Computational Physics Project Proposal
Monte Carlo Simulation of 3D Rutherford Scattering

Hani, Amal, Srijana

October 15, 2025

1. Simulation Description and Conceptual Elements

We propose a computational project to simulate the scattering of charged particles by a thin target using a Monte Carlo approach. The simulation is centered around the classical 3D Rutherford scattering scenario and aims to compute the **mean scattering angle** as the principal numerical result.

- **(1) Monte Carlo Simulation of Discrete Elements:** The system consists of N independent particles, each initialized with identical properties and launched toward a target. The path and interactions of each particle are simulated independently using random sampling techniques.
- **(2) 2nd Order ODE Evolution:** The initial trajectory of each particle (before interaction) is governed by Newton's second law, with particles propagating in straight lines under constant velocity (a trivial case of a 2nd order ODE). The formalism allows for future extensions to include more general forces if needed.
- **(3) Probabilistic Interactions:** Each particle undergoes a single elastic collision with the target, where the post-collision scattering angle θ is sampled from the Rutherford differential cross-section, using an accept-reject Monte Carlo method. The azimuthal angle ϕ is sampled uniformly in $[0, 2\pi)$, completing the probabilistic specification of post-collision states.
- **(4) Key Result:** The simulation's main result is the **mean scattering angle** $\langle\theta\rangle$ across all particles. Statistical uncertainty in this metric will be estimated by repeating the simulation with independent random seeds.
- **(5) Physical Validation:** The simulated angular distribution will be compared against the theoretical Rutherford distribution for validation. Conservation of energy and momentum will be explicitly checked for each collision event.

2. High-Level Pseudocode

Set simulation parameters: N particles, incident energy, particle charges, etc. For each repetition (for uncertainty estimation):
For each particle:

Set initial position and direction (e.g., +z axis)
Propagate trajectory using 2nd order ODE (straight line)
Sample θ from Rutherford distribution (accept-reject MC)
Sample ϕ uniformly from $[0, 2\pi)$
Compute new direction vector from θ, ϕ
Record θ (Optional: check conservation of energy/momentum)
Compute mean(θ) for this run
Aggregate mean(θ) across repetitions; report average and standard deviation
Compare angular histogram to theoretical Rutherford distribution for validation

3. Work Plan and Module Division

Module Overview and Responsibilities

- **Particle Initialization & Trajectory Module**
  Functions: `initialize_particles()`, `propagate_trajectory()`
  Purpose: Generate initial conditions and evolve trajectories using a 2nd order ODE (straight-line motion).

- **Monte Carlo Scattering Module**
  Functions: `sample_rutherford_angle()`, `sample_azimuthal_angle()`
  Purpose: Sample post-collision scattering angles according to the Rutherford distribution using accept-reject MC methods.

- **Collision & State Update Module**
  Functions: `apply_scattering()`, `update_particle_state()`
  Purpose: Compute new particle directions and states after scattering event; check conservation laws.

- **Data Collection & Analysis Module**
  Functions: `record_scattering_angles()`, `compute_mean_angle()`, `estimate_uncertainty()`, `plot_histograms()`
  Purpose: Aggregate results, compute mean scattering angle and uncertainty, produce comparison plots to theory.

- **Validation Module**
  Functions: `validate_conservation()`, `compare_to_theory()`
  Purpose: Quantitative checks of energy/momentum conservation, and comparison with analytic expectations.

Example Division of Labor (3 Members)

- **Srijana:** Particle Initialization & Trajectory, and Collision Module

- **Hani:** Monte Carlo Scattering Module (accept-reject implementation) and Data Recording

- **Amal:** Analysis, Uncertainty Estimation, Plotting, and Validation

Key Classes/Functions

- `Particle` class: holds position, velocity, and status (active/scattered)
- `sample_rutherford_angle()` function: implements MC accept-reject sampling from $P(\theta) \propto \frac{\sin\theta}{\sin^4(\theta/2)}$
- `simulate()` function: main simulation driver, orchestrates workflow
- `analyze_results()` function: computes and returns $\langle\theta\rangle$ and uncertainty
- `validate_conservation()` and `compare_to_theory()` functions: implement physical checks

References

- Rutherford, E. (1911). The Scattering of α and β Particles by Matter and the Structure of the Atom. Philosophical Magazine, 21, 669–688. - Additional textbooks or resources as required.