

Cormac Flanagan

Professor
Department of Computer Science and Engineering
University of California, Santa Cruz

August 12, 2020

Employment History

- 2018– *Full Professor of Computer Science and Engineering, UCSC*
- 2009–18 *Full Professor of Computer Science, UCSC*
- 2005–09 *Associate Professor of Computer Science, 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, Dublin, Ireland*

Education

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

Honors and Awards

- 2020 CSF Distinguished Paper Award for publication [C24].
“Transparent IFC Enforcement: Possibility and (In)Efficiency Results”. Maximilian Alghed and Cormac Flanagan. *Proceedings of the 33rd IEEE Computer Security Foundations Symposium (CSF), June 2020.*
- 2019 PLDI Most Influential Paper Award, a 10-year retrospective test-of-time award, for publication [C52].
“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.*

- 2017 PLDI Distinguished Artifact Award for the software accompanying [C32].
 “BigFoot: Static Check Placement for Dynamic Race Detection”. Dustin Rhodes, Cormac Flanagan, and Stephen N. Freund. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation* (PLDI), June 2017.
- 2016 Certificate of Merit from the Crown College Graduating Class of 2016 for “time and dedication to the students of Crown College”.
- 2015 Excellence in Teaching Award from the Senate Committee on Teaching for “absolute dedication to making learning a positive experience.”
- 2014 Certificate of Merit from the Crown College Graduating Class of 2014.
- 2014 Nominated for Senate Committee on Teaching Excellence in Teaching Award.
- 2013 ECOOP Best Paper Award for publication [C41].
 “RedCard: Redundant Check Elimination for Dynamic Race Detectors”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the European Conference on Object-Oriented Programming* (ECOOP), July 2013.
- 2012 PLDI Most Influential Paper Award, a 10-year retrospective test-of-time award, for publication [C69].
 “Extended Static Checking for Java”. Cormac Flanagan, K. Rustan M. Leino, Mark Lillibridge, Greg Nelson, James B. Saxe, and Raymie Stata. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation* (PLDI), June 2002.
- 2010 Theoretical Computer Science Top Cited Article Award 2005-10 for [J17].
 “Modular Verification of Multithreaded Programs”. Cormac Flanagan, Stephen N. Freund, Sanjit Seshia, and Shaz Qadeer. *Theoretical Computer Science* 338 (1-3), 2005.
- 2004 ISSTA Distinguished Paper Award for publication [C63].
 “Exploiting Purity for Atomicity”. Cormac Flanagan, Stephen N. Freund, and Shaz Qadeer. *Proceedings of the ACM International Symposium on Software Testing and Analysis* (ISSTA), June 2004.
- 2003 Publication [C84] selected as one of the 50 most influential contributions in the last twenty years of PLDI.
 “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), June 1993, 237–247. Also appeared in SIGPLAN Notices 28, 6 (June 1993).
- 1986 First place in the Irish Mathematics Examination taken by all Irish students at the end of high-school.

Research Grants and Gifts

- 2018–21 PI, NSF award CCF-1813133. “SHF: Small: Collaborative Research: Synchronicity: A Framework for Synthesizing Concurrent Software from Sequential and Cooperative Specifications”. \$300,000.
- 2014–17 PI, NSF award CCF-1421016. “SHF: Small: Collaborative Research: Fast and Precise Dynamic Race Detection: Eliminating State and Checking Redundancy”. \$301,004.
- 2013–17 Co-PI, NSF award CCF-1337278. “XPS: FP: DeCoP: Deterministic Cooperative Parallelism”. \$750,000 total, with 30% for my research group.
- 2014 PI, Mozilla Research Award, “Hygienic Macros for JavaScript”. \$53,167.
- 2013 PI, Mozilla Research Award, “Macros for JavaScript”. \$34,000.
- 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.
- 2005–09 **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 Scholarship in Computer Science, University College Dublin.
- 1988 Scholarship in Computer Science, University College Dublin.
- 1987 Scholarship in Computer Science, University College Dublin.

Scholarly and Creative Work

Edited Collections

- [E1] Logical Methods in Computer Science: Special Issue on Selected papers of the conference: Tools and Algorithms for the Construction and Analysis of Systems. DOI: 10.2168/LMCS-TACAS:2012. Cormac Flanagan and Barbara König (Editors).
- [E2] International Journal on Tools for Technology Transfer, 16 (2), April 2014. ISBN 1433-2779. Cormac Flanagan and Barbara König (Editors).
- [E3] ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI '13, Seattle, WA, USA, June 16-19, 2013. ISBN 978-1-4503-2014-6. Hans-Juergen Boehm and Cormac Flanagan (Editors).
Also published as ACM SIGPLAN Notices, 48, 6, June 2013.
- [E4] Tools and Algorithms for the Construction and Analysis of Systems – 18th International Conference, TACAS 2012, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2012, Tallinn, Estonia, March 24 – April 1, 2012. Proceedings. Lecture Notes in Computer Science 7214, Springer 2012, ISBN 978-3-642-28755-8. Cormac Flanagan and Barbara König (Editors).
- [E5] Design and Validation of Concurrent Systems. Cormac Flanagan, Susanne Graf, Madhusudan Parthasarathy, and Shaz Qadeer. September 2009. Dagstuhl Seminar Proceedings 09361.
- [E6] Proceedings of the 4th ACM Workshop Programming Languages meets Program Verification (PLPV), June 2010. Jean-Christophe Filliâtre and Cormac Flanagan.
- [E7] Proceedings of the 2004 ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE), 2004. Cormac Flanagan and Andreas Zeller (Editors).

Journal Articles (all peer-reviewed)

- [J8] “Multiple Facets for Dynamic Information Flow with Exceptions”. Thomas H. Austin, Thomas Schmitz, and Cormac Flanagan. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 39, 3 (July 2017).
- [J9] “Cooperative Types for Controlling Thread Interference in Java”. Jaeheon Yi, Tim Disney, Stephen N. Freund, and Cormac Flanagan. *Science of Computer Programming (SCP)*, 112, 1 (Nov 2015), 227–260.
- [J10] “Space Efficient Gradual Typing”. David Herman, Aaron Tomb, and Cormac Flanagan. *Higher-Order and Symbolic Computation*, 23, 2, (June 2010), 167–189.

- [J11] “FastTrack: Efficient and Precise Dynamic Race Detection”. Cormac Flanagan and Stephen N. Freund. *Communications of the ACM (CACM)*, 53, 11 (November 2010).
- [J12] “Hybrid Type Checking”. Kenneth Knowles and Cormac Flanagan. *ACM Transactions on Programming Languages and Systems*, 32, 2 (January 2010), 1–34.
- [J13] “Atomizer: A Dynamic Atomicity Checker for Multithreaded Programs”. Cormac Flanagan and Stephen N. Freund. *Science of Computer Programming*, 71, 2 (April 2008), 89–109.
- [J14] “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.
- [J15] “Type Inference Against Races”. Cormac Flanagan and Stephen N. Freund. *Science of Computer Programming*, 64, 1 (September 2006), 140–165.
- [J16] “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.
- [J17] “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.
- [J18] “Exploiting Purity for Atomicity”. Cormac Flanagan, Stephen N. Freund, and Shaz Qadeer. *IEEE Transactions on Software Engineering*, 31, 4 (April 2005), 275–291.
- [J19] “Automatic Software Model Checking via Constraint Logic”. Cormac Flanagan. *Science of Computer Programming*, 50, 1 (March 2004), 253–270.
- [J20] “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.
- [J21] “Annotation Inference for Modular Checkers”. Cormac Flanagan, Rajeev Joshi, and K. Rustan M. Leino. *Information Processing Letters*, 77, 2–4 (February 2001), 97–108.
- [J22] “Componential Set-Based Analysis”. Cormac Flanagan and Matthias Felleisen. *ACM Transactions on Programming Languages and Systems*, 21, 2 (March 1999) 370–416.
- [J23] “The Semantics of Future and an Application”. Cormac Flanagan and Matthias Felleisen. *Journal of Functional Programming*, 9, 1 (January 1999), 1–31.

Papers in Conference Proceedings (all peer-reviewed)

- [C24] “Transparent IFC Enforcement: Possibility and (In)Efficiency Results”. Maximilian Algehed and Cormac Flanagan. Proceedings of the 33rd IEEE Computer Security Foundations Symposium (CSF), June 2020.
CSF 2020 Distinguished Paper Award.
- [C25] “Optimizing Faceted Secure Multi-Execution”. Maximilian Algehed, Alejandro Russo, and Cormac Flanagan. Proceedings of the 32nd IEEE Computer Security Foundations Symposium (CSF), June 2019.
- [C26] “IDVE: An Integrated Development and Verification Environment for JavaScript”. Christopher Schuster and Cormac Flanagan. Conference Companion of the 3rd International Conference on Art, Science, and Engineering of Programming, Genova, Italy, April 1-4, 2019.
- [C27] “Secure serverless computing using dynamic information flow control”. Kaley Alpernas, Cormac Flanagan, Sadjad Fouladi, Leonid Ryzhyk, Mooly Sagiv, Thomas Schmitz, Keith Winstein. *Proceedings of the ACM on Programming Languages* (OOPSLA), October 2018, 118:1–118:26.
- [C28] “Faceted Secure Multi Execution”. Thomas Schmitz, Maximilian Algehed, Cormac Flanagan, Alejandro Russo. Proceedings of the ACM Conference on Computer and Communications Security (CCS), November 2018, 1617-1634.
- [C29] “ESVERIFY: Verifying Dynamically-Typed Higher-Order Functional Programs by SMT Solving”. Christopher Schuster, Sohun Banerjea, Cormac Flanagan. Proceedings of the 30th Symposium on Implementation and Application of Functional Languages (IFL), September 2018, 59–80.
- [C30] “A Better Facet of Dynamic Information Flow Control”. Minh Ngo, Natalia Bielova, Cormac Flanagan, Tamara Rezk, Alejandro Russo, Tommy Schmitz. WWW ’18 Companion: The 2018 Web Conference Companion, Apr 2018, Lyon, France. pp.1-9.
- [C31] “VerifiedFT: a verified, high-performance precise dynamic race detector”. James R. Wilcox, Cormac Flanagan, and Stephen N. Freund. *Proceedings of the ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming* (PPOPP), February 2018, 354-367.
- [C32] “BigFoot: Static Check Placement for Dynamic Race Detection”. Dustin Rhodes, Cormac Flanagan, and Stephen N. Freund. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation* (PLDI), June 2017.
PLDI 2017 Distinguished Artifact Award.
- [C33] “Precise, Dynamic Information Flow for Database-Backed Applications”. Jean Yang, Travis Hance, Thomas H. Austin, Armando Solar-Lezama, Cormac Flanagan, and Stephen Chong. *Proceedings of the ACM SIGPLAN*

Conference on Programming Language Design and Implementation (PLDI), June 2016.

- [C34] “Macrofication: Refactoring by Reverse Macro Expansion”. Christopher Schuster, Tim Disney, and Cormac Flanagan. *25th European Symposium on Programming (ESOP)*, April 2016.
- [C35] “Faceted Dynamic Information Flow via Control and Data Monads”. Thomas Schmitz, Dustin Rhodes, Thomas H. Austin, Kenneth Knowles, and Cormac Flanagan. *Principles of Security and Trust - 5th International Conference (POST)*, April 2016.
- [C36] “Reactive Programming With Reactive Variables”. Christopher Schuster and Cormac Flanagan. *Proceedings of the 15th International Conference on Modularity*, April 2016.
- [C37] “Array Shadow State Compression for Precise Dynamic Race Detection”. James R. Wilcox, Parker Finch, Cormac Flanagan, and Stephen N. Freund. *30th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, November 2015.
- [C38] “Game Semantics for Type Soundness”. Tim Disney and Cormac Flanagan. *30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, July 2015.
- [C39] “Dynamic Detection of Object Capability Violations Through Model Checking”. Dustin Rhodes, Tim Disney, and Cormac Flanagan. *Proceedings of the Dynamic Languages Symposium (DLS)*, October 2014.
- [C40] “Sweeten Your JavaScript: Hygienic Macros for ES5”. Tim Disney, Nathan Faubion, David Herman, and Cormac Flanagan. *Proceedings of the Dynamic Languages Symposium (DLS)*, October 2014.
- [C41] “RedCard: Redundant Check Elimination for Dynamic Race Detectors”. Cormac Flanagan and Stephen N. Freund. *Proceedings of the European Conference on Object-Oriented Programming (ECOOP)*, July 2013, 255–280.
ECOOP 2013 Best Paper Award.
- [C42] “A Functional View of Imperative Information Flow”. Thomas H. Austin, Cormac Flanagan, and Martin Abadi. *10th Asian Symposium on Programming Languages and Systems (APLAS)*, December 2012, 34–49.
- [C43] “Cooperative Types for Controlling Thread Interference in Java”. Jaeheon Yi, Tim Disney, Stephen N. Freund, and Cormac Flanagan. *International Symposium on Software Testing and Analysis (ISSTA)*, July 2012, 232–242.
- [C44] “Detecting Inconsistencies via Universal Reachability Analysis”. Aaron Tomb and Cormac Flanagan. *International Symposium on Software Testing and Analysis (ISSTA)*, July 2012, 287–297.

- [C45] “Multiple Facets for Dynamic Information Flow”. Thomas Austin and Cormac Flanagan. *Proceedings of the 38th ACM Symposium on Principles of Programming Languages (POPL)*, January 2012, 165–178.
- [C46] “Sound Predictive Race Detection in Polynomial Time”. Yannis Smaragdakis, Jacob M. Evans, Caitlin Sadowski, Jaeheon Yi, and Cormac Flanagan. *Proceedings of the 38th ACM Symposium on Principles of Programming Languages (POPL)*, January 2012, 387–400.
- [C47] “Virtual Values for Language Extension”. Tim Disney, Tom Austin, and Cormac Flanagan. *Object Oriented Programming, Systems, Languages, and Applications (OOPSLA)*, Oct 2011, 921–938.
- [C48] “Temporal Higher-Order Contracts”. Tim Disney, Cormac Flanagan, Jay McCarthy. *Proceedings of the 16th ACM SIGPLAN International Conference on Functional Programming (ICFP)*, Sept 2011.
- [C49] “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.
- [C50] “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.
- [C51] “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.
- [C52] “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.
- [C53] “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.
- [C54] “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.

- [C55] “Space Efficient Gradual Typing”. David Herman, Aaron Tomb, and Cormac Flanagan. *Eighth Symposium on Trends in Functional Programming*, (April 2007).
- [C56] “Unifying Hybrid Types and Contracts”. Jessica Gronski and Cormac Flanagan. *Eighth Symposium on Trends in Functional Programming*, (April 2007).
- [C57] “Type Reconstruction for General Refinement Types”. Kenneth Knowles and Cormac Flanagan. *16th European Symposium on Programming (ESOP)*, (March 2007), Springer-Verlag, 505–519.
- [C58] “Hybrid Type Checking”. Cormac Flanagan. *Proceedings of the 33rd ACM Symposium on Principles of Programming Languages* (January 2006), 245–256.
- [C59] “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).
- [C60] “Automatic Type Inference via Partial Evaluation”. Aaron Tomb and Cormac Flanagan. *Principles and Practice of Declarative Programming* (July 2005).
- [C61] “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).
- [C62] “Type Inference Against Races”. Cormac Flanagan and Stephen N. Freund. *Static Analysis, 11th International Symposium, SAS 2004*, Springer-Verlag (August 2004).
- [C63] “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.
- [C64] “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.
- [C65] “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.

- [C66] “A Type and Effect System for Atomicity”. Cormac Flanagan and Shaz Qadeer. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, June 2003, 338–349.
- [C67] “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), 189–3.
- [C68] “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), 180–4.
- [C69] “Extended Static Checking for Java”. Cormac Flanagan, K. Rustan M. Leino, Mark Lillibridge, Greg Nelson, James B. Saxe, and Raymie Stata. *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, June 2002, 234–245. Also appeared as SIGPLAN Notices 37, 5, June 2002.
- [C70] “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.
- [C71] “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), 191–2.
- [C72] “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.
- [C73] “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.
- [C74] “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.
- [C75] “Type-Based Race Detection for Java (summary)”. Cormac Flanagan and Stephen N. Freund. *Short topic presentation at the Fifteenth Annual IEEE Symposium on Logic in Computer Science, 2000*.

- [C76] “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.
- [C77] “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.
- [C78] “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).
- [C79] “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.
- [C80] “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.
- [C81] “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.
- [C82] “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.
- [C83] “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.
- [C84] “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.

Papers in Workshop Proceedings (all peer-reviewed)

- [W85] “Using Precise Taint Tracking for Auto-sanitization”. Tejas Saoji, Thomas H. Austin, and Cormac Flanagan. *Proceedings of the ACM SIGPLAN Workshop on Programming Languages and Analysis for Security (PLAS)*, October 2017.
- [W86] “Correctness of Partial Escape Analysis for Multithreaded Optimization”. Dustin Rhodes, Stephen N. Freund, and Cormac Flanagan. *Workshop on Formal Techniques for Java-like Programs (FTFJP)*, June 2017.
- [W87] “Live Programming by Example: Using Direct Manipulation for Live Program Synthesis”. Christopher Schuster and Cormac Flanagan. *Workshop on Live Programming Systems (LIVE)*, July 2016.
- [W88] “Live Programming for Event-Based Languages”. Christopher Schuster and Cormac Flanagan. *Workshop on Reactive and Event-based Languages & Systems*, October 2015
- [W89] “Contracts for Async Patterns in JavaScript”. Tim Disney and Cormac Flanagan. *Workshop on Script to Program Evolution (STOP)*, July 2015
- [W90] “A Lightweight Effect System for Javascript”. Christopher Schuster and Cormac Flanagan. *Proceedings of the 2015 Scripts to Programs Workshop (STOP)*, July 2015
- [W91] “Traveling through Time and Code: Omniscient Debugging and Beyond”. Christopher Schuster and Cormac Flanagan. *Future Programming Workshop*, October 2014.
- [W92] “Faceted Execution of Policy-Agnostic Programs”. Thomas H. Austin, Jean Yang, Cormac Flanagan and Armando Solar-Lezama. *Proceedings of the ACM SIGPLAN Eighth Workshop on Programming Languages and Analysis for Security (PLAS)*, June 2013, 15–26.
- [W93] “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.
- [W94] “Gradual Information Flow Typing”. Tim Disney and Cormac Flanagan. *Workshop on Script to Program Evolution (STOP)*, (January 2011).
- [W95] “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).
- [W96] “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.

- [W97] “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.
- [W98] “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).
- [W99] “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).
- [W100] “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).
- [W101] “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).
- [W102] “Status Report: Specifying JavaScript with ML”. David Herman and Cormac Flanagan. *Proceedings of the ACM Workshop on ML*, (October 2007), 47–52.
- [W103] “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.
- [W104] “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).
- [W105] “Dynamic Architecture Extraction”. Cormac Flanagan and Stephen N. Freund. *Workshop on Formal Approaches to Testing and Runtime Verification* (August 2006), 209–224.
- [W106] “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).
- [W107] “Automatic Synchronization Correction”. Cormac Flanagan and Stephen N. Freund. *Workshop on Synchronization and Concurrency in Object-Oriented Languages* (October 2005).

- [W108] “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).
- [W109] “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.
- [W110] “Software Model Checking via Iterative Abstraction Refinement Constraint Logic Queries”. Cormac Flanagan. *Workshop on Constraint Programming and Constraints for Verification*, (April 2004).
- [W111] “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.
- [W112] “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.
- [W113] “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.
- [W114] “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.

Invited Papers

- [115] “Cooperative Concurrency for a Multicore World (Extended Abstract)”. Jaeheon Yi, Caitlin Sadowski, Stephen Freund and Cormac Flanagan. *Second International Conference on Runtime Verification (RV)*, September 2011.
- [116] “Futures”. Cormac Flanagan. Article in *Encyclopedia of Parallel Computing*, Springer 2011.
- [117] “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 [C64].

Other Articles

- [118] “Practices of PLDI”. Hans Boehm, Jack Davidson, Kathleen Fisher, Cormac Flanagan, Jeremy Gibbons, Mary Hall, Graham Hutton, David Padua, Frank

- Tip, Jan Vitek, and Philip Wadler. *ACM SIGPLAN Notices*, 49, 4, April 2014, 33–38.
- [119] “Developments in Automated Verification Techniques”. Cormac Flanagan and Barbara König. *International Journal on Tools for Technology Transfer*, 16, 2 (April 2014), 123–125
- [120] “Extended Static Checking for Java”. Cormac Flanagan, K. Rustan M. Leino, Mark Lillibridge, Greg Nelson, James B. Saxe, and Raymie Stata. *SIGPLAN Notices* 48, 4S, April 2013, 22–33.
This paper is a reprint of [C69] commemorating its selection as the most influential paper from PLDI 2002.
- [121] “Introduction to the Workshop on Programming Languages Meets Program Verification”. Jean-Christophe Filliâtre and Cormac Flanagan. *ACM SIGPLAN Notices*, 44, 11, Nov 2009.
- [122] “Type-Based Race Detection for Java (summary)”. Cormac Flanagan and Stephen N. Freund. *15th Annual IEEE Symposium on Logic in Computer Science*, June 2000.

Technical Reports

- [T123] “FacetBook”. Thomas Schmitz and Cormac Flanagan. Technical Report UCSC-SOE-19-03, 2019.
- [T124] “BigFoot: Static Check Placement for Dynamic Race Detection”. Dustin Rhodes, Cormac Flanagan, and Stephen N. Freund. Williams College Technical Report CSTR-201702, 2017. This paper is an extended version of [C32].
- [T125] “The FASTTRACK2 race detector”. Cormac Flanagan and Stephen N. Freund. Williams College Technical Report CSTR-201701, 2017. This paper is a preliminary version of [C31].
- [T126] “Array Shadow State Compression for Precise Dynamic Race Detection”. James Wilcox, Parker Finch, Cormac Flanagan, and Stephen N. Freund. Williams College Technical Report CSTR-201510, 2015. This paper is an extended version of [C37].
- [T127] “Faceted Dynamic Information Flow Via Control And Data Monads”. Thomas Schmitz, Dustin Rhodes, Thomas H. Austin, Kenneth Knowles, and Cormac Flanagan. Technical Report UCSC-SOE-16-01, 2016. This paper is an extended version of [C35].
- [T128] “Game Semantics for Type Soundness”. Tim Disney and Cormac Flanagan. Technical Report UCSC-SOE-15-09, 2015. This paper is an extended version of [C38].

- [T129] “Union and Intersection Contracts”. Tim Disney and Cormac Flanagan. Technical Report UCSC-SOE-15-01, 2015.
- [T130] “Traveling Through Time And Code: Omniscient Debugging And Beyond”. Christopher Schuster and Cormac Flanagan. Technical Report UCSC-SOE-14-15, 2014. This paper is an extended version of [W91].
- [T131] “Typed Faceted Values for Secure Information Flow in Haskell”. Thomas H. Austin, Kenneth Knowles, and Cormac Flanagan. Technical Report UCSC-SOE-14-07, 2014.
- [T132] “RedCard: Redundant Check Elimination For Dynamic Race Detectors”. Cormac Flanagan and Stephen N. Freund. Technical Report UCSC-SOE-13-05, 2013. This paper is an extended version of [C41].
- [T133] “Faceted Execution of Policy-Agnostic Programs, Extended Version”. Thomas H. Austin, Jean Yang, Cormac Flanagan, and Armando Solar-Lezama. Technical Report UCSC-SOE-12-18, 2012. This paper is an extended version of [W92].
- [T134] “A Functional View of Imperative Information Flow”. Thomas H. Austin, Cormac Flanagan, Martin Abadi. Technical Report UCSC-SOE-12-15, 2012. This paper is an extended version of [C42].
- [T135] “Types for Precise Thread Interference”. Jaeheon Yi, Tim Disney, Stephen N. Freund, Cormac Flanagan . Technical Report UCSC-SOE-11-22, 2011.
- [T136] “Dynamic Information Flow Analysis for Featherweight JavaScript”. Thomas H. Austin, Tim Disney, Cormac Flanagan, Alan Jeffrey. Technical Report UCSC-SOE-11-19, 2011.
- [T137] “Virtual Values for Language Extension”. Thomas H. Austin, Tim Disney, and Cormac Flanagan. Technical Report UCSC-SOE-10-32, 2010.
- [T138] “Permissive Dynamic Information Flow Analysis”. Thomas H. Austin and Cormac Flanagan. Technical Report UCSC-SOE-09-34, Nov. 2009.
- [T139] “Compositional and Decidable Checking for Dependent Contract Types”. Kenneth Knowles and Cormac Flanagan. Technical Report UCSC-SOE-08-17, Aug. 2008.
- [T140] “Modular Verification of Multithreaded Programs”. Cormac Flanagan, Stephen N. Freund, Shaz Qadeer, and Sanjit A. Seshia. Williams College Technical Note 04-08, Nov. 2004.
- [T141] “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.

- [T142] “Type Inference Against Races (extended version)”. Cormac Flanagan and Stephen N. Freund. Williams College Technical Note 04-06, Sept. 2004.
- [T143] “Exploiting Purity for Atomicity (extended version)”. Cormac Flanagan, Stephen N. Freund, and Shaz Qadeer. Williams College Technical Note 04-05, July 2004.
- [T144] “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.
- [T145] “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.
- [T146] “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.
- [T147] “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.
- [T148] “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.
- [T149] “Well-Founded Touch Optimization of Futures”. Cormac Flanagan and Matthias Felleisen. Rice University Department of Computer Science Technical Report TR94-239, Oct. 1994.
- [T150] “The Semantics of Future”. Cormac Flanagan and Matthias Felleisen. Rice University Department of Computer Science Technical Report TR94-238, Feb. 1994.
- [T151] “PLT MrSpidey: Static Debugger Manual”. Cormac Flanagan. March, 1997. Available at <http://www.plt-scheme.org/software/mrspidey/-docs.html>.

Dissertation

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

Patents

- [P153] “Method and Apparatus For Automatically Inferring Annotations”. Cormac Flanagan and K. Rustan M. Leino. U.S. Patent 7,120,902, (2006).

- [P154] “Method and Apparatus For Organizing Warning Messages”. Cormac Flanagan and K. Rustan M. Leino. U.S. Patent 6,978,443, (2005).
- [P155] “Method and Apparatus For Verifying Data Local To A Single Thread”. Cormac Flanagan and Stephen N. Freund. U.S. Patent 6,817,009, (2004).
- [P156] “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).
- [P157] “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).
- [P158] “System and Method for Statically Detecting Poevential Race Conditions in Multithreaded Computer Programs”. Cormac Flanagan and Andrew Bernard. U.S. Patent 6,353,371, (2002).

Distributed software

- [Sw159] BIGFOOT SOFTWARE ARTIFACT.
The software system underlying PLDI paper [C32].
<https://users.soe.ucsc.edu/~cormac/bigfoot-artifact>.
2017 PLDI Distinguished Artifact Award.
- [Sw160] ROADRUNNER: A framework for writing dynamic analyses for concurrent Java programs, presented in publication [W96] and used as research infrastructure for [J11, C32, C37, C41, C46, C49, C51, C52, C53, C54, W98, T125, T126, T132].
Updated and extended during this review period.
<https://users.soe.ucsc.edu/~cormac/RoadRunner>.
- [Sw161] SAGE: A language based on hybrid type checking [J12, C58, W104].
Available from <http://sage.soe.ucsc.edu>.
- [Sw162] ESC/JAVA: Finds common errors by static analysis¹ [C69, C73, C71].
Recieved the PLDI 2012 Most Influential Paper Award.
- [Sw163] HOUDINI: Infers ESC/Java annotations¹ [C72].
- [Sw164] RCC/JAVA: Statically detects race conditions¹ [J16, C74].
- [Sw165] CALVIN: Verifies multithreaded Java programs¹ [J17, C70, C68, T140].
- [Sw166] JAVAFAE: An extensible parser and type checker for Java, and the front-end for ESC/JAVA¹.

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

Professional Activity

Leadership Positions in Professional Organizations

- 2016–19 Chair of Steering Committee: The ACM Conference on Programming Language Design and Implementation (PLDI).
- 2014– Editorial Board: PeerJ Computer Science.
- 2013–20 Steering Committee: The ACM Conference on Programming Language Design and Implementation (PLDI).
- 2012– Editorial Board: Foundations and Trends in Programming Languages.
- 2010– Steering Committee: International Workshop on Scripts to Program Evolution (STOP).
- 2014–17 Editorial Board: Scientific Programming.
- 2013–16 Associate Editor: ACM Transactions on Programming Languages and Systems (TOPLAS).
- 2015 Co-Organizer: Bay Area Programming Languages Seminar
- 2011–15 Co-Chair of Steering Committee: Programming Languages meets Program Verification Workshop Series (PLPV).
- 2013 Program Chair: The ACM Conference on Programming Language Design and Implementation (PLDI).
- 2012 Co-Chair: Tools and Algorithms for the Construction and Analysis of Systems (TACAS).
- 2011 Steering Committee: European Joint Conference on Theory and Practice of Software (ETAPS).
- 2011 Co-Organizer: Capabilities, Contracts, Objects, and Membranes Workshop, Stanford University, June 13-14, 2011.
- 2010 Program Co-Chair: The Fourth ACM SIGPLAN Workshop on Programming Languages meets Program Verification (PLPV).
- 2009 Co-Organizer: Dagstuhl Seminar on the Design and Validation of Concurrent Systems.
- 2004-07 Steering Committee: ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE).
- 2004-05 Chair of Steering Committee: ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE).
- 2004 Co-Chair: ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE).

Tutorials

- 2014 “Analysis Techniques to Detect Concurrency Errors”, UPMARC Summer School on Multicore Computing, Uppsala, Sweden, July 28–29, 2014. (Invited tutorial).
- 2006 “Static Analysis for Concurrency”, Summer School on Language-Based Techniques for Concurrent and Distributed Software. Eugene, Oregon, July 12-21, 2006. (Invited tutorial).
- 2005 “Atomicity for Reliable Concurrent Software”, Conference on Programming Language Design and Implementation. Chicago, Illinois, June 11, 2005.

Research Presentations

(excluding conference and workshop presentations of the papers listed above)

- 2019 **Keynote Speaker:** European Joint Conferences on Theory and Practice of Software (ETAPS). *Towards Efficient and Precise Concurrent Software Analysis*
- 2016 International Federation for Information Processing (IFIP) Working Group 2.3 on Programming Methodology Meeting. *RedCard: Redundant Check Elimination for Dynamic Race Detectors*
- 2015 International Federation for Information Processing (IFIP) Working Group 2.3 on Programming Methodology Meeting. *Yield-Oriented Programming*
- 2014 UCLA Computer Science Department Seminar: *Dynamic Analyses for Reliable Concurrency*
- 2014 **Keynote Speaker:** International Symposium on Software Testing and Analysis (ISSTA). *Dynamic Analyses for Reliable Concurrency*
Dagstuhl Workshop on Scripting Languages and Frameworks. *Multiple Facets for Dynamic Information Flow*
- 2013 International Federation for Information Processing (IFIP) Working Group 2.4 Software Implementation Technology Meeting. *RedCard: Redundant Check Elimination For Dynamic Race Detectors*
- 2012 Dagstuhl Workshop on Foundations of Scripting Languages. *Virtual Values for Language Extension*
- 2011 **Invited Speaker:** Second International Conference on Runtime Verification 2011. *Cooperative Concurrency for a Multicore World*
Capabilities, Contracts, Objects, and Membranes Workshop. *Temporal Higher-Order Contracts and Trace Semantics*
Stanford Research Institute, Seminar Speaker. *FastTrack: Efficient and Precise Dynamic Race Detection*
- 2010 UC Berkeley, 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*
 HP Transactional Memory Workshop. *Velodrome: A Sound and Complete Dynamic Atomicity Checker For Multithreaded Programs*
 Dagstuhl Seminar on the Design and Validation of Concurrent Systems: *Dynamic Checkers for Concurrent Software: FastTrack, Velodrome, SideTrack, SingleTrack*
 Dagstuhl Seminar on the Java Modeling Language. *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.

Program Committee Membership and other Reviewing Service

- 2020 Program Committee: 12th NASA Formal Methods Symposium (NFM)
- 2018 Program Committee: 7th International Conference on Principles of Security and Trust (POST)
- 2017 External Review Committee: The ACM Conference on Principles of Programming Languages (POPL)
- 2016 Distinguished Paper Committee: The ACM Conference on Programming Language Design and Implementation (PLDI)
 External Program Committee: The ACM Conference on Programming Language Design and Implementation (PLDI)
- 2015 External Program Committee: The ACM Conference on Programming Language Design and Implementation (PLDI)
- 2014 Program Committee: International Symposium on Engineering Secure Software and Systems (ESSOS)
 Program Committee: International Conference on Principles and Practice of Programming on the Java Platform: Virtual Machines, Languages, and Tools
 Program Committee: Workshop on Higher-Order Program Analysis (HOPA)

- Member of SIGPLAN Milner Award Selection Committee
- Member of Committee for PLDI 2004 Most Influential Paper Award (a ten-year retrospective award)
- 2013 Program Committee: Workshop on Scheme and Functional Programming
Facilitator: Facebook Faculty Summit session on Types and Code Evolution.
- 2012 Program Committee: The ACM Conference on Programming Language Design and Implementation (PLDI)
External Review Committee: The ACM Conference on Principles of Programming Languages (POPL)
External Review Committee: The ACM Conference on Principles and Practice of Parallel Programming (PPOPP)
External Review Committee: The ACM Conference on Object Oriented Programming, Languages, and Systems (OOPSLA)
Reviewer for Microsoft Research PhD Scholarship Programme
- 2011 Program Committee: International Conference on Computer Aided Verification (CAV)
Program Committee: International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI)
Program Committee: Second International Conference on Runtime Verification
Program Committee: Third NASA Formal Methods Symposium
Reviewer: Journal of Functional Programming (JFP); Principles of Programming Languages (POPL).
- 2010 Session Chair: ACM SIGPLAN Fifth Workshop on Programming Languages and Analysis for Security (PLAS)
Program Committee: ACM Symposium on Principles of Programming Languages (POPL)
Program Committee: European Symposium on Programming (ESOP)
Program Committee: First International Conference on Runtime Verification
Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging
External Review Committee: The ACM Conference on Programming Language Design and Implementation (PLDI)
Reviewer: ACM Transactions on Programming Languages and Systems (TOPLAS); Logical Methods in Computer Science (LMCS).
- 2009 Session Chair: The ACM Conference on Programming Language Design and Implementation (PLDI)
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
Program Committee: European Conference on Object-Oriented Programming (ECOOP)
Program Committee: The Ninth Workshop on Runtime Verification
Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging
Reviewer: 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 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
Program Committee: International Workshop on Foundations and Developments of Object-Oriented Languages (FOOL).
Program Committee: Workshop on Parallel and Distributed Systems: Testing and Debugging
Program Committee: The 2008 ACM SIGPLAN Workshop on ML.
Reviewer: 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 External Reviewer: ETH Zurich Research Commission.
External Reviewer: UC Microelectronics Program.
Program Committee: Workshop on Specification And Verification of Component-Based Systems
Program Committee: The Seventh Workshop on Runtime Verification
Program Committee: Workshop on Parallel and Distributed Systems: Testing

and Debugging

Reviewer: 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 (LMCS); 16th EACSL Annual Conference on Computer Science and Logic.

- 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.
- Reviewer:
- 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 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 Dustin Rhodes, Cormac Flanagan, and Stephen N. Freundniquies for Java-like Programs.

Program Committee: The 3rd Workshop on Parallel and Distributed Systems: Testing and Debugging.

Reviewer: 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 Program Committee: Fourth Workshop on Runtime Verification.

Reviewer: 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).

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).

Board Member

2008–14 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.

University and Public Service

Academic Senate Service

- 2017–18 Member of the Senate Committee on Planning and Budget
- 2016–17 Member of the Senate Committee on Planning and Budget
- 2015–16 Member of the Senate Committee on Planning and Budget
- 2010–11 Member of the Senate Committee for Educational Policy
- 2009–10 Member of the Senate Committee for Educational Policy

Service to the Baskin School of Engineering

- 2017–18 Member of the Review Team for the BSOE Reshaping Project
- 2013–14 Chair of Divisional Committee on Academic Personnel
- 2012–13 Member of Divisional Committee on Academic Personnel
- 2011–12 Member of Divisional Committee on Academic Personnel
- 2009–10 Member of CAP Ad hoc Committee

Service to Crown College

- 2015–18 Member of the Crown College Executive Committee

Service to the Department

- 2019–20 Co-Chair of the Personnel Committee
- 2018–19 Chair of the Faculty Recruitment Committee for Software Foundations
- 2017–18 Diversity Liaison on the Faculty Recruitment Committee for Distributed Systems
- 2017–18 Diversity Liaison on the Faculty Recruitment Committee for Data Science
- 2016–17 Chair of the Faculty Recruitment Committee for Big Data and Systems
- 2016–17 Chair of the Faculty Recruitment Committee for Data Driven Applications
- 2015–16 Member of the Distinguished Lecturer Series Committee
- 2015–16 Member of the Graduate Committee
- 2015–16 Member of the Planning Committee for Faculty Recruitment
- 2015–16 Member of Faculty Recruitment Committee
- 2014–15 Member of Faculty Recruitment Committee
- 2013–14 Member of Computer Engineering Faculty Recruitment Committee
- 2010–11 Graduate Director

2009–10 Graduate Director
2008–09 Graduate Director
2008–09 Member of the Faculty Recruitment Committee
2007–08 Assistant Graduate Director
2004–11 Member of the Graduate Committee
2003–04 Member of the Faculty Recruitment Committee

Other Service to the Campus

2013–14 Member of the UCSC Strategic Planning Task Force (Envision)

K–12 Outreach

2012 Invited talk at the Westlake Elementary Young Scientist Club

At Rice University

1994–95 Vice President, Rice Graduate Student Association
1993–94 Member of the Honor Council
1992–93 Computer Science Representative for the Graduate Student Association

Consulting

2006–12 Invited expert on the Ecma TC39 committee for standardization of the JavaScript programming language.
2006–12 Consultant for Mozilla Corporation.
2007–08 Consultant for Solidware Corporation.
2006 Consultant for Microsoft Corporation.