Alex Sanchez-Stern

Curriculum Vitae

Experience

Vocational

November 2024–Present
Senior Research Scientist, dmodel, San Francisco, CA
As a senior research scientist, I've been doing interpretability research, and also adapting to fill any engineering need we have at such a small company. That's involved managing projects, acting as technical lead, mentoring other engineers, and shipping artifacts to sell to customers.

June 2025–August 2025
Part-time Lecturer, University of Washington, Seattle, WA
While working at dmodel, I again taught Systems Programming at the University of Washington. This involves the usual managing course staff, designing course curriculum and materials, and lecturing three times a week.

June 2024–August 2024
Part-time Lecturer, University of Washington, Seattle, WA
While continuing my postdoctoral research part time, I taught Systems Programming at the University of Washington. This involves the usual managing course staff, designing course curriculum and materials, and lecturing three times a week.

Sepember 2021–September 2024
Postdoctoral Researcher, University of Massachusetts, Amherst, Amherst, MA
Worked on an extension to the TacTok proof synthesis tool with co-authors at UMass Amherst and University of Illinios, Urbana Champagne. Then developed an RL-based proof synthesis with the same collaborators. Also advised a new PhD student in her studies, advised a more senior PhD student through graduation (and placed her in a job), and worked on a masters students project on localization of errors in flakey tests.

- Co-PI on a DARPA proposal VERSE.
- Co-PI on a DARPA-PEARLS proposal.
- Submitted a paper to PLDI in my first three months, on inferring helper lemmas for Coq proofs.
- Now have 4 papers published during the postdoc with 2 more in the pipeline.

9410 42nd Ave NE – Seattle, WA 98115
(206) 902 6299 • *alex.sanchezstern@gmail.com*
alexsanchezstern.com

September 2016–June 2021 **Research Assistant**, University of California, San Diego, San Diego

Continued work begun at the University of Washington on the Herbgrind project for automatically diagnosing the causes of floating-point error in large numerical software, and began work on neural synthesis of machine-checkable proofs of program correctness.

Detailed achievements:

- Worked with collaborators at UCSD to produce Proverbot9001, a tool for neural proof synthesis.
  - Implemented in Python using PyTorch and Rust
  - Can find proofs for almost a quarter of all theorem statements in CompCert (a verified C compiler).
  - Published and presented as "Generating Correctness Proofs with Neural Networks" at MAPL 2020
  - Pre-print of the paper available at `http://proverbot9001.ucsd.edu/papers/proverbot9001.pdf`
  - Talk is available as part of MAPL proceedings at `https://youtu.be/rwBbYhOAnPo?t=11540`
- Worked with Collaborators in the Systems & Security groups to produce Scooter, a tool to make data migrations safer.
- Worked with collaborators at the UW as well as Sorin Lerner at UCSD to complete work on the Herbgrind tool and paper.
  - Implemented in 20,000 lines of code (C, python, and bash scripts).
  - Analyses programs up to 50,000 lines of code.
  - Published and presented "Finding Root Causes of Floating Point Error" at PLDI 2018
  - Pre-print of the paper available at `http://herbgrind.ucsd.edu/herbgrind-pldi18.pdf`
  - Talk slides available at `http://herbgrind.ucsd.edu/pldi18-talk/`
  - Talk video available at `https://www.youtube.com/watch?time_continue=1&v=bFL6PaPrz8Y`
- Continuing maintenance of the Herbie project with collaborators at the UW.

December 2013–September 2016 **Research Assistant**, University of Washington, Seattle

Worked with another research assistant to develop the Herbie system for automatically improving the accuracy of floating point code

Detailed achievements:

- Worked with Pavel Panchekha and Zachary Tatlock in developing the high level design of the system over the course of two years.
- Worked closely with Pavel Panchekha to write the implementation of the system, including specifically:
  - Independently developing the algebraic simplification system
  - Writing the top level code which controls the various subsystems
  - Developed the experimental loop variant of Herbie to continue the work described in the paper.
- Authored a paper on our work together with Pavel Panchekha, Zachary Tatlock, and James Wilcox.
  - Our paper was published at the Programming Languages Design and Implementation 2015 conference.
  - Paper and talk available at `http://herbie.uwplse.org/pldi15.html`
- Authored a second paper with Pavel Panchekha, Zachary Tatlock, Chen Qiu, and international collaborators Nasrine Damouche and Matthieu Martel on a new format and benchmark suite for cross-tool floating point benchmarks.
- Began work on a third project, Herbgrind, which I continued at UCSD

June 2013–September 2013 **College Tech**, Seattle Schools District, Seattle
Maintained existing educational and teacher machines, and set up and installed new machines, at a variety of schools in the Seattle Schools District.

September 2011–January 2013 **Assistant Operations Engineer**, Casa Latina, Seattle
Wrote tests and data aggregation and display code for the Machete job registration system, under James Carter.

July 2011–September 2011 **Intern**, Bensussen Deutsch & Associates, Inc, Woodinville
Performed market research, handled product returns, and managed product testing.

Publications

June 2024 **Rango: Adaptive Retrieval-Augmented Proving for Automated Software Verification**, *ICSE 2025*

June 2024 **QEDCartographer: Automating Formal Verification Using Reward-Free Reinforcement Learning**, *ICSE 2025*

December 2023 **Passport: Improving Automated Formal Verification Using Identifiers**, *TOPLAS 2023*

May 2023 **Proofster: Automated Formal Verification**, *ICSE 2023 Demo Track*

December 2022 **Data Driven Lemma Synthesis for Interactive Proofs**, *OOPSLA2022*

June 2021 **Scooter & Sidecar: a domain-specific approach to writing secure migrations**, *PLDI 2021*

June 2020 **Generating Correctness Proofs with Neural Networks**, *MAPL 2020*

January 2020 **REPLica: REPL Instrumentation for Coq Analysis**, *CPP 2020*

June 2018 **Finding Root Causes of Floating Point Error**, *PLDI 2018*

July 2016 **Towards a Standard Benchmark Format and Suite for Floating-Point Analysis**, *NSV 2016*

June 2015 **Automatically Improving Accuracy for Floating Point Expressions**, *PLDI 2015*, Distinguished Paper Award

Awards

2023 Panelist at the NeurIPS Machine Learning For Theorem Proving Tutorial

2015 Distinguished Paper – PLDI 2015

2015 Marygates Research Scholarship

Service

2024 PLDI PC

2023 OOPSLA ERC

2021 AIPLANS Committee

2020-2021 ACM Mentorship Program Mentor

2019-2020 ICFP Artifact Evaluation Committee

9410 42nd Ave NE – Seattle, WA 98115
(206) 902 6299 • *alex.sanchezstern@gmail.com*
alexsanchezstern.com

2018-2019 POPL Student Volunteer Captain

Education

2018–2021 **Doctor of Philosophy, Computer Science**, *University of California, San Diego*

2016–2018 **Candidate of Philosophy, Computer Science**, *University of California, San Diego*

2015–2016 **Masters of Science, Computer Science**, *The University of Washington*
Honors

2012–2015 **Bachelors of Science, Computer Science**, *The University of Washington*
Honors

PhD Thesis

Title *Hybrid-Neural Synthesis of Machine-Checkable Software Correctness Proofs*

Supervisor Professor Sorin Lerner

Description The correctness of large software artifacts has important impact on many aspects of the modern world. Machine-checkable software correctness proofs provide a guarantee that a piece of software adheres to some logical specification, however producing such proofs is labor-intensive, taking in some cases 23 person-years of highly skilled labor to prove properties of 10,000 line programs. This thesis work uses a hybrid-approach of machine learning and proof assistant search procedures to produce proofs of correctness for a large variety of software automatically or semi-automatically.

Masters Thesis

Title *Dynamic Analysis of Floating Point Errors with Herbgrind*

Supervisor Professor Zachary Tatlock

Description Numerical computation using floating point numbers is notoriously difficult to reason about, even in idealized environments. This thesis presents the development of a tool which can analyze the runtime behavior of programs written in a variety of environments and languages, and extract inaccurate floating point computation for improvement.

Bachelors Thesis

Title *Algebraic Simplification for the Herbie Project*

Supervisor Professor Zachary Tatlock

Description The ability to simplify arbitrary mathematical expressions is extremely useful in many applications, including the Herbie numerical synthesis tool, but is exponential in general. This thesis presents a set of data structures and heuristics that allow thousands of expressions to be simplified every second.

9410 42nd Ave NE – Seattle, WA 98115
(206) 902 6299 • *alex.sanchezstern@gmail.com*
alexsanchezstern.com