MAT 228A Notes

Sam Fleischer

October 20, 2016

1 Recall From Last Time

In a 2D grid, standard stensil:

$$\begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix} \tag{1}$$

When re-ordered in a vector, Laplacian matrix has -4 on the diagonal and 1 on the $1st$ and $n$thsup- and super-diagonals. In a 3D grid, the matrix has -6 on the diagonal and 1 on the $1st$, nth, and $\left(n^2\right)$th sub- and super-diagonals.

Using Gaussian Elimination gives fill-in between the bands. What is the operation count? Suppose a diagonal matrix with band-width b on the sub-diagonals and band-width a on the super-diagonals ($N \times N$ matrix). Work to factor? $\mathcal{O}(abN)$.

In 2D, $N = n^2$, $a = b = n$. So work is $\mathcal{O}\left(n^4\right) = \mathcal{O}\left(N^2\right)$. This is less than $\mathcal{O}\left(N^3\right)$ (work to factor unstructured matrix), but higher than the optimal. Back/forward solves is $\mathcal{O}(Nn) = \mathcal{O}\left(n^3\right) = \mathcal{O}\left(N^{\frac{3}{2}}\right)$.

In 3D, $N = n^3$, $a = b = n^2$. So work is $\mathcal{O}\left(n^7\right) = \mathcal{O}\left(N^{\frac{7}{3}}\right)$. This is less than $\mathcal{O}\left(N^3\right)$ but higher than the optimal. Back/forward solves is $\mathcal{O}\left(Nn^2\right) = \mathcal{O}\left(n^5\right) = \mathcal{O}\left(N^{\frac{5}{3}}\right)$.

2 Memory Allocation in 3D

We need to

- store the original matrix ($\mathcal{O}(N)$).
- store the factored matrix ($\mathcal{O}\left(N^{\frac{5}{3}}\right)$)

What does this mean in terms of computers we actually have? How big is this for $n = 100$ so we have a $100 \times 100 \times 100$ grid. So $N = 1{,}000{,}000$. $n^5 = 10^{10} = 10{,}000{,}000{,}000$. This takes about 20GB to store.. my computer only has 16GB.

3 Fast Fourier Transform

We know the eigenvalues of the 1D problem ($\sin(k\pi x)$).

Suppose

$$A\vec{u} = \vec{b} \tag{2}$$

Q is a matrix of eigenvectors, Λ diagonal matrix of eigenvalues.

$$AQ = Q\Lambda \qquad \Longrightarrow \qquad A = Q\Lambda Q^T \tag{3}$$

So

$$Q\Lambda Q^T\vec{u} = \vec{b} \implies Q^T\vec{u} = \Lambda^{-1}Q^T\vec{b} \implies \vec{u} = Q\Lambda^{-1}Q^T\vec{b} \tag{4}$$

Multiplication by Q^T can be done using FFT where work is $\mathcal{O}(N \log N)$. Multiplication by Q is inverse FFT where work is $\mathcal{O}(N \log N)$. This is a very special solver (rectangular domain).

4 Block Matrix

In 2D we have

$$A = \frac{1}{h^2} \begin{pmatrix} T & I & & & \\ I & T & I & & \\ & \ddots & \ddots & \ddots & \\ & & I & T & I \\ & & & I & T \end{pmatrix} \tag{5}$$

where

$$T = \begin{pmatrix} -4 & 1 & & & \\ 1 & -4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -4 & 1 \\ & & & 1 & -4 \end{pmatrix} \tag{6}$$

and I is the identity. Bob *thinks* we can exploit the structure of A to get work $\mathcal{O}\left(n^{\frac{3}{2}}\right)$ (nested dissection).

5 Convergence in 2D

In 1D, we did 2 norm (by knowing the eigenvectors) and max-norm (by just finding the inverse).

In 2D,

$$A\vec{e}^h = -\vec{\tau}^h \tag{7}$$

where $\vec{e}^h$ is the error and $\vec{\tau}^h$ is the LTE (local truncation error). For 2-norm, we use the same idea as in 1D - we know the eigenvectors (discrete eigenfunctions)! Their form is

$$u_{ij}^{km} = \sin(k\pi x_i)\sin(m\pi y_j) \tag{8}$$

The eigenvalues are

$$\lambda^{km} = \frac{2}{h^2}(\cos(k\pi h) + \cos(m\pi h) - 2) \tag{9}$$

Use these to find the spectral radius (and then the 2-norm).

We will use 3 steps for proving convergence max-norm in 2D.

1. Discrete maximum principle. Think of the operator L

$$L\vec{u} = \vec{f} \tag{10}$$

   If $Lu \geq 0$ for some region, then the maximum value of u is attained on the boundary (same idea as in 1D - if a function is concave up, its maximum is one of the boundaries). Similar statement about $Lu \leq 0$ (concave down).

2. If u is a discrete function, $u = 0$ on the boundary (lets just say discrete unit square), then

$$\|u\|_\infty = \frac{1}{8}\|Lu\|_\infty \tag{11}$$

   How can we use this? We can relate $\vec{e}$ and $\vec{\tau}$ by

$$L\vec{e} = -\vec{\tau} \tag{12}$$

   Using #2,

$$\|\vec{e}\|_\infty \leq \frac{1}{8}\|L3\|_\infty = \frac{1}{8}\tau_\infty = \mathcal{O}(h^2) \tag{13}$$

$$\frac{1}{h^2}(u_{i-1,j}+u_{i,j-1}-4u_{i,j}+u_{i+1,j}+u_{i+1,j+1}) \geq 0 \implies \frac{1}{4}u_{i-1,j} \tag{14}$$

If u_{ij} ia a max, than all neighbors must be equal. The function is constant.

Idea of (2.): If $L\vec{u}=f$, suppose u is zero on the boundary.

$$w_{ij} = \frac{1}{4}\left[\left(x_i - \frac{1}{2}\right)^2 + \left(y_j - \frac{1}{2}\right)^2\right] \tag{15}$$

Lw = 1. So

$$L(u + w\|\infty\|) = f + \|f\|_\infty \geq 0 \tag{16}$$

Finally we know $\max(u_w\|f\|_\infty)$ is on the boundary. So $u \leq u + \|f\|_\infty w \leq \max(w)\|f\|_\infty = \frac{1}{8}\|f\|_\infty$.