David Gurevich

dgurevich@uwaterloo.ca
(416) 414 0515
gurevich.ca

Technical Skills

- **Languages**: Python, C++, C, MATLAB, SQL, Shell scripting
- **Libraries**: NumPy, SciPy, Pandas, Matplotlib, Flask, OpenCV, FFTW, CTypes
- **Tools/Environments**: Git, Vim, JetBrains IDEs, gdb, CMake, Tableau, Microsoft Excel, Linux, Windows

Education

University of Waterloo — Waterloo, ON
Candidate for Bachelor of Mathematics, Computer Science — *Expected Graduation: 2025*

Relevant courses: Linear Algebra, Calculus, Microeconomics, Probability, Algorithm Design and Data Abstraction
GPA: 3.87/4.00

Experience

York University — Toronto, ON
Research Assistant — *January 2019 - August 2019*

- **Mathematical Research**: Conducted original epidemiological research to design an ODE model to simulate the dynamics of a measles outbreak in secondary schools, with implementation in MATLAB
- **Computational Model**: Designed and implemented an Agent-Based Model alternative to the ODE model in C++ with improved accuracy and realism
- **Software Optimization**: Implemented certain improvements in C++17 software (multi-threading, data structure optimization) to improve execution time by a factor of over 17,000, saving days of compute time
- **Conference Planning**: Supported the planning of the 2019 Annual Society for Mathematical Biology meeting. Developed Python and Shell scripts to automate repetitive tasks

Research in Flows, Inc — Brampton, ON
Software Engineer — *February 2018 - September 2020*

- **Research, Architecture, and Development**: Responsible for the ground-up design and development of digital signal processing system for high frequency signals, including purchasing OEM hardware, and production software development
- **Exceptional Performance**: Developed a high-throughput, multi-threaded Python application for acquisition and processing of high frequency analog signals within strict operational requirements
- **Digital Signal Processing**: Collaborated on and prototyped custom digital signal processing algorithms in MATLAB with final implementation in C++
- **Cost-saving Improvements**: Worked within strict budget requirements to create an effective low-cost solution, demonstrated in an ultrasonic flow-measurement application

Projects

- **3D Sound Synthesis**: Make any sound appear as though it is coming from somewhere else
  C++ (STL, FFTW, CMake), MATLAB
  - Designed a custom algorithm for 3D sound synthesis, given a position on the unit sphere
  - Prototyped in MATLAB, with extremely efficient implementation in well documented C++ code
  - Performed an extensive literature survey to determine the state-of-the-art in 3D sound synthesis
- **Trading Bot**: Python implementation of custom stock trading algorithm
  Python (NumPy, Pandas, Matplotlib)
  - Custom algorithm performs at or above market levels in almost all cases – as much as 220% above market
  - Developed mathematical method for prediction of scale of asset growth
  - Designed custom strategy backtesting framework for easier evaluation
  - Investigated and evaluated various data analysis techniques