DANIEL J. FREMONT

Jack Baskin School of Engineering
University of California, Santa Cruz
1156 High Street, MS SOE3
Santa Cruz, CA 95064

dfremont@ucsc.edu
+1 (831) 459-3597
Office: Engineering 2, Room 321
https://people.ucsc.edu/~dfremont/

RESEARCH INTERESTS

I use automated reasoning to improve the reliability of software, hardware, and cyber-physical systems. I develop practical algorithms for system design, verification, and testing, as well as theory for the core computational problems underlying them. Among other applications, I have used my tools to synthesize robotic controllers, verify quantitative security properties of programs, and systematically test and train machine learning models for autonomous cars.

EDUCATION

Ph.D.   **University of California, Berkeley**
Logic and the Methodology of Science, August 2019

Algorithmic Improvisation
Sanjit A. Seshia (chair), Antonio Montalbán, Stuart Russell, Alistair Sinclair, David Wagner

S.B.   **Massachusetts Institute of Technology**
Mathematics and Physics, June 2013

AWARDS AND HONORS

- C.V. Ramamoorthy Distinguished Research Award, UC Berkeley EECS Dept. (2019)
- NSF Graduate Research Fellowship (2015-2018)
- Best Paper Award, IoTDI 2016 (Conf. on Internet-of-Things Design and Implementation)
- Outstanding Graduate Student Instructor Award, UC Berkeley (2015)
- Sigma Pi Sigma (physics honor society; 2013)
- Explorer Grant, Singapore-MIT Alliance for Research and Technology (2010)
- National Merit Scholarship (2009)

WORK EXPERIENCE

UC Santa Cruz, Computer Science and Engineering Department
Assistant Professor   November 2019 – date

UC Berkeley, EECS Department
Postdoctoral Scholar (advisor Sanjit A. Seshia)   Sep.-Dec. 2019
Graduate Student Researcher (advisor Sanjit A. Seshia)   June 2013 – Aug. 2019
Graduate Student Instructor (advisor David Wagner)   August-December 2014

Caltech/JPL, Laboratory for Reliable Software   March-October 2018
Engineering Graduate Student (advisor Rajeev Joshi)   May-August 2017

MIT, Department of Mathematics
Undergraduate Student Researcher (advisor Henry Cohn)   June-September 2012
Undergraduate Student Researcher (advisor Abhinav Kumar) June-September 2011

MIT, Experimental Study Group
Teaching Assistant   February-May 2010

Manifold Studios (Cambridge, MA)
Co-Founder and Lead Programmer   Sep. 2009 – June 2011

PUBLICATIONS

Refereed Conference and Journal Papers

1. ***VerifAI: A Toolkit for the Design and Analysis of Artificial Intelligence-Based Systems,***
   Tommaso Dreossi, **Daniel J. Fremont**, Shromona Ghosh, Edward Kim, Hadi Ravanbakhsh, Marcell Vazquez-Chanlatte, and Sanjit A. Seshia,
   CAV 2019 (the 31st Intl. Conf. on Computer-Aided Verification).

2. ***Scenic: A Language for Scenario Specification and Scene Generation,***
   Daniel J. Fremont, Tommaso Dreossi, Shromona Ghosh, Xiangyu Yue, Alberto L. Sangiovanni-Vincentelli, and Sanjit A. Seshia,
   PLDI 2019 (40th Conf. on Programming Language Design and Implementation).

3. ***Reactive Control Improvisation***,
   Daniel J. Fremont and Sanjit A. Seshia,
   CAV 2018 (the 30th Intl. Conf. on Computer-Aided Verification).

4. ***Maximum Model Counting***,
   Daniel J. Fremont, Markus N. Rabe, and Sanjit A. Seshia,
   AAAI 2017 (the 31st AAAI Conf. on Artificial Intelligence).

5. ***Specification Mining for Machine Improvisation with Formal Specifications***,
    Rafael Valle, Alexandre Donzé, **Daniel J. Fremont**, Ilge Akkaya, Sanjit A. Seshia, Adrian Freed, and David Wessel,
    Computers in Entertainment, Vol. 14, No. 3, 2016.

6. ***Learning and Visualizing Music Specifications Using Pattern Graphs***,
    Rafael Valle, **Daniel J. Fremont**, Ilge Akkaya, Alexandre Donzé, Adrian Freed, and Sanjit A. Seshia,
    ISMIR 2016 (the 17th Intl. Society for Music Information Retrieval Conf.).

7. ***On the Hardness of SAT with Community Structure***,
    Nathan Mull, **Daniel J. Fremont**, and Sanjit A. Seshia,
    SAT 2016 (the 19th Intl. Conf. on Theory and Applications of Satisfiability Testing).

8. ***Control Improvisation with Probabilistic Temporal Specifications***,
    Ilge Akkaya, **Daniel J. Fremont**, Rafael Valle, Alexandre Donzé, Edward A. Lee, and Sanjit A. Seshia,
    IoTDI 2016 (the 1st Intl. Conf. on Internet-of-Things Design and Implementation).
    (best paper award)

9. ***Control Improvisation***,
    Daniel J. Fremont, Alexandre Donzé, Sanjit A. Seshia, and David Wessel,
    FSTTCS 2015 (the 35th IARCS Annual Conf. on Foundations of Software Technology and Theoretical Computer Science).
    Extended version: arXiv e-print 1704.06319, 2017.

10. ***On Parallel Scalable Uniform SAT Witness Generation***,
    Supratik Chakraborty, Daniel J. Fremont, Kuldeep S. Meel, Sanjit A. Seshia, and Moshe Y. Vardi **(alphabetical order)**,
    TACAS 2015 (the 21st Intl. Conf. on Tools and Algorithms for the Construction and Analysis of Systems).

11. ***Distribution-Aware Sampling and Weighted Model Counting for SAT***,
    Supratik Chakraborty, Daniel J. Fremont, Kuldeep S. Meel, Sanjit A. Seshia, and Moshe Y. Vardi **(alphabetical order)**,
    AAAI 2014 (the 28th AAAI Conf. on Artificial Intelligence).

Refereed Workshop Papers

12. *Evaluating Sampling Techniques for Testing Cyber-Physical Systems with Machine Learning Components*,
    Edward Kim, Shromona Ghosh, Tommaso Dreossi, **Daniel J. Fremont**, Sanjit A. Seshia, and Alberto L. Sangiovanni-Vincentelli,
    MT-CPS 2019 (Monitoring and Testing of Cyber-Physical Systems).

13. *Constrained Sampling and Counting: Universal Hashing Meets SAT Solving*,
    Kuldeep S. Meel, Moshe Y. Vardi, Supratik Chakraborty, **Daniel J. Fremont**, Sanjit A. Seshia, Dror Fried, Alexander Ivrii, and Sharad Malik,
    AAAI-16 Workshop on Beyond NP, 2016.

14. *Speeding Up SMT-Based Quantitative Program Analysis*,
    Daniel J. Fremont and Sanjit A. Seshia,
    SMT 2014 (the 12th Intl. Workshop on Satisfiability Modulo Theories).

Invited Papers

15. *Formal Specification for Deep Neural Networks*,
    Sanjit A. Seshia, Ankush Desai, Tommaso Dreossi, Daniel J. Fremont, Shromona Ghosh, Edward Kim, Sumukh Shivakumar, Marcell Vazquez-Chanlatte, and Xiangyu Yue **(alphabetical order after first author)**,
    ATVA 2018 (the 16th Intl. Symposium on Automated Technology for Verification and Analysis).

Technical Reports (not overlapping the papers above)

16. *The Reachability Problem for Affine Functions on the Integers*,
    Daniel Fremont,
    arXiv e-print 1304.2639.
    Presented at **RP 12** (the 6th Intl. Workshop on Reachability Problems), 2012.

Papers Under Review

17. *Control Improvisation*,
    Daniel J. Fremont, Alexandre Donzé, and Sanjit A. Seshia,
    In submission, **JACM** (Journal of the ACM). Extends the FSTTCS 2015 paper.

TEACHING AND MENTORING

Undergraduate Research Advisees
      Johnathan Chiu (UC Berkeley, December 2018 – date)
      William Brandon (UC Berkeley, May 2018 – June 2019)
      Nathan Mull (UC Berkeley, May 2015 – August 2016)

Directed Reading Program Mentoring
      From 2014-2019 I mentored 10 undergraduate students (one each semester) under the UC Berkeley Mathematics Department's Directed Reading Program. I helped each student learn about a topic in logic or theoretical computer science, culminating in a final presentation at the end of the semester.

Teaching Assistantships
      CS 170: Efficient Algorithms and Intractable Problems (UC Berkeley, Fall 2014)
      I taught 2 discussion sections of ~30 students each, held office hours, and designed homework and exam problems for this upper division algorithms course. I also gave a guest lecture on compression (Huffman and arithmetic coding).
      For this class I won UC Berkeley's **Outstanding Graduate Student Instructor Award**.

      8.022: Electricity and Magnetism (MIT, Spring 2010)
      I held office hours and graded homework for this theoretical physics course.

Training
      GSPDP 301: Mentoring in Higher Education (UC Berkeley, Spring 2018)
      Seminar on mentoring students in research. Part of the SMART (Student Mentoring and Research Teams) program I participated in (see below).

      Workshops on Teaching (UC Berkeley, Spring 2017)
      I participated in 5 workshops by the Graduate Student Instructor Teaching & Resource Center: "Enhancing Student Participation", "How Students Learn", "Creating Inclusive Classrooms", "Syllabus and Course Design", and "Teaching Large Lecture Courses".

      CS 375: Teaching Techniques for Computer Science (UC Berkeley, Fall 2014)
      Seminar on effective teaching, including classroom observation and feedback.

      SP.232: Undergrad Teaching (MIT, Spring 2010)
      Teaching seminar focusing on discussion sections.

TALKS AND POSTERS

Conference and Workshop Talks (see paper citations above)

1. *Scenic: A Language for Scenario Specification and Scene Generation,*
   PLDI 2019, Phoenix, AZ, June 24, 2019.

2. *Reactive Control Improvisation,*
   CAV 2018, Oxford, UK, July 14, 2018.

3. *Maximum Model Counting*,
   AAAI 2017, San Francisco, CA, February 8, 2017.

4. *Control Improvisation*,
   FSTTCS 2015, Bengaluru, India, December 18, 2015.

5. *Speeding Up SMT-Based Quantitative Program Analysis*,
   SMT 2014, Vienna, Austria, July 17, 2014.

6. *The Reachability Problem for Affine Functions on the Integers*,
   RP 12, Bordeaux, France, September 19, 2012.

Invited Talks

7. *Scenic: A Language for Scenario Specification and Scene Generation,*
   3D Scene Generation Workshop, CVPR 2019, Long Beach, CA, June 16, 2019.

8. *Beyond SAT: Quantitative Problems and Randomness in Formal Verification*,
   Dealer Colloquium, PARC, Palo Alto, CA, March 15, 2017.

9. *On the Hardness of SAT with Community Structure*,
   Theoretical Foundations of SAT Solving Workshop, Fields Institute, Toronto, Canada, August 15, 2016.

Seminars and Contributed Talks

10. *VerifAI: A Toolkit for the Design and Analysis of AI-Based Systems,*
    Boeing, DARPA Assured Autonomy meeting, Everett, WA, March 12, 2019.

11. *Safe Autonomy with Algorithmic Improvisation*,
    UCLA, ECE seminar, Los Angeles, CA, April 10, 2019.
    USC Information Sciences Institute, AI seminar, Marina del Rey, CA, March 22, 2019.
    HRL Laboratories, seminar, Malibu, CA, March 21, 2019.
    University of Colorado, Boulder, ECEE seminar, Boulder, CO, February 21, 2019.
    UCSC, CSE seminar, Santa Cruz, CA, February 13, 2019.
    UC Berkeley, DREAM/CPAR seminar, Berkeley, CA, February 4, 2019.

Notre Dame, CSE seminar, Notre Dame, IN, January 23, 2019.

12. *Algorithmic Improvisation*,
    Stanford University, CS seminar, Stanford, CA, December 6, 2018.
    UMD, CS seminar, College Park, MD, November 14, 2018.
    UCSC, CS seminar, Santa Cruz, CA, October 21, 2018.
    Columbia University, CS seminar, New York, NY, September 28, 2018.
    Princeton University, CS seminar, Princeton, NJ, September 27, 2018.
    Yale University, CS seminar, New Haven, CT, September 26, 2018.
    MIT, CSAIL seminar, Cambridge, MA, September 24, 2018.
    Caltech, CMS seminar, Pasadena, CA, September 21, 2018.
    USC, CS seminar, Los Angeles, CA, September 20, 2018.
    UCLA, Joint CS & ECE seminar, Los Angeles, CA, September 18, 2018.

13. *Control Improvisation*,
    Caltech, Prof. Richard Murray's group, Pasadena, CA, August 9, 2017.

14. *Control Improvisation*,
    Rice University, Prof. Moshe Vardi's group, Houston, TX, September 14, 2016.

15. *Control Improvisation for Music, Robotics, IoT, and Beyond*,
    SRC TECHCON, Austin, TX, September 13, 2016.

16. *Control Improvisation for Music, Robotics, IoT, and Beyond*,
    Berkeley EECS Annual Research Symposium (Hot Topics @ EECS session),
    Berkeley, CA, February 11, 2016.

17. *Model Counting and Uniform Random Sampling: Theory, Algorithms and Applications*,
    TerraSwarm e-Workshop, Berkeley, CA, April 30, 2015.

Conference and Workshop Posters (not overlapping the talks above)

1. *A Probabilistic Language for Scene Generation*,
   MAPL 2018, Philadelphia, PA, June 18, 2018.

2. *Maximum Model Counting*,
   SRC TECHCON, Austin, TX, September 12, 2017.

3. *On Parallel Scalable Uniform SAT Witness Generation*,
   TACAS 2015, London, UK, April 14, 2015.

4. *Distribution-Aware Sampling and Weighted Model Counting for SAT*,
   AAAI 2014, Québec, Canada, July 29-31, 2014.

SERVICE

Member, Program Committee, CAV (Computer-Aided Verification) 2020.

Member, Artifact Evaluation Committee, TACAS (Tools and Algorithms for the Construction and Analysis of Systems) 2019.

Reviewer for:
- Algorithmica: 2018, 2019
- CAV (Computer-Aided Verification): 2016, 2018, 2019
- CP (Constraint Programming): 2017, 2018
- TACAS (Tools and Algorithms for the Construction and Analysis of Systems): 2019
- VMCAI (Verification, Model Checking, and Abstract Interpretation): 2016
- VSTTE (Verified Software: Theories, Tools, and Experiments): 2015
- HVC (Haifa Verification Conference): 2016
- EAL 60 (Edward A. Lee Festschrift Symposium): 2017

Co-author, NSF proposal *CPS: Breakthrough: Control Improvisation for Cyber-Physical Systems* (award #1646208), 2016.

Volunteer teacher at the Julia Robinson Mathematics Festival: helped K-12 students solve recreational math problems and get excited about math (Emeryville 2014; Berkeley 2017, 2018).

Volunteer teacher at MIT Splash!: designed and taught classes on computer architecture (2009) and mathematical logic and set theory (2012) to high school students.

SOFTWARE (all open-source)

Scenic (with T. Dreossi, S. Ghosh, X. Yue; https://github.com/BerkeleyLearnVerify/Scenic)
A domain-specific probabilistic programming language for defining the environments of self-driving cars and other cyber-physical systems, generating random scenes for testing and training.

VerifAI (with T. Dreossi, S. Ghosh, et al.; https://github.com/BerkeleyLearnVerify/VerifAI)
A tool for simulation-based design and analysis of AI-based systems, guided by formal specifications. Uses the *Scenic* language (above) for environment modeling.

Kontest (with R. Joshi & R. Bocchino; https://math.berkeley.edu/~dfremont/kontest.html)
A concolic testing tool for LLVM I developed during my JPL internship and used to find several bugs in the flight software of the Mars 2020 rover.

MaxCount (with M. N. Rabe; https://github.com/dfremont/maxcount)
An approximate Max#SAT solver, based on our AAAI 2017 paper.

UniGen2 (with K. S. Meel; https://bitbucket.org/kuldeepmeel/unigen)
An approximate SAT model counter and uniform generator, based on our TACAS 2015 paper.