Pritam Gharat
(She/Her)
Senior Researcher
Microsoft Research, India

Email pritamgharat@microsoft.com
pritam01gharat@gmail.com
LinkedIn linkedin.com/in/Pritam-Gharat
Mobile 9867383270
Web https://pritammg.github.io/

Research Interests

Compilers, Program Analysis, Programming languages, Software Engineering.

Objective

Apply a strong foundation in traditional and AI-driven program analyses to advance compiler technology. Eager to contribute to product-grade compiler development, leveraging deep analysis expertise to improve performance and reliability while expanding knowledge of modern compiler architectures.

Experience

- **Senior Researcher** in Microsoft Research, India *(Mar '24 - Present)*
- **Post-Doctoral Researcher** in Microsoft Research, India *(Sep '21 - Feb '24)*
- **Research Associate** in Department of Computing, Imperial College, London *(Aug '18 - Jul '21)*
- **Invited Speaker** at Summer School for Pointer Analysis, IIT Bombay *(Jul '22)*
- **Teaching Assistant** in Department of Computing, Imperial College *(Oct '19 - Jan '20)*
- **Teaching Assistant** in CSE Department, IIT Bombay *(Aug '11 - Aug '18)*

Education

- **Ph.D.** in Computer Science & Engineering, IIT Bombay *(Jun '13 - Jul '18)*
  Thesis Title: *Generalized Points-to Graph: A New Abstraction of Memory in Presence of Pointers*
  Advisor: Prof. Uday P. Khedker
- **M.Tech.** in Computer Science & Engineering, IIT Bombay *(Jul '11 - Jun '13)*
- **B.E.** in Computer Science & Engineering, Mumbai University *(Jun '06 - May '10)*

Publications

- *Lightweight and modular resource leak checking (extended version)*. Narges Shadab, Pritam M. Gharat, Shrey Tiwari, Michael Ernst, Martin Kellogg, Akash Lal, Shuvendu Lahiri, Manu Sridharan. International Journal on Software Tools for Technology Transfer (STTT) 2025.
- *Inference of Resource Management Specifications*. Narges Shadab, Pritam M. Gharat, Shrey Tiwari, Michael Ernst, Martin Kellogg, Akash Lal, Shuvendu Lahiri, Manu Sridharan. Object Oriented Programming Systems Language and Applications (OOPSLA) 2023.
- *Combining Static Analysis Error Traces with Dynamic Symbolic Execution (Experience Paper)*. Frank Busse, Pritam M. Gharat, Cristian Cadar, Alastair Donaldson. ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA) 2022.
- *Generalized Points-to Graph: A New Abstraction of Memory in Presence of Pointers*. Pritam M. Gharat, Uday P. Khedker, Alan Mycroft. ACM Transactions on Programming Languages and Systems (TOPLAS) 2020.
- *Flow- and Context-Sensitive Points-to Analysis using Generalized Points-to Graphs*. Pritam M. Gharat, Uday P. Khedker, Alan Mycroft. 23rd Static Analysis Symposium (SAS) 2016.
- *CoS-SSA: SSA for Context-Sensitive Interprocedural Analysis*. Supriya Bhide, Uday P. Khedker, Pritam M. Gharat – Under preparation.
- *Excursions in Pointer Analysis*. Uday P. Khedker, Pritam M. Gharat – Book under preparation.

Projects

- **Rustler: Automating Translation of Legacy Code to Rust Using LLMs**
  (Ongoing)
  - Rustler supports automated workflows for translating C/C# code to Rust. These workflows batch files, extract and cluster patterns, and generate actionable insights using YAML-based configurations.
  - Combining offline summarization with natural language querying helps users understand complex codebases and identify reusable components that is useful for translation.
- **Inference of Resource Management Specifications**
  (In Collaboration With Narges Shadab (UCR), Manu Sridharan (UCR), Michael Ernst (UW), Martin Kellogg (NJIT), Shrey Tiwari (MSR), Akash Lal (MSR), Shuvendu Lahiri (MSR) (Sep. '21 - Oct. '23))
  - Developed an annotation-based modular resource leak checker in CodeQL for C# code.
  - Designed a fixed-point algorithm for automatic inference of annotations for the analysis.
  - This work was published in OOPSLA 2023.
- **Combining Static Analysis Error Traces with Dynamic Symbolic Execution**
  (In Collaboration With Frank Busse, Cristian Cadar, Alistair Donaldson at Imperial College (May '19 - Jul '21))
  - The goal of this work was efficient detection of bugs by combining static analysis and dynamic symbolic execution.
  - Worked on Clang Static Analyzer and Infer (static analyzers) and KLEE (dynamic symbolic execution tool).
  - Customized the search strategy of KLEE to explore only paths that agree with the static information, effectively detecting the bugs faster.
  - This work was published in ISSTA 2022.
- **Generalized Points-to Graph: A New Abstraction of Memory in Presence of Pointers**
  (Ph.D. Thesis under the guidance of Prof. Uday Khedker (IITB) (Jun '13 - Jul '18))
  - Designed and developed a summary-based flow- and context-sensitive pointer analysis in GCC.
  - Introduced the concept of Generalized Points-to Graph (GPG) as a new representation of a compact and yet precise procedure summary for scalable pointer analysis.
  - The implementation for GPG-based pointer analysis in GCC scaled to 158kLoC for C programs.
  - This work was published in SAS 2016 and TOPLAS 2020.
- **CoS-SSA: Context-Sensitive Interprocedural SSA**
  (In Collaboration with Supriya Bhide (IITB), Prof. Uday Khedker (IITB) (Ongoing))
  - Proposing an interprocedural SSA form, called the *context-sensitive SSA* (aka CoS-SSA), that overcomes the limitations of traditional intraprocedural SSA by constructing SSA variables for scalars and pointers that may be global or address-taken local.
  - The way traditional SSA brings in flow sensitivity for free for program analysis, CoS-SSA brings both flow and context sensitivity for free for program analysis.
- **Improving Interprocedural Analysis**
  (M.Tech + Ph.D. Dual Degree Research Proposal, guided by Prof. Uday Khedker (IITB), (Oct '12 - Dec '12))
  - Reforming Value Based Call Strings Method by eliminating the re-processing of flow functions and improving the efficiency.
  - Proposed a variant to k-CFA called as Var-k-CFA for higher order languages by building an analogy between Var-k-CFA and Value Based Call Strings Method.
- **Static Analysis of Object Oriented Languages**
  (M.Tech. Seminar, guided by Prof. Uday Khedker, Jan '12 - May '12)
  - A study of several static analyses designed for Object Oriented Languages that include Class Hierarchy Analysis, Pointer Analysis, Escape Analysis, Type Analysis.

- **Discovering use of Pointer Information in GCC** *(GCC Project, Spring '11)*
    - Points-to information generated by Pointer Analysis is used by other static analyses to perform further optimizations for better precision. The goal of the project was to discover how the points-to information is used by other optimizations in GCC.
- **Implementation of GVN PRE**
  (Course Project in Advanced Compilers, Autumn '12)
    - Implemented GVN PRE in Jikes RVM. Studied dynamic compilation and adaptive optimization system in Jikes RVM.

Honours and Distinctions

- TCS Research Fellowship *(Jul '13 - Jul '18)*.
- Sir Ratan Tata Trust Merit scholarship for two consecutive years - *2008* and *2009* for excellence in academics.
- Best Student Award from Tata Consultancy Services *(2010)*.
- Ranked 82 in Gate 2011.

Service

- APLAS 2025 (General Chair)
- ISEC 2024 (Program committee)
- CAV 2023 (Artifact evaluation)
- ISSTA 2021 (Artifact evaluation)
- SAS 2020 (Artifact evaluation)
- ISEC 2025 (Program committee)
- TACAS 2024 (Artifact evaluation)
- CGO 2021 (Artifact evaluation)
- PLDI 2020 (External review committee)

References

Will be provided when required.