Andrew Donald Gordon

Partner Research Manager, Microsoft Research
Honorary Professor, University of Edinburgh
https://www.microsoft.com/en-us/research/people/adg/
https://www.linkedin.com/in/andrew-d-gordon/

September 2023

Degrees

- BSc (Computer Science, First Class Honours) 1987, University of Edinburgh.
- PhD (Computer Science) 1992, University of Cambridge.

Awards and Honours

- Distinguished Dissertation in Computer Science, jointly awarded by the British Computer Society and the Conference of Professors and Heads of Computing, 1993.
- Most Influential ETAPS 1998 Paper, *Mobile Ambients*, with L. Cardelli, awarded by the European Association for Programming Languages and Systems, 2007.
- Most Influential POPL 2000 Paper, *Anytime, Anywhere: Modal Logics for Mobile Ambients*, with L. Cardelli, awarded by ACM SIGPLAN, 2010.
- Best Paper ETAPS 2013, *Deriving Probability Density Functions from Probabilistic Functional Programs*, with S. Bhat, J. Borgström, and C. Russo, awarded by the European Association for Programming Languages and Systems, 2013.
- ACM Fellow 2020, for "For contributions to programming languages: their principles, logic, usability, and trustworthiness." (ACM is the leading international association of computing professionals. ACM's most prestigious member grade recognizes the top 1% of ACM members for their outstanding accomplishments in computing and information technology and/or outstanding service to ACM and the larger computing community.)
- Best paper, Honorable mention for ACM CHI 2023 paper, *"What It Wants Me To Say": Bridging the Abstraction Gap Between End-User Programmers and Code-Generating Large Language Models*, with Michael Xieyang Liu, Advait Sarkar, Carina Negreanu, Ben Zorn, Jack Williams, and Neil Toronto.
- Best paper, Honorable mention for IEEE VL/HCC 2023 paper, *FxD: a functional debugger for dysfunctional spreadsheets*, with Ian Drosos, Nicholas Wilson, Sruti Srinivasa Ragavan, and Jack Williams.
- Honorary Professor, University of Edinburgh, since August 2023.

Professional Experience

- Research assistant, University of Edinburgh, summer 1987.
- Summer intern, Digital Systems Research Center, summer 1989.
- Research assistant, University of Cambridge, January 1991–October 1992.
- Visiting researcher and lecturer, Chalmers University, January–December 1993.
- Consultant, Lloyds Register, London, winter 1993.
- Research associate, University of Cambridge, January 1994–September 1994.
- Royal Society University Research Fellow, University of Cambridge, October 1994–October 1997.
- Consultant, Digital Systems Research Center, August 1996.
- Consultant, Digital Systems Research Center, March 1997.
- Researcher, Microsoft Research, November 1997–August 2002.
- Visiting Professor at the University of Provence, Marseille, April 1998.
- Senior Researcher, Microsoft Research, August 2002–August 2007.
- Visiting Professor at the University of Newcastle, March 2007–March 2010.
- Principal Researcher, Microsoft Research, August 2007–December 2012.
- Professor of Computer Security, University of Edinburgh, October 2010–July 2023.
- Principal Researcher / Joint Research Area Leader, Microsoft Research, December 2012–November 2013.
- Principal Researcher / Research Area Leader, Microsoft Research, November 2013–December 2017.
- Senior Principal Research Manager, Microsoft Research, January 2018–August 2023.
- Partner Research Manager, Microsoft Research, since September 2023.

Publications

Journal Publications

1. M. Abadi and A. D. Gordon. A bisimulation method for cryptographic protocols. *Nordic Journal of Computing*, 5:267–303, 1998.

2. R. L. Crole and A. D. Gordon. Relating operational and denotational semantics for input/output effects. *Mathematical Structures in Computer Science*, 9:125–158, 1999.

3. M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The spi calculus. *Information and Computation*, 148:1–70, 1999.

4. A. D. Gordon. Bisimilarity as a theory of functional programming. *Theoretical Computer Science*, 228:5–47, 1999.

5. A. D. Gordon, S. B. Lassen, and P. D. Hankin. Compilation and equivalence of imperative objects. *Journal of Functional Programming*, 9(4):373–426, 1999.

6. L. Cardelli and A. D. Gordon. Mobile ambients. *Theoretical Computer Science*, 240:177-213, 2000.

7. L. Cardelli, G. Ghelli, and A. D. Gordon. Types for the ambient calculus. *Information and Computation*, 177:160–194, 2002.

8. A. D. Gordon and L. Cardelli. Equational properties of mobile ambients. *Mathematical Structures in Computer Science*, 12:1–38, 2002.

9. S. Dal Zilio and A. D. Gordon. Region analysis and a π-calculus with groups. *Journal of Functional Programming*, 12(3):229–292, 2002.

10. W. Charatonik, S. Dal Zilio, A. D. Gordon, S. Mukhopadhyay, and J.-M. Talbot. The complexity of model checking mobile ambients. In *Theoretical Computer Science*, 308:277-331, 2003.

11. A. D. Gordon and A. Jeffrey. Typing correspondence assertions for communication protocols. *Theoretical Computer Science*, 300:379–409, 2003.

12. A. D. Gordon and A. Jeffrey. Authenticity by typing for security protocols. *Journal of Computer Security*, 11(4):451–521, 2003.

13. A. D. Gordon and A. Jeffrey. Types and effects for asymmetric cryptographic protocols. *Journal of Computer Security*, 12(3/4):435–484, 2003.

14. K. Bhargavan, C. Fournet, and A. D. Gordon. A semantics for web services authentication. *Theoretical Computer Science*, 340(1):102–153, 2005.

15. A. D. Gordon and R. Pucella. Validating a web service security abstraction by typing. *Formal Aspects of Computing*, 17:277–318, 2005.

16. L. Cardelli, G. Ghelli, and A. D. Gordon. Secrecy and group creation. *Information and Computation*, 196(2):127–155, 2005.

17. C. Calcagno, L. Cardelli, and A. D. Gordon. Deciding validity in a spatial logic for trees. *Journal of Functional Programming*, 15:543–572, 2005.

18. L. Cardelli and A. D. Gordon. Ambient logic. *Mathematical Structures in Computer Science*. To appear.

19. K. Bhargavan, R. Corin, C. Fournet, and A. D. Gordon. Secure sessions for Web services. *ACM Transactions on Information and System Security*, 10(2), 2007.

20. C. Fournet, A. D. Gordon, and S. Maffeis. A type discipline for authorization policies. *ACM Transactions on Programming Languages and Systems*, 29(5), 2007.

21. K. Bhargavan, C. Fournet, A. D. Gordon. Verifying policy-based web services security. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 30(6), 2008.

22. K. Bhargavan, C. Fournet, A.D. Gordon and S. Tse. Verified interoperable implementations of security protocols. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 31(1), 2008.

23. M. Becker, C. Fournet, and A. D. Gordon. SecPAL: Design and Semantics of a Decentralized Authorization Language. *Journal of Computer Security*, 18(4):597–643, 2010.

24. J. Bengtson, K. Bhargavan, C. Fournet, and S. Maffeis. Refinement types for secure implementations. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 33(2):8, 2011.

25. J. Borgström, A. D. Gordon, R. Pucella. Roles, stacks, histories: A triple for Hoare. *Journal of Functional Programming (JFP)*, 21(2):159–207, 2011.

26. G. M. Bierman, A. D. Gordon, C. Hriţcu, David E. Langworthy. Semantic subtyping with an SMT solver. *Journal of Functional Programming (JFP)* 22(1):31–105, 2012.

27. J. Borgström, A. D. Gordon, M. Greenberg, J. Margetson, J. Van Gael. Measure Transformer Semantics for Bayesian Machine Learning. *Logical Methods in Computer Science* 9(3), 2013.

28. F. Dupressoir, A. D. Gordon, J. Jürjens, D. A. Naumann. Guiding a General-Purpose C Verifier to Prove Cryptographic Protocols. *Journal of Computer Security* 22(5):823–866, 2014.

29. V. Vaglica, M. Sajeva, H. N. McGough, D. Hutchison, C. Russo, A. D. Gordon, A. V. Ramarosandratana, W. Stuppy, M. J. Smith. Monitoring internet trade to inform species conservation actions. *Endangered Species Research* 32:223–235, 2017.

30. Sooraj Bhat, Johannes Borgström, Andrew D. Gordon, and Claudio V. Russo. Deriving probability density functions from probabilistic functional programs. *Logical Methods in Computer Science*, 13(2), 2017.

31. Matt McCutchen, Judith Borghouts, Andrew D. Gordon, Simon Peyton Jones, and Advait Sarkar. Elastic sheet-defined functions: Generalising spreadsheet functions to variable-size input arrays. *J. Funct. Program.*, 30:e26, 2020.

32. Maria I. Gorinova, Andrew D. Gordon, Charles Sutton, and Matthijs Vákár. Conditional independence by typing. *ACM Trans. Program. Lang. Syst.*, 44(1):4:1–4:54, 2022.

33. Shuang Chen, Alperen Karaoglu, Carina Negreanu, Tingting Ma, Jin-Ge Yao, Jack Williams, Feng Jiang, Andy Gordon, and Chin-Yew Lin. LinkingPark: An automatic semantic table interpretation system. *J. Web Semant.*, 74:100733, 2022.

Refereed Conference and Workshop Publications

1. A. D. Gordon. The formal definition of a synchronous hardware-description language in higher order logic. In *International Conference on Computer Design, Cambridge, Massachusetts, October 11–14, 1992*, pages 531–534. IEEE Computer Society Press, 1992.

2. A. D. Gordon. An operational semantics for I/O in a lazy functional language. In *FPCA'93: Conference on Functional Programming Languages and Computer Architecture, Copenhagen*, pages 136–145. ACM Press, 1993.

3. R. L. Crole and A. D. Gordon. Factoring an adequacy proof (preliminary report). In *Functional Programming, Glasgow 1993*, Workshops in Computing, pages 9–27. Springer-Verlag, 1994.

4. A. D. Gordon. A mechanisation of name-carrying syntax up to alpha-conversion. In J. J. Joyce and C.-J. H. Seger, editors, *Higher Order Logic Theorem Proving and its Applications. Proceedings, 1993*, number 780 in Lecture Notes in Computer Science, pages 414–426. Springer-Verlag, 1994.

5. R. L. Crole and A. D. Gordon. A sound metalogical semantics for input/output effects. In L. Pacholski and J. Tiuryn, editors, *CSL'94 Computer Science Logic, Kazimierz, Poland, September 1994*, volume 933 of *Lecture Notes in Computer Science*, pages 339–353. Springer-Verlag, 1995.

6. A. D. Gordon. A tutorial on co-induction and functional programming. In *Functional Programming, Glasgow 1994*, Workshops in Computing, pages 78–95. Springer-Verlag, 1995.

7. A. D. Gordon. Bisimilarity as a theory of functional programming. In *Eleventh Annual Conference on Mathematical Foundations of Programming Semantics*, volume 1 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers B.V., 1995.

8. A. D. Gordon and K. Hammond. Monadic I/O in Haskell 1.3. In Paul Hudak, editor, *Proceedings of the Haskell Workshop, June 25, 1995, La Jolla, California*, pages 50–68, 1995. Available as Yale University Research Report YALEU/DCS/RR–1075.

9. A. D. Gordon and G. D. Rees. Bisimilarity for a first-order calculus of objects with subtyping. In *23rd ACM Symposium on Principles of Programming Languages (POPL'96)*, pages 386–395. ACM Press, 1996.

10. S. L. Peyton Jones, A. D. Gordon, and S. Finne. Concurrent Haskell. In *23rd ACM Symposium on Principles of Programming Languages (POPL'96)*, pages 295–308. ACM Press, 1996.

11. A. D. Gordon and T. Melham. Five axioms of alpha-conversion. In *Theorem Proving in Higher Order Logics: 9th International Conference, TPHOLs'96*, volume 1125 of *Lecture Notes in Computer Science*, pages 173–191. Springer-Verlag, 1996.

12. M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The spi calculus. In *4th ACM Conference on Computer and Communications Security (CCS'97)*, pages 36–47. ACM Press, April 1997.

13. M. Abadi and A. D. Gordon. Reasoning about cryptographic protocols in the spi calculus. In *Concurrency Theory (CONCUR'97)*, Lecture Notes in Computer Science, pages 59–73. Springer-Verlag, August 1997.

14. A. D. Gordon, S. B. Lassen, and P. D. Hankin. Compilation and equivalence of imperative objects. In *Foundations of Software Technology and Theoretical Computer Science (FST&TCS'97)*, volume 1346 of *Lecture Notes in Computer Science*, pages 74–87. Springer-Verlag, 1997.

15. A. D. Gordon. Operational equivalences for untyped and polymorphic object calculi. In A. D. Gordon and A. M. Pitts, editors, *Higher Order Operational Techniques in Semantics*, Publications of the Newton Institute, pages 9–54. Cambridge University Press, 1998.

16. M. Abadi and A. D. Gordon. A bisimulation method for cryptographic protocols. In *European Symposium on Programming (ESOP'98)*, volume 1381 of *Lecture Notes in Computer Science*, pages 12–26. Springer-Verlag, 1998.

17. L. Cardelli and A. D. Gordon. Mobile ambients. In *Foundations of Software Science and Computation Structures (FOSSACS'98)*, volume 1378 of *Lecture Notes in Computer Science*, pages 140–155. Springer-Verlag, 1998.

18. A. D. Gordon and P. D. Hankin. A concurrent object calculus: reduction and typing. In *3rd International Workshop on High-Level Concurrent Languages (HLCL'98)*, volume 16 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 1998.

19. L. Cardelli and A. D. Gordon. Types for mobile ambients. In *26th ACM Symposium on Principles of Programming Languages (POPL'99)*, pages 79–92. ACM Press, 1999.

20. L. Cardelli, G. Ghelli, and A. D. Gordon. Mobility types for mobile ambients. In *International Conference on Automata, Languages, and Programming (ICALP'99)*, volume 1644 of *Lecture Notes in Computer Science*, pages 230–239. Springer-Verlag, 1999.

21. A. D. Gordon and L. Cardelli. Equational properties of mobile ambients. In *Foundations of Software Science and Computation Structures*, Lecture Notes in Computer Science, pages 212–226. Springer-Verlag, 1999.

22. L. Cardelli and A. D. Gordon. Anytime, anywhere: Modal logics for mobile ambients. In *27th ACM Symposium on Principles of Programming Languages (POPL'00)*, pages 365–377. ACM Press, 2000.

23. L. Cardelli, G. Ghelli, and A. D. Gordon. Ambient groups and mobility types. In *Proceedings TCS2000*, volume 1872 of *Lecture Notes in Computer Science*, pages 333–347. Springer-Verlag, 2000.

24. L. Cardelli, G. Ghelli, and A. D. Gordon. Secrecy and group creation. In *Concurrency Theory (CONCUR'00)*, volume 1877 of *Lecture Notes in Computer Science*, pages 365–379. Springer-Verlag, 2000.

25. A. D. Gordon and D. Syme. Typing a multi-language intermediate code. In *28th ACM Symposium on Principles of Programming Languages (POPL'01)*, pages 248–260, 2001.

26. W. Charatonik, S. Dal Zilio, A. D. Gordon, S. Mukhopadhyay, and J.-M. Talbot. The complexity of model checking mobile ambients. In *Foundations of Software Science and Computation Structures (FOSSACS'01)*, volume 2030 of *Lecture Notes in Computer Science*, pages 152–167. Springer-Verlag, 2001.

27. L. Cardelli and A. D. Gordon. Logical properties of name restriction. In *Typed Lambda Calculi and Applications (TLCA'01)*, volume 2044 of *Lecture Notes in Computer Science*, pages 46–60. Springer-Verlag, 2001.

28. A. D. Gordon and A. Jeffrey. Typing correspondence assertions for communication protocols. In *Mathematical Foundations of Programming Semantics 17*, Electronic Notes in Theoretical Computer Science. Elsevier, 2001. Pages 99–120 of the Preliminary Proceedings, BRICS Notes Series NS-01-2, BRICS, University of Aarhus, May 2001.

29. A. D. Gordon and A. Jeffrey. Authenticity by typing for security protocols. In *14th IEEE Computer Security Foundations Workshop*, pages 145–159. IEEE Computer Society Press, 2001.

30. C. Fournet and A. D. Gordon. Stack inspection: Theory and variants. In *28th ACM Symposium on Principles of Programming Languages (POPL'02)*, pages 307–318, 2002.

31. W. Charatonik, A. D. Gordon, and J.-M. Talbot. Finite-control mobile ambients. In *European Symposium on Programming (ESOP'02)*, volume 2305 of *Lecture Notes in Computer Science*, pages 295–313. Springer-Verlag, 2002.

32. A. D. Gordon and A. Jeffrey. Types and effects for asymmetric cryptographic protocols. In *15th IEEE Computer Security Foundations Workshop*, pages 77–91. IEEE Computer Society Press, 2002.

33. D. Syme and A. D. Gordon. Automating type soundness proofs via decision procedures and guided reductions. In *9th International Conference on Logic for Programming Artificial Intelligence and Reasoning*, volume 2514 of *Lecture Notes in Computer Science*, pages 418–434. Springer-Verlag, 2002.

34. A. D. Gordon and R. Pucella. Validating a web service security abstraction by typing. In *2002 ACM Workshop on XML Security*, pages 18–29, 2002.

35. A. D. Gordon and A. Jeffrey. Typing one-to-one and one-to-many correspondences in security protocols. In *Software Security - Theories and Systems (ISSS 2002)*, volume 2609 of *Lecture Notes in Computer Science*, pages 263–282. Springer-Verlag, 2002.

36. C. Calcagno, L. Cardelli, and A. D. Gordon. Deciding validity in a spatial logic for trees. In *ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI)*, pages 62–73, 2003.

37. K. Bhargavan, C. Fournet, and A. D. Gordon. A semantics for web services authentication. In *31st ACM Symposium on Principles of Programming Languages (POPL'04)*, pages 198–209, 2004.

38. K. Bhargavan, C. Fournet, and A. D. Gordon. Verifying policy-based security for web services. In *11th ACM Conference on Computer and Communications Security (CCS'04)*, pages 268–277, October 2004.

39. K. Bhargavan, R. Corin, C. Fournet, and A. D. Gordon. Secure sessions for web services. In *2004 ACM Workshop on Secure Web Services (SWS'04)*, pages 56–66, 2004.

40. C. Fournet, A. D. Gordon, and S. Maffeis. A type discipline for authorization policies. In *European Symposium on Programming (ESOP'05)*, volume 3444 of *Lecture Notes in Computer Science*, pages 141–156. Springer-Verlag, 2005.

41. A. D. Gordon and A. Jeffrey. Secrecy despite compromise: Types, cryptography, and the pi-calculus. In *Concurrency Theory (CONCUR'05)*, volume 3653 of *Lecture Notes in Computer Science*, pages 186–201. Springer-Verlag, 2005.

42. K. Bhargavan, C. Fournet, A. D. Gordon, and G. O'Shea. An advisor for web services security policies. In *2005 ACM Workshop on Secure Web Services*, pages 1–9. ACM, 2005.

43. K. Bhargavan, C. Fournet, A. D. Gordon, and S. Tse. Verified interoperable implementations of security protocols. In *19th IEEE Computer Security Foundations Workshop (CSFW'06)*, pages 139–152, 2006.

44. K. Bhargavan, C. Fournet, and A. D. Gordon. Verified reference implementations of WS-Security protocols. In *3rd International Workshop on Web Services and Formal Methods (WS-FM 2006)*, volume 4184 of *Lecture Notes in Computer Science*, pages 88–106. Springer-Verlag, 2006.

45. M. Becker, C. Fournet, and A. D. Gordon. SecPAL: Design and Semantics of a Decentralized Authorization Language. In *20th IEEE Symposium on Computer Security Foundations (CSF'07)*, pages 3-15, Venice, July 6-8, 2007.

46. C. Fournet, A. D. Gordon, and S. Maffeis. A type discipline for authorization in distributed systems. In *20th IEEE Symposium on Computer Security Foundations (CSF'07)*, pages 31-48, Venice, July 6-8, 2007. IEEE Computer Society.

47. J. Borgström, A. D. Gordon, and A. Phillips. A chart semantics for the pi-calculus. In *14th International Workshop on Expressiveness in Concurrency (EXPRESS'07)*, Electronic Notes in Theoretical Computer Science 194(2):3-29 (January 2008).

48. K. Bhargavan, C. Fournet, A. D. Gordon, and N. Swamy. Verified implementations of the information card federated identity-management protocol. In *ACM Symposium on Information, Computer and Communication Security (ASIACCS '08)*, Tokyo, March 18-20, 2008.

49. K. Bhargavan, A. D. Gordon, and I. Narasamdya. Service combinators for farming virtual machines. In *COORDINATION 2008*, Oslo, June 4-6, 2008. Springer LNCS 5052:33-49.

50. J. Bengtson, K. Bhargavan, C. Fournet, and S. Maffeis. Refinement types for secure implementations. In *20th IEEE Computer Security Foundations Symposium (CSF 2008)*, pages 17-32, Pittsburgh, Pennsylvania, June 23-25, 2008. IEEE Computer Society.

51. A. D. Gordon, H. Hüttel, and R. Rydhof Hansen. Type inference for correspondence types. In *6th International Workshop on Security Issues in Concurrency (SecCo 2008)*, Toronto, August 23, 2008.

52. S. Maffeis, M. Abadi, C. Fournet, and A. D. Gordon. Code-carrying authorization. In *13th European Symposium on Research in Computer Security (ESORICS 2008)*, Malaga, October 6-8, 2008. Springer LNCS 5283:563-579.

53. I. Baltopoulos and A. D. Gordon. Secure compilation of a multi-tier web language. In *ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI 2009)*, Savannah, Georgia, January 24, 2009. ACM Press. Pages 27-38.

54. A. Mukhamedov, A. D. Gordon, and Mark Ryan. Towards a verified reference implementation of a trusted platform module. In *Seventeenth International Workshop on Security Protocols*, Cambridge, April 1-3, 2009. Springer LNCS 7028:69–81.

55. J. Borgström, K. Bhargavan, and A. D. Gordon. A compositional theory for STM Haskell. In *ACM SIGPLAN Haskell Symposium*, Edinburgh, September 3, 2009. Pages 69-80. ACM Press.

56. K. Bhargavan, C. Fournet, and A. D. Gordon. Modular verification of security protocol code by typing. In *37th ACM Symposium on Principles of Programming Languages (POPL'10)*, pages 198–209, 2010.

57. F. Dupressoir, A. D. Gordon, and J. Jürjens. Verifying authentication properties of C security protocol code using general verifiers. In *4th International Workshop on Analysis of Security APIs (ASA-4)*, 2010.

58. G. M. Bierman, A. D. Gordon, C. Hriţcu and D. Langworthy. Semantic subtyping with an SMT solver. In *International Conference on Functional Programming (ICFP'10*, pages 105–116, Baltimore, September 2010. ACM Press.

59. J. Borgström, A. D. Gordon, M. Greenberg, J. Margetson, J. Van Gael. Measure Transformer Semantics for Bayesian Machine Learning. *European Symposium on Programming (ESOP'11)*, pages 77–96.

60. I. G. Baltopoulos, J. Borgström, A. D. Gordon. Maintaining Database Integrity with Refinement Types. *European Conference on Object Oriented Programming (ECOOP 2011)*, pages 484–509.

61. F. Dupressoir, A. D. Gordon, J. Jürjens, D. A. Naumann. Guiding a General-Purpose C Verifier to Prove Cryptographic Protocols. *IEEE Symposium on Computer Security Foundations (CSF 2011)*, pages 3–17.

62. M. Aizatulin, A. D. Gordon, J. Jürjens. Extracting and verifying cryptographic models from C protocol code by symbolic execution. *ACM Conference on Computer and Communications Security (CCS 2011)*, pages 331–340.

63. M. Aizatulin, A. D. Gordon, J. Jürjens. Computational verification of C protocol implementations by symbolic execution. *ACM Conference on Computer and Communications Security (CCS 2012)*, pages 712–723.

64. J. A. Hewson, P. Anderson, A. D. Gordon. A Declarative Approach to Automated Configuration. *Proceedings of the 26th Large Installation System Administration Conference (LISA 2012)*, pages 51–66.

65. A. D. Gordon, M. Aizatulin, J. Borgström, G. Claret, T. Graepel, A. V. Nori, S. K. Rajamani, C. V. Russo. A model-learner pattern for Bayesian reasoning. In *40th ACM Symposium on Principles of Programming Languages (POPL'13)*, pages 403–416, 2013.

66. S. Bhat, J. Borgström, Andrew D. Gordon, Claudio V. Russo. Deriving Probability Density Functions from Probabilistic Functional Programs. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2013)*, pages 508–522, 2013.

67. G. Claret, S. K. Rajamani, A. V. Nori, A. D. Gordon, J. Borgström: Bayesian inference using data flow analysis. In *ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE 2013)*, pages 92–102.

68. J. A. Hewson, P. Anderson, A. D. Gordon. Constraint-Based Autonomic Reconfiguration. *7th IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2013)*, pages 101–110, 2013.

69. A. D. Gordon, T. Graepel, N. Rolland, C. V. Russo, J. Borgström, J. Guiver. Tabular: a schema-driven probabilistic programming language. In *41st ACM Symposium on Principles of Programming Languages (POPL'14)*, pages 321–334, 2014.

70. Andrew D. Gordon, Claudio V. Russo, Marcin Szymczak, Johannes Borgström, Nicolas Rolland, Thore Graepel, Daniel Tarlow. Probabilistic Programs as Spreadsheet Queries. In *European Symposium on Programming (ESOP'15)*, pages 1–25, 2015.

71. M. Allamanis, D. Tarlow, A.D. Gordon, Yi Wei. Bimodal Modelling of Source Code and Natural Language. In *International Conference on Machine Learning (ICML 2015)*, pages 2123-2132, 2015.

72. A. Scibior, Z. Ghahramani, A.D. Gordon. Practical probabilistic programming with monads. In *ACM Haskell Symposium (Haskell 2015)*, pages 165–176, 2015.

73. J. Borgström, A. D. Gordon, L. Ouyang, C.V. Russo, A. Scibior, M. Szymczak. Fabular: regression formulas as probabilistic programming. In *43rd ACM Symposium on Principles of Programming Languages (POPL'16)*, pages 271–283, 2016.

74. W. Chen, D. Aspinall, A. D. Gordon, C.A. Sutton, I. Muttik. Explaining unwanted behaviours in context. In *1st International Workshop on Innovations in Mobile Privacy and Security (IMPS 2016)*, 2016.

75. W. Chen, D. Aspinall, A. D. Gordon, C.A. Sutton, I. Muttik. A text-mining approach to explain unwanted behaviours. In *2016 European Workshop on System Security (EUROSEC 2016)*.

76. W. Chen, D. Aspinall, A. D. Gordon, C.A. Sutton, I. Muttik. On robust malware classifiers by verifying unwanted behaviours. In *Integrated Formal Methods - 12th International Conference, IFM 2016*, pages 326–341.

77. W. Chen, D. Aspinall, A. D. Gordon, C.A. Sutton, I. Muttik. More semantics more robust: improving Android malware classifiers. In *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WISEC 2016*, pages 147–158.

78. G. Barthe, G. P. Farina, M. Gaboardi, E. J. Gallego Arias, A. D. Gordon, J. Hsu, P.-Y. Strub. Differentially private Bayesian programming. In *23rd ACM Conference on Computer and Communications Security (CCS'16)*, pages 68–79.

79. J. Borgström, U. Dal Lago, A. D. Gordon, and M. Szymczak. A lambda-calculus foundation for universal probabilistic programming. In *International Conference on Functional Programming (ICFP'16)*, pages 33–46. ACM Press.

80. Advait Sarkar, Andrew D. Gordon, Simon Peyton Jones, and Neil Toronto. Calculation view: multiple-representation editing in spreadsheets. In *2018 IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2018, Lisbon, Portugal, October 1-4, 2018*, pages 85–93. IEEE Computer Society, 2018.

81. Advait Sarkar and Andrew D. Gordon. How do people learn to use spreadsheets? (Work in progress). In *Proceedings of the 29th Annual Workshop of the Psychology of Programming Interest Group, PPIG 2018, London, UK, September 5 - 7, 2018*.

82. Maria I. Gorinova, Andrew D. Gordon, and Charles A. Sutton. Probabilistic programming with densities in SlicStan: efficient, flexible, and deterministic. *Proc. ACM Program. Lang.*, 3(POPL):35:1–35:30, 2019.

83. Alan F. Blackwell, Luke Church, Martin Erwig, James Geddes, Andrew D. Gordon, Maria I. Gorinova, Atilim Gunes Baydin, Bradley Gram-Hansen, Tobias Kohn, Neil Lawrence, Vikash Mansinghka, Brooks Paige, Tomas Petricek, Diana Robinson, Advait Sarkar, Oliver Strickson. In *Proceedings of the 30th Annual Workshop of the Psychology of Programming Interest Group, PPIG 2019, Newcastle University, UK, August 28 - 30, 2019*.

84. Nima Joharizadeh, Advait Sarkar, Andrew D. Gordon, and Jack Williams. Gridlets: Reusing spreadsheet grids. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems, CHI 2020, Honolulu, HI, USA, April 25-30, 2020*, pages 1–7. ACM, 2020.

85. Jack Williams, Nima Joharizadeh, Andrew D. Gordon, and Advait Sarkar. Higher-order spreadsheets with spilled arrays. In *Programming Languages and Systems - 29th European Symposium on Programming, ESOP 2020, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2020, Dublin, Ireland, April 25-30, 2020, Proceedings*, volume 12075 of *Lecture Notes in Computer Science*, pages 743–769. Springer, 2020.

86. Shuang Chen, Alperen Karaoglu, Carina Negreanu, Tingting Ma, Jin-Ge Yao, Jack Williams, Andy Gordon, and Chin-Yew Lin. Linkingpark: An integrated approach for semantic table interpretation. In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020), Virtual conference (originally planned to be in Athens, Greece), November 5, 2020*, volume 2775 of *CEUR Workshop Proceedings*, pages 65–74. CEUR-WS.org, 2020.

87. Jack Williams, Carina Negreanu, Andrew D. Gordon, and Advait Sarkar. Understanding and inferring units in spreadsheets. In *IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2020, Dunedin, New Zealand, August 10-14, 2020*, pages 1–9. IEEE, 2020.

88. Sruti Srinivasa Srinivasa Ragavan, Advait Sarkar, and Andrew D. Gordon. Spreadsheet comprehension: Guesswork, giving up and going back to the author. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021*, pages 181:1–181:21. ACM, 2021.

89. Jack Williams and Andrew D. Gordon. Where-provenance for bidirectional editing in spreadsheets. In *IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2021, St Louis, MO, USA, October 10-13, 2021*, pages 1–10. IEEE, 2021.

90. Carina Negreanu, Alperen Karaoglu, Jack Williams, Shuang Chen, Daniel Fabian, Andrew D. Gordon, and Chin-Yew Lin. Rows from many sources: Enriching row completions from wikidata with a pre-trained language model. In *Companion of The Web Conference 2022, Virtual Event / Lyon, France, April 25 - 29, 2022*, pages 1272–1280. ACM, 2022.

91. Sruti Srinivasa Ragavan, Zhitao Hou, Yun Wang, Andrew D. Gordon, Haidong Zhang, and Dongmei Zhang. Gridbook: Natural language formulas for the spreadsheet grid. In *IUI 2022: 27th International Conference on Intelligent User Interfaces, Helsinki, Finland, March 22 - 25, 2022*, pages 345–368. ACM, 2022.

92. Michael Xieyang Liu, Advait Sarkar, Carina Negreanu, Benjamin G. Zorn, Jack Williams, Neil Toronto, Andrew D. Gordon. "What It Wants Me To Say": Bridging the Abstraction Gap Between End-User Programmers and Code-Generating Large Language Models. In *CHI 2023: Conference on Human Factors in Computing Systems, Hamburg, Germany, April 23-28, 2023*, pages 598:1–598:31. ACM, 2023.

93. Ian Drosos, Nicholas Wilson, Sruti Srinivasa Ragavan, Jack Williams, Andrew D. Gordon. FxD: a functional debugger for dysfunctional spreadsheets. In *IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2023, Washington, DC, USA, October 2-6, 2023*. IEEE, 2023.

94. Kasra Ferdowsi, Jack Williams, Ian Drosos, Andrew D. Gordon, Carina Negreanu, Nadia Polikarpova, Advait Sarkar, Benjamin Zorn. ColDeco: An End User Spreadsheet Inspection Tool for AI-Generated Code In *IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2023, Washington, DC, USA, October 2-6, 2023*. IEEE, 2023.

Invited Conference Publications

1. R. Boulton, A. Gordon, M. Gordon, J. Harrison, J. Herbert, and J. Van Tassel. Experience with embedding hardware description languages in HOL. In V. Stavridou, T. F. Melham, and R. T. Boute, editors, *Theorem Provers in Circuit Design: Theory, Practice and Experience: Proceedings of the IFIP TC10/WG 10.2 International Conference, Nijmegen, June 1992*, IFIP Transactions A-10, pages 129–156. North-Holland, 1992.

2. A. D. Gordon. Nominal calculi for security and mobility. In *DARPA Workshop on Foundations for Secure Mobile Code*, March 1997.

3. S. Dal Zilio and A. D. Gordon. Region analysis and a π-calculus with groups. In *Proceedings MFCS'00*, volume 1893 of *Lecture Notes in Computer Science*, pages 1–20. Springer-Verlag, 2000.

4. K. Bhargavan, C. Fournet, A. D. Gordon, and R. Pucella. TulaFale: A security tool for web services. In *International Symposium on Formal Methods for Components and Objects (FMCO'03)*, volume 3188 of *Lecture Notes in Computer Science*, pages 197–222. Springer-Verlag, 2004.

5. A. D. Gordon. Provable implementations of security protocols. In *Proceedings of the 21st IEEE Symposium on Logic in Computer Science (LICS'06)*, pages 345–346, 2006.

6. A. D. Gordon, R. Harper, J. Harrison, A. Jeffrey, P. Sewell. Robin Milner 1934–2010: verification, languages, and concurrency. POPL 2011:473-474.

7. M. Aizatulin, F. Dupressoir, A. D. Gordon, J. Jürjens. Verifying Cryptographic Code in C: Some Experience and the Csec Challenge. *Formal Aspects in Security and Trust* 2011:1–20.

8. A. D. Gordon, T. A. Henzinger, A. V. Nori, S. K. Rajamani. Probabilistic Programming. *Foundations of Software Engineering* 2014:167–181.

9. Judith Borghouts, Andrew D. Gordon, Advait Sarkar, and Neil Toronto. End-user probabilistic programming. In David Parker and Verena Wolf, editors, *Quantitative Evaluation of Systems, 16th International Conference, QEST 2019, Glasgow, UK, September 10-12, 2019, Proceedings*, volume 11785 of *Lecture Notes in Computer Science*, pages 3–24. Springer, 2019.

Invited Book Chapters

1. A. D. Gordon. Example: the binomial theorem. In M. J. C. Gordon and T. F. Melham, editors, *Introduction to HOL: A theorem proving environment for higher order logic*, chapter 7, pages 77–95. Cambridge University Press, 1993.

2. A. D. Gordon. Notes on nominal calculi for security and mobility. In R. Focardi and R. Gorrieri, editors, *Foundations of Security Analysis and Design*, Lecture Notes in Computer Science, pages 262–330. Springer-Verlag, 2001.

3. K. Bhargavan, C. Fournet, and A. D. Gordon. Policy advisor for WSE 3.0. In *Web Service Security: Scenarios, patterns, and implementation guidance for Web Services Enhancements (WSE) 3.0*, pages 324–330. Microsoft Press, 2006.

4. K. Bhargavan, C. Fournet, A. D. Gordon, and S. Tse. Verified interoperable implementations of security protocols. In M. Broy, J. Grünbauer, C. A. R. Hoare (editors), *Software system reliability and security: Proceedings of the NATO Summer School Marktoberdorf 2006*, pages 87–115. IOS Press, 2007.

5. A. D. Gordon Foreword. In P. Periorellis (editor), *Securing Web Services: Practical Usage of Standards and Specifications*, pages ix–x. Idea Group Inc (IGI), 2007.

6. A. D. Gordon and C. Fournet. Principles and applications of refinement types. In J. Esparza, B. Spanfelner, O. Grumberg (editors), *Logics and Languages for Reliability and Security: Proceedings of the NATO Summer School Marktoberdorf 2009*, pages 73–104. IOS Press, 2010. A preliminary version appears as Technical Report MSR-TR-2009-147, Microsoft Research, October 2009.

7. J. Borgström, A. D. Gordon, and R. Pucella. Roles, Stacks, Histories: A Triple for Hoare. To appear in a Festschrift volume for Tony Hoare, published by Springer. An extended version of this paper appears as Technical Report MSR-TR-2009-97, Microsoft Research, November 2009.

8. C. Fournet, K. Bhargavan, A. D. Gordon Cryptographic Verification by Typing for a Sample Protocol Implementation. FOSAD 2011:66-100.

9. A. D. Gordon, C. Russo, M. Szymczak, J. Borgström, N. Rolland, T. Graepel, D. Tarlow. Tabular: Probabilistic Inference from the Spreadsheet. *Foundations of Probabilistic Programming*, MIT Press, 2020.

Books and Collections

1. A. D. Gordon. *Functional Programming and Input/Output*. Distinguished Dissertations in Computer Science. Cambridge University Press, 1994. Paperback edition, 2008.

2. A. D. Gordon and A. M. Pitts, editors. *Higher Order Operational Techniques in Semantics*, Publications of the Newton Institute. Cambridge University Press, 1998.

3. A. D. Gordon, A. M. Pitts, and C. L. Talcott, editors. *Higher Order Operational Techniques in Semantics*, volume 10 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 1998.

4. A. D. Gordon and A. M. Pitts, editors. *Higher Order Operational Techniques in Semantics*, volume 26 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 1999.

5. A. D. Gordon, editor. *Foundations of Software Science and Computation Structures (FOSSACS'03)*, volume 2620 of *Lecture Notes in Computer Science*. Springer-Verlag, 2003.

6. A. D. Gordon, editor. *Theoretical Computer Science*, 333(1–2):1–327, March 2005. Special issue based on FOSSACS'03.

7. L. Aceto and A. D. Gordon, editors. *Algebraic Process Calculi: The First Twenty Five Years and Beyond*, 2005. Short Contributions from the *Workshop on Algebraic Process Calculi: The First Twenty Five Years and Beyond (PA-05)*, Bertinoro, Forlì, Italy, August 1-5, 2005. BRICS Note NS-05-03. Also appears as Volume 162, pages 1–340 of Electronic Notes in Theoretical Computer Science, Elsevier.

8. A. D. Gordon and D. Sands, editors. *Proceedings of the 2006 ACM Workshop on Formal Methods in Security Engineering (FMSE'06)*. ACM Press, 2006.

9. L. Aceto, M. Bravetti, W. Fokkink, and A. D. Gordon, editors. *Journal of Logic and Algebraic Programming*, 70(2):119–238, 2007. First special issue on *Algebraic Process Calculi: The First Twenty Five Years and Beyond*.

10. L. Aceto, M. Bravetti, W. Fokkink, and A. D. Gordon, editors. *Journal of Logic and Algebraic Programming*, 72(1):1–122, 2007. Second special issue on *Algebraic Process Calculi: The First Twenty Five Years and Beyond*.

11. L. Aceto, M. Bravetti, W. Fokkink, and A. D. Gordon, editors. *Journal of Logic and Algebraic Programming*, 75(1):1–166, 2008. Third special issue on *Algebraic Process Calculi: The First Twenty Five Years and Beyond*.

12. K. Bhargavan, A. D. Gordon, T. Harris, and P. Toft, editors. *Proceedings of the Joint MSR–HP Workshop on "The Rise and Rise of the Declarative Datacentre"*. Technical Report MSR–TR–2008–61, Microsoft Research. May 2008.

13. M. Abadi, P. Gardner, A. D. Gordon, and R. Mardare, editors. *Essays for the Luca Cardelli Fest*. Technical Report MSR–TR–2014–104, Microsoft Research. September 2014.

14. G. Barthe, A. D. Gordon, J.-P. Katoen, A. McIver, editors. *Challenges and Trends in Probabilistic Programming (Dagstuhl Seminar 15181)*. Dagstuhl Reports 5(4): 123-141, 2015.

15. Giuseppe Castagna, Andrew D. Gordon, editors. Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, POPL 2017, Paris, France, January 18-20, 2017. ACM 2017.

Dissertation

1. A. D. Gordon. *Functional Programming and Input/Output*. PhD Dissertation, Cambridge University, 1992.

Patents

1. Verifier to check intermediate language (2003) US Patent 6,560,774.
2. Ambient calculus-based modal logics for mobile ambients (2004) US Patent 6,826,751.
3. Verifying intermediate language code (2005) US Patent 6,851,108.
4. Verifier to check intermediate language (2007) US Patent 7,171,655.
5. Reviewing the security of trusted software components (2008) US Patent 7,437,718.
6. Automatically generating security policies for web services (2009) US Patent 7,559,080.
7. Ambient calculus-based modal logics for mobile ambients (2010) US Patent 7,721,335.
8. Security authorization queries (2011) US Patent 8,060,931.
9. Security assertion revocation (2012) US Patent 8,095,969.
10. Controlling the delegation of rights (2012) US Patent 8,201,215.
11. Semantic subtyping for declarative data scripting language by calling a prover (2013) US Patent 8,413,119.
12. Security language translations with logic resolution (2014) US Patent 8,656,503.
13. Bidirectional type checking for declarative data scripting language (2014) US Patent 8,762,942.
14. Security language expressions for logic resolution (2015) US Patent 8,938,783.
15. Type system for declarative data scripting language (2015) US Patent 8,949,784.
16. Modeling a data generating process using dyadic Bayesian models (2015) US Patent 9,104,961.
17. Security language translations with logic resolution (2016) US Patent 9,282,121.
18. Database access (2016) US Patent 9,418,086.
19. Relational database management (2020) US Patent 10,685,062.

20. Creating and handling lambda functions in spreadsheet applications (2020) US Patent 10,726,201.

21. System and method for generation and execution of elastic sheet-defined functions and arrays (2021) US Patent 10,891,434.

22. Natively handling approximate values in spreadsheet applications (2021) US Patent 10,902,194.

23. Spreadsheet cell calculation view providing multiple-representation editing (2021) US Patent 10,936,804.

24. Inferring cues for use with digital assistant (2021) US Patent 10,997,512.

25. Checking and/or completion for data grids (2021) US Patent 11,093,702.

26. Spreadsheet with reuse functionality (2022) US Patent 11,461,544.

27. Provenance aware editing for spreadsheets (2023) US Patent 11,636,254.