

Stephen N. Freund

Curriculum Vitae
September 12, 2017

Department of Computer Science
Williams College
Williamstown, MA 01267, USA
<http://www.cs.williams.edu/~freund>

(413) 597-4260 (phone)
(413) 597-4250 (fax)
freund@cs.williams.edu

EDUCATION

Ph.D.	Computer Science	Stanford University	June 2000
M.S.	Computer Science	Stanford University	Jan. 1998
B.S.	Computer Science (with distinction)	Stanford University	June 1995

PRINCIPAL EMPLOYMENT

Professor of Computer Science - Williams College, July 2014 – present
Associate Professor and Dept. Chair of Computer Science - Williams College, July 2011–June 2014
Associate Professor of Computer Science - Williams College, July 2008 – July 2011
Assistant Professor of Computer Science - Williams College, July 2002 – June 2008

OTHER POSITIONS

Visiting Scholar - University of Massachusetts, Amherst, May 2014 – present
Visiting Researcher - University of California, Santa Cruz, June 2005 – present
Consultant - HP Labs, Systems Research Center, June 2002 – March 2003
Member of Research Staff - Compaq Systems Research Center, Sept. 2000 – June 2002
Research Intern and Consultant - Compaq Systems Research Center, June 1999 – Sept. 2000
Research Assistant - Prof. John Mitchell, Stanford University, Sept. 1995 – June 2000
Research Assistant - Prof. Eric Roberts, Stanford University, Sept. 1993 – Sept. 1995

PRIMARY RESEARCH INTERESTS

Design and implementation of programming languages; race condition and atomicity checking; type-based program analysis; verification of multithreaded programs; programming environments.

COURSES TAUGHT AT WILLIAMS

CSCI 134: Introduction to Computer Science
CSCI 136: Data Structures and Advanced Programming
CSCI 334: Principles of Programming Languages
CSCI 434T: Compiler Design
CSCI 023: Independent Research and Development in Computer Science
CSCI 010: Introduction to C, Unix, and Software Tools

STUDENT RESEARCH SUPERVISED

- Matheus Cruz Correia de Carvalho Souza, “Optimizing Race Detection: Field Shadow State Compression,”
Summer, 2016
- Miranda Chaiken, “Dynamic Lock Adaptation,”
Summer, 2016
- David Moon, “Dynamic Verification of Concurrency Specifications,”
Honors Thesis, 2015–2016
- Alexander Majercik and Yitong Tseo, “UberLock: An Adaptive Locking Mechanism for Concurrent Programming,” Summer, 2015
- David Moon, “Optimizing Dynamic Race Detection with Hash Consing,”
Summer, 2015
- Emma Harrington, “Greed and Altruism on Stack Overflow,”
Honors Thesis, 2014–2015
*Winner, CRA Outstanding Undergraduate Researcher Awards
Second Place, Grace Hopper Conference Student Research Contest*
- Parker Finch, “Decoupling and Coalescing Race Checks,”
Honors Thesis, 2013–2014
- Emma Harrington, “Dynamic Escape Analysis for Race Checking,”
Summer 2013 and Winter 2014
- James Wilcox, “Optimizing Dynamic Race Detection in Array-Intensive Programs,”
Honors Thesis, 2012–2013
CRA Outstanding Undergraduate Researcher Awards Honorable Mention
- James Wilcox, “Whole-Program Cooperability Analysis,”
Summer 2012
- Parker Finch, “Optimizing Array Representations in Dynamic Race Detectors,”
Summer 2012 and Winter Study 2012
- Antal Spector-Zabusky, “Checking Temporal Properties of Concurrent Programs,”
Honors Thesis, 2011–2012
CRA Outstanding Undergraduate Researcher Awards Honorable Mention
- Antal Spector-Zabusky, “Visualizing Feasible Program Executions under a Relaxed Memory Model,”
Summer 2011
- Diogenese Nunez, “Statistical Sampling for Dynamic Concurrency Analyses,”
Summer 2010
- Caitlin Sadowski, “Precise Dynamic Prediction of Concurrency Errors,”
PhD Thesis Committee (UC Santa Cruz), 2010 – 2012
- Jaeheon Yi, “Dynamic Analysis of Large-Scale Programs,”
PhD Thesis Committee (UC Santa Cruz), 2008 – 2011
- Ben Wood, “Dynamic Heap Abstraction,”
Summer 2008
- Catalin Iordan, “Dynamic Heap Abstraction,”
Summer 2008
- Kenneth Knowles, “Executable Refinement Types: Hybrid Type Checking and Type Reconstruction,” MS Thesis Committee (UC Santa Cruz), 2008
- Ben Wood, “Hominy Grits: Specification and Inference of Synchronization Disciplines for Concurrent Programs,” Honors Thesis, 2007–2008
CRA Outstanding Undergraduate Researcher Awards Honorable Mention
- Ben Wood, “Sound and Precise Race Detection with Goldilocks,”
Summer 2007

Salvador Villa, “Fault Injection for Multithreaded Programs,”
 Summer 2007
 Paul Stansifer, “Alias Annotations for Faster Garbage Collection,”
 Honors Thesis, 2006–2007
 Aaron Tomb, “Hybrid Verification,”
 PhD Thesis Committee (UC Santa Cruz), 2006 – 2011
 Daniel Libicki, “The Glib Programming Language,”
 MS Thesis Committee (UC Santa Cruz), 2006
 Marina Lifshin, “Checking Atomicity Requirements in Multithreaded Programs,”
 Summer 2004
 Peter Applegate, “Solving Set Constraints with Substitutions Using Boolean Satisfiability,”
 Summer 2003

PROFESSIONAL ACTIVITIES

Program and Review Committees

- External Program Committee, PLDI, 2017
- External Review Committee, PLDI, 2016
- External Review Committee, OOPSLA, 2014
- External Review Committee, ASPLOS, 2014
- Workshop on Program Analysis and Software Techniques for Engineering, 2013 (**Co-Chair**)
- USENIX Workshop on Hot Topics in Parallelism (HotPar), 2013
- Workshop on Determinism and Correctness in Parallel Programming (WoDet), 2013
- External Review Committee, PLDI, 2013
- SPLASH Education Symposium (SPLASH-E), 2013
- International Conference on Runtime Verification, 2012
- Workshop on Foundations of Object-Oriented Languages, 2012
- ACM Conference on Programming Language Design and Implementation (PLDI), 2012
- Workshop on Formal Techniques for Java-like Programs, 2011 (**Chair**)
- ACM Symposium on Principles of Programming Languages (POPL), 2010
- TRANSACT 2010
- DEFECTS 2009
- External Review Committee, PLDI, 2009
- IBM Programming Languages Day, May 2008
- Verification and Analysis of Multi-threaded Java-like Programs (VAMP), 2007
- Generative and Transformational Techniques in Software Engineering, 2007
- Languages, Compilers, and Hardware Support for Transactional Computing, 2006
- Foundations and Developments of Object-Oriented Languages, 2006
- New England Programming Languages and Systems Seminar, June 2005 (**Chair**)
- New England Programming Languages and Systems Seminar, Feb. 2005, Oct. 2005
- Generative and Transformational Techniques in Software Engineering, 2005

Invited Lecturer, UPMARC Multicore Computing Summer School, 2014

Invited Tutorial Speaker, International Conference on Runtime Verification, 2012

Invited Lecturer, Reliable Software Systems Summer School, University of Oregon, July 2005

Chair, “Why Undergraduates Should Learn the Principles of Programming Languages” Report for the ACM SIGPLAN Education Board, 2010

GRE Computer Science Committee, *Education Testing Services (ETS)*, 2008–2010

Steering Committee, ACM SIGPLAN PL Curriculum Workshop, May 2008

Co-chair, “What to Teach about Programming Languages” Report Committee for the ACM SIGPLAN PL Curriculum Workshop, May 2008

External Honors Examiner for Swarthmore College, 2004, 2013

National Science Foundation Review Panelist, 2004, 2007, 2008, 2009, 2010, 2012, 2013, 2014, 2016
Book Proposal Reviewer for Addison-Wesley Publishing, 2004, 2007, 2008
External reviewer for promotion decisions, 2008, 2010, 2013
ACM SIGPLAN Education Board, 2009–2016

Member, Association of Computing Machinery

RESEARCH GRANTS

NSF #1439042, “XPS/RUI: SCORE: Scalability-Oriented Optimization”
Principal Investigator, with Emery Berger (UMass Amherst)
Williams award: \$252,000 (total: \$900,000), Sept. 2014 –Aug. 2018
NSF #1421051, “SHF/RUI: Fast and Precise Dynamic Race Detection: Eliminating State and Checking Redundancy”
Principal Investigator, with Cormac Flanagan (UCSC)
Williams award: \$198,993 (total: \$499,997), Sept. 2014 –Aug. 2017
NSF #1116825, “SHF/RUI: Static and Dynamic Analysis for Cooperative Concurrency”
Principal Investigator, with Cormac Flanagan (UCSC)
Williams award: \$134,059 (total: \$493,568), July 2011 –June 2014
NSF #0644130, “CAREER: Hybrid Atomicity Checking”
Principal Investigator
\$400,000, April 2007 – March 2012
NSF #0341387, “HDCCSR: Checking Atomicity for Improved Multithreaded Software Reliability”
Principal Investigator, with Cormac Flanagan (UCSC) and Martín Abadi (UCSC)
Williams award: \$218,000 (total: \$636,685), Sept. 2003 – Aug. 2007
NSF #0306486, “RUI: Modules and Parallel Specialization of Object Types”
Co-PI, with Kim Bruce
\$206,901, July 2003 – June 2006
NSF Graduate Research Fellowship
\$69,000, Sept. 1995 – June 1999

AWARDS

PLDI Distinguished Artifact Award for “BigFoot: Static Check Placement for Dynamic Race Detection”, 2017
ECOOP Best Paper Award, for “RedCard: Redundant Check Elimination for Dynamic Race Detectors,” 2013
Journal of Theoretical Computer Science Top Cited Article Award (2005–2010), for “Modular Verification of Multithreaded Programs,” 2010.
ACM SIGSOFT Distinguished Paper Award, for “Exploiting Purity for Atomicity,” 2004
CRA Outstanding Undergraduate Research Award — Honorable Mention, 1995
Phi Beta Kappa, elected 1995
Tau Beta Pi, elected 1994
GTech Corporate Fellowship, 1991–1995

PUBLICATIONS

in refereed journals, conferences, and workshops:

“BigFoot: Static Check Placement for Dynamic Race Detection,” with Dustin Rhodes and Cormac Flanagan. *Proceedings of the ACM Conference on Programming Language Design and Implementation*, pages 141–156, 2017

PLDI 2017 Distinguished Artifact Award.

- “Correctness of Partial Escape Analysis for Multithreading Optimization,” with Dustin Rhodes and Cormac Flanagan. *Workshop on Formal Techniques for Java-like Programs*, 5 pages, 2017.
- “Shadow State Compression for Precise Dynamic Race Detection,” with James Wilcox (Williams ’13), Parker Finch (Williams ’14), and Cormac Flanagan. *Automated Software Engineering*, 11 pages, 2015.
- “Cooperative Types for Controlling Thread Interference in Java,” with Jaeheon Yi, Tim Disney, and Cormac Flanagan. *Science of Computer Programming*, Volume 112(3), pages 227–260, 2015.
- “RedCard: Redundant Check Elimination for Dynamic Race Detectors,” with Cormac Flanagan. *European Conference on Object-Oriented Programming*, 25 pages, 2013.

ECOOP 2013 Best Paper Award.

- “Cooperative Types for Controlling Thread Interference in Java,” with Jaeheon Yi, Tim Disney, and Cormac Flanagan. *International Symposium on Software Testing and Analysis*, 11 pages, 2012.
- “Types for Precise Thread Interference,” with Jaeheon Yi, Tim Disney, and Cormac Flanagan. *Workshop on Foundations of Object-Oriented Languages*, 12 pages, 2011.
- “Adversarial Memory for Detecting Destructive Races,” with Cormac Flanagan. *Proceedings of the ACM Conference on Programming Language Design and Implementation*, pages 244-254, 2010.
- “FastTrack: Efficient and Precise Happens Before Race Detection,” with Cormac Flanagan. *Communications of the ACM*, Volume 53(11), pages 93–101, 2010.
- “The RoadRunner Dynamic Analysis Framework for Concurrent Programs,” with Cormac Flanagan. *Proceedings of the ACM Workshop on Program Analysis for Software Tools and Engineering*, pages 1-8, 2010.
- “FastTrack: Efficient and Precise Happens Before Race Detection,” with Cormac Flanagan. *Proceedings of the ACM Conference on Programming Language Design and Implementation*, pages 121-133, 2009.

Selected by SIGPLAN as a *Communications of the ACM* Research Highlight.

- “SingleTrack: A Dynamic Determinism Checker for Multithreaded Programs,” with Cormac Flanagan and Caitlin Sadowski. *European Symposium on Programming*, pages 394-409, 2009.
- “The Role of Programming Languages in Teaching Concurrency,” with Kim B. Bruce and Doug Lea. *Workshop on Curricula in Concurrency and Parallelism*, 3 pages, 2009.
- “Velodrome: A Sound and Complete Dynamic Atomicity Checker for Multithreaded Programs,” with Cormac Flanagan and Jaeheon Yi. *Proceedings of the ACM Conference on Programming Language Design and Implementation*, pages 293-303, 2008.
- “Types for Atomicity: Static Checking and Inference for Java,” with Cormac Flanagan, Marina Lifshin (Williams ’05), and Shaz Qadeer. *ACM Transactions on Programming Languages and Systems*, volume 30(4), pages 1–53, 2008.
- “Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs,” with Cormac Flanagan. *Science of Computer Programming*, volume 71(2), pages 89–109, 2008.
- “Programming Languages in a Liberal Arts Education,” with Kim B. Bruce. *SIGPLAN Workshop on Undergraduate Programming Language Curricula*, SIGPLAN Notices, volume 43(11), pages 45–49, 2008.
- “Programming Languages as Part of Core Computer Science,” with Kim B. Bruce. *SIGPLAN Workshop on Undergraduate Programming Language Curricula*, SIGPLAN Notices, volume 43(11), pages 50–54, 2008.

- “Type Inference Against Races,” with Cormac Flanagan. *Science of Computer Programming*, volume 64(1), pages 140–165, 2007.
- “Types for Safe Locking: Static Race Detection for Java,” with Martín Abadi and Cormac Flanagan. *ACM Transactions on Programming Languages and Systems*, volume 28(2), pages 207–255, 2006.
- “Dynamic Architecture Extraction,” with Cormac Flanagan. *Proceedings of the Workshop on Formal Approaches to Software Testing and Runtime Verification*, LNCS volume 4262, pages 209–224, 2006.
- “Sage: Hybrid Checking for Flexible Specifications,” with Jessica Gronski, Kenneth Knowles, Aaron Tomb, and Cormac Flanagan. *Workshop on Scheme and Functional Programming*, 12 pages, 2006.
- “Hybrid Types, Invariants, and Refinements for Imperative Objects,” with Cormac Flanagan and Aaron Tomb. *Workshop on Foundations and Developments of Object-Oriented Languages*, 12 pages, 2006.
- “Exploiting Purity for Atomicity,” with Cormac Flanagan and Shaz Qadeer. *IEEE Transactions on Software Engineering*, volume 31(4), 275–291, 2005.
- “Modular Verification of Multithreaded Programs,” with Cormac Flanagan, Shaz Qadeer, and Sanjit A. Seshia. *Theoretical Computer Science*, volume 338(1–3), pages 153–183, 2005.
- Theoretical Computer Science Top Cited Article (2005–2010) Award.**
- “Type Inference for Atomicity,” with Cormac Flanagan and Marina Lifshin (Williams ’05). *Proceedings of the ACM Workshop on Types in Language Design and Implementation*, pages 47–58, 2005.
- “Automatic Synchronization Correction,” with Cormac Flanagan. *Workshop on Synchronization and Concurrency in Object-Oriented Languages*, 10 pages, 2005.
- “Type Inference Against Races,” with Cormac Flanagan. *Proceedings of the Static Analysis Symposium*, pages 116–132, 2004.
- “Exploiting Purity for Atomicity,” with Cormac Flanagan and Shaz Qadeer. *Proceedings of the ACM International Symposium on Software Testing and Analysis*, pages 221–231, 2004.
- ACM SIGSOFT Distinguished Paper Award.**
- “Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs,” with Cormac Flanagan. *Proceedings of the ACM Symposium on Principles of Programming Languages*, pages 256–267, 2004.
- “Checking Concise Specifications for Multithreaded Software,” with Shaz Qadeer. *Journal of Object Technology*, volume 3(6), pages 81–101, 2004.
- “Checking Concise Specifications for Multithreaded Software,” with Shaz Qadeer. *Workshop on Formal Techniques for Java-like Programs*, 10 pages, 2003.
- “A Type System for the Java Bytecode Language and Verifier,” with John C. Mitchell. *Journal of Automated Reasoning*, volume 30(3–4), pages 271–321, 2003.
- “Run-Time Type Checking for Binary Programs,” with Mike Burrows and Janet Wiener. *Proceedings of the International Conference on Compiler Construction*, pages 90–105, 2003.
- “Thread-Modular Verification for Shared-Memory Programs,” with Cormac Flanagan and Shaz Qadeer. *Proceedings of the European Symposium on Programming*, pages 262–277, 2002.
- “Safe Asynchronous Exceptions For Python,” with Mark P. Mitchell. *Lightweight Languages Workshop*, 6 pages, 2002.
- “Detecting Race Conditions in Large Programs,” with Cormac Flanagan. *Proceedings of the ACM Workshop on Program Analysis for Software Tools and Engineering*, pages 90–96, 2001.
- “Type-Based Race Detection for Java,” with Cormac Flanagan. *Proceedings of the ACM Conference on Programming Language Design and Implementation*, pages 219–232, 2000.

- “Type-Based Race Detection for Java,” with Cormac Flanagan. Short topic at *IEEE Conference on Logic in Computer Science*, 2 pages, 2000.
- “A Type System for Object Initialization in the Java Bytecode Language,” with John C. Mitchell. *ACM Transactions on Programming Languages and Systems*, volume 21(6), pages 1196–1250, 1999.
- “A Formal Framework for the Java Bytecode Language and Verifier,” with John C. Mitchell. *Proceedings of the ACM Conference on Object-Oriented Programming: Systems, Languages and Applications*, pages 147–166, 1999.
- “A Type System for Object Initialization in the Java Bytecode Language,” with John C. Mitchell. *Proceedings of the ACM Conference on Object-Oriented Programming: Systems, Languages and Applications*, pages 210–227, 1998.
- “The Costs and Benefits of Java Bytecode Subroutines.” *Workshop on the Formal Underpinnings of the Java Paradigm*, 14 pages, 1998.
- “Adding Type Parameterization to the Java Language,” with Ole Agesen and John C. Mitchell. *Proceedings of the ACM Conference on Object-Oriented Programming: Systems, Languages and Applications*, pages 49–65, 1997.
- “A Type System for Object Initialization in the Java Bytecode Language,” with John C. Mitchell. *Proceedings of the Workshop on Higher Order Operational Techniques in Semantics (ENTCS, volume 10)*, 4 pages, 1997. Also presented at *Workshop on Security and Languages*, 1997.
- “Thetis: An ANSI C Programming Environment Designed for Introductory Use,” with Eric Roberts. *Proceedings of the ACM SIGCSE Technical Symposium on Computer Science Education*, pages 300–304, 1996.

in edited volumes, technical reports, patents, and other venues:

- “Teaching and Researching Programming Languages at a Liberal Arts College.” *The Programming Languages Enthusiast*, 2015. Available at: <http://www.pl-enthusiast.net/2015/03/16/teaching-and-researching-pl-at-a-liberal-arts-college/>.
- “Cooperative Concurrency for a Multicore World (Extended Abstract),” with Jaeheon Yi, Caitlin Sadowski, and Cormac Flanagan. *Proceedings of the International Conference on Runtime Verification*, 3 pages, 2011.
- “Why Undergraduates Should Learn the Principles of Programming Languages,” with Kim Bruce, Chair (Pomona College), Kathi Fisler (WPI), Dan Grossman (University of Washington), Matthew Hertz (Canisius College), Gary T. Leavens (University of Central Florida), Andrew Myers (Cornell University), Larry Snyder (University of Washington). 2010.
- “What a Programming Languages Curriculum Should Include,” with Kim Bruce, Robert Harper, Jim Larus, and Gary Leavens (lead authors). *Proceedings of the SIGPLAN Workshop on Undergraduate Programming Language Curricula*, SIGPLAN Notices, volume 43(11), pages 11–24, 2008.
- “Method and apparatus for verifying data local to a single thread,” with Cormac Flanagan. *United States Patent 6,817,009*, issued 2004.
- “Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs (Summary),” with Cormac Flanagan. *Proceedings of Workshop on Parallel and Distributed Systems: Testing and Debugging*, invited contribution, 2 pages, 2004.
- “Exploiting Purity for Atomicity (extended version),” with Cormac Flanagan and Shaz Qadeer. Williams College Technical Note 04-05, 23 pages, 2004.
- “Partial Type and Effect Inference for Rcc/Java is NP-Complete,” with Cormac Flanagan. Williams College Technical Note 04-01, 5 pages, 2004.
- “Checking Concise Specifications for Multithreaded Software (extended version),” with Shaz Qadeer. Williams College Technical Note 01-2002, 16 pages, 2002.

“Thread-Modular Verification for Shared-Memory Programs (extended version),” with Cormac Flanagan and Shaz Qadeer. Compaq Systems Research Center Technical Note 2001-03, 19 pages, 2001.

“A Type System for Java Bytecode Subroutines and Exceptions,” with John C. Mitchell. Stanford Computer Science Technical Note STAN-CS-TN-99-91, 20 pages, 1999.

dissertation:

Type Systems for Object-Oriented Intermediate Languages, Stanford University, 299 pages, 2000.

SOFTWARE

RoadRunner Dynamic Analysis Framework: <http://www.cs.williams.edu/~freund/rr/>
A framework for writing dynamic analyses for concurrent Java programs, 2009–present.

INVITED TALKS

BigFoot: Static Check Placement for Dynamic Race Detection

- Microsoft Research, Redmond, WA, Aug. 2017.

Data Race Detection: FastTrack and Beyond

- University of Massachusetts, Amherst, MA, April 2017.

SCORE: Scalability-Oriented Optimization

- NSF Workshop on Exploiting Parallelism and Scalability (XPS), Arlington, VA, June 2015.

Analysis Techniques to Detect Concurrency Errors

- UPMARC Summer School on Multicore Computing, Uppsala, Sweden, July 2014.

(Tutorial presented with Cormac Flanagan.)

Dynamic Analyses for Data Race Detection

- University of Massachusetts, Amherst, MA, March and November 2013.

Dynamic Analyses for Concurrency

- International Conference on Runtime Verification, Istanbul, Turkey, September 2012.

(Tutorial presented with John Erickson and Madan Musuvathi.)

Cooperative Concurrency for a Multicore World

- IBM Programming Languages Day, Hawthorne, NY, June 2012.

- University of Massachusetts, Amherst, MA, February 2012.

- University of Washington, Seattle, WA, November 2011.

Stopping the Software Bug Epidemic

- Faculty Lecture Series, Williams College, February 2011.

FastTrack and Jumble: Efficient and Precise Dynamic Detection of Destructive Races

- Cornell University, March 2011.

- Harvard University, November 2010.

FastTrack: Efficient and Precise Dynamic Race Detection

- Williams College, October 2009.

- University of Massachusetts, Amherst, MA, September 2009.

- UC Santa Cruz, Santa Cruz, CA, May. 2009.

Types for Concurrency

- Invited Keynote Lecture, Schloss Dagstuhl on Design and Validation of Concurrent Systems, Germany, Aug. 2009.

Squashing the Bugs: Dynamic and Static Checkers for Concurrency

- UC Santa Cruz, Santa Cruz, CA, Feb. 2009.

- Velodrome: Sound and Complete Atomicity Checking*
- Brown University, Providence, RI, March 2009.
 - Princeton University, Princeton, NJ, March 2009.
 - Pomona College, Claremont, CA, Jan. 2009.
 - Microsoft Research, Silicon Valley, CA, Nov. 2008.
 - Microsoft Research, Redmond, WA, Nov. 2008.
 - University of Massachusetts, Amherst, MA, Feb. 2008.
- Squashing the Bugs: Tools for Building Better Software and Atomizer: A Dynamic Bug Finder for Large Systems*
- Sigma Xi Lecture Series, Williams College, October, 2006.
- Practical Hybrid Type Checking*
- Stanford University, Stanford, CA, May 2006.
 - Microsoft Research, Redmond, WA, May 2006.
- Dynamic Heap Model Extraction*
- University of California, Santa Cruz, Santa Cruz, CA, May 2006.
- Lightweight Atomicity Checking*
- University of California, Santa Cruz, Santa Cruz, CA, Feb. 2006.
- Type Inference for Race Conditions and Atomicity*
- University of Washington, Seattle, WA, May 2006.
 - University of British Columbia, Vancouver, BC, May 2006.
 - Intel, Santa Clara, CA, Nov. 2005.
- Automatic Synchronization Correction*
- Microsoft Research, Mountain View, CA, Dec. 2005.
- Atomicity Checkers*
- University of California, Santa Cruz (2 lectures), Santa Cruz, CA, Oct. 2005.
- Lightweight Analyses for Reliable Concurrency*
- Reliable Software Systems Summer School (3 lectures), Eugene, OR, July 2005.
- Exploiting Purity for Atomicity*
- New England Programming Languages Seminar, Boston, MA, Feb. 2004.
- Finding Bugs in Software*
- Bronfman Science Lunch, Williams College, Williamstown, MA, Nov. 2003.
- Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs*
- Union College, Schenectady, NY, Nov. 2007.
 - Pomona College, Los Angeles, CA, Sept. 2005.
 - University of Illinois at Urbana-Champaign, Urbana, IL, Aug. 2004.
 - University of Pennsylvania, Philadelphia, PA, May 2004.
 - University of California, Berkeley, Berkeley, CA, Dec. 2003.
 - Stanford University, Stanford, CA, Sept. 2003.
 - AT&T Research, Florham Park, NJ, Aug. 2003.
- Safe Asynchronous Exceptions for Python*
- HP Labs, Palo Alto, CA, May 2003.
- Hobbes: A Run-Time Type Checker for Binary Programs*
- Microsoft Research, Mountain View, CA, May 2003.
- Better Abstraction via Race Freedom*
- New England Programming Languages Seminar, Yale, CT, Aug. 2002.
- Detecting Race Conditions in Large Programs*
- Brown University, Providence, RI, June 2002.
 - Microsoft Research, Redmond, WA, Sept. 2001.
 - Stanford University, Stanford, CA, Aug. 2001.

Type-Based Race Detection For Java

- Hamilton College, NY, Feb. 2002.
- Williams College, Williamstown, MA, Feb. 2002.
- Swarthmore College, PA, Feb. 2002.
- Carleton College, MN, Feb. 2002.
- Stanford University, Stanford, CA, Jan. 2000.
- AT&T Research, Florham Park, NJ, Jan. 1999.
- IBM TJ Watson Research Center, Hawthorne, NY, Jan. 1999.

Type Systems for Object-Oriented Intermediate Languages

- Stanford Computer Forum Annual Meeting, Stanford, CA, June 2000.
- Compaq Systems Research Center, Palo Alto, CA, May 2000.
- Johns Hopkins University, Baltimore, MD, May 2000.
- Microsoft Research, Redmond, WA, April 2000.
- AT&T Research, Florham Park, NJ, April 2000.
- Lucent Technologies Bay Area Research Lab, Palo Alto, CA, April 2000.

COLLEGE SERVICE AND COMMITTEES

Chair, Committee on Priorities and Resources (CPR), 2016–2017
Committee on Undergraduate Life (CUL), 2015–2016
Faculty Interview Committee, 2013–2014
First-Year Faculty Mentoring Program, 2009–2013, 2015–2016
Science Executive Committee, 2011–2014
Committee on Admission and Financial Aid (CAFA), 2011–2012
Advisory Group on Admission and Financial Aid (AGAFA), 2009–2011
Goldwater Fellowship Selection Committee, 2007, 2008
Committee on Priorities and Resources, 2006–2008
Honor System Committee, 2004–2005
Discipline Committee, 2004–2005
Committee on Student Course Evaluations and Pedagogy (CoSCEP), 2004–2005
Committee on Pedagogy and Its Evaluation (CoPE), 2003–2004
OCC Panel on Graduate School, 2004
First-Year Adviser, 2003–2005, 2006–2008, 2009–2014, 2015–2017
BIGP Advisory Committee, 2002–2008

DEPARTMENTAL SERVICE AND COMMITTEES

Department Chair, 2011–2014
Departmental Colloquium Organizer, 2010
COSSAC and Social Events Organizer, 2009
Library Liason, 2009–2010
Computer Facilities Manager, 2007–2008, 2010–2011
Web Pages and Documentation Support, 2007–2008, 2009–2010, 2011–2014, 2015–2016
Division III and Psychology Research Funds Committee, 2004–2005
TA and Tutor Manager, 2004–2005, 2006–2007, 2009–2010, 2016–2017
Web Pages and Documentation Manager, 2004–2005, 2006–2007, 2010–2011
Computer Facilities Support, 2003–2004, 2013–2014, 2015–2016