CSCI 7000: Software Engineering for Scientists

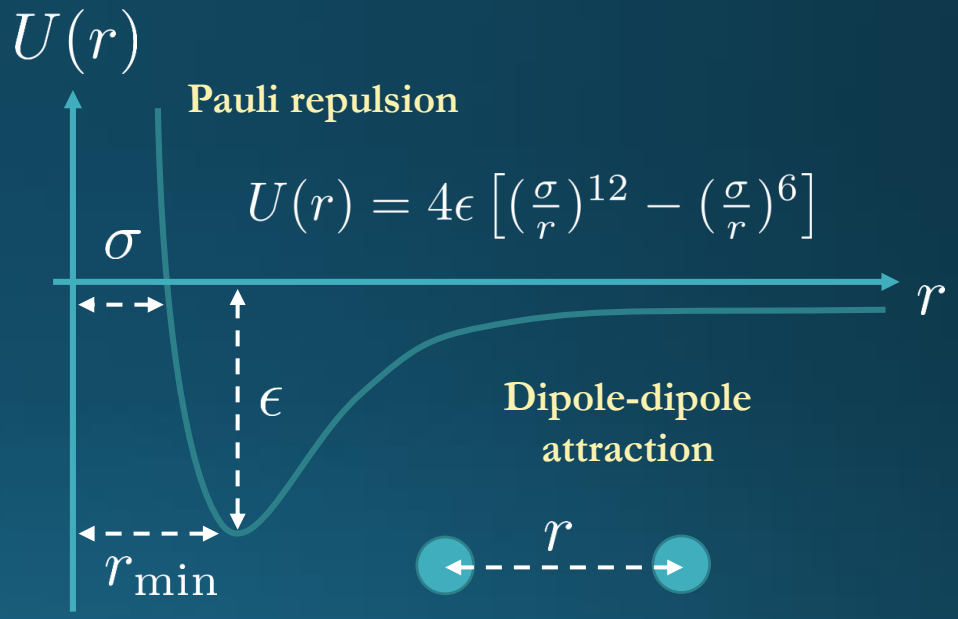
Monte Carlo Simulation of Lennard-Jones Fluids in Canonical Ensembles

Final Project Pitch

Presenter: Wei-Tse Hsu and Chi-Ju Wu

Lecturer: Dr. Ryan Layer

Date: 11.05.2019



Be Boulder.

University of Colorado Boulder

Department of Chemical and Biological Engineering, University of Colorado Boulder

Outline

Scientific backgrounds

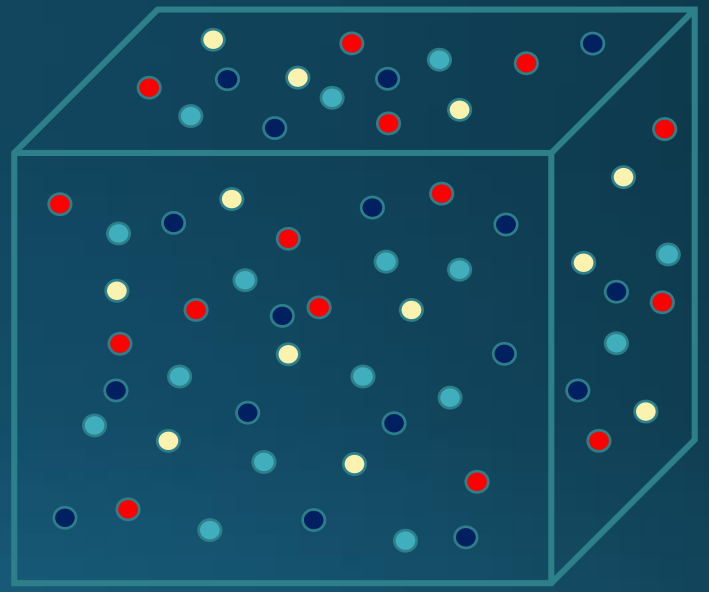
- System of interest
- Lennard-Jones potential
- Metropolis-Hastings algorithm
- Monte Carlo simulation

Software development plan

- Collaboration workflow
- Things to be done in software engineering

Goal: Develop a software package able to perform a Monte Carlo simulation of Lennard-Jones fluids

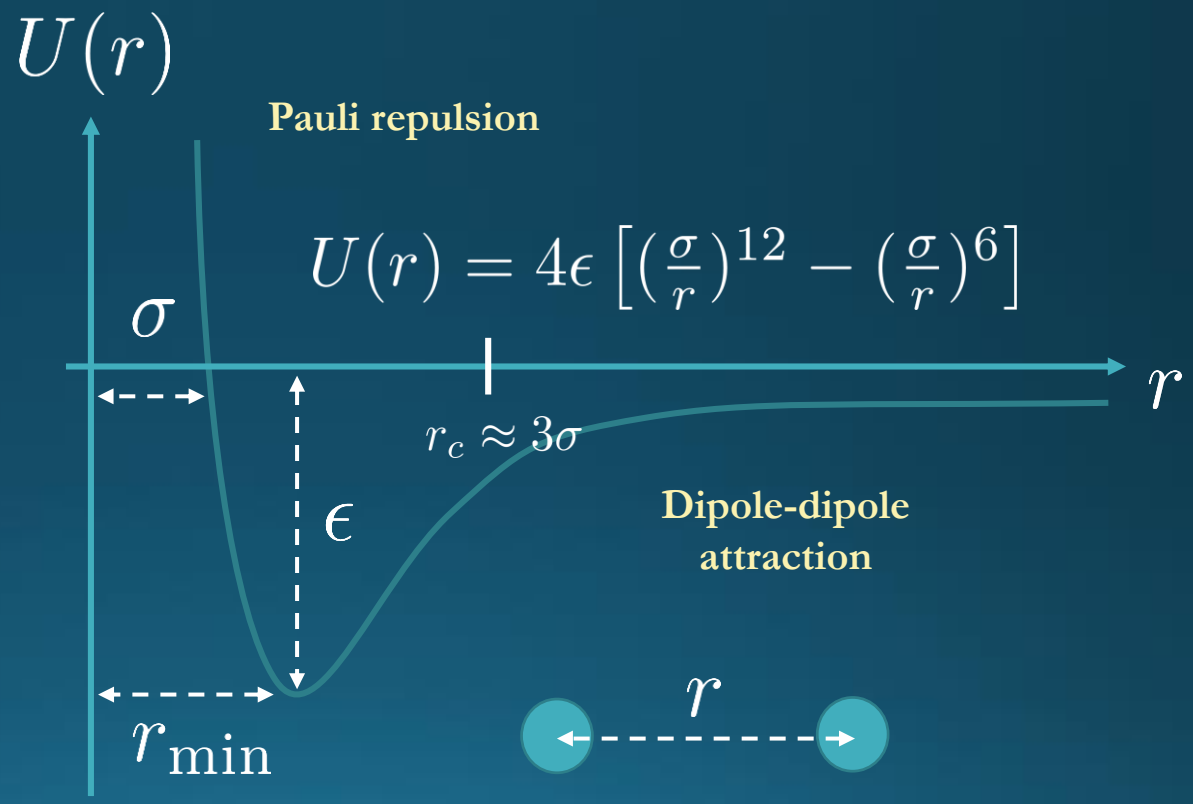
- 500 Lennard-Jones particles inside a box
- Simulation will be performed in a canonical ensemble (constant N, V, and T)
- 10,000,000 MC steps will be carried out.
- Task: Calculate the potential energy of the system and compare it with the NIST benchmark.



Lennard-Jones potential is a simple model which describes the van der Waals interactions between a pair of neutral particles

- r^{-6} term: attractive term
- r^{-12} term: repulsive term
- Tail correction to correct the effect of energy truncation

$$U_{\text{tail}} = \frac{8\pi N^2}{3V}\epsilon\sigma^3\left[\frac{1}{3}\left(\frac{\sigma}{r_c}\right)^9 - \left(\frac{\sigma}{r_c}\right)^3\right]$$



Metropolis-Hastings algorithm drives the energy of a system to its local minima

- Boltzmann distribution $P_k \propto \exp(-U_k/k_BT)$
- Metropolis-Hastings algorithm
  1) Calculate the initial energy U
  2) Choose a particle at random
  3) Propose a move for the particle
  4) Calculate the new/proposed energy U'
  5) Decided if the move should be accepted

$$P_{acc} = \exp(-\Delta U/k_BT), \text{if} \begin{cases} r \leq P_{acc}: \text{ accept} \\ r > P_{acc}: \text{ reject} \end{cases}$$

Why Metropolis-Hastings algorithm?

- If U' < U, the move will always be accepted.
- If U' > U, the move will not always be accepted.
- Therefore, by using Metropolis-Hastings algorithm, the system will be driven to a lower energy, which is more stable.

With Monte Carlo simulations, we can simulate the motion of the particles

Initialization (`class SystemSetup`)
- Parameters setup:
  - ✓ 500 particles, 10 million steps
  - ✓ $T^* = 0.9, \rho^* = 0.9, r_c = 3\sigma$
- Initial configuration: random placement

Energy calculation methods (`class Energy`)
- Periodic boundary condition
- Lennard-Jones potential
- Total pair energy
- Tail correction

Metropolis algorithm (`class MonteCarlo`)
- Calculate energy differences
- Accept/reject Monte Carlo moves
- Displacement adjustment

```
if acc_rate < 0.38:
    max_displacement *= 0.8
elif acc_rate > 0.42:
    max_displacement *= 1.2
```

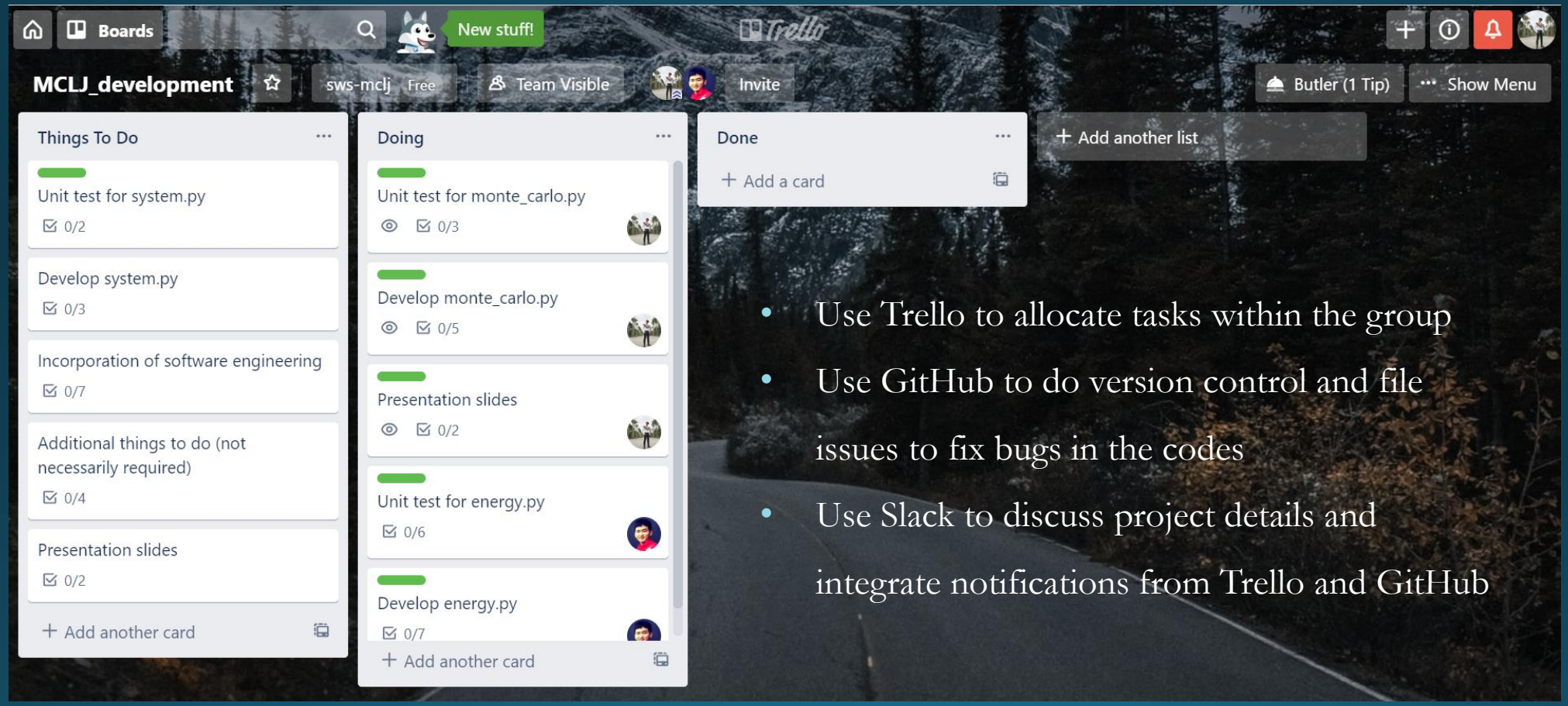
Data generation
- Trajectory file (`.xyz` file)
- Molecular visualization by VMD
- Energy as a function of MC steps
- Final configuration at 3D space

Things to be done in software engineering

- PEP8 coding style and clear documentation
- Unit tests and functional tests
- Continuous integration by TravisCI
- Test-driven development (TDD)
- Potential data structure: hash tables or tree
- Benchmarking and profiling
- Deployment of the software package

Software development plan

Collaboration workflow



- Use Trello to allocate tasks within the group
- Use GitHub to do version control and file issues to fix bugs in the codes
- Use Slack to discuss project details and integrate notifications from Trello and GitHub