

---

## Computer Skills

---

- Operating Systems : Unix/Linux, Windows, Mac-OsX
- Languages : C/C++, Python, Java, Ocaml, Lex, Yacc, Perl, C#, Verilog, Matlab

---

## Education

---

- **University of California, Santa Cruz, CA, PhD student in Computer Science**  
*2004- present*
  - Topic : Solving verification problems of large systems using scalable verification techniques
- **Indian Institute of Technology, Kharagpur, India**  
*2002- 04 M.S. Computer Science and Engineering*
  - Dissertation Topic : Development of Open-CTL Module Verifier and Stimulus Generation Tool
- **Indian Institute of Technology, Kharagpur, India**  
*1998- 2002 B.Tech (Hons.) Computer Science and Engineering*
  - Dissertation Topic : Mobile Hand-off and Channel Allocation Algorithm

---

## Experience

---

- **Graduate Student Researcher, Design & Verification Lab, UCSC**  
*2004 - present*
  - Project : Scalable verification techniques for large systems
  - Summary : Developed and implemented efficient, scalable algorithms to verify/test software/hardware designs
  - Implemented symbolic algorithms for the tool TICC: A Tool for Interface Compatibility and Composition.
- **Summer Intern, SRI International, Menlo Park, USA**  
*July 2009-Sept 2009*
  - Project : Typechecking an Expressive Type System for Simulink using YICES SMT Solver
  - Summary : Developed a tool that typechecks (and thus verifies) a Simulink design. The types are very expressive such as user given types, measurement units as types, dependent types, refinement types etc. The tool is written in the MATLAB language and is integrated with the Simulink environment. We also provide a translator from Simulink design to YICES SMT solver input language. I have also added Bounded Model Checking and k-induction into the tool. The tool can also generate test cases where the type checker fails. We can also simulate the failed scenario and dynamically monitor the design.
- **Summer Intern, NEC Research Lab, Princeton, USA**  
*June 2007-Sept 2007*
  - Project : Modular Verification of Software by Creating Interfaces
  - Summary : Developed and implemented algorithms to verify library functions by summarizing every function and creating an interface ( function call sequence graph)
- **Research Consultant, MSR Research, Redmond, USA**  
*May 2007*

---

<sup>1</sup>Updated on September 28, 2009

- Project : Probabilistic Reinforcement Learning in Software Testing
- Summary : Developed and implemented probabilistic learning algorithms to maximize the software testing coverage in online testing in NModel open source tool
- **Summer Intern, MSR Research, Redmond, USA**  
*June 2005-Sept 2005*
  - Project : Reinforcement Learning in Software Testing
  - Summary : Developed algorithms to maximize the software testing coverage in online testing in XRT, and SpecExplorer.
- **Graduate Research Assistant, Formal Verification Group, IIT Kharagpur, India**  
*Aug 2002- Aug 2004*
  - Project : Model Abstraction and Formal Verification (funded by SUN Microsystems, USA)
  - Summary : Worked on modeling and verifying systems specified in Open-RTCTL Developed a BDD-based Open-RTCTL model checker BLIF format front-end. Implemented BLIF parser in YACC and stored transition relation symbolically in a BDD.
- **Summer Intern, Dortmund University, Germany**  
*May 2001- July 2001*
  - Project : Compiler Optimization Tool for LANCE Embedded Processor
  - Summary : Worked on optimization of the LANCE compiler project. Developed and tested one of the code optimization techniques for embedded processors called Global Common Sub expression Elimination.

## Publications

---

- P. Roy, D. Parker, G. Norman, L. deAlfaro. Symbolic Magnifying Lens Abstraction in Markov Decision Processes. QEST08.
- Johannes Helander, Risto Serg, Margus Veanes and Pritam Roy. Adapting Futures: Scalability for Real-World Computing. RTSS07.
- L. deAlfaro, P.Roy. Solving Games via Three-Valued Abstraction Refinement. CONCUR07
- L. de Alfaro, P. Roy. Magnifying-Lens Abstraction for Markov Decision Processes. CAV07.
- Margus Veanes, Pritam Roy and Colin Campbell, Online Testing with Reinforcement Learning, in FATES/RV 06.
- B. Thomas Adler, L. de Alfaro, L. Dias Da Silva, M. Faella, A. Legay, V. Raman, P. Roy. Ticc: A Tool for Interface Compatibility and Com- position. in CAV06.
- Wolfgang Grieskamp, Nicolas Kicillof, Colin Campbell, Pritam Roy, Wolfram Schulte, Nikolai Tillman, Margus Veanes , Behavioral Composition in Symbolic Domains, IWAOM05.
- L.de Alfaro, L. Dias Da Silva, M. Faella, A. Legay, P. Roy, M. Sorea. Sociable Interfaces. FROCCOS05.
- Roy, P., Dasgupta, P., Chakrabarti, P.P., An Assertion Based Language for Generating Test Sequences for Complex Temporal Behavior, VDAT04.

## Research Interests

---

- Formal methods (mostly model checking) for hardware/software verification and testing
- Scalable verification techniques as abstraction-refinement, BDD / SMT-based symbolic algorithm
- Application of Game theory to hardware and software systems
- Solving practical verification problems using two player games and markov decision processes
- Interface theory for component based design, and verification