# Kalkin

kkalkin2@illinois.edu | kalkin04.github.io

## Education

**University of Illinois Urbana-Champaign** August 2026 –

*Mathematics PhD Student*

*GPA: N/A*

- Courses: Differentiable Manifolds†, Complex Analysis†, Hilbert Spaces†

**University of California San Diego** September 2022 – June 2026

*B.S. in Computer Engineering and Mathematics – 4.0 Major GPA in both*

*Overall GPA: 3.976/4.0, Magna Cum Laude*

- Math courses: Graph Theory*, Abstract Algebra I-III*, Algebraic Topology, Real Analysis I-III*, Analysis of PDEs, Grad Real Analysis I-III†, Numerical Linear Algebra†, Numerical Analysis†, Functional Analysis†, Probability Theory I-III†
- Engineering courses: Signals and Systems, Optimization, Stochastic Processes†, Digital Logic, Computer Architecture, Operating Systems, Algorithms, Graphics, Physics Simulation†, High Performance Computing†, Machine Learning†

* = honors, † = graduate level

## Academic Experience

**Graduate TA** August 2026 –

*University of Illinois Urbana-Champaign* *Urbana, IL*

- TA for calculus
- Responsibilities included holding office hours, proctoring exams, holding discussion sections, and grading

**Undergraduate TA** April 2023 – June 2026

*University of California San Diego* *La Jolla, CA*

- TA'd for courses in probability (2x), digital logic, signals and systems (2x), data structures (4x), and algorithms
- Responsibilities included holding office hours, proctoring exams, holding discussion sections, filling in for lecture, helping design homework and exam questions, and grading homework and exams

**Undergraduate Research Assistant** April 2023 – June 2024

*University of California San Diego* *La Jolla, CA*

- Created a neural net to predict trajectories of Argo floats and optimize the float distribution in order to to make the best use of the $70 million of annual U.S. government funding allocated towards this observing system
- Reduced the dimensionality of temperature and salinity data by 93% while capturing 99.9% of the variance via PCA
- Performed a detailed sensitivity analysis of the neural net over gridded boxes in the Southern Pacific Ocean to investigate which input parameters offer the most skill in different regions of the ocean

### Conference Presentations

Predicting Argo Float Trajectories, *Ocean Sciences Meeting* February 2024
Predicting Argo Float Trajectories, *UCSD Undergraduate Research Conference* April 2024

### Awards

Henry G. Booker Award Spring 2026
UCSD Triton Research & Experiential Learning Scholarship Spring 2024
UCSD Triton Research & Experiential Learning Scholarship Winter 2024

## Industry Experience

**3x Software Engineer Intern** 2024 – 2026

*Roblox* *San Mateo, CA*

- Reduced aerodynamic mesh memory by ~50% while preserving 99.9% accuracy on force and torque evaluation
- Decreased integration time of the aerodynamic force model by 3-4x using SIMD instruction
- Reduced network traffic by quantizing mesh data before serialization, resulting in faster join and load times
- Formulated and implemented a lightweight Fourier series-based turbulence model suited for Roblox's physics engine
- Synchronized the turbulence model between clients to ensure that physical behavior remains consistent
- Improved the aerodynamics, particle rendering and wind audio systems by allowing interaction with turbulent wind
- Designed an automatic fidelity selector to save aero force model runtime costs when accuracy was not important
- Refined the accuracy of the aero force model, correcting artificially high drag and helping improve stability in the solver
- Presented on 3 fluid simulation research papers in a reading group

## Projects (Handwritten)

**Distributed Wave Simulation** - *C++, OpenMPI*

- Approximated solutions to the wave equation over a 2D grid using a 5-point stencil method with stimuli and barriers that can be configured at runtime
- Parallelized the code using OpenMPI for ghost cell exchange, achieving large speedups with speedup efficiency $\approx 0.95$
- Periodically communicated and output the grid of simulation results into netCDF file format for visualization with ncview

**Custom Floating Point Microprocessor** - *SystemVerilog, Assembly, Python*

- Designed a custom Instruction Set Architecture for a microprocessor with 9-bit instruction length and 8-bit register width specialized to run programs for IEEE 754 16-Bit Floating Point operations
- Created a minimal MIPS-inspired load-store instruction set with 8 instructions to support necessary operations
- Wrote programs in the custom assembly language to perform floating point addition and conversions between fixed and floating point numbers
- Wrote an assembler in Python to convert programs written in the custom assembly language into machine code
- Implemented SystemVerilog modules for the hardware logic, verifying correctness of the assembly programs using ModelSim and synthesizability of the hardware using Quartus

**Fluid Solver** - *Python*

- Implemented the PIC, FLIP, and APIC methods in Python for 2D simulation of incompressible fluids
- Performed the pressure solve by writing a divergence operator that takes account of boundary conditions and solved the sparse system using conjugate gradient

**JPEG Compressor** - *C++*

- Implemented the JPEG Compression algorithm, allowing image size reduction by up to 30x without a noticeable loss in fidelity and formatting the output file using the JFIF for it to be easily viewable across all platforms

## Technical Skills

**Languages**: C++, Java, Python, C
**Tools/Libraries**: OpenMPI, SIMD, CUDA, Catch2, JUnit, Git, Github Actions, Linux, LaTeX, Pandas, Numpy