Introduction to Parallel Computing

HOMEWORK

PARALLELIZATION OF AN N-BODY SOLVER

The N-Body problem

This problem follows the behavior of a set of *N* bodies that are considered to interact directly through gravitational forces. For simplicity, each body is considered to be a point-like particle, and the force that body *i* exerts on body *j* can be calculated through Newton's law of universal gravitation:

$$\vec{F}_{ij} = G\frac{m_i m_j}{|\vec{r}_i - \vec{r}_j|^3}(\vec{r}_i - \vec{r}_j) \tag{1}$$

Where *G* is the gravitational constant, m_i and m_j are the mass of the bodies, and r_i and r_j are their positions. The net force acting on particle *j* will be the sum of the contributions from all other particles, i.e.:

$$\vec{F}_j = \sum_{i \neq j} G\frac{m_i m_j}{|\vec{r}_i - \vec{r}_j|^3}(\vec{r}_i - \vec{r}_j) \tag{2}$$

Using Newton's second law of motion we can find the acceleration for each individual particle:

$$\vec{a}_j = \frac{\vec{F}_j}{m_j} = \sum_{i \neq j} G\frac{m_i}{|\vec{r}_i - \vec{r}_j|^3}(\vec{r}_i - \vec{r}_j) \tag{3}$$

The acceleration can then be used to update the velocity and the position of particle *j*:

$$\begin{cases} \frac{d\vec{v}_j}{dt} = \vec{a}_j \\ \frac{d\vec{r}_j}{dt} = \vec{v}_j \end{cases} \tag{4}$$

Starting from an initial distribution of particle positions and velocities, equations (3) and (4) can be used to model the time evolution of the system of *N* bodies.

Solving the equations of motion numerically

There are many techniques for integrating equation (4) numerically, advancing the system from time t_n to $t_{n+1} = t_n + \Delta t$. For this problem we will be using a Leap-Frog method[1] that is second order accurate in time. This method allows for numerically integrating ordinary differential equations of the form:

$$\frac{dv}{dt} = F(x), \frac{dx}{dt} = v \tag{5}$$

By discretizing the equation in time in such a way that the values of x will be know at instants t_n, and the values of v are know at time $t_{n+1/2} = t_n + \Delta t/2$. For the N-body problem we will define the position of particle j, $\vec{r}_{j,n}$, at time t_n, and its velocity, $\overrightarrow{v}_{j,n+1/2}$, at time $t_{n+1/2}$. The time discretization of equation (4) then becomes:

$$\begin{cases} \overrightarrow{v}_{j,n+1/2} = \overrightarrow{v}_{j,n-1/2} + \overrightarrow{a}_{j,n}\Delta t \\ \vec{r}_{j,n+1} = \vec{r}_{j,n-1} + \overrightarrow{v}_{j,n+1/2}\Delta t \end{cases} \tag{6}$$

Where $\overrightarrow{a}_{j,n}$ is the acceleration of particle j calculated using equation (3) for the positions defined at time t_n.

Kinetic Energy

An important diagnostic for the behavior of the N-body system is its kinetic energy:

$$K = \sum_{i=1}^{N} \frac{1}{2} m_i v_i^2 \tag{7}$$

We will be using this quantity to assert that the multiple versions of the algorithm yield the same result.

HOMEWORK

Goals

The goals of this homework are: i) to compile and run the provided serial version of the above described N-body problem, ii) to parallelize the serial code using OpenMP, iii) to parallelize the serial code using MPI and iv) To analyze the performance of all versions on parallel hardware.

[1] See for example https://en.wikipedia.org/wiki/Leapfrog_integration

Implementation

Implementation can be done using any C compiler supporting OpenMP. The use of GCC is recommended. Any MPI library can be used for the MPI project. The number of particles used should be 16384, and the code should be run for 100 timesteps. The code must also print out the total kinetic energy at the beginning of each iteration, and the final kinetic energy for the test.

The attached file, *nbody.c*, implements a correct serial version of this algorithm, using the numerical techniques described above, and should be used as the basis for parallelizing the code. For simplicity it was considered that the gravitational constant is $G = 1$ and that the mass of all particles is the same and also equal to 1. The specific details of the implementation were also chosen for simplicity and not necessarily for performance.

The parallel implementations must parallelize both the particle advance and the kinetic energy calculations, which should of course give very similar results between all versions (negligible differences are to be expected from roundoff errors). Note that all versions of the code must initialize all particles with the same initial distribution, otherwise the results will not be consistent throughout the versions.

Deadline

The deadline for submission is Sunday, October 7, 23:59. Post-deadline submissions will incur a penalty of 1 point (in the 0/20 scale) per day *i.e.* submissions on October 8 (9) will be graded on the 0/19 (0/18) scale.

Documents to submit

The submission of the homework must include: i) the source for the 2 versions (OpenMP / MPI) of the numerical code, including the instructions for compilation and ii) a short report presenting the numerical results describing the parallelization strategy followed in the parallel codes, and discussing the speedup obtained from the serial version by the parallel codes (remember to include a brief description of the hardware used).