Polynomials & FFT

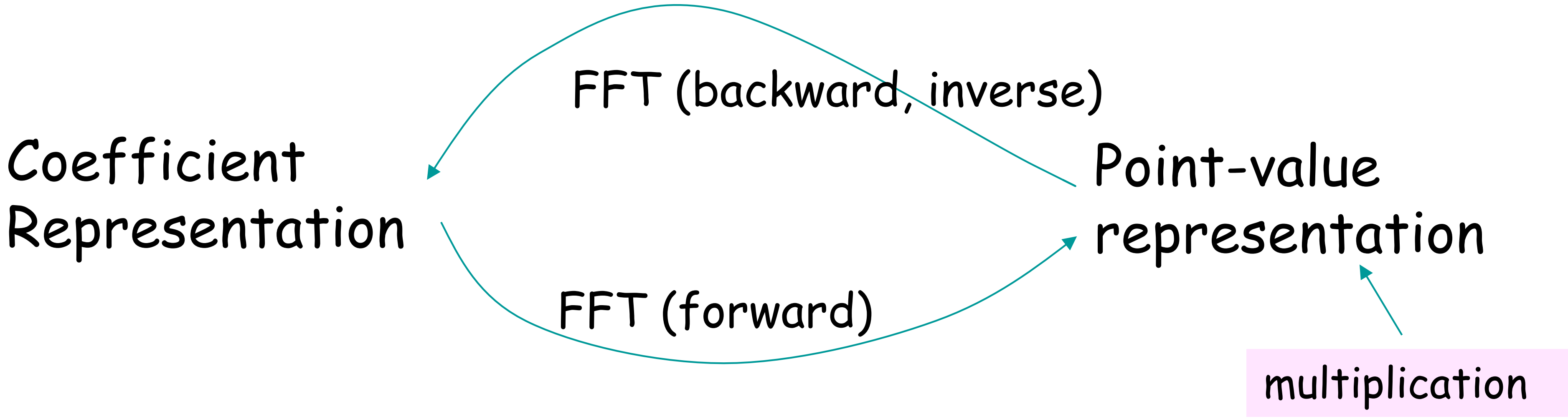
Reading

Cormen et al. : Chapter 30 (recommended)

Dasgupta et al. : 2.6

Lipson, *Elements of Algebra & Algebraic Computing* (available in the library)

Multiplying two degree-n polynomials: $O(n^2)$ time -> $O(n \log n)$



FFT (forward): Evaluation at multi-points.
FFT (backward): Interpolation.
Either step takes $O(n \log n)$ time.

Thus, two polynomials (represented by n coefficients) can be multiplied in $O(n \log n)$ time.

Polynomials

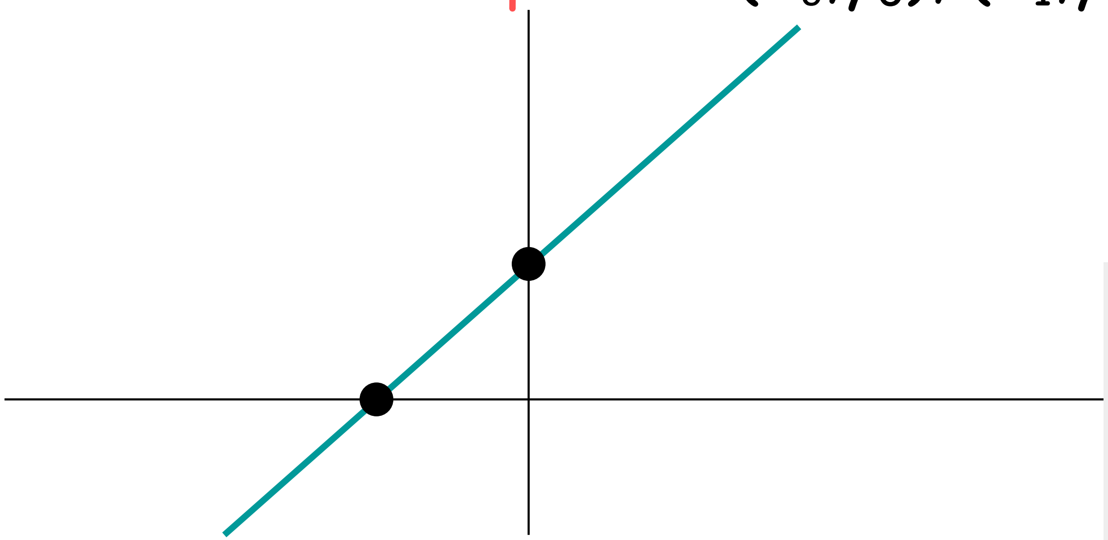
A degree $n-1$ polynomial $P(x)$ can be represented by

- n coefficients $a_0, a_1, a_2, \ldots a_{n-1}$

  I.e., $P(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{n-1}x^{n-1}$;

or

- values at n points: $(x_0,y_0), (x_1,y_1), \ldots, (x_{n-1},y_{n-1})$, where $y_i = P(x_i)$



Example

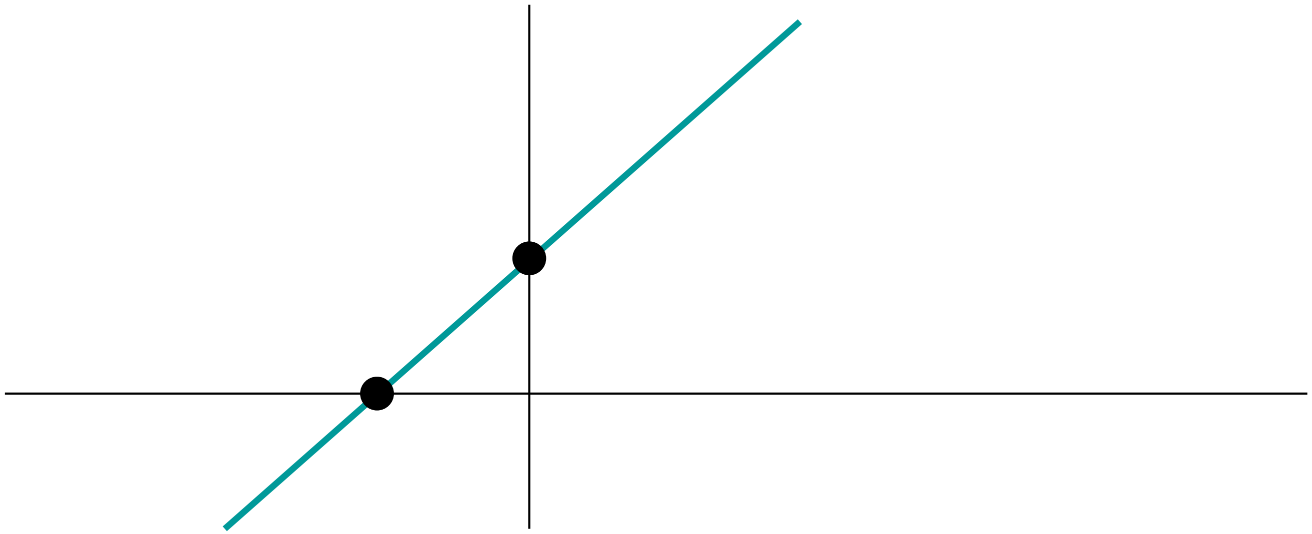
$P(x) = x + 1$ can be represented by two points $(0,1)$ and $(-1,0)$.

In fact, any two points on the green line can represent $p(x) = x+1$: $(1,2)$ & $(2,3)$

Polynomials

A degree n-1 polynomial P(x) can be represented by

- n coefficients $a_0, a_1, a_2, \ldots a_{n-1}$.
  I.e., $P(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{n-1}x^{n-1}$;
- values at n points: $(x_0,y_0), (x_1,y_1), \ldots, (x_{n-1},y_{n-1})$.



Interpolation Theorem. Given $\{ (x_0,y_0), (x_1,y_1), \ldots, (x_{n-1},y_{n-1}) \}$, there is a unique degree-(n-1) polynomial P(x) such that $P(x_i) = y_i$ for all i.

Evaluation of polynomials

How to evaluate $P(x) = a_0 + a_1x + a_2x^2 + \dots + a_{n-1}x^{n-1}$ at a certain value, say, x_1.

- E.g., $P(x) = 3x^2 + 5x - 1$; $P(x_1 = 0) = -1$; $P(x_2 = 1) = 7$; …

Horner's rule: n-1 multiplications (plus n additions).

$(a_{n-1} x_1 + a_{n-2})$

$(a_{n-1} x_1 + a_{n-2})\, x_1 + a_{n-3}$

$((a_{n-1} x_1 + a_{n-2})\, x_1 + a_{n-3})\, y + a_{n-4}$

.

.

$(..((a_{n-1} x_1 + a_{n-2})\, x_1 + a_{n-3})\, x_1 + a_{n-4})\, x_1 + \dots)\, x_1 + a_0$

Polynomial multiplications (convolution)

Given $P(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{n-1}x^{n-1}$ and

$Q(x) = b_0 + b_1x + b_2x^2 + \ldots + b_{n-1}x^{n-1}$,

computing $P(x) \times Q(x)$ requires $O(n^2)$ multiplications.

Degree of $P(x) \times Q(x)$: 2n-2.

$P(x) \times Q(x) =$

$a_0 b_0 \quad + (a_0 b_1 + a_1 b_0)x \quad + (a_0 b_2 + a_1 b_1 + a_2 b_0)x^2$

$+ \ldots$

$+ (a_0 b_i + a_1 b_{i-1} + a_2 b_{i-2} + \ldots + a_i b_0)x^i$

$+ \ldots$

$+ a_{n-1}b_{n-1}x^{2n-2}$

Point-Value representation

$P(x) \times Q(x)$, a degree 2n-2 polynomial, can be represented by

- either 2n-1 coefficients;
- or its values at 2n-1 distinct points of x.

Suppose we know the values of $P(x)$ and $Q(x)$ at 2n-1 points.

$P(x)$: $(x_0, y_0), (x_1, y_1), (x_2, y_2), \ldots, (x_{2n-2}, y_{2n-2})$.

$Q(x)$: $(x_0, z_0), (x_1, z_1), (x_2, z_2), \ldots, (x_{2n-2}, z_{2n-2})$.

Then computing $P(x) \times Q(x)$ is easy: 2n-1 multiplications.

- $y_0\, z_0$,
- $y_1\, z_1$,
- …
- $y_{2n-2}\, z_{2n-2}$.

These 2n-1 point-values uniquely represent $P(x) \times Q(x)$.

Which representation is better?

	Add	Multiply	Evaluate
Coefficient representation	$O(n)$	$O(n^2)$	$O(n)$
Point-value representation	$O(n)$	$O(n)$	$O(n^2)$

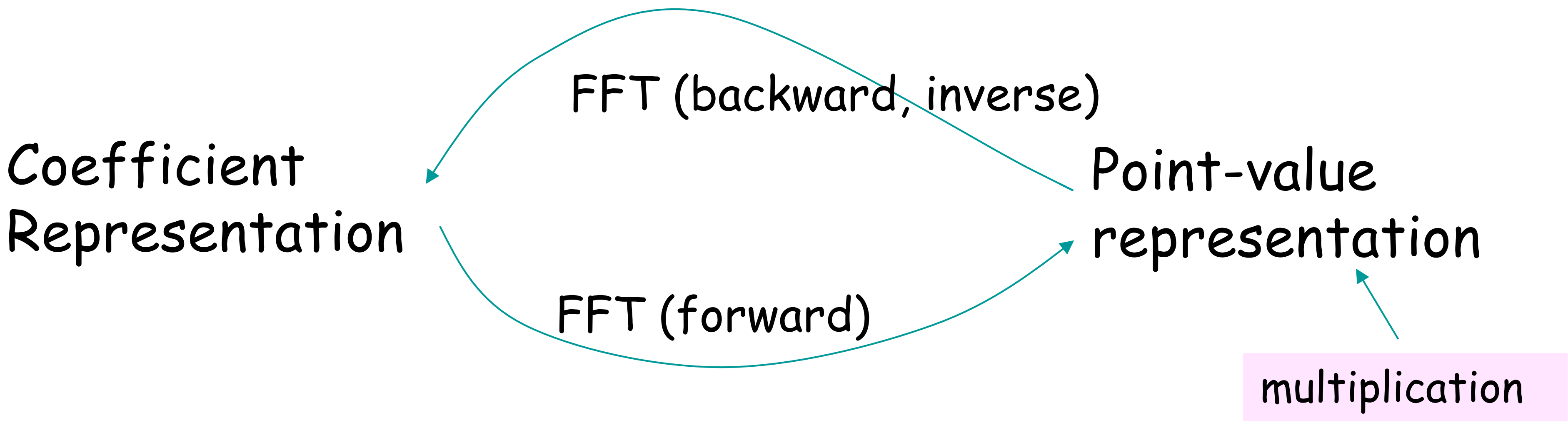
$P(x)$: $(x_0, y_0), (x_1, y_1), \ldots, (x_{n-1}, y_{n-1})$

To evaluate $P(x)$ with $x = x_n$, we calculate

$$\sum_{k=0 \text{ to } n-1} y_k \frac{\prod_{j \neq k} (x_n - x_j)}{\prod_{j \neq k} (x_k - x_j)}$$

Lagrange's formula

Conversion



FFT (forward): Evaluation at multi-points.
FFT (backward): Interpolation.
Either step takes O(n log n) time.

Thus, two polynomials (represented by coefficients) can be multiplied in O(n log n) time.

Forward transform: fast multi-point evaluation

Input: $P(x)$, represented by n coefficients a_0, a_1, a_2, a_{n-1}

To compute: $P(x)$ at some n points: $x_0, x_1, x_2, \ldots x_{n-1}$

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \\ y_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & x_0 & x_0^2 & \ldots & x_0^{n-1} \\ 1 & x_1 & x_1^2 & \ldots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \ldots & x_2^{n-1} \\ \\ 1 & x_{n-1} & x_{n-1}^2 & \ldots & x_{n-1}^{n-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \\ a_{n-1} \end{pmatrix}$$

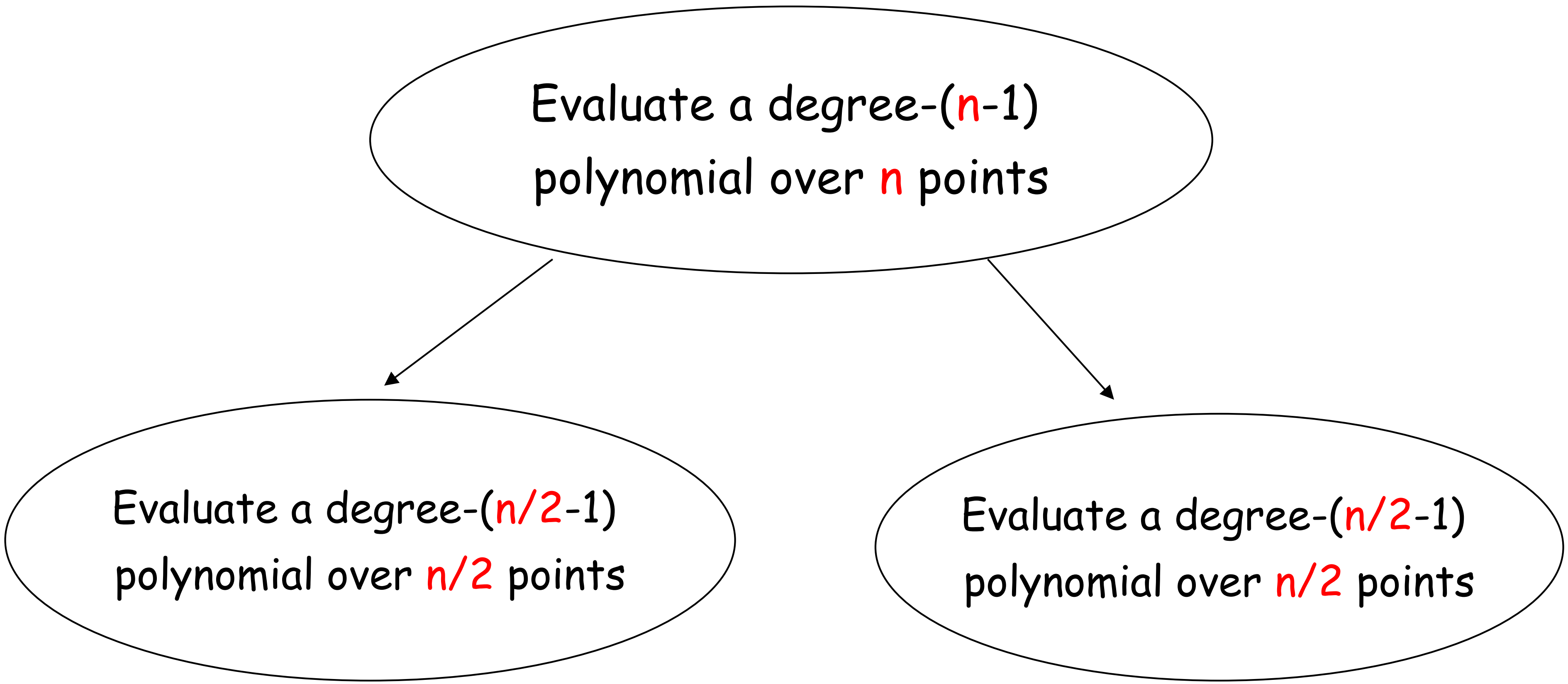
Output

Brute force: matrix-vector multiplication; $O(n^2)$ steps.

Divide and conquer: $O(n \log n)$? Yes, if the n points are chosen appropriately.

Divide & Conquer

$P(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \ldots + a_{n-1}x^{n-1}$ (n is a power of 2).



Divide & Conquer

$P(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \ldots + a_{n-1}x^{n-1}$ (n is a power of 2).

Define 2 polynomials with degree n/2 – 1 (and n/2 coefficients):

- $P_{even}(y) = a_0 + a_2y + a_4y^2 + \ldots + a_{n-2}y^{n/2-1}$ (degree $n/2-1$) ;
- $P_{odd}(y) = a_1 + a_3y + a_5y^2 + \ldots + a_{n-1}y^{n/2-1}$ (degree $n/2-1$)

Fact. $P(x) = P_{even}(x^2) + x\, P_{odd}(x^2)$

Divide & Conquer

$P(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \ldots + a_{n-1}x^{n-1}$ (n is a power of 2).

Define 2 polynomials.

- $P_{even}(y) = a_0 + a_2y + a_4y^2 + \ldots + a_{n-2}y^{n/2-1}$ (degree n/2-1) ;
- $P_{odd}(y) = a_1 + a_3y + a_5y^2 + \ldots + a_{n-1}y^{n/2-1}$ (degree n/2-1)

Fact. $P(x) = P_{even}(x^2) + x\, P_{odd}(x^2)$

Smart choice of points to evaluate.

- e.g., choose two points $x_i = 5$ and $x_j = -5$.

Then $x_i^2 = x_j^2$, and

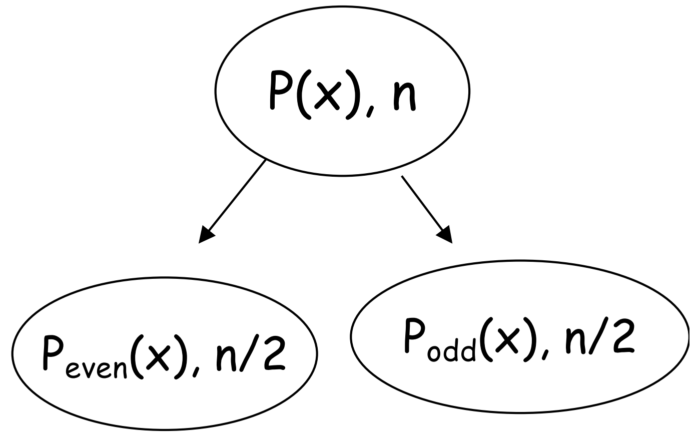
Almost for free

- $P(x_i) = P_{even}(x_i^2) + x_i\, P_{odd}(x_i^2)$, and
- $P(x_j) = P_{even}(x_j^2) + x_j\, P_{odd}(x_j^2) = \boxed{P_{even}(x_i^2) - x_i\, P_{odd}(x_i^2).}$

Recursive formulation

In general, choose n distinct points $x_0, x_1, x_2, \ldots x_{n-1}$ such that

- $x_0 = -x_{n/2}$ (then $x_0^2 = x_{n/2}^2$),
- $x_1 = -x_{n/2+1}$ (then $x_1^2 = x_{n/2+1}^2$)
- …
- $x_{n/2-1} = -x_{n-1}$ (then $x_{n/2-1}^2 = x_{n-1}^2$)



$\boxed{\text{Evaluate } P(x) \text{ (degree } n-1\text{) at } n \text{ points}}$

Evaluate $P_{even}(x)$ & $P_{odd}(x)$ at n points $x_0^2, x_1^2, \ldots, x_{n/2-1}^2, x_{n/2}^2, \cdots x_{n-1}^2$

=> Evaluate $P_{even}(x)$ & $P_{odd}(x)$ at n/2 points $x_0^2, x_1^2, \ldots, x_{n/2-1}^2$

For $0 \le i \le n/2 - 1$,

- $P(x_i) = P_{even}(x_i^2) + x_i P_{odd}(x_i^2)$, and
- $P(x_{n/2+i}) = P_{even}(x_i^2) - x_i P_{odd}(x_i^2)$

Recurrence

Evaluate two degree n/2-1 polynomials at n/2 points:

$P_{even}(x)$ at $x_0^2, x_1^2, \ldots, x_{n/2-1}^2$; $P_{odd}(x)$ at $x_0^2, x_1^2, \ldots, x_{n/2-1}^2$

For $0 \leq i \leq n/2 - 1$,

- $P(x_i) = P_{even}(x_i^2) + x_i P_{odd}(x_i^2)$, and
- $P(x_{n/2+i}) = P_{even}(x_i^2) - x_i P_{odd}(x_i^2)$

Let T(n) be the number of operations to evaluate a degree n-1 polynomial at some n points.

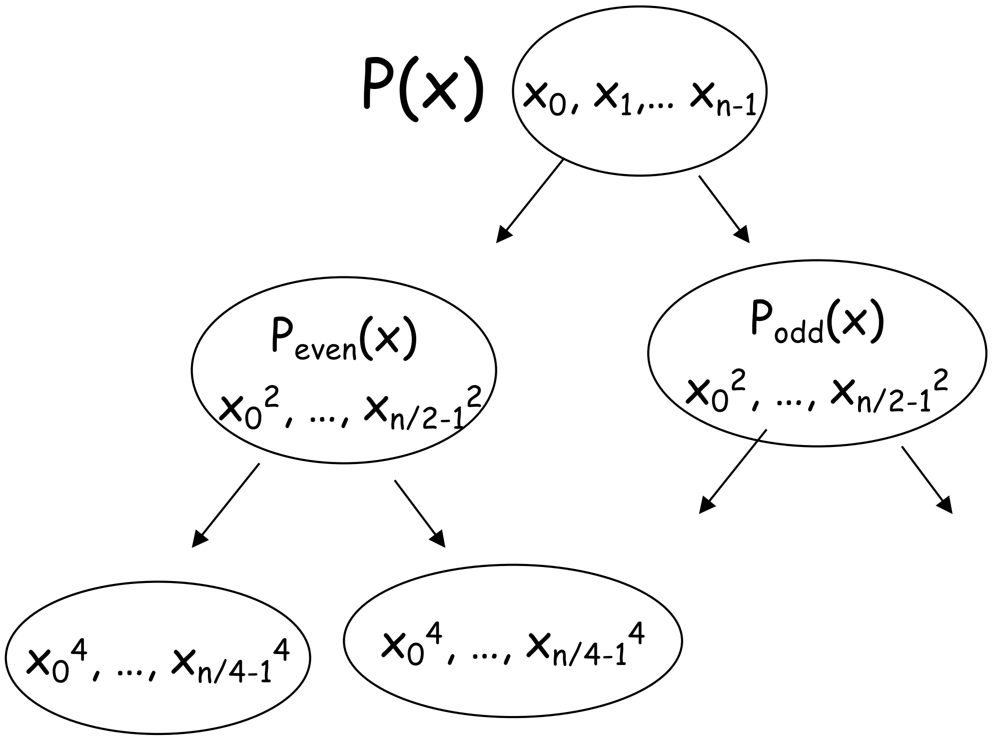
$T(n) = 2\ T(n/2) + c\ n$

Therefore, $T(n) = O(n \log n)$.

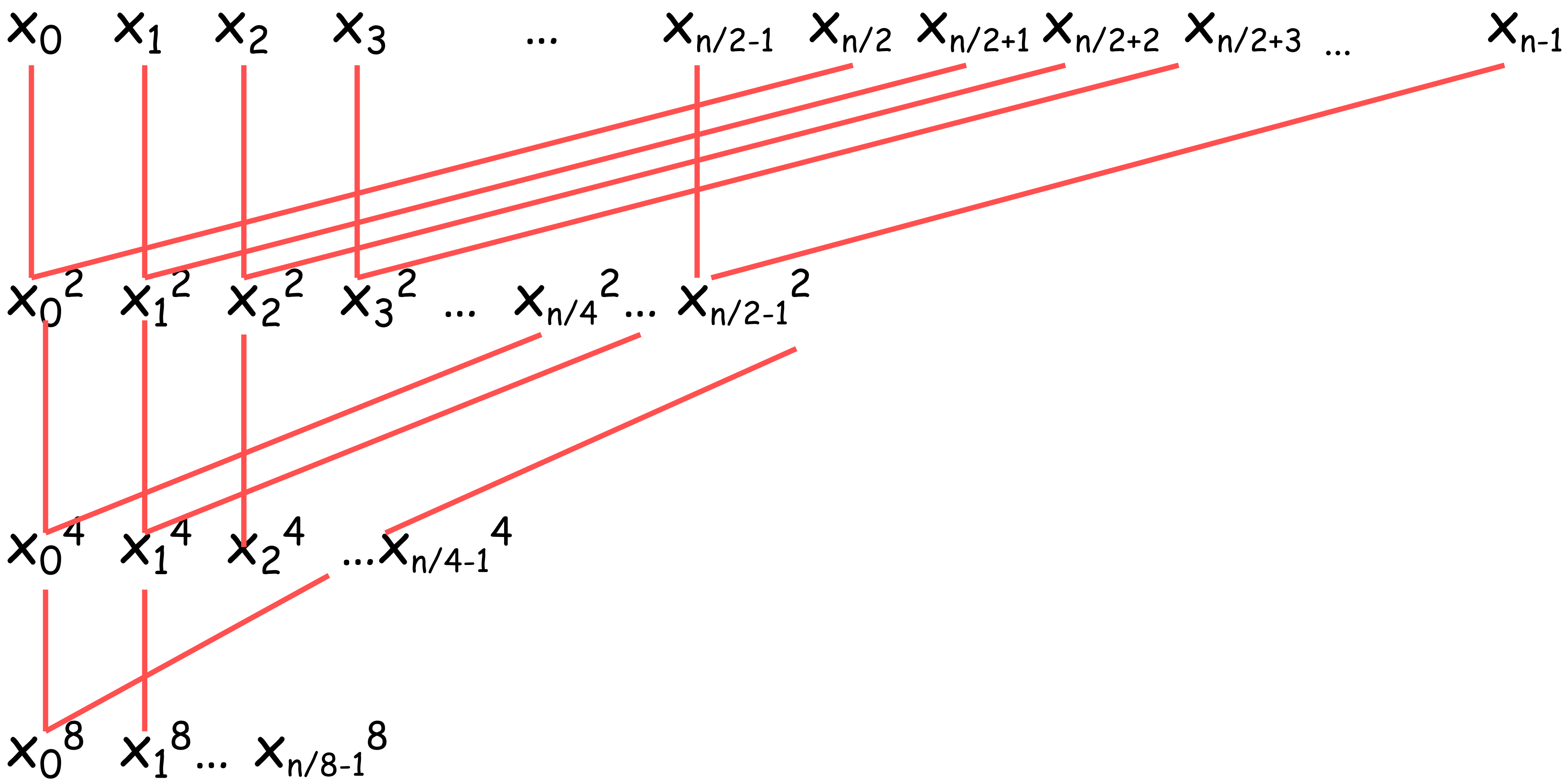
FFT Property: to realize the recursion

The points $x_0, x_1, x_2, \ldots x_{n-1}$ (n is a power of 2) are said to satisfy the FFT Property if

- $x_0 = -x_{n/2}$, $x_1 = -x_{n/2+1}$, ..., $x_{n/2-1} = -x_{n-1}$, and
- the n/2 (if > 1) points $x_0^2, x_1^2, \ldots, x_{n/2-1}^2$ also satisfies the FFT Property.



FFT Graph



x_0 x_1 x_2 x_3 … $x_{n/2-1}$ $x_{n/2}$ $x_{n/2+1}$ $x_{n/2+2}$ $x_{n/2+3}$ … x_{n-1}
${x_0}^2$ ${x_1}^2$ ${x_2}^2$ ${x_3}^2$ … ${x_{n/4}}^2$ … ${x_{n/2-1}}^2$
${x_0}^4$ ${x_1}^4$ ${x_2}^4$ … ${x_{n/4-1}}^4$
${x_0}^8$ ${x_1}^8$ … ${x_{n/8-1}}^8$

FFT Property: to realize the recursion

The points $x_0, x_1, x_2, \ldots x_{n-1}$ (n is a power of 2) are said to satisfy the FFT Property if

- $x_0 = -x_{n/2}$, $x_1 = -x_{n/2+1}$, ..., $x_{n/2-1} = -x_{n-1}$, and
- the n/2 (if > 1) points $x_0^2, x_1^2, \ldots, x_{n/2-1}^2$ also satisfies the FFT Property.

For example, n = 4.

Consider $(x_0, x_1, x_2, x_3) = (1, 2, -1, -2)$.

Then $x_0 = -x_2$, $x_1 = -x_3$

However, does $(x_0^2, x_1^2) = (1,4)$ satisfies FFT Property?

No. $x_0^4 = 1 \neq x_1^4 = 16$

What can be x_0^2 and x_1^2 so that $x_0^2 = - x_1^2$?

Complex numbers

Mathematics background

Def. ω is called an n-th root of unity if ω is a root of the equation $x^n - 1 = 0$. That is, $\omega^n = 1$.

Def. ω is called a primitive n-th root of unit if $\omega^n = 1$, and $\omega^k \neq 1$ for all $k = 1, \ldots, n-1$.

Example 1. $n = 2$. 1 is a 2nd root of unity.
-1 is a primitive 2nd root of unity.

For $n > 2$, the primitive n-th roots of unity are non-real complex numbers. Let $i = \sqrt{-1}$.

Background: Complex numbers

Note that $e^{i2\pi} = 1$, $e^{i\pi/2} = i$ and $e^{i\pi} = -1$ (polar representation),
where Let **i** = $\sqrt{-1}$.

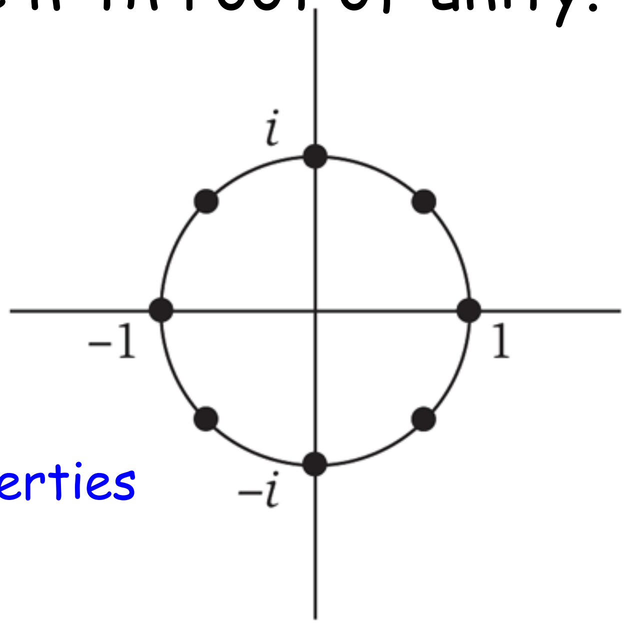
Fact. $\omega = e^{i2\pi/n}$ is a primitive n-th root of unity.

Example 2.

$n = 4$, $e^{i2\pi/n} = i$

- $i^1 = i$, $i^2 = -1$, $i^3 = -i$, $i^4 = 1$
- !? $i, -1, -i, 1$ satisfy the FFT properties

$n = 8$, $e^{i2\pi/n} = \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} i$



The 8th roots of unity in the complex plane.

Background: primitive n-th root of unity

Assume n is an even number.

Lemma 1. If ω is a primitive n-th root of unity, then ω^2 is a primitive n/2-th root of unity.

Proof.

- $(\omega^2)^{n/2} = \omega^n = 1$.
- For any $0 < k < n/2$,
  $(\omega^2)^k = \omega^{2k} \neq 1$ because $0 < 2k < n$.

Background: primitive n-th root of unity

Assume n is an even number.

Lemma 2. if ω is a primitive n-th root of unity, then $\omega^{n/2} = -1$.

Proof.

Recall that $\omega^n = (\omega^{n/2})^2 = 1$.

Thus, $(\omega^{n/2})$ satisfies the equation $x^2 = 1$.

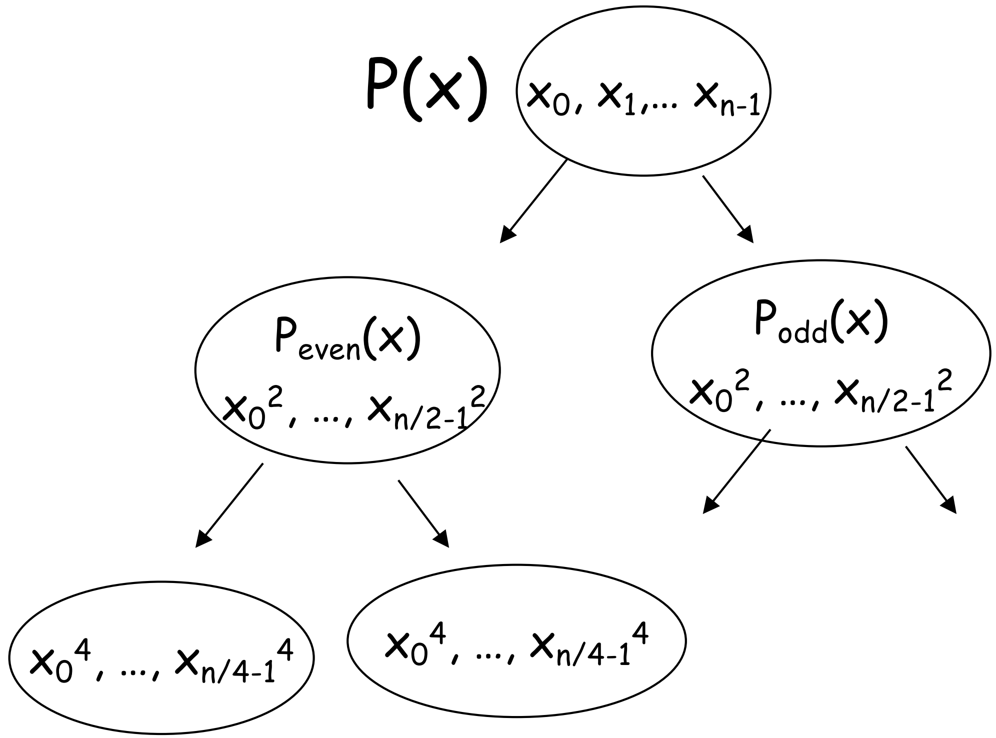
The equation $x^2 = 1$ has two roots, namely, 1 and -1.

Since $\omega^{n/2} \neq 1$, $\omega^{n/2} = -1$.

FFT Property

The points $x_0, x_1, x_2, \ldots x_{n-1}$ (n is a power of 2) are said to satisfy the FFT Property if

- $x_0 = -x_{n/2}$, $x_1 = -x_{n/2+1}$, …, $x_{n/2-1} = -x_{n-1}$, and
- the n/2 (if > 1) points $x_0^2, x_1^2, \ldots, x_{n/2-1}^2$ also satisfies the FFT Property.



Complex numbers

Let n be a power of 2, and let $\omega = e^{i2\pi/n}$, where $i = \sqrt{-1}$.

Claim. If ω is a primitive n-th root of unity. $(1, \omega^1, \omega^2, \ldots, \omega^{n-1})$ satisfies the FFT property.

By induction on n = 1, 2, 4, 8, 16, …. Base case n = 1 is trivial.

Lemma 2

ω is a primitive n-th root of unity $\Rightarrow$

for any $0 \le k \le n/2 - 1$, $\omega^{n/2+k} = \omega^{n/2}\omega^k = (-1)\omega^k = -\omega^k$;

therefore, $(\omega^{n/2+k})^2 = (\omega^k)^2$.

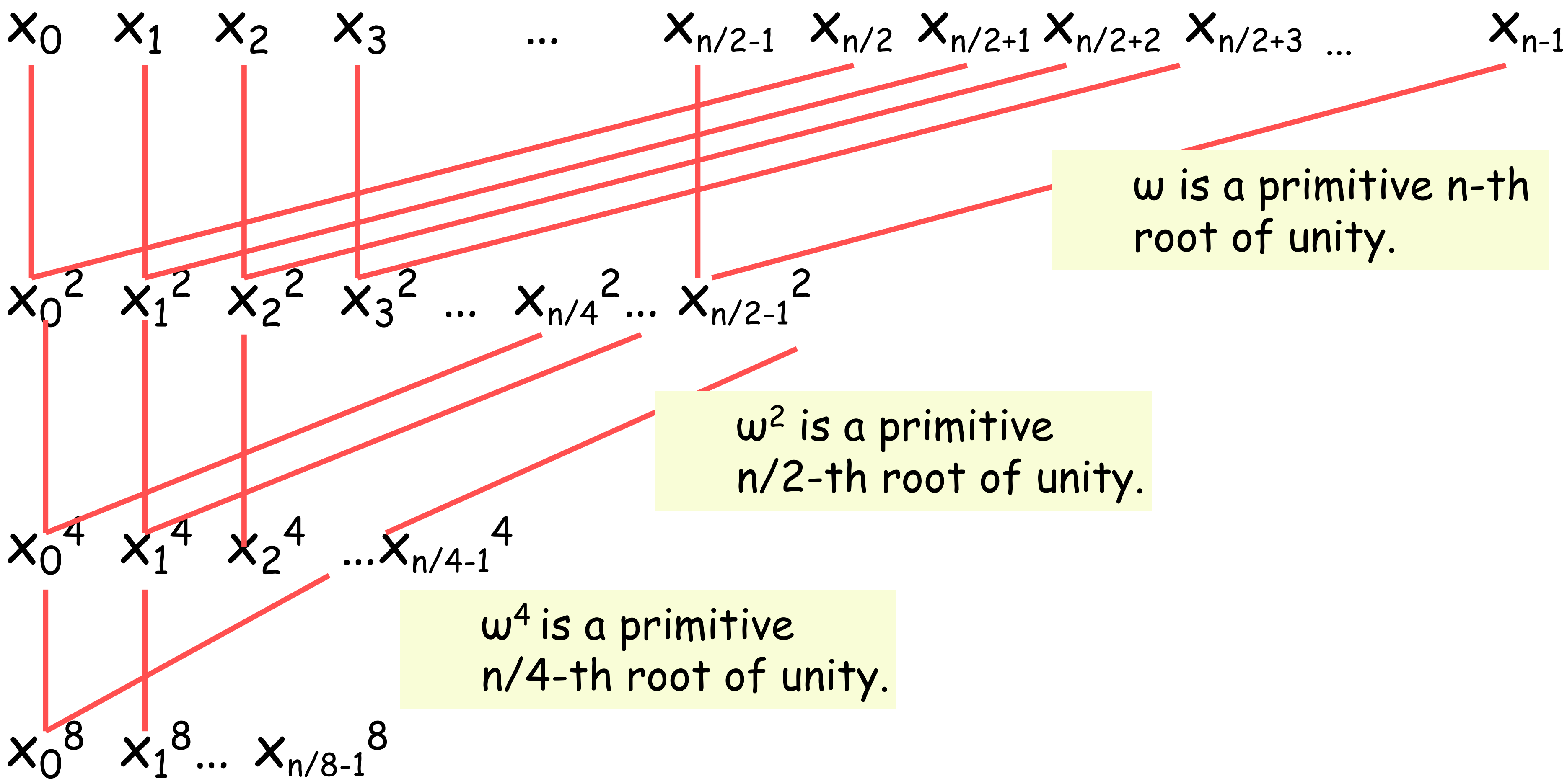
By Lemma 1, ω is a primitive n-th root of unity $\Rightarrow$

ω^2 is a primitive (n/2)-th root of unity.

By induction hypothesis, $(1, \omega^2, (\omega^2)^2, \ldots, (\omega^2)^{n/2-1})$ satisfies the FFT property.

Therefore, $(1, \omega^1, \omega^2, \ldots, \omega^{n-1})$ satisfies the FFT property.

FFT Graph: choosing $x_i = \omega^i$



DFT: evaluate P(x) at 1, ω^1, ω^2, …, ω^{n-1}

Let ω be a primtive n-th root of unity.

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \cdot \\ b_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega^1 & \omega^{1\cdot 2} & \cdots & \omega^{1\cdot n-1} \\ 1 & \omega^2 & \omega^{2\cdot 2} & \cdots & \omega^{2\cdot n-1} \\ & & \cdots & & \\ 1 & \omega^{n-1} & \omega^{(n-1)2} & \cdots & \omega^{(n-1)n-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \\ a_{n-1} \end{pmatrix}$$

??? given

The vector (b_0, b_1, …, b_{n-1}) is called the Discrete Fourier Transform of the vector (a_0, a_1, …, a_{n-1}).

Fast Fourier Transform: divide & conquer

FFT $(n, a_0, a_1, a_2, .., a_{n-1})$: $y_0, y_1, y_2, .., y_{n-1}$

if $n = 1$ then return a_0

P_{even}

$(z_0, z_1, z_2, .., z_{n/2-1}) = FFT\,(n/2, a_0, a_2, a_4\, .., a_{n-2})$

$(v_0, v_1, v_2, .., v_{n/2-1}) = FFT\,(n/2, a_1, a_3, a_5\, .., a_{n-1})$

P_{odd}

For $k = 0$ to $n/2 - 1$

$y_k = z_k + \omega^k v_k$

$y_{k+n/2} = z_k - \omega^k v_k$

return $(y_0, y_1, y_2, .., y_{n-1})$.

A summary

Given 2 degree n-1 polynomials P(x) and Q(x),
compute R(x) = P(x) Q(x).

To compute R(x) efficiently,

Step 0. Let N be a power of 2 such that $N-1 \geq 2n-2$. Treat P(x) and Q(x) as degree N-1 polynomials.

Step 1. Evaluate P(x) and Q(x) at N points $(1, \omega, \omega^2, \ldots, \omega^N)$, where ω is a primitive N-th root of unity.

Step 2. Multiply the two values at each of these N points.

Step 3. Interpolation: Find a polynomial that evaluates to the above values at the N points.

Step 1: FFT takes O(n log n) operations.
Step 2: O(n) operations.
Step 3: Inverse transform O(n log n) operations.

Inverse Transform: interpolation

Suppose we know that a degree n-1 polynomial P(x), when evaluated at points $(1, \omega^1, \omega^2, \ldots, \omega^{n-1})$, yields the values b_0, $b_1, b_2, \ldots, b_{n-1}$.

We want to find the coefficients $a_0, a_1, a_2, \ldots a_{n-1}$ of P(x).

I.e.,

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \\ . \\ b_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & \omega^1 & \omega^{1\cdot 2} & \ldots & \omega^{1\cdot n-1} \\ 1 & \omega^2 & \omega^{2\cdot 2} & \ldots & \omega^{2\cdot n-1} \\ & & \ldots & & \\ 1 & \omega^{n-1} & \omega^{(n-1)2} & \ldots & \omega^{(n-1)n-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \\ \\ a_{n-1} \end{pmatrix}$$

given

???

Inverse of ω

Lemma. If ω is a primitive n-th root of unity, then ω^{-1} is also a primitive n-th root of unity.

Proof.

$$
\begin{aligned}
(\omega^{-1})^n &= (\omega^{-1})^n \cdot 1 \\
&= (\omega^{-1})^n (\omega)^n \\
&= (\omega^{-1}\omega)^n \\
&= 1
\end{aligned}
$$

For any $0 < k < n$,

$$
\begin{aligned}
(\omega^{-1})^k &= (\omega^{-1})^k \cdot (\omega)^n \\
&= (\omega^{-1}\omega)^k \cdot (\omega)^{n-k} \\
&= \omega^{n-k} \\
&\neq 1 \quad (\text{because } 0 < n - k < n)
\end{aligned}
$$

Inverse transform

Let ω be a primtive n-th root of unity.

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \\ . \\ b_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega^1 & \omega^{1\cdot 2} & \dots & \omega^{1\cdot n-1} \\ 1 & \omega^2 & \omega^{2\cdot 2} & \dots & \omega^{2\cdot n-1} \\ & & \dots & & \\ 1 & \omega^{n-1} & \omega^{(n-1)2} & \dots & \omega^{(n-1)n-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \\ \\ a_{n-1} \end{pmatrix}$$

Given B

$V(\omega)$

Compute A

Lemma. $A = n^{-1} V(\omega^{-1}) B$

We use forward transform for ω^{-1} to perform inverse transform for ω

Vandermonde Matrix

Let $V(\omega) = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & \omega^1 & \omega^{1\cdot 2} & \ldots & \omega^{1\cdot(n-1)} \\ 1 & \omega^2 & \omega^{2\cdot 2} & \ldots & \omega^{2\cdot(n-1)} \\ & & \cdots & & \\ 1 & \omega^{n-1} & \omega^{(n-1)2} & \ldots & \omega^{(n-1)(n-1)} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1\ldots & 1 \\ 1 & \omega^1 & \omega^2 & \omega^3 \ldots & \omega^{n-1} \\ 1 & \omega^2 & \omega^{2\cdot 2} & \omega^{2\cdot 3} \ldots & \omega^{2\cdot(n-1)} \\ 1 & \omega^3 & \omega^{3\cdot 2} & \omega^{3\cdot 3} \ldots & \omega^{3(n-1)} \\ & & \cdots & & \\ 1 & \omega^{n-1} & \omega^{(n-1)2} & \ldots & \omega^{(n-1)(n-1)} \end{pmatrix}$

$V(\omega)$ is called the Vandermonde matrix.

Fact. In $V(\omega)$, The k-th row is identical to the k-th column.

A couple of lemmas about Vandermonde matrix would enable us to perform inverse transform easily!

Inverse of ω

Lemma. If ω is a primitive n-th root of unity, then ω^{-1} is also a primitive n-th root of unity.

Proof.

$$
\begin{aligned}
(\omega^{-1})^n &= (\omega^{-1})^n \cdot 1 \\
&= (\omega^{-1})^n (\omega)^n \\
&= (\omega^{-1}\omega)^n \\
&= 1
\end{aligned}
$$

For any $0 < k < n$,

$$
\begin{aligned}
(\omega^{-1})^k &= (\omega^{-1})^k \cdot (\omega)^n \\
&= (\omega^{-1}\omega)^k \cdot (\omega)^{n-k} \\
&= \omega^{n-k} \\
&\neq 1 \quad (\text{because } 0 < n - k < n)
\end{aligned}
$$

Inverse of Vandermonde matrix

Lemma. If ω is a primitive n-th root of unity, then

$$V(\omega)^{-1} = n^{-1}V(\omega^{-1}).$$

That is, $V(\omega) \cdot n^{-1}V(\omega^{-1}) = I$, where I = the identity matrix.

$$I = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & & & \\ 0 & 0 & \dots & 1 \end{pmatrix}$$

Proof. Let $M = V(\omega) \cdot V(\omega^{-1})$. Label the row and columns of M from 0 to n-1.

Case 1. What is M[k,j] if $0 \le k = j < n$?

- The k-th row of $V(\omega) = [\, 1\ \omega^k\ \omega^{k2} \dots \omega^{k(n-1)}\,]$, and

  the j-th column of $V(\omega^{-1}) = [1\ (\omega^{-1})^k\ (\omega^{-1})^{k2} \dots (\omega^{-1})^{k(n-1)}\,]$.
- $M[k,j] = 1 + \omega^k (\omega^{-1})^k + \omega^{k2} (\omega^{-1})^{k2} \dots + \omega^{k(n-1)} (\omega^{-1})^{k(n-1)} = n.$

Proof

Let $M = V(\omega) \cdot V(\omega^{-1})$.

Case 2. What is $M[k,j]$ if $0 \leq k \neq j < n$?

$M[k,j] = 1 + \omega^k (\omega^{-1})^j + \omega^{k2} (\omega^{-1})^{j2} \cdots + \omega^{k(n-1)} (\omega^{-1})^{j(n-1)}$

$= 1 + \omega^{k-j} + (\omega^{k-j})^2 \cdots + (\omega^{k-j})^{n-1}$ (G.P. Sum)

$$= \frac{(\omega^{k-j})^n - 1}{(\omega^{k-j}) - 1} = \frac{(\omega^n)^{k-j} - 1}{(\omega^{k-j}) - 1} = 0.$$

PS. Recall that ω and ω^{-1} are primitive n-th roots of unity.

Since $k \neq j$, then $0 < |k-j| < n$ and $\omega^{k-j} \neq 1$.

Conclusion

Let $M = V(\omega) \cdot V(\omega^{-1})$.

$$M = \begin{pmatrix} n & 0 & \dots & 0 \\ 0 & n & \dots & 0 \\ \dots & & & \\ 0 & 0 & \dots & n \end{pmatrix}$$

Thus $V(\omega) \cdot n^{-1}V(\omega^{-1}) = n^{-1}M = I$.

Inverse transform → Forward transform

Let A be the coefficient vector (a_0 , a_1 , a_2, … a_{n-1}), and let B be the value vector (b_0 , b_1 , b_2, … b_{n-1}).

Then $B = V(\omega)A \Leftrightarrow V(\omega)^{-1} B = A$.

Thus given the value vector B, we can compute A by computing $n^{-1} V(w^{-1}) B$.

How to compute $V(w^{-1}) B$?

Use forward transform.
Intuitively, $V(w^{-1}) B$ corresponds to the evaluation of a polynomial with B as coefficients at the points (1 , ω^{-1}, $\omega^{-1\cdot 2}$, …, $\omega^{-1\cdot(n-1)}$), where ω^{-1} is a primitive n-th root of unity.

Other than complex numbers

Z = the set of integers isn't a "field" and the primitive n-th root of unit is not well defined.

However, for any prime number p, $Z_p = \{0, 1, 2, \ldots, p-1\}$ is a field (w.r.t. mod p arithmetic).

For example, consider $Z_{13} = \{0, 1, 2, \ldots, 12\}$.

8 is a primitive 4-th root unity.

$8^1 = 8$; $8^2 = 12 \bmod 13$; $8^3 = 5$; $8^4 = 1$.

$8^{-1} = 5$ ($8 \times 5 = 1 \bmod 13$).

8^{-1} is also a primitive 4-th root unity.

Z_{13}

$$V(8) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 8 & 12 & 5 \\ 1 & 12 & 1 & 12 \\ 1 & 5 & 12 & 8 \end{pmatrix}$$

$$V(5) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 5 & 12 & 8 \\ 1 & 12 & 1 & 12 \\ 1 & 8 & 12 & 5 \end{pmatrix}$$

$$V(8)\,V(5) = \begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$$