Zhong Shao

November 2024

Address

Department of Computer Science
Yale University
51 Prospect Street
New Haven, CT 06520-8285, USA
Tel: +1 203 432 6828 Fax: +1 203 432 0593
Email: zhong.shao@yale.edu
URL: http://www.cs.yale.edu/homes/shao

Interests

Programming languages and operating systems, with a focus on language-based support for safety and security, certified system software, certified programming and compilation, formal methods and proof automation, concurrency and coordination, and type systems.

Education

Ph.D. in Computer Science, Princeton University, September 1994.
Thesis title: *Compiling Standard ML for efficient execution on modern machines*.
Advisor: Professor Andrew W. Appel.

M.A. in Computer Science, Princeton University, May 1991.

B.S. in Computer Science, University of Science and Technology of China, July 1988.

Professional Experience

Yale University, Department of Computer Science. Thomas L. Kempner Professor of Computer Science, since 2017; Department Chair, 2017–2023; Professor, since 2003; Associate Professor, 2000–2003; Assistant Professor, 1994–2000.

Co-Founder, CertiK, since 2017.

Microsoft Research, Redmond, WA. Summer 2008, Visiting Researcher.

DoCoMo Communications Laboratories USA, Palo Alto, CA. December 2007– August 2008, Consultant.

Bell Laboratories, Computing Sciences Research Center, Murray Hill, NJ, 1995–2001, Consultant. Worked on the Standard ML of New Jersey (SML/NJ) project.

Xerox Palo Alto Research Center, Summer 1993, Research Intern for Dr. Hans Boehm and Dr. John Ellis. Developed a set of runtime optimizations for Boehm's conservative garbage collector; built its interfaces in the GNU GCC and SRC Modula-3 compilers.

Bell Laboratories, Computing Sciences Research Center, Murray Hill, NJ, Summer 1991, Research Intern for Dr. David MacQueen. Designed a new separate compilation system for Standard ML; developed tools and optimizations for the SML/NJ project.

Princeton University, Department of Computer Science, 1989–1994, Research Assistant for Prof. Andrew W. Appel; Teaching Assistant for courses on systems programming and theory of algorithms.

Chinese Academy of Science, Institute of Software, Beijing, China, 1988–1989. Research Assistant. Worked for Prof. C.S. Tang and Prof. Huimin Lin on algebraic specifications of

abstract datatypes and semantics-based programming environments.

University of Science and Technology of China, Hefei, China, 1986–1988, Research Staff and Team Leader. Designed, developed, and commercialized a software system on educational management and timetable scheduling.

Awards

Communications of the ACM (CACM) Research Highlight, Building Certified Concurrent OS Kernels (with Ronghui Gu, Hao Chen, Jieung Kim, Jeremie Koenig, Newman Wu, Vilhelm Sjoberg, and David Costanzo), October 2019.

Member, Connecticut Academy of Science and Engineering, 2017-present.

National Science Foundation Expeditions in Computing Award, The Science of Deep Specification (with Andrew Appel, Adam Chlipala, Benjamin Pierce, Stephanie Weirich, Steven Zdancewic), 2015-2020.

USENIX 2nd Java Virtual Machine Research and Technology Symposium (JVM'02) Best Student Paper Award, Supporting Compatibility with Static Compilation (with Dachuan Yu and Valery Trifonov), August 2002.

National Science Foundation Faculty Early CAREER Development Award, 1995–1998.

Guo Mo-Ruo Award (for the best undergraduate student in computer science), University of Science and Technology of China, 1988.

Software

Key developer of the Standard ML of New Jersey (SML/NJ) compiler since 1990. Main architect and implementor of several latest releases (including version 110). SML/NJ is a production-quality compiler for Standard ML 1997 currently used by thousands of students, researchers, and developers worldwide. Worked on the compiler front-end (type-checker, module elaborator, abstract syntax, semantic analysis), the middle-end (FLINT-based intermediate languages, representation analysis, FLINT optimizations, CPS-based intermediate languages, CPS conversion, CPS optimizations, space-efficient closure conversion), and the backend and the runtime system (generation of abstract machine code, callee-save registers).

Leader of the Yale FLINT group which previously developed the systems software (i.e., compiler infrastructure, runtime systems) for advanced type-safe languages such as ML, Java, and safe dialects of C. FLINT was the first production-quality type-preserving compiler infrastructure. The FLINT system is currently used inside the SML/NJ compiler and by several research groups working on type-directed compilation and proof-carrying code. The FLINT group is currently in the middle of developing a practical infrastructure for building large-scale certified systems software, focusing on the development of new program verification techniques and integrated programming and proof tools.

The FLINT group has made multiple breakthroughs showing that building hacker-resistant concurrent OS kernels is not only feasible but also practical. The group developed a novel language-based account of certified concurrent abstraction layers, advocated abstraction over a particularly rich class of specification (called deep specification), and then constructed new methodologies and tools for formally specifying, programming, verifying, and composing abstraction layers. The group has successfully developed the CertiKOS certified OS kernel and verified its contextual functional correctness in Coq. CertiKOS is written

in 6500 lines of C and x86 assembly and runs on stock x86 multicore machines. This is the world's first proof of functional correctness of a complete, general-purpose concurrent OS kernel with fine-grained locking.

Publications ***Refereed journal and highly selective, refereed conference papers:***

[1] Y. Zhang, J. Koenig, Z. Shao, and Y. Wang. Fully Composable and Adequate Verified Compilation with Direct Refinements between Open Modules. *Proc. ACM Program. Lang. (PACMPL)* 9, POPL, Article 64, 31 pages, January 2025.

[2] A. Oliveira Vale, Z. Wang, Y. Chen, P. You, and Z. Shao. Compositionality and Observational Refinement for Linearizability with Crashes. *Proc. ACM Program. Lang. (PACMPL)*, 8, OOPSLA2, Article 352, 29 pages, October 2024.

[3] L. Qiu, Y. Kim, J. Shin, J. Kim, W. Honore, and Z. Shao. LiDO: Linearizable Byzantine Distributed Objects with Refinement-Based Liveness Proofs. *Proc. ACM Program. Lang. (PACMPL)*, 8, PLDI, Article 193, 25 pages, June 2024.

[4] A. Oliveira Vale, Z. Shao, and Y. Chen. A Compositional Theory of Linearizability (Extended Version). *Journal of ACM (JACM)*, Volumn 71, Issue 2, Article 14, 107 pages, April 2024.

[5] W. Honore, L. Qiu, Y. Kim, J. Shin, J. Kim, and Z. Shao. AdoB: Bridging Benign and Byzantine Consensus with Atomic Distributed Objects. *Proc. ACM Program. Lang. (PACMPL)*, 8, OOPSLA1, Article 109, 30 pages, April 2024.

[6] L. Zhang, Y. Wang, J. Wu, J. Koenig, and Z. Shao. Fully Composable and Adequate Verified Compilation with Direct Refinements between Open Modules. *Proc. ACM Program. Lang. (PACMPL)* 8, POPL, Article 72, 31 pages, January 2024.

[7] Y. Sang, N. Luo, S. Judson, B. Chaimberg, T. Antonopoulos, X. Wang, R. Piskac, and Z. Shao. Ou: Automating the Parallelization of Zero-Knowledge Protocols. *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communication Security (CCS'23)*, Copenhagen, Denmark, pages 534-548, November 2023. January 2023.

[8] A. Oliveira Vale, Z. Shao, and Y. Chen. A Compositional Theory of Linearizability. *Proc. ACM Program. Lang. (PACMPL)* 7, POPL, Article 38, 32 pages, January 2023.

[9] M. Liu, Z. Shao, H. Chen, M. Yoon, and J. Kim. Compositonal Virtual Timeline: Verifying Dynamic-Priority Partitions with Algorithmic Temporal Isolation . *Proc. ACM Program. Lang. (PACMPL)*, 6, OOPSLA2, Article 127, 29 pages, October 2022.

[10] W. Honore, J. Shin, J. Kim, and Z. Shao. Adore: Atomic Distributed Objects with Certified Reconfiguration. *Proceedings of the 2022 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'22)*, San Diego, California, 16 pages, June 2022.

[11] M. Yoon, J. Kim, R. Bradford, and Z. Shao. TimeDice: Schedulability-Preserving Priority Inversion for Mitigating Covert Timing Channels Between Real-time Partitions. *Proc. 52nd Annual IEEE/IFIP International Conference on Dependendable Systems and Networks (DSN'22)* Baltimore, Maryland, June 2022.

[12] A. Oliveira Vale, P.A. Mellies, Z. Shao, J. Koenig, and L. Stefanesco. Layered and Object-Based Game Semantics. *Proc. ACM Program. Lang. (PACMPL)* 6, POPL, Article 42, 32 pages, January 2022.

[13] Y. Wang, L. Zhang, Z. Shao, and J. Koenig. Verified Compilation of C Programs with a Nominal Memory Model. *Proc. ACM Program. Lang. (PACMPL)* 6, POPL, Article 25, 31 pages, January 2022.

[14] W. Honore, J. Kim, J. Shin, and Z. Shao. Much ADO about Failures: A Fault-Aware Model for Compositional Verification of Strongly Consistent Distributed Systems. *Proc. ACM Program. Lang. (PACMPL)*, 5, OOPSLA, Article 97, 31 pages, October 2021.

[15] M. Yoon, M. Liu, H. Chen, J. Kim, and Z. Shao. Blinder: Partition-Oblivious Hiearchical Scheduling. *Proceedings of the 30th USENIX Security Symposium (USENIX Security 2021)*, August 2021.

[16] J. Koenig and Z. Shao. CompCertO: Compiling Certified Open C Components. *Proceedings of the 2021 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'21)*, Virtual, Cananda, 15 pages, June 2021.

[17] J. Kim, R. Bradford, M. Del Giudice and Z. Shao. Adaptive Generative Modeling in Resource-Constrained Environments. *Proc. 24th ACM/IEEE Design, Automation, and Test in Europe (DATE'21)* Grenoble, France, February 2021.

[18] J. Kim, R. Bradford, M. Del Giudice and Z. Shao. Paired Training Framework for Time-Constrained Learning. *Proc. 24th ACM/IEEE Design, Automation, and Test in Europe (DATE'21)* Grenoble, France, February 2021.

[19] Y. Wang, X. Xu, P. Wilke, and Z. Shao. CompCertELF: Verified Separate Compilation of C Programs into ELF Object Files. *Proc. ACM Program. Lang. (PACMPL)*, 4, OOPSLA, Article 197, 28 pages, November 2020.

[20] J. Koenig and Z. Shao Refinement-Based Game Semantics for Certified Abstraction Layers. *Proceedings of the 35th ACM/IEEE Annual Symposium on Logic in Computer Science (LICS'20)*, pages 633-647, July 2020.

[21] V. Chen, M. Yoon, and Z. Shao. Task-Aware Novelty Detection for Visual-based Deep Learning in Autonomous Systems. *Proc. International Conference on Robotics and Automation (ICRA'20)*. Paris, France, pages 11060-11066, June 2020.

[22] J. Kim, R. Bradford, and Z. Shao. AnytimeNet: Controlling Time-Quality Tradeoffs in Deep Neural Network Architectures. *Proc. 23rd ACM/IEEE Design, Automation, and Test in Europe (DATE'20)* Grenoble, France, March 2020.

[23] J. Kim, R. Bradford, M. Yoon, and Z. Shao. ABC: Abstract Prediction Before Concreteness. *Proc. 23rd ACM/IEEE Design, Automation, and Test in Europe (DATE'20)* Grenoble, France, March 2020.

[24] M. Liu, L. Rieg, Z. Shao, R. Gu, D. Costanzo, J. Kim, and M. Yoon. Virtual Timeline: A Formal Abstraction for Verifying Preemptive Schedulers with Temporal Isolation. *Proc. ACM Program. Lang. (PACMPL)* 4, POPL, Article 20, 31 pages, January 2020.

[25] J. Shin, J. Kim, W. Honore, H. Vanzetto, S. Radhakrishnan, M. Balakrishnan, and Z. Shao. WormSpace: A Modular Foundation for Simple, Verifiable Distributed Systems. *Proc. 2019 ACM Symposium on Cloud Computing (SOCC'19)*, Santa Cruz, California, pages 299-311, November 2019.

[26] R. Gu, Z. Shao, H. Chen, J. Kim, J. Koenig, N. Wu, V. Sjoberg, and D. Costanzo. Building Certified Concurrent OS Kernels. *Communications of the ACM*, 62(10), pages 89-99, October 2019.

[27] V. Sjoberg, Y. Sang, S. Weng, and Z. Shao. DeepSEA: A Language for Certified System Software. *Proc. ACM Program. Lang. (PACMPL)*, 3, OOPSLA, Article 136, 27 pages, October 2019.

[28] X. Guo, M. Lesourd, M. Liu, L. Rieg, and Z. Shao. Integrating Formal Schedulability Analysis into a Verified OS Kernel. *Proc. 31st International Conference on Computer Aided Verification (CAV 2019)*, Part II, New York, USA, July 2019. Published as *Lecture Notes in Computer Science*, volume 11562, pages 496-514, Springer, 2019.

[29] M. Yoon and Z. Shao, ADLP: Accountable Data Logging Protocol for Publish-Subscribe Communication Systems, *Proceedings of the 39th IEEE International Conference on Distributed Computing Systems (ICDCS'19)*, July 2019.

[30] Y. Wang, P. Wilke, and Z. Shao. An Abstract Stack Based Approach to Verified Compositional Compilation to Machine Code. *Proc. ACM Program. Lang. (PACMPL)* 3, POPL, Article 62, 30 pages, January 2019.

[31] R. Gu, Z. Shao, J. Kim, X. Wu, J. Koenig, V. Sjoberg, H. Chao, D. Costanzo, and T. Ramananandro. Certified Concurrent Abstraction Layers. *Proceedings of the 2018 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'18)*, Philadelphia, Pennsylvania, pages 646-661, June 2018.

[32] Q. Carbonneaux, J. Hoffmann, T. Reps, and Z. Shao. Automated Resource Analysis with Coq Proof Objects. *Proc. 29th International Conference on Computer Aided Verification (CAV 2017)*, Part II, Heidelberg, Germany, July 2017. Published in *Lecture Notes in Computer Science*, volume 10427, pages 64-85, Springer, 2017.

[33] R. Gu, Z. Shao, H. Chen, X. Wu, J. Kim, V. Sjoberg, and D. Costanzo. CertiKOS: An Extensible Architecture for Building Certified Concurrent OS Kernels. *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI'16)*, Savannah, Georgia, pages 653-669, November 2016.

[34] H. Chen, X. Wu, Z. Shao, J. Lockerman, and R. Gu. Toward Compositional Verification of Interruptible OS Kernels and Device Drivers. *Proceedings of the 2016 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'16)*, Santa Barbara, California, pages 431-447, June 2016. An extended version of this paper appeared in *Journal of Automated Reasoning (JAR)*, volume 61, issue 1-4, pages 141-189, June 2018.

[35] D. Costanzo, Z. Shao, and R. Gu. End-to-End Verification of Information-Flow Security for C and Assembly Programs. *Proceedings of the 2016 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'16)*, Santa Barbara, California, pages 648-664, June 2016.

[36] J. Hoffmann and Z. Shao. Type-Based Amortized Resource Analysis with Integers and Arrays. *Journal of Functional Programming (JFP)*, volume 25, e17, 35 pages, October 2015.

[37] Q. Carbonneaux, J. Hoffmann, and Z. Shao. Compositional Certified Resource Bounds. *Proceedings of the 2015 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'15)*, Portland, Oregon, pages 467-478, June 2015.

[38] J. Hoffmann and Z. Shao. Automatic Static Cost Analysis for Parallel Programs. *Proceedings of the 24th European Symposium on Programming (ESOP'15)*, London, UK, April 2015. Published in Jan Vitek, editor, *Lecture Notes in Computer Science*, volume 9032, pages 132–157, Springer-Verlag, 2015.

[39] R. Gu, J. Koenig, T. Ramananandro, Z. Shao, X. Wu, S. Weng, H. Zhang, and Y. Guo. Deep Specifications and Certified Abstraction Layers. *Proceedings of the 42nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'15)*, Mumbai, India, pages 595–608, January 2015.

[40] H. Liang, X. Feng, and Z. Shao. Compositional Verification of Termination-Preserving Refinement of Concurrent Programs. *Proceedings of the 23rd EACSL Annual Conference on Computer Science Logic and 29th Annual IEEE Symposium on Logic in Computer Science (CSL-LICS'14)*, Article No. 65, Vienna, Austria, July 2014.

[41] Q. Carbonneaux, J. Hoffmann, T. Ramananandro, and Z. Shao. End-to-End Verification of Stack-Space Bounds for C Programs. *Proceedings of the 2014 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'14)*, Article No. 30, Edinburgh, UK, June 2014.

[42] D. Costanzo and Z. Shao A Separation Logic for Enforcing Declarative Information Flow Control Policies. *Proceedings of the 3rd Conference on Principles of Security and Trust (POST'14)*, Grenoble, France, April 2014. Published in *Lecture Notes in Computer Science*, volume 8414, pages 179–198, Springer-Verlag, 2014.

[43] H. Liang, J Hoffmann, X. Feng, and Z. Shao. Characterizing Progress Properties of Concurrent Objects via Contextual Refinements. *Proceedings of the 24th International Conference on Concurrency Theory (CONCUR'13)*, Buenos Aires, Argentina, pages 227–241, August 2013. Published in *Lecture Notes in Computer Science*, volume 8052, pages 227–241, Springer-Verlag, 2013.

[44] J. Hoffmann, M. Marmar, and Z. Shao. Quantitative Reasoning for Proving Lock-Freedom. *Proceedings of the 28th IEEE Annual Symposium on Logic in Computer Science (LICS'13)*, New Orleans, USA, pages 124–133, June 2013.

[45] P. Kazanzides, Y. Kouskoulas, A. Deguet, and Z. Shao. Proving the Correctness of Concurrent Robot Software. *Proc. IEEE International Symposium on Robotics and Automation (ICRA'12)*, St. Paul, Minnesota, USA, May 2012.

[46] A. Stampoulis and Z. Shao. Static and User-Extensible Proof Checking. *Proceedings of the 39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'12)*, Philadelphia, PA, pages 273-284., January 2012.

[47] G. Tan, Z. Shao, X. Feng, and H. Cai. Weak Updates and Separation Logic. *New Generation Computing*, volume 29, No.1, 2011, pages 1-29. Ohmsha, Ltd. and Springer.

[48] Z. Shao. Certified Software. *Communications of ACM*, 53(12), pages 56–66, December 2010.

[49] A. Stampoulis and Z. Shao. VeriML: Typed Computation of Logical Terms inside a Language with Effects. *Proceedings 2010 ACM SIGPLAN International Conference on Functional Programming (ICFP'10)*, Baltimore, Maryland, pages 333–344, September 2010.

[50] M. Fu, Y Li, X. Feng, Z. Shao, and Y. Zhang. Reasoning about Optimistic Concurrency using a Program Logic for History. *Proceedings of the 21st International Conference on Concurrency Theory (CONCUR'10)*, Paris, France, pages 388–402, August 2010. Published in *Lecture Notes in Computer Science*, volume 6269, pages 388–402, Springer-Verlag, 2010.

[51] R. Ferreira, X. Feng, and Z. Shao. Parameterized Memory Models and Concurrent Separation Logic. *Proceedings of the 19th European Symposium on Programming (ESOP'10)*, Paphos, Cyprus, March 2010. Published in Andrew Gordon, editor, *Lecture Notes in Computer Science*, volume 6012, pages 267–286, Springer-Verlag, 2010.

[52] X. Feng, Z. Shao, Y. Dong, and Y. Guo. Certifying Low-Level Programs with Hardware Interrupts and Preemptive Threads. *Proceedings of the 2008 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'08)*, Tuscon, AZ, pages 170–182, June 2008. An extended version of this paper appeared in *Journal of Automated Reasoning (JAR)*, special issue on Operating System Verification, 42(2-4):301-347, April 2009. Springer Science and Business Media B.V.2009.

[53] A. McCreight, Z. Shao, C. Lin, and L. Li. A General Framework for Certifying Garbage Collectors and Their Mutators. *Proceedings of the 2007 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'07)*, San Diego, CA, pages 468–479, June 2007.

[54] H. Cai, Z. Shao, and A. Vaynberg. Certified Self-Modifying Code. *Proceedings of the 2007 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'07)*, San Diego, CA, pages 66–77, June 2007.

[55] X. Feng, R. Ferreira, and Z. Shao. On the Relationship Between Concurrent Separation Logic and Assume-Guarantee Reasoning. *Proceedings of the 16th European Symposium on Programming (ESOP'07)*, Braga, Portugal, March 2007. Published in Rocco De Nicola, editor, *Lecture Notes in Computer Science*, volume 4421, pages 173–188, Springer-Verlag, 2007.

[56] X. Feng, Z. Shao, A. Vaynberg, S. Xiang, and Z. Ni. Modular Verification of Assembly Code with Stack-Based Control Abstractions, *Proceedings of the 2006 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'06)*, Ottawa, Canada, pages 401–414, June 2006.

[57] Z. Ni and Z. Shao. Certified Assembly Programming with Embedded Code Pointers, *Proceedings of the 33rd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'06)*, Charleston, SC, pages 320–333, January 2006.

[58] X. Feng and Z. Shao. Modular Verification of Concurrent Assembly Code with Dynamic Thread Creation and Termination, *Proceedings of the Tenth ACM SIGPLAN International Conference on Functional Programming (ICFP'05)*, Tallinn, Estonia, pages 254–267, September 2005.

[59] Z. Shao, V. Trifonov, B. Saha, and N. Papaspyrou. A Type System for Certified Binaries. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 27(1), pages 1–45, January 2005.

[60] D. Yu and Z. Shao. Verification of Safety Properties for Concurrent Assembly Code, *Proceedings of the Ninth ACM SIGPLAN International Conference on Functional Programming (ICFP'04)*, Snowbird, Utah, pages 175–188, September 2004.

[61] D. Yu, N.A. Hamid, and Z. Shao. Building Certified Libaries for PCC: Dynamic Storage Allocation. In *Science of Computer Programming*, 50(1-3), pages 101-127, 2004. An early version of this paper appeared in *Proceedings of the 2003 European Symposium on Programming (ESOP'03)*, Warsaw, Poland, April 2003. Published in Pierpaolo Degano, editor, *Lecture Notes in Computer Science*, volume 2618, pages 363–379, Springer-Verlag, 2003.

[62] B. Saha, V. Trifonov, and Z. Shao. Intensional Analysis of Quantified Types. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 25(2), pages 159–209, March 2003.

[63] S. Monnier and Z. Shao. Inlining as Staged Computation. *Journal of Functional Programming (JFP)*, 13(3), pages 647–676, May 2003.

[64] N.A. Hamid, Z. Shao, V. Trifonov, S. Monnier, and Z. Ni. A Syntactic Approach to Foundational Proof-Carrying Code. *Proceedings of the 17th IEEE Annual Symposium on Logic in Computer Science (LICS'02)*, Copenhagen, Denmark, pages 89–100, July 2002. An extended version of this paper appeared in *Journal of Automated Reasoning (JAR)*, 31(3-4), pages 191-229, October 2003.

[65] C. League, Z. Shao, and V. Trifonov. Type-Preserving Compilation of Featherweight Java. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 24(2), pages 112–152, March 2002.

[66] Z. Shao, B. Saha, V. Trifonov, and N. Papaspyrou. A Type System for Certified Binaries. *Proceedings of the 29th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'02)*, Portland, OR, pages 217–232, January 2002.

[67] S. Monnier, B. Saha, and Z. Shao. Principled Scavenging. *Proceedings of the 2001 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'01)*, Snowbird, UT, pages 81–91, June 2001.

[68] V. Trifonov, B. Saha, and Z. Shao. Fully Reflexive Intensional Type Analysis. *Proceedings of the Fifth ACM SIGPLAN International Conference on Functional Programming (ICFP'00)*, Montreal, Canada, pages 82–93, September 2000.

[69] Z. Shao and A.W. Appel. Efficient and Safe-for-Space Closure Conversion. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 22(1), pages 129–161, January 2000.

[70] C. League, Z. Shao, and V. Trifonov. Representing Java Classes in a Typed Intermediate Language. *Proceedings of the Fourth ACM SIGPLAN International Conference on Functional Programming (ICFP'99)*, Paris, France, pages 183–196, September 1999.

[71] Z. Shao. Transparent Modules with Fully Syntactic Signatures. *Proceedings of the Fourth ACM SIGPLAN International Conference on Functional Programming (ICFP'99)*, Paris, France, pages 220–232, September 1999.

[72] V. Trifonov and Z. Shao. Safe and Principled Language Interoperation. *Proceedings of the 1999 European Symposium on Programming (ESOP'99)*, Amsterdam, The Netherlands, March 1999. Published in S. Doaitse Swierstra, editor, *Lecture Notes in Computer Science*, volume 1576, pages 128–146, Springer-Verlag, 1999.

[73] Z. Shao, C. League, and S. Monnier. Implementing Typed Intermediate Languages. *Proceedings of the Third ACM SIGPLAN International Conference on Functional Programming (ICFP'98)*, Baltimore, MD, pages 313–323, September 1998.

[74] Z. Shao. Typed Cross-Module Compilation. *Proceedings of the Third ACM SIGPLAN International Conference on Functional Programming (ICFP'98)*, Baltimore, MD, pages 141–152, September 1998.

[75] Z. Shao. Flexible Representation Analysis. *Proceedings of the Second ACM SIGPLAN International Conference on Functional Programming (ICFP'97)*, Amsterdam, The Netherlands, pages 85–98, June 1997.

[76] A.W. Appel and Z. Shao. Empirical and Analytic Study of Stack vs. Heap Cost for Languages with Closures. *Journal of Functional Programming (JFP)*, 6(1), pages 47–74, January 1996.

[77] Z. Shao and A.W. Appel. A Type-Based Compiler for Standard ML. *Proceedings of the 1995 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'95)*, La Jolla, California, pages 116–129, June 1995.

[78] Z. Shao and A.W. Appel. Space Efficient Closure Representations. *Proceedings of the ACM SIGPLAN Conference on Lisp and Functional Programming (LFP'94)*, Orlando, FL, pages 150–161, June 1994.

[79] Z. Shao, J.H. Reppy, and A.W. Appel. Unrolling Lists. *Proceedings of the ACM SIGPLAN Conference on Lisp and Functional Programming (LFP'94)*, Orlando, FL, pages 185–195, June 1994.

[80] Z. Shao and A.W. Appel. Smartest Recompilation. *Proceedings of the 20th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'93)*, Charleston, SC, pages 439–450, January 1993.

[81] A.W. Appel and Z. Shao. Callee-save Registers in Continuation-Passing Style. *Lisp and Symbolic Computation*, 5(3), pages 189–219, 1992.

Other refereed journal, conference, and workshop papers:

[82] V. Chen, M. Yoon, and Z. Shao. Novelty Detection via Network Saliency in Visual-Based Deep Learning. *Proceedings of 2nd Workshop on Dependable and Secure Machine Learning*, pages 52-57, Portland, OR, June 2019.

[83] A. W. Appel, L. Beringer, A. Chlipala, B. C. Pierce, Z. Shao, S. Weirich, and S. Zdancewic. Position Paper: The Science of Deep Specifications, *Philosophic Transactions of the Royal Society A*, volume 375, issue 2104, 24 pages, October 2017.

[84] J. Kim, V. Sjoberg, R. Gu, and Z. Shao, Safety and Liveness of MCS Lock – Layer by Layer *Proceedings of the 15th Asian Symposium on Programming Languages and Systems (APLAS'17)*, Suzhou, China, November 2017. Published in *Lecture Notes in Computer Science*, volume 10695, pages 273–297, Springer-Verlag, 2017.

[85] J. Hoffmann and Z. Shao. Type-Based Amortized Resource Analysis with Integers and Arrays, *Proceedings of the 12th International Symposium on Functional and Logic Programming (FLOPS'14)*, Kanazawa, Japan, June 2014. Published in *Lecture Notes in Computer Science*, volume 8475, pages 152–168, Springer-Verlag, 2014.

[86] T. Ramananandro, Z. Shao, S.C. Weng, J. Keonig, and Y. Fu. A Compositional Semantics for Verified Separate Compilation and Linking. *Proceedings of the 4th ACM International Conference on Certified Programs and Proofs (CPP'14)*, Mumbia, India, pages 3–14. January 2015.

[87] A. Vaynberg and Z. Shao, Compositional Verification of a Baby Virtual Memory Manager, *Proceedings of the 2nd International Conference on Certified Programs and Proofs (CPP'12)*, Kyoto, Japan, December 2012. Published in *Lecture Notes in Computer Science*, volume 7679, pages 143–159, Springer-Verlag, 2012.

[88] D. Costanzo and Z. Shao, A Case for Behavior-Preserving Actions in Separation Logic, *Proceedings of the 10th Asian Symposium on Programming Languages and Systems (APLAS'12)*, Kyoto, Japan, December 2012. Published in *Lecture Notes in Computer Science*, volume 7705, pages 332–349, Springer-Verlag, 2012.

[89] Y. Guo, X. Feng, Z. Shao, and P. Shi, Modular Verification of Concurrent Thread Management, *Proceedings of the 10th Asian Symposium on Programming Languages and Systems (APLAS'12)*, Kyoto, Japan, December 2012. Published in *Lecture Notes in Computer Science*, volume 7705, pages 315–331, Springer-Verlag, 2012.

[90] Z. Zhang, X. Feng, M. Fu, Z. Shao, and Y. Li, A Structural Approach to Prophecy Variables, *Proceedings of the 9th Annual Conference on Theory and Applications of Models of Computation (TAMC'12)*, Beijing, China, May 2012. Published in *Lecture Notes in Computer Science*, volume 7287, pages 61–71, Springer-Verlag, 2012.

[91] W. Wang, Z. Shao, X. Jiang, and Y. Guo. A Simple Model for Certifying Assembly Programs with First-Class Function Pointers. *Proc. 4th IEEE & IFIP International Symposium on Theoretical Aspects of Software Engineering (TASE'11)*, Xian, China, August 2011.

[92] L. Gu, A. Vaynberg, B. Ford, Z. Shao, and D. Constanzo. CertiKOS: A Certified Kernel for Secure Cloud Computing. *Proc. 2nd ACM SIGOPS Asia-Pacific Workshop on Systems (APSys'11)*, Shanghai, China, July 2011.

[93] G. Tan, Z. Shao, X. Feng, and H. Cai. Weak Updates and Separation Logic. *Proceedings of the 7th Asian Symposium on Programming Languages and Systems (APLAS'09)*, Seoul, Korea, December 2009. Published in *Lecture Notes in Computer Science*, volume 5904, pages 178–193, Springer-Verlag, 2009.

[94] X. Feng, Z. Shao, Y. Guo, and Y. Dong. Combining Domain-Specific and Foundational Logics to Verify Complete Software Systems. *Proceedings of the 2nd IFIP Working Conference on Verified Software: Theories, Tools, and Experiments (VSTTE'08)*, Toronto, Canada, October 2008. Published in *Lecture Notes in Computer Science*, Springer-Verlag, 2008.

[95] Z. Ni, D. Yu,and Z. Shao. Using XCAP to Certify Realistic System Code: Machine Context Management. *Proceedings of the 20th International Conference on the Applications of Higher Order Logic Theorem Proving (TPHOLs'07)*, Kaiserslautern, Germany, September 2007. Published in *Lecture Notes in Computer Science*, volume 4732, pages 189–206, Springer-Verlag, 2007.

[96] C. Lin, A. McCreight, Z. Shao, Y. Chen, and Y. Guo. Foundational Typed Assembly Language with Certified Garbage Collection. *Proc. 1st IEEE & IFIP International Symposium on Theoretical Aspects of Software Engineering (TASE'07)*, Shanghai, China, pages 326–335, June 2007.

[97] X. Feng, Z. Ni, Z. Shao, and Y. Guo. An Open Framework for Foundational Proof-Carrying Code. *Proceedings of the 2007 ACM SIGPLAN International Workshop on Types in Language Design and Implementation (TLDI'07)*, Nice, France, pages 67–78, January 2007.

[98] N. Hamid and Z. Shao. Interfacing Hoare Logic and Type Systems for Foundational Proof-Carrying Code. *Proceedings of the 17th International Conference on the Applications of Higher Order Logic Theorem Proving (TPHOLs'04)*, Park City, Utah, September 2004. Published in Konrad Slind, editor, *Lecture Notes in Computer Science*, volume 3223, pages 118–135, Springer-Verlag, 2004.

[99] C. League, Z. Shao, and V. Trifonov. Precision in Practice: A Type-Preserving Java Compiler. *Proceedings of the 12th International Conference on Compiler Construction (CC'03)*, Warsaw, Poland, April 2003. Published in Gorel Hedin, editor, *Lecture Notes in Computer Science*, volume 2622, pages 106–120, Springer-Verlag, 2003.

[100] D. Yu, Z. Shao, and V. Trifonov. Supporting Binary Compatibility with Static Compilation. *Proceedings of the Second USENIX Java Virtual Machine Research and Technology Symposium (JVM'02)*, San Francisco, CA, pages 165–180, August 2002. *Winner of the Best Student Paper Award.*

[101] D. Yu, V. Trifonov, and Z. Shao. Type-Preserving Compilation of Featherweight IL (Extended Abstract). *Proceedings of the 2002 International Workshop on Formal Techniques for Java-like Programs (FTfJP'02)*, June 2002.

[102] C. League, V. Trifonov, and Z. Shao. Functional Java Bytecode. *Proceedings of the 2001 Workshop on Intermediate Representation Engineering for the Java Virtual Machine (IRE'01) at the 5th World Multi-conference on Systemics, Cybernetics, and Informatics*, Orlando, Florida, July 2001.

[103] C. League, V. Trifonov, and Z. Shao. Type-Preserving Compilation of Featherweight Java. *Proceedings of the Eighth ACM SIGPLAN International Workshop on Foundations of Object-Oriented Languages (FOOL'01)*, London, UK, January 2001.

[104] B. Saha, V. Trifonov, and Z. Shao. Fully Reflexive Intensional Type Analysis with Type Erasure Semantics. *Proceedings of the Third International Workshop on Types in Compilation (TIC'00)*, Montreal, Canada, September 2000.

[105] B. Saha and Z. Shao. Optimal Type Lifting. *Proceedings of the Second International Workshop on Types in Compilation (TIC'98)*, Kyoto, Japan, March 1998. Published in Xavier Leroy and Astushi Ohori, editors, *Lecture Notes in Computer Science*, volume 1473, pages 156–177, Springer-Verlag, 1998.

[106] Z. Shao and V. Trifonov. Type-Directed Continuation Allocation. *Proceedings of the Second International Workshop on Types in Compilation (TIC'98)*, Kyoto, Japan, March 1998. Published in Xavier Leroy and Astushi Ohori, editors, *Lecture Notes in Computer Science*, volume 1473, pages 116–135, Springer-Verlag, 1998.

[107] Z. Shao. Typed Common Intermediate Format. *Proceedings of the 1997 USENIX Conference on Domain-Specific Languages (DSL'97)*, Santa Barbara, CA, pages 89–102, October 1997.

[108] Z. Shao. An Overview of the FLINT/ML Compiler. *Proceedings of the First International Workshop on Types in Compilation (TIC'97)*, Amsterdam, The Netherlands, June 1997.

[109] H. Boehm and Z. Shao. Inferring Type Maps during Garbage Collection. *Proceedings of the OOPSLA'93 Workshop on Memory Management and Garbage Collection*, Washington, DC, September 1993.

[110] Z. Shao. The Practical University Timetable Problem and its Timetabling Algorithm. *Proceedings of the First National Conference for Young Computer Scientists*, Harbin, China, August 1987.

Unrefereed papers and technical reports not published elsewhere:

[111] S. Monnier and Z. Shao. Typed Regions. Technical Report YALEU DCS TR–1242, Dept. of Computer Science, Yale University, October 2002.

[112] G. Collins and Z. Shao. Intensional Analysis of Higher-Kinded Recursive Types. Technical Report YALEU DCS TR–1240, Dept. of Computer Science, Yale University, October 2002.

[113] D. Yu, V. Trifonov, and Z. Shao. Type-Preserving Compilation of Featherweight IL. Technical Report YALEU DCS TR–1228, Dept. of Computer Science, Yale University, April 2002.

[114] A.W. Appel, Z. Shao, V. Trifonov, and D. Walker. High-Assurance Common Language Runtime. Technical Report YALEU DCS TR–1225, Dept. of Computer Science, Yale University, December 2001.

[115] D. Teller and Z. Shao. Algorithm-Independent Framework for Verifying Integer Constraints. Technical Report YALEU DCS TR–1195, Dept. of Computer Science, Yale University, June 2000.

[116] A.W. Appel, E. Felten, and Z. Shao. Scaling Proof-Carrying Code to Production Compilers and Security Policies. Techical Report YALEU DCS TR–1182, Dept. of Computer Science, Yale University, January 1999.

[117] The ML2000 Working Group. Principles and a Preliminary Design for ML2000. March 1999.

[118] S. Monnier, M. Blume, and Z. Shao. Cross-Functor Inlining in FLINT. Technical Report YALEU DCS TR–1189, Dept. of Computer Science, Yale University, March 1999.

[119] C. League, Z. Shao, and V. Trifonov. Encoding Java Classes in a Typed Intermediate Language. Technical Report YALEU DCS TR–1173, Dept. of Computer Science, Yale University, December 1998.

[120] S. Monnier and Z. Shao. The FLINT Optimizer. Technical Report YALEU DCS TR–1172, Dept. of Computer Science, Yale University, December 1998.

[121] C. League and Z. Shao. Formal Semantics of the FLINT Intermediate Language. Technical Report YALEU DCS TR–1171, Dept. of Computer Science, Yale University, May 1998.

[122] Z. Shao. Parameterized Signatures and Higher-Order Modules. Technical Report YALEU DCS TR–1161, Dept. of Computer Science, Yale University, August 1998.

[123] Z. Shao. Compiling Standard ML for Efficient Execution on Modern Machines. Ph.D. Thesis. Technical Report CS–TR–475–94, Dept. of Computer Science, Princeton University, September 1994.

[124] Z. Shao. A Practical University Timetabling System. Zhong Shao. Bachelor's Thesis (in Chinese), University of Science and Technology of China, June 1988.

Grants

Compositional Certified Concurrent Abstraction Layers, National Science Foundation Grant CCF-2313433, $540,000. October 2023–September 2026.

ACE-PAVE: Privacy, Accountability, Verification, and Economics of Blockchain Systems (with Charalampos Papamanthou, Ben Fisch, Bryan Ford, Eran Tromer, Gur Huberman, Joan Feigenbaum, Rosario Gennaro, and Tal Malkin). Algorand Foundation, $1,150,000, August 2022 - July 2023.

PPoSS: Planning: High-Performance Certified Trust for Global-Scale Applications (with Abhishek Bhattacharjee, Anurag Khandelwal, and Lin Zhong), National Science Foundation Grant CCF 2118851, $250,000. October 2021–September 2022.

REFUEL: Verified Composition and Flattening of Unified Enclave Layers (with Jason Nieh, Ronghui Gu, Abhishek Bhattacharjee, and Gail Kaiser), Defense Advanced Research Projects Agency (DARPA), Award N66001-21-C-4018, $4,563,980, April 2021–March 2025.

ADVERT: Compositional Atomic Specifications for Distributed System Verification (with Ji Yong Shin and Robert Soule), National Science Foundation Grant CCF-2019285, $749,943. October 2020–September 2023.

Biking for Science and Health: Integration of Smart Sensor Technology with Public Bicycles for Urban Environmental Monitoring (with Xuhui Lee, Rob Dubrow, and Roman Kuc), Robert Wood Johnson Foundation, $366,358. August 2020-July 2023.

Partition-Oblivious Real-Time Hierarchical Scheduling (with Man-Ki Yoon and Jung-Eun Kim), National Science Foundation Grant CCF-1945541, $499,905. April 2020–March 2023.

DeepSEA: A Language for Programming and Synthesizing Certified Software (with Vilhelm Sjoberg and Ruzica Piskac), National Science Foundation Grant CCF-1763399, $800,000. June 2018–May 2022.

Measuring Heat Stress of Urban Residents with Smart Thermometers on Bicycles (with Xuhui Lee, Justin Farrell, and Roman Kuc), Leitner Award for Uncommon Environmental Collaborations, $75,000. 2018-2020.

Formal End-to-End Verification of Information-Flow Security for Complex Systems, National Science Foundation Grant CNS-1715154, $500,000. August 2017–July 2020.

Compositional Resource-Adaptive Certified System Software. Defense Advanced Research Projects Agency (DARPA), Award FA8750-16-2-0274, $396,717, August 2016–August 2018.

Expeditions in Computing: The Science of Deep Specification (with Andrew Appel, Adam Chlipala, Benjamin Pierce, Stefanie Weirich, and Steven Zdancewic). National Science

Foundation Grant CCF-1521523 (Yale FLINT component), $2,046,445. December 2015–November 2021.

CURB: Calculating and Understanding Resource Bounds to Detect Space/Time Vulnerabilities (with Alexey Loginov, Thomas Reps, and Jan Hoffmann). Defense Advanced Research Projects Agency (DARPA), Award FA8750-15-C-0082, $6,230,090 (Yale FLINT component: $563,547), April 2015–April 2019.

VeriQ: Formal Quantitative Software Verification in Realistic Application Scenarios (with Jan Hoffmann), National Science Foundation Grant CCF-1319671, $449,721, July 2013–June 2018.

Compositionality and Automation for Robotics Security (with Andrew Appel and Adam Chlipala), Defense Advanced Research Projects Agency (DARPA), Award FA8750-12-2-0293, $6,108,346 (Yale FLINT component: $2,799,966), August 2012–December 2016.

Reasoning Infrastructure for Security-Aware Software Development (with Bryan Ford and Joan Feigenbaum), Office of Naval Research Grant, Award N000141210478, $750,000, April 2012 – September 2015.

Making OS Kernels Crash-Proof by Design and Certification (with Bryan Ford), National Science Foundation Grant CNS-1065451, $1,116,262. August 2011–July 2016.

Advanced Development of Certified OS Kernels (with Bryan Ford), Defense Advanced Research Projects Agency (DARPA), Award FA8750-10-2-0254, $2,657,704, September 2010–December 2014.

Formal Reasoning about Concurrent Programs for Multicore and Multiprocessor Machines, National Science Foundation Grant CNS-0915888, $500,000, September 2009–August 2014.

Combining Foundational and Lightweight Formal Methods to Build Certifiably Dependable Software, National Science Foundation Grant CNS-0910670, $580,000, July 2009–June 2013.

Domain Specific Languages, Logics, and Proofs for Certified Software Design (with Paul Hudak), National Science Foundation Grant CCF-0811665, $850,000 (REU supplement: $19,238), July 2008–June 2013.

Microsoft Corporation Research Grant on Language and Compiler Support for Constructing Certified Systems Software, $100,000, April 2008–June 2009.

Certified Runtime Code Manipulation, National Science Foundation Program on Cyber Trust, CCF-0716540, $100,000, August 2007–July 2008.

Modular Development of Certified Concurrent Code, National Science Foundation Program on Cyber Trust, CCF-0524545, $400,000, August 2005–July 2008.

Intel Corporation Research Grant, $120,000, July 2004-June 2007.

Microsoft Corporation Research Grant, $72,000, April 2004–June 2006.

High-Assurance Common Language Runtime (with Valery Trifonov), National Science Foundation Program on Trusted Computing (TC), CNS-0208618, $400,000, August 2002–July 2005.

Microsoft Corporation Research Grant on Content and Curriculum, $35,000. January 2003–December 2004.

Scaling Proof-Carrying Code to Production Compilers and Security Policies—Technology Transfer Extension (with Andrew Appel, Valery Trifonov, and David Walker), Defense Advanced Research Projects Agency (DARPA), $1,346,386 (Yale FLINT component: $636,154), June 2002–June 2004.

FLINT—A Mobile-Code Infrastructure for Advanced Languages, National Science Foundation Information Technology Research (ITR) Award, CCF-0081590, $300,000, September 2000–August 2003.

Scaling Proof-Carrying Code to Production Compilers and Security Policies (with Andrew Appel and Edward Felten), Defense Advanced Research Projects Agency (DARPA), $2,224,772 (Yale FLINT component: $1,058,951), June 1999–June 2002.

Typed Common Intermediate Format, National Science Foundation Program on Software Engineering and Languages, CCR-9901011, $320,000, August 1999–July 2002.

Software Evolution using HOT Language Technology (with Paul Hudak and John Peterson), Defense Advanced Research Projects Agency (DARPA), $698,837, August 1996–July 1999.

Foundations of HOT Languages and Software Evolution (with Paul Hudak), National Science Foundation Grant CCR-9633390, $450,000, August 1996–July 1999.

Type-Directed Compilation, National Science Foundation Faculty Early CAREER Development Award CCR-9501624, $105,000, June 1995–May 1998.

Professional Activities

Significant leadership roles and service activities:

Chair of Steering Committee, *ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL)*, 2009-2010.

General Chair, *Thirty-sixth ACM Symposium on Principles of Programming Languages (POPL'09)*, January 2009.

Member at Large, *ACM SIGPLAN Executive Committee*, 2001–2005.

Chair of the ACM SIGPLAN Doctoral Dissertation Award Committee, 2003-2005.

Program Chair, *23rd European Symposium on Programming (ESOP)*, Grenoble, France, April 2014.

Program Co-Chair, *International Workshop on Certification of High-Level and Low-Level Programs*, IHP Trimester on Semantics of Proofs and Certified Mathematics. Paris, France, July 2014.

Steering Committee Chair and Co-Founder, *ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP)*, 2011–present.

Program Co-Chair, *First International Conference on Certified Programs and Proofs (CPP)*, 2011.

General Chair, *ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'03)*, New Orleans, LA, January 2003.

Program Chair, *Fifth Asian Symposium on Programming Languages and Systems (APLAS'07)*, Singapore, November 2007.

Member of Editorial Board, *Journal of Functional Programming*, 2001–2010.

Member of Advisory Board, Asian Association for Foundation of Software, 2003–present.

Other service activities:

Member of Program Committee, *52nd ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'25)*, Denver, CO. January 2025

Member of Program Committee, *2022 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA'22)*, Auckland, New Zealand, November 2022

Member of Program Committee, *6th Workshop on Principles of Secure Compilation (PriSC 2022)*, January 2022.

Member of Program Committee, *30th European Symposium on Programming*, Luxembourg, April 2021.

Member of Program Committee, *48th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'21)*, Copenhagen, Denmark. January 2021

30th European Symposium on Programming, Luxembourg, April 2021.

Member of Program Committee, *28th European Symposium on Programming*, Prague, Czech Republic, April 2019.

Member of Program Committee, *33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'18)*, Oxford, UK, July 2018.

Member of Program Committee, *45th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'18)*, San Francisco, CA, January 2018

Member of Program Committee, *30th IEEE Computer Security Foundation Symposium (CSF'17)*, Santa Barbara, CA, August 2017.

Member of External Program Committee, *ACM SIGPLAN International Conference on Programming Language Design and Implementation (PLDI'17)*, June 2017.

Invited Talk on CertiKOS: A Layered Architecture for Building Certified System Software. *10th Layered Assurance Workshop (LAW'16)*, Los Angeles, California, December 2016.

Invited Talk on Advanced Development of Certified OS Kernels. *8th Working Conference on Verified Software: Theories, Tools, and Experiments (VSTTE'16)*, Toronto, Canada, July 2016.

Member of External Review Committee, *Forty-third ACM Symposium on Principles of Programming Languages (POPL'16)*, St. Petersburg, FL, January 2016.

Co-Organizer, *Dagstuhl Seminar No. 15191 on Compositional Verification Methods for Next-Generation Concurrency*, Dagstuhl, Germany, May 2015.

Invited Talk on Clean-Slate Development of Certified OS Kernels. *Proceedings of the 4th ACM International Conference on Certified Programs and Proofs (CPP'15)*, Mumbia, India, pages 95–96. January 2015.

Member of External Review Committee, *ACM SIGPLAN International Conference on Programming Language Design and Implementation (PLDI'13)*, Seattle, WA, June 2013.

Member of Program Committee, *Fifth Working Conference on Verified Software: Theories, Tools, and Experiments (VSTTE'13)*, Atherton, California, May 2013.

Member of Program Committee, *22nd European Symposium on Programming*, Rome, Italy, April 2013.

Member of Program Committee, *International Workshop on Systems Software Verification (SSV'12)*, Sydney, November 2012.

Member of Program Committee, *ACM SIGPLAN Seventh Workshop on Programming Languages and Analysis for Security (PLAS'12)*, Beijing, China, June 2012.

Member of External Review Committee, *Thirty-ninth ACM Symposium on Principles of Programming Languages (POPL'12)*, Philadelphia, PA, January 2012.

Program Co-Chair, *2011 International Workshop on Syntax and Semantics of Low Level Languages (LOLA)*, 2011.

Member of Program Committee, *Third International Conference on Verified Software: Theories, Tools, and Experiments (VSTTE'10)*, Edinburgh, Scotland, August 2010.

Member of Program Committee, *Fifth International Workshop on Systems Software Verification (SSV'10)*, Vancouver, Canada, October 2010.

Member of Program Committee, *Tenth International Symposium on Functional and Logic Programming (FLOPS'10)*, Sendai, Japan, April 2010.

Member of Program Committee, *Fifth ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'10)*, Madrid, Spain, January 2010.

Member of Program Committee, *Fourth International Workshop on Systems Software Ver-*

ification (SSV'09), Aachen, Germany, June 2009.

Member of Program Committee, *Third International Workshop on Systems Software Verification (SSV'08)*, Sydney, Australia, February 2008.

Member of Program Committee, *Thirty-fifth ACM Symposium on Principles of Programming Languages (POPL'08)*, San Francisco, CA, January 2008.

Member of Program Committee, *First IEEE and IFIP International Symposium on Theoretical Aspects of Software Engineering (TASE'07)*, Shanghai, China, June 2007.

Member of Program Committee, *Eighth International Symposium on Trends in Functional Programming (TFP'07)*, New York, April 2007.

Member of Program Committee, *Sixteenth International Conference on Compiler Construction (CC'07)*, Braga, Portugal, March 2007.

Member of Editorial Board, *Journal of Computing Science and Engineering (JCSE)*, 2007–present.

Member of Program Committee, *Fourth Asian Symposium on Programming Languages and Systems (APLAS'06)*, Sydney, Australia, November 2006.

Member of Program Committee, *Fourth International Symposium on Automated Technology for Verification and Analysis (ATVA'06)*, Beijing, China, October 2006.

Member of Editorial Board, *Journal of Computer Science and Technology (JCST)*, 2006–present.

Member of Program Committee, *IJCAR Workshop on Programming Languages meets Program Verification (PLPV'06)*, Seattle, Washington, August 2006.

Member of Program Committee, *Seventh International Symposium on Trends in Functional Programming (TFP'06)*, Nottingham, UK, April, 2006.

Panel Organizer and Moderator, *The Future of Programming*, Yale Computer Science 35th Anniversary and Alumni Conference: Computer Science in the New Information Society, November 2005.

Member of Program Committee, *2005 ACM SIGPLAN Workshop on ML*, Tallinn, Estonia, September 2005.

Invited Speaker at the New England Programming Languages and Systems Symposium Series (NEPLS), Boston, MA, February 2005.

Member of Program Committee, *Thirty-second ACM Symposium on Principles of Programming Languages (POPL'05)*, Long Beach, CA, January 2005.

Invited Speaker on "The Essence of Proof-Carrying Code" at the *TYPES 2004 Conference*, Jouy-en-Josas, France, December 2004.

Member of Steering Committee, *ACM SIGPLAN International Conference on Functional Programming (ICFP)*, 2004–2006.

Member of Program Committee, *First Asian Symposium on Programming Languages and Systems (APLAS'03)*, Beijing, China, November 2003.

Member of Program Committee, *2003 ACM Workshop on Survivable and Self-Regenerative Systems (SSRS'03)*, Fairfax, VA, October 2003.

Member of Program Committee, *Eighth ACM SIGPLAN International Conference on Functional Programming (ICFP'03)*, Uppsala, Sweden, August 2003.

Member of Workshop Selection Committee, *2003 Conferences and Workshops on Principles, Logics, and Implementations of High-Level Programming Languages (PLI'03)*, Uppsala, Sweden, August 2003.

Member of Program Committee, *First International Workshop on Types in Programming (TIP'02)*, Dagstuhl, Germany, July 2002.

Member of Steering Committee, *ACM SIGPLAN Workshops on Types in Language Design and Implementation*, March 2002–present.

Invited Speaker, *Intel Research Forum on Language-Based Security*, Santa Clara, CA, January 2002.

Invited Speaker, *First International Workshop on Multi-Language Infrastructure and Interoperability (BABEL'01)*, Firenze, Italy, September 2001.

Invited Speaker, *Dagstuhl Seminar No. 01341 on Dependent Type Theory meets Practical Programming*, Dagstuhl, Germany, August 2001.

Invited Tutorial on Type-Based Certifying Compilation, *ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'01)*, Snowbird, UT, June 2001.

Moderator, *Programming Languages: Theory vs. Practice*, Alan J. Perlis Symposium, Sponsered by Department of Computer Science, Yale University, April 2000.

Panelist, *Typed Intermediate Languages for Compiling Object-Oriented Languages*, Seventh International Workshop on Foundations of Object-Oriented Languages, Boston, MA, January 2000.

Member of Program Committee, *ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'99)*, Atlanta, GA, May 1999.

Member of Program Committee, *Second ACM SIGPLAN International Workshop on Types in Compilation (TIC'98)*, Kyoto, Japan, March 1998.

Member of Program Committee, *Twenty-third ACM Symposium on Principles of Programming Languages (POPL'96)*, St. Petersburg, FL, January 1996.

Member of the ML2000 Working Group, 1993–2000.

Member of various review panels for National Science Foundation, 1996–present.

Reviewer for *Journal of Functional Programming*, *Software: Practice and Experience*, *ACM Transactions on Programming Languages and Systems*, *Journal of Information and Computation*, *ACM Transactions on Software Engineering and Methodology*, and a number of conferences on programming languages and compilers. Reviewer for *Cambridge University Press*, *Prentice Hall*, *McGraw Hill*, *Addison Wesley*, *Thomson* in the area of introductory programming, compilers, and programming languages. 1993–present.

Member of ACM, USENIX, and IEEE Computer Society, 1990–present

Teaching Experience

CS112 Introduction to Programming (five semesters).

CS210 A Second Course in Programming (two semesters).

CS421/521 Compilers and Interpreters (fifteen semesters).

CS422/522 Operating Systems (nine semesters).

CS428/528 Language-Based Security (six semesters).

CS430/530 Formal Semantics (five semesters).

CS535 Advanced Topics in Modern Compiler Implementation (one semester).

Graduate seminar on functional languages (two semesters).

Graduate seminar on secure internet programming (one semester).

Graduate seminar on understanding Java virtual machine (two semesters).

Students

Former postdocs and research scientists:

- Post-Doctoral Research Associate: Valery Trifonov (1997–2000).
- Post-Doctoral Research Associate: Nikolaos Papaspyrou (2000–2001).
- Software Engineer: Bandan Das (2012–2013). Current Employment: Software Engineer, Redhat.
- Post-Doctoral Research Associate: Liang Gu (2011–2015). Current Employment: CTO, Sangfor, China.
- Post-Doctoral Research Associate: Jan Hoffmann (2011–2015). Current Employment: Associate Professor, Carnegie Mellon University.
- Post-Doctoral Research Associate: Tahina Ramananandro (2012–2014). Current Employment: Senior Research Software Development Engineer, Microsoft Research.
- Post-Doctoral Research Associate: Hernan Vanzetto (2015–2017).
- Associate Research Scientist: Lionel Rieg (2016–2018). Current Employment: Assistant Lecturer, Verimag Labs, France.
- Post-Doctoral Research Associate: Pierre Wilke (2017–2018). Current Employment: Associate Professor, CentraleSupelec, France.

- Associate Research Scientist: Vilhelm Sjoberg (2015–2019). Current Employment: Principal Scientist, CertiK.
- Post-Doctoral Research Associate: Yuting Wang (2016–2019). Current Employment: Associate Professor, Shanghai Jiaotong University, China.
- Post-Doctoral Research Associate: Ji Yong Shin (2016–2020). Current Employment: Assistant Professor, Northeastern University.
- Associate Research Scientist: Jung-Eun Kim (2017–2021). Current Employment: Assistant Professor, North Carolina State University.
- Research Scientist: Man-Ki Yoon (2017–2022). Current Employment: Assistant Professor, North Carolina State University.
- Post-Doctoral Research Associate (CI Fellow): Anitha Gollamudi (2021–2022). Current Employment: Assistant Professor, University of Massachusetts at Lowell.
- Associate Research Scientist: Hao Chen (2019–2023). Current Employment: Researcher, CertiK Ltd.

Current postdocs and research scientists:

- Associate Research Scientist: Jeremie Koenig (2020–present).
- PostDoctoral Associate: Yoonseung Kim (2022–present).

Former Ph.D. students:

- Bratin Saha, Ph.D.(2002). Thesis title: *A Type System for Certified Runtime Type Analysis*. Current Employment:Vice President and General Manager, Machine Learning Services, Amazon AI.
- Christopher League, Ph.D.(2002). Thesis title: *A Type-Preserving Compiler Infrastructure*. Current Employment: Associate Professor, Long Island University.
- Stefan Monnier, Ph.D.(2003). Thesis title: *Principled Compilation and Scavenging*. Current Employment: Associate Professor, University of Montreal.
- Dachuan Yu (2004). Thesis title: *Safety Verification of Low-Level Code*. Current Employment: Software Architect, Orange Silicon Valley
- Nadeem A. Hamid (2004). Thesis title: *A Syntact Approach to Foundational Proof-Carrying Code*. Current Employment: Associate Professor, Berry College.
- Zhaozhong Ni (2006). Thesis title: *Modular Machine Code Verification*. Current Employment: VP of Engineering, CertiK Ltd
- Xinyu Feng (2007). Thesis title: *An Open Framework for Certified System Software*. Current Employment: Professor, Nanjing University.
- Hongxu Cai (2008). Thesis title: *Logic-based Verification of General Machine Code*. Current Employment: Staff Software Engineer, Google Inc. (Mountain View).
- Andrew McCreight (2008). Thesis title: *The Mechanized Verification of Garbage Collector Implementations*. Current Employment: Mozilla.
- Rodrigo Ferreira (2010). Thesis title: *Memory Consistency and Program Verification*. Current Employment: Co-founder, Pousadinhas.com.br Ltda.
- Alexander Vaynberg (2012). Thesis title: *Certifying Virtual Memory Manager Using Multiple Abstraction Levels*. Current Employment: Software Engineer, Google Inc. (New York).

- Antonis Stampoulis (2012). Thesis title: *VeriML: A Dependently-Typed, User-Extensible, and Language-Centric Approach to Proof Assistant*. Current Employment: Software Engineer and PL Researcher at Orignate NYC.
- Hongjin Liang (2014). Thesis title: *Refinement Verification of Concurrent Programs and Its Applications*. Current Employment: Associate Professor, Nanjing University.
- Shu-Chun Weng (2015). Thesis title: *DeepSpec: Modular Certified Programming with Deep Specifications*. Current Employment: Software Engineer, Google Inc. (Mountain View).
- David Costanzo (2016). Thesis title: *Formal End-to-End Verification of Information-Flow Security for Complex Systems*. Current Employment: Software Engineer, Google Inc. (Mountain View).
- Ronghui Gu (2016). Thesis title: *An Extensible Architecture for Building Certified Sequential and Concurrent OS Kernels*. Current Employment: Assistant Professor, Columbia University
- Quentin Carbonneaux (2018). Thesis title: *Modular and Certified Resource-Bound Analysis*. Current Employment: Researcher, Facebook.
- Newman Wu (2018). Thesis title: *A Compositional Automation Engine for Verifying Complex System Software*. Current Employment: Strategy and Core Developer, Virtu Financial.
- Jieung Kim (2019). Thesis title: *Modular and Compositional Development of Certified Concurrent Software Systems*. Current Employment: Assistant Professor, Yonsei University.
- Mengqi Liu (2020). Thesis title: *Real-Time CertiKOS: Compositional Verification of OS Kernels with Preemptive Scheduling and Temporal Isolation*. Current Employment: Researcher, Alibaba Group.
- Jeremie Koenig (2020). Thesis title: *Refinement-Based Game Semantics for Certified Components*. Current Employment: Associate Research Scientist, Yale University.
- Wolf Honore (2022). Thesis title: *The Atomic Distributed Object Model for Distributed System Verification*. Current Employment: Researcher, Amazon.
- Yuyang Sang (2024). Thesis title: *Objective DeepSEA: A Language for Developing Modular Certifiable System Software*. Current Employment: Researcher, Alibaba Group.

Current Ph.D. students:

- Richard Habeeb (2018–present). Research interest: *Programming Languages; Cyber-Physical Systems*.
- Tong Cheng (2018–present). Research interest: *Programming Languages; Formal Verification*.
- Arthur Oliveira Vale (2019–present). Research interest: *Programming Languages; Formal Verification*.
- Yixuan Chen (2019–present). Research interest: *Programming Languages; Operating Systems*.
- Yu Zhang (2019–present). Research interest: *Programming Languages; Formal Verification*
- Longfei Qiu Zhang (2021–present). Research interest: *Programming Languages; Formal Verification*

- Daniel Luick (2021–present). Research interest: *Programming Languages; Formal Verification*
- Ben Chaimberg (2021–present). Research interest: *Programming Languages; Formal Verification*
- Peixin You (2022–present). Research interest: *Programming Languages; Formal Verification*
- Christian Altamirano Modesto (2022–present). Research interest: *Programming Languages; Formal Verification*
- Justin Restivo (2023–present). Research interest: *Programming Languages; Formal Verification*
- Ben Siraphob (2023–present). Research interest: *Programming Languages; Formal Verification*
- Zhongye Wang (2024–present). Research interest: *Programming Languages; Formal Verification*

Undergraduate students (advising their senior projects): Chris Volkert (1995), Jonathan Traupman (1996), Lujo Bauer (1997), Ben Zhao (1997), Alex Hehmeyer (1997), Kenny Wolf (1997), Jesse Heitler (1997), Bret Martin (1997), David Auerbach (1998), Neil Inala (1998), John Richter (1999), Benjamin Christen (2000), John Garvin (2000-2001), Yichen Xie (2000), Daniel Dormont (2001), John Starks (2007), Alexander Thomson (2008), Aarlo Stone-Fish (2009), Eric Love (2011), John Whittaker (2014), Alexander Dobner (2015), Jacob Geiger (2015), Alexander Lew (2015), Hugh O'Cinneide (2015), Hengchu Zhang (2015), Jonathon Cai (2015), Jason Liu (2015), Adam Cimpeanu (2016), Kevin Abbott (2016), Jack Siegel (2017), Justin Wang (2017), Christopher Fu (2017-2018), Lee Danilek (2018), Valerie Chen (2018-2019), Collin Bentley (2019), Bradley Yam (2021), Kevin Tang (2021), Johan Todi (2021-2022), Daniel Hodeta (2022), Cody Lin (2022), Ben Cifu (2023), and Dylan Vroon (2023).

Research interns and visitors: Rudi Seitz (1996), Neil Inala (1996), Sukyoung Ryu (1999), Oukseh Lee (1999), David Teller (2000), Yichen Xie (2000-2001), John Garvin (2000-2001), Yuan Dong (2007-2008), Wei Wang (2008-2010), Ming Fu (2009-2010), Yong Li (2009-2010), Guillaume Claret (2010), Xinyu Jiang (2010-2011), Yu Zhang (2010-2011), Xinyu Feng (2011), Zhong Zhuang (2010-2012), Haozhong Zhang (2011-2013), Jinjiang Lei (2011-2013), Lin Yan (2012-2014), Hongjin Liang (2012-2013), Yu Guo (2012-2013), Yang Zhang (2013-2014), Hao Chen (2014-2016), Chanik Park (2015-2016), Zining Cao (2015-2016), Xiaorui Zhu (2015), Zefeng Zeng (2015-2016), Zhencao Zhang (2015-2016), Haiyong Sun (2016-2017), Lei Qiao (2016-2017), Yi Lyu (2017), Chen Liang (2017), Zhenguo Yin (2018), Xiaoqiang Wu (2020), Matthew Tu (2020), Alex Briasco-Stewart (2020-2021), Jeacy Espinoza (2021), Brandon Liu (2021), Jacob Dunefsky (2021), Albert Gong (2022), Gabriele Vanoni (2023), Zhongye Wang (2023), Haoran Ding (2023-2024), Eashan Hatti (2023-present), Jingqi Xiao (2024), and Yueyang Feng (2024).

Member of the Ph.D. thesis committee: Jan-Jan Wu (1995), Satish Pai (1996), Rajiv Mirani (1996), Kevin Lynch (1996), Sheng Liang (1997), Chih-Ping Chen (1999), Martin Sulzmann (1999), Mark Tullsen (2001), Bratin Saha (2002), Christopher League (2002), Zhanyong Wan (2002), Stefan Monnier (2003), Juan Chen (2004), Anthony Courtney (2004), Dachuan Yu (2004), Nadeem A. Hamid (2004), Zhaozhong Ni (2006), Xinyu Feng (2007), Liwen Huang (2008), Andrew McCreight (2008), Adam Poswalski (2008), Jeffrey Sarnat (2009), Rodrigo Ferreira (2010), Paul Liu (2011), Jan Hoffmann (2011), Alexander Vaynberg (2012), Antonis Stampoulis (2012), Amittai Aviram (2012), Adam Wright (2013),

Hongjin Liang (2014), Donya Quick (2014), Daniel Winograd-Cort (2015), Gordon Stewart (2015), Weiyi Wu (2015), Shu-Chun Weng (2015), Ronghui Gu (2016), David Costanzo (2016), Quentin Carbonneaux (2018), Joshua Lockerman (2018), Newman Wu (2018), Jieung Kim (2019), Mark Santoluctio (2020), Mengqi Liu (2020), Jeremie Koenig (2020), William Hallahan (2022), Wolf Honore (2022), Ning Luo (2023), Jialu Zhang (2023), Xiang Wu (2023), Samuel Judson (2023), and Yuyang Sang (2024).

University Activities

Department Chair, Yale Computer Science, 2017-2023.
Director of Undergraduate Studies, Yale Computer Science, 2003-2006, 2015.
Acting Chair, Graduate Admission Committee, Yale Computer Science, 2001.
Member, Graduate Admission Committee, Yale Computer Science, 1995-2000, 2008-2016.
Member, Advancement Committee for Engineering, Yale University, 2015-2017
Organizer, Weekly Systems Seminar (SPAM), Yale Computer Science, 1994-1996.
Organizer, Yale Computer Science Alan J. Perlis Symposium, 2000-2001.
Member, GSAS Allocation Committee, Yale University, 2013-2014.
Member, Ph.D. Comprehensive Exam Committee, Yale Computer Science, 1995-2001.
Member, Computing Committee, Yale Computer Science, 2003-2005.
Member, Financial Committee, Yale Computer Science, 2005-2006.
Member, Faculty Recruiting Committee, Yale Computer Science, 2005-2012, 2016.
Chair, Faculty Recruiting Committee, Yale Computer Science, 2015, 2017, 2023-2024.
Member, Teaching and Curriculum Committee, Yale Computer Science, 1996-2006.
Member, Curriculum 200X Committee, Yale Computer Science, 2001-2003.
Member, Library Committee, Yale Computer Science, 1996-2001.
First-year Graduate Student Coordinator, Yale Computer Science, 1997-2000.
Fellow, Silliman College, Yale University, 1995-present.
Freshman Advisor, Silliman College, Yale University, 1995-present.
Sophomore Advisor, Yale Computer Science, 1999-present.