	10017	11455	0	ITC'99
b17	30563	35197	0	ITC'99
b17_1	30697	35312	0	ITC'99
b18	87803	103k	0	ITC'99
b18_1	86107	102k	0	ITC'99
b19	173k	205k	0	ITC'99
b20	38511	43607	0	ITC'99
b20_1	38584	43667	0	ITC'99
b21	38363	43427	0	ITC'99
b21_1	39886	44978	0	ITC'99
b22	56931	64478	0	ITC'99
b22_1	57469	65046	0	ITC'99
des	2895	3388	0	OpenCores
DMA	29268	39715	0	Faraday
DSP	47663	64068	0	Faraday
ethernet	69546	82199	0	OpenCores
fpu	34039	37226	0	OpenCores
i2c	663	778	0	OpenCores
leon2	699k	755k	0	Gaisler
mem_ctrl	10015	12212	0	OpenCores
pci	25565	33838	0	OpenCores
RISC	81160	110k	0	Faraday
s1196	578	597	0	ISCAS
s1238	623	643	0	ISCAS
s13207	1703	1844	0	ISCAS
s1423	989	1068	0	ISCAS
s832	424	449	0	ISCAS
s838_1	363	431	0	ISCAS
s9234_1	1338	1479	0	ISCAS
sasc	786	909	0	OpenCores
spi	2890	3285	0	OpenCores
ss_pcm	433	511	0	OpenCores
systemcaes	10624	12617	0	OpenCores
systemcdes	3693	4376	0	OpenCores
tv80	7724	8378	0	OpenCores
usb_funct	18845	23913	0	OpenCores
usb_phy	681	794	0	OpenCores
vga_lcd	89751	105k	0	OpenCores
wb_conmax	64874	99445	0	OpenCores
wb_dma	81615	94562	0	OpenCores

to evaluate ML techniques in the context of both commercial and space-grade hardware design.

- **Faraday [10]:** A set of designs from Faraday Technology Corporation, a leading ASIC design service and IP provider. These designs represent a range of modern ASIC implementations, providing a robust platform for assessing ML techniques in the context of state-of-the-art ASIC design.
- **ITC'99[8]:** A collection of designs from the International Test Conference (ITC) 1999 benchmark suite. These designs, which cover a wide range of complexity and design styles, are ideal for evaluating the effectiveness of ML techniques in various test and diagnosis scenarios.
- **BOOM CPU designs from Chipyard [1]:** A set of modern, high-performance RISC-V CPU designs from the Chipyard framework. These designs represent state-of-the-art VLSI implementations and provide an excellent testbed for evaluating ML approaches in the context of advanced and complex designs.

3.4 Circuit Features

A critical aspect of applying ML techniques in the EDA domain lies in the ability to represent circuit designs in a format amenable to learning. Our benchmark suite offers a comprehensive set of circuit features, encapsulating the necessary information about the circuits into two primary forms: image-like representations and vector-based features.

- **Image-like representations:** These features capture the spatial distribution of various circuit components on the silicon area. The layout of a circuit can significantly influence the performance, power, and area (PPA) metrics of a VLSI design. For instance, closely packed gates can lead to higher power consumption and may affect the routability. Figure 2 showcases some examples of the image-like feature representations generated based on our EDALearn benchmark. Their representations capture the spatial layout and density of cells, nets, and macros in the design. These features can be easily stacked as tensors to be ingested by convolutional neural network (CNN) models, making them particularly suitable for tasks such as routability prediction, congestion estimation, and IR drop prediction.
- **Vector-based features:** These features capture circuit statistics and other design attributes as multi-dimensional vectors. They include parameters like the total number of cells, number of nets, design hierarchy depth, number and type of macros, target clock frequency, among others. These vector-based features serve as abstract representations of the designs and can be utilized by various types of ML models including decision trees, support vector machines, and fully connected neural networks. They are particularly useful for tasks that require a high-level understanding of the design, such as power and timing prediction.

By combining both spatial and statistical features, our EDALearn benchmark offers a holistic view of the circuits, enabling ML models to exploit both global and local information in their predictions. Moreover, the feature extraction process is flexible and extensible, allowing researchers to add more features that they believe may be relevant for their specific tasks. This hybrid approach paves the way for exploring complex relationships between circuit attributes

and design outcomes, fostering more insightful and accurate ML models for EDA.

3.5 EDA Tasks

Our EDALearn benchmark suite covers the following critical EDA tasks:

- **Power prediction:** Estimating the power consumption of a circuit design is a critical task in EDA. Excessive power consumption can lead to overheating, decreased reliability, and shorter battery life in mobile devices. ML models can be used to predict power consumption early in the design process, allowing designers to make necessary modifications to meet power constraints. This task specifically involves predicting both dynamic and leakage power for a given design, considering the impact of factors such as operating voltage, clock frequency, and gate capacitance.
- **Slack time prediction:** Timing analysis is a vital part of the EDA process to ensure that a circuit meets its performance targets. Slack time is the total time that you can delay a task without delaying the project. More slack time indicates a higher chance of meeting the timing requirements even with variations in gate delays. ML models can predict the slack time for each timing path, enabling designers to identify potential timing violations and make necessary adjustments early in the design process. Specifically, this task involves predicting the slack time at different stages of the design flow, such as post-synthesis, post-placement, and post-routing.
- **Routability prediction:** Assessing the feasibility of routing a design within given constraints, such as routing resources and timing requirements. ML models can be employed to predict routability early in the design process, reducing the need for time-consuming iterative refinements. Specifically, we extract labels for routing congestion and post-routing DRC violations, which are the two main metrics for routability.
- **IR drop prediction:** Estimating the voltage drop across the power distribution network, which can impact the performance and reliability of a design. ML-based approaches can provide accurate IR drop predictions in the early stages of design, allowing for more informed power optimization decisions.

In addition to the aforementioned EDA tasks, our EDALearn benchmark suite enables the benefits of cross-stage data analysis. This analysis involves comparing estimations generated by EDA tools at different stages of the design flow, such as power, timing, or routability estimations at synthesis, placement, and post-routing stages. By offering insights into the discrepancies and potential areas for improvement in the design flow, cross-stage data analysis can aid in the development of ML models tailored for EDA problems, leading to better optimization and more accurate predictions.

4 Benchmark Analysis

This section conducts an analysis of the data distribution observed in the EDALearn benchmark. Specifically, it compares the power and timing data at different stages of the implementation flow.

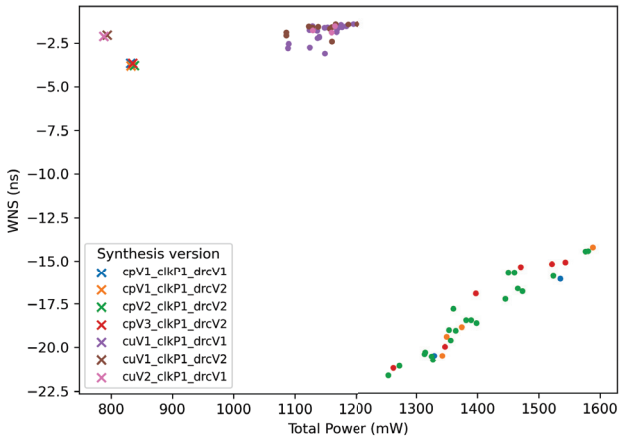


Figure 3: Distribution of WNS and total power from different synthesis/implementation results of Medium BOOM CPU design. Here we compare results based on both the `compile` and `compile_ultra` synthesis command.

Furthermore, it extends these comparisons across different technology nodes. The aim of these analyses is to shed light on potential research directions enabled by the EDALearn benchmark.

4.1 Trade-off between timing and power

Figure 3 shows the distribution of worst negative slack and total power from different synthesis/implementation results of Medium BOOM CPU design from chipyard. Here we compare the difference of distribution observed from results of the `compile` and `compile_ultra` synthesis command. It clearly shows that the results from both synthesis commands form two distinct clusters, indicating a noticeable difference in the outcomes based on the command used. The `compile` command results in one distribution, while the `compile_ultra` command results in another. Most notably, the `compile_ultra` command appears to produce superior results in terms of both worst negative slack time and total power consumption. It is found that the `compile_ultra` command results in less worst negative slack time, indicating a better timing performance. Similarly, the total power consumption is less with the `compile_ultra` command, suggesting improved energy efficiency.

4.2 Chipyard Boom designs

The discussion here centers on the distribution of Chipyard Boom CPU designs, which bear distinct attributes due to their larger size and incorporation of memory macros. As depicted in Figure 4, the worst negative slack time and total power of the Boom CPU designs, based on the FreePDK 45nm technology node, are distributed in unique clusters corresponding to each design size. An observable trend is that the total power consumed is commensurate with the size of the design, presenting a logical correlation between these two parameters. When scrutinizing the small and medium Boom CPUs, it becomes apparent that their worst negative slack times are notably similar. However, this isn't the case for the larger designs.

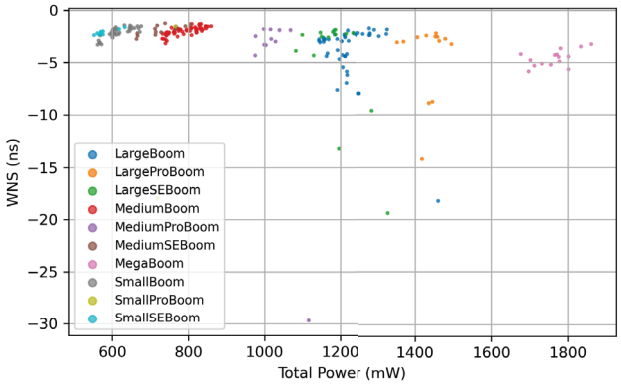


Figure 4: WNS and total power distribution of chipyard Boom designs implemented based on FreePDK45.

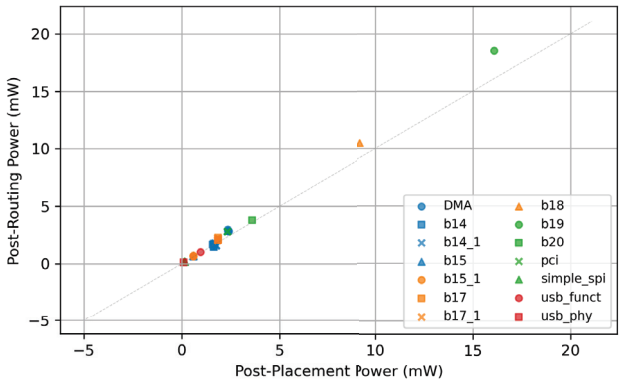


Figure 6: Total power distribution comparison between post-placement stage and post-routing stage of implementations based on the ASAP 7nm technology node.

For the large Boom CPUs and Mega Boom CPUs, the worst negative slack time is conspicuously higher, suggesting an increased complexity in managing timing for these larger designs. This analysis provides valuable insights into the intricate interplay between design size, power consumption, and timing performance.

4.3 Comparison between different stages or technology node

Figure 5 and Figure 6 respectively depict the distribution of total power consumption estimated at the post-routing stage for the FreePDK 45nm and ASAP 7nm technology nodes. A trend observed in both nodes is that the post-routing total power typically surpasses the post-placement total power, suggesting an over-optimistic estimation of total power at the post-placement stage by the golden tool. This increase can be attributed to the additional wire length incurred during the routing and optimization stage, which is commonly underestimated. Notably, the extent of this over-estimation

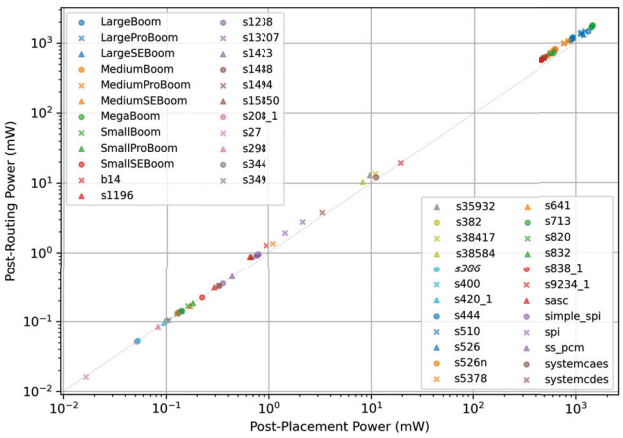


Figure 5: Total power distribution comparison between post-placement stage and post-routing stage of implementations based on the FreePDK 45nm technology node.

varies between different designs, as evidenced by Figure 5 and Figure 6. This observation illustrates the value of our EDALearn benchmark in enabling novel research directions, such as exploring more accurate power estimation models at the post-placement stage, identifying the sources of over-optimism in current tools, or developing ML models capable of more accurately predicting the impact of routing on power consumption.

In addition to offering insights into power estimations, Figure 5 and Figure 6 also facilitate a comparative view of the power distribution differences across two distinct technology nodes. This comparison not only underscores the diversity in power consumption patterns but also highlights the potential for disparities between different technology nodes. This paves the way for future research endeavors into the transferability of machine learning models, investigating whether models developed and trained on data from one technology node can perform accurately when applied to data from another node. This would further our understanding of how machine learning models can adapt to new scenarios, potentially leading to more robust and versatile ML solutions in electronic design automation.

5 Conclusion

In this work, we have introduced EDALearn, a comprehensive benchmark suite designed to facilitate the evaluation and comparison of ML-based methodologies within the realm of Electronic Design Automation. EDALearn addresses several limitations in existing benchmarks by providing a holistic, end-to-end flow that covers various critical EDA tasks including placement, routing, power analysis, timing analysis, and IR drop prediction. Our open-source reference flow fosters reproducibility and promotes research on ML transferability across different technology nodes, filling the gap left by previous benchmarks. Furthermore, EDALearn accommodates a broad variety of VLSI design instances and sizes, overcoming the narrow focus of prior benchmarks and making it a more representative solution for the modern, complex VLSI design scenario. The in-depth data analysis included in the suite enables users to gain a better understanding of data attributes and distribution, which

are essential for the development of effective ML models. In conclusion, the EDALearn benchmark suite is a significant step forward in the fusion of machine learning and EDA, providing a unified and systematic platform for performance evaluation, fostering collaboration and knowledge sharing, and encouraging innovation in ML-based algorithms and methodologies. We anticipate that this initiative will contribute substantially to the evolution and optimization of EDA tools and processes, steering the direction of future research in the field.

Acknowledgments

This work is supported by SRC 3104.001 and NSF 2106828.

References

[1] Alon Amid, David Biancolin, Abraham Gonzalez, Daniel Grubb, Sagar Karandikar, Harrison Liew, Albert Magyar, Howard Mao, Albert Ou, Nathan Pemberton, Paul Rigge, Colin Schmidt, John Wright, Jerry Zhao, Yakun Sophia Shao, Krste Asanović, and Borivoje Nikolić. 2020. Chipyard: Integrated Design, Simulation, and Implementation Framework for Custom SoCs. *IEEE Micro* 40, 4 (2020), 10–21. https://doi.org/10.1109/MM.2020.2996616

[2] Franc Brglez. 1985. A neutral netlist of 10 combinational benchmark circuits and a target translator in Fortran. In *Proc. Intl. Symp. Circuits and Systems, 1985.*

[3] Franc Brglez, David Bryan, and Krzysztof Kozminski. 1989. Combinational profiles of sequential benchmark circuits. In *1989 IEEE International Symposium on Circuits and Systems (ISCAS).* IEEE, 1929–1934.

[4] Zhuomin Chai, Yuxiang Zhao, Yibo Lin, Wei Liu, Runsheng Wang, and Ru Huang. 2022. CircuitNet: an open-source dataset for machine learning applications in electronic design automation (EDA). *arXiv preprint arXiv:2208.01040* (2022).

[5] Jingsong Chen, Jian Kuang, Guowei Zhao, Dennis J-H Huang, and Evangeline FY Young. 2020. PROS: A plug-in for routability optimization applied in the state-of-the-art commercial EDA tool using deep learning. In *Proceedings of the 39th International Conference on Computer-Aided Design.* 1–8.

[6] Animesh Basak Chowdhury, Benjamin Tan, Ramesh Karri, and Siddharth Garg. 2021. Openabc-d: A large-scale dataset for machine learning guided integrated circuit synthesis. *arXiv preprint arXiv:2110.11292* (2021).

[7] Lawrence T Clark, Vinay Vashishtha, Lucian Shifren, Aditya Gujja, Saurabh Sinha, Brian Cline, Chandarasekaran Ramamurthy, and Greg Yeric. 2016. ASAP7: A 7-nm finFET predictive process design kit. *Microelectronics Journal* 53 (2016), 105–115.

[8] Fulvio Corno, Matteo Sonza Reorda, and Giovanni Squillero. 2000. RT-level ITC'99 benchmarks and first ATPG results. *IEEE Design & Test of computers* 17, 3 (2000), 44–53.

[9] Mohamed A Elgammal, Kevin E Murray, and Vaughn Betz. 2021. RLPlace: Using reinforcement learning and smart perturbations to optimize FPGA placement. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 41, 8 (2021), 2532–2545.

[10] Faraday [n. d.]. http://www.faraday-tech.com/StructuredASIC/download.html.

[11] Gaisler [n. d.]. https://www.gaisler.com.

[12] R Goldman, K Bartleson, T Wood, K Kranen, V Melikyan, and E Babayan. 2013. 32/28nm educational design kit: Capabilities, deployment and future. In *2013 IEEE Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia).* IEEE, 284–288.

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

[14] Michael D Hutton, Jonathan Rose, Jerry P Grossman, and Derek G Corneil. 1998. Characterization and parameterized generation of synthetic combinational benchmark circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 17, 10 (1998), 985–996.

[15] Xun Jiang, Yuxiang Zhao, Yibo Lin, Runsheng Wang, Ru Huang, et al. 2024. CircuitNet 2.0: An Advanced Dataset for Promoting Machine Learning Innovations in Realistic Chip Design Environment. In *The Twelfth International Conference on Learning Representations.*

[16] Kevin E Murray, Scott Whitty, Suya Liu, Jason Luu, and Vaughn Betz. 2013. Titan: Enabling large and complex benchmarks in academic CAD. In *2013 23rd International Conference on Field programmable Logic and Applications.* IEEE, 1–8.

[17] OpenCores [n. d.]. https://www.opencores.org/.

[18] Peter Spindler and Frank M Johannes. 2007. Fast and accurate routing demand estimation for efficient routability-driven placement. In *2007 Design, Automation & Test in Europe Conference & Exhibition.* IEEE, 1–6.

[19] James E Stine, Ivan Castellanos, Michael Wood, Jeff Henson, Fred Love, W Rhett Davis, Paul D Franzon, Michael Bucher, Sunil Basavarajaiah, Julie Oh, et al. 2007. FreePDK: An open-source variation-aware design kit. In *2007 IEEE international conference on Microelectronic Systems Education (MSE'07).* IEEE, 173–174.

[20] Zhiyao Xie, Jingyu Pan, Chen-Chia Chang, Rongjian Liang, Erick Carvajal Barboza, and Yiran Chen. 2022. Deep learning for routability. In *Machine learning applications in electronic design automation.* Springer, 35–61.

[21] Tao-Chun Yu, Shao-Yun Fang, Hsien-Shih Chiu, Kai-Shun Hu, Philip Hui-Yuh Tai, Cindy Chin-Fang Shen, and Henry Sheng. 2019. Pin accessibility prediction and optimization with deep learning-based pin pattern recognition. In *Proceedings of the 56th Annual Design Automation Conference 2019.* 1–6.

[22] Guanglei Zhou and Jason H Anderson. 2023. Area-driven FPGA logic synthesis using reinforcement learning. In *Proceedings of the 28th Asia and South Pacific Design Automation Conference.* 159–165.