Max Planck Institute for Security and Privacy, Bochum, Germany

Pushing the Limit of Vectorized Polynomial Multiplications for NTRU Prime

Vincent Hwang

July 16, 2024

Goals

- Why homomorphisms of power-of-two dimensions frequently admit efficient vectorzation?
- Which homomorphisms admit efficient vectorization?
- Vectorization, formally: vectorization-friendliness, permutation-friendliness, Toeplitz matrix-vector product.
  - Driven by implementation experience.
- Polynomial multiplications in NTRU Prime (parameter set `ntrulpr761/sntrup761`)

$$\frac{\mathbb{Z}_{4591}[x]}{\langle x^{761} - x - 1\rangle} \cong \mathbb{F}_{4591^{761}}.$$

- $R = \mathbb{Z}_{4591}$ unless stated otherwise $\longrightarrow$ elements are stored as halfwords.

Vector Instruction Sets/Extensions

Vector Instruction Sets/Extensions

Overview of Vector Instruction Sets/Extensions		
	Armv8-A Neon	AVX2
Architecture	Armv8-A	x86
# vector registers	32	16
# bits in each vec.	128	256
# halfwords in each vec.	8	16
Vector-by-vector (halfword data)	add./sub./mul.	add./sub./mul.
Vector-by-scalar (halfword data)	mul.	None

Vectorization-Friendliness

Vectorization-Friendliness

- A transformation is vectorization-friendly if
  - it amounts to vector-by-vector arithmetic;
  - it results in subproblems of power-of-two dimensions.
- Let v be the number of elements in a vector.
- Conceptually, f is vectorization-friendly if it operates over chunks of size v.
  - $\exists f', f = f' \otimes I_v$.
- Formally (see paper for details):
  - $M_f = \prod_i (M_{f_i} \otimes I_v) S_{f_i}$.
  - M_f: matrix representation of f.
  - S_{f_i} must be a block diagonal matrix with each block
    - a diagonal matrix or
    - a cyclic/negacyclic shift matrix.
  - Diagonal matrix: component-wise multiplications.
  - Cyclic/negacyclic shift matrix: permutations/memory loads and stores.
- Dimension of f over R must be a multiple of v.

Example: Radix-2 Cooley–Tukey FFT

- Principal n-th root of unity ω_n:
  - $R = \mathbb{Z}_q$, prime q: n must divide $q - 1$.
- Radix-2, $n = 2^k$:

$$
\begin{tikzcd}
 & & & \frac{R[x]}{\left\langle x^{2^k} - 1 \right\rangle} \arrow[dll] \arrow[drr] & & & \\
 & \frac{R[x]}{\left\langle x^{2^{k-1}} - 1 \right\rangle} \arrow[dl] \arrow[dr] & & & & \frac{R[x]}{\left\langle x^{2^{k-1}} + 1 \right\rangle} \arrow[dl] \arrow[dr] & \\
\frac{R[x]}{\left\langle x^{2^{k-2}} - 1 \right\rangle} & & \frac{R[x]}{\left\langle x^{2^{k-2}} + 1 \right\rangle} & & \frac{R[x]}{\left\langle x^{2^{k-2}} - \omega_4 \right\rangle} & & \frac{R[x]}{\left\langle x^{2^{k-2}} + \omega_4 \right\rangle}
\end{tikzcd}
$$

Example: Vectorizing Radix-2 Cooley–Tukey

- $v | 2^k$.
- $R' = R[x] / \langle x^v - y \rangle$.
- $f = f' \otimes I_v$.
- f operates over R, and f' operates over R'.

Scalar View

$$\begin{array}{c} R[x] \Big/ \left\langle x^{2^k} - 1 \right\rangle \\ \Big\downarrow f \\ \prod_i R[x] \Big/ \left\langle x^v - \omega^i_{2^k/v} \right\rangle \end{array}$$

Vector View

$$\begin{array}{c} R'[y] \Big/ \left\langle y^{2^k/v} - 1 \right\rangle \\ \Big\downarrow f' \otimes I_v \\ \prod_i R'[y] \Big/ \left\langle y - \omega^i_{2^k/v} \right\rangle \end{array}$$

Permutation-Friendliness

Motivation of Permutation-Friendliness

- Vectorization with vector-by-vector arithmetic (conceptually):

$$f_0 \otimes I_v \longrightarrow M \longrightarrow f_1 \otimes I_v$$

Interleave

Vectorization-friendly

Permutation-friendly

- Power-of-two multiple of subproblems of equal power-of-two multiple dimensions.
- Rader-17:
  - 17 size-96 polynomials.
- Truncated Rader-17:
  - 16 size-96 polynomials.

Rader-17 FFT

- $4591 - 1 = 2 \cdot 3^3 \cdot 5 \cdot 17 \longrightarrow \exists \omega_{17}, \omega_3, \omega_2$.
- $R[x]/\langle x^{17} - 1\rangle \cong \prod_{i=0,\ldots,16} R[x]/\langle x - \omega_{17}^{i}\rangle$, mainly a size-16 cyclic convolution.
- Theory: [Rad68].

$$
\begin{tikzcd}[column sep=small]
 & & \frac{R[x]}{\langle x^{1632}-1\rangle} \arrow[dll] \arrow[dl] \arrow[dr] \arrow[drr] & & \\
\frac{R[x]}{\langle x^{96}-1\rangle} & \frac{R[x]}{\langle x^{96}-\omega_{17}\rangle} & \cdots & \frac{R[x]}{\langle x^{96}-\omega_{17}^{15}\rangle} & \frac{R[x]}{\langle x^{96}-\omega_{17}^{16}\rangle}
\end{tikzcd}
$$

Our Truncated Rader-17 FFT

- $4591 - 1 = 2 \cdot 3^3 \cdot 5 \cdot 17 \longrightarrow \exists \omega_{17}, \omega_3, \omega_2$.
- $R[x]/\langle \Phi_{17}(x)\rangle \cong \prod_{i=1,\ldots,16} R[x]/\left\langle x - \omega_{17}^i \right\rangle$ as a size-16 cyclic convolution.
- Theory: [Ber23].

$$\frac{R[x]}{\langle \Phi_{17}(x^{96})\rangle} = \frac{R[x]}{\langle x^{1536}+x^{1440}+\cdots+1\rangle}$$

$$\cancel{\frac{R[x]}{\langle x^{96}-1\rangle}} \qquad \frac{R[x]}{\langle x^{96}-\omega_{17}\rangle} \qquad \cdots \qquad \frac{R[x]}{\left\langle x^{96}-\omega_{17}^{15}\right\rangle} \qquad \frac{R[x]}{\left\langle x^{96}-\omega_{17}^{16}\right\rangle}$$

Permutation-Friendliness

- A transformation g is permutation-friendly if it is vectorization-friendly up to suitable interleaving.
- Power-of-two multiple of subproblems of equal power-of-two multiple dimensions.
- Rader-17:
  - 17 size-96 polynomials.
  - Not permutation-friendly.
- Truncated Rader-17:
  - 16 size-96 polynomials.
  - Permutation-friendly.
- Formally, $M_g = \prod_i S_{g_i} M_{g_i}$ (see paper for details).
  - M_g: matrix representation of g.
  - S_{g_i}: an interleaving matrix.
  - M_{g_i} vectorization-friendly.
- Dimension of g over R must be a multiple of v^2.

Toeplitz Matrix-Vector Product

Toeplitz Matrix-Vector Product

$$\left(a_0 + a_1 x + a_2 x^2 + a_3 x^3\right)\left(b_0 + b_1 x + b_2 x^2 + b_3 x^3\right) \in R[x]/\langle x^4 - \zeta\rangle .$$

$$\begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} a_0 & \zeta a_3 & \zeta a_2 & \zeta a_1 \\ a_1 & a_0 & \zeta a_3 & \zeta a_2 \\ a_2 & a_1 & a_0 & \zeta a_3 \\ a_3 & a_2 & a_1 & a_0 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

- Vector-by-scalar multiplications:

$$\begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} b_0 + \begin{pmatrix} \zeta a_3 \\ a_0 \\ a_1 \\ a_2 \end{pmatrix} b_1 + \begin{pmatrix} \zeta a_2 \\ \zeta a_3 \\ a_0 \\ a_1 \end{pmatrix} b_2 + \begin{pmatrix} \zeta a_1 \\ \zeta a_2 \\ \zeta a_3 \\ a_0 \end{pmatrix} b_3$$

- Neon vector-by-scalar multiplication instructions for $R[x]/\langle x^{mv} - \zeta\rangle$.

Transformations

Transformations

- Armv8-A Neon.
  - [HLY24]:
    - Rader-17, Good–Thomas, Cooley–Tukey, Bruun.
    - Vector-by-vector mul.
    - Vectorization-friendly but not permutation-friendly.
  - This work:
    - Truncated Rader-17, Good–Thomas, Cooley–Tukey, Toeplitz matrix-vector product.
    - Vector-by-scalar mul.
    - Vectorization-friendly + Toepliz matrix-vector product.
- AVX2, vector-by-vector mul.
  - [BBCT22]:
    - Schönhage, Nussbaumer.
    - Vectorization-friendly and permutation-friendly.
  - This work:
    - Truncated Rader-17, Good–Thomas, Cooley–Tukey, Bruun.
    - Vectorization-friendly and permutation-friendly.
    - $\frac{1}{4}$ small-dimensional polymul. compared to [BBCT22].
- Survey: `https://eprint.iacr.org/2023/1962`.

Results

Polynomial Multiplications

Table: Performance cycles of polynomial multiplications over $\mathbb{Z}_{4591}$ for `sntrup761`.

AVX2				
	[BBCT22]*	This work	[BBCT22]*	This work
	Haswell		Skylake	
`mulcore` $(\mathbb{Z}_{4591}[x])$	23 460	**12 336** $(1.90\times)$	20 070	**9 778** $(2.05\times)$
`polymul` $\left(\frac{\mathbb{Z}_{4591}[x]}{\langle x^{761}-x-1\rangle}\right)$	25 356	**12 760** $(1.99\times)$	21 364	**9 876** $(2.16\times)$
Neon				
	[HLY24]	This work	[HLY24]*	This work
	Cortex-A72		Apple M1 Pro	
`mulcore` $(\mathbb{Z}_{4591}[x])$	37 475	**29 909** $(1.25\times)$	8 120	**6 508** $(1.25\times)$
`polymul` $\left(\frac{\mathbb{Z}_{4591}[x]}{\langle x^{761}-x-1\rangle}\right)$	39 788	**30 912** $(1.29\times)$	9 091	**6 697** $(1.36\times)$

* Our own benchmarks.

Scheme (AVX2)

Table: Overall performance of our AVX2 implementation on Haswell and Skylake.

AVX2				
	Haswell		Skylake	
	[BBCT22]	This work	[BBCT22]	This work
Batch key gen.	154 552	**136 003** (−12.0%)	129 159	**118 939** (−7.9%)
	SUPERCOP	This work	SUPERCOP	This work
Encapsulation	47 464	**44 108** (−7.1%)	40 653	**36 486** (−10.3%)
Decapsulation	56 064	**50 080** (−10.7%)	47 387	**41 070** (−13.3%)

Scheme (Neon)

Table: Overall performance of our Neon implementation on Cortex-A72 and Apple M1.

	Neon			
	Cortex-A72		Apple M1 Pro	
	[HLY24]	This work	[HLY24]	This work
Key generation	6 574 055	**6 539 849** (-0.5%)	1 813 947	**1 806 741** (-0.4%)
Encapsulation	150 054	**140 107** (-6.6%)	64 924	**62 959** (-3.0%)
Decapsulation	159 286	**135 184** (-15.1%)	43 778	**38 196** (-12.8%)

Take Away

- Many more transformations other than Cooley–Tukey with efficient vectorization.
  - Truncated Rader-17.
- Vectorzation.
  - Vectorization-friendliness.
  - Permutation-friendliness.
  - Toeplitz matrix-vector product.
- Choose a $g \circ f$:
  - With vector-by-vector mul., decide if the all the following hold. Re-choose $g \circ f$ otherwise.
    - f vectorization-friendly.
    - g permutation-friendly.
  - With vector-by-scalar mul., decide if the all the following hold. Re-choose $g \circ f$ otherwise.
    - f vectorization-friendly.
    - g amounts to Toeplitz matrix-vector products.
- Results of polynomial multiplications:
  - AVX2 (Haswell and Skylake): 1.90 to 2.16 times faster.
  - Neon (Cortex-A72 and Apple M1 Pro): 1.25 to 1.36 times faster.

Thanks for listening

Paper (IACR ePrint): `https://eprint.iacr.org/2023/604`

Artifact: `https://github.com/vector-polymul-ntru-ntrup/NTRU_Prime_truncation`

Slides: `https://vincentvbh.github.io/slides/ACISP2024_2_97_slide.pdf`

Reference I

[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, pp. 845–862.

[Ber23] Daniel J. Bernstein, *Fast norm computation in smooth-degree abelian number fields*, Research in Number Theory **9** (2023), no. 4, 82.

[HLY24] Vincent Hwang, Chi-Ting Liu, and Bo-Yin Yang, *Algorithmic Views of Vectorized Polynomial Multipliers – NTRU Prime*, International Conference on Applied Cryptography and Network Security, Springer, 2024, pp. 24–46.

[Rad68] Charles M. Rader, *Discrete Fourier Transforms When the Number of Data Samples Is Prime*, Proceedings of the IEEE **56** (1968), no. 6, 1107–1108.

Vectorization for NTRU Prime?

- Radix-2 Cooley–Tukey.
  - $4591 - 1 = 2 \cdot 3^3 \cdot 5 \cdot 17 \longrightarrow k = 0, 1.$
  - Unfortunately, we don't have ω_{2^k} with a high-power 2^k in $\mathbb{Z}_{4591}$.

	[BBCT22]	[HLY24]	**This work**
ISA/extension	AVX2	Neon	**Neon/AVX2**
Domain	$\frac{R[x]}{\langle (x^{1024}+1)(x^{512}-1) \rangle}$	$\frac{R[x]}{\langle x^{1632}-1 \rangle}$	$\boldsymbol{\frac{R[x]}{\langle \Phi_{17}(x^{96}) \rangle}}$
FFT	Schönhage	Rader-17 + `GT`	**truncated Rader-17 +** `GT`
Image	$\left(\frac{R[x]}{\langle x^{64}+1 \rangle} \right)^{48}$	$\prod_i \frac{R[x]}{\langle x^{16}-\omega_{102}^i \rangle}$	$\boldsymbol{\left(\prod \frac{R[x]}{\langle x^{16} \pm 1 \rangle} \right)^{48}}$

Table: Summary of vectorization-friendly approaches. `GT` stands for Good–Thomas.

Comparisons of Permutation-Friendliness to Prior Works

Overview of Permutation-Friendly Approaches with AVX2		
	[BBCT22]	**This work**
Domain	$\left(\frac{R[x]}{\langle x^{64}+1\rangle}\right)^{48}$	$\boldsymbol{\left(\prod \frac{R[x]}{\langle x^{16}\pm 1\rangle}\right)^{48}}$
FFT	Nussbaumer	`CT + Bruun`
Image	$\left(\frac{R[z]}{\langle z^{8}+1\rangle}\right)^{768}$	$\boldsymbol{\left(\prod \frac{R[x]}{\langle x^{8}\pm 1\rangle} \times \prod \frac{R[x]}{\langle x^{8}\pm\sqrt{2}x^{4}+1\rangle}\right)^{48}}$
Follow up polymul.	Recursive `K`	`K`
Multiplication instruction	Vector-by-vector	**Vector-by-vector**