Lecture 11:

Recall: Goal: To compute $\vec{y} \overset{def}{=} \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{2m-1} \end{pmatrix} = F_n \vec{x} \overset{def}{=} F_n \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{2m-1} \end{pmatrix}$ $(n = 2m)$

Summary of FFT

Step 1: Split $\vec{x}$ into $\vec{x}\,' = \begin{pmatrix} x_0 \\ x_2 \\ \vdots \\ x_{2(m-1)} \end{pmatrix}$ and $\vec{x}\,'' = \begin{pmatrix} x_1 \\ x_3 \\ \vdots \\ x_{2m-1} \end{pmatrix}$

Step 2: Compute $\underset{\mathcal{O}(m^2)}{\vec{y}\,' = F_m \vec{x}\,'}$ and $\underset{\mathcal{O}(m^2)}{\vec{y}\,'' = F_m \vec{x}\,''}$, where $F_m = \underbrace{\frac{n}{2} \times \frac{n}{2}}_{}$ matrix

$$\frac{n}{2} \times \frac{n}{2} = m \times m$$

Step 3: Compute:

$$\begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{pmatrix} = \vec{y}\,' + \overbrace{\begin{pmatrix} \omega_n^0 \\ \omega_n^1 \\ \vdots \\ \omega_n^{m-1} \end{pmatrix}}^{\vec{\omega}_m} \otimes \vec{y}\,'' \quad \mathcal{O}(m)$$

and

$$\begin{pmatrix} y_m \\ y_{m+1} \\ \vdots \\ y_{2m-1} \end{pmatrix} = \vec{y}\,' - \overbrace{\begin{pmatrix} \omega_n^0 \\ \omega_n^1 \\ \vdots \\ \omega_n^{m-1} \end{pmatrix}}^{\vec{\omega}_m} \otimes \vec{y}\,'' \quad \mathcal{O}(m)$$

where:

$$\begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \otimes \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} = \begin{pmatrix} v_1 w_1 \\ v_2 w_2 \\ \vdots \\ v_n w_n \end{pmatrix}$$

Remark: Computational cost : $\mathcal{O}(m^2) + \mathcal{O}(m)$ (multiplication + addition)

$\quad \simeq \mathcal{O}(m^2)$

Computational cost for FFT: Assume $n = 2^\ell$.

Let C_m = computational cost of F_m. Then $C_1 = 1$.

Claim: $C_{2m} = 2C_m + 3m$

Proof: **Step 2:** $\vec{y}\,' = F_m \vec{x}\,'$, $\vec{y}\,'' = F_m \vec{x}\,''$ $(= 2C_m)$

Step 3:
$$y_j = \vec{y}_j\,' + w_n^j \vec{y}_j\,''$$
$$y_{j+m} = \vec{y}_j\,' - w_n^j \vec{y}_j\,''$$
for $j = 0, 1, 2, \ldots, m-1$

(= 1 multiplication + 1 addition + 1 substraction)

$\Downarrow$

Total : $3m$

$\therefore\ C_{2m} = 2C_m + 3m.$

Now, $n = 2^{\ell}$.

$$\therefore \quad C_{2^{\ell}} = 2C_{2^{\ell-1}} + 3 \cdot 2^{\ell-1}.$$

$$\therefore \quad 2^{-\ell} C_{2^{\ell}} = 2^{-(\ell-1)} C_{2^{\ell-1}} + \frac{3}{2} = 2^{-(\ell-2)} C_{2^{\ell-2}} + 2\left(\frac{3}{2}\right)$$

$$= \vdots$$

$$= 2^{0} C_{2^{0}} + \ell\left(\frac{3}{2}\right) = 1 + \frac{3}{2}\ell$$

$$\therefore \quad C_{2^{\ell}} = \underbrace{2^{\ell}}_{n} + \frac{3}{2} \overbrace{\ell}^{\log_2 n} \underbrace{2^{\ell}}_{n} = n + \frac{3}{2} n \log_2 n = O(n \log_2 n)$$

Butterfly diagram (Algorithmic visualization)

Consider F_4 (4x4 matrix). Goal: compute $\begin{pmatrix}y_0\\y_1\\y_2\\y_3\end{pmatrix} = F_4\begin{pmatrix}x_0\\x_1\\x_2\\x_3\end{pmatrix}$

[Recall: $\vec{y}_e = \vec{y}\,' = F_m\vec{x}\,'$. Denote $\vec{x}\,' := \vec{x}_e$, $\vec{x}\,'' = \vec{x}_o$]

$\vec{y}_o = \vec{y}\,'' = F_m\vec{x}\,''$

$$\begin{tikzcd}
\vec{y}_e := F_2\vec{x}_e = \begin{pmatrix}x_0\\x_2\end{pmatrix} = \begin{pmatrix}\cdot\\\cdot\end{pmatrix} \arrow[r] \arrow[dr] & \begin{pmatrix}y_0\\y_1\end{pmatrix} \\
\vec{y}_o := F_2\vec{x}_o = \begin{pmatrix}x_1\\x_3\end{pmatrix} = \begin{pmatrix}\cdot\\\cdot\end{pmatrix} \arrow[ur, "\vec{\omega}_4"'] \arrow[r, "-\vec{\omega}_4"'] & \begin{pmatrix}y_2\\y_3\end{pmatrix}
\end{tikzcd}$$

$$\vec{\omega}_4 = \begin{pmatrix}\omega_4^0\\\omega_4^1\end{pmatrix}$$

Diagram means:

$$\left.\begin{aligned}\begin{pmatrix}y_0\\y_1\end{pmatrix} &= \vec{y}_e + \vec{\omega}_4 \otimes \vec{y}_o\\ \begin{pmatrix}y_2\\y_3\end{pmatrix} &= \vec{y}_o - \vec{\omega}_4 \otimes \vec{y}_o\end{aligned}\right\}$$

Depend on $\vec{y}_o$ and $\vec{y}_e$.

For $F_2\vec{x}_e$: $1 = F_1\vec{x}_{ee} = (x_0)$

$1 = F_1\vec{x}_{eo} = (x_2)$

$\vec{\omega}_2^0$ $-\vec{\omega}_2^0$

$\begin{pmatrix}\cdot\\ \cdot\end{pmatrix} = F_2\vec{x}_e = \vec{y}_e$

For $F_2\vec{x}_o$: $1 = F_1\vec{x}_{oe} = (x_1)$

$1 = F_1\vec{x}_{oo} = (x_3)$

$\vec{\omega}_2^0$ $-\vec{\omega}_2^0$

$\begin{pmatrix}\cdot\\ \cdot\end{pmatrix} = F_2\vec{x}_o = \vec{y}_o$

Remark:

$\vec{x}_e = \begin{pmatrix} x_0 \\ x_2 \end{pmatrix}$, $\vec{x}_o = \begin{pmatrix} x_1 \\ x_3 \end{pmatrix}$

$\vec{x}_{ee} = (x_0)$ $\vec{x}_{oe} = (x_1)$

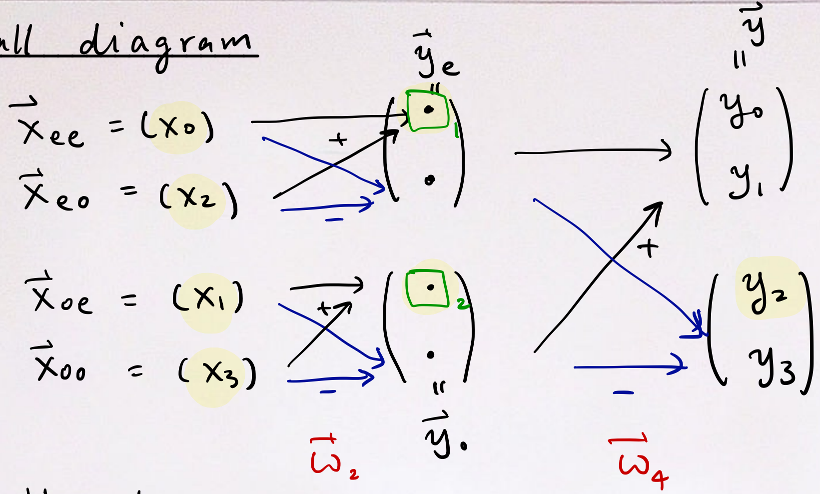
$\vec{x}_{eo} = (x_2)$ $\vec{x}_{oo} = (x_3)$

$$\vec{y}_e = \begin{pmatrix} x_0 + \vec{\omega}_2^0 \otimes (x_2) \\ x_0 - \vec{\omega}_2^0 \otimes (x_2) \end{pmatrix} = \begin{pmatrix} x_0 + x_2 \\ x_0 - x_2 \end{pmatrix}$$

$$\vec{y}_o = \begin{pmatrix} x_1 + \vec{\omega}_2^0 \otimes (x_3) \\ x_1 - \vec{\omega}_2^0 \otimes (x_3) \end{pmatrix} = \begin{pmatrix} x_1 + x_3 \\ x_1 - x_3 \end{pmatrix}$$

$\vec{\omega}_2^0 = (\omega_2^0) = (1)$

Overall diagram



(Butterfly diagram)

Using the diagram, find y_2.

$$y_2 = \square_1 - (\vec{\omega}_4)_0 \square_2 \quad ; \quad \square_1 = x_0 + x_2$$

$$= \square_1 - \square_2 \qquad \square_2 = x_1 + x_3$$

$$\therefore \quad y_2 = x_0 + x_2 - (x_1 + x_3)$$

Numerical Spectral method for PDE

Consider : $u = u(x,t)$ where :

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad \text{for} \quad t > 0 \quad \text{and} \quad x \in (0, 2\pi) \ni$$

$$u(x,0) = h(x) \quad (\text{initial condition})$$

$$u(0,t) = u(2\pi, t) \quad (\text{Boundary condition})$$

Divide $[0, 2\pi]$ into N partitions such that $x_j = j\left(\frac{2\pi}{N}\right)$ $\left(h = \frac{2\pi}{N}\right)$

Discretize PDE by:

$$\frac{u(x_k, t_{j+1}) - u(x_k, t_j)}{\Delta t} = \frac{u(x_{k+1}, t_j) - 2u(x_k, t_j) + u(x_{k-1}, t_j)}{h^2} \qquad (*)$$

for $k = 0, 1, 2, \ldots, N-1$

Denote $u(x_k, t_j)$ by $u_{k,j}$ and let $\vec{u}_j = \begin{pmatrix} u_{1,j} \\ u_{2,j} \\ \vdots \\ u_{k,j} \\ \vdots \end{pmatrix}$.

Then: $(*)$ becomes:

$$\frac{1}{\Delta t}\left(\vec{u}_{j+1} - \vec{u}_j\right) = D\vec{u}_j \quad \text{where } D = \begin{pmatrix} -2 & 1 & & & & 1 \\ 1 & -2 & 1 & & 0 & \\ & 1 & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ 0 & & & \ddots & & 1 \\ 1 & & & & 1 & -2 \end{pmatrix} / h^2$$

Now, let $\vec{u}_j = \sum_{k=0}^{N-1} a_k(j)\, e^{\overrightarrow{ikx}}$

Then, we get: $\frac{1}{\Delta t}\left(\sum_{k=0}^{N-1}\left(a_k(j+1)\, e^{\overrightarrow{ikx}} - a_k(j)\, e^{\overrightarrow{ikx}}\right)\right) = \sum_{k=0}^{N-1} a_k(j)\, \underbrace{D\, e^{\overrightarrow{ikx}}}_{\lambda_k^2\, e^{\overrightarrow{ikx}}}$

$$\Rightarrow \sum_{k=0}^{N-1}\left(\frac{a_k(j+1) - a_k(j)}{\Delta t}\right) e^{\overrightarrow{ikx}} = \sum_{k=0}^{N-1} a_k(j)\, \lambda_k^2\, e^{\overrightarrow{ikx}}$$

Comparing coefficients, we get:

$$\frac{a_k(j+1) - a_k(j)}{\Delta t} = \lambda_k^2 \, a_k(j)$$

$$\Rightarrow \quad a_k(j+1) = \left(1 + \Delta t \, \lambda_k^2\right) a_k(j) \quad \text{with}$$

$$a_k(0) = c_k \quad \text{where}$$

$$\begin{pmatrix} h(x_0) \\ h(x_1) \\ \vdots \\ h(x_{N-1}) \end{pmatrix} = \sum_{k=0}^{N-1} c_k \, e^{\overrightarrow{ikx}}$$

Iterative method for solving (huge) linear system

Recall: Numerical spectral method handles periodic functions.

Consider: (*) $\dfrac{d^2u}{dx^2} = f$. with $u(0) = A$ and $u(1) = B$

Recall that: $u''(x) \approx \dfrac{u(x+h) - 2u(x) + u(x-h)}{h^2}$ for small h

Partition $[0, 1]$ into $x_j = jh$ where $h = \dfrac{1}{n+1}$

Then: (*) is discretized as:

$$\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} = f(x_i)$$ or

$$A \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = \begin{pmatrix} f(x_1) - \frac{A}{h^2} \\ f(x_2) \\ \vdots \\ f(x_n) - \frac{B}{h^2} \end{pmatrix}$$ where

$$A = \frac{1}{h^2}\begin{pmatrix} -2 & 1 & & \\ 1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{pmatrix}$$ and $u(0) = A$, $u(1) = B$

Question: How to solve BIG linear system?

Method 1: Gaussian elimination

Computational cost : $O(n^3)$

Solution : exact

Method 2: LU factorization.

Decompose $A = LU$ (If A is symmetric positive-def., then $A = LL^T$ by Cholesky decomposition)

L: lower tri, U: upper tri

$LU\vec{x} = \vec{b}$ by solving $\left\{\begin{array}{l} L\vec{y} = \vec{b} \\ U\vec{x} = \vec{y} \end{array}\right.$ easy.

Computational cost : $O(n^3)$

Solution : exact.

Goal: Develop iterative method: find a sequence $\vec{x}_0, \vec{x}_1, \ldots$ such that $\vec{x}_k \to \vec{x}^* =$ solution of $A\vec{x} = \vec{f}$ as $k \to \infty$.

Remark: We can stop when error is small enough.

Method: Splitting method

Consider a linear system $A\vec{x} = \vec{f}$ where $A \in M_{n\times n}$ (n is BIG)

Split A as follows: $A = N + (A - N) = N - \underbrace{(N - A)}_{P} = N - P$

Then: $A\vec{x} = \vec{f} \iff (N - P)\vec{x} = \vec{f} \iff N\vec{x} = P\vec{x} + \vec{f}$

Develop an iterative scheme as follows:

$$(*) \quad N\vec{x}^{\,n+1} = P\vec{x}^{\,n} + \vec{f}$$

If $\{\vec{x}^{\,n}\}_{n=1}^{\infty}$ converges, then it converges to the sol. $\vec{x}^*$ of $A\vec{x} = \vec{f}$

Remark:

- N must be simple enough so that $(\ast)$ can be solved easily
- Will $\{\vec{x}^{\,n}\}_{n=1}^{\infty}$ converge? Suitable N.