DANIEL DELAYO

ddelayo@cs.stonybrook.edu ⋄ danieldelayo.github.io ⋄ orcid.org/0000-0001-7636-0107

EDUCATION

Stony Brook University
Fourth Year PhD Candidate — *May 2021 - Present*
under Michael A. Bender

B.S. in Computer Science — *August 2017 - May 2021*
College of Engineering and Applied Science — Honors College, Honors CS, Summa Cum Laude, 3.95

PUBLICATIONS

Exploring the Landscape of Distributed Graph Sketching (ALENEX 2025)
Single-machine streaming graph processing systems are typically bottlenecked by CPU performance, and distributed graph processing systems are typically bottlenecked by network latency. We present Landscape, a distributed graph-stream processing system that uses linear sketching to distribute the CPU work of computing graph properties to distributed workers with no need for worker-to-worker communication. As a result, it overcomes the CPU and network bottlenecks that limit other systems. DOI: 10.1137/1.9781611978339.11

GraphZeppelin: How to Find Connected Components (Even When Graphs Are Dense, Dynamic, and Massive) (TODS 2024)
The task of computing connected components becomes more difficult when graphs are very large, or when they are dynamic, meaning the edge set changes over time subject to a stream of edge insertions and deletions. GraphZeppelin uses new linear sketching data structures (CubeSketch) to solve the streaming connected components problem and requires space asymptotically smaller than the space required for a lossless representation of the graph. GraphZeppelin is optimized for massive dense graphs: GraphZeppelin can process millions of edge updates (both insertions and deletions) per second, even when the underlying graph is far too large to fit in available RAM. DOI: 10.1145/3643846

IncrementandFreeze: Every Cache, Everywhere, All of the Time (SPAA 2023)
We design and implement a parallel and cache-efficient algorithm for generating exact LRU cache hit-rate curves that runs many times faster than prior work. Exact LRU hit-rate curves can be used to optimize the cost to cache content. Reducing this cost can increase the availability and accessibility of content. DOI: 10.1145/3558481.3591085

Automatic Management of High-Bandwidth Memory (SPAA 2022)
High-Bandwidth Memory is an emerging type of memory that does not fit in the standard memory hierarchy; standard caching models do not apply. We present a caching model for High-Bandwidth Memory and verify it on Intel Knight's Landing processors.

We introduce Dynamic Priority, which gives a makespan constant-competitive with optimal and outperforms standard schemes. DOI: 10.1145/3490148.3538570

Write-Optimized IP Indexer for Cyber Security (Cluster Computing 2022)
We designed expiration strategies that allow our system, Diventi, to support IP range queries in external memory with millisecond response times. Diventi automatically expires the oldest IP data without compromising query latency or ingestion rate.

This work was in collaboration with Sandia National Labs. DOI: 10.1007/s10586-021-03463-5

SUBMITTED PUBLICATIONS

Don't Melt Your Cache: Low-Associativity with Heat-Sink
We show that low-associativity implementations of LRU are often *actually not* constant-competitive

algorithms. On the other hand, we give randomized eviction algorithms that *are constant-competitive*, and even a d-associative algorithm that, using any $d = \omega(1)$, and using $1 + o(1)$ resource augmentation, is $1 + o(1)$ competitive with the fully-associative LRU algorithm. Combined, our algorithms suggest a new way of thinking about the design of low-associativity caches, in which one intentionally designs randomized mechanisms that allow parts of the cache which are "overheating" to naturally cool down.

CUPCaKE: Fast and Compact Dynamic Connectivity
To date, dynamic connectivity implementations either achieve low update and query time, or asymptotically low space complexity on massive dense graphs, but not both. We show how to have our cake and eat it too – that is, we present a dynamic connectivity algorithm and a corresponding implementation that processes updates quickly, answers queries with low latency, and uses sublinear space for massive graphs.

WORKS IN PROGRESS

Cilk Prace: Fast and efficient parallel race detector
This work aims to create a provably good determinacy race detector that takes advantage of Cilk Reducers to make the work general and maintainable. Modern race detectors either do not certify a program is race free, have high space overhead, force the program to execute serially, or sacrifice maintainability for performance. Our work aims to create a race detection tool that has none of these flaws.

Cilk Increment And Freeze: LRU Cache hitrate instrumentation at program runtime
We turn Increment-and-Freeze into a cilktool to enable generation of LRU hitrate curves with minimal space and time overhead from real program executions as they happen. We use this tool to study address space partitioning based sampling and real cache curves of various programs.

Parallel Paging: Rethinking practical parallel computation
Recent work proposes that the integral of RAM allocation over time, or memory impact, is a contentious resource for parallel program execution. We explore the use of memory-impact based allocation and scheduling decisions to minimize a program's memory impact and therefore maximize a machine's utilization. We experiment with hit-rate curves, blind algorithms, and machine learning to guide these decisions.

TECHNICAL STRENGTHS

Computer Science	Algorithms, Data Structures, External-memory algorithms, Parallel algorithms, High-Performance Computing, LLVM
Parallel Programs	OpenMP, MPI, OpenCilk
Programming Languages	C++, C, Python; Familiar with Java, Prolog, SML

TEACHING EXPERIENCE

Teaching Assistant for Intro to Computer Science	Fall 2018
Teaching Assistant for Algorithms	Spring 2021
Teaching Assistant for Graduate System Security	Spring 2023
Substitute Lecture for Algorithms	Fall 2025

WORK EXPERIENCE

Stony Brook Computer Science Research Assistant Summer 2019 - Present

· Advised by Michael A. Bender on Theory, Memory Architecture, Data Structures and Algorithms.

Sandia National Labs Intern Summer 2021 - Summer 2024

· Collaborated with Sandians on the parallel and cache efficient theory and implementation of Write-Optimized Data Structures to solve cyber security challenges.

Stony Brook Physics Research Assistant Spring 2018 - Fall 2018

· Advised by Krishna Kumar on High-Energy Physics Simulations for the PREX-II and MOLLER experiments in collaboration with Jefferson Lab.

HONORS, AWARDS, AND SCHOLARSHIPS

NSF student travel grant for SPAA	Funding to attend SPAA 2024
CLASC Invited Student	Poster Presentation and funding for CLASC 2023
GAANN Fellowship	1 Year of PhD Funding (Fall 2022 and Spring 2023)
Graduated Summa Cum Laude	May 2021
Honors College	August 2017 - May 2021
Honors Computer Science	August 2017 - May 2021
SBU Presidential Scholarship	Tuition and Fees, August 2017 - May 2021