

Rohan Bavishi

SENIOR UNDERGRADUATE, COMPUTER SCIENCE, IIT KANPUR

C-324, Hall-1
IIT Kanpur
Kanpur, Uttar Pradesh, India
rbavishi@iitk.ac.in | rohan.bavishi95@gmail.com
Webpage : home.iitk.ac.in/~rbavishi
Github : github.com/rbavishi
+91-73-180-18920

EDUCATION **Indian Institute of Technology Kanpur**, Uttar Pradesh, India
Bachelor of Technology, Computer Science and Engineering, *Jul' 13 - Jul' 17 (Expected)*
GPA: 9.7/10 (Overall)

RESEARCH INTERESTS Program Analysis and Verification, Automated Debugging and Synthesis,
Compiler Optimizations, Decision Procedures

PUBLICATIONS **Rohan Bavishi**, Awanish Pandey, Subhajit Roy, "To Be Precise : Regression Aware Debugging" to appear in *Proceedings of the 2016 ACM International Conference on Object Oriented Programming Systems Languages & Applications (OOPSLA)*, Amsterdam, Netherlands

Rohan Bavishi, Awanish Pandey, Subhajit Roy, "Regression Aware Debugging for Mobile Applications" to appear in *Proceedings of the 1st International Workshop on Mobile Development (Mobile! 2016, part of SPLASH 2016)*, Amsterdam, Netherlands

AWARDS & ACHIEVEMENTS Awarded the **SIGPLAN PAC Scholarship** for paper presentation at **OOPSLA '16 Academic Excellence Award 2013-14**, IIT Kanpur
Secured an **All-India-Rank of 202** in JEE Advanced 2013 amongst 150,000 candidates
Secured an **All-India-Rank of 175** in JEE Mains 2013 amongst 20,00,000 candidates
Secured an **All-India-Rank of 33** in AMTI-Mathematics Olympiad

RESEARCH PROJECTS **New Strategy for Analysis of Concurrent Programs via Sequentialization**
Supervisor : Prof. Subhajit Roy *Aug '16 - Present*

- Using **CSeq** for code-to-code translation of concurrent programs into equivalent sequential ones
- Devising solving strategies to reduce verification time on existing backends like CBMC

Improving Bug Localization Using Interpolant-Based Proofs
Supervisor : Prof. Subhajit Roy *Jul '15 - Aug '16*

- Devised a new method to improve the quality of bug localizations, in terms of reduced superfluous program locations, for a given set of passing and failing test-cases
- Interpolants are constructed from passing tests to derive *soft* roadblocks. These roadblocks then discourage localizations violating these interpolants, thereby improving bug localization quality
- Upto 45% improvement in localization quality as compared to the state-of-the-art tool BugAssist.
- *Paper accepted in OOPSLA, one of the premier peer-reviewed conferences in Programming Languages*

Using SAT/QBF-Solvers to Detect Side-Channel Vulnerabilities in Hardware
Supervisors : Prof. Paolo Ienne and Mr. Andrew Becker *May '16 - Present*

- Summer internship project at the Processor Architecture Laboratory, EPFL, Switzerland
- Studied various side-channel attacks, mitigation techniques and their proofs of effectiveness using formal methods
- Developed a QBF-Encoding technique to verify whether a cryptographic circuit is secure against a popular side-channel attack based on fault-injection
- In the process of writing a paper and submitting to a peer-reviewed conference

Implementation of DirectFix in CBMC

Supervisor : Prof. Subhajit Roy

May '15 - Jul '15

- Ported the described *Component-Based-Synthesis* algorithm in [DirectFix](#) to CBMC
- Reproduced the experimental results provided in the paper, and devised further optimizations
- [Github Link](#)

ACADEMIC PROJECTS

Re-Inventing A Median Algorithm for Disk-Resident Data

Supervisor : Prof. Surender Baswana

Aug '14 - Nov '14

- Re-invented a two-pass *deterministic* algorithm to find the median of large data-sets (approx. 1 TB)
- The algorithm developed was similar to the one described in the [paper](#) by Munro-Paterson (1980)
- Carried out extensive tests to evaluate the performance of the algorithm
- [Report](#)

Peer-to-Peer Dropbox

Supervisor : Prof. Subhajit Roy

Aug '13 - Nov '13

- A linux application for backing-up and syncing files between two or more peers
- Users have a shared folder across different machines, with local copies. Changes made in any one copy are synced across all devices
- Linux *inotify* API used to track changes in the shared folder and *rsync* used to sync the modifications to ensure efficient transfer
- Multithreading with mutexes used to parallelize syncing and file-monitoring operations
- [Github Link](#)

COURSE PROJECTS

End-to-End Compiler for Perl-like Language

Course : Compilers | Supervisor : Prof. Subhajit Roy

Jan '15 - Apr '15

- Built an end-to-end compiler that takes a subset of the Perl language and outputs MIPS assembly
- Implemented Features such as *operator overloading*, *dynamic type-checking*, *variable function arguments*, *hashes*, *lists*, *type-based namespaces* etc.
- [Github Link](#)

Integer Superscalar Processor Simulator based on MIPS-R10K

Course : Computer Architecture | Supervisor : Prof. Mainak Chaudhuri

Jan '15 - Apr '15

- Implemented a superscalar processor simulator based on the MIPS R10K architecture with support for integer instructions only
- Implemented support for Out-of-Order execution, multiple instruction issue and commit, precise interrupts and branch-misprediction rollback
- Configurable parameters supported such as issue width, number of functional units, branch-prediction algorithm, RAS/BTB size, active-list size etc.
- Performed experiments on various synthesized benchmarks to measure performance gains over an in-order processor design. Repeated the experiment with varying parameter configurations
- [Report](#)

COMPUTER SKILLS

Languages: C, C++, Python, Bash, Verilog, L^AT_EX, Assembly (x86, MIPS)
SAT/SMT Solvers: MathSAT, Z3, Yices
Research Tools: CBMC (Proficient), KLEE, CSeq

EXTRA INTERESTS

Project Euler: Solved : 257/560 (*India Rank : 11*)
Hobbies: Competitive Programming, CTF & Wargames, Quizzing