# LI WEI (NEWAY) 李巍

weili3@andrew.cmu.com

Ph.D. Candidate ⋄ Department of Electrical and Computer Engineering ⋄ CMU

## RESEARCH INTERESTS

- Multi-modal ML and LLMs. [DAC'20, ASPDAC'21, TCAD'22, MLCAD'23, LAD'25]
- Scalable methods for physical design and testing. [ITC'22, DAC'24]
- Optimization methods for VLSI design and testing. [TCAD'21, DAC'23]

## EDUCATION

**Carnegie Mellon University, PA, USA**                                    Aug. 2021 – Present
Doctor of Philosophy, Department of Electrical and Computer Engineering
Supervisor: Professor Shawn Blanton & Professor Jose Moura

**The Chinese University of Hong Kong, Hong Kong**
Master of Philosophy, Department of Computer Science and Engineering          Aug. 2019 – July. 2021
Bachelor of Science, ELITE Stream, Department of Computer Science and Engineering   Aug. 2014 – Aug. 2018
Supervisor: Professor Bei Yu

## RESEARCH EXPERIENCE - IN UNIVERSITY

**PhD candidate, Carnegie Mellon University, United States**                    Sep. 2021 – Now
*Multi-modal agentic LLMs for EDA*

- Introduces a new paradigm where EDA agents move beyond rigid, predefined APIs and interact with a unified, multi-modal representation of the design environment.
- Outlines two challenging case studies—automated RTL debugging and logic diagnosis—to demonstrate the framework's effectiveness.
- Investigate the new problem-solving strategies that emerge from this increased agent autonomy.
- Study how this more complex interaction model impacts inference-time scaling laws, particularly the efficiency of repeated sampling.

*Differentiable and Scalable ATPG and Diagnosis*

- A transformative testing framework developed to resolve Automatic Test Pattern Generation (ATPG) and diagnosis.
- Shows a 30% fault coverage improvement over the industrial tool on the industrial circuits.
- Detects over 50% faults which are missed by the industrial tool on the industrial circuits.

*Graph modality in LLMs [ICLAD'25, Best Paper Honorable Mention Award]*

- Explore the graph modality integration into LLMs for VLSI
- An fully automated data collection pipeline
- Collect more than 10 billion training tokens

*Global Floorplanning using Semidefinite Programming [DAC'23]*

- A SDP-based method for finding the best locations of modules in a chip
- The average wirelength is reduced by at least from 3.02% to 20.01%
- The industrial case study shows **500%** quality improvement compared to the industrial tool.

*Pseudo-Exhaustive Physically-Aware Region Testing [ITC'22]*

- Comprehensively analyze both the physical layout and the logic netlist to identify single- or multi-output sub-circuits.
- Implemented a novel tensor-based representation of layout polygon coordinates that enables a neighborhood search strategy that reduces computational complexity from $O(n^2)$ to $O(dn)$.
- Implemented a GPU-based algorithm the physical sub-circuit extraction containing billions of sub-circuits.

*GNN study in logic locking [MLCAD'23]*

- Modeled their ability to identify circuit changes that stem from a logic lock as the ability to decide the isomorphism between logic netlists.
- Showed that GNNs are always upper bounded by heterogeneous Weisfeiler Lehman test in deciding the netlist isomorphism, and gave the conditions when GNNs reach the bound.

**MPhil Student, The Chinese University of Hong Kong, Hong Kong**          Aug. 2019 – June. 2019
*Routing Tree Construction [ASP-DAC'21, Best Paper Award]*
- Formalized special properties of the point cloud for the routing tree construction with theoretical proof.
- Proposed an adaptive flow, which used the cloud embedding obtained by a specifically-designed model based on special properties, to select the best approach and predict the best parameter.
- Outperformed previous methods by a large margin yet being extensible and flexible.

*Adaptive Layout Decomposition [DAC'20, TCAD'21]*
- Proposed an adaptive workflow for efficient decomposer selection and graph matching using graph embeddings.
- Designed a graph library construction algorithm based on graph embeddings for small graphs excluding isomorphic ones.
- Reduced the runtime by 87.7% while still preserving the optimality compared with the ILP-based decomposer.

**Research Assistant, The Chinese University of Hong Kong, Hong Kong**          Feb. 2019 – July. 2019
*Open-source Layout Decomposition Framework [TCAD'21]*
- Presented an open-source layout decomposition framework, with efficient implementations of various state-of-the-art simplification and decomposition algorithms.
- Discovered a set of issues of previous algorithms and proposed corresponding solutions.

*Acceleration and Compression of DNNs [ICTAI'19, Best Student Paper Award]*
- Proposed a unified framework to compress CNNs by combining both lowrankness and sparsity.
- Compressed the model with up to $4.9\times$ reduction of parameters at a cost of little loss.

**Research Assistant, Southern University of Science and Technology, China**          June. 2018 – Jan. 2019
*Testing of Auto-driving Systems [ICSE'20]*
- Introduced a joint optimization method to systematically generate adversarial perturbations to mislead steering of an autonomous driving system physically.
- Demonstrated the possibility of continuous physical-world tests for auto-driving scenarios as the first study.

*Fault Localization [ISSTA'19, Distinguished Paper Award]*
- Proposed a hierarchical DL approach to automatically learn the most effective features for precise fault localization.
- Significantly outperformed state-of-the-art with over 20% improvement.

## RESEARCH EXPERIENCE - IN INDUSTRY

**Intern, SoC Physical Design Group, Apple, United States**          May. 2024 – Aug. 2024
*Floorplan Encoder*
- Propose a novel floorplan encoder for the floorplanning task, the encoder is capable of encode the floorplan state, which is multi-modal, and includes multi-objects.
- Achieves 95% accuracy and shows 3X speedup to achieve the same quality compared to the industrial tool.

**Research Intern, Nvidia, United States**          May. 2023 – Aug. 2023
*Differentiable Global Routing [DAC'24]*
- A differentiabl eglobal router capable of concurrent optimization for millions of nets
- Reduced nets with overflow by more than 80%

**Intern, SoC Physical Design Group, Apple, United States**          June. 2022 – Sep. 2022
*Exploration of GNNs for Physical Design*
- Implemented a basic GNN model for predicting holder buffer before routing.
- Tried different methods: path-based, sub-circuit based, sub-graph based.

*Perfect Rectilinear Floorplanning*
- A Simulated Annealing based algorithm for perfect rectilinear floorplanning.
- Reinforcement learning, and supervised-learning that guides SA are also explored.
- **Employed in the industrial tool for all Apple chips starting from 2023.**

## TEACHING ASSISTANT

| | | |
|---|---|---|
| Spring 2023 | CMU | 18202 Mathematical Foundations of Electrical Engineering |
| Fall 2022,2023,2024,2025 | CMU | 18765 Digital System Testing and Testable Design |
| Spring 2020 | CUHK | CENG3420 Computer Organization and Design |
| Spring 2021 | CUHK | CENG2030 Fundamentals of Embedded Systems |

## SELECTED AWARDS AND HONORS

| | | |
|---|---|---|
| Best Paper Honorable Mention Award | ICLAD | 2025 |
| Jack and Mildred Bowers Scholarship | CMU | 2025 |
| Qualcomm Innovation Fellowship | Qualcomm Inc. | 2024 |
| Apple PhD fellowship in Integrated Systems | Apple Inc. | 2024 |
| Apple PhD fellowship in Integrated Systems | Apple Inc. | 2022 |
| Faculty Outstanding Thesis Award | Engineering Faculty, CUHK | 2021 |
| Dean's Fellowship | CMU | 2021 |
| Talent Development Scholarship | HKSAR Goverment | 2021 |
| Best Paper Award | ASP-DAC | 2021 |
| 1st Place Award in EDA elite challenge | Chinese Institute of Electronics | 2020 |
| Richard Newton Young Student Fellow | DAC | 2020 |
| Best Student Paper Award | ICTAI | 2019 |
| Distinguished Paper Award | ISSTA | 2019 |
| Full Postgraduate Studentship | CUHK | 2019-2021 |
| 2nd Place Award in CAD Contest | ICCAD | 2018 |
| ELITE Stream Student Scholarship | Faculty of Engineering, CUHK | 2018 |
| Undergraduate Admission Scholarship | Soong Ching Ling Foundation | 2015-2018 |

## PUBLICATIONS

**Journal Papers**

[J2] **Wei Li**, Yuzhe Ma, Yibo Lin, Bei Yu, "Adaptive Layout Decomposition with Graph Embedding Neural Networks", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (**TCAD**).

[J1] **Wei Li**, Yuzhe Ma, Qi Sun, Zhang Lu, Yibo Lin, Iris Hui-Ru Jiang, Bei Yu, David Z. Pan, "OpenMPL: An Open Source Layout Decomposer", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (**TCAD**).

**Conference Papers**

[C17] **Wei Li**, Yang Zou, Christopher Ellis, Ruben Purdy, Shawn Blanton and José Moura, "BRIDGE: Bridging Graph and Large Language Models in EDA", IEEE International Conference on LLM-Aided Design (**ICLAD**), 2025. **(Best Paper Honorable Mention Award, 1/94)**

[C16] Chris Nigh, Ruben Purdy, **Wei Li**, Subhasish Mitra, R.D. Blanton, " IC-PEPR: PEPR Testing Goes Intra-Cell", IEEE International Test Conference (**ITC**), 2025.

[C15] Ruben Purdy, Chris Nigh, **Wei Li**, R.D. Blanton, "CHEF: CHaracterizing Elusive Logic Circuit Failure" , IEEE VLSI Test Symposium (**VTS**) 2025.

[C14] Chris Nigh, Ruben Purdy,**Wei Li**, Subhasish Mitra, R.D. Blanton, "Faulty Function Extraction for Defective Circuits" , IEEE European Test Symposium (**ETS**) 2024.

[C13] **Wei Li**, Rongjian Liang, Anthony Agnesina, Haoyu Yang, Chia-Tung Ho, Anand Rajaram, Haoxing Ren, "DGR: Differentiable Global Routing", ACM/IEEE Design Automation Conference (**DAC**), San Francisco, 2024.

[C12] **Wei Li**, Ruben Purdy, Jose Moura, Shawn Blanton, "Characterize the ability of GNNs in attacking logic locking", ACM/IEEE Workshop on Machine Learning for CAD (**MLCAD**), Snowbird, Utah, Sep. 11–13, 2023.

[C11] **Wei Li**, Fangzhou Wang, Jose Moura, Shawn Blanton, "Global floorplanning via semidefinite programming", ACM/IEEE Design Automation Conference (**DAC**), San Francisco, July 9-13, 2023.

[C10] **Wei Li**, Chris Nigh, Danielle Duvalsaint, Subhasish Mitra, R.D. Blanton, "PEPR: Pseudo-Exhaustive Physical Region Testing", IEEE International Test Conference (**ITC**), Sep. 25 - Sep. 30, 2022.

[C9] **Wei Li**, Guojin Chen, Haoyu Yang, Ran Chen, Bei Yu, "Learning Point Clouds in EDA", ACM International Symposium on Physical Design (**ISPD**), Mar. 21–Mar. 24, 2021.

[C8] **Wei Li**, Yuxiao Qu, Gengjie Chen, Yuzhe Ma, Bei Yu, "TreeNet: Deep Point Cloud Embedding for Routing Tree Construction", IEEE/ACM Asian and South Pacific Design Automation Conference (**ASP-DAC**), Tokyo, Jan. 18–21, 2021. **(Best Paper Award)**

[C7] **Wei Li**, Jialu Xia, Yuzhe Ma, Jialu Li, Yibo Lin, Bei Yu, "Adaptive Layout Decomposition with Graph Embedding Neural Networks", ACM/IEEE Design Automation Conference (**DAC**), San Francisco, July 19-23, 2020.

[C6] Husheng Zhou, **Wei Li**, Yuankun Zhu, Yuqun Zhang, Bei Yu, Lingming Zhang, Cong Liu, "DeepBillboard: Systematic Physical-World Testing of Autonomous Driving Systems", ACM/IEEE International Conference on Software Engineering (**ICSE**), Seoul, May 23–29, 2020.

[C5] Yuzhe Ma, Zhuolun He, **Wei Li**, Tinghuan Chen, Lu Zhang, Bei Yu, "Understanding Graphs in EDA: From Shallow to Deep Learning", ACM International Symposium on Physical Design (**ISPD**), Taipei, Mar. 25–Apr. 01, 2020.

[C4] Yuzhe Ma, Ran Chen, **Wei Li**, Fanhua Shang, Wenjian Yu, Minsik Cho, Bei Yu, "A Unified Approximation Framework for Deep Neural Networks", The IEEE International Conference on Tools with Artificial Intelligence (**ICTAI**) 2019. **(Best Student Paper Award)**

[C3] **Wei Li**, Yuzhe Ma, Qi Sun, Yibo Lin, Iris Hui-Ru Jiang, Bei Yu, David Z. Pan, "OpenMPL: An Open Source Layout Decomposer", IEEE International Conference on ASIC (**ASICON**), Chongqing, China, Oct. 29–Nov. 1, 2019.

[C2] Xia Li, **Wei Li**, Yuqun Zhang, Yuqun Zhang, Lingming Zhang, "DeepFL: Integrating Multiple Fault Diagnosis Dimensions for Deep Fault Localization",The ACM SIGSOFT International Symposium on Software Testing and Analysis (**ISSTA**), 2019. **(Distinguished Paper Award)**

[C1] Bentian Jiang, Xiaopeng Zhang, Ran Chen, Gengjie Chen, Peishan Tu, **Wei Li**, Evangeline F. Y. Young, Bei Yu, "FIT: Fill Insertion Considering Timing", ACM/IEEE Design Automation Conference (**DAC**), Las Vegas, NV, June 2-6, 2019.

## TECHNICAL SKILLS

| | |
|---|---|
| **Languages** | Mandarin, Cantonese, English |
| **Programming Languages** | C/C++, Python, LaTeX |

# Beyond Rigid Tools:
# Multi-Modal Fluid Interaction for EDA Agents

Wei Li
Carnegie Mellon University

## Abstract

Recent advancements have positioned Large Language Model (LLM) agents at the forefront of Electronic Design Automation (EDA). However, the prevailing paradigm confines these agents to interacting with the design environment through a fixed set of text-based tool APIs. These predefined interfaces limit agent flexibility and fail to address the inherently multi-modal nature of hardware design, where artifacts like code, structural graphs, and simulation waveforms coexist. This research proposes a paradigm shift from rigid tools to **fluid interaction**, where agents are empowered to dynamically formulate queries and actions to explore, analyze, and correlate information across a unified, multi-modal representation of the design environment.

Our central hypothesis is that this increased freedom enables agents to develop more sophisticated and robust problem-solving strategies compared to traditional, tool-constrained methods. We will validate this framework across two complex, high-impact EDA tasks:

1. **Automated RTL Code Debugging:** A multi-agent system will diagnose and correct functional bugs by leveraging a holistic view of source code, specifications, and multi-modal simulation data.
2. **Logic Diagnosis and Diagnostic ATPG:** The system will perform fault diagnosis by interacting with both standard tools and a direct graph representation of the circuit, with its performance benchmarked against over 100 human expert reports.

Moreover, we will analyze the emergent interaction patterns and investigate the impact of our fluid-interaction model on inference-time scaling laws, exploring how increased agent autonomy affects the efficiency of repeated sampling. This work aims to establish a new paradigm for LLM-EDA interaction, paving the way for more capable, adaptable, and interpretable automated design assistants.

## 1 Background and Motivation

The automation of complex EDA tasks using LLM-based agents has progressed rapidly [13]. Specialized agents can now generate [7, 18] and debug RTL code [14], optimize standard cell layouts [6], automate verification [17] and design flows [12], and even craft heuristics for combinatorial optimization problems [2]. These pioneering efforts predominantly rely on agents that interact with the design environment through a set of predefined, text-based tools (APIs). While effective, this "rigid tools" paradigm imposes fundamental limitations. The fixed action space restricts the agent's reasoning pathways to what has been pre-programmed by human developers, hindering flexibility and adaptability.

Early evidence already suggests the profound impact of enhanced interaction capabilities. For instance, VerilogCoder [7] demonstrated that an agent with access to k-hop AST of RTL code—a limited yet more flexible form of interaction—was a key factor in substantially improving its functional bug-fixing success rate. This significant gain from a single, structured tool motivates our hypothesis that a fully programmatic and multi-modal interface could unlock even greater performance. Another recent study [5] also showed that, enabling agents to interact with graphs via code generation has unlocked superior capabilities than with fixed tool APIs.

This limitation is particularly acute in EDA, where design artifacts are inherently multi-modal. A hardware design is simultaneously a textual specification, a structural netlist graph, a behavioral simulation waveform, and a physical layout. Current agents, by processing only the textual outputs of discrete tools originally designed for human-centric workflows, are siloed from this rich, interconnected data landscape.

This proposal posits that the next leap in LLM-aided EDA requires a paradigm shift from using rigid tools to fluidly interacting with the design environment itself. We define "fluid interaction" as the agent's ability to

query, explore, analyze, and manipulate a multi-modal representation of the design. This approach allows the agent to dynamically formulate its own "tools" on the fly, adapting its strategy based on the specific context of the problem. Such a capability could unlock more sophisticated reasoning, as the agent is no longer a passive user of tools but an active explorer of the design space.

However, this raises critical scientific questions: (RQ1) Does greater autonomy in interaction lead to superior performance, or does the expanded action space introduce prohibitive complexity? Furthermore, do more powerful models derive a differential advantage from this richer environment? (RQ2) Can we identify emergent, generalizable interaction patterns from these behaviors and learn from them [3]? (RQ3) As explored in recent studies on inference scaling [1], how does this increased complexity impact the efficiency of test-time compute strategies like repeated sampling? Answering these questions is also essential for building the next generation of truly intelligent EDA agents.

## 2 Research Objectives and Methods

The primary objective of this research is to design, implement, and validate a multi-modal, fluid-interaction agentic framework for EDA problem-solving. We will test the hypothesis that enabling agents to generate code for environment interaction leads to superior performance and more sophisticated reasoning compared to fixed-API approaches.

### 2.1 Objective 1: Develop a Fluid Multi-Modal Agentic Framework

Inspired by the success of modular architectures [18, 7, 16], our framework will feature a multi-agent system based on the ReAct paradigm [15] centered around a Multi-Modal Design Representation (MMDR). The MMDR will serve as a unified, queryable database integrating diverse design artifacts, including:

- **Textual Data:** RTL source code, specifications, testbenches, and tool logs.
- **Graph Data:** Circuit netlists and design hierarchies represented as interactive graphs (e.g., using NetworkX or tabular methods), such as Abstract Syntax Trees (ASTs), Data Flow Graphs (DFGs), and Control Flow Graphs (CFGs).
- **Behavioral Data:** Simulation results and waveforms stored in a structured, queryable format (e.g., Pandas DataFrames).

The MMDR is designed for extensibility and is not limited to these modalities; it can be augmented to include physical layout images, congestion heatmaps, and static timing analysis (STA) reports in future work.

The multi-agent architecture will consist of a Planner Agent, multiple Specialist Agents, and a Reflector. The Planner Agent [8, 4], will decompose high-level tasks into a sequence of sub-goals. For each sub-goal, it will select an appropriate Specialist Agent (e.g., GraphAnalysisAgent, SimulationAgent). The core innovation lies in the action mechanism: agents will operate within a hybrid action space, allowing them to (1) call traditional academic or commercial tools, (2) generate programmatic code (e.g., Python scripts) to directly query and manipulate the MMDR, and (3) retrieve insights from an evolving knowledge base or from auxiliary ML-aided methods. To foster self-improvement, generated trajectories will be refined by the reflector and stored in the involving knowledge base ("playbook") [16] to enhance the agent's strategic capabilities within the multi-modal environment.

### 2.2 Objective 2: Case Study 1 - Automated Multi-Modal RTL Debugging

This case study will target RTL code debugging. The MMDR will be populated with the buggy RTL code, its specification, testbench, simulation failure logs organized in a tabular format, and the design's structural graphs (AST, DFG, CFG). The agent's goal is to autonomously identify and correct both syntax and function errors.

**Methodology:** The Planner Agent will hypothesize potential error sources (e.g., "incorrect state transition logic"). A Specialist Agent will then be tasked to interact with the multi-modal environment to validate this hypothesis by, for instance, generating code to traverse the CFG and trace signal dependencies, comparing the logic with the textual specification, or by programmatically analyzing the simulation waveform around the point of failure. This iterative "hypothesize-query-verify" loop mimics expert human debugging but with the scalability of an automated agent.

**Evaluation:** We will evaluate the bug fix rate on established RTL debugging benchmarks [11, 9, 10]. An ablation study will be conducted to quantify the performance contribution of each modality (e.g., waveform vs. graph vs. text-only).

## 2.3 Objective 3: Case Study 2 - Interactive Logic Diagnosis and Human Benchmarking

This objective pushes the boundaries of agent capabilities by tackling logic diagnosis for Single-Stuck-Line (SSL) and Bridge faults. We will provide the agent with a suite of five standard academic analysis tools, along with direct programmatic access to the circuit netlist graph, which is annotated with simulation values from test patterns.

**Methodology:** The multi-agent task is twofold: 1) to identify the fault location by intelligently using the provided tools and querying the annotated graph; 2) for advanced scenarios, to perform Diagnostic ATPG by generating test patterns to differentiate between fault candidates, without relying on an external ATPG tool.

**Evaluation:** We will benchmark the agent's performance (accuracy and runtime) against a unique, pre-collected dataset of over 100 human expert attempts on the same problems and environment. This will provide the first large-scale, quantitative comparison of LLM agents and human engineers on a complex, open-ended EDA reasoning task, offering insights that go beyond LLM-vs-LLM leaderboards.

## 2.4 Objective 4: Analyze Emergent Interaction Paradigms and Inference Scaling

This objective addresses the fundamental research questions of our proposal.

**Methodology for RQ1 & RQ2:** Through case-by-case analysis of the agent's programmatic query history, we will qualitatively identify and categorize emergent interaction patterns. We aim to discover if agents can autonomously develop novel debugging or diagnostic strategies that differ from human heuristics, potentially leading to new standardized methods for agent-environment interaction.

**Methodology for RQ3:** Our analysis will extend the methodology from recent work on inference scaling [1]. We will investigate how the relationship between coverage (pass@k) and sample count (k) is modulated by the complexity of the agent's action space. We will fit the data to mathematical models (e.g., exponentiated power law) to determine if the increased action space complexity enhances or diminishes the efficiency of sampling.

# 3 Expected Impact

This research is poised to make significant contributions to both the LLM-Aided Design field and the broader design automation and Machine Learning community.

1. **A New Paradigm for Agent-Environment Interaction:** The framework will pioneer a move away from static tool-use towards dynamic, programmatic interaction with multi-modal design data, providing powerful and flexible interaction insights for EDA agents.

2. **State-of-the-Art Performance on Unsolved EDA Problems:** This project tackles two critical and time-consuming tasks in the IC design flow: RTL debugging and fault diagnosis. Success in automating these areas would provide a significant practical impact, potentially saving thousands of engineering hours and accelerating the time-to-market for complex chips.

3. **The First Rigorous LLM Agent-vs-Human Benchmark in EDA:** Our quantitative comparison on logic diagnosis will provide invaluable, objective insights into the current capabilities and limitations of LLM agents relative to human experts, guiding future research in human-AI collaborative design.

4. **Insights into Agent Behavior and Inference Scaling:** The analysis of emergent interaction patterns and the study of inference scaling laws in a fluid environment will contribute fundamental knowledge to the broader field of autonomous AI agents, extending beyond the domain of EDA.

Ultimately, this work will pave the way for a new generation of LLM-based EDA tools that are not just assistants but active, intelligent partners in the hardware design and verification process.

# References

[1] Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

[2] Hongzheng Chen, Yingheng Wang, Yaohui Cai, Hins Hu, Jiajie Li, Shirley Huang, Chenhui Deng, Rongjian Liang, Shufeng Kong, Haoxing Ren, et al. Heurigym: An agentic benchmark for llm-crafted heuristics in combinatorial optimization. *arXiv preprint arXiv:2506.07972*, 2025.

[3] Michael Desmond, Ja Young Lee, Ibrahim Ibrahim, James M Johnson, Avirup Sil, Justin MacNair, and Ruchir Puri. Agent trajectory explorer: Visualizing and providing feedback on agent trajectories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 29634–29636, 2025.

[4] Lutfi Eren Erdogan, Nicholas Lee, Sehoon Kim, Suhong Moon, Hiroki Furuta, Gopala Anumanchipalli, Kurt Keutzer, and Amir Gholami. Plan-and-act: Improving planning of agents for long-horizon tasks. *arXiv preprint arXiv:2503.09572*, 2025.

[5] Ben Finkelshtein, Silviu Cucerzan, Sujay Kumar Jauhar, and Ryen White. Actions speak louder than prompts: A large-scale study of llms for graph inference. *arXiv preprint arXiv:2509.18487*, 2025.

[6] Chia-Tung Ho and Haoxing Ren. Large language model (llm) for standard cell layout design optimization. In *2024 IEEE LLM Aided Design Workshop (LAD)*, pages 1–6. IEEE, 2024.

[7] Chia-Tung Ho, Haoxing Ren, and Brucek Khailany. Verilogcoder: Autonomous verilog coding agents with graph-based planning and abstract syntax tree (ast)-based waveform tracing tool. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 300–307, 2025.

[8] Zhuofeng Li, Haoxiang Zhang, Seungju Han, Sheng Liu, Jianwen Xie, Yu Zhang, Yejin Choi, James Zou, and Pan Lu. In-the-flow agentic system optimization for effective planning and tool use. *arXiv preprint arXiv:2510.05592*, 2025.

[9] Shang Liu, Yao Lu, Wenji Fang, Mengming Li, and Zhiyao Xie. Openllm-rtl: Open dataset and benchmark for llm-aided design rtl generation. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, pages 1–9, 2024.

[10] Nathaniel Pinckney, Christopher Batten, Mingjie Liu, Haoxing Ren, and Brucek Khailany. Revisiting verilogeval: A year of improvements in large-language models for hardware code generation. *ACM Transactions on Design Automation of Electronic Systems*, 2025.

[11] Nathaniel Pinckney, Chenhui Deng, Chia-Tung Ho, Yun-Da Tsai, Mingjie Liu, Wenfei Zhou, Brucek Khailany, and Haoxing Ren. Comprehensive verilog design problems: A next-generation benchmark dataset for evaluating large language models and agents on rtl design and verification. *arXiv preprint arXiv:2506.14074*, 2025.

[12] Haoyuan Wu, Haisheng Zheng, Zhuolun He, and Bei Yu. Divergent thoughts toward one goal: Llm-based multi-agent collaboration system for electronic design automation. *arXiv preprint arXiv:2502.10857*, 2025.

[13] Kangwei Xu, Denis Schwachhofer, Jason Blocklove, Ilia Polian, Peter Domanski, Dirk Pflüger, Siddharth Garg, Ramesh Karri, Ozgur Sinanoglu, Johann Knechtel, et al. Large language models (llms) for electronic design automation (eda). *arXiv preprint arXiv:2508.20030*, 2025.

[14] Ke Xu, Jialin Sun, Yuchen Hu, Xinwei Fang, Weiwei Shan, Xi Wang, and Zhe Jiang. Meic: Re-thinking rtl debug automation using llms. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, pages 1–9, 2024.

[15] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

[16] Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, et al. Agentic context engineering: Evolving contexts for self-improving language models. *arXiv preprint arXiv:2510.04618*, 2025.

[17] Yujie Zhao, Zhijing Wu, Hejia Zhang, Zhongming Yu, Wentao Ni, Chia-Tung Ho, Haoxing Ren, and Jishen Zhao. Pro-v: An efficient program generation multi-agent system for automatic rtl verification. *arXiv preprint arXiv:2506.12200*, 2025.

[18] Yujie Zhao, Hejia Zhang, Hanxian Huang, Zhongming Yu, and Jishen Zhao. Mage: A multi-agent engine for automated rtl code generation. In *2025 62nd ACM/IEEE Design Automation Conference (DAC)*, pages 1–7. IEEE, 2025.