# William T. Hallahan

AKW 211, 51 Prospect Street, New Haven, CT 06511
+1 (603) 809-0314 • whallahan@binghamton.edu

## Research Interests

Program Verification, Synthesis, Symbolic Execution, Networks

## Education

**Yale University** **New Haven, CT**
*Computer Science, PhD* *2015–2022*
Thesis: Automated Approaches for Program Verification and Repair
Advisor: Ruzica Piskac

**College of the Holy Cross** **Worcester, MA**
*Bachelor of Arts in Mathematics, Computer Science (Double Major)* *2011–2015*
Thesis: Stability of the coefficients in the Kronecker product of a hook and a rectangle
Thesis Advisor: Cristina Ballantine

### Work Experience

**Assistant Professor** **Binghamton, NY**
*Binghamton University* *August 2022 - Present*

**Software Engineering and Research Intern** **Portland, OR**
*Galois* *June 2018 - August 2018*

**Software Engineering and Research Intern** **Santa Clara, CA**
*Barefoot Networks* *June 2017 - August 2017*

## Research

### Publications

W. Hallahan, R. Jhala, R. Piskac. **Counterexample-Guided Inference of Modular Specifications.** Under submission.

J. Kolesar, R. Piskac, W. Hallahan. **Checking Equivalence in a Non-Strict Language.** *OOPSLA*, 2022 (To Appear).

W. Hallahan, E. Zhai, R. Piskac. **Automated Analysis and Repair By Example for Firewalls. (Journal Version)** *FMSD*, 2020.

E. Campbell, W. Hallahan, P. Srikumar, C. Cascone, J. Liu, V. Ramamurthy, H. Hojjat, R. Piskac, R. Soulé, N. Foster. **Avenir: Managing Data Plane Diversity with Control Plane Synthesis.** *NSDI*, 2021.

K. Morton, W. Hallahan, E. Shum, R. Piskac, M. Santolucito. **Grammar Filtering For Syntax-Guided Synthesis.** *AAAI*, 2020.

W. Hallahan, A. Xue, R. Piskac. **G2Q: Haskell Constraint Solving.** *Haskell Symposium*, 2019.

W. Hallahan, A. Xue, M. Bland, R. Jhala, R. Piskac. **Lazy Counterfactual Symbolic Execution.** *PLDI*, 2019.

W.Hallahan, M. Santolucito, R. Piskac. **Live Programming by Example.** *CHI Demonstrations*,

2019.
J. Liu, W.Hallahan, C. Schlesinger, M. Sharif, J. Lee, R.Soulé, H. Wang, C. Caşcaval, N. McKeown, N.Foster. **p4v: Practical Verification for Programmable Data Planes.** *SIGCOMM*, 2018.
W. Hallahan, E. Zhai, R. Piskac. **Automated Analysis and Repair By Example for Firewalls.** *FMCAD*, 2017.
C. Ballantine, W. Hallahan. **Stability of coefficients in the Kronecker product of a hook and a rectangle.** *Journal of Physics A: Mathematical and Theoretical, Vol. 49 (5)*, 2015.

## Talks

**Counterexample Guided Inference of Modular Specifications**
*Simons Institute* *March 2021*

**Grammar Filtering For Syntax-Guided Synthesis**
*LiVe* *March 2021*

**Lazy Symbolic Execution: Writing, Debugging, and Repairing Programs**
*Cornell University* *July 2020*

**Grammar Filtering For Syntax-Guided Synthesis**
*AAAI* *February 2020*

**Data-driven Specification Synthesis for Modular Verification**
*Programming Languages Day, IBM T.J. Watson Research Center* *December 2019*

**G2Q: Haskell Constraint Solving**
*Haskell Symposium* *August 2019*

**Lazy Counterfactual Symbolic Execution**
*PLDI* *July 2019*

**Lazy Symbolic Execution: Counterfactual Examples and Haskell Constraint Solving**
*Microsoft Research Cambridge* *June 2019*

**Lazy Symbolic Execution: Counterfactual Examples and Haskell Constraint Solving**
*Imperial College London* *June 2019*

**Lazy Symbolic Execution: Counterfactual Examples and Haskell Constraint Solving**
*DiffBlue* *June 2019*

**Lazy Counterfactual Symbolic Execution**
*IBM Programming Languages Day, IBM T.J. Watson Research Center* *December 2018*

**Automated Analysis and Repair By Example for Firewalls**
*FMCAD* *October 2017*

**Automated Firewall Repair via Example-Based Synthesis**
*IBM Programming Languages Day, IBM T.J. Watson Research Center* *December 2016*

**Stability of the coefficients in the Kronecker product of a hook and a rectangle**
*College of the Holy Cross* *April 2015*

## Poster Presentations

**Building a Symbolic Execution Engine for Haskell**
*FMCAD* *October 2017*

**Automated Firewall Repair via Example-Based Synthesis**
*FMCAD* *October 2016*

**On the Kronecker Product of a Hook and a Box**
*JMM* *January 2015*

## Teaching

Advising Student Projects

**Higher Order Functions in Symbolic Execution** **Yale University**
*Raffa Levy* *Fall 2021*

**State Merging in Lazy Symbolic Execution** **Yale University**
*Rushyendra Maganty* *Summer 2019-Spring 2020*

**An online interface for live programming by example** **Yale University**
*Nathan Nuñez* *Summer 2019*

**Automated Code Repair** **Yale University**
*Max Levatich* *Summer 2019*

**Grammar Filtering For Syntax-Guided Synthesis** **Yale University**
*Kairo Morton, Elven Shum* *Summer 2019*

**Live Programming Interface** **Yale University**
*Griffin Solot-Kehl* *Spring 2019*

**Software Therapy Applications for Autism Spectrum Disorder** **Yale University**
*Stephanie Hickman* *Fall 2018*

**Synthesizing SDNs as Functional Reactive Programs** **Yale University**
*Vivek Gopalan* *Summer 2018*

**Symbolic Execution of Haskell Programs** **Yale University**
*Anton Xue* *Fall 2017-Spring 2019*

Advising Directed Reading

**Functional Programming Techniques** **Yale University**
*Laurence Lu* *Fall 2020*

Teaching Assistant

**Software Analysis and Verification** **Yale University**
*Taught by Ruzica Piskac* *Spring 2022*

**Software Analysis and Verification** **Yale University**
*Taught by Ruzica Piskac* *Fall 2020*

**Software Engineering** **Yale University**
*Taught by Ruzica Piskac* *Spring 2019*

**Software Analysis and Verification** **Yale University**
*Taught by Ruzica Piskac* *Fall 2018*

**Software Engineering** **Yale University**
*Taught by Ruzica Piskac* *Spring 2018*

**Software Analysis and Verification** **Yale University**
*Taught by Ruzica Piskac* *Fall 2017*

**Principles of Operating Systems** — **Yale University**
*Taught by Avi Silberschatz* — *Spring 2017*

**Introduction to Systems Programming & Computer Organization** — **Yale University**
*Taught by Stanley C. Eisenstat* — *Fall 2016*

Led Tutorial Session

**Algebraic Structures** — **College of the Holy Cross**
*Taught by Cristina Ballantine* — *Spring 2015*

## Service

**Program Committee**
*Haskell Symposium* — *2022*

**Artifact Evaluation Committee**
*VMCAI* — *2022*

**Web Chair**
*FMCAD* — *2021*

**Artifact Evaluation Committee**
*POPL* — *2021*

**Artifact Evaluation Committee**
*CAV* — *2019*

## Patents

John Nathan Foster, Jeongkeun Lee, Robert Soulé, William Hallahan, Cole Nathan Schlesinger, Han Wang, Steffen Julif Smolka. **Data Plane Program Verification**. US Patent Application 16/022,601, June 2018.

## Technical Skills

Haskell, OCaml, Python, C, C++, SMT-LIB, P4