Akshat Srivastava

Research & Engineering Portfolio

IOAA Gold Medalist · BL4S Global Shortlist

Focus Areas:
Computational Physics · Systems Engineering · Science Outreach

---

akshat.srivastava2255@gmail.com
bunchofcellulose.github.io
github.com/bunchofcellulose

January 11, 2026

UNIT I

Original Research & Computational Physics

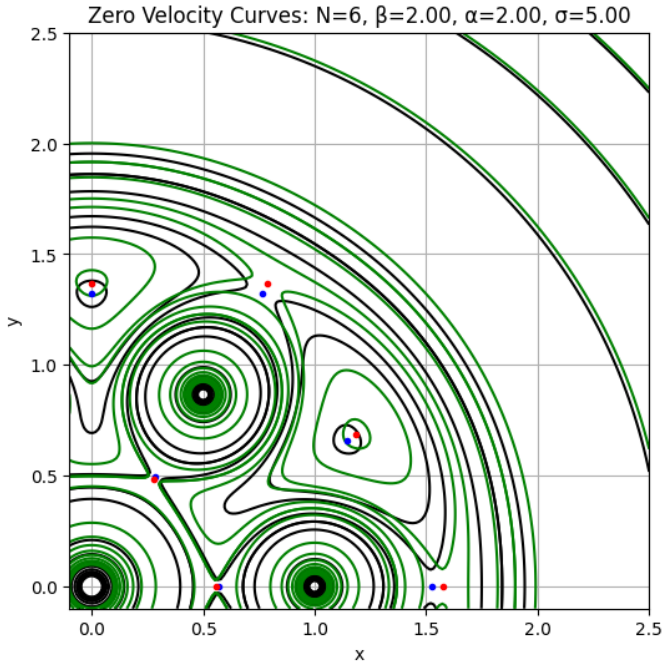
Stationary Points in Yukawa-corrected N+1 Maxwell Ring

Computational Astrophysics · Python · LaTeX

Code

Project Overview

This original research investigates the stability and equilibrium landscape of an $N + 1$ body ring system (a central mass surrounded by N peripheral bodies) under a modified gravitational potential. The study generalizes the classical Maxwell ring problem by introducing a screening parameter (σ) and amplitude (α) to model finite-range interactions, relevant in dusty plasma and screened gravity theories.



Technical Implementation

The potential is modeled as $V \propto \frac{1}{r}(1+\alpha e^{-\sigma r})$. I developed a custom bisection-refined gradient search algorithm to locate stationary points where $\nabla U(r, \theta) = 0$.

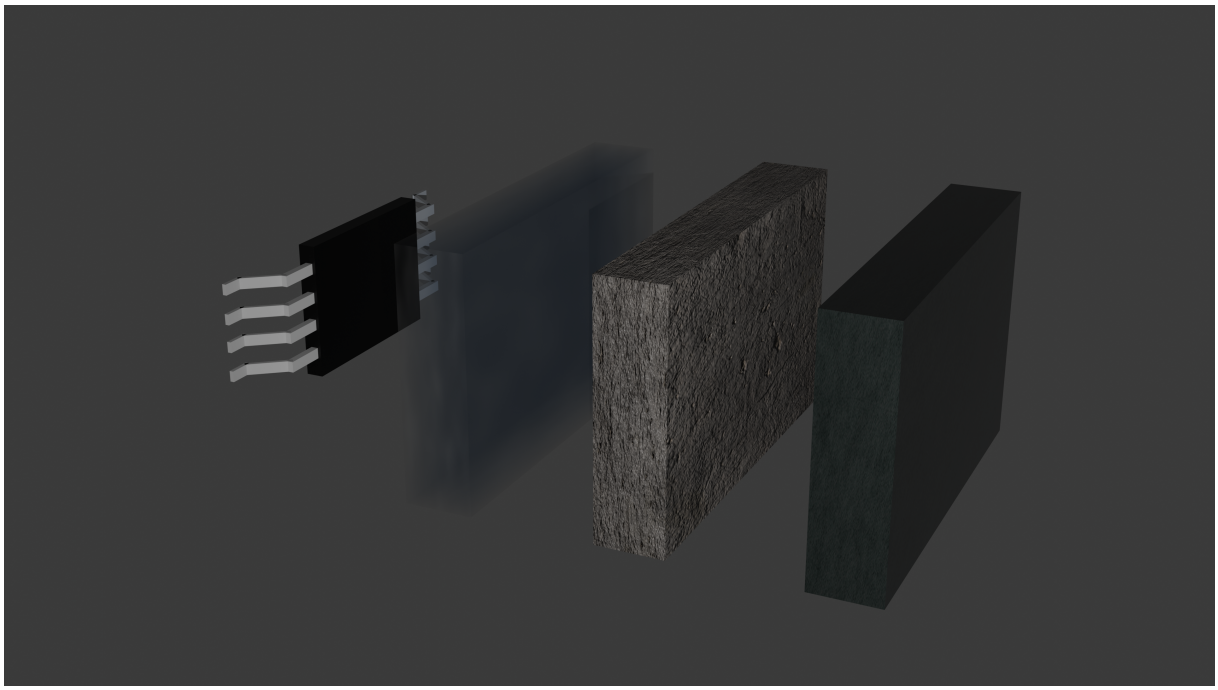
- **Performance:** Utilized `Numba` JIT compilation to accelerate iterative computations for $N = 3$ to 100.
- **Bifurcation Analysis:** Performed a sweep of the (α, σ, β) parameter space, identifying critical saddle-node bifurcations where families of equilibria are created or annihilated.
- **Topology:** Generated Zero Velocity Curves (ZVCs) via the Jacobi constant ($C_J = 2U - v^2$) to map allowed motion regions.

Characterization of Single Event Effects (SEEs) in EEPROMs

Particle Physics · Blender 3D · Experimental Design

Project Overview

Co-authored a research proposal shortlisted in the Global Top 50 for the 2025 CERN Beamline for Schools competition. We proposed an experiment to characterize Single Event Effects (data corruption) in memory chips caused by high-energy cosmic rays, evaluating the efficacy of various shielding materials.



Technical Implementation

- **3D Simulation (Blender):** Modeled the CERN T9 beamline geometry using particle systems. This verified that our detector arrangement (scintillators and calorimeters) had the correct geometric acceptance for the beam.
- **Electronics & PCB:** Designed a custom PCB interface for the M95640 EEPROM chips to allow for rapid read/write cycling during irradiation.
- **Statistical Rigor:** Defined the device vulnerability metric via the Saturated Cross-Section (σ_{sat}) and derived a two-sample t-statistic framework to quantitatively validate if shielding offered a significant reduction in errors.

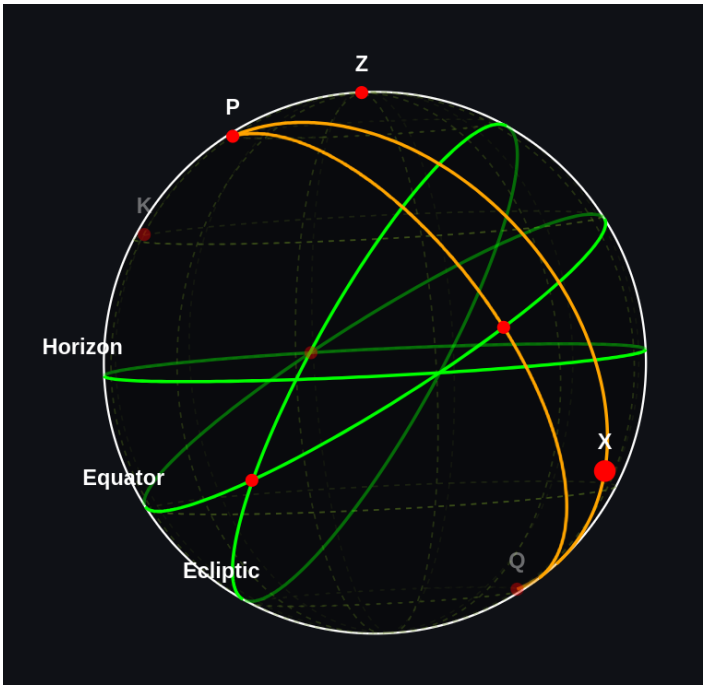
Celestial Sphere

Rust · WebAssembly · Computational Geometry

Code

Project Overview

A high-performance web application compiled to WebAssembly (WASM) that solves the difficulty of visualizing 3D spherical geometry on 2D surfaces. It allows students to visualize and manipulate spherical astronomy diagrams in real-time, serving as an interactive educational tool for spherical trigonometry.



Technical Implementation

Built using Rust and the Dioxus framework.

- **Interactive Geometry:** The engine handles spherical trigonometry math to snap points to great/small circles and calculate intersections dynamically.
- **State Management:** Implements complex state handling for grouping points and selecting nodes for geometric operations (e.g., generating a spherical triangle from three selected points).
- **Exportability:** Supports saving diagrams as JSON for state persistence or exporting as SVG for LaTeX inclusion.

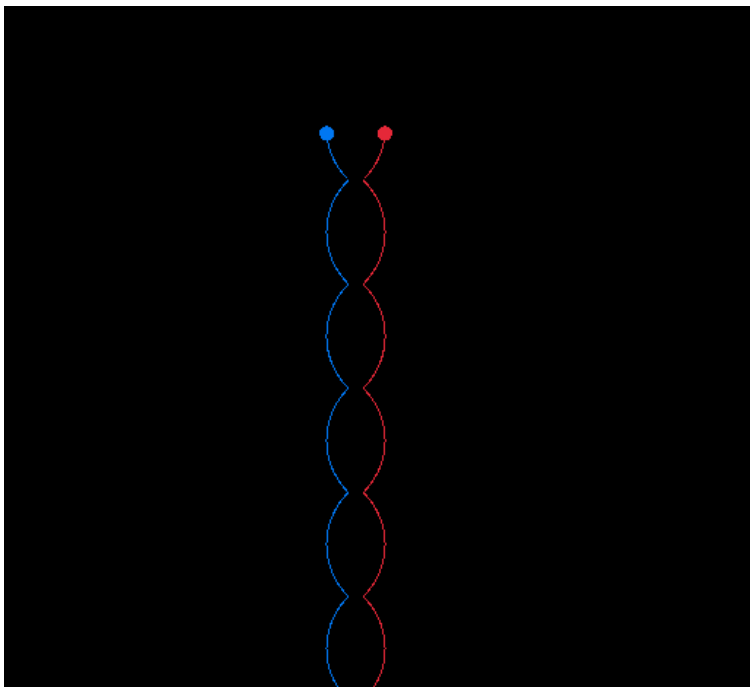
Oganesson: Relativistic Physics Engine

Code

Python · Symplectic Integration · FEM

Project Overview

A work-in-progress physics engine designed to simulate electrodynamics and gravity with high fidelity. Unlike standard engines that approximate physics for visual effect, Oganesson prioritizes numerical accuracy for scientific simulation.



Technical Implementation

- **Finite Element Method (FEM):** Implements FEM for accurate solving of Electric (**E**) and Magnetic (**B**) fields in complex geometries.
- **Symplectic Integration:** Utilizes a 4th-order leapfrog integrator. Unlike Runge-Kutta, this preserves the symplectic form $dp \wedge dq$, ensuring energy conservation over long timescales for orbital mechanics.
- **Relativity:** Handles particle dynamics under the Lorentz force with relativistic corrections.

Diffraction Simulator

Computational Optics · Python

Code

Project Overview

A wave optics simulation engine capable of generating high-fidelity interference patterns for arbitrary aperture geometries. Unlike standard simulators that rely on the Fraunhofer far-field approximation, this tool solves the wave equations directly. This ensures accuracy in both the Near-Field (Fresnel) and Far-Field regimes.

Technical Implementation

The simulation numerically solves the scalar Helmholtz Equation for a monochromatic wave field $U(\mathbf{r})$ propagating through space:

$$(\nabla^2 + k^2)U(\mathbf{r}) = 0$$

where $k = 2\pi/\lambda$ is the wavenumber.

By discretizing the spatial domain and applying boundary conditions defined by the user's slit geometry (circular, single, double, random shaped), the solver computes the complex field amplitude U at the screen. The resulting intensity is derived via $I \propto |U|^2$, capturing rigorous wave phenomena that simple geometric or far-field approximations miss.

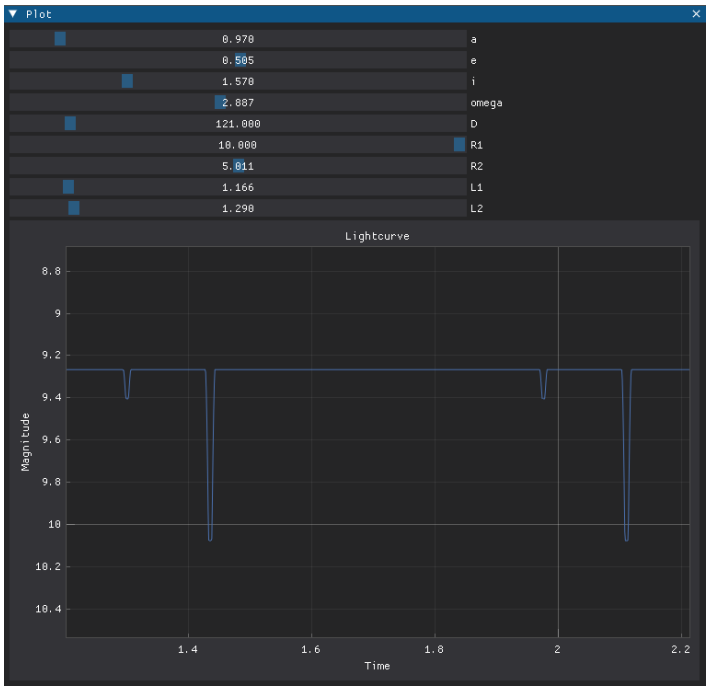
Binary Star Light Curve Tool

Code

Astrophysics · Photometry · DearPyGui

Project Overview

A desktop GUI application that simulates the photometric signature of binary star systems. It visualizes how orbital mechanics translate to observed flux brightness over time, aiding in the interpretation of eclipse data.



Technical Implementation

Built with Python and DearPyGui. The engine accepts orbital elements (eccentricity e, inclination i, period P) and integrates the total flux. It accounts for:

- **Eclipse Geometry:** Calculating the area of overlap between two stellar disks.
- **Limb Darkening:** Modeling the radial decrease in stellar brightness ($I(\mu)$) to produce realistic transit bottoms.

UNIT II

Computer Science &
Systems Engineering

Snotty Compiler

Code

Compiler Design · LL(1) Parser · Brainfuck

Project Overview

A statically typed programming language that transpiles to Brainfuck. The project aims to provide a "sane" programming interface (weak static typing) for an intentionally difficult target architecture (Brainfuck).

```
loop (let i = 1; i < 20; i = i + 1) {
    if i % 3 == 0 & i % 5 == 0 {
        puts("FizzBuzz");
    } else if i % 3 == 0 {
        puts("Fizz");
    } else if i % 5 == 0 {
        puts("Buzz");
    }  else {
        putchar(i + 80);
        putchar('\n');
    }
}
```

Technical Implementation

- **LL(1) Parser:** Implemented a custom recursive descent parser to handle syntax analysis and abstract syntax tree (AST) generation.
- **Memory Mapping:** The compiler manages the translation of high-level constructs (variables, loops) into the linear tape operations ($[\,] <> +-$) of Brainfuck.
- **Algorithms:** Invented and modified existing Brainfuck algorithms for efficient transpilation.

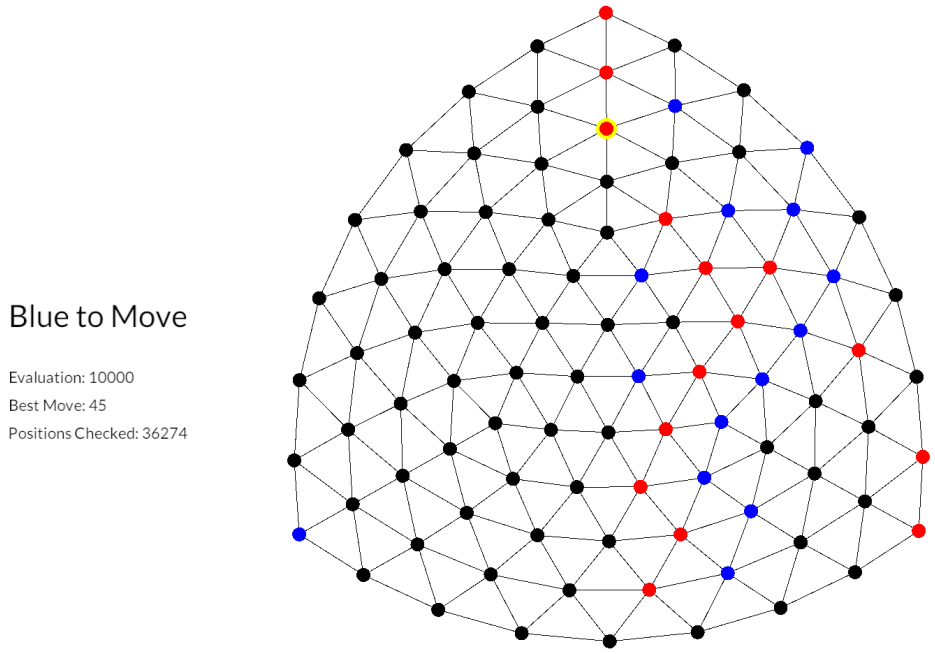
Y-Game AI Agent

Minimax · Alpha-Beta Pruning

Code

Project Overview

A strategy engine for the board game Y (a topological strategy game similar to Hex). The AI plays against human or machine opponents, attempting to form a connection between three sides of the board.



Technical Implementation

- **Minimax Algorithm:** Implemented to traverse the game tree and determine optimal moves.
- **Optimization:** Applied Alpha-Beta pruning to significantly reduce the search space by eliminating irrelevant branches. Transposition tables allow reusage of previously checked nodes.
- **Heuristics:** Developed custom evaluation functions to estimate win probabilities from non-terminal board states based on connectivity graphs.

UNIT III

Leadership, Outreach & Web Design

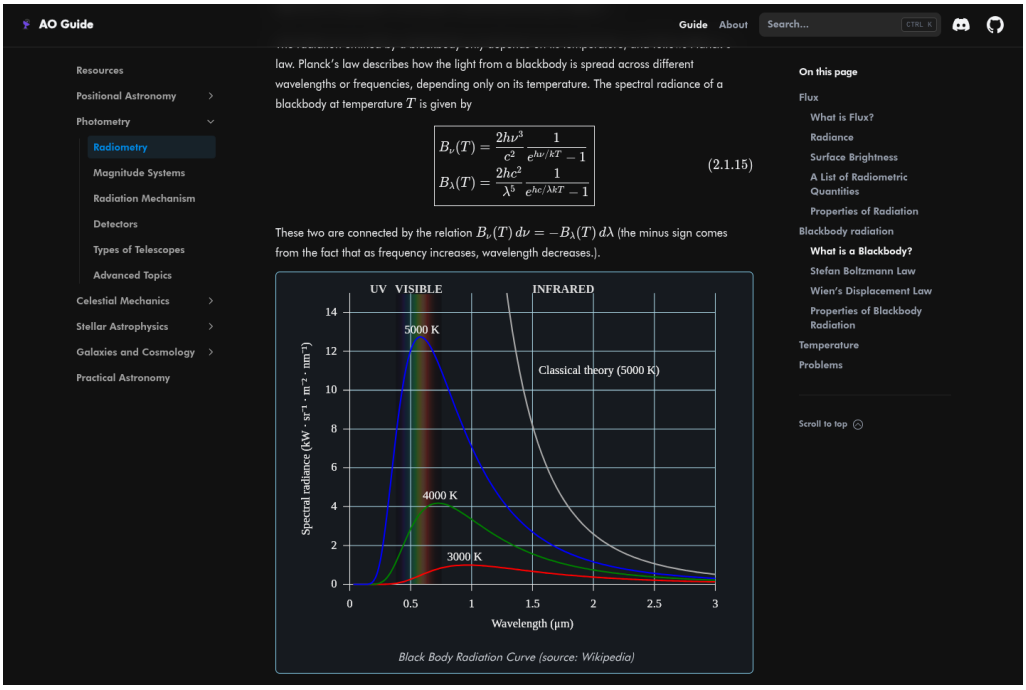
AO Guide (Astronomy Olympiad Guide)

EdTech Platform · Hugo · Open Source

Code

Project Overview

Founder and Lead Author of AO Guide, a comprehensive open-source resource for the International Olympiad on Astronomy and Astrophysics (IOAA). It bridges the gap between dense university textbooks and scattered handouts by providing a focused complete yet concise curriculum. It has consistently over 1.5k weekly views as of 3 months from launch, with over 4,000 learners globally.



Technical Implementation

- **Curriculum Design:** Strictly curated for the IOAA syllabus, emphasizing visual intuition over rote memorization.
- **Stack:** Built with Hugo (Static Site Generator) and hosted on Vercel.
- **Community:** structured as a living document on GitHub, allowing the global astronomy community to contribute problems and fix errata.

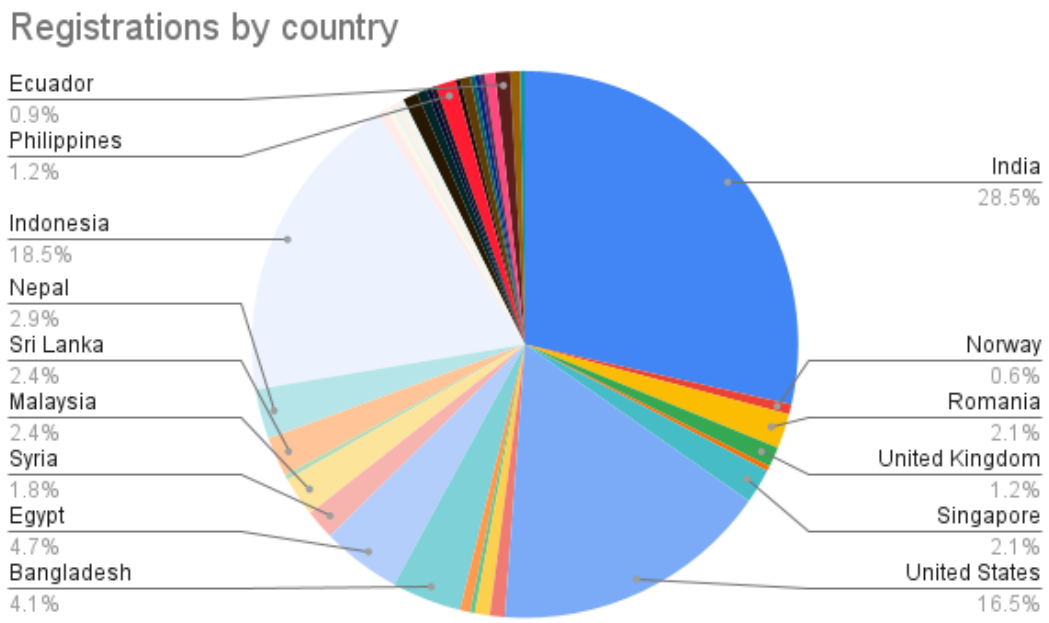
Online Astronomy Competition (OAC)

Global Online Event · Astronomy Outreach

Code

Project Overview

Founder and Organizer of the OAC, an international competition aimed at democratizing access to astronomy education. The platform provides high-quality problems to students regardless of their geographic location and education background.



Technical Implementation

- **Scale:** Successfully hosted over 340 registered participants with 150+ competitors.
- **Format Design:** Structured a two-stage competition (Objective Open Round + Subjective Invitational Round) to test both breadth and depth of knowledge.
- **Tech:** Developed the web portal using Hugo to handle registration and information dissemination.