

## Fast Fourier Transform

Consider two polynomials:

$$A(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1}$$

and

$$B(x) = b_0 + b_1x + \cdots + b_{n-1}x^{n-1}.$$

Let their product be

$$C(x) = c_0 + c_1x + \cdots + c_{2n-2}x^{2n-2}.$$

For all  $0 \leq k \leq 2n-2$ , the coefficient  $c_k$  of  $C(x)$  is:

$$c_k = a_0b_k + a_1b_{k-1} + \cdots + a_kb_0 = \sum_{i=0}^k a_ib_{k-i}.$$

(Assume that, for  $i \geq n$ ,  $a_i$  and  $b_i$  are zero.)

Takes  $O(n^2)$  multiplications.

## Fast Fourier Transform

A polynomial of degree  $d$  is *uniquely* characterized by its values at  $d + 1$  **distinct** points.

Let  $x_0, x_1, \dots, x_{2n-2}$  be  $2n - 1$  distinct points.

Let  $A(x_0), A(x_1), \dots, A(x_{2n-2})$  be the values of the polynomial  $A(x)$  at these  $2n - 1$  points.

Let  $B(x_0), B(x_1), \dots, B(x_{2n-2})$  be the values of the polynomial  $B(x)$  at these  $2n - 1$  points.

Then, the values of the product polynomial  $C(x)$  at these  $2n - 1$  points is :  $C(x_i) = A(x_i) \times B(x_i)$  for  $0 \leq i \leq 2n - 2$ .

## Fast Fourier Transform

Polynomial evaluation: going from coefficient representation to value representation:  $O(n)$  for each point using Horner's scheme.

Polynomial interpolation: going from value representation to coefficient representation:  $O(n^2)$  using Lagrange interpolation.

1. Choose  $N$  distinct points  $x_0, \dots, x_{N-1}$ .
2. Evaluate  $A(x)$  and  $B(x)$  at these  $N$  points.
3. Pointwise product:  $C(x_i) = A(x_i) \times B(x_i)$  for  $0 \leq i \leq N - 1$ .
4. Interpolate  $C(x_0), \dots, C(x_{N-1})$  to get  $C(x)$ .

Computing  $C(x)$  take  $O(n^2)$  time.

Choose the points of evaluation so that  $C(x)$  can be computed in  $O(n \log n)$  time.

## Fast Fourier Transform: evaluation

Assume that the number of coefficients  $N$  is a power of 2.

Express  $A(x)$  as:

$$[a_0 + a_2x^2 + \cdots + a_{N-2}x^{N-2}] + x [a_1 + a_3x^2 + \cdots + a_{N-1}x^{N-2}]$$

$$\text{Let } A_e(x^2) = a_0 + a_2x^2 + \cdots + a_{N-2}x^{N-2}.$$

$$\text{Let } A_o(x^2) = a_1 + a_3x^2 + \cdots + a_{N-1}x^{N-2}.$$

$$\text{Then, } A(x) = A_e(x^2) + x A_o(x^2).$$

Suppose the points are such that  $x_j = -x_{\frac{N}{2}+j}$  for  $0 \leq j \leq \frac{N}{2} - 1$ : *symmetry*.

- Evaluate  $A_e(x_j^2)$  and  $A_o(x_j^2)$  for  $0 \leq j \leq \frac{N}{2} - 1$ .
- For  $0 \leq j \leq \frac{N}{2} - 1$ , form:

$$A(x_j) = A_e(x_j^2) + x_j A_o(x_j^2)$$

$$A(x_{j+\frac{N}{2}}) = A_e(x_j^2) - x_j A_o(x_j^2).$$

Number of multiplications reduces by a factor of 2.

Evaluating  $A_e(x^2)$  same problem with half the number of coefficients and half the number of points.

Need points that satisfy the symmetry property recursively.

## Fast Fourier Transform: roots of unity

Consider the equation  $x^N - 1 = 0$ .

$N$  distinct roots expressible as :

$$1, \omega, \omega^2, \dots, \omega^{N-1}$$

$\omega = e^{\frac{2\pi i}{N}}$ : principal  $N$ -th root of unity.

Let  $x_j = \omega^j$  for  $0 \leq j \leq N - 1$ .

Properties of a principal  $N$ -th root of unity.:

- $\omega^N = 1$ .
- $\omega^j \neq 1$  for  $1 \leq j \leq N - 1$ .
- $\omega^j = -\omega^{\frac{N}{2}j}$  for  $0 \leq j \leq \frac{N}{2} - 1$ .

If  $w$  is a principal  $N$ -th root of unity,  $w^2$  is a principal  $\frac{N}{2}$ -th- root of unity.

## Fast Fourier Transform: Algorithm

$\text{FFT}(A, N, \omega) \rightarrow U$

$$A1 \leftarrow \text{FFT}(A_e, \frac{N}{2}, \omega^2)$$

$$A2 \leftarrow \text{FFT}(A_o, \frac{N}{2}, \omega^2)$$

For  $0 \leq j \leq \frac{N}{2} - 1$  Do:

$$U(j) = A1(j) + \omega^j A2(j)$$

$$U(j + \frac{N}{2}) = A1(j) - \omega^j A2(j)$$

$O(N \log N)$ .

## Fast Fourier Transform: Algorithm

To form the  $R$ , the values of the product polynomial  $C(x)$  at the  $N$  points  $\omega^j$  for  $0 \leq j \leq N - 1$ .

FFTEvaluation( $A(x), \omega, U$ ).

FFTEvaluation( $B(x), \omega, V$ ).

For  $0 \leq j \leq N - 1$  do:  $R[j] = U[j] \times V[j]$ .

## Fast Fourier Transform: Evaluation as Matrix vector product

FFT evaluates the polynomial  $A(x)$  at  $1, \omega, \omega^2, \dots, \omega^{N-1}$ .

$A$  is the vector of coefficients of  $A(x)$ .

Result in vector  $U$ .

Let  $\mathcal{W}$  be the  $N \times N$  matrix whose  $(j, k)$ -th entry is  $\omega^{jk}$ .

Vector  $U$  with values of polynomial  $A(x)$  at  $1, \omega, \omega^2, \dots, \omega^{N-1}$  is:

$$\mathcal{W} \times A = U.$$

## Fast Fourier Transform: Inverse Transform

Values of the product  $C(x)$  of  $A(x)$  and  $B(x)$  in the vector  $R$ .

Let  $C$  be the vector of (unknown) coefficients of  $C(x)$ .

Then,

$$\mathcal{W} \times C = R$$

To get  $C$  given  $R$ :

$$C = \mathcal{W}^{-1}R$$

Let:

$$\mathcal{W}\mathcal{I}_{jk} = \left(\frac{1}{\omega}\right)^{jk}.$$

for  $0 \leq j, k \leq N - 1$ .

Form the product of  $\mathcal{W}$  with  $\mathcal{W}\mathcal{I}$ :

$$(\mathcal{W} \times \mathcal{W}\mathcal{I})_{jk} = 0$$

if  $j \neq k$ .

$$(\mathcal{W} \times \mathcal{W}\mathcal{I})_{jk} = N$$

if  $j = k$ .

Conclude that:

$$\mathcal{W}^{-1} = \frac{1}{N}\mathcal{W}\mathcal{I}.$$

## Fast Fourier Transform: Inverse Transform

$\frac{1}{\omega}$  is a principal  $N$ -th root of unity if  $\omega$  is.

To get the coefficients  $C$  given  $R$ :

- FFTEvaluation( $R, N, \frac{1}{\omega}$ ) to get a vector  $S$ .
- For  $0 \leq j \leq N - 1$  do  $S[j] = S[j]/N$ .

$O(N \log N)$ .