Willow M. Ahrens

PH.D. COMPUTER SCIENCE AT CSAIL, MIT · COMPILERS, DATA STRUCTURES, ALGORITHMS

Office G740, 32 Vassar St, Cambridge, MA 02139

☎ (+1) 505-412-5239 | ✉ willow@csail.mit.edu | 🏠 willowahrens.io | 💾 willow-ahrens | 🎓 0QFbKQ4AAAAJ | 🆔 0000-0002-4963-0869

Education

Massachusetts Institute of Technology
Cambridge, MA

PH.D. COMPUTER SCIENCE, GPA: 4.9 / 5.0, ADVISOR: SAMAN AMARASINGHE
Sep. 2016 - Sep 2024

- Collaborated on intermediate languages, cost models, compiler passes, and algorithms for state-of-the-art research projects.
- Published 6 papers in top-tier conferences and journals, including PLDI, CGO, TOMS, and IPDPS.
- Presented at 15+ conferences, workshops, and research groups in academia and industry.
- Advised 4 undergraduates and 3 masters students. Proposed projects and provided weekly feedback. One student published in SPAA.
- Developed Finch.jl programming language and compiler for sparse and structured arrays.
- Discovered compiler algorithms to automatically adapt programs to input properties.

University of California, Berkeley
Berkeley, CA

BS IN EECS, MINOR IN MATH, GPA: 3.8 / 4.0, ADVISOR: JAMES DEMMEL
Sep. 2012 - May 2016

- Berkeley Benchmarking and Optimization Group
- Developed a reproducible linear algebra library in C, ReproBLAS.

Selected Publications

Finch: Sparse and Structured Array Programming with Control Flow
arXiv:2404.16730[cs]

W. AHRENS, T. F. COLLIN, R. PATEL, K. DEEDS, C. HONG, AND S. AMARASINGHE.
Apr. 2024

- Built the Finch array programming language and compiler.
- Finch is the first compiler to automatically specialize flexible control flow to diverse array data structures.
- Supports a familiar programming language of loops, statements, ifs, breaks, etc., over a wide variety of array structures, such as sparsity, run-length-encoding, symmetry, triangles, padding, or blocks. Finch reliably utilizes the key properties of structure, such as structural zeros, repeated values, or clustered non-zeros.
- The first compiler to support convolution over sparse arrays, as well as worst-case optimal joins and variable-width blocks.
- Demonstrated dramatic speedups in operations such as SpMV and SpGEMM, image processing, and graph analytics.

Looplets: A Language for Structured Coiteration.
CGO 2023

W. AHRENS, D. DONENFELD, F. KJOLSTAD, AND S. AMARASINGHE.
Feb. 2023

- Published in Proceedings of the 21st ACM/IEEE International Symposium on Code Generation and Optimization, in CGO 2023.
- Proposed the Looplets language, the key IR behind the Finch tensor compiler. Allows flexible iteration over a wide variety of sparse and structured arrays.

Autoscheduling For Sparse Tensor Algebra With An Asymptotic Cost Model.
PLDI 2022

W. AHRENS, F. KJOLSTAD, AND S. AMARASINGHE.
Jun. 2022

- Published in Proceedings of the 43rd ACM SIGPLAN International Conference on Programming Language Design and Implementation.
- Discovered an asymptotic notation for the runtime of sparse tensor programs.
- Built the first asymptotic autoscheduler for sparse tensor compilers.

Algorithms for Efficient Reproducible Floating Point Summation.
ACM Trans. Math. Softw.

W. AHRENS, J. DEMMEL, AND H. D. NGUYEN.
Jul. 2020

- Published in ACM Transactions on Mathematical Software, vol. 46, no. 3, p. 22:1-22:49, Jul. 2020.
- Discovered algorithms for bitwise identical sums of floating point numbers under reassociation, using only 6 words of memory.
- Implemented ReproBLAS library to implement performant versions of these algorithms for common linear algebra routines.

Tensor Algebra Compilation with Workspaces.
CGO 2019

F. KJOLSTAD, W. AHRENS, S. KAMIL, AND S. AMARASINGHE.
2019

- Published in 2019 IEEE/ACM International Symposium on Code Generation and Optimization (CGO), 2019, pp. 180-192.

A Fill Estimation Algorithm for Sparse Matrices and Tensors in Blocked Formats.
IPDPS 2018

W. AHRENS, H. XU, AND N. SCHIEFER.
2018

- Published in 2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2018, pp. 546-556.

Teaching

MIT Course 6.1200 (Mathematics For Computer Science)
Boston, MA

TEACHING ASSISTANT
Sep 2022 - Dec 2022

- Taught 6.1200 (formally 6.042), a proof-based course designed to teach the fundamentals of algorithmic thinking in computer science, with attention given to concepts such as induction, asymptotic analysis, graphs, and probability.
- Led two discussion sections with 30 students each, covering example problems and their solutions.
- With 2 other TAs, staffed the last in-person office hours before homework was due each week, with attendance regularly exceeding 40 students requesting individual attention.

MIT Glass Lab
Cambridge, MA

GLASSBLOWING INSTRUCTOR
Feb 2019 – Present

- Supervised pairs of beginner students one at a time for weekly two-hour sessions.
- Ensured student safety in their first experiences with handling 2400 °F glass in a crowded hot shop.
- Explained critical techniques in glassblowing, including gathering, marvering, blocking, and blowing.

Center for Access to Engineering Excellence
Berkeley, CA

TUTOR
Jan 2014 – May 2014

- Tutored groups of around 2-5 students at a time in lower-division Computer Science, Math, and Physics courses.

Mentorship Experience

Radha Patel
MASTERS STUDENT
2023-2024

- Investigated symmetric tensor processing in the Finch Programming Language.
- Implemented new symmetric tensor datastructures.
- Discovered new program transformations to optimize symmetric tensor kernels.

Cecilia Chen
UNDERGRADUATE STUDENT
2023

- Investigated graph processing in the Finch Programming Language.
- Implemented several kernels from the GAP benchmark suite in Finch. (Link)
- Discovered new program transformations to optimize symmetric tensor kernels.

Alexandra Dima
MASTERS STUDENT
2021-2023

- Built GSTACO graph processing framework, unifying tensor and graph interfaces.
- Compiled graph operations to Finch.
- Thesis: A. Dima, "GSTACO: A Generalized Sparse Tensor Algebra Compiler," Thesis, Massachusetts Institute of Technology, 2023. (Link)

Emily Lu
UNDERGRADUATE STUDENT
2023

- Built basic tensor user interface functions (e.g. indexing, slicing, etc.) by compiling to Finch.

Yiming Chen
UNDERGRADUATE STUDENT
2022

- Investigated distributional properties of recursive Kronecker sparse matrix generators.

Suzy Mueller
MASTERS STUDENT
2020

- Discovered algorithms for more efficient sparse tensor transpositions.
- Publication: S. Mueller, W. Ahrens, S. Chou, F. Kjolstad, and S. Amarasinghe, "Brief Announcement: Sparse Tensor Transpositions," in Proceedings of the 32nd ACM Symposium on Parallelism in Algorithms and Architectures (SPAA), 2020, pp. 559–561. (Link)

Tony Wang
UNDERGRADUATE STUDENT
2019

- Used term rewriting to recognize important kernels (such as matrix multiply) in Swizzles.jl array combinator language.

Honors

2017-2021	**CSGF Fellow**, DOE Computational Science Graduate Fellow	*Washington, D.C.*
2016	**Warren Y. Dere Design Award**, UC Berkeley	*Berkeley, CA*

Service

2025	**Program Committee**, PLDI 2025: 46th ACM SIGPLAN Conference on Programming Language Design and Implementation	*Seoul, South Korea*
2025	**Program Committee**, CC25: ACM SIGPLAN 2025 International Conference on Compiler Construction (CC'25) co-located with CGO, PPoPP and HPCA	*Las Vegas, NV*
2022-2024	**Organizer**, Sparse Roofline Benchmark Working Group, a multi-university collaboration to standardize benchmarking of sparse linear algebra	*Virtual*
2022-2024	**Editor**, GraphBLAS BinSparse Binary Sparse File Format Standards Committee	*Virtual*
2023	**Program Committee**, DRAGSTERS (Distributions, Relational Algebra, Graphs, Semi-Rings, Tensors, and All That) workshop at PLDI conference	*Orlando, FL*
2022	**Reviewer**, Parallel Computing	*N/A*
2020	**Reviewer**, IEEE Transactions on Computers	*N/A*
2020	**Reviewer**, IEEE Transactions on Computers	*N/A*
2019	**Reviewer**, IEEE Transactions on Computers	*N/A*
2021	**Reviewer**, IEEE Transactions on Parallel and Distributed Systems	*N/A*
2020	**Reviewer**, IEEE Transactions on Parallel and Distributed Systems	*N/A*
2019	**Reviewer**, IEEE Transactions on Parallel and Distributed Systems	*N/A*

Experience

2022 NSF I-Corps Fall Cohort #2 - South Regional Node Program — *Virtual*
ENTREPRENEURIAL LEAD — *Aug. 2022 - Nov. 2022*
- Interviewed 100 potential customers to validate the market for tensor algebra compilers.

Sandia National Laboratory — *Albuquerque, NM*
CSGF PRACTICUM INTERN, SUPERVISOR: ERIK BOMAN — *May 2019 – Aug 2019*
- Discovered algorithms to reorganize sparse matrix nonzeros into dense blocks. Proposed the 1D-VBR sparse matrix format. Julia.

MIT Julia Lab — *Cambridge, MA*
RESEARCH ASSISTANT — *Sep 2016 – May 2019*
- Worked with Professor Alan Edelman to develop Julia abstractions for scientific computing.
- Developed prototype Julia implementation for the CLIMA earth and atmosphere modeling project.
- Developed array programming infrastructure based on Julia base array interface.

Los Alamos National Laboratory — *Los Alamos, NM*
RESEARCH INTERN, SUPERVISOR: HAI AH NAM — *May 2016 – Aug 2016*
- Parallelized a coupled cluster doubles nuclear quantum physics simulation to run on Wolf cluster. C++/MPI.

NVIDIA — *Santa Clara, CA*
SOFTWARE ENGINEERING INTERN — *June 2014 – Aug 2014*
- Worked in a team to create a CPU profiler, intercepting dll calls and sampling using signal handlers. Created a small real-time system to handle stack traces and process them into various types of call graphs.
- Fixed bugs, conducted testing, and wrote a test for cuda-gdb.

Presentations

2024	**Panelist**, Working Toward an Interface for Sparse BLAS, Supercomputing 2024	*Atlanta, GA*
2024	**"binsparse: A Binary Sparse Tensor File Format"**, HPC Graph Toolkits and the GraphBLAS Forum, Supercomputing 2024	*Atlanta, GA*
2024	**"Sparse and Structured Array Programming"**, BU Principles of Programming and Verification (POPV), BU	*Boston, MA*
2024	**"Sparse and Structured Array Programming"**, Google	*Virtual*
2024	**"Finch: Sparse and Structured Array Programming"**, Ph.D. Defense, MIT	*Cambridge, MA*
2024	**"Finch: Sparse and Structured Array Programming"**, Sammaniversary, MIT	*Cambridge, MA*
2024	**"Compiling Control Flow in Sparse and Structured Array Programs"**, Sparse Workshop, PLDI	*Copenhagen, Denmark*
2024	**"Compilers for Sparse and Structured Array Programming"**, International Workshop on Efficient Generative AI, University of Edinburgh	*Edinburgh, UK*
2024	**"Finch: A Sparse and Structured Array Compiler in Julia"**, Cambridge Area Julia Users Network	*Cambridge, MA*
2024	**"Supporting the Sparse BLAS: File Formats and Benchmarks"**, Sparse BLAS Workshop 2, University of Tennessee	*Knoxville, TN*
2023	**"Sparse Compilers, Sparse Benchmarks"**, Sparse BLAS Workshop, University of Tennessee	*Knoxville, TN*
2023	**"Exploring the Design Space of Sparsity Through Compilers"**, The Sparse Rooflines Group	*Virtual*
2023	**"Exploring the Design Space of Sparsity Through Compilers"**, RelationalAI Virtual Talk	*Virtual*
2023	**"Finch: A Compiler for Sparse and Structured Data"**, Stanford University	*Stanford, CA*
2023	**"Finch: A Compiler for Sparse and Structured Data"**, Lawrence Berkeley National Lab	*Berkeley, CA*
2023	**"Finch: A Compiler for Sparse and Structured Data"**, University of Washington PLSE Group	*Seattle, WA*
2022	**"An Asymptotic Cost Model for Autoscheduling Sparse Tensor Programs"**, ADA Symposium	*Ann Arbor, Michigan*
2021	**"Contiguous Partitioning: Registers, Caches, and Distributed Memories"**, DOE CSGF Review	*Washington, D.C.*
2021	**"On Optimal Partitioning for Variable Block Row Format"**, MIT CRIBB Seminar	*Cambridge, MA*
2018	**"The Tensor Algebra Compiler (taco)"**, CSAIL Alliances Annual Meeting	*Cambridge, MA*
2018	**"For-Loops 2.0: Index Notation And The Future Of Array Compilers"**, JuliaCon 2018	*London, UK*

Complete Publications

Interface for Sparse Linear Algebra Operations. *arXiv:2411.13259 [cs]*

A. Abdelfattah, W. Ahrens, H. Anzt, et al. *Nov. 2024*

- Published in arXiv:2411.13259 [cs].

Galley: Modern Query Optimization for Sparse Tensor Programs. *arXiv:2408.14706 [cs]*

K. Deeds, W. Ahrens, M. Balazinska, and D. Suciu. *2024*

- Published on arXiv:2408.14706 [cs].

SySTeC: A Symmetric Sparse Tensor Compiler *arXiv:2406.09266 [cs]*

R. Patel, W. Ahrens, and S. Amarasinghe. *Jun. 2024*

- Published in arXiv:2406.09266 [cs].

Finch: Sparse and Structured Tensor Programming with Control Flow *arXiv:2404.16730[cs]*

W. Ahrens, T. F. Collin, R. Patel, K. Deeds, C. Hong, and S. Amarasinghe. *Apr. 2024*

- Published in arXiv:2404.16730 [cs].

Mechanised Hypersafety Proofs about Structured Data *PLDI 2024*

V. Gladshtein, Q. Zhao, W. Ahrens, S. Amarasinghe, and I. Sergey. *Jun. 2024*

- Published in Proceedings of the ACM on Programming Languages, Volume 8, Issue PLDI, p. 173:647-173:670. Jun. 2024

Binary Sparse Format Specification Version 0.1 *Online*

B. Brock, T. Davis, J. Kitchen, E. Welch, I. Virshup, W. Ahrens *Nov. 2023*

- Ongoing Specification: `https://github.com/GraphBLAS/binsparse-specification`

The Continuous Tensor Abstraction: Where Indices are Real *arXiv:2407.01742 [cs]*

J. Won, W. Ahrens, J. Emer, and S. Amarasinghe. *Jul. 2024*

- Published in arXiv:2407.01742 [cs].

Looplets: A Language for Structured Coiteration. *CGO 2023*

W. Ahrens, D. Donenfeld, F. Kjolstad, and S. Amarasinghe. *Feb. 2023*

- Published in Proceedings of the 21st ACM/IEEE International Symposium on Code Generation and Optimization, in CGO 2023.

Autoscheduling For Sparse Tensor Algebra With An Asymptotic Cost Model.   *PLDI 2022*

W. Ahrens, F. Kjolstad, and S. Amarasinghe.   *Jun. 2022*

- Published in Proceedings of the 43rd ACM SIGPLAN International Conference on Programming Language Design and Implementation.

Contiguous Graph Partitioning For Optimal Total Or Bottleneck Communication.   *arXiv:2007.16192 [cs]*

W. Ahrens.   *Jun. 2021*

- Published in arXiv:2007.16192 [cs].

Algorithms for Efficient Reproducible Floating Point Summation.   *ACM Trans. Math. Softw.*

W. Ahrens, J. Demmel, and H. D. Nguyen.   *Jul. 2020*

- Published in ACM Transactions on Mathematical Software, vol. 46, no. 3, p. 22:1-22:49, Jul. 2020.

Brief Announcement: Sparse Tensor Transpositions.   *SPAA 2020*

S. Mueller, W. Ahrens, S. Chou, F. Kjolstad, and S. Amarasinghe.   *2020*

- Published in Proceedings of the 32nd ACM Symposium on Parallelism in Algorithms and Architectures (SPAA), 2020, pp. 559-561.

On Optimal Partitioning For Sparse Matrices In Variable Block Row Format.   *arXiv:2005.12414 [cs]*

W. Ahrens and E. G. Boman.   *May 2020*

- Published in arXiv:2005.12414 [cs].

A Parallel Fill Estimation Algorithm for Sparse Matrices and Tensors in Blocked Formats.   *MIT Thesis*

W. Ahrens.   *2019*

- Thesis submitted to the Massachusetts Institute of Technology in 2019.

Tensor Algebra Compilation with Workspaces.   *CGO 2019*

F. Kjolstad, W. Ahrens, S. Kamil, and S. Amarasinghe.   *2019*

- Published in 2019 IEEE/ACM International Symposium on Code Generation and Optimization (CGO), 2019, pp. 180-192.

LATE Ain't Earley: A Faster Parallel Earley Parser.   *arXiv:1807.05642 [cs]*

W. Ahrens, J. Feser, and R. Hui.   *Jul. 2018*

- Published in arXiv:1807.05642 [cs].

A Fill Estimation Algorithm for Sparse Matrices and Tensors in Blocked Formats.   *IPDPS 2018*

W. Ahrens, H. Xu, and N. Schiefer.   *2018*

- Published in 2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2018, pp. 546-556.

Parallel Compact Hash Algorithms for Computational Meshes.   *SIAM J. Sci. Comput.*

R. Tumblin, W. Ahrens, S. Hartse, and R. Robey.   *Jan. 2015*

- Published in SIAM Journal on Scientific Computing, vol. 37, no. 1, pp. C31-C53, Jan. 2015.