+1 (217) 714 6242
shambwaditya.saha@gmail.com
samsaha.com
samsaha3
DBLP, Google Scholar

Shambwaditya "Sam" Saha

Professional Summary

Research Scientist advancing LLM reasoning and AI for code. Specializing in neuro-symbolic AI, integrating symbolic approaches (formal methods, constraint solving, program verification/synthesis) with ML techniques (reinforcement learning, post-training, agentic AI) to design systems capable of automated reasoning and verifiable code generation.

Employment

2023 - 2025 **Research Scientist**, *Software Engineering Institute (SEI)*, Carnegie Mellon University (CMU)

- **Co-authored research proposal** Auto Formal Methods, a multi-agent framework that automates formal verification from natural-language specs by translating requirements into proof obligations, orchestrating verification tasks, and invoking CBMC and KLEE to prove properties or find counterexamples; selected as a finalist for a $500K SEI Line Project grant.
- **Proposed and led research project** on agentic AI for vulnerability detection in >1M LOC codebases, overcoming LLM context limits by enabling dynamic code inspection through Code Property Graph (Joern) queries; achieved $\sim 90\%$ accuracy on NIST SARD benchmarks.
- **Led research project** on LLM-driven C++ code generation for AMD MPSoC, transforming high-level specs into compilable Vitis code using Chain-of-Thought prompting, retrieval-augmented generation over hardware documentation, and compiler-in-the-loop refinement.
- **Co-authored research proposal** on extracting models from code using LLMs; selected as a finalist in an institute-wide competition for a $2M, two-year SEI Line Project grant.

2020 - 2021 **Postdoctoral Researcher**, *Tufts University*, under Prof. Jeff Foster
Synthesizing programs using type inferencing, symbolic solving, and genetic programming.

2020 - 2020 **Research Internship**, *Runtime Verification*, Urbana, Illinois
Verification of Solidity bytecode for smart contracts used in the Ethereum Blockchain.

Education

2012 - 2019 **Integrated PhD, Computer Science**, *University of Illinois at Urbana-Champaign (UIUC)*
Thesis: Learning Frameworks for Program Synthesis. (Advisor: Madhusudan Parthasarathy)
Research Projects:

- **Specification Mining:** Generating precise contracts for C# programs.
- **Precondition Synthesis:** Synthesized preconditions to prevent failures in C# programs.
- **Invariant Synthesis:** Verified GPU and heap programs by synthesizing loop invariants.
- **Program Synthesis:** Developed a synthesis engine that solved SyGuS benchmarks.
- **Network Synthesis:** Automated SDN configuration changes to enforce policy updates.

2009 - 2011 **Master of Science, Computer Science**, *Chennai Mathematical Institute*, India
Thesis: A Survey of Automata and Logics Over Infinite Graphs. (Advisor: K. Narayan Kumar)

Publications

FACCT 2025 **Towards Compositional Assurance of Large Cyber-Physical Systems.** Gabriel A. Moreno, Mark Klein, Shambwaditya Saha, Farzaneh Derakhshan, Limin Jia. *Proceedings of the 1st Workshop on Formal Arguments for Cps Certification, 2025.*

OOPSLA 2021 **Synthesizing Contracts Correct Modulo a Test Generator.*** Angello Astorga, Shambwaditya Saha, Ahmad Dinkins, Felica Wang, P. Madhusudan, and Tao Xie. *ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications, 2021.*

JAR 2020 **A Learning-Based Approach to Synthesizing Invariants for Incomplete Verification Engines.** Daniel Neider, P. Madhusudan, Shambwaditya Saha, Pranav Garg, and Daejun Park. *Journal of Automated Reasoning, Volume 64, 2020.*

PLDI 2019 **Learning Stateful Preconditions Modulo a Test Generator.*** Angello Astorga, P. Madhusudan, Shambwaditya Saha, Shiyu Wang, and Tao Xie. *40th ACM SIGPLAN Conference on Programming Language Design and Implementation, 2019.*

SAS 2019 **Sorcar: Property-Driven Algorithms for Learning Conjunctive Invariants.** Daniel Neider, Shambwaditya Saha, Pranav Garg, and P. Madhusudan. *26th International Static Analysis Symposium, 2019.*

TOCL 2018 **Compositional Synthesis of Piece-Wise Functions by Learning Classifiers.*** Daniel Neider, Shambwaditya Saha, and P. Madhusudan. *ACM Transactions on Computational Logic, Volume 19, 2018*

CSL 2018 **A Decidable Fragment of Second Order Logic With Applications to Synthesis.*** P. Madhusudan, Umang Mathur, Shambwaditya Saha, and Mahesh Viswanathan. *27th EACSL Annual Conference on Computer Science Logic, 2018.*

TACAS 2018 **Invariant Synthesis for Incomplete Verification Engines.** Daniel Neider, Pranav Garg, P. Madhusudan, Shambwaditya Saha, and Daejun Park. *24th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, 2018.*

TACAS 2016 **Synthesizing Piece-Wise Functions by Learning Classifiers.*** Daniel Neider, Shambwaditya Saha, and P. Madhusudan. *22nd International Conference on Tools and Algorithms for the Construction and Analysis of Systems, 2016.*

CAV 2015 **Alchemist: Learning Guarded Affine Functions.*** Shambwaditya Saha, Pranav Garg, and P. Madhusudan. *27th International Conference on Computer Aided Verification, 2015.*

SOSR 2015 **NetGen: Synthesizing Data-plane Configurations for Network Policies.*** Shambwaditya Saha, Santhosh Prabhu, and P. Madhusudan. *1st ACM SIGCOMM Symposium on Software Defined Networking Research, 2015.*

DSSE 2015 **Syntax-Guided Synthesis.*** Rajeev Alur, Rastislav Bodik, Eric Dallal, Dana Fisman, Pranav Garg, Garvit Juniwal, Hadas Kress-Gazit, P. Madhusudan, Milo M. K. Martin, Mukund Raghothaman, Shambwaditya Saha, Sanjit A. Seshia, Rishabh Singh, Armando Solar-Lezama, Emina Torlak, and Abhishek Udupa. *Dependable Software Systems Engineering, NATO Science for Peace and Security Series, D: Information, Communication Security, 2015.*

* First Author or Equal Contribution

Academic Service

Reviewer **Use of LLMs for Program Analysis and Generation Track**, *Hawaii International Conference on System Sciences*, 2024.

Teaching

CS477 **Formal Software Development Methods**, *University of Illinois at Urbana Champaign*

Mentored students on mathematical models and methods for software specification and verification using formal logics, deductive verification, abstract interpretation, and testing.

CS126 **Software Design Studio**, *University of Illinois at Urbana Champaign*

Coached 20 students to improve their coding style by reviewing their coding assignments and giving them critical feedback on design, documentation, testing, and debugging.

CS498 **Data Visualization**, *Part of Data Mining Specialization at Coursera*, offered by UIUC

Mentored students on the concepts and tools and helped them with their final projects.

Talks

AutoFM: Automated Formal Methods, *Software Engineering Institute*, 2025

Extracting Models From Code, *Software Engineering Institute*, 2023

Specification Mining, *Software Engineering Institute*, 2023

Program Synthesis, *MathWorks*, 2020

Program Synthesis, *GE Research*, 2020

Network Synthesis, *Excape PI Meeting*, 2017

Network Synthesis, *Midwest Verification Week*, 2015

Alchemist: Learning Guarded Affine Functions, *CAV*, 2015

NetGen: Synthesizing Data-plane Configurations for Network Policies, *SOSR*, 2015

NetGen: Synthesizing Data-plane Configurations for Network Policies, *Excape PI Meeting*, 2015