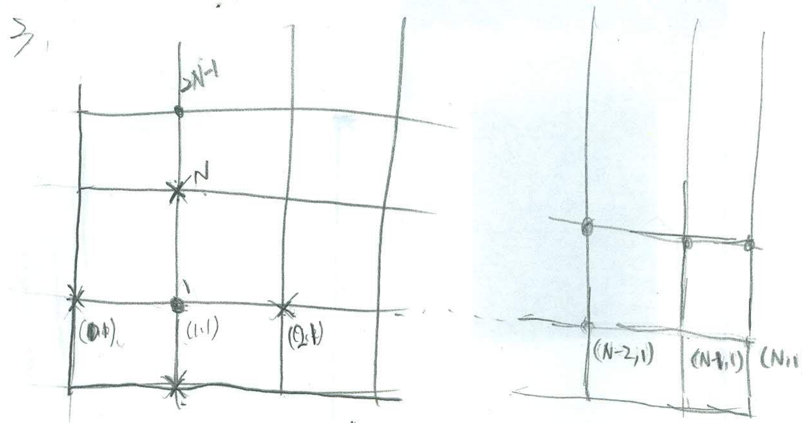
3.



$$(\Delta_h u)_{1,1} = \frac{1}{h^2}\begin{pmatrix} & 1 & \\ 1_{(=0)} & -4 & 1 \\ & 1_{(=0)} & \end{pmatrix}$$

$1 \sim N-1$

$N \sim 2N-2$

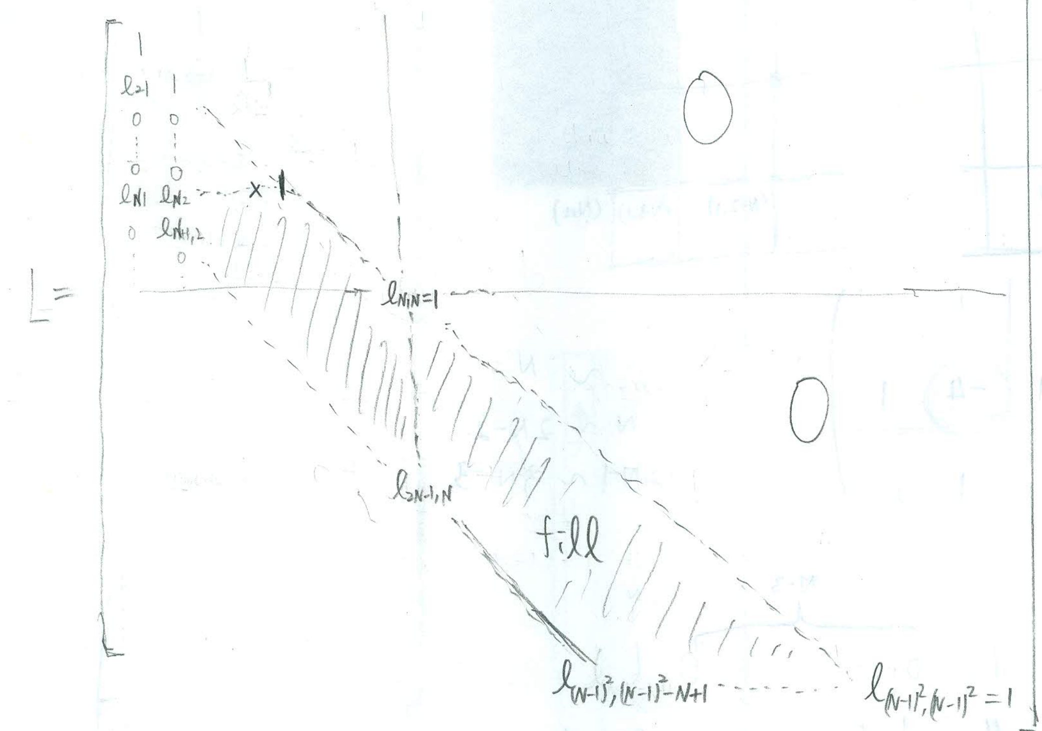
$2N-1 \sim 3N-3$

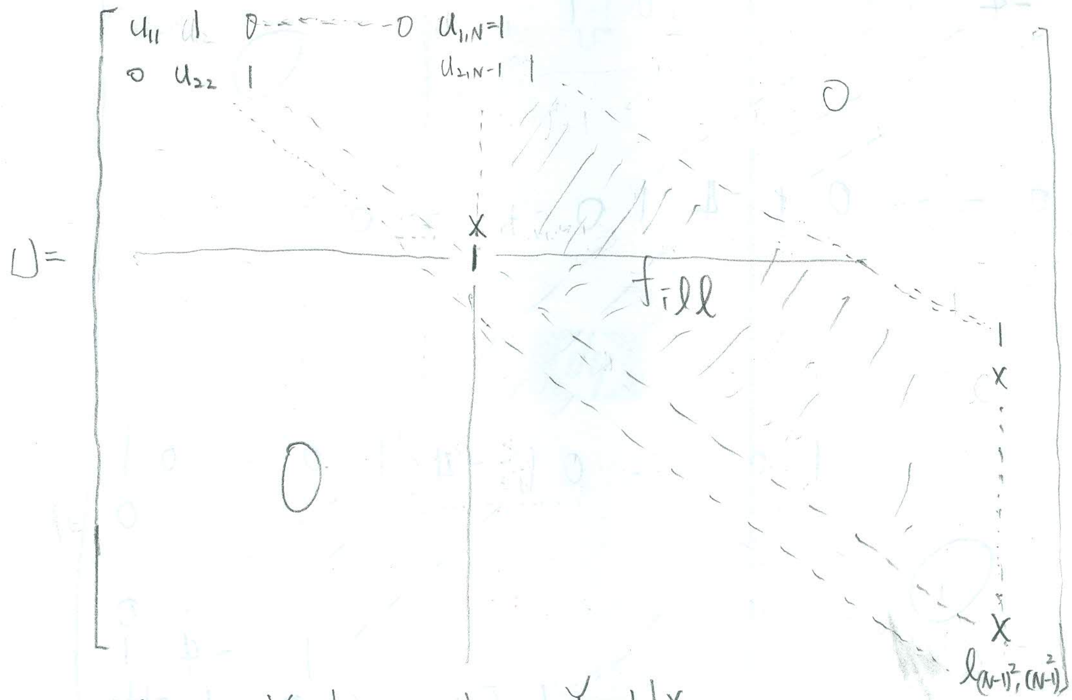
Let

$$A = \begin{bmatrix} -4 & 1 & \overbrace{0 \cdots 0}^{N-3} & \overset{N}{1} & & & & \\ 1 & -4 & 1 & 0 \cdots 0 & 1 & & & \\ 0 & \ddots & \ddots & \ddots & & \ddots & & O \\ \vdots & & & & & & & \\ 0 & & & & & & & \\ 1 & 0 \cdots 0 & 1 & -4 & 1 & 0 \cdots 0 & 1 & \\ & \ddots & & \ddots & \ddots & \ddots & 0 & \ddots \\ & & & & & & \vdots & \\ & & 1 & 0 \cdots 0 & 1 & -4 & 1 & 0 \cdots 0 \ \ 1 \\ & & & \ddots & & \ddots & \ddots & 0 \\ & O & & & & & & \vdots \\ & & & & & & & 0 \\ & & & & & 1 & -4 & 1 \\ & & & & 1 & 0 \cdots 0 & 1 & -4 \end{bmatrix}_{(N-1)^2 \times (N-1)^2}$$

(a)

$A = LU$, where





(1) solve $LY = b$, where $Y = UX$

need $N^3 + \cdots\cdots$

(2) solve $UX = Y$,

need $N^3 + \cdots\cdots$

by (1) (2) total need $2N^3 + \cdots\cdots \Rightarrow$ leading order is $2N^3$ #

(b) (1) Jacobi: $x^{(k)} = D^{-1}((L+U)x^{(k-1)} + b)$, $k = 1, 2, \ldots\ldots$

$$x_i^{(k)} = \left(-\sum_{\substack{j=1 \\ j\neq i}}^{n} (a_{ij} x_j^{(k-1)}) + b_i\right) \Big/ a_{ii} \quad \text{—— } (*)$$

∵ "L+U" has $2((N-1)^2-1+(N-1)^2-N+1)$ nonzero entries

∴ $(L+U)x^{(k-1)}$ need $2((N-1)^2-1+(N-1)^2-N+1)$ multip.

∵ "D^{-1}" has $(N-1)^2$ nonzero entries

∴ $D^{-1}((L+U)x^{(k-1)})$ need $4N^2+N^2+\cdots = 5N^2+\cdots$

⇒ leading order is $5N^2$ #

(2) Gauss seidel: $x^{(k)} = (D-L)^{-1}(Ux^{(k-1)} + b)$

$$x_i^{(k)} = \left(-\sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} + b\right) \Big/ a_{ii}$$

由 G-S 的 $x_i^{(k)}$ 的公式與 Jacobi 的 $x_i^{(k)}$ 的公式看來
所需的乘法是一樣.

(3). SOR: $x^{(k)} = (D-wL)^{-1}\left[(1-w)D+wU\right]x^{(k-1)} + w(D-wL)^{-1}b$

$$x_i^{(k)} = \frac{(1-w)x_i^{(k-1)} + w\left(-\sum_{j=1}^{i-1} a_{ij} x_i^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_i^{(k-1)} + b_i\right)}{a_{ii}}$$

多一個乘法　　多一個乘法

由 SOR 的 $x_i^{(k)}$ 的公式與 Jacobi 的 $x_i^{(k)}$ 的公式看來,
SOR 多了 $2N^2$ 個乘法 ⇒ leading order 是 $7N^2$ #

(c) As we explained in class.

If T has no Jordan block (i.e. diagonalizable)

Then $\|T^k x\| \leq \rho^k(T)\|x\|$

If T has nontrivial Jordan block,

then $\|T^k x\| \leq O(k) \cdot \rho^k(T)\|x\|$

In any case, we have $\|T^k x\| \leq (\rho(T)+\varepsilon)^k \|x\|$

Now $\rho(T_J)$, $\rho(T_g)$, $\rho(T_\omega) < 1$

So we have $\|T^k x\| \leq \left(\frac{\rho(T)+1}{2}\right)^k \|x\|$

Iteration and multiplication count:

Jacobi & G-S : $\frac{(\rho(T)+1)}{2} \cong 1 - Ch^2$

Iteration $= \frac{\log(h^2/1)}{\log(1-Ch^2)} \sim \frac{-\log h}{-Ch^2} = O(N^2 \log N)$

Multiplication $= O(N^2) \cdot O(N^2 \log N) = O(N^4 \log N)$

Slower than LU or Gauss elimination

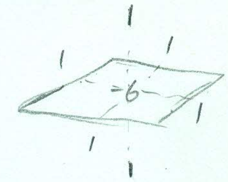
(d) SOR with $\omega = \omega^*$

$$\frac{\rho(T_\omega)+1}{2} \cong 1 - Ch$$

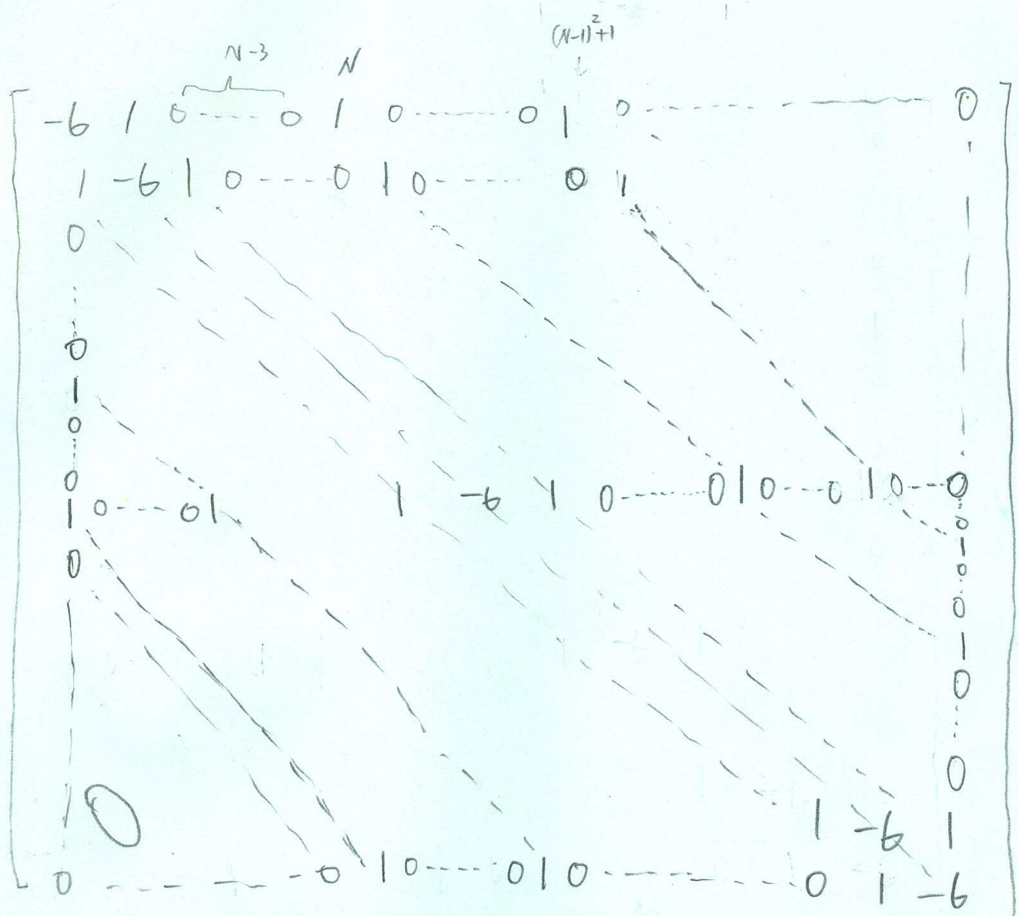
Iteration $= \dfrac{\log h^2}{\log(1-Ch)} = O(N \log N)$

multiplication $= O(N^3 \log N)$ (faster than LU)

(e) 3D matrix



A=



LU decomposition: $N^3 \cdot (N^2)^2 = N^7$ multiplications

Jacobi and Gauss: $O(N^3)$ multiplication each iteration

Total multiplication $= O(N^3) \cdot N^2 \log N = O(N^5 \log N)$

SOR with $\omega = \omega^*$

Total multiplication $= O(N^3) \cdot O(N \log N) = O(N^4 \log N)$