Ian Briggs

PhD Candidate in Computer Science
(801) 574-0929 ibriggs@cs.utah.edu

My research uses modern verification and synthesis tools to write high-performance, accurate numerical code. I have 7 years of experience working with floating point and automated reasoning.

EDUCATION

University of Utah—Ph. D.
Computer Science
August 2020 - Present

University of Utah—B.S/M.S.
Computer Science
August 2010 - May 2016

EXPERIENCE

University of Utah, Salt Lake City, UT — Research Associate
2016 - Present

Amazon Automated Reasoning, Seattle, WA — Summer Intern
2022

LLNL , Livermore, CA — Outside Collaborator
2018 - 2019

PUBLICATIONS

Choosing Mathematical Function Implementations for Speed and Accuracy — PLDI
Choosing the best in class combination of transcendental function calls for a given computation.

Keeping Science on Keel When Software Moves — CACM
Root causing result variability caused by compiler optimization.

FailAmp: Relativization Transformation for Soft Error Detection in Structured Address Generation — TACO
Detection of single event errors in address calculation in a fully automatic LLVM pass.

Scalable Yet Rigorous Floating-Point Error Analysis— SC
Improving on the state of the art to allow analysis of large floating point expressions.

Rigorous Estimation of Floating-Point Round-Off Errors with Symbolic Taylor Expansions — TOPLAS
The tightest rigorous bound for floating point error estimation.

Multi-Level Analysis of Compiler-Induced Variability and Performance Tradeoffs — HPDC
A tool to explore the effects compiler optimizations have on performance and accuracy.

FLiT: Cross-platform floating-point result-consistency tester and workload — IISWC
Examining how compiler optimizations affect computation results using a purpose built test suite.

Rigorous floating-point mixed-precision tuning. — POPL
Lowering the precision of computations to improve speed while retaining a rigorous error bound.

OTHER PUBLICATIONS

Synthesizing Mathematical Identities with E-Graphs — CoRR

FPDetect: Efficient Reasoning About Stencil Programs Using Selective Direct Evaluation. — TACO

Moving the Needle on Rigorous Floating-Point Precision Tuning. — NFM

ArcherGear: Data Race Equivalencing got Expeditious HPC Debugging — PPoPP

DiffTrace: Efficient Whole-Program Trace Analysis and Diffing for Debugging — CLUSTER