Atul S. Chaudhari, 2026, 14:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Deep Learning–Enabled Timing Optimization for Scalable and Efficient VLSI Design

Assistant Professor Atul S. Chaudhari, Aayush Mahesh Kotwal
Department of Computer Engineering,
MET's Institute of Engineering

Abstract- The continuous downscaling of semiconductor tech- nology has increased the need for reliable timing estimation and optimization in modern VLSI design. Conventional timing analysis methods depend heavily on post-routing data, which provides accurate results but demands substantial computation and slows down the overall design cycle. To overcome these limitations, recent work has moved toward incorporating deep learning models into earlier stages of the flow particularly during placement to enable faster and more predictive timing assessment. Prior studies have contributed in areas such as statistical timing analysis under process variability [1] and CNN- based frameworks for routability and power prediction, including RouteNet [4], PowerNet [2], and PROS [3]. However, these approaches operate mainly after placement or during routing, which restricts their usefulness for identifying timing issues at an early stage. The DTOC-P framework [5] marks an important shift by combining deep learning–based pre-route timing prediction with optimization capabilities of commercial EDA tools. It estimates arc delays and slew values during placement, pinpoints potential critical paths, and applies targeted refinement through iterative feedback using techniques such as buffer insertion and gate sizing. Additional features like continual learning and anomaly detection further improve its adaptability and reliability. This review paper provides an in-depth discussion of deep learning approaches applied to timing prediction and optimiza- tion, examining their methodologies, benefits, and challenges in comparison with DTOC-P. The analysis underscores how early-stage prediction and intelligent automation can shorten the timing-closure process, enhance accuracy, and reduce computa- tional effort in advanced technology nodes.

Keywords— Deep Learning, Timing Optimization, VLSI Design, Electronic Design Automation (EDA), Static Timing Analysis (STA)

I. INTRODUCTION

The rapid advancement of semiconductor technology has significantly increased the scale, performance expectations, and power constraints of modern VLSI designs. As feature sizes enter the nanometer era, achieving reliable timing be- havior during physical design has become more critical than ever. Accurate early-stage timing estimation helps ensure that performance targets are satisfied while also keeping power and area within acceptable limits. Traditional timing analysis, however, relies heavily on detailed post-routing information to achieve precision. Although effective, this dependence results in substantial computational effort and lengthens the design cycle [1].

To address variability in deep sub-micron technologies, researchers have proposed statistical timing analysis tech- niques that incorporate process and interconnect fluctuations into probabilistic models [1]. These methods offer improved accuracy under variability but struggle to scale efficiently for large industrial circuits. Meanwhile, deep learning has gained widespread adoption across fields such as computer vision and language modeling due to its ability to capture complex relationships from large datasets. Motivated by its success, EDA researchers have explored deep learning to enhance prediction accuracy, automate optimization decisions, and accelerate design turnaround time.

Several deep learning–based frameworks have been intro- duced for routability, power integrity, and congestion related prediction during physical

design. RouteNet [4] and PROS [3] employ convolutional neural networks to forecast routability and support optimization inside commercial tools, achieving improvements in congestion mitigation and timing health. PowerNet [2], on the other hand, uses a transferable CNN architecture for dynamic IR-drop prediction, helping mini- mize power integrity failures across different chip designs. Although these approaches demonstrate strong performance, they typically operate around or after the routing phase, which limits their potential for catching timing violations earlier in the design flow. The absence of robust pre-route timing prediction often forces designers to undergo multiple iterations, increasing runtime and computational cost.

The DTOC-P framework [5] was proposed to overcome this gap by combining early timing prediction with automated optimization in a practical, tool-compatible manner. DTOC- P estimates arc delay and slew during the placement stage itself, identifies potential critical paths, and collaborates with commercial place-and-route tools to perform focused opti- mizations such as buffer insertion, gate resizing, and threshold voltage adjustments. Its continual learning feature enables the model to adapt incrementally without full retraining, while anomaly detection improves reliability by preventing unsafe or abnormal optimization behavior.

This review paper provides an in-depth survey of deep learning–based timing prediction and optimization methods for VLSI design, with particular emphasis on the advancements introduced by DTOC-P. By comparing DTOC-P with related works—including PowerNet, RouteNet, PROS, and statistical timing models—this study outlines key trends, technical limitations, and future research directions in applying arti- ficial intelligence for intelligent, data-driven physical design automation.

II. PRELIMINARIES

The physical design flow of VLSI systems consists of sev- eral interconnected stages—floorplanning, placement, clock tree synthesis, routing, and ultimately timing closure. Each stage contributes to meeting performance, area, and manu- facturability goals. Among them, timing analysis is crucial because it evaluates whether signal delays across logic paths satisfy functional and frequency requirements. Key timing parameters such as arc delay, transition (slew), and setup/hold slack collectively determine whether the design can operate at its intended clock speed.

Conventional timing evaluation is carried out after detailed routing, when the exact interconnect layout becomes available. While this post-route analysis delivers the highest accuracy, it is computationally expensive and significantly extends design turnaround time. Since earlier stages like placement lack detailed interconnect and parasitic information, their timing estimates are often coarse. As a result, designers frequently cycle through multiple placement and routing iterations before timing closure is achieved—a process that becomes increas- ingly demanding at advanced nodes like 5 nm and below [1]. To reduce reliance on late-stage analysis, researchers have developed early prediction methods using statistical and ma- chine learning models. Statistical timing approaches incorpo- rate variability-aware models to estimate delay distributions and provide probabilistic timing bounds [1]. However, these techniques depend on simplified assumptions and analytical formulations that may not capture complex beh values.

Deep neural networks, especially convolutional neural net- works (CNNs), have demonstrated promising results in recog- nizing spatial and structural patterns within layout represen- tations [2], [4]. Models like RouteNet and PowerNet leverage CNN architectures for tasks such as routability and IR-drop prediction, showing that data-driven methods can outperform traditional heuristic-based estimators [2], [4]. Likewise, PROS embeds deep learning within commercial EDA flows to direct routability optimization in real time [3]. These advancements set the stage for frameworks like DTOC-P [5], which integrate early timing prediction with optimization actions to bridge the gap between initial design stages and final timing performance. DTOC-P expands on earlier ML-driven methods by unifying prediction and optimization into a single

workflow. It employs continual learning to update the model as new design data becomes available without rebuilding it from scratch. Addi- tionally, anomaly detection mechanisms remove qu essential. Since the original figure is removed, the concept can be de- scribed as follows: Machine-learning-based timing prediction frameworks—including DTOC-P—typically rely on a com- prehensive set of features extracted from both placement and routing stages. These features include geometric parameters such as arc length and pin positions, congestion information, parasitic data like resistance (Res.) and capacitance (Cap.) obtained from parasitic extraction (SPEF) files, and cell- level characteristics from library (Lib.) files. Pre-route timing analysis also provides coarse estimates of delay and slew that serve as initial guiding data.

All these inputs are fed into the learning model, which generates predictions for crucial timing metrics such as arc delay, output slew, and overall path delay. These predictions support early-stage optimization steps—such as resizing logic cells or inserting buffers—during the placement phase itself. By combining physical layout attributes and electrical char- acteristics into a unified model, such frameworks offer a strong balance between prediction accuracy and computational overhead. This enables earlier recognition of critical timing paths and leads to faster and more efficient timing closure in advanced VLSI design flows [5].

III. SYSTEM ARCHITECTURE OF DTOC-P FRAMEWORK

The DTOC-P framework integrates deep learning–based pre-route timing prediction with commercial Electronic Design Automation (EDA) tools to create a smart and automated timing optimization flow. Its architecture brings together pre- dictive modeling, continual learning, and anomaly detection in a unified system that links early timing estimation with post- route improvements. The framework operates through sev- eral coordinated stages, including the Training Design Phase, Model Training Phase, Continual Learning Mechanism, Model Inference Phase, and the Optimization Integration applied to test designs. Each stage contributes to refining prediction accuracy and guiding targeted timing optimizations during the placement process, making DTOC-P both practical and efficient for advanced VLSI design flows.

Training Design Phase

In this phase, training data is collected and organized using previously completed chip designs. Data is extracted from multiple stages of the physical design flow—such as placement, clock tree synthesis (CTS), and routing—to ensure the model is exposed to a wide range of layout scenarios. The extracted information includes spatial features like cell positions, net lengths, and congestion levels, as well as elec- trical characteristics such as resistance, capacitance, fan-out, input transition values, and arc delays [5]. These attributes are transformed into numerical feature sets that act as inputs for the learning model. This data preparation approach resembles that used in frameworks such as RouteNet and PowerNet, where detailed placement information and parasitic parameters form the foundation for accurate prediction models [2], [4].

Model Training Phase

During the model training phase, the deep learning net- work is taught to map placement-level features to accurate post-route timing outcomes. DTOC-P utilizes a ProxyNAS- driven architecture, which is well suited for learning complex nonlinear dependencies between physical layout attributes and timing behavior [5]. The model is trained to predict essential metrics such as arc delay and output slew. This approach aligns with earlier machine-learning frameworks like PowerNet and PROS, where CNN-based architectures were used to learn spa- tial patterns and temporal relationships within EDA datasets [2], [3]. Training is carried out by minimizing prediction errors using loss functions such as Mean Absolute Error (MAE) or Mean Squared Error (MSE), enabling the model to generalize effectively across new and unseen designs.

Continual Learning Mechanism

A key strength of DTOC-P is its continual learning capa- bility, which enables the model to evolve progressively as new design data is introduced [5]. Rather than performing full retraining for every

technology update or design change, the framework updates only the necessary portions of the model while preserving previously acquired knowledge. This incremental learning strategy reduces computational cost and enhances scalability. Such adaptability is especially important in advanced nanometer nodes, where frequent process shifts can quickly render fixed timing models obsolete [1]. By continuously refining its understanding over multiple design iterations, the system maintains long-term reliability and ef- fectiveness.

Model Inference Phase

After the model has been trained, it moves into the inference stage, where it generates timing predictions for new circuit designs. Using only placement-level information, DTOC-P estimates arc delay and output slew well before routing is per- formed, allowing designers to identify timing-critical regions early in the flow [5]. These early predictions help guide cor- rective optimization steps during placement itself. In addition, the framework integrates an anomaly detection module that screens out suspicious or low-confidence predictions to ensure safe and reliable optimization. This added layer of verification enhances system stability and distinguishes DTOC-P from earlier data-driven models such as RouteNet, which do not explicitly manage prediction outliers [4].

Test Design and Optimization Integration

In the final phase, DTOC-P integrates its predictions directly into commercial place and route (PR) tools for timing opti- mization. Based on the identified critical arcs, the tool applies actions such as gate resizing, buffer insertion, repowering, and threshold voltage swaps to improve timing performance [5]. The optimization cycle is followed by Static Timing Analysis (STA) to validate improvements. If violations remain, updated estimates are used to repeat the process until closure is achieved. This closed loop mechanism combines the predictive strengths of deep learning with the practical capabilities of industry standard tools, offering faster convergence and better timing closure than earlier frameworks like PROS or PowerNet [2], [3].

IV. WORKING FLOW

The DTOC-P framework operates through a structured workflow that links deep learning–based timing prediction with commercial Electronic Design Automation (EDA) tools to achieve early-stage timing optimization. The methodology progresses through several coordinated and iterative phases, allowing the system to maintain high prediction accuracy while remaining practical for real-world design environments. A concise overview of this workflow is presented below.

Data Extraction and Feature Preparation

The procedure starts by collecting design information from multiple phases of the VLSI workflow, including placement, clock tree synthesis (CTS), and routing, to encompass both physical and electrical aspects [5], [11], [13]. Key parameters such as cell positions, interconnect distances, net fan-out, re- sistance, capacitance, and signal transition times are recorded. Additionally, parasitic details obtained from SPEF files along with timing reports offer benchmark values like arc delays and output slew rates.

All collected features undergo standardization and normal- ization to create uniform training vectors. Such preprocessing is essential for ensuring model stability, consistency, and scalability across technology nodes, as demonstrated in several ML-based physical design studies [6], [7], [14].

Pre-route Timing Prediction Model Training

After preparing the features, a deep learning model inspired by the ProxyNAS architecture is trained to relate placement- level characteristics to timing results observed after routing [5]. Similar feature-to-metric learning approaches have been applied successfully in routability, congestion, and parasitic estimation frameworks such as RouteNet, PROS, and recent graph-based physical design models [3], [4], [8], [12].

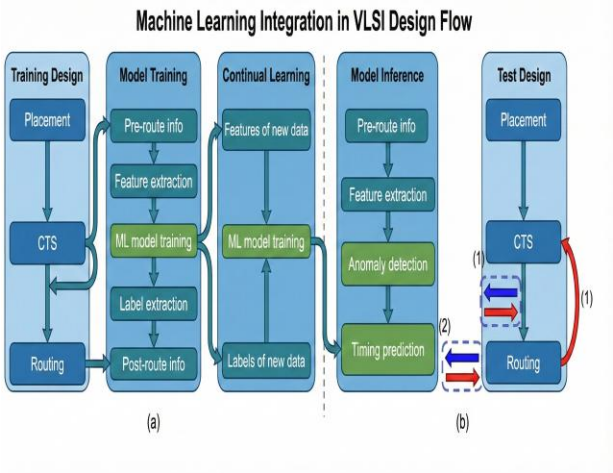
This model forecasts arc delays and output slew values, providing early insight into critical timing paths prior to the completion of detailed routing. Such early prediction follows the trend of integrating

neural network–based estimation di- rectly into the physical design flow for faster closure [9], [10], [16].

Identification of Critical Arcs

Once the model is trained, it is used to analyze new design placements to estimate timing metrics. Using the pre- dicted delays and slew values, the framework detects critical arcs—signal paths likely to breach timing constraints—and compiles them into a candidate violation set. This strategy aligns with earlier ML-based timing and congestion prediction paradigms, which aim to flag problematic paths before detailed routing [4], [8], [15].

By performing early detection, DTOC-P avoids unnecessary post-route iterations and substantially reduces overall design turnaround time, addressing a key bottleneck in large-scale industrial design flows [1], [6].



Generation and Ranking of Fix Actions

For every identified critical arc, several corrective mea- sures are proposed, including gate resizing, buffer insertion, repowering, and threshold voltage (Vt) swapping [5]. Similar optimization actions are also central to prior EDA-driven enhancement frameworks such as PROS, TimingNet, and reinforcement learning–based placement optimizers [3], [12], [17].

These interventions are prioritized based on predicted tim- ing benefit and implementation feasibility. DTOC-P integrates an anomaly detection system that screens out uncertain or potentially unsafe predictions, ensuring only reliable fixes proceed. This safety-aware mechanism provides additional robustness compared with earlier ML-based prediction models like RouteNet and PowerNet, which do not include explicit filtering mechanisms [2], [4], [7].

Optimization through Commercial EDA Tools

The highest-priority fixes are applied directly within com- mercial place-and-route (P&R) tools. Operations such as gate resizing, buffer insertion, and Vt swapping are conducted iteratively to enhance timing metrics. Similar ML-assisted optimization cycles have been explored in previous hybrid frameworks that combine data-driven prediction with conven- tional EDA algorithms [3], [11], [18].

Following each optimization iteration, Static Timing Anal- ysis (STA) is performed to verify timing improvements. If violations persist, DTOC-P updates estimates for local load, resistance, and capacitance, then continues the prediction- and-optimization loop until timing closure is achieved. This iterative refinement is consistent with modern industrial timing methodologies [14], [19].

Continual Learning for Model Updating

Unlike static machine learning models that require full retraining, DTOC-P incorporates a continual learning strategy that updates the model incrementally as new design data becomes available [5]. Continual adaptation has been shown to improve performance across technology nodes and changing layout characteristics, similar to incremental learning models studied in recent EDA research [9], [12], [20]. This feature reduces computational overhead while main- taining prediction accuracy, enabling the model to remain effective across diverse and increasingly complex design sce- narios. It also enhances scalability, addressing challenges faced by classical statistical timing models at advanced nodes [1], [6].

Final Validation and Reporting

After several optimization cycles, the framework produces a final design with improved timing performance and reduced violations. The outputs

include updated placement data, STA reports, and statistical evaluations of model accuracy. These results demonstrate DTOC-P's ability to accelerate timing clo- sure and improve resource efficiency compared to traditional post-routing optimization techniques [5], [3], [11].

Overall, the workflow highlights how integrating deep learn- ing predictions with commercial EDA tools enables faster, more intelligent, and more reliable timing optimization within the VLSI physical design process.

V. LITERATURE SURVEY

T. Huynh-Bao et al. (2017): Statistical Timing Anal- ysis for BEOL Variability at 5 nm [1]
This research proposed a statistical timing analysis method that addresses variability in both devices and interconnects, enhancing precision for advanced nodes like 5 nm. It successfully incorporates spatial correlation and BEOL variation effects, making it particularly useful for designs requiring variability awareness. Neverthe- less, the approach demands significant computational resources, limiting its suitability for large-scale circuits where efficiency and speed are crucial.

Z. Xie et al. (2020): PowerNet – Dynamic IR Drop Estimation Using Max-CNN [2]
PowerNet utilizes deep learning to perform early power integrity assessments by forecasting dynamic IR drop over chip layouts. This approach offers quicker estima- tions than conventional simulation methods while pre- serving strong accuracy. Its transfer learning capability allows the model to be reused across similar designs; however, its effectiveness decreases when applied to circuits with markedly different features.

J. Chen et al. (2020): PROS – Routability Optimiza- tion in Commercial EDA Tools [3]
PROS incorporates deep learning within commercial place and route (PR) tools to enhance routability and facilitate design closure. It showcases the effective use of AI-based predictions to steer layout optimizations. However, the approach depends strongly on consistent training datasets and struggles to generalize well to new design patterns or technology nodes.

Z. Xie et al. (2018): RouteNet – Routability Predic- tion Using CNN [4]
RouteNet employs convolutional neural networks to forecast routing congestion at an early stage of the phys- ical design process. It effectively pinpoints congested areas prior to routing, allowing for timely design modi- fications. However, its performance depends heavily on the quality of training data and tends to decline when used on layouts or technologies that differ substantially from those it was trained on.

DTOC-P (Base Paper): Deep-Learning-Driven Tim- ing Optimization Using Commercial EDA Tools with Practicality Enhancement [5]
DTOC-P introduces a two-stage framework that merges timing prediction before routing with optimization after prediction. It estimates arc delays and slew during place- ment and integrates these predictions into commercial place and route tools for iterative improvements like gate resizing and buffer insertion. The framework incorpo- rates continual learning to adapt to new data and uses anomaly detection to maintain safe and reliable opti- mization. However, it faces challenges such as large data demands, incomplete modeling of post-route parasitics, and overhead associated with tool integration.

Comparative Analysis
The reviewed literature presents a range of methodologies designed to enhance different stages of VLSI physical design using analytical approaches and machine learning techniques. The statistical timing analysis method for BEOL variability at the 5 nm node delivers high accuracy in variation-aware delay modeling but is limited by substantial computational overhead, reducing its practicality for large-scale circuits. Pow- erNet demonstrates the capability of deep learning for rapid dynamic IR-drop prediction, providing significant speedup over conventional simulations; however, its transfer learning effectiveness diminishes when applied to circuits that differ markedly from its training data.

Similarly, PROS integrates deep learning directly into com- mercial place-and-route tools to improve routability and clo- sure quality, but its performance depends heavily on the availability of consistent and representative datasets. RouteNet uses convolutional neural networks to forecast routing conges- tion early in the design process, supporting congestion-aware placement. Yet, its generalization capability declines when deployed on layouts or technology nodes outside its training distribution.

Compared to these approaches, the DTOC-P framework offers a more unified and practically deployable solution by combining early timing prediction with iterative, machine- learning–guided optimization embedded within commercial EDA tools. Unlike methods that focus solely on congestion analysis, IR-drop estimation, or statistical timing variation modeling, DTOC-P directly targets timing closure by pre- dicting pre-route delay and slew, applying targeted fixes, and leveraging continual learning for adaptability across evolv- ing design technologies. This coherence between prediction, optimization, and industrial compatibility makes DTOC-P a strong step toward robust, early-stage, and data-driven timing optimization in modern VLSI design.

advantages and key contributions

- Early Timing Prediction: DTOC-P provides precise delay and slew estimates early in the placement phase, minimizing dependence on routing analysis and enabling the prompt detection of timing problems.
- Integration with Commercial Tools: The framework integrates seamlessly with commercial place and route EDA tools, making it practical and applicable for real- world industrial use.
- Iterative Feedback Optimization: A closed-loop feed- back mechanism iteratively applies targeted fixes—such as gate resizing, buffer insertion, repowering, and thresh- old voltage swapping—until timing closure is success- fully reached.
- Continual Learning Adaptability: The predictive model incrementally updates with new data, avoiding full retraining and enhancing both efficiency and adapt- ability across various designs and technology nodes.
- Anomaly Detection for Reliability: Anomaly detection screens out unreliable or unsafe predictions, preserving design integrity and supporting robust optimization out- comes.

Faster and More Efficient Design Closure: By inte- grating early prediction, machine learning, and targeted optimization, DTOC-P delivers quicker convergence, enhanced accuracy, and better resource efficiency than conventional STA-based methods.

Disadvantages and Limitations

Although the DTOC-P framework [5] provides notable improvements in early timing prediction and optimization, it encounters some challenges and limitations that may impact its scalability and broader applicability. The main drawbacks are summarized as follows:

- High Dependence on Quality Training Data: The deep learning model's accuracy depends significantly on having access to large, high-quality datasets collected from diverse designs and technology nodes.
- Integration Overhead with Commercial Tools: While DTOC-P integrates with industrial EDA tools, the con- nection between predictive models and optimization commands can be intricate and time-intensive.
- Limited Modeling of Post-Route Effects: Because predictions occur prior to detailed routing, some post- route parasitic effects and variations might not be fully accounted for, resulting in slight accuracy reductions.
- Computational Cost of Model Training: Training deep learning models initially requires substantial computa- tional power—such as GPUs—and large volumes of design data, which may not always be readily available.
- Potential Generalization Issues: Models trained on particular datasets might struggle to perform well on different design styles or technology nodes unless they undergo fine-tuning or retraining.

Applications

The DTOC-P framework [5], along with other deep learn- ing–based approaches, finds wide applicability throughout multiple phases of VLSI physical design and optimization. Their capacity to forecast, assess, and enhance timing early in the design process positions them as essential tools for boost- ing overall design efficiency and reliability. Key application areas include:

- Early-Stage Timing Analysis: acilitates precise pre- routing estimates of delay and slew during place- ment, allowing designers to detect timing bottlenecks early—before detailed routing begins.
- Design Space Exploration: Supports rapid assessment of various design configurations, enabling improved trade-offs between timing, power, and area goals.
- Optimization Guidance for EDA Tools: Delivers data- driven insights to commercial place and route tools, enhancing decision-making for gate sizing, buffer inser- tion, and threshold voltage tuning.
- Technology Scaling Support: Enables timing opti- mization strategies to adapt across advanced technology nodes (such as 5 nm and beyond) by leveraging contin- ual learning and periodic model retraining.
- Predictive Maintenance in Design Flows: Identi- fies potential timing violations or degradation trends throughout iterations, allowing for proactive corrections and enhanced design convergence.

AI-Assisted CAD Automation:

Advances the integra- tion of artificial intelligence within Computer-Aided Design (CAD) workflows, promoting faster and smarter chip development.

VI. CONCLUSION

The ongoing miniaturization of semiconductor technology has significantly increased the complexity of VLSI design, driving the demand for smarter and more efficient timing anal- ysis and optimization methods. Although traditional routing-based timing analysis techniques provide accurate results, they are computationally intensive and typically occur late in the design process, often leading to multiple iterations and prolonged closure times. Deep learning offers a promising alternative by enabling early, data-driven predictions and opti- mizations, which reduce computational costs while maintain- ing accuracy.

The DTOC-P framework [5] marks a major advancement in integrating deep learning into commercial EDA workflows. By predicting timing metrics before routing during the placement phase, it enables earlier identification of timing violations and facilitates targeted optimizations such as gate resizing, buffer insertion, and threshold voltage adjustments. Additionally, its continual learning and anomaly detection features enhance adaptability, reliability, and efficiency across evolving designs and technology nodes.

Compared to earlier approaches like statistical timing mod- els [1], PowerNet [2], PROS [3], and RouteNet [4], DTOC-P distinguishes itself through its practical application in industry and its comprehensive integration of prediction and optimiza- tion. It effectively bridges the gap between academic research in deep learning and real-world design automation, paving the way for automated, learning-based timing closure.

Future developments could focus on improving model gen- eralization across varied layouts, enhancing the interpretabil- ity of features, and integrating multi-objective optimization that addresses power and area alongside timing. As machine learning technologies continue to advance, frameworks like DTOC-P are poised to play a crucial role in driving intelligent, adaptive, and resource-efficient VLSI design automation.

REFERENCES

1. T. Huynh-Bao, T. Chen, and Y. Cao, “Statistical timing analysis con- sidering device and interconnect variability for BEOL requirements in the 5-nm node and beyond,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 25, no. 5, pp. 1669–1680, May 2017.

2. Z. Xie, Z. Peng, M. D. Wong, and E. F. Y. Young, "PowerNet: Transferable dynamic IR drop estimation via maximum convolutional neural network," in Proceedings of the 25th Asia and South Pacific Design Automation Conference (ASP-DAC), 2020, pp. 13–18.
3. J. Chen, J. Kuang, G. Zhao, D. J. H. Huang, and E. F. Y. Young, "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 (ICCAD), 2020, pp. 1–8.
4. Z. Xie, H. Ren, P. Li, D. Z. Pan, and E. F. Y. Young, "RouteNet: Routability prediction for mixed size designs using convolutional neural network," in Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD), 2018, pp. 1–8.
5. S. K. Singh, Y. Yang, and D. Z. Pan, "DTOC-P: Deep learning driven timing optimization using commercial EDA tools with practicality enhancement," in Proceedings of the IEEE/ACM Design Automation Conference (DAC), 2023, pp. 1–6.
6. A. B. Kahng and S. Kang, "Machine learning applications in physical design," in Proceedings of the Design Automation Conference (DAC), 2017, pp. 1–6.