NCC-Sign: A New Lattice-based Signature Scheme using Non-Cyclotomic Polynomials

Kyung-Ah Shim, Jeongsu Kim, and Youngjoo An

Cryptographic Technology Research Team, NIMS

Contents

- Design Principles and Features
- Digital Signature Algorithm Specification
- Security Proof and Security Analysis
- Parameter Selection
- Implementation
- Future Plan

Design Principles and Features

- Majority of efficient lattice-based schemes are based on the structured lattices using power-of-2 cyclotomics by default.
  - CRYSTALS-Kyber, Saber, CRYSTALS-Dilithium, and Falcon use the 2n-th cyclotomic polynomial Ø(x)(X) = X^n+1, n is a power of 2.
- Potential threads on about on attacks exploited unnecessary algebraic structures
  - Q[x]/(Ø(x)) has many subfields
  - Q[x]/(Ø(x)) has small Galois Group
  - Z/q[x]/(Ø(x)) -> smaller ring: some attacks using ring homomorphisms
- Polynomial-time quantum attacks
  - Soliloquy, the cyclotomic case of Gentry's original FHE (STOC 2009) and the cyclotomic case of the Garg-Gentry-Halevi scheme under plausible assumptions
  - S-unit attack, Twisted-PHS: Approx-SVP on ideal lattices

Design Principles and Features

- Non-cyclotomic Lattice-based signature scheme
  - Stronger security guarantee than cyclotomic counterparts and better efficiency than unstructured lattice-based schemes.
  - The first lattice-based signature using a prime degree large Galois group inert modulus with Ø(x) = x^p − x − 1 to remove the structures that were the causes of the previous attacks.
- Fiat-Shamir with aborts paradigm
  - Our scheme combines the Bai-Galbraith scheme with several improvements from previous lattice-based schemes including CRYSTALS-Dilithium.
- Flexible Choice of Parameters
  - No parameter jumps
- Protection against Side-Channel Attacks.
  - Most of the side channel analysis targeted the data dependent side-channel leakage from the Gaussian sampling, the rejection sampling components and the computation of NTT.
  - Our scheme uses a uniform distribution and Toom-Cook and Karatsuba algorithm.

Features of NCC-Sign

- NCC-Sign: R_q=Z_q[X]/Ø(x)
  - NTRU Prime field, Ø(x) = x^p − x − 1
  - Cyclotomic counterpart: trinomial Ø(x) = x^n − x^n/2 − 1
- Several Optimizations
  - PK compression: use a random seed for a and omit the lower-bits of t, PK=(a, t=as_1+s_2)
  - Optimized SampleInBall function: use two separate polynomials, speed-up from 9% to 24%
- Security
  - R-LWE and SelfTargetRSIS (a variant of R-SIS) assumptions, Strong unforgeabilty in QROM
  - Beyond unforgeabilty: a signature can be identified with a unique public key and a message
- Cost Analysis
  - Best known algorithm: generic algorithms for finding short vectors in lattices
  - Parameters: concrete and conservative parameters
- Implementation
  - Toom-Cook and Karatsuba polynomial multiplications
  - All operations are implemented in constant time.

Digital Signature Algorithm Specification

- Key generation algorithm
  - ➢ Public key compression
    - ✓ Omit the lower-bits of t
    - ✓ Compute hints as part of a signature

Algorithm 2: $\text{Decompose}_q(r, \alpha)$

1 $r := r \mod^+ q$
2 $r_0 := r \mod^\pm \alpha$
3 **if** $r - r_0 = q - 1$ **then**
4 $\quad r_1 := 0$
5 $\quad r_0 := r_0 - 1$
6 **else**
7 $\quad r_1 := (r - r_0)/\alpha$
8 **return** (r_1, r_0)

Algorithm 3: $\text{UseHint}_q(h, r, \alpha)$

1 $m := (q-1)/\alpha$
2 $(r_1, r_0) := \text{Decompose}_q(r, \alpha)$
3 **if** $h = 1$ *and* $r_0 > 0$ **then**
4 $\quad$ **return** $(r_1 + 1) \mod^+ m$
5 **if** $h = 1$ *and* $r_0 \leq 0$ **then**
6 $\quad$ **return** $(r_1 - 1) \mod^+ m$
7 **return** r_1

Algorithm 4: $\text{Power2Round}_q(r, d)$

1 $r := r \mod^+ q$
2 $r_0 := r \mod^\pm 2^d$
3 **return** $((r - r_0)/2^d, r_0)$

Algorithm 5: $\text{HighBits}_q(r, \alpha)$

1 $(r_1, r_0) := \text{Decompose}_q(r, \alpha)$
2 **return** r_1

Algorithm 6: $\text{LowBits}_q(r, \alpha)$

1 $(r_1, r_0) := \text{Decompose}_q(r, \alpha)$
2 **return** r_0

Algorithm 7: $\text{MakeHint}_q(z, r, \alpha)$

1 $r_1 := \text{HighBits}_q(r, \alpha)$
2 $v_1 := \text{HighBits}_q(r + z, \alpha)$
3 **return** $[\![r_1 \neq v_1]\!]$

Digital Signature Algorithm Specification

- Optimized SampleInBall function
  - Choose two separate polynomials, c_i of degree p_i-1
  - $\mathbf{c} = \mathbf{c}_2 + X^{p_2}\mathbf{c}_1$

Algorithm 1: $\mathsf{SampleInBall}_{p,\tau}(\rho)$.
Create a random p-element array with τ ± 1's and $p - \tau$ 0's.
Use the input seed ρ (and an XOF) to generate the randomness needed in Step 3 and 4.

1 Initialize $\mathbf{c} = c_0 c_1 \ldots c_{p-1} = 00 \ldots 0$
2 **for** $i := p - \tau$ *to* $p - 1$ **do**
3 $\quad j \leftarrow \{0, 1, \ldots, i\}$
4 $\quad s \leftarrow \{0, 1\}$
5 $\quad c_i := c_j$
6 $\quad c_j := (-1)^s$
7 **return** $\mathbf{c}$

Digital Signature Algorithm Specification

- Key generation algorithm
  - Public key compression
  - Two separate seeds, ζ and ζ' to generate a public key a and a secret key $(\mathbf{s}_1, \mathbf{s}_2, K)$, respectively
  - Secret key K for deterministic signing

Algorithm 8: KeyGen

1 $(\zeta, \zeta') \leftarrow \{0,1\}^{256} \times \{0,1\}^{256}$
2 $(\xi_1, \xi_2, K) \in \{0,1\}^{256} \times \{0,1\}^{256} \times \{0,1\}^{256} := \mathrm{H}(\zeta')$
3 $\mathbf{a} \in R_q := \mathsf{ExpandA}(\zeta)$
4 $(\mathbf{s}_1, \mathbf{s}_2) \in S_\eta \times S_\eta := \mathsf{ExpandS}(\xi_1, \xi_2)$
5 $\mathbf{t} := \mathbf{a}\mathbf{s}_1 + \mathbf{s}_2$
6 $(\mathbf{t}_1, \mathbf{t}_0) := \mathsf{Power2Round}_q(\mathbf{t}, d)$
7 $tr \in \{0,1\}^{256} := \mathrm{H}(\zeta \parallel \mathbf{t}_1)$
8 **return** $(pk = (\zeta, \mathbf{t}_1), sk = (\zeta, tr, K, \mathbf{s}_1, \mathbf{s}_2, \mathbf{t}_0))$

bound ζ with the public key t_1

Digital Signature Algorithm Specification

- Signature generation algorithm
  - Recover the public key
  - Compute y, ay, high bit of ay
  - Compute c
  - Compute z=y+c s_1
  - Generate hint h
  - σ =(z, h, c)

Algorithm 9: $\mathrm{Sign}(sk, M)$

1 $\mathbf{a} \in R_q := \mathsf{ExpandA}(\zeta)$
2 $\mu \in \{0,1\}^{512} := \mathrm{H}(tr \parallel M)$ ← Bound the public key and the message
3 $\kappa := 0,\ (\mathbf{z}, \mathbf{h}) := \perp$
4 $\rho \in \{0,1\}^{512} := \mathrm{H}(K \parallel \mu)$ (or $\rho \leftarrow \{0,1\}^{512}$ for randomized signing)
5 **while** $(\mathbf{z}, \mathbf{h}) = \perp$ **do**
6 $\quad \mathbf{y} \in \tilde{S}_{\gamma_1} := \mathsf{ExpandMask}(\rho, \kappa)$
7 $\quad \mathbf{w} := \mathbf{ay}$
8 $\quad \mathbf{w}_1 := \mathsf{HighBits}_q(\mathbf{w}, 2\gamma_2)$
9 $\quad \tilde{c} \in \{0,1\}^{256} := \mathrm{H}(\mu \parallel \mathbf{w}_1)$
10 $\quad \mathbf{c} \in B_\tau := \mathsf{SampleInBall}_{p,\tau}(\tilde{c})$
11 $\quad \mathbf{z} := \mathbf{y} + \mathbf{cs}_1$
12 $\quad \mathbf{r}_0 := \mathsf{LowBits}_q(\mathbf{w} - \mathbf{cs}_2, 2\gamma_2)$
13 $\quad$ **if** $\|\mathbf{z}\|_\infty \geq \gamma_1 - \beta$ **or** $\|\mathbf{r}_0\|_\infty \geq \gamma_2 - \beta$ **then**
14 $\quad\quad (\mathbf{z}, \mathbf{h}) := \perp$
15 $\quad$ **else**
16 $\quad\quad \mathbf{h} := \mathsf{MakeHint}_q(-\mathbf{ct}_0, \mathbf{w} - \mathbf{cs}_2 + \mathbf{ct}_0, 2\gamma_2)$
17 $\quad\quad$ **if** $\|\mathbf{ct}_0\|_\infty \geq \gamma_2$ **or** the # of 1's in $\mathbf{h}$ is greater than ω
18 $\quad\quad$ **then**
19 $\quad\quad\quad (\mathbf{z}, \mathbf{h}) := \perp$
20 $\quad\quad \kappa := \kappa + 1$
21 **return** $\sigma = (\tilde{c}, \mathbf{z}, \mathbf{h})$

Digital Signature Algorithm Specification

- Signature verification algorithm (pk, M, σ)
  - Recover the public key from a seed
  - Validation check using hint

Algorithm 10: Verify$(pk, M, \sigma) = (\tilde{c}, \mathbf{z}, \mathbf{h})$

1 $\mathbf{a} \in R_q := \mathsf{ExpandA}(\zeta)$
2 $\mu \in \{0,1\}^{512} := \mathrm{H}(\mathrm{H}(\zeta \parallel \mathbf{t}_1) \parallel M)$
3 $\mathbf{c} := \mathsf{SampleInBall}(\tilde{c})$
4 $\mathbf{w}_1' := \mathsf{UseHint}_q(\mathbf{h}, \mathbf{az} - \mathbf{ct}_1 \cdot 2^d, 2\gamma_2)$
5 **return** $[\![\|\mathbf{z}\|_\infty < \gamma_1 - \beta]\!]$ **and** $[\![\tilde{c} = \mathrm{H}(\mu \parallel \mathbf{w}_1')]\!]$ **and**
$[\![\#$ of 1's in $\mathbf{h}$ is $\leq \omega]\!]$

Security Proof and Security Analysis

Definition 2.2 (Decision RLWE Problem.) Given a pair $(\mathbf{a}, \mathbf{t})$ decode with non-negligible advantage, whether it came from the RLWE distribution or it was generated uniformly at random from $R_q \times R_q$. The advantage of the adversary $\mathcal{A}$ in solving decisional RLWE problem over the ring R_q is

$$\begin{aligned}\mathrm{Adv}_{\chi}^{RLWE}(\mathcal{A}) := |Pr[b = 1 \,|\, a, t \leftarrow R_q; b \leftarrow \mathcal{A}(\mathbf{a}, \mathbf{t}) \\ - Pr[b = 1 \,|\, \mathbf{a} \leftarrow R_q, \mathbf{s}_1, \mathbf{s}_2 \leftarrow \chi; b \leftarrow \mathcal{A}(\mathbf{a}, \mathbf{a}\mathbf{s}_1 + \mathbf{s}_2)]| \,.\end{aligned}$$

Definition 2.3 (RSIS Problem.) The advantage of the adversary $\mathcal{A}$ to solve RSIS problem over the ring R_q is

$$\mathrm{Adv}_{\gamma}^{\mathrm{RSIS}}(\mathcal{A}) := Pr[0 < \|\vec{\mathbf{y}}\|_\infty \leq \gamma \wedge \begin{bmatrix}1 & \mathbf{a}_1 & \mathbf{a}_2\end{bmatrix} \cdot \vec{\mathbf{y}} = 0 \,|\, \mathbf{a}_1, \mathbf{a}_2 \leftarrow R_q; \vec{\mathbf{y}} \leftarrow \mathcal{A}(\mathbf{a}_1, \mathbf{a}_2)].$$

Definition 2.4. (SelfTargetRSIS Problem). For the cryptographic hash function H, the advantage of $\mathcal{A}$ to solve SelfTargetRSIS problem $\mathrm{Adv}_{\mathrm{H},\gamma}^{\mathrm{SelfTargetRSIS}}(\mathcal{A})$ is defined as

$$Pr\left[\begin{array}{c} 0 \leq \|\vec{\mathbf{y}}\|_\infty \leq \gamma \wedge \\ \mathrm{H}(\mu \| \begin{bmatrix}1 & \\ & \mathbf{a}_1\ \mathbf{a}_2\end{bmatrix} \cdot \vec{\mathbf{y}}) = \mathbf{c} \end{array} \middle| \mathbf{a}_1, \mathbf{a}_2 \leftarrow R_q; \left(\vec{\mathbf{y}} := \begin{bmatrix}\mathbf{r}_1 \\ \mathbf{r}_2 \\ \mathbf{c}\end{bmatrix}, \mu\right) \leftarrow \mathcal{A}^{|\mathrm{H}(\cdot)\rangle}(a_1, a_2) \right].$$

Security Proof and Security Analysis

- Existential unforgeability in (Q)ROM
  - For existential unforgeability against chosen-message attacks (EUF-CMA), existential unforgeability against no-message attacks (EUF-NMA) is sufficient if a signature scheme is zero-knowledge and deterministic.
  - Based on RLWE, SelfTargetRSIS problem, our scheme achieves zero-knowledge and EUF-NMA in (Q)ROM.
- Security analysis
  - RLWE, RSIS, SelfTargetRSIS, Primal attack, dual lattice attack
  - MATZOV: improved dual lattice attack

Parameter Selection

- Selection of p and q
  - NTRU Prime KEM: It requires relatively small p and q, p=653, q = 4621.
  - Dilithium: Module-LWE
    - a single prime q for the modulus for all security levels.
  - NCC-Sign
    - ✓ It needs inert modulus q, so our q is different at each security level.
    - ✓ Choose suitable p and q s.t. the expected number of repetitions is not too large for efficiency.
      - Expected number of repetitions in the rejection sampling is about $e^{p\beta(1/\gamma_1+1/\gamma_2)}$.
      - Satisfy $q \equiv 1 \mod 2\gamma_2$ for correct verification
      - Selection of q based on the SIS problem. The larger γ_2 the better efficiency, but less secure.
      - γ_1 : a power of two
      - γ_2 : $2\gamma_2 \approx \gamma_1$
      - $\eta = 2$.: Larger η makes the LWE problem harder, the cost of rejection sampling becomes inefficient since $\beta = 2\tau\eta$ in $e^{p\beta(1/\gamma_1+1/\gamma_2)}$.

p	q
1021	8348477, 8339581, 8333113
1429	8380087, 8376649, 8333131, 8332559
1913	8361623, 8343469, 8334383

Parameter Selection

- Cost analysis
  - Cost: cpu-cycles, Lattice estimator from https://github.com/malb/lattice-estimato
  - Core SVP estimate: BKZ-b calls the SVP oracle of dimension b
    - ✓ Solving shortest vector problem in a lattices of dimension b cost
      - classical security: $2^{0.265b}$ quantum security: $2^{0.292b}$
    - ✓ Classical security: $2^{a+0.292b}$, quantum security: $2^{a/2+0.265b}$

Parameter Selection

- Concrete parameters
  - ➢ It can be considered as optimized for key sizes.

Parameter/Seurity Level	I	III	V
p	1021	1429	1913
q	8339581	8376649	8343469
d [dropped bits from t] $(2^d\tau < \gamma_2)$	11	12	12
τ [# of ± 1's in c]	25	29	32
challenge entropy $[\log \binom{p}{\tau} + \tau]$	190	228	259
γ_1 [y coefficient range]	2^{17}	2^{18}	2^{19}
γ_2 [low-order rounding range]	$(q-1)/90$ $(= 92662)$	$(q-1)/56$ $(= 149583)$	$(q-1)/42$ $(= 198654)$
η [secret key range]	2	2	2
β	100	116	128
ω [max # of 1's in hint]	80	80	80
Exp. reps. $[\approx e^{p\beta(1/\gamma_1+1/\gamma_2)}]$	6.6	5.7	5.5
Public key size	1564	1997	2663
Secret key size	2266	3312	4402
Signature size	2458	3605	5055
Cost to SIS (BKZ b)	133.9 (411)	198.1 (629)	259.8 (839)
Quantum cost to SIS	115.9	173.9	229.7
Cost to LWE by estimator (BKZ b)	147.7 (413)	211.5 (641)	291.3 (924)
Quantum cost to LWE	123.0	182.0	255.6

Parameter Selection

- Conservative parameters
  - ➢ It can be considered as optimized for signing performance.

Parameter/Security Level	I^c	III^c	V^c
p	1201	1607	2039
q	17279291	17305741	17287423
d [dropped bits from t] $(2^d\tau < \gamma_2)$	12	13	13
τ [# of ± 1's in c]	32	32	32
challenge entropy $[\log \binom{p}{\tau} + \tau]$	241	254	265
γ_1 [y coefficient range]	2^{19}	2^{19}	2^{19}
γ_2 [low-order rounding range]	$(q-1)/70$ $(= 246847)$	$(q-1)/60$ $(= 288429)$	$(q-1)/58$ $(= 298059)$
η [secret key range]	2	2	2
β	128	128	128
ω [max # of 1's in hint]	80	80	80
Exp. reps. $[\approx e^{p\beta(1/\gamma_1+1/\gamma_2)}]$	2.5	3.02	3.95
Public key size	1984	2443	3091
Secret key size	2800	3914	4940
Signature size	3186	4251	5385
Cost to SIS (BKZ b)	155.5 (484)	218.1 (697)	289.7 (941)
Quantum cost to SIS	135.3	192.0	256.8
Cost to LWE (BKZ b)	167.3 (483)	229.3 (704)	298.1 (949)
Quantum cost to LWE	141.1	198.4	262.0

Cyclotomic counterpart

- Cyclotomic trinomial counterpart
  - $\phi(X) = X^n - X^{n/2} + 1$, q=2^23
  - Use the degree of the polynomial of the form 2^a3^b for flexible choices of parameters

Parameter/Security Level	I	III	V
n	1024	1458	1944
q	2^{23}	2^{23}	2^{23}
d [dropped bits from t] $(2^d\tau < \gamma_2)$	12	12	13
τ [# of ± 1's in c]	25	29	32
challenge entropy $[\log \binom{p}{\tau} + \tau]$	190	230	263
γ_1 [y coefficient range]	2^{18}	2^{18}	2^{19}
γ_2 [low-order rounding range]	2^{17}	2^{17}	2^{18}
η [secret key range]	2	2	2
β	100	116	128
ω [max # of 1's in hint]	80	80	80
Exp. reps. $[\approx e^{n\beta(1/\gamma_1 + 1/\gamma_2)}]$	3.23	6.92	4.15
Public key size	1440	2037	2462
Secret key size	2400	3377	4713
signature size	2529	3678	5135
Cost to SIS (BKZ b)	130.9 (411)	203.6 (658)	260.9 (853)
Quantum cost to SIS	114.4	180.1	232.0
Cost to LWE by estimator (BKZ b)	148.1 (414)	216.1 (657)	296.4 (943)
Quantum cost to LWE	123.3	186.2	260.4

Cost Analysis on Cost Models

- Cost models
  - Lattice estimator: We use the default option (MATZOV) in the lattice estimator for the cost estimation, but there exist other cost models. We provide cost estimates of the RLWE problem on other cost models for reference.
    - ✓ `bdd' means that solving a bounded distance decoding problem in the lattice is the best attack strategy. Bounded distance decoding problem can be easily converted to a unique shortest vector problem by the embedding approach
    - ✓ `usvp' means that solving unique shortest vector problem is the best estimated strategy.
    - ✓ `bkw' means that Blum-Kalai-Wasserman which needs quite many samples for the attack to succeed.

Cost Analysis on Cost Models

- Cost analysis on cost models
  - Lattice estimator: `https://github.com/malb/lattice-estimator`.

  - ABFKSW20: $0.125\beta \log_2 \beta - 0.547\beta + 10.4 + \log_2 64 + \log_2 8d$,
  - ABLR21: $0.125\beta \log_2 \beta - 0.654\beta + 25.84 + \log_2 64 + \log_2 8d$,
  - ADPS16: 0.292β,
  - BDGL16: $0.292\beta + 16.4 + \log_2 8d$,
  - CheNgu12: $0.270\beta \log \beta - 1.019\beta + 16.103 + \log_2 100 + \log_2 8d$,
  - LaaMosPol14: $0.265\beta + 16.4 + \log_2 8d$.

Cost model	I (128)	III (192)	V (256)	I^c (128)	III^c (192)	V^c (256)
ABFKSW20	259.6 (usvp)	363.5 (bkw)	478.8 (bkw)	307.1 (bkw)	403.5 (bkw)	500.1 (bkw)
ABLR21	229.9 (usvp)	363.5 (bkw)	478.8 (bkw)	274.2 (usvp)	403.5 (bkw)	500.1 (bkw)
ADPS16	123.2 (usvp)	190.1 (usvp)	273.3 (usvp)	143.7 (usvp)	208.5 (usvp)	280.3 (usvp)
BDGL16	150.7 (bdd)	217.5 (bdd)	300.8 (bdd)	171.3 (bdd)	236.1 (bdd)	308.0 (bdd)
CheNgu12	270.8 (bkw)	363.5 (bkw)	478.8 (bkw)	307.1 (bkw)	403.5 (bkw)	500.1 (bkw)
Kyber	154.4 (bdd)	218.7 (bdd)	299.2 (bdd)	174.2 (bdd)	236.6 (bdd)	306.1 (bdd)
MATZOV	147.7 (bdd)	211.5 (bdd)	291.3 (bdd)	167.3 (bdd)	229.3 (bdd)	298.1 (bdd)
GJ21	154.4 (bdd)	218.7 (bdd)	299.2 (bdd)	174.2 (bdd)	236.6 (bdd)	306.1 (bdd)
LaaMosPol14	139.3 (bdd)	200.0 (bdd)	275.5 (bdd)	158.1 (bdd)	216.9 (bdd)	282.0 (bdd)

Cost Analysis on Cost Models

- Comparison with CRYSTALS-Dilithium parameters
  - At the security level I, our costs for the concrete parameter are comparable to those of CRYSTALS-Dilithium.
  - At the other security levels, our costs are higher than those of CRYSTALS-Dilithium. Obviously, in the conservative parameters, our costs are higher than those of CRYSTALS-Dilithium at all the security levels.

Cost model/Security Level	2 (I)	3(III)	5(V)
ABFKSW20	261.0 (usvp)	363.4 (bkw)	454.7 (bkw)
ABLR21	231.1 (usvp)	363.0 (usvp)	454.7 (bkw)
ADPS16	123.8 (usvp)	182.5 (usvp)	252.0 (usvp)
BDGL16	151.2 (bdd)	209.7 (bdd)	279.6 (bdd)
CheNgu12	270.9 (bkw)	363.4 (bkw)	454.7 (bkw)
Kyber	154.8 (bdd)	211.1 (bdd)	278.7 (bdd)
MATZOV	148.1 (bdd)	204.0 (bdd)	271.0 (bdd)
GJ21	154.8 (bdd)	211.1 (bdd)	278.7 (bdd)
LaaMosPol14	139.7 (bdd)	192.8 (bdd)	256.3 (bdd)

Reference Implementation

- Reference implementation
  - Target Platform: Intel(R) Core(TM) i7-12700K CPU, 3.60GHz
  - Each result is an average of 100,000 measurements for each function using the C programming language with GNU GCC version 7.5.0 compiler.
  - CPU cycles required by the key generation, signing and verification.

Algorithm/Security Level	I	II	III
KeyGen	1,257,562	2,386,408	4,202,722
Sign	16,174,808	28,184,328	49,062,056
Verify	2,444,616	4,765,774	8,342,102

Algorithm/Security Level	I^c	III^c	V^c
KeyGen	1,727,508	2,965,942	4,700,228
Sign	11,768,076	20,816,964	42,227,652
Verify	3,400,702	5,876,246	9,324,876

Future Plan

- AVX2-Optimzed implementation
  - Optimization for polynomial multiplication: Toom-Cook, Karashuba
  - Optimized SampleInball, special for of the modulus q
  - Implementation of cyclotomic counterpart

Parameter	p	τ	κ	p_1, p_2	τ_1, τ_2	Exp.reps. (new)	Speed-up
I	1021	25	190	510,511	104,76	5.44	1.21
III	1429	29	228	714,715	120,88	4.76	1.19
V	1913	32	259	956,957	128,96	4.42	1.24
I^c	1201	32	241	600,601	132,98	2.27	1.09
III^c	1607	32	254	803,804	132,98	2.7	1.11
V^c	2039	32	265	1019,1020	132,98	3.43	1.15

p	q	$q-1$
1021	8290297 $(= 2^{23} - 2^{16} - 2^{15} - 2^3 + 1)$	$2^3 * 3^3 * 7 * 5483$
1447	8126431 $(= 2^{23} - 2^{18} - 2^5 - 2^1 + 1)$	$2 * 3 * 5 * 13 * 67 * 311$
1913	6287329 $(= 2^{23} - 2^{21} - 2^{12} - 2^5 + 1)$	$2^5 * 3^3 * 19 * 383$
1279	16736257 $(= 2^{24} - 2^{15} - 2^{13} + 1)$	$2^{13} * 3^2 * 227$
1621	16252861 $(= 2^{24} - 2^{19} - 2^6 - 2^2 + 1)$	$2^2 * 3 * 5 * 13 * 67 * 311$
2099	16515073 $(= 2^{24} - 2^{18} + 1)$	$2^{18} * 3^2 * 7$

Thanks.