Mitchell Miller
PHYS 240
3/19/10

Final Project Outline

My final project is to explore the molecular dynamics of argon. This is described in detail in chapter 16 of *A Survey of Computational Physics*. The main goal of this project is to determine whether a two dimensional array of argon molecules will form an ordered structure. In this simulation, the energy, volume, and total number of particles is fixed. Using Newton's Laws, one can simulate the effects of each molecule on the others. The force on any molecule can be calculated with the sum of central molecule – molecule potentials as follows:

$$F_i(r_0,\dots,r_{N-1}) = -\nabla_{r_i} U(r_0,\dots,r_{N-1}) \quad , \quad U(r_0,\dots,r_{N-1}) = \sum_{i<j} u(r_{ij}) = \sum_{i-0}^{N-2}\sum_{j=i+1}^{N-1} u(r_{ij})$$

$$f_{ij} = \frac{-du(r_{ij})}{dr_{ij}}\left[\frac{x_i - x_j}{r_{ij}}\widehat{e_x} + \frac{y_i - y_j}{r_{ij}}\widehat{e_x} + \frac{z_i - z_j}{r_{ij}}\widehat{e_z}\right.$$

Since computers must work with finite memory, the potential cannot be considered to have an effect at infinite distance. To fix this, a cutoff distance is implemented at a large distance. Since the slope at the cutoff point is undefined, energy cannot be conserved, however, at great distances, the potential is small enough that the error caused is the same order as that of round-off error.

It is clear that this method for calculating potential would become incredibly taxing for a computer as the number of particles gets large. To combat this, a more simplified calculation is performed called the Lennard - Jones Potential.

$$u(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \quad , \quad f(r) = \frac{-du}{dr} = \frac{48\epsilon}{r^2}\left[\left(\frac{\sigma}{r}\right)^{12} - \frac{1}{2}\left(\frac{\sigma}{r}\right)^{6}\right]$$

A table of values for the constants in these equations is provided but it is also common (and computationally easier) to use natural units for these equations, replacing the constants with 1.

The initial velocities for the particles are set using a random Gaussian distribution using the Monte-Carlo method. To compensate for the finite length limits of a computer simulation, some boundary conditions are imposed. Whenever a particle exits one side, an identical particle is simulated to come in the opposite side, maintaining the total number of particles and energy required for this simulation.

Finally, the Verlet Algorithm is used as one more simplification. This algorithm makes use of the forward or central differentiation methods learned in chapter 7.

$$F_i(r(t),t) = \frac{d^2 r_i}{dt^2} \cong \frac{r_i(t+h) + r_i(t-h) - 2r_i}{h^2} \quad , \quad r_i(t+h) \cong r_i(h) + hV_i(t) + \frac{h^2}{2}F_i(t) + O(h^3)$$

$$V_i(t+h) \cong V_i + h\left[\frac{F_i(t+h)+F_i(t)}{2}\right] + O(h^2), \quad \text{Starting Point: } r(t=-h) \cong r(0) - hV(0) + \frac{h^2}{2}F(0)$$

Using these equations, one can modify them and the code provided to create a two-dimensional simulation.