Emina Torlak

Paul G. Allen Center for Computer Science & Engineering, Box 352350, Seattle, WA 98195
emina@cs.washington.edu • homes.cs.washington.edu/∼emina

Research Interests

Tools and programming models for computer-aided design, verification, and synthesis of software.

Education

Massachusetts Institute of Technology, Ph.D., Computer Science 2009
Thesis: *A Constraint Solver for Software Engineering*
Advisor: Daniel Jackson

Massachusetts Institute of Technology, M.Eng., Computer Science 2004
Thesis: *Subtyping in Alloy*
Advisor: Daniel Jackson

Massachusetts Institute of Technology, B.Sc., Computer Science 2003

Employment

University of Washington Seattle, WA
Assistant Professor 2014–present

University of California Berkeley Berkeley, CA
Researcher 2012–2014

LogicBlox, Inc. Atlanta, GA
Senior Computer Scientist, Compiler Technologies Group 2011

IBM Research Hawthorne, NY
Research Staff Member, Programming Technologies Department 2008–2010

Awards and Honors

ACM SIGSOFT Distinguished Paper Award for [4] 2018
NSF CAREER Award 2017
Sloan Research Fellow 2016
The AITO Dahl-Nygaard Junior Prize 2016
Best Paper Award for [16] 2016
ACM SIGSOFT Distinguished Paper Award for [26] 2012

Publications

Refereed Conference Papers

[1] Eric Butler, Emina Torlak, and Zoran Popovic. A framework for computer-aided design of educational domain models. In *Verification, Model Checking, and Abstract Interpretation (VMCAI)*. 2018.

[2] Stephen Chang, Alex Knauth, and Emina Torlak. Symbolic types for lenient symbolic execution. In *Principles of Programming Languages (POPL)*. 2018.

[3] Milod Kazerounian, Niki Vazou, Austin Bourgerie, Jeff Foster, and Emina Torlak. Refinement types for Ruby. In *Verification, Model Checking, and Abstract Interpretation (VMCAI)*. 2018.

[4] Calvin Loncaric, Michael D. Ernst, and Emina Torlak. Generalized data structure synthesis. In *International Conference on Software Engineering (ICSE)*. Distinguished Paper Award, 2018.

[5] James Bornholt and Emina Torlak. Synthesizing memory models from framework sketches and litmus tests. In *Programming Language Design and Implementation (PLDI)*. 2017.

[6] Eric Butler, Emina Torlak, and Zoran Popovic. Synthesizing interpretable strategies for solving puzzle games. In *Foundations of Digital Games (FDG)*. 2017.

[7] Jonathan Jacky, Stefani Banerian, Michael D. Ernst, Calvin Loncaric, Stuart Pernsteiner, Zachary Tatlock, and Emina Torlak. Automatic formal verification for epics. In *International Conference on Accelerator and Large Experimental Control Systems (ICALEPCS)*. 2017.

[8] Luke Nelson, Helgi Sigurbjarnarson, Kaiyuan Zhang, Dylan Johnson, James Bornholt, Emina Torlak, and Xi Wang. Hyperkernel: Push-button verification of an OS kernel. In *Operating Systems Principles (SOSP)*. 2017.

[9] Konstantin Weitz, Steven Lyubomirsky, Stefan Heule, Emina Torlak, Michael D. Ernst, and Zachary Tatlock. SpaceSearch: A library for building and verifying solver-aided tools. In *International Conference on Functional Programming (ICFP)*. 2017.

[10] James Bornholt, Antoine Kaufmann, Jialin Li, Arvind Krishnamurthy, Emina Torlak, and Xi Wang. Specifying and checking file system crash-consistency models. In *Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. 2016.

[11] James Bornholt, Emina Torlak, Dan Grossman, and Luis Ceze. Optimizing synthesis with metasketches. In *Principles of Programming Languages (POPL)*. 2016.

[12] Eric Butler, Emina Torlak, and Zoran Popovic. A framework for parameterized design of rule systems applied to algebra. In *Intelligent Tutoring Systems (ITS)*. 2016.

[13] Calvin Loncaric, Emina Torlak, and Michael D. Ernst. Fast synthesis of fast collections. In *Programming Language Design and Implementation (PLDI)*. 2016.

[14] Pavel Panchekha and Emina Torlak. Automated reasoning for web page layout. In *Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*. 2016.

[15] Stuart Pernsteiner, Calvin Loncaric, Emina Torlak, Zachary Tatlock, Xi Wang, Michael D. Ernst, and Jonathan Jacky. Investigating safety of a radiotherapy machine using system models with pluggable checkers. In *Computer-Aided Verification (CAV)*. 2016.

[16] Helgi Sigurbjarnarson, James Bornholt, Emina Torlak, and Xi Wang. Push-button verification of file systems via crash refinement. In *Operating Systems Design and Implementation (OSDI)*. Best Paper Award, 2016.

[17] Konstantin Weitz, Doug Woos, Emina Torlak, Michael D. Ernst, Arvind Krishnamurthy, and Zachary Tatlock. Scalable verification of Border Gateway Protocol configurations with an SMT solver. In *Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA)*. 2016.

[18] Shaon Barman, Rastislav Bodik, Satish Chandra, Emina Torlak, Arka Bhattacharya, and David Culler. Toward tool support for interactive synthesis. In *New Ideas, New Paradigms, and Reflections on Programming & Software (Onward!)*. 2015.

[19] Michael Ernst, Dan Grossman, Jon Jacky, Calvin Loncaric, Stuart Pernsteiner, Zachary Tatlock, Emina Torlak, and Xi Wang. Toward a dependability case language and workflow for a radiation therapy system. In *Summit on Advances in Programming Languages (SNAPL)*. 2015.

[20] John Toman, Stuart Pernsteiner, and Emina Torlak. CRust: A bounded verifier for Rust. In *Automated Software Engineering (ASE)*. 2015.

[21] Vijayaraghavan Murali, Nishant Sinha, Emina Torlak, and Satish Chandra. What gives?: A hybrid algorithm for error trace explanation. In *Verified Software: Theories, Tools and Experiments (VSTTE)*. 2014.

[22] Emina Torlak and Rastislav Bodik. A lightweight symbolic virtual machine for solver-aided host languages. In *Programming Language Design and Implementation (PLDI)*. 2014.

[23] Rajeev Alur, Rastislav Bodik, Garvit Juniwal, Milo M. K. Martin, Mukund Raghothaman, Sanjit A. Seshia, Rishabh Singh, Armando Solar-Lezama, Emina Torlak, and Abhishek Udupa. Syntax-guided synthesis. In *Formal Methods in Computer-Aided Design (FMCAD)*. 2013.

[24] Emina Torlak and Rastislav Bodik. Growing solver-aided languages with Rosette. In *New Ideas, New Paradigms, and Reflections on Programming & Software (Onward!)*. 2013.

[25] Rastislav Bodik and Emina Torlak. Synthesizing programs with constraint solvers (Invited Tutorial). In *Computer Aided Verification (CAV)*. 2012.

[26] Emina Torlak. Scalable test data generation from multidimensional models. In *Foundations of Software Engineering (FSE)*. Distinguished Paper Award, 2012.

[27] Satish Chandra, Emina Torlak, Shaon Barman, and Rastislav Bodik. Angelic debugging. In *International Conference on Software Engineering (ICSE)*. 2011.

[28] Max Schaefer, Julian Dolby, Manu Sridharan, Emina Torlak, and Frank Tip. Correct refactoring of concurrent Java code. In *European Conference on Object-Oriented Programming (ECOOP)*. 2010.

[29] Emina Torlak and Satish Chandra. Effective interprocedural resource leak detection. In *International Conference on Software Engineering (ICSE)*. 2010.

[30] Emina Torlak, Mandana Vaziri, and Julian Dolby. MemSAT: Checking axiomatic specifications of memory models. In *Programming Language Design and Implementation (PLDI)*. 2010.

[31] Emina Torlak, Felix Sheng-Ho Chang, and Daniel Jackson. Finding minimal unsatisfiable cores of declarative specifications. In *Formal Methods (FM)*. 2008.

[32] Emina Torlak and Daniel Jackson. Kodkod: A relational model finder. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*. 2007.

[33] Jonathan Edwards, Daniel Jackson, and Emina Torlak. A type system for object models. In *Foundations of Software Engineering (FSE)*. 2004.

[34] Jonathan Edwards, Daniel Jackson, Emina Torlak, and Vincent Yeung. Faster constraint solving with subtypes. In *International Symposium on Software Testing and Analysis (ISSTA)*. 2004.

Refereed Journal Articles

[35] Rajeev Alur, Rastislav Bodik, 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 *Dependable Software Systems Engineering*, volume 40 of *NATO Science for Peace and Security Series*. IOS Press, 2015.

[36] Emina Torlak, Mana Taghdiri, Greg Dennis, and Joseph Near. Applications and extensions of Alloy: Past, present, and future. (Invited Paper). *Mathematical Structures in Computer Science (MSCS)*, Special Issue on Lightweight Formal Methods, 2013.

[37] Blaise Gassend, Marten Van Dijk, Dwaine Clarke, Emina Torlak, Srinivas Devadas, and Pim Tuyls. Controlled physical random functions and applications. *ACM Transactions on Information and System Security (TISSEC)*, 10:3:1–3:22, 2008.

Refereed Workshop Papers

[38] Konstantin Weitz, Doug Woos, Emina Torlak, Michael D. Ernst, Arvind Krishnamurthy, and Zachary Tatlock. Formal semantics and automated verification for the Border Gateway Protocol. In *ACM SIGCOMM Workshop on Networking and Programming Languages (NetPL)*. 2016.

[39] James Bornholt and Emina Torlak. Scaling program synthesis by exploiting existing code. In *Machine Learning for Programming Languages Workshop (ML4PL)*. 2015.

[40] James Bornholt, Emina Torlak, Luis Ceze, and Dan Grossman. Approximate program synthesis. In *Workshop on Approximate Computing Across the Stack (WAX)*. 2015.

[41] Emina Torlak and Greg Dennis. Kodkod for Alloy users. In *Alloy Workshop*. 2006.

Technical Reports and Theses

[42] Eric Butler, Emina Torlak, and Zoran Popovic. Synthesizing optimal domain models for educational applications. Technical Report UW-CSE-17-10-02, University of Washington Computer Science & Engineering, 2017.

[43] Konstantin Weitz, Dough Woos, Emina Torlak, Michael D. Ernst, Arvind Krishnamurthy, and Zachary Tatlock. Bagpipe: Verified BGP configuration checking. Technical Report UW-CSE-16-01-01, University of Washington Computer Science & Engineering, 2016.

[44] Emina Torlak. Scalable test data generation from multidimensional models. Technical Report UCB/EECS-2012-177, University of California, Berkeley, 2012.

[45] Shaon Barman, Ras Bodik, Satish Chandra, and Emina Torlak. Discovering algorithms in angelic programs. Technical Report RC25023, IBM Research, 2010.

[46] Emina Torlak. *A Constraint Solver for Software Engineering: Finding Models and Cores of Large Relational Specifications*. Ph.D. thesis, Massachusetts Institute of Technology, 2009.

[47] Emina Torlak and Daniel Jackson. The design of a relational engine. Technical Report MIT-CSAIL-TR-2006-068, Massachusetts Institute of Technology, 2006.

[48] Emina Torlak, Marten van Dijk, Blaise Gassend, Daniel Jackson, and Srinivas Devadas. Knowledge flow analysis for security protocols. Technical Report arXiv:cs/0605109v1, arXiv.org, 2006.

[49] Marten van Dijk, Emina Torlak, Blaise Gassend, and Srinivas Devadas. A generalized two-phase analysis of knowledge flows in security protocols. Technical Report arXiv:cs/0605097v1, arXiv.org, 2006.

[50] Emina Torlak. *Subtyping in Alloy*. Master's thesis, Massachusetts Institute of Technology, 2004.

Patents

[51] Shaon K. Barman, Satish Chandra, and Emina Torlak. Precise fault localization. U.S. Patent Application No. 13/006,126, Publication No. US 2012/0185731 A1, 2012.

[52] Satish Chandra and Emina Torlak. Systems and methods for resource leak detection. U.S. Patent Application No. 12/611,561, Publication No. US 2011/0107297 A1, 2011.

[53] Julian Dolby, Max Schaefer, Manu Sridharan, Frank Tip, and Emina Torlak. Correct refactoring of concurrent software. U.S. Patent Application No. 12/718,648, Publication No. US 2011/0219361 A1, 2011.

[54] Julian Dolby, Emina Torlak, and Mandana Vaziri. System and method for debugging memory consistency models. U.S. Patent Application No. 12/615,657, Publication No. US 2011/0113285 A1, 2011.

Grants

PI, *CAREER: The Next 700 Solver-Aided Languages* 2017-2022
NSF, CCF-1651225, $498,577

Co-PI, *ARION: Taming Heterogeneity with DSLs, Approximation, and Synthesis* 2017-2020
NSF, CCF-1723352, $850,000

Co-PI, *A Picture is Worth a Billion Bits: Adaptive Visualization of Big Data* 2015-2018
DARPA, $7,500,000

Co-PI, *General-Purpose Approximate Computing Across the System Stack* 2015-2020
NSF, CCF-1518703, $2,400,000

Co-PI, *Automated Probabilistic Programming Representation and Inference Languages* 2013-2017
DARPA, $1,369,735

Co-PI, *XPS: FP: Program Synthesis for Low-Power Spatial Architectures* 2013-2016
NSF, CCF-1337415, $749,877

Key Personnel, *DSL Technology for Exascale Computing (D-TEC)* 2012-2015
DOE, DE-SC0008923, $11,605,314

Co-PI, *Computed Aided Development for Mobile Applications* 2012-2013
Samsung Electronics, $126,086

Co-PI, *Software Synthesis for High Productivity Exascale Computing* 2012-2013
DOE, DE-SC0005136, $683,344

Presentations and Seminars

Solver-Aided Programming
Eighth Summer School on Formal Techniques (SSFT), Invited Tutorial May 2018

Finding Code That Explodes Under Symbolic Evaluation
60th Meeting of the IFIP Working Group 2.3 (Providence, RI), Invited Talk May 2018

Synthesis and Verification for All
Clojure/West 2017, Keynote Mar 2017
DARPA ISAT Augmented Developers Workshop, Invited Talk Feb 2017
RacketCon 2016, Keynote Sep 2016
ECOOP 2016, Junior DN Prize Lecture Jul 2016

Optimizing Synthesis with Metasketches
57th Meeting of the IFIP Working Group 2.4 (Victoria, BC, Canada), Invited Talk Apr 2016

57th Meeting of the IFIP Working Group 2.3 (Pasadena, CA), Invited Talk — Jan 2016

Synthesis and Verification for Everyone
Cornell University, Invited Seminar — Sep 2015
European Conference on Object-Oriented Programming (ECOOP), Invited Talk — Jul 2015

Formal, Machine-Checkable Dependability Cases for End-to-End Safety Properties
56th Meeting of the IFIP Working Group 2.4 (Boppard, Germany), Invited Talk — Jul 2015
Summit on Advances in Programming Languages (SNAPL), Conference Talk — May 2015
56th Meeting of the IFIP Working Group 2.3 (Istanbul, Turkey), Invited Talk — Mar 2015

Programming for Everyone: Languages That Automate Coding, Verification, and Debugging
TTI/Vanguard, Invited Talk — Sep 2014

Programming for Everyone: From Solvers to Solver-Aided Languages and Beyond
Stanford University, Stanford Software Seminar — Jul 2014
Fourth Summer School on Formal Techniques (SSFT), Invited Talk — May 2014
University of Washington, Invited Seminar — May 2014
Samsung Electronics, Invited Seminar — Apr 2014
University of Texas at Austin, ECE, Invited Seminar — Apr 2014
University of Pennsylvania, Invited Seminar — Apr 2014
Princeton University, Invited Seminar — Mar 2014
University of Southern California, Invited Seminar — Mar 2014
Microsoft Research, Redmond, Invited Seminar — Mar 2014
University of Wisconsin-Madison, Invited Seminar — Mar 2014
Purdue University, Invited Seminar — Feb 2014
University of California, Berkeley, Invited Seminar — Feb 2014
Northeastern University, Invited Seminar — Feb 2014
IBM T. J. Watson Research Center, Invited Seminar — Jan 2014
University of California, Los Angeles, Invited Seminar — Jan 2014

A Lightweight Symbolic Virtual Machine for Solver-Aided Host Languages
Programming Language Design and Implementation (PLDI), Conference Talk — Jun 2014
54th Meeting of the IFIP Working Group 2.4 (Pacific Grove, California), Invited Talk — Feb 2014

Growing Solver-Aided Languages with Rosette
Kestrel Institute, Invited Seminar — Dec 2013
Viewpoints Research Institute, Invited Seminar — Dec 2013
New Ideas, New Paradigms, and Reflections on Programming & Software (Onward!), Conference Talk — Oct 2013
54th Meeting of the IFIP Working Group 2.3 (St. Petersburg, Russia), Invited Talk — Jun 2013
Northeastern University, Invited Seminar — Apr 2013
Dagstuhl Seminar 13061 (Fault Prediction, Localization, and Repair), Invited Talk — Feb 2013

Synthesizing Programs with Constraint Solvers

Expeditions in Computer Augmented Program Engineering (ExCAPE) Summer School, Invited Lecture Series (with Rastislav Bodik) — Jun 2013

24th International Conference on Computer Aided Verification (CAV), Invited Tutorial (with Rastislav Bodik) — Jul 2012

Programming with Constraint Solvers: Toward a Shared Infrastructure for Code Checking, Angelic Execution, Debugging, and Synthesis

Georgia Institute of Technology, Invited Seminar — Nov 2012

SRI International, Invited Seminar — Sep 2012

Workshop on Intermediate Verification Languages (Boogie), Invited Talk — Jul 2012

Scalable Test Data Generation from Multidimensional Models

20th International Symposium on the Foundations of Software Engineering (FSE), Conference Talk — Nov 2012

52nd Meeting of the IFIP Working Group 2.4 (Vadstena, Sweden), Invited Talk — May 2012

University of California, Berkeley (ParLab), Invited Seminar — Aug 2011

Kodkod by Example: Code Checking, Data Repair, Debugging and Synthesis

Dagstuhl Seminar 12152 (Software Synthesis), Invited Tutorial — Apr 2012

MemSAT: Checking Axiomatic Specifications of Memory Models

First International SAT/SMT Summer School, Invited Lecture — Jun 2011

Purdue University (Secure Software Systems), Invited Seminar — Sep 2011

Programming Language Design and Implementation (PLDI), Conference Talk — Jun 2010

University of California, Berkeley (ParLab), Invited Seminar — Oct 2009

Effective Interprocedural Resource Leak Detection

32nd International Conference on Software Engineering (ICSE), Conference Talk — May 2010

Finding Minimal Unsatisfiable Cores of Declarative Specifications

15th International Symposium on Formal Methods (FM), Conference Talk — May 2008

Kodkod: A Constraint Solver for Relational Logic

Third International Workshop on Logic and Search (LaSh), Invited Talk — Jul 2010

IBM Research (Hawthorne), Invited Seminar — Apr 2008

Microsoft Research (Redmond), Invited Seminar — Apr 2008

NASA JPL (Lab for Reliable Software), Invited Seminar — Nov 2007

13th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS), Conference Talk — Mar 2007

Professional Service and Activities

Professional Societies

Member, IFIP Working Group 2.3: Programming Methodology

Member, IFIP Working Group 2.4: Software Implementation Technology

Program Committee Organization

The Future of Alloy Workshop, Program Committee Co-Chair	2018
Onward! 2017, Program Committee Chair	2017
SPLASH Posters Track, Co-Chair	2013
IBM Programming Languages Day, Chair	2010

Program and Review Committee Membership

Object-Oriented Programming Systems Languages and Applications (OOPSLA)	2019
Programming Language Design and Implementation (PLDI)	2018
Computer-Aided Verification (CAV)	2017
Workshop on Synthesis (SYNT)	2016
New Ideas, New Paradigms, and Reflections on Programming & Software (Onward!)	2016
Visions of 2025 and Beyond (V2025) at ICSE 2016	2016
Principles of Programming Languages (POPL)	2016
Computer-Aided Verification (CAV)	2015
Runtime Verification (RV)	2015
Summit on Advances in Programming Languages (SNAPL)	2015
Domain-Specific Language Design and Implementation (DSLDI)	2014
Generative Programming: Concepts & Experiences (GPCE)	2014
Workshop on Synthesis (SYNT)	2014
Constraints in Software Testing, Verification and Analysis (CSTVA)	2014
Principles of Programming Languages Off-the-Beaten Track (POPL OBT)	2014
Programming Language Design and Implementation (PLDI)	2012
Object-Oriented Programming Systems Languages and Applications (OOPSLA)	2012
International Workshop on Logic and Search (LaSh)	2010

External Review Committee Membership

Principles of Programming Languages (POPL), External Review Committee	2015
International Conference on Software Engineering (ICSE), Review Committee	2015
Object-Oriented Programming Systems Languages and Applications (OOPSLA), External Review Committee	2014
Programming Language Design and Implementation (PLDI), External Review Committee	2013
PLDI Student Research Competition, Selection Committee	2013

Refereeing and Reviewing

ACM Transactions on Software Engineering and Methodology (TOSEM)	2005, 2012
Mathematical Structures in Computer Science (MSCS)	2011
European Symposium on Programming (ESOP)	2010
International Conference on Software Engineering (ICSE)	2010
Science of Computer Programming	2010

Students

Current

James Bornholt, PhD	2020
Eric Butler, PhD	2018

Graduated

Vimala Jampala, MS, *Game, Setty, Match: Re-Implementing Setty in Rosette and Racket* 2015

Teaching

CSE403: Software Engineering (Undergrad, University of Washington) 2015-present

CSE507: Computer-Aided Reasoning for Software (Grad, University of Washington) 2014-present

CSE599: Advanced Computer-Aided Reasoning for Software (Grad, University of Washington) 2015

CS294: Program Synthesis for Everyone (Grad, University of California, Berkeley) 2012