

Patrick Meredith

Department of Computer Science
201 N. Goodwin
University of Illinois
Urbana, IL 61801

Phone: (217) 418-0418
pmeredit@illinois.edu
pmeredit@gmail.com

Research

I am a Post Doctorate Researcher in the Formal Systems Laboratory (fsl.cs.uiuc.edu) at the University of Illinois in Urbana-Champaign, where I was also previously a Graduate Research Assistant. I was also a member the LLVM group (llvm.org) at the University of Illinois as an M.S. student and undergraduate.

Current Research

I am interested primarily in the fields of runtime monitoring, programming languages, and compilers, with a secondary focus on computer architecture. My main research goals are providing efficient monitoring algorithms for different problem domains (e.g., user space software, embedded systems), making predictive runtime analysis feasible for large scale systems, and providing a new efficient rewriting system for the K-language framework.

Past Research

My work with the FSL group as a Ph.D. student can be divided into two main parts: runtime verification and language semantics. My work on a formal executable semantics of Verilog can be found at Verilog Semantics. My work on runtime monitoring of COTS components can be found at BusMOP, and my work with JavaMOP can be found at JavaMOP. The current version of K-Scheme (my M.S. thesis work), including an online interface, can be found at K-Scheme.

Publications

- [1] Choonghwan Lee, Dongyun Jin, Patrick O’Neil Meredith, and Grigore Roşu. Towards categorizing and formalizing the JDK API. Technical Report <http://hdl.handle.net/2142/30006>, Department of Computer Science, University of Illinois at Urbana-Champaign, March 2012.
- [2] Patrick O’Neil Meredith and Grigore Roşu. Efficient parametric runtime verification with deterministic string rewriting. Technical Report <http://www.ideals.illinois.edu/handle/2142/30467>, Department of Computer Science, University of Illinois at Urbana-Champaign, March 2012. submitted to FSE 2012.
- [3] Soha Hussein, Patrick O’Neil Meredith, and Grigore Roşu. Security-policy monitoring and enforcement with javamop. In *Proceedings of the 7th Worksop on Programming Languages and Analysis for Security (PLAS’12)*. ACM, 2012. to appear.
- [4] Dongyun Jin, Patrick O’Neil Meredith, Choonghwan Lee, and Grigore Roşu. Javamop: Efficient parametric runtime monitoring framework. In *Proceeding of the 34th International Conference on Software Engineering (ICSE’12)*. IEEE, 2012. to appear (demo paper).

- [5] Patrick O’Neil Meredith, Dongyun Jin, Dennis Griffith, Feng Chen, and Grigore Roşu. An overview of the MOP runtime verification framework. *Journal on Software Techniques for Technology Transfer (J. of STTT)*, April 2011.
- [6] Dongyun Jin, Patrick O’Neil Meredith, Dennis Griffith, and Grigore Roşu. Garbage collection for monitoring parametric properties. In *Programming Language Design and Implementation (PLDI’11)*, pages 415–424. ACM, 2011.
- [7] Patrick O’Neil Meredith, Dongyun Jin, Feng Chen, and Grigore Roşu. Efficient monitoring of parametric context-free patterns. *Journal of Automated Software Engineering*, 17(2):149–180, June 2010.
- [8] Patrick O’Neil Meredith, Michael Katelman, José Meseguer, and Grigore Roşu. A formal executable semantics of Verilog. In *Formal Methods and Models for Codesign (MEMOCODE’10)*, pages 179–188. IEEE, 2010.
- [9] Patrick O’Neil Meredith and Grigore Roşu. Runtime verification with the RV system. In *Runtime Verification (RV’10)*, volume 6418 of *Lecture Notes in Computer Science*, pages 136–152. Springer, 2010.
- [10] Feng Chen, Patrick O’Neil Meredith, Dongyun Jin, and Grigore Rosu. Efficient formalism-independent monitoring of parametric properties. In *Automated Software Engineering (ASE’09)*, pages 383–394, 2009.
- [11] Rodolfo Pellizzoni, Patrick O’Neil Meredith, Min-Young Nam, Mu Sun, Marco Caccamo, and Lui Sha. Handling mixed-criticality in soc-based real-time embedded systems. In *Embedded Software (Emsoft’09)*, pages 235–244, 2009.
- [12] Patrick O’Neil Meredith, Dongyun Jin, Feng Chen, and Grigore Roşu. Efficient monitoring of parametric context-free patterns. In *Automated Software Engineering(ASE ’08)*, pages 148–157. IEEE/ACM, 2008. **ACM Sigsoft Distinguished Paper Award.**
- [13] Rodolfo Pellizzoni, Patrick O’Neil Meredith, Marco Caccamo, and Grigore Rosu. Hardware runtime monitoring for dependable cots-based real-time embedded systems. In *Real-Time System Symposium (RTSS’08)*, pages 481–491, 2008.
- [14] Grigore Roşu Patrick O’Neil Meredith, Mark Hills. An executable rewriting logic semantics of K-Scheme. In Danny Dube, editor, *Workshop on Scheme and Functional Programming (SCHEME’07)*, pages 91–103. Laval University, 2007.
- [15] Patrick O’Neil Meredith, Balpreet Pankaj, Swarup Sahoo, Chris Lattner, and Vikram Adve. How successful is data structure analysis in isolating and analyzing linked data structures? Technical Report Department of Computer Science UIUCDCS-R-2005-2658, University of Illinois at Urbana-Champaign, 2005.

Relevant Employment Experience

I am co-founder of Runtime Verification, Inc. We work to commercialize the research performed by the Formal Systems Laboratory. Our website can be found at [Runtime Verification](#). During

the first half of 2010 we had an SBIR grant under NASA where we developed a predictive analysis system called RVpredict. I was responsible for leading a team of two programmers and doing much of the programming myself for the system. RVpredict uses many techniques to handle the huge¹ amounts of data that it must stream through in order to predict errors in real world programs.

Teaching Experience

- **CS522: Programming Language Semantics [Fall 2009]**, guest lecturer. Gave a lecture on Verilok (K definition of Verilog).
- **CS422: Programming Language Design [Fall 2007]**, guest lecturer. Gave a lecture on K-Scheme.
- **CS105: Computer Science for Non-technical Majors [Spring 2007]**, teaching assistant, webmaster, database master. Tasks included teaching weekly laboratory sections, administering the website and the database used in class.
- **CS105: Computer Science for Non-technical Majors [Fall 2006]**, teaching assistant. Tasks included teaching weekly laboratory sections.

Presentations

- **Efficient Monitoring of Parametric Context-Free Patterns**
L'Aquila Italy, September 2008 (ASE'08)
- **MOP Tutorial**
Auckland New Zealand, November 2009 (ASE'09)
- **Efficient Formalism-Independent Monitoring of Parametric Properties**
Auckland New Zealand, November 2009 (ASE'09)
- **A Formal Executable Semantics of Verilog (MEMOCODE'10)**
Grenoble France, July 2010
- **Runtime Verification with the RV System Tutorial (RV'10)**
Malta, November 2010

Honors and Awards

- Siebel Fellowship, 2008
- Siebel Scholar, 2005
- Tau Beta Pi, 2004
- Alpha Lambda Delta, 2002

¹Traces of 30+ GB compressed have been successfully predicted.

- Phi Eta Sigma, 2002
- Phi Kappa Phi, 2002
- Deans List, 2002-2003²
- Eagle Scout, 1999

Education

Department of Computer Science, University of Illinois

Ph.D. Computer Science, University of Illinois, 2012, GPA: 4.0

M.S. Computer Science, University of Illinois, 2007, GPA: 3.98

B.S. Computer Science, University of Illinois, 2007, *Highest Honors*, GPA: 3.88

Relevant Coursework

- ECE598sv: Formal Hardware/SoC Verification (audit)
- CS522: Programming Language Semantics (audit)
- CS533: Parallel Computer Architecture
- CS598mc: Real Time Systems
- CS523: Advanced Operating Systems
- ECE598bl: System on a Chip Design
- CS422: Programming Language Design
- ECE512: Computer Micro-architecture
- CS526: Advanced Topics in Compiler Construction
- ECE511: Computer Architecture
- CS598cz: Virtual Machines and Run-time Optimization
- CS418: Computer Graphics
- LING406: Computation Linguistics
- CS475: Formal Models of Computation
- CS433: Computer System Organization
- CS326(426): Compiler Construction
- CS323(423): Operating System Design
- CS321(421): Programming Languages and Compilers
- CS257: Numerical Methods

²In 2004-2005 half my hours were for graduate credit; I did not have enough *undergraduate* hours to make the Deans List despite a 4.0 GPA both semesters.

- CS232: Computer Architecture II
- CS273: Introduction to Theoretical Computer Science
- CS231: Computer Architecture I
- CS225: Data Structures and Software Principles
- CS172: Discrete Math Structures
- CS125: Introduction to Computer Science

Reviews

I have been a reviewer for several conferences and journals including: AMAST, IEEE Trans. on VLSI Systems, RTAS, ICDCN, ISSRE, SPLASH, ASE, FSE, ISSTA, PLDI, and ICSE.