Machine learning in physical design

Junrui Yu[1, 5, †] **Yanru Li**[2,†]**, Xinyi Liu**[3,†]**, Zhide Yang**[4,†]

[1]Eurasia International School, Henan University, Kaifeng, Henan, China
[2]Xingjian College of Science and Liberal Arts, Guangxi University, Nanning, Guangxi, China
[3]Department of Electronic Information Engineering, Xidian University, Xi'an, Shanxi, China
[4]Guangdong Country Garden School, Foshan, China

[5]1928140100@henu.edu.cn
[†] These authors contributed equally and should be considered co-first authors.

Abstract. Machine learning is a highly effective instrument in constructing models that can expeditiously produce accurate prognostications. As the complexity of integrated circuit design continues to increase and process nodes continue to evolve, and physical design faces more challenges from modeling and optimization. To address these challenges, machine learning has been introduced into physical design. Thus, in this paper, we discuss the application of machine learning in physical design, covering topics such as Clock Tree Synthesis (CTS), Placement and Routing, IR-Drop and Static Timing Analysis (STA). The essay explores how machine learning can be used to overcome challenges in these areas, such as reducing peak current and clock skew in CTS, optimizing placement parameters and decision-making, predicting routability and reducing IR-drop effects. This paper also discusses various machine learning techniques (ML), such as reinforcement learning, convolutional neural networks and transfer learning. To conclude, we provide insights into how machine learning can be applied to improve various aspects of physical design.

Keywords: CTS, Placement, Routing, Machine Learning, IR-Drop.

1. Introduction

The integrated circuits (ICs) design process can be divided into two parts: front-end design (also known as logic design) and back-end design (also known as physical design). The front-end design mainly includes specification formulation, detailed design, Hardware description language coding, simulation verification, logic synthesis and static timing synthesis. The back-end design process includes Design for Test, layout planning, clock tree compositing, Place & Route, and physical verification of layouts. This paper focuses on back-end design (physical design) and static timing synthesis.

With groundbreaking innovations in IC design and integration, some chips have up to 1.2 trillion transistors [1]. Moore's Law predictions over the past half-century have brought the number of transistors to billions, increasing design complexity, chip integration difficulty, and design effort costs. To overcome this challenge, electronic design and automation (EDA) vendors are beginning to introduce machine learning into their products to shorten the design work time and reduce manufacturing costs [2].

Since EDA tools for ICs are a key enabler in the semiconductor industry, we present the use of machine learning algorithms for physical design and STAs in EDA.

In this paper, we review some recent important studies to solve some EDA important problems by applying ML. By learning from the existing clock tree data, machine learning can predict the optimal clock tree structure to achieve the minimum clock delay and power consumption and improve the performance of the design. In routing placement, machine learning can quickly predict the optimal layout to achieve minimum power consumption and maximum performance. In IR (voltage), machine learning can predict the optimal power network structure by learning from the historical data of the chip power network to achieve minimum power consumption and maximum stability. In terms of STA, machine learning can quickly predict the timing performance of chips by learning from historical data to achieve higher design accuracy and faster design speed. The predictions conducted by machine learning summarized in this paper will play a greater role in ICs in the semiconductor industry.

2. Clock Tree Synthesis

Clock tree synthesis is a process that distributes clock signals to clock gates in the circuit to optimize clock performance. However, CTS faces two significant challenges: variable effects and high-power consumption. Clock skew, clock jitter, clock delay, and clock load are variable effects that must be addressed in CTS due to their sensitivity to changes in the manufacturing process. Additionally, clock networks consume significant power due to their large fan-out size and switching frequency.

2.1. Machine Learning for the Variation Effect Synthesized by Clock Trees

2.1.1. Machine Learning for Reducing Peak Current. Starting all registers simultaneously causes a surge in battery demand current in zero clock skew, significantly increasing voltage drop and voltage noise, reducing transistor speed, and increasing clock jitter. To minimize peak clock current and minimize clock deviation simultaneously, the researchers used SARSA(State-Action-Reward-State-Action) Q-Learning and attenuating epsilon greedy strategies to optimize the clock arrival distribution [3].

First, the researchers randomly added or removed buffers and registers to do full exploration mode, and then used the epsilon greedy strategy to reduce random actions. Then agents gradually change behavior to maximize cumulative rewards. Experimental results show that this method can significantly reduce the peak current and IR drop and explore the optimization opportunities in CTS more comprehensively than the heuristic algorithms used in existing EDA tools.

2.1.2. Machine Learning for Reducing Time Skew. Clock skew refers to the difference in time when clock signals arrive at different components, which affects circuit performance and stability. Measures such as adding clock buffers, optimizing clock routing, adjusting clock frequency, etc., can be employed to reduce the effect of clock skew.

In this area, some researchers solve the optimization problem of highly discrete buffer and wire sizes of ASIC clock networks and the expensive problem of ordinary simulation of complex models of non-tree clock networks [4]. The authors' support vector machines-based approach replaces expensive circuit-level simulations. In the support vector machines program, the link resistor is first removed, the clock network is changed into a tree network for hierarchical optimization, and then the link resistor is added back for new optimization in the second step. Experimental results show that this method can reduce clock skew by an average of 43% with a very small increase in power consumption.

2.2. Machine Learning for Optimizing Clock Power Consumption

2.2.1. Predict Transient Clock Power with Artificial Neural Networks (ANNs). To quickly estimate the transient clock power consumption of buffers, clock gating cells (CGCs), and flip-flops before CTS, the researchers trained four different ANNs to estimate the number of CGCs and clock buffers and their respective wire loads, and to identify the gated or ungated state of each CGC, then calculate the power consumption with $P_{clk} = C \times V^2 \times F$, where C is clock load capacitance, V is clock voltage and F is clock frequency. The results show that the average error between the actual clock power consumption curve and the predicted curve is only 2%. However, the clock buffers and CGCs used in the simplified model is single and the generalization ability is not strong enough [5].

2.2.2. Predict Clock Power with Convolutional Neural Networks (CNNs). The training objectives, input elements and estimated power of the model in the literature are similar to those in the literature, but ANNs are replaced by CNNs in the algorithm [5,6]. Although CNN's training time and calculation cost are higher and overfitting problems may occur, they can handle more complex clock tree structures. The model uses CTS pre-netlist and layout diagram to plan the image, uses CNNs to estimate the parameters of the clock tree network, and further enhances CNN network by K-means clustering and Linear programming optimization.

2.3. Machine Learning for Overall Optimization of CTS

To optimize CTS from an overall perspective, the researchers propose a universal CTS framework consisting of four models: First, a location feature extractor using CNNs (ResNet-50 pre-trained) and transfer learning. Second, a regression model using clock power consumption, clock line length, and maximum clock deviation is realized to describe the optimized clock tree. Third, an adversarial learning generator to optimize and classify CTS through strategy gradient reinforcement learning. Fourth, an adversarial learning monitor using a previously trained regression model [7]. The specific process from feature extraction to the establishment of an adversarial learning model to predict CTS outcome and success is shown in Figure 1.

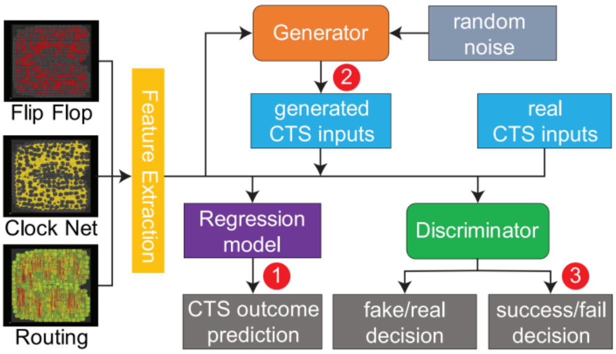

Figure 1. An overview of this general CTS framework [7].

The framework significantly improved prediction error, clock power consumption, clock line length, and maximum deviation, and attained a high score of 0.952 in the CTS success and failure classification task.

3. Placement and Routing

Placement is a crucial stage in IC design flow and also an important part in physical design, which refers to the allocation of the logic components, including circuits based on logic gates and blocks which are functionally required in the chip into the physical layout of the chip. Whether the quality of placement is excellent and closely related to the logic interconnects and the geometric position of the logic components. Since the quality of the layout scheme can only be accurately evaluated after routing, the feedback loop in the design process is long. Therefore, modern industry has some requirements for placement and routing that are improving routability while reducing congestion in routing.

Routing is also a time-consuming stage of the chip design flow. In this stage, the logic components and functional blocks assigned in the placement step will connect with each other rationally. Routing is closely linked to placement, and a good routing solution improves chip area utilization, timing performance, and routability.

3.1. Machine Learning in Placement

Traditional Placers Enhancement: Nowadays, for large-scale optimization problems, many studies traditional placers enhancements mentioned in the latest studies focus on using Central Processing Units (CPUs) for a large number of numerical calculations, and do not delve into Graphics Processing Units (GPUs). However, the exploration of GPUs will break through existing research results. The thought that the placement problem in an analytical way is pretty similar to the process of building a neural network model is the inspiration of DREAMPlace, which is based on an advanced analytical placement algorithm [8]. DREAMPlace uses the toolkit PyTorch in deep learning to achieve crucial operators of hand-optimized, which is over 30* speed up compared with tools based on CPUs. Moreover, reinforcement learning (RL) could also be used to optimize the placement parameters in EDA tools. Agnesina presents the framework based on RL that RL trains an agent and adjusts parameters autonomously [9].

Placement Decision Making: The generalization ability could be enhanced using learning-based approaches that apply RL. DeepPlace integrates RL with a cell placer based on the gradient to complete the placement of macros and standard cells. To have a better bridge between placement and routing, DeepPR based on RL is invented to achieve both placement and routing goals [10].

3.2. Machine Learning in Routing

Learning-aided Routability Prediction: The routing design in the stage of placement requirement is under consideration. However, much research focuses on overcoming the problem that it is tough to predict routing design and information correctly and quickly in placement. Optimizing the prediction of congestion count and location could reduce turnaround time in the design process [11].

Reinforcement Learning for Routing: Reinforcement learning is an effective method to handle problems in the routing stage, because it also is seen as a procedure containing a decision-making process. Generally, RL deals with violations of design rules in standard cell routing that are included in initial routing choices generated by a genetic algorithm [12].

4. The IR-Drop

IR(Voltage) drop is a phenomenon in integrated circuits that causes voltage fluctuations in power and ground networks. The allocation of power supply within a chip is attained via the Power Delivery Network (PDN) [13]. The PDN's metal layers are inherently subject to resistivity constraints. The flow of current throughout the PDN results in a voltage decline in accordance with Ohm's law, whereby the magnitude of voltage reduction is defined as V=I.R. Due to technology scaling, chip functionality is significantly impacted by even the slightest drop in supply voltage [14]. Therefore, IR drop analysis has become an indispensable procedure in chip signoff.

There are two main types for IR drop analysis: static IR drop and dynamic IR drop. Static IR drop is predominantly attributed to the voltage division of metal interconnects within the power network, consequent to their inherent resistance. A voltage decrement is observed as electrical current courses through the internal power interconnects. Consequently, the Static IR drop is substantially dependent on the configuration and specifics of the power network. Thus, the Static IR drop primarily takes into account the influence of resistance and scrutinizes its ramifications. Dynamic IR drop is a voltage deviation that arises from current fluctuations during circuit switching. This occurrence is observed at the trigger edge of the clock. The clock edge transition not only induces a significant number of transistor switches but also triggers modifications in combinational logic circuits.

4.1. Vectored IR-Drop Analysis

Vectored IR-drop analysis is used to verify power integrity in a chip's power delivery network. The extensive runtimes associated with dynamic IR-drop analysis necessitate reducing the number of test patterns to a subset of IR vectors representing worst scenarios. One of the vectored methods is a machine learning-based method called ML-Aided Vectored IR-Drop Estimation and Classification (MAVIREC) for vector-based dynamic IR drop estimation, which evaluates the power integrity of a power delivery network that is integrated on a chip [15]. Unlike the traditional slow heuristic method, MAVIREC leverages machine learning techniques for fast analysis and recommends a larger subset of test patterns that exercise worst-case scenarios. MAVIREC profiles 100K-cycle vectors in under 30 minutes and offers enhanced coverage and precision recommendations, surpassing contemporary industrial flow. MAVIREC's IR drop predictor shows a 10X speedup with under 4mV RMSE (Root Mean Square Error) [15].

4.2. Vectorless IR-Drop Analysis

Vectorless IR-drop analysis is a valuable approach utilized in analyzing power integrity of an on-chip power delivery network without using simulation patterns from value change dump (VCD) files [16]. This stands in opposition to the approach of vectored IR-drop analysis, which necessitates a considerable quantity of simulation patterns to encompass the full range of conceivable situations.

The vectors IR drop analysis approach has been regarded as the preferred method for mitigating IR issues during physical design. This is primarily due to its relatively faster and earlier estimation capabilities, which are of utmost importance, especially for large chipsets. Vector-based analysis, on the other hand, can be significantly sluggish and prone to inaccuracies, given the inherent complexities involved in obtaining precise power simulation patterns in the early stages of the design process [17].

5. Static Timing Analysis

STA is a technology used to check whether the design meets the timing rules required for the operation of the final product correctly. In the field of electronic design, timing analysis is a crucial step, which helps to ensure that the design meets high-speed signal processing requirements. By establishing a database of inputs and analyzing these inputs, STA can ensure that the design's timing meets specifications, thereby avoiding potential performance issues. During the course of time-sequence analysis, the design undergoes multiple versions of change. STA tools recalculate the timing of the design each time a design change is made, and check for potential time violations. These time violations may be caused by various factors, such as improper circuit layout, changes in component performance, and variations in load. In order to ensure the corrective measures in the design, STA tools re-run the tool several times after each design change to check for potential time violations. In summary, STA is an important tool that helps ensure electronic design projects' correctness and reliability. Using STA, designers can avoid performance bottlenecks and system failures caused by timing issues.

5.1. Machine Learning for Reducing Signal Delay

The on-chip power supply variation has become a dominant factor influencing the circuits' signal delay. In the literature, they proposed an efficient STA method [18]. By utilizing the spatial correlations of IR-drop, they can consider on-chip power supply varying in the STA. To be specific, they first identified the characterization of voltage-delay properties. Then they extracted the correlation between IR-drop and Distance by conducting IR-drop. As a result, they constructed the table, including the relationship between distance and the delay variation caused by IR-drop. Finally, they perform STA considering on-chip variations. In the experiments, the nominal path delay was calculated by Spice simulation. Then, the nominal delay of the capture path was multiplied with the corresponding derating factor (with respect to R-drop) to obtain the path delay with on-chip power supply variations. They use STA to evaluate the accuracy of the proposed method by comparing the relative delay between capture and launch paths to the criterion ratio, they specified. During STA, they found that compared with the traditional practical method, this method reduced the extra design margin (pessimism) of the traditional method and improved the accuracy of calculating the path delay ratio.

5.2. Machine Learning for Speeding Up Timing Closure

The automatic timing closure solution for relatively timed circuits is presented in the literature [19]. Using ML techniques, we can achieve the goal of timing closure for relatively timed circuits. Using iterators and features can significantly accelerate the development process and minimize the overall runtime. Using the elements generated by the model-driven hardware generator framework and will use it to reason the inference of the testing set generated by the physical system. We will also extract the real labels of delays, waveforms, and their interdependence to better understand and analyze these systems' behaviors. In literature, placement stage timing and physical information are extracted as sequence features during placement stage, while the residual path delay modeled to verify the mismatch routing path delays [20].

6. Conclusion

There are lots of research in the field of using ML methods to optimate physical design in IC flow, which is a high-demand industry. Because the rapidly rising complexity and ultra-high level of integration in physical design, efficient methods based on ML that could reduce the processing time and improve the accuracy of prediction are needed. In summary, this paper shows insight into the advanced methods of using ML algorithms for physical design in IC flow. As mentioned before, the methods that have been developed could reduce peak current and clock skew in CTS, optimize placement parameters and decision-making, predict routability and reduce IR-drop effects. Although breakthroughs have been made, we still expect more achievements on applying ML for physical design. For example, it is still hard for only using ML algorithms to fulfill the demands in real application. Therefore, the combination of modern methods and the traditional method is of great significant. Current ML methods aim to solve simplified problems and always be restricted in some small spaces. However, advanced algorithms researchers are working on are supposed to make ML methods more useful.

References

[1] A. WIlliams. 2019. LARGEST CHIP EVER HOLDS 1.2 TRILLION TRANSISTORS. Retrieved April 5, 2022

[2] L. Wang and M. Luo, "Machine Learning Applications and Opportunities in IC Design Flow," 2019 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), Hsinchu, Taiwan, 2019, pp. 1-3, doi: 10.1109/VLSI-DAT.2019.8742073.

[3] S. A. Beheshti-Shirazi, A. Vakil, S. Manoj, I. Savidis, H. Homayoun, and A. Sasan, “A Reinforced Learning Solution for Clock Skew Engineering to Reduce Peak Current and IR Drop,” in *Proceedings of the 2021 on Great Lakes Symposium on VLSI*, in GLSVLSI ’21. New York, NY, USA: Association for Computing Machinery, 2021, pp. 181–187. doi: 10.1145/3453688.3461754.

[4] R. Samanta, J. Hu and P. Li, "Discrete Buffer and Wire Sizing for Link-Based Non-Tree Clock Networks," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 18, no. 7, pp. 1025-1035, July 2010, doi: 10.1109/TVLSI.2009.2019088.

[5] Y. Kwon, J. Jung, I. Han and Y. Shin, "Transient Clock Power Estimation of Pre-CTS Netlist," 2018 IEEE International Symposium on Circuits and Systems (ISCAS), Florence, Italy, 2018, pp. 1-4, doi: 10.1109/ISCAS.2018.8351430.

[6] S. Nagaria and S. Deb, "Designing of an Optimization Technique for the Prediction of CTS Outcomes using Neural Network," 2020 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS), Chennai, India, 2020, pp. 312-315, doi: 10.1109/iSES50453.2020.00075.

[7] Y.-C. Lu, J. Lee, A. Agnesina, K. Samadi, and S. Lim, "GAN-CTS: A Generative Adversarial Framework for Clock Tree Prediction and Optimization," Nov. 2019, pp. 1–8. doi: 10.1109/ICCAD45719.2019.8942063.

[8] Y. Lin et al., "DREAMPlace: Deep Learning Toolkit-Enabled GPU Acceleration for Modern VLSI Placement," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 40, no. 4, pp. 748-761, April 2021, doi: 10.1109/TCAD.2020.3003843.

[9] A. Agnesina, K. Chang and S. K. Lim, "VLSI Placement Parameter Optimization using Deep Reinforcement Learning," 2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD), San Diego, CA, USA, 2020, pp. 1-9.

[10] R. Cheng and J. Yan, "On Joint Learning for Solving Placement and Routing in Chip Design," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. W. Vaughan, Eds., Curran Associates, Inc., 2021, pp. 16508–16519.

[11] W. -K. Cheng, Y. -Y. Guo and C. -S. Wu, "Evaluation of routability-driven macro placement with machine-learning technique," 2018 7th International Symposium on Next Generation Electronics (ISNE), Taipei, Taiwan, 2018, pp. 1-3, doi: 10.1109/ISNE.2018.8394712.

[12] Y. Lin, T. Qu, Z. Lu, Y. Su and Y. Wei, "Asynchronous Reinforcement Learning Framework and Knowledge Transfer for Net-Order Exploration in Detailed Routing," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 9, pp. 3132-3142, Sept. 2022, doi: 10.1109/TCAD.2021.3117505.

[13] (IR)IR Drop Analysis in Physical Design | IR Analysis in VLSI. (2020, May 2). IR Drop Analysis in Physical Design | IR Analysis in VLSI. https://teamvlsi.com/2020/07/ir-analysis-in-asic-design-effects-and.html

[14] T. -Y. Wu, S. Gharahi and J. A. Abraham, "An area efficient on-chip static IR drop detector/evaluator," 2009 IEEE International Symposium on Circuits and Systems, Taipei, Taiwan, 2009, pp. 2009-2012, doi: 10.1109/ISCAS.2009.5118186.

[15] V. A. Chhabria, Y. Zhang, H. Ren, B. Keller, B. Khailany and S. S. Sapatnekar, "MAVIREC: ML-Aided Vectored IR-Drop Estimation and Classification," 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, 2021, pp. 1825-1828, doi: 10.23919/DATE51398.2021.9473914.

[16] P. Huang, C. Ma, and Z. Wu, "Fast Dynamic IR-Drop Prediction Using Machine Learning in Bulk FinFET Technologies," Symmetry, vol. 13, no. 10, p. 1807, Sep. 2021, doi: 10.3390/sym13101807.

[17] Z. Xie et al., "PowerNet: Transferable Dynamic IR Drop Estimation via Maximum Convolutional Neural Network," 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC), Beijing, China, 2020, pp. 13-18, doi: 10.1109/ASP-DAC47756.2020.9045574.

[18] S. Bian, M. Shintani, M. Hiromoto, and T. Sato, "LSTA: Learning-Based Static Timing Analysis for High-Dimensional Correlated On-Chip Variations," in *Proceedings of the 54th Annual Design Automation Conference 2017*, in DAC ’17. New York, NY, USA: Association for Computing Machinery, 2017. doi: 10.1145/3061639.3062280.

[19] T. Sharma, S. Kolluru, and K. S. Stevens, "Learning Based Timing Closure on Relative Timed Design," in *VLSI-SoC: Design Trends*, A. Calimera, P.-E. Gaillardon, K. Korgaonkar, S. Kvatinsky, and R. Reis, Eds., Cham: Springer International Publishing, 2021, pp. 133–148..

[20] T. Yang, G. He and P. Cao, "Pre-Routing Path Delay Estimation Based on Transformer and Residual Framework," 2022 27th Asia and South Pacific Design Automation Conference (ASP-DAC), Taipei, Taiwan, 2022, pp. 184-189, doi: 10.1109/ASP-DAC52403.2022.9712484.