Mason DiCicco

Worcester Polytechnic Institute, UH320
Worcester, MA 01609
(860) 882 7015
mtdicicco@wpi.edu

Education

2020-2025 **Ph.D. Computer Science**, *Worcester Polytechnic Institute*, 4.0 GPA.
Advised by Daniel Reichman

2016-2020 **B.S. Computer Science**, *University of Connecticut*, 3.9 GPA.
Concentration: Theory and Algorithms

2016-2020 **B.A. Mathematics**, *University of Connecticut*, 3.9 GPA.
Concentration: Pure Mathematics

Coursework

Graduate Level: Combinatorics, Analysis, Algebra, AI, Reinforcement Learning, Deep Learning, Networks.

Mathematics: Abstract/Linear Algebra, Differential Equations, Analysis, Topology, Probability.

Computer Science: Systems Programming, Computer Architecture, Algorithms and Complexity, Machine Learning, Computational Geometry, Operating Systems, Modern Cryptography, Numerical Methods.

Research Experience

September 2020 - Present **Graduate Student**, *Worcester Polytechnic Institute*, Advisor: Daniel Reichman.
Studying topics in theoretical computer science, proving hardness results in communication and learning complexity, statistical inference, and circuit complexity.

September 2022 **Research Visit**, *Santa Fe Institute*, Host: Cris Moore.
Developed combinatorial methods for proving upper bounds on the expected length of the longest common subsequence of random strings (i.e., the Chvatal-Sankoff constant).

Publications

Accepted papers

May 2021 - September 2022 **The Learning and Communication Complexity of Subsequence Containment**, *WPI*, CS Department.
Proved tight bounds on the complexity of subsequence detection. (ISIT 2023.)

January 2024 - October 2024 **The Karp Dataset**, *WPI*, CS Department.
Introduced a dataset of NP-hardness reductions. (NeurIPS MATH-AI Workshop 2024.)

Preprints and other projects

November 2023 - January 2024 **Nearest Neighbor Complexity and Boolean Circuits**, *WPI*, CS Department.
Studied the representational complexity of the nearest-neighbor classification rule. (Submitted to ITCS 2025)

December 2022 - May 2023 **Inoculation Strategies for Bounded Degree Graphs**, *WPI*, CS Department.
Proved tight bounds on the price of anarchy for an abstract model of epidemic containment. (Submitted to TCS)

May 2018 - August 2018 **Markov Coupling**, *UCONN*, Math Department REU.
Proved upper bounds on the coupling-times of Markov chains under various restrictions. https://opencommons.uconn.edu/srhonors_theses/709/

Professional Experience

May 2023 - July 2023 — **Naval Research Lab**, *Washington, D.C.*.
- Implementated physics-informed neural networks, universal differential equations for practical applications.
- Generalized spectral neural operators to improve performance on localized data.

June 2019 - August 2019 — **Naval Information Warfare Center**, *San Diego, CA*.
- Researched topological data analysis approaches to analyzing neural network behavior.
- Neuromorphic computing: Developed a spiking neural network (SNN) simulator.

July 2015 - September 2015 — **United Technologies Aerospace Systems**, *Windsor Locks, CT*.
- Created an accurate computer simulation of the Integrated Fuel Pump Control system.

Talks

March 2024 — **Communication complexity and linear arrangements**,
Discrete Math Seminar, WPI.

July 2024 — **Nearest neighbor compolexity and boolean circuits**,
CS Theory Seminar, Tufts University.

November 2023 — **Threshold circuit lower bounds from communication complexity**,
Discrete Math Seminar, WPI.

September 2023 — **Introduction to nearest neighbor complexity**,
Discrete Math Seminar, WPI.

September 2022 — **Expected length of the longest common subsequence**,
Discrete Math Seminar, WPI.

October 2021 — **The communication complexity of subsequence detection**,
Discrete Math Seminar, WPI.

Teaching Experience

(Many instances) — **Foundations of Computer Science**, *Teaching Assistant*, WPI.
Introduction to finite automata, context-free grammars, and Turing machines.

(Many instances) — **Introduction to Machine Learning**, *Teaching Assistant*, WPI.
Regression, MLE, clustering, neural networks.

(Many instances) — **Algorithms: Design and Analysis**, *Teaching Assistant*, WPI.
Divide and conquer, dynamic programming, etc., amortized analysis, NP completeness

Specific Skills

Programming — Python, Julia, Java, C, C++, Javascript (node.js)

Implementations — Deep Learning, Convolutional/Recurrent Networks, Reinforcement Learning, Few-Shot Learning, Meta Learning, Evolutionary Algorithms, Neuromorphic Computing (Pytorch, Tensorflow, OpenAI Gym.)