Fourier transform algorithms. (Crandall and Pomerance 2001, §9.1.1, 9.5.2, 9.5.3)

Jonas Hoecht, Heidelberg university.
Proseminar *prime numbers and cryptography*.
11.04.2022.

1 Grammar school multiplication (GSM).

Definition. Base-B representation. $x \in \mathbb{N}_0$, $B, D \in \mathbb{N}$. $(x_i)_{i \in \mathcal{D}}$ with $0 \leq x_i < B$ and $\mathcal{D} = \{0, 1, \ldots, D-1\}$ with minimal D:

$$x = \sum_{i=0}^{D-1} x_i B^i. \tag{1}$$

Proposition. Grammar school multiplication. $x, y \in \mathbb{N}_0$ with base-B representations $x = (x_i)_B, y = (y_i)_B$ of length $\leq D$, $z = xy$. With

$$w_n = \sum_{i+j=n}^{D-1} x_i y_j = \sum_{i}^{D-1} x_i y_{n-i}, \tag{2}$$

an *acyclic convolution* of x and y, $w = x \times_A y$, $z = (z_n)_B$ of length $\leq 2D$ is obtained from w_n by *adjusting carry*.
Complexity: $\mathcal{O}(D^2)$.

2 Discrete Fourier Transform (DFT).

Definition. Signal. $D \in \mathbb{N}$, R a domain with $D^{-1} \in R$, $g_D \in R$ a primitive D^{th} root of 1. Then we call $x = (x_n)$ of length D, $x_n \in R$ a *signal* in R.

Proposition. Discrete Fourier Transform (DFT). $x = (x_n)$ a signal in R. Then $\mathcal{F}(x) = p = (p_k)$ with

$$p_k = \sum_{j=0}^{D-1} x_j g_D^{-jk} \tag{3}$$

is a well-defined sequence of elements of R, the so-called *discrete Fourier transform of* x.
Complexity: $\mathcal{O}(D^2)$.

Theorem. DFT is bijective. $x = \mathcal{F}^{-1}(p) = (x_j)$ defined by

$$x_j = \frac{1}{D} \sum_{k=0}^{D-1} p_k g_D^{jk}. \tag{4}$$

3 Fast Fourier Transform (FFT).

Observation. Danielson-Lanczos identity. D even.

$$p_k = \underbrace{\sum_{j=0}^{D/2-1} x_{2j} (g_D^2)^{-jk}}_{=:p_k^g} + g_D^{-k} \underbrace{\sum_{j=0}^{D/2-1} x_{2j+1} (g_D^2)^{-jk}}_{=:p_k^u}. \tag{5}$$

D Fourier-coefficients p_k from 2 Fourier transforms of size $D/2$.

Algorithm. Fast Fourier Transform (FFT). Iterative application of this identity yields the FFT algorithm (Cooley and Tukey 1965).
Complexity: $\mathcal{O}(D \log D)$.
Applications: data compression, spectral analysis, differential equations, telecommunication, e.g. mp3, MRT, digital oscilloscope.

4 FFT multiplication.

Theorem. Convolution theorem. x, y signals of length D. Then,

$$x \times y = \mathcal{F}^{-1}(\mathcal{F}(x) \star \mathcal{F}(y)), \ (p \star q)_n = p_n q_n. \tag{6}$$

The $\mathcal{O}(D \log D)$ FFT converts an $\mathcal{O}(D^2)$ cyclic convolution to an $\mathcal{O}(D)$, i.e. asymptotically negligible, dyadic product (!).
(Technical remark: *zero-padding* required to make this Theorem applicable to GSM, then $\times_A \mapsto \times$).

Algorithm. FFT multiplication Input: $x, y \in \mathbb{N}_0$ with base-B representations of lengths $\leq D$. Output: base-B representation of the product $z = xy$.

1. zero-pad x, y.
2. $p = \mathcal{F}(x)$, $q = \mathcal{F}(x)$.
3. $Z = p \star q$.
4. $z = \mathcal{F}^{-1}(Z)$.
5. round, adjust carry, delete leading zeros, return z.

Conjectured lower bound for the **complexity**: $\Omega(D \log D)$.

- **Schönhage-Strassen-algorithm** $\mathcal{O}(D \log D \cdot \log \log D)$ (Schönhage and Strassen 1971).
- **Fürer-algorithm** $\mathcal{O}(D \log D \cdot 2^{\mathcal{O}(\log^\star D)})$ (Fürer 2009).
- **Harvey-van-der-Hoeven-algorithm** $\mathcal{O}(D \log D)$ (!) (Harvey and Van Der Hoeven 2021).

5 Summary

Using a number-theoretic version of the FFT (Cooley and Tukey 1965), multiplication of large integers can be done in $\mathcal{O}(D \log D)$ instead of $\mathcal{O}(D^2)$ (Harvey and Van Der Hoeven 2021).
This is relevant e.g. for public-key cryptography, where large prime numbers need to be multiplied.

References

[1] James W Cooley and John W Tukey. "An algorithm for the machine calculation of complex Fourier series". In: *Mathematics of computation* 19.90 (1965), pp. 297–301.

[2] Richard Crandall and Carl Pomerance. *Prime numbers*. Springer, 2001.

[3] Martin Fürer. "Faster integer multiplication". In: *SIAM Journal on Computing* 39.3 (2009), pp. 979–1005.

[4] David Harvey and Joris Van Der Hoeven. "Integer multiplication in time O (n log n)". In: *Annals of Mathematics* 193.2 (2021), pp. 563–617.

[5] Arnold Schönhage and Volker Strassen. "Schnelle multiplikation grosser zahlen". In: *Computing* 7.3 (1971), pp. 281–292.