Basic Numerical Analysis

November 13, 2012

Problem Set 6

1. A warm-up.
Write a simple program that does the Fourier- and inverse Fourier-trasnform for a data vector of an arbitrary size. Begin with small data, say,
$\mathbf{x}$ = (2, 3, 5, -1, -2.8, 2.5, -1, 0.1). Check if your code returns the same vector after FFT + inverse FFT. Try to use a publicly available library such as FFTW or one in Numerical Recipes.

2. Solve the 2D Poisson equation

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\phi = \rho(x, y) \tag{1}$$

subject to the (somewhat artificial) boundary conditions

$$\phi = 0 \tag{2}$$

at the boundaries, where the system is defined on the domain $0 < x < 1, 0 < y < 1$. Let us use a Gaussian distribution $\rho(x, y) = \exp[-70((x - 0.5)^2 + (y - 0.5)^2]$.

3. Solve the same equation as for 2 above but use a direct method. Namely, distribute mass (or charge) elements such that the distribution approxiates $\rho(x, y)$. Then sum up the potential generated by all the distributed mass elements (particles). Note the green function for the 2D Poisson equation is $\sim \log r$.
Advanced Write an MPI program to solve the same problem. Use 4 processors. Divide the domain equally in x-direction, such that processor 0 holds $0 < x <= 0.25$, processor 1 holds $0.25 < x <= 0.5$, ...
The mass (charge) distribution can be divided, but you may also use $\rho(0 < x < 1; 0 < y < 1)$ as a shared information. MPIBcast and MPIAllgather should suffice.