Max Planck Institute for Security and Privacy

Practical Aspects of Schönhage and Nussbaumer FFTs

Vincent Hwang

May 3, 2024

FFT-Based Polymul./Intmul.

	Complexity	Type	Constraint	Work
Schönhage–Strassen (1971)	$O(n \lg n \lg \lg n)$	Intmul.	None	[SS71]
Schönhage (1977)	$O(n \lg n \lg \lg n)$	Polymul.	2^{-1} or 3^{-1}	[Sch77]
Nussbaumer (1980)	$O(n \lg n \lg \lg n)$	Polymul.	2^{-1} or 3^{-1}	[Nus80]
Cantor–Kaltofen (1991)	$O(n \lg n \lg \lg n)$	Polymul.	None	[CK91]
Fürer (2009)	$n \lg n 2^{\Theta(\lg^* n)})$	Intmul.	None	[Für09]
Harvey–van der Hoeven (2021)	$O(n \lg n)$	Intmul.	None	[HvdH21]

Practicality of Schönhage/Nussbaumer

- Scheme-dependent (over $\mathbb{Z}_{2^k}$ or $\mathbb{Z}_q$ for an odd q).
- Platform-dependent (width of mul. instructions).
- Revising the cost with incomplete transformations.
- Sometimes practically fastest and sometimes not.
  - NTRU Prime (`sntrup761/ntrulpr761`) with AVX2
    - $\mathbb{Z}_{4591}[x]/\langle x^{761} - x - 1 \rangle$.
    - Schönhage/Nussbaumer is slower than multiplication-based FFT.
  - Saber on Cortex-M3
    - $\mathbb{Z}_{8192}[x]/\langle x^{256} + 1 \rangle$.
    - Schönhage/Nussbaumer is faster.

FFT-Based Polynomial Multiplications

- Monic degree-n $\boldsymbol{g}$.
- Ring hom. $f : R[x]/\langle \boldsymbol{g} \rangle \hookrightarrow S$, various FFTs.
  - Cooley–Tukey.
  - Schönhage/Nussbaumer.
  - Rader, Good–Thomas, Bruun, and many more.
- $\boldsymbol{ab} = f^{-1}\left(f(\boldsymbol{a})f(\boldsymbol{b})\right)$.
- Two f, one $\cdot_S$, and one f^{-1}.

$$\begin{tikzcd}
\boldsymbol{a} \arrow[r, "f"] & f(\boldsymbol{a}) \arrow[dr] & & \\
 & & f(\boldsymbol{a})f(\boldsymbol{b}) = f(\boldsymbol{ab}) \arrow[r, "f^{-1}"] & \boldsymbol{ab} \\
\boldsymbol{b} \arrow[r, "f"] & f(\boldsymbol{b}) \arrow[ur, "\cdot_S"'] & &
\end{tikzcd}$$

Cooley–Tukey FFT

- $n = 2^h, h \geq 0$.
- If (i) $\exists \omega_n \in R$ and (ii) $\exists n^{-1} \in R$, then
  $$R[x]/\langle x^n - 1\rangle \cong \prod R[x]/\left\langle x^{\frac{n}{2}} \pm 1\right\rangle \cong \cdots \cong \prod_i R[x]\Big/\left\langle x - \omega_n^{\mathtt{bitrev}(i)}\right\rangle \cong R^n.$$
- $\mathtt{bitrev}$: n-bit bit-reversal.

$$\begin{tikzcd}[column sep=small]
 & & & \frac{R[x]}{\langle x^n - 1\rangle} \arrow[dll] \arrow[drr] & & & \\
 & \frac{R[x]}{\left\langle x^{\frac{n}{2}} - 1\right\rangle} \arrow[dl] \arrow[dr] & & & & \frac{R[x]}{\left\langle x^{\frac{n}{2}} + 1\right\rangle} \arrow[dl] \arrow[dr] & \\
\frac{R[x]}{\left\langle x^{\frac{n}{4}} - 1\right\rangle} & & \frac{R[x]}{\left\langle x^{\frac{n}{4}} + 1\right\rangle} & & \frac{R[x]}{\left\langle x^{\frac{n}{4}} - \omega_4\right\rangle} & & \frac{R[x]}{\left\langle x^{\frac{n}{4}} + \omega_4\right\rangle} \\
\vdots & & \vdots & & \vdots & & \vdots \\
R^{\frac{n}{4}} & & R^{\frac{n}{4}} & & R^{\frac{n}{4}} & & R^{\frac{n}{4}}
\end{tikzcd}$$

n copies of R in total.

Analyzing Cooley–Tukey

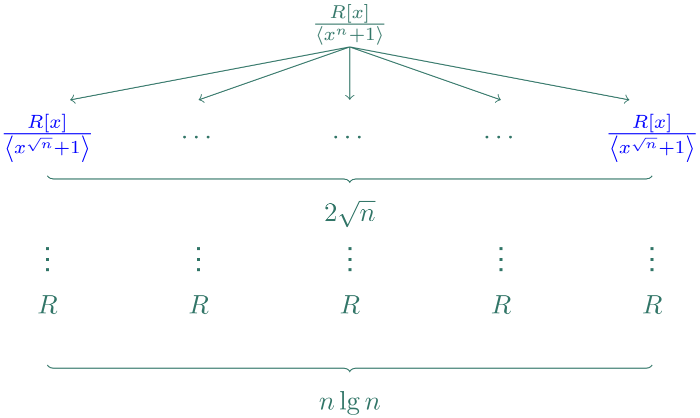
- $\mathcal{T}_{\cdot}$: # mul. in the transformation.
- $\mathcal{T}_{+}$: # add. in the transformation.
- $\mathcal{T}_{\#}$: # subproblems after the transformation.

$$\begin{cases} \mathcal{T}_{\cdot}(n) = 2\mathcal{T}_{\cdot}\left(\frac{n}{2}\right) + \frac{1}{2}n \\ \mathcal{T}_{+}(n) = 2\mathcal{T}_{+}\left(\frac{n}{2}\right) + n \\ \mathcal{T}_{\#}(n) = 2\mathcal{T}_{\#}\left(\frac{n}{2}\right) + [\![n = 1]\!] \end{cases}$$

$$\begin{cases} \mathcal{T}_{\cdot}(n) = \frac{1}{2}n \lg n \\ \mathcal{T}_{+}(n) = n \lg n \\ \mathcal{T}_{\#}(n) = n \end{cases}$$

Nussbaumer FFT

- $n = 2^{2^h}, h \geq 0.$
- $R[x]/\langle x^n + 1\rangle \cong \dfrac{\left(R[y]/\langle y^{\sqrt{n}}+1\rangle\right)[x]}{\langle x^{\sqrt{n}}-y\rangle} \hookrightarrow \dfrac{\left(R[y]/\langle y^{\sqrt{n}}+1\rangle\right)[x]}{\langle x^{2\sqrt{n}}-1\rangle} \cong \prod_i \dfrac{\left(R[y]/\langle y^{\sqrt{n}}+1\rangle\right)[x]}{\langle x-y^{\mathtt{bitrev}(i)}\rangle}$



Analyzing Nussbaumer

- No mul. in the transformation.
- $\mathcal{T}_+$: # add. in the transformation.
- $\mathcal{T}_\#$: # subproblems after the transformation.

$$\begin{cases} \mathcal{T}_+(n) = 2\sqrt{n}\mathcal{T}_+\left(\sqrt{n}\right) + \frac{1}{2}n\lg n + n \\ \mathcal{T}_\#(n) = 2\sqrt{n}\mathcal{T}_\#\left(\sqrt{n}\right) \cdot [\![n > 2]\!] + 2 \cdot [\![n = 2]\!] \end{cases}$$

$$\begin{cases} \mathcal{T}_+(n) = \frac{1}{2}n\lg n\lg\lg n + n\lg n - n \\ \mathcal{T}_\#(n) = n\lg n \end{cases}$$

Comparisons

	Cooley–Tukey	Schönhage/Nussbaumer
$\mathcal{T}_{\cdot}$	$\frac{1}{2}n \lg n$	0
$\mathcal{T}_{+}$	$n \lg n$	$\Theta\left(n \lg n \max\left(\lg \lg n, 1\right)\right)$
$\mathcal{T}_{\#}$	n	$n \lg n$

- Computation we will go through in this talk.
- Computation requiring a more careful analysis (memory op., add.).

Incomplete Transformation

We often stop earlier as long as problem sizes $\leq m$, a certain (platform-dependent) constant (usually 4 to 8).
Let $\mathcal{C}(m)$ be the cost of the fastest approach multiplying polynomials in $R[x]/\langle x^m + 1\rangle$.

- Schoolbook: $\Theta(m^2)$, m^2 mul.
- Karatsuba: $\Theta(m^{1.58})$.

	Cooley–Tukey	Schönhage/Nussbaumer
$\mathcal{T}_{\cdot}$	$\frac{1}{2}n \lg n \cdot \frac{1}{\lg m}$	0
$\mathcal{T}_{+}$	$n \lg n \cdot \frac{1}{\lg m}$	$\Theta\left(n \lg n \max\left(\lg \log_m n, 1\right)\right)$
$\mathcal{T}_{\#}$	$\frac{n}{m}$	$n \lg n \cdot \frac{1}{m \lg m}$
$\#$ mul., $3\mathcal{T}_{\cdot} + \mathcal{T}_{\#} \cdot \mathcal{C}(m)$	$n \lg n \cdot \frac{3}{2 \lg m} + nm$	$n \lg n \cdot \frac{m}{\lg m}$

- $\#$ mul.: $\frac{\text{Cooley–Tukey}}{\text{Schönhage/Nussbaumer}} \sim \frac{3}{2m}$.

An Unsuccessful Story of Schönhage/Nussbaumer I

- NTRU Prime (`sntrup761`/`ntrulpr761`): $\mathbb{Z}_{4591}[x]/\langle x^{761} - x - 1\rangle \cong \mathbb{F}_{4591^{761}}$.
- Choose a polynomial modulus $\boldsymbol{g}$ with deg ≥ 1521.
- Compute in $\mathbb{Z}_{4591}[x]/\langle \boldsymbol{g}\rangle$.
- Haswell, AVX2, 256-bit vector registers (packed 16-bit elements).
  - Mulmod. in $\mathbb{Z}_{4591}$ is not very fast.

	Schönhage/Nussbaumer	Mul.-based
Work	[BBCT22]	[Hwa24]
Poly. ring	$\frac{\mathbb{Z}_{4591}[x]}{\langle (x^{1024}+1)(x^{512}-1)\rangle}$	$\frac{\mathbb{Z}_{4591}[x]}{\langle \Phi_{17}(x^{96})\rangle}$
Cycles	$23,460$	$12,336$
Comment		Complicate tran. (omitted)

An Unsuccessful Story of Schönhage/Nussbaumer II

	Schönhage/Nussbaumer	Mul.-based
Poly. ring	$\frac{\mathbb{Z}_{4591}[x]}{\langle (x^{1024}+1)(x^{512}-1)\rangle}$	$\frac{\mathbb{Z}_{4591}[x]}{\langle \Phi_{17}(x^{96})\rangle}$
Total cycles	$23,460$	$12,336$
Tran. cycles	$10,500$	$9,378$
Small polymul. cycles	$12,960$	$2,958$

- Over the same coefficient ring $\mathbb{Z}_{4591}$, there are too many small polymul. in Schönhage/Nussbaumer $(4\times)$.
- Mulmod. in $\mathbb{Z}_{4591}$ is not very fast $\longrightarrow$ huge impact for Schönhage/Nussbaumer.

A Successful Story of Schönhage/Nussbaumer I

- Saber (`saber`):
  - Matrix-vector product $M \cdot v$ over $\mathbb{Z}_{2^{13}}[x]/\langle x^{256}+1 \rangle$.
  - Approaches:
    - Schönhage/Nussbaumer over $\mathbb{Z}_{2^{13}}$.
    - RNS $(\mathbb{Z}_{3329}, \mathbb{Z}_{7681})$ for mul.-based Cooley–Tukey.
- Cortex-M3, Armv7-M, 32-bit registers.
- Mul. in $\mathbb{Z}_{2^{1,\dots,32}}$ are very fast.
- Mulmod. in $\mathbb{Z}_{3329}, \mathbb{Z}_{7681}$ are slow.

	Schönhage/Nussbaumer	Mul.-based
Work	Submitted	[ACC+22]
Poly. ring	$\frac{\mathbb{Z}_{2^{13}}[x]}{\langle x^{256}+1 \rangle}$	$\frac{(\mathbb{Z}_{3329} \times \mathbb{Z}_{7681})[x]}{\langle x^{256}+1 \rangle}$
Cycles	272k	391k
Comment		Cooley–Tukey, RNS

A Successful Story of Schönhage/Nussbaumer II

	Schönhage/Nussbaumer	Mul.-based
Poly. ring	$\frac{\mathbb{Z}_{2^{13}}[x]}{\langle x^{256}+1\rangle}$	$\frac{(\mathbb{Z}_{3329}\times\mathbb{Z}_{7681})[x]}{\langle x^{256}+1\rangle}$
Total cycles	272k	391k
Tran. cycles	171k	284k
Small polymul. cycles	101k	107k

- $\#$ small polymul. in Schönhage/Nussbaumer is roughly $2\times$ than in RNS.
- Mul. in $\mathbb{Z}_{2^{21}}$ is extremely fast compared to mulmod. in $\mathbb{Z}_{3329} \times \mathbb{Z}_{7681}$.
- The large $\#$ of small polymul. over $\mathbb{Z}_{2^{21}}$ is still fast.
- Transformation in Schönhage/Nussbaumer is fast since we only need add., whereas we need a lot of mul. (in $\mathbb{Z}_{3329} \times \mathbb{Z}_{7681}$) in the mul.-based one.

Takeaway

- Schönhage/Nussbaumer is fast when mul. in R is fast.
- Choose a scheme comes with $R = \mathbb{Z}_{2^k}$. For example, Saber ($k = 13$) and NTRU ($k = 11 \sim 14$).
- Implement on a platform with 32-bit mul. instruction and no other longer mul. instructions.
  - Cortex-M3.
- Schönhage/Nussbaumer should be the champion.

Thank you for listening.

Reference I

[ACC+22] Amin Abdulrahman, Jiun-Peng Chen, Yu-Jia Chen, Vincent Hwang, Matthias J. Kannwischer, and Bo-Yin Yang, *Multi-moduli NTTs for Saber on Cortex-M3 and Cortex-M4*, IACR Transactions on Cryptographic Hardware and Embedded Systems **2022** (2022), no. 1, 127–151, https://tches.iacr.org/index.php/TCHES/article/view/9292.

[BBCT22] Daniel J. Bernstein, Billy Bob Brumley, Ming-Shing Chen, and Nicola Tuveri, *OpenSSLNTRU: Faster post-quantum TLS key exchange*, 31st USENIX Security Symposium (USENIX Security 22), 2022, https://www.usenix.org/conference/usenixsecurity22/presentation/bernstein, pp. 845–862.

[CK91] David G. Cantor and Erich Kaltofen, *On Fast Multiplication of Polynomials over Arbitrary Algebras*, Acta Informatica **28** (1991), no. 7, 693–701, https://link.springer.com/article/10.1007/BF01178683.

Reference II

[Für09] Martin Fürer, *Faster Integer Multiplication*, SIAM Journal on Computing **39** (2009), no. 3, 979–1005, `https://doi.org/10.1137/070711761`.

[HvdH21] David Harvey and Joris van der Hoeven, *Integer multiplication in time O (n log n)*, Annals of Mathematics **193** (2021), no. 2, 563–617, `https://annals.math.princeton.edu/2021/193-2/p04`.

[Hwa24] Vincent Hwang, *Pushing the Limit of Vectorized Polynomial Multiplication for NTRU Prime*, To appear at ACISP 2024, currently available at `https://eprint.iacr.org/2023/604`.

[Nus80] Henri J. Nussbaumer, *Fast Polynomial Transform Algorithms for Digital Convolution*, IEEE Transactions on Acoustics, Speech, and Signal Processing **28** (1980), no. 2, 205–215, `https://ieeexplore.ieee.org/document/1163372`.

Reference III

[Sch77] Arnold Schönhage, *Schnelle Multiplikation von Polynomen über Körpern der Charakteristik 2*, Acta Informatica **7** (1977), no. 4, 395–398, `https://link.springer.com/article/10.1007/bf00289470`.

[SS71] Arnold Schönhage and Volker Strassen, *Schnelle Multiplikation großer Zahlen*, Computing **7** (1971), no. 3-4, 281–292, `https://link.springer.com/article/10.1007/BF02242355`.