

---

## Summary

- Ten years of experience with designing domain-specific languages and building optimizing compilers that help programmers more easily write high-performance software.
- Worked on a variety of production-grade compilers, including the **XLA TPU compiler** and **IBM XL C/C++/Fortran**.
- Was a primary contributor to the **TACO tensor algebra compiler**, which generates fast code for performing sparse tensor algebra computations given high-level specifications of the desired computation and input/output data structures.
- Multiple publications, including a distinguished paper award, at top programming language and compiler conferences such as **OOPSLA**, **PLDI**, and **CGO**.
- Working experience with the **LLVM** and **MLIR** compiler frameworks.
- Proficient in **C++** and **C** and have working experience with **Python**, **JavaScript**, **MATLAB**, **Java**, **SQL**, and **Bash**.

---

## Education

- 01/2018–08/2022 **PhD, Computer Science**, *Massachusetts Institute of Technology*.  
Dissertation: Format Abstractions for the Compilation of Sparse Tensor Algebra.  
Advisor: Prof. Saman Amarasinghe.
- 09/2015–01/2018 **SM, Computer Science**, *Massachusetts Institute of Technology*.  
Thesis: Unified Sparse Formats for Tensor Algebra Compilers.  
Advisor: Prof. Saman Amarasinghe.
- 09/2010–04/2015 **BASc, Computer Engineering**, *University of Waterloo*.  
Cumulative GPA: 94%. (Graduated on Dean's Honours List with distinction.)

---

## Industry and Research Experience

- 05/2024–Present **Senior Software Engineer**, *Google*, Sunnyvale, CA.
- 10/2022–04/2024 **Software Engineer**, *Google*, Sunnyvale, CA.
- Contribute to the XLA TPU compiler with a focus on building out and maintaining the MLIR-based middle end and LLVM-based back end for SparseCores (specialized processors in TPUs that are designed to accelerate embedding-based ML workloads).
  - Lead bring-ups of the SparseCore compiler stack for new generations of TPUs, with responsibility for ensuring the compiler supports and utilizes new hardware features.
- 09/2015–08/2022 **Research Assistant**, *Massachusetts Institute of Technology*, Cambridge, MA.
- Generalized the TACO tensor algebra compiler theory to support efficiently computing with sparse tensors that may be stored in a wide variety of data structures. This helps speed up real-world sparse tensor algebra computations by letting users store their data in specialized data structures that are optimized for the data.
  - Developed a more user-friendly compiler front end for Simit, a programming language for computing on sparse systems using linear algebra. This new front end was later also adapted for use in the compiler for GraphIt, a graph programming language.
  - Developed a web interface for TACO (hosted at <http://tensor-compiler.org/codegen>).
- 06/2018–08/2018 **Research Intern**, *Microsoft*, Redmond, WA.
- Designed and implemented a prototype framework that can improve the performance of deep learning models in TensorFlow by automatically applying user-defined, high-level architectural optimizations. Worked with Drs. Minjia Zhang and Yuxiong He.

- 01/2015–08/2015 **Research Assistant**, *University of Waterloo*, Waterloo, ON, Canada.
- Prototyped mathematical (SMT) models that can be used to automatically synthesize designs for capillary electrophoresis devices given user-specified requirements and constraints. Worked with Prof. Derek Rayside.
- 09/2014–12/2014 **Compiler Optimization Developer Co-op**, *IBM*, Markham, ON, Canada.
- 01/2014–04/2014 ◦ Implemented enhancements to the IBM XL C/C++/Fortran compiler’s automatic simdization feature, such as adding the ability to vectorize type conversion operations (which yielded up to ~30% performance gains on some SPEC benchmarks).
- Investigated and fixed defects in other parts of the XL compiler’s high-level optimizer.
- 05/2013–08/2013 **Software Engineering Intern**, *OptumSoft*, Menlo Park, CA.
- Designed and implemented a GraphLab-inspired distributed graph processing framework prototype, using the TACC programming language developed by OptumSoft.
- 08/2012–12/2012 **Software Developer Co-op**, *Sybase*, Waterloo, ON, Canada.
- 01/2012–04/2012 ◦ Developed tools and investigated techniques for improving the performance and optimality of Sybase SQL Anywhere’s join optimizer. Worked with Dr. Anisoara Nica.
- 05/2011–08/2011 ◦ Developed a replacement system for aggregating Sybase product crash reports and usage statistics. System was implemented entirely in Sybase SQL.

---

## Publications

### Conference Publications

- OOPSLA 2022 Stephen Chou and Saman Amarasinghe. Compilation of Dynamic Sparse Tensor Algebra. In *Proceedings of the ACM on Programming Languages*, Volume 6, Issue OOPSLA2, 2022.
- CGO 2022 Daniel Donenfeld, Stephen Chou, Saman Amarasinghe. Unified Compilation for Lossless Compression and Sparse Computing. In *Proceedings of the 2022 IEEE/ACM International Symposium on Code Generation and Optimization*, 2022.
- OOPSLA 2021 Rawn Henry, Olivia Hsu, Rohan Yadav, Stephen Chou, Kunle Olukotun, Saman Amarasinghe, Fredrik Kjolstad. Compilation of Sparse Array Programming Models. In *Proceedings of the ACM on Programming Languages*, Volume 5, Issue OOPSLA, 2021.
- OOPSLA 2020 Ryan Senanayake, Changwon Hong, Ziheng Wang, Amalee Wilson, Stephen Chou, Shoaib Kamil, Saman Amarasinghe, Fredrik Kjolstad. A Sparse Iteration Space Transformation Framework for Sparse Tensor Algebra. In *Proceedings of the ACM on Programming Languages*, Volume 4, Issue OOPSLA, 2020.
- PLDI 2020 Stephen Chou, Fredrik Kjolstad, Saman Amarasinghe. Automatic Generation of Efficient Sparse Tensor Format Conversion Routines. In *Proceedings of the 41st ACM SIGPLAN International Conference on Programming Language Design and Implementation*, 2020.
- OOPSLA 2018 Stephen Chou, Fredrik Kjolstad, Saman Amarasinghe. Format Abstraction for Sparse Tensor Algebra Compilers. In *Proceedings of the ACM on Programming Languages*, Volume 2, Issue OOPSLA, 2018.
- OOPSLA 2017 Fredrik Kjolstad, Shoaib Kamil, Stephen Chou, David Lugato, Saman Amarasinghe. The Tensor Algebra Compiler. In *Proceedings of the ACM on Programming Languages*, Volume 1, Issue OOPSLA, 2017. [Distinguished Paper Award]

### Peer-Reviewed Short/Workshop Publications

- SPAA 2020 Suzanne Mueller, Peter Ahrens, Stephen Chou, Fredrik Kjolstad, Saman Amarasinghe. Sparse Tensor Transpositions. In *Proceedings of the 32nd ACM Symposium on Parallelism in Algorithms and Architectures (brief announcement)*, 2020.

- Chocs Avancées David Lugato, Fredrik Kjolstad, Stephen Chou, Saman Amarasinghe, Shoaib Kamil. Taco: compilation et génération de code d'expressions tensorielles. In *Chocs Avancées*, No. 12, 2018.
- ASE 2017 Fredrik Kjolstad, Stephen Chou, David Lugato, Shoaib Kamil, Saman Amarasinghe. taco: A Tool to Generate Tensor Algebra Kernels. In *Proceedings of the 32nd IEEE/ACM International Conference on Automated Software Engineering (tools paper and demo)*, 2017.
- DBTest 2013 Anisoara Nica and Stephen Chou. Using Similarity Distance for Performance Prediction of the Query Optimization Process. In *Proceedings of the Sixth International Workshop on Testing Database Systems*, 2013.

---

## Patents

- 01/2013 **Resource Estimation For A Query Optimization Process.**  
Anisoara Nica and Stephen Chou.  
Patent number: US 9,298,771 B2.

---

## Teaching Experience

- 10/2021 **Workshop Instructor**, *Boston College*, Chestnut Hill, MA.  
  - Prepared and delivered a two-hour Introduction to Unix Shell workshop, designed for undergraduate computer science and engineering students, that covered the basics of Bash and Vim.
- 09/2020–12/2020 **Teaching Assistant**, *Massachusetts Institute of Technology*, Cambridge, MA.  
  - Handled logistics associated with running the Dynamic Computer Language Engineering course (6.818), which had 30+ students. This included keeping course website up to date, answering student questions on Piazza, and preparing and grading mini-quizzes that were given after every lecture.
  - Prepared and graded semester-long course project, which required students to implement an optimized virtual machine for a dynamic programming language.
  - Prepared and delivered nine recitations over the semester that were designed to aid students with their project. Also held weekly office hours to assist students one-on-one.

---

## Awards & Honors

- 10/2017 **Distinguished Paper Award**, *OOPSLA 2017*.
- 06/2015 **Albert Sherwood Barber Medal for Best Overall Work Term and Academic Performance**, *University of Waterloo*.
- 02/2015 **First in Class Engineering Scholarship for 4A term**, *University of Waterloo*.
- 06/2014 **First in Class Engineering Scholarship for 3B term**, *University of Waterloo*.
- 10/2013 **First in Class Engineering Scholarship for 3A term**, *University of Waterloo*.
- 07/2012 **iAnywhere Solutions Inc. Scholarship**, *University of Waterloo*.

---

## Invited Talks

### Format Abstractions for Compilation of Sparse Tensor Algebra

- 07/2021 Google Compiler ML Reading Group
- 07/2020 MIT Fast Code Seminar
- 02/2019 ADA Liaison Meeting Talk
- 01/2019 Invited Workshop on Compiler Techniques for Sparse Tensor Algebra, MIT

---

## Professional Service

Journal Review **IEEE TC:** Transactions on Computers (2019)

Conference **MLSys:** Conference on Machine Learning and Systems (2022)

Review **Euro-Par:** European Conference on Parallel and Distributed Computing (2020)

**SPAA:** Symposium on Parallelism in Algorithms and Architectures (2019, 2021)

**PLDI:** Programming Language Design and Implementation (2019)

**CGO:** Symposium on Code Generation and Optimization (2017, 2018, 2022)

Committees **CGO 2020** Artifact Evaluation Committee