

ISAAC GROSOFF

Computer Science Department
Carnegie Mellon University
Pittsburgh, PA, 15213

isaacg1.github.io
igrosof@cmu.edu
(206) 718-6712

RESEARCH INTERESTS

Design and performance analysis of stochastic computer systems, including theory and implementation.
Queueing behavior and scheduling policies for resource allocation, especially in multicore systems.

CURRENT RESEARCH

Multiserver scheduling policies, scheduling policies for tail metrics.

EDUCATION

2017 - pres: Pursuing PhD in Computer Science. Carnegie Mellon University, Pittsburgh, PA.
Advisor: Prof. Mor Harchol-Balter

2013 - 2017: M.E. and B.S. in Computer Science. Massachusetts Institute of Technology, Cambridge, MA. GPA 4.96/5
Master's Thesis in information-theoretic cryptography: "Secure communication: CDS, PIR, PSM"
Master's Advisor: Prof. Vinod Vaikunathanathan
Bachelor's Advisor: Prof. Frans Kaashoek

REFEREED PUBLICATIONS

Isaac Grosof, Ziv Scully, Mor Harchol-Balter. "Load Balancing Guardrails: Keeping Your Heavy Traffic on the Road to Low Response Times." *Proceedings of the ACM Measurement and Analysis of Computer Systems – SIGMETRICS*: Volume 3, Issue 2, Article 42 (June 2019), pp. 42:1 – 42:31, 2019.

Conference version appeared in *Proceedings of ACM Simetrics/Performance 2019 Conference on Measurement and Modeling of Computer Systems (SIGMETRICS 19)*, Pheonix, AZ. June 2019.

Winner of SIGMETRICS 2019 Best Student Paper Award.

Isaac Grosof, Ziv Scully, Mor Harchol-Balter. "SRPT for Multiserver Systems." *Performance Evaluation*, vol. 127-128, Nov. 2018, pp. 154-175.

Conference version appeared in *36th International Symposium on Computer Performance, Modeling, Measurements, and Evaluation (Performance 2018)*, Toulouse, France, December 2018.

Winner of Performance 2018 Best Student Paper Award.

Erik D. Demaine, Isaac Grosof, Jayson Lynch, and Mikhail Rudoy. "Computational Complexity of Motion Planning of a Robot through Simple Gadgets." *Ninth International Conference on Fun with Algorithms*. La Maddalena, Italy. 2018.

Erik D. Demaine, Isaac Grosof, and Jayson Lynch. "Push-Pull Block Puzzles are Hard." *International Conference on Algorithms and Complexity*. Athens, Greece. 2017.

Benjamin Grosof, Janine Bloomfield, Paul Fodor, Michael Kifer, Isaac Grosof, Miguel Calejo, and Theresa Swift. "Automated Decision Support for Financial Regulatory/Policy Compliance, using Textual Rulelog." *RuleML 2015*. Berlin, Germany. 2015.

OTHER PUBLICATIONS

Isaac Grosof. "Open Problem - M/G/k/SRPT Under Medium Load." *Stochastic Systems*. Sep. 2019.

EMPLOYMENT

Summer 2019: Facebook, Menlo Park, CA.

- Research Intern to develop a machine-learning-based SSD admission policy for Facebook's Tao caching architecture.
- Improved the tradeoff between hit ratio and SSD write rate.

Summer 2018: Microsoft Research, Seattle, WA.

- Research Intern to develop novel FPGA algorithms for linear algebra and solving linear programs.

Summer 2016: Jane Street Capital, LLC, New York City, NY.

- Software developer for non-obtrusive data collection about in-house trading.
- Software developer responsible for updating trading simulation package to accommodate new trade specification format.

Summer 2015: Coherent Knowledge, Seattle, WA.

- Knowledge Engineer to build demonstrations for the financial and natural language domains using the declarative logic programming language Ergo.

2013 - 2014: MIT Undergraduate Research Opportunities Program, Cambridge, MA.

- Researcher in Complexity Theory proving computational hardness of block puzzles and related agent motion problems.

Summer 2014: EMC Isilon, Seattle, WA.

- Software developer to replace the previous ad-hoc build platform with a modern Jenkins-based build platform.

PROJECTS

2014 - pres: Author of new programming language: *Pyth*

- Pyth is one of the best programming languages for solving tasks with the shortest possible programs.
- Pyth is an open-source language written in Python which has an online interpreter and detailed documentation.
- Available at <https://github.com/isaacg1/pyth>

SKILLS

Strong mathematical background, including abstract logic, probability, and mathematical proofs.

Proficient with Python, OCaml, Rust, C++.

Experience with Haskell, C, Java.

Experience in functional programming (e.g., Scheme) and logic programming (e.g., Prolog).

Experience with Linux, Linux shell scripting.