The Fast Fourier Transform (FFT)

Gilad Lerman
Notes for Math 5467

1 FFT for signals

We recall that the DFT of a signal $x_M = (x(0), \ldots, x(M-1)) \in \mathbb{C}^M$ has the form

$$\hat{x}_M(n) = \sum_{k=0}^{M-1} x_M(k) e^{\frac{-2\pi i k n}{M}} = \sum_{k=0}^{M-1} x_M(k) W_M^{kn},$$

where

$$W_M = e^{\frac{-2\pi i}{M}}.$$

It can be evaluated directly in $O(M^2)$ computations. Indeed, for each $n = 0, \ldots, M-1$, there are $M-1$ additions and M multiplications involved in computing $\hat{x}(n)$. In addition, there are $M-2$ multiplications involved in computing $W_M^2, \ldots, W_M^{M-1}$ and a constant C_M required to compute W_M, so $(2M-1)M + M - 2 + C_M = O(M^2)$ computations. We will not distinguish here between the cost of computing products or additions of numbers, even though they are complex.

Cooley and Tukey have proposed in 1965 an $O(M \log M)$ algorithm for computing the DFT of a signal in $\mathbb{C}^M$, where $M = 2^m$ (see [2, 3, 1] for the history of this algorithm). It applies a "divide and conquer" strategy by dividing recursively the signal into two signals with half of that length containing its even and odd indices respectively. That is, the basic step of their algorithm decomposes $X_{2L} \in \mathbb{C}^{2L}$, where $L = M/2, M/4, \ldots, 1$, into the signals X_L^{even} and X_L^{odd} in $\mathbb{C}^L$ as follows:

$$\begin{aligned} X_L^{\text{even}}(\ell) &= X_{2L}(2\ell),\ \ell = 0, \ldots, L-1, \\ X_L^{\text{odd}}(\ell) &= X_{2L}(2\ell+1),\ \ell = 0 \ldots, L-1. \end{aligned}$$

This division has the following property:

Lemma 1.1. *If $L \in \mathbb{N}$, then for all $n = 0, \dots, L-1$,*

$$\hat{X}_{2L}(n) = \hat{X}_L^{\text{even}}(n) + \hat{X}_L^{\text{odd}}(n) W_{2L}^n, \tag{1}$$

and

$$\hat{X}_{2L}(n+L) = \hat{X}_L^{\text{even}}(n) - \hat{X}_L^{\text{odd}}(n) W_{2L}^n. \tag{2}$$

We denote the number of operations needed to calculate the DFT of a signal in $\mathbb{C}^L$ by $\#L$. A direct consequence of Lemma 1.1 is the following proposition.

Lemma 1.2. *If $L \in \mathbb{N}$, then given the numerical value of W_{2L}*

$$\#(2L) \leq 2 \cdot \#L + 4L.$$

This lemma now leads to the main theorem.

Theorem 1.1. *If $M = 2^m$, where $m \in \mathbb{Z}$, $m \geq 0$, and C_M is the number of operations required to calculate W_M, then*

$$\#M \leq 2M \log_2 M + C_M. \tag{3}$$

We will prove the above propositions in order.

Proof of Lemma 1.1. For $n = 0, \dots, L-1$

$$\begin{aligned}
\hat{X}_{2L}(n) &= \sum_{k=0}^{2L-1} X_{2L}(k) W_{2L}^{kn} \\
&= \sum_{\substack{k=0 \\ k=2\ell}}^{2L-1} X_{2L}(k) W_{2L}^{kn} + \sum_{\substack{k=0 \\ k=2\ell+1}}^{2L-1} X_{2L}(k) W_{2L}^{kn} \\
&= \sum_{\ell=0}^{L-1} X_{2L}(2\ell) W_{2L}^{2\ell n} + \sum_{\ell=0}^{L-1} X_{2L}(2\ell+1) W_{2L}^{2\ell n} W_{2L}^n \\
&= \sum_{\ell=0}^{L-1} X_L^{\text{even}}(\ell) W_{2L}^{2\ell n} + \sum_{\ell=0}^{L-1} X_L^{\text{odd}}(\ell) W_L^{kn} W_{2L}^n \\
&= \hat{X}_L^{\text{even}}(n) + \hat{X}_L^{\text{odd}}(n) W_{2L}^n.
\end{aligned} \tag{4}$$

In order to obtain (2) from (4), we use the fact that $\hat{X}_L^{\mathrm{even}}$ and $\hat{X}_L^{\mathrm{odd}}$ are L-periodic, i.e., for $n = 0, \ldots, L-1$:

$$\hat{X}_L^{\mathrm{even}}(n+L) = \hat{X}_L^{\mathrm{even}}(n) \quad \text{and} \quad \hat{X}_L^{\mathrm{odd}}(n+L) = \hat{X}_L^{\mathrm{odd}}(n)$$

as well as

$$W_{2L}^{L} = -1\,.$$

□

Proof of Lemma 1.2. We compute $\hat{X}_{2L}(n)$, $n = 0, \ldots, 2L-1$, following the formulas of Lemma 1.2. We note that we need to compute $\hat{X}_L^{\mathrm{even}}(n)$ and $\hat{X}_L^{\mathrm{odd}}(n)$ for all $n = 0, \ldots, L-1$, and that this computation requires $2\cdot\#L$ operations. We then need to compute $W_{2L}^2, \ldots, W_{2L}^{L-1}$ via $L-2$ multiplications. We also have L multiplications (computing $\hat{X}_L^{\mathrm{odd}}(n) W_{2L}^n$, for $n = 0, \ldots, L-1$), $L-1$ additions of (1) and $L-1$ subtractions of (2). Therefore,

$$\#(2L) \leq 2\cdot\#L + 4L - 2 \leq 2\cdot\#L + 4L\,. \tag{5}$$

□

Proof of Theorem 1.1. The idea of the algorithm is to take the signal of length $M = 2^m$ and recursively divide it into its odd and even components of half length. We assume that $W_M \equiv W_{2^m}$ is given. We then compute the FFT in bottom-up procedure, applying (1) and (2).

We note that $W_{M/2}$, $W_{M/4}$, ..., W_2, W_1 are computed during this recursive procedure and accounted for in the estimate provided by Lemma 1.2. Indeed, at the last level of the computation, where the FFT of the full signal (of length M) is computed according to the FFTs of the even and odd signals of lengths $M/2$ (following (1) and (2)), the algorithm computes W_M^n, $n = 0, \ldots, M/2-1$. The values of W_M^n, $n = M/2, \ldots, M-1$ are obtained by multiplying by -1 those values respectively (since $W_M^{M/2} = -1$). Moreover, we note that

$$W_{M/2^q} = W_M^{2^q} = \quad \text{for} \quad q = 1, \ldots, m.$$

Therefore, assuming that W_M has been computed, the algorithm has computed the values of $W_{M/2^q}$ for $1 \leq q \leq m$ and they have been accounted in the operation count of Lemma 1.2 (for $L = M/2$).

Next, we prove (3). Given the value of W_M (costing C_M operations), we only need to verify that

$$\#M \leq 2M \log_2 M. \tag{6}$$

We prove it by induction, which we relate to the general scheme of the algorithm as follows. For arbitrarily fixed $M = 2^m$, we let $q = 0, \ldots, m$ denote the different levels of the algorithm from bottom to top. That is, $q = 0$ is the bottom level of M signals of length 1 and $q = m$ is the top level with one signal of length M. So the qth step of the induction estimates the number of operation for any arbitrary signal of length 2^q out of the $M/2^q$ signals at level q (so eventually for level $q = m$ (7) is obtained). That is, we will show that

$$\#L \leq 2L \log_2 L \quad \text{for} \quad L = 2^q, \; q = 0, \ldots, m. \tag{7}$$

For $q = 0$, $L = 1$ and

$$\hat{x}_1(0) = x_1(0).$$

No operations are needed to compute the DFT in that case and indeed $2 \cdot L \log_2 L = 2 \cdot 1 \cdot \log_2 1 = 0$.

Assume that the formula is true for $L = 2^{q-1}$. We will verify it for $L' = 2L = 2^q$. By applying Lemma 1.1 (while using the remark above that W_{2^q} is known for all $q = 0, \ldots, m$ since W_M is known) and the induction assumption we obtain that

$$\begin{aligned} \#L' &= \#(2L) \leq 2\#L + 4L \\ &= 2 \cdot 2 \cdot L \log_2 L + 4L = 4L(\log_2 L + 1) \\ &= 4 \cdot 2^{q-1} \cdot (q - 1 + 1) = 2 \cdot 2^q \cdot q \\ &= 2 \cdot L' \cdot \log_2 L' \end{aligned}$$

□

2 FFT for Images

We recall that if $x \in \mathbb{C}^{M_1 \times M_2}$ indexed by $x(k_1, k_2)$, where $0 \leq k_1 \leq M_1 - 1$ and $0 \leq k_2 \leq M_2 - 1$, then

$$\hat{x}(n_1, n_2) = \sum_{k_1=0}^{M_1-1} \sum_{k_2=0}^{M_2-1} x(k_1, k_2) e^{-2\pi i(\frac{k_1 n_1}{M_1} + \frac{k_2 n_2}{M_2})}$$

We denote

$$\hat{x}(n_2 \mid k_1) = \sum_{k_2=0}^{M_2-1} x(k_1, k_2) e^{-\frac{2\pi i k_2 n_2}{M_2}},$$

and note that

$$\hat{x}(n_1, n_2) = \sum_{k_1=0}^{M_1-1} \hat{x}(n_2 \mid k_1) e^{-\frac{2\pi i k_1 n_1}{M_1}}$$

Therefore, in order to compute $\hat{x}(n_1, n_2)$, for $n_1 = 0, \ldots, M_1 - 1$ and $n_2 = 0, \ldots, M_2 - 1$, we compute $\hat{x}(n_2 \mid k_1)$, $n_2 = 0, \ldots, M_2 - 1$, $k_1 = 0, \ldots, M_1 - 1$ in $O(M_1 M_2 \log_2 M_2)$ operations, using the FFT of one-dimensional signals. We then compute the FFT of the signals $\hat{x}(n_2 \mid k_1)$, $k_1 = 0, \ldots, M_1 - 1$, for all $n_2 = 0, \ldots, M_2 - 1$ in $O(M_2 M_1 \log_2 M_1)$ operations. Thus the total amount of computation is $O(M_1 M_2 \log_2(M_1 M_2))$.

References

[1] J. W. Cooley. The re-discovery of the fast fourier transform algorithm. *Microchimica Acta*, 93(1-6):33–45, 1987.

[2] J. W. Cooley, P. A. W. Lewis, and P. D. Welch. Historical notes on the fast fourier transform. *Proceedings of the IEEE*, 55(10):1675–1677, Oct 1967.

[3] M. T. Heideman, D. H. Johnson, and C. S. Burrus. Gauss and the history of the fast fourier transform. *Archive for History of Exact Sciences*, 34(3):265–277, 1985.