Benjamin F Jones

Contact

Email benjaminfjones@gmail.com

Github http://github.com/benjaminfjones

Webpage http://bfj7.com

Education

2002-2007 **PhD, Mathematics**; University of Notre Dame (Notre Dame, IN)

Thesis title: On the Singular Chern Classes of Schubert Varieties Via Small Resolution

1997-2002 **BSc, Mathematics**; University of Utah (Salt Lake City, UT)

Graduated Cum Laude, Minor: Physics

Experience

Software Engineer, Groq Inc.: (2017 – Present)

Compiler development for a novel tensor stream processor: compiler backend development in Haskell, compiler optimization, QoR optimization, interface between hardware and software team.

Research Engineer, Galois Inc.: (2012 – 2017)

Haskell development, DSL and language development, automated theorem proving (SMT solving, model checking, and custom decision procedures), interactive theorem proving (Coq).

Assistant Professor, University of Wisconsin, Stout: (2010 – 2012)

Research in representation theory and algebraic geometry, teaching freshman honors calculus, upper level undergraduate courses in algebra, and senior level courses in programming languages.

Technical Experience

Projects

BLT: **[Github]** A novel decision procedure for integer linear programming that outperforms traditional branch and bound solvers on certain classes of problems. This work was published at the 2015 SMT Workshop **[full text]**.

LIMA: **[Github]** A domain specific language for implementing and modeling fault-tolerant distributed systems. This is joint work with Lee Pike as part of NASA contract NNL14AA08.

Selected Talks & Papers

Language for Unified Verification and Implementation for Distributed Avionics. Journal of Aerospace Information Systems, 2018.

Modular Model-Checking of a Byzantine Fault-Tolerant Protocol. NASA Formal Methods, 2017.

Bounded Integer Linear Constraint Solving via Lattice Search. 13th International Workshop on Satisfiability Modulo Theories, 2015.

See http://bfj7.com for more talks and papers.

Programming Languages

Haskell: 8 years experience in both large projects (>200k SLOC) and small; DSL, parser, compiler, and interpreter design; extensive use of property-based and unit testing; familiarity with the foreign function interface and mainstream debugging and profiling tools.

Python: 4 years experience doing open source work and computational mathematics, using numpy, cython, matplotlib, contributions to **SageMath**.

C/C++: Used on and off in mostly small scale projects; For example, the BLT project described above is a C++ library with a set of high-level Haskell bindings.

---

benjaminfjones@gmail.com • http://bfj7.com • 706 296 5614