CSE 498/598k: Homework #2

Due: 10/28/04

Problems 1–3 may be done in pairs. If you choose to work in pairs on this problem, then place both names on this part of the assignment, as well as giving your partner's name on your copy of question #4.

Problem 4 must be done individually.

1. Write a program that simulates the solar system with 9 planets and the sun. The only force present will be gravity. The input file (`solar.inp`) contains all initial positions and momenta as well as the force constants needed. To simplify the problem, we will assume that the motion happens in a 2-dimensional plane. You will use the Leapfrog integrator discussed in class to integrate the sytem in time. Recall Leapfrog:

   - $\frac{1}{2}$-Kick:
   - Drift:
   - Calculate forces:
   - $\frac{1}{2}$-Kick:

   At first, you will implement the force calculation using all-pairs. Once you have determined that this is working, you will be required to implement cutoff force calculation. For cutoff, you will only compute forces between objects that are within a certain distance r_c. You will be required to print out the energies, positions, and momenta. An example output file, `solar.out`, is provided.

2. Similar to problem #1, you will implement a united-atom model of butane. This time, however, there will be only bonded forces: bonds, angles, and dihedrals. The necessary equations to calculate the energies and forces, as well as pertinent constants, can be found in the file `butane.inp`. For this problem, you should run your simulation long enough so that you "discover" the three conformations of butane. This will most likely require 10^6 or greater simulation steps. You will need to keep track of the dihedral energies at each step in order to determine how long the simulation spent in each conformation. An example output file, `butane.out`, is provided.

3. Similar to problem #1, you will implement the solar system problem inside of the framework ProtoMol2. The focus of this problem is to extend an existing piece of software. ProtoMol2 does not contain a "gravity" force, so you will add one and test it on a 3-dimensional model of the solar system. A more detailed set of instructions will be given shortly. Input and output files, `force.inp` and `force.out` are provided. You will find a tarball of the ProtoMol2 source code at

   `/afs/nd.edu/coursefa.04/cse/cse598k.01/www/hw/hw2/protomol2.tar.gz`

   Copy the source code into your directory. Unzip and untar the source with the command:

   ```
   gzip -dc protomol2.tar.gz | tar -xf -
   ```

   You may now remove the file `protomol2.tar.gz`. Enter the `protomol2` directory. In order to compile ProtoMol2, you will need to run the following commands:

   ```
   aclocal; autoheader; autoconf; automake -a;
   ./configure --with-gcc
   make clean; make depend; make
   ```

This will take about 30 minutes to compile on a relatively new processor. In theory, this should work on the Sun machines and on Linux. You can ignore the output from the `make depend` command. If you have any problems compiling, let me know. Once compiled, the executable can be found in `protomol2/applications/protomol-app/protomol`.

4. Reading questions:

   (a) In Sagui and Darden 1999, they quote a study that says, *"...the P3M approach is significantly more efficient than FMM for any conceivable system size, despite the superior asymptotic scaling of the latter [$O(N)$ versus $O(N \log N)$]."* What do they mean? Order the following algorithms according to increasing parallel scalability: MG, P^3M, PME, and FMM.

   (b) In Carlson 2002, how does she suggest multiple protein structures (MPS) be generated? Give the authors and paper title of the first study that employed MPS.

   (c) Using the FlexE paper as a reference, give the pros and cons of using "averaged" protein structures, whether grid-based or not. When is it appropriate to use these?

   (d) Describe simple protein models that are used to predict folding. (HPLS02)

   (e) Mention three difficulties of sampling and approaches used to overcome them. (BeSt97)

   (f) What limits the applicability of MTS integrators? What are two key ideas used in this paper to increase the longest time step of MTS integrators? (MaIz03)

5. Deliverables.

   In your dropboxes, I have created a folder called `hw2` for you to submit all of your files. It is very important that you follow the guidelines given to you as this assignment will be graded using scripts that test the output of your program. I will provide you with a set of initial conditions to test your programs, but I will use a different set to grade them. If you use C or C++, then I am more than happy to help you with your programming. If you choose to use a different language such as Java or Fortran, then I will not be able to help you very much.