DNAP ASSIGNMENT-3

AMIT KUMAR

7. We consider the radix-2 fft algorithm, then the algorithm actually divides the whole series of DFT into halves containing even and odd terms & work our way back.

this algorithm of radix 2 works only for step size $= 2^m$ $(m \in \mathbb{N})$, and we work out the complexity for them only.

say $\hat{w}_q$ represent the DFT for the frequency point q

$$\hat{w}_q = \frac{1}{\sqrt{n}} \sum_{p=0}^{2^m - 1} \exp\left(\frac{-i 2\pi q p}{n}\right) w_p$$

where $\{w_p\}$ are data pts.

We can divide the series as:

$$\frac{1}{\sqrt{n}} \sum_{p=0}^{w_2 - 1} w_{2p} \left(\exp\left(\frac{-i 2\pi q \cdot p}{n/2}\right)\right)$$

$$+ \sum_{p=0}^{n/2 - 1} w_{2p+1} \exp\left(\frac{-i 2\pi q}{n}\right) \exp\left(\frac{i 2\pi q p}{n/2}\right)$$

we can write this as:

$$\frac{1}{\sqrt{n}} \left\{ DFT\left(\frac{n}{2} \text{ points (even points)}\right) + \varphi^q DFT\left(\frac{n}{2} \text{ points (odd points)}\right) \right\}.$$

where $\varphi = \exp\left(\frac{-i 2\pi}{n}\right)$

Again we can divide the $w/2$ pts into $w/4$ pts each.

so the chain goes as:

$$DFT_n = DFT^{e}_{w/2} + \varphi^{e} DFT^{o}_{w/2}$$

$$= DFT^{ee}_{w/4} + DFT^{eo}_{w/4} (\varphi^{(e)} + (DFT^{oe}_{w/4} + DFT^{oo}_{w/4}) \varphi^{(1)}$$

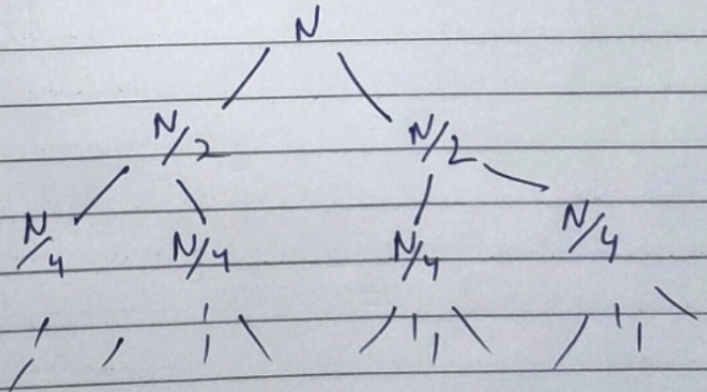
ie. we keep multiplying phases & computing DFTs.

(*) How long can this split goes on?

(*) But DFT of single number is that number it self. what true by remains is multiplying the phase factor & adding them up.

what we need to do is to this recursively multiply phases & add up.

so



At each single step, what matters are the no of operations we're doing at each step.

So at each step we have $O(N)$ operations complexity & there are $m = \log_2 N$ steps to compute them.

So net complexity of algorithm given by

$O(N \log_2 N)$ [By product of complexities.]

Assignment 3

Amit Kuamr

Click here for my GitHub link: https://github.com/amitkumar20ms0012/Computational_Physics_A.git