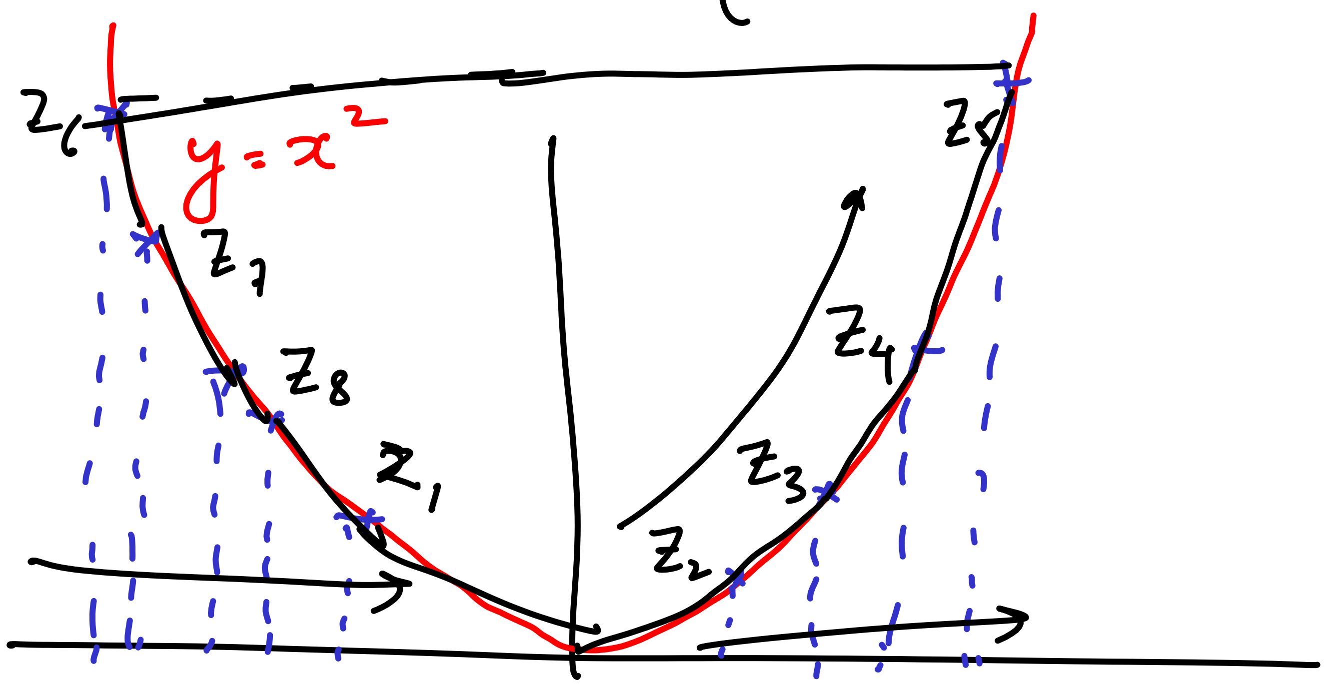
CSL 356 Lecture 31 Oct 15

A lower bound for planar convex hull.

Consider an input $S = (x_1, x_2 .. x_n)$ that we want to sort.

Construct $S' = \{(x_1, x_1^2), (x_2, x_2^2) .. (x_n, x_n^2)\}$



Construct CH (S'). CH(S') will contain all points of S' as boundary points

Claim: We can read out the sorted order of S from CH (S') in O(n) time.

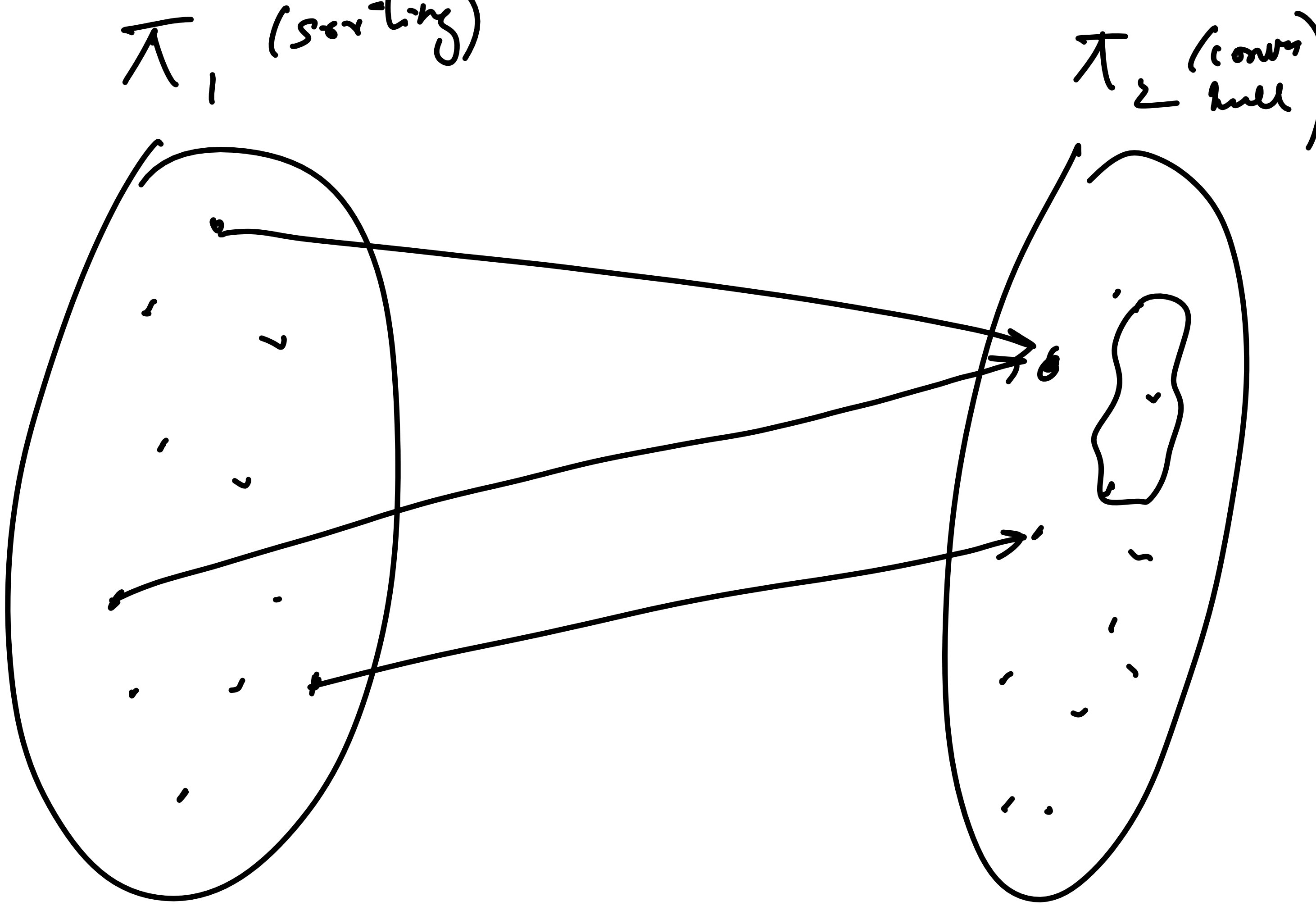
We have reduced the problem of sorting S to $CH(S')$ in $O(n)$ time

$$\underset{\pi_1}{\text{Sorting}} \;\propto_{O(n)}\; \underset{\pi_2}{\text{Convex Hull}}$$

time for reduction

$\Rightarrow$ The time to construct convex hull is asymptotically as much as sorting.

Reduction involves constructing an instance of π_2 given an instance of π_1. Then solve π_2 and map the solution of π_2 to π_1



- lower bound of π_2 is at least as much as the lower bound of π_1

- upper bound of π_2 is also an upper bound of π_1

Model of computation for Convex hulls require testing polynomials of degree ≥ 2

$$x_i^2 - x_i x_j \overset{?}{\leq} 0$$

$\boxed{x_i - x_j} \overset{?}{\leq} 0 \leftarrow$ degree 1 comparison

Given a polynomial,

$$P(x) = \underline{a_0} + \underline{a_1} x + \underline{a_2} x^2 + \cdots \underline{a_{n-1}} x^{n-1}$$

We want to evaluate $P(x_0)$ for some x_0

We need $O(n)$ multiplications and additions

We want to evaluate $P(x)$ at

$x_0, x_1, x_2, x_3 \;\; - - - - \;\; x_{n-1}$

$(x_i \neq x_j)$

$\Rightarrow \;\; \Omega(n^2)$ algorithm if we use the previous method

<u>Can we do better</u> ?

An alternate representation of polynomials is the point evaluation given at n distinct points for degree n polynomial

so
$$\begin{matrix} (x_0, & P(x_0)) \\ (x_1 & P(x_1)) \\ \vdots & \\ (x_{n-1} & P(x_{n-1})) \end{matrix}$$
gives the polynomial $P(x)$

Evaluation and Interpolations enable us to switch between the representations

(Verify - that interpolation using standard formulae takes $O(n^2)$-time)

Question : Can we do better?

If we choose $x_0, x_1 \cdots x_{n-1}$ "carefully" - then we can do it

$$P(x) = \underline{a_0} + a_1 x^1 + \underline{a_2} x^2 + \cdots a_{n-1} x^{n-1}$$

(assume n is a power of 2)

$$P_E(x) = a_0 + a_2 x + a_4 x^2 + \cdots a_{n-2} x^{\frac{n}{2}-1}$$

(even coefficients)

$$P_O(x) = a_1 + a_3 x + a_5 x^2 + \cdots a_{n-1} x^{n/2-1}$$

(odd coefficients)

Then

$$P(x) = P_E(x^2) + x \cdot P_O(x^2)$$

If $x_0 = -x_{n/2}$

then

$$P(x_0) = P_E(x_0^2) + x_0 \cdot P_O(x_0^2)$$

$$P(x_{n/2}) = P_E(x_{n/2}^2) + x_{n/2} \cdot P_O(x_{n/2}^2)$$

$$= P_E(x_0^2) - x_0 \cdot P_O(x_0^2)$$

Choose

$x_1 = -x_{n/2+1}$ $\quad x_2 = -x_{n/2+2} \cdots$

$$x_i = -x_{n/2+1}$$

$$x_0 \quad x_1 \quad x_2 \quad \cdots \quad x_{n/2} \quad x_{n/2+1} \quad \cdots \quad x_{n-1}$$

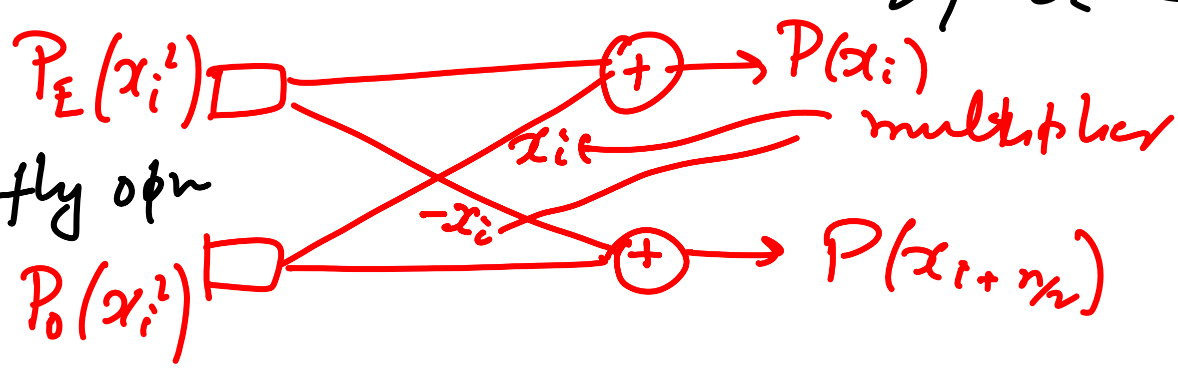
To evaluate the polynomial $P(x)$ at x_i and $x_{i+n/2}$,
we evaluate the polynomials $P_O(x_i^2)$ and $P_E(x_i^2)$.

These are $\frac{n}{2}$-coefficient polynomials (half the size of $P(x)$)

$$\boxed{\begin{aligned} P(x_i) &= P_E(x_i^2) + x_i \cdot P_O(x_i^2) \\ P(x_{i+n/2}) &= P_E(x_i^2) - x_i\, P_O(x_i^2) \end{aligned}}$$

with an extra multiplication and two extra additions we have the values of $P(x_i)$ and $P(x_{i+n/2})$ $0 \le i \le \frac{n}{2}-1$

Butterfly opn



$$P(x)^{x_0, x_1, \ldots x_{n-1}}_{a_0\, a_1\, a_2 \ldots a_{n-1}} = P_O(x)^{x_0^2\, x_1^2 \ldots x_{n/2-1}^2}_{a_1\, a_3\, a_5 \ldots}$$

$$+ P_E(x)^{x_0^2\, x_1^2 \ldots x_{n/2-1}^2}_{a_0\, a_2\, a_4 \ldots} + O(n)$$

multip' & additions

by choosing $\boxed{x_i = -x_{n/2+i}}$

$P_O(\,)$ has to be evaluated at $a_1\ a_3\ a_5\ a_7$

$x_0^2, \; x_1^2, \; \ldots \quad x_{n/2-1}^2$

We must choose $x_0^2 = -x_{n/4}^2$

$\Rightarrow \; x_{n/4}/x_0 = \sqrt{-1}$

For the next level of recursion, i.e. $\frac{n}{4}$ coeff polynomials to be evaluated at $\frac{n}{4}$ points, we would like to satisfy $x_0^4 = -1\, x_{n/8}^4$

$\vdots$

At the j^{th} level of recursion

$$x_0^{2^{j-1}} = -x_{\frac{n}{2^j}}^{2^{j-1}} \qquad j = 1, 2, \ldots \log n$$

$$x_0^{n/2} = -x_1^{n/2} \qquad \text{for } j = \log n$$

$$\left(\frac{x_1}{x_0}\right)^{n/2} = -1 \Rightarrow \frac{x_1}{x_0} = \boxed{(-1)^{\frac{1}{n/2}}}$$

This is n^{th} root of unity, say ω

$$\omega^n = 1 \qquad (\omega)^{n/2} = -1$$

. If we choose

$x_0 = 1 = w^0 \quad x_1 = w, \quad w_2 = w^2 \ldots$

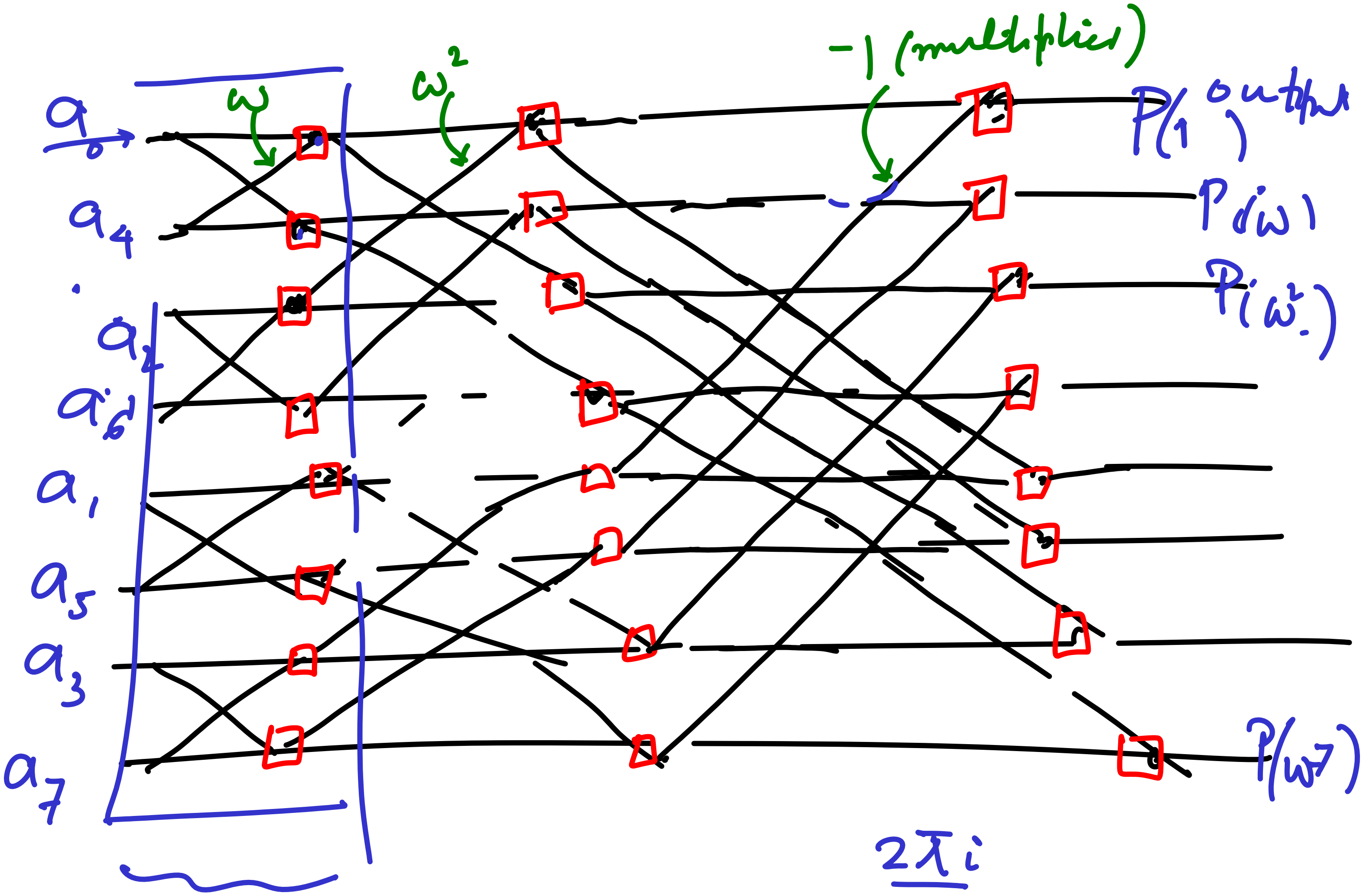
$x_{n-1} = w^{n-1}$

then the above equations are satisfied

The time to compute a n-coeff polynomial at n points

$$T(n) = 2T\left(\frac{n}{2}\right) + O(n)$$

$$\Rightarrow T(n) = O(n \log n)$$ multiplication + additions



lowest level of recursion

$\omega = e^{\frac{2\pi i}{n}}$

$\omega^8 = 1$

A 3 stage FFT network

Multiply two polynomials

$$P_A(x) = a_0 + a_1 x + a_2 x^2 + \cdots a_{n-1} x^{n-1}$$

$$P_B(x) = b_0 + b_1 x + b_2 x^2 + \cdots b_{n-1} x^{n-1}$$

$$P_{AB}(x) = P^1(x) \times P^2(x) = a_0 b_0 +$$

$$(a_1 b_0 + b_1 a_0) x$$

<u>convolution</u> $(a_3 b_0 + a_1 b_2 + b_2 a_1 + b_3 a_0) x^2$

$$\vdots$$

$$(a_k b_0 + a_{k-1} b_1 + a_{k-2} b_2 + \cdots a_0 b_k) x^k$$

Go for the point, value representation of $P_A(x)$ and $P_B(x)$ in $2n-1$ points

Multiplication in this representation is simple since $P_A(x_0) \cdot P_B(x_0) = P_{AB}(x_0)$