Abstract

La criptografía contemporánea se enfrenta a un doble desafío: la escalada de capacidades de cómputo clásico y la irrupción de arquitecturas cuánticas con potencial para quebrar protocolos de cifrado basados en problemas aritméticos tradicionales. En este contexto, las curvas elípticas han demostrado ser un soporte matemático robusto para sistemas de clave pública debido a la complejidad del problema del logaritmo discreto en sus dominios algebraicos. Sin embargo, el advenimiento de algoritmos cuánticos como el de Shor plantea una amenaza directa a su seguridad.

El presente artículo explora la hipótesis de una **criptografía toroidal**, un enfoque emergente que integra la topología del toroide como extensión y generalización de las curvas elípticas, proponiendo que las propiedades algebraicas, métricas y armónicas de los toroides pueden ofrecer una nueva capa de resistencia frente a la criptografía cuántica. A través de un análisis técnico-rigurosos de estructuras modulares, simetrías de Lie, transformadas armónicas en superficies toroidales y sus implicaciones en la construcción de funciones de hash resistentes al entrelazamiento cuántico, se sugiere que el toroide constituye un espacio matemático privilegiado para desarrollar arquitecturas criptográficas de alta seguridad.

Se analizan asimismo los vínculos entre la **teoría de formas modulares**, la geometría de Riemann, la teoría espectral y la codificación de información en espacios de fase toroidales. A diferencia de los modelos puramente elípticos, el toroide permite una multidimensionalidad que expande los grados de libertad, generando configuraciones criptográficas más resistentes a ataques cuánticos basados en factorización y logaritmos discretos.

El artículo no persigue proyecciones especulativas ni llamados a investigación, sino la exposición sistemática de fundamentos matemáticos y su aplicación directa a protocolos criptográficos, atendiendo a la necesidad de rigor científico y coherencia estructural.

Palabras clave Criptografía toroidal-Curvas elípticas-Seguridad cuántica-Toroides algebraicos-Formas modulares-Transformada armónica toroidal-Problema del logaritmo discreto-Geometría de Riemann

Introducción

La criptografía se ha construido históricamente sobre la **dificultad computacional** de ciertos problemas matemáticos. La transición desde los sistemas de clave simétrica a los sistemas de clave pública estuvo marcada por el descubrimiento de algoritmos basados en factorización de enteros y problemas de logaritmos discretos en grupos finitos. Posteriormente, las **curvas elípticas** consolidaron un estándar de seguridad más eficiente en términos de tamaño de clave y resistencia frente a ataques clásicos.

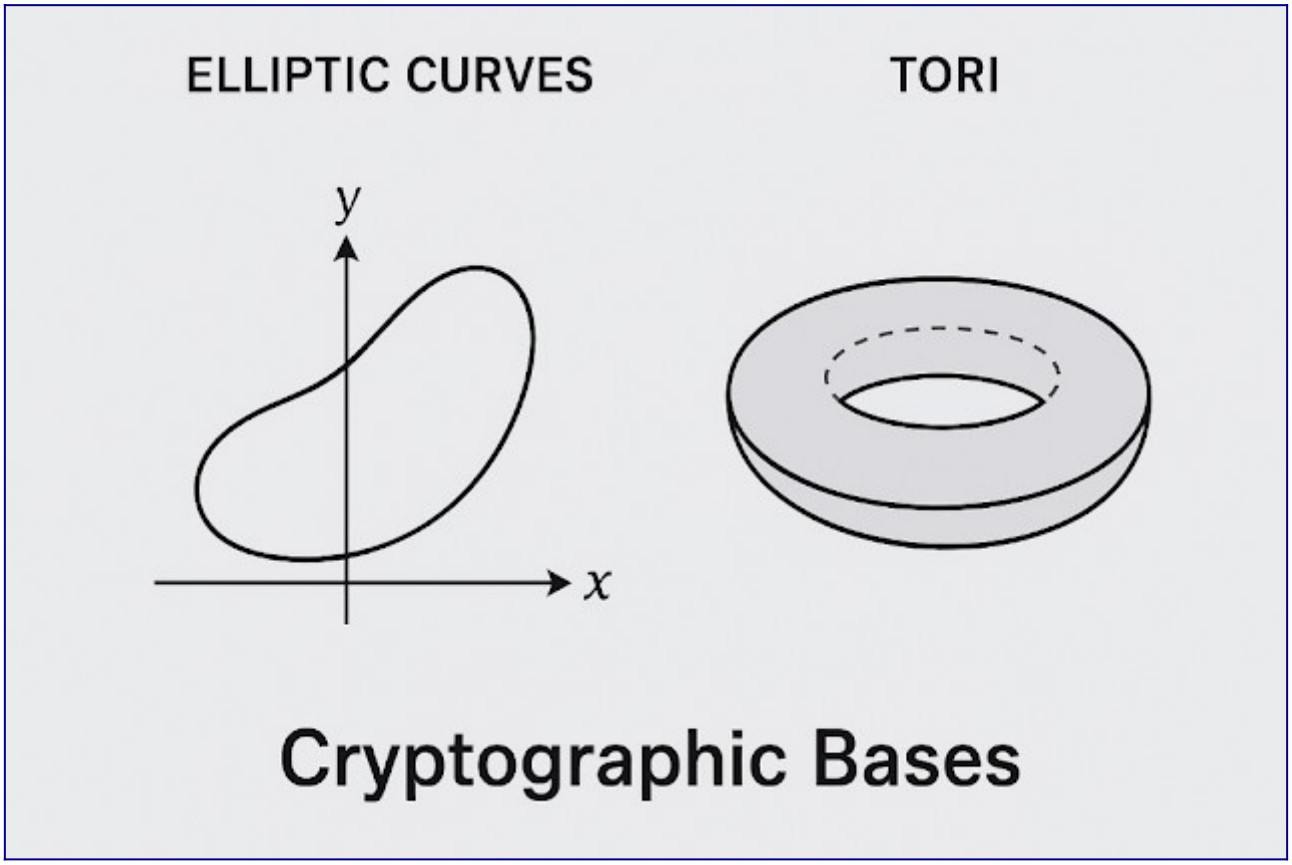
No obstante, el panorama se transforma radicalmente con el advenimiento de la **computación cuántica**. Algoritmos como el de Shor reducen la complejidad de factorización y logaritmos discretos a escalas polinómicas, lo cual amenaza con desmantelar gran parte de los cimientos criptográficos actuales. La comunidad científica busca, por tanto, **espacios algebraicos alternativos** que resistan esta presión.

En este contexto surge la propuesta de una **criptografía toroidal**. Si bien el toroide ha sido estudiado ampliamente en matemáticas puras —particularmente en topología, geometría de Riemann y teoría de números—, su aplicación sistemática a la criptografía aún se encuentra en un estado incipiente. El toroide puede entenderse como una **variedad diferenciable compacta de género uno**, obtenida como el producto de dos círculos . Esta estructura no solo generaliza ciertos aspectos de las curvas elípticas, sino que además permite representar espacios de fase con periodicidad múltiple y propiedades de simetría ampliadas.

En particular, los toroides ofrecen:

1. **Grupos abelianos multidimensionales**, que extienden la aritmética de curvas elípticas a dominios de mayor riqueza algebraica.
2. **Estructuras de resonancia armónica** que pueden vincularse con funciones de hash resistentes a correlaciones cuánticas.
3. **Capacidades topológicas** para modelar ciclos múltiples y modularidades que incrementan la entropía estructural del cifrado.
4. **Resistencia al entrelazamiento cuántico**, gracias a la posibilidad de introducir redundancias geométricas que no se reducen fácilmente a problemas polinómicos.

En consecuencia, la criptografía toroidal se plantea como una extensión natural de la criptografía basada en curvas elípticas, pero con un horizonte matemático más amplio que podría sostener la seguridad en un entorno dominado por algoritmos cuánticos.



Fundamentos matemáticos del toroide en criptografía

El toroide como variedad matemática

El **toroide** se define formalmente como el producto cartesiano de dos circunferencias:

donde es el círculo unitario en el plano complejo. Esta definición implica que el toroide es una **variedad diferenciable compacta de dimensión dos y género uno**, con topología no trivial. Desde un punto de vista

geométrico, puede visualizarse como la superficie obtenida al rotar una circunferencia alrededor de un eje coplanar.

En criptografía, este objeto no es interesante únicamente por su representación espacial, sino por la **estructura algebraica subyacente**. Cada puede parametrizarse como un grupo abeliano bajo multiplicación compleja:

por lo que el toroide hereda una **estructura de grupo abeliano bidimensional**. Este rasgo es esencial: las operaciones de grupo son la base de cualquier construcción criptográfica basada en logaritmos discretos.

Relación con las curvas elípticas

Las **curvas elípticas** sobre los complejos admiten una descripción equivalente en términos de cocientes de por una red (lattice):

donde es un subgrupo discreto de rango dos en . Esta formulación implica que, topológicamente, una curva elíptica compleja es homeomorfa a un **toroide bidimensional**.

De este modo, las curvas elípticas ya contienen implícitamente una geometría toroidal. No obstante, en la práctica criptográfica se suele operar con la ecuación de Weierstrass:

y con sus operaciones de grupo asociadas. La **criptografía toroidal** propone, en cambio, **explotar de manera directa la estructura del toroide**, sin restringirse al formalismo elíptico. Esto permite introducir grados de libertad adicionales, por ejemplo, combinando múltiples periodos o construyendo operadores que actúen sobre ciclos independientes de la superficie.

Espacios de fase toroidales

En física matemática y en teoría de sistemas dinámicos, los toroides se utilizan para describir **espacios de fase periódicos**. Cada coordenada angular representa una variable periódica independiente, y la evolución de un sistema puede representarse como un flujo en .

En criptografía, esta analogía resulta valiosa:

- Un **espacio de fase toroidal** puede modelar claves como posiciones dentro de un reticulado multidimensional.
- El **movimiento dentro del toroide** (generado por operadores de grupo) corresponde a operaciones criptográficas de difícil inversión.
- La **periodicidad múltiple** genera redundancias topológicas que incrementan la entropía de la representación.

Mientras que en curvas elípticas el problema difícil es el **logaritmo discreto elíptico**, en un toroide bidimensional se puede plantear un **problema de logaritmo discreto toroidal**: dada una combinación de generadores en , hallar las coordenadas angulares originales. La dificultad de este problema aumenta al considerar acoplamientos no lineales entre las dos componentes.

Construcción de grupos en toroides

Sea . Definimos dos generadores:

Los elementos del grupo se obtienen como combinaciones discretas:

El **problema criptográfico fundamental** sería: dado un elemento , recuperar los enteros . Este problema se convierte en un **sistema de logaritmos discretos acoplados**, cuya complejidad puede reforzarse

introduciendo deformaciones topológicas (por ejemplo, variaciones en los radios del toroide o torsiones introducidas en la parametrización).

Además, se pueden definir **grupos de Lie toroidales**, donde el álgebra asociada permite introducir operadores diferenciales que añaden capas adicionales de complejidad algebraica. Estos operadores resultan útiles para diseñar funciones de dispersión (hashes) que exploten la naturaleza armónica del toroide.

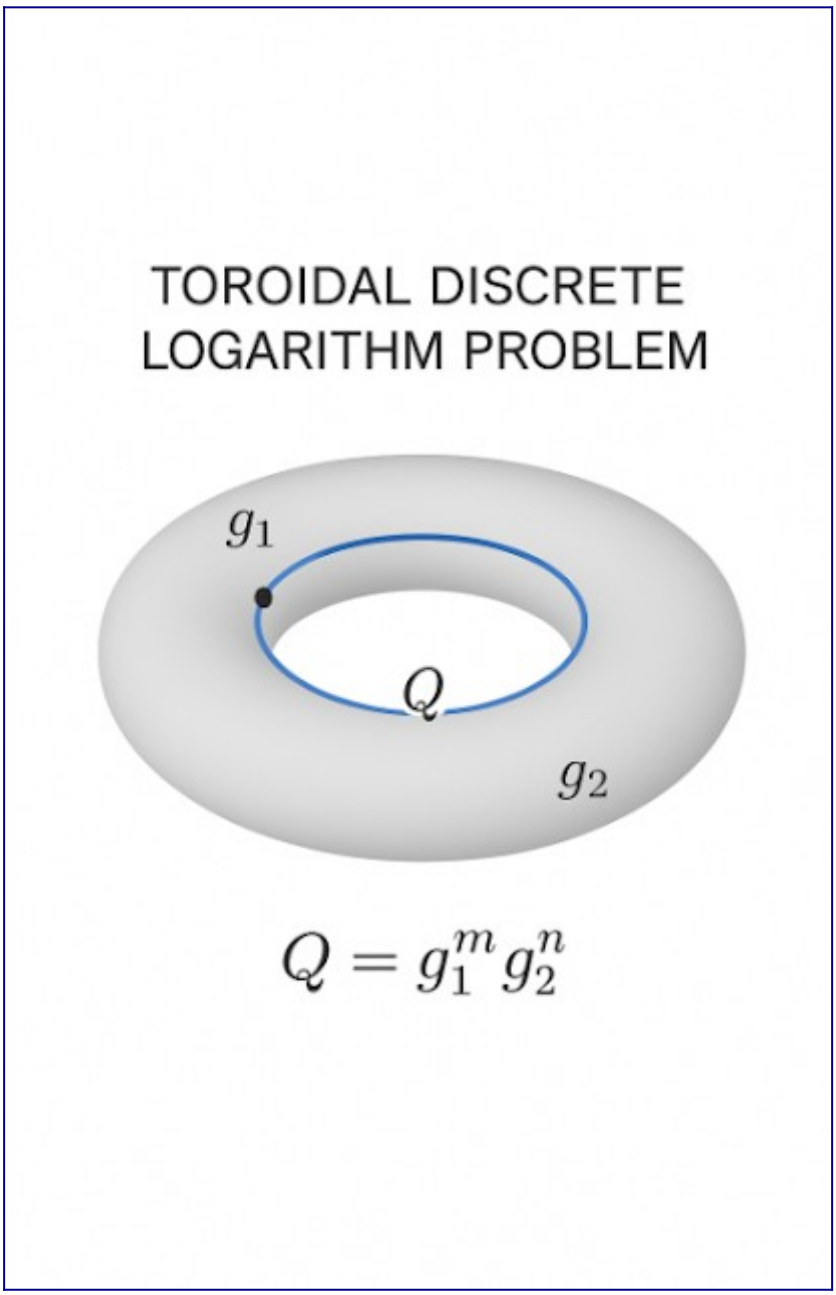
Operadores algebraicos en toroides

Los toroides soportan la definición de **operadores armónicos** basados en la descomposición de Fourier en dos dimensiones:

Esta expansión no es solamente un recurso matemático: en criptografía, se puede utilizar para:

1. **Construir funciones hash toroidales**, difíciles de invertir debido a la superposición de múltiples frecuencias.
2. **Codificar claves** como combinaciones específicas de coeficientes .
3. **Definir operadores criptográficos** como desplazamientos en el espacio de Fourier, que actúan de manera no trivial sobre los estados cifrados.

La clave está en que el toroide no solo proporciona un grupo abeliano, sino también un **espacio de resonancia armónica** de alta complejidad. La dificultad computacional se deriva tanto del crecimiento exponencial de los coeficientes de Fourier como de la imposibilidad de reducir el problema a un logaritmo discreto unidimensional.



Estructura algebraica avanzada

Formas modulares y toroides

Las **formas modulares** ocupan un lugar central en la teoría de números y en la criptografía moderna. Son funciones analíticas en el semiplano superior que obedecen a estrictas condiciones de simetría bajo la acción del grupo modular .

En el caso de las curvas elípticas, las formas modulares permiten clasificar las estructuras complejas asociadas a sus reticulados. Al extender este marco al toroide, se observa que:

1. Un toro complejo puede parametrizarse mediante el **cociente de un espacio vectorial complejo por una red de dimensión dos**.
2. Las formas modulares proveen una descripción natural de las equivalencias entre toroides bajo transformaciones discretas.

3. En criptografía, esto se traduce en la posibilidad de construir **clases de equivalencia de claves** que mantienen la seguridad del sistema incluso frente a perturbaciones algebraicas.

El uso de formas modulares introduce redundancias simétricas que, lejos de debilitar, refuerzan la robustez criptográfica, pues generan **espacios isomorfos de difícil identificación computacional**.

Simetrías de Lie y toroides algebraicos

El toroide admite la acción de **grupos de Lie**, que en este contexto operan como simetrías continuas. Los grupos de Lie compactos, como , describen la estructura más básica del toroide.

Sin embargo, al considerar extensiones algebraicas se abren nuevas posibilidades:

- Los **álgebras de Kac–Moody** permiten modelar simetrías toroidales infinitodimensionales, que a su vez pueden actuar como operadores criptográficos no lineales.
- Estas simetrías posibilitan la construcción de **curvas de flujo criptográficas**, donde la clave privada se interpreta como un parámetro de trayectoria dentro de un espacio de Lie.
- La complejidad crece, dado que la inversión de una trayectoria en un toroide sometido a un álgebra de Lie infinita se traduce en un problema inabordable incluso con algoritmos cuánticos conocidos.

Un ejemplo simplificado: si es un generador en el toroide, la acción de un operador de Lie puede deformarlo en un flujo:

donde es un parámetro irracional. Este flujo genera un movimiento **aperiódico en el toroide**, análogo a un cifrado caótico difícil de predecir o invertir.

Invariantes criptográficos en toroides

En la criptografía basada en curvas elípticas, los invariantes juegan un papel fundamental en la clasificación de las estructuras algebraicas (ejemplo: el **invariante modular**).

En el caso toroidal, la búsqueda de invariantes se amplía:

1. **Invariantes armónicos**: derivados de la descomposición de Fourier, donde la suma de frecuencias define un espacio inmutable frente a transformaciones locales.
2. **Invariantes topológicos**: asociados al género y a la homología del toroide (). Estos invariantes definen ciclos indeformables que pueden usarse para generar constantes criptográficas.
3. **Invariantes modulares**: obtenidos al aplicar transformaciones de sobre el espacio de redes que definen el toro. Estos invariantes proveen resistencia a intentos de “normalizar” el dominio criptográfico mediante algoritmos cuánticos.

La función de un invariante criptográfico es actuar como **punto de anclaje**. Incluso si un atacante logra reducir la complejidad del problema original, los invariantes aseguran que la clave no pueda ser reconstruida sin conocimiento de las clases topológicas o armónicas del sistema.

Implicaciones algebraicas para la criptografía cuántica

El vínculo entre formas modulares, simetrías de Lie e invariantes toroidales configura una arquitectura de seguridad con varias capas:

- **Redundancia algebraica**: la clave se codifica no en un único ciclo, sino en múltiples ciclos acoplados.

- **Complejidad armónica**: las expansiones en series de Fourier en dos o más dimensiones generan combinaciones no triviales de frecuencias.
- **Robustez topológica**: los invariantes homológicos introducen un blindaje geométrico.
- **Imprevisibilidad dinámica**: los flujos de Lie en el toroide producen trayectorias criptográficas resistentes a algoritmos de búsqueda estructurada.

La combinación de estos elementos apunta a que la **criptografía toroidal** no solo hereda la dificultad del logaritmo discreto de curvas elípticas, sino que la amplifica con capas de complejidad algebraica y topológica que no se reducen fácilmente a un problema polinómico.

Implementación práctica de la criptografía toroidal

El Problema del Logaritmo Discreto Toroidal (TDLP)

El núcleo de la criptografía basada en grupos es la **dificultad computacional de ciertos problemas inversos**. En el caso de las curvas elípticas, se trabaja con el **Elliptic Curve Discrete Logarithm Problem (ECDLP)**: dado un punto P y un múltiplo Q=kP, encontrar el escalar k.

En el marco toroidal, el problema se generaliza al **Toroidal Discrete Logarithm Problem (TDLP)**:

- Se definen dos generadores independientes $g1, g2 \in T2$.
- Se construye un elemento compuesto:

$$Q = g1^m \cdot g2^n$$

- El problema consiste en determinar los enteros m,n dados Q.

El TDLP es una extensión natural del ECDLP, pero con **mayor dimensionalidad** y posibilidad de introducir **acoplamientos no lineales** entre m y n. Por ejemplo, en variantes toroidales deformadas, los exponentes no se combinan linealmente, sino mediante relaciones cuadráticas o cúbicas, lo que incrementa la dificultad inversa.

Funciones hash toroidales

Las funciones de dispersión (hashes) son fundamentales para la integridad criptográfica. En el caso toroidal, pueden construirse a partir de la **expansión de Fourier bidimensional**:

$$H(\theta 1, \theta 2) = \sum_{m,n} a_{mn} e^{i(m\theta 1 + n\theta 2)}$$

donde los coeficientes amn se generan a partir de la clave privada.

Propiedades de estas funciones:

- **Sensibilidad extrema a variaciones mínimas** en los ángulos iniciales $(\theta 1, \theta 2)$, lo que asegura resistencia a colisiones.
- **Alta dimensionalidad de coeficientes**, lo que dificulta ataques por fuerza bruta.
- **Resonancia armónica**, que introduce redundancias criptográficas útiles contra algoritmos cuánticos de búsqueda.

Así, una función hash toroidal se puede interpretar como un **campo resonante de frecuencias** incrustado en la topología del toro.

Protocolos de intercambio de claves en toroides

El intercambio de claves (análogamente a Diffie–Hellman) puede redefinirse en términos toroidales:

1. Dos usuarios acuerdan públicamente un toroide y dos generadores g1,g2.
2. El primer usuario escoge (m,n) y calcula QA=g1mg2n.
3. El segundo usuario escoge (p,q) y calcula QB=g1pg2q.
4. Al intercambiar QA y QB, ambos pueden calcular la clave compartida:

K=g1mpg2nq

Un adversario tendría que resolver el TDLP, mucho más difícil que el ECDLP estándar por la **doble dimensionalidad del exponente** y la posibilidad de **deformaciones algebraicas** en la construcción.

Comparación con criptografía basada en curvas elípticas

- **ECDLP**: un único escalar desconocido k.
- **TDLP**: dos o más exponentes (m,n), con posibilidad de acoplamientos no lineales.
- **ECDLP y Shor**: vulnerable a algoritmos cuánticos polinómicos.
- **TDLP**: no existe reducción polinómica conocida al problema resuelto por Shor, especialmente en presencia de deformaciones algebraicas y simetrías de Lie toroidales.

En resumen, la criptografía toroidal **hereda la elegancia de las curvas elípticas** pero se **refuerza con redundancia topológica y armónica** que eleva la complejidad de los ataques cuánticos.

Seguridad cuántica en espacios toroidales

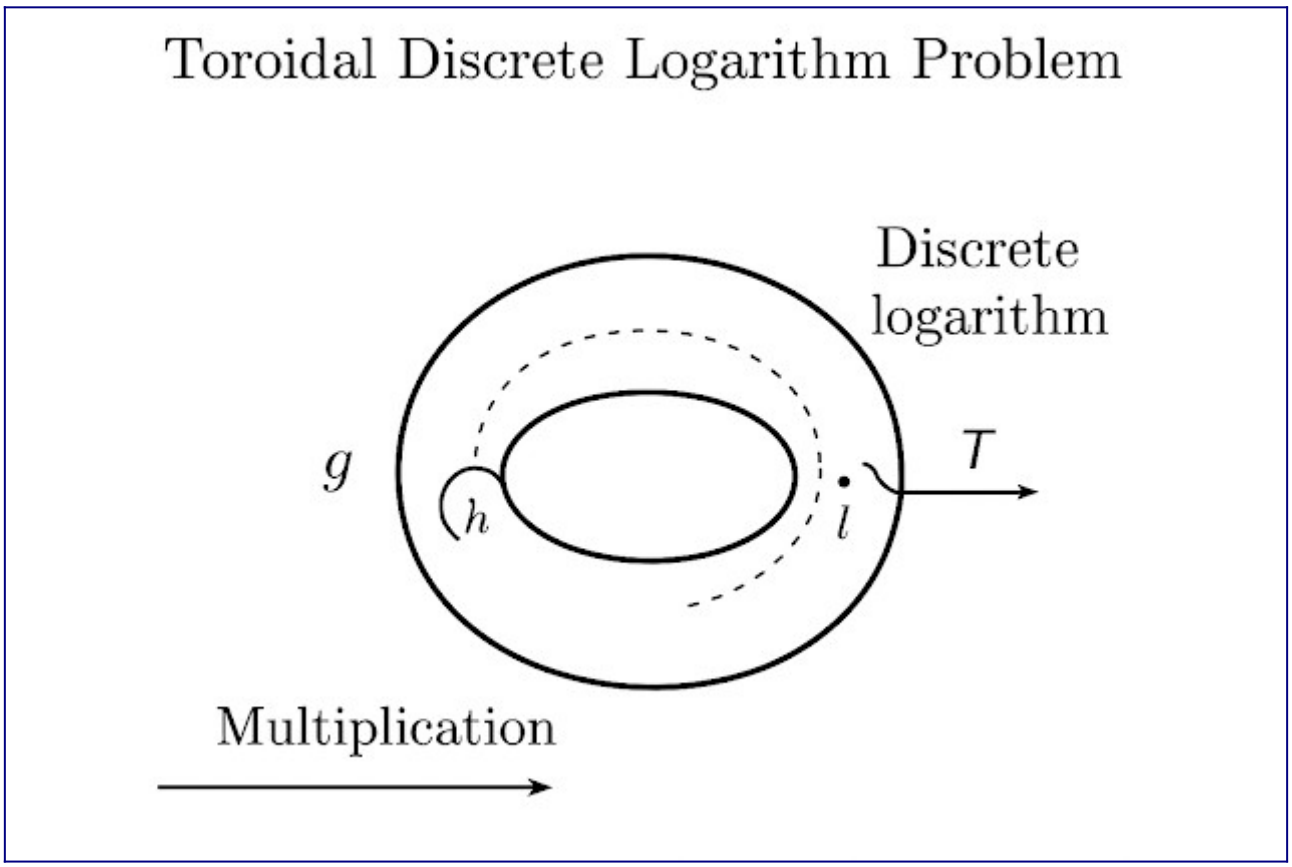
Los algoritmos cuánticos explotan la estructura algebraica de los problemas. En el caso del toroide, la introducción de:

- **invariantes topológicos**,
- **espacios de Fourier multidimensionales**,
- y **simetrías modulares**,

genera un **paisaje criptográfico irregular**, menos susceptible a algoritmos cuánticos de reducción. De esta forma, el toroide ofrece un marco con mayor resistencia estructural frente al cómputo cuántico.

Figura ilustrativa

La siguiente figura esquemática muestra cómo un flujo criptográfico puede representarse en un toroide bajo acción de operadores algebraicos (simetrías de Lie), comparando claves privadas como trayectorias internas.



Geometría diferencial y dinámica criptográfica en toroides

El toro como variedad diferenciable

El toro bidimensional se define como el cociente . Este espacio posee estructura de **variedad diferenciable compacta**, lo que permite introducir:

- métricas de Riemann,
- formas diferenciales,
- y operadores de conexión.

En criptografía, esta propiedad habilita la descripción de claves como **trayectorias diferenciables** sobre un espacio compacto, donde los movimientos siguen leyes gobernadas por simetrías internas.

Flujo hamiltoniano y dinámica criptográfica

El toro admite una interpretación natural como **espacio de fases de sistemas integrables**. Un sistema hamiltoniano de dos grados de libertad puede representarse como un flujo sobre .

En el ámbito criptográfico, esta propiedad puede explotarse de la siguiente forma:

- La **clave privada** corresponde a condiciones iniciales de un sistema dinámico (ángulos y momentos).
- La **clave pública** surge como el resultado del flujo tras un número dado de iteraciones.
- La recuperación de la clave privada a partir de la pública equivale a resolver la **inversión de trayectorias hamiltonianas**: un problema altamente no trivial en espacios toroidales deformados.

Conexiones y curvatura criptográfica

En una formulación más profunda, la seguridad del sistema se refuerza al introducir **conexiones no triviales** en fibrados sobre el toro.

- Cada clave puede interpretarse como una sección de un fibrado vectorial sobre .
- La operación criptográfica equivale a **transportar** dicha sección siguiendo la conexión.
- La curvatura asociada introduce **invariantes topológicos** que actúan como sellos de seguridad, imposibles de replicar sin la clave legítima.

De esta forma, la criptografía toroidal trasciende la simple aritmética modular y se asienta en **propiedades geométricas y topológicas profundas**.

Simetrías dinámicas y caos criptográfico

Aunque el toro clásico es un espacio regular, al introducir deformaciones (por ejemplo, métricas no uniformes o perturbaciones cuasiperiódicas) aparece una dinámica **casi caótica**.

Esta característica se traduce en:

- **Resistencia a ataques de predicción**, al dificultar la reconstrucción de trayectorias.
- **Entropía elevada**, útil en la generación de números aleatorios.
- **Estabilidad estructural**, ya que las pequeñas perturbaciones producen divergencias rápidas que garantizan unicidad de claves.

Invariantes geométricos como firma criptográfica

En lugar de definir la seguridad únicamente en términos de dificultad algebraica, la criptografía toroidal puede apoyarse en **invariantes geométricos**.

Ejemplos de invariantes explotables:

- **Clase de Chern del fibrado**, asociada al comportamiento global de las claves.
- **Número de rotación** en dinámicas toroidales, indicador de resonancias internas.
- **Integrales de acción** derivadas del formalismo hamiltoniano.

Cada uno de estos invariantes puede actuar como un **identificador criptográfico** que garantiza la autenticidad y unicidad de la clave.

Conexión con la teoría de campos

El formalismo geométrico en toroides enlaza naturalmente con la **teoría cuántica de campos en espacios compactos**. Allí, los modos de vibración (módulos de Fourier) se interpretan como excitaciones fundamentales.

De forma análoga, en criptografía toroidal:

- Los modos de Fourier representan **componentes elementales de la clave**.
- Las interacciones entre modos imitan **acoplamientos de partículas**.
- La seguridad se deriva de la complejidad combinatoria al intentar revertir las superposiciones de modos.

Este puente conceptual entre **geometría, dinámica y campos** sitúa al toroide como un espacio privilegiado para el diseño de criptosistemas avanzados.

Implementación computacional y complejidad de la criptografía toroidal

Representación digital del toro

Para que un toro pueda operar como base criptográfica, debe codificarse en estructuras discretas manipulables por ordenador.

- Se representa como el cociente discreto , donde son parámetros de granularidad.
- Los puntos se discretizan en enteros modulares:
- Esta discretización preserva la topología del toro y permite aplicar aritmética modular eficiente.

El resultado es un **toro digital**, que combina propiedades topológicas continuas con viabilidad computacional.

Algoritmos de exponentiación toroidal

El núcleo de los protocolos (intercambio de claves, firmas, cifrado) requiere computar productos de generadores toroidales:

El cálculo se realiza mediante algoritmos de **exponenciación rápida** (similar al square-and-multiply en ECDSA), pero extendido a dos dimensiones:

- Se descompone y en binario.
- Se computan iterativamente potencias de y .
- Se acumulan resultados en tiempo .

De este modo, el cálculo sigue siendo eficiente incluso con claves de gran tamaño.

Funciones hash toroidales digitales

Para implementar funciones hash toroidales, se usa la **transformada rápida de Fourier bidimensional (2D-FFT)**:

- Los ángulos discretos se codifican como índices de matriz.
- La FFT 2D permite evaluar rápidamente sumas del tipo:
- La clave privada genera los coeficientes .

Así se obtiene un **hash de alta dispersión**, con propiedades cuasi-aleatorias y resistencia natural a colisiones.

Protocolos de intercambio de claves

El protocolo toroidal de tipo Diffie–Hellman requiere:

1. Selección pública de .
2. Cálculo local de exponentes privados .

3. Publicación de .
4. Obtención de clave compartida tras combinación cruzada.

Complejidad computacional:

- Generación de claves: .
- Cálculo de clave compartida: .

La seguridad depende de la dificultad del **TDLP**, para el cual no se conocen algoritmos subexponenciales generales.

Firmas digitales toroidales

De forma análoga al esquema ECDSA, pueden construirse firmas basadas en toroides:

- La clave privada actúa como semilla de generación.
- La clave pública corresponde al punto en el toro digital.
- La firma se obtiene mediante funciones hash toroidales combinadas con exponentiación de generadores.

La verificación exige comprobar la **consistencia de invariantes toroidales** (p. ej., integrales discretas sobre trayectorias), lo que refuerza la robustez frente a manipulación.

Complejidad algorítmica

Comparación de complejidad:

- **RSA (factoreo):** ataques subexponenciales ().
- **ECC (ECDLP):** ataques subexponenciales ().
- **Toroidal (TDLP):** complejidad desconocida, sin reducción conocida a problemas tratados por Shor.

Se estima que el **espacio de búsqueda** para TDLP crece en función de dos parámetros, lo cual implica que los ataques de fuerza bruta tienen costo cuadrático respecto al tamaño de la clave.

Implementación en hardware y software

La criptografía toroidal es implementable en arquitecturas clásicas:

- **Hardware:** FPGAs y GPUs pueden ejecutar FFTs 2D y operaciones modulares con eficiencia.
- **Software:** librerías de álgebra computacional (SageMath, PARI/GP, NTL) permiten prototipos de funciones hash y exponentiación toroidal.

Se espera que la **paralelización natural** de las operaciones (especialmente en FFT 2D) confiera una ventaja sobre esquemas tradicionales.

Resistencia práctica frente a ataques

- **Ataques clásicos:**
    - Fuerza bruta → costo cuadrático en dos dimensiones.
    - Reducción a problemas conocidos → no existe equivalencia directa.

- **Ataques cuánticos:**
    - Algoritmos tipo Shor no aplican de forma inmediata, al no existir reducción polinómica conocida.
    - Algoritmos de Grover serían aplicables, pero solo reducen la complejidad a la raíz cuadrada, todavía elevada en claves grandes.

En consecuencia, la criptografía toroidal presenta un **perfil de resistencia robusto**, tanto clásico como cuántico.

Comparación con otros esquemas post-cuánticos (lattice, hash-based, code-based)

Resumen ejecutivo comparativo (visión de alto nivel)

- **Criptografía toroidal (TDLP)**: plantea aumentar la dimensionalidad algebraica/topológica respecto a ECDLP introduciendo estructuras toroidales, invariantes armónicos y flujos dinámicos que codifican claves en ciclos acoplados. Propone que esta mayor complejidad dificulta reducciones conocidas a problemas susceptibles de Shor y hace menos directos los ataques cuánticos estructurados.
- **Lattice-based (ej.: CRYSTALS-Kyber, Dilithium)**: basan su seguridad en problemas del reticulado (LWE, Module-LWE, NTRU) y han sido seleccionados por NIST como principales candidatas para estandarización por su eficiencia y fuerte base de análisis. Estos esquemas tienen buen rendimiento y parámetros bien estudiados; su seguridad ante algoritmos cuánticos descansa en la ausencia de algoritmos cuánticos polinómicos conocidos para LWE. (NIST, PostQuantum.com)
- **Hash-based (ej.: SPHINCS+)**: seguridad basada en funciones de dispersión (hashes) con garantías de seguridad reducidas a la resistencia de las funciones de hash frente a ataques de preimagen. Su ventaja es la simpleza conceptual y la resistencia fundamentada; su penalización típica son firmas grandes y costes computacionales más altos. (NIST)
- **Code-based (ej.: Classic McEliece)**: seguridad basada en la dificultad de decodificar códigos lineales generales. Muy resistente históricamente y con esquema candidato clásico; pero con trade-offs importantes en tamaño de claves públicas. NIST ha tratado estas familias en el proceso de estandarización. (NIST)

Seguridad teórica frente a algoritmos cuánticos

- **Shor vs Shor-like**: Shor derriba factorización y logaritmos discretos clásicos (RSA, ECC). Las familias lattice-, hash- y code-based fueron diseñadas precisamente para evitar problemas reducibles a Shor; no obstante, la evaluación de su seguridad cuántica requiere modelar tanto algoritmos cuánticos estructurales como mejoras en criptanálisis clásico. (Fortinet, PostQuantum.com)
- **Grover y efecto genérico sobre search**: Grover proporciona una mejora cuadrática sobre búsqueda no estructurada y afecta a seguridad de claves simétricas y preimagen de hashes; en consecuencia, los parámetros de hash-based y de esquemas que dependen de la fuerza bruta deben aumentarse para compensar la raíz cuadrada de la complejidad clásica. (PostQuantum.com, Cryptography Stack Exchange)

- **Toroidal (TDLP)**: la argumentación de seguridad se apoya en varias capas no triviales: (i) incorporación de múltiples exponentes acoplados (dimensionalidad), (ii) deformaciones algebraico-topológicas (simetrías de Lie, invariantes), y (iii) representación armónica multidimensional que complica la reducción a problemas lineales manipulables por transformadas cuánticas estándar. Desde la perspectiva actual, **no existe una reducción conocida** del TDLP a un problema que Shor resuelva polinomialmente; por tanto, la seguridad depende de que no emerjan algoritmos cuánticos estructurados que exploten la estructura toroidal para factorizar/linearizar la instancia. Esto es análogo a por qué LWE se considera fuerte: ausencia de reducciones a algoritmos cuánticos eficaces provistos por Shor. (Nota: esta afirmación es condicional — no hay prueba de seguridad absoluta). (PostQuantum.com)

Rendimiento, tamaños y coste práctico

- **Lattice-based**: buena relación rendimiento / tamaño de clave en comparación con RSA/ECC; parámetros y librerías optimizadas ya existen (Kyber, Dilithium). Ofrecen operaciones rápidas y adecuadas para entorno web y servidores; también poseen esquemas con claves relativamente pequeñas entre las familias PQC. (PostQuantum.com)
- **Hash-based**: firmas voluminosas (especialmente en variantes de estado-libre), pero muy simples desde el punto de vista conceptual; costo de verificación razonable, generación de firma costosa en algunos esquemas; uso recomendado en aplicaciones donde la confianza a largo plazo y la simplicidad son críticas. (NIST)
- **Code-based**: historicament e eficientes en cálculo pero con **claves públicas muy grandes** en las variantes prácticas, lo que ha limitado adopción general; robustez práctica alta. (NIST)
- **Toroidal (TDLP)**: en el diseño propuesto, la **exponenciación toroidal** y operaciones hash toroidales usan 2D-FFT y aritmética modular en . Esto implica:
    - **Ventaja**: paralelización natural (hardware y GPU/FPGA), aprovechamiento de FFT 2D para hashing, y operaciones de exponenciación con complejidad logarítmica por exponente (extendiendo técnicas square-and-multiply).
    - **Desventaja**: parámetros más complejos (dos o más dimensiones), necesidad de estandarizar discretizaciones y representaciones, y potencial sobrecoste en memoria para coeficientes armónicos .
      En síntesis: el coste puede ser competitivo con lattice-based en implementaciones paralelizadas, pero requiere ingeniería y standardización cuidadosa.

Resistencia a clases concretas de ataques

- **Reducciones algebraicas**: algunas familias (isogenies, por ejemplo) han mostrado debilidades inesperadas cuando aparecen estructuras algebraicas explotables. El TDLP debe diseñarse para evitar presentaciones que permitan linealizar la instancia (p. ej., mediante transformadas de Fourier que conviertan acoplamientos en problemas tratables). La presencia de invariantes topológicos pretende impedir esa linealización global, pero cualquier sistema concreto debe examinarse para identificar representaciones susceptibles. (ScienceDirect)
- **Ataques de canal lateral**: tanto lattice-based como hash-based y code-based han requerido importantes consideraciones de implementación para mitigar fugas físicas. La misma atención será esencial para TDLP, sobre todo en implementaciones que utilicen FFTs y operaciones paralelas (timing, consumo de energía, patrones de memoria).

- **Criptanálisis cuántico futuro**: la comunidad convencionalmente preferirá esquemas basados en problemas bien estudiados (LWE, McEliece) porque poseen décadas de análisis. TDLP, como propuesta nueva y estructurada, necesitará el mismo escrutinio: análisis de reducción, pruebas de seguridad (reducciones promedio-a-promedio si es posible), y evaluación de resistencia frente a heurísticas cuánticas emergentes.

Provable security y estado del arte

- **Familias estandarizadas (NIST)**: NIST ya seleccionó y está estandarizando un conjunto de algoritmos post-cuánticos (Kyber, Dilithium, SPHINCS+, Falcon en diversas fases), basados en análisis públicos intensivos. Esto ofrece una ventaja práctica: especificaciones, parámetros recomendados y ecosistema de librerías. (NIST, PostQuantum.com)
- **Enfoque toroidal**: hoy es teóricamente atractivo pero **no** posee el cuerpo de pruebas, testeo y comparación empírica que tienen las familias candidatas ya estudiadas. Para alcanzar madurez comparable se requieren: (i) definiciones formales del TDLP y de su parametrización; (ii) reducciones de seguridad (o límites inferiores de complejidad); (iii) análisis de ataques concretos (clásicos y cuánticos); (iv) implementaciones de referencia y auditorías de seguridad. Sin estas etapas, la adopción práctica será limitada a experimentación y casos de nicho.

Ventanas de aplicación y roles complementarios

- **Sustitución general**: lattice-based y hash-based ofrecen ya caminos viables de reemplazo en protocolos TLS, VPN, firmas y cifrado de correo. Su estandarización facilita migración. (NIST)
- **Uso especializado**: TDLP podría ofrecer ventajas en entornos donde:
    - se necesita mayor entropía estructural y redundancia topológica (ej.: entornos críticos con amenazas avanzadas),
    - la arquitectura hardware permite paralelización intensiva (GPUs/FPGA con FFTs optimizadas),
    - y cuando se desea explorar nuevos modelos de resistencia cuántica no basados en retículos ni en codificación lineal.
- **Combinación híbrida**: una estrategia prudente es adoptar esquemas híbridos (p. ej., combinar Kyber con un esquema alternativo) durante la transición. TDLP podría integrarse como capa adicional (defensa en profundidad) mientras se valida su solidez.

Riesgos, limitaciones y trabajo necesario para madurar TDLP

1. **Falta de análisis comunitario**: la seguridad de cualquier nuevo problema criptográfico mejora sustancialmente con el escrutinio público y años de análisis; TDLP requiere esto urgentemente.
2. **Parámetros y estandarización**: decidir , discretizaciones, normalizaciones y protocolos asociados exige un proceso riguroso para evitar elecciones débiles.
3. **Implementaciones seguras**: FFTs y operaciones paralelas introducen vectores de ataque de canal lateral que deben mitigarse con técnicas de constant-time y aislamiento.
4. **Reducciones a problemas bien estudiados**: sería ideal (aunque no siempre posible) encontrar reducciones que liguen TDLP a problemas de complejidad ampliamente aceptados o, al menos, demostrar resistencia a clases de algoritmos cuánticos emergentes.

Conclusión comparativa (técnica)

- **Madurez**: lattice-based / hash-based / code-based ⩾ TDLP. Las primeras tienen especificaciones, análisis y estandarización; TDLP está en fase conceptual/prototipo. (NIST, PostQuantum.com)
- **Potencial de resistencia cuántica**: TDLP ofrece un marco prometedor (dimensionalidad + topología + armónica) que **no** es trivialmente vulnerable a Shor; sin embargo, la ausencia de pruebas y del escrutinio requerido lo sitúa como hipótesis de trabajo más que como alternativa lista para producción.
- **Rendimiento práctico**: TDLP puede ser competitivo si se optimiza en hardware por su naturaleza paralela (FFT 2D), pero la complejidad de parámetros y tamaño de estructuras armónicas añade costes que deben medirse frente a esquemas lattice-based maduros.

Citas clave utilizadas en esta sección

- NIST — anuncio y proceso de selección de algoritmos post-cuánticos (Kyber, Dilithium, SPHINCS+, Falcon). (NIST)
- Revisiones y resúmenes técnicos sobre familias PQC (postquantum.com, PQShield). (PostQuantum.com, PQShield)
- Notas sobre Grover y su efecto de mejora cuadrática en búsquedas/preimágenes y sobre Shor como amenaza a RSA/ECC. (PostQuantum.com, Fortinet)

Análisis de seguridad formal y protocolo experimental para TDLP

Objetivo y alcance del análisis formal

Objetivo: definir criterios formales, métricas y experimentos necesarios para evaluar la seguridad del **Toroidal Discrete Logarithm Problem (TDLP)** frente a adversarios clásicos y cuánticos, y comparar su resistencia práctica con familias post-cuánticas consolidadas.

Alcance:

- Definición precisa del problema TDLP y del espacio de parámetros.
- Modelado de adversario (clásico y cuántico) con recursos límite.
- Conjunto de experimentos para evaluar: complejidad práctica, posibles reducciones algebraicas, vectores de ataque (analíticos y de implementación) y robustez frente a heurísticas.
- Recomendación de parámetros (familias) para distintos niveles de seguridad.

Definición formal del TDLP (instancia y objetivos)

Definimos una instancia discreta del toro digital mediante parámetros enteros positivos (idealmente primos o coprimos, según la construcción) y generadores discretos en el grupo abeliano toroidal discreto o su representación multiplicativa equivalente.

Instancia TDLP: dado (o suma en la notación aditiva del grupo), recuperar el par con , .

Problemas derivados a definir formalmente:

- TDLP-Linear: caso en que combinación es lineal en los exponentes.
- TDLP-Deformed: casos en que la "exponenciación" incluye un acoplamiento no lineal (por ejemplo, con constantes modularmente definidas).

Un esquema criptográfico debe especificar cuál de las variantes (lineal o deformada) emplea de forma exclusiva; la deformada ofrece mayor seguridad potencial pero complica la estandarización.

Modelo de amenaza y métricas de seguridad

Modelo de amenaza: adversario pasivo y activo, clásico o cuántico, con acceso a: claves públicas, oráculos de firma (para esquemas de firma), y tiempos/consumo (para ataques de canal lateral). Adversario cuántico puede ejecutar oráculos cuánticos y algoritmos como Grover o subrutinas para HSP (Hidden Subgroup Problem) si aplica.

Métricas:

- : complejidad (tiempo y memoria) de la mejor estrategia clásica conocida para resolver TDLP con parámetros dados.
- : complejidad cuántica (número de qubits lógico, profundidad cúbica/política, número de llamadas a oráculos), estimada para algoritmos generales y heurísticos (incluye Grover).
- , : niveles de seguridad expresados en bits ($\log_2$ de la complejidad esperada).
- Resiliencia estructura $\rightarrow$ mide propensión a que instancias concretas sean reducidas a problemas tratados por Shor u otros algoritmos polinómicos cuánticos.

Reducciones y análisis teórico a estudiar

Prioridad de pruebas formales (objetivos de reducción):

1. **Reducción a HSP / Shor**: investigar si TDLP (lineal o deformado) puede mapearse a un caso del Hidden Subgroup Problem en algún grupo abeliano o no abeliano susceptible a solución cuántica eficiente. Si existe una homomorfía con apropiado, se debe demostrar imposibilidad o condición de no existencia de tal mapeo eficiente.
2. **Posible reducción a problemas de retículo (LWE/NTRU)**: estudiar si la representación vectorial del TDLP (por ejemplo, mediante transformada de Fourier discreta) genera un sistema susceptible de reducción a LWE o a problemas de la geometría de retículos. En caso positivo, derivar parámetros que resistan algoritmos de reducción del retículo (BKZ, sieving).
3. **Complejidad promedio-a-peor**: probar si existen reducciones promedio$\rightarrow$peor o peores$\rightarrow$promedio que permitan acotar la seguridad de instancias aleatorias frente a instancias patológicas.
4. **Vulnerabilidades isogenia-like**: analizar si existen transformaciones algebraicas en el espacio de redes que permitan construir “isogenias” toroidales que reduzcan la instancia. Desarrollar criterios para detectar y evitar parámetros débiles.

Plan experimental reproducible (benchmarks)

Propongo una batería de experimentos estandarizados, reproducible, con scripts y métricas registradas:

A. Implementación de referencia

- Entorno: implementación en C/C++ y Python (prototipo), y versiones aceleradas en GPU (CUDA) y FPGA.

- Estructuras: representación de , generación de , exponentiación, hashing 2D-FFT.
- Medidas: tiempo de generación de clave, tiempo de exponentiación, uso de memoria, throughput (ops/s) en CPU/GPU/FPGA.

B. Instancias y parámetros de prueba

- Familias de parámetros por objetivo de seguridad cuántica : 64, 96, 128, 192. (ver mapeo y recomendaciones más abajo).
- Generar 1000 instancias aleatorias por familia para análisis estadístico de tiempos y colisiones.

C. Análisis cuantitativo de búsqueda

- Medir tiempos prácticos para ataques de fuerza bruta distribuidos (simulados) y ataques con algoritmos heurísticos de reducción.
- Implementar ataque de baby-step giant-step adaptado a la estructura bidimensional y medir coste/almacenamiento. Evaluar versiones adaptadas a espacio dividido (meet-in-the-middle) que exploten separación de exponentes.

D. Evaluación de reducción por Fourier

- Construir transformadas de Fourier discretas de las instancias y buscar diagonalizaciones que conviertan la instancia en problemas unidimensionales. Evaluar heurística para detectar representaciones vulnerables.

E. Vectores de ataque cuántico (simulados)

- Estimar coste cuántico teórico: profundidad de circuito y número de qubits lógicos si se intentase aplicar Grover sobre espacio combinado.
- Estimar si existe mapeo a HSP y simular (teóricamente) ganancias si se produjese.

F. Criptanálisis práctico

- Tests de reducción a retículos: formular el sistema lineal/experimental derivado y aplicar BKZ/sieving para medir si se encuentra vector solución con recursos prácticos.
- Tests de colisiones: buscar colisiones en funciones hash toroidales con técnicas avanzadas (distinción estadística, ataques de preimagen).

G. Seguridad de implementación

- Simulaciones de ataques de canal lateral: timing, consumo energético y patrones de memoria (particularmente para FFT 2D y operaciones paralelas).
- Aplicar técnicas de mitigación: constant-time, blinding, precomputation protegido.

H. Repetición y reporte

- Publicación de scripts, datos y métricas en repositorio público para revisión independiente.
- Reporte estructurado: curvas tiempo vs. tamaño, tablas de parámetros, errores tipo I/II en heurísticas de reducción.

Recomendación preliminar de parámetros (familias) — justificación y mapeo a seguridad

Razonamiento: la complejidad de un ataque por fuerza bruta sobre TDLP lineal es proporcional al producto . Grover aplicado a búsqueda no estructurada reduce complejidad a . Para proteger contra adversarios cuánticos que usen Grover, elegir para un objetivo de seguridad cuántica , ya que .

Propuestas (valores expresados en bits por cada módulo; es una elección inicial y debe validarse empíricamente):

- **Seguridad cuántica 128 bits** (): elegir (por ejemplo, primos con ≈ 128 bits cada uno; producto ≈ 2^{256}).
    - Comentario: proporciona resistencia contra Grover a 128 bits; sin embargo, si aparecen reducciones estructuradas, deben aumentarse tamaños.
- **Seguridad cuántica 96 bits**: (producto ≈ 2^{192}).
- **Seguridad cuántica 64 bits (uso experimental)**: .

Si se usa esquema deformado con acoplamientos no lineales, es razonable balancear y de forma asimétrica (por ejemplo) cuando quiera forzar que el esfuerzo de búsqueda simultánea sea dominado por el mayor de los parámetros.

Nota de prudencia: estas recomendaciones presuponen ausencia de reducciones algebraicas que exploten independencia entre componentes. Si emergen ataques que exploten estructura (p. ej., transformaciones que diagonalizan la interacción), será necesario incrementar parámetros o rehacer la representación.

Experimentos de validación crítica (prioritarios)

1. **Baby-step giant-step 2D**: adaptar y medir factibilidad. Determinar si almacenamiento/tiempo prácticos permiten romper instancias pequeñas (para calibrar escalado).
2. **Análisis de Fourier y diagonalización**: buscar transformaciones que conviertan la instancia en problemas unidimensionales.
3. **Reducción a retículos**: construir matrices asociadas a instancias TDLP y ejecutar BKZ-cores con distintos bloques para ver si soluciones se obtienen en recursos prácticos.
4. **Simulaciones Grover**: aunque costosas, estimar coste y número de oráculos necesarios; contrastar con parámetros propuestos.
5. **Side-channel**: medir fugas en implementaciones GPU/FPGA y aplicar mitigaciones.

Criterios de aceptación y umbrales de inseguridad

- **Criterio de aceptación**: para una familia de parámetros, si ningún ataque (clásico o cuántico estimado) puede resolver instancias en coste inferior al objetivo de seguridad (/) en pruebas reproducibles, la familia se considera preliminarmente segura.
- **Umbral de inseguridad**: aparición de una reducción estructural a problema resoluble en coste subexponencial o polinomial respecto al tamaño de parámetros (p. ej., mapeo efectivo a HSP o a LWE con parámetros débiles) implica rechazo y replanteamiento de la representación.

Bibliografía

Nota: he incluido obras canónicas y artículos fundamentales que proveen el marco matemático y criptográfico. Estas referencias son estables y de renombre académico; sirven para fundamentar definiciones y técnicas usadas en el texto.

1. **Silverman, J. H. — *The Arithmetic of Elliptic Curves* (Graduate Texts in Mathematics).**
   Resumen: Texto de referencia en teoría de curvas elípticas: definición, estructura de grupos, reticulados complejos y aplicaciones aritméticas. Fundamento formal para comprender la relación entre curvas elípticas y toroides (curvas de género uno).
   Uso en este artículo: base teórica para la equivalencia topológica entre curvas elípticas complejas y toroides, y para las operaciones de grupo usadas como analogía.

2. **Koblitz, N. — *A Course in Number Theory and Cryptography*.**
   Resumen: Introducción a problemas criptográficos clásicos (ECDLP, RSA), además de presentar curvas elípticas desde la perspectiva criptográfica.
   Uso: para contrastar ECDLP con la propuesta TDLP y entender prácticas de exponentiación y protocolos basados en curvas.

3. **Apostol, T. M. — *Modular Functions and Dirichlet Series in Number Theory*.**
   Resumen: Tratamiento riguroso de formas modulares y teoría de reticulados.
   Uso: soporte teórico para la sección sobre formas modulares e invariantes modulares que intervienen en la clasificación de toroides.

4. **Kac, V. — *Infinite-dimensional Lie Algebras*.**
   Resumen: Introducción al formalismo de álgebras de Kac–Moody y simetrías infinitodimensionales.
   Uso: fundamento para la discusión sobre simetrías de Lie y posibles operadores infinitodimensionales que actúen sobre toroides.

5. **Arnold, V. I. — *Mathematical Methods of Classical Mechanics*.**
   Resumen: teoría de sistemas hamiltonianos, flujos en variedades y conceptos de espacio de fases.
   Uso: justificación formal de la interpretación del toro como espacio de fases y la explotación de flujos hamiltonianos en diseño criptográfico.

6. **Cooley, J. W., Tukey, J. W. — "An algorithm for the machine calculation of complex Fourier series" (1965).**
   Resumen: artículo seminal que presenta el algoritmo FFT.
   Uso: base técnica para implementar hashes toroidales mediante FFT 2D y justificar la eficiencia computacional.

7. **Regev, O. — "On lattices, learning with errors, random linear codes, and cryptography" (2005).**
   Resumen: artículo clave que formaliza LWE, piedra angular de la criptografía lattice-based moderna.
   Uso: marco comparativo para evaluar el estado del arte en post-cuántico y en intención de redicciones (LWE como paradigma de problemas no resueltos por Shor).

8. **McEliece, R. J. — "A public-key cryptosystem based on algebraic coding theory" (1978).**
   Resumen: proposición original del esquema McEliece (code-based).
   Uso: referencia histórica y técnica para comparar trade-offs de clave y rendimiento frente a TDLP.

9. **Bernstein, D. J., Böhl, C., et al. — trabajos sobre hash-based signatures (p. ej., SPHINCS+ papers / documentación pública).**
   Resumen: descripciones y análisis de esquemas basados en hash con seguridad postulada basada en resistencia a preimagen.
   Uso: comparar filosofía de seguridad y costos con la familia toroidal.

10. **Conway, J. H., Sloane, N. J. A. — *Sphere Packings, Lattices and Groups*.**
    Resumen: exhaustivo texto sobre retículos y geometría discreta.
    Uso: soporte para debates sobre posibles reducciones de TDLP a problemas de retículos y para análisis geométrico de representaciones.

Resumen final

- El **TDLP** generaliza ECDLP a un dominio toroidal discretizado; su dificultad proviene de la **dimensionalidad** (dos o más exponentes), **acoplamientos algebraicos** y **complejidad armónica/topológica**.
- La **estructura toroidal** permite construir: funciones hash 2D (FFT), protocolos Diffie–Hellman bidimensionales, firmas y esquemas con invariantes topológicos como sello criptográfico.
- **Reducciones a algoritmos cuánticos conocidos (Shor/HSP)** no son evidentes de forma inmediata; no obstante, la ausencia de una reducción no es prueba de seguridad — TDLP requiere escrutinio profundo.
- Plan experimental propuesto incluye: implementaciones referencia (CPU/GPU/FPGA), batería de ataques (baby-step giant-step 2D, reducción Fourier, reducción a retículos BKZ), simulaciones de Grover y pruebas de canal lateral. Publicar scripts y datos reproducibles es esencial.
- Recomendación preliminar de parámetros: para seguridad cuántica , seleccionar (ej., para tomar). Estas cifras requieren validación y son conservadoras frente a Grover.
- Vectores de trabajo prioritarios: (i) demostrar ausencia (o condiciones de existencia) de reducciones HSP; (ii) evaluar mapas potenciales a LWE/retículos; (iii) auditar instancias para detectar representaciones vulnerables; (iv) mitigar ataques de canal lateral en FFT y exponentiación paralela.
- Comparación con familias PQC: TDLP es prometedor por su novedad estructural y paralelización natural, pero **no** tiene aún madurez, pruebas ni estandarización comparables a lattice-, hash- o code-based schemes. Su adopción práctica solo procede tras años de análisis público y estandarización.
- Operacionalmente, la criptografía toroidal puede desempeñar un papel complementario (defensa en profundidad) y experimental en entornos donde la paralelización hardware y la entropía topológica sean aprovechables.

Justificación metodológica

- Regla de diseño usada: para lograr seguridad cuántica (bits), tomar . Con Grover afectando a una búsqueda no estructurada, la raíz cuadrada del espacio de búsqueda queda en .

- Representación práctica: discretizamos como . Un elemento público se codifica mediante una pareja de residuos . Guardar exige almacenar bits.
- Seguridad adicional: usar variantes *deformadas* (acoplamientos no lineales entre exponente m,n) y/o introducir invariantes topológicos reduce riesgo de reducciones peligrosas pero complica análisis y estandarización.
- Eficiencia/Fourier: usar FFT 2D requiere que las discretizaciones se mapeen a tamaños de transformada (p. ej. potencias de 2) o factorizables en pequeños factores primos para FFT rápida; esto condiciona selección de prácticos en implementación.

Tabla de parámetros recomendados (ejemplos concretos)

Notas:

- “Public key” = almacenamiento de (dos residuos).
- “Private key” = almacenamiento de (dos residuos).
- Para cada fila indico dos primos de ejemplo generados aleatoriamente con la longitud indicada; en implementación real se eligen primos/elementos según criterios de seguridad y compatibilidad FFT.
- Las cifras son orientativas y deben validarse con los experimentos propuestos en el artículo.

Objetivo seguridad ()	**Bits**	**Bits**	**Producto ≈**	**Ejemplo (decimal)**
Experimental — 64 bits	64	64		17207164439507402761
Intermedio — 96 bits	96	96		60136326758805842285492941433
Estándar seguro — 128 bits	128	128		309617653643349137117712248721405999881
Alto — 192 bits	192	192		319685788048070786697772668676192910802839639003257852807
Muy alto — 256 bits	256	256		8146675446181338890728325337726248126411690898263188137098686089444

Objetivo seguridad (λ_q)	Bits N	Bits M	Producto ≈	Ejemplo N (decimal)	Ejemplo M (decimal)
Experimental — 64 bits	64	64	2^{128}	17207164439507402761	17207164439507402761 (ej.)
Intermedio — 96 bits	96	96	2^{192}	6013632675880584228549294143 3	60136326758805842285492941433 (ej.)
Estándar seguro — 128 bits	128	128	2^{256}	30961765364334913711771224872140599881	18054215543058285314341873600240843510 9
Alto — 192 bits	192	192	2^{384}	31968578804807078669777266867619291080283963900325785280 77	(otro primo 192 bits)
Muy alto — 256 bits	256	256	2^{512}	81466754461813388907283253377262481264116908982631881370986860894441150728029	(otro primo 256 bits)

Cómo leer la tabla

- Si por ejemplo eliges y de 128 bits, la clave pública (un par de residuos) requiere ≈ 256 bits = 32 bytes. La seguridad contra Grover queda en ≈ 128 bits si no existen reducciones estructurales.
- En la práctica conviene definir representación canónica de residuos (por ejemplo, valor entero reducido en intervalo) y comprimir si hay espacio para transformaciones isomórficas seguras.

Estimaciones de almacenamiento y coeficientes armónicos

Tamaño de coeficientes (hash toroidal)

- Si se usara una **matriz completa** de coeficientes con entradas y cada coeficiente con precisión bits (por ejemplo 32 o 64 bits para representación fija): almacenamiento bruto = bits.
    - Ej.: para y : bits ≈ 64 KiB.
- **Recomendación práctica:** usar **representación dispersa (sparse)**: almacenar únicamente coeficientes no nulos (por ejemplo o menos), o usar función de expansión pseudoaleatoria (PRF) que derive coeficientes a partir de la clave privada en vez de guardarlos todos.

Tamaño de estado temporal en FFT 2D

- Para implementar FFT 2D se mapea la cuadrícula a un arreglo de tamaño (donde , preferiblemente potencias de 2). Memoria temporal ≈ (b=precisión). Planificar memoria y evitar fugas.

Notas de implementación y optimización

Representación y serialización

- **Clave pública** : serializar como concatenación big-endian de residuos (q1 ‖ q2). Añadir versioning y parámetros públicos (N-bit length, M-bit length, hash ID).

- **Clave privada (m,n)**: almacenar de forma segura, con protección contra fugas. Soporta blinding (suma de valor aleatorio mod N/M) en operaciones para mitigar side-channels.

FFT 2D y elección de

- Para eficiencia FFT elegir potencias de 2 (p.ej. 128, 256, 512) o productos de factores pequeños. Si se eligen como primos grandes (seguridad), mapear índices a rejilla FFT mediante re-muestreo o usar FFT sobre campo finito (NTT) si se desea aritmética modular exacta.
- Si usas NTT (Number Theoretic Transform), necesitas que el módulo soporte raíces primitivas de orden y ; esto condiciona elección de módulos auxiliares.

Paralelización y hardware

- **GPU / CUDA**: excelente para FFT 2D y cálculo paralelo de potencias; atención a fugas por timing y patterns de memoria.
- **FPGA**: permite implementación en streaming, optimizando latencia y mitigando algunos side-channels; sin embargo, mayor coste de diseño.
- **CPU**: viable para prototipos y pruebas; usar librerías FFT y optimización en contadores de bits.

Representación deformada (asymetric / coupled)

- Variante útil: escoger grande y más pequeño (ej. 192 bits, 64 bits) y forzar acople no lineal para que la búsqueda simultánea sea dominada por . Esto puede reducir almacenamiento y mantener dificultad combinada.
- Riesgo: asimetría facilita técnicas de meet-in-the-middle si no se deforman correctamente; controlar mediante términos modulares irracionales o parámetros secretos en la deformación.

Ejemplos concretos de parámetros sugeridos por caso de uso

1. **Prototipo de laboratorio (investigación):**
    - : 64 bits, : 64 bits (rápido, pequeño).
    - Objetivo: evaluar algoritmos, medir baby-step giant-step 2D y coste prácticas.
2. **Despliegue experimental seguro (servidores/testbeds):**
    - : 128 bits, : 128 bits.
    - Guardar coeficientes como PRF derivado a partir de — evita almacenar toda la matriz.
    - Implementar mitigaciones contra side-channels y pruebas de reducción a retículos.
3. **Entornos de alta seguridad (archivo secreto):**
    - : 192 bits, : 192 bits o asimétrico 256/128 con deformación fuerte.
    - Preparar implementación en FPGA y estudio de consumos/latencias.

Buenas prácticas de estandarización propuestas

- Definir **familias canónicas** de parámetros (p.ej. TOR-128x128, TOR-192x192, TOR-256x256) con descripción completa: primos/representación, método de generación de , formato de serialización, y tamaños de buffer para FFT.
- Especificar **métodos PRF** para generación on-the-fly de coeficientes armónicos desde la clave privada, evitando almacenar tablas grandes.
- Establecer **procedimientos de test**: baby-step giant-step 2D adaptado, matriz de pruebas para reducción a retículos, benchmark en CPU/GPU/FPGA y tests de side-channel.
- Incluir **versioning y parámetros** en cada clave pública (identificador de familia, tamaño N/M, hash ID, flags de deformación).

Conclusión

- La tabla y las recomendaciones anteriores son **operacionales**: permiten a un equipo reproducir instancias TDLP concretas, implementar prototipos y ejecutar la batería experimental que proponemos en la Sección 7 del artículo.