NTRU and Lattices

Mathilde Kermorgant

May 2024

Introduction

On the eve of quantum computers, most cryptosystems in use have been proven to break under quantum attacks, and none have been proven to resit them. Nevertheless, some cryptosystems continue to survive theoretical quantum attacks. One in particular, NTRU, short for "Number Theory Research Unit" or "Number Theorists 'R' Us", relies on the difficulty of factoring polynomials in a special kind of polynomial ring or finding short vectors in n dimensional lattices. No polynomial time quantum algorithms exist to solve these problems. Developed at Brown in 1996 by Hoffstein, Pipher and Silverman, it has the potential to become a cryptographic standard in the post-quantum world.

1 Polynomial Rings

NTRU's security relies on the difficulty of the following problem: given $h(x)$ a polynomial, find $f(x)$ and $g(x)$ such that $f(x)h(x) \equiv g(x)$ in some special polynomial ring. Those special polynomial rings and their arithmetic are defined below:

Definition 1.1. The polynomial ring over a ring R in variable X is the set of expressions of the form $a_0X^0 + ... + a_nX^n$ with coefficients $a_0, ..., a_n \in R$. We write this polynomial ring $R[X]$.

For NTRU, we restrict our attention to polynomial rings over $\mathbb{Z}$ and over $\mathbb{Z}/p\mathbb{Z}$ where p is a positive integer.

Definition 1.2. The ring of convolution of polynomials of rank N over $\mathbb{Z}$ is the quotient ring

$$R = \frac{\mathbb{Z}[X]}{(X^N - 1)}$$

The ring of convolution of polynomials of rank N over $\mathbb{Z}/p\mathbb{Z}$ where p is a positive integer is the quotient ring

$$R_p = \frac{\mathbb{Z}/p\mathbb{Z}[X]}{(X^N - 1)}$$

While at first these may seem daunting the additional information or constraints brought by taking the quotient of the polynomial ring by (X^N-1) can be summed up in the following way: taking the quotient by the ideal generated by X^N-1 is identifying X^N-1 with 0 in the new ring. Hence, in the ring of convolution, $X^N-1=0$ so $X^N=1$.

Example 1.1. *Take the convolution of polynomials R_5 of rank 3 over $\mathbb{Z}/5\mathbb{Z}$. The polynomial $1+x+x^3+17x^{16}-2x^{101}$ in R_5 is equal to $1+1+x+2x+3x^2=2+3x+3x^2$, which is much simpler.*

Remark (Very important). Every polynomial in the ring of convolution of polynomials of rank N has a unique representation as $a_0+a_1X+...+a_{N-1}X^{N-1}$ where a_0, ..., a_{N-1} are in $\mathbb{Z}$ or $\mathbb{Z}/p\mathbb{Z}$ depending on the choice of rings

While every polynomial in R has a unique representative in R_p simply by taking each coefficient modulo p, the mapping is not one to one. A polynomial in R_p may represent multiple polynomials in R. To avoid such concerns, we define a canonical way from taking a polynomial in R to a polynomial in R_p:

Definition 1.3 (Center-Lift)**.** Let $a(X)\in R_p$. The center-lift of $a(X)$ to R is the unique polynomial $a'(X)$ satisfying $a'(X) \bmod(q)\equiv a(x)$ with coefficients $-\frac{q}{2}<a'_i\leq\frac{q}{2}$.

Proposition 1.1. *Let $a(X)=a_0+a_1X+...+a_{N-1}X^{N-1}, b(X)=b_0+b_1X+...+b_{N-1}X^{N-1}\in R$ or $\mathbb{R}_p$. Then, $a(X)+b(X)=a_0+b_0+(a_1+b_1)X+...+(a_{N-1}+b_{N-1})X^{N-1}$.*

Proposition 1.2. *Let $a(X)=a_0+a_1X+...+a_{N-1}X^{N-1}, b(X)=b_0+b_1X+...+b_{N-1}X^{N-1}\in R$ or $\mathbb{R}_p$. Then, the multiplication $a(X)\star b(X)=c(X)$ is given by $c_i=\sum_{j+k\equiv i}a_jb_k$ where c_i is the coefficient of X^i in $c(X)$ and $0\leq i\leq N-1$.*

Remark (Disclaimer). The center lift does not commute with addition and multiplication. Nonetheless, they are congruent modulo p.

Remark. This is equivalent to multiplication in $\mathbb{Z}[X]$ or $\mathbb{Z}/p\mathbb{Z}[X]$ and then reducing modulo X^N-1. Moreover, the addition and multiplication operation inherit associativity, commutativity and distributivity from associativity, commutativity and distributivity in $\mathbb{Z}[X]$ or $\mathbb{Z}/p\mathbb{Z}[X]$.

Example 1.2. *Take the convolution of polynomials R_5 of rank 3 over $\mathbb{Z}/5\mathbb{Z}$. Let $a(X)=2+3X+3X^2$, and $b(X)=3+X$.*

$a(X)\star b(X)=(2\cdot3+3\cdot1)+(2\cdot1+3\cdot3)X+(3\cdot3+3\cdot1)X^2=4+X+2X^2$. $a(X)b(X)=6+11X+12X^2+3X^3\cong 4+X+2X^2$ in R_5, so the convolution of two polynomials is coincides with polynomial multiplication and reduction modulo X^3-1. The center lift of $a(X)\star b(X)$ is $-1+X+2X^2$.

The product of the center lifts of $a(X)$ and $b(X)$ is $(2-2X-2X^2)\star(-2+X)=-6+6X+2X^2$. Hence, the center-lift of the product is not equal to the product of the center lifts: $-1+X+2X^2\neq-6+6X+2X^2$. Nonetheless $-1+X+2X^2\equiv-6+6X+2X^2$ mod(5) so they are congruent modulo 5.

Now that we have addition and multiplication of polynomials in R and R_p, we may begin doing arithmetic.

In $\mathbb{Z}[X]$, most polynomials do not have an inverse. In fact, the only two units of $\mathbb{Z}[X]$ are 1 and -1. This is not true in R and R_p, where we can make use of the additional constraint $X^N = 1$. For instance, X and X^{N_1} do not have inverses in $\mathbb{Z}[X]$, but are inverses in R: $X \star X^{N-1} = X^N = 1$. Because integers don't have inverses, few polynomials in R have inverses, but in R_p, because the non-zero coefficients are invertible, the story is different...

Proposition 1.3. *Let p be prime. $a(X) \in R_p$ has a multiplicative inverse if and only if*

$$\gcd(a(X), X^N - 1) = 1 \text{ in } \mathbb{Z}/p\mathbb{Z}[X].$$

The proof requires quite a bit of background about polynomial rings, which is not included in this exposition. The bottom line is that exactly like integers, given polynomials $a(X)$ and $b(X)$ such that $\deg(a(X)) > \deg(b(X))$, there exists $r(X)$ with $\deg(b(X)) > \deg(r(X))$ and $v(X)$ such that $a(X) = v(X) \star b(X) + r(X)$. Further, given polynomials $a(X)$ and $b(X)$ the extended Euclidian algorithm provides $u(X)$ and $v(X)$ such that $a(X) \star u(X) + b(X) \star v(X) = \gcd(a(X), b(X))$.

Proof. Assume $\gcd(a(X), X^N - 1) = 1$ in $\mathbb{Z}/p\mathbb{Z}[X]$.. Then there exist $u(X)$ and $v(X)$ such that $a(X) \star u(X) + (X^N - 1) \star v(X) = 1$. Reducing this equation in R_p yields $a(X) \star u(X) \equiv 1$. Therefore, $u(X)$ is the multiplicative of $a(X)$.

On the other hand, assume $a(X)$ has a multiplicative inverse $u(X)$. Then, $a(X) \star u(X) \equiv 1$ in R_p. This means that there exists $v(X)$ such that $a(X) \star u(X) = 1 + (X^N - 1) \star v(X)$, which implies that $a(X) \star u(X) - (X^N - 1) \star v(X) = 1$. Therefore, $\gcd(a(X), X^N - 1) = 1$ in $\mathbb{Z}/p\mathbb{Z}[X]$. □

One last definition, and we are ready to go.

Definition 1.4. Let d_1, $d_2 \in \mathbb{N}$. We define $\mathcal{T}(d_1, d_2)$ to be the set of $a(X) \in R$ such that $a(X)$ has d_1 coefficients equal to 1, d_2 coefficients equal to -1 and all other coefficients equal to 0.

2 NTRUEncrypt

Addition, multiplication, inverses, special polynomials... all we need to make a cryptosystem in rings of convolution of polynomials.

2.1 The NTRU cryptosystem, encryption and decryption

Public information

- $N > 1$ a fixed prime, $p > 1$, $q > 1$ such that $\gcd(p, q) = \gcd(N, q) = 1$ and d such that $q > (6d + 1)p$. This implies that q is much larger than p, which is important for message decryption.
- $R = \frac{\mathbb{Z}[X]}{(X^N - 1)}$, $R_p = \frac{\mathbb{Z}/p\mathbb{Z}[X]}{(X^N - 1)}$, $R_q = \frac{\mathbb{Z}/q\mathbb{Z}[X]}{(X^N - 1)}$

Alice private and public key

Using the public information, Alice builds her private keys: $f(X) \in \mathcal{T}(d+1, d)$ and $g(X) \in \mathcal{T}(d, d)$. f is invertible in R_p and R_q, and Alice randomly generates f as many times as needed. She then computes $F_p(X) = f(X)^{-1}$ in R_p and $F_q(X) = f(X)^{-1}$ in R_q using the extended Euclidian algorithm.

For her public key, Alice computes $h(X) = F_q(X) \star g(X) \bmod(q)$. All she needs to decrypt the message are $f(X)$ and $F_p(X)$.

Bob's message encryption

Bob's message is a polynomial $m(X)$ with coefficients $-\frac{p}{2} < m_i \leq \frac{p}{2}$. Notice that it is a center lift from R_p to R. :)

Then, Bob chooses a random perturbation $r(X) \in \mathcal{T}(d, d)$ and computes $e(X) = p \cdot h(X) \star r(X) + m(X) \bmod(q)$.

Alice's decryption

To decrypt $e(X)$, Alice computes $a(X) \equiv f(X) \star e(X) \bmod(q)$. She then takes the center lift $a'(X)$ of $a(X)$ to R, and then computes $b(X) \equiv F_p(X) \star a(X) \bmod(p)$. If the parameters were well chosen, $b(X) = m(X)$ and Alice has successfully decrypted the message.

Example 2.1. *For example, choose the (small and unsafe) parameters $p = 3$, $q = 256$, $N = 11$ and $d = 4$.*

Alice can randomly generate $f(X) = -X^10 - X^9 + X^7 + X^6 + X^4 - X^3 + X^2 + X - 1$ with $F_q(X) = 143 * X^10 + 115 * X^9 + 146 * X^8 + 254 * X^7 + 232 * X^6 + 167 * X^5 + 161 * X^4 + 138 * X^3 + 42 * X^2 + 138 * X + 1$ and $F_p(X) = 2 * X^10 + 2 * X^9 + X^6 + X^5 + X^4 + X^3 + 2 * X^2 + X + 2$, and $g(X) = X^10 + X^8 - X^7 + X^4 - X^3 - X^2 + X - 1$. Notice that while f has very small coefficients, its inverse F_q doesn't seem to have restrictions on the size of its coefficients. To create her public key, Alice computes $h(X) = F_q(X) \star g(X) = 221 * X^10 + 207 * X^9 + 104 * X^8 + 76 * X^7 + 160 * X^6 + 215 * X^5 + 73 * X^4 + 47 * X^3 + 140 * X^2 + 43 * X + 250$.

Bob wants to send the message $m(X) = X^10 - X^8 + X^4 + X^3 - 1$. To do so, he generates a random perturbation in $\mathcal{T}(d, d)$: $r(X) = -X^7 - X^6 + X^5 + X^4 + X^3 - X^2 - X + 1$, and computes $e(X) = p \cdot h(X) \star r(X) + m(X) = 212 * X^10 + 96 * X^9 + 59 * X^8 + 28 * X^7 + 173 * X^6 + 44 * X^5 + 12 * X^4 + 63 * X^3 + 173 * X^2 + 32 * X + 133$ and sends $h(X)$ to Alice. The encrypted message doesn't look like anything special.

Now that Alice has $h(X)$, she wants to decrypt it using her private keys f and F_p. She first computes the center-lift $a'(X)$ of $a(X) \equiv f(X) \star e(X) \bmod(q) \equiv 254 * X^10 + 9 * X^9 + 7 * X^8 + 250 * X^7 + 251 * X^6 + 253 * X^5 + 7 * X^4 + 252 * X^3 + 2 * X^2 + X + 251$, and then computes $b(X) \equiv F_p(X) \star a'(X) \bmod(p) \equiv X^10 - X^8 + X^4 + X^3 - 1 = m(X)$.

2.2 Verifying that the encryption and decryption work

Theorem 2.1. *If the parameters satisfy the conditions above, then $b(X) = m(X)$.*

Proof. Alice beings by computing $a(X) \equiv f(X) \star e(X) \equiv f(X) \star p \cdot h(X) \star r(X) + f(X) \star m(X) \bmod(q) \equiv p \cdot f(X) \star F_q(X) \star g(X) \star r(X) + f(X) \star m(X) \bmod(q) \equiv p \cdot g(X) \star r(X) + f(X) \star m(X) \bmod(q)$.

Alice then takes the lift of $a(X) \equiv p \cdot g(X) \star r(X) + f(X) \star m(X)$. To know what "shape" the lift has, we check what the largest coefficient of $a(X)$ is. To do so, we analyze it component by component:

- $\underline{g(X) \star r(X)}$: Since $r(X)$ and $g(X)$ are chosen to be in $\mathcal{T}(d, d)$ the largest possible coefficient is if all the 1 and -1 match up for a total sum of $2d$.
- $\underline{f(X) \star m(X)}$: $f(X) \in \mathcal{T}(d+1, d)$ and $m(X)$ is a lift in R_p so has all coefficients of magnitude less than $\frac{p}{2}$. Hence, the largest possible coefficient is if all the 1 match up with the $\frac{p}{2}$ and all the -1 match up with the $-\frac{p}{2}$ for a total of $(2d+1)\frac{p}{2}$.

If the largest coefficient of $p \cdot g(X) \star r(X)$ and the largest coefficient of $f(X) \star m(X)$ coincide, the largest total coefficient is $p2d + (2d+1)\frac{p}{2} = (3d + \frac{1}{2})p$. Similarly, the smallest possible coefficient is also of magnitude less than $(3d+\frac{1}{2})p$. Since by assumption $q > (6d+1)p$, the largest coefficient of $a(X)$ is less than $\frac{q}{2}$ and the smallest coefficient of $a(X)$ is larger than $\frac{q}{2}$. This implies that the lift of $a(X)$ from R_q is $a(X)$ itself, i.e. that $a'(X) = a(X)$.

From there, Alice computes $b(X) \equiv F_p(X) \star a(x) \bmod(p) \equiv F_p(X) \star p \cdot g(X) \star r(X) + f(X) \star m(X) \bmod(p) \equiv F_p(X) \star f(X) \star m(X) \bmod(p) \equiv m(X) \bmod(p)$ in R_p. □

3 NTRUEncrypt and lattices

While at first glance, all the chosen parameters seem independent, we know that $f(X) \star h(X) \equiv g(X)$ in R_q. We know that $f(X)$ and $g(X)$ have very small coefficients, and $h(X)$ doesn't have any constraints on its coefficients (notice how similar it is to 1TRU). Therefore, breaking $NTRU$ amounts to finding small ternary polynomials $f(X)$ and $g(X)$ such that $f(X) \star h(X) \equiv g(X)$ in R_q given $h(X)$.

We can translate this into a lattice problem, meaning that if the lattice problem can be solved fast, then the cryptosystem would be broken. In practice, attacks on NTRU can be made using LLL lattice basis reduction algorithm, and some are fast if NTRU's parameters aren't well chosen. As such, understanding NTRU from a lattice perspective is essential to guarantee its security. Let

$$M_{NTRU} = \begin{pmatrix} I & H \\ 0 & qI \end{pmatrix}$$

where

$$H = \begin{pmatrix} h_0 & h_1 & ... & h_{N-1} \\ h_{N-1} & h_0 & ... & h_{N-2} \\ \vdots & \vdots & \ddots & \vdots \\ h_1 & h_2 & ... & h_0 \end{pmatrix}$$

We represent pairs of polynomials in the lattice $(a(X), b(X)) = (a_0, ..., a_{N-1}, b_0, ..., b_{N-1})$.

Notice immediately that when performing the multiplication $(a(X), b(X))M_{NTRU}$, $a(X)$ aligns with the identity matrix and $b(X)$ the 0 matrix so that the first N coordinates are

simply the coordinates of $a(X)$. Moreover, $a(X)$ aligns with H and $b(X)$ with qI. The last N coordinates of the result is the coordinates of $a(X)H+qb(X)$. While $qb(X)$ is straightforward, $a(X)H$ is not. The first coordinate is given by $a_0h_0 + a_1h_{n-1} + ... + a_{N-1}h_1 = \sum_{j+k\equiv 0} a_jh_k$, which is exactly the first coordinate of the convolution product of $a(X)$ and $h(X)$. Similarly, the i^{th} coordinate (coefficient in front of X^{i-1} is given by $a_0h_{i-1}+...+a_{N-1}h_i = \sum_{j+k\equiv i-1} a_jh_k$ the i^{th} coordinate of the convolution product of $a(X)$ and $h(X)$.

Therefore, $a(X)H + qb(X) = a(X) \star h(x) + qb(X)$, so that $(a(X), b(X))M_{NTRU} = (a(X), a(X) \star h(X) + qb(X))$.

Proposition 3.1. *Assume $f(X) \star h(X) \equiv g(X) \bmod(q)$. Let $u(x)$ be the polynomial such that $f(X) \star h(X) = g(X) + qu(X)$. Then, $(f(X), -u(X))M_{NTRU} = (f(X), g(X))$ so the vector $(f(X), g(X))$ is in the NTRU lattice.*

Proof. Based on the discussion above, we may perform the succinct computation:

$$(f(X), -u(X)) \begin{pmatrix} I & H \\ 0 & qI \end{pmatrix} = (f(X), f(X) \star h(X) - qu(X) = (f(X), g(X)).$$

Since $(f(X), g(X))$ are a linear combination of lattice vectors, they are in the lattice. □

The last proposition of this exposition accomplishes the translation of breaking NTRU into a shortest lattice vector problem:

Proposition 3.2. *Let (N, p, q, d) be NTRUEncrypt parameters that satisfy the necessary conditions. Let $\mathcal{L}_h$ be the NTRU lattice associated to Alice's private key pair $(f(X), g(X))$. Then:*

1. *det($\mathcal{L}_h$) $= q^N$*
2. *$\|(f(X), g(X))\| \approx \sqrt{4d}$*
3. *According to the Gaussian heuristic, the shortest vector in the lattice has length $\sqrt{\frac{Nq}{\pi e}}$.*

Proof.

1. $\det(\mathcal{L}_h) = q^N$ is simply $\det(M_{NTRU})$. Since $M_{NTRU})$ is an upper triangular matrix, its determinant is the product of its diagonal elements, i.e. q^N.
2. By definition, $f(X)$ has $d + 1$ coordinates equal to 1 and d corrdinates equal to -1, and $g(X)$ has d coordinates equal to 1 and d coordinates equal to -1. The norm of the resulting vector is the square root of $4d + 1$ copies of 1^2, so $\sqrt{4d + 1} \approx \sqrt{4d}$.
3. The Gaussian Heuristic yields that the shortest lattice vector has length approximately :

$$\sqrt{\frac{2N}{2\pi e}}\det(\mathcal{L}_h)^{1/2N} = \sqrt{\frac{N}{\pi e}}(q^N)^{1/2N} = \sqrt{\frac{Nq}{\pi e}}$$

□

This proposition doesn't in and of itself say anything about $(f(X), g(X))$ being a shortest vector. However, carefully choosing parameters such that $p = 3$ $d = N/p$, $q \approx 6pd \approx 2pN$ yields that

$$\frac{\|(f(X), g(X))\|}{\text{GH for shortest vector}} \approx \frac{\sqrt{4d}}{\sqrt{\frac{Nq}{\pi e}}} \approx \sqrt{4\pi e/3}\sqrt{\frac{N}{Nq}} \approx \sqrt{4\pi e/9}\sqrt{\frac{1}{2N}} \approx 1.38\frac{1}{\sqrt{N}}$$

. For large enough choices of N, the pair $(f(X), g(X))$ is a short vector in the lattice.

This choice of parameters can seem forced, but in practice, safe choices look like $(N = 509, p = 3, q = 2048)$ or $(N = 821, p = 3, q = 4096)$.

Conclusion

NTRU encryption represents a significant advancement in cryptographic algorithms, offering robust security, resilience to quantum threats, and practicality for implementations. At the core of NTRU's strength lies its use of hard mathematical problems such as the Shortest Vector Problem in lattices and the factorization problem in convolution rings.

One of NTRU's key advantages is its resilience quantum attacks (for now), an important consideration for the near future. While traditional public-key cryptosystems are vulnerable to quantum attacks, NTRU's security has yet to be provably broken. Indeed, RSA relies on the difficulty of factoring large integers, which Shor's algorithm (a quantum algorithm) does in polynomial time. NTRU relies on the difficulty of the SVP or factorization in convolution rings, for which no known quantum algorithms exist. However, this does not imply that such an algorithm can't exist.

Moreover, NTRU is practical in its implementation, both in terms of computational efficiency and parameter sizes. Unlike traditional cryptosystems that require large key sizes and extensive computational resources, NTRU achieves strong security with manageable overheads. For reference, for a 256 bit security margin, the parameters $(N = 821, p = 3, q = 4096)$ suffice for NTRU. For RSA encryption, the parameters p and q must be at least on the order of 2^{1024}.

References

- J. Hoffstein, J. Pipher, J. H. Silverman, *An Introduction to Mathematical Cryptography*, chapter 7, Springer Undergraduate Texts in Mathematics, 2014
- J. Hoffstein, J. Pipher, J. H. Silverman, "NTRU: a New High Speed Public Key Cryptosystem", Brown University, 1996
- J. N. Gaithuru, M. Salleh, M. Bakhtiari, "Identification of Influential Parameters for NTRU Decryption Failure and Recommendation of Extended Parameter Selection Criteria for Elimination of Decryption Failure", IAENG International Journal of Computer Science vol.44, 2017