Report

High Performance Programming
Assignment 4

by Jiayi Yang and Dominik Lehnert

0. Content

1. Problem description

The problem at hand deals with the calculation of mutual gravitational influences between a number of N objects in a two dimensional space. This so called N-Body problem is solved through the application of Newton's law of gravitation. The gravitational force of each of the N objects on all other N-1 objects is calculated for a number of n steps using a given step size Δt. The program initializes the simulation by reading all necessary information about the N objects and their position from the provided input files.

This assignment is based on work done in the previous assignment. Different to the last assignment is the application of the Barnes-Hut method instead of the $O(N^2)$ algorithm with the goal of improving computational complexity.

2. Solution description

Program execution is started with the input of the following command lien parameters:

N *as the number of stars/particles to simulate*
filename *of the file to read the initial configuration from*
nsteps *for the number of timesteps*
delta_t *for the timestep size* Δ*t*
graphics *as 1 or 0 for weather graphics should be shown.*

In a first step the input data is read from the specified input file (parameter: *filename*). The consistency of the input data is checked before the section of the program is started.

Barnes-Hut method is then used to conduct the calculation of the forces. This method is based on the premise that under certain conditions a number of objects can be casted into a single object, characterized with a total mass according to the sum of the objects and a new center of gravity derived from the single objects. Applying this method therefore allows to lower the number of objects N entering the computation and thereby lowering the computational complexity.

For the implementation of Barnes-Hut method so called quadtrees are applied. Based on recursive subdivision of the domain into four squares, the whole domain gets divided until only one object remains in each square.

This quadtree is then recursively traversed, calculating the resulting forces on all objects/ groups of objects.

The file “quadtree.h” contains the function headers together with the typedef of structure “treeNode” which is used to represent a quadtree. In file “quadtree.c” the methods to insert new nodes, split nodes and to free the quadtrees memory are implemented.

The following steps are executed iteratively over the number of n time steps and the given step size.

First the quadtree is built, starting with all objects in one square and iteratively subdividing it until only one object remains in each square. Secondly the forces are calculated by traversing the quadtree based on the condition:

$$\theta = \frac{\text{width of current box containing particles}}{\text{distance from particle to center of box}}$$

with the given theta_max used to determine whether a square gets further broken down, or if it and all of its included objects are seen as one leaf and are treated as one object in the calculations. Lastly the resulting positions are stored for the next iteration.

In the final code section the calculated final result is written to an output file called *result.gal*, using the same format as the input file.

3. Optimization measures

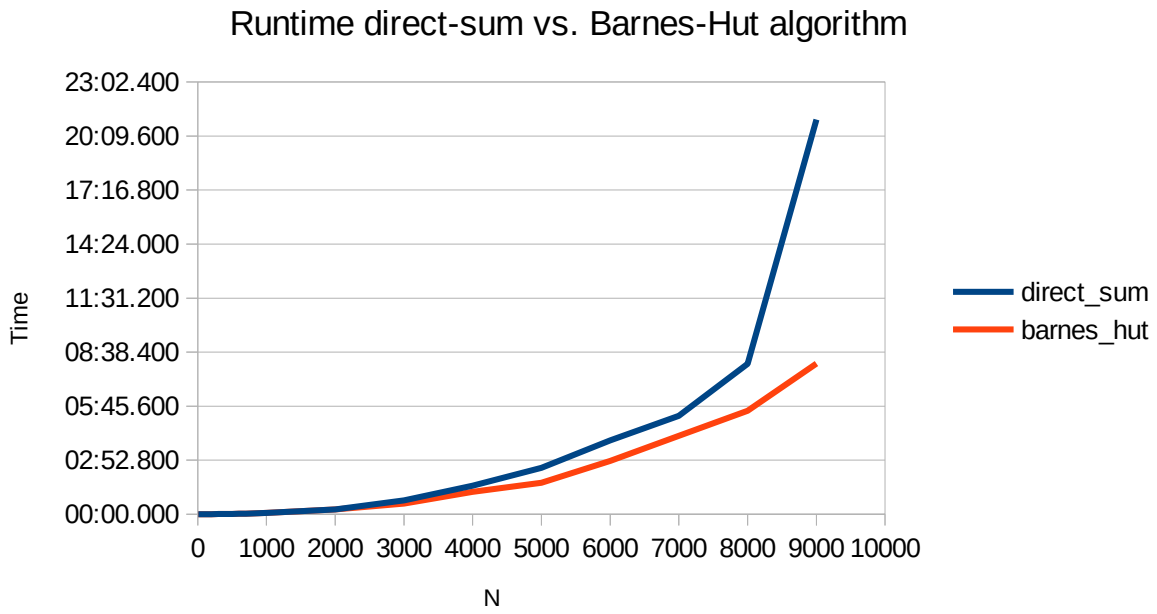
Besides considering fast code execution throughout the whole development process, three measures have been specifically implemented to achieve short run time.

- Code compilation using the -03 flag, ensuring accurate results while achieving a high grade of compiler optimization.
- Const keyword
- Loop unrolling

These measures described before allowed us to achieve the following run times.

Processor specifications:
Intel® Core™ i7-7500U CPU @ 2.70GHz × 4

Results:



The expected complexities $O(N^2)$ and O(n log n) can be observed for the Direct Sum and Barnes Hut algorithms.

4. Division of labor

The work for the assignment was split up, wherein Jiayi focused on coding and debugging and Dominik on the report and measurements.