PC5215, Numerical Recipes with Applications

Lab 1, starting 2:00pm 18 Aug for lab briefing, due Friday, 4 September 2020

Submit in hard copy your codes in Python with a report of the steps and analysis, *as well as answers.*

1. Use the "Numerical Recipes" LU decomposition routines [ludcmp(), lubksb()] to solve the following 4×4 linear equations, and verify (by hand or using other software, such as matlab, or a consistent check) that the answers are correct.

$$\begin{bmatrix} 1 & 3 & 3 & -5 \\ 2 & -4 & 7 & -1 \\ 7 & 1/2 & 3 & -6 \\ 9 & -2 & 3 & 8 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 3 \\ -10 \end{bmatrix}$$

2. Consider square grids of $L \times L$ resistor network shown below (for the case of L=4). Given that the point A has voltage 0 and B voltage 1, compute the current flowing from B to A, and the total resistance of the network. We assume that each link has a resistance $r = 1/\sqrt{L}$. Use the same linear solver as in problem 1 above. Report the results on L = 1, 2, 4, 8, 16, 32, …, (as large as you can get), and the CPU times needed. Then Compare with the answers and CPU times with scipy linalg.solve() results as well. Also check your computer answer against hand calculation for the case of L=1 and 2.

To set up the linear equations, you need (1) Ohm's law $I_{ij} = (V_j - V_i)/r$ for each link, (2) conservation of current $\Sigma\, I_{ij} = 0$ at each node. And make linear equations for the voltage at each node. Be careful about the sign of the current. The program should work for general L.