6.896
2/18/04
L5.1

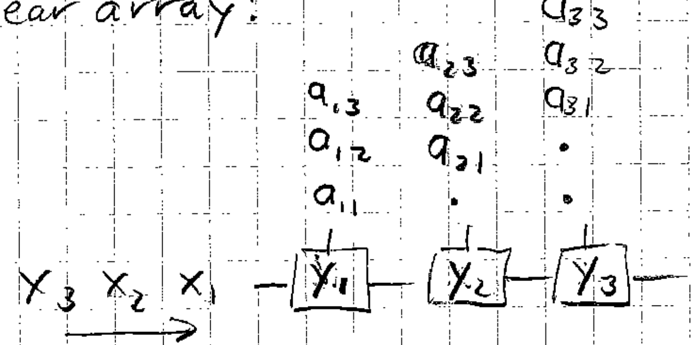
Matrix computations

- dense matrices
- mesh networks (1D & 2D)
- word model

Matrix-vector mult.

$N \times N$ matrix $A = (a_{ij})$
N-vector $x = (x_j)$
Compute N-vector $y = (y_i)$, where $y_i = \sum_{j=1}^{N} a_{ij} x_j$

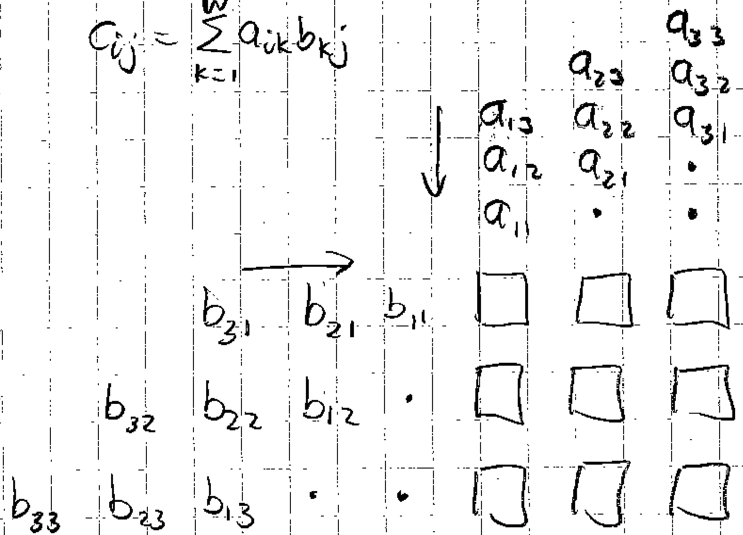
Linear array:



$\Theta(N)$ time ($2N$ steps)
$\Theta(N)$ HW

Matrix mult.

$N \times N$ matrices.
Compute $C = AB$
$c_{ij} = \sum_{k=1}^{N} a_{ik} b_{kj}$



$\Theta(N)$ time ($3N$ steps)
$\Theta(N^2)$ HW

Torus: mesh + "end around"
N steps:

$a_{11}b_{11}$ $a_{23}b_{31}$ $a_{32}b_{21}$

$a_{13}b_{32}$ $a_{22}b_{22}$ $a_{31}b_{12}$

$a_{12}b_{23}$ $a_{21}b_{13}$ $a_{33}b_{33}$

6.896
2/18/04
L5.2

Simulating torus on mesh:

Each step of torus simulated by 2 mesh steps.

Gaussian elimination

$Ax = b$, solve for x (A symmetric, positive definite or irreducible diag dominant $\Rightarrow$ pivot on diagonal)

$$\begin{pmatrix} 2 & -3 & 1 \\ 1 & -1 & -2 \\ 3 & 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 7 \\ -2 \\ 0 \end{pmatrix}$$

Tabular method:

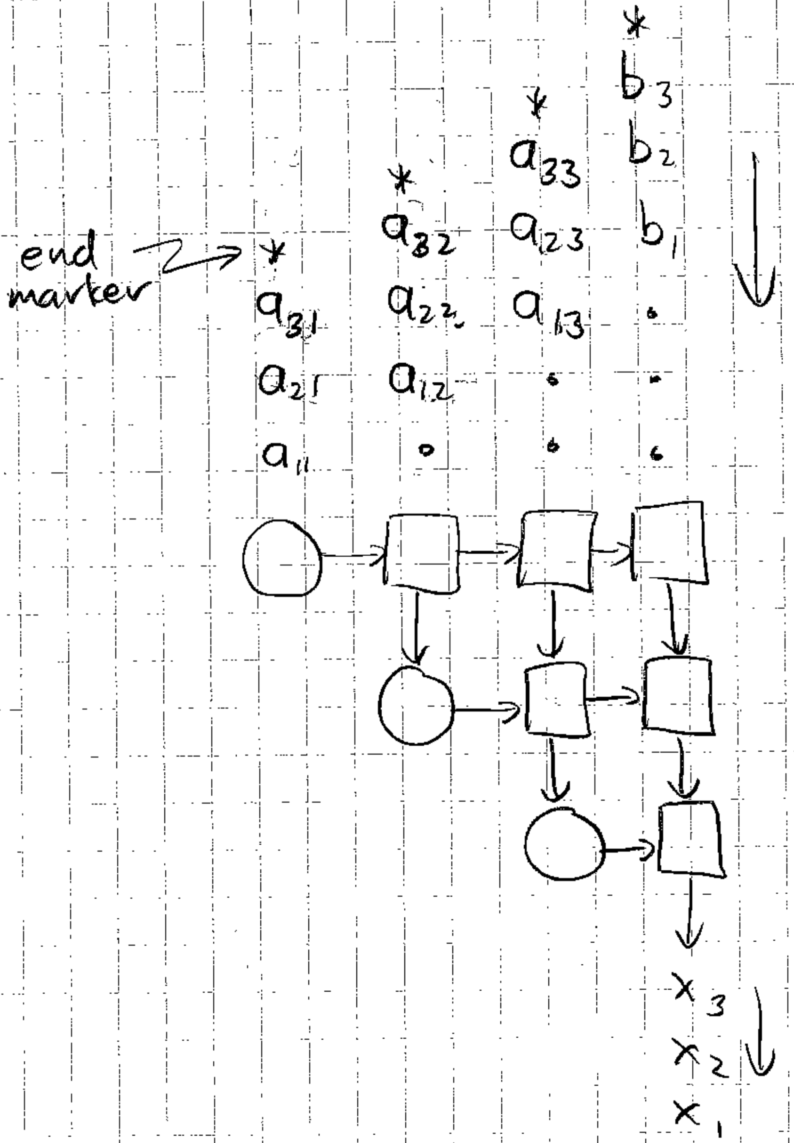
$$\begin{array}{ccc|c} 2 & -3 & 1 & 7 \\ 1 & -1 & -2 & -2 \\ 3 & 1 & -1 & 0 \end{array} \Rightarrow \begin{array}{ccc|c} 1 & -\frac{3}{2} & \frac{1}{2} & \frac{7}{2} \\ 1 & -1 & -2 & -2 \\ 3 & 1 & -1 & 0 \end{array} \Rightarrow \begin{array}{ccc|c} 1 & -\frac{3}{2} & \frac{1}{2} & \frac{7}{2} \\ \underline{0} & \frac{1}{2} & -\frac{5}{2} & -\frac{11}{2} \\ \underline{0} & \frac{11}{2} & -\frac{5}{2} & -\frac{21}{2} \end{array}$$

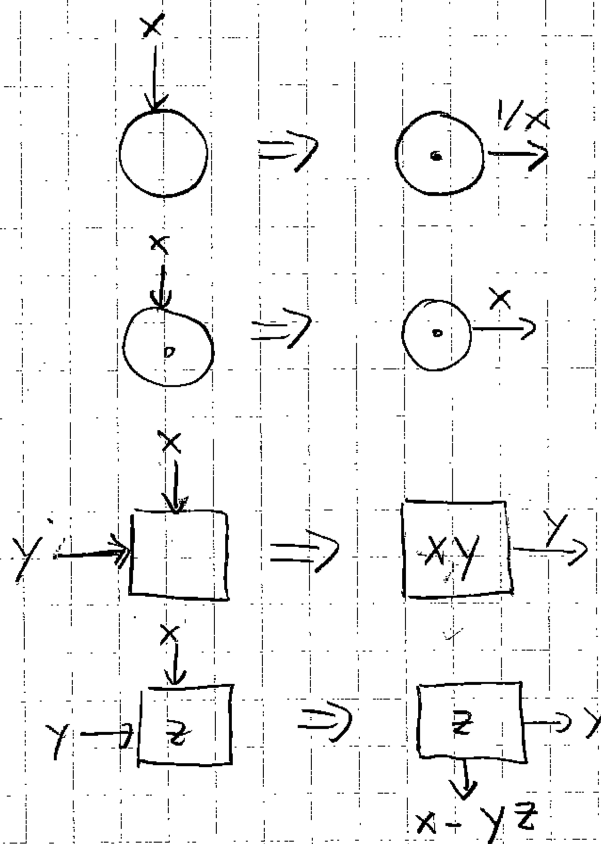
$$\Rightarrow \begin{array}{ccc|c} 1 & -\frac{3}{2} & \frac{1}{2} & \frac{7}{2} \\ 0 & 1 & -5 & -11 \\ 0 & \frac{11}{2} & -\frac{5}{2} & -\frac{21}{2} \end{array} \Rightarrow \begin{array}{ccc|c} 1 & \underline{0} & -7 & -13 \\ 0 & 1 & -5 & -11 \\ 0 & \underline{0} & 25 & 50 \end{array} \Rightarrow \begin{array}{ccc|c} 1 & 0 & -7 & -13 \\ 0 & 1 & -5 & -11 \\ 0 & 0 & 1 & 2 \end{array}$$

$$\Rightarrow \begin{array}{ccc|c} 1 & 0 & \underline{0} & 1 \\ 0 & 1 & \underline{0} & -1 \\ 0 & 0 & 1 & 2 \end{array} \qquad x = \begin{pmatrix} 1 \\ -1 \\ 2 \end{pmatrix}$$

$$\boxed{a_{ij}^{(k)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)} a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}}}$$

6.896
2/18/04
L5.3

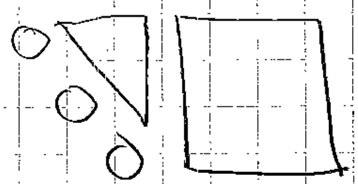




$\Theta(N)$ time
$\Theta(N^2)$ HW

Similar alg with pivoting (Leighton pp. 82-92)

Matrix inverse:



«Inverse bad numerically»

6.896
2/18/04
L5.4

Transitive closure of a digraph

Problem: Given a directed graph, for all pairs (i,j), determine if $\exists$ directed path from i to j.

Adjacency matrix

$A = (a_{ij})$ $\quad a_{ij} = \begin{cases} 1 & \text{if } i \to j \text{ is an edge} \\ 0 & \text{otherwise} \end{cases}$

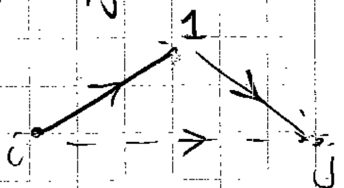
(Assume $a_{ii} = 1$ for simplicity)

Trans. closure $A^* = (a^*_{ij})$, $a^*_{ij} = 1$ iff $\exists$ path $i \xrightarrow{*} j$

Step 1: $\forall i,j$ s.t. $a_{i1}^{(0)} = 1$ and $a_{1j}^{(0)} = 1$, set $a_{ij}^{(1)} = 1$.

I.e., $a_{ij}^{(1)} \leftarrow a_{ij}^{(0)} \vee a_{i1}^{(0)} a_{1j}^{(0)}$

Shortcut node 1:



$A = A^{(0)} \to A^{(1)}$

Step 2 $\quad a_{ij}^{(2)} \leftarrow a_{ij}^{(1)} \vee a_{i2}^{(1)} a_{2j}^{(1)}$ $\qquad A^{(1)} \to A^{(2)}$

⋮

Step k $\quad \boxed{a_{ij}^{(k)} \leftarrow a_{ij}^{(k-1)} \vee a_{ik}^{(k-1)} a_{kj}^{(k-1)}}$ $\qquad A^{(k-1)} \to A^{(k)}$

6.896
2/18/04
L5.5

Theorem $A^{(N)} = A^*$

Proof. (Induction on k)

Claim: $a_{ij}^{(k)} = 1$ iff $\exists$ path $i \overset{*}{\Rightarrow} j$ in orig graph only going through nodes $1, 2, \ldots, k$.

($\Rightarrow$) Easy.

($\Leftarrow$) Case 1. $i \overset{*}{\Rightarrow} j$ through $1, \ldots, k-1$ $a_{ij}^{(k-1)} = 1$.
Case 2. $i \overset{*}{\Rightarrow} k$ through $1, \ldots, k-1$ $a_{ik}^{(k-1)} = 1$.
$k \overset{*}{\Rightarrow} j$ through $1, \ldots, k-1$ $a_{kj}^{(k-1)} = 1$. $\boxtimes$

$\Theta(N)$ time on $N \times N$ mesh.
Idea: Same computation as G.E.:

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)} a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}}$$

Also, shortest paths:

$$a_{ij} = \begin{cases} \text{weight of edge from } i \text{ to } j. \\ \infty \text{ if no edge.} \end{cases}$$

a^*_{ij} = weight of $\underbrace{\text{min-weight path}}_{\text{sum edge weights}}$ from i to j.

$$\boxed{a_{ij}^{(k)} = \min\left(a_{ij}^{(k-1)},\ a_{ik}^{(k-1)} + a_{kj}^{(k-1)}\right)}$$

Homework: min spanning tree.

Use thm: An edge $i \to j$ with weight a_{ij} belongs to MST iff $\nexists$ path from i to j with every edge having weight $< a_{ij}$.