

# Cormac Flanagan

Computer Science Department  
University of California, Santa Cruz

September 9, 2011

## Professional Experience

- 2009– *Full Professor of Computer Science* (with tenure), UCSC
- 2005–09 *Associate Professor of Computer Science* (with tenure), UCSC
- Fall 2006 *Visiting Researcher*, Max Planck Institute for Software Systems
- 2003–05 *Assistant Professor of Computer Science*, UCSC
- 2002–03 *Principal Research Scientist*, Hewlett Packard Corporation.
- 1998–02 *Principal Research Scientist*, Compaq Computer Corporation.
- 1997–98 *Principal Research Scientist*, Digital Equipment Corporation.
- 1990–91 *Software Engineer*, Peregrine Expert Systems.

## Education

- Ph.D. 1997 *Computer Science*, Rice University.  
Dissertation Title: “Componential Set-Based Analysis.”  
Advisor: Matthias Felleisen
- M.Sc. 1995 *Computer Science*, Rice University.
- B.Sc. 1990 *Computer Science and Mathematics*, University College Dublin.

## Honors and Awards

- 2010 *Theoretical Computer Science Top Cited Article Award 2005-10* for [7].  
*Modular Verification of Multithreaded Programs*. Cormac Flanagan, Stephen N. Freund, Sanjit Seshia, and Shaz Qadeer. *Theoretical Computer Science* 338 (1-3), 2005.
- 2005-09 *Alfred P. Sloan Foundation Fellowship*.
- 2004 *ACM SIGPLAN Distinguished Paper Award* [46], *International Symposium on Software Testing and Analysis*.
- 2003 “The Essence of Compiling with Continuations” [72] selected as one of the 50 most influential contributions in the last twenty years of the *Programming Language Design and Implementation Conference*.

## Grants and Awards

- 2011-14 PI, NSF award CCF-1116883. “Static and Dynamic Analysis for Cooperative Concurrency”. \$359,509.
- 2009-11 PI, NSF award CNS-0905650. “Next-Generation Infrastructure for Trustworthy Web Applications”. \$300,000.
- 2010 PI, IBM Innovation Grant. “Concurrency Types for X10: Race-Freedom, Atomicity, and Determinism”. \$20,000.
- 2010 PI, ONR STTR award (subcontract from MZA Associate Corporation, Albuquerque). “Mathematically Rigorous Methods for Determining Software Quality”, \$28,000.
- 2007-10 Co-PI, NSF award P200A070588-08. “Graduate Assistance in Areas of National Need (GAANN)”. \$384,390.
- 2007-10 PI, NSF award CCR-0707885. “A JML Community Infrastructure – Revitalizing Tools and Documentation to Aid Formal Methods Research”. \$150,000, part of a multi-university project funded at \$895,000.
- 2005-07 Alfred P. Sloan Foundation Fellowship. \$45,000.
- 2005-06 PI, UC MICRO award. “Lightweight Transactions for Robust Error Handling”. \$42,174.
- 2005 PI, Microsoft Research award. “Lightweight Transactions for Robust Error Handling”. \$49,192.
- 2003-08 PI, NSF award CCR-0341179. “Checking Atomicity for Improved Multi-threaded Software Reliability”. \$257,773.
- 2002-05 Irish Research Council Basic Research Award. “Automated Verification of Security Protocols”. Euro 173,000. (Collaborator)
- 1997 NSF-NATO Postdoctoral Fellowship awarded but declined by me in favor of a position at the Systems Research Center.
- 1996–97 Lodieska Stockbridge Vaughan Fellowship, Rice University.
- 1987,88,89 Scholarships in Computer Science, University College Dublin.
- 1986 First place in the Irish Leaving Certificate examination in Mathematics.

## Consulting

- 2006– Invited expert on the Ecma TC39 committee for standardization of the JavaScript programming language.
- 2006– Consultant for Mozilla Corporation.
- 2007–08 Consultant for Solidware Corporation.
- 2006 Consultant for Microsoft Corporation.

## Board Member

- 2008– Engineering Advisory Board for UCSC Extension.
- 2007–08 Software Engineering and Quality Program Advisory Board, UCSC Extension.
- 2006–07 Software Quality Engineering and Management Program Advisory Board, UCSC Extension.

## Publications

Publications are available at: <http://www.soe.ucsc.edu/~cormac>

## Journal Publications

- [1] “FastTrack: Efficient and Precise Dynamic Race Detection”. Cormac Flanagan and Stephen N. Freund. *Communications of the ACM (CACM)*, 53, 11 (November 2010).
- [2] “Hybrid Type Checking”. Kenneth Knowles and Cormac Flanagan. *ACM Transactions on Programming Languages and Systems*, 32, 2 (January 2010), 1–34.
- [3] “Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs”. Cormac Flanagan and Stephen N. Freund. *Science of Computer Programming*, 71, 2 (April 2008), 89–109.
- [4] “Types for Atomicity: Static Checking and Inference for Java”. Cormac Flanagan, Stephen N. Freund, Marina Lifshin, and Shaz Qadeer. *ACM Transactions on Programming Languages and Systems*, 30, 4 (July 2008), 1–53.
- [5] “Type Inference Against Races”. Cormac Flanagan and Stephen N. Freund. *Science of Computer Programming*, 64, 1 (September 2006), 140–165.
- [6] “Types for Safe Locking: Static Race Detection for Java”. Martin Abadi, Cormac Flanagan, and Stephen N. Freund. *ACM Transactions on Programming Languages and Systems*, 28, 2 (March 2006), 207–255.
- [7] “Modular Verification of Multithreaded Programs”. Cormac Flanagan, Stephen N. Freund, Shaz Qadeer, and Sanjit Seshia. *Theoretical Computer Science*, 338, 1-3 (June 2005), 153–183.
- [8] “Exploiting Purity for Atomicity”. Cormac Flanagan, Stephen N. Freund, and Shaz Qadeer. *IEEE Transactions on Software Engineering*, 31, 4 (April 2005), 275–291.
- [9] “Automatic Software Model Checking via Constraint Logic”. Cormac Flanagan. *Science of Computer Programming*, 50, 1 (March 2004), 253–270.

- [10] “DrScheme: A Programming Environment for Scheme”. Robert Bruce Findler, John Clements, Cormac Flanagan, Matthew Flatt, Shriram Krishnamurthi, Paul Steckler, and Matthias Felleisen. *Journal of Functional Programming*, 12, 2, (March 2002), 159–182.
- [11] “Annotation Inference for Modular Checkers”. Cormac Flanagan, Rajeev Joshi, and K. Rustan M. Leino. *Information Processing Letters*, 77, 2–4 (February 2001), 97–108.
- [12] “Componential Set-Based Analysis”. Cormac Flanagan and Matthias Felleisen. *ACM Transactions on Programming Languages and Systems*, 21, 2 (March 1999) 370–416.
- [13] “The Semantics of Future and an Application”. Cormac Flanagan and Matthias Felleisen. *Journal of Functional Programming*, 9, 1 (January 1999), 1–31.

### Refereed Conference and Workshop Publications

- [14] “Types for Precise Thread Interference”. Jaeheon Yi, Tim Disney, Stephen Freund, and Cormac Flanagan. *International Workshop on Foundations of Object-Oriented Languages (FOOL)*, (Oct 2011).
- [15] “Virtual Values for Language Extension”. Tim Disney, Tom Austin, and Cormac Flanagan. *Object Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, (Oct 2011).
- [16] “Temporal Higher-Order Contracts”. Tim Disney, Cormac Flanagan, Jay McCarthy. *Proceedings of the The 16th ACM SIGPLAN International Conference on Functional Programming (ICFP)*, (Sept 2011).
- [17] “Cooperative Reasoning for Preemptive Execution”. Jaeheon Yi, Caitlin Sadowski, and Cormac Flanagan. *Proceedings of the ACM SIGPLAN Conference on Principles and Practice of Parallel Programming (PPOPP)*, (February 2011), 147–156.
- [18] “Correct Blame for Contracts: No More Scapegoating”. Christos Dimoulas, Robert Bruce Findler, Cormac Flanagan, and Matthias Felleisen. *Proceedings of the 38th ACM Symposium on Principles of Programming Languages (POPL)* (January 2011), 215–226.
- [19] “Gradual Information Flow Typing”. Tim Disney and Cormac Flanagan. *Workshop on Script to Program Evolution (STOP)*, (January 2011).
- [20] “Adversarial Memory for Detecting Destructive Races”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, (June 2010), 244–254.

- [21] “Permissive Dynamic Information Flow Analysis”. Thomas Austin and Cormac Flanagan. *Proceedings of the ACM SIGPLAN Fourth Workshop on Programming Languages and Analysis for Security (PLAS)*, (June 2010).
- [22] “The RoadRunner Dynamic Analysis Framework for Concurrent Programs”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the 2010 ACM Workshop on Program Analysis For Software Tools and Engineering, PASTE 2010* (June 2010), 1–8.
- [23] “Effects for Cooperable and Serializable Threads”. Jaeheon Yi and Cormac Flanagan. *Proceedings of the Fifth ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI)*, (Jan 2010), 3–14.
- [24] “FastTrack: Efficient and Precise Dynamic Race Detection”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, (June 2009), 121–133.
- [25] “SideTrack: Generalizing Dynamic Atomicity Analysis”. Jaeheon Yi, Caitlin Sadowski, and Cormac Flanagan. *Proceedings of the Workshop on Parallel and Distributed Systems: Testing, Analysis, and Debugging (PADTAD)*, (July 2009).
- [26] “Efficient Purely-Dynamic Information Flow Analysis”. Thomas Austin and Cormac Flanagan. *Proceedings of the ACM SIGPLAN Fourth Workshop on Programming Languages and Analysis for Security (PLAS)*, (June 2009).  
This paper was one of two PLAS papers selected for publication in ACM SIGPLAN Notices 44 (8), pages 20–21 (August 2009).
- [27] “SingleTrack: A Dynamic Determinism Checker for Multithreaded Programs”. Caitlin Sadowski, Stephen N. Freund, and Cormac Flanagan. *18th European Symposium on Programming (ESOP)*, (March 2009), Springer-Verlag, 394–409.
- [28] “Compositional and Decidable Checking for Dependent Contract Types”. Kenneth Knowles and Cormac Flanagan. *Proceedings of the Workshop on Programming Languages meets Program Verification (PLPV)*, (Jan 2009).
- [29] “Proving Correctness of a Dynamic Atomicity Analysis in Coq”. Caitlin Sadowski, Jaeheon Yi, Kenneth Knowles, and Cormac Flanagan. *Third Informal ACM SIGPLAN Workshop on Mechanizing Metatheory*, (September 2008).
- [30] “Velodrome: A Sound and Complete Dynamic Atomicity Checker for Multithreaded Programs”. Cormac Flanagan, Stephen N. Freund, and Jaeheon Yi. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, (June 2008), 293–303.
- [31] “Status Report: Specifying JavaScript with ML”. David Herman and Cormac Flanagan. *Proceedings of the ACM Workshop on ML*, (October 2007), 47–52.

- [32] “Cartesian Partial-Order Reduction”. Guy Gueta, Cormac Flanagan, Eran Yahav, and Mooly Sagiv. *14th International SPIN Workshop on Model Checking Software*, (July 2007), 95–112.
- [33] “Space Efficient Gradual Typing”. David Herman, Aaron Tomb, and Cormac Flanagan. *Eighth Symposium on Trends in Functional Programming*, (April 2007).
- [34] “Unifying Hybrid Types and Contracts”. Jessica Gronski and Cormac Flanagan. *Eighth Symposium on Trends in Functional Programming*, (April 2007).
- [35] “Type Reconstruction for General Refinement Types”. Kenneth Knowles and Cormac Flanagan. *16th European Symposium on Programming (ESOP)*, (March 2007), Springer-Verlag, 505–519.
- [36] “Sage: Hybrid Checking for Flexible Specifications”. Jessica Gronski, Kenneth Knowles, Aaron Tomb, Stephen N. Freund, and Cormac Flanagan. *Workshop on Scheme and Functional Programming* (September 2006).
- [37] “Dynamic Architecture Extraction”. Cormac Flanagan and Stephen N. Freund. *Workshop on Formal Approaches to Testing and Runtime Verification* (August 2006), 209–224.
- [38] “Hybrid Type Checking”. Cormac Flanagan. *Proceedings of the 33rd ACM Symposium on Principles of Programming Languages* (January 2006), 245–256.
- [39] “Hybrid Types, Invariants, and Refinements for Imperative Objects”. Cormac Flanagan, Stephen N. Freund, and Aaron Tomb. *International Workshop on Foundations and Developments of Object-Oriented Languages* (January 2006).
- [40] “Automatic Synchronization Correction”. Cormac Flanagan and Stephen N. Freund. *Workshop on Synchronization and Concurrency in Object-Oriented Languages* (October 2005).
- [41] “Extending JML for Modular Specification and Verification of Multi-Threaded Programs”. Edwin Rodriguez, Matthew Dwyer, Cormac Flanagan, John Hatcliff, Gary T. Leavens, and Robby. *European Conference on Object Oriented Programming* (July 2005).
- [42] “Automatic Type Inference via Partial Evaluation”. Aaron Tomb and Cormac Flanagan. *Principles and Practice of Declarative Programming* (July 2005).
- [43] “Type Inference for Atomicity”. Cormac Flanagan and Stephen N. Freund and Marina Lifshin. *Proceedings of the ACM SIGPLAN Workshop on Types in Language Design and Implementation, TLDI 2005*, (January 2005).
- [44] “Dynamic Partial-Order Reduction for Model Checking Software”. Cormac Flanagan and Patrice Godefroid. *Proceedings of the 32nd ACM Symposium on Principles of Programming Languages* (January 2005).

- [45] “Type Inference Against Races”. Cormac Flanagan and Stephen N. Freund. *Static Analysis, 11th International Symposium, SAS 2004*, Springer-Verlag (August 2004).
- [46] “Exploiting Purity for Atomicity”. Cormac Flanagan, Stephen N. Freund, and Shaz Qadeer. *Proceedings of the ACM International Symposium on Software Testing and Analysis, ISSTA 2004* (July 2004), 221–231. This paper received an ACM SIGPLAN Distinguished Paper Award.
- [47] “Verifying Commit-Atomicity Using Model-Checking”. Cormac Flanagan. *11th International SPIN Workshop on Model Checking Software*, (Susanne Graf and Laurent Mounier, eds.), Springer-Verlag (April 2004), 252–266.
- [48] “Software Model Checking via Iterative Abstraction Refinement Constraint Logic Queries”. Cormac Flanagan. *Workshop on Constraint Programming and Constraints for Verification*, (April 2004).
- [49] “Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the 31st ACM Symposium on Principles of Programming Languages* (January 2004), 256–267.
- [50] “Theorem Proving using Lazy Proof Explication”. Cormac Flanagan, Rajeev Joshi, Xinming Ou, and James B. Saxe. *Computer Aided Verification, 15th International Conference, CAV 2003*, (Warren A. Hunt Jr. and Fabio Somenzi, eds.), Springer-Verlag (July 2003), 355–367.
- [51] “A Type and Effect System for Atomicity”. Cormac Flanagan and Shaz Qadeer. *Proceedings of the ACM SIGPLAN 2003 Conference on Programming Language Design and Implementation 2003*, (June 2003), 338–349.
- [52] “Transactions for Software Model Checking”. Cormac Flanagan and Shaz Qadeer. *Workshop on Software Model Checking, SoftMC 2003*, Electronic Notes in Theoretical Computer Science, Volume 89, 3, 2003.
- [53] “Thread-Modular Model Checking”. Cormac Flanagan and Shaz Qadeer. *SPIN Model Checking Software, 10th International SPIN Workshop*, (Thomas Ball and Sriram K. Rajamani, eds.), Springer-Verlag (May 2003), 213–224.
- [54] “Automatic Software Model Checking using CLP”. Cormac Flanagan. *Programming Languages and Systems, 12th European Symposium on Programming, ESOP 2003*, (Pierpaolo Degano, ed.), Springer-Verlag (April 2003), 189–203.
- [55] “Types for Atomicity”. Cormac Flanagan and Shaz Qadeer. *Proceedings of the ACM SIGPLAN Workshop on Types in Language Design and Implementation, TLDI 2003*, (January 2003), 1–12.
- [56] “A Modular Checker for Multithreaded Programs”. Cormac Flanagan, Shaz Qadeer, and Sanjit A. Seshia. *Computer Aided Verification, 14th International Conference, CAV 2002*, (Ed Brinksma and Kim Guldstrand Larsen, eds.), Springer-Verlag (July 2002), 180–194.

- [57] “Extended Static Checking for Java”. Cormac Flanagan, K. Rustan M. Leino, Mark Lillibridge, Greg Nelson, James B. Saxe, and Raymie Stata. *Proceedings of the 2002 ACM Conference on Programming Language Design and Implementation, PLDI 2002*, SIGPLAN Notices 37, 5, (June 2002), 234–245.
- [58] “Thread-Modular Verification for Shared-Memory Programs”. Cormac Flanagan, Stephen N. Freund, and Shaz Qadeer. *11th European Symposium on Programming, ESOP 2002*, (Daniel Le Métayer, ed.), Springer-Verlag (April 2002), 262–277.
- [59] “Predicate Abstraction for Software Verification”. Cormac Flanagan and Shaz Qadeer. *Conference Record of POPL 2002: The 29th Symposium on Principles of Programming Languages*, (January 2002), 191–202.
- [60] “Detecting Race Conditions in Large Programs”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the 2001 ACM Workshop on Program Analysis For Software Tools and Engineering, PASTE 2001* (June 2001), 90–96.
- [61] “Houdini, an Annotation Assistant for ESC/Java”. Cormac Flanagan and K. Rustan M. Leino. *FME 2001: Formal Methods for Increasing Software Productivity, International Symposium of Formal Methods Europe*, (José Nuno Oliveira and Pamela Zave, eds.), Springer-Verlag (March 2001), 500–517.
- [62] “Avoiding Exponential Explosion: Generating Compact Verification Conditions”. Cormac Flanagan and James B. Saxe. *Conference Record of POPL 2001: The 28th ACM Symposium on Principles of Programming Languages*, ACM SIGPLAN Notices 36, 3, (January 2001), 193–205.
- [63] “Type-Based Race Detection for Java”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the 2000 ACM SIGPLAN Conference on Programming Language Design and Implementation*, SIGPLAN Notices 35, 5 (June 2000), 219–232.
- [64] “Object Types against Races”. Cormac Flanagan and Martin Abadi. *CONCUR ’99: Concurrency Theory, 10th International Conference*, (Jos C. M. Baeten and Sjouke Mauw, eds.), Springer-Verlag (August 1999), 288–303.
- [65] “Types for Safe Locking”. Cormac Flanagan and Martin Abadi. *Programming Languages and Systems, 8th European Symposium on Programming, ESOP’99*, (S. Doaitse Swierstra, ed.), Springer-Verlag (March 1999), 91–108.
- [66] “A New Way of Debugging Lisp Programs”. Cormac Flanagan and Matthias Felleisen. *Proceedings of the Conference on the 40th Anniversary of Lisp: Lisp in the Mainstream*, (November 1998).
- [67] “DrScheme: A Pedagogic Programming Environment for Scheme”. Robert Bruce Findler, Cormac Flanagan, Matthew Flatt, Shriram Krishnamurthi, and

Matthias Felleisen. *Programming Languages: Implementations, Logics, and Programs, 9th International Symposium*, (Hugh Glaser and Pieter H. Hartel and Herbert Kuchen, eds.), Springer-Verlag (September 1997), 369–388.

- [68] “Componential Set-Based Analysis”. Cormac Flanagan and Matthias Felleisen. *Proceedings of the ACM SIGPLAN ’97 Conference on Programming Language Design and Implementation, PLDI ’97*, SIGPLAN Notices 32, 5 (June 1997), 235–248.
- [69] “Catching Bugs in the Web of Program Invariants”. Cormac Flanagan, Matthew Flatt, Shriram Krishnamurthi, Stephanie Weirich, and Matthias Felleisen. *Proceedings of the ACM SIGPLAN’96 Conference on Programming Language Design and Implementation, PLDI ’96*, SIGPLAN Notices 31, 5 (May 1996), 23–32.
- [70] “pHluid: The Design of a Parallel Functional Language Implementation on Workstations”. Cormac Flanagan and Rishiyur S. Nikhil. *Proceedings of the 1996 ACM SIGPLAN International Conference on Functional Programming (ICFP)*, SIGPLAN Notices 31, 6 (May 1996), 169–179.
- [71] “The Semantics of Future and its use in Program Optimizations”. Cormac Flanagan and Matthias Felleisen. *Conference Record of the 22nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, (January 1995), 209–220.
- [72] “The Essence of Compiling with Continuations”. Cormac Flanagan, Amr Sabry, Bruce F. Duba, and Matthias Felleisen. *Proceedings of the ACM SIGPLAN’93 Conference on Programming Language Design and Implementation, PLDI ’93*, SIGPLAN Notices 28, 6 (June 1993), 237–247.

### Invited Papers

- [73] “Cooperative Concurrency for a Multicore World (Extended Abstract)”. Jaeheon Yi, Caitlin Sadowski, Stephen Freund and Cormac Flanagan. *Second International Conference on Runtime Verification*, 2011.
- [74] “Futures”. Cormac Flanagan. Article in *Encyclopedia of Parallel Computing*, Springer 2011.
- [75] “Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs (Summary)”. Cormac Flanagan and Stephen N. Freund. *Parallel and Distributed Systems: Testing and Debugging Workshop, part of 18th International Parallel and Distributed Processing Symposium, IPDPS 2004* (April 2004). This invited paper is a summary of [49].

## Collections

- [76] “Design and Validation of Concurrent Systems”. Cormac Flanagan, Susanne Graf, Madhusudan Parthasarathy, and Shaz Qadeer. Dagstuhl Seminar Proceedings 09361. <http://drops.dagstuhl.de/opus/volltexte/2010/2549>
- [77] “Proceedings of the 4th ACM Workshop Programming Languages meets Program Verification, PLPV 2010”. Jean-Christophe Fillitre and Cormac Flanagan. ACM 2010.
- [78] “Proceedings of the 2004 ACM SIGPLAN-SIGSOFT Workshop on Program Analysis For Software Tools and Engineering, PASTE 2004”. Cormac Flanagan and Andreas Zeller. ACM 2004.

## Dissertation

- [79] “Componential Set-Based Analysis”. Cormac Flanagan. Doctoral Thesis, Rice University (July 1997).

## Abstracts

- [80] “Type-Based Race Detection for Java (summary)”. Cormac Flanagan and Stephen N. Freund. *15th Annual IEEE Symposium on Logic in Computer Science* (June 2000). This paper is a summary of [63].

## Technical Reports

- [81] “Types for Precise Thread Interference”. Jaeheon Yi, Tim Disney, Stephen N. Freund, Cormac Flanagan . Technical Report UCSC-SOE-11-22, 2011.
- [82] “Dynamic Information Flow Analysis for Featherweight JavaScript”. Thomas H. Austin, Tim Disney, Cormac Flanagan, Alan Jeffrey. Technical Report UCSC-SOE-11-19, 2011.
- [83] “Virtual Values for Language Extension”. Thomas H. Austin, Tim Disney, and Cormac Flanagan. Technical Report UCSC-SOE-10-32, 2010.
- [84] “Permissive Dynamic Information Flow Analysis”. Thomas H. Austin and Cormac Flanagan. Technical Report UCSC-SOE-09-34, Nov. 2009.
- [85] “Compositional and Decidable Checking for Dependent Contract Types”. Kenneth Knowles and Cormac Flanagan. Technical Report UCSC-SOE-08-17, Aug. 2008.
- [86] “Modular Verification of Multithreaded Programs”. Cormac Flanagan, Stephen N. Freund, Shaz Qadeer, and Sanjit A. Seshia. Williams College Technical Note 04-08, Nov. 2004.

- [87] “An Explicating Theorem Prover for Quantified Formulas”. Cormac Flanagan, Rajeev Joshi, and James B. Saxe. Hewlett-Packard Labs Technical Report HPL-2004-199, Nov. 2004.
- [88] “Type Inference Against Races (extended version)”. Cormac Flanagan and Stephen N. Freund. Williams College Technical Note 04-06, Sept. 2004.
- [89] “Exploiting Purity for Atomicity (extended version)”. Cormac Flanagan, Stephen N. Freund, and Shaz Qadeer. Williams College Technical Note 04-05, July 2004.
- [90] “Partial Type And Effect Inference for Rcc/Java in NP-Complete”. Cormac Flanagan and Stephen N. Freund. Williams College Technical Note 04-01, Feb. 2004.
- [91] “Thread-Modular Verification For Shared-Memory Programs”. Cormac Flanagan, Stephen Freund, and Shaz Qadeer. Systems Research Center Technical Note SRC-TN-2001-003, Nov. 2001.
- [92] “Houdini, an Annotation Assistant for ESC/Java”. Cormac Flanagan and K. Rustan M. Leino. Systems Research Center Technical Note SRC-TN-2000-003, Dec. 2000.
- [93] “Modular and Polymorphic Set-Based Analysis: Theory and Practice”. Cormac Flanagan and Matthias Felleisen. Rice University Department of Computer Science Technical Report TR96-266, Jan. 1996.
- [94] “Set Based Analysis for Full Scheme and Its Use in Soft-Typing”. Cormac Flanagan and Matthias Felleisen. Rice University Department of Computer Science Technical Report TR95-254, Oct. 1995.
- [95] “Well-Founded Touch Optimization of Futures”. Cormac Flanagan and Matthias Felleisen. Rice University Department of Computer Science Technical Report TR94-239, Oct. 1994.
- [96] “The Semantics of Future”. Cormac Flanagan and Matthias Felleisen. Rice University Department of Computer Science Technical Report TR94-238, Feb. 1994.
- [97] “PLT MrSpidey: Static Debugger Manual”. Cormac Flanagan. March, 1997. Available at <http://www.plt-scheme.org/software/mrspidey/docs.html>.

## Patents

- [1] “Method and Apparatus For Automatically Inferring Annotations”. Cormac Flanagan and K. Rustan M. Leino. U.S. Patent 7,120,902, (2006).
- [2] “Method and Apparatus For Organizing Warning Messages”. Cormac Flanagan and K. Rustan M. Leino. U.S. Patent 6,978,443, (2005).
- [3] “Method and Apparatus For Verifying Data Local To A Single Thread”. Cormac Flanagan and Stephen N. Freund. U.S. Patent 6,817,009, (2004).
- [4] “System and Method for Dynamic Detecting Unchecked Error Condition Values in Computer Programs”. Cormac Flanagan and Mike Burrows. U.S. Patent 6,378,087, (2002).
- [5] “System and Method for Lexing and Parsing Program Annotations”. Raymond Paul Stata, Cormac Flanagan, K. Rustan M. Leino, Mark Lillibridge, and James B. Saxe. U.S. Patent 6,353,925, (2002).
- [6] “System and Method for Statically Detecting Poolevential Race Conditions in Multithreaded Computer Programs”. Cormac Flanagan and Andrew Bernard. U.S. Patent 6,353,371, (2002).

## Distributed software

- ROADRUNNER** A framework for writing dynamic analyses for concurrent Java programs, presented in publication [22] and used as research infrastructure for publications [1, 17, 20, 24, 25, 27, 30].  
Available from <http://www.cs.williams.edu/~freund/rr/>.
- SAGE** A language based on hybrid type checking [2, 36, 38].  
Available from <http://sage.soe.ucsc.edu/>.
- ESC/JAVA** Finds common errors by static analysis<sup>1</sup> [57, 62, 59].
- HOUDINI** Infers ESC/Java annotations<sup>1</sup> [61].
- RCC/JAVA** Statically detects race conditions<sup>1</sup> [6, 63].
- CALVIN** Verifies invariants in multithreaded Java programs<sup>1</sup> [7, 58, 56].
- JAVAFE** An extensible parser and type checker for Java<sup>1</sup>.

<sup>1</sup> Available from <http://www.hp1.hp.com/downloads/crl/jtk/> as part of the Java Programming Toolkit Source Release.

## Invited Talks

- 2011 Invited Speaker at the Second International Conference on Runtime Verification 2011: *Cooperative Concurrency for a Multicore World*
- 2011 Stanford Research Institute Seminar Speaker: *FastTrack: Efficient and Precise Dynamic Race Detection*
- 2010 UC Berkeley Par Lab Seminar Speaker: *FastTrack: Efficient and Precise Dynamic Race Detection*
- 2009 Stanford University Seminar Speaker: *Velodrome: A Sound and Complete Dynamic Atomicity Checker For Multithreaded Programs*
- 2009 HP Transactional Memory Workshop: *Velodrome: A Sound and Complete Dynamic Atomicity Checker For Multithreaded Programs*
- 2009 Dagstuhl Seminar on the Design and Validation of 11 Concurrent Systems: *Dynamic Checkers for Concurrent Software: FastTrack, Velodrome, SideTrack, SingleTrack*
- 2009 Dagstuhl Seminar on JML *Specifications for Reliable Concurrent Software: Atomicity and Determinism*
- 2007 Colloquium Speaker, Max Plank Institute for Software Systems, Saarbrucken, Germany.
- 2005 Distinguished Seminar Speaker, IBM T. J. Watson Research Center, Hawthorne, New York.  
Invited to the Workshop on Construction and Analysis of Safe, Secure and Interoperable Smart devices (CASSIS), Nice, France.
- 2004 Invited Speaker, Workshop on Parallel and Distributed Systems: Testing and Debugging (PADTAD), Santa Fe, New Mexico.  
Invited Speaker, Workshop on Constraint Programming and Constraints for Verification (CP+CV), Barcelona, Spain.
- 2003 Computer Science Colloquium, San Jose State University.
- 2002 Massachusetts Institute of Technology.  
University of California at Berkeley.

## Invited Tutorials

- 2006 “Static Analysis for Concurrency”, Summer School on Language-Based Techniques for Concurrent and Distributed Software. Eugene, Oregon, July 12-21, 2006.

## Other Tutorials (by submission)

- 2005 “Atomicity for Reliable Concurrent Software”, Conference on Programming Language Design and Implementation. Chicago, Illinois, June 11, 2005.

## External Professional Activities

- 2011 **Steering Committee Member:** European Joint Conference on Theory and Practice of Software (ETAPS)
- Steering Committee Chair:** Programming Languages meets Program Verification Workshop Series (PLPV)
- Co-Organizer:** CAPabilities, Contracts, Objects, and Membranes (CAPCOM) Workshop, Stanford University, June 13-14, 2011
- Program Committee: International Conference on Computer Aided Verification (CAV'11)
- Program Committee: International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI'11)
- Program Committee: Second International Conference on Runtime Verification (RV'11)
- Program Committee: Third NASA Formal Methods Symposium
- 2010 **Program Co-Chair:** The Fourth ACM SIGPLAN Workshop on Programming Languages meets Program Verification (PLPV'10)
- Member of Steering Committee:** International Workshop on Scripts to Program Evolution (STOP)
- Session Chair: ACM SIGPLAN Fifth Workshop on Programming Languages and Analysis for Security (PLAS'10)
- Program Committee: ACM Symposium on Principles of Programming Languages (POPL'10)
- Program Committee: European Symposium on Programming (ESOP'10)
- Program Committee: First International Conference on Runtime Verification (RV'10)
- Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging (PADTAD'10)
- External Review Committee: The ACM Conference on Programming Language Design and Implementation (PLDI'10)
- 2009 **Co-Organizer:** Dagstuhl Seminar on the Design and Validation of Concurrent Systems
- Session Chair: The ACM Conference on Programming Language Design and Implementation (PLDI'09)
- Reviewer for ACM SIGPLAN Outstanding Doctoral Dissertation Award.
- Proposal Reviewer: Czech Science Foundation Debugging.
- Member of Proposal Review Panel, National Science Foundation.
- Program Committee: First International Workshop on Script to Program Evolution

- Program Committee: International Symposium on the Implementation and Application of Functional Languages
- Program Committee: Tenth Symposium on Trends in Functional Programming (TFP'09)
- Program Committee: European Conference on Object-Oriented Programming (ECOOP'09)
- Program Committee: The Ninth Workshop on Runtime Verification (RV'09).
- Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging (PADTAD'09)
- 2008 Reviewer for ACM SIGPLAN Outstanding Doctoral Dissertation Award.  
Member of Proposal Review Panel, National Science Foundation.  
Proposal Reviewer: Netherlands Organisation for Scientific Research  
Program Committee: Workshop on Specification and Verification of Component Based Systems (SAVCBS'08).  
Program Committee: International Workshop on Foundations and Developments of Object-Oriented Languages (FOOL'08).  
Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging (PADTAD'08)  
Program Committee: The 2008 ACM SIGPLAN Workshop on ML.
- 2007 External Reviewer: ETH Zurich Research Commission.  
External Reviewer: UC Microelectronics Program.  
Program Committee: Workshop on Specification And Verification of Component-Based Systems (SAVCBS'07)  
Program Committee: The Seventh Workshop on Runtime Verification (RV'07).  
Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging (PADTAD'07)
- 2005-07 **Steering Committee Member:** ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering.
- 2006 Member of Proposal Review Panel, National Science Foundation.  
Posters Selection Committee: ACM SIGSOFT Symposium on Foundations of Software Engineering (FSE).  
External Reviewer: University of California Micro Program.  
External Reviewer: UC Microelectronics Program.  
Program Committee: Workshop on Memory Systems Correctness and Performance.  
Program Committee: Workshop on Formal Aspects of Testing and Runtime Verification.

- Program Committee: Workshop on Multithreading in Hardware and Software: Formal Approaches to Design and Verification.
- Program Committee: Workshop on Scheme and Functional Programming.
- Program Committee: The 13th International SPIN Workshop on Model Checking of Software.
- Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging.
- 2004/5 **Steering Committee Chair:** ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering.
- 2005 Program Committee: The ACM Conference on Programming Language Design and Implementation (PLDI).
- Program Committee: The ACM Workshop on Types in Language Design and Implementation (TLDI).
- Program Committee: The 12th International SPIN Workshop on Model Checking of Software.
- Program Committee: Workshop on Specification and Verification of Component-Based Systems.
- Program Committee: The Fifth Workshop on Runtime Verification (RV'05).
- Program Committee: Workshop on Evaluation of Software Defect Detection Tools.
- Program Committee: Workshop on Formal Techniques for Java-like Programs.
- Program Committee: The 3rd Workshop on Parallel and Distributed Systems: Testing and Debugging.
- 2004 **Co-Chair:** ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering.
- Program Committee: Fourth Workshop on Runtime Verification.
- 2003 Program Committee: The Thirtieth ACM Symposium on Principles of Programming Languages (POPL).
- 2002 Program Committee: The ACM Workshop on Program Analysis For Software Tools and Engineering (PASTE).
- 2001 Program Committee: The ACM Conference on Programming Language Design and Implementation (PLDI).

## Reviewer of Technical Papers

- 2011 Journal of Functional Programming (JFP); Principles of Programming Languages (POPL).
- 2010 ACM Transactions on Programming Languages and Systems (TOPLAS); Logical Methods in Computer Science.
- 2009 ACM Transactions on Programming Languages and Systems (TOPLAS); International Conference on Computer Aided Verification (CAV); International Conference on Object Oriented Programming, Systems, Languages and Applications (OOPSLA).
- 2008 ACM Transactions on Programming Languages and Systems (TOPLAS); International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA); Principles of Programming Languages (POPL); Programming Language Design and implementation (PLDI); and Formal Methods in System Design.
- 2007 ACM Transactions on Programming Languages and Systems (TOPLAS); ACM Transactions on Software Engineering and Methodology (TOSEM); ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA); Programming Language Design and Implementation (PLDI); International Conference on Functional Programming (ICFP); Logical Methods in Computer Science; 16th EACSL Annual Conference on Computer Science and Logic.
- 2006 ACM Transactions on Programming Languages and Systems (TOPLAS); International Journal on Software Tools for Technology Transfer (STTT); ACM Transactions on Software Engineering and Methodology (TOSEM); Computer-Aided Verification (CAV); Principles of Programming Languages (POPL); Programming Language Design and Implementation (PLDI).
- 2005 ACM Transactions on Programming Languages and Systems (TOPLAS); Transactions on Software Engineering (TSE); Science of Computer Programming (SCP); Journal of Object Technology (JOT); European Symposium on Programming (ESOP); Foundations of Object-Oriented Languages (FOOL); Principles and Practice of Parellel Programming (PPoPP); Principles of Programming Languages (POPL); Computer-Aided Verification (CAV); Tools and Algorithms for the Construction and Analysis of Systems (TACAS); International Conference of Functional Programming (ICFP).
- 2004 ACM Transactions on Programming Languages and Systems (TOPLAS); Journal of Computer Security (JCS); International Symposium on Software Testing and Analysis (ISSTA); Tools and Algorithms for the Construction and Analysis of Systems (TACAS); Static Analysis Symposium (SAS); Journal on Formal Aspects of Computing; International Conference of Functional Programming (ICFP).

## Student Advising and Supervision

### Adviser to Continuing Ph.D. Students

- Jaeheon Yi
- Thomas H. Austin
- Tim Disney
- Kenneth Knowles (currently on leave)

### Doctoral Dissertation Reading Committee Chair

- Jaeheon Yi. *Cooperability: A New Property for Multithreading*. UCSC, 8/2011.
- Aaron Tomb. *Program Inconsistency Detection: Universal Reachability Analysis and Conditional Slicing*. UCSC, 2/2011.

### Doctoral Dissertation Reading Committee Member

- Nicholas D. Matsakis. *Intervals: Data-race-free Parallel Programming*. **ETH Zurich**, 5/2011.
- Sean Halle. *A Study of Frameworks for Collectively Achieving the Productivity, Portability, and Adoptability Goals of Parallel Software*. UCSC, 6/2011.
- Vishwanath Raman. *Game Relations, Metrics, and Refinements*. UCSC, 6/2010.
- Pritam Roy. *Interval-Based Abstraction Refinement* UCSC, 12/2009.
- Avik Chadhuri. *Foundations of Access Control for Secure Storage*. UCSC, 10/2008.
- Nathan Whitehead. *Combining Reason and Authority for Code Authentication and Verification*. UCSC, 3/2008.
- Andrew Edward Santosa. *A Framework for Program Reasoning Based on Constraint Traces*. **National University of Singapore**, 3/2008.
- Jennifer Bevan. *Software Instabilities: Identification, Analysis, and Visualization*. UCSC, 10/2006.
- Sung Kim. *Adaptive Bug Prediction by Analyzing Project History*. UCSC, 8/2006.
- Philippe Meunier. *Modular Set-Based Analysis from Contracts*. **Northeastern University**, 5/2006.

### **Doctoral Qualifying Exam Committee Member**

- Andrey Chudnov. *Information Flow Monitor Inlining for JavaScript*. **Stevens Institute of Technology**, 2/2011.
- Thomas Austin. *Dynamic Information Flow Analysis for JavaScript in a Web Browser*. 6/2010.
- Caitlin Sadowski. *Precise Dynamic Concurrency Error Prediction*. 5/2010.
- Arjun Guha. *Static Types for JavaScript: Theory and Practice*. **Brown University**, 10/2009.
- Alamelu Sankaranarayanan. *GPSim: A Framework for Modeling Current and Future Manycore Architectures*. 8/2009.
- Kenneth Knowles. *Semantic Design for Concurrent Functional Data Structures*. 3/2009.
- Jaeheon Yi. *Dynamic Analysis of Large-Scale Programs*. 12/2008.
- Avik Chadhuri. *Foundations of Access Control for Secure Storage*. 1/2007.
- Aaron Tomb. *Hybrid Verification of Object-Oriented Programs*. 5/2006.
- Nathan Whitehead. *Combining Reason and Authority for Code Authentication and Verification*. 10/2005.
- Kai Pan. *An Investigation of Program Slice Encoding and Its Applications*. 11/2004.
- Sung Kim. *Semantic API Framework*. 11/2004.
- Jennifer Bevan. *Software Instabilities: Identification, Analysis, and Visualization*. 10/2003.

### **Masters of Science Thesis Reading Committee Member**

- Daniel Libicki. *The GLIB Programming Language*. 3/2006.
- Spencer Tu. *EOS: A System for Evaluateable Objects in Scheme*. 3/2005.
- Michael K. Baker. *Object-Oriented Change Analysis*. 6/2004.

### **Masters of Science Project Reading Committee Member**

- Suchit Agarwal. *Distributed Hash Tables in Large Scale LANs*. 4/2010.

## **University Service**

### **Campus Service**

2010–2011 Member of the Committee for Educational Policy

2009–2010 Member of the Committee for Educational Policy

### **School of Engineering Service**

2009–2010 Member of the CAP Ad hoc Committee

### **Service in the Department of Computer Science**

2008–11 Graduate Director

2008–09 Member of the Faculty Recruitment Committee

2007–08 Assistant Graduate Director

2004– Member of the Graduate Committee

2003–04 Member of the Faculty Recruitment Committee

### **At Rice University**

1994–95 Vice President, Rice Graduate Student Association

1993–94 Member of the Honor Council

1992–93 Computer Science Rep. for the Graduate Student Association