P740.HW1.tex

Due 02/05/07

1. Here are two web sites with "movies" of tsunami.

   (a) SUMATRA (12/26/04): http://www.abelard.org/briefings/tsunami.php/ (An animation of the first few hours).

   (b) SUN (12/06/06): http://www.nso.edu/press/tsunami/ (a 9 frame movie).

   From the data in each movie determine the velocity with which the tsunami moves. This is an order of magnitude calculation.
   A simple theory of the velocity of a tsunami on earth gives $C = \sqrt{gh}$ where g is g and h is the depth of the ocean. Estimate the average depth of the ocean.

2. In the calculation of the $\omega - \mathbf{k}$ relationship for a sound wave, done in class, we used a linearized set of equations, Eq. (5) and (6). As these equations are linear and homogeneous nothing sets a scale for the size of things. Suppose you know the size of δP.

   (a) Find a relationship between $\delta P/P_0$, $|\delta \mathbf{v}|/c_0$ and $\delta\rho/\rho_0$ for the case $\mathbf{k} \parallel \delta\mathbf{v}$.

   (b) The threshold for hearing is a pressure fluctuation of size $P_s = 2 \times 10^{-5}$ Pa. Normal speech involves pressure fluctuations of size $P_{ns} \approx 10^6\ P_s$.

      i. Calculate the pressure fluctuation of normal speech.

      ii. Calculate $\delta\rho/\rho_0$ for normal speech.

      iii. Calculate $|\delta\mathbf{v}|$ for normal speech.

3. Random walk simulation. If the diffusion constant of some particles is $D = L^2/\tau$ (here L=l in Notes) the average motion of a particle is supposed to be the same as the average of random walks with step size L, each step taking duration τ. Suppose you wanted to simulate a one dimensional random walk, with L and τ known numbers, out to time $t = N_T\tau$, i.e., an N_T step random walk.

(a) In MATLAB you might construct a **for** loop something like

```
    L=0.025;
    tau=0.0010;
x=0;
t=0;
    for ii=1:NT
        rg=rand(1,1); % rg is uniformly distributed between 0 and 1
          if rg > 0.5
              f=+1;
          else
              f=-1;
          end
        x=x+f*L; % update x
        t=t+tau; % update time
    end
```

(b) At time $t = NT * \tau$ the particle is at x. You should probably do this for many particles and average. So insert this **for** loop inside of another: **for** jj = 1:N, where N is the number particles you are going to average over.

(c) You should probably do this calculation for several values of t. Insert the ii and jj **for** loops in a third: **for** kk= 1:NTmax, where $NT = 1 \cdots NTmax$ are the times you are going to consider.

(d) Suppose you choose $NTmax = 100$. A particle that does a 100 step walk has also done a 99 step walk. A particle that does a 99 step walk has also done a 98 step walk. Do you need **for** kk= 1:NTmax? Or do you just need to collect numerical data thoughtfully?

(e) In the ii **for** loop: you can drop L and tau in the sums and do just one multiplication at the end. That is, the structure of the calculation is independent of specific numbers. So put specifics in only when you need to.

(f) The sequence of values of f in the ii **for** loop is a random sequence of ± 1. You can produce such a sequence with

$$R(1:N) \;=\; -0.5 + rand(1,N); \tag{1}$$

$$f(1:N) \;=\; sign(R(1:N)). \tag{2}$$

The step $x = x + f$ can be carried out by **cumsum(f)**. You may be able to dispense with the ii **for** loop altogether.

Calculate $< x >$ and $< x^2 >$ as a function of t. Here $< \cdots >$ stands for "an average over realizations" of the random walk. That is "physics speak" for averaging over the random walk of some number of particles. Expect $< x^2 > \propto Dt$. Plot $< x >$ and $< x^2 >$ as a function of t. Fnd the numerical factor C in $< x^2 >= CDt$. Run **tic-toc** for your calculation. Your grade on this problem will be determined by the accuracy of your value of C and the result from **tic-toc**. [For non-Matlab users **tic-toc** is a clock which determines how much time a calculation takes.]