Research Statement

Wei Li, CMU ECE

As silicon complexity scales toward trillion-transistor systems, the demand for specialized Artificial Intelligence (AI) hardware is outpacing the productivity of traditional Electronic Design Automation (EDA) methodologies. Designing a flagship chip at advanced nodes now requires massive engineering teams and months of "human-in-the-loop" iterations (see left of Figure 1). Exemplified by unprecedented industry moves such as NVIDIA's $2 billion investment in Synopsys, the industry is transitioning from human-driven, tool-assisted flows toward autonomous EDA, where AI moves the human role from active 'in-the-loop' intervention to strategic 'on-the-loop' supervision.

However, realizing this vision requires overcoming four fundamental challenges: (i) Fragmentation in optimizations, where upstream decisions lack visibility into downstream physical and manufacturing constraints; (ii) Modality gaps between heterogeneous design representations and current AI architectures, preventing AI from capturing the full design context; (iii) Scalability limits of existing AI approaches for billion-scale combinatorial problems; and (iv) Lack of Robustness, where AI designs fail physical verification because AI-for-design lacks feedback from real-world manufacturing constraints. In light of these challenges, my research aims to enable the transition toward truly end-to-end autonomous EDA, grounded in three pillars: Perception (understanding multi-modal data), Action (executing scalable optimizations for superior power, performance and area), and Grounding (anchoring AI in physical reality).

I am well positioned to contribute to this transition because: (i) I have cross-disciplinary expertise that combines AI, EDA, and hardware testing, evidenced by publications in top venues and three Best Paper Awards. Specifically, my work fused geometric deep learning [1] and Large Language Models (LLMs) [2] with EDA, fundamentally rethinking how AI perceives chip topology and semantics. My works in robust system [3], [4] and hardware testing [5] bridge the gap between algorithmic design and physical reality, delivering solutions developed in close collaboration with and actively evaluated by major industry collaborators. (ii) My work combines mathematical rigor with scalability, replacing heuristics with differentiable and analytical solvers (e.g., Semidefinite Programming (SDP) [6], differentiable routing [7]) to ensure global optimality; (iii) Finally, my accomplishments deliver proven industrial value, with algorithms deployed in industrial production flows, earning recognition through the Qualcomm Innovation Fellowship and two Apple PhD Fellowships in Integrated Systems.

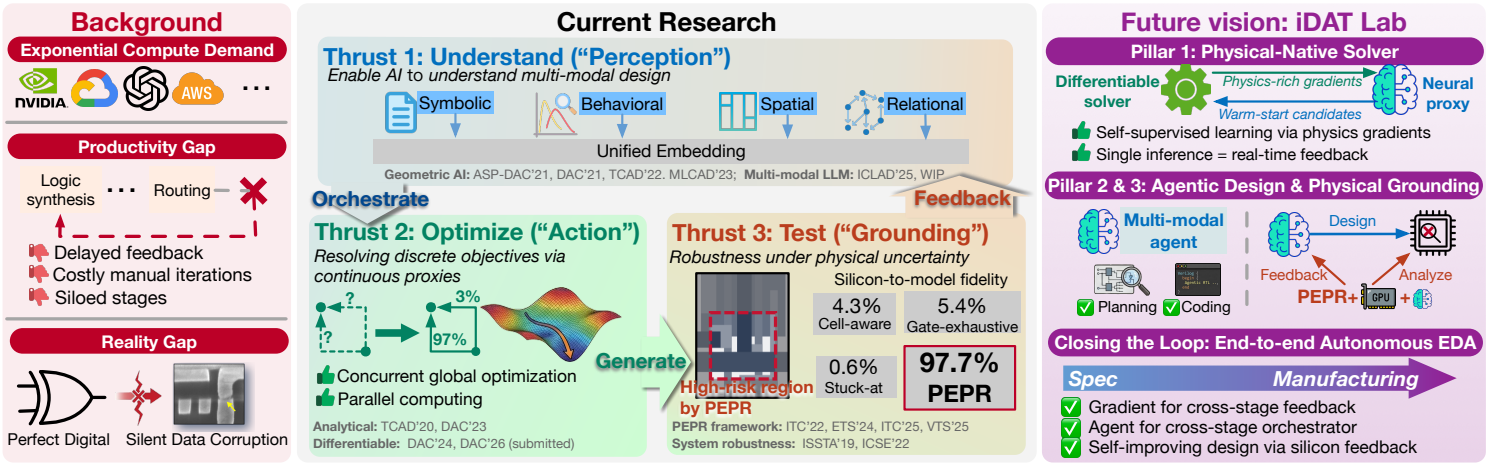

Figure 1: Overview of background, current research, and future vision towards End-to-End Autonomous EDA.

1 Current Research: Foundations of Autonomous EDA

Thrust 1: Multi-Modal Perception for EDA (The "Eyes")

Autonomous EDA begins with perception, the capability for AI to understand the design context. However, design context presents unique challenges to standard AI models due to its complex, highly structured, and multi-modal nature. My research asks: *How can we enable AI to perceive the complex, multi-modal nature of integrated circuits?*

Multi-Modal Representations in EDA tasks. My research introduced geometric deep learning for EDA by viewing chips as native multi-modal geometric structures. Our work [8], [9] modeled pins as point clouds and systematically studied which point-cloud neural architectures best encode pins to guide routing tree construction. This approach was among the first to demonstrate that geometric learning could outperform traditional heuristics in routing tasks, and was recognized with the Best Paper Award at ASP-DAC 2021. In parallel, we developed Graph Neural Network (GNN) architectures tailored to layout decomposition [1], [10] and logic locking [11], laying early foundations for multi-modal learning in EDA with 100+ follow-up studies in just a few years.

Theoretical Limits of Graph Learning. Blind application of GNNs in EDA often fails because graphs associated with hardware fundamentally differ from graphs in other fields. In [11], we rigorously proved that message-passing GNNs are upper-bounded by the heterogeneous and directed Weisfeiler-Lehman test in distinguishing netlist isomorphism. We further identified that heterophily (dissimilar neighbors) is a key characteristic of layout graphs for the layout decomposition task, leading to new GNN architectures designed specifically for the graph coloring problem [12]. This theoretical grounding ensures mathematically sound models beyond mere empirical tuning, and has inspired 40+ subsequent works by others.

BRIDGES: Unifying Graph Learning and LLMs. Recognizing that chips are described by both semantics (RTL) and topology (netlist), we developed BRIDGES [2], the first framework to effectively fuse LLMs with GNNs for EDA tasks. BRIDGES enables an LLM to "understand" graph-modality information such as the netlist structure. By projecting graph embeddings into the LLM's token space, BRIDGES achieves 2x-10x improvements over text-only baselines across multiple EDA tasks. The framework was open-sourced and reached 50+ stars and downloads on Github and Hugging Face within months. This work was recognized as the Best Paper Honorable Mention at ICLAD'25, and has attracted interest from industry, e.g., NVIDIA, MediaTek, and Apple, motivating follow-up discussions with industry leaders on deploying BRIDGES in industrial workflows.

Thrust 2: Analytical and Differentiable Optimization (The "Hands")

Most EDA tasks are combinatorial optimization problems plagued by discrete constraints (e.g., binary digits). My approach maps discrete problems into continuous landscapes, and the global gradients can guide the optimization out of local minima.

Analytical Optimizations. I specialize in reformulating discrete, geometric design challenges into globally solvable analytical models. We developed a global floorplanning framework using SDP [6], providing global optimality guarantees under specific objective relaxations. This analytical approach was evaluated from industry for its potential to automate the floorplanning of complex SoCs, a task that currently consumes days of human effort. Similarly, OpenMPL [13], our open-source analytical solvers for layout decomposition, became a standard benchmarking suite in the community, accumulating 80+ stars on GitHub.

Differentiable Programming. Differentiable Programming facilitates (1) GPU parallelism and (2) global gradient-based guidance that escapes the local optima that traps traditional tools. In collaboration with NVIDIA, we developed DGR [7], shifting global routing from sequential heuristics to a concurrent, differentiable optimization paradigm. DGR formulates global routing as a candidate path selection problem for millions of nets simultaneously. By optimizing the entire routing solution concurrently, DGR resolves congestion holistically rather than locally. This work, currently under a joint patent filing with NVIDIA, reduced congestion overflows by 23.9% on average compared to state-of-the-art academic global routers. Extending this differentiable paradigm to hardware testing, we introduce DEFT [14], reformulating discrete ATPG as a continuous optimization task using a novel reparameterization technique. DEFT optimizes a global objective

to minimize pattern count while maximizing coverage across all hard-to-detect (HTD) faults simultaneously. On industrial benchmarks, DEFT improved the detection of Hard-to-Detect (HTD) faults by 21.1% to 48.9% compared to a leading commercial tool, under the same pattern budget and comparable runtime.

Thrust 3: Robustness Under Physical Uncertainty (The "Grounding")

A design is good only if grounded in reality. My research philosophy is that robustness must be verified not just in abstract models, but under the stochastic uncertainty of the physical world. My pursuit of robustness began at the system level, where we pioneered realistic and continuous physical-world tests for autonomous driving systems [4]. Our work concerning system robustness, including DeepBillboard [4] and DeepFL [3] (Best Paper Award, ISSTA 2019), has received nearly 600 citations. Bringing this "reality-first" philosophy to EDA, I recognized that legacy fault models fail to capture the silicon reality. We introduced PEPR (Pseudo-Exhaustive Physically-aware Region Testing)[5] to bridge the gap between abstract netlists and physical layout by identifying all "physical regions" susceptible to defects that legacy models miss. Analysis of 30,000 failing industrial 14nm chips demonstrated that fault behavior characterized by PEPR matched 97.7% defects (vs. 0.6% for stuck-at fault and 5.4% for gate-exhaustive, see Figure 1). Building on this success, we expanded PEPR into a comprehensive suite for reliability, such as intra-cell testing [15], diagnosis [16], and faulty function extraction [17]. This body of work discusses the industry pain point of silent data corruption in data centers, and is conducted in long-term collaboration with Stanford University. It has and continues to attract funding and active collaboration from Broadcom, Google, Intel, Qualcomm, and other industry leaders.

Future Research Agenda: The iDAT Lab

My long-term goal is to realize Autonomous EDA as a unified, end-to-end design engine, shifting the human role from active intervention to strategic supervision. To achieve this, I will establish the **intelligent Design Automation & Testing (iDAT)** lab. We envision a future where: (1) heuristic black boxes are replaced by differentiable solvers or learned proxies; (2) upstream stages receive instant and comprehensive feedback from downstream constraints; (3) multi-modal agents orchestrate design flows by reasoning over feedback signals and dynamically invoking tools with optimized arguments; and (4) manufacturing-aware design becomes self-improving with every tapeout.

Pillar 1: The Physics-Native Design Engine (Near-Term, 1-2 Years)

We will evolve differentiable solvers into real-time physics learning engines via an "AI-warm-start, differentiable-fine-tune" paradigm (See Figure 1). By training neural proxies through physics-rich gradients from analytical solvers, we enable single-inference generation of high-quality design candidates. For instance, gradients in DGR, representing the congestion sensitivity for each grid, can be used to train a customized GNN to predict a global routing solution directly. These candidates serve as "warm starts" that are subsequently fine-tuned by the differentiable solver. Together, these techniques enable near-interactive "what-if" analyses ("if I move this macro, how does it affect congestion?") where designers receive physical feedback in seconds.

Pillar 2: Multi-Modal EDA Agents (Mid-Term, 1-4 Years)

While Pillar 1 establishes the "differentiable hands", Pillar 2 constructs the "multi-modal brain" to orchestrate solvers and tools.

Multi-Modal Reasoning for RTL Generation. We will extend our BRIDGES framework [2] to verify that design intelligence requires an abstract multi-modal environment, which integrates graph (dataflow), text (specifications), and waveforms (behavior). My hypothesis is that functional correctness in RTL generation stems from joint reasoning over these diverse modalities. Preliminary results on NVIDIA's CVDP benchmark demonstrate that such multi-modal inputs can elevate functional correctness from 40% to 85% compared to text-only LLMs.

Bridging Logic and Silicon via Diagnosis. Beyond RTL generation, I will develop agents to navigate the intricate and non-obvious link between digital logic and physical silicon. By analyzing tester responses and physical layouts in real-time, these agents will mimic expert reasoning to isolate defects. Success in this

high-complexity diagnosis domain is a foundational step; it validates the agent's ability to combine physical and logical constraints, proving its potential for autonomous end-to-end design.

Pillar 3: Scalable Testing and Intelligent Yield Learning (Mid-Term, 1-4 Years)

My third pillar anchors abstract design models in the physical reality of manufacturing.

To transform PEPR from a post-silicon analysis tool into a ubiquitous sign-off engine, I will pursue hardware-algorithm co-optimization. First, acknowledging that region analysis is inherently a massive parallel geometric problem, we will re-architect PEPR as a GPU-native engine to achieve orders-of-magnitude speedup, leveraging my experience in CUDA [7], [14] and AI acceleration [18]. Second, we will evolve PEPR from "exhaustive" to "predictive," and utilize active learning agents to direct computational resources toward high-risk regions, making physical-aware verification tractable at full-chip scale.

Beyond testing, I aim to answer a foundational question: *How can we turn physical failure data into design wisdom?* Current pre-silicon yield models are either hand-crafted rules-of-thumb or proprietary black boxes. We will propose data-driven yield learning to close the loop. By correlating silicon failure data with upstream design features [17], we can make agents evolve with every tapeout, allowing designers to visualize risk regions during the pre-silicon phase.

Closing the Loop: End-to-End Autonomous Engine (Long-Term, 3-5 Years)

My ultimate objective is the full integration of these pillars into a unified design engine.

Cross-Stage Gradient Propagation. Upstream tools currently optimize for abstract objectives without downstream visibility. They rely on loose proxies, failing to capture the true physical costs of design decisions and often deviate from actual constraints. I propose to bridge this gap by back-propagating physical gradients from downstream solvers (e.g., DGR's congestion or DEFT's testability) or learned neural proxies to guide upstream tasks (e.g., congestion- and testability-aware logic synthesis). This mathematically minimizes physical violations by providing real-time, physics-based feedback during early design stages.

Multi-Modal Agent as Cross-Stage Orchestrator. The multi-modal agent will serve as the *Strategic Orchestrator*, managing the end-to-end design closure. Unlike current human-centric flows, these agents leverage multi-modal perception to perform intelligent planning and decision that shift the design process from a slow, human-in-the-loop cycle to an autonomous, 24/7 accelerated flow. To realize this vision, my research includes three phases: First, we will integrate downstream multi-modal feedback (e.g., timing reports) into the agent's reasoning loop, enabling it to proactively orchestrate tool chains and optimize hyper-parameters. Building upon this reasoning core, we will investigate long-horizon memory mechanisms. This is critical to prevent the agent from "getting lost" in the vast design context and hundreds of EDA command sequences during multi-day autonomous runs. Finally, these capabilities will be integrated into and evaluated on OpenROAD, the leading open-source EDA flow, in addition to planned collaborations with commercial EDA vendors.

Collaboration Plans and Funding opportunities

My research is inherently interdisciplinary, bridging the gap between machine learning, circuit optimization, and physical manufacturing. To realize the vision of the iDAT Lab and an end-to-end Autonomous EDA system, I plan to collaborate broadly across the EECS community : (i) partnering with ML researchers to further scale geometric deep learning and multi-modal LLMs for billion-scale netlist and layout problems; (ii) collaborating with computer architecture and VLSI design groups to integrate my differentiable solvers (e.g., DGR, DEFT) into production flows to evaluate system-level PPA; and (iii) working with manufacturing and testing experts to evolve the PEPR framework for advanced-node reliability and yield learning.

My funding strategy leverages federal, industrial, and consortium support. I will target NSF (FuSe2, Future CoRe) for foundational AI-EDA research, DARPA (ERI 2.0) for autonomous EDA and hardware assurance, and CHIPS Act (NAPMP, NSTC) for manufacturing-aware design. Furthermore, I will pursue SRC (GRC, JUMP 2.0) and direct research gifts from long-term industry collaborators (e.g., Apple, NVIDIA, Google, Intel, Broadcom) to address industry pain points.

References

[1] Y. Ma, Z. He, **Li, Wei**, L. Zhang, and B. Yu, "Understanding graphs in EDA: From shallow to deep learning," in *Proceedings of the 2020 International Symposium on Physical Design (ISPD)*, 2020, pp. 119–126.

[2] **Li, Wei**, Y. Zou, C. Ellis, R. Purdy, R. D. Blanton, and J. M. F. Moura, "BRIDGES: Bridging graph modality and large language models within EDA tasks," in *IEEE International Conference on LLM-Aided Design (ICLAD)*, 2025, pp. 77–84.

[3] X. Li, **Li, Wei**, Y. Zhang, and L. Zhang, "DeepFL: Integrating multiple fault diagnosis dimensions for deep fault localization," in *Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA)*, 2019, pp. 169–180.

[4] H. Zhou, **Li, Wei**, Z. Kong, *et al.*, "DeepBillboard: Systematic physical-world testing of autonomous driving systems," in *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering (ICSE)*, 2020, pp. 347–358.

[5] **Li, Wei**, C. Nigh, D. Duvalsaint, and S. Mitra, "PEPR: Pseudo-exhaustive physically-aware region testing," in *IEEE International Test Conference (ITC)*, 2022, pp. 314–323.

[6] **Li, Wei**, F. Wang, J. M. F. Moura, and R. D. Blanton, "Global floorplanning via semidefinite programming," in *Proceedings of the 60th ACM/IEEE Design Automation Conference (DAC)*, 2023, pp. 1–6.

[7] **Li, Wei**, R. Liang, A. Agnesina, *et al.*, "DGR: Differentiable global router," in *Proceedings of the 61st ACM/IEEE Design Automation Conference (DAC)*, 2024, pp. 1–6.

[8] **Li, Wei**, Y. P. Qu, G. Chen, Y. Ma, and B. Yu, "TreeNet: Deep point cloud embedding for routing tree construction," in *Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2021, pp. 164–169.

[9] **Li, Wei**, G. Chen, H. Yang, R. Chen, and B. Yu, "Learning point clouds in EDA," in *Proceedings of the 2021 International Symposium on Physical Design (ISPD)*, 2021, pp. 55–62.

[10] **Li, Wei**, Y. Ma, Y. Lin, and B. Yu, "Adaptive layout decomposition with graph embedding neural networks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 5030–5042, 2022.

[11] **Li, Wei**, R. Purdy, J. M. F. Moura, and R. D. Blanton, "Characterize the ability of GNNs in attacking logic locking," in *ACM/IEEE 5th Workshop on Machine Learning for CAD (MLCAD)*, 2023, pp. 1–6.

[12] **Li, Wei**, R. Li, Y. Ma, S. O. Chan, D. Z. Pan, and B. Yu, "Rethinking graph neural networks for the graph coloring problem," *arXiv preprint arXiv:2208.06975*, 2022.

[13] **Li, Wei**, Y. Ma, Q. Sun, *et al.*, "OpenMPL: An open-source layout decomposer," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 40, no. 11, pp. 2331–2344, 2020.

[14] **Li, Wei**, Y. Zou, Y. Liang, J. M. F. Moura, and R. D. Blanton, "DEFT: Differentiable automatic test pattern generation," in *Proceedings of the 64th ACM/IEEE Design Automation Conference (DAC)*, Submitted, 2026.

[15] C. Nigh, R. Purdy, **Li, Wei**, S. Mitra, and R. D. Blanton, "IC-PEPR: PEPR testing goes intra-cell," in *IEEE International Test Conference (ITC)*, 2025, pp. 301–309.

[16] R. Purdy, C. Nigh, **Li, Wei**, and R. D. Blanton, "CHEF: Characterizing elusive logic circuit failures," in *IEEE 43rd VLSI Test Symposium (VTS)*, 2025, pp. 1–7.

[17] C. Nigh, R. Purdy, **Li, Wei**, S. Mitra, and R. D. Blanton, "Faulty function extraction for defective circuits," in *Proceedings of the IEEE*, 2024.

[18] Y. Ma, R. Chen, **Li, Wei**, *et al.*, "A unified approximation framework for compressing and accelerating deep neural networks," in *IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, 2019, pp. 376–383.