Madhusudan Parthasarathy

Curriculum Vitae

Department of Computer Science
Univ. of Illinois at Urbana-Champaign
(217) 721 7977
(217) 244 1323
(217) 265 4035
madhu@illinois.edu
http://madhu.cs.illinois.edu

Education

1994 **Bachelor of Science (B.Sc.) in Mathematics**, *Loyola College, Madras University*, Chennai, India

1996 **Master of Science (M.Sc.)in Theoretical Computer Science**, *Institute of Mathematical Sciences, Anna University*, Chennai, India

2001 **Doctor of Philosophy (Ph.D) in Theoretical Computer Science**, *Institute of Mathematical Sciences, University of Madras*, Chennai, India

Research Interests

Automated software verification, formal methods, security and privacy, programming languages, software engineering, program synthesis, interpretable machine learning, neuro-symbolic learning, logic, and automata theory.

Academic Positions

2018-present Professor, Dept. of Computer Science, University of Illinois at Urbana-Champaign

2011-2018 Associate Professor, Dept. of Computer Science, University of Illinois at Urbana-Champaign

2005-2011 Assistant Professor, Dept. of Computer Science, University of Illinois at Urbana-Champaign

2002-2004 Postdoctoral Researcher, Department of Computer and Information Sciences, University of Pennsylvania

Experience

Aug 2019 – Aug 2020 **Sabbatical**, Halıcıoğlu Data Science Institute (HDSI), University of California, San Diego (UCSD).

Oct 2012 – Aug 2013 **Visiting Researcher**, Microsoft Research, Bangalore, India
Part of Sabbatical year. Initiated and was involved in developing a MOOC platform for India called Massively Empowered Classrooms (MEC) that provides online resources in a blended learning platform for CS undergraduates in India. Also directed first course on this platform on Design and Analysis of Algorithms. `https://www.mecr.org/`

2000 **Research Assistant**, RWTH, Aachen, Germany
∼9 month visit during Ph.D. visiting the research group of Prof. Wolfgang Thomas, and helping teach a course on model-checking.

Online bibliography databases

- I publish usually under the name "P. Madhusudan", and sometimes under "Madhusudan Parthasarathy" or "Parthasarathy Madhusudan"
- DBLP: https://dblp.org/pid/m/PMadhusudan
- Google Scholar: https://scholar.google.com/citations?user=V828uG8AAAAJ ; **h-index: 46**
- Semantic Scholar: https://www.semanticscholar.org/author/P.-Madhusudan/145104529
- Orcid ID: 0000-0002-9782-721X

Awards, Honors, and Highlights

- Served on several NSF panels

2008 National Science Foundation (NSF) Faculty Early Career Development (CAREER) Award

2009 List of Teachers Ranked as Excellent by Their Students (for CS373: Theory of Computation: Spring 2009)

2010 Best paper award for paper titled: *"VEX: Vetting Browser Extensions For Security Vulnerabilities"*, USENIX Security Symposium. Washington D.C, USA, August 2010.

2011 Invited paper for Research Highlight in the Communications of the ACM: *"Vetting browser extensions for security vulnerabilities with VEX"*

2012 PC Co-Chair, International Conference on Computer Aided Verification, Berkeley, CA, 2012.

2012 Invited talk at IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS) on *Automated Reasoning and Natural Proofs for Programs Manipulating Data Structures*, Hyderabad, India.

2015 Invited Tutorial at Conference on Computer Aided Verification on *Machine-learning based methods for synthesizing invariants.*, 2015.

2017 Invited Tutorial at Conference on Runtime Verification on *Machine-learning State Properties*, 2017.

2018 Best Paper Award at Mathematical Foundations of Computer Science (MFCS) for the paper *Lagrange's theorem for binary squares*

2021 Invited Tutorial at ETAPS on *Learning Logical Expressions with Applications to Verification, Specification Mining, and Proofs*, 2021.

2022 Distinguished Paper Award at POPL 2022 for the paper *Learning formulas in finite variable logics*

Invited talks and lectures

- "Making the Stack Visible: Visibly Pushdown Automata," *Logic and Computational Complexity (LCC) 2005 Workshop with LICS,* Chicago, IL, June 2005.
- "Mining Dynamic Interfaces," *Foundations of Interface Technologies (FIT) 2005 Workshop with CONCUR,* San Francisco, CA, August 2005.
- "Automata theory for nested structures," *GALOP'06: Games for Logic and Programming Languages, part of FLoC (Federated Logic Conference)*, Seattle, WA, 2006.
- "Visibly pushdown automata for XML," *EROW: Workshop on Emerging Research Opportunities in Web Data Management (held with ICDT)*, Barcelona, Spain, January 2007.
- "Learning Algorithms and Formal Verification," Invited tutorial, *8th Int'l Conference on Verification, Model Checking and Abstract Interpretation (VMCAI)*, Nice, France, January 2007.
- "Learning algorithms and formal verification," *Institute of Mathematic Sciences*, Chennai, India, February, 2007.

- "Logic, Automata, and Algorithms," *Invited course at Universita degli Studi di Salerno*, Salerno, Italy, June 2007.
- "Multi-stack Automata: A New Tractable Subclass,", *Microsoft Research*, Redmond, WA, May 2007.
- "Analysing heaps using automata," *IFIP Working Group 2.2 (International Federation for Information Processing)*, Nancy, France, September, 2007.
- "Monitoring Serializability," *Microsoft Research*, Redmond, WA, August 2008.
- "Finding Concurrency Bugs through Atomicity Violations," *UPCRC Seminar (audience: UIUC, Microsoft, Intel)*, Urbana, IL, October, 2008.
- "Annotations for race-freedom," *Dagstuhl Workshop on Design and Validation of Concurrent Systems,* Dagstuhl, Germany, September 2009.
- "Annotations for Race-freedom," Chennai Mathematical Institute (CMI), Chennai, India, August, 2009.
- "Provable annotations for race-freedom," *Technische Universitat Darmstadt (Darmstadt University)*, Darmstadt, Germany, September, 2009.
- "Correctness projects in UPCRC," *UPCRC Summit (audience: UIUC, Microsoft, Intel)*, Urbana, IL, March, 2010.
- "Deciding automata with auxiliary storage," *Invited talk at International Conference on Implementation and Application of Automata (CIAA),* Winnipeg, Canada, August, 2010.
- "The role of automata theory in software verification," *CS Dept, University of Wisconsin,* Madison, Wisconsin, September 2010.
- "The role of automata theory in software verification," *CERIAS Security Seminar, University of Purdue*, Purdue, Indiana, October, 2010.
- "Synthesizing Programs using Bounded Domains and Occam's Razor," *Invited talk at the 1st Workshop on Synthesis (SYNT 2012)*, Berkeley, California, July 2012.
- "Automated Reasoning and Natural Proofs for Programs Manipulating Data Structures", *Invited talk at 32nd International Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, Hyderabad, India, December, 2012.
- "Synthesizing Programs over Bounded Data Domains," Workshop on Verification of Infinite-State Systems, co-held with FSTTCS, Hyderabad, India, December, 2012.
- "Automata and Learning Based Methods in Software Verification," *Invited Series of lectures for the AlgoSyn: Fall School on Algorithmic Game Theory and Learning*, a series of lectures on learning based techniques for program verification to a group of graduate students who gathered from all over Germany and some from other parts of Europe, RWTH Aachen, Aachen, Germany, October, 2013.
- "Machine-learning based methods for synthesizing invariants," Tutorial at Conference on Computer Aided Verification (CAV 2015), San Francisco, CA, July 20, 2015.
  Material available at `http://madhu.cs.illinois.edu/CAV15Tutorial/`
- Multiple talks at the NSF Expeditions ExCAPE meetings held semi-annually in the years 2013-2017.
- "Foundations for Natural Proofs and Quantifier Instantiation" *Talk at Microsoft Research*, Seattle, WA, 2017.
- "Machine-learning State Properties," *Invited tutorial at the 17th International Conference on Runtime Verification*, Seattle, WA, 2017.
- Invited talk, February 8-9, 2018. Invited talk at a workshop on Program Synthesis and Machine Learning, University of Washington, Seattle.
- "Learning Logical Expressions with Applications to Verification, Specification Mining, and Proofs", *Invited tutorial at the European Joint Conferences on Theory and Practice of Software (ETAPS), 2021.*
- "Exploring Combinations of Neural and Logic Learning", *Invited talk at "When deep learning meets logic": workshop on neural-symbolic integration sponsored by Samsung Research*, 2021.

Some funding:

- Gift from Amazon, unrestricted funds, $50K, 2021.
- Discovery Partners Institute (DPI) Seed Grant for "Trustworthy and Robust AI", $125K, 2020.
- Discovery Partners Institute (DPI) Seed Grant for "Privacy in the Era of Big Data", $125K, 2020.
- NSF Small: Automating Software Verification using Natural Proofs, single PI, $500K, 2015-2018.
- NSF Expeditions in Computing: ExCAPE: Expeditions in Computer Augmented Program Engineering,

multi-PI grant, $10M, (my share: $500K plus some central funds and centrally funded postdocs), 2012-2017.
- Intel Parallel Center, $2M, (my share $\sim$ 9%, i.e., $180K), 2012-2013.
- NSF TC: Small: TC: Collaborative Research: Formal Security Analysis of Access Control Models and Extensions, $475K (my share: $200K), 2009-2012.
- UPCRC: Universal Parallel Computing Research Center (Lead: Correctness group), Microsoft/Intel, $10M (my share $\sim$ 5%, i.e., $300K), 2008-2011.
- NSF TC: Small: Keeping Jack in the Box: Confining the Role of Untrusted Inputs in Web Scenarios, 2 PIs, $450K (my share: $225K), 2009-2012.
- NSF CSR-EHCS (EHS), TM:Compositional Technology for Safety-Critical Modular Systems, multi-PI grant, $300K (my share: $\sim$$50K), 2008-2009.
- NSF Career Grant, single PI, "The Automata Theoretic Method in Software Verification", $400K, 2008–2012.
- Gift from Microsoft Research, unrestricted funds, $10K, 2005.

Conference Program Committees

- Program Committee, FORMATS and FTRTFT 2004 Joint Conference on Formal Modelling and Analysis of Timed Systems (FORMATS) and Formal Techniques in Real-Time and Fault Tolerant System (FTRTFT).
- Program Committee, Games in Design and Verification (GDV), 2005 (with CAV 2005). Program Committee, 17th Int'l Conference on Computer Aided Verification (CAV), Edinburgh, Scotland, 2005.
- Program Committee, 25th Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS), Hyderabad, India, 2005.
- Program Committee, ACM Symposium on Applied Computing (SAC): Technical Track on Software Verification, 2006.
- Program Committee, 34th International Colloquium on Automata, Languages and Programming (ICALP), Wroclaw, Poland, 2007.
- Program Committee, 19th International Conference on Concurrency Theory (CONCUR), Toronto, Canada, 2008.
- Program Committee, Sixth ASIAN Symposium on Programming Languages and Systems (APLAS), Bangalore, India, 2008.
- Program Committee, Annual IEEE Symposium on Logic in Computer Science (LICS), Los Angeles, USA, 2009.
- Program Committee: 16th International Symposium on Temporal Representation and Reasoning (TIME), Brixen-Bressanone, Italy, 2009.
- Program Committee: 21st Int'l Conf on Computer Aided Verification (CAV), Grenoble, France, 2009. External Review Committee: ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI), Dublin, Ireland, 2009.
- Program Committee, 27th Symposium on Theoretical Aspects of Computer Science (STACS), Nancy, France, 2010.
- Program Committee: 8th Int'l Symposium on Automated Technology for Verification and Analysis (ATVA), Singapore, 2010.
- Program Committee: 30th Int'l Conference on Foundations of Software Tech. and Theoretical Comp. Sc. (FSTTCS), Chennai, India, 2010.
- Program Committee: 23rd Int'l Conf on Computer Aided Verification (CAV), Snowbird, UT, USA, 2011.
- Program Committee: Automated Technology for Verification and Analysis, 9th International Symposium, ATVA, Taipei, Taiwan, 2011.
- Program Committee: IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS), Mumbai, India, 2011.

- Program Committee: 18th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR), Mérida, Venezuela, 2012.
- Program Committee: 39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL), Philadelphia, USA, 2012.
- Program Committee: 38th Int'l Symp. on Mathematical Foundations of Computer Science (MFCS), IST Austria, Austia, 2013.
- Program Committee: 25th Int'l Conf on Computer Aided Verification (CAV), Saint Petersburg, Russia, 2013.
- Program Committee: Conference on Highlights of Logic, Games, and Automata, Paris, France, 2013.
- Program Committee: 21st International Static Analysis Symposium (SAS), Munich, Germany, 2014.
- Program Committee: 25th Conference on Concurrency Theory (CONCUR), Rome, Italy, 2014.
- Program Committee: 35th annual ACM SIGPLAN conference on Programming Language Design and Implementation (PLDI), Edinburgh, UK, 2014.
- Program Committee: 42nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL), Mumbai, India, 2015.
- Program Committee: 10th Symposium of the Trustworthy Global Computing (TGC), Madrid, Spain, 2015.
- Program Committee: 36th annual ACM SIGPLAN conference on Programming Language Design and Implementation (PLDI), Portland, OR, USA, 2015.
- External Review Committee: 28th International Conference on Computer Aided Verification (CAV), Toronto, Canada, 2016.
- Program Committee: 43rd International Colloquium on Automata, Languages, and Programming (ICALP), Rome, Italy, 2016.
- Program Committee: 23rd International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS), Uppsala, Sweden, 2017.
- Program Committee: 5th Conference on Highlights of Logic, Automata and Games, London, UK, September 12–15, 2017.
- Program Committee: The 17th International Conference on Runtime Verification (RV 2017), Seattle, Washington, September 13-16, 2017.
- Program Committee, Workshop on Automated Deduction for Separation Logics (ADSL), affiliated with the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2018) and part of the Federated Logic Conference 2018 (FLOC 2018), Oxford, UK, 2018.
- Program Committee, Workshop on Logic and Learning, affiliated with the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2018) and part of the Federated Logic Conference 2018 (FLOC 2018), Oxford, UK, 2018.
- Program Committee, 29th International Conference on Concurrency Theory (CONCUR 2018), Beijing, China, 2018.
- Program Committee: 46th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL), Lisbon, Portugal, 2019.
- Program Committee, 46th International Colloquium on Automata, Languages and Programming (ICALP 2019), Patras, Greece, 2019.
- Program Committee, 46th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL), Cascais, Portugal, 2019.
- Program Committee, 46th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL), Cascais, Portugal, 2019.
- 14th-15th International Conference on Language and Automata Theory and Applications (LATA 2020 and LATA 2021), Milan, Italy, 2021.
- Program Committee, 49th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL), Philadelphia, United States, 2022.

- Program Committee, 17th International Computer Science Symposium in Russia (CSR), 2022, online.
- Program Committee, OOPSLA/SPLASH, Auckland, New Zealand, 2022.

Conferences Chaired or Organized

- Program Chair (Organizer), Workshop on Software Verification, 2005 (part of FSTTCS 2005) Program Chair, Workshop on Games in Design and Verification (GDV'06); co-located with FLoC (Federated Logic Conference), Seattle, USA, 2006.
- Program Chair, 9th International Workshop on Verification of Infinite-State Systems, (INFINITY), Lisbon, Portugal, 2007.
- Organizer, Workshop on Security and Reliability in Software Systems, with FSTTCS, Bangalore, India, 2008.
- Organizer: Dagstuhl Workshop on Design and Validation of Concurrent Systems, Dagstuhl, Germany, August, 2009.
- Program Co-Chair, 24th Int'l Conf on Computer Aided Verification (CAV), Berkeley, USA, 2012. Program Co-Chair and co-Organizer: SYNT 2015: 4th Workshop on Synthesis (with Conference on Computer Aided Verification-CAV), San Francisco, USA, 2015.
- Program Co-Chair, Workshop on Formal Methods and Security, held with PLDI 2016, Santa Barbara, USA, 2016.
- Program Co-Chair, Workshop on Formal Methods and Security, held with PLDI 2018 (upcoming), Philadelphia, USA, 2018.
- Program Co-Chair, Workshop on Formal Methods and Security, held with PLDI 2018 (upcoming), Philadelphia, USA, 2018.

Conference Publications (including conferences that publish proceedings in journals)

[1] Adithya Murali, Lucas Pena, Eion Blanchard, Christof Löding, and P. Madhusudan. Model-guided synthesis of inductive lemmas for fol with least fixpoints. *Conditionally accepted to a conference.*, 2022.

[2] Zhengyao Lin, Paul Krogmeier, Adithya Murali, and P. Madhusudan. Synthesizing axiomatizations using reasoning and logic learning. *Conditionally accepted to a conference.*, 2022.

[3] Adithya Murali, Atharva Sehgal, Paul Krogmeier, and P. Madhusudan. Composing neural learning and symbolic reasoning with an application to visual discrimination. In Luc De Raedt, editor, *Proc. Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria*, pages 3358–3365. ijcai.org, 2022.

[4] Paul Krogmeier and P. Madhusudan. Learning formulas in finite variable logics. *Proc. ACM Program. Lang.*, 6(Conference on Principles of Programming Languages (POPL)):1–28, 2022 **(Distinguished Paper Award)**.

[5] Angello Astorga, Shambwaditya Saha, Ahmad Dinkins, Felicia Wang, P. Madhusudan, and Tao Xie. Synthesizing contracts correct modulo a test generator. *Proc. ACM Program. Lang.*, 5(Conf. on Object-Oriented Programming, Systems, Languages & Applications (OOPSLA)):1–27, 2021.

[6] Umang Mathur, Adithya Murali, Paul Krogmeier, P. Madhusudan, and Mahesh Viswanathan. Deciding memory safety for single-pass heap-manipulating programs. *Proc. ACM Program. Lang.*, 4(Conference on Principles of Programming Languages (POPL)):35:1–35:29, 2020.

[7] Paul Krogmeier, Umang Mathur, Adithya Murali, P. Madhusudan, and Mahesh Viswanathan. Decidable synthesis of programs with uninterpreted functions. In Shuvendu K. Lahiri and Chao Wang, editors, *Computer Aided Verification - 32nd Int'l Conf., CAV 2020, Los Angeles, CA, USA, Proceedings, Part II*, volume 12225 of *Lecture Notes in Computer Science*, pages 634–657. Springer, 2020.

[8] Adithya Murali, Lucas Peña, Christof Löding, and P. Madhusudan. A First-Order Logic with Frames. In Peter Müller, editor, *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 515–543. Springer, 2020.

[9] Umang Mathur, P. Madhusudan, and Mahesh Viswanathan. What's decidable about program verification modulo axioms? In Armin Biere and David Parker, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 26th Int'l Conf. TACAS 2020, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2020, Dublin, Ireland, April 25-30, 2020, Proceedings, Part II*, volume 12079 of *Lecture Notes in Computer Science*, pages 158–177. Springer, 2020.

[10] Umang Mathur, P. Madhusudan, and Mahesh Viswanathan. Decidable verification of uninterpreted programs. *Proc. ACM Program. Lang.*, 3(Conference on Principles of Programming Languages (POPL)):46:1–46:29, 2019.

[11] Faria Kalim, Karl Palmskog, Jayasi Mehar, Adithya Murali, Indranil Gupta, and P. Madhusudan. Kaizen: Building a performant blockchain system verified for consensus and integrity. In Clark W. Barrett and Jin Yang, editors, *2019 Formal Methods in Computer Aided Design, FMCAD 2019, San Jose, CA, USA, October 22-25, 2019*, pages 96–104. IEEE, 2019.

[12] Salvatore La Torre and P. Madhusudan. Reachability in concurrent uninterpreted programs. In Arkadev Chattopadhyay and Paul Gastin, editors, *39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2019, December 11-13, 2019, Bombay, India*, volume 150 of *LIPIcs*, pages 46:1–46:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[13] Angello Astorga, P. Madhusudan, Shambwaditya Saha, Shiyu Wang, and Tao Xie. Learning stateful preconditions modulo a test generator. In Kathryn S. McKinley and Kathleen Fisher, editors, *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2019, Phoenix, AZ, USA, June 22-26, 2019*, pages 775–787. ACM, 2019.

[14] Daniel Neider, Shambwaditya Saha, Pranav Garg, and P. Madhusudan. Sorcar: Property-driven algorithms for learning conjunctive invariants. In Bor-Yuh Evan Chang, editor, *Static Analysis - 26th International Symposium, SAS 2019, Porto, Portugal, October 8-11, 2019, Proceedings*, volume 11822 of *Lecture Notes in Computer Science*, pages 323–346. Springer, 2019.

[15] P. Ezudheen, Daniel Neider, Deepak D'Souza, Pranav Garg, and P. Madhusudan. Horn-ice learning for synthesizing invariants and contracts. *Proc. ACM Program. Lang.*, 2(Conf. on Object-Oriented Programming, Systems, Languages & Applications (OOPSLA)):131:1–131:25, 2018.

[16] Christof Löding, P. Madhusudan, and Lucas Peña. Foundations for natural proofs and quantifier instantiation. *Proc. ACM Program. Lang.*, 2(Conference on Principles of Programming Languages (POPL)):10:1–10:30, 2018.

[17] P. Madhusudan, Dirk Nowotka, Aayush Rajasekaran, and Jeffrey O. Shallit. Lagrange's theorem for binary squares. In Igor Potapov, Paul G. Spirakis, and James Worrell, editors, *43rd Int'l Symp. on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK*, volume 117 of *LIPIcs*, pages 18:1–18:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018 **(Best Paper Award)**.

[18] Daniel Neider, Pranav Garg, P. Madhusudan, Shambwaditya Saha, and Daejun Park. Invariant synthesis for incomplete verification engines. In Dirk Beyer and Marieke Huisman, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 24th Int'l Conf. of TACAS 2018, part of ETAPS, Thessaloniki, Greece, April, 2018, Proceedings, Part I*, volume 10805 of *Lecture Notes in Computer Science*, pages 232–250. Springer, 2018.

[19] Alex Gyori, Pranav Garg, Edgar Pek, and P. Madhusudan. Efficient incrementalized runtime checking of linear measures on lists. In *2017 IEEE International Conference on Software Testing, Verification and Validation, ICST 2017, Tokyo, Japan, March 13-17, 2017*, pages 310–320. IEEE Computer Society, 2017.

[20] Pranav Garg, Daniel Neider, P. Madhusudan, and Dan Roth. Learning invariants using decision trees and implication counterexamples. In Rastislav Bodík and Rupak Majumdar, editors, *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20 - 22, 2016*, pages 499–512. ACM, 2016.

[21] Christof Löding, P. Madhusudan, and Daniel Neider. Abstract learning frameworks for synthesis. In Marsha Chechik and Jean-François Raskin, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 22nd International Conference, TACAS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*, volume 9636 of *Lecture Notes in Computer Science*, pages 167–185. Springer, 2016.

[22] Daniel Neider, Shambwaditya Saha, and P. Madhusudan. Synthesizing piece-wise functions by learning classifiers. In Marsha Chechik and Jean-François Raskin, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 22nd International Conference, TACAS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*, volume 9636 of *Lecture Notes in Computer Science*, pages 186–203. Springer, 2016.

[23] Shambwaditya Saha, Pranav Garg, and P. Madhusudan. Alchemist: Learning guarded affine functions. In Daniel Kroening and Corina S. Pasareanu, editors, *Computer Aided Verification - 27th International Conference, CAV 2015, San Francisco, CA, USA, July 18-24, 2015, Proceedings, Part I*, volume 9206 of *Lecture Notes in Computer Science*, pages 440–446. Springer, 2015.

[24] Shambwaditya Saha, Santhosh Prabhu, and P. Madhusudan. Netgen: synthesizing data-plane configurations for network policies. In Jennifer Rexford and Amin Vahdat, editors, *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research, SOSR '15, Santa Clara, California, USA, June 17-18, 2015*, pages 17:1–17:6. ACM, 2015.

[25] Pranav Garg, Christof Löding, P. Madhusudan, and Daniel Neider. ICE: A robust framework for learning invariants. In Armin Biere and Roderick Bloem, editors, *Computer Aided Verification - 26th International Conference, CAV 2014, Held as Part of the Vienna Summer of Logic, VSL 2014, Vienna, Austria, July 18-22, 2014. Proceedings*, volume 8559 of *Lecture Notes in Computer Science*, pages 69–87. Springer, 2014.

[26] Anna Lisa Ferrara, P. Madhusudan, Truc L. Nguyen, and Gennaro Parlato. Vac - verifier of administrative role-based access control policies. In Armin Biere and Roderick Bloem, editors, *Computer Aided Verification - 26th International Conference, CAV 2014, Held as Part of the Vienna Summer of Logic, VSL 2014, Vienna, Austria, July 18-22, 2014. Proceedings*, volume 8559 of *Lecture Notes in Computer Science*, pages 184–191. Springer, 2014.

[27] Andrew Cross, B. Ashok, Srinath Bala, Edward Cutrell, Naren Datha, Rahul Kumar, Viraj Kumar, Parthasarathy Madhusudan, Siddharth Prakash, Sriram K. Rajamani, Satish Sangameswaran, Deepika Sharma, and William Thies. Online learning versus blended learning: an exploratory study. In Mehran Sahami, Armando Fox, Marti A. Hearst, and Michelene T. H. Chi, editors, *First (2014) ACM Conference on Learning @ Scale, L@S 2014, Atlanta, GA, USA, March 4-5, 2014*, pages 179–180. ACM, 2014.

[28] Ankush Desai, Pranav Garg, and P. Madhusudan. Natural proofs for asynchronous programs using almost-synchronous reductions. In Andrew P. Black and Todd D. Millstein, editors, *Proceedings of the 2014 ACM International Conference on Object Oriented Programming Systems Languages & Applications, OOPSLA 2014, part of SPLASH 2014, Portland, OR, USA, October 20-24, 2014*, pages 709–725. ACM, 2014.

[29] Edgar Pek, Xiaokang Qiu, and P. Madhusudan. Natural proofs for data structure manipulation in C using separation logic. In Michael F. P. O'Boyle and Keshav Pingali, editors, *ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI '14, Edinburgh, United Kingdom - June 09 - 11, 2014*, pages 440–451. ACM, 2014.

[30] Haohui Mai, Edgar Pek, Hui Xue, Samuel Talmadge King, and Parthasarathy Madhusudan. Verifying security invariants in expressos. In Vivek Sarkar and Rastislav Bodík, editors, *Architectural Support for Programming Languages and Operating Systems, ASPLOS '13, Houston, TX, USA - March 16 - 20, 2013*, pages 293–304. ACM, 2013.

[31] Pranav Garg, Christof Löding, P. Madhusudan, and Daniel Neider. Learning universally quantified invariants of linear data structures. In Natasha Sharygina and Helmut Veith, editors, *Computer Aided Verification - 25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings*, volume 8044 of *Lecture Notes in Computer Science*, pages 813–829. Springer, 2013.

[32] Xiaokang Qiu, Pranav Garg, Andrei Stefanescu, and Parthasarathy Madhusudan. Natural proofs for structure, data, and separation. In Hans-Juergen Boehm and Cormac Flanagan, editors, *ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI '13, Seattle, WA, USA, June 16-19, 2013*, pages 231–242. ACM, 2013.

[33] Pranav Garg, P. Madhusudan, and Gennaro Parlato. Quantified data automata on skinny trees: An abstract domain for lists. In Francesco Logozzo and Manuel Fähndrich, editors, *Static Analysis - 20th International Symposium, SAS 2013, Seattle, WA, USA, June 20-22, 2013. Proceedings*, volume 7935 of *Lecture Notes in Computer Science*, pages 172–193. Springer, 2013.

[34] Anna Lisa Ferrara, P. Madhusudan, and Gennaro Parlato. Policy analysis for self-administrated role-based access control. In Nir Piterman and Scott A. Smolka, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 19th International Conference, TACAS 2013, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2013, Rome, Italy, March 16-24, 2013. Proceedings*, volume 7795 of *Lecture Notes in Computer Science*, pages 432–447. Springer, 2013.

[35] Anna Lisa Ferrara, P. Madhusudan, and Gennaro Parlato. Security analysis of role-based access control through program verification. In Stephen Chong, editor, *25th IEEE Computer Security Foundations Symposium, CSF 2012, Cambridge, MA, USA, June 25-27, 2012*, pages 113–125. IEEE Computer Society, 2012.

[36] Parthasarathy Madhusudan. Automated reasoning and natural proofs for programs manipulating data structures. In Deepak D'Souza, Telikepalli Kavitha, and Jaikumar Radhakrishnan, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2012, December 15-17, 2012, Hyderabad, India*, volume 18 of *LIPIcs*, pages 34–35. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.

[37] Parthasarathy Madhusudan, Xiaokang Qiu, and Andrei Stefanescu. Recursive proofs for inductive tree data-structures. In John Field and Michael Hicks, editors, *Proceedings of the 39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2012, Philadelphia, Pennsylvania, USA, January 22-28, 2012*, pages 123–136. ACM, 2012.

[38] Emre Uzun, Vijayalakshmi Atluri, Shamik Sural, Jaideep Vaidya, Gennaro Parlato, Anna Lisa Ferrara, and Parthasarathy Madhusudan. Analyzing temporal role based access control models. In Vijay Atluri, Jaideep Vaidya, Axel Kern, and Murat Kantarcioglu, editors, *17th ACM Symposium on Access Control Models and Technologies, SACMAT '12, Newark, NJ, USA - June 20 - 22, 2012*, pages 177–186. ACM, 2012.

[39] Azadeh Farzan, P. Madhusudan, Niloofar Razavi, and Francesco Sorrentino. Predicting null-pointer dereferences in concurrent programs. In Will Tracz, Martin P. Robillard, and Tevfik Bultan, editors, *20th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-20), SIGSOFT/FSE'12, Cary, NC, USA - November 11 - 16, 2012*, page 47. ACM, 2012.

[40] Rohit Chadha, P. Madhusudan, and Mahesh Viswanathan. Reachability under contextual locking. In Cormac Flanagan and Barbara König, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 18th International Conference, TACAS 2012, Held as Part of the European*

Joint Conferences on Theory and Practice of Software, ETAPS 2012, Tallinn, Estonia, March 24 - April 1, 2012. Proceedings, volume 7214 of *Lecture Notes in Computer Science*, pages 437–450. Springer, 2012.

[41] Salvatore La Torre, P. Madhusudan, and Gennaro Parlato. Sequentializing parameterized programs. In Sebastian S. Bauer and Jean-Baptiste Raclet, editors, *Proceedings Fourth Workshop on Foundations of Interface Technologies, FIT 2012, Tallinn, Estonia, 25th March 2012.*, volume 87 of *EPTCS*, pages 34–47, 2012.

[42] P. Madhusudan and Sanjit A. Seshia, editors. *Computer Aided Verification - 24th International Conference, CAV 2012, Berkeley, CA, USA, July 7-13, 2012 Proceedings*, volume 7358 of *Lecture Notes in Computer Science*. Springer, 2012.

[43] Parthasarathy Madhusudan. Synthesizing reactive programs. In Marc Bezem, editor, *Computer Science Logic, 25th International Workshop / 20th Annual Conference of the EACSL, CSL 2011, September 12-15, 2011, Bergen, Norway, Proceedings*, volume 12 of *LIPIcs*, pages 428–442. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2011.

[44] P. Madhusudan and Gennaro Parlato. The tree width of auxiliary storage. In Thomas Ball and Mooly Sagiv, editors, *Proceedings of the 38th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2011, Austin, TX, USA, January 26-28, 2011*, pages 283–294. ACM, 2011.

[45] P. Madhusudan, Gennaro Parlato, and Xiaokang Qiu. Decidable logics combining heap structures and data. In Thomas Ball and Mooly Sagiv, editors, *Proceedings of the 38th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2011, Austin, TX, USA, January 26-28, 2011*, pages 611–622. ACM, 2011.

[46] Rajesh K. Karmani, P. Madhusudan, and Brandon M. Moore. Thread contracts for safe parallelism. In Calin Cascaval and Pen-Chung Yew, editors, *Proceedings of the 16th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, PPOPP 2011, San Antonio, TX, USA, February 12-16, 2011*, pages 125–134. ACM, 2011.

[47] P. Madhusudan and Xiaokang Qiu. Efficient decision procedures for heaps using STRAND. In Eran Yahav, editor, *Static Analysis - 18th International Symposium, SAS 2011, Venice, Italy, September 14-16, 2011. Proceedings*, volume 6887 of *Lecture Notes in Computer Science*, pages 43–59. Springer, 2011.

[48] Pranav Garg and P. Madhusudan. Compositionality entails sequentializability. In Parosh Aziz Abdulla and K. Rustan M. Leino, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 17th International Conference, TACAS 2011, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2011, Saarbrücken, Germany, March 26-April 3, 2011. Proceedings*, volume 6605 of *Lecture Notes in Computer Science*, pages 26–40. Springer, 2011.

[49] Salvatore La Torre, P. Madhusudan, and Gennaro Parlato. Model-checking parameterized concurrent programs using linear interfaces. In Tayssir Touili, Byron Cook, and Paul B. Jackson, editors, *Computer Aided Verification, 22nd International Conference, CAV 2010, Edinburgh, UK, July 15-19, 2010. Proceedings*, volume 6174 of *Lecture Notes in Computer Science*, pages 629–644. Springer, 2010.

[50] Salvatore La Torre, Parthasarathy Madhusudan, and Gennaro Parlato. The language theory of bounded context-switching. In Alejandro López-Ortiz, editor, *LATIN 2010: Theoretical Informatics,*

9th Latin American Symposium, Oaxaca, Mexico, April 19-23, 2010. Proceedings, volume 6034 of *Lecture Notes in Computer Science*, pages 96–107. Springer, 2010.

[51] Francesco Sorrentino, Azadeh Farzan, and P. Madhusudan. PENELOPE: weaving threads to expose atomicity violations. In Gruia-Catalin Roman and Kevin J. Sullivan, editors, *Proceedings of the 18th ACM SIGSOFT International Symposium on Foundations of Software Engineering, 2010, Santa Fe, NM, USA, November 7-11, 2010*, pages 37–46. ACM, 2010.

[52] Sruthi Bandhakavi, Samuel T. King, P. Madhusudan, and Marianne Winslett. VEX: vetting browser extensions for security vulnerabilities. In *19th USENIX Security Symposium, Washington, DC, USA, August 11-13, 2010, Proceedings*, pages 339–354. USENIX Association, 2010.

[53] Azadeh Farzan, P. Madhusudan, and Francesco Sorrentino. Meta-analysis for atomicity violations under nested locking. In Ahmed Bouajjani and Oded Maler, editors, *Computer Aided Verification, 21st International Conference, CAV 2009, Grenoble, France, June 26 - July 2, 2009. Proceedings*, volume 5643 of *Lecture Notes in Computer Science*, pages 248–262. Springer, 2009.

[54] Salvatore La Torre, P. Madhusudan, and Gennaro Parlato. Reducing context-bounded concurrent reachability to sequential reachability. In Ahmed Bouajjani and Oded Maler, editors, *Computer Aided Verification, 21st International Conference, CAV 2009, Grenoble, France, June 26 - July 2, 2009. Proceedings*, volume 5643 of *Lecture Notes in Computer Science*, pages 477–492. Springer, 2009.

[55] P. Madhusudan and Mahesh Viswanathan. Query automata for nested words. In Rastislav Královic and Damian Niwinski, editors, *Mathematical Foundations of Computer Science 2009, 34th International Symposium, MFCS 2009, Novy Smokovec, High Tatras, Slovakia, August 24-28, 2009. Proceedings*, volume 5734 of *Lecture Notes in Computer Science*, pages 561–573. Springer, 2009.

[56] Salvatore La Torre, P. Madhusudan, and Gennaro Parlato. Analyzing recursive programs using a fixed-point calculus. In Michael Hind and Amer Diwan, editors, *Proceedings of the 2009 ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2009, Dublin, Ireland, June 15-21, 2009*, pages 211–222. ACM, 2009.

[57] Azadeh Farzan and P. Madhusudan. The complexity of predicting atomicity violations. In Stefan Kowalewski and Anna Philippou, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 15th International Conference, TACAS 2009, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2009, York, UK, March 22-29, 2009. Proceedings*, volume 5505 of *Lecture Notes in Computer Science*, pages 155–169. Springer, 2009.

[58] Azadeh Farzan and P. Madhusudan. Monitoring atomicity in concurrent programs. In Aarti Gupta and Sharad Malik, editors, *Computer Aided Verification, 20th International Conference, CAV 2008, Princeton, NJ, USA, July 7-14, 2008, Proceedings*, volume 5123 of *Lecture Notes in Computer Science*, pages 52–65. Springer, 2008.

[59] Lars E. Olson, Carl A. Gunter, and P. Madhusudan. A formal framework for reflective database access control policies. In Peng Ning, Paul F. Syverson, and Somesh Jha, editors, *Proceedings of the 2008 ACM Conference on Computer and Communications Security, CCS 2008, Alexandria, Virginia, USA, October 27-31, 2008*, pages 289–298. ACM, 2008.

[60] Salvatore La Torre, P. Madhusudan, and Gennaro Parlato. An infinite automaton characterization of double exponential time. In Michael Kaminski and Simone Martini, editors, *Computer Science Logic, 22nd International Workshop, CSL 2008, 17th Annual Conference of the EACSL, Bertinoro, Italy, September 16-19, 2008. Proceedings*, volume 5213 of *Lecture Notes in Computer Science*, pages 33–48. Springer, 2008.

[61] Salvatore La Torre, P. Madhusudan, and Gennaro Parlato. Context-bounded analysis of concurrent queue systems. In C. R. Ramakrishnan and Jakob Rehof, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 14th International Conference, TACAS 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29-April 6, 2008. Proceedings*, volume 4963 of *Lecture Notes in Computer Science*, pages 299–314. Springer, 2008.

[62] Sruthi Bandhakavi, Prithvi Bisht, P. Madhusudan, and V. N. Venkatakrishnan. CANDID: preventing sql injection attacks using dynamic candidate evaluations. In Peng Ning, Sabrina De Capitani di Vimercati, and Paul F. Syverson, editors, *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA, October 28-31, 2007*, pages 12–24. ACM, 2007.

[63] Salvatore La Torre, Parthasarathy Madhusudan, and Gennaro Parlato. A robust class of context-sensitive languages. In *22nd IEEE Symposium on Logic in Computer Science (LICS 2007), 10-12 July 2007, Wroclaw, Poland, Proceedings*, pages 161–170. IEEE Computer Society, 2007.

[64] Azadeh Farzan and P. Madhusudan. Causal dataflow analysis for concurrent programs. In Orna Grumberg and Michael Huth, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 13th International Conference, TACAS 2007, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2007 Braga, Portugal, March 24 - April 1, 2007, Proceedings*, volume 4424 of *Lecture Notes in Computer Science*, pages 102–116. Springer, 2007.

[65] P. Madhusudan. Learning algorithms and formal verification (invited tutorial). In Byron Cook and Andreas Podelski, editors, *Verification, Model Checking, and Abstract Interpretation, 8th International Conference, VMCAI 2007, Nice, France, January 14-16, 2007, Proceedings*, volume 4349 of *Lecture Notes in Computer Science*, page 214. Springer, 2007.

[66] Viraj Kumar, P. Madhusudan, and Mahesh Viswanathan. Visibly pushdown automata for streaming XML. In Carey L. Williamson, Mary Ellen Zurko, Peter F. Patel-Schneider, and Prashant J. Shenoy, editors, *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 1053–1062. ACM, 2007.

[67] Azadeh Farzan and P. Madhusudan. Causal atomicity. In Thomas Ball and Robert B. Jones, editors, *Computer Aided Verification, 18th International Conference, CAV 2006, Seattle, WA, USA, August 17-20, 2006, Proceedings*, volume 4144 of *Lecture Notes in Computer Science*, pages 315–328. Springer, 2006.

[68] Rajeev Alur, Swarat Chaudhuri, and P. Madhusudan. Languages of nested trees. In Thomas Ball and Robert B. Jones, editors, *Computer Aided Verification, 18th International Conference, CAV 2006, Seattle, WA, USA, August 17-20, 2006, Proceedings*, volume 4144 of *Lecture Notes in Computer Science*, pages 329–342. Springer, 2006.

[69] Viraj Kumar, P. Madhusudan, and Mahesh Viswanathan. Minimization, learning, and conformance testing of boolean programs. In Christel Baier and Holger Hermanns, editors, *CONCUR 2006 - Concurrency Theory, 17th International Conference, CONCUR 2006, Bonn, Germany, August 27-30, 2006, Proceedings*, volume 4137 of *Lecture Notes in Computer Science*, pages 203–217. Springer, 2006.

[70] Rajeev Alur and P. Madhusudan. Adding nesting structure to words. In Oscar H. Ibarra and Zhe Dang, editors, *Developments in Language Theory, 10th International Conference, DLT 2006, Santa*

Barbara, CA, USA, June 26-29, 2006, Proceedings, volume 4036 of *Lecture Notes in Computer Science*, pages 1–13. Springer, 2006.

[71] Rajeev Alur, Swarat Chaudhuri, and P. Madhusudan. A fixpoint calculus for local and global program flows. In J. Gregory Morrisett and Simon L. Peyton Jones, editors, *Proceedings of the 33rd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2006, Charleston, South Carolina, USA, January 11-13, 2006*, pages 153–165. ACM, 2006.

[72] Rajeev Alur, P. Madhusudan, and Wonhong Nam. Symbolic computational techniques for solving games. *STTT*, 7(2):118–128, 2005.

[73] Rajeev Alur, P. Madhusudan, and Wonhong Nam. Symbolic compositional verification by learning assumptions. In Kousha Etessami and Sriram K. Rajamani, editors, *Computer Aided Verification, 17th International Conference, CAV 2005, Edinburgh, Scotland, UK, July 6-10, 2005, Proceedings*, volume 3576 of *Lecture Notes in Computer Science*, pages 548–562. Springer, 2005.

[74] P. Madhusudan, P. S. Thiagarajan, and Shaofa Yang. The MSO theory of connectedly communicating processes. In Ramaswamy Ramanujam and Sandeep Sen, editors, *FSTTCS 2005: Foundations of Software Technology and Theoretical Computer Science, 25th International Conference, Hyderabad, India, December 15-18, 2005, Proceedings*, volume 3821 of *Lecture Notes in Computer Science*, pages 201–212. Springer, 2005.

[75] Rajeev Alur, Salvatore La Torre, and P. Madhusudan. Perturbed timed automata. In Manfred Morari and Lothar Thiele, editors, *Hybrid Systems: Computation and Control, 8th International Workshop, HSCC 2005, Zurich, Switzerland, March 9-11, 2005, Proceedings*, volume 3414 of *Lecture Notes in Computer Science*, pages 70–85. Springer, 2005.

[76] Rajeev Alur, Viraj Kumar, P. Madhusudan, and Mahesh Viswanathan. Congruences for visibly pushdown languages. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005, Lisbon, Portugal, July 11-15, 2005, Proceedings*, volume 3580 of *Lecture Notes in Computer Science*, pages 1102–1114. Springer, 2005.

[77] Rajeev Alur, Pavol Cerný, P. Madhusudan, and Wonhong Nam. Synthesis of interface specifications for java classes. In Jens Palsberg and Martín Abadi, editors, *Proceedings of the 32nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2005, Long Beach, California, USA, January 12-14, 2005*, pages 98–109. ACM, 2005.

[78] Rajeev Alur, Swarat Chaudhuri, Kousha Etessami, and P. Madhusudan. On-the-fly reachability and cycle detection for recursive state machines. In Nicolas Halbwachs and Lenore D. Zuck, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 11th International Conference, TACAS 2005, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2005, Edinburgh, UK, April 4-8, 2005, Proceedings*, volume 3440 of *Lecture Notes in Computer Science*, pages 61–76. Springer, 2005.

[79] Christof Löding, P. Madhusudan, and Olivier Serre. Visibly pushdown games. In Kamal Lodaya and Meena Mahajan, editors, *FSTTCS 2004: Foundations of Software Technology and Theoretical Computer Science, 24th International Conference, Chennai, India, December 16-18, 2004, Proceedings*, volume 3328 of *Lecture Notes in Computer Science*, pages 408–420. Springer, 2004.

[80] Rajeev Alur, Mikhail Bernadsky, and P. Madhusudan. Optimal reachability for weighted timed games. In Josep Díaz, Juhani Karhumäki, Arto Lepistö, and Donald Sannella, editors, *Automata, Languages*

and Programming: 31st International Colloquium, ICALP 2004, Turku, Finland, July 12-16, 2004. Proceedings, volume 3142 of *Lecture Notes in Computer Science*, pages 122–133. Springer, 2004.

[81] Rajeev Alur and P. Madhusudan. Decision problems for timed automata: A survey. In Marco Bernardo and Flavio Corradini, editors, *Formal Methods for the Design of Real-Time Systems, International School on Formal Methods for the Design of Computer, Communication and Software Systems, SFM-RT 2004, Bertinoro, Italy, September 13-18, 2004, Revised Lectures*, volume 3185 of *Lecture Notes in Computer Science*, pages 1–24. Springer, 2004.

[82] Rajeev Alur and P. Madhusudan. Visibly pushdown languages. In László Babai, editor, *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*, pages 202–211. ACM, 2004.

[83] Rajeev Alur, Kousha Etessami, and P. Madhusudan. A temporal logic of nested calls and returns. In Kurt Jensen and Andreas Podelski, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 10th International Conference, TACAS 2004, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2004, Barcelona, Spain, March 29 - April 2, 2004, Proceedings*, volume 2988 of *Lecture Notes in Computer Science*, pages 467–481. Springer, 2004.

[84] Rajeev Alur, Salvatore La Torre, and P. Madhusudan. Modular strategies for infinite games on recursive graphs. In Warren A. Hunt Jr. and Fabio Somenzi, editors, *Computer Aided Verification, 15th International Conference, CAV 2003, Boulder, CO, USA, July 8-12, 2003, Proceedings*, volume 2725 of *Lecture Notes in Computer Science*, pages 67–79. Springer, 2003.

[85] Patricia Bouyer, Deepak D'Souza, P. Madhusudan, and Antoine Petit. Timed control with partial observability. In Warren A. Hunt Jr. and Fabio Somenzi, editors, *Computer Aided Verification, 15th International Conference, CAV 2003, Boulder, CO, USA, July 8-12, 2003, Proceedings*, volume 2725 of *Lecture Notes in Computer Science*, pages 180–192. Springer, 2003.

[86] Rajeev Alur, Salvatore La Torre, and P. Madhusudan. Playing games with boxes and diamonds. In Roberto M. Amadio and Denis Lugiez, editors, *CONCUR 2003 - Concurrency Theory, 14th International Conference, Marseille, France, September 3-5, 2003, Proceedings*, volume 2761 of *Lecture Notes in Computer Science*, pages 127–141. Springer, 2003.

[87] P. Madhusudan. Model-checking trace event structures. In *18th IEEE Symposium on Logic in Computer Science (LICS 2003), 22-25 June 2003, Ottawa, Canada, Proceedings*, pages 371–380. IEEE Computer Society, 2003.

[88] Rajeev Alur, Salvatore La Torre, and P. Madhusudan. Modular strategies for recursive game graphs. In Hubert Garavel and John Hatcliff, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 9th International Conference, TACAS 2003, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2003, Warsaw, Poland, April 7-11, 2003, Proceedings*, volume 2619 of *Lecture Notes in Computer Science*, pages 363–378. Springer, 2003.

[89] P. Madhusudan and P. S. Thiagarajan. A decidable class of asynchronous distributed controllers. In Lubos Brim, Petr Jancar, Mojmír Kretínský, and Antonín Kucera, editors, *CONCUR 2002 - Concurrency Theory, 13th International Conference, Brno, Czech Republic, August 20-23, 2002, Proceedings*, volume 2421 of *Lecture Notes in Computer Science*, pages 145–160. Springer, 2002.

[90] Martin Leucker, P. Madhusudan, and Supratik Mukhopadhyay. Dynamic message sequence charts. In Manindra Agrawal and Anil Seth, editors, *FST TCS 2002: Foundations of Software Technology and*

Theoretical Computer Science, 22nd Conference Kanpur, India, December 12-14, 2002, Proceedings, volume 2556 of *Lecture Notes in Computer Science*, pages 253–264. Springer, 2002.

[91] Deepak D'Souza and P. Madhusudan. Timed control synthesis for external specifications. In Helmut Alt and Afonso Ferreira, editors, *STACS 2002, 19th Annual Symposium on Theoretical Aspects of Computer Science, Antibes - Juan les Pins, France, March 14-16, 2002, Proceedings*, volume 2285 of *Lecture Notes in Computer Science*, pages 571–582. Springer, 2002.

[92] P. Madhusudan and B. Meenakshi. Beyond message sequence graphs. In Ramesh Hariharan, Madhavan Mukund, and V. Vinay, editors, *FST TCS 2001: Foundations of Software Technology and Theoretical Computer Science, 21st Conference, Bangalore, India, December 13-15, 2001, Proceedings*, volume 2245 of *Lecture Notes in Computer Science*, pages 256–267. Springer, 2001.

[93] P. Madhusudan and P. S. Thiagarajan. Distributed controller synthesis for local specifications. In Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen, editors, *Automata, Languages and Programming, 28th International Colloquium, ICALP 2001, Crete, Greece, July 8-12, 2001, Proceedings*, volume 2076 of *Lecture Notes in Computer Science*, pages 396–407. Springer, 2001.

[94] P. Madhusudan. Reasoning about sequential and branching behaviours of message sequence graphs. In Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen, editors, *Automata, Languages and Programming, 28th International Colloquium, ICALP 2001, Crete, Greece, July 8-12, 2001, Proceedings*, volume 2076 of *Lecture Notes in Computer Science*, pages 809–820. Springer, 2001.

[95] Orna Kupferman, P. Madhusudan, P. S. Thiagarajan, and Moshe Y. Vardi. Open systems in reactive environments: Control and synthesis. In Catuscia Palamidessi, editor, *CONCUR 2000 - Concurrency Theory, 11th International Conference, University Park, PA, USA, August 22-25, 2000, Proceedings*, volume 1877 of *Lecture Notes in Computer Science*, pages 92–107. Springer, 2000.

[96] P. Madhusudan and P. S. Thiagarajan. Controllers for discrete event systems via morphisms. In Davide Sangiorgi and Robert de Simone, editors, *CONCUR '98: Concurrency Theory, 9th International Conference, Nice, France, September 8-11, 1998, Proceedings*, volume 1466 of *Lecture Notes in Computer Science*, pages 18–33. Springer, 1998.

Journal Publications

[1] Daniel Neider, P. Madhusudan, Shambwaditya Saha, Pranav Garg, and Daejun Park. A learning-based approach to synthesizing invariants for incomplete verification engines. *Journal of Automated Reasoning*, 64(7):1523–1552, 2020.

[2] Daniel Neider, Shambwaditya Saha, and P. Madhusudan. Compositional synthesis of piece-wise functions by learning classifiers. *ACM Transactions on Computational Logic*, 19(2):10:1–10:23, 2018.

[3] Pranav Garg, Christof Löding, P. Madhusudan, and Daniel Neider. Quantified data automata for linear data structures: a register automaton model with applications to learning invariants of programs manipulating arrays and lists. *Formal Methods in System Design*, 47(1):120–157, 2015.

[4] Rajeev Alur, Rastislav Bodík, Eric Dallal, Dana Fisman, Pranav Garg, Garvit Juniwal, Hadas Kress-Gazit, P. Madhusudan, Milo M. K. Martin, Mukund Raghothaman, Shambwaditya Saha, Sanjit A. Seshia, Rishabh Singh, Armando Solar-Lezama, Emina Torlak, and Abhishek Udupa. Syntax-guided synthesis. In Maximilian Irlbeck, Doron A. Peled, and Alexander Pretschner, editors, *Dependable Software Systems Engineering*, volume 40 of *NATO Science for Peace and Security Series, D: Information and Communication Security*, pages 1–25. IOS Press, 2015.

[5] Emre Uzun, Vijayalakshmi Atluri, Jaideep Vaidya, Shamik Sural, Anna Lisa Ferrara, Gennaro Parlato, and P. Madhusudan. Security analysis for temporal role based access control. *Journal of Computer Security*, 22(6):961–996, 2014.

[6] Rémi Bonnet, Rohit Chadha, P. Madhusudan, and Mahesh Viswanathan. Reachability under contextual locking. *Logical Methods in Computer Science*, 9(3), 2013.

[7] Sruthi Bandhakavi, Nandit Tiku, Wyatt Pittman, Samuel T. King, P. Madhusudan, and Marianne Winslett. Vetting browser extensions for security vulnerabilities with VEX. *Commun. ACM*, 54(9):91–99, 2011.

[8] Rajeev Alur, Swarat Chaudhuri, and P. Madhusudan. Software model checking using languages of nested trees. *ACM Trans. Program. Lang. Syst.*, 33(5):15:1–15:45, 2011.

[9] Prithvi Bisht, Parthasarathy Madhusudan, and V. N. Venkatakrishnan. CANDID: dynamic candidate evaluations for automatic prevention of SQL injection attacks. *ACM Trans. Inf. Syst. Secur.*, 13(2):14:1–14:39, 2010.

[10] Rajeev Alur and P. Madhusudan. Adding nesting structure to words. *J. ACM*, 56(3):16:1–16:43, 2009.

[11] Wonhong Nam, P. Madhusudan, and Rajeev Alur. Automatic symbolic compositional verification by learning assumptions. *Formal Methods in System Design*, 32(3):207–234, 2008.

[12] Rajeev Alur, Salvatore La Torre, and P. Madhusudan. Modular strategies for recursive game graphs. *Theor. Comput. Sci.*, 354(2):230–249, 2006.

[13] P. Madhusudan, Wonhong Nam, and Rajeev Alur. Symbolic computational techniques for solving games. *Electr. Notes Theor. Comput. Sci.*, 89(4):578–592, 2003.

[14] P. Madhusudan and P. S. Thiagarajan. Branching time controllers for discrete event systems. *Theor. Comput. Sci.*, 274(1-2):117–149, 2002.

Doctoral Students Advised or Co-advised

- Sruthi Bandhakavi, graduated 2012, worked on finding security vulnerabilities, now at Google.
- Xiaokang Qiu, graduated 2013, worked on natural proofs in program verification, now an Assistant Professor at University of Purdue.
- Francesco Sorrentino, graduated 2014, worked on finding errors in concurrent software, now working in industry.
- Pranav Garg, graduated 2015, worked on automatic invariant synthesis, now in Amazon Research.
- Edgar Pek, graduated 2015, worked on building verified systems, now at Oracle Labs.
- Shambwaditya Saha, graduated in 2020, worked on program synthesis.
- Adithya Murali, working on automated logical reasoning for program verification and in neuro-symbolic learning (current)
- Paul Krogmeier, working on learning logics (current)
- Angello Astorga, working on specification mining using learning (current)

Post-doctoral Researchers Mentored

- Gennaro Parlato, for several years on theoretical aspects of program verification, now a professor at University of Southampton, UK.
- Daniel Neider, for two to three years, on learning techniques in program verification, then a Research Group Leader at Max Plank Institute for Software Systems, Kaiserslautern, Germany, and now Professor at Carl von Ossietzky Universität Oldenburg.
- Karl Palmskog, on building verified distributed systems, then a postdoctoral researcher at UT Austin, and now Lecturer at KTH Royal Institute of Technology, Sweden.