De Euclides al Qubit

La singularidad de la aritmética modular

Por Aingeru García Blas

Índice

1. Presentación

1.1. ¿Por qué he elegido esta temática?

Cuando se nos presentó a los alumnos la posibilidad de hacer un trabajo de Criptografía para la asignatura de Matemática Discreta, no dudé ni un instante en mi participación.

Como contextualización, podríamos decir que soy un apasionado de las ciencias, particularmente de las matemáticas, la física y la informática. Pese a no ser un experto, mi intención durante estos años académicos es la de especializarme lo máximo posible, puesto que estas ciencias que nos ayudan a comprender los entresijos del universo es algo que me despierta una sed de conocimiento que continúa expandiéndose.

Mi curiosidad hacia la criptografía lleva muchos años presente, siempre ha sido una gran afición y me he interesado por la criptografía clásica. Concretamente entorno al contexto de la máquina Enigma y la Segunda Guerra Mundial. Comencé sintiendo mucha atracción por el tema con la película Enigma (2001) y literatura relacionada (especial mención a el Criptonomicón), para continuar luego investigando y estudiando diferentes temas.

Con este trabajo me gustaría transmitir la pasión que me despierta esta temática, aunando los conocimientos adquiridos en clase, mis intereses y pasiones al respecto. Durante todo el trabajo tendremos un factor común o, también podríamos llamarlo path. Éste path es fruto de la relación que, tras un ejercicio de abstracción sobre toda la materia investigada, he decidido sea el hilo conductor a través del viaje espacio temporal que he preparado.

Espero que pueda apreciarse la inspiración que todo esto me produce, lo que he disfrutado realizando el trabajo y por supuesto que el recorrido sea agradable y satisfactorio.

1.2. ¿Qué es la criptografia?

Entendemos por Criptografía (del griego kripto (secreto) y grafia (escritura)) a la ciencia o arte que cifra y o codifica con diferentes técnicas mensajes para hacerlos incomprensibles para aquellos que no son los destinatarios. Nótese que las técnicas de cifrado tienen como objectivo principal ocultar el significado del mensaje y no el mensaje en sí.

Históricamente ha habido una necesidad de ocultar información en toda clase de contextos. Previo al auge de la culturización de la población y debido a la escasa cantidad de gente letrada, los sistemas de cifrado de la antigüedad no necesitaban mucha complejidad para ocultar los mensajes. A medida que las sociedades y culturas han ido avanzando, los sistemas de cifrado han ido adaptándose a esas circunstancias volviéndose más complejos y sofisticados para poder seguir manteniendo el *contenido* de los mensajes en secreto.

Se puede observar que durante toda la historia se han utilizado toda clase de técnicas para el cálculo de los cifrados, desde el *método Polibio*, al *análisis de frecuencias* medieval para romper códigos, pasando por *jeroglíficos, transposición, sustitución a sistemas de claves*, sin obviar la maquina **Enigma**, o incluso idiomas sin ningún tipo de conexión con otros, utilizados como sistema de codificación natural para desconocedores de ese lenguaje como el caso de al que se llegó a denominar cómo: 'Código Euskera'.

Guerras, economías de los diferentes países, política, compra-venta de información; todas estas variables han ayudado a la proliferación de sistemas de codificación y ámbitos de ocultación en general, generando una ciencia muy llamativa para las mentes curiosas, sobre todo dentro del mundo de la **matemática e ingeniería**.

1.3. Bases Matemáticas

Para poder generar esos sistemas de cifrado y codificación, la Criptografía necesita de herramientas que provienen de diferentes áreas de las matemáticas. Se pueden llegar a utilizar una gran variedad de ramas de esta ciencia como los *logaritmos discretos, curvas elípticas, ecuaciones multivariables* o la interesante y compleja criptografía basada en *retículos o lattices*, etcétera...

De todas las posibles ramas, vamos a centrarnos en la **Teoría de Números**, concretamente en la *Aritmética modular*.

2. La Criptografía en el pasado

Como ya hemos visto, el interés por codificar información puede remontarse incluso milenios atrás, de todos los ejemplos, culturas del pasado y toda la historia por contar en relación a este tema, nos vamos a centrar en una persona que considero un punto de inflexión en la matemática.

2.1. Euclides

Euclides (325 ac - 265 ac) fue un matemático griego, al que gracias a sus obras puede ser considerado uno de los pensadores más influyentes de la historia. Es conocido como el padre de la geometría y su obra *Elementos*, es uno de los pilares básicos de las matemáticas que se estudian hoy en día.

De entre sus abundantes contribuciones, vamos a centrarnos en una de las que podría denominarse como uno de los ejes de la criptografía; estas bases se han mostrado inmutables a el paso del tiempo, validando y asombrándonos a partes iguales con su relevancia, complejidad y visión por parte de Euclides; que se siguen aplicando más de 2.000 años después y no hay indicio de que vayan a dejar de utilizarse en un futuro cercano.

A continuación se asentarán las bases para comprender el resto de contextos, y poder disfrutar de un viaje matemático desde Euclides hasta el Qubit.

2.2. Algoritmo de Euclides

En su obra *Elementos*, Euclides explica cómo poder obtener el máximo común divisor de entre dos números cualesquiera simplemente dividiendo el número mayor por el número menor. Sí la división es exacta, se toma el número menor como *mcd* (máximo común divisor). Si no es exacta, el divisor pasa a ser el dividendo y el resto pasa a ser el divisor, el *mcd* será el último número por el cual podamos dividir.

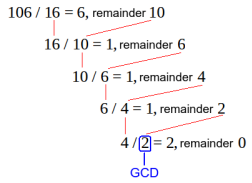


Figura 1: M.C.D

Un ejemplo simple de una función recursiva que programé en Python:

```python
# Algoritmo de Euclides recursivo.
# Implementación en Python.
def mcd(a, b):
    while(b != 0):
        mod = a % b
        a = b
        b = mod
    return a
```

2.3. Aritmética Modular

La aritmética modular es un área de la matemática discreta introducido por Euclides, el cual nos permite trabajar con clases de equivalencia entre números. Estas clases de equivalencia las llamamos clases de congruencia. La percepción más moderna de éste sistema aritmético es el presentado por Carl Friedrich Gauss en su obra *Disquisitiones Arithmeticae*, en 1801.

2.3.1. Concepto de módulo

Una idea que sirve para comprender el concepto de módulo, es la idea de *Aritmética de reloj*, la cual acota el universo en el cual los números $\mathbb{Z}$ pueden existir en base al módulo n en el que estemos trabajando. En el caso del reloj, $n = 60$, con lo que, por ejemplo, 62 no tiene sentido en éste $\mathbb{Z}_n$, y será:

$$62 \pmod{60} = 2$$

Dos números cualesquiera que tengan el mismo módulo serán congruentes, es decir, dos numero cualesquiera que sean divididos por un número o módulo y tenga el mismo resto serán equivalentes o congruentes,

y se representa:

$$a \equiv b \pmod{n}$$

Nótese que el $\mathbb{Z}_n$ estará constituido por $\mathbb{Z}_n - 1$ números y será $\mathbb{Z}_n = \{0...\mathbb{Z}_n - 1\}$

2.3.2. Inverso modular

El inverso modular será una herramienta extremadamente útil que nos ayudará a deshacer operaciones más adelante. Sabemos que por definición:

$$a \cdot a^{-1} \pmod{n} = 1$$

Esto implica que, dentro de $\mathbb{Z}_n$ habrá un número que multiplicado por su inverso, será igual a 1. Para saber si un número tiene inverso, el *mcd* entre dicho número y el módulo tiene que ser la unidad, es decir, el número a y el *módulo n*, deben de ser ***coprimos*** o primos relativos.

Éste concepto es muy similar en la adición, pero a diferencia del multiplicativo, el inverso será el complemento. Con lo que:

$$a + a^{-1} \pmod{n} = 0$$

Por ejemplo, si tomamos $n = 27$ (número de letras del alfabeto, incluyendo la ñ) y queremos cifrar la letra F, a la cual le corresponde la posición número 5 del alfabeto (empezando por 0) y para ello rotamos 12 caracteres a la derecha: $5 + 12 = 17 \Rightarrow Q$. Su inverso será el número que sumado a 12 y pasado a módulo n sea 0. Es decir:

$$(15 + 12) \pmod{27} = 0$$

Para descifrar Q, no tenemos más que hacer $(17 + 15) \pmod{27} = 5 \Rightarrow F$.

2.4. Algoritmo de Euclides extendido

Llegados a este punto, veamos el método de cálculo para poder obtener el inverso modular de un número. Una vez más recurriremos al algoritmo de Euclides, pero ésta vez en su versión extendida, la cual trata

Sabiendo que un $ax + bn = mcd(a, n)$, siempre que a y n sean coprimos (es decir, $mcd(a, n) = 1$). El algoritmo de Euclides extendido, nos ayudará a expresar ésta combinación lineal de tal manera, que podremos calcular el valor de x. Ya que:

$$a \cdot x \equiv 1 \pmod{n}$$

Veamos un ejemplo, pese a que no sean primos relativos.

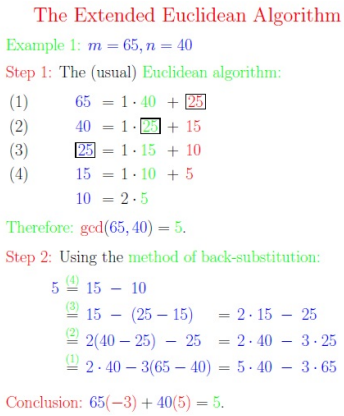


Figura 2: Ejemplo de Algoritmo de Euclides extendido, a y n no son coprimos

2.5. Algoritmos de cifrado mediante Aritmética Modular: Vigènere.

El cifrado *Vigenère* (incorrectamente atribuido a Blaise de Vigenère en el siglo XIX, ya que Giovan Battista Belass dejó constancia del mismo ya en 1533). Es un sistema basado en el cifrado Cesar, pero a diferencia de éste dispone de una tabla la cual, teniendo la letra que queremos cifrar y una clave arbitraria, obtendremos el valor cifrado.

Vemos que $\mathbb{Z}_n$ en este caso es 27, el numero de letras del alfabeto. Con lo que será nuestro *universo*.

Veamos un ejemplo:

Si deseamos cifrar la letra P, y tenemos como clave L, obtendremos la A en la tabla *Vigenère*.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Figura 3: Tabla de claves Vigenere

Si X es la letra a cifrar, K la clave y L es la cantidad de letras que contiene el alfabeto, el cifrado sería:

$$E(X_i) = (X_i + K_i) \pmod{L}$$

3. La Criptografía en el presente

3.1. Breve explicación e influencia de Euclides

El enfrentamiento intelectual entre las personas que quieren ocultar información y los que por el contrario, deseaban conocer su contenido, ha generado que con el paso del tiempo y el acceso a la educación, haya habido un incremento progresivo en la complejidad de los criptosistemas. Siempre basándose en los fundamentos anteriormente expuestos de Euclides.

La llegada de la informática y la revolución electrónica generó un cambio en las comunicaciones y con ello la manera en que se debía ocultar la información, los diferentes tipos de comunicaciones han evolucionado de la misma manera que la necesidad de encontrar más y mejores sistemas de criptografía.

En la actualidad existen gran variedad de sistemas, se podrían englobar en criptosistemas simétricos, asimétricos, híbridos y por funciones de *hash* (aunque estas últimas sean irreversibles).

3.2. Algoritmo RSA

Su origen data del 1977, en el Massachusetts Institute of Technology, cuando sus creadores **R**ivest, **S**hamir y **A**dleman lo hicieron público. Un dato curioso es que Clifford Rocks, un británico, lo hubiese diseñado en 1973 pero no disponer de los recursos computacionales adecuados pasó inadvertido - y esto, no se conoció hasta 1997 ya que era información confidencial.

Éste sistema es un criptosistema asimétrico de claves públicas, lo cual implica que tanto el emisor como el receptor deberán de disponer de dos claves: una pública y una privada. El emisor cifrará su mensaje con la clave pública del receptor y el receptor descifrará el mensaje con su propia clave privada.

La clave de éste algoritmo radica en utilizar, como veremos a continuación, números muy grandes de tal manera que la factorización de los mismos requiriese de una cantidad de cómputos inconmensurables, algo que en términos prácticos, no pueda ser resuelto en un tiempo apropiado.

Éste criptosistema está constituido por 3 fases: generación de claves (ambas partes), cifrado (emisor) y descifrado (receptor).

Veamos cómo funcionan cada una de ellas.

3.2.1. Generación de claves

- Se eligen arbitrariamente dos números **primos**, $\boldsymbol{p}$ y $\boldsymbol{q}$. Cuanto más grandes, más difícil será descomponer el producto en los mismos.

$$n = p \cdot q$$

- Hecho ésto, sabemos que $\boldsymbol{p}$ y $\boldsymbol{q}$ serán los únicos divisores primos de $\boldsymbol{n}$ (a parte del 1).

- Ahora se calcula la función de Euler; $\boldsymbol{\phi(n)}$, que indicará la cantidad de *números primos relativos con* $\boldsymbol{n}$, es decir, números que en módulo n serán invertibles.

$$\boldsymbol{\phi(n)} = (p-1)(q-1)$$

- Elegimos un número $\boldsymbol{r}$ que sea *primo relativo* de ϕ(n) y ya tendremos nuestra **clave pública** compuesta por **(n, r)**. Siendo $\boldsymbol{r}$ el exponente de cifrado.

- Para la clave privada a la cual llamaremos $\boldsymbol{s}$, calculamos el inverso modular de $\boldsymbol{r}$ en módulo ϕ(n):

$$s = r^{-1} \pmod{\phi(n)}$$

  O lo que es lo mismo, s y r serán congruentes en módulo ϕ(n):

$$r \cdot s \equiv 1 \pmod{\phi(n)}$$

  Nótese que para éste cálculo, utilizaremos el algoritmo de Euclides extendido.

Ahora se tiene que realizar un intercambio de claves públicas entre el emisor - receptor. De ésta manera, cifraremos nuestro mensaje con la clave pública de éste último. En muchos casos, éste intercambio se realiza mediante el protocolo Diffie y Helman.

3.2.2. Cifrado

Ésta operación deberá de aplicarse para cada uno de los códigos que deseemos cifrar. Obviamente, es interesante que primero pasemos el texto plano a otra codificación, para generar una capa extra de ofuscación.
Una vez se tienen $\boldsymbol{n}$, $\boldsymbol{r}$ y $\boldsymbol{s}$, podemos proceder al cifrado. Siendo $\boldsymbol{mc}$ el mensaje cifrado y $\boldsymbol{t}$ el mensaje en texto plano, procederemos a su cifrado mediante la exponenciación binaria en módulo $\boldsymbol{n}$. Nótese que en éste caso, $\boldsymbol{r}$ será la clave pública del receptor, una vez realizado el intercambio de claves públicas.

$$mc = t^r \pmod{n}$$

3.2.3. Descifrado

A su vez, el receptor procederá a descifrar el mensaje gracias a su **clave privada**. Para ello, lo que, a mi juicio es algo mágico y bonito, realizaremos la misma operación de cifrado. Pero, en lugar de utilizar $\boldsymbol{r}$ como exponente, utilizaremos la propia clave privada del receptor, de la siguiente manera.

$$t = mc^s \pmod{n}$$

Llegamos a éste método ya que se cumple el **Teorema de Euler**:

$$mcd(a,\ n) = 1$$
$$\wedge$$
$$\mathrm{a}^{\phi(n)} \equiv 1 \pmod{n}$$

Como anécdota, me gustaría contar brevemente la siguiente. Recuerdo que, esto mismo entró en el último ejercicio del examen del Bloque 1 de la asignatura de Matemática Discreta. Creo que lo hice correctamente pero, en lugar de deducir que iba a obtener $\boldsymbol{t}$, hice las operaciones de exponenciación binaria etc. El examen daba la opción de obviar las operaciones y justificar la respuesta si eras capaz de deducir por dónde iban los tiros. Algo me olía, pero, tras hacer las operaciones y pese a los nervios; me emocioné mucho cuando todo cuadró, no porque viese que lo había realizado correctamente - si no porque las matemáticas son preciosas. Haciendo la misma operación - **cifras y descifras**.

3.2.4. Ejemplos de Laboratorio en R

```
##############################
######### GENERACIÓN #########
#########    DE      #########
#########  CLAVES    #########
##############################

# Paquetes que son de gran ayud para la utilización
# de números primos inversos modulares etc.

library("numbers")
library(help="numbers")

# Veamos varias funciones que implementamos en el laboratorio
# muy descriptivas que nos ayudarán a reproducir el proceso
# de generación de claves públicas, cifrado y descifrado.

primo_relativo <- function(m,umbral)
{
    num<-umbral;
    while(GCD(m,num)!=1)
    {
        num<-num+1;
    };
    return(num);
}

claves_grandes_RSA <- function(a,b)
{
    if(!isPrime(a))
    {
        pr = nextPrime(a)
        p = pr
    }
    else p = a

    if(!isPrime(b))
    {
        pr = nextPrime(b)
        q = pr
    }
    else q = b

    n = p*q
    m = (p - 1)*(q - 1)
    r = primo_relativo(m)
    s = modinv(r, m)

    return(c(n,r,s))
}
```

Como detalle, gracias al paquete "numbers", R nos permite **factorizar** enteros con la funcíon *primefactors*

con un límite de $\mathbf{2^{53} - 1}$.

```
##############################
#########  CIFRADO  ##########
#########     Y     ##########
######### DESCIFRADO #########
##############################

# Paquetes que son de gran ayuda
# para la gestión de números primos
# inversos modulares etc.

library("numbers")
library(help="numbers")

# Veamos varias funciones que implementamos en el laboratorio
# muy descriptivas que nos ayudarán a reproducir el proceso
# de generación de claves públicas, cifrado y descifrado.

codifica <- function(txt)
{
    strtoi(charToRaw(txt),16L)
}

decodifica <- function(codetxt)
{
    rawToChar(as.raw(codetxt))
}

cifrar <- function(codevector,r,n)
{

    vcifrado = codevector
    for (i in 1:length(codevector)) # codevector[i]^r mod n
      vcifrado[i] = modpower(codevector[i], r, n)

    print(vcifrado)
    return(vcifrado)
}

descifra <- function(vectorcifrado,s,n)
{
    cifrar(vectorcifrado, s, n)
}

# Ejemplo, asumo que la variable texto ya tendrá
# un valor correspondiente

n<-9797
r<-7
s<-2743

texto_cifrado <- cifrar(codifica(texto), r, n )
texto_plano <- decodifica(descifra(texto_cifrado, s, n))
}
```

3.3. Problema RSA

Si se hacen pública la **clave privada**, es obvio que cualquier persona puede descifrar el mensaje por el método visto anteriormente. Si se hacen públicos $\boldsymbol{p}$ y $\boldsymbol{q}$, se puede obtener $\boldsymbol{\phi(n)}$, y con ello sabiendo que disponemos públicamente de $\boldsymbol{r}$; el inverso modular mediante el algoritmo de Euclides extendido. El inverso modular de $\boldsymbol{r}$ será $\boldsymbol{s}$, y podremos descifrar sin inconveniente alguno.

A su vez, conociendo $\boldsymbol{\phi(n)}$, se podría llegar a obtener la **clave privada** a través del mismo método.

Si únicamente disponemos de la *clave pública* (n, r), no nos queda otra posibilidad más que **factorizar** $\boldsymbol{n}$ para intentar obtener los dos números primos $\boldsymbol{p}$ y $\boldsymbol{q}$. Y he aquí la problemática general del algoritmo RSA - la posibilidad de conseguir factorizar $\boldsymbol{n}$ cuando $\boldsymbol{p}$ y $\boldsymbol{q}$ son números muy grandes, es teóricamente posible, pero - en la práctica; es un problema de eficiencia dada la carencia de recursos que puedan realizar semejante cómputo, no es realmente factible en un tiempo apropiado.

Digamos pues, que dada la **insolubilidad** de la clase de problemas que requieran factorización o logaritmos discretos - RSA es seguro... ¿o no?.

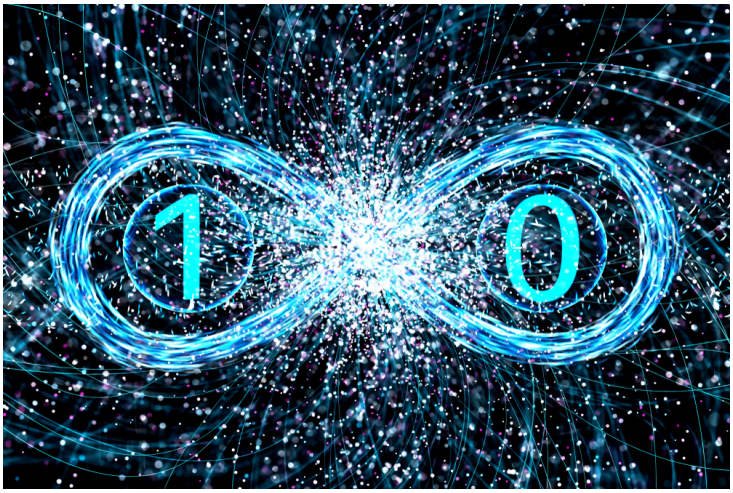
4. La Criptografía en el (ya no tan lejano) futuro

4.1. Física Cuántica

Ahora que parecía estábamos llegando a uno de los puntos álgidos en la historia de la humanidad donde la comprensión y percepción de nuestra realidad era más certera o incluso axiomática - van y lo cambian. A volver a empezar.

Las leyes de la Física se han visto sometidas a minuciosos estudios desde los anales de la historia por las mejores mentes que han pisado éste planeta. Esto ha llevado al no poco fascinante y tremendamente complejo descubrimiento de la Física Cuántica, la cual cambia completamente nuestra concepción de multitud de cosas, ya que las leyes que, al menos desde nuestro punto de vista, rigen nuestro universo - no son enteramente válidas cuando las extrapolamos al mundo subatómico.

Cómo es lógico, éste atractivo nuevo mundo ha generado una gran multitud de lineas de investigación y, en éste contexto, una de ellas nos interesa especialmente: **La Computación Cuántica**.



4.1.1. El Qubit y la computación Cuántica

La computación cuántica permite implementar de manera física equipos que se aprovechen de las características o comportamiento cuántico para poder incrementar de manera exponencial la capacidad de cómputo, además de introducir nuevos paradigmas en la programación de algoritmos.

Así como la computación clásica tiene como unidad mínima el bit, la computación cuántica dispone del Qubit (Quantum bit). El cual tiene un gran ventaja con respecto al primero, al aprovecharse de comportamientos que únicamente se dan a nivel de partículas subatómicas.

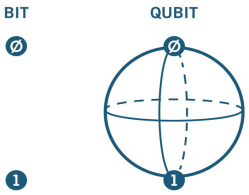


Figura 4: Qubit

4.1.2. Principio de Superposición, entrelazamiento y coherencia.

Superposición de estados

Como introducción a éste concepto podemos recurrir a la idea que propuso Schrödinger mediante el ejercicio mental del famoso *gato de Schrödinger*. Partiendo de la base de que una partícula puede adoptar únicamente dos estados, como hasta hace poco inducía la Física Clásica, consideremos la posibilidad de que mientras no observemos dicha partícula, ésta **estará en ambos estados simultáneamente** y una vez observada, adquirirá uno de los dos el cual nosotros seremos capaces de percibir.

- Ésta es la idea fundamental de la **superposición de estados**; una partícula puede encontrarse en todos los posibles estados simultáneamente hasta que un observador la perciba.

Esto es lo ocurre con los qubits, pueden estar simultáneamente en los dos estados que su antecesor el bit era capaz de adoptar. Si utilizamos el qubit como unidad mínima de cálculo, podemos intuir gracias a este concepto que la capacidad de cómputo del equipo que lo aloje, se verá **incrementado exponencialmente** con respecto a la computación clásica. Por ejemplo, un equipo clásico tardaría millones de años en encontrar los factores primos de un número de 2048 bits, pero los qubits lo harían en cuestión de minutos.

Los qubits son frágiles, existe ruido o incoherencia que los puede hacer inestables y por ende, proporcionar resultados incorrectos. Para incrementar ésta coherencia actualmente se están trabajando en el desarrollo de equipos y métodos de corrección de errores.

De ésta manera, veamos cómo se representan los estados cuánticos del qubit, mediante la notación Dirac o 'bra-ket', que representan los estados del *vector bidimensional* que reside en un espacio vectorial complejo:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

Donde α y β son números complejos que representan las amplitudes de probabilidad de que cada uno de esos estados, los cuales vienen determinados por el segundo axioma de la teoría de probabilidades:

$$|\alpha|^2 + |\beta|^2 = 1.$$

El estado puro del qubit se logra cuando existe un ket completo, lo cual define una superposición coherente de ambos estados y las probabilidades del estado de superposición vienen determinadas por:

$$\alpha = \cos\left(\tfrac{\theta}{2}\right) \quad y \quad \beta = e^{i\phi}\sin\left(\tfrac{\theta}{2}\right)$$

Vemoas más detalles mediante la esfera de Bloch:

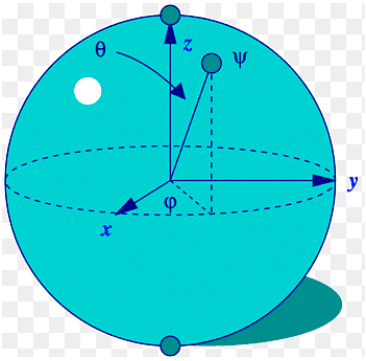


Figura 5: Esfera de Bloch

- El ángulo θ representa la superposición de $|0\rangle$ y $|1\rangle$, donde $0 \leq \theta \leq \pi$
- Y el ángulo ϕ representa la fase del qubit, donde $0 \leq \phi < 2\pi$

Es un espacio geométrico bidimensional, en los cuales tienen dos grados más de libertad (ϕ y ψ). Para poder describir el estado de un sistema de n componentes, en la física clásica se podría hacer con únicamente n bits, sin embargo, en la cúantica se necesitan 2^{n-1}.

Con lo que, la computación cuántica, lo que trata es de que un ordenador con n qubits, pueda coexistir en un estado de superposición cuántica con sus 2^n estados lógicos, simultáneamente.

$|00...|0\rangle$
$|11...|1\rangle$

Entrelazamiento

Otra característica cuántica que resulta fundamental en éste nuevo tipo de computación, es el **entrelazamiento**.

El entrelazamiento es un fenómeno que no ocurre en el mundo macroscopico, si no que únicamente se da a nivel de las partículas subatómicas - en el caso que vamos a tratar, estas partículas serán **fotones de luz**. Es una correlación entre dos partículas que dicta el estado del sistema. Ésto se podría reducir a que, si tenemos un sistema con dos partículas y una de ellas experimenta y cambio de estado, la otra partícula lo experimentará también.
Debido a éste efecto, también se le denomina teleportación cuántica.

Actualmente los científicos están trabajando en mejorar y corregir las posibles pérdidas de coherencia cuántica para estabilizar los sistemas y puedan resultar tan eficientes como se espera de ellos.

4.1.3. Algoritmos de Shor y Grover (1994, 1996)

Llegados a éste punto, podemos comenzar a realizar nuestro propio *entrelazamiento* y ver cómo todo quiere empezar a encajar.

En 1994, el Americano Peter Shor (profesor de Matemática Aplicada en el M.I.T), hizo un descubrimiento un tanto importante: consiguió diseñar un algoritmo cuántico el cual, dado un entero N, obtiene sus factores primos. Es decir, éste algoritmo radica en la factorización de cualquier entero, no importa cuán grande sea.

También existe la versión clásica de éste algoritmo, pero, como se ha mencionado previamente, no es eficiente. Ésta versión, utiliza el Algoritmo de Euclides y su Aritmética Modular. Sin embargo, la versión cuántica del algoritmo de Shor sí que es eficiente, y mucho. Pero, al menos hasta el momento, de manera teórica, ya que no existen los medios físicos que lo puedan ejecutar.

El algoritmo de Shor tarda un tiempo polinómico y se necesitan el siguiente número de puertas cuánticas utilizando multiplicaciones rápidas:

$$O\big((\log N)^2(\log\log N)(\log\log\log N)\big)$$

Éste algoritmo, explota la característica cuántica de la **superposición**, ya que se basa en encontrar la frecuencia o periodo con la que se da una función, y para ello necesita que el ordenador cuántico esté en muchísimos estados simultáneamente. Esto permite evaluar la función en muchos puntos a la vez.

Si un ordenador cuántico con un número suficiente de qubits pudiese operar sin sucumbir al ruido cuántico y otras problemáticas de pérdida de **coherencia**, el *Algoritmo de Shor* funcionaría.

Así como también lo haría el Algoritmo de Grover (por Love K. Grover, 1996), cuya funcionalidad radica en la capacidad de invertir una funcíón:

$$y = f(x)$$

Dado un valor y, podríamos obtener x mediante un método que sobrepasa la eficiencia de las soluciones clásicas.

4.1.4. El dilema

La computación cuántica está siendo desarrollada a un ritmo sin precedentes, los gigantes tecnológicos destinan gran parte de sus inversiones a esta fascinante nueva tecnología. Ya es una realidad, y está cerca.

Hace tan sólo un par de días, China anunció de manera oficial que ya dispone de su primer ordenador cuántico, el cual es capaz de realizar en tan sólo un milisegundo lo que el ordenador más potente de la actualidad tardaría **30 billones de años**.

Pensar esto es vertiginoso, pero, ¿qué significa esto? Pues muchas cosas, y entre ellas que Algoritmos como el de Shor o Grovel podrán ser implementados y, con ello sistemas que radican en la solubilidad ineficiente de ciertos problemas como es la factorización de números enteros, como el RSA, podrían ser rotos en un tiempo irrisorio.

Teniendo en cuenta que, estos sistemas criptográficos son el pilar de la seguridad entre nuestra comunicaciones, nos encontramos ante un gran pero a su vez emocionalmente dilema.

4.1.5. Criptografía Cuántica

Aunque aún está en una fase de desarrollo muy temprana, con la idea de la computación cuántica se comenzó a investigar paralelamente cómo no sucumbir a los problemas que podrían surgir con la llegada de la misma.

Si la computación cuántica iba a permitir la implementación de técnicas que romperían la integridad de nuestros sistemas criptográficos en segundos, había que buscar una alternativa robusta a esos sistemas que se adaptara al nuevo tipo de tecnologías.

Veamos brevemente en qué consiste la idea actual de éste nuevo sistema criptográfico.

El fotón

Dada su naturaleza cuántica intrínseca, la partícula asociada a estos nuevos criptosistemas es el fotón. Gracias a su comportamiento y capacidad de superposición de estados, no es posible su observación sin generar una perturbación (recordar el caso del gato de Schrödinger basado en el *principio de incertidumbre* de Heisenberg). Lo cual, con los métodos apropiados se puede implementar un **sistema inquebrantable**.

Pero claro, ¿cómo se puede enviar la información deseada mediante fotones a alguien? Bien, imaginémonos las diferentes posibilidades de **polarización** de las que dispone un fotón, es decir, la dirección hacia la que vibrará el campo eléctrico del mismo. Cada uno de estos espines, representará información - o 1 o 0. Por lo que para enviar cierta información, deberíamos de enviar una sucesión de fotones. Las comunicaciones se efectuarán por un canal cuántico privado, y otro convencional totalmente público.

Veamos la siguiente imagen:

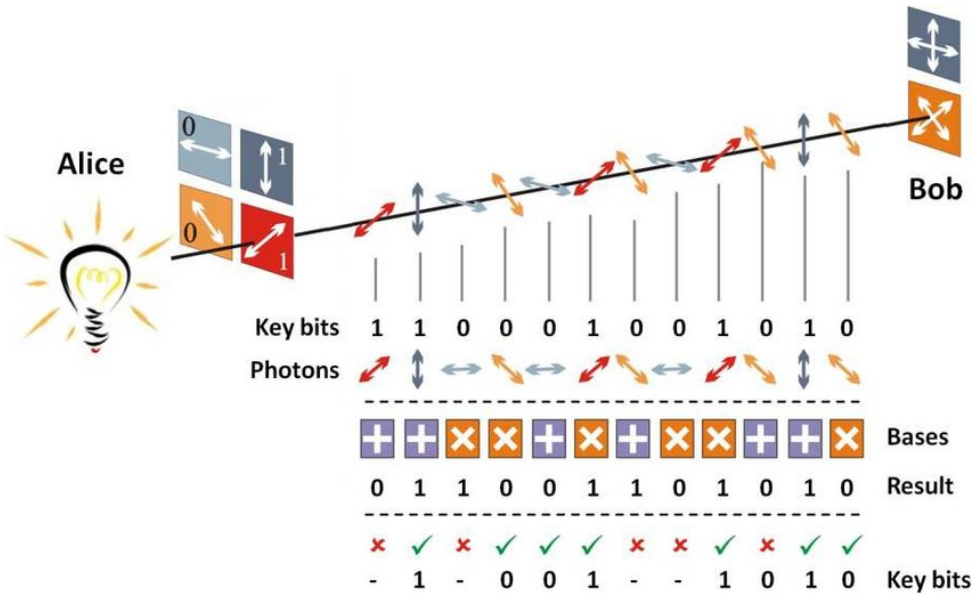


Figura 6: Registro de polaridades

Emisión (canal cuántico privado): Cuando el receptor envía dicha sucesión de fotones, estos mismos pasarán por unos filtros los cuales forzarán o corregirán la polarización de los mismos en el caso de ser diferente. Cuando esto ocurra, el receptor registrará dichas polarizaciones.

Recepción (canal cuántico privado): Por su parte, el emisor recibirá dicha secuencia sin realmente saber las polarizaciones, con lo que pasarán por los filtros de recepción, los cuales pueden ser o no iguales que con los que el emisor envió, con un 50 % de posibilidad de error.

Intercambio (canal público): Ahora es cuando viene el momento de contrastar los registros de las polarizaciones de la secuencia de fotones entre ambas partes. El receptor comunica al emisor qué filtros han sido utilizados para la recepción, y éste último le responde de vuelta con cuales han sido los filtros que ha escogido correctamente. De ésta manera, ambos podrán saber qué fotones serán parte de la clave y cuales se desecharan.

Viendo esto y recordando la naturaleza de superposición de los fotones así como la de la coherencia cuántica, podemos comprender cómo si una tercera persona hubiese interceptado u observado dichos fotones una vez registradas las polaridades, el estado de los mismos sería perturbado y habría inconsistencias una vez el emisor y el receptor intercambien sus datos registrados. Es decir, verían que la información ha sido alterada, por el simple hecho de haber existido un 'man in the middle'.

Para poder implementar de manera universal una idea tan ingeniosa como esta, aún queda mucho trabajo por hacer, en China ya han construido plantas en las cuales gracias a repetidores se consigue alargar la distancia a la que el envío de fotones es posible entre dos puntos.

4.1.6. A hombros de gigantes

Hace unos años, el NIST (National Institute of Standards and Technology) de Estados Unidos, comenzó un proceso para solicitar, evaluar y estandarizar uno o más algoritmos cuánticos que puedan resistir algoritmos de clave pública en la era ***Post-Cuántica***.

A esta iniciativa se han sumado docenas de soberbios grupos de investigación los cuales están planteando soluciones muy interesantes e innovadoras. Actualmente se encuentran en la ronda 3 y de entre los finalistas, se encuentra un equipo llamado SABER (website) cuya propuesta consiste en un sistema de retículos (lattice) mediante aprendizaje modular con redondeo (Modular Learning with rounding).

Con objeto de encontrar soluciones a problemas sabemos no tardaran en llegar; en una de las competiciones con más prestigio a nivel mundial; uno de los equipos finalistas presenta una propuesta que gira en torno a, entre otras cosas, las enseñanzas de Euclides.

> "*Si he visto más lejos es porque estoy sentado sobre los hombros de gigantes.*"
>
> –Sir Isaac Newton

4.1.7. La singularidad de la aritmética modular

Algo que era inevitable y ha conseguido la Física Cuántica es que nos replanteemos todo lo conocido, de tal manera que hoy en día hay estudios tan espectaculares y aparentemente extraños que incluso sugieren que nuestra consciencia pueda ser fruto de conexiones cuánticas debido a patrones fractales dentro de las neuronas.

Pese a que todo esto puede sonar a ciencia ficción ahora mismo, estamos en una época pre-cuántica, pre-qubit, época que está desafiando las leyes que rigen nuestro universo e incluso hace que nos planteemos la naturaleza de nuestra consciencia o realidad. Al mismo tiempo seguimos mirando al futuro en busca de soluciones, pero sin olvidarnos del pasado y utilizando las grandes ideas de ilustres como Euclides; Es maravilloso pensar que, en ésta época pre-cuántica, sus escritos aún nos estén ayudando.

Este es el momento donde se produce una **singularidad (espacio-temporal)** de la Aritmética Modular, donde el pasado se da la mano con el presente para mirar al futuro.

5. Conclusiones

5.1. Reflexión

El retomar los estudios para mí ha sido una decisión muy importante. Tenía claro que lo que me gusta está orientado a la Criptografía, Inteligencia Artificial y Videojuegos. Éste trabajo me ha permitido inmiscuirme de nuevo en éste entorno científico de investigación, pasando horas leyendo diferentes fuentes, tocando código, viendo vídeos / documentales, tomando notas, aprendiendo LaTeX... No ha hecho más que recordarme que, este ambiente es lo mío; es con lo que más disfruto y a lo que me quiero dedicar.

No conocía el concepto de Módulo. Tras las clases sobre Aritmética Modular, se ha despertado en mí un interés grandísimo sobre este tema y al investigar, he descubierto un mundo. Me encanta y seguiré con ello.

He entrado de nuevo sabiendo que optaré por la *mención en computación* y efectivamente, así será. La pregunta es - ¿y después? ¿*Criptografía*? ¿*IA*? Ya lo iremos viendo, poco a poco - quiero reforzar y ampliar mis bases matemáticas lo máximo posible. Supondría un gran reto para mí el poder dedicarme a algo relacionado con la Criptografía, pero estoy seguro de que volcaría toda mi pasión y ganas en ello.

Lo que está claro es que mi universo lo definirá $\pmod{C}$, siendo C no enteros si no las diferentes ramas de la computación.

Eskerrik asko por la oportunidad de hacer éste trabajo.

5.2. Opinión personal

5.2.1. Sobre la Criptografía

La criptografía está en constante evolución y actualmente nos encontramos en un punto crítico. Es un momento muy emocionante en el cual es complicado dilucidar qué ocurrirá - con lo que, es un muy buen momento para especializarse en este arte y disfrutar de todos los entresijos que lo rodean ya que, no solo el ingenio si no también la creatividad jugará un papel muy importante en el porvenir.

Indagando en estos temas, se me venía a la cabeza la frase que dice: “Cuanto más sabes, más te das cuenta de que no sabes nada” - yo agregaría - “, pero a su vez - ¡más quiero saber!”.

El mundo criptográfico es abrumador y tremendamente complejo, pero es tan bonito que es inevitable sentir pasión por él.

Al parecer, el tema de la *Criptografía Cuántica* está aún muy verde, y aún existen algunas personas con dudas de que realmente vaya a dejar obsoletos los sistemas asimétricos de claves públicas. Habrá que mantenerse actualizados e ir preparándonos todo lo posible, por si acaso.

Será interesante seguir los resultados que vaya publicando el NIST en futuras rondas, veamos los cambios que van presentando los participantes y cuales se convierten en los estándares Post-Quantum.

5.2.2. Laboratorio de Matemática Discreta

La verdad es que he disfrutado muchísimo de los Laboratorios. Me han encantado y creo que son realmente útiles, es una manera de poder ver una aplicación directa de lo que estamos viendo en clase.

Sobre la organización, pues muy buena la verdad, acorde con lo que estábamos viendo en clase. Quizá algo rápido en algunos momentos, al menos así me lo pareció el día que vimos la función de Euler. Pero nada especial, luego en casa, al reorganizar pensamientos te das cuenta de que ha quedado todo claro.

Tan sólo decir, que hay una cosa *no* me ha gustado del Laboratorio; ¡que se haya terminado!

5.3. Making of: De Euclides al Qubit.

He intentado diseminar toda la información y crear una estructura que la presente de manera cronológica y ordenada, pero con un factor común en todo momento: Euclies y su Aritmética modular. No sé si he conseguido sea obvio el guiño, pero la idea es transmitir la *'existencia'* de un viaje en el tiempo, desdoblar el momento en el que se está avanzando al futuro (algoritmos para la criptografía cuántica) mediante la utilización de técnicas matemáticas ancestrales (Aritmética modular).

Un pequeño guiño para hacer el proyecto un poco más entretenido y con juego para el lector; *'Euclides toca nuestro presente a través de un agujero de gusano'.*

Bien, para el desarrollo de trabajo, he utilizado muchas herramientas diferentes durante la semana y media que he estado investigando, redactando .etc.

Sistemas Operativos
Como sistemas operativos, he utilizado GNU/Linux Debian 11 Bullseye (sobremesa), y MacOS Big Sur (portátil nuevo, primera vez utilizando MacOS).

Entornos o IDEs
En éste caso únicamente he utilizado R Studio para Linux (Debian), con el fin de poder cacharrear con las funciones que realizamos en el laboratorio.

LaTeX

Para el marcado con LaTeX, he utilizado TexMaker, pero luego pasé a Overleaf (y me ha gustado bastante). Como referencia, he recurrido tanto a Stack Overflow como Manual de Latex. No había utilizado nunca - y tras utilizarlo por primera vez; me ha encantado. Es una herramienta terriblemente potente y ahora entiendo porque hace tantos años que venía escuchando hablar de ello.

Música
Como dato extra, mientras he realizado el trabajo he estado escuchando las siguiente bandas sonoras:

- Enigma soundtrack - (Película Engima 2001)
- A Beautiful mind soundtrack - Película Una mente maravillosa (2001)
- The Imitation Game - Película, tema principal (2014)

6. Bibliografía

Referencias

https://community.ibm.com/HigherLogic/System/DownloadDocumentFile.ashx?DocumentFileKey=6cd7c160-a884-b2fb-b682-b15947a84c73

https://es.wikipedia.org/wiki/Cúbit
https://www.monografias.com/trabajos76/utilidad-aritmetica-modular-sistemas-criptograficos/utilidad-aritmetica-modular-sistemas-criptograficos2.shtml

https://www.slideserve.com/quasar/richard-cleve

https://www.nist.gov/news-events/news/2020/07/nists-post-quantum-cryptography-program-enters-selection-round

https://cybercamp.es/sites/default/files/contenidos/videos/adjuntos/cybercamp2017-las$_m atematicas_e n_l a_e volucion_d e_l a_c$

Docuemnt RSA-Gazteleraz.pdf de eGela

https://es.wikipedia.org/wiki/Algoritmo$_d e_S hor$

https://www.xatakaciencia.com/matematicas/ideas-matematicas-que-han-influido-en-la-historia

https://es.wikipedia.org/wiki/Euclides

https://www.bbvaopenmind.com/tecnologia/mundo-digital/entender-la-criptografia-cuantica/

https://azure.microsoft.com/es-es/overview/what-is-quantum-computing/how-it-works

https://www.techedgegroup.com/es/blog/introduccion-computacion-cuantica-ii

https://www.fayerwayer.com/2013/09/qubit-la-unidad-fundamental-del-futuro-informatico-y-tecnologico/

https://manualdelatex.com/simbolos

https://es.wikipedia.org/wiki/Cifrado$_d e_V igenère$

https://link.springer.com/book/10.1007/978-3-319-59879-6

Diffuse helman

http://www.criptored.upm.es/crypt4you/temas/RSA/leccion3/leccion03.htmlapartado3-2

https://theconversation.com/puede-explicarse-la-conciencia-con-fisica-cuantica-166510

https://www.uco.es/ ma1beyem/tema6.htm

https://ww2.mathworks.cn/matlabcentral/mlc-downloads/downloads/submissions/32852/versions/1/screenshot.jpg

https://www.codex.com/wp-content/uploads/2019/08/quantumcomputingmedium.jpg

https://www.researchgate.net/profile/Alberto$_C arrasco-Casado/publication$/309731586/$figure/fig$11/$AS$: 669033707872267@1536521474236/$BB84-protocol-basic-scheme.jpg$