Research Statement

Jiong Yang
Ph.D. Student at Georgia Institute of Technology
https://al-jiongyang.github.io

Overview

Automated reasoning lies at the intersection of artificial intelligence and formal methods, serving as a foundation for verifying hardware, accelerating software development, and securing large-scale cloud systems. Over the past three decades, a virtuous cycle between applications and techniques has emerged: real-world demands drive automated reasoning from a theoretical pursuit into a core technology, which in turn fuels broader adoption. As progress continues, scaling to increasingly complex problems remains a persistent challenge due to inherent computational complexity, and with its widespread adoption, vulnerability to subtle flaws that can propagate downstream has emerged as a growing concern. To address these challenges, my research pushes the frontier of **efficient** and **trusted** automated reasoning from both theoretical and practical perspectives, while extending its **impact** to emerging domains such as quantum computing and machine learning. I leverage techniques such as *randomized algorithms*, *proof certification*, *parallelization*, and *causal reasoning* to enhance scalability and strengthen trust in reasoning systems.

The first two thrusts of my research advance *efficiency* and *trustworthiness*. For *efficiency*, I improved the performance of approximate counting and SAT solving through novel randomized algorithmic designs and effective system engineering. For instance, we introduced a rounding technique to mitigate under- and over-estimation errors in approximate counting, allowing the algorithm to terminate earlier and solving 23% of instances that were previously out of reach for the state of the art. This work proposed a new randomized scheme for approximate counting and was recognized with the **Distinguished Paper Award** at CAV 2023. Building on this foundation, I turned to enhancing the *trustworthiness* of reasoning systems, as flaws in reasoning engines can expose downstream applications to vulnerabilities. We developed the first formally certified approximate counter that provides provably reliable model count estimates by combining a static proof of its probabilistic guarantees in the Isabelle/HOL proof assistant with dynamic per-run certification of solver calls. This certified reasoning framework established a new paradigm for verifying randomized algorithms and was recognized with the **Distinguished Paper Award** at CAV 2024.

The third thrust of my research extends automated reasoning to emerging domains, demonstrating its *practical impact* in quantum computing and machine learning. In quantum computing, we developed a SAT based quantum circuit compiler that minimizes costly and error-prone SWAP gates arising from limited qubit connectivity. By combining incremental and parallel SAT solving, our approach reduced compilation time from hours to minutes-achieving a 26× speedup-on important quantum applications. This work received the **Best Student Paper Runner-Up Award** at SAT 2024. In machine learning, verifying neural networks presents a fundamental tension between scalability and soundness: scalable verifiers often produce unsound results, while certified ones are prohibitively slow. To resolve this, we designed an efficient certified verifier for Binarized Neural Networks (BNNs) that achieves reliability without sacrificing efficiency. Our verifier introduces a native representation of BNN constraints in model counting, together with specialized reasoning and proof-checking pipelines, achieving a 218× speedup over existing methods and enabling the first practical certified neural network verification. This work was recognized with the **Best Student Paper Runner-Up Award** at SAT 2025.

Thrust 1: Redefining the Efficiency Frontier of Counting

My first research thrust focuses on bridging the gap between SAT solving and model counting, with the long-term goal of making counting as nearly fast as solving through randomization and approximation. I address this challenge by tightly integrating *algorithmic design* with *system engineering*, leveraging the strengths of both.

Algorithmic Design. Novel algorithmic design is key to efficient counting. Current randomized algorithms partition the solution space using hashing functions and estimate counts by enumerating solutions within a sampled cell and scaling by the number of cells. While effective in moderate settings, their performance degrades under low-error requirements, preventing adoption in critical domains. To overcome this bottleneck, we first developed a new analysis that decomposes approximation error into under- and over-estimation components, revealing that overall error is dominated by the larger of the two. We then introduced a *rounding* algorithm that leverages this decomposition to selectively correct under- and over-estimation by adjusting outlier cells upward or downward to balance the two errors. The reduced overall error enables early termination of the randomized procedure, achieving a 4× improvement in efficiency [1]. This work was recognized with the **Distinguished Paper Award** at CAV 2023. We further advanced counting efficiency with a single-call-per-partition algorithm that exhaustively divides the solution space until each cell contains roughly one solution, yielding a constant-factor approximation based on the number of partitions. This approach reduces SAT calls by an order of magnitude, marking a significant step toward practical counting [5]. For instance, in verifying BNN fairness, a task that previously timed out after 5,000 seconds now completes in 798 seconds with rounding, and just 60 seconds via single-call partitioning, achieving an overall speedup of more than 83×.

System Engineering. Effective system design further drives practical performance. For BNN verification, the CNF representation is inherently unnatural, and the reliance of modern SAT solvers on the resolution proof system severely limits efficiency. Over the past decade, solvers with native support for Pseudo-Boolean (PB) constraints have emerged, offering a more natural encoding for BNNs. However, enabling quantitative reasoning on BNNs via PB encoding requires a PB model counter—yet none exists, and developing an approximate PB counter further demands an expressive solver capable of jointly reasoning over PB and XOR constraints. To bridge this gap, we built the first PB-XOR solver, LinPB, capable of solving PB and XOR constraints natively via cutting planes and Gaussian elimination, and further implemented the first approximate PB model counter, ApproxMC-PB. In evaluations on 1,076 BNN instances, ApproxMC-PB solved 110 more cases than CNF-based counters with re-encoding [8]. We further enhanced practical efficiency through preprocessing techniques that trade approximation quality for runtime. In projected model counting—where only a subset of variables is counted—efficiency improves as the projection set shrinks. To exploit this, we introduced the notion of *upper bound support*, a reduced variable set whose projected count upper-bounds the original model count. Our implemented preprocessor for identifying such supports significantly accelerates counting, helping solve 208 additional instances on a benchmark suite of 2,632 instances [7].

Thrust 2: Establishing the Trust Foundations of Randomized Systems

Ensuring reliable outputs from automated reasoning systems is critical, as errors can propagate to downstream applications with serious consequences. To improve trustworthiness, we developed a *formally certified counter* that guarantees correctness and applied *causal reasoning* to uncover the inner workings of SAT solvers.

Certified Counting. While existing certification techniques have secured deterministic reasoning engines, such as UNSAT proof checking for SAT solvers, how to certify complex randomized systems like probabilistic approximate counters has long remained unclear. The core challenge lies in the probabilistic guarantee—traditional certification cannot properly handle probabilistic failures. To address this, we established a new paradigm for verifying randomized algorithms, developing a certified approximate counter with formally verified guarantees on output quality. Our approach first proves the algorithm's probabilistic guarantee in the Isabelle/HOL theorem prover and then verifies each execution via proof certification, together producing certified results *modulo randomness*. This pipeline creates a rigorous bridge between algorithm-level correctness and solver-level execution, effectively turning the formalized randomized algorithm into a verified proof checker. The framework includes thousands of lines of formal proof for ApproxMC and customized proof certification for ApproxMC and CNF-XOR solving steps. Experiments show that certificate generation incurs negligible overhead, while our checker, CertCheck, is able to certify 84.7% of instances [2]. This work was recognized with the **Distinguished Paper Award** at CAV 2024. Building on this foundation, we further developed a certified binarized neural network verifier, addressing long-standing concerns about unsound results in neural network verification [4].

Causal Explanation. Over the past three decades, SAT solving has achieved remarkable progress, with modern CDCL-based solvers capable of handling industrial benchmarks containing millions of variables within seconds. Yet, despite this success, their internal behavior remains largely opaque, and they continue to struggle with certain benchmark classes of only hundreds of variables—a sharp contrast to their practical effectiveness. To uncover the principles behind these seemingly weak yet powerful systems, we introduced CausalSAT, a framework that applies *causal reasoning* to reveal interactions among solver components and guide transparent, interpretable solver design [6]. CausalSAT begins by generating observational data from solver executions and then learns a structured graph that captures causal dependencies between components. Given a query—for example, whether clauses with low literal block distance (LBD) yield higher utility—CausalSAT estimates the causal effect of LBD on clause utility and provides a principled answer. Using this framework, we quantitatively verified hypotheses previously treated as "rules of thumb", such as the rapid decline in utility of high-LBD clauses, and explored new questions, including which branching heuristics maximize clause utility. Experiments on practical benchmarks show that CausalSAT effectively models solver behavior, verifies four empirical rules, and answers three previously unexplored questions, providing actionable insights into modern SAT solver design.

Thrust 3: Empowering Impact through Automated Reasoning

We apply automated reasoning techniques to real-world scenarios to demonstrate their practical impact in emerging domains such as quantum computing and machine learning. Specifically, we developed a SAT-based *quantum circuit compiler* to optimize quantum circuit execution on physical devices and designed a *certified neural network verifier* to eliminate unsound verification results.

Quantum Compilation. Near-term quantum computing faces significant challenges, including limited qubit connectivity and noisy quantum operations. To address connectivity constraints, circuit mapping is required to execute quantum circuits on actual hardware. This process involves determining the initial qubit placement and inserting SWAP operations to relocate non-adjacent qubits for nearest-neighbor interaction. Because SWAPs are costly and error-prone, reducing their number is critical for improving the success rate of quantum circuit execution. To minimize SWAPs, we developed a novel circuit mapping method, SATmapper, which combines incremental and parallel SAT solving with an innovative SAT encoding of the mapping problem. This approach not only improves solver-based mapping but also provides a smooth trade-off between compilation quality and time. On a benchmark suite of 78 instances covering three quantum algorithms and two hardware topologies, our method achieves a 26× speedup over state-of-the-art solver-based approaches, reducing compilation time from hours to minutes for important quantum applications used in the Amazon Quantum Cloud Computing Service [3]. This contribution was recognized with the **Best Student Paper Runner-Up Award** at SAT 2024.

Certified Neural Network Verification. Neural networks are increasingly deployed in safety-critical applications, where verification must be both scalable and trustworthy. However, existing approaches either lack soundness guarantees—making them vulnerable to errors—or rely on certification pipelines with poor scalability, limiting practical adoption. To achieve both scalability and trustworthiness, we developed an efficient certified verifier for Binarized Neural Networks (BNNs), where neurons are constrained to Boolean values. Our method introduces a native representation of BNN constraints in an approximate model counter, ApproxMC-BNN, along with specialized proof-generation and checking pipelines that support BNN reasoning natively. This design ensures scalable yet formally certified quantitative verification. On a robustness verification benchmark suite, our certified approach achieves a 218× speedup over CNF-based baselines and produces fully certified results for 86% of queries, compared to only 4% for prior methods [4]. This work establishes the foundation for practical certified neural network verification and was recognized with the **Best Student Paper Runner-Up Award** at SAT 2025.

Future Work

My long-term research goal is to drive the virtuous cycle of automated reasoning between applications and techniques through *algorithm-system co-design* and *reasoning under uncertainty*. By uniting the strengths of theory and practice, I aim to redefine the efficiency frontier of automated reasoning through tighter integration across all layers—from complexity analysis and algorithmic design to system engineering and hardware acceleration. Meantime, I seek to establish the next generation of trusted reasoning systems that operate with negligible overhead, preserve privacy by design, and enable transparent solver architectures with predictable instance behavior. Finally, by embracing reasoning under uncertainty via randomness and approximation, my vision is to create automated reasoning systems that not only capture but also harness the inherent uncertainty of emerging domains such as probabilistic programming, quantum simulation, and large language models—driving the next wave of practical breakthroughs.

Pushing the Frontier of Efficiency. I envision the next frontier of counting systems to drive breakthroughs across four dimensions: revolutionary algorithmic design, unprecedented reasoning capability, innovative system engineering, and hardware-aware acceleration. To revolutionize algorithmic design, the central challenge in approximate counting is to further reduce the number of required SAT calls. The best known upper bound remains $O\left(\log n \cdot \log(1/\delta)/\varepsilon^2\right)$ [CMV16] in terms of the number of variables n, tolerance ε, and confidence δ. Our recent work [5] suggests a theoretical—though currently impractical—construction achieving $O(\log n + \log(1/\varepsilon) + \log(1/\delta))$.

Bridging this theoretical insight with a practical algorithm would mark a paradigm shift in counting efficiency. To expand reasoning capability, I aim to probe the limits of current algorithms via proof complexity. Known exponential lower bounds for exact counting [BW98; Bea+13] reveal fundamental weaknesses even on monotone CNFs—formulas trivial for SAT. Exploring approximation opportunities for such instances, or devising fundamentally stronger algorithms, opens a promising path forward. From a system perspective, enhancing CNF-XOR solving is critical. Our observations show that ApproxMC fails to solve even a simple instance with a single clause and 4,000 variables within one hour—highlighting that enumerating solutions for large-scale XORs remains a major bottleneck. The next generation of systems must overcome this barrier to enable efficient large-scale CNF-XOR reasoning. Finally, on the hardware frontier, while parallel SAT solving has achieved remarkable success, no effective parallel model counters exist today. I envision a similar *parallelization revolution* for model counting, enabled by cube-and-conquer strategies and parallel CNF-XOR solving, to fully harness modern multi-core and distributed architectures.

Shaping the Future of Trust. The future of reasoning systems should enable cost-negligible certification, privacy-preserving reasoning, explainable solver design, and interpretable instance difficulty. First, achieving certified counting without sacrificing efficiency requires completing the certification pipeline—extending certification to preprocessors such as Arjun [SM22] and PB counters like ApproxMC-PB [8] —and evolving it alongside future algorithmic advances [1]; [5]. Second, to support privacy-sensitive applications, we envision privacy-preserving counting algorithms. Building on zero-knowledge solving [Luo+22], we aim to develop secure counting frameworks that protect proprietary techniques and problem instances. Third, we leverage causal reasoning to deepen understanding of solver behavior. Building on our causal analysis of clause management heuristics, we plan to uncover causal connections between solver components and runtime, inspiring new solver architectures. Finally, we apply causal reasoning to interpret instance hardness. Our ongoing work has identified a causal link between the clause-to-variable ratio and runtime in random 3SAT instances, consistent with the phase transition phenomenon [MSL92]. Extending this analysis to industrial benchmarks such as bounded model checking could reveal runtime-critical factors, explain solver success on real-world problems, and guide resource allocation across problem classes.

Driving the Next Cycle of Impact. The recent advances in counting open up new opportunities across combinatorics, probabilistic programming, quantum simulation, and large language models (LLMs). The first direction targets long-standing open combinatorial problems such as the Dedekind number [Jäk23; Van+24] and the Knight's Tour problem [McK97] through model counting. Our ongoing work leverages symmetry reduction and parallel acceleration to compute previously unknown counts for large instances, pushing beyond the current computational limits. The second direction focuses on probabilistic inference—the principal computational bottleneck in probabilistic programs—using weighted model counting. Going beyond existing BDD-based inference [HVM20], we aim to build scalable inference via modern top-down counters and explore approximate inference through randomization. The third direction explores accelerating quantum circuit simulation on classical computers via model counting [MBL24]. Effectively handling negative and complex weights introduced by recent encoding will unlock this potential through approximate counting. The fourth direction seeks to scale BNN verification from feedforward architectures toward LLMs [Zha+23]. Our recent progress has established the foundation for verifying binarized models with millions of neurons, opening the path to reasoning about quantized LLMs [Xia+23] with broader impact.

Referenced Own Publications

[1] Jiong Yang and Kuldeep S. Meel. "Rounding Meets Approximate Model Counting." In: *Proc. of CAV*. 2023.
Distinguished Paper Award.

[2] Yong Kiam Tan*, Jiong Yang*, Mate Soos, Magnus O. Myreen, and Kuldeep S. Meel. "Formally Certified Approximate Model Counting." In: *Proc. of CAV*. * Yong Kiam Tan and Jiong Yang contributed equally. 2024.
Distinguished Paper Award.

[3] Jiong Yang, Yaroslav A. Kharkov, Yunong Shi, Marijn J. H. Heule, and Bruno Dutertre. "Quantum Circuit Mapping Based on Incremental and Parallel SAT Solving." In: *Proc. of SAT*. 2024.
Best Student Paper Runner-Up Award.

[4] Jiong Yang, Yong Kiam Tan, Mate Soos, Magnus O. Myreen, and Kuldeep S. Meel. "Efficient Certified Reasoning for Binarized Neural Networks." In: *Proc. of SAT*. 2025.
Best Student Paper Runner-Up Award.

[5] Yash Pote ⓡ Kuldeep S. Meel ⓡ Jiong Yang. "Towards Real-Time Approximate Counting." In: *Proc. of AAAI*. ⓡ denotes random order. 2025.

[6] Jiong Yang, Arijit Shaw, Teodora Baluta, Mate Soos, and Kuldeep S. Meel. "Explaining SAT Solving Using Causal Reasoning." In: *Proc. of SAT*. 2023.

[7] Jiong Yang, Supratik Chakraborty, and Kuldeep S. Meel. "Projected Model Counting: Beyond Independent Support." In: *Proc. of ATVA*. 2022.

[8] Jiong Yang and Kuldeep S. Meel. "Engineering an Efficient PB-XOR Solver." In: *Proc. of CP*. 2021.

Other References

[Bea+13] Paul Beame, Jerry Li, Sudeepa Roy, and Dan Suciu. "Lower bounds for exact model counting and applications in probabilistic databases." In: *Proc. of UAI*. 2013.

[BW98] Beate Bollig and Ingo Wegener. "A very simple function that requires exponential size read-once branching programs." In: *Information Processing Letters* (1998).

[CMV16] Supratik Chakraborty, Kuldeep S. Meel, and Moshe Y. Vardi. "Algorithmic Improvements in Approximate Counting for Probabilistic Inference: From Linear to Logarithmic SAT Calls." In: *Proc. of IJCAI*. 2016.

[HVM20] Steven Holtzen, Guy Van den Broeck, and Todd Millstein. "Scaling exact inference for discrete probabilistic programs." In: *Proc. of OOPSLA* (2020).

[Jäk23] Christian Jäkel. "A computation of the ninth Dedekind number." In: *Journal of Computational Algebra* (2023).

[Luo+22] Ning Luo, Timos Antonopoulos, William R. Harris, Ruzica Piskac, Eran Tromer, and Xiao Wang. "Proving UNSAT in Zero Knowledge." In: *Proc. of CCS*. 2022.

[McK97] Brendan McKay. "Knight's Tours on an 8 × 8 Chessboard." In: *Technical Report TR-CS-97-03* (1997).

[MBL24] Jingyi Mei, Marcello Bonsangue, and Alfons Laarman. "Simulating Quantum Circuits by Model Counting." In: *Proc. of CAV*. 2024.

[MSL92] David Mitchell, Bart Selman, and Hector Levesque. "Hard and easy distributions of SAT problems." In: *Proc. AAAI*. 1992.

[SM22] Mate Soos and Kuldeep S. Meel. "Arjun: An Efficient Independent Support Computation Technique and its Applications to Counting and Sampling." In: *Proc. of ICCAD*. 2022.

[Van+24] Lennart Van Hirtum, Patrick De Causmaecker, Jens Goemaere, Tobias Kenter, Heinrich Riebler, Michael Lass, and Christian Plessl. "A Computation of the Ninth Dedekind Number Using FPGA Supercomputing." In: *ACM Trans. Reconfigurable Technol. Syst.* (2024).

[Xia+23] Guangxuan Xiao, Ji Lin, Mickaël Seznec, Hao Wu, Julien Demouth, and Song Han. "SmoothQuant: Accurate and Efficient Post-Training Quantization for Large Language Models." In: *Proc. of ICML*. 2023.

[Zha+23] Yichi Zhang, Ankush Garg, Yuan Cao, Lukasz Lew, Behrooz Ghorbani, Zhiru Zhang, and Orhan Firat. "Binarized Neural Machine Translation." In: *Proc. of NeurIPS*. 2023.