Siddharth Bhat

Education

PhD **University of Cambridge**.
(2024 - Ongoing)

PhD **University of Edinburgh (*moved to Cambridge*)**.
(2022 - 2024)

Master by Research **International Institute of Information Technology Hyderabad India**.
(2020 - 2021)

Undergraduate **International Institute of Information Technology Hyderabad India**.
2015 - 2020

Publications

Verifying Wu's Method can Boost Symbolic AI to Rival Silver Medalists and AlphaGeometry to Outperform Gold Medalists at IMO Geometry: Shiven Sinha, Ameya Prabhu, Ponnurangam Kumaraguru, **Siddharth Bhat**, Matthias Bethge. NeurIPS 2024 Workshop MATH-AI

Verifying Peephole Rewriting in SSA Compiler IRs: **Siddharth Bhat**, Alex Keizer, Chris Hughes, Andres Goens, Tobias Grosser. ITP 2024

Towards Neural Synthesis for SMT-Assisted Proof-Oriented Programming: Saikat Chakraborty, Gabriel Ebner, **Siddharth Bhat**, Sarah Fakhoury, Sakina Fatima, Shuvendu Lahiri, Nikhil Swamy. ICSE 2024

Rewriting Optimization Problems into Disciplined Convex Programming Form: Ramon Fernandez Mir, **Siddharth Bhat**, Andres Goens, Tobias Grosser. CICM 2024

Guided Equality Saturation: Thomas Koehler, Andres Goens, **Siddharth Bhat**, Tobias Grosser, Phil Trinder, Michel Steuwer. POPL 2024

Lambda the Ultimate SSA: **Siddharth Bhat**, Tobias Grosser. CGO 2022

QSSA: An SSA based IR for Quantum Computing: Anurudh Peduri, **Siddharth Bhat**, Tobias Grosser. CC 2021

Optimizing Geometric Multigrid Computation using a DSL Approach: Vinay Vasista, Kumudha KN, **Siddharth Bhat**, Uday Bondhugula. Supercomputing (SC), Nov 2017

Word Embeddings as Tuples of Feature Probabilities: **Siddharth Bhat**, Alok Debnath, Souvik Banerjee, Manish Shrivastava Representation Learning for NLP, 2020

Internship Experience

Sep-Nov '24 **Amazon Web Services, Automated Reasoning Group**, *Austin*.
Deciding memory (non)interference in `lnsym`, a Lean-based ARM symbolic simulator

Jul-Sep '23 **Microsoft Research**, *Redmond*.
Retrieval Augmented theorem proving for the Fstar proof assistant.

July 1-10 ’23 **Adjoint School**, *Glasgow*.
Researched Markov categories and their relationship to probabilistic programming.

Winter 2019 **Teaching Assistant for Natural Language: Applications**, *IIIT-H*.
Monitored projects, took sessions on word embeddings, involving `word2vec`, `GloVe`, `fasttext`.

May-Jul ’19 **Intern at Tweag.io**, *Paris, France*.
Re-implemented portions of GHC(Glasgow Haskell Compiler) runtime for `Asterius` (link), a Haskell to WebAssembly compiler. Involved Haskell, C, and WebAssembly.

Winter 2018 **Teaching Assistant for Principles of Programming Languages**, *IIIT-H*.
Course covers the book "Essentials of Programming Languages" by Dan Friedman. Helped write lecture notes, set assignments, graded assignments and exams.

Summer 2018 **Visiting research intern at ETH Zurich**, *Zurich, Switzerland*.
Investigating formal verification of polyhedral compilation. `PolyIR (Link)` is a formal specification of polyhedral programs.

Summer 2018 **GSoC mentor, Polly Labs**.
Mentoring a project to enable Polly’s loop optimisations into Chapel.

Mar-Dec ’17 **ETH Zurich, Research Intern at SPCL**, *Zurich, Switzerland*.
Worked on Polly, a polyhedral loop optimizer for LLVM.

Jan-Mar ’17 **Course content contributor**, *IIIT-H*.
Wrote lecture notes for the Intro to programming course (link)

May-Jul ’16 **Research Intern**, *IISC Bangalore*, Bangalore.
Worked on PolyMage, DSL compiler for optimising loop transforms. Contributed to ISL and PLUTO. Implemented tiling patterns, optimised PolyMage for stencils.

Summer 2016 **Selected for GSoC 2016**, *Google*.
Binding SymEngine, a symbolic math library to Haskell. Had to drop this to intern at IISc, Bangalore. Still maintain the library (symengine.hs)

Summer 2015 **GSoC 2015**, *Google*.
Worked on VisPy, a pure Python graphics library which uses OpenGL internally for performace. Successfully completed.

Open Source Contributions

Coq Submitted issues, bug-fixes, helped improve developer documentation.

VE-LLVM Collaboration with VE-LLVM, a formal semantics of the LLVM compiler toolchain in Coq

Polly Implementing support for Fortran, added unified memory abilities to the CUDA backend within Polly, a polyhedral loop optimiser for LLVM. (Link to commits)

Symengine.hs GSoC 2016. Haskell bindings to SymEngine, a C++ symbolic manipulation library.

VisPy GSoC 2015. Rewrote scene graph for performance. Added visuals, high level API for easy use of plotting. Implemented auto-resizing with **Cassowary**, a linear optimisation library.

Rust Contributed to the Rust compiler and ecosystem. Found compiler errors, fixed libraries. Was part of *Piston*, group of Rust programmers that experimented with writing game engines.

Haskell Contributed to the Haskell ecosystem. Reported and fixed bugs in *stack*, *stackage*, *diagrams*, *GHC*, etc. (Link to GHC commits).

PLUTO Source to Source C optimiser for loop nests. Improved the PLUTO API that had gone out of sync with master. Discovered bugs in PLUTO for diamond tiling transforms

PolyMage DSL Compiler than generates C code. Uses **Polyhedral Compilation** Extended the compiler to add stencils, time iterated-stencils.

PPSSPP PPSSP is a C++ open source PSP emulator. Wrote most of the touch handling code. Implemented atomic locks for audio performance.

My Projects

Lean-MLIR Formal semantics for the MLIR compiler framework, defined within the Lean4 proof assistant.

lz An MLIR based compiler backend for the Lean4 proof assistant.

Lean4 Metaprogramming Book A textbook on metaprogramming in Lean4. I wrote the chapters on tactics and metaprogramming for embedded DSLs.

Lean-to A Jupyter kernel for the Lean4 proof assistant.

Simplexhc A custom compiler for a subset of Haskell. The goal is to try and apply *polyhedral compilation* ideas to compile a lazy, pure, functional programming language. with LLVM as a backend. Has **64 stars** on github.

Sublime Bookmarks A plugin for sublime text to quickly jump between pieces of your codebase. **26k downloads** and counting.

Cellular Automata A collection of Cellular Automata written in Haskell. Uses **Comonads** for abstraction. **130 stars** on Github.

Teleport A simple tool to switch between projects written in Haskell. Shows how to write “real world Haskell". Published as a **Literal Haskell tutorial**. **90 stars** on github

TIMi A visual interpreter of the **template instantiation machine** to understand evaluation of lazy functional languages. **51** stars on github.

Miscellaneous

Barvinok Talk at ETH Zurich: Slides describing the barvinok algorithm to count lattice points in polyhedra

FunctionalConf '19 Talk on implementing embedded probabilistic programming languages in Haskell (Slides)

Haskell Exchange 2020 Talk on optimizing `smallpt-hs` (a port of a raytracer to haskell) to beat `C++` performance (Slides)

FPIndia Talk on egg: fast and extensible equality saturation. (Slides)

Theory seminar, winter '19 Talk on impossibility of compass-straightedge constructions using field theory.

math.se Answer on **math.stackexchange. 8312 reputation, top 4% overall**. Abstract algebra and differential/algebraic geometry.