Yen-Hsiang Chang

☎ (217)898-6189 | ✉ yenhsiangc@berkeley.edu | 🏠 yen-hsiang-chang.github.io | yen-hsiang-chang

Research Interests

Parallel Programming and Algorithms

My research interests lie in the general area of high-performance computing, particularly in parallel programming and algorithms, with the focus on graph algorithms and applied numerical linear algebra.

Education

University of California at Berkeley

Aug. 2023 - Now

Doctor of Philosophy

Berkeley, California

- Major: Electrical Engineering and Computer Sciences

University of Illinois at Urbana-Champaign (UIUC)

Aug. 2018 - May. 2022

Bachelor of Science in Grainger Engineering

Champaign, Illinois

- Major: Computer Engineering, Minor: Mathematics
- Cumulative GPA: 3.99/4.00, Major GPA: 4.00/4.00, Minor GPA: 4.00/4.00
- Graduated with Highest Honors, on completion of an undergraduate thesis of superior quality

Research Experiences

Graduate Researcher, instructed by Prof. James Demmel & Dr. Aydın Buluç *Aug. 2023 - Now*

BeBOP and PASSION lab, Berkeley

- Researched on extracting parallelism from the approximate minimum degree algorithm, which reduces fill-ins in Cholesky factorization, by relaxing the minimum degree and independence criteria.
- Developed a parallel randomized minimum cut algorithm in shared memory using parallel tree contractions and parallel batched updates and queries.

Undergraduate Researcher, instructed by Prof. Wen-mei Hwu, Prof. Rakesh Nagi & Prof. Jinjun Xiong *May. 2021 - May. 2022*

Coordinated Science Laboratory, UIUC

- Researched on graph mining and implemented local k-clique counting kernels on GPUs.
- Researched on maximal clique enumeration, with the focus on implementing variants of Bron-Kerbosch algorithm on GPUs.
- Designed efficient parallel maximal clique enumeration kernels for multi-GPUs, with the characteristics of mitigating load imbalance using a worker list and reducing memory footprint by splitting complicated sets into monotonic sets that can be stored using compact representations.
- Researched on generalizing the worker list technique to mitigate load imbalance on GPUs for other domains.
- Published the paper **”Parallelizing Maximal Clique Enumeration on GPUs”** in **PACT’23**

Undergraduate Researcher, instructed by Prof. Wen-mei Hwu & Prof. Jinjun Xiong *Jun. 2019 - May. 2022*

IBM-Illinois Center for Cognitive Computing Systems Research (C3SR)

- Researched on MLModelScope, an HW/SW agnostic, extensible, and customizable platform for evaluating and profiling ML models across datasets/frameworks/hardware, and within AI application pipelines.
- Developed MLModelScope Agents in different frameworks, primarily in PyTorch and ONNX Runtime.
- Published the paper **”MLHarness: A Scalable Benchmarking System for MLCommons”** in **BENCH’21**.

Publications

Parallelizing Maximal Clique Enumeration on GPUs | Link | Code

Mohammad Almasri*, Yen-Hsiang Chang*, Izzat El Hajj, Rakesh Nagi, Jinjun Xiong, and Wen-mei Hwu *Oct. 2023*

(*Equal contribution)

Published in 32nd International Conference on Parallel Architectures and Compilation Techniques (PACT’23) *Vienna, Austria*

- Parallelized the Bron-Kerbosch algorithm for single-GPU and multi-GPUs, with a geometric mean speedup of 4.9× (up to 16.7×) on single GPU and scaled efficiently to multiple GPUs.
- Proposed to parallelize maximal clique enumeration on GPUs by performing depth-first traversal of independent sub-trees in parallel, instead of performing breadth-first traversal to avoid explosion in the number of tree nodes at deep levels.
- Proposed a worker list for dynamic load balancing, as well as partial induced subgraphs and a compact representation of excluded vertex sets to regulate memory consumption.

MLHarness: A Scalable Benchmarking System for MLCommons | Link | Code

Yen-Hsiang Chang, Jianhao Pu, Wen-mei Hwu, and Jinjun Xiong *Nov. 2021*

Published in 2021 BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench’21) *Virtual*

- Proposed MLHarness, a scalable benchmarking harness system for MLCommons.
- MLHarness codifies the standard benchmark process as defined by MLCommons including models, datasets, DL frameworks, and software and hardware systems.
- MLHarness provides an easy and declarative approach for model developers to contribute their models and datasets to MLCommons.
- MLHarness includes the support of a wide range of models with varying inputs/outputs modalities so that it can scalably benchmark these models across different datasets, frameworks, and hardware systems.

Honors & Awards

International

2022	**17th Place**, 2022 Google Hash Code World Finals
2021	**11th Place**, 44th Annual World Finals of the International Collegiate Programming Contest
2020	**6th Place**, Microsoft Q# Coding Contest – Summer 2020
2020	**Round 4 Qualifier (top 110)**, 2020 Topcoder Open Algorithm Competition
2019	**112th Place**, 2019 Google Code Jam Round 3

Domestic

2021	**ECE Alumni Association Scholarship**, Outstanding scholastic record in ECE Department, UIUC
2020	**10th Place**, 2020 ICPC North America Championship
2020	**Midwest Champion**, 2020 ICPC North America Championship
2020	**2nd Place**, 2020 ICPC North America Championship Cyber Challenge
2018-22	**Dean's List**, Grainger College of Engineering, UIUC

Selected Projects

Convex Relaxations for Sparse Matrix Reordering

Aug. 2024 - Dec. 2024

for EE227BT (Convex Optimization) at Berkeley

- Investigated the opportunity of solving sparse matrix reordering problems using convex relaxations.
- Showed that convex relaxations achieve reordering quality comparable to that from heuristic and spectral algorithms, but the execution time is too high to make it practical.

Parallel Randomized Minimum Cuts and Parallel Tree Contractions

Jan. 2024 - Apr. 2024

for CS267 (Applications of Parallel Computers) at Berkeley

- Implemented a parallel randomized minimum cut algorithm in shared memory using parallel tree contractions and parallel batched updates and queries.
- The implementation is scalable but not competitive against the state-of-the-art deterministic parallel solver due to the inherent huge constants in the parallel data structures used.

Randomized SVD for Serverless Systems

Aug. 2023 - Dec. 2023

for CS262A (Advanced Topics in Computer Systems) at Berkeley

- Integrated a newly proposed serverless message interface, FaaS Message Interface, into a loosely coupled randomized SVD algorithm.
- Demonstrated that high performance linear algebra kernels can be executed in the serverless setting with comparable performance and significantly better accessibility when compared to supercomputers.

On the Hardness of Approximate Nearest Neighbor Search

Aug. 2023 - Dec. 2023

for MATH221 (Matrix Computations / Numerical Linear Algebra) at Berkeley

- Investigated the hardness of approximate nearest neighbor search by analyzing condition numbers of intermediate and final results.
- Examined a special case of approximate nearest neighbor search where the query is guaranteed to be close to a database point, and showed that typical solutions to approximate nearest neighbor search using dimensionality reduction can be simplified.

Improvements to the Hungarian LAP Solver on GPU

Aug. 2021 - Dec. 2021

for ECE508 (Manycore Parallel Algorithms) at UIUC

- Compared two state-of-the-art GPU-accelerated Hungarian LAP solvers of classical and alternating tree variants of the algorithm.
- Optimized CUDA kernels based on the bottlenecks found from profiling tools, including NVIDIA Nsight Systems.

GPU Convolution Kernel Optimizations

Aug. 2020 - Dec. 2020

for ECE408 (Applied Parallel Programming) at UIUC

- Designed and developed an optimized neural-network convolutional layer with tensor cores.
- Analyzed and fine-tuned CUDA kernels through the use of profiling tools, including NVIDIA Nsight Compute.

Skills

Languages	C/C++, Python
Libraries/Tools	CUDA, OpenMP, MPI
Other	Git, Docker, LaTeX