Alley Stoughton

45 Custer St, PO Box 300047
Jamaica Plain, MA 02130, USA
alley.stoughton@icloud.com
alleystoughton.us
+1.785.341.3041

January 31, 2025

Education

University of Edinburgh
PhD Computer Science, January 1987

University of California, Los Angeles
MS Computer Science, December 1981
BS Mathematics/Computer Science, June 1979

Employment

Department of Computer Science, Boston University
Research Professor, October 2018–present

Hariri Institute for Computing and Computational Science & Engineering, Boston University
Research Fellow, March 2017–December 2018

IMDEA Software Institute
Researcher, October 2015–August 2016

MIT Lincoln Laboratory
Technical Staff, September 2012–April 2015

Department of Computer Science, Tufts University
Lecturer, January–May 2012

Department of Computing and Information Sciences, Kansas State University
Associate Professor, August 1993–May 2010

School of Cognitive and Computing Sciences, University of Sussex
Lecturer in Computer Science, April 1988–July 1993
Research Fellow, September 1986–March 1988

Department of Computer Sciences, Chalmers University of Technology
Visiting Research Fellow, January–July 1986

Information Sciences Institute, Los Angeles
Research Assistant, 1981–82

Computer Science Department, University of California, Los Angeles
Programmer, 1980–81 and 1976–79
Teaching Assistant, 1979–80

Publications

A. Stoughton, C. Chen, M. Gaboardi and W. Qu. Formalizing Algorithmic Bounds in the Query Model in EasyCrypt. *Proceedings of the 13th International Conference on Interactive Theorem Proving (ITP 2022)*, pp. 30:1–30:21. Schloss Dagstuhl—Leibniz-Zentrum für Informatik, 2022.

A. Stoughton and M. Vassena. PLAS'20: 15th Workshop on Programming Languages and Analysis for Security. *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS 2020)*, pp. 2151–2152. ACM, 2020.

J. B. Almeida, C. Baritel-Ruet, M. Barbosa, G. Barthe, F. Dupressoir, B. Grégoire, V. Laporte, T. Oliveira, A. Stoughton and P.-Y. Strub. Machine-Checked Proofs for Cryptographic Standards: Indifferentiability of Sponge and Secure High-Assurance Implementations of SHA-3. *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS 2019)*, pp. 1607–1622. ACM, 2019. Also available as Report 2019/1155 of *Cryptology ePrint Archive*, 2019.

R. Canetti, A. Stoughton and M. Varia. EasyUC: Using EasyCrypt to Mechanize Proofs of Universally Composable Security. *Proceedings of the 32nd IEEE Computer Security Foundations Symposium (CSF '19)*, pp. 167–183. IEEE Computer Society, 2019. An extended version of this paper is available as Report 2019/582 of *Cryptology ePrint Archive*, 2019.

A. Stoughton and M. Varia. Mechanizing the Proof of Adaptive, Information-theoretic Security of Cryptographic Protocols in the Random Oracle Model. *Proceedings of the 30th IEEE Computer Security Foundations Symposium (CSF '17)*, pp. 83–99. IEEE Computer Society, 2017.

J. Hughes, C. Sparks, A. Stoughton, R. Parikh, A. Reuther and S. Jagannathan. Building Resource Adaptive Software Systems (BRASS): Objectives and System Evaluation. *SIGSOFT Softw. Eng. Notes*, vol. 41, no. 1. ACM, 2016.

A. Stoughton, A. Johnson, S. Beller, D. Chen, K. Chadha, K. Foner and M. Zhivich. You Shot My Battleship! A Case Study in Secure Programming. *Proceedings of the Ninth Workshop on Programming Languages and Analysis for Security (PLAS '14)*, pp. 2–14. ACM, 2014.

A. Stoughton. Experimenting with Formal Languages Using Forlan. *FDPE '08: Proceedings of the 2008 International Workshop on Functional and Declarative Programming in Education*, pp. 41–50. ACM, 2008.

A. Stoughton. A Functional Model-View-Controller Software Architecture for Command-oriented Programs. *WGP '08: Proceedings of the ACM SIGPLAN Workshop on Generic Programming*, pp. 1–12. ACM, 2008.

A. Stoughton. Experimenting with Formal Languages. *36th SIGCSE Technical Symposium on Computer Science Education*, workshop abstract, p. 566. ACM, 2005.

C. Haack, B. Howard, A. Stoughton and J. Wells. Fully Automatic Adaptation of Software Components Based on Semantic Specifications. *9th International Conference on Algebraic Methodology and Software Technology (AMAST)*, Lecture Notes in Computer Science, vol. 2422, pp. 83–98. Springer-Verlag, 2002.

A. Stoughton. Infinite Pretty-printing in eXene. *Trends in Functional Programming*, vol. 3, pp. 13–24. Intellect, 2002.

A. Stoughton. An Operational Semantics Framework Supporting the Incremental Construction of Derivation Trees. *Second Workshop on Higher-Order Operational Techniques in Semantics (HOOTS II)*, Electronic Notes in Theoretical Computer Science, vol. 10, 12 pp. Elsevier Science B. V., 1998.

A. Stoughton. Porgi: a Proof-Or-Refutation Generator for Intuitionistic propositional logic. *CADE-13 Workshop on Proof Search in Type-Theoretic Languages*, Rutgers University, pp. 109–116, 1996.

A. Stoughton. Mechanizing logical relations. *Ninth International Conference on the Mathematical Foundations of Programming Semantics*, Lecture Notes in Computer Science, vol. 802, pp. 359–377. Springer-Verlag, 1994.

A. Jung and A. Stoughton. Studying the fully abstract model of PCF within its continuous function model. *International Conference on Typed Lambda Calculi and Applications*, Lecture Notes in Computer Science, vol. 664, pp. 230–244. Springer-Verlag, 1993.

A. Stoughton. Parallel PCF has a unique extensional model. *Sixth Annual IEEE Symposium on Logic in Computer Science*, pp. 146–151. IEEE, 1991.

A. Stoughton. Interdefinability of parallel operations in PCF. *Theoretical Computer Science*, 79:357–358, 1991.

A. Stoughton. Equationally fully abstract models of PCF. *Fifth International Conference on the Mathematical Foundations of Programming Semantics*, Lecture Notes in Computer Science, vol. 442, pp. 271–283. Springer-Verlag, 1990.

A. Stoughton. *Fully Abstract Models of Programming Languages.* Research Notes in Theoretical Computer Science, 123 pp. Pitman/Wiley, 1988. A revision with additions of the University of Edinburgh PhD thesis of the same name, Technical Report CST–40–86, Computer Science Department, University of Edinburgh, 1986.

A. Stoughton. Substitution revisited. *Theoretical Computer Science*, 59:317–325, 1988. Previously appeared as Technical Report 1/87, Computer Science Subject Group, University of Sussex, 1987.

D. Parker, G. Popek, G. Rudisin, A. Stoughton, B. Walker, E. Walton, J. Chow, D. Edwards, S. Kiser and C. Kline. Detection of mutual inconsistency in distributed systems. *IEEE Transactions on Software Engineering*, SE–9(3):240–247, 1983.

V. Kini, D. Martin and A. Stoughton. Tools for testing denotational semantic definitions of programming languages. Technical Report ISI/RR–83–112, 77 pp., Information Sciences Institute, 1983.

V. Kini, D. Martin and A. Stoughton. Testing the INRIA Ada formal definition: The USC-ISI formal semantics project. *ADATec Conference on Ada*, pp. 120–128. ACM, 1982.

A. Stoughton. Access Flow: A protection model which integrates access control and information flow. *1981 Symposium on Security and Privacy*, pp. 9–18. IEEE, 1981.

G. Popek, M. Kampe, C. Kline, A. Stoughton, M. Urban and E. Walton. UCLA Secure Unix. *National Computer Conference*, pp. 355–364, 1979.

Workshops

A. Stoughton. Experimenting with Formal Languages. Workshop given at the *36th SIGCSE Technical Symposium on Computer Science Education*, February 23, 2005.

Grants

Principal investigator together with co-principal investigators R. Canetti and M. Varia of Boston University subaward from Riverside Research (principal investigator

M. Clark) of DARPA HARDEN grant *Universal Composability for Preventing Adversarial Composition Techniques (UC-PACT)*, 2022-2024.

Co-principal investigator with a. shelat (principal investigator), A. Ahmed, D. Wichs, V. Vaikuntanathan, A. Chlipala, R. Canetti, A. Bestavros, A. Lapets, A. Stoughton, M. Venkitasubramaniam of IARPA HECTOR grant *ACHILLES: Assured CryptograpHic Integration of muLtiple Languages for Encypted Systems*, 2019–2020.

Principal investigator, together with co-principal investigators R. Canetti, A. Kfoury and M. Varia, of NSF grant CNS-1801564 *Towards Mechanized Proofs of Composable Security Properties*, 2018–2022 (extended 2023).

Co-principal investigator with D. Schmidt, B. Howard and M. Dwyer of NSF/DARPA grant CCR-9633388, *Logical Support for High-Assurance Software Evolution*, 1996–98.

Co-principal investigator with D. Schmidt and B. Howard of ONR grant N 00014-94-1-0866, *Type Theories in ML*, 1994–96.

Co-principal investigator with D. Schmidt of NSF grant CCR-9302962, *Analysis and Classification of Programming Languages*, 1993–96.

Co-principal investigator with E. Robinson of ESPRIT grant BRA 6811, *Categorical Logic in Computer Science* (Sussex University was an associate site), 1992–95.