Sep 25, 2020

Gaussian elimination with sparsity

A sparse, $PA = LU$, are L, U also sparse?

$PAQ = LU$ $\quad P, Q$ permutation

$$Ax = b \Rightarrow \underbrace{PAQ}Q^T x = Pb$$

$$LUy = Pb \qquad Q^T x = y \quad x = Qy$$

choosing Q to encourage sparsity in L, U

"sparse direct"

Compressed sparse column (CSC)

$$y = Ax$$

$$y = \sum_j A(:,j)x_j$$

$$O(\underline{nnz}(A))$$

$$\begin{pmatrix} x & x & x & x & x \\ x & x & & & \\ x & & x & & \\ x & & & x & \\ x & & & & x \end{pmatrix}$$

first step: $L(2:5,1) = A(2:5,1)/A(1,1)$

$A(2:5,2:5) \mathrel{-}= L(2:5,1)\, A(1,2:5)$

$$\begin{pmatrix} x & & & & x \\ & x & & & x \\ & & x & & x \\ & & & x & x \\ x & x & x & x & x \end{pmatrix}$$

$L(2:5,1) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ x \end{pmatrix}$ $\quad A(1,2:5) = \begin{pmatrix} 0 & 0 & 0 & x \end{pmatrix}$

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x \end{pmatrix}$$

$$\underset{L}{\begin{pmatrix} x & & & \\ & x & & \\ & & \ddots & \\ x & \cdots & \cdots & x \end{pmatrix}} \quad \underset{U}{\begin{pmatrix} x & & & x \\ & x & & \vdots \\ & & \ddots & \\ & & & x \end{pmatrix}}$$

$O(nnz(L) + nnz(U))$

$= O(nnz(A))$

Matrices $\Longleftrightarrow$ graphs

Graph $= (V, E)$

$V = \{1, 2, \ldots, n\}$

E = set of pairs of vertices

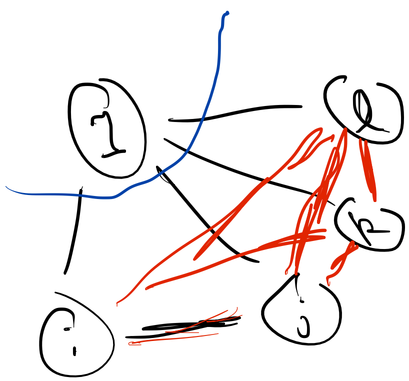
edges

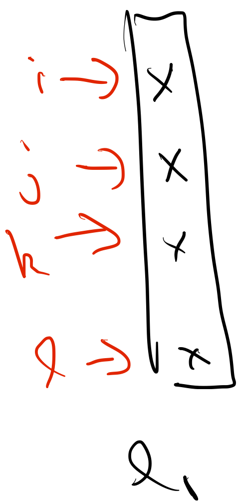
$$A_{ij} = \begin{cases} 0 & (i,j) \notin E \\ \neq 0 & (i,j) \in E \end{cases}$$

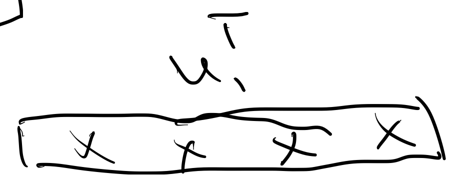
symmetric matrices $\Longleftrightarrow$ undirected graphs

$$A = \begin{bmatrix} a_{11} & a_{12}^T \\ a_{21} & A_{22} \end{bmatrix}$$

where do we get fill-in?

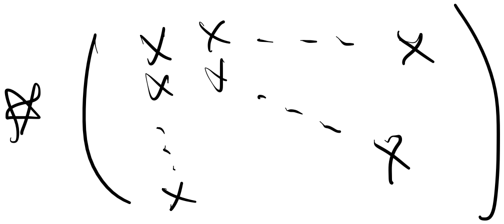




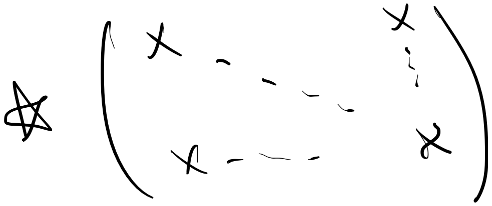


any pair of neighbors of node 1 gets filled in

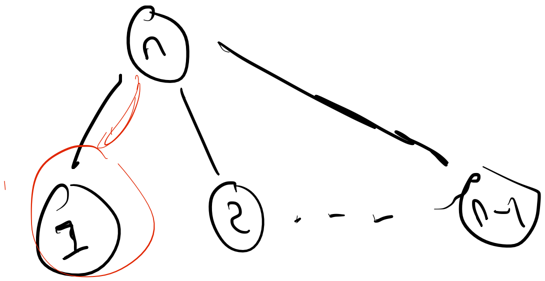
(equiv. add edges to graph)



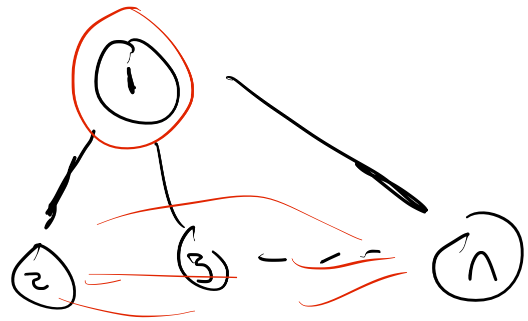
first node is neighbors with everyone



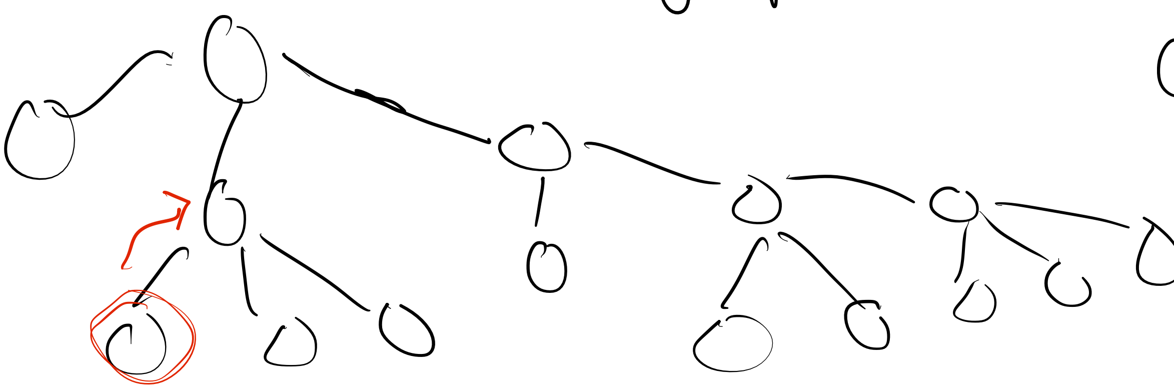
first node is only neighbors with one other node



<u>tree</u>



Fact: no fill-in when graph is a tree, go leaves up



Cor: $O(n)$ time to solve $Ax = b$

Most graphs are not trees

One heuristic: go by increasing degree

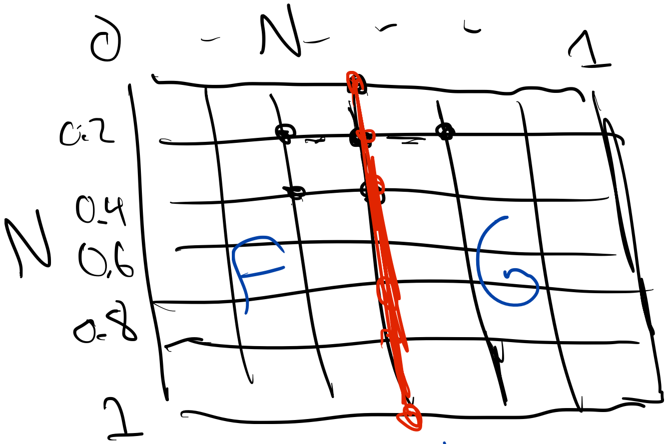
degree = # of neighbors of node

Nested dissection

A_{ij} $i = (0.4, 0.6]$, $j = (0.6, 0.8]$

Basic idea: block structure of tree

A is $N^2 \times N^2$



indices of A: points on mesh

non-zeros: Adjacent points

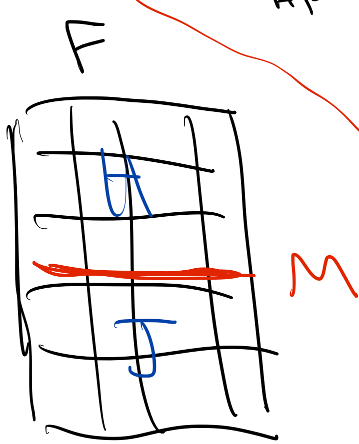
$nnz(A) = O(N^2)$

vertex^H separator: $O(N)$ points

$O(N^2)$

$A_{FF} = L_F U_F$

$$A = \begin{pmatrix} A_{FF} & 0 & A_{FH} \\ 0 & A_{GG} & A_{GH} \\ A_{HF} & A_{HG} & A_{HH} \end{pmatrix} \Longrightarrow \begin{pmatrix} L_F U_F & 0 & L_F^{-1} A_{FH} \\ 0 & A_{GG} & A_{GH} \\ A_{HF} U_F^{-1} & A_{HG} & A_{HH} - A_{HF} A_{FF}^{-1} A_{FH} \end{pmatrix}$$



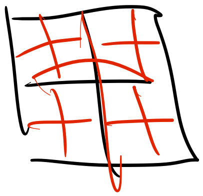
$$A_{FF} = \begin{pmatrix} A_{HH} & 0 & A_{MH} \\ 0 & A_{JJ} & A_{JM} \\ A_{HM} & A_{MJ} & A_{MM} \end{pmatrix}$$

Key concept: vertex separator much smaller than blocks

Can prove: 2D meshes $\Rightarrow O(N^3)$ instead of $O(N^6)$

N

N

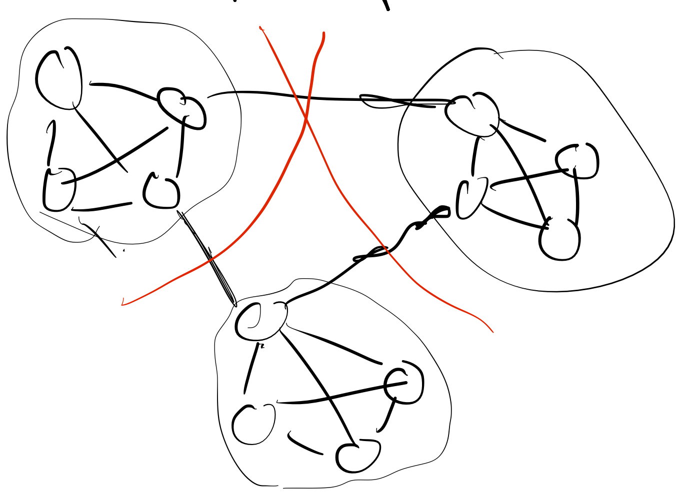


$O(n^3)$ $\quad n = N^2 \quad$ $O(n^{3/2})$

How do we find good separators?

1. Physical geometry (meshes)
2. Graph partitioning

can do this with eigenvectors!



$$\begin{bmatrix} M_1 & & \\ & M_2 & \\ & & M_n \end{bmatrix}$$