

YIHONG ZHANG | CV

Website: <https://effect.systems>
E-mail: yz489@cs.washington.edu

GitHub: @yihozhang
Twitter: @yihongz_bot

Research Interests

E-graphs and Equality Saturation. Staging and Abstraction without Regrets. Program Synthesis. PL for ML.

Research Experience

- Practical Deforestation for Haskell Programs, advised by Lionel Parreaux. *June 2021—Current.*
- Faster and Worst-Case Optimal E-Matching, advised by Zachary Tatlock. *Jan 2021—Current.*
- A Genetic Algorithm for Quality Counterfactual Explanations, advised by Dan Suciu. *Aug 2020—Mar 2021.*

Education Background

Master - University of Washington

Sept 2021-June 2022

- M.S. in Computer Science

Bachelor - University of Washington

Sept 2018-June 2021

- B.S. with Departmental Honors in Computer Science, *Magna Cum Laude*. GPA 3.90/4.00.
- Thesis: *Faster and Worst-Case Optimal E-Matching*, advised by Zachary Tatlock.
- Selected Coursework:

Graduate:

- Principles of Programming Languages
- Principles of Database Systems
- Computer-Aided Reasoning for Software
- Software Engineering
- Database Internals
- Artificial Intelligence
- Deep Processing Techniques for NLP

Undergrad:

- Advanced PL & Verification
- Design and Implementation of DSLs
- Toolkits for Modern Algorithms
- Distributed Systems
- Operating Systems
- Advanced Logic
- Axiomatic Set Theory

Publications

- **Yihong Zhang**, Remy Wang, Max Willsey, Zachary Tatlock. *Relational E-matching*. In submission.
- **Yihong Zhang**. *Faster and Worst-Case Optimal E-matching via Reduction to Conjunctive Queries*. Programming Language Design and Implementation (PLDI) 2021 Student Research Competition. **Awarded 1st place at the undergraduate division.**
- Maximilian Schleich, Zixuan Geng, **Yihong Zhang**, and Dan Suciu. *GeCo: Quality Counterfactual Explanations in Real Time*. PVLDB, 14(9): 1681-1693, 2021. doi:10.14778/3461535.3461555.

Employment

Research Assistant - Hong Kong University of Science and Technology

June 2021-Sept 2021

- Working with Lionel Parreaux on the optimization of functional programs.

Teaching Assistant - University of Washington

Sept 2020-June 2021

- Responsibility includes grading student's assignments, hosting office hours and teaching sections, and designing homework.

Software Engineer Intern - Oracle NetSuite

June 2020-Sept 2021

- Worked with the team on several features and issues for an enterprise-resource planning platform.

Honors

- 1st Place, PLDI Student Research Competition (Undergraduate Division). June 2021.
- 1st Place, International Collegiate Programming Contest (ICPC), UW Qualifier. Oct 2019.
- 5th Place, International Collegiate Programming Contest (ICPC), Pacific Northwest Region. Nov. 2018.
- Dean's List, University of Washington. Dec 2018—June 2020.

Course Projects

- Sdl: A Staged Datalog Compiler using Lightweight Modular Staging (LMS)
 - Class project for CSE 544 Principles of Database Systems.
 - In this project, I build a staged Datalog compiler using Lightweight Modular Staging (LMS). Experiments show that it achieves up to 10x speedup compared to Souffle Datalog tool.
- MaxDuet: Combining Statistical Top-down Deductions and Bottom-up Enumerations for Programming by Example
 - Class project for CSE 573 Artificial Intelligence.
 - In this project, I add support for enumerative search to MaxFlash for PBE solving. The enumerated programs are used to directly solve the synthesis problem and guide the witness functions during probabilistic deductions. This is an attempt to unify enumerative search, deductive search, and stochastic search into a framework for program synthesis. Experiments on the SyGuS benchmarks shows mixed (aka negative) results.
- Cornelius: Killing Equivalent and Redundant Mutants with E-graphs
 - Class project for CSE 503 Software Engineering with Ben Kushigian, Ishan Chatterjee, and Gabrielle Strandquist.
 - We propose to utilize E-graph to eliminate equivalent and redundant mutants for mutation testing. This addresses the phase-ordering problem faced by Trivial Compiler Equivalence (TCE), a known technique for eliminating equivalent mutants. Experiments on a pure Java subset show that it discovers much more equivalent and redundant mutants than TCE in less time.
- Sager: A Demonic Graph Synthesizer for Worst-Case Performance
 - CSE 507 Computer-Aided Reasoning for Software with Mike He.
 - We propose to use symbolic evaluations to synthesize input instances that determine the worst-case complexity of a given algorithm. We build a prototype implementation with Rosette and show that it makes Shortest-Path Faster Algorithm (SPFA) and its several variants asymptotically worse.

Programming for Fun

- Hatafun: Embedding the type system of Datafun (ICFP 2016) in Haskell.
- egraph-sql: A minimal implementation of e-graphs as relational databases.