

# Rotation Project: Molecular Dynamics of the Lennard Jones Fluid from Scratch

The Jackson Group

July 24, 2025

For a rotation project every student will write, from scratch in Python, a 3D molecular dynamics (MD) code that reproduces the key structural (and optionally transport) observables of Rahman's seminal simulation of liquid argon (A. Rahman, *Correlations in the Motion of Atoms in Liquid Argon*, *Phys. Rev.* **136**, A405–A411 (1964)). You will implement an NVE Lennard–Jones (LJ) MD simulation with periodic boundary conditions and compare your calculated radial distribution function  $g(r)$  to Rahman's results (Figure 2 in Rahman's paper). In writing this code, the book by Allen and Tildesley should be with you every step of the way - you can access it freely through UIUC here: <https://academic.oup.com/book/27866>. At a minimum you will need to read Chapters 1, 2, 3, 8, and Appendix B to complete the project. I do not expect everyone to finish this project during the allotted rotation time, but I do expect you to work hard to make progress and to regularly interact with me when you need help and/or feedback. You are allowed to use whatever resources you would like to accomplish the project, but you can expect me to ask you questions about whatever code appears and I will expect you to answer these questions convincingly - don't blindly copy others' code or code from ChatGPT or similar.

## Step 1: Familiarity with Python

If you are comfortable with programming in Python, move on to Step 2. If you do not know programming, Python, or need to brush up, I would recommend:

<https://pythonnumericalmethods.studentorg.berkeley.edu/notebooks/Index.html>

## Step 2: A Basic MD Simulation

- **Reduced units:** Set  $\sigma = 1$ ,  $\varepsilon = 1$ ,  $m = 1$ . Later map to argon with  $\sigma = 3.405 \text{ \AA}$ ,  $\varepsilon/k_B \approx 119.8 \text{ K}$ .
- **System size:** Choose  $N = 256$  or  $N = 512$  atoms; cubic box with periodic boundary conditions.
- **State point of simulation:** Reduced density  $\rho^* = N/L^3 = 0.8442$ , temperature  $T^* \approx 0.722$  (canonical “argon-like” LJ state point).
- **Initialization:** Positions on a cubic (or FCC) lattice. Velocities from the Maxwell–Boltzmann distribution at  $T^*$ ; shift to zero total momentum.
- **Forces:** Lennard–Jones potential with cutoff  $r_c = 2.5\sigma$  and either shifted-potential or shifted-force treatment (decide, implement, document).
- **Integrator:** Velocity Verlet with tentative  $\Delta t^* \sim 0.002$ . Verify energy drift and adjust  $\Delta t^*$ .
- **Diagnostics:** Calculate the total, kinetic, potential energies, configurational temperature, and (via virial) pressure.

## Step 3: Thermalization

- **Equilibrate with velocity rescaling**, then switch to NVE for production.
- **Production length:**  $\sim 10^5\text{--}10^6$  steps (depending on  $N$  and  $\Delta t$ ) to get converged statistics.

- **Radial distribution function:**

$$g(r) = \frac{1}{4\pi r^2 \rho \Delta r} \left\langle \sum_{i \neq j} \mathbf{1}\{r \leq |\mathbf{r}_i - \mathbf{r}_j| < r + \Delta r\} \right\rangle.$$

Ensure proper normalization ( $g(r) \rightarrow 1$  at large  $r$ ) and compare peak positions/heights to Rahman or standard LJ benchmarks.

- **Sanity checks:** Equipartition  $\langle K \rangle = \frac{3}{2} N k_B T$ ; total momentum stays  $\approx 0$  in NVE.

## Optional: Transport

- **Velocity autocorrelation function (VACF):**  $C_v(t) = \langle \mathbf{v}(0) \cdot \mathbf{v}(t) \rangle$ . Identify the characteristic negative dip/backscattering discussed by Rahman.
- **Diffusion coefficient  $D$  (optional):** via Green–Kubo (integrated VACF) or Einstein relation (slope of the mean-squared displacement).

## Questions to Think About

1. Why are reduced LJ units convenient, and how do you map back to real units for Argon?
2. What ensemble is generated by velocity rescaling? Why switch to NVE for production?
3. Sources of energy drift: finite  $\Delta t$ , truncated LJ, floating point; how do you diagnose and control each?
4. How would you add analytical tail corrections to energy/pressure? Do they affect  $g(r)$ ?
5. How would you estimate error bars on  $g(r)$ , VACF, and  $D$  (block averaging, integrated autocorrelation times)?
6. Where is the code slow? How could you speed it up?

## Suggested Resources

- M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids*, 2nd ed., Oxford (2017).
- D. Frenkel and B. Smit, *Understanding Molecular Simulation*, 2nd ed., Academic Press (2002).

## Stretch Goals

- Implement Langevin thermostat to sample NVT.
- Implement tail corrections and quantify their impact.
- Compute the static structure factor  $S(k)$  and relate it to  $g(r)$ .
- Explore how the size of your simulated system affects the results.