

Curriculum Vitae

Prof. Dr. Byron Cook, FREng

Affiliations: University College London and Amazon

Email: byroncook@gmail.com

URL: <http://www0.cs.ucl.ac.uk/staff/b.cook/>

Short biography

Dr. Byron Cook is a *Professor of Computer Science* at University College London, as well as *Vice President and Distinguished Scientist* at Amazon. Byron's interests include many topics in formal methods, logic, and their applications, *e.g.* biological systems, hardware design, networking, operating systems, programming languages, and security. Byron's work has gained significant attention, *e.g.* a substantial publication record, numerous keynote speaker invitations, and press coverage from Economist, Financial Times, Science, Scientific American, TechCrunch, Vogue, and Wired. Byron's recent work at Amazon has focused on automated logic-based reasoning for cloud security, with applications to cryptography, networks, policies, and virtualization. Prior to joining Amazon, Byron was particularly well known for his work on automatic methods for proving program termination and the TERMINATOR termination prover. This work represented a breakthrough, challenging the prevailing opinion in computer science at the time that automatic termination proving was impossible. Byron was also known for his contributions to SLAM and the Microsoft product Static Driver Verifier, which is sometimes credited for the revival of automatic program verification research.

Publications

Refereed conference articles

1. *Generating and Exploiting Automated Reasoning Proof Certificates*
Haniel Barbosa *et. al*
Communications of the ACM (CACM) 2023
2. *Partitioning strategies for distributed SMT solving*
Amalee Wilson, *et. al*
FMCAD 2023
3. *Model checking boot code from AWS data centers*
Byron Cook, *et. al*
FMSD 57(1):34-52 (2021)
4. *Code-level model checking in the software development workflow at Amazon Web Services*
Nathan Chong, *et. al*
SPE 51(4):772-797 (2021)_{i/li}
5. *Using model checking tools to triage the severity of security bugs in the Xen hypervisor*
Byron Cook, *et. al*
FMCAD 2020

6. *Stratified Abstraction of Access Control Policies*
John Backes, *et al*
CAV [International Conference on Computer-Aided Verification], Los Angeles, 2020
7. *Block Public Access: Trust Safety Verification of Access Control Policies*
Malik Bouchet, *et al*
ESEC/FSE20 [28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 20)], Sacramento, 2020
8. *Reachability Analysis for AWS-based Networks*
John Backes, *et al*
CAV [International Conference on Computer-Aided Verification], New York City, 2019
9. *Formal reasoning about the security of Amazon Web Services*
Byron Cook
CAV [International Conference on Computer-Aided Verification], Oxford, 2018
10. *Semantic-based Automated Reasoning for AWS Access Policies using SMT*
John Backes, *et al*
FMCAD [International Conference on Formal Methods in Computer Aided Design], Austin, 2018
11. *Model checking boot code from AWS data centers*
B. Cook, K. Khazem, D. Kroening, S. Tasiran, M. Tautschnig and M. Tuttle.
CAV [International Conference on Computer-Aided Verification], Oxford, 2018
12. *Continuous formal verification of Amazon s2n*
Andrey Chudnov, Nathan Collins, Byron Cook, Josiah Dodds, Brian Huffman, Stephen Magill, Colm MacCarthaigh, Eric Mertens, Eric Mullen, Serdar Tasiran, Aaron Tomb, and Edwin Westbrook
CAV [International Conference on Computer-Aided Verification], Oxford, 2018
13. *SideTrail: Verifying Time-Balancing of Cryptosystems*
K. Athanasiou, B. Cook, M. Emmi, C. MacCarthaigh, D. Schwartz-Narbonne and S. Tasiran
VSTTE [Working Conference on Verified Software: Theories, Tools, and Experiments], Oxford, 2018
14. *T2: temporal property verification*
Marc Brockschmidt, Byron Cook, Samin Ishtiaq, Heidy Khlaaf, and Nir Piterman
TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], Eindhoven, 2016
15. *On automation of CTL* verification for infinite-state systems*
Byron Cook, Heidy Khlaaf and Nir Piterman
CAV [International Conference on Computer-Aided Verification], San Francisco, 2015
16. *Spatial interpolants*
Aws Albarghouthi, Josh Berdine, Byron Cook and Zachary Kincaid
ESOP [European Symposium on Programming], London, 2015
17. *Fairness for infinite-state systems*
Byron Cook, Heidy Khlaaf and Nir Piterman
TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], London, 2015
18. *Proving nontermination via safety*
Hong Yi Chen, Byron Cook, Carsten Fuhs, Kaustubh Nimkar, Peter W. O’Hearn
TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], Grenoble, 2014

19. *Faster temporal reasoning for infinite-state programs*
Byron Cook, Heidy Khlaaf, Nir Piterman
FMCAD [International Conference on Formal Methods in Computer Aided Design], Lausanne, 2014
20. *Disproving termination with overapproximation*
Byron Cook, Carsten Fuhs, Kaustubh Nimkar, Peter O'Hearn
FMCAD [International Conference on Formal Methods in Computer Aided Design], Lausanne, 2014
21. *Finding instability in biological models*
Byron Cook, Jasmin Fisher, Benjamin A. Hall, Samin Ishtiaq, Garvit Juniwal, Nir Piterman
CAV [International Conference on Computer-Aided Verification], Vienna, 2014
22. *Better termination proving through cooperation*
Marc Brockschmidt, Byron Cook and Carsten Fuhs
CAV [International Conference on Computer-Aided Verification], St. Petersburg, 2013
23. *Reasoning about nondeterminism in programs*
Byron Cook and Eric Koskinen
PLDI [International Conference on Programming Language Design and Implementation], Seattle, 2013
24. *Ramsey vs. lexicographic termination proving*
Byron Cook, Abigail See, and Florian Zuleger
TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], Rome, 2013
25. *At the interface of biology and computation*
Alex S. Taylor, Nir Piterman, Samin Ishtiaq, Jasmin Fisher, Byron Cook, Caitlin Cockerton, Sam Bourton, and David Benque
CHI [ACM SIGCHI Conference on Human Factors in Computing Systems], Paris, 2013
26. *BMA: visual tool for modeling and analysis of biological networks* (tool paper)
David Benque, Sam Bourton, Caitlan Cockerton, Byron Cook, Jasmin Fisher, Samin Ishtiaq, Nir Piterman, Alex Taylor, Moshe Vardi
CAV [International Conference on Computer-Aided Verification], Berkeley, 2012
27. *Temporal property verification as a program analysis task*
Byron Cook, Eric Koskinen, Moshe Vardi
CAV [International Conference on Computer-Aided Verification], Snowbird, 2011
28. *SLayer: Memory safety for systems-level code*
Josh Berdine, Byron Cook, Samin Ishtiaq
CAV [International Conference on Computer-Aided Verification], Snowbird, 2011
29. *Making prophecies with decision predicates*
Byron Cook and Eric Koskinen
POPL [International Symposium on Principles of Programming Languages], Austin, 2011
30. *Proving stabilization of biological systems*
Byron Cook, Jasmin Fisher, Elzbieta Krepska, Nir Piterman
VMCAI [International Conference on Verification, Model Checking, and Abstract Interpretation], Savannah, 2011
31. *Tractable Reasoning in a Fragment of Separation Logic*
Byron Cook, Christoph Haase, Joel Ouaknine, Matthew Parkinson and James Worrell
CONCUR [International Conference on Concurrency Theory], Aachen, 2011

32. *Ranking function synthesis for bit-vector relations*
Byron Cook, Daniel Kroening, Philipp Ruemmer, and Christoph Wintersteiger
TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], Paphos, 2010
33. *Finding heap-bounds for hardware synthesis*
Byron Cook, Ashutosh Gupta, Stephen Magill, Andrey Rybalchenko, Jiri Simsa, Satnam Singh, and Viktor Vafeiadis
FMCAD [International Conference on Formal Methods in Computer Aided Design], Austin, 2010
34. *Proving that non-blocking algorithms don't block*
Alexey Gotsman, Byron Cook, Matthew Parkinson, and Viktor Vafeiadis
POPL [International Symposium on Principles of Programming Languages], Savannah, 2009
35. *Proving conditional termination*
Byron Cook, Sumit Gulwani, Tal Lev-Ami, Andrey Rybalchenko, and Mooly Sagiv
CAV [International Conference on Computer-Aided Verification], Princeton, 2008
36. *Scalable shape analysis for systems code*
Hongseok Yang, Oukseh Lee, Josh Berdine, Cristiano Calcagno, Byron Cook, Dino Distefano, and Peter O'Hearn
CAV [International Conference on Computer-Aided Verification], Princeton, 2008
37. *Ranking abstractions*
Aziem Chawdhary, Byron Cook, Sumit Gulwani, Mooly Sagiv, and Hongseok Yang
ESOP [European Symposium on Programming], Budapest, 2008
38. *Proving thread termination*
Byron Cook, Andreas Podelski, and Andrey Rybalchenko,
PLDI [International Conference on Programming Language Design and Implementation], San Diego, 2007
39. *Thread-modular shape analysis*
Alexey Gotsman, Josh Berdine, Byron Cook, and Mooly Sagiv,
PLDI [International Conference on Programming Language Design and Implementation], San Diego, 2007
40. *Local reasoning for storable locks and threads*
Alexey Gotsman, Josh Berdine, Byron Cook, Noam Rinetzkzy, and Mooly Sagiv
APLAS [Asian Symposium on Programming Languages and Systems], Singapore, 2007
41. *Proving that programs eventually do something good*
Byron Cook, Alexey Gotsman, Andreas Podelski, Andrey Rybalchenko, and Moshe Vardi
POPL [International Symposium on Principles of Programming Languages], Nice 2007
42. *Variance analyses from invariance analyses*
Josh Berdine, Aziem Chawdhary, Byron Cook, Dino Distefano, and Peter O'Hearn
POPL [International Symposium on Principles of Programming Languages], Nice 2007
43. *Shape analysis for composite data structures*
Josh Berdine, Cristiano Calcagno, Byron Cook, Dino Distefano, Peter O'Hearn, Thomas Wies, and Hongseok Yang
CAV [International Conference on Computer-Aided Verification], Berlin, 2007
44. *Arithmetic strengthening for shape analysis*
Stephen Magill, Josh Berdine, Edmund Clarke, and Byron Cook,
SAS [International Static Analysis Symposium], Denmark, 2007

45. *Proving termination by divergence*
Domagoj Babic, Byron Cook, Alan Hu, and Zvonimir Rakamaric
SEFM [International Conference on Software Engineering and Formal Methods], London, 2007
46. *Shape analysis by graph decomposition*
Roman Manevich, Josh Berdine, Byron Cook, Ganesan Ramalingam, and Mooly Sagiv
TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], Braga, 2007
47. *Over-approximating Boolean programs with unbounded thread creation*
Byron Cook, Daniel Kroening, Natasha Sharygina
FMCAD [International Conference on Formal Methods in Computer Aided Design], San Jose, 2007
48. *Interprocedural shape analysis with separated heap abstractions*
Alexey Gotsman, Josh Berdine, and Byron Cook
SAS [International Symposium on Static Analysis], Seoul, 2007
49. *Automatic termination proofs for programs with shape-shifting heaps*
Josh Berdine, Byron Cook, Dino Distefano, and Peter O'Hearn
CAV [International Conference on Computer-Aided Verification], Seattle, 2006
50. *Terminator: Beyond safety*
Byron Cook, Andreas Podelski, and Andrey Rybalchenko
CAV [International Conference on Computer-Aided Verification], Seattle, 2006
51. *Repair of Boolean programs with an application to C*
Andreas Griesmayer, Roderick Bloem, and Byron Cook
CAV [International Conference on Computer-Aided Verification], Seattle, 2006
52. *Termination proofs for systems code*
Byron Cook, Andreas Podelski, and Andrey Rybalchenko
PLDI [International Conference on Programming Language Design and Implementation], Ottawa, 2006
53. *Thorough static analysis of device drivers*
Thomas Ball, Ella Bounimova, Byron Cook, Vladimir Levin, Jakob Lichtenberg, Con McGarvey, Bohus Ondrusek, Sriram K. Rajamani, Abdullah Ustuner
EuroSys [European Systems Conference], Leuven, 2006
54. *Abstraction-refinement for termination*
Byron Cook, Andreas Podelski, Andrey Rybalchenko
SAS [International Symposium on Static Analysis], London, 2005
55. *Using Stålmarck's algorithm to prove inequalities*
Byron Cook and George Gonthier
ICFEM [International Conference on Formal Engineering Methods], Manchester, 2005
56. *Predicate abstraction via symbolic decision procedures*
Shuvendu Lahiri, Thomas Ball, and Byron Cook
CAV [International Conference on Computer-Aided Verification], Edinburgh, 2005
57. *Cogent: Accurate theorem proving for program verification*
Byron Cook, Daniel Kroening, and Natasha Sharygina
CAV [International Conference on Computer-Aided Verification], Edinburgh, 2005
58. *Zapato: Automatic theorem proving for predicate abstraction refinement*
Thomas Ball, Byron Cook, Shuvendu K. Lahiri, and Lintao Zhang
CAV [International Conference on Computer-Aided Verification], Boston, 2004

59. *Refining approximations in software predicate abstraction*
Thomas Ball, Byron Cook, Satyaki Das, and Sriram K. Rajamani
TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], Barcelona, 2004
60. *SLAM and Static Driver Verifier: technology transfer of formal methods inside Microsoft*,
Thomas Ball, Byron Cook, Vladimir Levin and Sriram K. Rajamani.
IFM [International Conference on Integrated Formal Methods], Kent, 2004
61. *Accurate theorem proving for program verification*,
Byron Cook, Daniel Kroening, Natasha Sharygina,
ISoLA [Leveraging Applications of Formal Methods], Paphos, 2004
62. *A symbolic approach to predicate abstraction*
Shuvendu K. Lahiri, Randall E. Bryant, and Byron Cook
CAV [International Conference on Computer-Aided Verification], Boulder, 2003
63. *A proof engine approach to solving combinational design automation problems*
Gunnar Andersson, Per Bjesse, Byron Cook, and Ziyad Hanna
DAC [Design Automation Conference], Las Vegas, 2002
64. *A framework for microprocessor correctness statements*
Mark Aagaard, Byron Cook, Nancy Day, and Robert Jones
CHARME [Advanced Research Working Conference on Correct Hardware Design and Verification Methods], Edinburgh, 2001
65. *Combining stream-based and state-based verification techniques for microarchitectures*
Mark Aagaard, Byron Cook, and Nancy Day
FMCAD [International Conference on Formal Methods in Computer Aided Design], Austin, 2000
66. *Formal verification of explicitly parallel microprocessors*
Byron Cook, John Launchbury, John Matthews, and Dick Kieburtz
CHARME [Advanced Research Working Conference on Correct Hardware Design and Verification Methods], Bad Herrenalb, 1999
67. *On embedding a microarchitectural design language within Haskell*
John Launchbury, Jeff Lewis and Byron Cook
ICFP [International Conference on Functional Programming], Paris, 1999
68. *Microprocessor specification in Hawk*
John Matthews, John Launchbury, and Byron Cook
ICCL [International Conference on Computer Languages], Chicago, 1998

Refereed journal articles

69. *Model checking boot code from AWS data centers*
B. Cook *et al.*, Journal of Formal Methods in Systems Design (2020)
70. *One-click formal methods*
P. Bolignano *et al.*, IEEE Software Magazine, November/December 2019
71. *Verifying Increasingly Expressive Temporal Logics for Infinite-State Systems*
B. Cook, H. Khlaaf, and N. Piterman.
Journal of the the ACM, 64, 2, Article 15 (May 2017), 39 pages.
72. *Drug target optimization in chronic myeloid leukemia using innovative computational platform*
R. Chuang, B. Hall, D. Benque, B. Cook, S. Ishtiaq, N. Piterman, A. Taylor, M. Vardi, S. Koschmieder, B. Gottgens, and J. Fisher
Scientific Reports, 5:8190, Nature Publishing Group, February 2015

73. *Relations*
Tauba Auerbach, Byron Cook, David Reinfurt
Bulletins of the Serving Library, 2014
74. *Mathematical artifacts*
Byron Cook
Parkett 94, 2014
75. *Ranking function synthesis for bit-vector relations*
Byron Cook, Daniel Kroening, Philipp Rümmer, Christoph Winterstieger
International Journal on Formal Methods in System Design, March, 2013
76. *Proving termination of nonlinear command sequences*
Domagoj Babic, Byron Cook, Alan J. Hu, Zvonimir Rakamaric
International Journal on Formal Aspects of Computing (special issue from SEFM), 2013
77. *Temporal property verification as a program analysis task (extended version)*
Byron Cook, Eric Koskinen, Moshe Vardi
International Journal on Formal Methods in System Design (special issue from CAV), 2012
78. *Proving program termination*
Byron Cook, Andreas Podelski, Andrey Rybalchenko
Communications of the ACM, Volume 54 Issue 5, May 2011
79. *Summarization for termination*
Byron Cook, Andreas Podelski, and Andrey Rybalchenko
International Journal on Formal Methods in System Design, Vol 35, pp. 369–387
80. *Software engineering and formal methods*
Mike Hinchey, Michael Jackson, Patrick Cousot, Byron Cook, Jonathon P. Bowen
Communications of the ACM, Vol. 51, 2008, pp. 54-59
81. *Verification of Boolean programs with unbounded thread creation*
Byron Cook, Daniel Kroening, and Natasha Sharygina
Journal of Theoretical Computer Science, Vol. 388, 2007, pp. 227-242
82. *Predicate abstraction via symbolic decision procedures*
Shuvendu Lahiri, Tom Ball, and Byron Cook
Journal of Logic Methods in Computer Science, Vol. 3(1:2), 2007, pp. 1-20
83. *Design automation with mixtures of proof strategies for propositional logic*
Gunnar Andersson, Per Bjesse, Byron Cook, and Ziyad Hanna
IEEE Transactions on CAD, Vol. 22(8), 2003, pp. 1042-1048
84. *A framework for microprocessor correctness statements*
Mark Aagaard, Byron Cook, Nancy Day, and Robert Jones
International Journal on Software Tools for Technology Transfer, Vol 4(3), 2002, pp. 298-312

Books

85. *Program termination*
Byron Cook
Cambridge University Press (forthcoming)
86. *Computer Aided Verification*
Byron Cook, Tayssir Touili, Paul Jackson (Eds.)
Springer, 2010

87. *Formal Methods for Industrial Critical Systems*
Maria Alpuente, Byron Cook, Christophe Joubert (Eds.)
Springer, 2009
88. *Proceedings of the 8th International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI)*
Byron Cook and Andreas Podelski (Eds.)
Springer, 2007

Journal special issues

89. *Special Issue on Satisfiability Modulo Theories*
Byron Cook and Roberto Sebastiani (Eds.)
Journal on Satisfiability, Boolean Modeling and Computation

Workshop articles

90. *Learning to decipher the heap for program verification*
Mark Brockschmidt, Yuxin CHen, Byron Cook, Pushmeet Kohli, and Daniel Tarlow
Workshop on Constructive Machine Learning at ICML, 2015
91. *Symbolic model checking for asynchronous Boolean programs*
Byron Cook, Daniel Kroening, and Natasha Sharygina
SPIN [SPIN Workshop on Model Checking of Software], San Francisco, 2005
92. *Specifying superscalar microprocessors in Hawk*
Byron Cook, John Launchbury, and John Matthews
FTH [International Workshop on Formal Techniques for Hardware], Marstrand, 1998
93. *Disposable memo functions*
Byron Cook and John Launchbury
Haskell Workshop, Amsterdam, 1997

Invited articles

94. *Formal reasoning about the security of Amazon Web Services*
Byron Cook
CAV [International Conference on Computer-Aided Verification], Oxford, 2018
95. *Principles of program termination*
Byron Cook
Lecture notes from 2008 Marktoberdorf summer school (Marktoberdorf)
96. *Advances in Program Termination and Liveness*
Byron Cook
VMCAI [International Conference on Verification, Model Checking, and Abstract Interpretation], Savannah, 2009
97. *Computing bounds on space and time for hardware compilation*
Byron Cook
FMCAD [Formal Methods in Computer Aided Design], Portland, 2008
98. *Automatically proving program termination*
Byron Cook
CAV [International Conference on Computer-Aided Verification], Berlin, 2007

99. *Bringing hardware and software closer together with termination analysis*
Byron Cook
MEMOCODE [International Conference on Formal Methods and Models for Codesign], Nice, 2007
100. *Automatically Proving Concurrent Programs Correct*
Byron Cook
SEFM [IEEE International Conference on Software Engineering and Formal Methods], London, 2007
101. *Finding bugs in device drivers with Static Driver Verifier*
Byron Cook
ASM [International Conference on Abstract State Machines], Paris, 2005
102. *Finding API usage rule violations in Windows device drivers using Static Driver Verifier*
Byron Cook
ISoLA [Leveraging Applications of Formal Methods], Paphos, 2004

Grants

- “IRIS: Interface Reasoning for Interacting Systems” (EPSRC program grant), 6.1m GBP
- “Compositional Security Analysis for Binaries” (EPSRC), 290k GBP
- “Boosting Automated Verification Using Cyclic Proof” (EPSRC), 550k GBP
- “Program Verification Techniques for Understanding Security Properties of Software” (EPSRC), 877k GBP
- “Resource Reasoning” (EPSRC), 3.1m GBP

Patents

- US Patent No. 11,095,523, “VIRTUAL NETWORK VERIFICATION SERVICE.”
- US Patent No. 11,108,805, “AUTOMATED PACKETLESS NETWORK REACHABILITY ANALYSIS.”
- US Patent No. 10,630,695, “SECURITY POLICY MONITORING SERVICE.”
- US Patent No. 10,469,324, “VIRTUAL NETWORK VERIFICATION SERVICE.”
- US Patent No. 10,664,379, “AUTOMATED SOFTWARE VERIFICATION SERVICE”
- US Patent No. 10,652,266, “AUTOMATED THREAT MODELING USING MACHINE-READABLE THREAT MODELS.”
- US Patent No. 10,757,128, “SECURITY POLICY ANALYZER SERVICE AND SATISFIABILITY ENGINE”
- US Patent No. 10,769,250, “TARGETED SECURITY MONITORING USING SEMANTIC BEHAVIORAL CHANGE ANALYSIS”

Awards

- Elected *Fellow of the Royal Academy of Engineering*, 2019. Text from the award: “Byron Cook is a world-renowned leader in the field of formal verification. For over 20 years Byron has worked to bring this field from academic hypothesis to mechanised industrial reality. Byron has made major research contributions, built influential tools, led teams that operationalised formal verification activities, and helped establish connections between others that have dramatically accelerated growth of the area. Byrons tools have been applied to a wide array of topics, e.g. biological systems, computer operating systems, programming languages, and security. Byrons Automated Reasoning Group at Amazon is leading the field to even greater success”
- Honorable mention, NSA Annual Best Scientific Cybersecurity Paper Competition, 2019. This is a competition run by NSA that examines all computer security papers published in the last year and selects those papers they view as most impactful. Each year there is 1 winner, between 0-2 honorable mentions.
- Roger Needham Award, 2009. The Roger Needham Award is made annually by the British Computer Society for a distinguished research contribution in computer science by a UK based researcher within ten years of their PhD. The award includes a 5,000 GBP cash prize, and a public lecture at the UK’s Royal Society in London.

Press coverage

- *Next Generation Security with Automated Reasoning, an Artificial Intelligence Technology* AWS Podcast no.266, Oct 2018
<https://aws.amazon.com/podcasts/aws-podcast/#266>
- *What are Amazon Zelkova and Tiros? AWS looks to reduce S3 configuration errors*, IDG, August, 2018
<https://www.csoonline.com/article/3298166/cloud-security/what-are-amazon-zelkova-and-tiros-aws->
- *Amazon tests out two tools to help keep its cloud secure*, Wired, Lily Hay Newman, July, 2018
<https://www.wired.com/story/aws-cloud-security-tools-leaks/>
- *Amazon is quietly doubling down on cryptographic security*, TechCrunch, August, 2018
<https://techcrunch.com/2018/08/30/amazon-aws-cryptography-security/>
- *How Do You Explain The Unreasonable Effectiveness Of Cloud Security?* highscalability.com, September 2018
<http://highscalability.com/blog/2018/9/19/how-do-you-explain-the-unreasonable-effectiveness-of->
- *Amazon Adds Crypto-Based Security Tools*, pymnts.com, September 2018
<https://www.pymnts.com/amazon/2018/ai-security-software-tools-aws-arg/>
- *Why the blue screen of death no longer plagues Windows users* , Nick Heath, ZDNet, September, 2013
- *Modelling: Computing Cancer*, Neil Savage, Nature, November 2012
- *Dr. Byron Cook: Geek of the week*, Richard Morris, Simple-Talk, September, 2010
- *Sign Of The Times*, Cameron Bird, Wired, December 2009
- *A Good Sign*, Angela Saini, Science, July 2009
- *Optic Nerve*, Dodie Kazanjain, Vogue, January 2009

- *All Shook Down*, Hiya Swanhuysen, San Francisco Weekly, December 2008
- *Byron Cook: Terminator - Proving good things will eventually happen* (Video interview), Charles Torre, MSDN, July 2007
- *Byron Cook: Inside Terminator* (Video interview), Charles Torre, MSDN, September 2007
- *Send in the Terminator*, Gary Stix, Scientific American, December 2006
- *Testers aim to kill off dreaded blue screens*, Mary Branscombe, Financial Times, November 22nd, 2006
- *Microsoft creates an application terminator*, Andy Patrizio, Internet News, August, 2006
- *Microsoft researcher aims to make software more predictable*, Tony Baer, Computer Wire, August, 2006,
- *Microsoft bug-checking tools promise fewer crashes*, Joris Evens, CNET, May 26, 2006
- *Microsoft's secret bug squasher*, Simson Garfinkel, Wired, November, 2005
- *Researching a path to fewer bugs*, Patrick Meader, Visual Studio Magazine, February 2003
- *Building a better bug-trap* , Economist, June 19th, 2003

Graduate students

- Alexey Gotsman. Completed PhD: 2009. Now at IMDEA (Madrid Institute for Advanced Studies)
- Eric Koskinen. Completed PhD: 2012. Now at Stevens Institute of Technology
- Kaustubh Nimkar. Completed PhD: 2015. now at Bloomberg
- Heidy Khlaaf. Completed PhD: 2018. Now at Adelard LLP
- Paul Subotic. Completed PhD: 2019. Now at Amazon

Teaching

- *Program termination*, 4 graduate-level lecture hours at University of California, Berkeley, 2009
- *Program termination*, 6 graduate-level lecture hours at the International Summer School on Trends in Concurrency (Prague), 2008
- *Program termination*, 5 graduate-level lecture hours at the Marktoberdorf Summer School, 2008
- *Program termination*, 6 graduate-level lecture hours at the International Summer School on Trends in Concurrency (Prague), 2008
- *Program termination*, 24 graduate-level lecture hours at Carnegie Mellon University, 2008
- *Program termination*, 6 graduate-level lecture hours at Imperial College, 2008
- *Program termination*, 6 graduate-level lecture hours at Cambridge University, 2007

- *Introduction to C++*, 20 undergraduate-level lecture hours at The Evergreen State College, 1998
- *Constructing Applets in Java*, 10 undergraduate-level lecture hours at The Evergreen State College, 1998
- *C++ and Java, Object Oriented Programming*, 20 undergraduate-level lecture hours at The Evergreen State College, 1997
- *C programming language: Introduction*, 20 undergraduate-level lecture hours at The Evergreen State College, 1997
- *Web programming in Perl*, 10 undergraduate-level lecture hours at The Evergreen State College, 1997
- *Introduction to Computer Science II*, 20 undergraduate-level lecture hours at Portland Community College, 1997
- *Introduction to Computer Programming*, 20 undergraduate-level lecture hours at Oregon Institute of Technology, 1997
- *Introduction to Data Structures*, 20 undergraduate-level lecture hours at Oregon Institute of Technology, 1997
- *Introduction to Programming for the World Wide Web*, 20 undergraduate-level lecture hours at The Evergreen State College, 1996
- *Introduction to UNIX Operating System*, 20 undergraduate-level lecture hours at The Evergreen State College, 1996
- *Introduction to Computer Science I*, 20 undergraduate-level lecture hours at Portland Community College, 1996

Tools developed

- ZELKOVA* : Constraint-based IAM policy reasoning tool. Internal Amazon tool used by Amazon/AWS products S3, Macie, Trusted Advisor, Config, and various internal tools. Discussed in <https://www.youtube.com/watch?v=Wvyc-VEU0ns&t=2214> and <https://www.wired.com/story/aws-cloud-security-tools-leaks/> .
- TIROS* : Constraint-based EC2 network reasoning tool. Internal Amazon tool used by forthcoming Amazon/AWS product and various internal tools. Discussed in <https://www.wired.com/story/aws-cloud-security-tools-leaks/> .
- BMA* : Biological model analysis tool
<http://biomodelanalyzer.org/> Used in drugs discovery research by companies such as AstraZeneca (as discussed in <https://www.youtube.com/watch?v=00Evk79JW78>)
- TERMINATOR and T2 : Program termination prover
<http://github.com/mmjb/T2>
- SLAYER* : Shape analysis engine
<http://research.microsoft.com/SLayer>
- SLAM : Symbolic software model checker
<http://research.microsoft.com/SLAM>

- Static Driver Verifier : Device driver correctness tool (Microsoft Windows product released through the Windows Device Driver Development Kit)
<https://docs.microsoft.com/en-us/windows-hardware/drivers/devtest/static-driver-verifier>
- ZAPATO : Microsoft internal decision procedure framework used in SLAM. Note that ZAPATO led to ZAP, which led to Z3
- PROVER CL : Propositional SAT solver
<http://www.prover.com/products/ppi/cl.xml>
- PROVER SL : Symbolic model checker for finite-state systems
<http://www.prover.com/products/ppi/sl.xml>
- HLSPEC : High-level microprocessor design language and tools (Intel internal)
- HAWK : High-level microprocessor design language and tools
<http://www.cse.ogi.edu/PacSoft/projects/Hawk/>

For tool names marked with *, Byron was involved in the founding, strategy, and/or design, but did not write/maintain the source code.

Invited, plenary, keynote and tutorial talks

- Marktoberdorf Summer School, 2023
- Kings college, Cambridge annual Turing lecture, 2023. See <https://www.youtube.com/watch?v=thPPk9GmyxI>
- Workshop on Dependable and Secure Software Systems, 2021. See https://www.youtube.com/watch?v=g-DH_b5bF
- Newton Institute / Vistas in Verified Software, 2022. See <https://www.youtube.com/watch?v=PaPoujuU0kM>
- Ask me anything interview. PLDI [International Conference on Programming Language Design and Implementation], 2020 See <https://www.youtube.com/watch?v=jGgQmnPH0dQ&t=1h19m56s>
- NASA Formal Methods Symposium, Moffett Field, 2020
- IBM Programming Languages Day, 2019
- Graz Security Week, 2019
- MODELS 2019, Munich
- FLoC [Federated Logic Conference], Oxford, 2018. FLoC occurs every 4 years as the amalgamation of CAV, CSF, FM, FSCD, ICLP, IJCAR, ITP, LICS, SAT
 - Plenary speaker. See <https://www.youtube.com/watch?v=JfjLKB027nw>
 - Invited speaker, “Formal methods in Industry” special session
 - Verification and Deduction Mentoring Workshop
- SPLASH/OOPSLA, 2018, <https://www.youtube.com/watch?v=91PR0d2uijo>
- Philosophical Society of Washington, Lecture 2382, Washington DC, 2017
<https://www.youtube.com/watch?v=eJ88cIDUNXY>
- FMCAD[Formal Methods in Computer Aided Design], Vienna, 2017
- SPIN Symposium, Santa Barbara, 2017

- PiP [POPL Workshop on Principles in Practice], Paris, 2017
- N40AI [POPL Workshop on Next 40 years of abstract interpretation], Paris, 2017
- AWS Re:Invent, Las Vegas, 2016
<https://www.youtube.com/watch?v=U40bWY6oVtU>
- CAV Mentoring Workshop, San Francisco, 2015
- WST [International Workshop on Termination], Bertinoro (Italy), 2013
- Workshop on Software Correctness and Reliability, Zurich, 2013
https://www.youtube.com/watch?v=_LqXVnq_rWM
- POPL (tutorial) [Symposium on Principles of Programming Languages], Philadelphia, 2012
- Dutch Model Checking Day, Amsterdam, 2012
- SIGPLAN Programming Languages Mentoring Workshop, Philadelphia, 2012
- Manycore workshop, Birmingham, 2012
- CAV workshop on Applications of Formal Methods in Systems Biology, Berkeley, 2012
- Bright Club, London, April 2012
- HCSS [International Conference on High Confidence Software and Systems], Annapolis, 2012
- CADE [International Conference on Automated Deduction], Wroclaw, 2011
- ECOOP Summer School [European Conference on Object-Oriented Programming], Lancaster, 2011
- Dutch Model Checking Day, Delft, 2011
- IFIP Working Group 2.3, Santa Barbara, 2011
- Workshop on Theory Engineering, Cambridge, 2010
- IFIP Working Group 2.3, Zurich, 2010
- Royal Society, London, 2009
- Midlands Graduate School Christmas Seminar, 2009
- HCSS [International Conference on High Confidence Software and Systems], Baltimore, 2009
- NFM [NASA Formal Methods Symposium], Moffett Field, 2009
- VMCAI [International Conference on Verification, Model Checking, and Abstract Interpretation], Savannah, 2009
- IFM [Integrated Formal Methods], Dusseldorf, 2009
- Infinity [International Workshop on Verification of Infinite-State Systems], Bologna, 2009
- Workshop on Applied Logic: Inductive and Deductive Reasoning, 2009
- IFIP Working Group 2.3, Cambridge, 2008
- Marktoberdorf Summer School, 2008
- FMCAD [Formal Methods in Computer Aided Design], Portland, 2008
- HCSS [International Conference on High Confidence Software and Systems], Baltimore, 2008

- International Summer School on Trends in Concurrency (Prague), 2008
- CAV Workshop on Numerical Abstractions for Software Verification, 2008
- Science of Security Workshop, Berkeley, 2008
- QCon Enterprise Software Development Conference, San Francisco, 2007
- CAV [International Conference on Computer-Aided Verification] Berlin, 2007
- HCSS [International Conference on High Confidence Software and Systems], Baltimore, 2007
- MEMOCODE [International Conference on Formal Methods and Models for Codesign] (Nice), 2007
- SEFM [IEEE International Conference on Software Engineering and Formal Methods] London, 2007
- Seminar on the Challenge of Software Verification, Dagstuhl, 2006
- WST [International Workshop on Termination], Seattle, 2006
- SVV [International Workshop on Software Verification and Validation], Seattle, 2006
- AVoCS [International Workshop on Automated Verification of Critical Systems], Nancy, 2006
- ARW [Automated Reasoning Workshop], Bristol, 2006
- ICSSR [International Computer Science Symposium in Russia], St. Petersburg, 2006
- ESCAR [CADE Workshop on Empirically Successful Classical Automated Reasoning], 2005
- DISPROVING [Workshop on Disproving - Non-Theorems, Non-Validity, Non-Provability] Tallinn, 2005
- ASM [International Workshop on Abstract State Machines], Paris, 2005
- Combination of Decision Procedures Summer School, Stanford, 2004
- HCSS [Conference on High Confidence Software and Systems], Baltimore, 2004
- ISoLA [Leveraging Applications of Formal Methods] (Paphos)
- Colloquium L'ingnierie du logiciel, Paris, 2004
- DAC [Design Automation Conference], Las Vegas 2001
- University colloquium lectures at Stanford, Berkeley, Carnegie Mellon, Harvard, MIT, ETH, INRIA, University of Toronto, University of Birmingham, University of Manchester, University of British Columbia, University of Utah, University of Colorado, Oxford, Cambridge, Chalmers, and New York University. See <https://vimeo.com/81641895> for an example.
- Microsoft Techfest¹ in 2011, 2009, 2007, 2006, and 2005
<https://www.youtube.com/watch?v=V91oBk-nWCg>
https://archive.org/details/Microsoft_Research_Video_104033

¹Techfest is a Microsoft event in which researchers give lectures and make demos available to Microsoft employees and the press. The event is high-profile (>30,000 attendees) and the lectures are selected using a competitive process.

Panel discussions

- NASA Formal Methods Symposium, Moffett Field, 2009
- Science of Security Workshop, Berkeley, 2008
- SMT [International Workshop on Satisfiability Modulo Theories], Princeton, 2008
- HCSS [International Conference on High Confidence Software and Systems], Baltimore, 2008
- University of Illinois Affiliates Conference, 2006
- MEMOCODE [International Conference on Formal Methods and Models for Codesign] Verona, 2005

Professional activities

- Advisory board
 - DeepSpec/NSF
- Program committee appointments:
 - 2017– : no longer accepting program committee requests
 - VMCAI [International Conference on Verification, Model Checking, and Abstract Interpretation], 2017
 - CAV [International Conference on Computer-Aided Verification], 2015
 - CAV [International Conference on Computer-Aided Verification], 2014
 - CAV [International Conference on Computer-Aided Verification], 2013
 - RTA [Rewriting Techniques and Applications], 2013
 - PLDI [International Conference on Programming Language Design and Implementation], 2012
 - FMCAD [Formal Methods in Computer-Aided Design], 2012
 - SAS [International Static Analysis Symposium], 2011
 - FM [International Symposium on Formal Methods], 2011
 - PADL [International Symposium on Practical Aspects of Declarative Languages], 2011
 - SAS [International Static Analysis Symposium], 2010
 - CAV [International Conference on Computer-Aided Verification], 2010
 - CAV [International Conference on Computer-Aided Verification], 2009
 - FMICS [Formal Methods for Industrial Critical Systems], 2009
 - WST [International Workshop on Termination], 2009
 - POPL [Symposium on Principles of Programming Languages], 2008
 - TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], 2008
 - SSV [International Workshop on Systems Software Verification], 2008
 - LPAR [International Conference on Logic for Programming Artificial Intelligence and Reasoning], 2007

- VMCAI [International Conference on Verification, Model Checking, and Abstract Interpretation], 2007
- TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], 2007
- SMT [International Workshop on Satisfiability Modulo Theories], 2007
- SV [International Workshop on System Verification], 2007
- TV [Thread Verification Workshop], 2006
- PDPAR [Pragmatics of Decision Procedures in Automated Reasoning], 2006
- SoftMC [Software Model Checking Workshop], 2005
- CUFP [ICFP Workshop on Commercial Users of Functional Programming], 2004
- SoftMC [Software Model Checking Workshop], 2003
- CHARME [Advanced Research Working Conference on Correct Hardware Design and Verification Methods], 2003
- Co-organizer, HCSS [International Conference on High Confidence Software and Systems], Annapolis, 2013
- External review committee, PLDI [International Conference on Programming Language Design and Implementation], 2013
- External review committee, PLDI [International Conference on Programming Language Design and Implementation], 2010
- Steering committee, International Workshop on Satisfiability Modulo Theories
- Co-chair, CAV [International Conference on Computer-Aided Verification], Edinburgh, 2010
- Co-chair, FMICS [Formal Methods for Industrial Critical Systems], Eindhoven, 2009
- Workshops chair, CAV [International Conference on Computer-Aided Verification], Princeton, 2008
- Associate Editor, ACM Transactions on Programming Languages and Systems, 2009-2012
- Guest co-editor, Journal on Satisfiability, Boolean Modeling, and Computation (Special Issue on Satisfiability Modulo Theories),
- Program chair for tools papers, TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], 2008
- Co-organizer, *Seminar on Deduction and Decision Procedures*, Schloss Dagstuhl, 2007
- Co-chair, AHA [International Symposium on Automatic Heap Analysis], 2007
- Co-chair, VMCAI [International Conference on Verification, Model Checking, and Abstract Interpretation], 2007
- Co-organizer, SSPV [Symposium on SAT-solvers and Program Verification], 2006
- Co-chair, PDPAR [Pragmatics of Decision Procedures in Automated Reasoning], 2006
- Program chair for tools papers, TACAS [International Conference on Tools and Algorithms for the Construction and Analysis of Systems], 2007
- Co-organizer of SoftMC'05 and SoftMC'03 [CAV Workshop on Software Model Checking]
- Co-organizer of CFDP'05 [Cambridge Forum on Decision Procedures]

- External Ph.D. reviewer: Albert Oliveras (Barcelona), Stephen Magill (CMU), Daron Vroon (Georgia Tech), Jonathan Heusser (Queen Mary), Jules Villard (Cachan), Elzbieta Krepska (VU Amsterdam)
- Grant reviewing: UK Engineering and Physical Sciences Research Council (EPSRC), European Research Council (ERC)
- Committee EAPLS Best PhD Dissertation Award 2011

Work history

Academic positions

- University College London, Professor (joint with Microsoft and now Amazon), 2012-Current
- Queen Mary, University of London, Professor (joint with Microsoft), 2008-2012
- Carnegie Mellon University, Visiting Professor, 2008
- Queen Mary, University of London, Visiting Professor, 2006-2008
- Chalmers University, Visiting Lecturer, 2005-2006
- The Evergreen State College, Adjunct Lecturer, 1996-1998
- Portland Community College, Adjunct Lecturer, 1996-1997
- Oregon Institute of Technology, Adjunct Lecturer, 1997

Industrial positions

- Amazon and AWS, 2014-Current
 - Vice President and Distinguished Scientist, 2021-current
 - Senior Principal Applied Scientist, Engineer, and Director, 2014-2021
- Microsoft and Microsoft Research 2002-2014
 - Principal Researcher, group manager (Programming, Principles, and Tools group), MSR-Cambridge senior leadership team member 2011-2014
 - Principal Researcher, 2009-2011,
 - Senior Researcher, 2008-2009,
 - Researcher, 2004-2008
 - Software developer (in the Base OS kernel team), 2002-2004
- Prover Technology AB, Pre-sales engineer, 1999-2002
- Intel Strategic CAD Labs, Software developer, 1998-1999

Education

- Ph.D. The Oregon Graduate Institute of Science and Technology, 2005, Advisor: John Launchbury
- B.Sci. The Evergreen State College, 1995.
- Secondary/high school – The Jefferson County Open School (Colorado), 1990

References

- Eric Brandwine – Amazon
Contact information available on request
- Prof. Dr. Jasmin Fisher – University College London
`jasmin.fisher@ucl.ac.uk`
- Prof. Dr. Daniel Kroening – Oxford University and Amazon
`kroening@cs.ox.ac.uk`
- Brad Martin – DARPA
Contact information available on request
- Prof. Dr. Peter O’Hearn – University College London and Facebook
`p.ohearn@ucl.ac.uk`
- Prof. Dr. Moshe Vardi – Rice University
`vardi@cs.rice.edu`