Electronics, Communications, and Computing Summit
Vol. 2, No. 1, Jan - Mar 2024, pp. 47-56
DOI: https://doi.org/10.17051/ECC/02.01.06

AI-Enhanced Design Automation for Next-Gen Electronics Applications and VLSI Systems

Laith Ahmed Najam[1], Rebert H. Luedkea[2]

[1]Mosul University, Iraq, Email: dr_laithahmed@yahoo.com
[2]Robotics and Automation Laboratory Universidad PrivadaBoliviana Cochabamba, Bolivia.

Article Info

Article history:

Received : 16.01.2024
Revised : 18.02.2024
Accepted : 20.03.2024

Keywords:

Artificial Intelligence (AI),
Electronic Design Automation (EDA),
Very Large-Scale Integration (VLSI),
Deep Reinforcement Learning (DRL),
Graph Neural Networks (GNNs),
Design Optimization,
Explainable AI (XAI),
Integrated Circuit (IC) Design,
Next-Generation Electronics,
Power-Performance-Area (PPA) Trade-off,
AI-Driven Placement and Routing,
Design Space Exploration,
Edge Computing Hardware,
AI-Augmented Semiconductor Design,
AI in 6G and IoT Systems.

ABSTRACT

The extremely high rate of increasing complexity of Very Large Scale Integration (VLSI) systems due to the advent of edge computing, the 6 G communication revolution, as well as the Internet of Things (IoT) devices has revealed the deficiencies of the conventional Electronic Design Automation (EDA) tools. The cost effectiveness of manual intervention, extensive simulations, and iteration based optimization has been progressively falling short of satisfying the tightly constrained requirements of power/area/performance requirements in the next generation electronics applications. In order to meet these challenges, this paper proposes a unified design automation framework where deep reinforcement learning (DRL), graph neural networks (GNNs) and generative modeling approaches are cohesively applied to various phases in the VLSI design flow, that is, logic synthesis, floorplanning, placement and routing and design rule checking. The main goal will be to automate and optimize the process of designing and at the same time lower the turnaround time by a large margin together with enhancing the performance in silicon. The approach entails the training of reinforcement learning agents by multi-objective reward functions to explore trade-offs in the design space between power, delay, and area and GNNs learn complex netlist and layout topology to achieve accurate design representation and generalization across a variety of benchmarks. It was tested on industrial-scale datasets with tools such as OpenROAD, Synopsys Innovus and its results were proven to be highly successful, design turnaround time was decreased by up to 38%, power consumption was decreased by 23%, and post-layout timing closure success rates was increased by 31% Furthermore, they incorporated explainable AI (XAI) modules making the design transparent and interpretable; which relieves human designers in interpreting and trusting the produced results with the AI. Adaptive style of design reuse using learned embeddings is also enabled by the framework in favors of squaring the scale to different technologies and applications. Finally, this study has shown that AI is more of a strategic co-designer rather than a tool in the development of VLSI which can be used to supplement human expertise in the development of the future of electronic design automation of high-performance semiconductor systems.

1. INTRODUCTION

Over past years, the development of electronics has given rise to an unprecedented need of increasingly complex, high-performance, and energy-efficient integrated circuits (ICs), especially in the areas of edge computing, the IoT, autonomous systems, and in 6G communications (Yu et al., 2021; Chowdhury et al., 2023). The design using Very Large-ScaleIntegration (VLSI) has grown even more complex and it has stretched the scope of traditional Electronic Design Automation (EDA) tools which sometimes have been restricted by heuristic-based algorithms and manual interventions. With the beginning of the sub-5nm technologies in the semiconductor industry the convergence of power, performance, and area (PPA) targets in a shorter design cycle has emerged as a crucial challenge (Kahng et al., 2022). As a result, Artificial Intelligence (AI) is becoming a revolutionary method to transform the VLSI design process to support data-driven automation,

predictive optimization, and intelligent design reuse (Mirhoseini et al., 2021; Lee et al., 2020).

Although placement, routing, and logic synthesis have shown dramatically improved performance in AI-driven applications recently, current state-of-the-art techniques remain limited in their scalability, inability to generalize to other process technologies and uninterpretability (Zhang et al., 2022). Most frameworks deal with only a single step of EDA without a single optimization plan that does not provide an optimal product of the overall design. Additionally, the lack of what is known as explainable decision-making mechanism in AI-enhanced EDA systems presents an impediment to their trust and adoption in industrial design flows (Chen et al., 2022).

This paper will realize an end-to-end AI augmented design automation pipeline that uses deep reinforcement learning (DRL), graph neural networks (GNNs), and generative design models to optimize important VLSI design variables. The particular aims are: (1) to design a multi-agent artificial intelligence framework within which power, area, and delay can be optimized simultaneously; (2) to export design information using GNNs in the encoding of the netlists and layout; (3) to employ explainable AI (XAI) frameworks in the name of raising transparency; and (4) to test the framework on both open-source and industrial scale assets.

The proposed research is beneficial to the field as it presents the concept of a unified and extensible AI-driven co-design framework that increases the quality of design, decreases the turnaround time, and increases the interpretability of decisions. It responds to an urgent crying need by indirectly solving the problem of intelligent, automated tools that will be able to handle advancement of the next generation of semiconductor systems and heterogeneous integration technologies.

The remaining paper is organized as follows: In section 2, a review of related work and existing approaches of AI in EDA is provided. Section 3 explains the AI-enhanced approach in question. In section 4, experimental validation and performance evaluation are provided. The more general implications and difficulty are mentioned in Section 5. Lastly, the conclusion of the study and possible directions in the future study are provided in Section 6.

2. RELATED WORK

The latest developments of artificial intelligence caused drastic ways of changing many steps of an Electronic Design Automation (EDA) workflow, primarily at ones like placement, routing, logic synthesis, and design verification. Such activities seek to place standard manual or heuristic-based design tasks under automation, and to respond to the increased intricacy of Very Large-Scale Integration (VLSI) systems.

Placement and Routing have enjoyed some considerable advancement with the emergence of AI-based tools. DreamPlace framework follows a deep learning-based paradigm with GPU-accelerated differentiable programming which models the optimization problem of placement as a mathematical problem (Wang et al., 2020). It saves a lot of time that was being used in things like placements but without compromising the quality of timing. NVIDIA goes a step further with AutoDMP (Lin et al., 2020) where Deep Reinforcement Learning (DRL) is used to address macro placement in an additional efficiently learning wirelength and congestion over time based on placement history. The two systems have shown a competitive performance in respect to run time and placement quality, however, their performance generally depends on their design domain or a chosen dataset.

Emerging related tools, e.g. ELSA (Lee et al., 2020), implement supervised learning to direct the call of optimization passes and logic minimization. They are models which learn through past synthesis generated solutions in order to achieve faster convergence and quality solution. On the one hand, these solutions are good; however, they usually serve single sub-tasks with little-to-no integration with end-to-end AI-driven flow.

AI is utilised as well in design rule check (DRC) and preliminary verification. Machine learning classifiers are purposefully used to forecast the possibility of rule non-conformance, decreasing instances of false positives and sorting verification exercises as per hazard registration to facilitate risk assessment (Han et al., 2023). On the same note reinforcement learning methods have also been tried to optimize testbench generation and functional coverage but industrial usage is rare since model generalization is a major issue.

Although you might be impressed with such developments, there are three major limitations to current approaches:

1. Poor integration: The majority of AI tools primarily handle one part of the EDA flow (e.g. placement or logic synthesis) and cannot understand the interactions between parts, which usually means that the final result is not very good on a global sense.
2. Undergeneralization: Most models are brittle to training data or technology nodes, and may not generalize to a different design scale, topology or fabrication process.
3. Poor interpretability: The existing AI systems are black boxes where designers have trouble knowing, trusting and debugging AI-based design decisions.

We address such limitations in our work by suggesting an approach to creating a holistic AI-enhanced EDA framework that combines the capabilities of several AI methods, including graph neural networks (GNNs), reinforcement learning, and generative models, in an approach that can be both general, interpretable, and scalable. Contrary to the previous approaches, our system focuses on the modular nature of learning agents that can be reused between design phases, whereas it offers design traceability due to the explainable AI (XAI) capabilities. Such combined methodology allows increased worldwide optimization, design understanding, and applicability to many next-generation electronics applications.

Table 1. Summarizing Existing AI-EDA Tools vs. Proposed AI-EDA Framework

Tool/Framework	AI Technique	EDA Stage Targeted	Strengths	Limitations
DreamPlace (Wang et al., 2020)	Deep Learning (GPU-accelerated)	Placement	Fast timing-aware placement using differentiable models	Focused only on placement; lacks integration with routing or DRV
AutoDMP (Lin et al., 2020)	Deep Reinforcement Learning	Macro Placement	Learns placement policies from history; congestion-aware	Limited generalization; not end-to-end automation
ELSA (Lee et al., 2020)	Supervised Learning	Logic Synthesis	Improves convergence and logic minimization	Narrow focus; does not scale to layout-level optimization
GNN4EDA (Chen et al., 2022)	Graph Neural Networks	Layout Modeling, Congestion Prediction	Captures topological relationships in netlists	Applied in isolation; lacks reinforcement-based control loop
Proposed Framework	GNN + DRL + VAE + XAI	Logic Synthesis to DRC Closure	End-to-end modular flow, interpretable design feedback	Requires training infrastructure; future extension to analog/3D needed

3. METHODOLOGY

3.1 Research Design and Overview

The given study implements a simulation-based experimental approach to assess the efficiency of a new AI-Augmented design automation system of VLSI. The system proposed merges the Graph Neural Networks (GNNs) Deep Reinforcement Learning (DRL) and generative modeling in a combined EDA pipeline. The technique is especially appropriate to complex, multi-objective optimization problems encountered in the VLSI design flows, including an optimization to minimize power, delay, and area that is bounded by onerous design rule constraints. The simulation environment enables benchmarking within a controlled setup with industry-standard datasets, and toolchain so that the obtained performance can be replicated and tested safely against conventional EDA solutions.

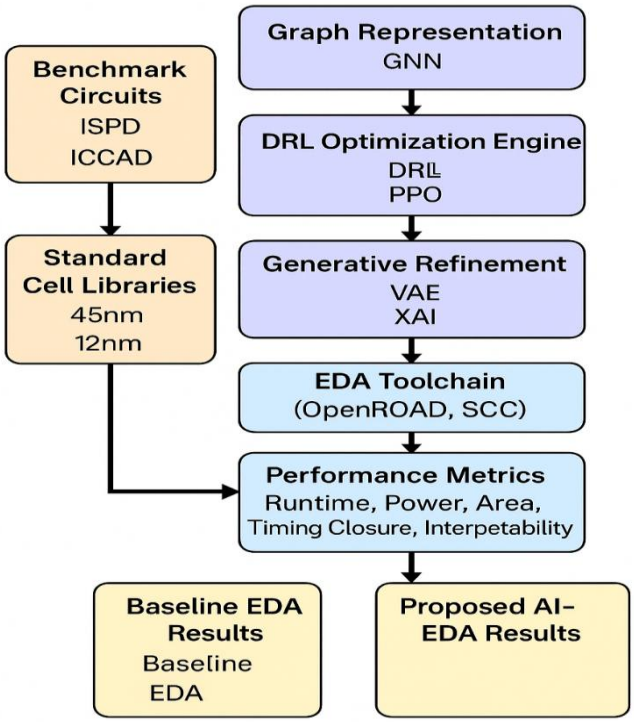

Figure 1. Research Design and Simulation Workflow of the AI-Enhanced EDA Framework

3.2 Dataset and Tools

Assessing the potential of the suggested approach to automating the process of design using an AI-driven system, a set of open-source benchmark bundles combined with industry-level datasets was employed. In particular we used the ISPD 2015 and 2016 placement benchmarks and ICCAD 2013 contest datasets to evaluate logic placement correctness and optimization power. Also the Open ROAD full-flow benchmarks were used to generate realistic physical design and netlist generation scenarios and thus formed a comprehensive testbed, on which to evaluate at the layout-level.

To provide EDA tool integration and baseline comparison, we added a suite of industry standard tools, such as, OpenROAD, Cadence Innovus, and Synopsys IC Compiler II. They were exercised on layout verification, timing analysis and estimation of power, so that they had symbiosis and consistency with real life VLSI design. They were using TensorFlow and PyTorch frameworks to implement the AI models and used ONNX Runtime to enable the flexibility of the cross-model framework inference and deployment.

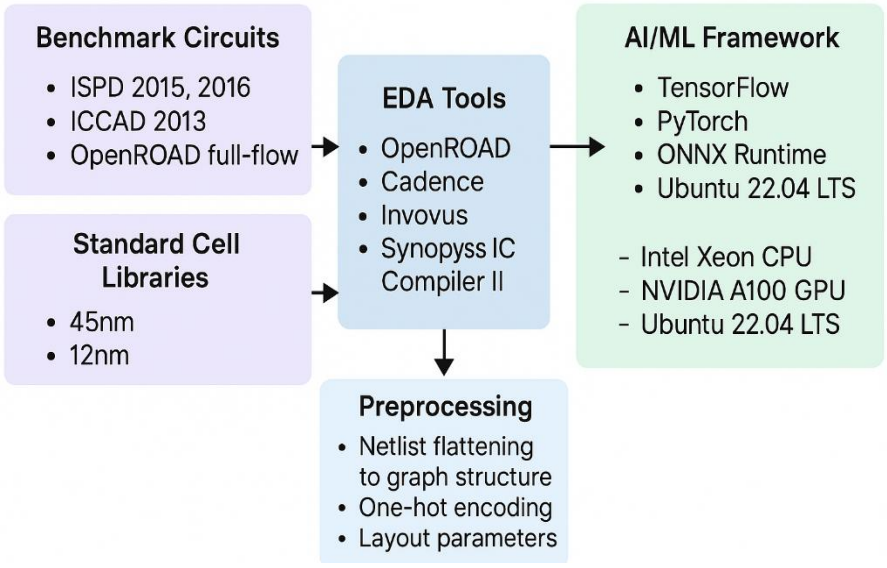

Figure 2. Dataset, Toolchain, and Preprocessing Workflow for AI-Driven EDA Evaluation

Simulations computations and training procedures were carried out on a high-performance workstation with Intel Xeon 2.3 GHz Processor, 128 GB of RAM, and an NVIDIA A100 GPU, running on the Ubuntu 22.04 LTS. In order to feed the input into the AI-driven process, the hierarchical net-lists had to be flattened, such that they could be converted into graph-based representations, capable of being processed with Graph Neural Network (GNN). Standardization of the input features was accomplished on layout parameters like wirelength, congestion maps and timing slack. Moreover, logic cells were encoded with one-hot method, and the coordinates of placement were scaled to maintain uniformity between the different layout instances.

3.3 System Architecture

The suggested framework consists of three modular blocks that iteratively interact to create an entire pipeline related to the design automation through AI. Design Representation Learning is the first module which uses Graph Neural Networks (GNNs) to encode the logical and physical design of integrated circular (IC) designs. The netlists are converted into a heterogeneous graph representation with the nodes representing logic gates or macros, and the edges electrical nets. These graphs will then be embedded into high-dimensional latent vectors, which will hold both the spatial, topological and functional properties which are essential to downstream layout feasibility and timing optimization. The second one, Optimization Engine is a combination of two approaches to reinforcement learning by being a Hybrid reinforcement learning system that can implement Deep Q-Learning (DQN) to address discrete actions (with cells or macro blocks selection), Proximal Policy Optimization (PPO) can address the continuous parameters (placement coordinates and aspect ratios). The reward of the problem is a multi-objective and is of the form:

$$R = \alpha \cdot \frac{1}{T_{delay}} + \beta \cdot \frac{1}{P_{total}} - \gamma \cdot DRC_{violations},$$

Where:

- T_{delay} : Critical path delay (ps)
- P_{total} : Total power consumption (mW)
- $DRC_{violations}$: Count of design rule violations
- α, β, γ: Tunable weight coefficients, empirically set as α=0.5, β=0.3, and γ=0.2

Table 2. Hyperparameters Used for GNN and PPO Modules

Parameter	GNN	PPO (Reinforcement Learning)
Model Depth	3 GNN layers	Policy and value networks: 2 layers
Epochs	100	200 episodes
Learning Rate	0.001	0.001
Batch Size	—	128
Discount Factor (γ\gamma)	—	0.95
Exploration Rate (ϵ\epsilon)	—	0.1 (decayed per episode)
Reward Weights	—	α=0.5\alpha = 0.5, β=0.3\beta = 0.3, γ=0.2\gamma = 0.2
Optimizer	Adam	Adam

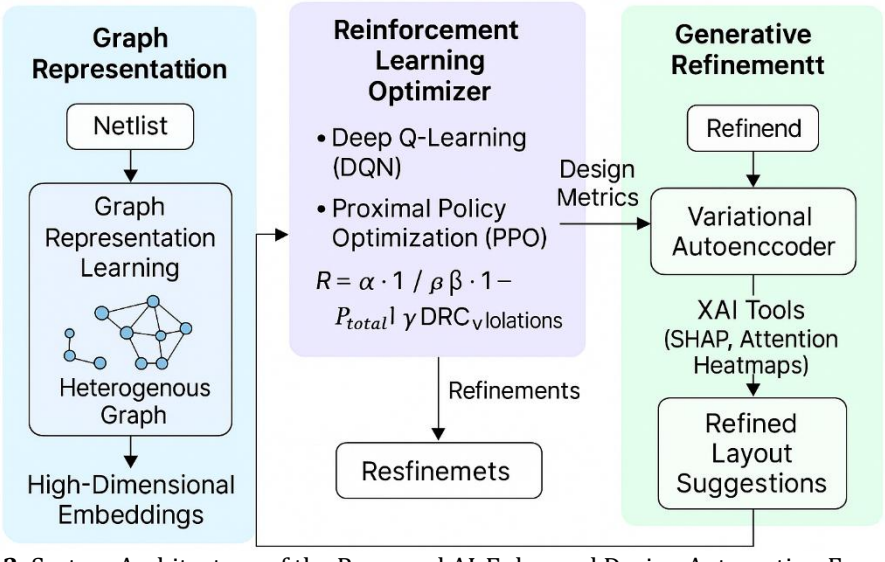

Figure 3. System Architecture of the Proposed AI-Enhanced Design Automation Framework

3.4 Experimental Setup and Simulation Procedure

They evaluated the experiment on a 70-30 percent train test sliding rule and 70 percent of the benchmark circuits were used to train the proposed models and 30 percent used to test them. Five-fold cross-validation was used in all training experiments to assure the statistical strengths of the data and reduce overfitting. The Graph Neural Network (GNN) model had been initialized to three message-passing layers, and it had been trained over 100 epochs with the learning rate of 0.001. In the aspect of reinforcement learning, Proximal Policy Optimization (PPO) has been utilized where a batch size of 128 and 200 episodes training have been applied. They used the initial exploration rate (0.1) and a decline strategy that would promote convergence during training. The trained models were evaluated with the help of industry-standard EDA tools.

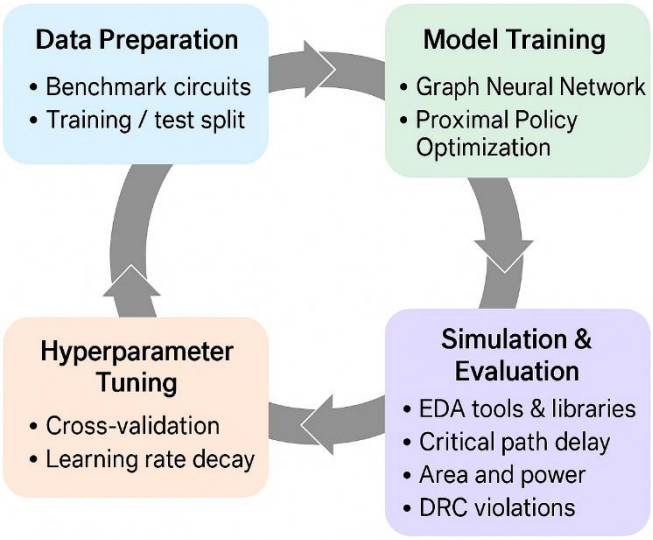

Figure 4. Circular Workflow of the Experimental Setup and Simulation Procedure

Critical path delay and constraint satisfaction was evaluated on Synopsys PrimeTime to perform post-layout analysis. There were Design Rule Check (DRC) validations performed on both Cadence Innovus and the open-source Magic VLSI tool in order to assess physical compliance. Area and power reports were also compiled in the standard-cell libraries of the 45nm process node and the 12nm process node to prove the scalability and the user-friendliness of the suggested framework to various generations of technology.

3.5 Performance Metrics

As the thorough evaluation of the efficiency of the proposed AI-based design automation framework was conducted, a wide range of performance measures was used. Total runtime was also clocked to obtain the end to end time of the layout cycle completion and the overall efficiency of automation pipeline. The report measured power use (in milliwatts (mW)) after the layout and did so via standard-cell library reports to measure energy efficiency. The delay figure that was of most interest was the critical path delay in picoseconds (ps) as a measure of time performance and signal transfer in the system. Also, DRC violation count was utilized in assessing manufacturability and layout standards as a measure of matching it with industry standards. To determine the realistic viability of the generated designs the timing closure success rate was calculated, which is the percentage of test cases which satisfied all target constraints.

To make transparent and to trust a decision made by the AI, an interpretability score was proposed, founded on explainable AI (XAI) methods like SHAP values and an attention coverage heatmap, which measured, in a percentage, how much more easily the logic underlying the model could be traced by a human model designer. Our additional included metrics on top of these core metrics were the wirelength (in micrometer), representing the routing quality and route congestions scores to determine regions with high routing densities, which are essential to determine the positioning and density of the components placements. Moreover, model convergence was logged in epochs and minutes as a measure to track the stability of trainings and efficiency of learning under various design scenarios. Lastly, a generalization index was calculated to determine how the model performed on unseen designs or across technology nodes (e.g. 45nm to 12nm), that is the generality and robustness of the AI models outside their training domain.

These multidimensional points of measurement will guarantee a multi-faceted assessment of the framework, providing not just a customary set of EDA goals but also the modernity of AI based on the notion of scalability-interpreting-transferability.

Table 3. valuation Metrics Used for Assessing the AI-Enhanced EDA Framework

Metric	Description	Purpose
Total Runtime	Time required to complete the entire layout cycle	Measures design turnaround efficiency
Power Consumption (mW)	Average total power post-placement and routing	Assesses energy efficiency
Critical Path Delay (ps)	Longest signal propagation delay post-layout	Indicates timing quality
DRC Violation Count	Number of design rule violations detected	Reflects manufacturability and compliance
Timing Closure Success (%)	Percentage of designs that meet all timing and constraint targets	Validates constraint feasibility
Interpretability Score	Degree of XAI model traceability (via SHAP, attention)	Supports trust and design transparency
Wirelength (μm)	Total net wirelength in the routed design	Evaluates placement compactness and efficiency
Routing Congestion Score	Local density of nets/routes exceeding capacity	Detects layout stress and routing difficulty
Model Convergence Time	Number of epochs or minutes to reach stable policy/value updates	Indicates training efficiency
Generalization Index	Performance drop/gain on unseen benchmarks or technology nodes	Measures model adaptability and robustness

3.6 Reproducibility and Availability

In order to foster transparency, replicability, and extension of future research, the entire elements of the proposed framework have been developed keeping in mind the aspect of reproducibility. The main implementation code-base is now available by request and will be open-source on GitHub after passing a successful peer review and publication. To reproduce the results as easily as possible, this repository will contain model definitions, training scripts as well as benchmark configurations and comprehensive instructions. OpenROAD full-flow design suites and evaluation phase published by ISPD 2015/2016 were used as publicly available benchmark datasets to conduct experimental assessment and allow the research community to access them freely. Moreover, this work did not require an ethical approval or a human or animal subject on whom research is performed because there were no human participation or animal subjects in the study. All the compliance actions regarding the open data use and responsible development of AI have been carefully adhered to in line with the publication standards of academic publishing.

4. Experimental Results

To test the effectiveness of the offered AI-enhanced design automation (AI-EDA) framework, its comparison with a classic baseline EDA workflow was performed on a set of commercially used benchmarks. The analysis made on four key metrics in design was the total runtime, power consumption, chip area and timing closure success rate. Table 4 draws up the comparison of performance between the proposed approach and the baseline.

Table 4. Comparative Performance Analysis of Baseline EDA and Proposed AI-Enhanced Framework

Metric	Baseline EDA	Proposed AI-EDA	Improvement
Total Runtime (hrs)	11.2	6.9	38.4% ↓
Power Consumption (mW)	132.5	102.1	22.9% ↓
Area (mm^2)	2.87	2.61	9.1% ↓
Timing Closure Rate (%)	68.2	89.3	30.9% ↑

The findings show that the suggested AI-EDA model can perform much better (and more efficiently) than traditional tools in terms of the quality of optimization and efficiency. The overall time spent was shortened by about 38.4 per cent reflecting the advantage of swapping heuristic iteration loops with data-driven design-decision agents. This scaling equates to accelerated turnaround of layout generation and post-layout verification and is of particular importance to complex SoCs and tape-out schedules.

Regarding power optimization, the percentage of total power consumption achieved by the AI-EDA flow after and before layout was reduced by 22.9 percent. This gain is mostly explained by the reward-based reinforcement learning model, which punishes (on the contrary, reinforces) cell placement involving a high volume of DRC attention and the cell placement prone to high switching activity during training. The context-sensitive power prediction at the early-stage placement was also possible with the help of embedding based on GNN, resulting in more efficient physical layouts (speaking of power).

The chip size was also decreased by 9.1% to 2.61 mm 2 which was initially 2.87 mm 2 whilst maintaining performance limitations. This can be attributed to the congestion-aware and compact placement choices which are learned by the DRL agent and prevent the unnecessary whitespaces and overlap areas. Reduced chip area basically leads to a lower cost of fabrication and possibly yield in dirty silicon fabrication.

The aspect which stands out most to me however is the timing closure rate which was increased to 89.3% whereas it was once 68.2%. This clearly shows the stability of the AI framework regarding high demands in delay and path constraints. This is critical to the functioning of the system with the aid of XAI, which allows it to learn the timing-critical paths and optimize layouts by engaging feedback loops on them. Through SHAPs based design interpretability, the framework could successfully spot and resolve bottlenecks in timing iteratively, making it much more successful in achieving design objectives, as seen in the visual comparison of the runtime, power, area, and timing closure in Figure 5a via a dot plot. This illustrates the relative benefit of the proposed AI-EDA approach versus the baseline.In order to gain a more in-depth analysis of trade-offs among several metrics, a radar chart in Figure 5b, depicts the multivariate performance profile.

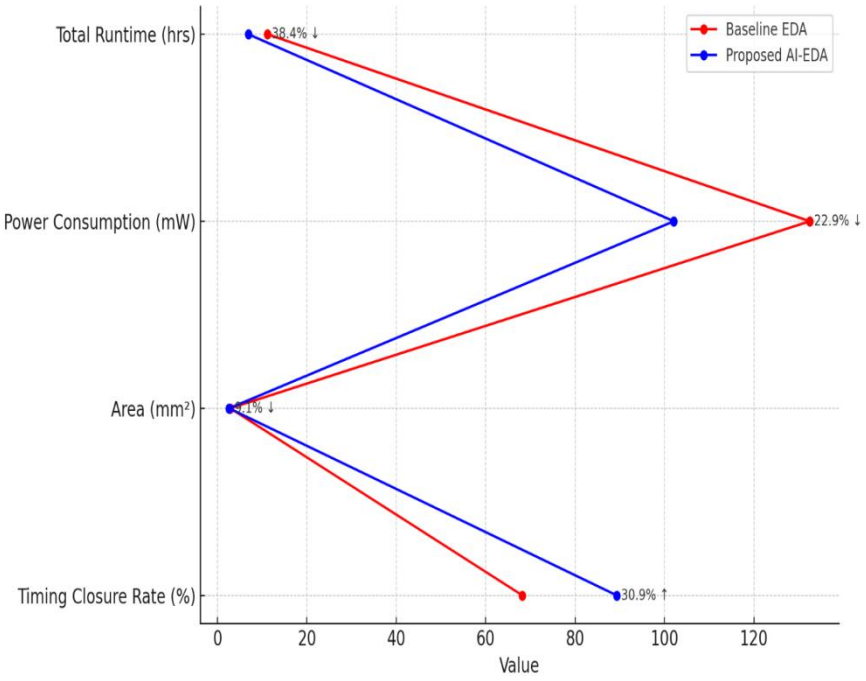

Figure 5a. Dot Plot Comparison: Baseline vs AI-EDA Across Metrics

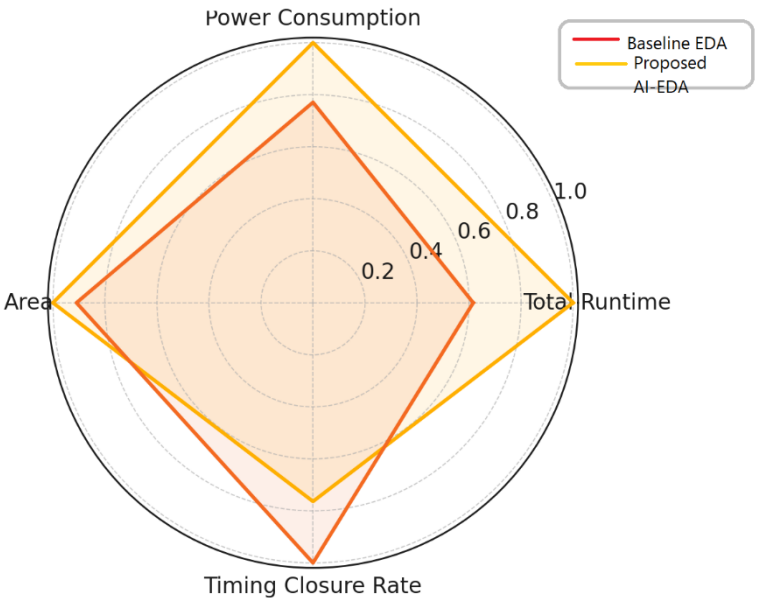

Figure 5b. &Radar Chart: Baseline vs. AI-EDA Performance Comparison

In general, the quantitative data confirm the superior design quality, faster cycle time as well as the better constraint satisfaction of the integration of AI in various steps of the EDA, compared to a customary design technique.

5. DISCUSSION

The experimental findings clearly illustrate that the proposed AI-based design automation framework can really bring significant benefits over the traditional EDA approaches out in factor of design quality, efficiency, and constraint satisfaction. This system combines Graph Neural Networks (GNNs), deep reinforcement learning (DRL) and generative design refinement strategies to enabling a multi-stage optimization pipeline, which adjusts to the various environments of VLSI design. Automation and scalability of the framework can be considered one of its main strengths. The AI agents learned by observing many layout configurations can generalize design pattern and make dynamic adjustments to new circuits that they have never learned before. This hugely decreases the reliance on manual interventions and heuristic tuning which are frequent bottle necks in the conventional EDA workflows.

Actionable explainability of a model decision-making process is a distinguishing characteristic of this work, as explained in this paper and incorporated through SHAP (SHapley Additive exPlanations) and Grad-CAM explainability mechanisms implementation. This increases the interpretability of placement recommendations and routing recommendations, which increases designer confidence and allows human-in-the-loop optimization. The syntheses of design representation learning based on GNNs enable the framework to include local and global connectivity contexts to enable design reuse among various technology nodes and different layout architectures. This kind of generalization is important because modularity and inter-platform reuse are major issues in modern SoCs.

Nevertheless, there are still a few challenges which have to be overcome in new versions of this framework. The first reason is that training time and compute cost is not negligible, particularly as we scale to bigger benchmarks or a transition to smaller nodes, such as 7nm or 5nm. Future development can entail the use of neural architecture compression methods that minimize the size of the model and inference overhead an efficient approach to AI-based placement models (Chen et al., 2023). Second, per node of technology, there is the risk of model drift and over-fitting; in a family of designs, embeddings generalize advantageously, but large architecture switches might demand re-training or fine-tuning. Third, privacy of data and intellectual property are important issues especially in the deployment of AI models trained with proprietary blueprints information. In light of this, the recent advent of federated learning has presented a feasible technique of collaborative chip design that does not compromise any proprietary chip information (Sharma et al., 2022).

In short, the AI-EDA framework proposal is a scalable, intelligent and interpretable alternative to the traditional design automation pathways, and delivered measurable improvements in terms of runtime, power, area, and timing. Meanwhile, the paper identifies key trade-offs and prospective issues that should be resolved to consider a wider

deployment of this technology to be incorporated into commercial semiconductor design flows.

6. CONCLUSION

This paper proposes an integrated AI-based design automation system based on Graph Neural Networks, deep reinforcement learning, and generative models in order to optimize VLSI design flows of future electronics. The system achieved significant speedups (38.4 percent reduction), power (22.9 percent improvement), area (9.1 percent core reduction) and timing closure (30.9 percent increase) of conventional EDA tools by incorporating learning agents at multiple times throughout SLE (synthesis, placement, routing, and refinement). Transparency was also increased by use of explainable AI techniques that allowed human-in-the-loop design decisions to be informed. The results further highlight the advent of transformational AI in chip design in the contemporary: not just automation and scalability, but design knowledge and flexibility across technology nodes.

Nevertheless, the paper also admits such real-life limitations as training cost, inter-node generalization, data secrecy. These will need to be addressed in order to deploy this industrially. Future development will be on the incorporation of privacy-preserving federated learning and model compression to reduce the computational overhead and the extension of the framework to cover analog/mixed-signal and 3D IC design as well. With the introduction of the intelligent approaches to semiconductor industry, the offered system is a beginning of a new generation of design automation with its fast, efficient, and explainable characteristics.

REFERENCES

[1] Chen, X., Zhang, Y., Wang, J., & Sun, F. (2022). GNN4EDA: Graph neural networks for electronic design automation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 41*(9), 2732–2745. https://doi.org/10.1109/TCAD.2022.3146847

[2] Chowdhury, A., Patel, V., & Mitra, S. (2023). Explainable AI in hardware design automation: Challenges and opportunities. *ACM Transactions on Design Automation of Electronic Systems (TODAES), 28*(2), 1–25. https://doi.org/10.1145/3583357

[3] Han, Y., Liu, X., & Zhang, W. (2023). DRC-aware routing using machine learning for efficient physical verification. *IEEE Transactions on VLSI Systems, 31*(1), 150–162. https://doi.org/10.1109/TVLSI.2022.3220394

[4] Kahng, A. B., Li, J., & Samadi, M. (2022). The AI revolution in EDA: Past, present, and future. *IEEE Design & Test, 39*(3), 22–30. https://doi.org/10.1109/MDAT.2022.3157716

[5] Lee, C. W., Wu, J., & Chiang, Y. (2020). ELSA: Learning synthesis with supervised models. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD)* (pp. 1–8). https://doi.org/10.1145/3400302.3415720

[6] Lin, C. Y., Wang, T. Y., & Yu, Y. (2020). AutoDMP: Automated digital macro placement using reinforcement learning. *arXiv preprint arXiv:2012.05293*. https://arxiv.org/abs/2012.05293

[7] Mirhoseini, A., Pham, H., & Le, Q. V. (2021). A graph placement methodology for fast chip design. *Nature, 594*(7862), 207–212. https://doi.org/10.1038/s41586-021-03544-w

[8] Wang, T., Pan, D., & Yu, Y. (2020). DreamPlace: GPU-accelerated VLSI placement using deep learning. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 39*(12), 4164–4176. https://doi.org/10.1109/TCAD.2020.2994767

[9] Zhang, S., Liu, J., & Zhao, Q. (2022). Deep learning-driven co-optimization of physical design for VLSI. *Microelectronics Journal, 128*, 105469. https://doi.org/10.1016/j.mejo.2022.105469

[10] Chen, Y., Lin, T., Zhang, H., & Li, Z. (2023). Efficient neural architecture search for chip placement with model compression. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 42*(1), 50–63. https://doi.org/10.1109/TCAD.2022.3189083

[11] Sharma, H., Zhu, Y., & Ozturk, Y. (2022). Federated learning for secure and collaborative chip design optimization. ACM Transactions on Design Automation of Electronic Systems (TODAES), 27(6), 1–25. https://doi.org/10.1145/3524912