P. Koumoutsakos, M. Troyer
ETH Zentrum, CLT F 12
CH-8092 Zürich

Fall semester 2014

Set 9 - Molecular Dynamics and Cell Lists

Issued: November 21, 2014
Hand in: November 28, 2014, 8:00am

In this exercise sheet we will write a molecular dynamics solver for N particles interacting via a Lennard-Jones $V_{LJ}(r)$ potential (fig. 1) in two dimensions. The position $\vec{x}_i$ and velocity $\vec{v}_i$ of a particle i is govern by Newton's equation of motion

$$\frac{\mathrm{d}\vec{x}_i}{\mathrm{d}t} = \vec{v}_i \tag{1}$$

$$\frac{\mathrm{d}\vec{v}_i}{\mathrm{d}t} = \vec{a}_i = \frac{1}{m_i}\vec{F}_i\,, \tag{2}$$

where $\vec{F}_i$ is the sum of all forces acting on the particle of mass m_i.

To integrate these coupled ordinary differential equations we will use the popular Velocity Verlet algorithm

$$\vec{x}_i(t+\Delta t) = \vec{x}_i(t) + \Delta t \cdot \vec{v}_i(t) + \frac{1}{2}(\Delta t)^2 \cdot \vec{a}_i(t) + \mathcal{O}((\Delta t)^3) \tag{3}$$

$$\vec{v}_i(t+\Delta t) = \vec{v}_i(t) + \Delta t \cdot \frac{\vec{a}_i(t) + \vec{a}_i(t+\Delta t)}{2} + \mathcal{O}((\Delta t)^2)\,. \tag{4}$$

We will calculate the forces in a simple N^2 loop in the first exercise and concentrate on optimizing the force calculation for the individual particle pairs. In the second exercise we will improve the scaling of the loop by introducing cell lists.

Question 1: Molecular Dynamics for the N-body problem

Here we will implement the Verlet algorithm for the N-Body problem. For simplicity you may assume all particles to have identical masses and set them to unity. As an (optional) starting point we provide a skeleton code `nbody_skeleton.cpp`.

a) Write a function that, given the positions of two particles at positions $\vec{x}$ and $\vec{y}$, calculates the Lennard-Jones force

$$\vec{F}_{LJ}(\vec{x},\vec{y}) = -\nabla_{\vec{x}} V_{LJ}(\vec{x},\vec{y}).$$

The Lennard-Jones potential V_{LJ} only depends on the relative shortest distance of the two particles with respect to the periodic boundary conditions $r = |\vec{r}| = |\vec{x} - \vec{y}|$:

$$V_{LJ}(r) = \varepsilon\left(\left(\frac{r_m}{r}\right)^{12} - 2\left(\frac{r_m}{r}\right)^{6}\right)\,, \tag{5}$$

where the parameters ε and r_m control the depth and the width of the potential.

Truncate the potential at a radius r_c smaller than the diameter of the simulation box and impose periodic boundary conditions. Your code will spend most of its time in this function, so try to make it as fast as possible. Explain what optimizations you implemented.

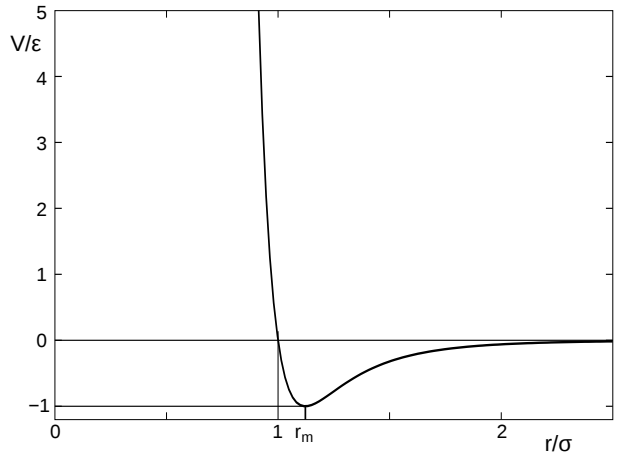

Figure 1: The Lennard-Jones potential (from `http://en.wikipedia.org/wiki/Lennard-Jones_potential`).

b) Use the force calculation to implement a time step of the (velocity) Verlet algorithm for a system of N point particles.

c) Write a function that measures the potential and kinetic energy of a configuration.

$$E_{\text{pot}} = \sum_{i \neq j} V_{LJ}(\vec{x}_i, \vec{x}_j) - V_{\text{shift}}, \qquad E_{\text{kin}} = \frac{1}{2} \sum_i m_i \vec{v}_i^2 \,.$$

Note: Due the cutoff at radius r_c we introduced a discontinuty in the potential $V_{LJ}(\vec{x}_i, \vec{x}_j)$. We compensated for this by shifting the potential by $V_{\text{shift}} = V(r_c)$.

d) Put the pieces together and test your code: Choose a set of particle positions and velocities $\{\vec{x}_i, \vec{v}_i\}$ as initial condition and evolve the system in time. For the parameters of the Lennard-Jones potential you may use $\varepsilon = 5$, $r_m = 0.1$ and a cutoff radius $r_c = 2.5\sigma = 2.5 \cdot 2^{-1/6} \cdot r_m$. Plot the particle positions for several time steps and check that the time evolution looks physically plausible. Monitor the kinetic and potential energies and check that the total energy $E = E_{\text{pot}} + E_{\text{kin}}$ is conserved during the simulation. If the force calculation produces infinities you may need to reduce the time step ($\Delta t \sim 10^{-4}$ should work for the mentioned parameters).

Question 2: Cell Lists

a) Extend your serial implementation of the MD code from the previous question to use cell lists. Comment on the theoretical advantages and measure the scaling of the algorithm with particle number. Compare to your previous results.

b) Parallelize your code using OpenMP and explain your parallelization strategy.

c) Show the strong and weak scaling behavior of your code up to 24 cores.

Summary

Summarize your answers, results and plots into a PDF document. Furthermore, elucidate the main structure of the code and report possible code details that are relevant in terms of accuracy or performance. Send the PDF document and source code to your assigned teaching assistant.