

Phys 304: Homework 10

Mary Smith*

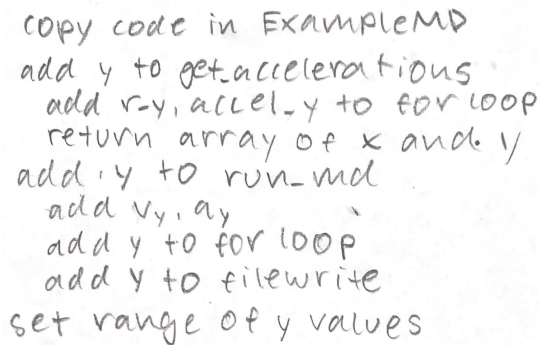
Haverford College Department of Physics

(Dated: April 21, 2024)

[This homework took me about 8 hours. I learned how to code for molecular dynamics, and how to implement a Velocity-Verlet algorithm. I also became familiar with the VMD software and how to use it to visualise physical systems. The assignment was just the right length, given how involved the background information was.]

1. BUILDING A MOLECULAR DYNAMICS SIMULATION

Python code can be built in order to model the behavior of a system of random Argon atoms in one dimension, shown in [1]. This code can then be modified in order to simulate the atoms' motion in two dimensions, according to the pseudocode in Figure 1. In order to understand this code, the background of molecular dynamics must be understood.



```
copy code in ExampleMD
add y to get_accelerations
add r-y, accel-y to for loop
return array of x and y
add y to run_md
add v_y, a_y
add y to for loop
add y to filewrite
set range of y values
```

FIG. 1: [The pseudocode in order to modify the ExampleMD.py file to work in two-dimensions.]

1.1. Background: The Lennard-Jones potential

In order to observe particles interact only through the van der Waals interaction, a mathematical potential function must be created that is able to model this interaction. The van der Waals interaction is a combination of two forces; the long-range attractive London dispersion and the short-range Pauli exclusion principle. This means that the potential function must take both of these forces into account.

The most common potential function that is used to model such molecular dynamics is the Lennard-Jones potential model. This model considers both the London dispersion forces as well as the Pauli exclusion principle:

$$E_{lj}(r) = -4\epsilon\left(\frac{\sigma}{r}\right)^6 + 4\epsilon\left(\frac{\sigma}{r}\right)^{12}, \quad (1)$$

where σ is the distance at which the potential energy between the two particles is zero, ϵ is the potential energy at the equilibrium separation, and r is the distance between the two atoms.

1.2. Background: Force and acceleration

Using Newton's second law of motion, the force between two particles can be found from the interaction energy:

$$f = \frac{-dE(r)}{dr}, \quad (2)$$

where f is the force, r is the position, and $E(r)$ is the interaction energy. This means that the force is the negative derivative of the energy with respect to the position of the particles. This means that the force can be calculated from the Lennard-Jones potential in Equation 1 as

$$F_{lj} = 48\epsilon\frac{\sigma^{12}}{r^{13}} - 24\epsilon\frac{\sigma^6}{r^7} \quad (3)$$

In order to calculate the accelerations, an empty matrix of zeros was created and then filled for each particle's position using Newton's 2nd law $\mathbf{f} = m\mathbf{a}$.

*Electronic address: masmith@haverford.edu

1.3. Background: Integration

Newton's laws of motion can then be applied to find the particle's trajectory.

$$\begin{aligned}\mathbf{x}_i(t + \delta t) &= \mathbf{x}_i(t) + \mathbf{v}_i(t)\delta t + \frac{1}{2}\mathbf{a}_i(t)\delta t^2 \\ \mathbf{v}_i(t + \delta t) &= \mathbf{v}_i(t) + \frac{1}{2}[\mathbf{a}_i(t) + \mathbf{a}_i(t + \delta t)]\delta t,\end{aligned}\tag{4}$$

where δt is the timestep, \mathbf{x}_i is the particle position, \mathbf{v}_i is the velocity, and \mathbf{a}_i is the acceleration. These two equations are together the Velocity-Verlet algorithm, which is used to calculate updated velocities and accelerations at a series of time steps. This acts as an integrator, and effectively integrates the highly non-linear equations in Eq. 4. This means that the integration step is only valid for very small timestep values, usually in the order of 10^{-15}s .

1.4. Background: Initialisation

Lastly, in order to run a molecular dynamics simulation, the original particle positions and velocities must be calculated. The initial structure is extremely important. If the initial structure is unrepresentative of the equilibrium structure, it will take longer than can be reasonably simulated to reach the equilibrium structure. For the purposes of this simulation, the particle positions are based on knowledge of the system.

The particle velocities are more general, as the total kinetic energy and therefore particle velocities are only dependent on the temperature of the simulation. These can be calculated in Equation 5.

$$E_K = \sum_{i=1}^N \frac{m_i |v_i|^2}{2} = \frac{3}{2} N k_B T, \tag{5}$$

where E_K is the total kinetic energy, T is the temperature, m_i is the mass of particle i , N is the number of particles, and k_B is the Boltzmann constant.

1.5. Results: The simulation

From the background discussed in the previous sections, and the pseudocode in Figure 1, a simulation can be built of the behavior of 10 Argon atoms. The initial behavior of the atoms can be seen in Figure 2.

After 5000 time steps, the attractive and repulsive forces the atoms exert on each other can be observed in Figure 3.

Lastly, the final positions of the particles after 10000 time steps can be found in Figure 4. The attractive and repulsive forces have moved the particles completely away from their initial positions, although some remnant of the initial behavior can be seen in their positions.

[1] e. a. McCluskey, A. R., Appl. Crystallogr. **52**, 665 (2019),
URL doi.org/10.1107/S1600576719004333.

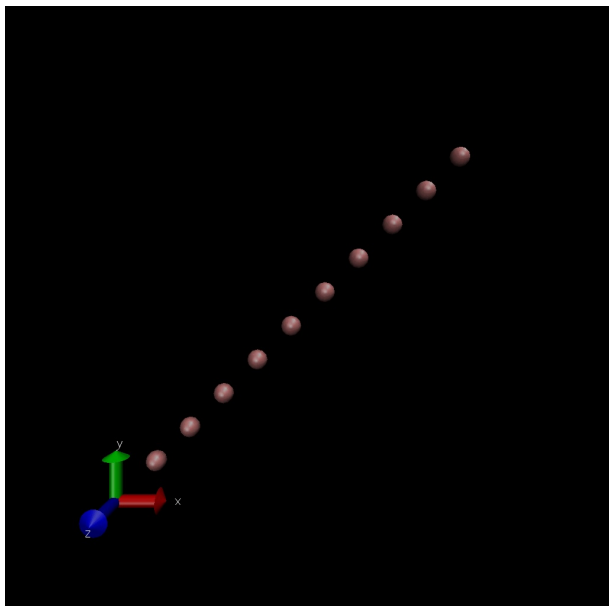


FIG. 2: [The initial positions of the Argon atoms.]

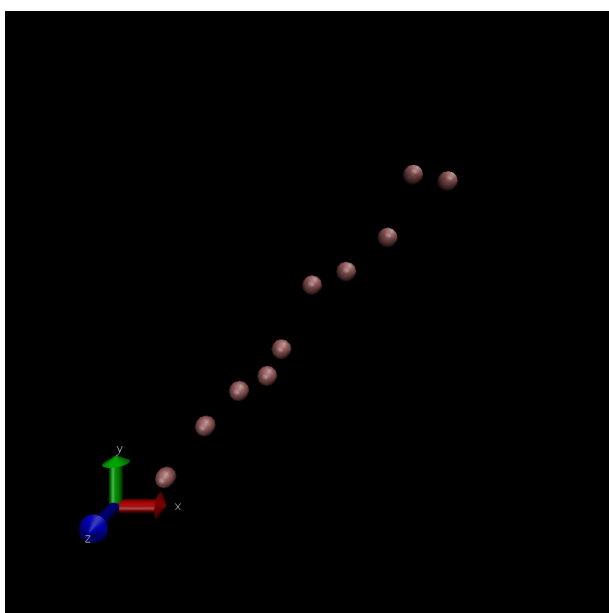


FIG. 3: [The positions of Argon atoms after half of the time steps have passed.]

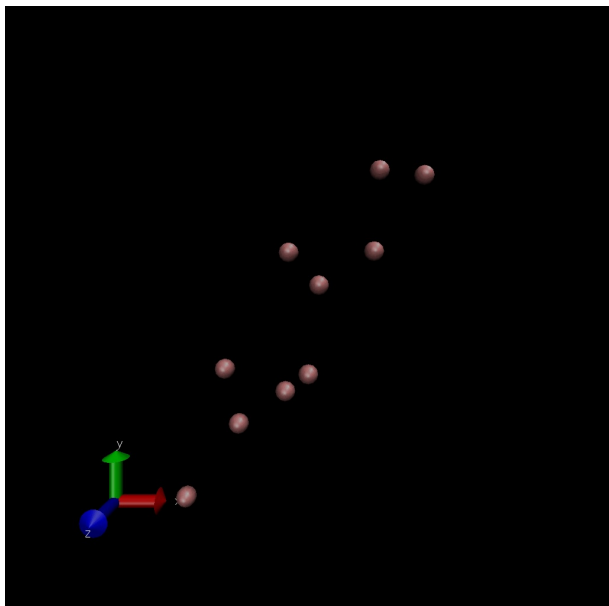


FIG. 4: [The positions of Argon atoms after all 10000 time steps have passed.]