Trabajo de Fin de Máster

Máster Universitario en Ciberseguridad e Inteligencia de Datos

Ataques y aplicaciones a CRYSTALS-Dilithium

Attacks and applications on CRYSTALS-Dilithium

Édgar Pérez Ramos

La Laguna, 20 de mayo de 2024

Dña. **Pino Teresa Caballero Gil**, con N.I.F. 45534310Z, Catedrática de Universidad adscrita al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutora.

C E R T I F I C A

Que la presente memoria titulada:

"*Ataques y aplicaciones a CRYSTALS-Dilithium*"

ha sido realizada bajo su dirección por D. **Édgar Pérez Ramos**, con N.I.F. 42199515N.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 20 de mayo de 2024

Agradecimientos

En primer lugar, me gustaría agradecer a Pino por su ayuda a lo largo de este curso, por ser siempre referente de trabajo y tesón, por haber confiado en mí y por todas las oportunidades que me ha brindado.

En segundo lugar, me gustaría agradecer a Atlantis Technology, a la Cátedra de Ciberseguridad Binter Universidad de La Laguna y al grupo de investigación CryptULL por haberme permitido disfrutar y aprovechar este año de investigación, del cual he podido extraer grandes resultados y experiencias inolvidables.

Por último, agradecer a mi familia y a Amanda por el apoyo incondicional.

“…Amon Dín, la llama de la esperanza.”

Licencia

Resumen

CRYSTALS-Dilithium es uno de los tres esquemas de firma digital finalistas del proceso de selección del National Institute of Standards and Technology. Se trata de un algoritmo basado en retículos, que sigue las pautas del conocido esquema de Fiat-Shamir. Concretamente, su seguridad se basa en la dificultad del problema de encontrar vectores más cortos en retículos. Este trabajo se presenta a modo de compendio de una serie de artículos publicados por el autor relacionados con CRYSTALS-Dilithium. En primer lugar, se realiza una introducción a los fundamentos matemáticos del esquema. En segundo lugar, se presentan dos publicaciones dedicadas al primer ataque teórico cleptográfico por backdoor a CRYSTALS-Dilithium. En tercer lugar, se presenta una publicación sobre el estudio y la exploración de la compleja función Transformada Teórica de Números (NTT, Number Theoretic Transform), utilizada en las multiplicaciones de polinomios del estándar de firma. Finalmente, se incluye una serie de publicaciones didácticas dirigidas al nivel de Secundaria, con el propósito de mejorar la comprensión de los conceptos matemáticos involucrados en la criptografía post-cuántica basada en retículos.

Palabras clave: Criptografía post-cuántica, Retículos, CRYSTALS-Dilithium, Cleptografía, Ataque backdoor, NTT, GeoGebra

Abstract

CRYSTALS-Dilithium is one of the three digital signature schemes included in the third round of post-quantum standard selection by the National Institute of Standards and Technology. It is a latticebased algorithm that adheres to the well-known Fiat-Shamir scheme guidelines. Specifically, its security is based on the difficulty of the problem of finding shorter vectors in lattices. This work is presented as a compendium of a series of papers published by the author related to CRYSTALS-Dilithium. First, an introduction to the mathematical foundations of the scheme is given. Secondly, two publications dedicated to the first theoretical kleptographic backdoor attack on CRYSTALS-Dilithium are presented. Thirdly, a publication on the study and exploration of the complex Number Theoretic Transform (NTT) function used in the polynomial multiplications of the signature standard is presented. Finally, it includes a series of pedagogical publications aimed at secondary school level, with the purpose of improving the understanding of the mathematical concepts involved in lattice-based post-quantum cryptography.

Keywords: Post-quantum cryptography, Lattices, CRYSTALS-Dilithium, Kleptography, Backdoor, NTT, GeoGebra

Índice general

Índice de Figuras

Capítulo 1

Introducción

Desde tiempos antiguos, la criptografía ha sido una herramienta clave en la protección de la información. Mensajes cifrados han sido hallados en inscripciones egipcias que datan de hace más de 4.000 años, mientras que los griegos antiguos empleaban técnicas de transposición y sustitución de letras para resguardar sus comunicaciones. Durante la Edad Media, los monjes copistas desarrollaron métodos para ocultar información en manuscritos, y los soldados utilizaban claves secretas para enviar mensajes en pleno campo de batalla.

En el siglo XX, la criptografía experimentó avances significativos que marcaron un antes y un después en su evolución. Durante la Segunda Guerra Mundial, la criptografía desempeñó un papel crucial en los conflictos bélicos, con el desarrollo y la rotura de códigos desempeñando un papel determinante en el resultado de las batallas. Alan Turing, pionero en el campo de la computación, lideró el equipo que logró descifrar el código Enigma utilizado por las fuerzas alemanas, contribuyendo así de manera decisiva a la victoria de los Aliados.

Tras la guerra, el enfoque en la criptografía se desplazó hacia el desarrollo de sistemas de clave simétrica, donde tanto el cifrado como el descifrado utilizan la misma clave. Sin embargo, la distribución segura de estas claves se convirtió en un desafío significativo. Es en este contexto que surgió el revolucionario concepto de criptografía de clave asimétrica, introducido por Whitfield Diffie y Martin Hellman en los años setenta. Este enfoque permitió la creación de un par de claves: una pública y otra privada. La clave pública se utiliza para cifrar los mensajes, mientras que la clave privada se utiliza para descifrarlos, proporcionando una solución elegante al problema de la distribución segura de claves en la era digital. Esta innovación sentó las bases para la seguridad en internet y la protección de la privacidad en las comunicaciones electrónicas.

En la actualidad, la criptografía desempeña un papel fundamental en la seguridad de la información en todos los ámbitos de la sociedad moderna, como la seguridad informática, la banca, la medicina, las comunicaciones militares, entre otros. La criptografía también se utiliza para proteger la privacidad de las comunicaciones en línea y para garantizar la seguridad de las transacciones financieras en internet.

Sin embargo, con el advenimiento de la computación cuántica, se plantea un desafío sin precedentes para la seguridad de los sistemas criptográficos actuales. La promesa de la computación cuántica de resolver problemas computacionales de manera exponen-

cialmente más rápida que los computadores clásicos podría comprometer seriamente la seguridad de la infraestructura de comunicaciones y la privacidad de los datos sensibles.

En respuesta a este desafío, ha surgido un campo emergente dentro de la criptografía conocido como criptografía post-cuántica. Este enfoque busca desarrollar algoritmos y protocolos de seguridad que sean inmunes a los ataques de los computadores cuánticos, garantizando así la seguridad y la privacidad de la información en la era post-cuántica.

Por esta razón, en el año 2016 arrancó la carrera de la criptografía post-cuántica. Ese año el *National Institute of Standards and Technology* (NIST) inició un proceso de selección de esquemas criptográficos que pudieran ser resistentes a los ordenadores cuánticos. A finales de 2017 se publicaron los algoritmos que habían pasado la primera fase de la convocatoria, un total de 69. Más tarde, en 2019 tras aplicar un cribado exhaustivo de los 69 iniciales quedaron 26. Durante esa ronda se llevaron a cabo pruebas rigurosas para evaluar la seguridad y el rendimiento de cada uno de los algoritmos, así como para identificar posibles vulnerabilidades y mejorar la seguridad de los algoritmos. En la tercera ronda de evaluación en 2018, se seleccionaron 15 candidatos finales para continuar en el proceso de estandarización. Durante esta ronda, se llevaron a cabo pruebas adicionales y se trabajó con la comunidad criptográfica para identificar posibles vulnerabilidades y mejorar la seguridad de los algoritmos preseleccionados.
Finalmente, según (1) en julio de 2022, el NIST anunció los algoritmos seleccionados como estándares finales (2), diferenciando entre:

- FIPS 203: CRYSTALS-Kyber. Es un KEM basado en el problema de aprendizaje con errores (*Learning With Errors*, LWE) sobre anillos (3).
- FIPS 204: CRYSTALS-Dilithium.
- FIPS 205: SPHINCS+.

El NIST invitó a la comunidad criptográfica mundial a proporcionar comentarios sobre estos borradores hasta noviembre de 2023, con el fin de recibir retroalimentación y asegurarse de que los estándares sean completos y no tengan omisiones antes de su finalización.

En este trabajo se presenta un estudio en profundidad de las bases matemáticas de los nuevos estándares CRYSTALS-Kyber, (3) y CRYSTALS-Dilithium (4), ambos basados en la teoría de retículos (5), (6).

Se espera que tanto Kyber como Dilithium se utilicen ampliamente en la industria y el gobierno para proteger privacidad de las comunicaciones y la autenticidad de los documentos digitales, los mensajes en sistemas informáticos y de las comunicaciones durante las próximas décadas.

Este trabajo se estructura de la siguiente forma. En el capítulo 2 se analizan en profundidad los conceptos matemáticos asociados a CRYSTALS-Dilithium y CRYSTALS-Kyber, como son los retículos, los problemas del vector más corto, más cercano y aprendizaje sobre errores, además de describir brevemente la Transformada Teórica de Números y el esquema de Dilithium. En el capítulo 3 se desarrollan las contribuciones asociadas a los artículos o aportaciones a congresos publicadas. Finalmente, en los anexos A, B y C se adjuntan los respectivos artículos.

Capítulo 2

Conceptos básicos y antecedentes

Para el desarrollo de este capítulo, donde se deben definir y asentar las bases matemáticas de CRYSTALS, nos hemos basado en (7), además de mejorar y reforzar ciertos conceptos. Por lo tanto, antes de comenzar a estudiar en profundidad los retículos se deben controlar algunos conceptos algebraicos previos que van a ayudar a conocer y entender mejor los cimientos de este objeto matemático y sus propiedades.

2.1. Preliminares

2.1.1. Grupos

Definición 2.1.1.1 *Sea G un conjunto no vacío y $\#$ una operación binaria definida en G. Se dice que el par $(G, \#)$ es un grupo si se cumplen las siguientes propiedades:*

1. *Propiedad asociativa.* $\forall a, b, c \in G \ : \ (a \# b) \# c = a \# (b \# c)$
2. *Existencia del elemento neutro.* $\exists\, e \in G \ : \ e \# u = u = u \# e, \qquad \forall u \in G$
3. *Existencia del elemento opuesto.* $\forall\, a \in G, \exists\, b \in G \ : \ a \# b = e = b \# a$

Si se da la propiedad conmutativa respecto a la operación # se habla de un *grupo abeliano.* Además, en general se suele denotar las operaciones de la siguiente manera: $(G, +)$ en notación aditiva y $(G, \cdot)$ en notación multiplicativa.

Definición 2.1.1.2 *Sea $(G, \cdot)$ es un grupo y $H \subset G$, se dice que $(H, +)$ es un subgrupo de $(G, +)$ si también tiene estructura de grupo con la misma operación.*
En caso de que ocurra se denota:

$$(H, +) \leq (G, +) \qquad o \qquad H \leq G$$

A continuación se introduce el teorema de caracterización de subgrupos, el cuál es fundamental y se necesita en definiciones siguientes.

Teorema 2.1.1.1 *Si $(G, \cdot)$ es un grupo y $H \subset G$ entonces son equivalentes las siguientes afirmaciones:*

- $(H, +) \leq (G, +)$
- • $\forall a, b \in H \Rightarrow a \cdot b \in H$

- $1_G \in H$
- $\forall a \in H \Rightarrow a^{-1} \in H$

- $H \neq \emptyset$
- $\forall a, b \in H \Rightarrow a \cdot b^{-1} \in H$

2.1.2. Anillos

En esta sección se recordará la definición de anillo y de cuerpo. El objetivo que se persigue es adquirir las herramientas necesarias para abordar los espacios vectoriales y, más adelante, los retículos.

Definición 2.1.2.1 *Sea A un conjunto no vacío dotado de dos operaciones binarias. Se dice que la terna $(A, +, \cdot)$ es un anillo si se cumplen las siguientes condiciones:*

- *Respecto a la operación $+$:*
  1. *Propiedad asociativa.* $\forall a, b, c \in A \ : \ (a + b) + c = a + (b + c)$
  2. *Propiedad conmutativa.* $\forall a, b \in A \ : \ a + b = b + a$
  3. *Existencia del elemento neutro.* $\exists\, e \in A \ : \ e + u = u = u + e, \qquad \forall u \in A$
     Se denota al elemento neutro por 0_A
  4. *Existencia del simétrico.* $\forall\, a \in A, \exists\, b \in A \ : \ a + b = e = b + a$
     Al elemento simétrico de $a \in A$ *se le denota como* $-a$ *y se le conoce por el opuesto de a.*

- *Respecto a la operación $\cdot$:*
  1. *Propiedad asociativa.* $\forall a, b, c \in A \quad : \quad (a \cdot b) \cdot c = a \cdot (b \cdot c)$
  2. *Propiedad distributiva.* $\forall a, b, c \in A \quad : \quad (a + b) \cdot c = a \cdot c + b \cdot c$
  3. *En caso de existir,* $\exists\, e \in A : e \cdot a = a \cdot e = a$ *El elemento neutro del producto se denota por* 1_A.

En el caso de que se cumpla la propiedad conmutativa y exista el elemento neutro de la operación $\cdot$ se habla de un anillo conmutativo y unitario.

Para explicar qué es un cuerpo, se requiere tener en cuenta la noción de unidad. Es importante recordar cómo se define:

Definición 2.1.2.2 *Sea $(A, +, \cdot)$ un anillo unitario. Se dice que $a \in A \setminus 0_A$ es una unidad de A si $\exists\, b \in A$:*

$$a \cdot b = 1_A = b \cdot a$$

Al elemento $b \in A$ se le llama inverso de $a \in A$ y se escribe: $b = a^{-1}$. Además, el conjunto de las unidades de A se denota de la siguiente manera:

$$A^* = \{a \in A \setminus \{0_A\} \ : \ a \cdot b = 1_A = b \cdot a, \ \exists\, b \in A\}$$

Definición 2.1.2.3 *Sea $(A, +, \cdot)$ un anillo conmutativo y unitario. Se dice que A es un cuerpo si $A^* = A \setminus \{0_A\}$.*

Para abordar los conjuntos cocientes y los problemas relacionados con los retículos, resulta imprescindible contar con la definición de los ideales.

Definición 2.1.2.4 *Sea A un anillo conmutativo e I un subconjunto de A no vacío. Se dice que I es un ideal si:*

1. $0_A \in I$
2. $\forall a \in A, \forall b \in I,\ \ a \cdot b \in I$
3. $\forall a, b \in I,\ \ a - b \in I$

Definición 2.1.2.5 *Sean A un anillo conmutativo y unitario e I un ideal de A. En A se define la siguiente relación binaria:*

$$\forall a, b \in A\ :\ a \sim_I b \iff a - b \in I$$

La relación $\sim_I$ es una relación de equivalencia. Se denota por A/I al conjunto cociente asociado a $\sim_I$, es decir:

$$A/I = \{a + I : a \in A\}$$

Además A/I tiene estructura de anillo conmutativo y unitario con las siguientes operaciones:

- *Adición*

$$\begin{aligned} +\colon A/I \times A/I &\longrightarrow A/I \\ (a+I, b+I) &\longmapsto (a+b)+I, \end{aligned}$$

- *Producto*

$$\begin{aligned} \cdot\colon A/I \times A/I &\longrightarrow A/I \\ (a+I, b+I) &\longmapsto (a\cdot b)+I, \end{aligned}$$

Ejemplo 1 *$A = \mathbb{Z}$ y $I = (3)$, es decir, I es el conjunto de los múltiplos de 3.*
Por tanto: $A/I = \{0 + I, 1 + I, 2 + I\}$.
Este concepto adquiere importancia cuando se trabaja con ideales y anillos de polinomios del estilo $\mathbb{Z}_p$, con p un número primo.

2.1.3. Espacios vectoriales.

Definición 2.1.3.1 *Sea V un espacio vectorial sobre un cuerpo K, con V distinto de vacío y dotado con dos operaciones:*

- *Adición*

$$\begin{aligned} +\colon V \times V &\longrightarrow V \\ (u, v) &\longmapsto u + v, \end{aligned}$$

es una operación interna que cumple las siguientes condiciones:

1. *Conmutativa.* $\forall u, v \in V,\ \ u + v = v + u$

2. *Asociativa.* $\forall u, v, w \in V,\ (u + v) + w = u + (v + w)$
3. *Existencia del elemento neutro.* $\exists\, e \in V\ :\ e + u = u + e = u,\ \forall u \in V$
4. *Existencia del opuesto.* $\forall u \in V, \exists\, v \in V\ :\ u + v = e = v + u$

- *Producto*

$$\begin{aligned} \cdot : K \times V &\longrightarrow V \\ (a, v) &\longmapsto a \cdot v, \end{aligned}$$

es una operación externa tal que:

1. *Asociativa.* $\forall a, b \in K, \forall u \in V\ :\ a \cdot (b \cdot u) = (a \cdot b) \cdot u$
2. *Existencia del elemento neutro.* $\exists\, e \in K, \forall u \in V\ :\ e \cdot u = u$
3. *Distributiva.* $\forall a \in K, \forall u, v \in V\ :\ a \cdot (v + u) = a \cdot v + a \cdot u$

A la terna $(V, +, \cdot)$ se le conoce como espacio vectorial sobre el cuerpo K.

Definición 2.1.3.2 *Sea V un espacio vectorial sobre un cuerpo K y un subconjunto $B \subset V$ es una base si:*

1. *B es linealmente independiente*
2. *B es un sistema de generadores de V*

Las bases revelan la estructura de los espacios vectoriales de una manera concisa. Una base es el menor conjunto (finito o infinito) que genera V, con $B = \{v_i\}_{i \in I}$ y $B \subseteq V$. Esto significa que cualquier vector v puede ser expresado como una combinación lineal de elementos de la base.

$$\forall u \in V, \quad u = \sum_{i=1}^{n} a_i \cdot v_i, \quad a_i \in K$$

Es posible abordar la dimensión del espacio vectorial utilizando la referencia bibliográfica proporcionada por (8).

- Si B es una base finita, es decir tiene n vectores, entonces $dim(V) = n$
- Si B es una base infinita, entonces $dim(V) = \infty$

2.1.4. Espacios normados

Para la siguiente defínción y ejemplo se ha utilizado como referencia (9).

Definición 2.1.4.1 *Un espacio vectorial $\mathbb{V}$ se denomina espacio normado, si para cada $x \in \mathbb{V}$, se define un número real, que se denota por $||x||$ y que satisface las siguientes propiedades:*

- $\| x \| \geq 0,\ \forall x \in \mathbb{V}$ *(Positividad)*
- $\| x \| = 0 \iff x = 0,\ \forall x \in \mathbb{V}$ *(Definido)*

- $\| \alpha \cdot x \| = |\alpha| \cdot \| x \| \quad \forall \alpha \in \mathbb{R}$ *y* $\forall x \in \mathbb{V}$ *(Homogeneidad)*
- $\| x + y \| \leq \| x \| + \| y \|, \quad \forall x, y \in \mathbb{V}$ *(Desigualdad triangular)*

La cantidad $\| x \|$ es conocida como la norma de x. Generalmente, se denota como $(\mathbb{V}, \cdot)$ al espacio vectorial normado. En el caso de que no se verifique la segunda condición de la Def. 2.1.4.1 pero sí las restantes, se dice que $\| \cdot \|$ es una seminorma.

Ejemplo 2 *Supongamos que $x \in \mathbb{R}^n$, con $n \in \mathbb{N}$, entonces se definen las normas siguientes:*

$$||x||_p = (\sum_{i=1}^{n} x_i^p)^{\frac{1}{p}}, \qquad y \qquad ||x||_\infty = \sup_{i \in \{1,n\}} x_i \tag{2.1}$$

También se pueden considerar los siguientes ejemplos:

- *Sea $p(x) \in \mathbb{P}_n$ un polinomio de grado n, con coeficientes a_i, entonces la norma infinito se define del siguiente modo:*

$$||p||_\infty = max\{|a_0|, |a_1|, \ldots, |a_n|\} \tag{2.2}$$

- *Sea $A \in \mathcal{M}(\mathbb{R})_{m \times n}$, se define la norma infinito de una matriz como:*

$$||A||_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^{n} |a_{ij}| \tag{2.3}$$

Se ha realizado un breve repaso de las herramientas necesarias para estudiar los retículos. Ahora, los retículos son objetos geométricos que pueden describirse intuitivamente como los puntos de intersección de una malla, similar a la red infinita de una portería de fútbol. Esta malla no necesariamente es ortogonal y puede tener un número arbitrario de dimensiones, (ver Fig. 2.1).

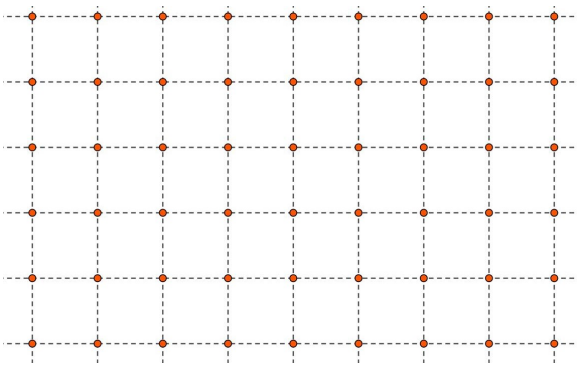

Figura 2.1: Retículo ortonormal sobre el plano euclídeo

2.2. Retículos

2.2.1. Primeras definiciones

Definición 2.2.1.1 *Sea V un espacio vectorial sobre K, con K cuerpo y $B = \{v_1, v_2, \ldots, v_n\}$ una base de un subespacio vectorial de V, y A un anillo contenido en K. Entonces el*

retículo $\mathcal{L} \subset V$ generado por $\{v_1, v_2, \cdots, v_n\}$ es el conjunto:

$$\mathcal{L}(v_1, v_2, \ldots, v_n) = \{ \sum_{i=1}^{n} a_i \cdot v_i : a_i \in A \}$$

En el resto del trabajo se considera $V = \mathbb{R}^m$ y $\mathbb{Z}$ el anillo A, es decir:

$$\mathcal{L}(v_1, v_2, \ldots, v_n) = \{ \sum_{i=1}^{n} a_i \cdot v_i : a_i \in \mathbb{Z} \}$$

Por tanto, un retículo siempre se puede generar a partir de una base del espacio vectorial en el que se defina, mediante todas las combinaciones lineales de elementos de esa base. De hecho, diferentes conjuntos de vectores pueden generar el mismo retículo o, en otras palabras, un mismo retículo puede ser definido a partir de varias bases diferentes. Nótese que en el rango del retículo es n, pues hay n vectores linealmente independientes y su dimensión es m, ya que $V = \mathbb{R}^m$.

Se denota $B = \{v_1, v_2, \ldots, v_n\}$ a la base del retículo que puede ser representado de la forma siguiente: $B = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^{m \times n}$, que nos da lugar a una representación equivalente:

$$\mathcal{L}(B) = \{ Bx : x \in \mathbb{Z}^n \}$$

Se puede enunciar el siguiente el resultado.

Proposición 2.2.1.1 *Dado $B \in \mathbb{R}^{m \times n}$, con columnas linealmente independientes. $\mathcal{L} \subset \mathbb{R}^n$ es de rango máximo si y solo si $< B > = \{ Bx : x \in \mathbb{R}^n \} = \mathbb{R}^n$. Es decir,*

$$n = m \iff < B > = \{ Bx : x \in \mathbb{R}^n \}$$

Nota 2.2.1 *De esta proposición se puede deducir que el rango del retículo se caracteriza como la dimensión del espacio que genera su base. O de otra manera,*

$$rango(\mathcal{L}(B)) = dim(< B >)$$

Definición 2.2.1.2 *Sean $\mathcal{L}' = \mathcal{L}'(B')$ y $\mathcal{L} = \mathcal{L}(B)$ retículos generados por las bases B' y B respectivamente. Si $B' \subset B$ entonces se dice que $\mathcal{L}'$ es un subretículo de $\mathcal{L}$.*

Ejemplo 3 *Sea el retículo $\mathcal{L} = \mathcal{L}((1,0,0),(0,1,0),(0,0,1))$. Entonces un subretículo de $\mathcal{L}$ puede ser $\mathcal{L}' = \mathcal{L}((1,0,0),(0,1,0))$, ya que:*

$$B' = \{(1,0,0),(0,1,0)\} \subsetneq B = \{(1,0,0),(0,1,0),(0,0,1)\}$$

Ejemplo de un no subretículo:

$$\mathcal{L}'' = ((\frac{1}{2},0),(0,\frac{1}{2})) \nsubseteq \mathcal{L} = ((1,0),(0,1))$$

Puesto que:

$$\{(\frac{1}{2},0),(0,\frac{1}{2})\} \nsubseteq \{(1,0),(0,1)\},$$

ya que no se puede expresar $(\frac{1}{2},0)$ como combinación lineal de de la base B.

Para construir un subretículo basta con construirlo a partir de una base en la que se hayan suprimido vectores de la base original.

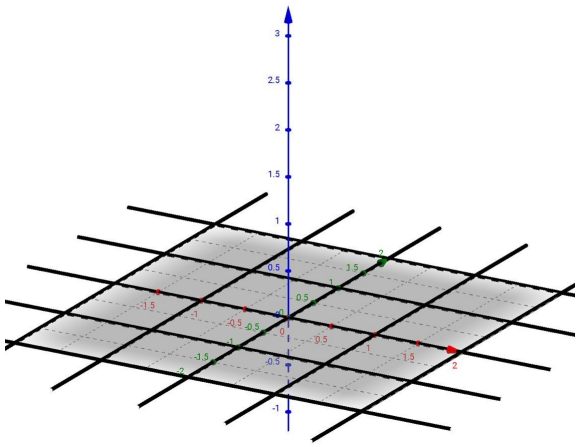

Figura 2.2: Subretículo de dimensión 2 en el espacio

2.2.2. Bases y caracterización algebraica

En general existe más de una base que genera el espacio, es decir, B y B' generan el mismo retículo. Se denota al retículo por $\Lambda = \mathcal{L}(B)$ sin hacer referencia a ninguna base en particular.

Supongamos que $B = \{v_1, v_2, \ldots, v_n\}$ es una base de nuestro retículo $\mathcal{L}(B)$ y sea $B' = \{w_1, w_2, \ldots, w_n\}$ otro conjunto de vectores de V. Entonces se puede escribir como combinación lineal cada $w_j \in B'$ con los elementos de V. Es decir:

$$\begin{cases} w_1 = \alpha_{1,1}v_1 + \alpha_{1,2}v_2 + \ldots + \alpha_{1,n}v_n \\ w_2 = \alpha_{2,1}v_1 + \alpha_{2,2}v_2 + \ldots + \alpha_{2,n}v_n \\ \vdots \\ w_n = \alpha_{n,1}v_1 + \alpha_{n,2}v_2 + \ldots + \alpha_{n,n}v_n \end{cases} \tag{2.4}$$

Sabiendo que $\alpha_{i,j} \in \mathbb{Z}$, $\forall i, j \in \{1, 2, \ldots, n\}$. De esta forma, la matriz que define ese sistema de ecuaciones, llamada matriz de cambio de base, es:

$$A = \begin{pmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,n} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n,1} & \alpha_{n,2} & \cdots & \alpha_{n,n} \end{pmatrix} \tag{2.5}$$

Proposición 2.2.2.1 *Dada una matriz A con coeficientes enteros y A^{-1} su matriz inversa. Entonces A^{-1} es de coeficientes enteros si y solo si $det(A) = \pm 1$.*

Se procede de izquierda a derecha:

Hipótesis: $A, A^{-1} \in \mathcal{M}_{n\times n}(\mathbb{Z})$.
Tesis: $det(A) = \pm 1$.

$$1 = det(I) = det(A \cdot A^{-1}) = det(A)det(A^{-1})$$

Pero $det(A), det(A^{-1}) \in \mathbb{Z}$, pues tienen entradas enteras.
Además se sabe que las únicas unidades en $\mathbb{Z}$ son $\pm 1 \Rightarrow det(A) = \pm 1$.

Ahora se avanza en dirección opuesta.
Hipótesis: $det(A) = \pm 1$ y $A \in \mathcal{M}_{n\times n}(\mathbb{Z})$.
Tesis: $A^{-1} \in \mathcal{M}_{n\times n}(\mathbb{Z})$.
En primer lugar, se sabe que:

$$A^{-1} = \frac{1}{det(A)} \cdot (A^*)^t \tag{2.6}$$

Por hipótesis $det(A) = \pm 1 \Rightarrow A^{-1} = \pm(A^*)^t$.
Por definición, si $A \in \mathcal{M}_{n\times n}(\mathbb{Z}) \Rightarrow A^* \in \mathcal{M}_{n\times n}(\mathbb{Z})$. Entonces:

$$A^{-1} = \pm(A^*)^t \in \mathcal{M}_{n\times n}(\mathbb{Z})$$

■

Definición 2.2.2.1 *Se define como matriz unimodular a la matriz $U \in \mathcal{M}_{n\times n}(\mathbb{Z})$ si $det(U) = \pm 1$.*

Proposición 2.2.2.2 *Dadas dos bases B, B' de un retículo Λ, existe $U \in \mathcal{M}_{n\times n}(\mathbb{Z})$ unimodular tal que $B' = BU$.*

Sean $B' = [w_1, w_2, \ldots, w_n]$ y $B = [v_1, v_2, \ldots, v_n]$.
Entonces es fácil ver de Eq. 3.2 que:

$$B' = B \cdot A^t = [v_1, v_2, \ldots, v_n] \cdot \begin{pmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,n} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n,1} & \alpha_{n,2} & \cdots & \alpha_{n,n} \end{pmatrix}^t$$

Como $det(A) = \pm 1 \Rightarrow det(A^t) = \pm 1$, luego A^t es unimodular. Por tanto,

$$B' = BU$$

■

Nota 2.2.2.1 *Así, para descubrir si dos bases generan el mismo retículo solo hace falta descubrir la correspondiente matriz unimodular U. De hecho, es suficiente comprobar que $B^{-1}B'$ es unimodular.*

Ejemplo 4 *Considérese las bases:*

$$B' = \{(1,0,0), (0,1,0), (0,0,1)\} \text{ y } B = \{(1,-1,1), (0,1,1), (0,0,-1)\}$$

Entonces la matriz de cambio de base es:

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 1 & -1 \end{pmatrix}$$

Dado que esta matriz es triangular inferior, el cálculo de su determinante es instantáneo, $det(A) = -1$. *Por tanto,* A *es unimodular, y según la Prop. 2.2.2.1, se sabe que* A^{-1} *es también una matriz de coeficientes enteros:*

$$A^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & 1 & -1 \end{pmatrix}$$

2.2.3. Definición alternativa

Existe una definición alternativa para los retículos, pero antes es necesario ver dos definiciones previas.

Definición 2.2.3.1 *Sea* $S \subset \mathbb{R}^n$*, se dice que* S *es un subconjunto discreto si:*

$$\exists\, \epsilon > 0 \ : \ \forall x \in S,\ S \cap \{\, w \in \mathbb{R}^n : \| \, x - w \, \| < \epsilon \,\} \ = \ \{x\}$$

Para el caso de un retículo:

$$\exists\, \epsilon > 0,\ \ \forall v \in \Lambda \ \ : \ \ S \cap B(v, \epsilon) \ \ = \ \ \{v\}$$

Para el caso de un retículo $\mathcal{L}$, la definición anterior implica que es posible tomar una bola de radio ϵ centrado en un punto del retículo de forma que dentro de esa bola únicamente se encuentre ese punto del retículo (ver Fig. 2.3):

$$\exists\, \epsilon > 0,\ \forall v \in \mathcal{L} \ : \ S \cap B(x, \epsilon) \ = \ \{x\} \tag{2.7}$$

Definición 2.2.3.2 *Un subconjunto* $\mathcal{L} \subset \mathbb{R}^m$ *es un retículo si y solo si es u subgrupo aditivo discreto.*

Ejemplo 5 *A continuación se dan ejemplos de retículos.*

1. *El conjunto* $\{0\}$.
2. $\mathbb{Z}$ *es un subgrupo aditivo de* $\mathbb{R}$*, por lo que es un retículo. De igual manera, se sabe que el producto cartesiano de un número finito de grupos es también grupo, y pasa lo mismo con los conjuntos discretos. Por tanto,* $\mathbb{Z}^m$ *también es un retículo.*
3. *El conjunto de los números pares,* $2\mathbb{Z}$ *es un grupo aditivo, y además discreto, pues bastaría con tomar una bola de radio* $r < 2$*. Nuevamente esta caracterización se puede generalizar a* $k\mathbb{Z}$*, con* $k \in \mathbb{Z}$.

2.2.4. Caracterización geométrica y determinante

Definición 2.2.4.1 *Dados* $v_1, v_2, \ldots, v_n$ *vectores linealmente independientes, se define el paralelepípedo fundamental como:*

$$\mathcal{P}(B) \ = \ \mathcal{P}(v_1, v_2, \ldots, v_n) \ = \ \{ \sum_{i=1}^{n} x_i v_i \ : \ x_i \in [\frac{-1}{2}, \frac{1}{2}) \,\}$$

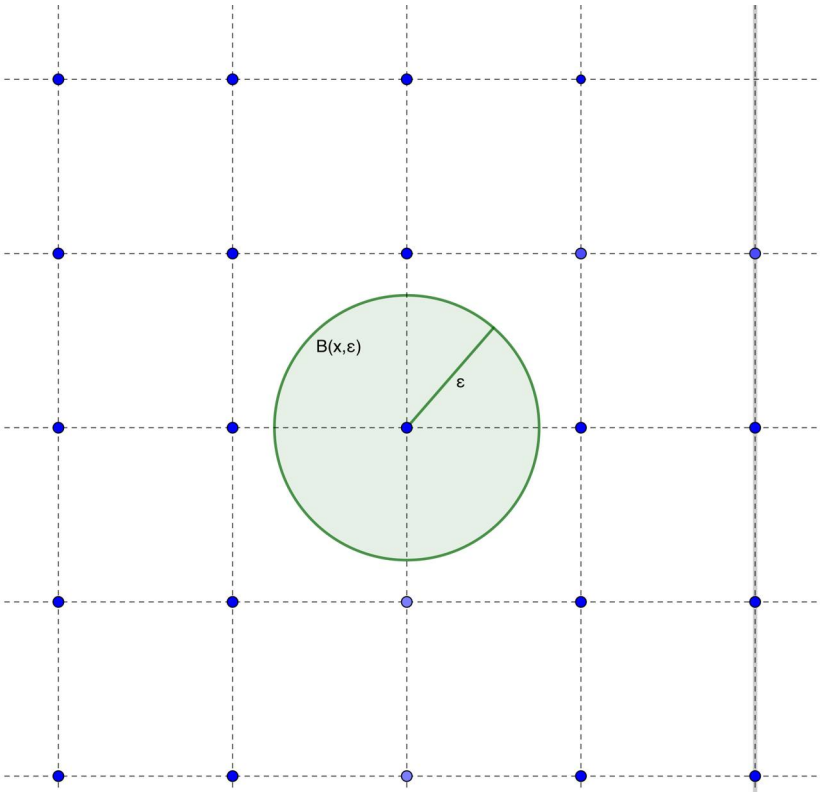

Figura 2.3: Ilustración de un conjunto discreto

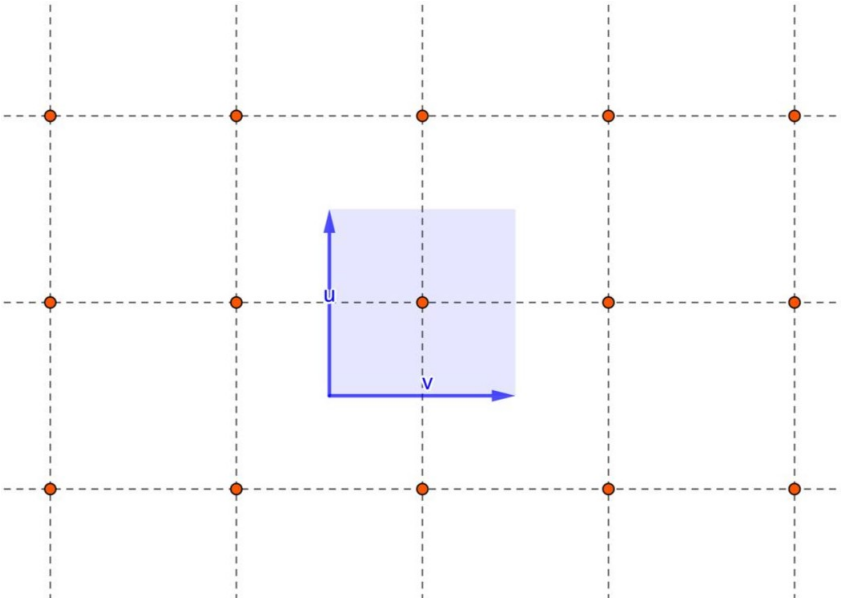

Figura 2.4: $\mathcal{P}((1,0),(0,1))$, con $v = (1,0)$ y $u = (0,1)$

En la imagen de la Fig. 2.4 se observa $\mathcal{P}((1,0),(0,1))$.

Como se puede apreciar, el paralelepípedo fundamental es la región semiabierta delimitada por los vectores $v_1, v_2, \ldots, v_n$. Se tiene el siguiente resultado con el que se puede teselar con nuestro paralelepípedo todo el espacio.

Proposición 2.2.4.1

$$\bigcup_{v\in\mathcal{L}}(v+\mathcal{P}(B)) = \mathbb{R}^n$$

Sea $p \in \mathbb{R}^n$. Se puede escribir p de la siguiente manera:

$$p \;=\; \sum_{i=1}^{n} x_i v_i \;=\; \sum_{i=1}^{n}[x_i]v_i + \sum_{i=1}^{n}(x_i - [x_i])v_i,$$

donde $[x_i]$ es x_i redondeado tomando la parte entera Por ejemplo:

$$x_i = 1{,}9 \Rightarrow [x_i] = 2, \qquad x_i = 1{,}2 \Rightarrow [x_i] = 1$$

Además,

$$\frac{-1}{2} \leq a - [a] < \frac{1}{2}$$

Por tanto,

1. $\displaystyle\sum_{i=1}^{n}[x_i]v_i \in \mathcal{L}$
2. $\displaystyle\sum_{i=1}^{n}(x_i - [x_i])v_i \in \mathcal{P}(B)$

De lo que se concluye que:

$$p = \underbrace{\sum_{i=1}^{n}[x_i]v_i}_{\in\mathcal{L}} + \underbrace{\sum_{i=1}^{n}(x_i - [x_i])v_i}_{\in\mathcal{P}(B)} \Rightarrow \mathbb{R}^n = \bigcup_{v\in\mathcal{L}}(v+\mathcal{P}(B))$$

Solo queda ver que no se superponen los paralelepípedos fundamentales. Supongamos que $(v+\mathcal{P}(B)) \cap (w+\mathcal{P}(B)) \neq \varnothing$. Entonces para ciertos $\alpha, \beta \in \mathcal{P}(B)$:

$$v + \alpha = w + \beta \Rightarrow v - w = \beta - \alpha$$

Dado que $v-w$ es una combinación lineal entera de vectores y $\beta-\alpha$ es una combinación lineal que vive en $(-1,1)$, pues $\alpha, \beta \in \mathcal{P}(B)$. Entonces la única opción es que: $v - w = 0 \Rightarrow v = w$.

■

Definición 2.2.4.2 *Dado un retículo $\mathcal{L} \subset \mathbb{R}^m$ se define su determinante denotado como $det(\mathcal{L})$ como el volumen $n-dimensional$ de su paralelepípedo fundamental.*
Esta cantidad es invariante, ya que no depende de la base escogida. En caso de que el retículo sea de rango máximo $(n = m)$ se tiene que:

$$det(\mathcal{L}) = |det(B)|$$

Se presenta a continuación un resultado de particular interés:

Proposición 2.2.4.2

$$\mathbb{Z}^n/\mathcal{L} \;=\; vol(\mathcal{P}(B)) \;=\; del(\mathcal{L})$$

Se puede encontrar la demostración en (10).

Una forma alternativa de calcular el determinante de un retículo es la siguiente.

Proposición 2.2.4.3 *Dado un retículo $\mathcal{L} \subset \mathbb{R}^n$ de rango máximo con base B se tiene que:*

$$det(\mathcal{L}) = \prod_{i=1}^{n} \parallel b_i^* \parallel$$

donde $B^* = [b_1^*, b_2^*, \ldots, b_n^*]$ es la base de Gram-Schmidt correspondiente.

A continuación se describe el proceso de ortogonalización de Gram-Schmidt de manera matricial:

$$B = [b_1, b_2, \ldots, b_n] = [b_1^*, b_2^*, \ldots, b_n^*] \cdot \begin{pmatrix} 1 & \mu_{21} & \mu_{31} & \cdots & \mu_{n1} \\ 0 & 1 & \mu_{32} & \cdots & \mu_{n2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix} =$$

$$= [\frac{b_1^*}{\parallel b_1^* \parallel}, \ldots, \frac{b_n^*}{\parallel b_n^* \parallel}] \cdot \begin{pmatrix} \parallel b_1^* \parallel & \mu_{21} \cdot \parallel b_1^* \parallel & \mu_{31} \cdot \parallel b_1^* \parallel & \cdots & \mu_{n1} \cdot \parallel b_1^* \parallel \\ 0 & \parallel b_2^* \parallel & \mu_{32} \cdot \parallel b_2^* \parallel & \cdots & \mu_{n2} \cdot \parallel b_2^* \parallel \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \parallel b_n^* \parallel \end{pmatrix}$$

Por lo que:

$$B = B^*T \Rightarrow det(B) = det(B^*)det(T)$$

Como los vectores $\frac{b_i^*}{\|b_i^*\|}$ son ortonormales, el determinante de la matriz de dichos vectores es 1 o -1. Además, al ser la segunda matriz triangular superior se obtiene:

$$det(\mathcal{L}) = |det(B)| = \prod_{i=1}^{n} \parallel b_i^* \parallel$$

■

Proposición 2.2.4.4 *Para cualquier retículo de base $B \in \mathbb{R}^{m \times n}$:*

$$det(\mathcal{L}) = \sqrt{det((B^t)B)}$$

Del resultado anterior se deduce que:

$$B = B^*T$$

con T una matriz triangular superior con unos en la diagonal. Entonces:

$$\sqrt{det((B^t)B)} = \sqrt{det(((B^*T)^t)B^*T)} = \sqrt{det(T^tB^*B^*T)} =$$

$$= \sqrt{det(T^*) \cdot det(B^{*t}B^*)det(T)}$$

Por hipótesis $det(T) = 1$, luego:

$$det(B^{*t}B^*) = \prod_{i=1}^{n} < b_i^*, b_i^* > = \prod_{i=1}^{n} \| b_i^* \|^2 =$$

$$= (\prod_{i=1}^{n} \| b_i^* \|)^2 = (det(\mathcal{L}))^2$$

Por tanto,

$$det(\mathcal{L}(B) = \sqrt{det(B^t B)}$$

■

Nota 2.2.4.1 *Si B es una matriz cuadrada invertible, entonces $det(B) = det(B^t)$, por lo que $det(\mathcal{L}) = |det(B)|$.*

Ejemplo 6 *Tomando la base $B = \{(1, 0, 0), (1, 2, 3)\}$, si se calcula el producto $B^t B$, queda una matriz A de rango 2×2, que es:*

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 3 \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 \\ 0 & 2 \\ 0 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 14 \end{pmatrix} \tag{2.8}$$

Por consiguiente, $det(A) = 13$, y $det(\mathcal{L}) = \sqrt{13}$.

2.3. La Transformada Teórica de Números o *NTT*

La NTT es un caso especial de la transformada discreta de Fourier sobre un cuerpo finito. Hay varios tipo de NTT, pues también se pueden dar convoluciones concretas para $\mathbb{Z}_q[x]/(x^n)$, y $\mathbb{Z}_q[x]/(x^n - 1)$. En el caso que se trata tanto en CRYSTALS-Kyber como en Dilithium es: $\mathbb{Z}_q[x]/(x^n + 1)$

Definición 2.3.0.1 *Sea $q \in \mathbb{N}$ un número primo tal que $q \equiv 1 \ (mod\ 2n)$ tal que la $2n$-ésima raíz de la unidad ψ_{2n} existe en $\mathbb{Z}_q$. Se denota $w_n = \psi_{2n}^2$ (la n-ésima raíz de la unidad en $\mathbb{Z}_q$). Se define entonces la Transformada Teórica de Números basada en la NWC, (Negative Wrapped Convolution) de $a = (a_0, ..., a_{n-1}) \in \mathbb{Z}_q^n$ siendo a el vector de coeficientes de $a(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1} \in \mathbb{Z}_q[x]/(x^n + 1)$ como:*

$$NTT_\psi(a) = (\hat{a}_0, \hat{a}_1, ..., \hat{a}_{n-1}) \tag{2.9}$$

Donde:

$$\hat{a}_j = \sum_{i=0}^{n-1} a_i \psi_{2n}^i w_n^{ij} \ \ (mod\ q) \tag{2.10}$$

Realmente la NTT_ψ se puede definir en $\mathbb{Z}_q^n$ como la aplicación:

$$\begin{array}{rccc} NTT_\psi: & \mathbb{Z}_q^n & \longrightarrow & \mathbb{Z}_q^n \\ & (a_0, a_1, ..., a_{n-1}) & \longmapsto & (\hat{a}_0, \hat{a}_1, ..., \hat{a}_{n-1}) \end{array} \tag{2.11}$$

con $\hat{a}_j$ definidos como *Ec. (2.10).*

Véase que al ser $w_n = \psi_{2n}^2$, se tiene que la expresión *Ec. (2.10)* es equivalente a:

$$\hat{a}_j = \psi_{2n}^{2j+1} \sum_{i=0}^{n-1} a_i \psi_{2n}^i \pmod q$$

Además en este documento se denota de igual manera $NTT(a) := NTT_\psi(a)$ pues no se está considerando otro tipo de convoluciones.

Definición 2.3.0.2 *Partiendo de las mismas hipótesis de la definición de NTT_ψ se define la Transformada Teórica de Números Inversa (Inverse Number Theoretic Transform, INTT) basada en la NWC de $\hat{a} = (\hat{a}_0, \hat{a}_1, .., \hat{a}_{n-1})$ siendo a el vector de coeficientes de $\hat{a}(x) = \hat{a}_0 + \hat{a}_1 x + \hat{a}_2 x^2 + \cdots + \hat{a}_{n-1} x^{n-1} \in \mathbb{Z}_q[x]/(x^n + 1)$ como:*

$$INTT_\psi(\hat{a}) = (a_0, a_1, ..., a_{n-1}) \tag{2.12}$$

donde:

$$a_i = n^{-1} \psi_{2n}^{-i} \sum_{j=0}^{n-1} \hat{a}_j w_n^{-ij} \pmod q \tag{2.13}$$

Proposición 2.3.0.1 *Sea $a(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1} \in \mathbb{Z}_q[x]/(x^n + 1)$ y $a = (a_0, ..., a_{n-1}) \in \mathbb{Z}_q^n$ su vector de coeficientes adecuado, entonces:*

$$INNT_\psi(NTT_\psi(a)) = a \tag{2.14}$$

Esta proposición ya indica que la NTT_ψ es una correspondencia biyectiva.

Proposición 2.3.0.1 *(Propiedad de la NWC). Sea $c(x) \in \mathbb{Z}_q[x]/(x^n + 1)$ la convolución de dos polinomios $a(x), b(x) \in \mathbb{Z}_q[x]/(x^n + 1)$, denotando por c, a, b los vectores de coeficientes de los respectivos polinomios, y $\circ$ al producto punto a punto de dos vectores, entonces:*

$$NTT_\psi(c) = NTT_\psi(a) \circ NTT_\psi(b) \tag{2.15}$$

Como consecuencia directa de las proposiciones anteriores, se tiene que:

$$c = INNT_\psi(NTT_\psi(a) \circ NTT_\psi(b)) \tag{2.16}$$

Por tanto, se puede obtener la multiplicación de dos polinomios en $\mathbb{Z}_q[x]/(x^n+1)$ usando NTT_ψ y $INTT_\psi$ siempre que se asuman las hipótesis.

Definición 2.3.0.3 *Sea $n \in \mathbb{N}$ una potencia de 2 y $b \in \mathbb{Z}$, con $b \geq 0$, se define el bit-reverso de b respecto de n como:*

$$br_n(b) = br_n(b_{log_2(n-1)} 2^{log_2(n-1)} + \cdots + b_1 2 + b_0) =$$

$$b_0 2^{log_2(n-1)} + \cdots + b_{log_2(n-2)} 2 + b_{log_2(n-1)} \tag{2.17}$$

Donde b_i representa el i-ésimo bit en la forma binaria de b. Como se ve es simplemente revertir el orden de los bits y devolver la representación entera.

Ahora como el bit reverso (br) de los enteros $\{0, ..., n-1\}$ forman una permutación de este conjunto, la expresión que se tenía es equivalente a:

$$\mathbb{Z}_q[x]/(x^n+1) = \prod_{j=0}^{n-1} \mathbb{Z}_q[x]/(x - \psi_{2n}^{2br(j)+1}) \tag{2.18}$$

La razón detrás de esta reversión de bits en los índices de exponenciación es que se da un mejor acceso en memoria a la hora de implementar.

Ejemplo 7 *Se pretende observar el cálculo de $a(r_i)$. Para este propósito, se considera el valor de $i = 1$. En consecuencia:*

$$r_1 = r^{brv(128+1)} = r^{brv(129)}$$

Pero, 129 en bits es 10000001, sin embargo si se aplica brv resulta que $brv(10000001) = 10000001$, ya que si se le da la vuelta queda el mismo número, pues es palíndromo. Por lo tanto:

$$r_1 = r^{10000001},$$

con $r = 1753$ y todas las operaciones mod q.

Ahora con *Ec. (2.19)* se sabe que si existe una $2n$-ésima raíz de la unidad, entonces es posible recurrir a una multiplicación punto a punto en la NTT, pues al ser reducido con un polinomio de grado 1 solo quedan constantes.

En Kyber no se puede recurrir a una multiplicación punto a punto de términos constantes, ya que solo existe hasta la n-ésima raíz de la unidad. Luego, se puede aplicar el Teorema Chino del Resto hasta polinomios de grado 2, es decir:

Considerando w_n la n-ésima raíz de la unidad con los parámetros de Kyber, se cumple que:

$$\mathbb{Z}_q[x]/(x^n+1) = \prod_{j=0}^{\frac{n}{2}-1} \mathbb{Z}_q[x]/(x^2 - w_n^{2br(j)+1}) \tag{2.19}$$

2.3.1. Complejidad de la NTT

La complejidad de aplicar NTT/INTT de manera directa es de $O(n^2)$, la cuál es la misma complejidad que se da en la multiplicación de polinomios directa con anillos de la forma $K[x]$ con K cuerpo. Por tanto, la eficiencia de esta transformada se encuentra en los algoritmos que rebajan el orden. Se observa en las siguientes subsecciones, y llegan a conseguir un orden de $O(n \cdot log(n))$.

2.3.2. Ventajas de la NTT

La NTT presenta ciertas propiedades que favorecen a los esquemas de cifrado:

- La NTT es una correspondencia lineal y biyectiva, lo cuál resultará útil para la implementación de Kyber.

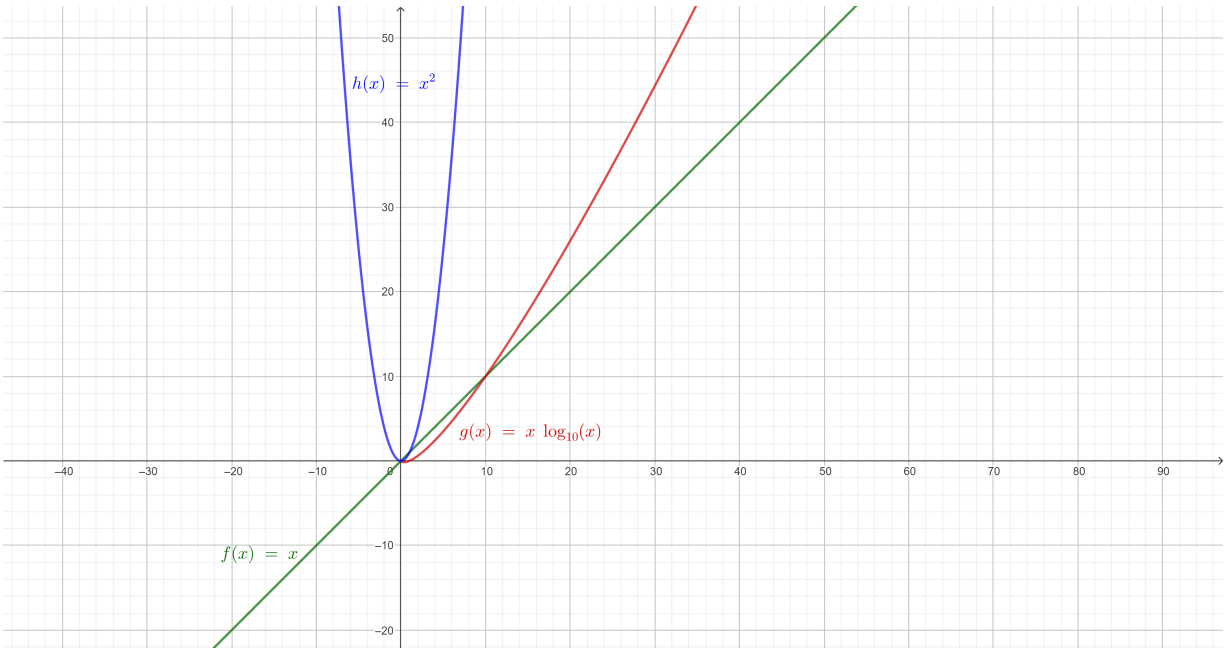

Figura 2.5: Representación de las funciones de orden

- Debido a que la NTT es una correspondencia biyectiva, conserva aleatoriedad de un vector de coeficientes. Con esto se puede generar un vector aleatorio y verlo como un vector ya transformado de la NTT. Esto es posible ya que es sobreyectiva por lo que tendrá un vector que lo tenga como imagen, y además no hay problemas con la aleatoriedad pues la conserva ya que, al ser biyectiva, si la imagen no fuera aleatoria se podría conocer el elemento de entrada, por lo que no sería aleatorio.
- En cuanto al acceso de memoria en una implementación, como la transformada lleva n puntos en n puntos, es posible guardar $\hat{a}$ donde originalmente se encontraba a en memoria.

2.4. Problemas asociados a los retículos

A continuación se define el problema del vector más corto, teniendo en cuenta lo siguiente. Un problema de decisión es aquel en el que la cuestión tiene una respuesta binaria: 'SÍ' o 'NO'. Los problemas de optimización son aquellos en las que se busca minimizar o maximizar el valor de una salida dentro de un grupo de salidas generadas para una entrada específica. Nótese que cualquier problema de optimización siempre puede ser manejado como uno de decisión, fijando de antemano un valor objetivo.

Definición 2.4.1 *Dada una base $B \in \mathbb{Z}^{m \times n}$ de un retículo $\mathcal{L}$, se define el problema del vector más corto (Shortest Vector Problem, SVP) de la siguiente manera:*

- *Búsqueda: Encontrar un vector $v \in \mathcal{L} \setminus \{0\}$ tal que $\| v \| = \lambda_1(\mathcal{L})$.*
- *Optimización: Encontrar $\lambda_1(\mathcal{L})$.*
- *Decisión: Dado un número racional $r \in \mathbb{Q}$, determinar si $\lambda_1(\mathcal{L}) \leq r$.*

A continuación se plantea una variante del problema del vector más corto, que en particular puede resultar de interés cuando el valor objetivo es difícil de conseguir.

Definición 2.4.2 *Dada una base $B \in \mathbb{Z}^{m \times n}$ de un retículo $\mathcal{L}$, se habla del problema del vector más corto (en su variante de aproximación, SVP_γ), con $\gamma(n) > 1$ si:*

- *Búsqueda: Encontrar un vector $v \in \mathcal{L} \setminus \{0\}$ tal que $\| v \| = \gamma(n) \cdot \lambda_1(\mathcal{L})$.*
- *Optimización: Encontrar $z \in \mathbb{Q}$ tal que $z \leq \lambda_1(\mathcal{L}) \leq \gamma(n) \cdot z$.*
- *Promesa: Dado un número racional $r \in \mathbb{Q}$ y los siguientes conjuntos:*

$$SI = \{(B, r) : \lambda_1 \leq r\}$$

$$NO = \{(B, r) : \lambda_1 \geq \gamma(n) \cdot r\}$$

Este problema consiste en diferenciar acertadamente dada una instancia $I \in SI \cup NO$, a cuál pertenece I. La variante promesa se puede denotar de la siguiente manera: $GapSVP_\gamma$.

Algunas observaciones que se deducen a partir de la definición anterior:

- $SI \cap NO = \varnothing$.
- Para toda instancia I, $I \in SI \cup NO$.
- Si $I \notin SI \cup NO$, entonces I toma como válida cualquiera de las dos respuestas.

A raíz del problema del vector más corto nace el problema del vector más cercano. Para ello se define la distancia mínima como sigue:

$$dist(\mathcal{L}, t) = \min_{v \in \lambda \setminus \{0\}} \| v - t \| \tag{2.20}$$

2.4.1. Problema del vector más cercano

Definición 2.4.3 *Dada una base $B \in \mathbb{Z}^{m \times n}$ de un retículo $\mathcal{L}$ y $t \in \mathbb{Z}^m$, se habla del problema del vector más cercano (Closest Vector Problem, CVP) si:*

- *Búsqueda: Encontrar un vector $v \in \mathcal{L} \setminus \{0\}$ tal que $dist(\mathcal{L}, t) = \| v - t \|$ sea mínima.*
- *Optimización: Encontrar $dist(\mathcal{L}, t)$.*
- *Decisión: Dado un número racional $z \in \mathbb{Q}, z > 0$, decidir si existe un $v \in \mathcal{L} \setminus \{0\}$ tal que $dist(\mathcal{L}, t) \leq z$.*

También se considera una versión aproximada de este problema:

Definición 2.4.4 *Dada una base $B \in \mathbb{Z}^{m \times n}$ de un retículo $\mathcal{L}$ y $t \in \mathbb{Z}^m$, se define el problema del vector más cercano (en su variante de aproximación, CVP_γ), con $\gamma(n) > 1$ si:*

- *Búsqueda: Encontrar un vector $v \in \mathcal{L} \setminus \{0\}$ tal que $dist(\mathcal{L}, t) = \| v - t \| \leq \gamma(n) \cdot dist(\mathcal{L}, t)$.*
- *Optimización: Encontrar $z \in \mathbb{Q}$ tal que $z \leq dist(\mathcal{L}, t) \leq \gamma(n) \cdot z$.*

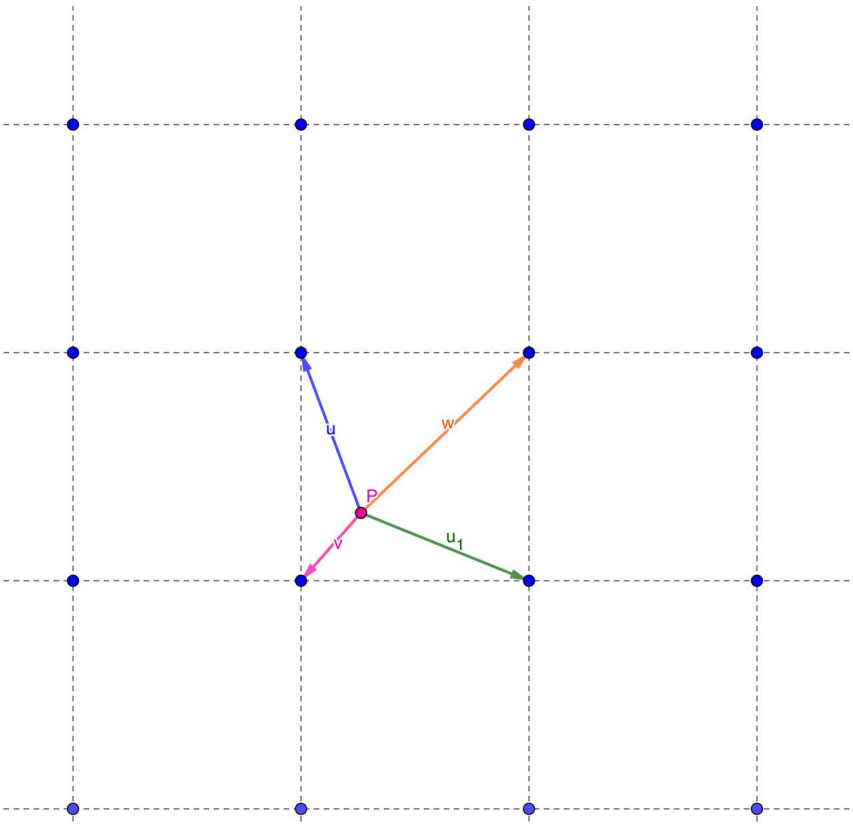

Figura 2.6: Representación del problema del vector más cercano en el plano euclídeo

- *Promesa: Dado un número racional $x \in \mathbb{Q}$ y los siguientes conjuntos:*

$$SI = \{(B, t, x) : dist(\mathcal{L}, t) \leq x\}$$

$$NO = \{(B, t, x) : dist(\mathcal{L}, t) \geq \gamma(n) \cdot x\}$$

Como ya se mencionó anteriormente, este problema consiste en diferenciar de forma acertada dada una instancia $I \in SI \cup NO$, a cuál pertenece I. La variante promesa se puede denotar de la siguiente manera: $GapCVP_{\gamma}$.

2.4.2. Algoritmos de resolución

Sea $\mathcal{L}$ un retículo con una base dada $B \in \mathbb{Z}^{n \times n}$. Sea $w \in \mathcal{L}$. Entonces existen $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{Z}$ tal que:

$$\| w \|^2 = \| \alpha_1 v_1 + \alpha_2 v_2 + \ldots + \alpha_n v_n \|^2$$

Aplicando las propiedades de la norma (ver en Def. 2.1.4.1 y para más detalle consultar en (9)) y el producto interno se tiene que:

$$\| w \|^2 = < \alpha_1 v_1 + \alpha_2 v_2 + \ldots + \alpha_n v_n \, , \, \alpha_1 v_1 + \alpha_2 v_2 + \ldots + \alpha_n v_n > =$$
$$= \alpha_1 \alpha_1 < v_1, v_1 > + \ldots \alpha_n \alpha_n < v_n, v_n > = \alpha_1{}^2 \| v_1 \| + \ldots \alpha_n{}^2 \| v_n \|$$

Por tanto,

$$\| w \|^2 = \sum_{i=1}^{n} \alpha_i^2 \| v_i \|^2 \tag{2.21}$$

Como los coeficientes deben ser enteros, entonces los vectores más cortos de $\mathcal{L}$ deben ser los más cortos de $\{\pm v_1, \pm v_2, \ldots, \pm v_n\}$.

Ahora se analiza el caso del vector más cercano. Sea $x \in \mathbb{R}^n$. Entonces:

$$x = \beta_1 v_1 + \ldots + \beta_n v_n, \qquad con\ b_i \in \mathbb{R},\ \ \forall i = 1, \ldots, n$$

Sea $w \in \mathcal{L}$, con $w = \alpha_1 v_1 + \alpha_2 v_2 + \ldots + \alpha_n v_n$, siendo $\alpha_i \in \mathbb{Z}, \forall i = 1, \ldots, n$. Entonces:

$$\| w - x \|^2 = \| (\alpha_1 - \beta_1)v_1 + (\alpha_2 - \beta_2)v_2 + \ldots + (\alpha_n - \beta_n)v_n \|^2 =$$
$$= < (\alpha_1 - \beta_1)v_1 + \ldots + (\alpha_n - \beta_n)v_n \ , \ (\alpha_1 - \beta_1)v_1 + \ldots + (\alpha_n - \beta_n)v_n > =$$
$$= (\alpha_1 - \beta_1)^2 \| v_1 \|^2 + \ldots + (\alpha_n - \beta_n)^2 \| v_n \|^2$$

Luego,

$$\| w - x \|^2 = \sum_{i=1}^{n} (\alpha_i - \beta_i) \| v_i \|^2 \tag{2.22}$$

Se tiene que $\alpha_i \in \mathbb{Z}$ y $b_i \in \mathbb{R}, \forall i = 1, \ldots, n$. Se puede minimizar la expresión tomando $\alpha_i = [\beta_i]$, siendo $[\beta_i]$ la parte entera de $\beta_i, \forall i = 1, \ldots, n$.

Como conclusión se deduce que cuánto más cortos y más ortogonales son los vectores de la base, mejores son las soluciones a la hora de resolver SVP y CVP. Si se piensa por el contrarrecíproco, es decir, si se quiere complicar el problema, cuánto mayor sea la base, más largos y menos ortogonales sean nuestros vectores, mayor es la dificultad de la resolución.

2.5. El problema de aprendizaje con errores

Considérese un vector secreto de coeficientes enteros $s = (s_1, s_2, \ldots, s_n) \in \mathbb{Z}^n$, y un sistema lineal de m ecuaciones de coeficientes conocidos cuya solución es s (con $m \geq n$), entonces se tiene que:

$$\begin{cases} a = a_{1,1}s_1 + a_{1,2}s_2 + \ldots + a_{1,n}s_n \\ b = a_{2,1}s_1 + a_{2,2}s_2 + \ldots + a_{2,n}s_n \\ \vdots \\ m = a_{m,1}s_1 + a_{m,2}s_2 + \ldots + a_{m,n}s_n \end{cases} \tag{2.23}$$

Resolver este sistema es relativamente sencillo aplicando cualquier método adecuado para esta tarea, como puede ser la *eliminación Gaussiana*. Sin embargo, eso varía cambiando ligeramente la situación. Concretamente se puede introducir un pequeño error en cada ecuación, dejándolas de la forma siguiente:

$$\begin{cases} a = a_{1,1}s_1 + a_{1,2}s_2 + \ldots + a_{1,n}s_n + e_1 \\ b = a_{2,1}s_1 + a_{2,2}s_2 + \ldots + a_{2,n}s_n + e_2 \\ \vdots \\ m = a_{m,1}s_1 + a_{m,2}s_2 + \ldots + a_{m,n}s_n + e_m \end{cases} \tag{2.24}$$

$$\Downarrow$$

$$\begin{cases} a \approx a_{1,1}s_1 + a_{1,2}s_2 + \ldots + a_{1,n}s_n \\ b \approx a_{2,1}s_1 + a_{2,2}s_2 + \ldots + a_{2,n}s_n \\ \vdots \\ m \approx a_{m,1}s_1 + a_{m,2}s_2 + \ldots + a_{m,n}s_n \end{cases} \tag{2.25}$$

Resolver este problema no es tan simple, pues a la hora de aplicar la reducción por filas se acumulan errores de forma que el resultado final estaría posiblemente muy lejos del valor real.

Ejemplo 8 *En base a (12), considérese que existe un vector secreto* $s = (2, 0)$*, y los errores* $e_1 = e_2 = -1$*, entonces el sistema de ecuaciones con errores queda:*

$$\begin{cases} 2s_1 + 3s_2 \approx 5 \\ 3s_1 - s_2 \approx 7 \end{cases}$$

Si en este sistema se tuvieran igualdades, la solución sería $s = (2'36, 0'09)$ *pero como se puede comprobar, ese resultado es diferente del vector secreto inicial* $(2, 0)$*. En la Fig. 2.7 se puede ver gráficamente la solución sin alterar (rosa) y la solución con errores añadidos (azul).*

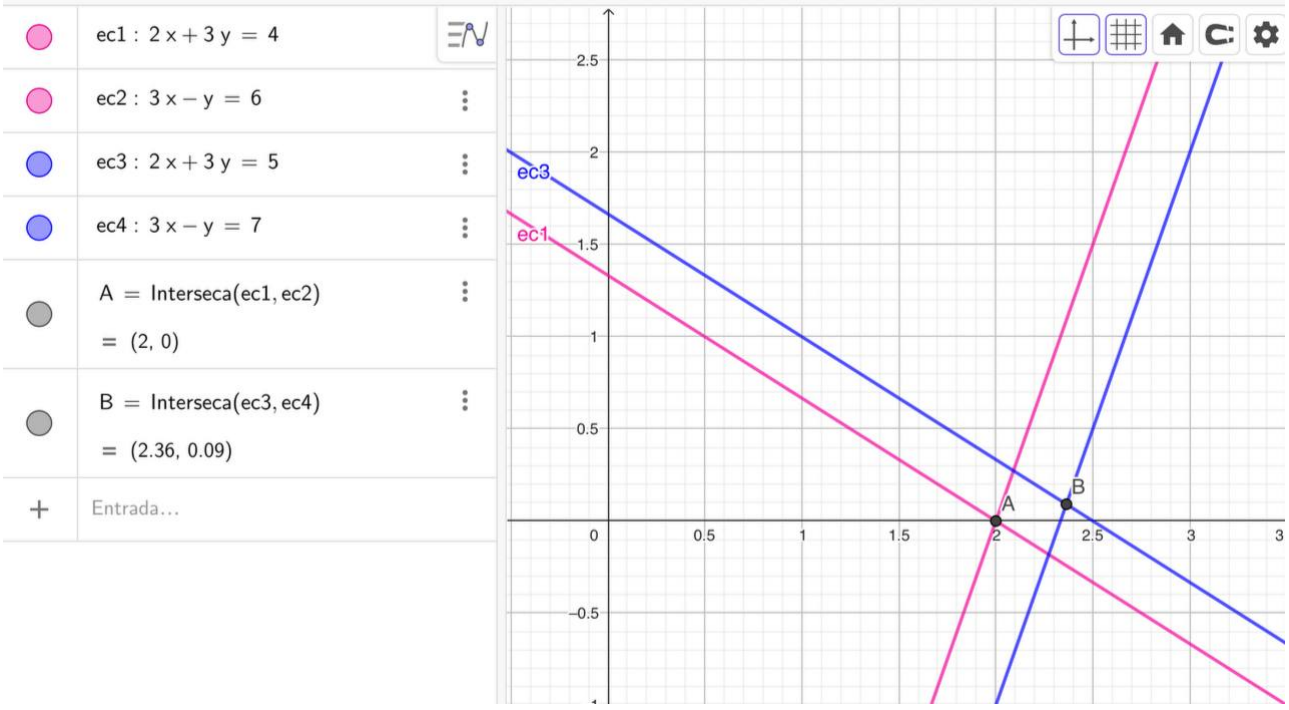

Figura 2.7: Representación gráfica del sistema de ecuaciones original y alterado

2.5.1. Formalización del problema

Sean $n \in \mathbb{N}$ y $q \in \mathbb{Z}$ (cuyo tamaño es similar a n).Considérese además m vectores $b_1, b_2, \ldots, b_m \in \mathbb{Z}_q^n$. El retículo Λ generado por la base de vectores $B = \{b_1, b_2, \ldots, b_m \in \mathbb{Z}_q^n\}$ sería por tanto el conjunto:

$$\Lambda = \{\sum_{i=1}^{m} z_i \cdot b_i \ : \ z_i \in \mathbb{Z}\}$$

Fijada una distribución de probabilidad $\mathcal{X}$ sobre $\mathbb{Z}_q$, que de alguna forma permite elegir un cierto término de error de manera controlada. La selección de un error e de acuerdo con esta distribución se denota $e \leftarrow \mathcal{X}$. Estos parámetros sirven para fijar la llamada *distribución LWE* sobre $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

Definición 2.5.1 *Distribución LWE. Sean $n, q \in \mathbb{N}$, $s \in \mathbb{Z}_q^n$ y $\mathcal{X}$ una distribución de probabilidad sobre $\mathbb{Z}_q$. Se dice que la distribución LWE módulo q asociada a s, denotada $\mathcal{A}_{s,\mathcal{Z}}$, es la definida eligiendo un vector $a \in \mathbb{Z}_q$ uniformemente al azar y eligiendo un error $e \leftarrow \mathcal{X}$, seleccionando así:*

$$(a, b) \in \mathbb{Z}_q^n \times \mathbb{Z}_q, \quad con \ b \ = < s, a > + \ e, \quad mod \ q \tag{2.26}$$

De forma natural se puede asociar al problema LWE un problema de búsqueda.

- Búsqueda LWE. Sean $n, m, q \in \mathbb{N}$ y sea s escogido uniformemente al azar en $\mathbb{Z}_q^n$. y además se considera $\mathcal{A}_{s,\mathcal{Z}}$ la distribución anteriormente definida. El problema consiste en calcular s dados m elementos de $\mathbb{Z}_q^n \times \mathbb{Z}_q$ elegidos de manera independiente.

2.5.2. Aprendizaje con errores sobre anillos

Según (13), siendo $n, m, q \in \mathbb{Z}$, el problema de aprendizaje con errores se basa en encontrar un vector $s \in \mathbb{Z}_q^n$ tal que $As + e = b \ mod \ q$, donde $A \in \mathbb{Z}_q^{m \times n}$, $e, b \in \mathbb{Z}_q^n$. El vector e es el error y viene dado por la distribución $\mathcal{X}^m$ de probabilidad en $\mathbb{Z}_q^m$.

Se toma el anillo $\mathcal{K}_q = \mathbb{Z}_q[x]/(x^n + 1)$, con n siendo $2^{n'-1}$ tal que $(x^n + 1)$ es el $2^{n'}$ - ésimo polinomio ciclotómico. Entonces el par (A, b) viene dado por $A \in \mathcal{K}_q$ y $b = As + e$ mod qK con $s \in \mathcal{K}_q$ y e obtenido de la distribución de probabilidad $\mathcal{X}$ mod q. Esto es el llamado *Module-LWE*.

La principal ventaja del esquema *Ring-LWE* es la eficiencia, en términos de velocidad, como del tamaño de la clave del texto cifrado. En cambio sus desventajas son la preocupación de que la estructura adicional pueda permitir ataques más eficientes y que las compensaciones entre eficiencia y seguridad solo se pueden escalar de forma bastante aproximada.

2.6. Complejidades computacionales

Se introducen una serie de resultados para entender las complejidades computacionales de cada problema. Para profundizar más se puede consultar (6).

Teorema 2.6.1 *El problema CVP en su versión de decisión es NP - completo.*

El siguiente resultado muestra que para $\gamma \geq 1$ encontrar soluciones a SVP_γ no es más difícil que encontrar soluciones a CVP_γ. Nótese que para $\gamma = 1$ resulta la versión primigenia.

Teorema 2.6.2 *$\forall \gamma \geq 1$ dado acceso a un oráculo que resuelva $GapCVP_\gamma$, es posible resolver $GapSVP_\gamma$ en tiempo polinomial.*

En pocas palabras, si no se resuelve en tiempo polinomial $GapSVP_\gamma$, tampoco $GapCVP$. Además, se debe aclarar que el término *oráculo* es una máquina abstracta usada para estudiar problemas de decisión y que hace referencia a la opacidad del proceso del cálculo del ejercicio en cuestión.
Para el caso aproximado se tiene que el siguiente resultado:

Teorema 2.6.3 *Para toda $c > 0$ y $\gamma(n) = n^{\frac{c}{log(log(n))}}$. Entonces:*

$$GapCVP_\gamma \in NP - Difcil$$

En cuanto al problema del vector más corto, $GapSVP_\gamma$, se conjetura que no existen algoritmos clásicos (pre-cuánticos) que puedan lograr soluciones aproximadas con factores de aproximación lineales. Incluso, se considera la conjetura de que no existen algoritmos cuánticos que puedan resolver este problema con factores de aproximación lineales.

2.6.1. Glosario

En esta sección se repasan algunos términos relacionados con las complejidades computacionales.

Definición 2.6.1 *Sean $f, g : \mathbb{N} \to \mathbb{R}^+ \cup \{0\}$ funciones. Entonces se llama notación asintótica a:*

- *$f(n) \in \mathcal{O}(g(n))$ si existe una constante $c \in \mathbb{R}^+$ y un $n_0 \in \mathbb{N}$ tales que:*

$$\forall n \geq n_0, \;\; f(n) \leq c \cdot g(n) \tag{2.27}$$

con $\mathcal{O}(g(n))$ cota superior asintótica.

$$\mathcal{O}(f(n)) \;\; = \;\; \{h : \mathbb{N} \to \mathbb{N} \;/\; \exists\, c > 0, n_0 > 0 : \;\; h(m) \leq c \cdot f(m), \;\; \forall m \geq n_0\}$$

Siendo $\mathcal{O}(f)$ el conjunto de las funciones acotadas por $c \cdot f$ para algún $c \in \mathbb{R}$.

En la teoría de la complejidad computacional, los problemas computacionales son modelados como subconjuntos $L \subset \{(0,1)^*\}$, llamados lenguajes, donde para todo $x \in \{(0,1)^*\}$ o bien $x \in L$ o bien $x \notin L$. Se introduce a continuación el tiempo polinomial.

Definición 2.6.2 *Un algoritmo $\mathcal{A}$ ejecuta en tiempo polinomial, si existe un polinomio $p : \mathbb{N} \longrightarrow \mathbb{N}$ tal que:*
$\forall x \in \{(0,1)^*\}$ el cómputo de $\mathcal{A}[x]$ termina a lo sumo en $p(|x|)$ pasos, donde $|x|$ denota el largo de la cadena de caracteres de x.

2.6.2. P vs NP

Sin entrar en profundidad, se sabe que hay distintas clases y las más importantes son: P, NP - Completo y NP - Difícil.

Definición 2.6.3 *La clase P se define como el conjunto de todos los problemas de decisión que pueden ser decididos por un algoritmo (determinista) en tiempo polinomial.*

La clase NP es el conjunto de los problemas que se pueden comprobar en tiempo polinomial.

Definición 2.6.4 *Un lenguaje de decisión $L \subset \{(0,1)^*\}$ está en NP si existe un polinomio $p : \mathbb{N} \longrightarrow \mathbb{N}$ y un algoritmo en tiempo polinomial $\mathcal{M}$ tal que para todo $x \in \{(0,1)^*\}$:*

$$x \in L \iff \exists\, u \in \{(0,1)^{p(|x|)}\} \ : \ \mathcal{M}(x,u) = 1 \tag{2.28}$$

Si $x \in L$ y $u \in \{(0,1)^*\}$ satisfacen que $\mathcal{M}(x,u) = 1$, se dice que u es un certificado para x.

Para poder tratar con cierto rigor sobre las clases NP-Difícil y NP-Completo se debe introducir la reducción en tiempo polinomial, ya que es un ingrediente clave en la definición de los mismos.

Definición 2.6.5 *Un lenguaje $L \subset \{(0,1)^*\}$ se reduce en tiempo polinomial al lenguaje $L' \subset \{(0,1)^*\}$, denotado $L \leq_p L'$, si existe una función computable en tiempo polinomial $f : \{(0,1)^*\} \longrightarrow \{(0,1)^*\}$ tal que:*

$$\forall x \in \{(0,1)^*\},\ x \in L \iff f(x) \in L' \tag{2.29}$$

Es decir, una reducción en tiempo polinomial se puede ver como una función que computa instancias que pertenecen a L a instancias que también pertenecen a L'. Como se tiene una doble implicación se puede manipular de igual manera con los contrarrecíprocos. La ventaja de esta definición es que si se tiene una función que transforma instancias de L en L' y además un algoritmo que resuelve L', también se tiene un algoritmo que resuelva L.

Definición 2.6.6 *Un lenguaje L' es NP-Difícil si para todo $L \in NP$, $L \leq_q L'$.*

Definición 2.6.7 *Un lenguaje L' es NP-Completo si es NP-Difícil y $L' \in NP$.*

Lo que estas definiciones quieren decir es que todo problema $L' \in$ NP-Difícil es tan complicado como cualquiera NP, esto es porque existe una reducción polinomial de L a L' por definición. Además, si se impone que $L' \in NP$ entonces se obtienen los problemas NP-Completos, que serían los más complicados de resolver.

Ahora que ya se ha dado una noción sobre las distintas clases computacionales se puede entender la relevancia del problema $P\, vs\, NP$. Es lógico ver que todo problema que se resuelve fácilmente se comprueba con un esfuerzo parecido, por lo que es claro que $P \subset PN$, sin embargo, ¿$NP \subset P$? En el caso de que $P = NP$ esto supondría que para cualquier problema existe un algoritmo que resuelve de manera eficaz el ejercicio en cuestión, sin necesidad de emplear la fuerza bruta.

Este dilema es uno de los 7 problemas del milenio. La comunidad científica cree y avala que se avanza en dirección a que $P \neq NP$, y es por eso que se escogen los problemas SVP, CVP y LWE como base de algunos esquemas (como es el nuestro en este caso: CRYSTALS-Dilithium), debido a que es realmente difícil resolverlos en un tiempo razonable, ya sea usando un ordenador clásico como uno cuántico.

2.7. CRYSTALS-Dilithium

En este capítulo se incluye una introducción al esquema de firma CRYSTALS - Dilithium basado en criptografía de clave pública. La dificultad de este esquema de firma digital se basa en el problema de encontrar los vectores más cortos en un retículo y en el ya mencionado *LWE*.

La criptografía de clave pública (también llamada asimétrica) emplea un par de claves (pública y privada) para definir los distintos roles, como pueden ser emisor/receptor, firmante/verificador, etc. de las dos entidades involucradas en el proceso. Se basa en problemas computacionales que son eficientes de realizar en un sentido, pero que no es factible realizarlos en sentido contrario (unidireccionalidad).

Definición 2.7.1 *Un esquema de firma de clave pública Sig es una terna de algoritmos ejecutables en tiempo polinomial $Sig = (\mathcal{K}, \mathcal{F}, \mathcal{V})$, donde:*

- *$\mathcal{K}$ es el algoritmo de generación de claves, es un algoritmo probabilístico que, recibiendo como entrada al parámetro de seguridad $l \in \mathbb{N}$, produce como salida un par (pk, sk) de clave pública y de clave privada.*
- *$\mathcal{F}$ es el algoritmo de firma, es un algoritmo probabilístico que, recibiendo como mensaje de entrada $\mathcal{M}$ codificado como una cadena de $p(l)$ bits, para un cierto polinomio p junto con la clave secreta sk, da como salida el mensaje M y su firma σ que, de nuevo, se codifica como una cadena de bits de longitud polinomial l.*
- *$\mathcal{V}$ es el algoritmo de verificación, es un algoritmo determinista que, recibiendo como entrada un par (M, σ) y una clave pública pk, da como salida un bit que puede ser 0 o 1 indicando el fracaso o el éxito de la operación.*

2.7.1. Esquema inicial

A continuación se presentan dos esquemas. El primero de ellos es una versión simplificada denominada *Template*, que se utiliza con el objetivo de facilitar la comprensión de la versión final. Para ello, se desarrollan algunos conceptos fundamentales relacionados con la fase de firma digital.

Funciones hash. Una función hash es un algoritmo matemático que transforma cualquier dato entrante en una serie de caracteres de salida, con una longitud fija o variable, dependiendo del algoritmo hash que se esté utilizando. CRYSTALS-Dilithium utiliza alguna de las implementaciones de SHA-3 *(Secure Hash Algorithm 3)*, que se estandarizó en el 2015 por el NIST.

El conjunto B_τ está implicado en la proceso de crear la firma. B_τ es el conjunto de elementos de $\mathcal{R}_q$ que tienen τ coeficientes que únicamente son 1 o -1 y el resto son 0. Véase en la Fig. 2.8 el algoritmo que los genera de forma aleatoria.

Se expone un resultado para caracterizar los conjuntos cocientes, teniendo anillos e ideales de polinomios.

Proposición 2.7.1 *Sean A un cuerpo e I un ideal de A, siendo $I = p(x)$ con $p(x) \in A[x]$ un polinomio de grado n. Entonces se tiene que:*

$$A/I \ = \ \{(a_0 + a_1 x + a_2 x^2 + \dots a_{n-1} x^{n-1}) + I : a_i \in A\} \tag{2.30}$$

SampleInBall(ρ)
01 Initialize $\mathbf{c} = c_0 c_1 \dots c_{255} = 00 \dots 0$
02 **for** $i := 256 - \tau$ to 255
03 $\quad j \leftarrow \{0, 1, \dots, i\}$
04 $\quad s \leftarrow \{0, 1\}$
05 $\quad c_i := c_j$
06 $\quad c_j := (-1)^s$
07 **return c**

Figura 2.8: Algoritmo que calcula un elemento de B_τ de forma aleatoria

Ejemplo 9 *Sean $\tau = 2$ y el anillo $\mathbb{Z}_3[x]/(x^4 + 1)$, por Ec. (2.30) se tiene que:*

$$\mathbb{Z}_7[x]/(x^3 + 1) = \{(a_0 + a_1 x + a_2 x^2) + I / a_i \in \{0, 1, \dots, 6\}, \forall i\}$$

Por tanto, en este caso:

$$B_\tau = \{(a_0 + a_1 x + a_2 x^2) + I / a_i \in \{0, 1, -1\}$$

Elementos de B_τ podrían ser:

- $1 + I$.
- $(1 + x) + I$.
- $(-1 - x^2) + I = (6 + 6x^2) + I$. *Pues se considera $\mathbb{Z}_7$ y $-1 = 6$.*

Reducciones modulares:

- Caso α par. Se define la reducción modular como sigue:

$$r' = r \; mod^{\pm} \; \alpha, \text{ con } r' \text{ el único } r \in (-\frac{\alpha}{2}, \frac{\alpha}{2}] : r' \equiv r \; mod \; \alpha$$

- Caso α impar. Se define la reducción modular como sigue:

$$r' = r \; mod^{\pm} \; \alpha, \text{ con } r' \text{ el único } r \in [-\frac{\alpha - 1}{2}, \frac{\alpha - 1}{2}] : r' \equiv r \; mod \; \alpha$$

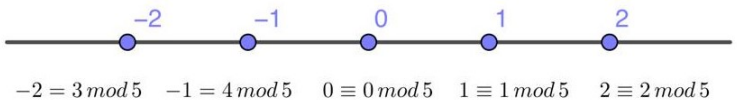

Figura 2.9: Representación gráfica de una reducción modular con $\alpha = 5$

En las Fig. 2.10 y 2.11 se pueden apreciar los algoritmos complementarios y la versión inicial del esquema de firma. Ahora que ya se han introducido los conceptos necesarios, se procede a estudiar cada algoritmo del *Template*.

Como en cualquier esquema de firma, el esquema completo de CRYSTALS-Dilithium consta de tres algoritmos: Generación de clave, proceso de firma y proceso de verificación.

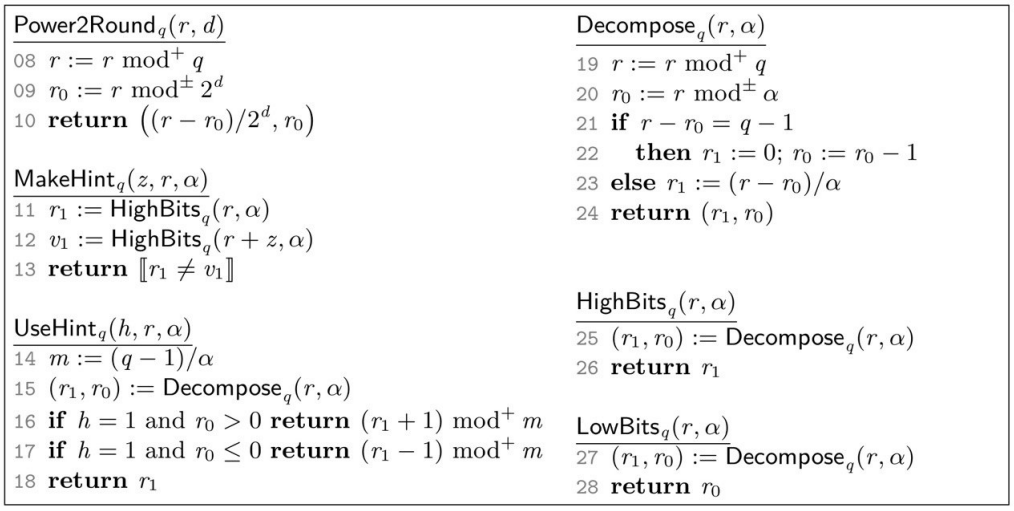

Figura 2.10: Algoritmos complementarios al esquema CRYSTALS-Dilithium

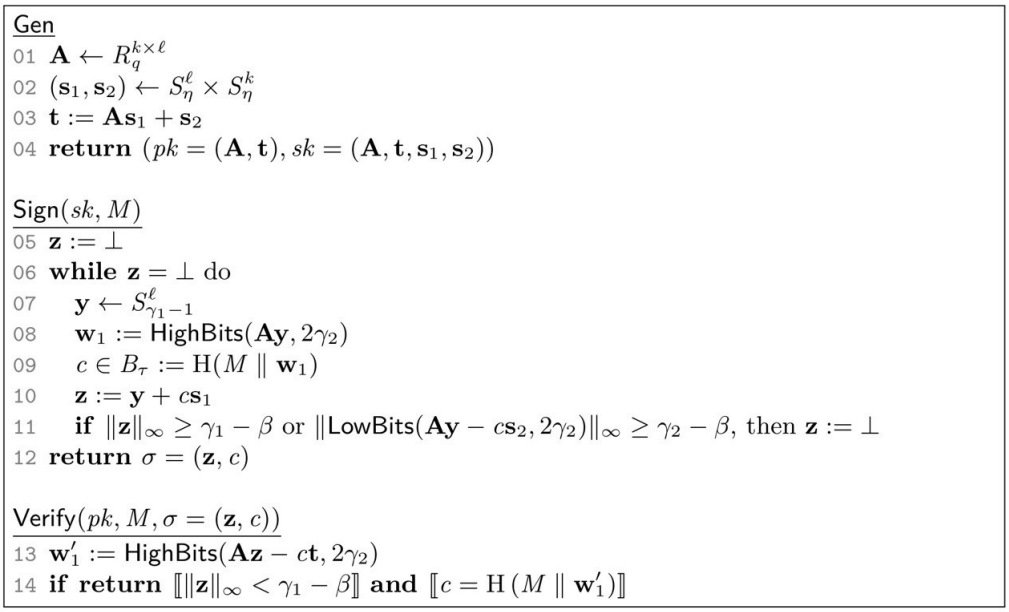

Figura 2.11: *Template,* Primera versión de CRYSTALS-Dilithium

- Generación de clave. El algoritmo empieza generando una matriz aleatoria de dimensión $k \times l$, en particular se suele tomar 5×4 o 6×5 e incluso 8×7. Los elementos de esta matriz pertenecen a $\mathcal{R}_q$. Se sabe que $\mathcal{R}_q = \mathbb{Z}_q[x]/(x^n + 1)$, donde $q = 2^{23} - 2^{13} + 1$ y $n = 256$. Dado que q es un número primo, se tiene que $\mathbb{Z}_q$ es un cuerpo, por lo que esto implica que $\mathbb{Z}_q[x]$ también lo es. Se deduce de Eq. 2.30 que los elementos de $\mathcal{R}_q$ denotando como $I = (x^{256} + 1)$ se expresan de la siguiente forma:

$$\mathcal{R}_q \ = \ \{(a_0 + a_1 x + a_2 x^2 + \ldots + a_{255} x^{255} \ + \ I : a_i \in \mathbb{Z}_q\}$$

  En el segundo paso se generan dos vectores s_1 y s_2, siendo $(s_1, s_2) \in S_\mu^l \times S_\mu^k$ y S_μ el conjunto de todos los elementos de $\mathcal{R}_q$ tal que $\| \ w \ \|_\infty \leq \mu$.
  Luego se computa $\mathbf{t} := As_1 + s_2$. En esta fase se aprecia cómo se relaciona con el problema de L.W.E. siendo s_1 el 'vector secreto' y s_2 el 'vector error'. Para finalizar este algoritmo se devuelve la clave tanto pública $pk = (A, t)$ como privada $sk = (A, t, s_1, s_2)$.
  Se recuerda lo siguiente:

$$||x||_\infty = sup\{x_n\}_{n \in \mathbb{N}}$$

- Proceso de firma. Para iniciar este algoritmo son necesarios dos parámetros: el mensaje $\mathcal{M}$ y la clave secreta generada anteriormente.

  El proceso comienza con un bucle *while* que sirve para crear la firma digital. Se obtiene un vector $y \in S_{\gamma_1 - 1}^l$ de forma aleatoria, donde γ_1 se escoge arbitrariamente de manera que no sea ni lo suficientemente largo ni tampoco lo suficientemente corto, para que no sea fácil forzarla ni tampoco la firma puntual revele la clave.

  En el siguiente paso se define w_1, que es el resultado de aplicar HighBits a Ay y $2\gamma_2$. Nótese que γ_2 se obtiene de expresar todo elemento w del producto Ay como:

$$w = w_1 \cdot 2\gamma_2 + w_0, \qquad |w_0| \leq \gamma_2 \tag{2.31}$$

  A continuación se define un vector c, donde $c \in B_\tau$, que es el resultado de aplicar el algoritmo hash t al mensaje $\mathcal{M}$ con una longitud fija w_1. Se computa el candidato a firma $z = y + c \cdot s_1$, recordando que $s_1 \in sk$.

  Para terminar este proceso la firma z tiene que cumplir dos condiciones:

  1. $\| \ z \ \|_\infty \ \geq \ \gamma_1 - \beta$, siendo β el coeficiente más grande del producto $c \cdot s_i$, con $i = 1, 2$.
  2. $\| \ LowBits(Ay - c \cdot s_2, 2\gamma_2) \ \|_\infty \ \geq \ \gamma_2 - \beta$.

  En el caso de que no se cumpla ninguna de las dos condiciones se repite el proceso. Finalmente una vez concluye este bucle *while*, se obtiene la firma digital:

$$\sigma = (z, c)$$

- Proceso de verificación. Para ejecutar este algoritmo se necesita como parámetro de entrada la clave secreta sk, el mensaje $\mathcal{M}$ y la firma $\sigma = (z, c)$. Bastará con definir $w_2 = HighBits(Az - ct, 2\gamma_2)$, y luego comprobar las condiciones pertinentes. En caso de que se cumplan ambas se nos devolverá un 1, y en caso contrario un 0.

Nota 2.7.1 *En sí, el esquema planteado es un tanto ineficiente, debido a que la matriz generada es una matriz de polinomios de tamaño $k \times l$. Es decir, se operan $k \cdot l$ polinomios, añadiendo que cada polinomio puede tener a lo sumo 256 coeficientes y además se encuentran en el intervalo $[0, q)$ con $q = 2^{23} - 2^{13} + 1 = 8380417$, por lo que se tiene que la matriz que genera la clave es de unas dimensiones muy considerables.*

2.7.2. Esquema final

A continuación se presenta la versión final de CRYSTALS-Dilithium, prestando especial atención a algunos aspectos que destacan con respecto al *Template*.

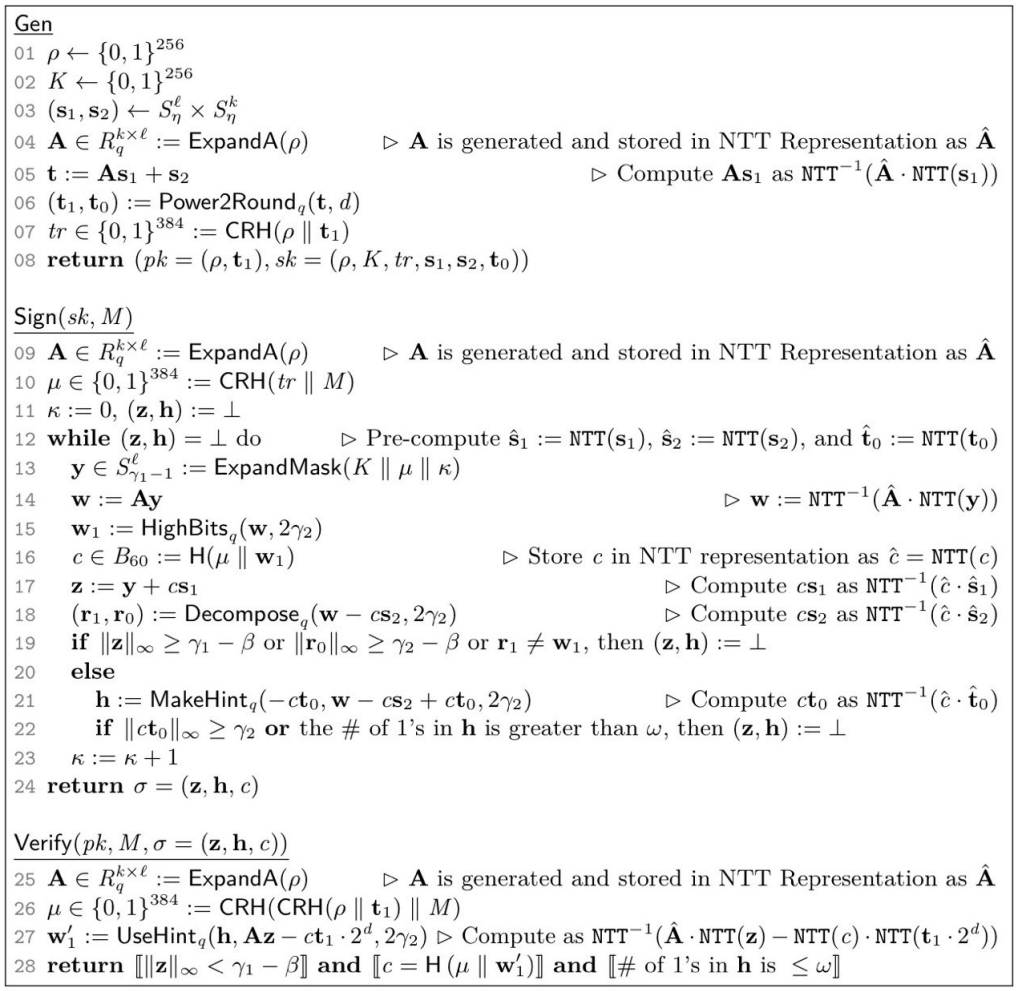

Figura 2.12: Última versión de CRYSTALS-Dilithium

En la Fig. 2.12 se pueden ver algunas diferencias con respecto al *Template*, como pueden ser:

- El uso de semillas para iniciar los procesos aleatorios, como puede ser los vectores y la matriz en el primer algoritmo.

- Se utiliza la Transformación Teórica de Números, para optimizar gran parte de los cálculos que están involucrados en la generación aleatoria de los parámetros.
- Se utilizan nuevos algoritmos (que ya han sido introducidos en la Fig. 2.10) tales como $Decompose_q$, $MakeHint_q$ o $Power2Round_q$.

En las próximas líneas, se exponen en mayor profundidad algunos de los procesos y algoritmos que se emplean en el esquema en cuestión.

- Expansión de la matriz. La función *ExpandA(p)* asigna uniformemente una semilla $p \in \{0,1\}^{256}$ a una matriz $A \in \mathcal{R}_q^{k \times l}$ en la representación NTT. Esto es debido a que la matriz solo es necesaria para la multiplicación y dadas sus dimensiones en aras de ahorrar coste computacional se aplica la Transformación Teórica de Números. Por lo que si se tiene $A \in \mathcal{R}_q^{k \times l}$ como entrada, el valor de salida es $\hat{A} \in \mathbf{Z}_q^{256}$.

- Muestreo de vectores. Con la función *ExpandS* se generan los vectores secretos en la fase de generación de clase, asignando una semilla p' a $(s_1, s_2) \in S_\mu^l \times S_\mu^k$. También se puede hablar de *ExpandMask* que se utiliza para generar de manera determinista la aleatoriedad del esquema de firma.

- Resistencia a las colisiones. En el esquema de firma Dilithium se utiliza una función hash resistente a las colisiones que se asigna a $\{0,1\}^{384}$. Estas funciones tienen la propiedad de que es difícil encontrar dos entradas que tengan la misma salida. Si se tiene una función hash con más entradas que salidas necesariamente habrá colisiones.

Capítulo 3

Contribuciones

En este capítulo se explican las aportaciones realizadas en este Trabajo Fin de Máster. Concretamente, se destacan tres publicaciones aceptadas, que están incluidas en el Apéndice A.1, el Apéndice A.3 y el Apéndice A.4. Por otra parte, el Apéndice A.2 y el Apéndice A.5 corresponden a dos *Extended Abstracts* que han sido aceptados y cuyos *Full Papers* están en proceso de revisión.

3.1. Sobre un ataque backdoor a CRYSTALS-Dilithium

El problema tratado en el trabajo recogido en el Apéndice A.1 es el de estudiar, analizar y atacar el esquema de firma CRYSTALS-Dilithium. Para este propósito, se ha tomado como referencia el artículo (14) al elaborar el documento.

El ataque propuesto se basa en cleptografía, una rama de la criptografía dedicada a investigar métodos y técnicas para sustraer información o claves secretas de un usuario sin su conocimiento. Para ello se utiliza la noción de SETUP, introducida en (15). El artículo se compone de dos ataques, cada uno enfocado a un elemento del par que compone la firma digital de Dilithium, es decir, a cada elemento de la firma $\sigma = (z, c)$ le corresponde un ataque.

Para realizar el ataque diseñado, se supone la siguiente situación: Alice y Bob quieren compartir un documento o mensaje, sin embargo, Alice no desea que nadie más lea el documento y para ello lo cifra con una clave que solo conoce Bob. Además, para que Bob se asegure de que el documento es de ella, Alice firma el documento cifrado. Entonces, la idea del atacante es exfiltrar el documento o mensaje, sin que Alice ni Bob se percaten. No obstante, al estar cifrado y desconocer la clave de descifrado, el atacante decide exfiltrar el documento o mensaje a través de la firma.

La firma $\sigma = (z, c)$ de CRYSTALS - Dilithium, donde z es de la siguiente forma:

$$z = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_k \end{pmatrix}, \quad p_i \in \mathcal{R}_q, \quad \forall i \in \{1, 2, \ldots, k\} \tag{3.1}$$

Es decir, $256 \cdot k$ coeficientes componen z, donde cada $p_{i,j} \in \mathbb{Z}_q$ tal que:

$$z = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,256} \\ p_{2,1} & \alpha_{2,2} & \cdots & p_{2,256} \\ \vdots & \vdots & \ddots & \vdots \\ p_{k,1} & p_{k,2} & \cdots & p_{k,256} \end{pmatrix} \cdot (X^k)^t, \;\; X = \begin{pmatrix} 1 \\ x \\ x^2 \\ \vdots \\ x^{255} \end{pmatrix} \tag{3.2}$$

Por otra parte, $c \in \mathcal{B}_\tau$, donde

$$\mathcal{B}_\tau = \{a_0 + a_1 \cdot x + \ldots + a_{255} \cdot x^{255} : a_i \in \{0, 1, -1\}\} \tag{3.3}$$

Además, en $\mathcal{B}_{tau}$ hay exactamente τ $1s$ ó $-1s$ y el resto son $0s$.

Por tanto, a la hora de exfiltrar el mensaje o documento M, este se codifica en bits, denotandose como M_{bits}. Una vez se ha realizado este paso, se le aplica una operación XOR con una semilla ρ aleatoria generada por el atacante, obteniendo así secreto perfecto:

$$M_\rho = M_{bits} \oplus \rho \tag{3.4}$$

Posteriormente, el atacante debe realizar lo siguiente, dependiendo de si elige extraer M_ρ a través de z o de c:

- Si se extrae a través de z, cada bit de M_ρ debe ser codificado según la paridad en z, con la correspondencia siguiente:

$$\begin{aligned} \varphi \colon M_\rho &\longrightarrow A \\ x &\longmapsto a, \\ \varphi(0) = [0], \;\; \varphi(1) &= [1] \end{aligned}$$

  Donde $\varphi(0) \in \{0, 2, \ldots, q-1\}$, $\varphi(1) \in \{1, 3, \ldots, q-2\}$. También se puede realizar la correspondencia del bit 0 a los impares y del bit 1 a los pares

- Si se extrae a través de c se debe tener en consideración que la longitud de M_ρ puede ser mayor que la de c, que c justamente cuenta con 256 coeficientes, luego, para ello se crean las funciones *ChangeSign*, *IntroduceBits* y *ReduceBits*

Una vez se ha realizado la correcta codificación del mensaje o documento M_ρ en la firma σ, dado que la semilla ρ es almacenada por el atacante, por las propiedades de la operación XOR se tiene que:

$$M_{bits} = M_\rho \oplus \rho \tag{3.5}$$

Luego solo restaría volver a codificar los bits en el mensaje o documento perteneciente a Alice.

3.2. Sobre la Transformada Teórica de Números (*NTT*)

El problema tratado en el trabajo recogido en el Apéndice A.3 es el de abordar y estudiar en profundidad la Transformada Teórica de Números (*Number Theoretic Transform, NTT*). La NTT es una técnica realmente importante y compleja en el campo de la teoría de números computacional y la criptografía. Es una variante de la Transformada Rápida

de Fourier (FFT), que se ha adaptado para trabajar con anillos.

Los estándares de cifrado y de firma digital post-cuánticos como CRYSTALS-Kyber y CRYSTALS-Dilithium se basan en la teoría de retículos. Específicamente, los componentes clave de estos esquemas son polinomios del anillo $\mathcal{R}_q = \mathbb{Z}_q/(x^n + 1)$, con $q = 2^{23} - 2^{13} + 1$ y $n = 256$. Sin embargo, el a la hora de llevar a la práctica las operaciones con matrices compuestas por polinomios de este tipo, son ineficientes, es por ello que se utiliza la NTT. La NTT es fundamental en la implementación eficiente de algoritmos para multiplicación de polinomios y otros problemas en teoría de números.

El trabajo recogido en el Apéndice A.3 pretende arrojar luz y comprensión sobre la NTT, lo que es crucial, ya que uno de los desafíos principales al estudiar los esquemas post-cuánticos recientemente estandarizados es la falta de atención dedicada a esta función. A lo largo del artículo se refuerzan los siguientes puntos con detenimiento:

- Teoría de anillos preliminar necesaria para comprender la Transformada Teórica de Números, como la k-ésima raíz primitiva de la unidad y el Teorema Chino del Resto
- Se desglosan los distintos tipos de convoluciones que se pueden encontrar a la hora de multiplicar polinomios con coeficientes en un anillo, particularizando en la *Negative Wrapped Convolution* que es la que se trata en CRYSTALS-Kyber
- Complejidad de la NTT frente a la complejidad de la multiplicación de polinomios ($n \cdot \log(n)$ frente a n^2)
- Algoritmos Cooley-Tukey y Gentleman-Sande, que son los que mejoran en gran medida el rendimiento de la NTT y son los involucrados en la multiplicación de polinomios. La idea de dichos algoritmos se puede ver en Fig. 3.1:

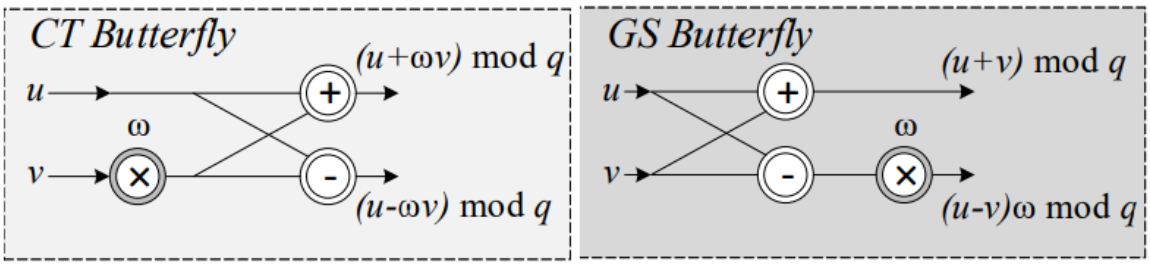

Figura 3.1: Algoritmos Cooley-Tukey y Gentleman-Sande

- Por último, se presenta una implementación eficiente en Python, en la que se explica paso a paso cada sección del algoritmo

Este trabajo representa un gran avance pues aborda y estudia en profundidad un tema de gran complejidad como es la Transformada Teórica de Números, que además es de suma importancia en los actuales estándares post-cuánticos basados en retículos. Su abordaje exhaustivo representa un avance considerable en la comprensión y aplicación de la criptografía post-cuántica.

3.3. Sobre recursos didácticos con retículos y GeoGebra

El problema que se trata en el trabajo recogido en el Apéndice A.4 consiste en el desarrollo de material didáctico y pedagógico sobre la matemática subyacente a los estándares de criptografía post-cuántica basada en retículos. Para el desarrollo de este trabajo hay dos motivos:

- El bajo rendimiento de los estudiantes en las pruebas PISA (véase en (16) y Fig. 3.2)

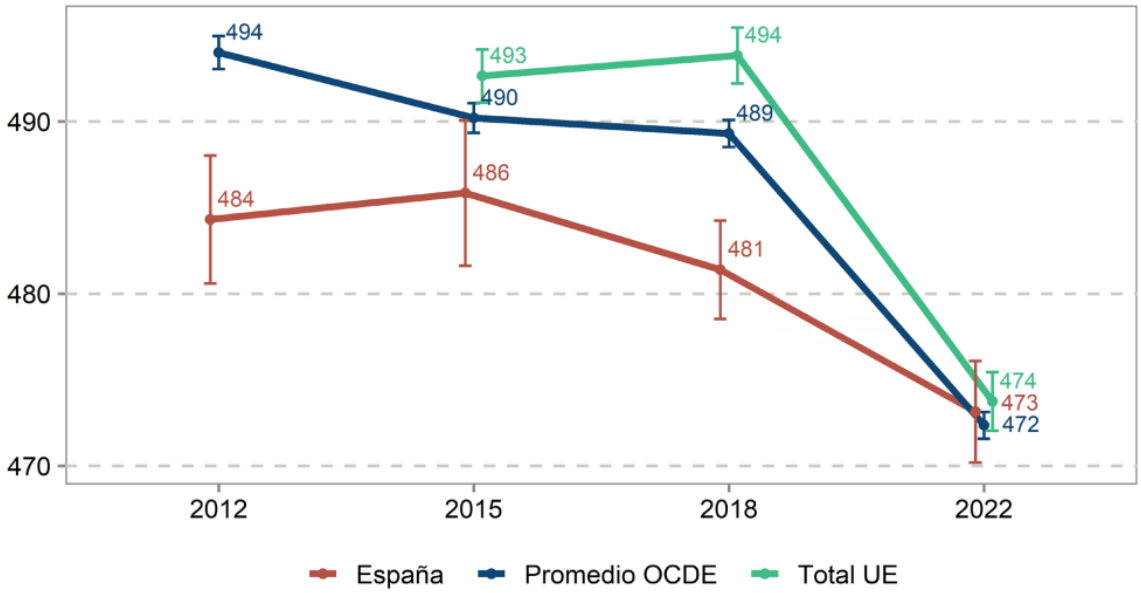

Figura 3.2: Evolución de las puntuaciones de matemáticas

- La carencia de información sobre la teoría de retículos enfocado al estudiante pre-universitario y universitario, un tema de gran relevancia y creciente interés

Con este fin, se ha desarrollado una actividad utilizando GeoGebra, la cual está disponible para su consulta en el enlace proporcionado en (17). Esta actividad incluye tanto preguntas de respuesta breve como de opción múltiple, junto con ejercicios dinámicos diseñados para fortalecer la comprensión de los conceptos. Dentro de los conceptos trabajados se incluyen:

- Operaciones con vectores
- Producto escalar
- Ortogonalidad
- Sistema de referencias afines y cartesianos
- Distancia taxi

La propuesta busca que los docentes desarrollen actividades matemáticas que demuestren aplicaciones prácticas del temario enseñado en clase, como la criptografía post-cuántica. Además, pretende que a través de esta última, refuercen los conceptos impartidos en el aula de forma interactiva y empírica.

Capítulo 4

Conclusiones y líneas futuras

En este Trabajo Fin de Máster se ha explorado en profundidad el esquema de firma CRYSTALS-Dilithium, aportando un novedoso ataque teórico por backdoor empleando técnicas de cleptografía. Por otro lado, se ha trabajado y se ha estudiado en la compleja función Transformada Teórica de Números, NTT. Se ha dedicado un artículo al estudio de la NTT, arrojando luz sobre sus complejos procedimientos y aportando una implementación didáctica y eficiente. Para finalizar, se han desarrollado actividades didácticas totalmente innovadoras relativas a las matemáticas subyacentes de los nuevos estándares post-cuánticos basados en retículos.

Como trabajos futuros se plantea la opción de seguir mejorando el ataque cleptográfico, presentando además de forma generalizada. También se considera la opción de continuar creando material didáctico que ayude a comprender las matemáticas relacionadas con la criptografía post-cuántica, tanto en Secundaria, como en Bachillerato y como en entornos universitarios.

Finalmente, este Trabajo Fin de Máster, no solo por su contenido, es totalmente novedoso por haber sido presentado en forma de compendio por publicaciones, el cuál sin duda representa un avance en materia de criptografía post-cuántica.

Capítulo 5

Conclusions and future lines

This work has been a continuation of our own Final Degree Project carried out in 2023, (7). Throughout this Master's Thesis, the CRYSTALS-Dilithium signature scheme has been extensively explored, introducing a novel theoretical backdoor attack using kleptographic techniques. Additionally, significant work has been dedicated to studying the complex Number Theoretic Transform function, NTT. An article has been devoted to the study of NTT, shedding light on its intricate procedures and providing a didactic and efficient implementation. Finally, entirely innovative didactic activities related to the underlying mathematics of the new post-quantum lattice-based standards have been developed.

As future work, the option to further improve the kleptographic attack, presenting it in a more generalized form, is considered. The possibility of continuing to create didactic material to aid in understanding the mathematics related to post-quantum cryptography is also under consideration, targeting Secondary Education, Baccalaureate, and university environments.

In conclusion, this Master's Thesis, not only due to its content but also due to its presentation in the form of a compendium of publications, undoubtedly represents an advancement in post-quantum cryptography.

Capítulo 6

Bibliografía

[1] NIST, “NIST Announces First Four Quantum-Resistant Cryptographic Algorithms,"*NIST News*, July 2022. [Online]. Disponible en: https://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms

[2] NIST: “NIST: an official website of the US government", *NIST News*, August 2023. [Online], Disponible en: https://www.nist.gov/news-events/news

[3] Avanzi, R., Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J., Schwabe, P., Seiler, G., Stehlé, D., “CRYSTALS-Kyber algorithm specifications and supporting documentation”, NIST PQC Round, 2019. [Online], Disponible en: https://pq-crystals.org/kyber/data/kyber-specification-round2.pdf

[4] Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., Stehlé, D., “CRYSTALS-Dilithium, Algorithm Specifications and Supporting Documentation (Version 3.1)", 2022. [Online], Disponible en: https://pq-CRYSTALS.org/dilithium/data/dilithium-specification-round3-20210208.pdf

[5] Sailada, S., Vohra, N., Subramanian, N., “Crystal Dilithium Algorithm For Post Quantum Cryptography: Experimentantion and Usecase for eSign", en *First International Conference on Electrical, Electronics, Information and Communication Technologies*, 2022

[6] Scarone-Etchamendi, B., “Criptografía Post Cuántica basada en Reticulados", Trabajo fin de grado, Universidad de la República de Uruguay, 2018

[7] Pérez-Ramos, É., “Estudio e Implementación del esquema de firma CRYSTALS-Dilithium”, Trabajo fin de grado, Universidad de la Laguna, 2023. [Online], Disponible en: https://riull.ull.es/xmlui/handle/915/36280

[8] Merino-González, L. M., y Santos-Aláez, E. “Algebra Lineal Con Métodos Elementales", Madrid: Thomson Paraninfo, 2006

[9] Águila-Hernández, E.J., “Algunos Tópicos en Teoría de Aproximación”, Trabajo fin de grado, Universidad de La Laguna, 2023. [Online], Disponible en: https://riull.ull.es/xmlui/handle/915/33370

[10] Chi, D.P., Choi, J. W., Kim, J. S. y Kim, T.,“Lattice Based Cryptography for Beginners,” Cryptology ePrint Archive, Paper 2015/938, 2015. [Online], Disponible en: https://eprint.iacr.org/2015/938

[11] Zahid, A. Z., “Lattices, Cryptography, and NTRU", Trabajo fin de grado, St. Mary’s College of California, 2017

[12] Harrigan, S., “Lattice-Based Cryptography and the Learning with Errors Problem”, 2017. [Online], Disponible en: https://mysite.science.uottawa.ca/mnevins/papers/StephenHarrigan2017LWE.pdf

[13] Miguel-Salgado, A., “Criptografía Postcuántica", Trabajo fin de grado, Universidad del País Vasco, 2021

[14] Ravi, P., Bhasin, S., Chattopadhyay, A., Roy, S., “Backdooring Post-Quantum Cryptography: Kleptographic Attacks on Lattice-based KEMs”, Cryptology ePrint Archive, 2022

[15] Young, A., Yung, M., “Kleptography: Using cryptography against cryptography”, Advances in Cryptology—EUROCRYPT’97: International Conference on the Theory and Application of Cryptographic Techniques, pp. 62–74, Springer, 1997

[16] Instituto Nacional de Evaluación Educativa. “PISA 2022. Programa para la Evaluación Internacional de los Estudiantes. Informe español.",2023. [Online], Disponible en: https://www.libreria.educacion.gob.es/search/?q=PISA+2022

[17] Pérez-Ramos, É. “Una breve introducción a la teoría de retículos", 2024. Disponible en: https://www.geogebra.org/m/cm2e42fk

Apéndice A

Apéndices

A.1. Artículo aceptado “Theoretical Approach to Backdoor Attacks on the Template of CRYSTALS-Dilithium”

Édgar Pérez-Ramos, Pino Caballero-Gil
“Theoretical Approach to Backdoor Attacks on the Template of CRYSTALS-Dilithium”
IEEE International Conference on Communications ICC
Denver, CO, USA. 9-13 June, 2024
Indexado en GII-GRIN con GGS Rating Class 2 A+
Indexado en LiveSHINE: A+

Theoretical Approach to Backdoor Attacks on the Template of CRYSTALS-Dilithium

Édgar Pérez-Ramos
Department of Computer Engineering and Systems
University of La Laguna
Tenerife, Spain
alu0101207667@ull.edu.es

Pino Caballero-Gil
Department of Computer Engineering and Systems
University of La Laguna
Tenerife, Spain
pcaballe@ull.edu.es

***Abstract*—Anticipating the potential of quantum processors to compromise the security of widely employed cryptographic algorithms, there is a current need for extensive research in post-quantum cryptography across all communication technologies, including 6G. Therefore, post-quantum cryptography has recently witnessed the emergence of new and prominent representatives, such as the new standard for quantum-safe digital signatures, called CRYSTALS-Dilithium. This work presents a theoretical approach to two backdoor attacks on that algorithm, in which the objective is to extract the message through the signature. Although the hypotheses of this work are very theoretical, since they assume that the message is encrypted and that the attacker possesses the private key, given the enormous relevance of the new standard post-quantum encryption, this study is of great interest. In particular, it relies on installing a backdoor in the signing process to later be able to extract the message through the signature. This document provides several details of the attacks, thus demonstrating the feasibility of leaking the original message during the signing phase.**

***Index Terms*—Post-Quantum Cryptography, Kleptography, Backdoor Attack, CRYSTALS-Dilithium, 6G**

I. Introduction

In current and future wireless networks, digital signatures play a crucial role in security and authentication systems as they enable the verification of both authenticity and integrity of digital documents, ensuring they originate from a trusted origin and have not undergone alterations. However, with the emergence and threat of quantum computing, cryptographic algorithms employed in traditional electronic signature methods, such as RSA signature and Elliptic Curve Digital Signature Algorithm, prove to be vulnerable. Therefore, it is imperative to replace these schemes with new digital signatures that are resistant to quantum computing.

The National Institute of Standards and Technology (NIST) has recently dedicated several years of effort to the pursuit of standardizing algorithms capable of withstanding the challenges posed by quantum computing. In 2022, the four finalists of this comprehensive process were unveiled, featuring prominent algorithms such as CRYSTALS-Kyber, [1], for encryption, and CRYSTALS-Dilithium, [2], for digital signatures.

Since then, numerous efforts have been dedicated to verifying the strength of these schemes, given the importance they will take on in the coming years. In particular, NIST has recently developed the corresponding drafts of Federal Information Processing Standards (FIPS), FIPS 203 [3] and FIPS 204 [4], which specify the Module-Lattice Key-Encapsulation Mechanism (ML-KEM) and the Module-Lattice Digital Signature Algorithm (ML-DSA), derived from CRYSTALS-Kyber and CRYSTALS-Dilithium, respectively.

This work has been inspired by [5]. However, there are substantial differences, since in [5] the secret key is exfiltrated through the public key, whereas in this work, the hidden message is decoded through the signature. Thus, in line with the research line of [5] on CRYSTALS-Kyber, the aim of this work is to establish a connection between backdoor usage, kleptography and CRYSTALS-Dilithium. Surprisingly, despite the abundance of research related to kleptographic attacks on various public key cryptography systems like RSA and elliptic curve cryptography, there appears to be limited information regarding such attacks against digital signature schemes. Therefore, this study aims to provide new insights to address this knowledge gap.

This work is structured as follows. Section II provides essential background information for a better understanding of the proposed attacks. Section III gives an overview of the two attacks. Section IV includes an explanation of the implementations, accompanied by their results and experiments. Section V closes the paper with some conclusions and open questions.

II. Background

Before introducing the basis for the proposed attacks, some preliminary concepts and terms are introduced below.

A backdoor is a hidden or secret point that allows remote users access to a victim system. It can reach a victim system through various means: they can be pre-installed in the system, downloaded via infected files and applications with phishing, or exploited by cybercriminals through system vulnerabilities.

This research has been possible thanks to the agreement between Atlantis Technology and the University of La Laguna, the project PID2022-138933OB-I00: ATQUE funded by MCIN/AEI/10.13039/501100011033/FEDER, EU, and the Cybersecurity Chairs of the University of La Laguna financed by Binter SA, and by the NextGenerationEU within the framework of the Recovery and Resilience Facility.

Kleptography is a branch of cryptography that focuses on studying methods and techniques to steal secret information or encryption keys from a cryptographic system without detection by the legitimate user or system owner. This term was proposed by Adam Young and Moti Yung in 1996 [6].

Unlike conventional cryptanalysis, where the goal is to break a cryptographic system, kleptography aims to subvert the system to gain unauthorized access to confidential information. Some examples of kleptographic attacks may include manipulating cryptographic hardware or software to unintentionally leak secret information, introducing malicious components into the cryptographic system, or using interference signals to reveal secret information. Kleptography can be said to explore how adversaries can potentially implant covert backdoors to compromise cryptographic systems.

According to [6], a kleptographic attack typically contains in its core an efficient algorithm called Secretly Embedded Trapdoor with Universal Protection (SETUP). This algorithm can be integrated within a cryptosystem to covertly leak secret information through the output of the cryptosystem.

Several researchers have demonstrated the theoretical feasibility and potential dangers of kleptographic attacks against certain digital signatures, such as those based on discrete logarithm [7]. Regarding CRYSTALS-Dilithium, to the best of our knowledge, the only bibliographic reference to a similar attack is [8], which is a survey on attack approaches based on subliminal channels against different NIST post-quantum signature schemes.

For the next definitions, the following notation is used. Let $q, m, n, k \in \mathbb{Z}$. The ring of integers modulo a prime q is denoted $\mathbb{Z}_q$. The set of n-vectors over $\mathbb{Z}_q$ is denoted $\mathbb{Z}_q^n$. If the polynomial ring $\mathbb{Z}_q(x)/\phi(x)$ over $\mathbb{Z}_q$ with reduction polynomial $\phi(x)$ is denoted $\mathcal{R}_q$, let $r \in \mathcal{R}_q^k$ be a *module* of dimension k.

Definition 1: The Learning With Errors (LWE) is a search problem that consists of solving a system of m equations with errors, by finding $s \in \mathbb{Z}_q^n$ such that $A \cdot s + e = b \ mod \ q$, where $A = (a_1, ..., a_m)$ is a $m \times n$ matrix with $a_i \in \mathbb{Z}_q^n$ $\forall i \in \{1, ..., m\}$, and e, $b \in \mathbb{Z}_q^n$.
Thus, the definition of the LWE problem consists of hiding the value of a secret vector s through the introduction of an error vector e.

Definition 2: The Module-Learning With Errors (MLWE) is a search problem that consists of solving a system of m equations with errors, by finding $s \in \mathcal{R}_q^n$ such that $A \cdot s + e = b$, where $A \in \mathcal{R}_q^{m \times n}$ and e, $b \in \mathcal{R}_q^m$.

Thus, the formulation of the MLWE is similar to that of LWE, where the ring of integers $\mathbb{Z}$ is replaced by the polynomial ring $\mathcal{R}_q = \mathbb{Z}_q(x)/\phi(x)$ with reduction polynomial $\phi(x) = x^n + 1$. In this way, all variables used in relation with MLWE are polynomials, and the MLWE can be seen as a search problem defined on many LWE instances in the polynomial ring. This involves introducing as a new parameter the degree n of the polynomial. In particular, typical parameters in the MLWE, used hereinafter in this document, as stated in [2], are $q = 2^{23} - 2^{13} + 1$ and $n = 256$.

The post-quantum signature scheme called CRYSTALS-Dilithium derives its security from the hardness of the MLWE problem. It is composed of three algorithms, whose templates described in [2]

In the template, the Gen algorithm first generates the first part of the public key, which is a matrix A. Then, it samples random secret key vectors s_1 and s_2, which are the private key. Then, the second part of the public key is computed with $t = As_1 + s_2$.

The Sign algorithm first generates a vector of polynomials y. Then, the signer computes Ay and sets w_1 to be the "high-order" bits of the coefficients in this vector, and the challenge c is created as the hash of the message and w_1. After that, a potential signature is computed as $z = y + cs_1$, which is modified to prevent the secret key from being leaked from it.

The Verify algorithm first computes w_1' to be the high-order bits of $Az - ct$. Then, it accepts the signature if all the coefficients of z are low enough and c is the hash of the message and w_1'.

For a more in-depth understanding of Dilithium digital signature, readers may refer to [2].

III. Backdoor Attacks on Dilithium Template

The proposed attacks against CRYSTALS-Dilithium are based on a black-box cryptographic module that introduces a backdoor into the Sign algorithm in order to obtain information about the message. As in [5], the attacker only requires access to the public outputs of the algorithm, with no other type of interaction or access required. Another assumption is that the attacker knows both the user's public key pk and the user's secret key sk.

In the two proposed attacks, the attacker manages to discover the original binary message M from the output of the signature scheme $Sign(sk, M) = \sigma(z, c)$. An example of application of this attack could be the following. Suppose that, during the signature creation process, the attacker installs a backdoor. Then, it is assumed that the attacked user sends the encrypted signed document or message through one channel and the digital signature through another channel. Then, the attacker proceeds by intercepting the signature, and managing to extract the document from it.

It is important to clarify that there is some variation regarding the definition of kleptography as outlined in [6]. In the First Attack, there is a non-zero probability of detection since the vector c is modified, and its integrity is checked during the signature verification phase. However, in the Second Attack, by altering z and successfully passing security tests while keeping the vector c unchanged, there is a higher likelihood of effectiveness with a proper utilization of the backdoor.

First Attack

This attack consists of hiding the bit-encoded message M within the second part of the signature c. Hereinafter, M encoded in bits is denoted as M_{bits}.

Note that according to [2], the second part of the signature $c \in \mathcal{B}_\tau$, that is, there are exactly τ elements that are 1s or -1s, while the rest are equal to zero. Thus, for the attack to be successful, it is necessary to guarantee that $\alpha = \tau$, for which the following three cases are distinguished:

1) $\alpha = \tau$. In this case, the attacker does not have to change anything. For the attack, a few random 1s are selected from M_{bits}, which are replaced with -1s to hide M in c without raising suspicion.
2) $\alpha < \tau$. In this scenario, since the number of 1s in M_{bits} is less than τ, exactly $\beta = (\tau - \alpha)$ -1s are generated for the attack, which are added to the vector c.
3) $\alpha > \tau$. In this case, given the excess of 1s in M_{bits} with respect to τ, it is necessary for the attack to use some compression technique, here called ReduceBits function, in order to reduce the amount of 1s in M_{bits}.

The message extraction process could be carried out in the two phases of signature creation or signature verification. However, this paper is focused on the attack during the signature creation phase because the backdoor is assumed to be installed in this step, just before computing z. In particular, the attack algorithm is presented in detail in Algorithm 1.

Algorithm 1 First Attack on Dilithium Template

Require: sk, M, τ
Ensure: $\sigma = (z, c)$
$z := \perp$
while $z = \perp$ **do**
  $y \leftarrow \mathcal{S}^l_{\gamma_1 - 1}$
  $w_1 := \mathbf{HighBits}(Ay, 2\gamma_2)$
  $c \in \mathcal{B}_\tau := \mathbf{ConvertToBits}(M, \tau)$ ▷ Here is the backdoor
  $z := y + cs_1$
  if $\| z \|_\infty \geq \gamma_1 - \beta$ or $\| \mathbf{LowBits}(Ay - cs_2, 2\gamma_2) \|_\infty \geq \gamma_2 - \beta$ **then**
    $z := \perp$
  end if
end while
return $\sigma = (z, c)$

If compared with Dilithium template [2], it can be seen that the main difference between it and Algorithm 1 is found in the new ConvertToBits function shown in Algorithm 2, where the aforementioned substitution of vector c takes place.

An interesting observation is that $c = M_{bits}$ is calculated before z. In other words, c is manipulated and then, implicitly, z is also manipulated, as $z = y + c \cdot s_1$. Therefore, with this backdoor technique, not only M is being encoded in the second part of the signature c but also in its first part z. This implies that z also depends on M. An intriguing line for future research could focus on studying the exfiltration through z of M, instead of c, as shown in Algorithm 6.

Algorithm 2 ConvertToBits

Require: M, τ
Ensure: c
$M_{bits} = [m_0, m_1, \cdots, m_b]$ ▷ message M is turned into bits
$\rho \in \{0,1\}^b$ ▷ A seed of the same length as M_{bits} is chosen
$M_\rho = M_{bits} \oplus \rho$
$\alpha :=$ The number of 1s in M_ρ
if $\tau = \alpha$ **then**
  $c = \mathbf{ChangeSign}(\mathbf{M}_\rho)$ ▷ some 1s are changed to -1s randomly
else if $\tau > \alpha$ **then**
  $\beta := \tau - \alpha$
  $c = \mathbf{IntroduceBits}(\beta, \mathbf{M}_\rho)$ ▷ β -1s is added to M_ρ until $\alpha = \tau$
else if $\tau < \alpha$ **then**
  while $\tau < \alpha$ **do**
    $M_\rho = \mathbf{ReduceBits}(M_\rho)$
    $c = \mathbf{ChangeSign}(M_\rho)$
  end while
end if
return c

Algorithm 3 ChangeSign

Require: M_ρ
Ensure: M'_ρ
$\alpha :=$ The number of 1s in M_ρ
$\theta := len(M_\rho)$
Initialize $M'_\rho = m_0 m_1 \ldots m_\theta = 000 \ldots 0$
$s := 0$
for $i := 0$ to $\theta - 1$ **do**
  if $M_\rho[i] = 0$ **then**
    $M'_\rho[i] = 0$
  else
    if s is even **then**
      $M'_\rho[i] = -1$
    end if
    s = s + 1
  end if
end for
return M'_ρ

The following observations are related to the three distinguished cases mentioned above:

1) For the effective concealment of M bits within c, a strategy of perfect secrecy is utilized. Initially, a random seed

Algorithm 4 IntroduceBits

Require: $M_\rho,\ \beta$
Ensure: M'_ρ
$M'_\rho := M_\rho$
$\theta := len(M_\rho)$
$s := 0$
for $i := 0$ to $\theta - 1$ **do**
  if $s < \beta$ **then**
    if $M'_\rho[i] = 0$ **then**
      $M'_\rho[i] = -1$
      s = s+1 ▷ s is used as a counter to keep track of the entered bits
    end if
  end if
end for
return M'_ρ

Algorithm 5 ReduceBits

Require: M_ρ, τ
$\alpha :=$ The number of 1s in M_ρ
$\theta := len(M_\rho)$
$\beta := \alpha - \tau$ ▷ The number of 1s to eliminate
$M'_\rho,\ v, v_\beta := \perp$
for $i := 0$ to θ **do**:
  if $M_\rho[i] = 1$ **then**
    $v.append(i)$ ▷ the 1s positions are stored
  end if
end for
$v_\beta = sample(v, \beta)$ ▷ β random positions are chosen
for $i := 0$ to θ **do**:
  if $i \in v_\beta$ **then**:
    $M'_\rho.append(0)$ ▷ Introduce a 0 to remove the 1
  else
    $M'_\rho.append(M_\rho[i])$
  end if
end for
return $\phi = (M'_\rho, v_\beta)$

ρ matching the length of $M_{bits}$is generated. Following this, a bit-by-bit XOR operation bit by bit and perform the subsequent operations with the resulting new vector M_ρ. During the recovery phase, the seed is added to the exfiltrated vector to retrieve the original M_{bits}.

2) With the the objective of hiding M_{bits} within c, and since α must coincide with τ, in some cases a random selection of some 1s is made to be replaced by -1s through the Algorithm 3 in order to minimize user detection.
3) In those cases where β 1s are missing, they are introduced randomly using the IntroduceBits function shown in Algorithm 4. In this way, the required amount is reached, and the sign of those entered is also changed to prevent the user from noticing the change.
4) The ReduceBits function is used in the scenario where $\tau < \alpha$. In this case the number of 1s in the encoded message M has to be reduced so that it is possible to revert to the previous situation. In fact, an efficient method is necessary to properly extract the message when there is an excess of 1s. Once the required number of 1s has been reduced, the scenario is that of $\tau = \alpha$. The v_β vector is stored by the attacker and is not shown to the user, in order to use it for message retrieval.

Note that in the process of recovering message M through the Algorithm 6, two options are possible:

1) $\tau = \alpha$
2) $\tau > \alpha$. The case where $\alpha > \tau$ is not considered, as demonstrated in Algorithm 2, as it ultimately reduces to the scenario where $\alpha = \tau$.

Algorithm 6 Recovering the message M

Require: $M'_\rho,\ \tau,\ v_\beta,\ \rho$
Ensure: M_1, M_2, M_3
$M_1 = |M'_\rho|$ ▷ Case when $\alpha = \tau$
$M_2 = M'_\rho \setminus U$ ▷ Case when $\alpha < \tau$. $U = \bigcup_{i=1}^{\beta} \{-1s\}$
$M_3 = |M'_\rho| + Rebuild(v_\beta)$ ▷ Case when $\alpha > \tau$
$M^i_{bits} = M_i \oplus \rho$ ▷ M_{bits} is recovered by applying the XOR operation
return M^i_ρ

Second Attack

This attack consists of hiding the message encoded in M bits within the first part of the z signature. The main concept of this attack is to hide M_{bits} in z, using a seed ρ and an XOR operation and randomly assigning 0 bits to even (or odd) numbers and 1 bits to odd (or even) numbers.

The form of z is:

$$z = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_k \end{pmatrix}, \quad p_i \in \mathcal{R}_q, \quad \forall i \in \{1, 2, \ldots, k\} \tag{1}$$

Indeed, $256 \cdot k$ coefficients $p_{i,j} \in \mathbb{Z}_q$ exist such that:

$$z = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,256} \\ p_{2,1} & \alpha_{2,2} & \cdots & p_{2,256} \\ \vdots & \vdots & \ddots & \vdots \\ p_{k,1} & p_{k,2} & \cdots & p_{k,256} \end{pmatrix} \cdot (X^k)^t, \quad X = \begin{pmatrix} 1 \\ x \\ x^2 \\ \vdots \\ x^{255} \end{pmatrix} \tag{2}$$

In order to establish a bidirectional relationship that enables both message encoding and retrieval, it is essential to lay the

mathematical foundations. To do so, the following equivalence relation is defined. Set $x, y \in \mathbb{Z}$, then:

$$x \sim_A y \iff x \wedge y \text{ are even} \tag{3}$$

Then, the quotient set is:

$$A = \mathbb{Z} \backslash \sim_A = \{[x], [y] : x \text{ odd and } y \text{ even (or vice versa)}\} \tag{4}$$

The equality $|A| = 2$ is evident. Additionally, considering $M_{bit} = \{0, 1\}$ leads to $|M_{bit}| = 2$. Subsequently, a bijective mapping is established as follows:

$$\begin{aligned} \varphi \colon M_{bit} &\longrightarrow A \\ x &\longmapsto a, \\ \varphi(0) = [0], \quad \varphi(1) &= [1] \end{aligned}$$

This establishes a bijective function facilitating both encoding and retrieval of M_{bits}. Subsequently, the algorithm forming the basis of the proposed attack is presented, situated in the signature creation phase.

Observations from Algorithm 8 (the first three) and Algorithm 9 (the last two) can be discussed as follows:

1) Initial assignment of a value from the set $\{0, 1\}$ to the variable x introduces randomness to the algorithm, assigning even numbers if $x = 0$ and odd numbers if $x = 1$, thereby reducing user detectability.
2) In this step, the seed ρ is generated and undergoes the XOR operation with M_{bits}. This process yields M_ρ, establishing a secure foundation for subsequent operations. Like v_β, ρ is stored for the attacker and is not displayed to the user, so that the message can be retrieved later.
3) During the intermediate phase of the algorithm, a detailed analysis of the length of M_ρ is performed. If the value of $\theta \leq 255 \cdot k - 1$, the $Parity$ function is applied to evaluate whether noise should be introduced to meet the condition $\theta = 255 \cdot k - 1$. On the other hand, if $\theta \geq 255 \cdot k$, M_ρ is divided into two parts, M_1 and M_2. For the first one, the $Parity$ function is applied, while for M_2, a procedure similar to the one explained in the previous attack, based on $ReduceBits$, is employed.
4) Ultimately, since z consists of $256 \cdot k$ elements, it is transformed into a $k \times 1$ matrix, where each element is represented by $p_i \in \mathcal{R}_q$, thus, $z \in \mathcal{R}_q^k$.
5) Within the *for* loop, when $M_\rho[i] = 1$, an even number within the range $[0, q)$ is assigned. By adding the value of x to it yields a number that can be either even or odd. On the other hand, if $M_\rho[i] = 0$, an odd number is sampled, and once again, the variable x is added to obtain the final outcome.
6) If $\theta < 255 \cdot k - 1$, noise is introduced. Specifically, starting from position $z[\theta]$, random elements are selected from $\mathbb{Z}_q$. This approach allows us to accurately determine the stopping point in the decoding of z while introducing a random component to make it less perceptible to the user.

Algorithm 7 Second Attack on Dilithium Template

Require: sk, M_{bits}, τ
Ensure: $\sigma = (z, c)$
$z := \perp$
while $z = \perp$ **do**
  $y \leftarrow \mathcal{S}^l_{\gamma_1 - 1}$
  $w_1 := \mathbf{HighBits}(Ay, 2\gamma_2)$
  $c \in \mathcal{B}_\tau := \mathbf{H}(M, w_1)$
  $z := \mathbf{Parity}(M_{bits})$ ▷ Here is the backdoor
  if $\| \ z \ \|_\infty \geq \ \gamma_1 - \beta$ or $\| \ \mathbf{LowBits}(Ay - cs_2, 2\gamma_2) \ \|_\infty \geq \gamma_2 - \beta$ **then**
    $z := \perp$
  end if
end while
return $\sigma = (z, c)$

Algorithm 8 Parity

Require: M_{bits}
Ensure: z, x, θ, ρ
$x \leftarrow \{0, 1\}$ ▷ Randomly choose between assigning zero to evens or odds
$\rho \leftarrow \{0, 1\}^\theta$ ▷ Where $\theta =:= len(M_{bits})$
$M_\rho = M_{bits} \oplus \rho$
if $\theta \leq 256 \cdot k$ **then**
  ParityKernel(M_ρ, θ, x)
else
  Break$(M_\rho) = (M_1, M_2)$
  $z_1, z_2 =$ **ParityKernel**(M_1, θ_{M_1}, x), **StoreIndex**(M_2)
  $z = z_1$
end if
Rebuild(z) ▷ z is restructured as a matrix
return z, x, θ, ρ

Algorithm 9 ParityKernel

Require: M_ρ, θ, x
Ensure: z
$z := \{0\}^{256 \cdot k}$ ▷ Initialize z to zero
for $i := 0$ to $(\theta - 1)$ **do**:
  if $M_\rho[i] = 1$ **then**
    $z[i] = sample(2\mathbb{Z}_q) + x$
  else
    $z[i] = sample(\mathbb{Z}_q \setminus 2\mathbb{Z}_q) + x$
  end if
end for
if $\theta < 256 \cdot k$ **then**
  for $i := \theta$ to $(255 \cdot k - 1)$ **do**:
    $z[i] = sample(\mathbb{Z}_q)$ ▷ Introduce some noise
  end for
end if
Rebuild(z) ▷ z is restructured as a matrix
return z

IV. Implementations

Preliminary implementations of versions of the aforementioned attacks can be found in [9] and [10]. In order to carry out implementations from scratch of the First and the Second Attack described above, Python programming language and the Google Colaboratory environment were used. During the implementation, firstly it was necessary to import various libraries. Furthermore, several auxiliary functions were implemented with different objectives.

Tables I and II gather the results of the experiments, where the effectiveness of ConvertToBits and Parity functions was tested hundreds of thousands of times. Furthermore, certain values were varied to diversify the experimentation, such as the size of τ, the word length and the value of k.

TABLE I
SUMMARY OF EXPERIMENTS FOR THE FIRST ATTACK.

Experiments of the First Attack			
Iterations	M **size**	τ **size**	**Success**
100000	16	60	100%
100000	50	60	100%
100000	$[0, 1000]$	$[1, 100]$	100%
100000	$[0, 1000]$	60	100%

TABLE II
SUMMARY OF EXPERIMENTS FOR THE SECOND ATTACK.

Experiments of the Second Attack			
Iterations	M **size**	k **size**	**Success**
100000	30	8	100%
100000	1000	8	100%
100000	$[0, 100]$	8	100%
200000	50	$[1, 20]$	100%

As can be seen in Table I, the four experiments carried out were 100% successful. In particular, in the fourth experiment of the First Attack, 10^5 iterations were conducted, divided into 100 batches of 1000 each. This analysis involved assessing the code's efficacy by varying the size of τ within the range $[0, 100]$ and the length of the random string within the range $[0, 1000]$. The absence of errors found throughout these test runs is notable.

Also as can be seen in Table II, the four experiments carried out were 100% successful. In particular, in the fourth experiment of the Second Attack, an examination of the range $[1, 20]$ for the size of k was done. This involved executing $2{\cdot}10^5$ iterations, random strings sized at 50. The results indicate the successful completion of the experiment.

In particular, the performed implementation focused on encoding the signature through the aforementioned functions and subsequently decoding it. Note that these processes were successfully carried out in all cases, as can be verified in Table I and Table II. This confirms the feasibility and effectiveness of our approach in a controlled experimental environment.

It is important to note that this study represents a theoretical attack, where the ability to manipulate and decode signatures using the identified vulnerabilities has been demonstrated. However, it is necessary to clarify that implementation in a real-world environment is currently a pending task, which would be a crucial step in assessing the true effectiveness and practical viability of the proposed approach.

V. Conclusions and Future Work

This work presents a theoretical study of two kleptographic backdoor attacks against the post-quantum digital signature template CRYSTALS-Dilithium. In particular, two algorithms have been designed to extract the message M from the signature $\sigma = (z, c)$. One attack is via z while the other is via c, both based on prior knowledge of the user's private key and the attacker's intervention during the signature creation phase. In order to carry out several experiments to analyse the functionality of the proposed attacks, didactic implementations have been carried out in Python, which have shown the success of both attacks. As part of the ongoing work, a third attack is being developed using dynamic intervals instead of using parity, analysing the scheme in scenarios where only one of the keys is known, and conducting experiments to evaluate the effectiveness of M message recovery in different cases where the signature is not involved. Also planned as future work is the study of the computational cost of the attack, its implementation in a different environment, different options for error correction codes, and possible defenses.

References

[1] R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. Schanck, P. Schwabe, G. Seiler, D. Stehlé, "CRYSTALS-Kyber algorithm specifications and supporting documentation", NIST PQC Round, vol. 2, no. 4, pp. 1–43, 2019.

[2] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, D. Stehlé, "CRYSTALS-Dilithium: Algorithm specifications and supporting documentation (version 3.1)", NIST Post-Quantum Cryptography Standardization Round, vol. 3, 2021.

[3] National Institute of Standards and Technology (NIST), "FIPS 203 (Draft) Module-Lattice-based Key-Encapsulation Mechanism Standard", 2023, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.203.ipd.pdf

[4] National Institute of Standards and Technology (NIST), "FIPS 204 (Draft) Module-Lattice-Based Digital Signature Standard", 2023,https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.204.ipd.pdf

[5] P. Ravi, S. Bhasin, A. Chattopadhyay, S. Roy, "Backdooring Post-Quantum Cryptography: Kleptographic Attacks on Lattice-based KEMs", Cryptology ePrint Archive, 2022.

[6] A.Young, M.Yung, "Kleptography: Using cryptography against cryptography", Advances in Cryptology—EUROCRYPT'97: International Conference on the Theory and Application of Cryptographic Techniques, pp. 62–74, Springer, 1997.

[7] G. Teşeleanu, "Threshold kleptographic attacks on discrete logarithm based signatures", International Conference on Cryptology and Information Security in Latin America, pp. 401–414, Springer, 2017.

[8] H. Galteland, K. Gjøsteen, "Subliminal channels in post-quantum digital signature schemes", Cryptology ePrint Archive, 2019.

[9] É. Pérez-Ramos, Bits.ipynb: Implementation of the first kleptographic attack on CRYSTALS-Dilithium, Google Colaboratory. Available at: https://rb.gy/59hk59, 2023.

[10] É. Pérez-Ramos, Parity.ipynb: Implementation of the second kleptographic attack on CRYSTALS-Dilithium, Google Colaboratory. Available at: https://rb.gy/yfjo3r, 2023.

A.2. Artículo presentado “Theoretical Backdoor Attack on CRYSTALS-Diltihium”

Édgar Pérez-Ramos, Pino Caballero-Gil
“Theoretical Backdoor Attack on CRYSTALS-Diltihium”
19th International Conference on Computer Aided Systems Theory (Eurocast)
Gran Canaria, ESP. 26 February- 1 March, 2024
Indexado en GII-GRIN con GGS Rating Class WiP

Theoretical Backdoor Attack on CRYSTALS-Dilithium

E. Pérez-Ramos[1 [0009−0008−0409−3079]] and P. Caballero-Gil[1 [0000−0002−0859−5876]]

Department of Computer Engineering and Systems. University of La Laguna. 38271 La Laguna. Tenerife. Spain
alu0101207667@ull.edu.es pcaballe@ull.edu.es

Extended abstract

Post-quantum cryptography has gained prominence in recent years as a response to the quantum threat. The National Institute of Standards and Technology (NIST) initiated an algorithm selection process in 2015 to address this issue. After a thorough search and evaluation, encryption algorithms such as CRYSTALS-Kyber and digital signature algorithms such as CRYSTALS-Dilithium, among others, have been selected.

CRYSTALS-Dilithium is a lattice-based algorithm, which follows the well-known Fiat-Shamir scheme. Specifically, its security is based on the difficulty of the problem of finding shortest vectors in lattices. A comprehensive overview of the algorithm can be found at [1].

This paper focuses on a theoretical study of a possible backdoor attack based on [2] that can be inserted in the signature creation process in CRYSTALS-Dilithium. The aim of this attack is to exploit the properties of lattices, modular algebra and the theory of the kleptography to extract the bit-encoded message or document. This scenario occurs in situations where the signature is sent over one channel and the signed message or document is transmitted over another channel.

The attack strategy proposed for CRYSTALS-Dilithium relies on a black-box cryptographic module, which introduces a backdoor into Dilithium's Sign procedure. This vulnerability allows the attacker to extract information about the message being processed. Similar to the approach presented in [2], the attacker can carry out this exploit with just access to the algorithm's public outputs, without necessitating any other forms of interaction or access. Furthermore, it is assumed that the attacker possesses knowledge of both the user's public key pk and the user's secret key sk. In particular, the attack is composed of the following two algorithms: Alg. 1 and Alg. 2.

In the first algorithm, Alg. 1, the function IntervalsKernel is used to assign each bit of M_{bits} to an element within the associated interval. Furthermore, if the length exceeds the available capacity, the StoreIndexes function is used to store the positions of the bits with value 1, allowing the message to be reconstructed later.

In conclusion, this paper introduces a theoretical backdoor attack on the post-quantum digital signature scheme CRYSTALS-Dilithium, assuming the attacker

Algorithm 1 Intervals

$x = \textbf{LightHouse}(M_{bits},\ q)$
if $\theta \leq 256 \cdot k$ **then**
    $\textbf{IntervalsKernel}(M_{bits}, \theta, x)$ ▷ We allocate the bits of M_{bits} according to the x indications.
else
    $\textbf{Break}(M_{bits}) = (M_1, M_2)$
    $z_1, z_2 = \textbf{IntervalsKernel}(M_1, \theta_{M_1}, x),\ \ \textbf{StoreIndexes}(M_2)$
    $z = (z_1, z_2)$
end if
$\textbf{Rebuild}(z)$ ▷ z is restructured as a matrix
return z, x, θ

Algorithm 2 LightHouse

$\alpha := \textbf{CountOnes}(M_{bits})$ ▷ We count the number of ones
$\theta := length(Mbits)$
$x := [\frac{\alpha}{\theta} \cdot q]$ ▷ In this way we calculate the proportion of 1s in M_{bits}
return x

knows the user's private and public keys. The attack focuses on the signature creation phase, particularly targeting z. Ongoing research includes exploring attack implementation and the impact on other variables when modifying z.

References

1. Bai, S., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., Stehlé, D.: "CRYSTALS-Dilithium, Algorithm Specifications and Supporting Documentation (Version 3.1)", NIST Post-Quantum Cryptography Standardization Round, vol.3, 2021.

2. Ravi, P., Bhasin, S., Chattopadhyay, A., Roy, S.S.: Backdooring Post-Quantum Cryptography: Kleptographic Attacks on Lattice-based KEMs. Cryptology ePrint Archive, 2022.

A.3. Artículo presentado “La Transformada Teórica de Números para Kyber”

Néstor Antuñano-Cabrera, Édgar Pérez-Ramos, Candelaria Hernández-Goya, Pino Caballero-Gil
“La Transformada Teórica de Números para Kyber”
IX Jornadas Nacionales de Investigación en Ciberseguridad (JNIC)
Sevilla, ESP. 27-29 Mayo, 2024

La Transformada Teórica de Números para Kyber

Néstor Antuñano-Cabrera
Universidad de La Laguna
Tenerife, España
alu0101440460@ull.edu.es

Édgar Pérez-Ramos
Universidad de La Laguna
Tenerife, España
alu0101207667@ull.edu.es

Candelaria Hernández-Goya
Universidad de La Laguna
Tenerife, España
mchgoya@ull.edu.es

Pino Caballero-Gil
Universidad de La Laguna
Tenerife, España
pcaballe@ull.edu.es

***Resumen*—La Transformada Teórica de Números es un método eficiente para la multiplicación de dos polinomios de grado alto, ampliamente usado para sistemas criptográficos basados en retículos como Kyber y Dilithium. Este documento se centra en el caso concreto de Kyber. Incluye una revisión de los conceptos básicos del álgebra de anillos y posteriormente una explicación sobre la convolución de polinomios usando la Transformada Teórica de Números. Además se introducen algunos algoritmos básicos como el radix-2 basado en los algoritmos de Cooley-Tukey y Gentleman-Sande. Finalmente se describe una implementación en Python de la Transformada Teórica de Números generalizada y la correspondiente convolución de polinomios en Kyber.**

***Index Terms*—Criptografía post-cuántica, CRYSTALS-Kyber, NTT**

Tipo de contribución: *Investigación en desarrollo*

I. Introducción

La criptografía de clave pública actual basada principalmente en la dificultad de factorizar primos grandes y la resolución de logaritmos discretos, posee vulnerabilidades frente a ataques cuánticos. Por ello, la criptografía post-cuántica es una alternativa fundamental para el futuro próximo.

En diciembre de 2016, el Instituto Nacional de Estándares y Tecnología (*National Institute of Standards and Technology*, NIST) inició un proceso para estandarizar algoritmos criptográficos resistentes a la computación cuántica, publicando una solicitud de propuestas de algoritmos de criptografía post-cuántica. Esta iniciativa fue un paso importante en el esfuerzo por desarrollar sistemas criptográficos que sean seguros frente a ordenadores cuánticos y clásicos, y que puedan interoperar con protocolos y redes de comunicaciones existentes.

El objetivo de la iniciativa era recibir propuestas de algoritmos que pudieran servir como estándares para firmas digitales y mecanismos de encapsulación de claves (*Key Encapsulation Mechanism*, KEM), que son fundamentales para la seguridad en la era post-cuántica. Para la fecha límite, en noviembre de 2017, el NIST había recibido 69 algoritmos elegibles. Posteriormente en agosto de 2023, NIST dio el siguiente paso para la estandarización de algoritmos criptográficos resistentes a ataques desarrollados con ordenadores cuánticos, publicando borradores de estándares para tres de los cuatro algoritmos seleccionados en 2022 [1]:

- FIPS 203: CRYSTALS-Kyber. Es un KEM basado en el problema de aprendizaje con errores (*Learning With Errors*, LWE) sobre anillos [2].
- FIPS 204: CRYSTALS-Dilithium [3].
- FIPS 205: SPHINCS+ [4].

El NIST invitó a la comunidad criptográfica mundial a proporcionar comentarios sobre estos borradores hasta noviembre de 2023, con el fin de recibir retroalimentación y asegurarse de que los estándares sean completos y no tengan omisiones antes de su finalización.

Por otra parte, la Transformada Teórica de Números (*Number Theoretic Transform*, NTT) es esencial para el desarrollo tanto de CRYSTALS-Kyber, como de CRYSTALS-Dilithium porque permite realizar multiplicaciones de polinomios de manera eficiente, una operación clave en criptosistemas basados en retículos como Kyber. La NTT es una variante de la Transformada Rápida de Fourier (*Fast Fourier Transform*, FFT) adaptada para trabajar en un anillo de números enteros módulo un número primo [5], [6].

La eficiencia de CRYSTALS-Kyber depende en gran medida de la rapidez con la que se pueden realizar estas multiplicaciones polinómicas, y la NTT permite hacerlo de manera rápida y eficiente. Por tanto, la NTT es una herramienta crucial para optimizar el rendimiento de Kyber, especialmente en lo que respecta a la generación de claves, el cifrado y el descifrado, que son procesos computacionalmente intensivos.

En resumen, sin la NTT, los algoritmos como CRYSTALS-Kyber no serían prácticos para su uso en aplicaciones reales debido a las demandas del cálculo computacional que implican. La NTT permite que Kyber sea un esquema de cifrado viable y seguro para la criptografía en la era post-cuántica.

Este trabajo se estructura como sigue. La sección II incluye algunos preliminares esenciales de la Teoría de Anillos, la Transformada Teórica de Números y el Teorema Chino del Resto. La sección III se dedica a los algoritmos Butterfly, mientras que la sección IV trata sobre la convolución en Kyber. En la sección V se aportan datos sobre la implementación de la NTT. Finalmente la sección VI cierra el trabajo con algunas conclusiones y trabajos futuros.

II. Preliminares

A continuación se define la notación utilizada y se incluye una breve introducción a la multiplicación polinomial y a la convolución.

II-A. Teoría de Anillos

Sea $\mathbb{Z}$ el anillo de los enteros, y sean $n \in \mathbb{Z}$ positivo y q un número primo, se considera $\mathbb{Z}_q = \{[0]_q, [1]_q, ..., [q-1]_q\}$ el anillo de los enteros módulo q.

Se denota a lo largo de todo el documento $\mathbb{Z}_q[x]/I$ al anillo cociente sobre el anillo de polinomios $\mathbb{Z}_q[x]$, donde q es un número primo e $I = (x^n + 1)$. Véase en [7] que cualquier $a(x) \in \mathbb{Z}_q[x]/I$, tiene grado menor o igual a $n-1$. Debe tenerse en cuenta que al hacer modulo $x^n + 1$, resulta que $x^n = -1$ por lo que los polinomios del anillo no tendrán grado igual a n, sino menor.

Definición 2.1: (k-ésima raíz primitiva de la unidad [7]). Sea R un anillo conmutativo y unitario, es decir un anillo conmutativo con identidad multiplicativa 1, entonces ψ es una k-ésima raíz de la unidad en R si y solo si

$$\psi^k = 1, \quad \psi^i \neq 1, \forall i \in \{0, 1, .., k-1\}. \tag{1}$$

En Kyber se utilizan los parámetros $q = 3329$, $n = 256$ siendo el anillo $\mathbb{Z}_q[x]/I = \mathbb{Z}_{3329}[x]/(x^{256}+1)$.

El anillo $\mathbb{Z}_{3329}$ en concreto, tiene una particularidad que lo hace ligeramente distinto a los demás y es que para el uso de algoritmos en la multiplicación de polinomios, como se ve más adelante, es necesaria la existencia de una $2n$-ésima raíz de la unidad. En CRYSTALS-Kyber, $2n = 512$, y se comprueba en la implementación desarrollada que no existe dicha $2n$-ésima raíz primitiva de la unidad.

Definición 2.2: (Negative Wrapped Convolution [8]). Sean $q \in \mathbb{N}$ un número primo e $I = (x^n + 1)$ el ideal generado por $x^n+1 \in \mathbb{Z}_q[x]$, y $a(x), b(x) \in \mathbb{Z}_q[x]/I$, supóngase que ambos polinomios $a(x), b(x)$ tienen un grado exacto de $n-1$, si fueran de grado menor se rellenan los coeficientes con ceros.

Siendo $a = (a_0, ..., a_{n-1})$, $b = (b_0, ..., b_{n-1}) \in \mathbb{Z}_q^n$ los vectores de coeficientes de $a(x), b(x)$ respectivamente.

Se define la *Negative Wrapped Convolution* (NWC) de $a(x)$ y $b(x)$ como la multiplicación de estos polinomios

$$c(x) = \sum_{k=0}^{n-1} c_k x^k \tag{2}$$

siendo:

$$c_k = \sum_{i=0}^{k} a_i b_{k-i} + \sum_{i=k+1}^{n-1} a_i b_{k+n-i} \pmod q, \tag{3}$$

$\forall k \in \{0, 1, .., n-1\}$
Estos coeficientes c_k no son más que los coeficientes de la multiplicación de polinomios en $\mathbb{Z}_q[x]$ reducido módulo $I = (x^n + 1)$.

II-B. Transformada Teórica de Números

La NTT es un caso especial de la transformada discreta de Fourier sobre un cuerpo finito. Hay varios tipo de NTT, pues también se pueden dar convoluciones concretas para $\mathbb{Z}_q[x]$, y $\mathbb{Z}_q[x]/(x^n - 1)$. En el caso de $\mathbb{Z}_q[x]/(x^n + 1)$ se tiene la convolución definida anteriormente, y la NTT en Kyber está basada en ella.

Definición 2.3: (Transformada Teórica de Números basada en la NWC [8]). Sea $q \in \mathbb{N}$ un número primo tal que $q \equiv 1 \pmod{2n}$ tal que la $2n$-ésima raíz de la unidad ψ_{2n} existe en $\mathbb{Z}_q$. Se denota $w_n = \psi_{2n}^2$ (la n-ésima raíz de la unidad en $\mathbb{Z}_q$).

Se define entonces la Transformada Teórica de Números basada en la NWC de $a = (a_0, ..., a_{n-1}) \in \mathbb{Z}_q^n$ siendo a el vector de coeficientes de $a(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1}x^{n-1} \in \mathbb{Z}_q[x]/(x^n + 1)$ como:

$$NTT_\psi(a) = (\hat{a}_0, \hat{a}_1, ..., \hat{a}_{n-1}) \tag{4}$$

Donde:

$$\hat{a}_j = \sum_{i=0}^{n-1} a_i \psi_{2n}^i w_n^{ij} \pmod q \tag{5}$$

Realmente la NTT_ψ se puede definir en $\mathbb{Z}_q^n$ como la aplicación:

$$\begin{array}{rccc} NTT_\psi: & \mathbb{Z}_q^n & \longrightarrow & \mathbb{Z}_q^n \\ & (a_0, a_1, ..., a_{n-1}) & \longmapsto & (\hat{a}_0, \hat{a}_1, ..., \hat{a}_{n-1}) \end{array} \tag{6}$$

con $\hat{a}_j$ definidos como *Ec.* (5).

Véase que al ser $w_n = \psi_{2n}^2$, se tiene que la expresión *Ec.* (5) es equivalente a:

$$\hat{a}_j = \psi_{2n}^{2j+1} \sum_{i=0}^{n-1} a_i \psi_{2n}^i \pmod q$$

Además en este documento se denota de igual manera $NTT(a) := NTT_\psi(a)$ pues no se está considerando otro tipo de convoluciones.

Definición 2.4: (Transformada Teórica de Números Inversa basada en la NWC [8]). Partiendo de las mismas hipótesis de la definición de NTT_ψ se define la Transformada Teórica de Números Inversa (*Inverse Number Theoretic Transform*, INTT) basada en la NWC de $\hat{a} = (\hat{a}_0, \hat{a}_1, .., \hat{a}_{n-1})$ siendo $\hat{a}$ el vector de coeficientes de $\hat{a}(x) = \hat{a}_0 + \hat{a}_1 x + \hat{a}_2 x^2 + \cdots + \hat{a}_{n-1}x^{n-1} \in \mathbb{Z}_q[x]/(x^n + 1)$ como:

$$INTT_\psi(\hat{a}) = (a_0, a_1, ..., a_{n-1}) \tag{7}$$

donde:

$$a_i = n^{-1} \psi_{2n}^{-i} \sum_{j=0}^{n-1} \hat{a}_j w_n^{-ij} \pmod q \tag{8}$$

Proposición 2.1: Sea $a(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1}x^{n-1} \in \mathbb{Z}_q[x]/(x^n + 1)$ y $a = (a_0, ..., a_{n-1}) \in \mathbb{Z}_q^n$ su vector de coeficientes adecuado, entonces:

$$INNT_\psi(NTT_\psi(a)) = a \tag{9}$$

Esta proposición ya indica que la NTT_ψ es una correspondencia biyectiva.

Proposición 2.2: (Propiedad de la NWC [8]). Sea $c(x) \in \mathbb{Z}_q[x]/(x^n+1)$ la convolución de dos polinomios $a(x), b(x) \in \mathbb{Z}_q[x]/(x^n + 1)$, denotando por c, a, b los vectores de coeficientes de los respectivos polinomios, y $\circ$ al producto punto a punto de dos vectores, entonces:

$$NTT_\psi(c) = NTT_\psi(a) \circ NTT_\psi(b) \tag{10}$$

Como consecuencia directa de las proposiciones anteriores, se tiene que:

$$c = INNT_\psi(NTT_\psi(a) \circ NTT_\psi(b)) \tag{11}$$

Por tanto, se puede obtener la multiplicación de dos polinomios en $\mathbb{Z}_q[x]/(x^n + 1)$ usando NTT_ψ y $INTT_\psi$ siempre que se asuman las hipótesis.

En Kyber, como se ha mencionado anteriormente, no existe dicha 256-raíz de la unidad, y por tanto, lo anterior por ahora no es aplicable.

En la definición y teorema siguientes se introduce una justificación para el procedimiento que se emplea en Kyber.

II-C. Teorema Chino del Resto

Definición 2.5: (Ideales coprimos [9]). Sean $I, J \subseteq R$ ideales, con R anillo conmutativo y unitario, entonces I, J son ideales coprimos si y solo si $\exists i \in I,\ j \in J$ tal que $i + j = 1$.

Teorema 2.1: (Teorema Chino del Resto Generalizado [9]). Sea R un anillo conmutativo y unitario, e $I_1, I_2, .., I_k \subseteq R$ ideales coprimos entre sí, considerando I como la intersección de todos los I_j, los siguientes anillos son isomorfos:

$$R/I \cong R/I_1 \times R/I_2 \times \cdots \times R/I_k \tag{12}$$

Considérese en $R = \mathbb{Z}_q[x]$ el ideal $H = (x^n + 1)$ con $n \in \mathbb{N}$ una potencia de 2 y ψ_{2n} la $2n$-ésima raíz de la unidad, se cumple por definición que $\psi_{2n}^{2n} = 1$ y al ser q primo, entonces $\mathbb{Z}_q$ es un cuerpo, luego se puede asegurar que $\psi_{2n}^{n} = -1$ por tanto se tiene que el polinomio generador x^n+1 cumple que $x^n+1 = x^n-\psi_{2n}^n = [x^{\frac{n}{2}}-\psi_{2n}^{\frac{n}{2}}][x^n+\psi_{2n}^{\frac{n}{2}}]$.

Así, si se define $I_1 = (x^{\frac{n}{2}} - \psi_{2n}^{\frac{n}{2}})$ y $I_2 = (x^n + \psi_{2n}^{\frac{n}{2}})$ se tiene que $I = I_1 \cap I_2 = H$.

Además se observa que si:

- $a(x) = (-2^{-1}\psi_{2n}^{-\frac{n}{2}})(x^n - \psi_{2n}^{\frac{n}{2}}) \in I_1$
- $b(x) = (2^{-1}\psi_{2n}^{-\frac{n}{2}})(x^n + \psi_{2n}^{\frac{n}{2}}) \in I_2$

se da que:

$$a(x) + b(x) = -2^{-1}\psi_{2n}^{-\frac{n}{2}}x^n + 2^{-1}\psi_{2n}^{\frac{n}{2}}x^n + 2^{-1} + 2^{-1} = 2^{-1} + 2^{-1} = 2 \cdot 2^{-1} = 1$$

Luego, se reúnen las hipótesis del teorema y se concluye que:

$$\mathbb{Z}_q[x]/(x^n + 1) \cong \mathbb{Z}_q[x]/(x^{\frac{n}{2}} - \psi_{2n}^{\frac{n}{2}}) \times \mathbb{Z}_q[x]/(x^{\frac{n}{2}} + \psi_{2n}^{\frac{n}{2}})$$

Es posible aplicar esto recursivamente (ver Fig. 1) pero dado que las raíces de x^n+1 son ψ_{2n}^{2j+1}, con $j \in \{0, 1, ...n-1\}$, de manera análoga teniendo la descomposición de:

$$x^n + 1 = \prod_{j=0}^{n-1}(x^n - \psi_{2n}^{2j+1}) \tag{13}$$

y aplicando el Teorema Chino del Resto:

$$\mathbb{Z}_q[x]/(x^n + 1) \cong \prod_{j=0}^{n-1}\mathbb{Z}_q[x]/(x - \psi_{2n}^{2j+1}) \tag{14}$$

Definición 2.6: (Bit-reverso [8]). Sea $n \in \mathbb{N}$ una potencia de 2 y $b \in \mathbb{Z}$, con $b \geq 0$, se define el bit-reverso de b respecto de n como:

$$br_n(b) = br_n(b_{log_2(n-1)}2^{log_2(n-1)} + \cdots + b_1 2 + b_0) = b_0 2^{log_2(n-1)} + \cdots + b_{log_2(n-2)}2 + b_{log_2(n-1)} \tag{15}$$

Donde b_i representa el i-ésimo bit en la forma binaria de b. Como se ve es simplemente revertir el orden de los bits y devolver la representación entera.

Ahora como el bit reverso (br) de los enteros $\{0, ..., n-1\}$ forman una permutación de este conjunto, la expresión que se tenía es equivalente a:

$$\mathbb{Z}_q[x]/(x^n + 1) \cong \prod_{j=0}^{n-1}\mathbb{Z}_q[x]/(x - \psi_{2n}^{2br(j)+1}) \tag{16}$$

La razón detrás de esta reversión de bits en los índices de exponenciación es que se da un mejor acceso en memoria a la hora de implementar.

Ahora con *Ec. (17)* se sabe que si existe una $2n$-ésima raíz de la unidad, entonces es posible recurrir a una multiplicación punto a punto en la NTT, pues al ser reducido con un polinomio de grado 1 solo quedan constantes.

En Kyber no se puede recurrir a una multiplicación punto a punto de términos constantes, ya que solo existe hasta la n-ésima raíz de la unidad. Luego, se puede aplicar el Teorema Chino del Resto hasta polinomios de grado 2, es decir:

Considerando w_n la n-ésima raíz de la unidad con los parámetros de Kyber, se cumple que:

$$\mathbb{Z}_q[x]/(x^n + 1) \cong \prod_{j=0}^{\frac{n}{2}-1}\mathbb{Z}_q[x]/(x^2 - w_n^{2br(j)+1}) \tag{17}$$

II-D. Complejidad de la NTT

La complejidad de aplicar NTT/INTT de manera directa es de $O(n^2)$, la cuál es la misma complejidad que se da en la multiplicación de polinomios directa con anillos de la forma $K[x]$ con K cuerpo. Por tanto, la eficiencia de esta transformada se encuentra en los algoritmos que rebajan el orden. Se observa en las siguientes subsecciones, y llegan a conseguir un orden de $O(n\log(n))$.

II-E. Ventajas de la NTT

La NTT presenta ciertas propiedades que favorecen a los esquemas de cifrado:

- La NTT es una correspondencia lineal y biyectiva, lo cuál resultará útil para la implementación de Kyber.
- Debido a que la NTT es una correspondencia biyectiva, conserva aleatoriedad de un vector de coeficientes. Con esto se puede generar un vector aleatorio y verlo como un vector ya transformado de la NTT. Esto es posible ya que es sobreyectiva por lo que tendrá un vector que lo tenga como imagen, y además no hay problemas con la aleatoriedad pues la conserva ya que, al ser biyectiva, si la imagen no fuera aleatoria se podría conocer el elemento de entrada, por lo que no sería aleatorio.
- En cuanto al acceso de memoria en una implementación, como la transformada lleva n puntos en n puntos, es posible guardar $\hat{a}$ donde originalmente se encontraba a en memoria.
- Como se ve en el algoritmo de Cooley-Tukey (ver Fig. 2), se pueden calcular con cierta recursividad los coeficientes $\hat{a}_j$, haciendo que se ahorre en términos de coste computacional en la transformación.

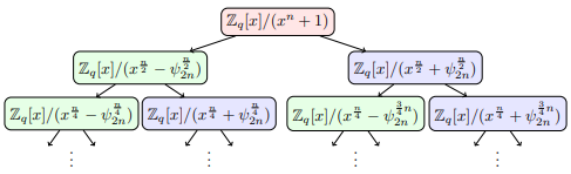

Figura 1. Teorema Chino del Resto en $\mathbb{Z}_q[x]/(x^n + 1)$

III. Algoritmos-*Butterfly*: Cooley-Tukey y Gentleman-Sande

En esta sección se desarrollan los algoritmos que mejoran en gran medida el rendimiento de la NTT, explicando el procedimiento por el cuál se llegan a tener los esquemas *Butterfly* [10].

III-A. Base teórica de los algoritmos Butterfly.

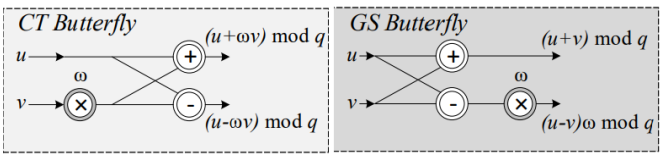

Figura 2. Algoritmos Butterfly

De la definición de NTT_ψ (2.3), los coeficientes transformados Ec. (5) :

$$\hat{a}_j = \sum_{i=0}^{n-1} a_i \psi_{2n}^i w_n^{ij} \pmod q$$

Se divide el sumatorio en los índices pares e impares por i, recordar que n es potencia de 2 luego divisible por 2, queda:

$$\hat{a}_j = \sum_{i=0}^{\frac{n}{2}-1} a_{2i} \psi_{2n}^{2i} w_n^{2ij} + \sum_{i=0}^{\frac{n}{2}-1} a_{2i+1} \psi_{2n}^{2i+1} w_n^{(2i+1)j} \pmod q$$

De aquí en adelante las operaciones asumen la aplicación de módulo q.

Usando simplemente la propiedad de potencias, y sacando el factor ψ_{2n} y w_n^j, pues j no depende de la variable i en el sumando, se tiene que $\hat{a}_j$ es:

$$\sum_{i=0}^{\frac{n}{2}-1} a_{2i} (\psi_{2n}^i)^2 (w^{ij})_n^2 + \psi_{2n} w_n^j \sum_{i=0}^{\frac{n}{2}-1} a_{2i+1} (\psi_{2n}^i)^2 (w_n^{ij})^2$$

Ahora, la idea es sustituir $j + \frac{n}{2}$ en lo anterior y considerar las propiedades de estas raíces de la unidad que se vieron en las secciones II-B, II-C, estas son: $\psi_{2n}^n = -1$ y $w_n = \psi_{2n}^2$. Resulta que $\hat{a}_{j+\frac{n}{2}}$ es:

$$\sum_{i=0}^{\frac{n}{2}-1} a_{2i} (\psi_{2n}^i)^2 (w^{ij})_n^2 - \psi_{2n} w_n^j \sum_{i=0}^{\frac{n}{2}-1} a_{2i+1} (\psi_{2n}^i)^2 (w_n^{ij})^2$$

Se define:

$$\hat{a}'_j := \sum_{i=0}^{\frac{n}{2}-1} a_{2i} (\psi_{2n}^i)^2 (w^{ij})_n^2 \pmod q$$

$$\hat{a}''_j := \sum_{i=0}^{\frac{n}{2}-1} a_{2i+1} (\psi_{2n}^i)^2 \pmod q$$

Luego, teniendo en cuenta que $\psi_{2n} w_n^j = \psi_{2n}^{2j+1}$ y sustituyendo por $\hat{a}'_j$, $\hat{a}''_j$:

$$\hat{a}_j = \hat{a}'_j + (\psi_{2n}^{2j+1}) \hat{a}''_j \pmod q$$

$$\hat{a}_{j+\frac{n}{2}} = \hat{a}'_j - (\psi_{2n}^{2j+1}) \hat{a}''_j \pmod q$$

Donde además estos $\hat{a}'_j$, $\hat{a}''_j$ corresponden al cálculo de una NTT sobre $\frac{n}{2}$ puntos (de la propia definición) y así recursivamente se obtienen los coeficientes.

Véase que lo que se está haciendo es separar en cada etapa k en la que se aplica esta recursividad con una distancia de $n/(k+1)$ en los índices de $\hat{a}_j$ tal y como está descrito en Fig. 3 [11]. Por ejemplo en la primera etapa, se está a distancia $\frac{n}{2}$: $\hat{a}_j$ y $\hat{a}_{j+\frac{n}{2}}$.

Esta es una de las bases del algoritmo Butterfly, el cuál se denota así por que las operaciones en cada subetapa al representarlo esquemáticamente se 'asemejan' a la figura de una mariposa, como se muestra en Fig. 2.

Para el caso de la INTT se procede manera análoga, de la expresión de la $INTT_\psi$:

$$a_i = n^{-1} \psi_{2n}^{-i} \sum_{j=0}^{n-1} \hat{a}_j w_n^{-ij} \pmod q$$

Desarrollando y definiendo los siguientes elementos:

$$\hat{b}'_j := \hat{a}_j + \hat{a}_{j+\frac{n}{2}} \pmod q$$

$$\hat{b}''_j := (\hat{a}_j + \hat{a}_{j+\frac{n}{2}}) \psi^{-(2j+1)} \pmod q$$

Nos queda en la sustitución la expresión:

$$a_{2i} = (\psi_{2n}^2)^{-i} \sum_{j=0}^{\frac{n}{2}-1} \hat{b}'_j (w^2)_n^{-ij} \pmod q$$

$$a_{2i+1} = (\psi_{2n}^2)^{-i} \sum_{j=0}^{\frac{n}{2}-1} \hat{b}''_j (w^2)_n^{-ij} \pmod q$$

Por tanto, es posible aplicar ahora un $INTT_\psi$ basada en $\frac{n}{2}$ puntos. De nuevo es posible hacer una recurrencia análoga a la que hecha con la NTT_ψ.

III-B. Cooley-Tukey

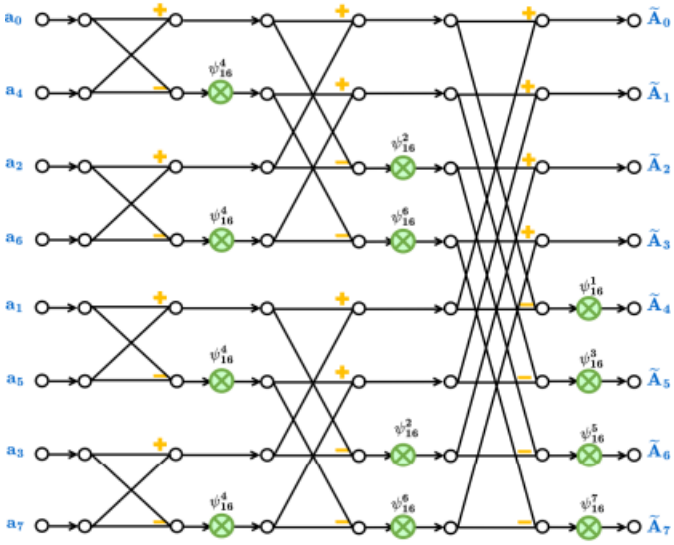

Figura 3. Butterfly-Cooley-Tukey para un $n = 8$

Una vez introducidas las bases, se describe el algoritmo de Cooley-Tukey [12] en el pseudocódigo incluido como

Algoritmo 1. La notación usada es la siguiente: m indica la etapa actual de la NTT_ψ, mientras que k se refiere a la distancia a la que están los índices en dicha etapa. A su vez, el cálculo de índices allí descrito se usa para indexar cada uno de los elementos, sumar la distancia k y realizar la operación mariposa.

Véase que la salida de este algoritmo está automáticamente generada en bit-reverse, tal y cómo se refleja en Fig. 3. La lista de entrada, cuyos elementos corresponden a los coeficientes de un polinomio en $\mathbb{Z}_q[x]/(x^8+1)$, tienen la correspondencia de índices en bit reverse (sobre 3-bits pues $3 = \log_2(8)$) : $[a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7] \rightarrow [a_0, a_4, a_2, a_6, a_1, a_5, a_3, a_7]$ por lo que está en reversión.

Esto no es un problema ni requiere modificaciones, ya que el conjunto de datos de entrada $INTT_\psi$ está organizado de acuerdo con el método de bit-reversión y luego se aplica el proceso de bit-reversión nuevamente en la salida de manera predeterminada. Por lo tanto, al realizar convoluciones de polinomios que requieran la aplicación simultánea de NTT_ψ e $INTT_\psi$, se obtiene el resultado correcto.

Algoritmo 1 NTT_ψ basada en el algoritmo Cooley-Tukey Butterfly

Entrada: Vector $a = (a_0, a_1, .., a_{n-1})$ de $\mathbb{Z}_q$.
Salida: $a \leftarrow NTT_\psi(a)$
$m \leftarrow 1$
$k \leftarrow \frac{n}{2}$
$\psi \leftarrow RaizPrimitiva(2n, q)$ ▷ Previamente creada
while $m < n$ **do**
  for $i = 0$ to $i = m - 1$ **do**
    $j_1 \leftarrow 2 \cdot i \cdot k$
    $j_2 \leftarrow j_1 + k - 1$
    $S \leftarrow \psi^{bitrev(m+i,n)}$
    for $j = j_1$ to $j = j_2$ **do**
      $u \leftarrow a[j]$
      $v \leftarrow a[j + t]$
      $a[j] \leftarrow u + v \cdot S \pmod q$
      $a[j + t] \leftarrow u - v \cdot S \pmod q$
    end for
  end for
  $m \leftarrow m \cdot 2$
  $k \leftarrow k/2$
end while
return a

III-C. Gentleman-Sande

Siguiendo el pseudocódigo del Algoritmo 2, y utilizando un razonamiento similar al explicado anteriormente, se observa que al aplicar la recursividad inversa, los índices m y k, así como la exponenciación de ψ^{-1}, se comportan de manera opuesta a la utilizada en el algoritmo de Cooley-Tukey. Se concluye con la correspondiente operación mariposa descrita en la sección III-A.

Es importante notar que dado que $\psi^{2n} = 1$, entonces $\psi^{2n-1} \cdot \psi = 1$. Asi, se puede deducir que $\psi^{-1} = \psi^{2n-1}$. Por lo tanto, se calcula la inversa de ψ elevando a la potencia $2n - 1$, como se indica en el Algoritmo 2.

Algoritmo 2 $INTT_\psi$ basada en el algoritmo Gentleman-Sande Butterfly

Entrada: Vector $a = (a_0, a_1, .., a_{n-1})$ de $\mathbb{Z}_q$.
Salida: $a \leftarrow INTT_\psi(a)$
$m \leftarrow \frac{n}{2}$
$k \leftarrow 1$
$\psi \leftarrow RaizPrimitiva(2n, q)$ ▷ Previamente creada
$\psi^{-1} \leftarrow \psi^{2n-1}$
$n^{-1} \leftarrow n^{q-1}$
while $m \geq n$ **do**
  for $i = 0$ to $i = m - 1$ **do**
    $j_1 \leftarrow 2 \cdot i \cdot k$
    $j_2 \leftarrow j_1 + k - 1$
    $S \leftarrow \psi^{bitrev(m+i,n)}$
    for $j = j_1$ to $j = j_2$ **do**
      $u \leftarrow a[j]$
      $v \leftarrow a[j + t]$
      $a[j] \leftarrow (u + v) \pmod q$
      $a[j + t] \leftarrow (u - v) \cdot S \pmod q$
    end for
  end for
  $m \leftarrow m/2$
  $k \leftarrow k \cdot 2$
end while
$a \leftarrow a \cdot n^{-1}$ ▷ Se multiplica cada componente por n^{-1}
return a

III-D. Convolución de polinomios, con una $2n$-ésima raíz de la unidad en $\mathbb{Z}_q$

Una vez se cuenta con estos algoritmos, al estar diseñados en base a la existencia de una $2n$-ésima raíz de la unidad en $\mathbb{Z}_q$, la convolución en este caso consiste en aplicar la ecuación siguiente:

$$c = INNT_\psi(NTT_\psi(a) \circ NTT_\psi(b)) \tag{18}$$

Para ello es conveniente dar un criterio que asegurare la existencia de dicha raíz primitiva de la unidad.

Proposición 3.1: (Criterio de existencia de una $2n$-ésima raíz de la unidad en $\mathbb{Z}_q$ [13]). Sea q un número primo tal que $2n$ divide a $q - 1$, entonces existe una $2n$-ésima raíz de la unidad en $\mathbb{Z}_q$.
Por lo que, en caso de cumplir este criterio, bastaría con implementar los algoritmos correspondientes. Se incluye un caso para ilustrar la implementación realizada en Python con los siguientes datos: número primo $q = 257 = 2^8 + 1$ y $n = 16$. Dado que $257 - 1 = 2^8$ es divisible por $2n = 32$, se tiene la existencia de la raíz de la unidad.

Realmente es necesario otro requerimiento para aplicar la NTT_ψ, y es que se necesita que en todo momento se pueda dividir n por 2 hasta llegar 1, luego el valor de n debe ser una potencia de 2.

IV. Convolución en Kyber

En Kyber se cuenta con los parámetros $q = 3329$ y $n = 256$, donde no existe una 512-ésima raíz de la unidad en $\mathbb{Z}_{3329}$. Por tanto, no se pueden aplicar directamente los

algoritmos descritos, se requiere una pequeña modificación inicial.

Tal y como se describió en la sección II-C *Teorema Chino del Resto* se necesita una multiplicación punto a punto en $\mathbb{Z}_q[x]/(x^2 - w_n^{2br(i)+1})$ por lo que se tiene una multiplicación de polinomios de grado 1, y posteriormente una reducción modular con un polinomio de grado 2.

La estrategia a seguir es separar un polinomio dado en el anillo de Kyber con $n = 256$, en dos polinomios nuevos, uno con sus coeficientes pares y otro con sus coeficientes impares. Ahora, cada polinomio tiene grado $n = 128$, por lo que si existe una 256-raíz de la unidad en $\mathbb{Z}_{3329}$ para cada uno. Esto permite aplicar la NTT a cada uno de estos polinomios. El resultado obtenido en estas dos transformaciones tiene un sentido algebraico derivado de lo estudiado para Kyber en la sección II-C. Esto es, se parte de:

$$a(x) = a_0 + \cdots + a_{255}x^{255}$$
$$b(x) = b_0 + \cdots + b_{255}x^{255}$$

- Se separan los polinomios en su parte par e impar:

$$a_{par}(x) = a_0 + a_2x \cdots + a_{254}x^{128}$$
$$a_{impar}(x) = a_1 + a_3x + \cdots + a_{255}x^{128}$$
$$b_{par}(x) = b_0 + b_2x \cdots + b_{254}x^{128}$$
$$b_{impar}(x) = b_1 + b_3x + \cdots + b_{255}x^{128}$$

- Se aplica NTT a los coeficientes asociados, denotando la NTT_ψ y $INTT_\psi$ de un polinomio como la transformada de sus coeficientes asociados:

$$NTT_\psi(a_{par}(x)) = (\hat{a}_0, \hat{a}_2, .., \hat{a}_{254})$$
$$NTT_\psi(a_{impar}(x)) = (\hat{a}_1, \hat{a}_3, ..., \hat{a}_{255})$$
$$NTT_\psi(b_{par}(x)) = (\hat{b}_0, \hat{b}_2, ..., \hat{b}_{254})$$
$$NTT_\psi(b_{impar}(x)) = (\hat{b}_1, \hat{b}_3, ..., \hat{b}_{255})$$

- Se usa la expresión algebraica que tiene estas transformada en conjunto sobre $\mathbb{Z}_q[x]/(x^2 - w_n^{2br(i)+1})$:

$$NTT_{Kyber}(a(x)) = (\hat{a}_0 + \hat{a}_1x, \hat{a}_2 + \hat{a}_3x, .., \hat{a}_{254} + \hat{a}_{255}x)$$
$$NTT_{Kyber}(b(x)) = (\hat{b}_0 + \hat{b}_1x, \hat{b}_2 + \hat{b}_3x, .., \hat{b}_{254} + \hat{b}_{255}x)$$

- Se aplica en este dominio la multiplicación punto a punto en $\mathbb{Z}_q[x]/(x^2 - w_n^{2br(i)+1})$, es decir:

$$NTT_{Kyber}(a(x)) \circ NTT_{Kyber}(b(x)) = (\hat{c}_0 + \hat{c}_1x, \hat{c}_2 + \hat{c}_3x, .., \hat{c}_{254} + \hat{c}_{255}x)$$

  Donde estos c_k se pueden calcular sin tener que multiplicar y posteriormente hacer reducción de módulo a cada uno, pues ya se sabe la forma que tiene ese cociente cuando es una multiplicación de polinomios de grado 1. Esta es:

$$c_{2i} = a_{2i} \cdot b_{2i} + a_{2i+1} \cdot b_{2i+1}w^{2br(i)+1}$$
$$c_{2i+1} = a_{2i} \cdot b_{2i+1} + a_{2i+1} \cdot b_{2i}$$

  con $i \in \{0, 1, 2..., 255\}$.
- Por último, se consideran estos coeficientes c_k obtenidos de la multiplicación punto a punto y se aplica la $INTT_\psi$ a los coeficientes pares e impares, tal y como se precedió con NTT_ψ. Juntando en un solo polinomio los coeficientes obtenidos que corresponden a la parte par e impar, se obtiene la convolución de polinomios. Esto es, denotando $INTT_{Kyber}$ a la forma descrita de aplicar $INTT_\psi$ se obtiene:

$$a(x) \cdot b(x) = INTT_{Kyber}(c(x))$$

V. Implementación de la NTT_ψ, $INTT_\psi$ y la convolución en Kyber

En esta sección se incluye la implementación realizada en Google Colab desarrollada en Python, de la NTT_ψ, $INTT_\psi$, usando Cooley-Tukey, Gentleman-Sande respectivamente tal y como se describió en los Algoritmos 1 y 2, así como la convolución en Kyber, se puede consultar en [15].

Además como medida de eficiencia se muestran los tiempos de ejecución de dicha implementación y una ya creada en el módulo $SymPy$, una biblioteca de Python para matemáticas simbólicas. También se incluye una comparación de tiempos de ejecución de la multiplicación de polinomios de Kyber usando una multiplicación directa desde el mismo módulo de $SymPy$, que ya cuenta con algoritmos eficientes para multiplicar polinomios de alto grado, y un tipo de convolución para un anillo de polinomios.

Cabe resaltar que la implementación de NTT e INTT de $SymPy$ hace uso de un algoritmo similar pero para un caso distinto de convolución que no es aplicable para $\mathbb{Z}_q[x]/(x^n + 1)$ pero es lo más cercano que dispone Python con librerías reconocidas.

Se ha escogido la plataforma de Google Colab y Python con el fin de obtener una verificación de estos algoritmos y facilitar el aprendizaje sobre la NTT. Python es un lenguaje de alto nivel sencillo de manejar, y Google Colab proporciona un entorno apto para introducir de manera didáctica la implementación realizada.

Para ello se crean las funciones necesarias para la implementación, y posteriormente las funciones NTT, INTT y KyberConvolution. Además, se han implementado funciones más básicas como por ejemplo, el *bit-reverse*. Seguidamente se describen las principales funciones.

V-A. *Función de la n-ésima raíz de la unidad*

Para implementar una función en Python que calcule la n-ésima raíz de la unidad en $\mathbb{Z}_q$ se ha planteado una búsqueda iterativa que devuelve el elemento si cumple las dos condiciones necesarias para que sea n-ésima raíz de la unidad. Véase que no solo se calcula una n-ésima raíz de la unidad si no que es la primera de ellas.

En Fig. 4 y Fig. 5 se muestran los resultados de aplicar esta función en el cuerpo de Kyber y en $\mathbb{Z}_{17}$, ilustrando así que es una función general. Además, en Fig. 6 se muestra con esta misma función que no existe una 512-raíz de la unidad en $\mathbb{Z}_{3329}$.

```
# Ejemplo de la función primera_raiz_primitiva para el
#número q primo de kyber 3329=13*2^8+1
PrimeraRaizPrimitiva(256,3329)

17
```

Figura 4. Primera-Raiz-Primitiva para $Kyber$ y $n = 256$

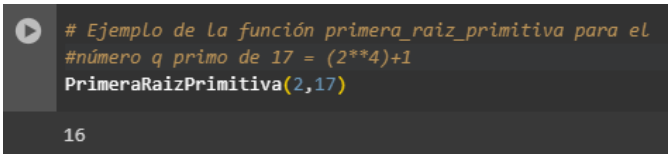

Figura 5. Primera-Raiz-Primitiva para $q = 17$ y $n = 2$

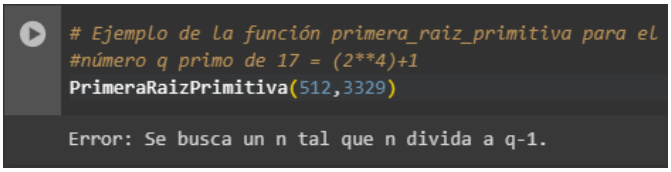

Figura 6. Primera-Raiz-Primitiva para Kyber y $n = 512$

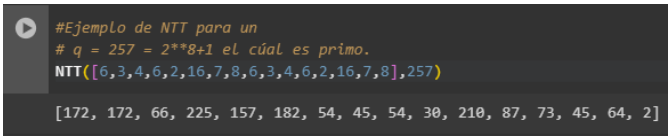

Figura 7. NTT aplicada a $q = 257$ y $n = 16$

V-B. *Funciones NTT y $INTT$*

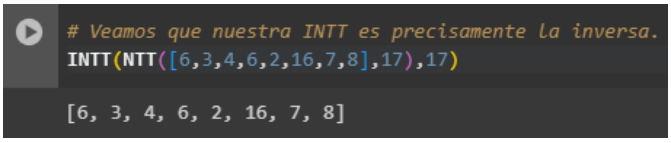

Figura 8. INTT aplicada a un elemento del dominio de NTT

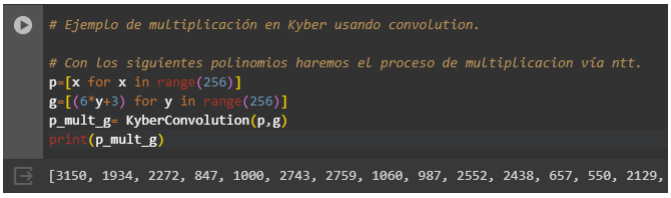

Figura 9. Kyber Convolution aplicada a polinomios de $\mathbb{Z}_{3329}[x]/(x^{256}+1)$

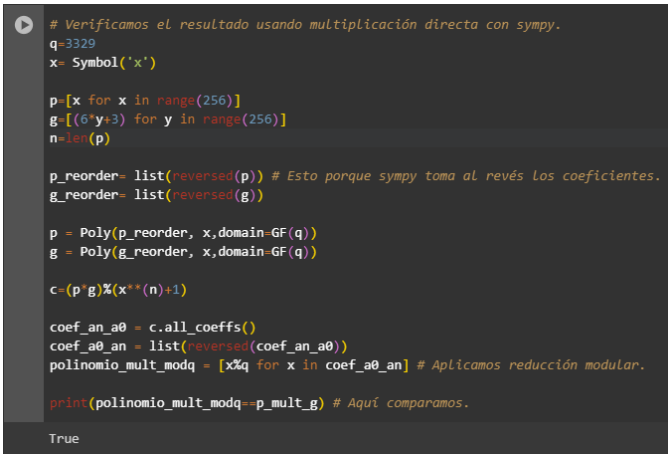

Figura 10. Verificación de la multiplicación vía NTT.

A continuación se muestran los resultados obtenidos en la implementación presentada para las funciones principales, mostrando en Fig. 7 un caso de aplicación de la NTT_ψ y en Fig. 8 una comprobación de que la $INTT_\psi$ calcula el inverso de la transformada de manera correcta.

Esta implementación se consigue usando como punto de partida los Algoritmos 1 y 2, aplicados sobre una lista de entrada que tomaran los coeficientes de su respectivo polinomio de menor a mayor, siendo el output de la misma naturaleza.

V-C. *Función Kyber-Convolution*

En esta función se implementa el proceso descrito en la sección IV.

Se muestra en Fig. 9 un ejemplo definiendo dos polinomios de gran tamaño con $n = 256$ en $\mathbb{Z}_{3329}[x]/(x^{256}+1)$, creados a partir de expresiones simples utilizando un bucle, donde se verifica el resultado utilizando multiplicación directa por $SymPy$ en Fig. 10.

V-D. *Comparativa en tiempos de ejecución*

Se muestra la comparativa de tiempos de ejecución para las diferentes NTT de la implementación desarrollada y la de $SymPy$. Se incluye también comparación de la multiplicación de polinomios vía NTT y multiplicación directa.

Se debe recordar que realmente la NTT de $SymPy$ no es aplicable para la convolución en Kyber y es por ello, por lo que se incluyen los tiempos de la multiplicación directa de $SymPy$. Para la medición de los tiempos de ejecución se ha usado el paquete *time* de Python.

Se destacan los resultados realizados con los parámetros $q = 257$, $n = 16$ para la NTT, INTT, INTT $\circ$ NTT y parámetros $q = 3329$, $n = 256$ para la multiplicación en la I. Realmente tardan menos de 1 segundo pero al usar el módulo de time se ralentizan esos pasos, de todas formas interesa la comparativa por lo que no supone un inconveniente.

Tabla I
COMPARATIVA PARA LA IMPLEMENTACIÓN DE LA NTT

Código a comparar	En $SymPy$ (s)	Implementación propia (s)
NTT	1,0076167583465576	1,006410837173462
INTT	1,007662057876587	1,007622480392456
INTT $\circ$ NTT	1,009049892425537	1,006197214126587
Multiplicación	2,2660510540008545	1,2369062900543213

Se puede observar que existe una gran diferencia entre la multiplicación directa y la convolución, lo cuál es de esperar pues $SymPy$ además de hacer multiplicación directa calcula la reducción módulo $x^{256}+1$, una operación costosa. Además, debido a que la versión NTT de $SymPy$ es para un tipo de convolución que no necesita exponenciación de la raíz primitiva en los algoritmos Butterfly, tendría que ser menos costosa por lo que si están con valores de ejecución próximos entre sí significa que la implementación presentada no es una mala aproximación para lograr ilustrar la Transformada Teórica de Números.

VI. CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo se aporta un detallado desarrollo de la multiplicación polinomial sobre anillos cocientes mediante el uso de la Transformada Teórica de Números, elemento de algunos esquemas de cifrado resistentes a la computación

cuántica que se suele pasar por alto debido a su complejidad. Sin embargo, este elemento es crucial a la hora de lograr una implementación eficiente de los algoritmos que involucran criptografía post-cuántica basada en retículos. Por ello, se incluye en este documento un análisis que parte de la base teórica necesaria y una descripción de los algoritmos implicados (Cooley-Tukey y Gentleman-Sande), haciendo pruebas de implementación en Python y verificando la eficiencia de esta comparando tiempos de ejecución.

Una de las líneas de trabajo futuro es el estudio e implementación del esquema de cifrado CRYSTALS-Kyber al completo, haciendo uso de la eficiente multiplicación de polinomios que se ha presentado en este documento.

AGRADECIMIENTOS

Este trabajo ha sido posible gracias a las Cátedras de Ciberseguridad de la Universidad de La Laguna patrocinadas por Binter y por INCIBE. Además forma parte de los proyectos: PID2022-138933OB-I00 financiado por MCIN/AEI/ 10.13039/501100011033/FEDER, UE, y SCITALA C064/23 ULL-INCIBE financiado con fondos del Plan de Recuperación, Transformación y Resiliencia, con financiación de la UE (Next Generation).

REFERENCIAS

[1] NIST: “NIST: an official website of the US government”, en: https://www.nist.gov/news-events/news/2023/08/nist-standardize-encryption-algorithms-can-resist-attack-quantum-computers.

[2] CRYSTALS Kyber: “Crystals Kyber resource”, en https://pq-crystals.org/kyber/index.shtml.

[3] CRYSTALS Dilithium: “Crystals Dilithium resource”, en https://pq-crystals.org/dilithium/index.shtml.

[4] SPHINCS+: “Stateless Hash-Based Digital Signature Standard”, en https://doi.org/10.6028/NIST.FIPS.205.ipd.

[5] Hung Nguyen, Linh Tran. “Design of Polynomial NTT and INTT Accelerator for Post-Quantum Cryptography CRYSTALS-Kyber”, en https://link.springer.com/article/10.1007/s13369-022-06928-w.

[6] Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, Damien Stehlé. “Algorithm Specifications And Supporting Documentation”, en https://pq-crystals.org/kyber/resources.shtml.

[7] Lauritzen N. “Concrete Abstract Algebra: From Numbers to Gröbner Bases”. Cambridge University Press, 2003.

[8] Zhichuang Liang, Yunlei Zhao. “Number Theoretic Transform and Its Applications in Lattice-based Cryptosystems: A Survey”, en https://doi.org/10.48550/arXiv.2211.13546.

[9] Kenneth Ireland, Michael Rosen. “A Classical Introduction to Modern Number Theory”. Springer-Verlag, 1990.

[10] Mojtaba Bisheh Niasar, Reza Azarderakhsh, Mehran Mozaffari Kermani. “High-Speed NTT-based Polynomial Multiplication Accelerator for CRYSTALS-Kyber Post-Quantum Cryptography”, en https://eprint.iacr.org/2021/563.

[11] Sin-Wei Chiu, Keshab K. Parhi. “Long Polynomial Modular Multiplication using Low-Complexity Number Theoretic Transform”, en https://arxiv.org/abs/2306.12519.

[12] Longa, P., Naehrig, M. (2016). “Speeding up the Number Theoretic Transform for Faster Ideal Lattice-Based Cryptography. In: Foresti, S., Persiano, G. (eds) Cryptology and Network Security. CANS 2016”. Lecture Notes in Computer Science(), vol 10052. Springer, Cham. en: https://doi.org/10.1007/978-3-319-48965-0_8

[13] Tom Apostol. “Introducción A La Teoría Analítica De Los Números”. Springer-Verlag, 1984.

[14] SymPy: “SymPy”, en https://www.sympy.org/en/index.html. Consultado en: 24 de marzo de 2024.

[15] Antuñano Cabrera, N.: “Implementación NTT”https://colab.research.google.com/drive/1oUBPXxWyBsXkIiBWhCK6LchLmpE_Gqvd?usp=sharing. Consultado en: 24 de marzo de 2024.

A.4. Artículo presentado “GeoGebra para introducir los fundamentos de la criptografía basada en retículos”

Édgar Pérez-Ramos, Pino Caballero-Gil, Héctor Reboso-Morales
“GeoGebra para introducir los fundamentos de la criptografía basada en retículos”
IX Jornadas Nacionales de Investigación en Ciberseguridad (JNIC)
Sevilla, ESP. 27-29 Mayo, 2024

GeoGebra para introducir los fundamentos de la criptografía basada en retículos

Édgar Pérez-Ramos
Universidad de La Laguna
Tenerife, España
alu0101207667@ull.edu.es

Pino Caballero-Gil
Universidad de La Laguna
Tenerife, España
pcaballe@ull.edu.es

Héctor Reboso-Morales
Universidad de La Laguna
Tenerife, España
hreboso@ull.edu.es

***Resumen*—El avance y la amenaza que representa la computación cuántica plantean desafíos significativos para la seguridad de la información. Esta situación ha generado la necesidad imperiosa de desarrollar sistemas criptográficos capaces de resistir ataques cuánticos. En este contexto, la criptografía post-cuántica ha surgido como un campo de investigación crucial. Sin embargo, uno de los principales desafíos que enfrenta este campo es la escasez de recursos didácticos disponibles. El presente trabajo aborda esta problemática mediante la creación de recursos didácticos centrados en retículos utilizando GeoGebra. Se ha diseñado una actividad que introduce y relaciona los retículos con diversos conceptos matemáticos impartidos en el aula. La actividad desarrollada es totalmente novedosa y tiene como objetivo hacer más accesible la comprensión de la criptografía post-cuántica al proporcionar herramientas visuales y manipulables que ayuden a los estudiantes a interiorizar los principios matemáticos y computacionales implicados en la seguridad de la información en la era post-cuántica.**

***Index Terms*—Retículos, GeoGebra, Criptografía post-cuántica**

Tipo de contribución: Formación e innovación educativa

I. Introducción

En la actualidad, la criptografía juega un papel esencial en todas las tecnologías de la comunicación, ya que es necesaria para proteger la seguridad de los sistemas y mensajes. Entre las técnicas criptográficas más relevantes, el cifrado se utiliza para proteger secretos, mientras que la firma digital se emplea para verificar la autenticidad e integridad de documentos y mensajes digitales. Sin embargo, con el posible despliegue futuro de la computación cuántica, se ha descubierto que muchos algoritmos criptográficos utilizados actualmente en diferentes tecnologías, como RSA o ECDSA, serán totalmente vulnerables, motivo por el cual emana la necesidad de nuevos esquemas criptográficos resistentes a la llamada amenaza cuántica.

El Instituto Nacional de Normas y Tecnología (NIST) ha dedicado recientemente varios años de esfuerzo a la búsqueda de algoritmos estandarizados capaces de resistir los retos que plantea la computación cuántica. En 2022, se dieron a conocer los cuatro finalistas de este exhaustivo proceso, entre los que destacaban algoritmos como CRYSTALS-Kyber, [1], diseñado para el cifrado, y CRYSTALS-Dilithium, [2], destinado a las firmas digitales.

Desde entonces, se han dedicado numerosos esfuerzos a verificar la solidez de estos esquemas, dada la importancia que adquirirán en los próximos años. En particular, el NIST ha desarrollado recientemente los correspondientes borradores de los Estándares Federales de Procesamiento de la Información (FIPS), FIPS 203 [3] y FIPS 204 [4], que especifican el Mecanismo de Encapsulación de Claves Módulo-Lattice (ML-KEM) y el Algoritmo de Firma Digital Módulo-Lattice (ML-DSA), derivados de CRYSTALS-Kyber y CRYSTALS-Dilithium, respectivamente.

Los retículos son el factor común de la mayoría de los algoritmos resistentes a la cuántica (CRYSTALS-Kyber, CRYSTALS-Dilithium y FALCON, [5]). No obstante, la familiarización con los fundamentos de la teoría de retículos puede ser vista como una necesidad en diversos niveles educativos, con el fin de preparar de manera más efectiva a los futuros ingenieros y científicos.

Este trabajo se estructura de la siguiente forma: En primer lugar, en la Sección II se presentan las herramientas necesarias para comprender el trabajo. Se expone qué es el GeoGebra, se justifica para qué tipo de alumnado está enfocada la actividad y se introducen los conceptos matemáticos que se trabajan en el ejercicio. En la Sección III se desarrolla y explica la actividad, que puede consultarse en [6] y la sección IV trata sobre una propuesta de cuestionarios para la evaluación de la actividad. Por último, se dedica una última sección a los trabajos futuros y las conclusiones.

II. Preliminares

Los retículos son el objeto algebraico más predominante en los estándares tanto de cifrado como de firma actuales. En la literatura, el material didáctico disponible sobre los retículos suele ser escaso y en ocasiones puede llegar a ser complejo y abstracto para el alumnado poco experimentado. Es por ello que en este trabajo se ha desarrollado una serie de actividades con el software GeoGebra, [7].

GeoGebra es un proyecto de software de matemáticas libre que nació en el año 2001 por parte de Markus Hohenwarter. GeoGebra combina aspectos de la geometría, el álgebra, el cálculo y otros campos de las matemáticas. Es ampliamente utilizado en entornos educativos, desde escuelas primarias hasta niveles universitarios, así como por investigadores, profesionales en matemáticas y disciplinas relacionadas. Algunas características son:

- Interfaz dinámica: GeoGebra proporciona una interfaz dinámica que permite a los usuarios crear, manipular y explorar objetos matemáticos en tiempo real. Esto facilita la comprensión de conceptos matemáticos abstractos al permitir la visualización interactiva.
- Geometría interactiva: Los usuarios pueden construir y manipular figuras geométricas, como puntos, líneas, segmentos, polígonos, círculos y mucho más. Estas figuras

pueden ser arrastradas, rotadas, escaladas y modificadas fácilmente.
- Álgebra dinámica: GeoGebra permite trabajar con expresiones algebraicas, ecuaciones y funciones de manera interactiva. Los usuarios pueden graficar funciones, resolver ecuaciones, encontrar derivadas e integrales, y realizar manipulaciones algebraicas.
- Visualización de datos: GeoGebra permite la visualización de datos mediante la creación de gráficos de funciones, diagramas de dispersión, histogramas y otros tipos de representaciones visuales. Esto facilita el análisis y la interpretación de datos en contextos matemáticos y científicos.
- Herramientas adicionales: Además de sus capacidades principales en geometría y álgebra, GeoGebra también incluye herramientas para trabajar con cálculo diferencial e integral, estadísticas, probabilidad, y más.

Por tanto, por todas las ventajas anteriormente mencionadas, se ha escogido GeoGebra por su alta eficiencia y utilidad en entornos didácticos, como por ejemplo se evidencia en [8] y [9].

Este trabajo puede encajar a partir de 4º de ESO, incluyendo el Bachillerato, puesto que, como aparece en [10] tanto Matemáticas A como Matemáticas B coinciden en:

- Criterios de evaluación:
  - Competencia específica 7.2 " Seleccionar entre diferentes herramientas, incluidas las digitales, y formas de representación (pictórica, gráfica, verbal o simbólica) valorando su utilidad para compartir información."
- Saberes básicos:
  - Sentido espacial:
    - Figuras geométricas de dos y tres dimensiones. "Propiedades geométricas de objetos matemáticos y de la vida cotidiana: investigación con programas de geometría dinámica."
    - Movimientos y transformaciones. "Transformaciones elementales en la vida cotidiana: investigación con herramientas tecnológicas como programas de geometría dinámica, realidad aumentada..."
  - Sentido algebraico:
    - Pensamiento computacional. "Resolución de problemas mediante la descomposición en partes, la automatización y el pensamiento algorítmico."

Por otra parte, cabe destacar que una motivación fundamental para desarrollar este trabajo ha sido el último Informe PISA (se puede consultar en [11]), en el cual España ha obtenido su peor resultado desde que comenzó a realizarse dicha prueba. En particular, el alumnado español de último curso de ESO ha bajado 8 puntos en matemáticas respecto a la edición anterior. La evolución puede verse en Fig. 1

Por lo tanto, tras estos resultados crear material didáctico útil sobre matemáticas se vuelve de carácter urgente. La aspiración es abordar las deficiencias identificadas y fortalecer las habilidades numéricas y conceptuales de los estudiantes.

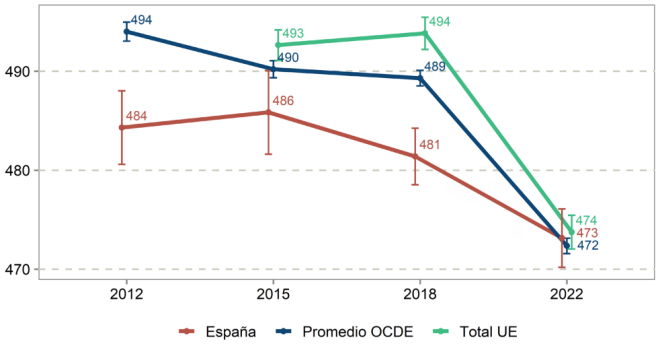

Figura 1. Evolución de las puntuaciones de matemáticas

Conceptos matemáticos

La actividad desarrollada en [6] tiene un doble propósito: emplear los principios matemáticos enseñados en Secundaria y Bachillerato para introducir la criptografía post-cuántica a los estudiantes, o utilizar la criptografía post-cuántica como herramienta para mejorar la comprensión de diversos conceptos matemáticos abordados en el aula. Estos conceptos incluyen:

- Retículos
- Producto escalar de dos vectores
- Ortogonalidad
- Sistema de referencia afín y usual
- Distancia euclídea y taxi

A continuación se procede a definir de manera formar los conceptos previamente mencionados.

Formalmente un retículo se puede definir de la siguiente forma:

Definición 1: Sean V un espacio vectorial sobre un cuerpo K, $\{v_1, v_2, \ldots, v_n\}$ una base de un subespacio vectorial de V, y A un anillo contenido en K. Entonces el retículo $\mathcal{L} \subset V$ generado por la base $\{v_1, v_2, \cdots, v_n\}$ es el conjunto:

$$\mathcal{L}(v_1, v_2, \ldots, v_n) = \{ \sum_{i=1}^{n} a_i \cdot v_i : a_i \in A \} \tag{1}$$

Generalmente se considera que $A = \mathbb{Z}$ y para esta actividad $V = \mathbb{Z}^m$, de forma que el retículo $\mathcal{L}(v_1, v_2, \ldots, v_n)$ definido a partir de una base $v_i \in \mathbb{Z}^m$:

$$\mathcal{L}(v_1, v_2, \ldots, v_n) = \{ \sum_{i=1}^{n} a_i \cdot v_i : a_i \in \mathbb{Z}, \} \tag{2}$$

Por tanto, un retículo siempre se puede generar a partir de una base del espacio vectorial en el que se defina, mediante todas las combinaciones lineales de elementos de esa base. No obstante, lejos de esta formalidad, el mensaje que se quiere transmitir al alumnado es que un retículo no es más que las combinaciones lineales enteras de un conjunto de vectores y que distintas bases pueden dar lugar a un mismo retículo.

Definición 2: En un espacio vectorial $\mathbb{V}$, un producto interno (o producto escalar) es una aplicación tal que:

$$\langle \cdot, \cdot \rangle : \mathbb{V} \times \mathbb{V} \longrightarrow \mathbb{K},$$
$$(u, v) \longmapsto a = \langle u, v \rangle$$

donde $\mathbb{V}$ es un espacio vectorial y $\mathbb{K}$ el cuerpo ordenado sobre el que está definido, siendo $\mathbb{R}$. Esta operación binaria

debe satisfacer las siguientes condiciones siendo $a, b \in \mathbb{K}$ y $u, v, w \in \mathbb{V}$:

- Linealidad tanto por la izquierda como por la derecha. Es decir:

$$\langle au + bv, w\rangle = a \cdot \langle u, w\rangle + b \cdot \langle v, w\rangle \tag{3}$$

y

$$\langle u, av + bw\rangle = \bar{a} \cdot \langle u, v\rangle + \bar{b} \cdot \langle u, w\rangle \tag{4}$$

- Hermiticidad: $\langle u, v\rangle = \overline{\langle v, u\rangle}$
- Definida positiva: $\langle u, u\rangle \geq 0$ y $\langle u, u\rangle = 0 \iff u = 0$, $\forall u \in \mathbb{V}$

Usualmente se suele representar a esta operación por "$\cdot$". Luego, si además consideramos $\mathbb{V} = \mathbb{R}^2$ y $\mathbb{K} = \mathbb{R}$ podremos considerar el producto escalar común que conocen los estudiantes de cursos de ESO y Bachillerato. Además, para conectar esta definición con los retículos sobre el plano, se recuerda la siguiente relación:

$$u \cdot v = \| u \| \cdot \| v \| \cdot cos(\alpha) \tag{5}$$

siendo $u, v \in \mathbb{R}^2$ y α el ángulo formado entre los dos vectores. A continuación se recuerda la definición de la norma, asociada a la Eq. 5. Para profundizar aún más sobre ello se puede consultar [12].

Definición 3: Un espacio vectorial $\mathbb{V}$ se denomina espacio normado, si para cada $x \in \mathbb{V}$, se define un número real, que se denota por $\| x \|$ y que satisface las siguientes propiedades:

- $\| x \| \geq 0, \ \ \forall x \in \mathbb{V}$ (Positividad)
- $\| x \| = 0 \iff x = 0, \ \ \forall x \in \mathbb{V}$ (Definido)
- $\| \alpha \cdot x \| = |\alpha| \cdot \| x \| \ \ \forall \alpha \in \mathbb{R}$ y $\forall x \in \mathbb{V}$ (Homogeneidad)
- $\| x + y \| \leq \| x \| + \| y \|, \ \ \forall x, y \in \mathbb{V}$ (Desigualdad triangular)

La cantidad $\| x \|$ es conocida como la norma de x. Generalmente, se denota como $(\mathbb{V}, \cdot)$ al espacio vectorial normado. En el caso de que no se verifique la segunda condición de la Def. 3 pero sí las restantes, se dice que $\| \cdot \|$ es una seminorma.

A partir de la Eq. 5 sabemos que si $\alpha = (2k + 1) \cdot \frac{\pi}{2}$, con $k \in \mathbb{Z}$ entonces $u \cdot v = 0$. Es en este instante donde se relaciona los retículos con el producto escalar y con la ortogonalidad, pudiendo así trabajar con retículos de bases ortogonales.

Definición 4: Dado un espacio vectorial $\mathbb{V}$ sobre un cuerpo $\mathbb{K}$ con un producto interno $\langle \cdot, \cdot \rangle$, dos vectores u y v en $\mathbb{V}$ se dicen ortogonales si su producto interno es igual a cero, es decir:

$$\langle u, v\rangle = 0 \tag{6}$$

A continuación, para poder enlazar los retículos con bases ortogonales y el concepto tanto de sistema de referencia afín como cartesiano, nos basaremos en [13], donde se pueden consultar todas las definiciones y ejemplos necesarios.

Definición 5: Una colección de puntos $\{p_0, p_1, \ldots, p_k\}$, con $k \in \mathbb{N}$ en un espacio afín $\mathcal{A}$ se dice afínmente independiente si los vectores $\{\overrightarrow{p_0p_1}, \overrightarrow{p_1p_2}, \ldots \overrightarrow{p_{k-1}p_k}\}$ son linealmente independientes.

Definición 6: Dado un espacio afín $\mathcal{A}$ con $dim(\mathcal{A}) = n$, siendo $n \in \mathbb{N}$, un sistema de referencia $\mathcal{R}$ en $\mathcal{A}$ es un sistema de referencia ordenado $\{p_0, p_1, \ldots, p_n\}$ de $n + 1$ puntos afínmente independientes o equivalentemente satisfaciendo:

$$\langle \{p_0, p_1, \ldots, p_n\} \rangle = \mathcal{A} \tag{7}$$

Una vez definido un sistema de referencia afín, se puede definir el sistema de referencia afín euclidiano en $\mathbb{R}^n$ o comúnmente llamado el usual o cartesiano.

Definición 7: En el espacio afín euclidiano $\mathbb{R}^n$ dotado de estructura afín canónica, el sistema de referencia

$$\mathcal{R}_0 = \{(0, 0, \ldots, 0), B_0\} \tag{8}$$

donde $B_0 = \{(1, 0, 0, \ldots, 0), \ldots, (0, 0, 0, \ldots, 1)\}$ es la base canónica de $\mathbb{R}^n$, esto es, que todo punto $x \in \mathbb{R}^n$ se puede expresar en términos de $\mathcal{R}_0$

En general, en el aula se trabaja con el sistema de referencia definido por $\{(0, 0), (1, 0), (0, 1)\}$ en el caso del plano, y $\{(0, 0, 0), (1, 0, 0), (0, 1, 0), (0, 0, 1)\}$ en el espacio. Cuando se trata de transmitir conceptos abstractos como la Def. 6 en el aula, el objetivo es enseñar que el producto escalar, ya sea 0 o distinto de 0, permite la creación de sistemas de referencia cartesianos o, alternativamente, la imposición de un sistema de referencia afín utilizando los puntos y vectores disponibles. Esto se hace con el fin de construir retículos que pueden tener bases ortogonales (un escenario usual) o bases no ortogonales. A través de este enfoque, los alumnos no solo comprenden cómo construir ejes perpendiculares que sigan una unidad de medida, sino que también aprenden a construir ejes que no sean perpendiculares utilizando puntos arbitrarios.

Por último, y a raíz de la Def. 8 introducimos la distancia taxi, la cuál sobre el plano puede tener una representación geométrica similar a la de los retículos. Este concepto permite tanto al docente como al alumnado agrupar las definiciones anteriores y acercarlas a la realidad, hacerlas un poco más tangibles.

Definición 8: La distancia taxi, también conocida como distancia de Manhattan o distancia rectilínea, es una métrica utilizada en geometría para calcular la distancia entre dos puntos en un espacio euclidiano con coordenadas rectangulares. Formalmente, la distancia taxi entre dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ en un plano euclidiano se expresa como:

$$d_{\text{taxi}}(P, Q) = |x_1 - x_2| + |y_1 - y_2| \tag{9}$$

Donde $|x_1 - x_2|$ representa la diferencia absoluta entre las coordenadas horizontales de los puntos y $|y_1 - y_2|$ representa la diferencia absoluta entre las coordenadas verticales de los puntos. La suma de estas diferencias absolutas proporciona la distancia taxi entre los dos puntos. En la Fig. 2 se puede observar un ejemplo ilustrativo sobre el siguiente concepto.

III. Actividad de GeoGebra

Como se mencionó anteriormente, la actividad desarrollada en [6] puede tener dos propósitos: emplear los principios matemáticos enseñados en Secundaria y Bachillerato para introducir la criptografía post-cuántica a los estudiantes, o utilizar la criptografía post-cuántica como herramienta para mejorar la comprensión de diversos conceptos matemáticos abordados en el aula.

Luego, a la hora de presentar dicho recurso, no es necesario que se impartan los contenidos matemáticos descritos en el

Figura 2. Ejemplo de la aplicación de la distancia taxi, [14]

presente trabajo. En cambio, la actividad puede servir como una manera alternativa de introducir esos conceptos o como un complemento motivador para reforzar los temas ya abordados en el aula.

Una breve introducción a la teoría de retículos

La actividad de GeoGebra titulada "Una breve introducción a la teoría de retículos", tiene por objetivo ofrecer nociones sobre la matemática subyacente a la criptografía post-cuántica, a través de breves definiciones y diferentes ejercicios, tanto de respuesta corta como respuesta larga y abierta.

En un primer instante, se presenta el por qué de la criptografía post-cuántica y la amenaza de la computación cuántica, de modo que el alumno sienta curiosidad por el tema. Acto seguido, se hace una comparación entre dos definiciones de retículo, tanto informal (ver Fig. 3) como formal (Def. 1).

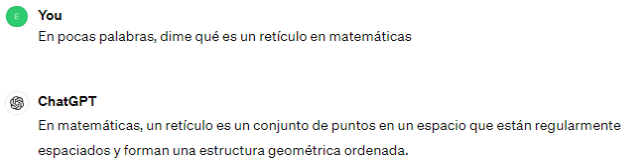

Figura 3. Definición informal de retículo

ChatGPT es una herramienta relativamente novedosa. Se ha elegido utilizarla en esta actividad (en particular al principio, para no perder el interés del lector) con el objetivo de generar un mayor compromiso por parte del alumnado. Dado que esta herramienta suele estar asociada a un estigma que la aleja de los entornos educativos convencionales, se busca presentar una definición atractiva que despierte el interés y la participación de los estudiantes.

Una vez se han dado las definiciones correspondientes, para reforzar estos conceptos se presentan ejemplos simples, representaciones dinámicas como se pueden ver en Fig. 4 y Fig. 5 y preguntas cortas y abiertas, como se puede observar en Fig. 6 y Fig. 7.

Una de las principales ventajas de estas representaciones dinámicas, es que el propio estudiante puede hacer variar los vectores y así ver reflejados los cambios. De esta manera, mediante el aprendizaje interactivo los estudiantes pueden manipular los conceptos y se involucran de manera directa

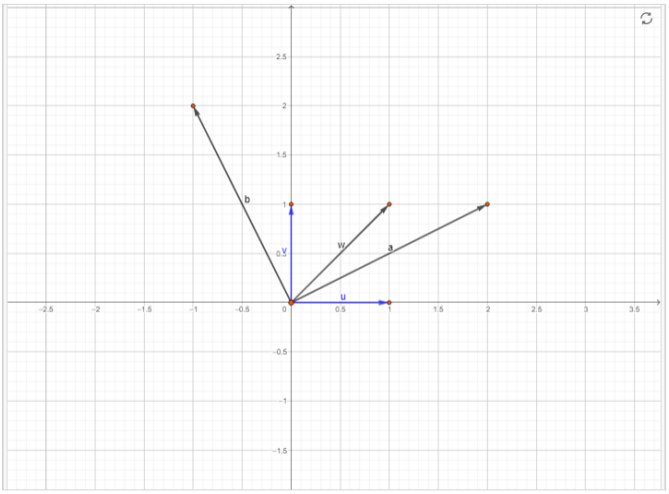

Figura 4. Primer ejemplo de los elementos de un retículo

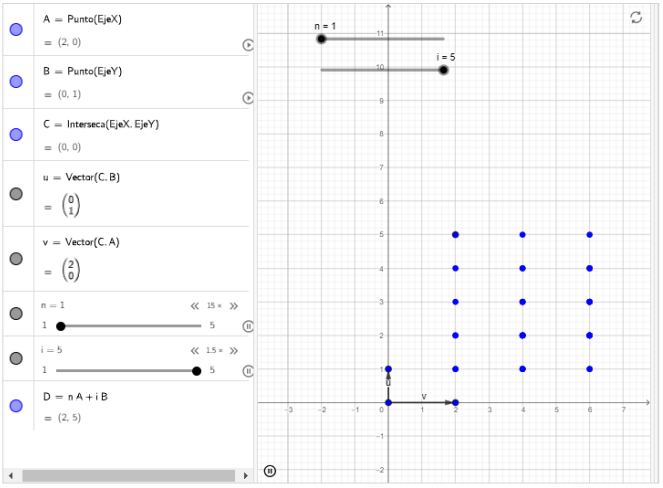

Figura 5. Ejemplo dinámico de los elementos de un retículo

en la construcción de su conocimiento. Es por ello, que también se propone la construcción de sus propios retículos en preguntas abiertas.

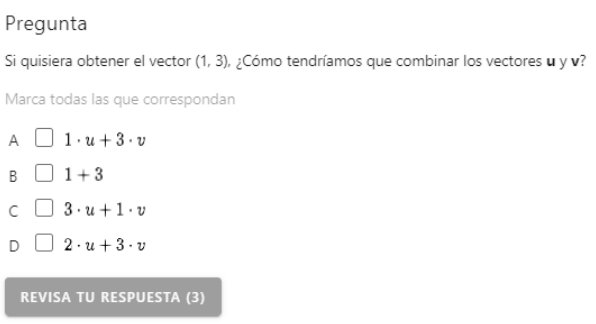

Figura 6. Primera pregunta corta

Además, como se puede observar en Fig. 8 se emplean también las representaciones dinámicas para demostrar de manera empírica que el único retículo nulo es aquel compuesto por los vectores nulos. De manera alternativa, el estudiante

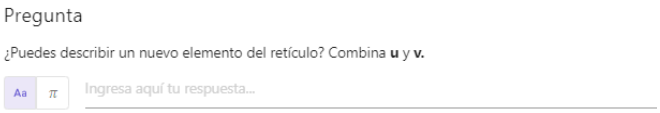

Figura 7. Pregunta abierta en la actividad

aprende de forma interactiva que al utilizar vectores no nulos con coeficientes infinitos, el retículo resultante será infinito.

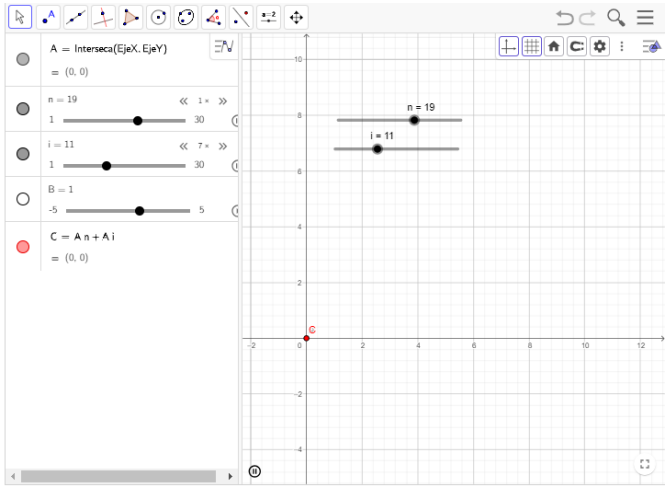

Figura 8. Retículo compuesto por vectores cero

En la segunda parte de la actividad, se busca intentar relacionar los conceptos mencionados anteriormente de forma que el lector pueda llegar a comprender sus conexiones con éxito. En este punto se hace mención al producto escalar entre dos vectores y derivando del mismo, la ortogonalidad. Así mismo, como se puede ver en Fig. 9 también se introducen los sistemas de referencia afines y usuales.

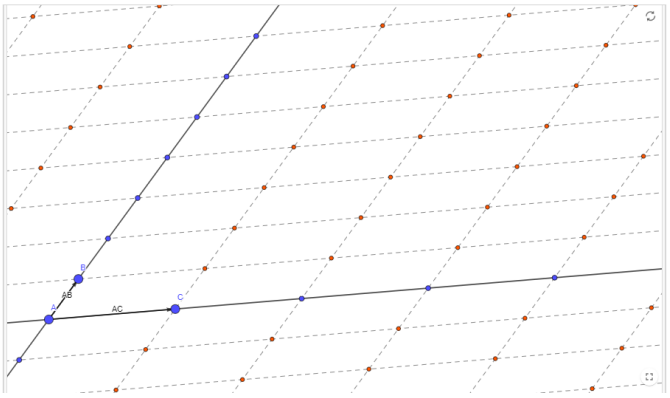

Figura 9. Retículo no ortogonal

Para completar la actividad y proporcionarle una conclusión más integradora y práctica, se mencionan las diversas ramas de las matemáticas, con especial énfasis en la geometría y la topología. Este enfoque nos conduce a la ya mencionada distancia taxi en Def. 8. Para este ejemplo práctico se ha propuesto una imagen de la ciudad de Barcelona con vista aérea, pues esta destaca por su disposición urbana, ordenada y representativa. Se puede observar en Fig. 10.

Figura 10. Barcelona desde el aire

Para aprovechar las capacidades de GeoGebra, se integra la Fig. 10 en el software y se superpone elementos de un retículo en cada intersección tanto de las calles verticales como horizontales (ver Fig. 11. Esto permite plantear al estudiante cómo llegar de un punto A a un punto B moviéndose únicamente a través de los puntos del retículo. De esta manera, se evidencia que la solución no es trazar una línea recta entre A y B, sino que existen múltiples rutas posibles, resaltando así la naturaleza no única de la solución y la aplicación directa y tangible de los conceptos aprendidos.

IV. Diseño de evaluación

Un primer estudio sobre el impacto del recurso didáctico propuesto podría realizarse con varias poblaciones, como puede ser un grupo de profesores y un grupo de alumnos. De este modo, podría medirse con más precisión el alcance del trabajo desarrollado, consiguiendo la opinión de los profesionales que imparten el recurso y la opinión de quienes lo reciben, a esta metodología la denominamos “Enfoque de Doble Audiencia”. La metodología es la siguiente:

- Se llevan a cabo varios cuestionarios durante el proceso: un primer cuestionario demográfico para comprender la composición de la población, un segundo cuestionario intermedio y un último cuestionario para evaluar el progreso a lo largo del taller.

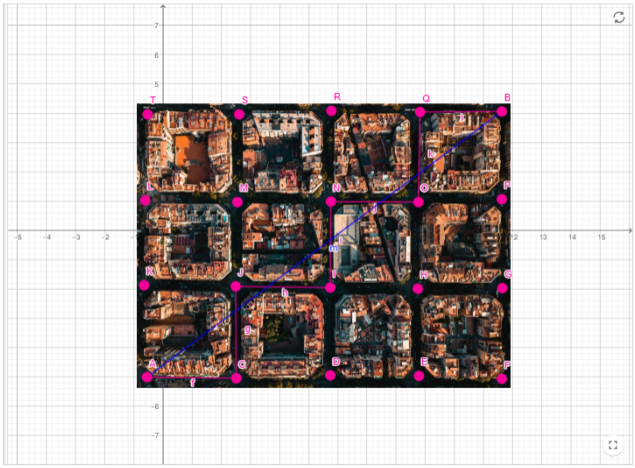

Figura 11. Distancia taxi sobre un retículo

- Después del cuestionario inicial, se sugiere realizar una charla donde se exponen el desarrollo y los fundamentos teóricos de la criptografía post-cuántica. Tras su finalización, la cual puede tener una duración variable de 20 a 30 minutos, se administra el cuestionario intermedio.
- En la segunda parte del taller, se presenta el recurso didáctico. Dependiendo de la audiencia (docentes o estudiantes), esta parte adopta un enfoque más pragmático, donde se muestra la utilidad de la herramienta para una mejor comunicación y explicación de los conceptos matemáticos, o más formativo, donde simplemente se exponen las relaciones y los términos matemáticos directamente al alumnado.

Las preguntas que se proponen para incluir en los cuestionarios son las siguientes:

- Preguntas a los docentes:
  - **Q1** ¿Sabes algo sobre las bases de la criptografía post-cuántica?
  - **Q2** ¿Te parecen sencillos los conceptos asociados a la criptografía post-cuántica?
  - **Q3** ¿Consideras que la criptografía post-cuántica puede tener interés en el aula?
  - **Q4** ¿Consideras que la criptografía post-cuántica puede aplicarse en el aula para explicar conceptos matemáticos?
  - **Q5** ¿Te parece útil la actividad desarrollada para comprender algunos conceptos de las matemáticas?
  - **Q6** ¿Te gustaría aplicar estas actividades en el aula?
  - **Q7** Indica el concepto que te parezca más interesante:
    - Retículo
    - Paralelepípedo
    - El problema de aprendizaje sobre errores
    - El problema del vector más corto
    - El problema del vector más cercano
    - Criptografía de clave pública
  - **Q8** ¿Qué concepto crees que sale más reforzado tras la actividad?
    - Producto escalar
    - Sistema de referencia
    - Combinaciones lineales de vectores
    - Ortogonalidad
    - Determinantes y matrices
- Preguntas al alumnado:
  - **Q1** ¿Sabes algo sobre las bases de la criptografía post-cuántica?
  - **Q2** ¿Te parecen sencillos los conceptos asociados a la criptografía post-cuántica?
  - **Q3** ¿Te parece interesante la criptografía post-cuántica?
  - **Q4** ¿Qué nivel de abstracción consideras que tiene la criptografía post-cuántica?
  - **Q5** ¿Te parece útil la actividad desarrollada para comprender algunos conceptos de las matemáticas?
  - **Q6** ¿Te han ayudado las actividades de GeoGebra a comprender los conceptos?
  - **Q7** Indica el concepto que te parezca más interesante:
    - Retículo
    - Paralelepípedo
    - El problema de aprendizaje sobre errores
    - El problema del vector más corto
    - El problema del vector más cercano
    - Criptografía de clave pública
    - Polinomios
  - **Q8** ¿Qué concepto consideras que sale más reforzado tras la actividad?
    - Producto escalar
    - Sistema de referencia
    - Combinaciones lineales de vectores
    - Ortogonalidad
    - Determinantes y matrices

Se sugiere responder a las preguntas **Q1** - **Q6** utilizando la escala de Likert, mientras que las preguntas **Q7** y **Q8** se abordan con respuestas múltiples. Cabe destacar que varias de las preguntas se repiten a lo largo de los cuestionarios que se realizan, de ese modo se permite medir la evolución y el impacto a lo largo del taller.

Conclusiones

Este trabajo ha presentado una metodología innovadora para la enseñanza de los retículos utilizando GeoGebra. Durante la actividad, se han explorado conceptos fundamentales del tema, como la estructura del propio retículo, mediante ejemplos dinámicos, preguntas de distinto tipo y su relación con otros conceptos aparentemente disímiles, como el producto escalar, la ortogonalidad, referencias afines y la distancia taxi. Esta investigación se distingue por su originalidad, al no haber sido precedida por trabajos similares. Como trabajo futuro, se planea llevar a cabo talleres con estudiantes y docentes para recopilar resultados estadísticos que permitan evaluar la utilidad de la actividad, así como contemplar posibles modificaciones en la misma para su mejora continua.

Agradecimientos

Este trabajo ha sido posible gracias al acuerdo entre Atlantis SL y la Universidad de La Laguna, y a las Cátedras de Ciberseguridad patrocinadas por Binter, y por INCIBE mediante iniciativa realizada en el marco de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiada por

la Unión Europea (Next Generation). Además forma parte del proyecto PID2022-138933OB-I00 financiado por MCIN/AEI/ 10.13039/501100011033/FEDER, UE.

REFERENCIAS

[1] R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. Schanck, P. Schwabe, G. Seiler, D. Stehlé, "CRYSTALS-Kyber algorithm specifications and supporting documentation", NIST PQC Round, vol. 2, no. 4, pp. 1–43, 2019.

[2] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, D. Stehlé, "CRYSTALS-Dilithium: Algorithm specifications and supporting documentation (version 3.1)", NIST Post-Quantum Cryptography Standardization Round, vol. 3, 2021.

[3] National Institute of Standards and Technology (NIST), "FIPS 203 (Draft) Module-Lattice-based Key-Encapsulation Mechanism Standard", 2023, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.203.ipd.pdf

[4] National Institute of Standards and Technology (NIST), "FIPS 204 (Draft) Module-Lattice-Based Digital Signature Standard", 2023,https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.204.ipd.pdf

[5] P. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Prest, T. Ricosset, G. Seiler, W. Whyte, Z. Zhang, "Falcon: Fast-Fourier lattice-based compact signatures over NTRU." *Submission to the NIST's post-quantum cryptography standardization process*, 2018, https://www.di.ens.fr/~prest/Publications/falcon.pdf

[6] Pérez, É. "Una breve introducción a la teoría de retículos", 2024. Disponible en: https://www.geogebra.org/m/cm2e42fk

[7] GeoGebra. (s/f). GeoGebra. Disponible en: https://www.geogebra.org/

[8] N. Arbain, Nurbiha A. Shukor. "The Effects of GeoGebra on Students Achievement". *Procedia - Social and Behavioral Sciences*, vol. 172, pp. 208–214, 2015. Disponible en: https://doi.org/10.1016/j.sbspro.2015.01.356

[9] Dogan, M., İçel, R. "The role of dynamic geometry software in the process of learning: GeoGebra example about triangles". *Journal of Human Sciences*, vol. 8, pp. 1441—1458. Disponible en: https://www.j-humansciences.com/ojs/index.php/IJHS/article/view/1547

[10] Ministerio de Educación y Formación Profesional. "Real Decreto 217/2022, de 29 de marzo, por el que se establece la ordenación y las enseñanzas mínimas de la Educación Secundaria Obligatoria". *BOE*, núm. 76, de 30 de marzo de 2022. Referencia: BOE-A-2022-4975.

[11] Instituto Nacional de Evaluación Educativa. "PISA 2022. Programa para la Evaluación Internacional de los Estudiantes. Informe español.",2023. Disponible en: https://acortar.link/afkH6B

[12] Águila Hernández, E.J., "Algunos Tópicos en Teoría de Aproximación", Trabajo fin de grado, Universidad de La Laguna, 2023. [Online], Disponible en: https://riull.ull.es/xmlui/handle/915/33370

[13] López, Francisco J. Geometría III. Departamento de Geometría y Topología, Universidad de Granada. Granada, España. [Online], Disponible en: https://www.ugr.es/~fjlopez/_private/Geometria_III.pdf

[14] El País. "Manhattan, distancias y "el juicio de Pitágoras. Ciencia/Materia, 2016. Disponible en: https://acortar.link/8BQt7X

A.5. Artículo pre-aceptado “Using GeoGebra to Learn the Basics of Post-Quantum Cryptography”

Édgar Pérez-Ramos, Pino Caballero-Gil, Héctor Reboso-Morales
“GeoGebra para introducir los fundamentos de la criptografía basada en retículos”
Frontiers in Education (FIE)
Washington DC, USA. 13-16 October, 2024
Indexado en GII-GRIN con GGS Rating Class 3 B-
Indexado en Computing Research and Education, CORE: C
Indexado en LiveSHINE: c

WIP: Using GeoGebra to Learn the Basics of Post-Quantum Cryptography

Édgar Pérez-Ramos, Pino Caballero-Gil, Héctor Reboso-Morales
Department of Computer Engineering and Systems
University of La Laguna
Tenerife, Spain
alu0101207667@ull.edu.es, pcaballe@ull.edu.es, hreboso@ull.es

***Keywords*—Mathematics, Educational software, High school.**

EXTENDED ABSTRACT

This work in progress research paper describes a proposal to use the GeoGebra environment as a useful didactic and conceptual tool to introduce the basic concepts of the increasingly relevant area of Post-Quantum Cryptography (PQC).

Currently, cryptography plays an essential role in all communication technologies because it is necessary to protect the security of systems and messages. Among the most relevant cryptographic techniques, encryption is used to protect secrets while digital signature is used to verify the authenticity and integrity of digital documents and messages. However, with the possible future deployment of quantum computing, it has been discovered that many cryptographic algorithms currently used in different technologies, such as RSA or ECDSA, will be totally vulnerable. Hence the need arises for new cryptographic schemes that are resistant to the so-called quantum threat.

In 2016 the race for Post-Quantum Cryptography began. That year, the National Institute of Standards and Technology (NIST) began a process of selecting cryptographic schemes that could withstand quantum computers. Finally, the selected algorithms identified as final standards were [1]:

- Encryption: CRYSTALS-Kyber
- Digital signature: CRYSTALS-Dilithium, FALCON, and SPHINCS+

The common factor of most quantum-resistant algorithms (CRYSTALS-Kyber, CRYSTALS-Dilithium and FALCON) is their basis in lattice theory. This, introducing the basic concepts of lattice theory can now be considered a necessity at different educational levels for the best preparation of future engineers and scientists.

Formally, a lattice can be defined as follows:
Definition. Let V be a vector space over a field K, $\{v_1, v_2, , v_n\}$ a basis of a subspace of V, and A a ring contained in K. Then the lattice $\mathcal{L} \subset V$ generated by the basis $\{v_1, v_2, \cdots, v_n\}$ is the set:

$$\mathcal{L}(v_1, v_2, , v_n) = \{\sum_{i=1}^{n} a_i \cdot v_i : a_i \in A\} \quad (1)$$

This formal definition of what a lattice is can be very complex even for mathematics and engineering students, so it would be recommended to introduce the concept in a didactic way in high school. One of the main current challenges in the field of Post-Quantum Cryptography is its totally innovative nature, which is accompanied by an almost absolute scarcity of information and teaching resources, especially when compared to the number of proposals to introduce concepts of classical cryptography [2]. For this reason, the purpose of this work is to create teaching material about lattice, which allows introducing the basic concepts to high school students and those beginning their university studies. Thus, various interactive activities in Geogebra are proposed here, which allow students to experiment with different resources, pose questions and establish connections between lattice and the real world to assimilate the concept in both a theoretical and practical way. Furthermore, in the designed activities the mathematical concept of lattice is linked to others that form part of the curriculum taught in high school, such as: scalar product, orthogonality, affine and Cartesian reference systems, basic concepts of topology, etc.

Furthermore, in order to provide a more practical approach to this work, the creation of an environment in Google Colaboratory is used as a support tool. In this way, the student is able to directly input the necessary data to construct the lattice, visualize its elements, calculate the associated determinant, the volume of its parallelepiped, and many other relevant aspects.

To develop the necessary study of the results of the proposal, several face-to-face sessions have been planned with students from different high school courses, to collect statistical data that allow evaluating the impact and effectiveness of the developed teaching resources.

ACKNOWLEDGMENT

This research is possible thanks to the agreement between Atlantis SL and the University of La Laguna, the PID2022-138933OB-I00 project, and the Cybersecurity Chair financed by NextGenerationEU and Recovery and Resilience Facility.

REFERENCES

[1] NIST: "PQC Standardization Process: Announcing Four Candidates to be Standardized, Plus Fourth Round Candidates", *NIST News*, 2022. [Online]. Available in: https://acortar.link/WQeR80

[2] Deeb, F.A., Hickey, T.J. Teaching introductory cryptography using a 3D escape-the-room game. IEEE Frontiers in Education Conference. 2019.