Quantum Factoring Postponed

Now FFTs, polynomial multiplication, integer multiplication.

Main Thm for today

Integer multiplication for n-bit integers in time $O(n \text{ polylog } n)$.

(in the RAM model).

On the way:

1. Polynomial multiplication
2. Fourier Transforms (FFTs).

R is some ring.

$$A(x) = \sum_{i=0}^{n} a_i x^i \in R[x]$$

$$B(x) = \sum_{j=0}^{n} b_j x^j \in R[x]$$

Want to compute $A(x) \cdot B(x)$ using as few ring operations as possible.

If

$$C(x) = A(x) \cdot B(x) \qquad C(x) = \sum c_\ell x^\ell.$$

then

$$c_\ell = \sum_{i+j=\ell} a_i b_j$$

Naive formula needs $O(n^2)$ operations.

Can we do it faster?

A really good idea:

1. Evaluate A, B on some set S of pts, with $|S| \geq 2n+1$

2. Multiply pointwise using $|S|$ multiplications.

3. Interpolate $C(x)$ of degree $\leq 2n$ from these.

Already shown that with $O(n^2)$ additions and scalings (if the ring R is not horrible) and $O(n)$ multiplications, we can multiply polys in $R[x]$.

FFT:
When R is nice we can do interpolation and evaluation at a nice set S superfast.

Defn $\omega \in R$ is called a principal

m'th root of 1 if

1. $\omega^m = 1$
2. for all $j \in \{1, 2, \ldots, m-1\}$

$\omega^j - 1$ is invertible in R.

Defn

Let ω be a principal m^{th} root of 1.

$$F(\omega) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{m-1} & \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(m-1)} & \\ \vdots & & & & & \\ 1 & \omega^{m-1} & \cdots & & \omega^{(m-1)^2} & \end{bmatrix}$$

is the Fourier Transform matrix associated to m, ω.

Observation:

$F(\omega) \cdot \vec{a} =$ evaluations of $\sum a_i X^i$

at $1, \omega, \ldots, \omega^{m-1}$.

Fact. $F(\omega)$ is invertible in R.

Proof: Vandermonde + ω is principal m^{th} root

Fact $\quad F(\omega) \cdot F(\omega^{-1}) = m \cdot I$.

i,j entry of LHS is

$$\sum_k \omega^{ik} \omega^{-kj} = \sum_k \omega^{k\cdot(i-j)}$$

$$= \begin{cases} \dfrac{\omega^{k\cdot m} - 1}{\omega^k - 1} = 0 & i \not\equiv j \bmod m \\ m & i - j = 0 \bmod m \end{cases}$$

Fact: If principal m^{th} root of 1 exists, m is invertible in R.

Thm If R has a principal

$m = 2^k$ th root of unity ω, then the map $R^m \to R^m$ sending $a \mapsto f(\omega) \cdot a$ can be computed in $O(m \log m)$ operations.

Proof. Want to evaluate

$$A(x) = \sum_{i=0}^{m-1} a_i x^i \quad \text{on} \quad \Omega_k = \{1, \omega, \omega^2, \ldots, \omega^{m-1}\}$$

Write $A(x) = B(x^2) + x \cdot C(x^2)$

$\deg(B), \deg(C) \leq \left(\frac{m}{2} - 1\right)$.

Evaluating A at Ω_k reduced to evaluating B and C at all pts in $\{y^2 : y \in \Omega_k\} = \Omega_{k-1}$

$\{1, \omega^2, \ldots, \omega^{m-2}\}$

recurse - $\{1, \omega^2, \omega^4, \ldots, \omega^{n-2}\}$
is the set of powers of ω^2.

Then compute

$$A(\omega^j) = B(\omega^{2j}) + \omega^j \cdot C(\omega^{2j})$$

for each $j = 0, \ldots, n-1$

Total runtime. $T(n)$.

Satisfies $T(n) \leq 2T(\frac{n}{2}) + O(n)$

Solving: $T(n) \leq O(n \log n)$.

Thm In any (initial) commutative ring R, can multiply (degree n) polynomials in $R(x)$ in $O(n \log^2 n)$ operations over R.

Plan Introduce artificially a principal root of unity in R.

Try) Consider $S = R[Y] / (Y^m - 1)$ (something like)

for some m (being a power of 2).

use FFT over S?

S is huge, S operations require lot of R-work.

Instead, we do something else.

Let m be a power of 2 bigger than n.

$$A(x) = \sum_{i=0}^{m-1} a_i x^i$$

$\sqrt{m}-1$ $i\sqrt{m}$

$$= \sum_{i=0}^{} \alpha_i(X)\, X$$

where $\text{degree}(\alpha_i) < \sqrt{m}$

$$B(X) = \sum_{j=0}^{\sqrt{m}-1} \beta_j(X)\, X^{j\sqrt{m}}$$

Let $Y = X^{\sqrt{m}}$.

$$A(X) = \sum \alpha_i(X)\, Y^i = Q_A(Y)$$

$$B(X) = \sum \beta_j(X)\, Y^j = Q_B(Y)$$

View these as polynomials in

$S[Y]$ where $S = R[X]/\langle X^{\sqrt{m}}+1 \rangle$

Claims 1. S has a primitive $\sqrt{m}$'th root of 1.

2. Multiplying these two

$Q_A(Y)$, $Q_B(Y)$

is enough to find

$A(X) - B(X)$.