UCB Math 228B, Spring 2015: Problem Set 7

Due May 7

1. Write a function `nestdiff` that for a positive integer input argument n computes a nested dissection permutation vector p for a regular n-by-n grid. To compute a separator for a rectangular block, simply split across the longest edge size.

As an example, for $n = 5$ you could renumber the nodes according to below:

$$\begin{array}{ccccc} 21 & 22 & 23 & 24 & 25 \\ 16 & 17 & 18 & 19 & 20 \\ 11 & 12 & 13 & 14 & 15 \\ 6 & 7 & 8 & 9 & 10 \\ 1 & 2 & 3 & 4 & 5 \end{array} \quad \longrightarrow \quad \begin{array}{ccccc} 7 & 8 & 25 & 17 & 18 \\ 5 & 6 & 24 & 15 & 16 \\ 9 & 10 & 23 & 19 & 20 \\ 3 & 4 & 22 & 13 & 14 \\ 1 & 2 & 21 & 11 & 12 \end{array}$$

which gives the permutation

$$p = [1, 2, 6, 7, 16, 17, 21, 22, 11, 12, 4, 5, 9, 10, 19, 20, 24, 25, 14, 15, 3, 8, 13, 18, 23].$$

Try to verify that your code is correct, for example by studying the output for small n or by plotting the sparsity pattern of a permuted Poisson model problem. Also make sure it is fast enough to handle sizes up to $n = 320$.

2. Study the fill-in for the Poisson model problem of size n (see lecture slides for Python and MATLAB code) when factorized using LU factorizion (`scipy.sparse.linalg.splu` in Python, `lu` in MATLAB). Use the following three different reordering schemes:

(a) Natural ordering (no reordering)

(b) Approximate minimum degree (`colamd`)

(c) Nested dissection (from problem **1.**)

Permute both the rows and the columns of A using each of the schemes and compute LU factorizations of the permuted matrices. Use the problem sizes $n = 10$, 20, 40, 80, 160, and 320, and plot the total number of non-zeros in the LU factors as a function of n in a log-log graph. Try to determine the dependency on n, that is, $O(n^\alpha)$ or $O(n^\alpha \log n)$ for some value of α.

Note that `splu` in Python reorders with `colamd` by default, and you need to pass the argument `'natural'` to suppress that.