Centro de Investigación y de Estudios
Avanzados del Instituto Politécnico Nacional

Unidad Zacatenco

Departamento de Computación

De la búsqueda de endomorfismos eficientes en curvas elípticas binarias

Tesis que presenta

Daniel Idelfonso Cervantes Vázquez

para obtener el Grado de

Maestro en Ciencias en Computación

Director de tesis:

Dr. Francisco José Rambó Rodríguez Henríquez

Ciudad de México **Agosto 2016**

Agradecimientos

Al Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, particularmente al Departamento de Computación. Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por el apoyo económico durante estos dos años de estudios.

A Thomaz Oliveira, al Dr. Francisco (mi asesor), al Dr. Luis Julian Dominguez y al Dr. Guillermo Morales(ambos mis sinodales), por sus comentarios, correciones y recomendaciones.

A Sofía Reza, por su ayuda en cuestiones administrativas y su gran apoyo durante mi estancia en este departamento.

A mi familia por su constante apoyo sin el cual, este tiempo hubiera sido más difícil.

Resumen

En los últimos años el uso de curvas elípticas en criptografía ha ganado popularidad, debido a que permiten usar llaves criptográficas cortas. En particular, las curvas elípticas binarias ofrecen una característica adicional, pueden implementarse de forma eficiente en los procesadores actuales ya que la aritmética en campos binarios permite aprovechar su arquitectura. La operación fundamental en los protocolos criptográficos basados en curvas elípticas es la multiplicación escalar de puntos, la cual es también la más costosa. En esta tesis se realiza un análisis de eficiencia y seguridad de dicha operación en dos familias de curvas. La primera familia consiste de las llamadas curvas de Galbraith-Lin-Scot (GLS) binarias, que pueden ser equipadas con un endomorfismo eficiente para acelerar la multiplicación escalar. Para estas curvas presentamos dos construcciónes de endomorfismos, la primera es una alternativa del endomorfismo GLS y la segunda es la construcción de una familia de endomorfismos, ambas construcciones permiten realizar el cálculo de la multiplicación escalar de una manera más rápida; la segunda familia que se presenta es la conformada por las curvas anómalas binarias, también llamadas de Koblitz, estas curvas se caracterizan por hacer uso del endomorfismo de Frobenius para acelerar la multiplicación escalar. En esta familia se presenta un nuevo endomorfismo sobre extensiones del campo $\mathbb{F}_4$, que requiere la mitad de las operaciones que el de Frobenius. Además, se presenta un análisis de seguridad de las posibles vulnerabilidades que surgen al usar endomorfismos para el cálculo de la multiplicación escalar.

Abstract

Nowadays, the use of elliptic curves in cryptography has gained popularity since they allow to use short cryptographic keys. In particular, binary elliptic curves can be implemented efficiently in current processors because their field arithmetic can take advantage of processor architecture. The most important operation in a elliptic curves based protocol is scalar point multiplication, which is also the most expensive. In this thesis we show an efficiency and security analysis of the scalar multiplication in two curve families. The first family is the binary Galbraith-Lin-Scot (GLS) family, which can be equipped with an efficient endomorphism to accelerate the scalar multiplication. We show two endomorphism constructions, at first we present an alternative to the GLS endomorphism followed by an endomorphism family propposed by us, both constructions allow acelerate the scalar multiplication; In addition, we present a security analysis due to the potential vulnerabilities of using endomorphism in scalar multiplication. The second family consist of the anomalous curves also know as Koblitz curves, these curves are characterized by the use of the Frobenius endomorphism to accelerate scalar multiplication. In this family we present a new endomorphism that works on the $\mathbb{F}_4$ finite field extensions, which requires half of the Frobenius operations.

ÍNDICE GENERAL

Agradecimientos i

Resumen iii

Abstract v

Índice de figuras xi

Índice de tablas xiii

Índice de algoritmos xv

1. Introducción 1
- 1.1. Criptografía de llave pública. . . . 3
  - 1.1.1. Curvas elípticas 5
- 1.2. Estado del arte 7
- 1.3. Planteamiento del problema y objetivos 7
- 1.4. Organización del documento 9

2. Contexto matemático 11
- 2.1. Grupos 12
  - 2.1.1. Grupos cíclicos 15
  - 2.1.2. Clases laterales 17
  - 2.1.3. Homomorfismos 19
  - 2.1.4. Teoremas de homomorfismos 23
- 2.2. Anillos 23
  - 2.2.1. Teorema de Fermat y Euler 27
- 2.3. Retículos 27

2.3.1. Espacios vectoriales . . . 27
2.3.2. Aspectos básicos de retículos . . . 32
2.3.3. Algoritmos para reducción de retículos . . . 34

3. Curvas elípticas **37**
3.1. Curvas elípticas binarias . . . 40
3.2. Isogenias . . . 41
3.2.1. La fórmula de Vélu . . . 43
3.2.2. Polinomios de División . . . 44
3.3. Multiplicación Escalar . . . 45
3.4. Endomorfismos eficientes . . . 50
3.4.1. Aplicación a la multiplicación escalar . . . 51
3.4.2. Descomposición escalar . . . 52
3.4.3. Curvas de Koblitz . . . 53

4. Análisis y desarrollo **57**
4.1. Convenciones y notación . . . 59
4.2. Curvas GLS binarias . . . 59
4.2.1. Nuestra construcción . . . 60
4.2.2. El polinomio característico . . . 62
4.2.3. Un nuevo endomorfismo . . . 63
4.3. Análisis de Seguridad . . . 68
4.3.1. 2-recomposición . . . 68
4.3.2. 3-recomposición . . . 69
4.4. 4-recomposición . . . 70
4.5. Curvas de Koblitz . . . 70
4.5.1. Una mejora ante Frobenius . . . 71
4.6. Análisis de eficiencia . . . 73
4.6.1. Análisis 3-GLV y 4-GLV en $\mathbb{F}_{q^2}$ con $q = 2^{127}$ . . . 74
4.6.2. Curvas de Koblitz en $\mathbb{F}_4$ . . . 77

5. Conclusiones y trabajo a futuro **79**
5.1. Conclusiones . . . 79
5.2. Trabajo a futuro . . . 80

Apéndices **86**

A. Fundamentos matemáticos **89**
A.1. Conjuntos . . . 89
A.1.1. Particiones y relaciones de equivalencia . . . 92

A.1.2. El conjunto de los enteros 93
A.2. Anillos de polinomios 96

B. Códigos **99**

C. 3-Decomposición **103**
C.1. 3-GLV 103
C.1.1. Ejemplo pequeño 104

ÍNDICE DE FIGURAS

1.1. Modelo básico de comunicación 2
1.2. Intercambio de llaves de Diffie-Hellman. 4
1.3. Esquema de Criptografía de llave pública 5

2.1. Teorema fundamental de los homomorfismos 23

3.1. Tablas del campo $\mathbb{F}_4$ 40

4.1. Diagrama de construcción de $\psi = \phi(\pi(\phi^{-1}))$. 61
4.2. Diagrama de construcción de $\psi = \phi(\pi)$. 63
4.3. Diagrama de construcción de $\varphi = \phi(\pi(\nu))$. 64
4.4. Diagrama extendido de la construcción de $\varphi = \phi(\pi(\nu))$. . . . 65
4.5. Diagrama de construcción de $\hat{\tau} = \phi(\pi^{(2)})$. 72

A.1. Ejemplo de una función entre los conjuntos X e Y. 92

ÍNDICE DE TABLAS

2.1. Tabla de grupo de un grupo $(G, *)$ 14
2.2. Ejemplo 2.1.9 14
2.3. Ejemplo 2.1.21 16
2.4. Ejemplo de ejecución del algoritmo gausiano 35

4.1. Tiempos (en ciclos de reloj) para la arimética de campo y operaciones de curva elíptica en una plataforma Intel Sandy Bridge 74
4.2. Comparación entre 2GLV, 3GLV y 4GLV 77
4.3. Conteo de operaciones requeridas por φ_3 77
4.4. Operaciones requeridas por τ y $\hat{\tau}$ 78

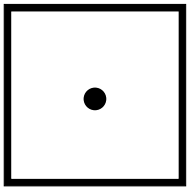
ÍNDICE DE ALGORITMOS

2.3.1.Gram-Schmidt 32
2.3.2.Aproximación de Babai 34
2.3.3.Reducción gausiana para retículos. 34
2.3.4.Reducción de retículos LLL 36

3.3.1.Método binario para multiplicación escalar. 46
3.3.2.Cálculo de NAF para un entero positivo. 48
3.3.3.Método binario NAF para multiplicación escalar. 48
3.3.4.Truco de Shamir-Strauss 49
3.4.1.2-Descomposición escalar balanceada 53
3.4.2.τ-NAF [29] 55
3.4.3.Multiplicación escalar en base τ 55

4.6.1.Multiplicación escalar protegida 2GLV 75
4.6.2.Multiplicación escalar protegida 3GLV 76
4.6.3.Multiplicación escalar protegida 4GLV 76

A.1.1Euclides extendido 95



Capítulo 1

Introducción

-How long do you want these messages to remain secret?[...]
+I want them to remain secret for as long as men are capable of evil.[1]

Neal Stephenson
-Cryptonomicon

Desde los tiempos antiguos la raza humana ha tenido la necesidad de ocultar cierta información que en su momento consideró valiosa. En ocasiones es deseable compartir esta información sólo a un determinado receptor o conjunto de receptores. Por ejemplo en el caso militar, es indispensable que únicamente el bando aliado pueda conocer los secretos. Un problema significativo dentro de este escenario es que los canales de comunicación son en su mayoría inseguros. La inseguridad radica en que un posible adversario

[1] -¿Por cuanto tiempo quieres que estos mensajes sigan siendo secretos?[...]
+Quiero que sigan siendo secretos mientras los hombres sean capaces del mal.

podría ver la información o incluso modificarla. En la figura 1.1 se observa el escenario usual al querer compartir alguna información.

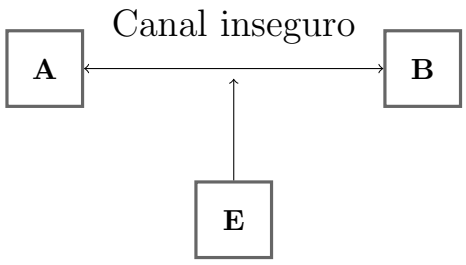


Figura 1.1: Modelo básico de comunicación
El emisor **A** envia información al receptor **B**, al ser un canal inseguro un intruso **E** puede interceptar el mensaje, verlo o incluso, modificarlo.

Con el paso de los siglos la raza humana ha ido perfeccionado la manera de ocultar la información y de esta forma obtener *seguridad*. La seguridad carece de una definición concreta. El motivo de esta carencia es que cierta aplicación puede ser segura bajo ciertos términos, sin embargo otra no necesariamente es segura bajo estos mismos términos. La seguridad puede definirse en términos de los siguientes *servicios de seguridad*:

- *Confidencialidad*. Garantizar el acceso y manipulación de la información únicamente por las entidades autorizadas.
- *Integridad*. Asegurar que la información no sea, ni ha sido, modificada por alguna entidad no autorizada.
- *Autenticación*. Garantizar que una entidad es quien dice ser.
- *No Repudio*. Evitar que una entidad niegue alguna acción o compromiso efectuado.

La criptografía se encarga del estudio y diseño de técnicas matemáticas que permitan entablar comunicaciones seguras a través de un canal inseguro. En un principio la manera en la que la criptografía actuaba para proveer seguridad en las comunicaciones esta dada por lo siguiente:

- Ambas entidades establecen un *secreto en común*. A este secreto lo denominaremos *llave*.
- La información a enviar (*texto claro*) se *cifra* con la llave.

••

- El receptor descifra el mensaje recibido usando la llave.

Los primeros cifrados conocidos fueron los cifrados por sustitución, es decir se cifraba sustituyendo los simbolos del lenguaje por otros simbolos a manera de una permutación del alfabeto. En este caso la llave consistía en la permutación a utilizar. Un claro ejemplo es el cifrado *César* el cual consistía en rotar el alfabeto un cierto número de veces.

Ejemplo 1.0.1 (Cifrado César)**.** Cifrado César usando una rotación de 3. En este caso la llave es 3.

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
$\hookleftarrow$ x 3
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

Texto claro: CESAR
Texto cifrado: FHVDU

Posteriormente surgieron cifrados por trasposición, en los cuales no se sustituían los símbolos por otros sino que se cambiaba el orden de éstos[2]. Cabe destacar que para decifrar era necesario contar con la misma "llave" con la cual se cifró. Las técnicas de cifrado fueron mejorando con el paso del tiempo. A su vez, la realización de comunicaciones a distancias enormes fué posible. Los cifrados de aquella época, a los cuales llamaremos cifrados clásicos o simétricos, carecían de una manera segura de establecer llaves entre 2 entidades a distancia. Este problema abierto se solucionó cuando en 1976, Whitfield Diffie y Martin Hellman propusieron un protocolo criptográfico para intercambiar llaves de manera segura. El protocolo de intercambio de llaves de Diffie-Hellman sigue el esquema presentado en la figura 1.2. El intercambio de llaves de Diffie-Hellman basa su seguridad en un problema matemático conocido como *el problema del Logaritmo Discreto*.

1.1. Criptografía de llave pública.

Diffie y Hellman mediante su protocolo de intercambio de llaves abren las puertas a nuevos protocolos de cifrado llamados de *llave pública*. Estos protocolos tienen la característica de contar con un par de llaves a diferencia

[2] Algunos cifrados modernos como *DES* o *AES* utilizan ambas técnicas.

...

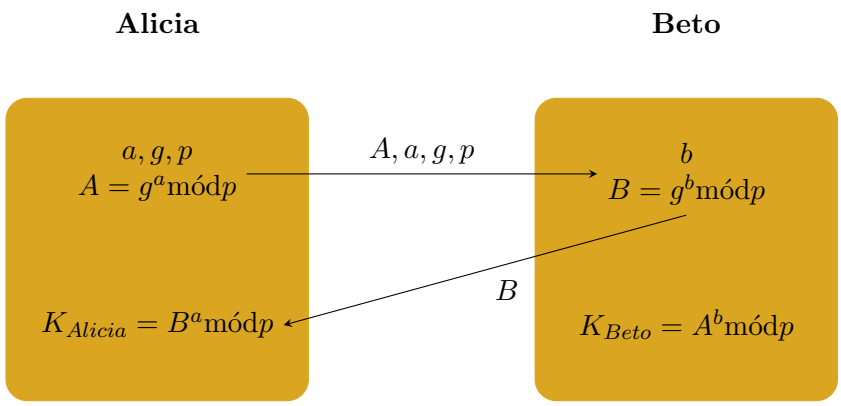


Figura 1.2: Intercambio de llaves de Diffie-Hellman.
Alicia escoge a, g, p y calcula A. Envía A, a, g, p a Beto. Beto escoge b aleatoriamente y calcula B y K_{Beto} . Beto Envía B a Alicia. Alicia calcula K_{Alicia}. La llave compartida es $K = K_{Alicia} = K_{Beto}$. Este esquema es susceptible ante ataques de intruso de en medio.

de los protocolos de la criptografía clásica. Una de estas llaves es de carácter público mientras que la otra es privada. La llave pública es utilizada para que los emisores puedan cifrar información dirigida a un receptor. La llave privada es utilizada por el receptor para descifrar la información previamente cifrada con su llave pública. La figura 1.3 muestra como funciona este tipo de protocolos de manera general.

La seguridad de estos protocolos descansa en problemas matemáticos, difíciles de resolver. Un problema se considera difícil de resolver, si no se conoce algún algoritmo capaz de solucionarlo con complejidad polinomial en tiempo [5]. Algunos ejemplos de cifrados de llave pública son:

- ***RSA*** [26]. Debe su nombre a *Ronald* ***R****ivest, Adi* ***S****hamir* y *Leonard* ***A****dleman*; RSA basa su seguridad en el problema de factorización entera. Desarrollado en 1978. Esto es, hallar una factorización en números primos de algún entero *suficientemente grande*.

- ***ElGamal***[7]. Propuesto por Taher Elgamal en 1985. Basa su seguridad en la dificultad de resolver el problema del logaritmo discreto en campos finitos.

....

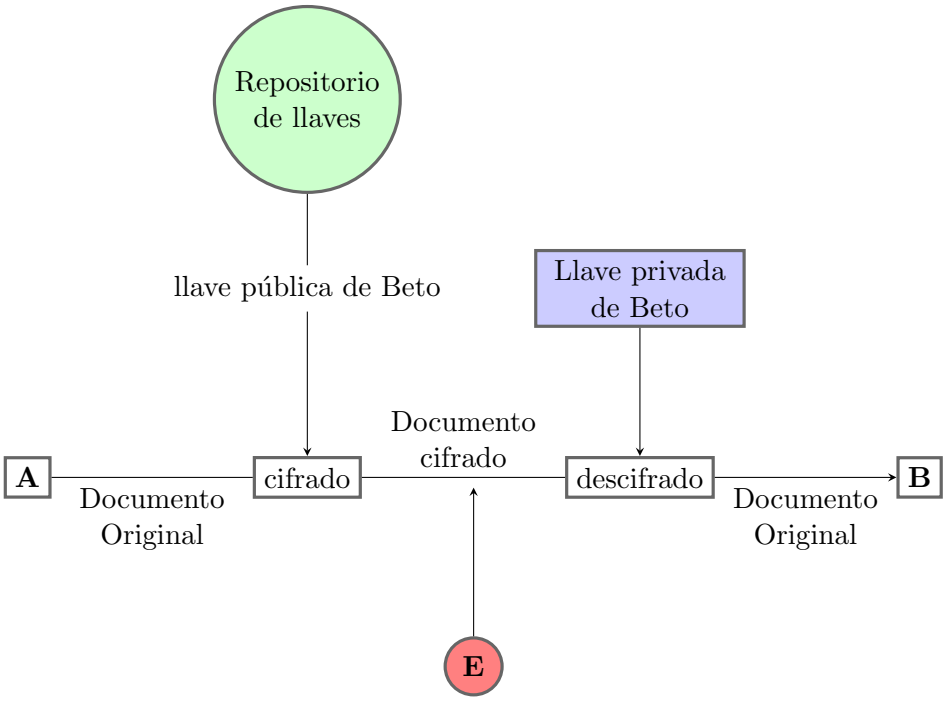


Figura 1.3: Esquema de Criptografía de llave pública

1.1.1. Curvas elípticas

En 1985 Miller[22] y Koblitz[18] propusieron, de manera independiente, el uso de curvas elípticas para el diseño de cifrados de llave pública. Las curvas elípticas poseen ciertas características geométricas y algebráicas las cuales permiten definir en ellas una instancia del problema del logaritmo discreto. Estos cifrados requieren del uso de operaciones aritméticas en los puntos de las curvas elípticas. La operación principal en la criptografía de curvas elípticas es la multiplicación escalar de un punto P por un entero k que consiste en sumar un punto consigo mismo k veces. Se conocen fórmulas explicitas para la suma y el doblado de puntos y varios métodos para calcular kP de manera eficiente, sin embargo surge la siguiente pregunta, la cual motiva este proyecto, ¿Qué tan rápido podemos realizar esta operación?.

Existen diversos métodos rápidos conocidos hasta el momento, como son:

- Métodos genéricos como
  - Técnicas de *peinado (comb en inglés)* las cuales precomputan tablas que dependen de P. Estas técnicas se aplican cuando el

punto base P es fijo y conocido *a priori*. Un ejemplo de esto es la generación de firmas ECDSA.

- Cadenas de adición. Estas son útiles cuando el escalar k es fijo. Por ejemplo en el proceso de decifrado RSA.
- Técnicas de *ventana (windowing en inglés)*. Son útiles cuando el punto base P no es comocido *a prior*. Un ejemplo de esto el *acuerdo de llaves* de Diffile-Hellman
- Técnicas de *multi-exponenciación* simultanea. Útiles para calcular expresiones $k_0P_0 + k_1P_1 + \cdots + k_tP_t$. Usadas por ejemplo en la verificación de firma ECDSA.

- Codificación del exponente. Reemplazan la representación binaria del escalar k, por una representación con menos términos diferentes de cero.

- Métodos para curvas elípticas particulares como

  - Elegir un campo base adecuado, esto es, con una aritmética eficiente. Por ejemplo seleccionar un campo $\mathbb{F}_p$ con un primo de Mersenne[6] o una extensión óptima.
  - Elegir una representación del campo base que acelere la aritmética. Por ejemplo usando un trinomio irreducible en el caso de las extensiones de campos binarios.
  - Elegir una representación adecuada del punto. Por ejemplo las coordenadas lambda[25] que reducen el costo de la suma y doblado de puntos en curvas elípticas binarias.
  - Elegir una curva con propiedades especiales. Por ejemplo las curvas de Koblitz[18] que explotan el endomorfismo de Frobenius.

- Métodos mixtos. Estos métodos hacen uso de una combinación de dos o más métodos. Por ejemplo las curvas GLS [9] que tienen la propiedad de contar con un endomorfismo eficiente y utilizan técnicas de multiexponenciación. Otro ejemplo podrían ser ser las curvas de Koblitz que además de sus propiedades, hacen uso de una codificación τNAF al escalar.

Entre otros.

1.2. Estado del arte

Como se mencionó al final de la sección anterior, parte de la comunidad científica trata de resolver el problema de calcular la multiplicación escalar de una manera eficiente. Muchos enfoques han surgido, desde algoritmos, hasta nuevas familias de curvas. A continuación se mencionan de manera breve algunos trabajos de suma importancia al tratar de responder la interrogante mencionada en la sección anterior.

- En el año 2001, Robert Gallant, Robert Lambert y Scott Vanstone en [10] propusieron una manera eficiente de realizar la arítmetica en algunas curvas elípticas dotadas con endomorfismos. El método es conocido como GLV (Por las siglas de los autores) y consiste en calcular

$$kP = k_1P + k_2\varphi(P),$$

  donde $\varphi : \mathscr{E} \to \mathscr{E}$ es un endomorfismo de la curva $\mathscr{E}$ y $P \in \mathscr{E}$. La mayor eficiencia de este radica en el ahorro de doblados al realizarse simultáneamente las dos multiplicaciones escalares. A pesar de la eficiencia del método no se conocen muchas curvas las cuales se puedan equipar con endomorfismos eficientes.

- En el año 2009, Steven Galbraith, Xibin Lin y Michael Scott en [9] proponen una familia de curvas con sus respectivos endomorfismos definidas sobre $\mathbb{F}_{p^2}$ afirmando que en sus curvas se puede aplicar el método GLV. Estas curvas son llamadas curvas *GLS*.

- En 2013, Benjamin Smith en [28] propone una manera de construir endomorfismos eficientes sobre una familia de $\mathbb{Q}$-curvas propuesta por Yuji Hasegawa en 1997[14]. Estas curvas son de *buena reducción* para números primos mayores o iguales que 5.

- En 2015 Craig Costello y Patrick Longa [6] realizan una implementaciíon de estas $\mathbb{Q}$-curvas utilizando el primo de Mersenne $p = 2^{127} - 1$. En este trabajo se realiza una descomposición en 4 sumandos.

1.3. Planteamiento del problema y objetivos

En la actualidad la criptografía desempeña una función vital en la seguridad informática. No sólo protege los datos sino que además, como se mencionó con anterioridad ofrece algunos otros servicios de seguridad como son:

- Confidencialidad.
- Integridad.
- Autenticación.
- No repudio.

Para poder proveer de estos servicios a las entidades participantes es necesario contar con las características siguientes:

- ⋆ Velocidad
- ⋆ Seguridad

Las curvas elípticas han ganado popularidad a lo largo de los años y se han incluido en la mayoria de las bibliotecas dedicadas a seguridad como `PGP`[35], `OpenSSL`[33] entre otros. En la vida diaria poder realizar comunicaciones de manera rápida juega un papel relevante. Existen curvas rápidas y seguras como `Curve25519`[3], K-163 entre otras[23].

A pesar de tener una variedad de curvas, la respuesta a la interrogante: ¿Qué tan rápido podemos realizar la multiplicación escalar? no es clara. Es decir el problema de encontrar métodos o curvas más rápidas sigue abierto. Con base en el trabajo de Ben Smith, se pretende extender su propuesta encontrando curvas en característica 2. La problemática a resolver es la siguiente: ¿Es posible encontrar endomorfismos eficientes en curvas elípticas binarias para agilizar el proceso de la multiplicación escalar?.

El objetivo principal que se pretende alcanzar en esta tesis, es el de proporcionar endomorfismos eficientes en curvas elípticas binarias para acelerar la multiplicación escalar. De manera particular se hace enfoque en las curvas elípticas binarias debido a la carencia de resultados en éstas, con respecto a los endomorfismos de $\mathbb{Q}$-curvas, y a que las operaciones aritméticas(Suma y doblado) pueden realizarse de manera eficiente[25]. Como objetivos particulares se tienen:

- Adaptar el método de creación de endomorfismos en $\mathbb{Q}$-curvas primas propuesto por Ben Smith en [28] a curvas GLS binarias. La finalidad de esta meta es la de acelerar la multiplicación escalar en estas curvas.
- Proponer un nuevo endomorfismo para curvas de Koblitz.

[3] Utilizada por Whatsapp[32].

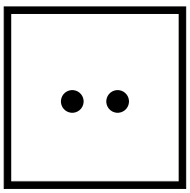
1.4. Organización del documento

Esta tesis ha sido organizada en 5 capítulos. En el capítulo 2 se exponen y analizan algunos conceptos y resultados matemáticos indispensables para la comprensión, tanto de la construcción de los endomorfismos , como de la descomposición escalar. El contenido de este capítulo abreva del material disponible en [16], [11], [8] y [20]

Los objetos matemáticos de interés en esta tesis, son las curvas elípticas. En el capítulo 3 se hace referencia a la multiplicación escalar entre otros aspectos necesarios para poder comprender el desarrollo de la tesis. El contenido de este capitulo abreva del material disponible en [16], [31], [13] y [27].

En el capítulo 4 se expone de manera detallada el desarrollo de esta tesis. Se presentan las curvas elípticas en las cuales se trabajó, así como un analisis de seguridad relativos a la descomposición escalar. Por último en el capítulo 5 se presentan los resultados obtenidos y las conclusiones, así como el trabajo a futuro.

Están disponibles 3 apéndices. En el apéndice A se mencionan conceptos básicos de las matemáticas. En el apéndice B están algunos códigos en MAGMA utilizados en los ejemplos y contruscciones del capítulo 4. El apéndice C expone un ejemplo de una 3-descomposición con base en algunos resultados obtenidos en el capítulo 4.



Capítulo 2

Contexto matemático

> A mathematician, like a painter or a poet, is a maker of patterns. If his patterns are more permanent than theirs, it is because they are made with ideas.[1]
>
> *G. H. Hardy*
> *-A Mathematician's Apology*

En el campo de las matemáticas existen estructuras u objetos, los cuales son utilizados por la criptografía para poder construir protocolos y ofrecer ciertos servicios de seguridad. Ejemplos de estas estructuras son los grupos, anillos y campos. Con base en lo anterior, surge la necesidad de conocer estas estructuras para su posterior uso en la criptografía. Este capítulo trata de suplir esta necesidad presentando conceptos matemáticos básicos aplicables a la criptografía. Los conceptos se presentarán distribuidos en 4 secciones.

[1] Un matemático, así como un pintor o un poeta, es un creador de patrones. Si sus patrones son más permanentes que el de ellos, es porque están hechos de ideas.

La primera sección introduce el concepto de *conjunto*, con particular enfoque en funciones y aritmética modular. Posteriormente en la sección 2 se presenta una estructura matemática conocida como *grupo*. Se estudiarán las propiedades de los grupos y funciones entre grupos(llamadas morfismos) y. En la sección 3 se estudiará otra estructura matemática denominada *anillo*. Como parte de esta sección se presentán los *campos* y *anillos de polinomios*. Por último, en la sección 4 se hará un breve repaso de algebra lineal y se presentan las *retículas* de manera particular, con enfoque a la criptografía.

2.1. Grupos

El objetivo de esta sección es el de introducir el concepto de Grupo así como algunas propiedades de estos, funciones entre grupos y sus propiedades.

Definición 2.1.1 (Operación binaria)**.** Una *operación binaria* sobre un conjunto G es una función $* : G \times G \to G$. Para $a, b \in G$ se acostumbra escribir $a * b$ en lugar de $*((a, b))$, pero aún es más común escribir solamente ab. Utilizaremos esta última notación para casí todo el escrito y la primera cuando sea necesario especificar la operación binaria.

- Una operación binaria se dice que es *asociativa* si para cualesquiera $a, b, c \in G$ se cumple que $(a * b) * c = a * (b * c)$.
- Se dice que es *conmutativa* si para cualesquiera $a, b \in G$ se cumple que $a * b = b * a$.

Ejemplo 2.1.2. Considere el conjunto $\mathbb{R}$. La resta usual es una operación binaria, no asociativa y no conmutativa. Por ejemplo

- Si se toma $3-((-3)-3) \neq (3-(-3))-3$ se puede ver la no asociatividad y,
- tomando $3 - (-3) \neq (-3) - 3$ se prueba la no conmutatividad.

Ejemplo 2.1.3. Considere el conjunto $\mathbb{Z}$. La suma usual es una operación binaria asociativa y conmutativa.

Definición 2.1.4. Sea G un conjunto no vacio y $*$ una operación binaria sobre G; entonces el par $(G, *)$ recibe el nombre de:

- *Semigrupo*, si $*$ es asociativa.
- *Monoide*, si $*$ es asociativa y existe un *elemento neutro*, es decir, existe $e_G \in G$ tal que $e_G g = g e_G = g$, para toda $g \in G$.

- *Grupo*, si $(G, *)$ es un monoide y se cumple la propiedad del inverso para la operación*, es decir, para cada $g \in G$ existe $g^{-1} \in G$ tal que $gg^{-1} = g^{-1}g = e_G$.
- *Grupo conmutativo* o *Abeliano*[2] si $(G, *)$ es un grupo y la operación $*$ es conmutativa. En caso contrario se dice que el grupo no es conmutativo o que es *no abeliano*.

Observación 2.1.5. Cuando el grupo esté bien definido escribiremos únicamente e en lugar de e_G. En caso de ser necesario especificar el grupo usaremos e_G. El elemento neutro e tambien es llamado *identidad*.

Observación 2.1.6.

- El elemento neutro de un monoide es único.
- En un grupo se cumple la *ley de cancelación por la izquierda*. Si $g, x, y \in G$ y $gx = gy$ entonces

$$\begin{aligned} g^{-1}gx &= g^{-1}gy \\ ex &= ey \\ x &= y \end{aligned}$$

  De manera análoga se cumple la *ley de cancelación por la derecha*. Si $xg = yg$ entonces

$$\begin{aligned} xg^{-1}g &= yg^{-1}g \\ xe &= ye \\ x &= y \end{aligned}$$

- Por el inciso anterior, el inverso g^{-1} del elemento g es único. Si suponemos que no, entonces tenemos que $gg^{-1} = e = gh$ para algún $h \in G$; por el inciso anterior tenemos que en este caso $g^{-1} = h$. □
- Se cumple que $(g^{-1})^{-1} = g$ y que para toda $a, b \in G$ se tiene $(ab)^{-1} = b^{-1}a^{-1}$.

Definición 2.1.7. Sea $*$ una operación binaria en un conjunto G y $H \subset G$. H es *cerrado* bajo $*$ si para todo $a, b \in H$, se tiene que $a * b \in H$. En este caso la operación binaria de H esta dada por la restricción de $*$ a H. Esta nueva operación binaria es conocida como la *operación inducida* de $*$ en H.

[2] Llamado así en honor al matemático Niels Henrik Abel.

Definición 2.1.8 (Orden)**.** El orden de un grupo $(G, *)$, es la cardinalidad del conjunto G. Decimos que el grupo $(G, *)$ es finito si $\mid G \mid = n \in \mathbb{N}$

Una manera común de representar un grupo finito es mediante su *tabla de grupo*. Esta tabla consiste en ubicar ordenadamente los resultados de la operación binaria correspondiente como se aprecia en la tabla 2.1.

$*$	e	g_1	$\dots$	g_j	$\dots$	g_n
e	$(e * e)$	$(e * g_1)$	$\dots$	g_j	$\dots$	g_n
g_1	$g_1 * e$	$g_1 * g_1$	$\dots$	$g_1 * g_j$	$\dots$	$g_1 * g_n$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
g_i	g_i	$g_i * g_1$	$\dots$	$g_i * g_j$	$\dots$	$g_i * g_n$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
g_n	g_n	$g_n * g_1$	$\dots$	$g_n * g_j$	$\dots$	$g_n * g_n$

Tabla 2.1: Tabla de grupo de un grupo $(G, *)$ con $G = \{e, g_1, g_2, \dots, g_n\}$

Ejemplo 2.1.9. Consideremos como ejemplo el subconjunto $G = \{1, -1, i, -i\}$ del conjunto $\mathbb{C}$. Tomemos como operación binaria $*$ a la multiplicación usual de los números complejos. La tabla de grupo de $(G, *)$ es la que se observa en la figura 2.2

$*$	1	-1	i	$-i$
1	1	-1	i	$-i$
-1	-1	1	$-i$	i
i	i	$-i$	-1	1
$-i$	$-i$	i	1	-1

Tabla 2.2: Tabla de grupo del ejemplo 2.1.9

El hecho de que el producto es cerrado en este conjunto confirma que es una operación binaria en $G = \{1, -1, i, -i\}$. La asociatividad se *hereda* (esta es la forma habitual de describir que un subobjeto tiene cierta propiedad porque la tiene el objeto), al igual que el neutro y la conmutatividad, pues estas propiedades las tiene el producto usual de los complejos.

Notación 2.1.10. Se ha definido un grupo como un par $(G, *)$, sin embargo cuando sea posible abreviaremos esto escribiendo únicamente G como grupo.

Definición 2.1.11 (Subgrupo)**.** Sean G un grupo y H un subconjunto no vacío de G; Si H es cerrado bajo la operación de G, y forma un grupo con

la operación inducida de G, entonces H es un *subgrupo* de G. Denotamos por $H \leq G$ o $G \geq H$ el hecho de que H es un subgrupo de G y por $H < G$ o $G > H$ el que H sea un subgrupo de G pero $H \neq G$.

Ejemplo 2.1.12. El grupo del ejemplo 2.1.9 es un subgrupo del grupo $(\mathbb{C}, +)$.

Definición 2.1.13. Sea G un grupo. El *subgrupo impropio* de G, es G mismo, todos los demás subgrupos de G son llamados *sugrupos propios*. El subgrupo $\{e\}$ es llamado el *subgrupo trivial*, todos los demás subgrupos son llamados *subgrupos no triviales*.

El siguiente teorema nos permite decidir cuando un subconjunto de un grupo es un subgrupo.

Teorema 2.1.14. *Un subconjunto H de un grupo G es un subgrupo de G si y sólo si*

1. *H es cerrado bajo la operación de G.*
2. *El elemento identidad e de G está en H.*
3. *Para toda $a \in H$ se cumple que $a^{-1} \in H$.*

Proposición 2.1.15. *Sea G un grupo y H un subconjunto no vacío de G, entonces $H < G$ si y sólo si para cualesquiera $a, b \in H$ se cumple que $ab^{-1} \in H$.*

2.1.1. Grupos cíclicos

Notación 2.1.16. Sea $(G, *)$ un grupo y $a \in G$ denotaremos por a^k a la acción de aplicar la operación $*$ en a, $(k-1)$ veces en si mismo.

$$a^k = \underbrace{a * a * a * a * \cdots * a}_{(k-1) \text{ veces}}.$$

Teorema 2.1.17. *Sea G un grupo y $a \in G$, entonces*

$$H = \{a^n \mid n \in \mathbb{Z}\} = \{e, a, a^2, a^3, \dots\}$$

es un subgrupo de G y es el subgrupo más pequeño[3] *que contiene al elemento a.*

[3]En el sentido de que, si existe otro conjunto conteniendo a a, digamos $a \in K < G$, entonces $H \subset K$ y $H < K$.

Con base en el teorema 2.1.17 se procederá a definir cierta clase especial de grupo.

Definición 2.1.18 (Subgrupo cíclico)**.** Sea G un grupo y $a \in G$. El subgrupo H de G caracterizado en el teorema 2.1.17 es llamado *subgrupo cíclico de G generado por a*. Este subgrupo es denotado por $\langle a \rangle$.

Ejemplo 2.1.19. Denotamos por $n\mathbb{Z}$ el grupo cíclico generado por n, bajo la adición. Ejemplo de esto es $2\mathbb{Z} = \{\ldots, -4, -2, 0, 2, 4, \ldots\}$.

Definición 2.1.20 (Grupo cíclico)**.** Un elemento a de un grupo G, *genera* a G y es un *generador* para G si $\langle a \rangle = G$. Un grupo G es *cíclico* si existe un elemento $a \in G$ que genera a G.

Ejemplo 2.1.21. El conjunto $\mathbb{Z}_4 = \{0, 1, 2, 3\}$ junto con la operación de suma módulo 4 forma un grupo finito. La tabla de grupo se puede observar en la tabla 2.3. Este grupo es un grupo cíclico puesto que $\mathbb{Z}_4 = \langle 1 \rangle = \langle 3 \rangle$. El subgrupo cíclico $\langle 2 \rangle$ esta formado por el conjunto $\{0, 2\}$ y la operación inducida de $\mathbb{Z}_4$.

$*$	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

Tabla 2.3: Tabla de grupo del ejemplo 2.1.21

Observación 2.1.22. En general si $n \in \mathbb{Z}$, entonces el conjunto $\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}$ junto con la suma módulo n forma un grupo.

Teorema 2.1.23. *Todo grupo cíclico es abeliano.*

Teorema 2.1.24. *Todo subgrupo de un grupo cíclico es cíclico.*

Definición 2.1.25 (Producto directo de grupos)**.** Sean $G_0, G_1, G_2, \ldots, G_n$ grupos. Abreviamos el producto cartesiano de $G_0 \times G_1 \times G_2 \times \cdots \times G_n$ como $\prod_{i=0}^{n} G_i$. Definimos la operación de 2 elementos $(a_0, a_1, \ldots, a_n)(b_0, b_1, \ldots, b_n)$ como el elemento $(a_0 b_0, a_1 b_1, \ldots, a_n b_n) \in \prod_{i=0}^{n} G_i$. Entonces $\prod_{i=0}^{n} G_i$ es el *producto directo* de los grupos G_i bajo esta operación.

Ejemplo 2.1.26. Consideremos el grupo $\mathbb{Z}_2 \times \mathbb{Z}_3$. Este grupo tiene $2 \cdot 3 = 6$ elementos. Los elementos son $(0,0), (0,1), (0,2), (1,0), (1,1), (1,2)$. Afirmamos que $\mathbb{Z}_2 \times \mathbb{Z}_3$ es cíclico. Un generador es $(1,1)$.

$$\begin{aligned}
(1,1) &= (1,1) \\
2(1,1) &= (1,1) + (1,1) = (0,2) \\
3(1,1) &= (1,1) + (1,1) + (1,1) = (1,0) \\
4(1,1) &= (1,1) + (1,1) + (1,1) + (1,1) = (0,1) \\
5(1,1) &= (1,1) + (1,1) + (1,1) + (1,1) + (1,1) = (1,2) \\
6(1,1) &= (1,1) + (1,1) + (1,1) + (1,1) + (1,1) + (1,1) = (0,0)
\end{aligned}$$

.

Observación 2.1.27. En el ejemplo anterior se usa la notación $n(a,b)$. Cuando el grupo tiene operación aditiva en lugar de escribir $(a,b)^n$ para indicar que la operación se aplica $n-1$ veces en si mismo, se escribe $n(a,b)$.

Teorema 2.1.28. *El grupo $\mathbb{Z}_m \times \mathbb{Z}_n$ es cíclico e isomorfo a $\mathbb{Z}_{mn}$ si y sólo si m y n son primos relativos, esto es $mcd(m,n) = 1$.*

Corolario 2.1.29. *El grupo $\prod_{i=0}^{n} \mathbb{Z}_{m_i}$ es cíclico e isomorfo a $\mathbb{Z}_{m_0 m_1 m_2 \ldots m_n}$ si y sólo si los enteros m_i son primos relativos entre sí. Esto es que para cada 2 enteros m_i y m_j se tiene que $mcd(m_i, m_j) = 1$.*

2.1.2. Clases laterales

Sea H un subgrupo de un grupo G de orden finito o infinito definiremos dos relaciones de equivalencia en G, $\sim_I$ y $\sim_D$.

Teorema 2.1.30. *Sea H un subgrupo de un grupo G. Definimos la relación $\sim_I$ en G como*

$$a \sim_I b \Leftrightarrow a^{-1}b \in H$$

y definimos $\sim_D$ como

$$a \sim_D b \Leftrightarrow ab^{-1} \in H \ .$$

Entonces $\sim_I$ y $\sim_D$ son relaciones de equivalencias.

Definición 2.1.31. Sea H un subgrupo de un grupo G. El subconjunto $aH = \{ah \mid h \in H\}$ de G es llamado, *clase lateral izquierda* de H conteniendo al elemento a, mientras que el subconjunto $Ha = \{ha \mid h \in H\}$ de G es llamado, la *clase lateral derecha* de H, que contiene al elemento a.

Ejemplo 2.1.32. sea $3\mathbb{Z} = \{\dots, -9, -6, -3, 0, 3, 6, 9, \dots\}$ un subgrupo de $\mathbb{Z}$. En este caso tenemos un grupo bajo la adición. Consideremos $a = 1$, entonces

$$1 + 3\mathbb{Z} = \{\dots, -8, -5, -2, 1, 4, 7, 10, \dots\}$$

es la clase lateral izquierda de $3\mathbb{Z}$ que contiene al 1.

Ejemplo 2.1.33. Sea $2\mathbb{Z} = \{\dots, -4, -2, 0, 2, 4, \dots\}$ un subgrupo aditivo de $\mathbb{Z}$ entonces

$$2\mathbb{Z} + 3 = \{\dots, -1, 1, 3, 5, 7, \dots\}$$

es la clase lateral derecha de $2\mathbb{Z}$ que contiene al 3.

Observación 2.1.34. En los dos ejemplos anteriores se puede verificar que $1 + 3\mathbb{Z} = 3\mathbb{Z} + 1$ y que $2\mathbb{Z} + 3 = 3 + 2\mathbb{Z}$. Esto sucede porque $\mathbb{Z}$ es un grupo abeliano.

Observación 2.1.35. Para un elemento $g \in G$ y $H < G$, podemos construir una función inyectiva $\varphi : H \to gH$ donde $\varphi(h) = gh$. Esta función es inyectiva puesto que si tomamos $\varphi(h_0) = \varphi(h_1)$ tendríamos que $gh_0 = gh_1$, y por las leyes de cancelación llegamos a que $h_0 = h_1$. Además por definición de gH φ es suprayectiva. Lo anterior nos dice que *toda clase lateral (derecha o izquierda) de $H < G$ tiene la misma cardinalidad que H.*

Teorema 2.1.36 (Lagrange)**.** *Sea H un subgrupo de un grupo finito G. Entonces el orden de H es un divisor del orden de G.*

Corolario 2.1.37. *Todo grupo de orden primo es cíclico.*

Corolario 2.1.38. *El orden de un elemento de un grupo finito G es un divisor del orden de G.*

Definición 2.1.39 (Índice)**.** Sea $H < G$. El número de clases laterales izquierdas de H en G es el *índice* $(G : H)$ de H en G.

Observación 2.1.40. El índice $(G : H)$ puede ser finito o infinito. Si G es finito entonces $(G : H)$ es finito y $(G : H) =\mid G \mid / \mid H \mid$.

Observación 2.1.41. Aunque el índice se definió en terminos de clases laterales izquierdas, tambien está bien definido usando clases laterales derechas. Esto nos dice que el número de clases laterales derechas e izquierdas son iguales.

Teorema 2.1.42. *Sean H, K subgrupos de un grupo G tales que $K \leq H \leq G$. Si $(H : K)$ y $G : H$ son finitos, entonces $(G : K) = (G : H)(H : K)$.*

Por el teorema de *Lagrange*, si se tiene un grupo finito G, entonces el orden de cualquier subgrupo de G divide al orden de G. El recíproco no siempre es cierto, es decir que si m es un divisor del orden de G no necesariamente existe un subgrupo de orden m. Sin embargo el siguiente teorema nos dice cuándo es que existe este subgrupo.

Teorema 2.1.43. *Si m es un divisor del orden de un grupo abeliano finito G, entonces G tiene un subgrupo de orden m.*

2.1.3. Homomorfismos

En esta sección presentamos cierto tipo de funciones, las cuales están definidas entre grupos. Se presentan ciertas propiedades que cumplen y algunas aplicaciones que brindan a la teoría de grupos.

Definición 2.1.44 (Homomorfismo)**.** Sean $(G, \cdot)$ y $(G', \otimes)$ grupos. Una función $\phi : G \to G'$ es un *homomorfismo* de grupos si satisface que

$$\phi(a \cdot b) = \phi(a) \otimes \phi(b) \tag{2.1.1}$$

para todo elemento $a, b \in G$.

Observación 2.1.45. La definición 2.1.1 nos dice que ϕ preserva la operación de ambos grupos. Es decir que si se toman 2 elementos en el grupo G y se aplica ϕ a ab, entonces la imagen de $\phi(ab)$ corresponde a la respectiva operación del grupo G' de las imágenes de $\phi(a)$ y $\phi(b)$.

Observación 2.1.46. Siempre existe un homomorfismo entre cualesquiera dos grupos. Este homomorfismo es llamado el *morfismo trivial*. Sean G y G' dos grupos y e' el neutro G', entonces el homomorfismo trivial $\phi : G \to G'$ está definido por $\phi(a) = e'$ para toda $a \in G$.

Ejemplo 2.1.47. Sea $r \in \mathbb{Z}$ y sea $\phi_r : \mathbb{Z} \to \mathbb{Z}$ definida por $\phi(n) = rn$ para toda $n \in \mathbb{Z}$. ϕ_r es un homomorfismo dado que

$$\begin{gathered} \phi(n+m) = r(n+m) \\ \phi(n+m) = rn + rm \\ \phi(n+m) = \phi(n) + \phi(m). \end{gathered}$$

En este caso ϕ_0 es el homomorfismo trivial.

Observación 2.1.48. En el ejemplo anterior, el homomorfismo ϕ_1 es llamado el *morfismo identidad*. El homomorfismo identidad $\phi : G \to G$ consiste en $\phi(a) = a$ para toda $a \in G$.

Ejemplo 2.1.49 (Reducción módulo n)**.** Sea $\gamma : \mathbb{Z} \to \mathbb{Z}_n$ definida por $\gamma(m) = r$, donde r es el residuo de aplicar el algoritmo de la división cuando m es dividido por n. γ es un homomorfismo.

Ejemplo 2.1.50. Sea $G = \prod_{i=0}^{n} G_i$ un producto directo de grupos. La función *proyección* $\pi_i : G \to G_i$ definida por $\pi(g_0, g_1, \ldots, g_n) = g_i$ es un homomorfismo para cada $i = 0, 1, \ldots, n$.

Definición 2.1.51. Sea $\phi : G \to G'$ un homomorfismo de grupos, entonces decimos que ϕ es un

- *Monomorfismo* si ϕ es inyectiva.
- *Epimorfismo* si ϕ es suprayectiva.
- *Isomorfismo* si es biyectiva.
- *Automorfismo* si ϕ es un isomorfismo y $G = G'$.

Teorema 2.1.52. *Sean G y G' dos grupos tales que $\#G = \#G'$ entonces, existe un isomorfismo $\phi : G \to G'$.*

Propiedades de los homomorfismos

Notación 2.1.53. Si H es un subgrupo de un grupo G y ϕ es un homomorfismo de G a otro grupo G', entonces denotamos por $\phi(H)$ al conjunto $\{\phi(h) \mid h \in H\}$.

Teorema 2.1.54. *Sean G y G' dos grupos y $\phi : G \to G'$ un homomorfismo de grupos.*

- *Si e es la identidad de G entonces $\phi(e)$ es la identidad de G'.*
- *Si $a \in G$, entonces $\phi(a^{-1}) = \phi(a)^{-1}$.*
- *Si $H < G$, entonces $\phi(H)$ es un subgrupo de G'.*
- *Si K' es un subgrupo de G', entonces*

•

Definición 2.1.55 (Núcleo)**.** Sea $\phi : G \to G'$ un homomorfismo de grupos. El subgrupo $\phi^{-1}(\{e'\}) = \{x \in G \mid \phi(x) = e'\}$ es llamado el *núcleo (Kernel*[4]*)* de ϕ y es denotado por $Ker(\phi)$.

Ejemplo 2.1.56. Sea $\phi : G \to G'$ el homomorfismo trivial, entonces el núcleo de ϕ es G.

Teorema 2.1.57. *Un homomorfismo de grupos es un monomorfismo si y sólo si* $Ker(\phi) = \{e\}$.

Teorema 2.1.58. *Sea* $\phi : G \to G'$ *un homomorfismo de grupos, y sea* $H = Ker(\phi)$. *Sea* $a \in G$. *Entonces el conjunto*

$$\phi^{-1}(\{\phi(a)\}) = \{x \in G \mid \phi(x) = \phi(a)\}$$

es la clase lateral izquierda aH *de* H, *de igual forma, es la clase lateral derecha* Ha *de* H. *En este caso las clases laterales derecha e izquierda de* H *coinciden.*

Definición 2.1.59 (subgrupo normal)**.** Un subgrupo H de un grupo G es *normal* si sus clases laterales derechas e izquierdas coinciden, esto es

$$gH = Hg$$

para toda $g \in G$. Denotamos que H es normal en G por $H \triangleleft G$ o $G \triangleright H$.

Observación 2.1.60. Todos los subgrupos de un grupo abeliano son normales.

Corolario 2.1.61. *Si* $\phi : G \to G'$ *es un homomorfismo de grupos, entonces* $Ker(\phi)$ *es un subgrupo normal de* G.

Teorema 2.1.62. *La composición de homomorfismos de grupos es un homomorfismo. Esto es, sean* G, G' *y* $\hat{G}$ *grupos. Sean* $\phi : G \to G'$ *y* $\gamma : G' \to \hat{G}$ *homomorfismos de grupos, entonces la composición* $\gamma(\phi) : G \to \hat{G}$ *es un homomorfismo.*

Grupo cociente

Teorema 2.1.63. *Sea* $\phi : G \to G'$ *un homomorfismo de grupos con kernel* H. *Entonces el conjunto de las clases laterales de* H *forma un grupo cociente,* G/H, *donde la operación binaria esta dada por* $(aH)(bH) = (ab)H$. *La función* $\mu : G/H \to \phi(G)$ *definida por* $\mu(aH) = \phi(a)$ *es un isomorfismo.*

[4]De la raíz germánica Kern que traducido al español es núcleo o hueso.

.

.

Observación 2.1.64. La notación G/H se lee G sobre H, G módulo H o G mód H. Sin embargo, G dividido por H es incorrecto.

Ejemplo 2.1.65. Consideremos el mapa $\gamma : \mathbb{Z} \to \mathbb{Z}_n$ definido en el ejemplo 2.1.49. El núcleo de γ es $n\mathbb{Z}$. El teorema 2.1.63 nos dice que $\mathbb{Z}/n\mathbb{Z}$ es isomorfo a $\mathbb{Z}_n$. Las clases laterales de $n\mathbb{Z}$ son llamadas *clases de residuos módulo* n. Por ejemplo tomemos $n = 5$, las clases laterales de $5\mathbb{Z}$ son

$$\begin{aligned} 5\mathbb{Z} = &\quad \{\dots, -10, -5, 0, 5, 10, \dots\}, \\ 1 + 5\mathbb{Z} = &\quad \{\dots, -9, -4, 1, 6, 11, \dots\}, \\ 2 + 5\mathbb{Z} = &\quad \{\dots, -8, -3, 2, 7, 12, \dots\}, \\ 3 + 5\mathbb{Z} = &\quad \{\dots, -7, -2, 3, 8, 13, \dots\}, \\ 4 + 5\mathbb{Z} = &\quad \{\dots, -6, -1, 4, 9, 14, \dots\}. \end{aligned}$$

El isomorfismo $\mu : \mathbb{Z}/5\mathbb{Z} \to \mathbb{Z}_5$ del teorema 2.1.63 asigna cada clase lateral de $5\mathbb{Z}$ al menor entero no negativo contenido en esa clase. Esto es, $\mu(5\mathbb{Z}) = 0$, $\mu(1 + 5\mathbb{Z}) = 1$, etc.

Ejemplo 2.1.66. Consideremos el grupo cociente $\mathbb{Z}/5\mathbb{Z}$ definido en el ejemplo 2.1.65. Podemos sumar dos clases laterales, por ejemplo, $(2+5\mathbb{Z})+(4+5\mathbb{Z})$ escogiendo $2 + 4 = 6$. Podemos ver que $6 \in (1 + 5\mathbb{Z})$. De igual forma podemos considerar $7 \in (2 + 5\mathbb{Z})$ y $-6 \in (4 + 5\mathbb{Z})$; la suma $(7 + (-6)) = 1$ está tambien en la clase lateral $(1 + 5\mathbb{Z})$.

Observación 2.1.67. Hasta ahora se ha definido una operación de clases laterales del grupo cociente a partir del núcleo de una homomorfismo. A continuación se presenta una condición que nos dice para que tipo de subgrupos es posible definir esta operación sin considerar primero un homomorfismo.

Teorema 2.1.68. *Sea H un subgrupo de un grupo G, entonces la multiplicación de clases laterales izquierdas dada por la ecuación*

$$(aH)(bH) = (ab)H, \tag{2.1.2}$$

está bien definida si y sólo si, H es un subgrupo normal de G.

Corolario 2.1.69. *Sea H un subgrupo normal de un grupo G, entonces las clases laterales de H forman un grupo G/H bajo la operación binaria $(aH)(bH) = (ab)H$.*

Definición 2.1.70 (Grupo cociente)**.** El grupo G/H en el corolario 2.1.69 es el *grupo cociente* de G sobre H.

Ejemplo 2.1.71. Se sabe que $\mathbb{Z}$ es un grupo abeliano con la suma. El subgrupo $n\mathbb{Z}$ es un subgrupo normal de $\mathbb{Z}$. El corolario 2.1.69 nos permite construir el grupo cociente $\mathbb{Z}/n\mathbb{Z}$ sin necesidad de utilizar un homomorfismo. Como se observó en el ejemplo 2.1.65 $\mathbb{Z}/n\mathbb{Z}$ es isomorfo a $\mathbb{Z}_5$.

•

••

2.1.4. Teoremas de homomorfismos

Teorema 2.1.72. *Sea H un subgrupo normal del grupo G. Entonces la función $\gamma : G \to G/H$ dada por $\gamma(x) = xH$ es un homomorfismo con núcleo H.*

Notación 2.1.73. Denotamos por $G \cong G'$ el hecho de que el grupo G es isomorfo al grupo G'.

Teorema 2.1.74 (El teorema fundamental de los homomorfismos)**.** *Sea $\phi : G \to G'$ un homomorfismo de grupos con núcleo H, entonces $\phi(G)$ es un grupo, y $\mu : G/H \to \phi(G)$ dado por $\mu(gH) = \phi(g)$ es un isomorfismo. Si $\gamma : G \to G/H$ es el homomorfismo dado por $\gamma(g) = gH$, entonces $\phi(g) = \mu(\gamma(g))$ para cada $g \in G$. En la figura 2.1 se puede ver el diagrama de esos grupos y los homomorfismos relacionados.*

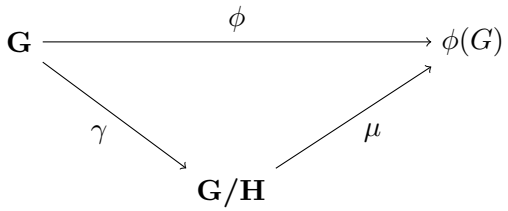


Figura 2.1: Teorema fundamental de los homomorfismos

Observación 2.1.75. El isomorfismo μ del teorema 2.1.74 es llamado *natural* o *canónico*. El mismo adjetivo es utilizado para referirse al homomorfismo γ del mismo teorema.

Teorema 2.1.76 (Primer teorema de isomorfismos)**.** *Sea $\phi : G \to G'$ un morfimo de grupos con núcleo K, y sea $\gamma_K : G \to G/K$ el homomorfismo canónico. Existe un único isomorfismo $\mu : G/K \to \phi(G)$ tal que, $\phi(x) = \mu(\gamma_K(x))$ para cada $x \in G$.*

2.2. Anillos

Definición 2.2.1. Un *Anillo* es una tupla $(R, +, \cdot)$ consistente de un conjunto R y dos operaciones binarias, $+$ y $\cdot$, llamadas *suma* y *multiplicación*, definidas en R con las siguientes propiedades:

- $(R, +)$ es un grupo abeliano.

.
...

- $(R, \cdot)$ es un semigrupo.
- Para todo $a, b, c \in R$, se cumplen *la ley distributiva izquierda*, $a \cdot (b + c) = (a \cdot b) + (c \cdot b)$, y la *ley distributiva derecha* $(a+b) \cdot c = (a \cdot c) + (b \cdot c)$.

Ejemplo 2.2.2. $\mathbb{Z}, \mathbb{R}$ y $\mathbb{Q}$ con sus operaciones de suma y multiplicación usual, son ejemplos de anillos conocidos.

Notación 2.2.3. Similar a la notación en grupos, la multiplicación entre dos elementos a, b será denotada simplemente por ab. La ley distributiva izquierda por ejemplo, la escribiremos como $a(b + c) = ab + ac$.

Ejemplo 2.2.4. El conjunto $n\mathbb{Z}$ es un grupo con la operación de suma. Podemos definir la multiplicación en este conjunto como la multiplicación usual de enteros. Esta multiplicación es una operación binaria. Se sabe que $(nr)(ns) = n(nrs)$, esto muestra que la multiplicación es cerrada en $n\mathbb{Z}$.

Ejemplo 2.2.5. Sabemos que $\mathbb{Z}_n$ es un grupo cíclico con la suma módulo n. Si definimos la operación binaria $\cdot$, como la multiplicación módulo n, entonces $(n\mathbb{Z}, +, \cdot)$ es un anillo.

Ejemplo 2.2.6. Sean $R_1, R_2, \ldots, R_n$ anillos. Podemos construir el conjunto $R_1 \times R_2 \times \cdots \times R_n$ de todas las n-tuplas ordenadas $(r_1, r_2, \ldots, r_n)$ donde $r_i \in R_i$. Podemos definir la suma y la multiplicación componente a componente(igual que para grupos). Entonces el anillo $R_1 \times R_2 \times \cdots \times R_n$ con las operaciones mencionadas es el *producto directo* de anillos.

Teorema 2.2.7. *Si R es un anillo con neutro aditivo 0, entonces para cualesquiera $a, b \in R$ se cumple que*

- $0a = a0 = 0$,
- $a(-b) = (-a)b = -(ab)$,
- $(-a)(-b) = ab$.

Definición 2.2.8 (Homomorfismo)**.** Sean $(R, +, \cdot)$ y $(R', \oplus, \otimes)$, una función $\phi : R \to R'$ es un homomorfismo si cumple las siguientes dos condiciones para cualesquiera $a, b \in R$.

- $\phi(a + b) = \phi(a) \oplus \phi(b)$.
- $\phi(a \cdot b) = \phi(a) \otimes \phi(b)$.

Observación 2.2.9. En particular un homomorfismo de anillos, con la notación anterior, satisface $\phi(c \cdot (a + b)) = \phi(c) \otimes \phi(a) \oplus \phi(c) \otimes \phi(b)$.

•

••••

Observación 2.2.10. Dado que un anillo es un grupo abeliano con la operación de suma, entonces un homomorfismo de anillos, en particular, es un homomorfismo de grupos. Esto significa que todos los resultados de homomorfismos entre grupos se mantienen validos en anillos.

Definición 2.2.11. Un homomorfismo entre anillos es un

- *Monomorfismo* si es inyectivo,
- *Epimorfismo* si es suprayectivo,
- *Isomorfismo* si es biyectivo.

Definición 2.2.12. Sea $(R, +, \cdot)$ un anillo.

- $(R, +, \cdot)$ es un *anillo conmutativo* si la operación $\cdot$ es conmutativa.
- $(R, +, \cdot)$ es un *anillo con unitario* si $(R, \cdot)$ es un monoide.

Notación 2.2.13. Denotamos por 1 al elemento neutro de la multiplicación. Este elemento es llamado *uno* o *identidad*.

Observación 2.2.14. En un anillo con unitario se satisfacen las siguientes propiedades para el elemento unidad.

- $\underbrace{(1+1+\cdots+1)}_{n \text{ sumandos}}\underbrace{(1+1+\cdots+1)}_{m \text{ sumandos}} = \underbrace{(1+1+\cdots+1)}_{nm \text{ sumandos}}$,
- $(n \cdot 1)(m \cdot 1) = (nm) \cdot 1 = nm$.

Ejemplo 2.2.15. El conjunto $\mathbb{Z}$ es un anillo conmutativo con unitario. El 1 es el neutro aditivo y el 0 el neutro multiplicativo.

Definición 2.2.16. Sea R un anillo con unitario $1 \neq 0$. Un elemento u es llamado unidad de R si tiene inverso multiplicativo. Si todo elemento en R distinto de cero es una unidad, entonces R es un *anillo con división*. Un *campo* es un anillo conmutativo con división.

Observación 2.2.17. Similar a los grupos, un *subanillo* de un anillo R, es un subconjunto que con las operaciones inducidas de R es un anillo por si mismo. Este concepto aplica de igual forma para un *subcampo*. De igual forma un anillo es finito si el conjunto R es finito.

Definición 2.2.18. Sea R un anillo y 0 el neutro aditivo de R. Sean $0 \neq r, s \in R$. El elemento r es llamado *divisor izquierdo de cero* si $rs = 0$; respectivamente es llamado *divisor derecho de cero* si $sr = 0$. Usualmente se omite si es un divisor derecho o izquierdo de cero y se dice que r y s son divisores de cero.

•

Ejemplo 2.2.19. El conjunto $\mathbb{Z}_4$ es un anillo con divisores de cero. Consideremos $2 \cdot 2 = 4 \equiv 0$ mód 4. Entonces el 2 es un divisor de cero.

Observación 2.2.20. El teorema 2.2.7 nos dice que el cero siempre es un divisor de cero.

Teorema 2.2.21. *En el anillo $\mathbb{Z}_n$ los divisores de cero son los elementos distintos de cero que no son primos relativos con n.*

Ejemplo 2.2.22. En $\mathbb{Z}_{12}$ los divisores de cero son $2, 3, 4, 6, 8, 9$ y 10.

Corolario 2.2.23. *Sea p un primo. El anillo $\mathbb{Z}_p$ no tiene divisores de cero*

Teorema 2.2.24. *Las leyes de cancelación para la multiplicación en un anillo R se cumplen si y sólo si, R no tiene divisores de cero.*

Definición 2.2.25. Un anillo conmutativo con unitario sin divisores de cero (distintos de cero) es llamado un *dominio entero.*

Teorema 2.2.26. *Todo campo es un dominio entero.*

Teorema 2.2.27. *Todo dominio entero finito es un campo.*

Definición 2.2.28. Sea R un anillo. Si existe un entero positivo n tal que $n \cdot a = 0$ para toda $a \in R$, entonces el menor entero que cumple esta propiedad es llamado la *característica* del anillo R. Si no existe este entero, se dice que la característica es 0.

Teorema 2.2.29. *La característica de un dominio entero es siempre* 0 *o un número primo p.*

Corolario 2.2.30. *Todo campo finito tiene característica un número primo p y cardinalidad(orden) p^n para $n \in \mathbb{Z}^+$*

Ejemplo 2.2.31. El anillo $\mathbb{Z}_n$ tiene caractrística n, en tanto que, $\mathbb{R}, \mathbb{Q}$ y $\mathbb{C}$ tienen característica 0.

Observación 2.2.32. Los elementos distintos de cero en un campo forman un grupo bajo la multiplicación. Este grupo es usualmente llamado grupo multiplicativo.

•

•

2.2.1. Teorema de Fermat y Euler

Teorema 2.2.33 (Pequeño teorema de Fermat)**.** *Sea $a \in \mathbb{Z}$ y p un primo que no divide a a, entonces $p \mid a^{p-1} - 1$, esto es $a^{p-1} \equiv 1 (mód\ p)$.*

Corolario 2.2.34. *Si $a \in \mathbb{Z}$, entonces $a^p \equiv a (mód\ p)$*

Teorema 2.2.35. *El conjunto $\mathbb{Z}_n^*$ de todos los elementos distintos de cero de $\mathbb{Z}_n$ que no son divisores de cero forma un grupo bajo la multiplicación.*

Observación 2.2.36. El teorema 2.2.35 nos dice que la cardinalidad del conjunto $\mathbb{Z}_n^*$ concuerda con la función de Euler, $\varphi(n)$ de la definición A.1.43.

Ejemplo 2.2.37. En el ejemplo 2.2.22 se dice que los divisores de cero de $\mathbb{Z}_{12}$ son $2, 3, 4, 6, 8, 9$ y 10. Los elementos que no son divisores de cero son $1, 5, 7$ y 11. Y $\varphi(12) = 4$.

Teorema 2.2.38 (Teorema de Euler)**.** *Si $a \in \mathbb{Z}$ es primo relativo a $n \in \mathbb{Z}$ entonces, $n \mid a^{\varphi(n)-1}$. Esto es $a^{\varphi(n)} \equiv 1 (mód\ n)$.*

2.3. Retículos

A continuación se presenta el concepto de retículo. A pesar de existir una amplia teoría relativa a este objeto matemático, en este documento nos referiremos únicamente a la aplicación de estos a la criptografía. Antes de hablar de retículos se expondrán brevemente las bases de los espacios vectoriales ya que algunas propiedades y resultados de estos son aplicados a los retículos.

2.3.1. Espacios vectoriales

Definición 2.3.1. Sea V un conjunto no vacio, K un campo y

$$+ : \begin{array}{ccc} V \times V & \to & V \\ (v_1, v_2) & \mapsto & v_1 + v_2 \end{array} \quad , \quad * : \begin{array}{ccc} K \times V & \to & V \\ (c, v) & \mapsto & c * v \end{array}$$

operaciones binarias. Un *espacio vectorial* es una tupla $(V, +, *)$ en la cual las operaciones llamadas respectivamente *suma* y *producto por escalar* , satisfacen:

1. $(V, +)$ es un grupo abeliano.
2. Para $u, v \in V$ y $c \in K$ se cumplen las leyes distributivas. (Ver definición 2.2.1)

∴

3. La operación $*$ es asociativa con respecto a los elementos de K. Esto es, para cualesquiera $c_1, c_2 \in K$ y $v \in V$ se cumple que $(c_1c_2)v = c_1(c_2v)$.

4. El neutro multiplicativo de K, digamos 1, satisface que $1 * v = v$ para todo $v \in V$.

Los elementos que conforman un espacio vectorial son llamados *vectores*. Es común abreviar el término "espacio vectorial sobre un campo K" escribiendo que $(V, +, \cdot)$ es un *K-espacio vectorial*.

Notación 2.3.2. Al igual que en grupos, anillos y campos, cuando el contexto lo permita indicaremos que V es un espacio vectorial omitiendo el campo K y las operaciones.

Ejemplo 2.3.3. El plano euclidiano $\mathbb{R}^2$, es un $\mathbb{R}-$espacio vectorial donde la operación suma consiste en sumar entrada a entrada los vectores y la multiplicación escalar está dada por multiplicar cada entrada del vector por el escalar. Por ejemplo

- $(3, 4) + (4, 7) = (3 + 4, 4 + 7) = (7, 11)$
- $\frac{1}{2} \cdot (8, 12) = (\frac{1}{2}8, \frac{1}{2}12) = (4, 6)$.

En general el conjunto $\mathbb{R}^n$ es un $\mathbb{R}-$espacio vectorial

Definición 2.3.4. Sea V un espacio vectorial y W un subconjunto de V. Decimos que W es un *subespacio vectorial* de V si satisface que es un espacio vectorial con las operaciones de V.

Teorema 2.3.5. *Sea W un subconjunto de un K-espacio vectorial V, entonces W es un subespacio vectorial de V si y solo si satisface:*

1. *$W \neq \emptyset$*
2. *si $v, w \in W$ entonces $v + w \in W$.*
3. *si $v \in W$ y $c \in K$ entonces $cv \in W$.*

Ejemplo 2.3.6. Sea $\mathbb{R}^2$ como en el ejemplo 2.3.3. Fijemos dos enteros, digamos, $a, b \in \mathbb{Z}$, entonces el conjunto $W = \{(x, y) \in \mathbb{R}^2 \mid ax + by = 0\} \subset V$ es un subespacio vectorial de V. Verificamos que se cumplen las tres condiciones del teorema 2.3.5.

1. W es no vacio. Puesto que, para cualesquiera $a, b \in \mathbb{Z}$ se cumple que $a \cdot 0 + b \cdot 0 = 0$, entonces $(0, 0) \in W$.

•

...

2. Sean $v = (x_1, y_1), w = (x_2, y_2) \in W$ verifiquemos que la suma $(x_1 + x_2, y_1 + y_2)$ está en W. Ya que $v, w \in W$ se tiene que $ax_1 + by_1 = 0$ y que $ax_2 + by_2 = 0$, sumando ambas ecuaciones obtenemos $a(x_1 + x_2) + b(y_1 + y_2) = 0$. Esto muestra que $(x_1 + x_2, y_1 + y_2) \in W$.

3. Sea $c \in \mathbb{R}$ y $v = (x, y) \in W$. Tenemos que $cv = c(x, y) = (cx, cy)$. Como la multiplicación es asociativa y distributiva en $\mathbb{R}$ tenemos que $a(cx) + b(cy) = c(ax) + c(by) = c(ax + by)$. Como $(x, y) \in W$, entonces se cumple que $ax + by = 0$, multiplicando esta ecuación por c obtenemos $c(ax + by) = c \cdot 0 = 0$, lo cual demuestra que $cv \in W$.

Como se cumplen las condiciones del Teorema 2.3.5, entonces verificamos que W es un subespacio vectorial de V.

Definición 2.3.7. Sean $S = \{v_1, v_2, \ldots, v_n\}$, un conjunto de n vectores de un K-espacio vectorial V. El conjunto de todas las combinaciones lineales de estos vectores forma un espacio vectorial y es llamado *espacio generado* por $\{v_1, v_2, \ldots, v_n\}$ y se denota $\langle S\rangle = \langle\{v_1, v_2, \ldots, v_n\}\rangle$. En notación de conjuntos:

$$\begin{aligned}\langle S\rangle &= \{v \in V \mid v \text{ es una combinación lineal de } v_1, v_2, \ldots, v_n\} \\ &= \left\{\sum_{i=1}^{n} c_i v_i \mid v_i \in S, c_1, c_2, \ldots, c_n \in K\right\}\end{aligned}$$

Ejemplo 2.3.8. Sea $\mathbb{R}^3$ un $\mathbb{R}$-espacio vectorial y $S = \{(1, 1, 0), (0, 1, 0)\}$. El espacio generado $\langle S\rangle$ es el conjunto

$$\begin{aligned}\langle S\rangle &= \{a(1, 1, 0) + b(0, 1, 0) \mid a, b \in \mathbb{R}\} \\ &= \{(a, b, 0) \mid a, b \in \mathbb{R}\}\end{aligned}$$

El espacio generado por S coincide con un plano en $\mathbb{R}^3$.

Ejemplo 2.3.9. Sea $S = \{(1, 0), (0, 1)\}$ un subconjunto de vectores del $\mathbb{R}$-espacio vectorial $\mathbb{R}^2$. El conjunto

$$\begin{aligned}\langle S\rangle =& \{a(1, 0) + b(0, 1) \mid a, b \in \mathbb{R}\} \\ & \{(a, b) \mid a, b \in \mathbb{R}\}\end{aligned}$$

coincide con la definición de $\mathbb{R}^2$, por lo tanto $\langle S\rangle = V$. Decimos que $\langle S\rangle$ genera a $\mathbb{R}^2$.

Definición 2.3.10. Sea V un K-espacio vectorial. Se dice que un conjunto no vacío $S = \{v_1, v_2, \ldots, v_n\}$ de vectores de V es *linealmente dependiente* si existen escalares $c_1, c_2, \ldots, c_n$ no todos cero tales que $c_1v_1 + c_2v_2 + \cdots + c_nv_n = 0$. Se dice que S es *linealmente independiente* si no es linealmente dependiente.

Ejemplo 2.3.11. Sea $W = \{(1,2),(2,4)\}$ un subconjunto de $\mathbb{R}^2$ como $\mathbb{R}$-espacio vectorial. Afirmamos que W es linealmente dependiente. Si tomamos los escalares 2 y (-1) obtenemos

$$\begin{aligned} 2(1,2) + (-1)(2,4) =& (2,4) + (-2,-4) \\ & (2 + (-2), 4 + (-4)) \\ & (0,0). \end{aligned}$$

Dado que existe una combinación lineal de los vectores de W que forman el vector $(0,0)$ tal que los escalares no son todos cero concluimos que, W es linealmente dependiente. Si consideramos el subconjunto $S = \{(1,0),(0,1)\}$, afirmamos que es linealmente independiente. Sean $a, b \in \mathbb{R}$, tenemos

$$\begin{aligned} a(1,0) + b(0,1) =& (a,0) + (0,b) \\ & (a,b) \end{aligned}$$

Ahora $(a,b) = (0,0)$ si y sólo si $a = 0$ y $b = 0$, por lo que se concluye que el conjunto S es linealmente independiente.

Definición 2.3.12. Un conjunto S de vectores en un espacio vectorial V es linealmente dependiente si contiene un subconjunto finito linealmente dependiente.

Teorema 2.3.13. *Sea $\{v_1, v_2, \ldots, v_n\}$ un conjunto linealmente independiente de vectores de un espacio vectorial V. La representación de cada vector $v \in \langle\{v_1, v_2, \ldots, v_n\}\rangle$ como combinación lineal de estos vectores es única.*

Definición 2.3.14. Sea V un espacio vectorial. Una *base* para V es un subconjunto de V que es linealmente independiente y genera a V.

Ejemplo 2.3.15. De los ejemplos 2.3.9 y 2.3.11 tenemos que el conjunto $S = \{(1,0),(0,1)\}$ es una base para el espacio vectorial $\mathbb{R}^2$.

Teorema 2.3.16. *Sea V un espacio vectorial generado por un conjunto finito de vectores $v_1, v_2, \ldots, v_m$. Entonces todo conjunto linealmente independiente de vectores de V es finito y además no contiene más de m elementos.*

•

Corolario 2.3.17. *Sea V un espacio vectorial no nulo. Si β y β' son dos bases finitas de V, entonces $|\beta| = |\beta'|$.*

Definición 2.3.18. De acuerdo al corolario 2.3.17, podemos definir la *dimensión* de un espacio vectorial como la cardinalidad de una base cualquiera de V. Denotamos la dimensión de un espacio vectorial V por dim V.

Corolario 2.3.19. *Sea V un espacio vectorial de dimensión finita y sea $n = dim\ V$. Entonces:*

- *Todo subconjunto de V con más de n vectores es linealmente dependiente.*
- *Todo subconjunto de V con menos de n vectores no puede generar a V.*

Corolario 2.3.20. *Sea V un espacio vectorial de dimensión finita y sea W un subespacio de V. Entonces*

- *dim W $\leq$ dim V.*
- *dim W $=$ dim V si y sólo si $W = V$.*

Definición 2.3.21. Sean $v, w \in V \subset \mathbb{R}^m$. Esto es $v = (x_1, x_2, \ldots, x_m)$, y $w = (y_1, y_2, \ldots, y_m)$.

- El *producto punto* de v y w esta dado por

$$v \cdot w = x_1y_1 + x_2y_2 + \cdots + x_my_m.$$

- Decimos que v y w son *ortogonales* si $v \cdot w = 0$.
- Denotamos por $||v||$ a la cantidad $\sqrt{x_1^2 + x_2^2 + \cdots + x_m^2}$. $||v||$ es llamada la *norma euclidiana*.

Observación 2.3.22. El producto punto y la norma euclidiana están relacionados por la siguiente formula

$$v \cdot v = ||v||^2.$$

Definición 2.3.23. Sea V un espacio vectorial y $\beta = \{v_1, v_2, \ldots, v_n\}$ una base para V.

- Decimos que β es una *base ortogonal* si satisface que $v_i \cdot v_j = 0$ para toda $i \neq j$.

- Decimos que β es una *base ortonormal* si es una base ortogonal y además $||v_i|| = 1$ para toda i.

Teorema 2.3.24 (Algoritmo de Gram-Schmidt)**.** *Sea $\beta = \{v_1, v_2, \ldots, v_n\}$ una base de un espacio vectorial $V \subset \mathbb{R}^m$. El algoritmo 2.3.1 genera una base ortogonal $\beta^*\{v_1^*, v_2^*, \ldots, v_n^*\}$ para V.*

Algoritmo 2.3.1: Gram-Schmidt

Entrada: Base $\beta = \{v_1, v_2, \ldots, v_n\}$
Salida: Base ortogonal $\beta^*\{v_1^*, v_2^*, \ldots, v_n^*\}$
inicio
  $v_1^* \leftarrow v_1$
  para $i \leftarrow 2, 3, \ldots, n$ **hacer**
    $\mu_{ij} \leftarrow (v_i \cdot v_j^*)/||v_j||^2$ para $1 \leq j < i$
    $v_i^* \leftarrow v_i - \sum_{j=1}^{i-1} \mu_{ij} v_j^*$
  devolver $\beta^*\{v_1^*, v_2^*, \ldots, v_n^*\}$

2.3.2. Aspectos básicos de retículos

El contenido de esta sección ha sido tomado de [16].

Definición 2.3.25. Sean V un conjunto no vacío y R un anillo conmutativo con unitario. Sean las operaciones $+, *$ como en la definición 2.3.1 con la diferencia de que los coeficientes se toman del anillo R, entonces, decimos que $(V, +, *)$ es un *R-módulo.*

Observación 2.3.26. Los conceptos y resultados de dimensión y bases en espacios vectoriales se mantienen para los R-módulos dado que no dependen de alguna propiedad restringida a los campos.

Definición 2.3.27. Sea $V \subset \mathbb{R}^m$. Decimos que V es un *Retículo* si V es un $\mathbb{Z}$-módulo.

Definición 2.3.28. Un *Retículo integral (o entero)* es un retículo en el que los vectores tienen coordenadas enteras. Esto es, sea $V \subset \mathbb{Z}^n$,decimos que V es un retículo integral(o entero) si V es un retículo.

Ejemplo 2.3.29. Considerar el retículo $L \subset \mathbb{R}^3$ generado por los tres vectores

$$v_1 = (2, 1, 3), \qquad v2 = (1, 2, 0), \qquad v_3 = (2, -3, -5).$$

Al ser generada por vectores con entradas en $\mathbb{Z}$, y dado que los escalares son tomados de $\mathbb{Z}$, todos los vectores que conforman este retículo tienen coordenadas enteras.

Dos problemas

Definición 2.3.30 (El problema del vector más corto)**.** Sea L un retículo. El problema del vector más corto(SVP por sus siglas en inglés) consiste en encontrar el vector más corto (diferente del vector cero) en un retículo. Esto es, encontrar un vector, diferente de cero, $v \in L$ tal que su norma euclidiana $||v||$ sea mínima.

Definición 2.3.31 (El problema del vector más cercano)**.** Dado un vector $w \in \mathbb{R}^m$ que no esta en un retículo L, encontrar un vector $v \in L$ que este cercano a w. Esto es encontrar un vector $0 \neq v \in L$ que minimize la norma eucidiana de $||w - v||$.

Observación 2.3.32. Existe más de un vector que cumpla con las características antes mencionadas, por ejemplo, en $\mathbb{Z}^2$ los cuatro vectores $(0, \pm 1), (\pm 1, 0)$ satisfacen el problema del vector más corto.

Existe una variante del problema del vector más corto. Esta variante, extiende el concepto y busca una base *corta*.

Definición 2.3.33 (Problema de la base más corta)**.** Este problema consiste en encontrar una base $v_1, \ldots, v_n$ corta para un retículo. Es corta en el sentido de que , por ejemplo

$$\max_{1 \leq i \leq n} ||v_i|| \sum_{i=1}^{n} ||v_i||^2$$

sea mínimo. Existen diferentes versiones de este problema

Teorema 2.3.34 (Babai)**.** *Sea $L \subset \mathbb{R}^n$ un retículo con base $v_1, v_2, \ldots, v_n$ y sea $w \in \mathbb{R}^n$ un vector cualquiera. Si los vectores en la base son lo suficientemente ortogonales con respecto a los demás, entonces el algoritmo 2.3.2 resuelve el problema del vector más cercano.*

Observación 2.3.35. Decimos que dos vectores v y w son *suficientemente ortogonales* si satisfacen que $v \cdot w \approx 0$.

Algoritmo 2.3.2: Aproximación de Babai

Entrada: Base $\beta = \{v_1, v_2, \ldots, v_n\}$ suficientemente ortogonal, vector $w \in \mathbb{R}^n$
Salida: Vector $v \in L$ cercano a w
inicio
  Encontrar $t_1, \ldots, t_n \in \mathbb{R}$, tales que $w = t_1 v_1 + \cdots + t_n v_n$
  $a_i \leftarrow \lfloor t_i \rceil$ para $i = 1, \ldots, n$.
  devolver $v = a_1 v_1 + a_2 v_2 + \cdots + a_n v_n$

Observación 2.3.36. En el algoritmo 2.3.2 se hace uso del operador $\lfloor x \rceil$. Este operador denota el entero más cercano a x.

2.3.3. Algoritmos para reducción de retículos

Existen algunos algoritmos que permiten obtener una solución al problema de la base más corta. A continuación se muestran dos algoritmos principales.

Proposición 2.3.37 (Reducción gausiana de retículos)**.** *Sea $L \subset \mathbb{R}^2$ un retículo de dimensión 2 con base v_1 y v_2. El algoritmo 2.3.3 termina y da como resultado una buena base para L. Esto es una base suficientemente ortogonal y corta.*

Algoritmo 2.3.3: Reducción gausiana para retículos.

Entrada: Base $\beta = \{v_1, v_2\}$
Salida: Vector $v \in L$ cercano a w
inicio
  Ciclo
    si $||v_2|| < ||v_1||$ **entonces**
      Intercambiar v_1 y v_2.
    $m \leftarrow \lfloor v_1 \cdot v_2 / ||v_1||^2 \rceil$.
    si $m = 0$ **entonces**
      devolver v_1, v_2
    $v_2 \leftarrow v_2 - m v_1$.

Observación 2.3.38. El algoritmo 2.3.3 es un análogo del algoritmo de Euclides (Teorema A.1.29). Este algoritmo tiene complejidad $O(\log^2 \text{máx}(||v_2||, ||v_1||))$,

Iteración	v_1	v_2	m
1	$(6513996, 6393464)$	$(66586820, 65354729)$	10
2	$(1446860, 1420089)$	$(6513996, 6393464)$	5
3	$(-720304, -706981)$	$(1446860, 1420089)$	-2
4	$(6252, 6127)$	$(-720304, -706981)$	-115
5	$(-1324, -2376)$	$(6252, 6127)$	-3
6	$(2280, -1001)$	$(-1324, -2376)$	0

Tabla 2.4: Ejemplo de ejecución del algoritmo gausiano

esto es, polinomial con respecto al tamaño de la norma de los vectores de entrada.

Ejemplo 2.3.39. Sea L el retículo generado por la base $v_1 = (66586820, 65354729)$ y $v_2 = (6513996, 6393464)$. Aplicando el algoritmo 2.3.3 a este par de vectores obtenemos lo siguiente

- Calculamos $||v_1||^2 \approx 8.71{*}10^{15}$ y $||v_2|| \approx 8.33{*}10^{13}$. Dado que v_2 es más corto que v_1 los intercambiamos. Entonces $v_1 = (6513996, 6393464)$ y $v_2 = (66586820, 65354729)$.
- Sustraemos un múltiplo de v_1 a v_2. El múltiplo es

$$m = \left\lfloor \frac{v_1 \cdot v_2}{||v_1||^2} \right\rceil = \lfloor 10.2221 \rceil = 10.$$

- Como $m \neq 0$, entonces reemplazamos v_2 por $v_2 - mv_1 = (1446860, 1420089)$.
- Este nuevo vector tiene norma $||v_2||^2 \approx 4.11 * 10^12$. Como es más pequeño que $||v_1||^2$, entonces los intercambiamos de nuevo. Tenemos que $v_1 = (1446860, 1420089)$ y $v_2 = (6513996, 6393464)$.
- Repetimos este proceso hasta que el algoritmo termine. La tabla 2.4 muestra el cambio de los vectores en cada iteración del algoritmo.
- De acuerdo a la tabla 2.4 la base corta es $v_1 = (2280, -1001)$ y $v_2 = (-1324, -2376)$.

Algoritmo LLL para reducción de retículos

Definición 2.3.40. Sea $\beta = \{v_1, v_2, \ldots, v_n\}$ una base para un retículo L y sea $\beta^* = \{v_1^*, v_2^*, \ldots, v_n^*\}$ la base ortogonal obtenida por el algoritmo de

Gram-Schmidt (Algoritmo 2.3.1). La base β se dice que es *LLL-reducida*[5] si satisface las siguientes dos condiciones

(*Condición de tamaño*)

$$\mu_{ij}| = \frac{|v_i \cdot v_j^*|}{||v_j||^2} \leq \frac{1}{2} \qquad \text{para toda } 1 \leq j < i \leq n.$$

(*Condición de Lovász*)

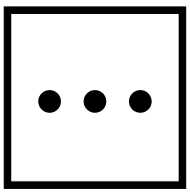
$$||v_i^* + \mu_{i,i-1} v_{i-1}||^2 \geq \frac{3}{4}||v_{i-1}^*||^2$$

Teorema 2.3.41. *Sea $\beta = \{v_1, v_2, \ldots, v_n\}$ una base para un retículo L, el algoritmo 2.3.4 termina en un número finito de pasos y regresa una base LLL-reducida. Este algoritmo es de orden polinomial en tiempo, de manera más precisa, sea $B = \max |v_i|$, entonces, el algoritmo ejecuta el ciclo principal no más de $O(n^2 \log(n) + n^2 \log(B))$ veces.*

Algoritmo 2.3.4: Reducción de retículos LLL

Entrada: Base $\beta = \{v_1, v_2\}$
Salida: Base LLL-reducida $\{v_1, v_2\}$
inicio
  $k \leftarrow 2$
  $v_1^* \leftarrow v_1$
  mientras $k \leq n$ **hacer**
    para $j = 1, 2, \ldots, n$ **hacer**
      $v_k \leftarrow v_k - \lfloor \mu_{kj} \rceil v_j^*$ `// [Reducción de tamaño]`
    si $||v_i^* + \mu_{i,i-1} v_{i-1}||^2 \geq \frac{3}{4}||v_{i-1}^*||^2$ `// [Condición de Lovász]`
     entonces
      $k \leftarrow k + 1$
    en otro caso
      Intercambiar v_{k-1} y v_k
      $k \leftarrow max(k - 1, 2)$

[5] El algoritmo LLL propuesto por Arjen Lenstra, Hendrik Lenstra y László Lovász en 1982.



Capítulo 3

Curvas elípticas

It is possible to write endlessly on elliptic curves. (This is not a threat.)[1]

Serge Lang
-Elliptic Curves:Diophantine Analysis

En este capítulo se pretende cubrir la teoría elemental relativa a las curvas elípticas binarias. En la primera sección se define el concepto de curva elíptica binaria y la aritmética de campos binarios. En la segunda sección se introducen de manera breve las isogenias y de manera específica una fórmula para generarlas, la fórmula de *Vélu*. Por último hablaremos de uno de los principales temas de esta tesis, la multiplicación escalar de puntos en una curva elíptica binaria. El contenido de este capítulo es tomado del material disponible en [16], [31], [13] y [27].

[1] Es posible escribir interminablemente acerca de curvas elípticas.(Esto no es una amenaza.)

Definición 3.0.1 (Curva Elíptica)**.** Una *Curva Elíptica* E sobre un campo K está determinada por la siguiente ecuación [2]

$$E: \ y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

donde los coeficientes $a_1, a_2, a_3, a_4, a_6 \in K$ y satisfacen que $\Delta \neq 0$, donde Δ está definido por las siguientes ecuaciones

$$\begin{aligned} &b_2 = a_1^2 + 4a_2, \qquad b_4 = 2a_4 + a_{13}, \qquad b_6 = a_3^2 + 4a_6, \\ &b_8 = a_1^2a_6 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2 \\ &\Delta = -b_2^2b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6. \end{aligned}$$

La última condición de la definición nos asegura que una curva elíptica es *suave*, o *no singular*.

Notación 3.0.2. Denotamos por $E(K)$ al conjunto de los elementos de una curva elíptica. Estos elementos estan en el conjunto K^2 y son tales que son raíces de la ecuación que determina a E. A estos elementos los denominamos *puntos*.

Notación 3.0.3. Al campo K sobre el cuál está definida una curva elíptica E se le llama *campo base*.

El conjunto $E(K)$, que contiene los puntos de una curva elíptica y un punto al infinito que denotaremos como $\mathcal{O}$, forma un grupo abeliano con la operacion suma la cual se define a continuación:
Sean P, Q puntos de una curva $E(K)$, donde $P = (x_1, y_1), Q = (x_2, y_2)$. Definimos $R = P + Q = (x_3, y_3)$ donde

$$\begin{aligned} \lambda =& \frac{y_2 - y_1}{x_2 - x_1} \\ x_3 =& \lambda^2 - x_1 - x_2 \\ y_3 =& \lambda(x_1 - x_3) - y_1. \end{aligned}$$

En el caso particular cuando $P = Q = (x, y)$ decimos que tenemos la operación de *doblado* pues $R = 2P = (x_1, y_1)$ y se calcula de la siguiente manera:

[2] Esta ecuación es conocida como la ecuación elíptica de Weierstrass

$$\begin{aligned}\lambda =& \frac{3x^2 + a_4}{2y} \\ x_1 =& \lambda^2 - 2x \\ y_1 =& \lambda(x - x_1) - y\end{aligned}$$

si $x_1 = x_2$ pero $y_1 \neq y_2$ entonces $P + Q = \mathcal{O}$. si $P = Q = (x, y)$, pero $y = 0$ entonces $P + Q = \mathcal{O}$.

Observación 3.0.4. En el caso de una curva elíptica como grupo el punto al infinito $\mathcal{O}$ es el elemento neutro y se tiene que, $P + \mathcal{O} = P = \mathcal{O} + P$. (Definición 2.1.4) .

Notación 3.0.5. En esta sección y en el resto del documento se utilizará la letra $\mathbb{F}$ para denotar un campo finito. Si se tiene que $|\mathbb{F}| = q$, entonces lo denotaremos por $\mathbb{F}_q$. El corolario 2.2.30 nos dice que $q = p^k$ para algún primo p y un entero positivo k.

Teorema 3.0.6 (Hasse)**.** *Sea E una curva elíptica definida sobre un campo $\mathbb{F}_q$, entonces*

$$\#E(\mathbb{F}_q) = q + 1 - t_q \qquad \textit{donde } |t_q| \leq 2\sqrt{q}.$$

Ejemplo 3.0.7. Trabajemos con el campo $\mathbb{F}_9 = \{a+bi \mid a, b \in \mathbb{F}_3, i^2 = -1\}$ y la curva elíptica

$$E : Y^2 = X^3 + (1 + i)X + (2 + i).$$

Por ensayo y error podemos encontrar que hay 10 puntos en $E(\mathbb{F}_9)$,

$$\begin{gathered}(2i, 1 + 2i), \quad (2i, 2 + i), \quad (1 + i, 1 + i), \quad (1 + i, 2 + 2i), \quad (2, 0), \\ (2 + i, i), \quad (2 + i, 2i), \quad (2 + 2i, 1), \quad (2 + 2i, 2), \quad \mathcal{O}.\end{gathered}$$

De acuerdo al Teorema de Hasse (3.0.6) se tiene que $E(\mathbb{F}_9) = 3^2 + 1 - t_9$, donde t_9 es menor que $2\sqrt{3} \approx 3.4641$ por lo que el teorema se cumple en este ejemplo.

Definición 3.0.8. El número $E(\mathbb{F})$ es llamado el *orden* de una curva elíptica. El orden de un punto $P \in E$ esta dado por $|\langle P \rangle|$.

$+$	0	1	u	$u+1$
0	0	1	u	$u+1$
1	1	0	$u+1$	u
u	u	$u+1$	0	1
$u+1$	$u+1$	u	1	0

$\cdot$	0	1	u	$u+1$
0	0	0	0	0
1	0	1	u	$u+1$
u	0	u	$u+1$	1
$u+1$	0	$u+1$	1	u

Figura 3.1: Tablas del campo $\mathbb{F}_4 = \mathbb{F}_2[u]/\langle p(u)\rangle$, donde $p(u) = u^2 + u + 1$

3.1. Curvas elípticas binarias

Definición 3.1.1. Sea $\mathbb{F}_2[x]$ el anillo de polinomios del campo $\mathbb{F}_2 = \{0, 1\}$. Sea $p(x)$ un polinomio irreducible de grado d en $\mathbb{F}_2[x]$. El conjunto cociente $\mathbb{F}_2[x]/\langle p(x)\rangle$ forma un campo con la suma usual de polinomios y la multiplicación módulo el polinomio $p(x)$. Es usual llamar a este campo , *campo binario de grado d* y lo denotamos por $\mathbb{F}_{2^d}$. Decimos que el campo $\mathbb{F}_{2^d}$ es una *extensión de grado d* del campo $\mathbb{F}_2$.

Ejemplo 3.1.2. Consideremos el anillo $\mathbb{F}_2[u]$ y sea $p(u) = u^2 + u + 1$ un polinomio irreducible en este anillo. Construimos el campo $\mathbb{F}_{2^2} = \mathbb{F}_4 = \{0, 1, u, u+1\}$. Las tablas de grupo bajo multiplicación y suma se presentan en la figura 3.1. Como ejemplo tomemos los elementos u y $u + 1$, si los multiplicamos obtenemos

$$\begin{aligned} u(u+1) &= u^2 + u \ (\text{mód } u^2 + u + 1) \\ &= u + 1 + u \ (\text{mód } u^2 + u + 1) \\ &= 1. \end{aligned}$$

Entonces u es el inverso multiplicativo de $u + 1$ y viceversa.

Definición 3.1.3. Una curva elíptica definida por la ecuación

$$E : Y^2 + XY = X^3 + aX^2 + b,$$

donde $a, b \in \mathbb{F}_{2^m}$ y $b \neq 0$, se dice que es una *curva elíptica binaria.*

Definición 3.1.4. Una curva de *Koblitz* es una curva elíptica binaria E_a donde los coeficientes descritos en la definición 3.1.3 satisfacen que $b = 1$ y $a \in \mathbb{F}_2$.

Ejemplo 3.1.5. Sea $E_0 : Y^2 + XY = X^3 + 1$ una curva de Koblitz definida sobre $\mathbb{F}_2$, entonces

$$E_0(\mathbb{F}_2) = \{(0, 1), (1, 0), (1, 1), \mathcal{O}\}.$$

••

Podemos ver que el teorema de Hasse (3.0.6) se cumple en esta curva elíptica binaria.

$$E(\mathbb{F}_2) = 4 = p + 1 - t_2 = 2 + 1 - (-1).$$

En general se cumple para todas las curvas elípticas binarias.

Definición 3.1.6. La función *traza* en $\mathbb{F}_{2^m}$ es la función $\mathrm{Tr} : \mathbb{F}_{2^m} \to \mathbb{F}_2$ definida por

$$\mathrm{Tr}(c) = c + c^2 + c^{2^2} + \cdots + c^{2m-1}.$$

Teorema 3.1.7 (Teorema 3.18, página 86 [13])**.** *Sean $E_1 : y^2 + xy = x^3 + ax^2 + b$ y $E_2 : y^2 + xy = x^3 + \bar{a}x^2 + \bar{b}$ curvas elípticas binarias. Se cumple que $E_1 \cong E_2$ si y sólo si $Tr(a) = Tr(\bar{a})$ y $b = \bar{b}$.*
Más aún, si se cumplen estas condiciones, se tiene que existe $s \in \mathbb{F}_{2^m}$ tal que $\bar{a} = s^2 + s + a$ y se tiene el isomorfismo $\phi : E_1 \to E_2$ definido por:

$$\phi((x, y)) \mapsto (x, y + sx).$$

Observación 3.1.8. En curvas elípticas binarias, se dice que el isomorfismo ϕ del teorema 3.1.7 es una *involución*, esto es que es su propio inverso ($\phi^{-1} = \phi$). Es fácil comprobarlo. Sea $P = (x, y)$ un punto de una curva elíptica binaria, aplicamos ϕ a este punto y tenemos $\phi(P) = \tilde{P} = (x, y + sx)$. Ahora aplicamos ϕ a $\tilde{P}$ y tenemos $\phi(\tilde{P}) = (x, (y + sx) + sx) = (x, y + sx + sx) = (x, y) = P$. De aquí concluimos que ϕ es su propio inverso que es lo mismo que decir que ϕ^2 es el endomorfismo identidad en E.

3.2. Isogenias

Definición 3.2.1. Sean E_1 y E_2, curvas elípticas. Una *isogenia* entre E_1 y E_2 es un homomorfismo

$$\phi : E_1 \to E_2$$

que satisface $\phi(\mathcal{O}) = \mathcal{O}$. Decimos que E_1 y E_2 son *isógenas* si existe una isogenia entre ellas con $\phi(E_1) \neq \{\mathcal{O}\}$

Ejemplo 3.2.2. Para $m \in \mathbb{Z}$ podemos definir la isogenia

$$[m] : E \to E,$$

llamada *multiplicación por m*. Esta isogenia está definida por

$$[m](P) = \underbrace{P + P + P + \cdots + P}_{m-1 veces}.$$

Entonces si $m = 2$, tenemos que $[m](P) = P+P$ y si $m < 0$, la definimos por $[m](-P)$, por ejemplo $m = -3$ tenemos que $[-m](-P) = [-(-3)](-P) = (-P) + (-P) + (-P)$.

Observación 3.2.3. Una isogenia puede ser tal que $\phi(E_1) = \{\mathcal{O}\}$ o de lo contrario, ser suprayectiva. Dado que dos curvas son isógenas si $\phi(E_1) \neq \{\mathcal{O}\}$, entonces podemos decir que una isogenia entre dos curvas es suprayectiva siempre que sea diferente de la isogenia *cero* definida por $[0](P) = \mathcal{O}$.

Proposición 3.2.4. *Sea E una curva elíptica y sea $m \in \mathbb{Z}$, $m \neq 0$. Entonces la multiplicación por m, definida en el ejemplo 3.2.2, es no-constante, esto es, si fijamos $Q \in E$, existen al menos un punto $P \in E$ tal que $P \neq \mathcal{O}$ y $[m](P) \neq Q$.*

Teorema 3.2.5. *Sea $\phi : E_1 \to E_2$ una isogenia. Entonces, para cualesquiera puntos $P, Q \in E$, se cumple que $\phi(P + Q) = \phi(P) + \phi(Q)$.*

Observación 3.2.6. Dado que las curvas elípticas son un grupo bajo la adición, entonces los resultados de la sección 2.1 son válidos. Del teorema 3.2.5 podemos concluir, de manera particular que, los resultados de la sección 2.1.3 aplican a estas.

Definición 3.2.7. Sea $P = (x, y)$ un punto de una curva elíptica E_1. Sea E_2 una curva elíptica isógena a E_1. Podemos describir una isogenia $\phi : E_1 \to E_2$ de la siguiente manera:

$$\phi(P) = \phi((x, y)) = (r_1(x), yr_2(x)).$$

Donde r_1 y r_2 son *funciones racionales*, es decir, se pueden escribir como un cociente de dos polinomios, digamos $p(x)$ y $q(x)$, tales que son primos relativos entre sí. Entonces $r_1(x) = \dfrac{p_1(x)}{q_1(x)}$ y $r_2(x) = \dfrac{p_2(x)}{q_2(x)}$. Definimos el grado de ϕ como

$$grad(\phi) = \text{máx}\{grad(p_1(x)), grad(q_1(x))\}.$$

Teorema 3.2.8. *Sea $\phi : E_1 \to E_2$ una isogenia de grado d, entonces existe una isogenia $\hat{\phi} : E_2 \to E_1$ tal que $\hat{\phi}(\phi) = [d]$, para $[d]$ como en el ejemplo 3.2.2.*

••

••

Definición 3.2.9. Sea $\hat{\phi}$ como en el teorema 3.2.8. Entonces $\hat{\phi}$ es llamada la *isogenia dual*.

3.2.1. La fórmula de Vélu

En la sección 2.1.4 se mencionó una forma de construir homomorfismos entre un grupo G y algún grupo cociente de este. A continuación se presenta una fórmula para generar isogenias a partir de un subgrupo de una curva elíptica. Esta fórmula fue propuesta en 1971 por *Jacques Vélu* en [30].

Observación 3.2.10. Las curvas elípticas que se han mencionado siguen el modelo de Weierstrass. La fórmula de Vélu es aplicable a curvas con este modelo, sin embargo, existen otras formulas para curvas con base en otros modelos como los de Edwards o *Huff* por mencionar algunos.

Teorema 3.2.11 (Teorema 12.16, página 392 [31])**.** *Sea E una curva elíptica dada por la ecuación de Weierstrass*

$$E\text{: } y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

con todas las a_i en un campo K. Sea C un subgrupo de $E(K)$. Entonces existen una curva elíptica E_2 y una isogenia $\phi : E \to E_2$ tal que $C = Ker(\phi)$.

Para un punto $Q = (x_Q, y_Q) \in C$ tal que $Q \neq \mathcal{O}$ definimos

$$\begin{aligned} g_Q^x =& 3x_Q^2 + 2a_2x_Q + a_4 - a_1y_Q \\ g_Q^y =& -2y_Q - a_1x_Q - a_3 \\ v_Q =& \begin{cases} g_Q^x & (si 2Q = \mathcal{O}) \\ 2g_Q^x - a_1g_Q^y & (si 2Q \neq \mathcal{O}) \end{cases} \\ u_Q =& (g_Q^y)^2. \end{aligned}$$

Sea C_2 el conjunto que contiene a los puntos de orden dos en C. Se escoge $R \subset C$ tal que se tiene una unión disjunta

$$C = \{\mathcal{O}\} \cup C_2 \cup R \cup (-R)$$

Una manera de definir los conjuntos R y $(-R)$ es tal que, si $-P \in (-R)$ entonces $P \in R$. Sea $S = R \cup C_2$. Definimos los elementos v y w como

$$v = \sum_{Q \in S} v_Q, \qquad w = \sum_{Q \in S} (u_Q + x_Q v_Q).$$

..

...

Entonces la curva E_2 está definida por la siguiente ecuación

$$Y^2 + A_1XY + A_3Y = X^3 + A_2X^2 + A_4X + A_6,$$

donde

$$A_1 = a_1, \quad A_2 = a_2, \quad A_3 = a_3,$$
$$A_4 = a_4 - 5v, \quad A_6 = a_6 - (a_1^2 + 4a_2)v - 7w.$$

La isogenia ϕ está definida por

$\phi(x, y) = (X, Y)$, *donde*

$$X = x + \sum_{Q \in S} \left(\frac{v_Q}{x - x_Q} + \frac{u_Q}{(x - x_Q)^2} \right),$$

$$Y = y - \sum_{Q \in S} \left(u_Q \frac{2y + a_1x + a_3}{(x - x_Q)^3} + v_Q \frac{a_1(x - x_Q) + y - y_Q}{(x - x_Q)^2} + \frac{a_1u_Q - g_Q^x g_Q^y}{(x - x_Q)^2} \right).$$

3.2.2. Polinomios de División

El contenido de est sección puede encontrarse en la sección 3.2 de la referencia [31]

Definición 3.2.12. Un *polinomio de división* $\rho_m \in \mathbb{Z}[x, y, A, B]$ está dado por

$$\begin{aligned}
&\rho_0 = 0 \\
&\rho_1 = 1 \\
&\rho_2 = 2y \\
&\rho_3 = 3x^4 + 6Ax^2 + 12Bx - A^2 \\
&\rho_4 = 4y(x^6 + 5Ax^4 + 20Bx^3 - 5A^2x^2 - 4ABx - 8B^2 - A^3) \\
&\rho_{2m+1} = \rho_{m+2}\rho_m^3 - \rho_{m-1}\rho_{m+1}^3, \qquad \text{para } m \geq 2 \\
&\rho_{2m} = (2y)^{-1}(\rho_m)(\rho_{m+2}\rho_{m-1}^2 - \rho_{m-2}\rho_{m+1}^2) \qquad \text{para } m \geq 3.
\end{aligned}$$

••

••••

Teorema 3.2.13. *Sea $P = (x, y)$ un punto de una curva elíptica $y^3 = x^3 + Ax + B$ y sea n un entero positivo, entonces*

$$nP = \left(\frac{\phi_n(x)}{\rho_n^2(x)}, \frac{\omega_n(x,y)}{\rho_n(x,y)^3} \right).$$

Donde

$$\begin{aligned} \phi_m =& x\rho_m^2 - \rho_{m+1}\rho_{m-1} \\ \omega_m =& (4y)^{-1}(\rho_{m+2}\rho_{m-1}^2 - \rho_{m-2}\rho_{m+1}^2). \end{aligned}$$

Observación 3.2.14. Estos polinomios de división nos generan la isogenia $[n]$.

3.3. Multiplicación Escalar

Al principio de este capítulo se mencionó que las curvas elípticas forman un grupo aditivo. Se presentó una fórmula para obtener el doble de un punto P en una curva E, es decir, $2P$. Sin embargo es común tener la necesidad de calcular algún otro múltiplo del punto. En esta sección se presentan algoritmos para realizar este cálculo de una manera eficiente.

Definición 3.3.1. Sea k un entero. Sea E una curva elíptica y P un punto en la curva. La operación

$$kP = \underbrace{P + P + P + \cdots + P}_{K-1 veces}$$

es llamada *multiplicación escalar* de un punto.

Observación 3.3.2.

- En el ejemplo 3.2.2 se definió la isogenia de multiplicación por un entero m. Esta isogenia coincide con la definición anterior, sin embargo haremos de lado la notación de función y nos enfocaremos a la notación que se presentó en la sección 2.1.1, de manera concreta en el ejemplo 2.1.26.

- En la literatura a la multiplicación escalar se le puede encontrar como multiplicación de puntos.

∎

En criptografía se utilizan curvas elípticas definidas sobre campos finitos. Por cuestiones de seguridad supondremos que $\#E(\mathbb{F}_q) = rh$, donde r es un número primo y h es un entero *pequeño* ($q \approx r$). Del teorema 2.1.43 y del teorema de Hasse(3.0.6) podemos decir que existe un subgrupo de $E(\mathbb{F}_q)$, digamos N de orden r. Los puntos que consideraremos son puntos que pertenecen a N. Del colorario 2.1.37 tenemos que N es cíclico, esto es $N = \langle P \rangle$ para un punto $P \in N$.

Algunas consideraciones a tomar son

- El punto P a utilizar tiene orden r.
- la representación binaria de un entero la denotaremos como $k = (k_{t-1}, \ldots, k_1, k_0)_2$, donde $t \approx m = \lceil log_2(q) \rceil$.
- Denotaremos por A a la operación de adición y D a la de doblado.

El algoritmo 3.3.1 calcula kP con base en la representación binaria de k. El número esperado de *bits en uno* en la representación binaria de k es de $t/2 \approx m/2$. De aquí, el tiempo esperado de este algoritmo es el tiempo que toma realizar $m/2$ adiciones de puntos y m doblados de puntos.

$$\frac{m}{2}A + mD.$$

Algoritmo 3.3.1: Método binario de derecha a izquierda para multiplicación escalar.

Entrada: $k = (k_{t-1}, \ldots, k_1, k_0)_2$, $P \in \mathbb{F}_q$
Salida: kP
inicio
  $Q \leftarrow \mathcal{O}$
  para $i \leftarrow t-1, t-2, \ldots, 0$ **hacer**
    $Q \leftarrow 2Q$
    si $k_i = 1$ **entonces**
      $Q \leftarrow Q + P$.
  devolver Q

Definición 3.3.3 (NAF)**.** Una *forma no-adyacente*, conocido como *NAF* por sus siglas en inglés, de un entero positivo k es una expresión

$$k = \sum_{i=0}^{l-1} k_i 2^i$$

donde $k_i \in \{0, \pm 1\}$, $k_{l-1} \neq 0$ y son tales que no existen dos dígitos consecutivos que sean diferentes de cero, esto es, si $k_i \neq 0$ entonces, $k_{i+1} = k_{i-1} = 0$. La longitud de esta forma no adyacente es l.

Teorema 3.3.4. *Propiedades del NAF*

- *k tiene una única representación NAF denotada NAF(k).*
- *NAF(k) tiene menos digitos distintos de cero que cualquier otra representación signada de k.*
- *La longitud de NAF(k) es a lo más $t + 1$.*
- *Si la longitud de NAF(k) es l, entonces $2^l/3 < k < 2^{l+1}/3$.*
- *La densidad de dígitos diferentes de cero es aproximadamente $1/3$.*

El algoritmo 3.3.2 calcula NAF(k) de una manera eficiente. El algoritmo 3.3.3 es una modificación del algoritmo 3.3.1 para que utilice NAF(k) en lugar de la representación binaria(no signada) de k.

Ejemplo 3.3.5. Consideremos el entero $k = 1122334455$. A continuación se presenta la comparación entre su representación binaria y su representación NAF obtenida con el algoritmo 3.3.2 .

$$(k)_2 = 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1$$
$$NAF_2(k) = 1\ 0\ 0\ 0\ 1\ 0\ \bar{1}\ 0\ 0\ \bar{1}\ 0\ 1\ 0\ \bar{1}\ 0\ \bar{1}\ 0\ 0\ 0\ \bar{1}\ 0\ 0\ \bar{1}\ 0\ 0\ 0\ 0\ \bar{1}\ 0\ 0\ \bar{1}$$

Sean P y Q dos puntos en una curva elíptica E. En ocasiones es necesario calcular $k_0P + k_1Q$ de una manera eficiente. La estrategia natural sería la de calcular por separado k_0P y k_1Q y luego sumarlos. El algoritmo 3.3.4 nos permite calcular $k_0P + k_1Q$ de una manera simultánea.

Observación 3.3.6. El número de sumas que realiza el algoritmo 3.3.4 es igual al número de sumas necesarios si se calculara de manera individual k_0P y k_1Q. Sin embargo, el número de doblados es la mitad del necesario en el cálculo no simultaneo de k_0P y k_1Q.

Algoritmo 3.3.2: Cálculo de NAF para un entero positivo.

Entrada: Entero positivo k
Salida: NAF(k)
inicio
  $i \leftarrow 0$
  mientras $k \geq 1$ **hacer**
    si k *es impar* **entonces**
      $k_i \leftarrow 2 - (k \text{ mód } 4)$.
      $k \leftarrow k - k_i$
    en otro caso
      $k_i \leftarrow 0$
    $k \leftarrow k/2$
    $i \leftarrow i + 1$
  devolver $(k_{i-1}, k_{i-2}, \ldots, k_1, k_0)$.

Algoritmo 3.3.3: Método binario NAF de izquierda a derecha para multiplicación escalar.

Entrada: Entero positivo k, $P \in E(\mathbb{F}_q)$
Salida: $Q = kP$
inicio
  Usar el algoritmo 3.3.2 para calcular NAF$(k) = \sum_{i=0}^{l-1} k_i 2^i$ $Q \leftarrow \mathcal{O}$
  para $i = l-1, l-2, \ldots, 0$ **hacer**
    $Q \leftarrow 2Q$
    si $k_i = 1$ **entonces**
      $Q \leftarrow Q + P$.
    si $k_i = -1$ **entonces**
      $Q \leftarrow Q - P$.
  devolver Q

Algoritmo 3.3.4: Multiplicación escalar simultánea (Truco de Shamir-Strauss, algoritmo 3.48 de [13])

Entrada: Enteros positivos k_0 y k_1, puntos $P, Q \in E(\mathbb{F}_q)$
Salida: $R = k_0P + k_1Q$
inicio
  Usar el algoritmo 3.3.2 para calcular $\text{NAF}(k_0) = \sum_{i=0}^{l-1} a_i 2^i$ y $\text{NAF}(k_1) = \sum_{i=0}^{l-1} b_i 2^i$
  $R \leftarrow \mathcal{O}$
  para $i = l-1, l-2, \ldots, 0$ **hacer**
    $R \leftarrow 2R$
    si $a_i = 1$ **entonces**
      $R \leftarrow R + P$.
    si $a_i = -1$ **entonces**
      $R \leftarrow R - P$.
    si $b_i = 1$ **entonces**
      $R \leftarrow R + Q$.
    si $b_i = -1$ **entonces**
      $R \leftarrow R - Q$.
  devolver R

3.4. Endomorfismos eficientes

Definición 3.4.1. Un *endomorfismo* de una curva elíptica E, es un morfismo de la curva en sí misma. Es decir, ϕ es un endomorfismo si es un morfismo y

$$\phi : E \to E.$$

Observación 3.4.2. En la definición 2.1.51 se hace mención de los automorfismos de un grupo. La diferencia entre un automorfismo y un endomorfismo, es principalmente que, un endomorfismo no necesariamente es un isomorfismo.

Observación 3.4.3. Una isogenia puede comportarse como un endomorfismo. Ejemplo de esto es la multiplicación por m. Ver Ejemplo 3.2.2

Definición 3.4.4. Sea ϕ un endomorfismo de una curva elíptica E y $P \in E$. Decimos que ϕ es *eficiente* si satisface que el número de operaciones necesarias para calcular $\phi(P)$ es equivalente al de calcular *unos pocos* doblados.

Observación 3.4.5. Así como una isogenia puede comportarse como un endomorfismo, un endomorfismo puede comportarse a modo de una isogenia. Un endomorfismo puede escribirse de manera explícita en forma de funciones racionales. Ver definición 3.2.7

Notación 3.4.6. La notación ϕ^n indica la composición de el morfismo ϕ consigo mismo $n-1$ veces.

Definición 3.4.7. El polinomio característico de un endomorfismo ϕ es el polinomio $\chi_\phi(X)$ mónico y de menor grado en $\mathbb{Z}[X]$ tal que, $\chi(\phi(P)) = \mathcal{O}$ para todo punto $P \in E$.

Ejemplo 3.4.8.

- Sea E una curva elíptica sobre $\mathbb{F}_q$. Definimos el endomorfismo *negación* por

  $$\begin{aligned} \phi : E &\to E \\ \phi(P) &\mapsto -P. \end{aligned}$$

  Este es un caso particular de la multiplicación por m tomando $m = -1$. El polinomio característico de este endomorfismo es $X - m$.

- Sea E una curva elíptica definida sobre F_q y $P = (x, y) \in E$. Definimos el endomorfismo de *Frobenius* por

  $$\phi(P) = \phi((x, y)) \mapsto (x^q, y^q), \qquad \phi(\mathcal{O}) \mapsto \mathcal{O}.$$

El polinomio característico del morfismo de Frobenius es $X^2 - tX + q$ donde, $t = q + 1 - E(\mathbb{F}_q)$.

Observación 3.4.9. Como se mencionó en la sección 3.3, en criptografía de curvas elípticas, trabajamos con un subgrupo cíclico, $\langle P\rangle < E(\mathbb{F}_q)$ de orden r, donde $E(\mathbb{F}_q) = rh$. Si tenemos un endomorfismo ϕ, este satisface que $\phi(P) \in \langle P\rangle$. De aquí , podemos concluir que $\phi(P) = \lambda P$ para alguna $\lambda \in \{1, 2, 3, \ldots, r-1\}$. El entero λ es una raíz del polinomio característico de ϕ definido en $\mathbb{Z}_r[X]$.

Ejemplo 3.4.10. Sea $q = 2^m$ y $E : y^2 + xy = x^3 + ax^2 + b$ una curva elíptica definida sobre $\mathbb{F}_{q^2}$ tal que $a \in \mathbb{F}_{q^2}$ y $b \in \mathbb{F}_q$. Dado un punto $P = (x, y) \in E$, definimos el endomorfismo $\phi : E \to E$ por

$$\phi(P) = \phi((x, y)) = (x^q, y^q + ux^q).$$

Donde u es tal que $u^2 = u + 1$ como en el ejemplo 3.1.2. El polinomio característico de ϕ es $X^2 + 1$. Si tomamos $m = 7, a = u$ y $b = t^2$ obtenemos una curva de orden $E(\mathbb{F}_{q^2}) = 16130 = 2 \cdot 5 \cdot 1613$. Tomamos un punto P de orden $r = 1613$. Las raíces del polinomio $X^2 + 1 \in \mathbb{Z}_r$ son 127 y 1486. Podemos verificar que $\phi(P) = 1486P$ para todo punto P de orden 1613.

3.4.1. Aplicación a la multiplicación escalar

Se ha mencionado que un endomorfismo ϕ tiene un equivalente entero esto es que $\phi(P) = \lambda P$ para algún entero $\lambda \in \{1, \ldots, r\}$. Haciendo uso de este hecho podemos acelerar de cierta forma la multiplicación escalar. La estrategia consiste en expresar el escalar k como una combinación lineal de 1 y λ, es decir, $k = k_0 + k_1\lambda(\ \mathrm{mód}\ r)$. La complejidad de los algoritmos 3.3.1 y 3.3.3 está dada en términos de la longitud de la representación binaria y NAF respectívamente. Digamos que la representación binaria de k tiene longitud t entonces, es deseable que la representación binaria de los escalares k_0 y k_1 tenga longitud, aproximadamente, $\frac{t}{2}$. Si expresamos a k de esta manera entonces podemos calcular kP como

$$kP = k_0P + k_1\phi(P).$$

Si utilizamos el algoritmo 3.3.4 para calcular $k_0P + k_1\phi(P)$ entonces se requieren t sumas y $\frac{t}{2}$ doblados. Se obtiene una aceleración en comparación con el algoritmo 3.3.3 que requiere la misma cantidad de sumas pero t doblados.

Observación 3.4.11. Este método para acelerar la multiplicación escalar fue propuesto por Gallant, Lambert y Vanstone en el 2001[10] y es conocido como método *GLV*.

3.4.2. Descomposición escalar

En la sección anterior se dijo que, de ser posible expresar el escalar k como una combinación lineal de 1 y λ entonces, se podría acelerar la multiplicación escalar. Sin embargo no se mencionó cómo calcular los enteros que satisfacen esa combinación lineal. A continuación se presenta una técnica que nos permite resolver este problema.

Definición 3.4.12. El problema de expresar un entero como combinación lineal de n enteros λ_i, para $i \in \{0, 2, \ldots, n-1\}$, es llamado *descomposición escalar*. El problema inverso es llamado *recomposición escalar*. La recomposición escalar consiste en que dados los enteros λ_i se escogen de manera *aleatoria* n enteros a_i y calculamos el entero k de la siguiente manera:

$$k = \sum_{i=0}^{n-1} a_i \lambda_i.$$

Notación 3.4.13. Existe una ambigüedad al momento de tratar de diferenciar una descomposición escalar en dos enteros, por ejemplo, de una de tres enteros. Para indicar en número de enteros en que se descompone un escalar escribiremos que se tiene una *2-descomposición* en el caso de dos enteros y una *n-descomposición* en el caso de n enteros.

Definición 3.4.14. Decimos que una n-descomposión es *balanceada* si los enteros a_i tales que $k = \sum_{i=0}^{n-1} a_i \lambda_i$ satisfacen que

$$|a_0| \approx |a_1| \approx \cdots \approx |a_{n-1}|.$$

En este caso el operador $|\cdot|$ indica la longitud de alguna representación del entero, por ejemplo binaria o NAF.

El algoritmo 3.4.1 encuentra una 2-descomposición balanceada para un escalar k con base en un entero λ y un primo r. La complejidad de este algoritmo es igual a la complejidad del algoritmo de Euclides ya que la parte principal es el algoritmo extendido de Euclides y la parte restante consiste en un número constante de pasos.

Ejemplo 3.4.15. Consideremos la curva y los parámetros del ejemplo 3.4.10. Tenemos que $E(\mathbb{F}_{q^2}) = 16130 = 2 \cdot 5 \cdot 1613$ donde $q = 2^7$. Consideramos $r = 1613$ y tenemos que para el endomorfismo definido en ese ejemplo se tiene que $\phi(P) = \lambda P = 1486P$ para todo punto $P \in E(\mathbb{F}_{q^2})$ de orden 1613. Aplicamos el algoritmo 3.4.1 a los enteros $k = 575$ y r.

Algoritmo 3.4.1: 2-Descomposición escalar balanceada (algoritmo 3.74 de [13])

Entrada: Enteros positivos k, r y $\lambda \in \{1, 2, \ldots, \}$
Salida: Enteros k_0, k_1 tales que, $k = k_0 + k_1(\text{ mód }) r$ y
$|k_0| \approx |k_1| \approx \sqrt{r}$
inicio
Aplicar el algoritmo extendido de Euclides (algoritmo A.1.1) a los enteros k y r. El algoritmo genera una serie de ecuaciones $ax_i + by_i = u = n_i$ y $ax_{i+1} + by_{i+1} = v = n_{i+1}$. Sea l el mayor índice tal que $n_l \geq \sqrt{r}$.
$(a_1, b_1) \leftarrow (n_{l+1}, -x_{l+1})$.
si $(n_l^2 + x_l^2) \leq (n_{l+2}^2 + x_{l+2}^2)$ **entonces**
$(a_2, b_2) \leftarrow (r_l, -x_l)$
en otro caso
$(a_2, b_2) \leftarrow (r_{l+2}, -x_{l+2})$
$c_1 \leftarrow \lfloor b_2 k/r \rceil,\ c_2 \leftarrow \lfloor -b1k/r \rceil$.
$k_1 \leftarrow k - c_1 a_1 - c2_a 2,\ k_2 \leftarrow -c_1 b_1 - c_2 b_2$.
devolver (k_1, k_2)

- Obtenemos los vectores $(a_1, b_1) = (38, 13)$ y $(a_2, b_2) = (13, -38)$.
- Y los escalares $k_1 = -22$ y $k_2 = 8$.

Es fácil comprobar que $k = k_1 + k_2\lambda(\text{mód } r)$, simplemente sustituimos y tenemos que

$$-22 + 8(1486) = -22 + 11888 = 11866 \qquad y,$$
$$11866 \equiv 575(\text{mód } r).$$

3.4.3. Curvas de Koblitz

Las curvas de Koblitz (definición 3.1.4) conocidas como *curvas anómalas* fueron introducidas en 1992 por por Neal Koblitz [19]. En su artículo Koblitz menciona que estas curvas tienen buenas propiedades criptográficas. A continuación se describe brevemente cómo es posible usar el morfismo de Frobenius (ejemplo 3.4.8) para acelerar la multiplicación escalar en estas curvas.

Definición 3.4.16. El endomorfismo de Frobenius para una curva de Koblitz está dado por

$$\tau : E \to E$$
$$\tau((x, y)) \mapsto (x^2, y^2)$$

Notación 3.4.17. Es común encontrar en la literatura que, el morfismo de Frobenius en curvas de Koblitz se representa con la letra griega *tau* (τ). En lo que resta del capítulo se mantendrá esa notación para efectos del morfismo en sí, como un escalar y como variable en el polinomio característico.

Teorema 3.4.18. *Sea $\tau^2 + \mu\tau - 2$ el polinomio característico del endomorfismo de Frobenius, $\mu = q + 1 - E(\mathbb{F}_q)$. El valor de μ es conocido como la traza de Frobenius.*

En el caso particular de una curva de Koblitz , el polinomio característico del morfismo de Frobenius es $\tau^2 + \mu\tau - 2$ donde $\mu = (-1)^{1-a}$. Conociendo esto podemos utilizar el algoritmo 3.4.1 para obtener una representación de un entero k como $k_0 + k_1\tau \in \mathbb{Z}[\tau]$.

Definición 3.4.19. Una representación NAF *τ-ádica* o *τ-NAF* de un elemento $0 \neq k \in \mathbb{Z}[\tau]$ es una expresión

$$k = \sum_{i=0}^{l-1} u_i \tau^i \tag{3.4.1}$$

donde cada $u_i \in \{0, \pm 1\}$, $u_{l-1} \neq 0$ y no hay dos dígitos consecutivos que sean diferentes de cero. La longitud de la representación τ-NAF es l.

El algoritmo 3.4.2 calcula la representación $\tau - NAF$ de un elemento en $\mathbb{Z}[\tau]/(^2+\mu\tau-2)$. Usando el algoritmo 3.4.1 para obtener una representación de un entero k como $k_0 + k_1\tau \in \mathbb{Z}[\tau]$ tenemos que la longitud de τ-NAF(k) es aproximadamente $m = Log_2(k)$. Sin embargo la densidad de ceros en τ-NAF(k) es aproximadamente 1/3 [15] entonces el algoritmo 3.4.3 calcula kP realizando aproximadamente

$$\frac{m}{3}A.$$

La aceleración de la multiplicación escalar en estas curvas descansa principalmente en el cambio realizado en el algoritmo 3.4.3 con respecto al algoritmo 3.3.3. Este cambio es el que se observa en la línea 6 del algoritmo 3.4.3, se cambia la operación $2P$ por $\tau(P)$. De la ecuación 3.4.1 podemos ver

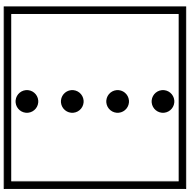
Algoritmo 3.4.2: τ-NAF [29]

Entrada: $k = r_0 + r_1\tau$
Salida: τ-NAF(k)
inicio
  $i \leftarrow 0$
  mientras $r_0 \neq 0$ *o* $r_1 \neq 0$ **hacer**
    si r_0 *es impar* **entonces**
      $u_i \leftarrow 2 - (r_0 - 2r_1(\ \text{mód}\ 4)),\ r_0 \leftarrow r_0 - u_i$
    en otro caso
      $u_i \leftarrow 0$
    $t \leftarrow r_0,\ r_0 \leftarrow r_1 + \mu r_0/2,\ r_1 \leftarrow -t/2.\ i \leftarrow i + 1.$
  devolver $(u_{i-1}, u_{i-2}, \ldots, u_1, u_0)$

que el endomorfismo de Frobenius requiere de calcular x^2 e y^2. Estas operaciones en campos binarios son muy eficientes en comparación con el costo que requiere la operación de doblado, es por esto que sólo consideramos el número de sumas al mencionar los cálculos que realiza este algoritmo.

Algoritmo 3.4.3: Multiplicación escalar en base τ

Entrada: Integer k, point $P \in E$
1 **inicio**
2 $U \leftarrow \tau\text{-}NAF(k)$
3 $Q \leftarrow \mathcal{O}$
4 $P1 \leftarrow -P$
5 **para** $i = 0, 1, \ldots, \#U - 1$ **hacer**
6 $Q \leftarrow \tau(Q)$
7 **si** $U[i] = 1$ **entonces**
8 $Q \leftarrow Q + P$
9 **si** $U[i] = -1$ **entonces**
10 $Q \leftarrow Q + P1$
11 **devolver** Q



Capítulo 4

Análisis y desarrollo

> Existe más de una manera de ganar un juego. Se puede ganar sin siquiera pelear.
>
> *Sora*
> *-No game no life.*

El uso de curvas elípticas como base para protocolos de seguridad ha ido aumentando en estos últimos años. Las ventajas que ofrecen las curvas elípticas en la criptografía de llave pública frente a otros esquemas de cifrado como puede ser *RSA* son principalmente dos: menor tamaño en las llaves y, mejor seguridad con llaves cortas. Esto hace *fácil* poder compartir llaves en dispositivos de poco almacenamiento. De igual forma el número de operaciones necesarias por algún protocolo puede verse reducido al tener una llave más corta. Sin embargo como se mencionó en el capítulo 3, las operaciones en las curvas elípticas como la suma y el doblado requieren de varias operaciones en el campo base. Esto último podría ser perjudicial al momento de realizar una implementación y hasta cierto punto se podría pensar que

este inconveniente contrasta con el tamaño de la llave, es decir que, a pesar de tener una llave corta, las operaciones son costosas, computacionalmente hablando.

Dada la popularidad que tienen estas curvas que se utilizan, por ejemplo en *Bitcoin*, certificados digitales o incluso, aplicaciones móviles como *Whatsapp* o *Signal*, es necesario poder subsanar de cierta forma el costo de las operaciones o, en otro caso, reducir el número de operaciones requeridas para el cálculo de la multiplicación escalar que, es la base de muchos protocolos basados en curvas elípticas. A lo largo de las secciónes 3.3 y 3.4, se muestran unas cuantas técnicas relacionadas con la multiplicación escalar en curvas elípticas. De manera concisa estas técnicas aceleran la multiplicación escalar. Como se pudo observar, algunas técnicas consisten en ontener una *buena* representación del escalar, como la representación NAF. Otra técnica que se mencionó es la técnica *GLV* que hace uso de endomorfismos eficientes en una curva elíptica para descomponer un escalar de de longitud n en bits, en dos escalares de longitud aproximada $\frac{n}{2}$ y posteriormente utilizar el truco de Shamir para así reducir el número de doblados necesario.

En este capítulo se exponen dos enfoques en la resolución de este problema. En primer lugar se analizan las denominadas curvas **GLS** binarias que, son tales que se puede aplicar el método *GLV* en ellas. Sin embargo, hasta el momento sólo se tiene una descomposición en dos, usando un endomorfismo. Como primer propuesta de solución se pretende extender la descomposición en dos, a una descomposición en 3 usando un segundo endomorfismo. Se presenta el método a seguir para la construcción de este endomorfismo y su análisis con respecto a la *eficiencia* que puede ofrecer. En segundo lugar se analizan las curvas de Koblitz pero en el campo $\mathbb{F}_4$ y sus extensiones. En la sección 3.4.3 del capítulo anterior se expone de manera breve una forma de acelerar la multiplicación escalar en curvas de Koblitz sobre $\mathbb{F}_2$ y sus extensiones usando el endomorfismo de Frobenius. El endomorfismo de Frobenius es eficientemente calculable en $\mathbb{F}_2$, sin embargo, en $\mathbb{F}_4$ se requieren cálculos extras por lo que es posible que deje de ser tan eficiente. Como propuesta se tiene la siguiente: construir un nuevo endomorfismo con las propiedades del endomorfismo de Frobenius que pueda suplir a este endomorfismo en el caso de $\mathbb{F}_4$. Para finalizar este capítulo se analiza una posible vulnerabilidad que surge en el proceso de recomposición al tener más de un endomorfismo.

4.1. Convenciones y notación

A lo largo de este capítulo haremos uso de ciertos campos y endomorfismos. A continuación se presenta la estructura de estos campos y la notación de los endomorfismos que se utilizan posteriormente.

- Los elementos de los campos finitos binarios $\mathbb{F}_{2^m}$ los tomaremos como polinomios de grado a lo más $m-1$ en la variable t.
- Usaremos la letra q para referirnos a 2^m.
- Los elementos de una extensión cuadrática los denotaremos de la forma $a+bu \in \mathbb{F}_{2^2m}$ donde $a, b \in \mathbb{F}_{2^m}$ y u es una raíz del polinomio x^2+x+1.
- El morfismo de Frobenius para la sección de curvas GLS binarias se representa con la letra griega *pi* (π) y en la sección de curvas de Koblitz con la letra griega *tau* (τ). En caso de ser necesario especificar la potencia a la que se elevan los elementos del campo, lo denotaremos por $\pi^{(n)}$. Por ejemplo $\pi^{(2)}$ es equivalente a $\pi((x,y)) \mapsto (x^2, y^2)$.
- El método GLV originalmente hace uso de un endomorfismo, sin embargo, con base en la descomposición escalar, es posible extenderse a más endomorfismos. En caso de tener una n-descomposición decimos que tenemos un n-GLV. Con esta notación el GLV presentado en la sección 3.4 es un 2-GLV.

4.2. Curvas GLS binarias

Las curvas GLS binarias ofrecen rapidez en la multiplicación escalar al contar con un endomorfismo muy eficiente. Estas curvas fueron propuestas en 2009 por Steven D. Galbraith, Xibin Lin y Michael Scott [9] y se pensaron para campos $\mathbb{F}_{p^2}$ para un primo $p > 3$. Sin embargo en el mismo año Darrel Hankerson, Koray Karabina y Alfred Menezes muestran que es posible adaptar estas curvas a campos binarios [12] . A continuación se muestra la construcción de la curva y del endomorfismo por parte de Hankerson, Karabina y Menezes. Posteriormente se muestra una alternativa de esta construcción. En el ejemplo 3.4.10 se menciona un endomorfismo para una cierta curva binaria, ahora la retomaremos de manera formal.

Definición 4.2.1. Una *curva GLS binaria* es una curva elíptica binaria

$$E : y^2 + xy = x^3 + ax^2 + b \tag{4.2.1}$$

Donde $a \in \mathbb{F}_{q^2}$ y $b \in \mathbb{F}_q$

Notación 4.2.2. Denotaremos por $\tilde{\text{Tr}}$ a la función traza de $\mathbb{F}_{q^2}$ en $\mathbb{F}_2$ dada por

$$\tilde{\text{Tr}}(a) = \sum_{i=0}^{2m-1} a^{2^i}.$$

De acuerdo al artículo de Hankerson *et al.* es posible construir un endomorfismo en una curva GLS E' de la siguiente manera:

- Considerar la curva $E(\mathbb{F}_q) : y^2 + xy = x^3 + ax + b$.
- Sea $\tilde{a} \in \mathbb{F}_{2^q}$ tal que $\tilde{\text{Tr}}(\tilde{a}) = 1$.
- Construimos la curva $\tilde{E}(\mathbb{F}_q) : y^2 + xy = x^3 + \tilde{a}x^2 + b$.
- E y $\tilde{E}$ son isomorfas en $\mathbb{F}_{q^4}$. Del teorema 3.1.7 tenemos que existe un isomorfismo ϕ entre E y $\tilde{E}$ dado por $\phi((x, y)) \mapsto (x, y + sx)$ donde $s \in \mathbb{F}_{q^4}/\mathbb{F}_{q^2}$ es tal que $s^2 + s = a + \tilde{a}$. Aplicamos este isomorfismo a E.
- Aplicamos el endomorfismo de Frobenius en E.
- Aplicamos de nuevo el isomorfismo ϕ, ahora en $\tilde{E}$.

En la figura 4.1 se puede observar el diagrama de esta construcción. De manera explicita tenemos el endomorfismo

$$\begin{aligned}\psi((x, y)) = \; & \phi(\pi(\phi((x, y)))) \\ & \phi(\pi((x, y + sx))) \\ & \phi((x^q, y^q + s^q x^q)) \\ & (x^q, y^q + s^q x^q + s x^q) \\ & (x^q, y^q + (s^q + s)x^q)\end{aligned}$$

Observación 4.2.3. Se dice que E es la *torcedura cuadrática* (*quadratic twist* en inglés) de E sobre $\mathbb{F}_{q^2}$.

4.2.1. Nuestra construcción

Ahora proponemos una construcción más secilla para este tipo de curvas y del endomorfismo.

- Tomamos un elemento $a \in \mathbb{F}_{q^2}$ tal que $\tilde{\text{Tr}}(a) = 1$ y un elemento $b \in \mathbb{F}_q$.
- Construimos la curva $E : y^2 + xy = x^3 + ax^2 + b$.

•••

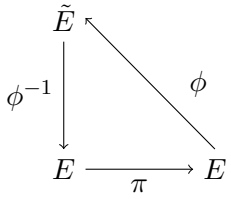


Figura 4.1: Diagrama de construcción de $\psi = \phi(\pi(\phi^{-1}))$.

- Aplicamos el mapa de Frobenius π^q a E. Esto consiste en elevar a la q los coeficientes de la curva.
- Obtenemos la curva $E^{(q)} : y^2 + xy = x^3 + a^q x^2 + b$.
- Si se cumple que $\tilde{\mathrm{Tr}}(a) = \tilde{\mathrm{Tr}}(a^q)$ entonces $E \cong E^{(q)}$. En efecto, $\tilde{\mathrm{Tr}}(a) = \tilde{\mathrm{Tr}}(a^q)$.
- Del teorema 3.1.7 tenemos que existe un isomorfismo $\phi : E^{(q)} \to E$ definido por $\phi((x, y)) = (x, y + sx)$ para una s tal que $s^2 + s = a + \tilde{a}$.
- Componemos los morfismos π y ϕ y obtenemos $\psi = \phi(\pi)) : E \to E$ definido por $\psi((x, y)) = (x^q, y^q + sx^q)$.

Demostración. Veamos que $\tilde{\mathrm{Tr}}(a) = \tilde{\mathrm{Tr}}(a^q)$. Primero observemos que si $a \in \mathbb{F}_q$ entonces $\tilde{\mathrm{Tr}}(a) = 0$. Tenemos que

$$\begin{aligned}\tilde{\mathrm{Tr}}(a) &= \sum_{i=0}^{2m-1} a^{2^i} \\ &= \sum_{i=0}^{m-1} a^{2^i} + \sum_{i=m}^{2m-1} a^{2^i}. \\ &= \mathrm{Tr}(a) + \sum_{i=m}^{2m-1} a^{2^i}\end{aligned}$$

Dado que $a \in \mathbb{F}_q$, entonces $a^q = a^{2^m} = a = a^{2^0}$. Entonces $a^{2^{m+1}} = a^{2^m} \cdot 2 = a^{2^1}$. Por inducción tenemos que $a^{2^{m+n}} = a^{2^n}$. De aquí tenemos que

$$\begin{aligned}\tilde{\mathrm{Tr}}(a) &= \mathrm{Tr}(a) + \sum_{i=0}^{2m-1} a^{2^i} \\ &= \mathrm{Tr}(a) + \mathrm{Tr}(a) \\ &= 0.\end{aligned}$$

...
.

Dado que si $a \in \mathbb{F}_q$ entonces $a^q = a$ concluimos que $\tilde{\mathrm{Tr}}(a) = \tilde{\mathrm{Tr}}(a^q)$ Ahora analizamos el caso cuando $a \in \mathbb{F}_{q^2}$. Podemos escribir a a como $a = b + cu$. Calculamos $\tilde{\mathrm{Tr}}(a)$

$$\begin{aligned}\tilde{\mathrm{Tr}}(a) &= \sum_{i=0}^{2m-1} (b+cu)^{2^i} \\ &= \sum_{i=0}^{2m-1} (b)^{2^i} + \sum_{i=0}^{2m-1} (cu)^{2^i} = \qquad \tilde{\mathrm{Tr}}(b) + \mathrm{Tr}(cu) \\ &= \tilde{\mathrm{Tr}}(cu).\end{aligned}$$

Ahora calculamos $\tilde{\mathrm{Tr}}(a^q)$. Considerando la aritmética de los campos binarios tenemos que $a^q = (b+cu)^q = b^q + (cu)^q = b + c(u+1) = (b+c) + cu$ entonces

$$\begin{aligned}\tilde{\mathrm{Tr}}(a^q) &= \sum_{i=0}^{2m-1} ((b+c)+cu)^{2^i} \\ &= \sum_{i=0}^{2m-1} (b+c)^{2^i} + \sum_{i=0}^{2m-1} (cu)^{2^i} = \qquad \tilde{\mathrm{Tr}}(b+c) + \mathrm{Tr}(cu) \\ &= \tilde{\mathrm{Tr}}(cu).\end{aligned}$$

De aquí tenemos que $\tilde{\mathrm{Tr}}(a) = \tilde{\mathrm{Tr}}(a^q)$. Por lo tanto E es isomorfa a $E^{(q)}$ y la construcción es correcta.

□

Corolario 4.2.4. *Sea $z = a + bu \in \mathbb{F}_{q^2}$ entonces, $\bar{Tr}(z) = \bar{Tr}(bu)$.*

En la figura 4.2 se puede observar el diagrama que describe al endomorfismo $\psi = \phi(\pi)$. A diferencia de la construcción propuesta por Hankerson *et al.* nosotros empezamos con una curva en $\mathbb{F}_{q^2}$ y sólo hacemos uso de la composición de dos morfismos.

4.2.2. El polinomio característico

El polinomio característico de la función ψ es $X^2 + 1$. Usaremos nuestra construcción para probar que en efecto, este es el polinomio característico de ψ.

•••

••

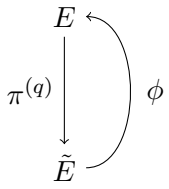


Figura 4.2: Diagrama de construcción de $\psi = \phi(\pi)$.

Demostración. Calculemos ψ^2.

$$\begin{aligned}\psi^2((x,y)) &= ((x^q)^q, (y^q + sx^q)^q + s(x^q)^q) \\ &= (x^{q^2}, y^{q^2} + s^q x^{q^2} + sx^{q^2}) \\ &= (x, y + (s^q + s)x).\end{aligned}$$

Es conocido que en una curva elíptica binaria si $P = (x, y)$, entonces $-P = (x, y + x)$. Con base en esto sólo necesitamos probar que $s^q + s = 1$.
Tenemos que $s \in \mathbb{F}_{q^2}$, entonces $s = r + tu$. Sabemos que $s^2 + s = a + a^q$ y que $a + a^q = b$ para $a = b + cu$. De aquí podemos ver que

$$\begin{aligned}b &= s^2 + s \\ &= ((r+t)^2 + t^2 u) + (r + tu) \\ &= (r^2 + t^2 + r) + (t^2 + t)u.\end{aligned}$$

Dado que $b \in \mathbb{F}_q$ tenemos que $t^2 + t$ necesariamente es cero. Las raíces del polinomio $t^2 + t$ en $\mathbb{F}_q$ son 1 y 0. Como $\tilde{\mathrm{Tr}}(a) = 1$ entonces $t \neq 0$. De aquí obtenemos que $t = 1$. Con base en lo anterior tenemos que $s^q + s = 1$. Por lo tanto, $\psi^2(P) = -P$. □

Notación 4.2.5. Nos referiremos al endomorfismo ψ como *endomorfismo GLS*.

4.2.3. Un nuevo endomorfismo

Hasta ahora contamos con unas curvas binarias en las cuales es posible definir un endomorfismo eficiente. Con base en esta configuración es posible utilizar el método 2GLV. Parte del propósito de esta tesis es el de encontrar endomorfismos eficientes en curvas binarias para acelerar la multiplicación escalar. En el año 2013 Benjamin Smith [28] propone el uso de una familia

...

...

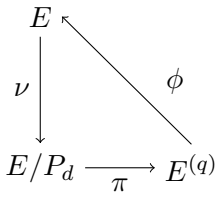


Figura 4.3: Diagrama de construcción de $\varphi = \phi(\pi(\nu))$.

de curvas en las cuales se pueden definir endomorfismos eficientes usando la fórmula de Vélu y propiedades de las curvas elípticas definidas sobre los racionales. Estas curvas fueron expuestas en 1997 por Yuji Hasegawa [14] y son conocidas como $\mathbb{Q}$-curvas. En 2015 Craig Costello y Patrick Longa[6] realizaron una implementación mezclando el método de Smith para $\mathbb{Q}$-curvas, con el endomorfismo GLS para obtener un 4-GLV usando la estrategia que proponen Longa y Francisco Sica [21]. A continuación se presenta una adaptación de la construcción que propone Smith al caso binario.

La construcción

Teorema 4.2.6. *Tomemos una curva GLS $E : y^2 + xy = x^3 + ax^2 + b$. Existe un endomorfismo en E dado por la siguiente construcción:*

- *Obtener un punto de d-torsión en E, $d \not\equiv 0 (mód\ 2)$. Digamos P_d.*
- *Construir la isogenia ν dada por la fórmula de Vélu.*
- *Construir la curva E/P_d dada por la fórmula de Vélu.*
- *Construir la curva $E^{(q)}$ obtenida al aplicar el mapa de Frobenius a E/P_d.*
- *Obtener una isogenia $\phi : E^{(q)} \to E$.*
- *Componer las isogenias ν, π y ϕ para obtener un endomorfismo de E.*

Afirmamos que la isogenia ϕ existe.

La figura 4.3 nos ofrece el diagrama que muestra la construcción de el endomorfismo $\varphi = \phi(\pi(\nu))$.

Demostración. La figura 4.4 nos muestra el diagrama de los morfismos que usaremos para probar esta afirmación. De la construcción del endomorfismo GLS tenemos que el morfismo ρ existe y más aún, $\rho(\pi)$ coincide con el endomorfismo GLS en la curva E/P_d. La isogenia $\hat{\nu}$ es la isogenia dual de ν. De aquí podemos considerar $\phi = \hat{\nu}(\rho)$ lo cuál prueba que existe una isogenia de $E^{(q)}$ a E. □

•••

••••

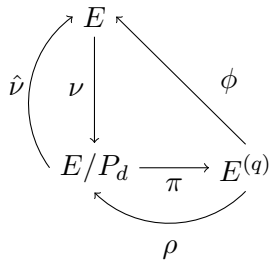


Figura 4.4: Diagrama extendido de la construcción de $\varphi = \phi(\pi(\nu))$.

Teorema 4.2.7. *El polinomio característico de φ_d es $X^2 + d^2$.*

Demostración. Sabemos que una isogenia Φ de grado d satisface que $\hat{\Phi}(\Phi) = [d]$, donde $\hat{\Phi}$ es la isogenia dual de Φ. Además tenemos que $[d]^2 = [d^2]$ y que $\psi^2(P) = -P$ para cualquier punto $P \in E$ donde ψ es el endomorfismo GLS. De igual forma nos apoyaremos de la figura 4.4 para observar el comportamiento de las isogenias.
Para probar que el polinomio característico de φ_d es $X^2 + d^2$ procedemos a expresar φ_d^2 como la composición al cuadrado de estas isogenias. Por cuestión visual usaremos la notación $f \circ g$ para referirnos a la composición $f(g)$.

$$\begin{aligned}\varphi_d^2 &= (\hat{\nu} \circ \rho \circ \pi \circ \nu)^2 \\ &= \hat{\nu} \circ \rho \circ \pi \circ \nu \circ \hat{\nu} \circ \rho \circ \pi \circ \nu\end{aligned}$$

Denotaremos por $[\bar{d}]$ al endomorfismo $[d] : E/P_d \to E/P_d$ para diferenciarlo del endomorfismo $[d] : E \to E$. Recordemos que $\hat{\nu} \circ \nu = [\bar{d}]$, entonces

$$\varphi_d^2 = \hat{\nu} \circ \rho \circ \pi \circ [\bar{d}] \circ \rho \circ \pi \circ \nu.$$

Además tenemos que $\rho \circ \pi$ es el endomorfismo GLS en E/P_d al que denotaremos por G. Con base en esto tenemos que

$$\varphi_d^2 = \hat{\nu} \circ G \circ [\bar{d}] \circ G \circ \nu.$$

Dado que $[d]$ y $[\bar{d}]$ consisten en multiplicar a un punto $P \in E$ y $P' \in E/P_d$ por d, respectívamente. Podemos aprovechar una propiedad de los morfismos y obtener

$$\varphi_d^2 = \hat{\nu} \circ G \circ G \circ \nu \circ [d].$$

$\bullet\bullet\bullet$

$\blacksquare$

Sabemos que el polinomio característico de G es $X^2 + 1$, esto nos dice que $G^2 = [-\bar{1}]$. De aquí obtenemos

$$\begin{aligned}\varphi_d^2 &= \hat{\nu} \circ [-\bar{1}] \circ \nu \circ [d] \\ &= \hat{\nu} \circ \nu \circ [-d] \\ &= [d] \circ [-d] \\ &= [-d^2].\end{aligned}$$

De aquí podemos concluir que $\varphi_d^2 = [-d^2]$, es decir $\varphi_d^2(P) = -d^2P$. Por lo tanto el polinomio característico de φ_d es $X^2 + d^2$.

□

Observación 4.2.8. En realidad tenemos una familia de endomorfismos parametrizada por d. El valor de d debe ser un número impar debido a que la característica del campo en nuestro caso es 2.

Observación 4.2.9. Se podría decir que esta familia de endomorfismos es un caso general del endomorfismo GLS. Si consideramos $d = 1$ obtenemos el endomorfismo GLS. Para nuestros términos diremos que φ_d es un múltiplo del endomorfismo GLS.

Ejemplo útil

A continuación se presenta un ejemplo de este endomorfismo que posteriormente utilizaremos para extender el 2-GLV a 3-GLV y 4-GLV. Los cálculos son realizados con el conjunto de bibliotecas provisto por MAGMA[4].

Consideramos $d = 3$ y la curva GLS $E : y^2 + xy = x^3 + ux^2 + b$. La configuración del campo $\mathbb{F}_q$ y $\mathbb{F}_{q^2}$ se encuentra en el código B.1.

Del código 4.1, obtenemos el siguiente endomorfismo

$$\varphi_3(x, y) = (\hat{x}^q, \hat{y}^q + u\hat{x}^q). \tag{4.2.2}$$

...

Donde

$$\begin{aligned}\hat{x} &= \gamma_1(x^9 + bx^3 + b^2x),\\ \hat{y} &= \gamma_2\left\{y\left[x^{12} + x^{11} + b(x^8 + x^6 + x^5) + b^2(x^4 + x^3 + x^2 + b)\right]\right.\\ &\quad \left. + x^{11} + x^{10} + x^9 + b\left[x^8 + x^7 + x^6 + b^2\right]\right\}\end{aligned}$$

Definimos

$$\begin{aligned}w &= x^8 + x^6 + b^2\\ v &= x^{12} + x^{11} + x^{10} + x^9 + bx^8 + bx^6 + b^2x^4 + b^2x^3 + b^3\\ z &= (wv)^{-1}\\ \gamma_1 &= zv\\ \gamma_2 &= zw\end{aligned}$$

Observación 4.2.10. En la construcción presentada en el código 4.1 se utilizan polinomios de división en lugar de un punto de d-torsión y se considera $b = t$.

Código 4.1: Construccion de φ_3 en Magma

```
1  //Cargamos archivo de configuración
2  load config.magma;
3
4  //Creamos la curva elíptica E
5  E := EllipticCurve([F_q2|1,u,0,0,t]);
6  //Definimos el grado de la isogenia
7  d := 3
8  //Usando la fórmula de Vélu construimos la curva E_Pd
9  //y la isogenia nu
10 E_Pd, nu := IsogenyFromKernel(E, DivisionPolynomial(E, d));
11 //Elevamos a la q los coeficientes de la curva E_Pd
12 coefE_Pd_q := [c^(q) : c in Eltseq(E_Pd)];
13 //Generamos la curva Eq
14 Eq := EllipticCurve(coefE_Pd_q);
15 //Generamos el mapa de Frobenius pi
16 pi := map<E_Pd -> Eq | P :-> Eq ! [P[1]^(q), P[2]^(q), P[3]]>;
17 //Generamos el isomorfismo phi
18 phi := Isomorphism(Eq, E);
19
20 //Componemos los morfismos
21 Endo := nu*pi*phi;
```

Si consideramos $q = 2^7$ tenemos que $\#E = 2 \cdot 5 \cdot 1613 = hr$ donde el cofactor $h = 2 \cdot 5$ y $r = 1613$. Tomamos un punto $P \in E$ de orden r. Con la ayuda de

...

..

Magma podemos calcular las raíces de $X^2+9 \in \mathbb{Z}_r[X]$. Obtenemos que estas raíces son $(381, 1232)$. Podemos verificar que $\varphi_3(P) = 1613P$, lo que verifica que el polinomio característico de φ_d es $X^2 + d^2$. El código relacionado a la búsqueda de las raíces se pueden encontrar en el código B.3. En la sección 4.6 retomaremos este endomorfismo y mostraremos una forma de acelerar la multiplicación escalar usándolo.

4.3. Análisis de Seguridad

La técnica de recomposición resultante de escoger n escalares y formar un nuevo escalar usando estos escalares puede tener cierto sesgo en algunos casos y ser vulnerable ante ataques como el de Bleichenbacher[3]. Un análisis de esta recomposición es aportado por Mehdi Tibouchi *et al.*[2] en el caso de las curvas GLS primas. A continuación se presenta el análisis de seguridad en la 2-recomposición siguiendo la estrategia de Tibouchi y posteriormente el caso 3 y 4 de la recomposición.

4.3.1. 2-recomposición

De acuerdo al trabajo de Tibouchi *et al.* es suficiente con probar que la función

$$\begin{array}{rccl} F: & [0, 2^{m/2}] & \to & \mathbb{Z}/r\mathbb{Z} \\ & (k_0, k_1) & \mapsto & k_0 + \lambda k_1. \end{array}$$

es inyectiva. De serlo, se puede concluir que los escalares k_0 y k_1 se distribuyen uniformemente con lo que se garantiza seguridad. Ahora procederemos a demostrar que en efecto, F es inyectiva.

Demostración. Consideramos dos tuplas $(a, b) \neq (\bar{a}, \bar{b})$ tales que $F(a, b) = F(\bar{a}, \bar{b})$. Como se ha mencionado $a, b, \bar{a}$ y $\bar{b}$ son menores que $\sqrt{r}$. Dado que $F(a, b) = F(\bar{a}, \bar{b})$ tenemos que

$$\begin{array}{rcll} (a - \bar{a}) + \lambda(b - \bar{b}) & \equiv 0 & & (\text{mód } r) \\ (a - \bar{a})^2 - (b - \bar{b})^2 & \equiv 0 & & (\text{mód } r) \end{array}$$

$$\begin{array}{rcl} \text{Hacemos un cambio de variables} & d = a - \bar{a} \text{ y } \bar{d} & = b - \bar{b}. \end{array}$$

$$\begin{array}{rcll} d^2 - \bar{d}^2 & \equiv 0 & & (\text{mód } r) \\ (d - \bar{d})(d + \bar{d}) & \equiv 0 & & (\text{mód } r) \end{array}$$

...

...

Dado que d y $\bar{d}$ son menores que $\sqrt{r}$, entonces, $(d-\bar{d})(d+\bar{d}) < 2r$. De aquí $(d-\bar{d})(d+\bar{d}) = r$. Esto es una contradicción porque r es un primo. Por lo tanto F es inyectiva como afirmamos. □

Afirmamos que la distribución de los valores $k = k_0 + k_1\lambda$ para $(k_0, k1)$ es uniforme en $[0, \sqrt{r}]$. En efecto, F es inyectiva, por lo tanto, para cada par (k_0, k_1) existe uno y sólo un valor de K. De aquí, para cada k_0, k_1 en el intervalo $[0, \sqrt{r}]$ existen $\frac{1}{r}$ opciones. De aquí, la distribución es uniforme.

4.3.2. 3-recomposición

Ahora analizamos una 3-recomposición. Similiar a la 2-descomposición, necesitamos probar que $F_3(k_0, k_1, k_2) = k_0 + k_1\lambda_0 + k_2\lambda_1$ es inyectiva. Donde $\psi = [\lambda_0]$ y $\varphi_3 = [\lambda_1]$.

Demostración. Consideramos dos diferentes tuplas $(a, b, c), (\bar{a}, \bar{b}, \bar{c})$. Suponemos que $F(a, b, c) = F(\bar{a}, \bar{b}, \bar{c})$. Necesitamos que $a, b, c, \bar{a}, \bar{b}$ y $\bar{c}$ tengan la misma longitud en bits . Digamos que tienen longitud ℓ donde $\ell = \frac{1}{3}$ de la longitud en bits de r. Obtenemos las siguientes ecuaciones.

$$\begin{array}{rll} (a-\bar{a}) + \lambda_0(b-\bar{b}) & \equiv \lambda_1(\bar{c}-c) & (\text{mód } r) \\ ((a-\bar{a}) + \lambda_0(b-\bar{b}))^2 & \equiv 9(\bar{c}-c)^2 & (\text{mód } r) \end{array}$$

$$\begin{array}{rll} \text{Hacemos un cambio de variables} & d = a-\bar{a}\ ,\ e = b-\bar{b}. & \text{y } f = c-c \end{array}$$

$$\begin{array}{rll} d^2 + 2de - e^2 & = 9g^2 & (\text{mód r}) \end{array}$$

Analizamos la longitud en bits de ambos lados. El lado derecho tiene $2\ell + 3$ bits, esto porque multiplicar por 9 añade 3 bits y elevar al cuadrado duplica el número de bits. En el lado izquierdo tiene 2 elementos de 2ℓ bits y uno de $2\ell + 1$ bits. Ahora $(d^2 - e^2)$ tiene $2\ell - 1$ bits, sumar $2de$ incrementa a lo más en un bit. De aquí tenemos que las longitudes en bits no coinciden. Esto es una contradicción. Por lo tanto F_3 es inyectiva. □

De manera análoga al caso de una 2-descomposición, podemos concluir que bajo estas condiciones, los escalares k_i se distribuyen de manera uniforme.

...

....

4.4. 4-recomposición

Ahora analizamos el caso de una 4-recomposición. Usando el hecho que $\psi = [\lambda_0]$ y $\varphi_3 = [\lambda_1]$ tenemos que una 4-recomposición está dada por

$$k = k_0 + \lambda_0 k_1 + \lambda_{12} + \lambda_0\lambda_1 k_3 (\text{mód } r).$$

Usar φ_3 y ψ conlleva un problema. Dado que $\lambda_1 = 3\lambda_0$ entonces, $\lambda_0\lambda_1 = -3$. Con base en esto obtenemos la siguiente ecuación

$$k = (k_0 + k_3) + k_1\lambda_0 + k_2\lambda_1 (\text{mód } r).$$

Esto genera un sesgo. Usar enteros k_i de un cuarto de la longitud en bits de r reduce el total de posibles valores de K. Más aún, genera colisiones por lo que se vuelve inseguro. Afirmamos que estas colisiones existen con base en los siguientes argumentos.

- Es posible obtener a lo más r valores. Usando todos los posibles valores de k_i.
- Al ser F_3 inyectiva y contar con escalares de a lo más un cuarto de la longitud en bits de r no podemos generar r valores diferentes.
- Entonces existen colisiones.

Con base en estos argumentos podemos concluir que la distribución de las k_i para la 4-recomposición usando ψ y φ_3 no es uniforme. Por lo tanto, no existe una manera segura de elegir k como una 4-recomposición.

4.5. Curvas de Koblitz

En 1992 Neal Koblitz[19] propone unas curvas binarias llamadas *curvas anómalas*. Posteriormente estas curvas acuñaron el nombre de curvas de Koblitz (ver definición 3.1.4). En la sección 3.3 se presentó una técnica la cual hace que en estas curvas, pueda efectuarse de manera rápida. Esta técnica consiste en aprovecharse del endomorfismo de Frobenius, con base en esto crear una representación τNAF (definición 3.4.19) y aplicar el algoritmo 3.4.2. Esta técnica es desarrollada en curvas de Koblitz binarias en $\mathbb{F}_2$. Ahora analizaremos que sucede en las curvas de Koblitz definidas en $\mathbb{F}_4$.

Definición 4.5.1 (Curvas de Koblitz en $\mathbb{F}_4$ [19])**.** Una curva de Koblitz en $\mathbb{F}_4$ esta definida por la siguiente ecuación:

$$E_{a,\gamma} : \; y^2 + xy = x^3 + a\gamma x^2 + \gamma.$$

...

Donde

- $\gamma \in \mathbb{F}_4$ y satisface $\gamma^2 = \gamma + 1$.
- $a \in \{0, 1\}$.

El endomorfismo de Frobenius en estas curvas está dado por

$$\tau((x, y)) = (x^4, y^4) \tag{4.5.1}$$

y tiene polinomio característico $\tau^2 - \mu\tau + 4$ para $\mu = (-1)^a$

4.5.1. Una mejora ante Frobenius

Introducimos una definición que pretende ampliar la definición 4.5.1.

Definición 4.5.2 (Curva de Koblitz en $\mathbb{F}_4$ *generalizada*)**.** Una curva de Koblitz en $\mathbb{F}_4$ *generalizada* esta dada por la siguiente ecuación

$$E_{\gamma_0,\gamma_1} : \ y^2 + xy = x^3 + \gamma_0 x^2 + \gamma_1.$$

Donde

- $\gamma_i \in \{0, 1, u, u + 1\}$.
- $\gamma_1 \neq 0$.
- Cuando se cumple alguno de los siguientes casos se obtiene una curva de Koblitz.
  - $\gamma_0 = \gamma_1 \in \{1, u, u + 1\}$.
  - $\gamma_0 = 0$ y $\gamma_1 \in \{1, u, u + 1\}$

Observación 4.5.3. Quizá la principal diferencia entre la definición **??** y la 4.5.1 es que γ_0 y γ_1 no tienen que ser iguales. Dado que $u^4 = u, (u + 1)^4 = (u + 1), 1^4 = 1$ y $0^4 = 0$ tenemos que el endomorfismo de Frobenius, en efecto, es un endomorfismo en estas curvas.

Teorema 4.5.4. *El endomorfismo de Frobenius tiene polinomio característico* $X^2 + 3X + 4$ *en estas curvas*

Demostración. La curva E_{γ_0} definida sobre F_4 tiene 2 puntos. Del Teorema 3.4.18 tenemos que, en efecto el polinomio característico es X^2+3X+4. □

Entre estas curvas generalizadas podemos pensar en un caso particular que posee buenas propiedades, las cuales permiten construir un endomorfismo diferente a τ y que resulta ser más eficiente. Las curvas

$$E_{\gamma_0,1} : y^2 + xy = x^3 + \gamma_0 x^2 + 1. \tag{4.5.2}$$

Por simplicidad escribiremos únicamemte E_{γ_0} para referirnos a esta curva.

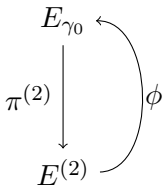


Figura 4.5: Diagrama de construcción de $\hat{\tau} = \phi(\pi^{(2)})$.

Teorema 4.5.5. *Sea $\pi^{(2)}$ el morfismo de Frobenius definido por $\pi^{(2)}((x,y)) = (x^2, y^2)$. Las curvas E_{γ_0} y $\pi^{(2)}(E_{\gamma_0}) = E^{(2)}$ son isomorfas.*

Demostración. Del teorema 3.1.7 tenemos que dos curvas son isomorfas si $\tilde{Tr}(a) = \tilde{Tr}(\bar{a})$ y $b = \bar{b}$. En nuestro caso, sabemos que $1^2 = 1$ entonces, sólo necesitamos verificar que $\tilde{Tr}(\gamma_0) = \tilde{Tr}(\gamma_0^2)$. Esto es, verificar que $\tilde{Tr}(u) = \tilde{Tr}(u^2)$ y que $\tilde{Tr}(u+1) = \tilde{Tr}((u+1)^2)$. En el caso de 1 y 0, sabemos que su traza es 0 puesto que, estan en $\mathbb{F}_q$. De la aritmética de $\mathbb{F}_{q^2}$ tenemos que $u^2 = u+1$ y que $(u+1)^2 = u$, entonces sólo basta con demostrar que $\tilde{Tr}(u+1) = \tilde{Tr}(u)$. Del corolario **??** tenemos que para $z = a + bu \in \mathbb{F}_{q^2}$ se cumple que $\tilde{Tr}(z) = \tilde{Tr}(bu)$. De aquí tenemos que $\tilde{Tr}(u+1) = \tilde{Tr}(u)$. Por lo tanto E_{γ_0} y $E^{(2)}$) son isomorfas □

Corolario 4.5.6. *Existe un endomorfismo $\hat{\tau}$ en E_{γ_0} definido por $\hat{\tau}((x,y)) = (x^2, y^2 + sx^2)$.*

Observación 4.5.7. Éste es un análogo de la construcción GLS de la sección 4.2.1.

En la figura 4.5 se puede observar el diagrama de esta construcción.

Teorema 4.5.8. *El polinomio característico de $\hat{\tau}$ es $X^2 + \hat{\mu}X + 2$. Donde $\hat{\mu} = 1$ si $\gamma_0 = u, u+1, 1$ y $\mu = -1$ si $\gamma_0 = 1, 0$.*

Demostración. Calculamos $\hat{\tau}^2$ y obtenemos que

$$\begin{aligned}\hat{\tau}^2 = &(x^4, y^4 + s^2x^4 + sx^4)\\ &(x^4, y^4 + (s^2+s)x^4).\end{aligned}$$

De aquí tenemos dos casos. Si $\gamma_0 = 0, 1$ entonces $s^2 + s = 1, 0$. Si $\gamma_0 = u, u+1, 0, 1$ entonces $s^2 + s = 1$. Con base en esto obtenemos

$$\hat{\tau}^2 = \begin{cases} \tau & \text{si } s^2 + s = 0. \\ -\tau & \text{si } s^2 + s = 1. \end{cases}$$

De aquí y del teorema 4.5.4 obtenemos la siguiente relación

$$\hat{\tau}^4 \pm 3\hat{\tau}^2 + 4 = 0.$$

Consideramos el caso cuando $\hat{\tau}^2 = -\tau$.

$$\begin{aligned} \hat{\tau}^4 - 3\hat{\tau}^2 + 4 &= 0 \\ \hat{\tau}^4 - 4\hat{\tau}^2 + 4 &= -\hat{\tau}^2 \\ (\hat{\tau}^2 - 2)^2 &= -\hat{\tau}^2 \end{aligned}$$

Usando el hecho que $\hat{\tau}$ es un endomorfismo, entonces $-\hat{\tau}^2(P) = \hat{\tau}^2(-P)$. Realizamos un cambio de variable del lado derecho reemplazando $-\hat{\tau}^2$ por $\hat{\tau}^-$ donde $\hat{\tau}^- = \hat{\tau}(-P)$.

$$\begin{aligned} (\hat{\tau}^2 - 2)^2 &= \hat{\tau}^{-2} \\ \hat{\tau}^2 - 2 &= \hat{\tau}^- \\ \hat{\tau}^2 - \hat{\tau}^- - 2 &= 0 \\ -\hat{\tau}^{-2} - \hat{\tau}^- - 2 &= 0 \\ \hat{\tau}^2 + \hat{\tau} + 2 &= 0. \end{aligned}$$

De aquí tenemos que $\hat{\tau}^2 + \hat{\tau} + 2$ es el polinomio característico de $\hat{\tau}$ para el caso $\hat{\tau}^2 = -\tau$.
Ahora analizamos el caso cuando $\hat{\tau}^2 = \tau$.

$$\begin{aligned} \hat{\tau}^4 + 3\hat{\tau}^2 + 4 &= 0 \\ \hat{\tau}^4 + 4\hat{\tau}^2 + 4 &= \hat{\tau}^2 \\ (\hat{\tau}^2 2)^2 &= +\hat{\tau}^2 \\ \hat{\tau}^2 + 2 &= \hat{\tau} \\ \hat{\tau}^2 - \hat{\tau} + 2 &= 0. \end{aligned}$$

De aquí tenemos que $\hat{\tau}^2 - \hat{\tau} + 2$ es el polinomio característico de $\hat{\tau}$ para el caso $\hat{\tau}^2 = \tau$. □

4.6. Análisis de eficiencia

A continuación se presenta un analisis de eficiencia con base en el uso de los endomorfismos propuestos. Primero se presenta el análisis respecto a usar el endomorfismo φ_d en conjunto con las curvas GLS-binarias usando una 3-descomposición y una 4-descomposición. Posteriormente se presenta el análisis de $\hat{\tau}$ en relación de τ.

Operación de campo	$\mathbb{F}_{2^{254}}$		Operación en Curva Elíptica	GLS $E(\mathbb{F}_{2^{254}})$	
	Ciclos	op/M[a]		Ciclos	op/M[a]
Multiplicación	94	1.00	Doblado(D)	450	4.79
Reducción mod.	11	0.12	Adición completa (FA)	1102	11.72
Raíz cuadrada	15	0.16	Adición mixta (MA)	812	8.64
Elevar al cuadrado	13	0.14	4-NAF	1540	16.38
Inversión	969	10.30	2-GLV-4-NAF	918	9.76

Tabla 4.1: Tiempos (en ciclos de reloj) para la arimética de campo y operaciones de curva elíptica en una plataforma Intel Sandy Bridge[25]
[a]Operaciones con respecto a la multiplicación.

Notación 4.6.1. A partir de ahora y como en la sección 3.3, estamos considerando curvas elípticas con orden *quasi-primo*, esto es que $\#E(\mathbb{F}_q) = hr$ donde r es un primo y $E(\mathbb{F}_q) \approx r$. Denotaremos por $|k| = n$ a la longitud de la representación binaria de un entero k.

4.6.1. Análisis 3-GLV y 4-GLV en $\mathbb{F}_{q^2}$ con $q = 2^{127}$

Usaremos como marco de referencia el trabajo presentado por Oliveira, López, Aranha y Rodríguez-Henríquez[25] en el que realizan una implementación usando una curva GLS en el campo $\mathbb{F}_{q^2}$ donde $q = 2^{127}$. Hasta el momento de realización de este trabajo tienen el mejor tiempo en cuanto a curvas elípticas binarias. Utilizaremos los tiempos que reportan para hacer las estimaciones. Estos tiempos se pueden ver en la tabla 4.1 tomada del artículo citado previamente.

El algoritmo 4.6.1 calcula $kP = k_0P + k_1\psi(P)$ de manera protegida. Este algoritmo es presentado en el trabajo de Oliveira *et al.* y nos servirá para realizar las estimaciones. Nos enfocaremos en el cíclo principal (línea 6). El valor de l usando un w-NAF regular[17] es de aproximadamente $\frac{n}{2(w-1)} + 1$ con $n = |k| = 254$. Con base en esto podemos estimar el total operaciones que está dado por

$$\frac{n}{2(w-1)}\left[(w-1)D + 2MA\right]. \tag{4.6.1}$$

Con base en esto y en la tabla 4.1 tenemos que el ciclo principal tomando $w = 4$ requiere 127882 ciclos de reloj.

Observación 4.6.2. El algoritmo 4.6.1 hace referencia a un "paso lineal (linear pass)". Este paso lineal es una técnica diseñada para prevenir información sensible ante ataques de "canal lateral" asociados con los patrones del

acceso a la "caché" del CPU. En este análisis no estamos considerando el costo de realizar el paso lineal.

Observación 4.6.3. En el trabajo de Oliveira *et al.* mencionan que en el algoritmo 4.6.1 es más rápido calcular ψ en cada ejecución que pre-computar la evaluación de $\psi(iP)$. El costo de calcular ψ es despreciable, es equivalente a una suma en $\mathbb{F}_{2^{254}}$.

Algoritmo 4.6.1: Multiplicación escalar protegida 2GLV

Entrada: Enteros positivos k_0, k_1, $P \in E$
Salida: $Q = k_0P + k_1\psi(P)$

inicio
  Calcular w-NAF regular de longitud l de k_0 y k_1. $R \leftarrow \mathcal{O}$
  para $i = 1, 3, \ldots, 2^{w-1} - 1$ **hacer**
    $P_i \leftarrow iP$
  $Q \leftarrow P_{k_0,l-1} + \psi(P_{k_0,l-1})$
  para $i = l-2, l-3, \ldots, 0$ **hacer**
    $Q \leftarrow 2^{w-1}Q$
    Realizar un "paso lineal" (linear pass) para recuperar $P_{k_0,i}, P_{k_1,i}$.
    $Q \leftarrow Q + P_{k_0,i} + \psi(P_{k_1,i})$
  devolver Q

Podemos modificar el algoritmo 4.6.1 para el caso de una 3 y 4 descomposición usando el endomorfismo φ_3 de la sección 4.2.3. Los algoritmos 4.6.2 y 4.6.3 muestran estas modificaciones. En este caso, considerando que los escalares tienen longitud $n/3$ y $n/4$ en el caso de 3 y 4 GLV respectivamente, la longitud del w-NAF cambia a

$$\frac{n}{3(w-1)} + 1 \qquad \frac{n}{4(w-1)} + 1$$

respectivamente. Con base en esto estimamos las operaciones en el ciclo principal de los algoritmos 4.6.2 y 4.6.3.

$$\frac{n}{3(w-1)}\left[(w-1)D + 3MA\right]. \qquad \frac{n}{4(w-1)}\left[(w-1)D + 4MA\right].$$

Las cuales en términos de ciclos de reloj indican 109794 y 101156 ciclos respectivamente. En comparación con el algoritmo 4.6.1 vemos una mejora

Algoritmo 4.6.2: Multiplicación escalar protegida 3GLV

Entrada: Enteros positivos $k_0, k_1, k_2,\ P \in E$.
Salida: $Q = k_0P + k_1\psi(P) + k_2\varphi_3(P)$.

inicio
  Calcular w-NAF regular de longitud $\bar{l}$ de k_0, k_1 y k_2. $R \leftarrow \mathcal{O}$
  $\bar{P} \leftarrow \varphi_3(P)$
  para $i = 1, 3, \ldots, 2^{w-1} - 1$ **hacer**
    $P_i \leftarrow iP$
    $\bar{P}_i \leftarrow i\bar{P}$
  $Q \leftarrow P_{k_0,\bar{l}-1} + \psi(P_{k_0,\bar{l}-1})$
  para $i = \bar{l} - 2, \bar{l} - 3, \ldots, 0$ **hacer**
    $Q \leftarrow 2^{w-1}Q$
    Realizar un "paso lineal" (linear pass) para recuperar $P_{k_0,i}, P_{k_1,i}, \bar{P}_{k_2,i}$.
    $Q \leftarrow Q + P_{k_0,i} + \psi(P_{k_1,i} + \bar{P}_{k_2,i})$
  devolver Q

Algoritmo 4.6.3: Multiplicación escalar protegida 4GLV

Entrada: Enteros positivos $k_0, k_1, k_2, k_3,\ P \in E$.
Salida: $KP = k_0P + k_1\psi(P) + k_2\varphi_3(P) + k_3\varphi_3(\psi(P))$.

inicio
  Calcular w-NAF regular de longitud $\hat{l}$ de k_0, k_1, k_2 y k_3. $R \leftarrow \mathcal{O}$
  $\bar{P} \leftarrow \varphi_3(P)$
  para $i = 1, 3, \ldots, 2^{w-1} - 1$ **hacer**
    $P_i \leftarrow iP$
    $\bar{P}_i \leftarrow i\bar{P}$
  $Q \leftarrow P_{k_0,\hat{l}-1} + \psi(P_{k_0,\hat{l}-1})$
  para $i = \hat{l} - 2, \hat{l} - 3, \ldots, 0$ **hacer**
    $Q \leftarrow 2^{w-1}Q$
    Realizar un "paso lineal" (linear pass) para recuperar $P_{k_0,i}, P_{k_1,i}, \bar{P}_{k_2,i}, \bar{P_{k_3,i}}$.
    $Q \leftarrow Q + P_{k_0,i} + \psi(P_{k_1,i} + \bar{P}_{k_2,i} + \psi(\bar{P_{k_3,i}}))$
  devolver Q

Algoritmo	Ciclos de reloj	Ganancia (en ciclos) con respecto a 2-GLV
2-GLV	127882	n/a
3-GLV	109794	18088
4-GLV	101156	26726

Tabla 4.2: Comparación entre 2GLV, 3GLV y 4GLV

Elemento	Operación				
	Suma	Multiplicación	Elevar al cuadrado	Elevar a la q	Inverso
$\hat{x}$	2	3	0	1	0
$\hat{y}$	13	5	6	1	0
φ_3	17	14	6	2	1

Tabla 4.3: Conteo de operaciones requeridas por φ_3. La fila φ_3 indica el total de las operaciones necesarias.

de aproximadamente 18000 ciclos en el caso 3-GLV y de 26700 ciclos enl 4-GLV. En la tabla 4.2 se puede observar esta comparación.

En el ciclo principal obtenemos ganancia, sin embargo no estamos considerando el costo de calcular φ_3. Con base en la tabla 4.1 y mediante un conteo de operaciones podemos estimar el costo de este endomorfismo. La tabla 4.3 muestra este conteo. El costo de calcular φ_3 es de aproximadamente 2363 ciclos, equivalente a 5.2 doblados. Si consideramos el costo de calcular $\varphi_3(P)$ y de precomputar $c\varphi_3(P)$ para $c = 3, 5, 7$, en el caso $w = 4$, tenemos aproximadamente 6149 ciclos. A pesar de ser una cantidad elevada de ciclos, la diferencia con respecto al ciclo principal del 2-GLV , aún contando este precomputo sigue superando los 11000 ciclos.

El costo total de la multiplicación escalar reportado por Oliveira *et al.* es de 114800 ciclos. Con base en esto observamos que hay un factor de reducción de aproximadamente .89 con respecto a nuestra estimación de su ciclo principal. Aplicando este factor de reducción a los ciclos que requiere nuestro ciclo principal y el costo del precomputo obtenemos una aproximación de 103189 en el caso de 3-GLV y de 95501 ciclos en el 4-GLV.

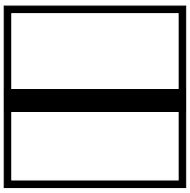
4.6.2. Curvas de Koblitz en $\mathbb{F}_4$

La tabla 4.4 muestra una comparación directa de los ciclos necesarios por τ y $\hat{\tau}$. Con base en esta tabla podemos concluir que requiere la mitad de operaciones que τ lo que en la práctica podría verse reflejado en una aceleración de aproximadamente el doble. En este caso multiplicar por u es

Operación	Endomorfismo	
	τ	$\hat{\tau}$
Elevar al cuadrado	4	2
Multiplicación por u	0	1

Tabla 4.4: Operaciones requeridas por τ y $\hat{\tau}$

equivalente a una suma en $\mathbb{F}_q$ lo que hace despreciable este costo.



Capítulo 5

Conclusiones y trabajo a futuro

5.1. Conclusiones

El objetivo principal de este trabajo era el de encontrar endomorfismos eficientes para poder agilizar el cálculo de la multiplicación escalar en curvas elípticas binarias. En estre trabajo se hizo particular enfoque en curvas GLS y curvas de Koblitz en $\mathbb{F}_4$. De manera general, con base en los resultados obtenidos, se concluye que el objetivo principal de este trabajo fué alcanzado. El motivo principal es que se encontró una familia de endomorfismos, en algunos casos eficientes, para curvas GLS y un nuevo endomorfismo en curvas de Koblitz en $\mathbb{F}_4$ los cuales permiten un realizar la multiplicación escalar más rápido utilizando un 3-GLV y un 4-GLV. Al momento de escribir esta tesis no se conocen resultados en curvas elípticas binarias relacionados con un 3-GLV y un 4-GLV.

En la sección 4.2.1 dimos una construcción alternativa a la propuesta por Hankerson *et al.*[12] y ofrecimos una demostración del polinomio característico resultante de esta construcción. Posteriormente en la sección 4.2.3 se propuso una construcción, basada en la propuesta por Benjamin Smith[28],

que generaliza el endomorfismo GLS obteniendo "múltiplos" de este endomorfismo. Estos múltiplos, representados por φ_d, pueden ser utilizados para realizar un 3-GLV y un 4-GLV, sin embargo, resultan no ser tan eficientes al ser utilizados en grados mayores que 5. En el caso de φ_3, a pesar de no ser tan eficiente en comparación con el endomorfismo GLS al ser usado en un 3-GLV y un 4-GLV se obtiene una reducción de operaciones en comparación con el método clásico GLV usando el endomorfismo GLS. Esto último no siempre sucede como puede observarse en el ejemplo C.1.1 en el que usar el método "tradicional" es igual o incluso más rápido, sin embargo en campos más grandes si se obtiene una mejora como se observa en la sección 4.6.1. En la sección mencionada obtenemos que con nuestro endomorfismo obtenemos una mejora significativa con respecto a la utilizada como marco de comparación. Por otro lado, desde el punto de vista de seguridad, en la sección 4.3 se realizó un análisis referente a la la técnica de recomposición que es suceptible a ataques. Este análisis nos permite concluir que, a pesar de que, la 4-descomposición ofrece una mejora en cuanto el tiempo necesario para calcular la multiplicación escalar, la 4-recomposición no es segura. Sin embargo una 3-recomposición resulta ser más rápida y al mismo tiempo segura.

En la sección 4.5 se realizó un analisis referente a las curvas de Koblitz en $\mathbb{F}_4$ y se definió de manera general un modelo para estas curvas. Se demostraron los nuevos resultados que surgen de esta generalización , por ejemplo el cambio en los polinomios característicos del endomorfismo de Frobenius respecto de los parámetros de la curva. En la sección 4.5.1 se propone un nuevo endomorfismo que satisface las propiedades del endomorfismo de Frobenius pero a la mitad del costo de este. Este endomorfismo permite realizar una descomposición τNAF idéntica a la de las curvas de Koblitz en $\mathbb{F}_2$. Con base en esto, tenemos que los resultados y propiedades en relación al τNaf son válidos con este nuevo endomorfismo permitiendo que las implementaciones existentes puedan utilizarse en conjunto con este endomorfismo.

5.2. Trabajo a futuro

A pesar de que los objetivos de este trabajo se cumplieron de manera satisfactoria, se han descubierto otras áreas de oportunidad adicionales que podrían complementar los resultados obtenidos. A continuación se menciona este posible trabajo

- Realizar una implementación en C para poder estimar de manera más precisa los tiempos que toma realizar la multiplicación escalar usando

••••

el endomorfismo GLS y el propuesto.

- Realizar un análisis de seguridad al realizar una n-descomposición utilizando $n-1$ endomorfismos.
- Verificar si existe una forma de adaptar algún otro método de generación de endomorfismos. Por ejemplo el propuesto en [34].
- Realizar un análisis respecto del nuevo endomorfismo en curvas de Koblitz en $\mathbb{F}_4$ para saber si es posible adaptarlo a otras extensiones como $\mathbb{F}_8$ o $\mathbb{F}_{16}$.

....
..

BIBLIOGRAFÍA

[1] Apostol, T. *CALCULUS, VOLUME I, 2ND ED.* Wiley India Pvt. Limited, 2007.

[2] Aranha, D. F., Fouque, P.-A., Gérard, B., Kammerer, J.-G., Tibouchi, M., and Zapalowicz, J.-C. *GLV/GLS Decomposition, Power Analysis, and Attacks on ECDSA Signatures with Single-Bit Nonce Bias.* Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, pp. 262–281.

[3] Bleichenbacher, D. *Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1.* Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 1–12.

[4] BOSMA, W., CANNON, J., and PLAYOUST, C. The magma algebra system i: The user language. *Journal of Symbolic Computation 24*, 3â4 (1997), 235 – 265.

[5] Cormen, T. H., Stein, C., Rivest, R. L., and Leiserson, C. E. *Introduction to Algorithms*, 2nd ed. McGraw-Hill Higher Education, 2001.

[6] Costello, C., and Longa, P. Fourq: four-dimensional decompositions on a q-curve over the mersenne prime. Cryptology ePrint Archive, Report 2015/565, 2015. `http://eprint.iacr.org/`.

[7] Elgamal, T. A public key cryptosystem and a signature scheme based on discrete logarithms. *Information Theory, IEEE Transactions on 31*, 4 (Jul 1985), 469–472.

[8] Fraleigh, J., and Katz, V. *A first course in abstract algebra.* Addison-Wesley world student series. Addison-Wesley, 2003.

[9] Galbraith, S., Lin, X., and Scott, M. Endomorphisms for faster elliptic curve cryptography on a large class of curves. *Journal of Cryptology 24*, 3 (2011), 446–469.

[10] Gallant, R., Lambert, R., and Vanstone, S. Faster point multiplication on elliptic curves with efficient endomorphisms. In *Advances in Cryptology â CRYPTO 2001*, J. Kilian, Ed., vol. 2139 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2001, pp. 190–200.

[11] Guerrero Lara, Ernesto . Pérez Terrazas, E. *Álgebra abstracta: de grupos a preliminares de la Teoria de Galois*. UADY, Mérida, Yuc., México, 2010.

[12] Hankerson, D., Karabina, K., and Menezes, A. Analyzing the galbraith-lin-scott point multiplication method for elliptic curves over binary fields. *IEEE Transactions on Computers 58*, 10 (2009), 1411–1420.

[13] Hankerson, D., Menezes, A. J., and Vanstone, S. *Guide to Elliptic Curve Cryptography*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.

[14] Hasegawa, Y. Q-curves over quadratic fields. *manuscripta mathematica 94*, 1 (1997), 347–364.

[15] Hernández, A. F. Implementación multinúcleo de la multiplicación escalar en curvas de koblitz, 2012. `http://delta.cs.cinvestav.mx/~francisco/Tesiscompleta_Faz.pdf`.

[16] Hoffstein, J., Pipher, J., and Silverman, J. *An Introduction to Mathematical Cryptography*, 1 ed. Springer Publishing Company, Incorporated, 2008.

[17] Joye, M., and Tunstall, M. *Exponent Recoding and Regular Exponentiation Algorithms*. 2009, pp. 334–349.

[18] Koblitz, N. Elliptic curve cryptosystems. *Mathematics of Computation 48*, 177 (1987), 203–209.

[19] Koblitz, N. *Advances in Cryptology — CRYPTO '91: Proceedings*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992, ch. CM-Curves with Good Cryptographic Properties, pp. 279–287.

[20] Lara Rodriguez, José Alejandro . Rubio Barrios, C. J. *Álgebra Lineal*. UADY, Mérida, Yuc., México, 2011.

[21] Longa, P., and Sica, F. Four-dimensional gallant–lambert–vanstone scalar multiplication. *Journal of Cryptology 27*, 2 (2014), 248–283.

[22] Miller, V. Use of elliptic curves in cryptography. In *Advances in Cryptology â CRYPTO â85 Proceedings*, H. Williams, Ed., vol. 218 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1986, pp. 417–426.

....

....

[23] NIST. Recommended elliptic curves for federal government use, 1999.

[24] Niven, I., and Zuckerman, H. *An Introduction to the Theory of Numbers*. Wiley Eastern, 1993.

[25] Oliveira, T., López, J., Aranha, D. F., and Rodríguez-Henríquez, F. Two is the fastest prime: lambda coordinates for binary elliptic curves. *Journal of Cryptographic Engineering 4*, 1 (2014), 3–17.

[26] Rivest, R. L., Shamir, A., and Adleman, L. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM 21*, 2 (Feb. 1978), 120–126.

[27] Silverman, J. H. *The arithmetic of elliptic curves*. Graduate texts in mathematics. Springer, New York, Berlin, 1986. 2e tirage corrigé 1992.

[28] Smith, B. Families of fast elliptic curves from q-curves. In *Advances in Cryptology - ASIACRYPT 2013*, K. Sako and P. Sarkar, Eds., vol. 8269 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 61–78.

[29] Solinas, J. A. Efficient arithmetic on koblitz curves. *Designs, Codes, and Cryptography* (2000), 195–249.

[30] Vélu, J. Isogénies entre courbes elliptiques. *C. R. Acad. Sci. Paris Sér. A-B 273* (1971), A238–A241.

[31] Washington, L. C. *Elliptic Curves: Number Theory and Cryptography, Second Edition*, 2 ed. Chapman & Hall/CRC, 2008.

[32] WhatsApp. Whatsapp encryption overview, 2016.

[33] Young, E., and Hudson, T. Openssl, 1998.

[34] Zhou, Z., Hu, Z., Xu, M., and Song, W. Efficient 3-dimensional glv method for faster point multiplication on some gls elliptic curves. *Information Processing Letters 110*, 22 (2010), 1003–1006.

[35] Zimmermann, P. *PGP Source Code and Internals*. MIT Press, Cambridge, MA, USA, 1995.

Apéndices

Apéndice A

Fundamentos matemáticos

A.1. Conjuntos

El contenido de este apéndice abreva del material disponible en [8], [1], [5] y [24].

Definición A.1.1 (Conjunto)**.** Un *conjunto* es una coleción bien definida de objetos.

- Un conjunto S esta compuesto de *elementos*. Si a es uno de estos elementos, lo denotaremos por $a \in S$.
- Existe un único conjunto el cual no contiene elementos. Este conjunto es el *conjunto vacío* y se denota por $\emptyset$.
- Es posible describir un conjunto de 2 formas, dando una propiedad característica de los elementos o listando sus elementos. La manera usual de enlistar los elementos de un conjunto es encerrandolos en llaves($\{,\}$) y separandolos por comas; por ejemplo, $\{1, 2, 3, 14\}$. Si un conjunto es descrito por una propiedad $P(x)$ para los elementos x, se preserva la notación de llaves $\{x \mid P(x)\}$ y se lee: "el conjunto de todas

las x tales que la proposición $P(x)$ referente a x es cierta". Entonces $\{2,4,6\} = \{x \mid x \text{ es un número entero, positivo y menor que } 8\}$.

- Un conjunto es *bien definido* en el sentido de que, si S es un conjunto y a un elemento, a esta definitivamente en S, denotado por $a \in S$, o a definitivamente no esta en S, denotado por $a \notin S$.

Definición A.1.2 (Subconjunto)**.** Un conjunto B es un *subconjunto* de un conjunto A, denotado por $B \subseteq A$ o $A \supseteq B$, si todo elemento de B pertenece al conjunto A. La notación $B \subset A$ o $A \supset B$ es utilizada cuando $B \subseteq A$, pero $B \neq A$.

Un hecho que nos brinda la definición A.1.2 es que, para cualquier conjunto A, existen al menos 2 subconjuntos, A mismo y $\emptyset$.

Definición A.1.3. Sea A un conjunto, entonces A es un *subconjunto impropio* de A. Cualquier otro subconjunto de A se dice que es un *subconjunto propio* de A.

Ejemplo A.1.4. Sea $S = \{1,2,3\}$. S tiene un total de 8 subconjuntos: $\emptyset, S, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}$.

Definición A.1.5 (Producto cartesiano)**.** Sean A y B conjuntos. El conjunto $A \times B = \{(a,b) \mid a \in A \text{ y } b \in B\}$, es llamado el *producto cartesiano* de A con B.

Ejemplo A.1.6. Sean $A = \{1,2,3\}$ y $B = \{3,4\}$, entonces

$$A \times B = \{(1,3),(1,4),(2,3),(2,4),(3,3),(3,4)\}$$

Observación A.1.7. El concepto de producto cartesiano puede extenderse de manera natural aplicandose a n conjuntos.

Notación A.1.8. Denotamos el producto cartesiano de un conjunto S consigo mismo como $S^2 = S \times S$. El conjunto S^n denota el producto cartesiano de S consigo mismo $n-1$ veces, esto es, $S^n = \underbrace{S \times S \times \cdots \times S}_{n-1 \text{ veces}}$

A lo largo de este documento se utilizarán ciertos conjuntos conocidos. A continuación se presenta la notación que se utilizará para cada uno de ellos.

- $\mathbb{N}\{1,2,3,\dots\}$ es el conjunto de los números naturales.
- $\mathbb{Z} = \{\dots,-3,-2,-1,0,1,2,3,\dots\}$ es el conjunto de los enteros.
- $\mathbb{Q} = \{x \mid x = \frac{m}{n}, m,n \in \mathbb{Z}, n \neq 0\}$ es el conjunto de los números racionales.

••••

- $\mathbb{R}$ es el conjunto de los números reales.
- $\mathbb{C}$ el conjunto de los números complejos.
- $\mathbb{Z}^+, \mathbb{Q}^+$ y $\mathbb{R}^+$ denotan el conjunto de los elementos positivos de $\mathbb{Z}, \mathbb{Q}$ y $\mathbb{R}$ respectívamente.
- $\mathbb{Z}^+, \mathbb{Q}^+, \mathbb{R}^+$ y $\mathbb{C}$ denotan el conjunto de los elementos no cero de $\mathbb{Z}^+, \mathbb{Q}^+, \mathbb{R}^+$ y $\mathbb{C}$ respectívamente.

Definición A.1.9 (Cardinalidad)**.** El número de elementos en un conjunto S es llamado la *cardinalidad* del conjunto S y es usualmente denotada por $\mid S \mid$ ó $\#S$.

Ejemplo A.1.10. La cardinalidad del conjunto S del ejemplo A.1.4 es 3.

Definición A.1.11 (Relación)**.** Una *relación* entre los conjuntos A y B es un subconjunto $\mathscr{R}$ de $A \times B$. Decimos que un elemento $a \in A$ esta relacionado con un elemento $b \in B$ si $(a, b) \in \mathscr{R}$. Lo denotamos por $a\mathscr{R}b$.

Definición A.1.12 (Función)**.** Una función[1] ϕ de un conjunto X a un conjunto Y es una relación entre X e Y con la propiedad de que cada $x \in X$ aparece como primer término en exáctamente un par ordenado (x, y) en ϕ. Una función es tambien llamada *mapa* o *asignación*. Se escribe como $\phi : X \to Y$ y denotamos que $(x, y) \in \phi$ como $\phi(x) = y$. El *dominio* de ϕ es el conjunto X y el conjunto Y es el *codominio* de ϕ. El *rango* de ϕ es el conjunto $\phi(X) = \{\phi(x) \mid x \in X\}$.

Ejemplo A.1.13. Podemos ver la suma de los números reales como una función $+ : (\mathbb{R} \times \mathbb{R}) \to \mathbb{R}$, esto es, como una asignación de $(\mathbb{R} \times \mathbb{R})$ en $\mathbb{R}$. Por ejemplo la acción de $+$ en $(2, 3) \in \mathbb{R} \times \mathbb{R}$ esta dada en notación de funcion por $+((2, 3)) = 5$. En notación de conjuntos escribimos $((2, 3), 5) \in +$. Usualmente escribimos $2 + 3 = 5$.

Definición A.1.14. Una función $\phi : X \to Y$ es *inyectiva(uno a uno)* si $\phi(x_1) = \phi(x_2)$ únicamente cuando $x_1 = x_2$. La función ϕ es *suprayectiva(sobreyectiva, suryectiva, subyectiva)* si el rango de ϕ es Y. Si ϕ es inyectiva y suprayectiva, decimos que ϕ es *biyectiva*.

Definición A.1.15. Sea $\phi : X \to Y$ un función inyectiva y suprayectiva. Intercambiar el primer y segundo miembro de todo par $(x, y) \in \phi$ nos proporciona un conjunto de parejas ordenadas (y, x). Tenemos un subconjunto de $Y \times X$ el cual es una función inyectiva y suprayectiva de Y a X. Esta función es llamada *función inversa* y es denotada por $\phi^{-1} : Y \to X$.

[1] El término *función* fue introducido en las matemáticas por Gottfried Wilhelm Leibniz, que utilizaba este término para designar cierto tipo de fórmulas matemáticas.

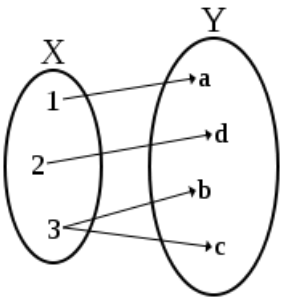


Figura A.1: Ejemplo de una función entre los conjuntos X e Y.

Definición A.1.16. Dos conjuntos tienen la misma cardinalidad si existe una función biyectiva entre ellos.

Definición A.1.17 (Composición de funciones)**.** Sean X, Y y Z conjuntos. Sean $\gamma : X \to Y$ y $\phi : Y \to Z$ funciones. Definimos la composición de funciones $\phi(\gamma) : X \to Z$ por el conjunto $W = \{\phi(y) \mid y \in \gamma(X)\} \in Z$. Esto es, aplicar la función ϕ al rango de γ.

A.1.1. Particiones y relaciones de equivalencia

Definición A.1.18 (Partición)**.** Una *partición* de un conjunto S es una colección de subconjuntos no vacios de S tales que cada elemento de S esta en exáctamente uno de estos subconjuntos. Los subconjuntos son llamados *clases*. Denotaremos por $\bar{x}$ a la clase que contiene el elemento $x \in S$.

Ejemplo A.1.19. Separando a $\mathbb{Z}^+$ en, el conjunto de los números pares y el de los impares obtenemos una partición de $\mathbb{Z}^+$ en dos clases. Por ejemplo podemos escribir

$$\bar{14} = \{2, 4, 6, 8, 10, 12, 14, 16, 18, \dots\}.$$

Definición A.1.20 (Relación de equivalencia)**.** Una relación de equivalencia $\mathscr{R}$ en un conjunto S, es una relación que satisface las siguientes tres propiedades para todo $x, y, z \in S$.

- *Reflexiva*: $x\mathscr{R}x$.
- *Simétrica*: Si $x\mathscr{R}y \Rightarrow y\mathscr{R}x$.
- *Transitiva*: Si $x\mathscr{R}y$ y, $y\mathscr{R}z$, entonces $x\mathscr{R}z$.

Teorema A.1.21 (Relaciones de equivalencia y particiones)**.** *Sea S un conjunto no vacio y sea $\sim$ una relación de equivalencia en S. Entonces $\sim$ produce una partición de S, donde*

$$\bar{a} = \{x \in S \mid x \sim a\}.$$

A.1.2. El conjunto de los enteros

Definición A.1.22. Un entero b es divisible por otro entero $a \neq 0$ si existe un entero x tal que $b = ax$. Se escribe $a|b$ y se lee a "divide a" b. Si a no divide a b lo denotamos por $a \nmid b$.

Teorema A.1.23. *Sean a, b, c enteros*

1. *$a|b$ implica que $a|bc$ para cualquier c.*
2. *$a|b$ y $b|c$ implica que $a|c$.*
3. *$a|b$ y $a|c$ implica que $a|bx + cy$ para cualesquiera enteros x, y.*
4. *$a|b$ y $b|a$ implica que $a = \pm b$*
5. *si $a|b$ y $a > 0, b > 0$ entonces $a \leq b$.*

Observación A.1.24. En el teorema anterior, el punto 3 puede extenderse a cualquier conjunto finito de enteros, esto es, si a divide a los enteros $b_1, b_2, \ldots, b_n$ entonces, $a|\sum_{i=1}^{n} b_i x_i$ para cualesquiera enteros x_i. Decimos que $\sum_{i=1}^{n} b_i x_i$ es una *combinación lineal* de las b_i.

Teorema A.1.25 (Algoritmo de la división)**.** *Dados dos enteros cualesquiera a y b, con $a > 0$, existen los enteros q y r tales que $b = qa + r$, donde $0 \leq r < a$. Si $a \nmid b$ entonces r satisface las desigualdades estrictas $0 < r < a$.*

Definición A.1.26. El entero a es *un divisor común* de b y c si $a|b$ y $a|c$. Puesto que solo existe un número finito de divisores de cualquier entero diferente de cero, solamente existen un número finito de divisores comunes de b y c, excepto en el caso $b = c = 0$. Si alguno, b o c, es diferente de cero entonces el mayor entero que es divisor común de b y c es llamado *máximo común divisor*. El máximo cumún divisor de dos enteros b, c es denotado por $mcd(b, c)$; en ocasiones cuando el contexto lo permite es sólo denotado por (b, c). El $(b, c) \geq 1$.

Teorema A.1.27. *Si g es el máximo común divisor de b y c, entonces existen los enteros y_0 y x_0 tales que $g = (b, c) = bx_0 + cy_0$.*

Definición A.1.28. Se dice que los enteros a y b son *primos relativos* si $mcd(a, b) = 1$. Esto es que el único divisor común de a y b es el 1.

Teorema A.1.29 (Algoritmo euclidiano)**.** *Dados los enteros b y $c > 0$ se hace una aplicación repetida del algoritmo de la división, para obtener una serie de ecuaciones*

$$\begin{aligned}
b &= cq_1 + r_1 && 0 < r_1 < c,\\
c &= r_1q_2 + r_2 && 0 < r_2 < r_1\\
r_1 &= r_2q_3 + r_3 && 0 < r_3 < r_2\\
&\vdots\\
r_{j-2} &= r_{j-1}q_j + r_j && 0 < r_j < r_{j-1}\\
r_{j-1} &= r_jq_{j+1}.
\end{aligned}$$

El máximo común divisor de b y c es r_j, es decir, el último residuo diferente de cero en el proceso de la división.

Observación A.1.30. El proceso del teorema anterior es finito, termina cuando la división es exacta. Dado que los residuos decrecen en cada aplicación de la división eventualmente un residuo será cero. Los valores y y x tales que $mcd(b, c) = bx + cy$ se obtienen eliminando los residuos en el conjunto de ecuaciones.

Ejemplo A.1.31. Sean $b = 963$ y $c = 657$. Aplicamos el algoritmo euclidiano en by c y obtenemos las siguientes ecuaciones:

$$\begin{aligned}
963 &= 657 \cdot 1 + 306\\
657 &= 305 \cdot 2 + 45\\
306 &= 45 \cdot 6 + 36\\
45 &= 36 \cdot 1 + 9\\
36 &= 9 \cdot 4.
\end{aligned}$$

De las ecuaciones obtenemos que $mcd(963, 657) = 9$. Más aún, 9 puede expresarse como una combinación lineal de 963 y 657 eliminando los residuos 36, 45 y 306 de la siguiente manera:

$$\begin{aligned}
9 &= 45 - 36 \cdot 1\\
&= 45 - (306 - 45 \cdot 6)\\
&= -306 + 7 \cdot 45\\
&= -306 + 7(657 - 306 \cdot 2)\\
&= 7 \cdot 657 - 15 \cdot 306\\
&= 7 \cdot 65715(963 - 657)\\
&= 22 \cdot 657 - 15 \cdot 963.
\end{aligned}$$

Algoritmo A.1.1: Euclides extendido

Entrada: Enteros positivos a, b tal que$a \leq b$
Salida: $d = mcd(a, b)$ y los enteros x, y que satisfacen $ax + by = d$.
inicio
  $u \leftarrow a, \quad v \leftarrow b.$
  $x_1 \leftarrow 1,\ y_1 \leftarrow 0,\ x_2 \leftarrow 0,\ y_2 \leftarrow 1.$
  mientras $u \neq 0$ **hacer**
    $q \leftarrow \lfloor v/u \rfloor,\ r \leftarrow v - qu,\ x \leftarrow x_2 - qx_1,\ y \leftarrow y_2 - qy_1.$
    $v \leftarrow u,\ u \leftarrow r,\ x_2 \leftarrow x_1, x_1 \leftarrow x,\ y_2 \leftarrow y_1,\ y_1 \leftarrow y.$
  $d \leftarrow v,\ x \leftarrow x_2,\ y \leftarrow y_2.$
  devolver d, x, y

El algoritmo A.1.1 nos permite obtener los valores y y x tales que si $d = mcd(a, b)$ entonces $ax + by = d$.

Definición A.1.32. Se dice que un entero $p > 1$ es un *número primo*, o simplemente que es un primo, en caso de que no exista algún divisor d de p que satisfaga $1 < d < p$. Si un entero $a > 1$ no es un primo, entonces se dice que es un número *compuesto*

Ejemplo A.1.33. Los números 2,3,5 y 7 son primos, mientras que 4,6,8 y 9 son compuestos.

Teorema A.1.34. *Sea p un primo y a y b enteros. Si $p|ab$, entonces $p|a$ o bien $p|b$.*

Teorema A.1.35. *Todo entero positivo puede expresarse como un producto de primos.*

Ejemplo A.1.36. $6 = 3 \cdot 2$, $35 = 7 \cdot 5$, $1048584 = 2^3 \cdot 3 \cdot 43691$

Definición A.1.37. Esta forma de expresar un número como un produto de primos es llamada *factorización*. Los primos que forman parte de una factorización son llamados *factores*. Así 6 se *factoriza* como $2 \cdot 3$.

Teorema A.1.38 (Teorema fundamental de la aritmética)**.** *La factorización de cualquier entero positivo n en primos es única independientemente del orden de los primos.*

Teorema A.1.39 (Euclides)**.** *El número de primos es infinito.*

Definición A.1.40. Si un entero m, diferente de cero, divide a la diferencia $a-b$, se dice que a *es congruente* con b módulo m y se escribe $a \equiv b(\text{mód } m)$. Si $m \nmid (a-b)$, entonces se dice que a no es congruente con b módulo m y se escribe $a \not\equiv b(\text{mód } m)$.

Observación A.1.41. La congruencia módulo m es una relación de equivalencia.

Definición A.1.42. Si $x \equiv y(\text{mód } m)$ entonces y recibe el nombre de *residuo de x módulo m.*

Definición A.1.43. El número $\varphi(n)$ es el número de enteros positivos menores o iguales que n y son primos relativos con n. Este número es conocido como la función indicatriz φ de Euler.

A.2. Anillos de polinomios

Definición A.2.1. Sea R un anillo. Un *polinomio $f(x)$ con coeficientes en R* es una suma formal infinita

$$\sum_{i=0}^{\infty} a_i x^i = a_0 + a_1 x + \cdots + a_n x^n + \ldots,$$

donde cada $a_i \in R$ y $a_i = 0$ excepto para un número finito de valores de i. Los elementos a_i son llamados *coeficientes* de $f(x)$. Si se cumple para alguna i que $a_i \neq 0$ entonces, el mayor valor de i que cumple esto es llamado el *grado* del polinomio $f(x)$. Si todos los coeficientes son cero, entonces el grado del polinomio está indefinido. Si $a_i \neq 0$ únicamente para $i = 0$, entonces $f(x)$ es denominado *polinomio constante.*

Observación A.2.2. Todo elemento en R es un polinomio constante.

Es posible definir una operación de suma y una multiplicación en estos polinomios.

Definición A.2.3. Sean $f(x) = a_0 + a_1 x + \cdots + a_n x^n + \ldots$ y $g(x) = b_0 + a_1 x + \cdots + b_n x^n + \ldots$ entonces la suma de polinomios está definida por

$$f(x) + g(x) = (a_0 + b_0) + (a_1 + b_1)x + \cdots + (a_n + b_n)x^n + \ldots$$

.

La multiplicación está dada por

$$f(x)g(x) = d_0 + d_1x + \cdots + d_nx^n + \ldots, \text{ donde } d_n = \sum_{i=0}^{n} a_i b_{n-i}.$$

Teorema A.2.4. *El conjunto $R[x]$ de todos los polinomios en un indeterminada x con coeficientes en un anillo R, es un anillo con las operaciones definidas previamente. Si R es un anillo conmutativo, entonces $R[x]$ lo es, y si R es un anillo con unitario entonces $R[x]$ lo es.*

Ejemplo A.2.5. Sea $\mathbb{Z}_2[x]$, tenemos $(x+1)^2 = (x+1)(x+1) = x^2 + (1+1)x + 1 = x^2 + 1$.

Teorema A.2.6. *Si F es un campo entonces $F[x]$ es un dominio entero.*

Si tenemos dos indeterminadas, digamos x, y, y un anillo R, entonces podemos formar el anillo $(R[x])[y]$, el cual consta de polinomios en la indeterminada y, donde los coeficientes son polinomios en la indeterminada x con coeficientes en R. De manera natural $(R[x])[y]$ es isomorfo a $(R[y])[x]$. La manera usual de denotar este anillo es escribiendo $R[x, y]$. Este anillo es denominado anillo de polinomios en dos indeterminadas. Esta idea puede extenderse y tener un *anillo de polinomios en n indeterminadas* $R[x_1, x_2, \ldots, x_n]$.

Notación A.2.7. La letra F denotará un campo.

Teorema A.2.8. *Sea F un subcampo de un campo E. Sea $\alpha \in E$, y x una indeterminada. La función $\phi_a : F[x] \to E$ definida por*

$$\phi(a_0 + a_1x + \cdots + a_nx^n) = a_0 + a_1\alpha + \cdots + a_n\alpha^n$$

para $f(x) = (a_0 + a_1x + \cdots + a_nx^n) \in F[x]$, es un morfismo de $F[x]$ a E. Este morfismo es llamado evaluación en α.

Definición A.2.9. Sean $F, E, \alpha, x, f(x)$ y ϕ como en el teorema A.2.8 . Denotamos a $\phi_\alpha(f(x))$ como $f(\alpha)$. Si $f(\alpha) = 0$, entonces decimos que α es un *cero* o una *raíz* de $f(x)$.

Definición A.2.10 (Algoritmo de la división)**.** Sea F un campo y $F[x]$ su anillo de polinomios. Sean $f(x), g(x) \in F[x]$ de grado $n \geq 0$ y $m > 0$ respectivamente entonces, existen polinomios únicos $q(x), r(x) \in F[x]$ tales que

$$f(x) = g(x)q(x) + r(x),$$

donde $r(x) = 0$ o de grado menor a m. Decimos que $g(x)$ es un factor de $f(x)$ si $q(x) \neq 0$ y $r(x) = 0$.

Teorema A.2.11. *Un elemento $a \in F$ es una raiz de $f(x) \in F[x]$ si y sólo si, $x - a$ es un factor de $f(x)$ en $F[x]$.*

Corolario A.2.12. *Un polinomio $f(x) \in F[x]$ de grado n puede tener a lo sumo n raices.*

Definición A.2.13. Un polinomio $f(x) \in F[x]$ de grado $m \geq 1$ es *irreducible sobre F* o es un *polinomio irreducible en $F[x]$* si, $f(x)$ no puede expresarse como producto de dos polinomios de grado menor. Esto es no existen $g(x), h(x) \in F[x]$ tales que $f(x) = g(x)h(x)$. Si existen estos polinomios decimos que $f(x)$ es *reducible*.

Teorema A.2.14. *Sea $p(x)$ un polinomio irreducible en $F[x]$. Si $p(x)|r(x)s(x)$ para $r(x), s(x) \in F[x]$, entonces $p(x)|r(x)$ o $p(x)|s(x)$.*

Apéndice B

Códigos

En este apéndice se encuentran todos los códigos utilizados en magma. Los códigos pueden ser encontrados en la siguiente dirección `http://computacion.cs.cinvestav.mx/~dcervantes/Tesis/codigos`

Configuración

Código B.1: Configuración de campo binario

```
1  //Grado de la extensión
2  m := 127;
3  q := 2^m;
4  //Se genera el campo F_2
5  F_2 := GF(2);
6  //Se genera F_2[x]
7  F_2P<x> := PolynomialRing(F_2);
8  //Generando un polinomio irreducible de grado m
9  p := IrreduciblePolynomial(F_2, m);
10 //Se construye la extensión F_q
11 F_q<t> := ext<F2 | p>;
12 //Se crea F_q[y]
```

```
13 F_qP<y> := PolynomialRing(F_q);
14 //Se construye F_{q^2}
15 F_q2<u> := ext<F_q | y^2 + y + 1>;
```

Código B.2: Construcción de E y de φ

```
1  //Cargamos archivo de configuración
2  load config.magma;
3
4  //Elegimos un parámetro b aleatorio
5  b := Random(F_q);
6
7  //Creamos la curva elíptica E
8  E := EllipticCurve([F_q2|1,u,0,0,b]);
9  //Definimos el grado de la isogenia
10 d := 3
11 //Usando la formula de Vélu construimos la curva E_Pd
12 //y la isogenia nu
13 E_Pd, nu := IsogenyFromKernel(E, DivisionPolynomial(E, d));
14 //Elevamos a la q los coeficientes de la curva E_Pd
15 coefE_Pd_q := [c^(q) : c in Eltseq(E_Pd)];
16 //Generamos la curva Eq
17 Eq := EllipticCurve(coefE_Pd_q);
18 //Generamos el mapa de Frobenius pi
19 pi := map<E_Pd -> Eq | P :-> Eq ! [P[1]^(q), P[2]^(q), P[3]]>;
20 //Generamos el isomorfismo phi
21 phi := Isomorphism(Eq, E);
22
23 //Componemos los morfismos
24 Endo := nu*pi*phi;
```

Código B.3: Calculando las raices de $X^2 + 9$

```
1  //Se factoriza el orden de E
2  FactO := FactoredOrder(E);
3  //Se elige el factor primo más grande
4  //Magma presenta la factorización en orden ascendente,
5  //por eso elegimos el último elemento
6  r := FactO[#FactO][1];
7
8  //Calculamos el cofactor
9  H := 1;
10 for i := 1 to (#FactO - 1) do
11     H *:= FactO[i][1];
12 end for;
13 //=============================================
14 //Se elige un punto P de manera aleatoria
15 P := Random(E);
16 //Forzando a P a tener orden r
```

```
17  P := H * P;
18  //================================================
19
20  //Construimos el campo F_r = Z_r
21  F_r := GF(r);
22  //Generamos el anillo F_r[l]
23  F_rP<l> := PolynomialRing(F_r);
24
25  //Calculamos las raices del polinomio característico de Endo
26  RootsEndo := Roots(l^2 + 9);
27
28  //Comprobamos cuál de las raices satisface la ecuación
29  //Endo(P) = kP
30  for roots in RootsEndo do
31      if Integers()!roots[1] * P eq Endo(P) then
32          Lambda1 := Integers()!roots[1];
33      end if;
34  end for;
```

En el siguiente código se calculan las raices del polinomio característico de ψ. De hacerse en conjunto con el código anterios, las lineas 1-25 pueden ser ignoradas.

Código B.4: Calculando las raices de $X^2 + 1$

```
1   //Se factoriza el orden de E
2   FactO := FactoredOrder(E);
3   //Se elige el factor primo más grande
4   //Magma presenta la factorización en orden ascendente,
5   //por eso elegimos el último elemento
6   r := FactO[#FactO][1];
7
8   //Calculamos el cofactor
9   H := 1;
10  for i := 1 to (#FactO - 1) do
11      H *:= FactO[i][1];
12  end for;
13  //================================================
14  //Se elige un punto P de manera aleatoria
15  P := Random(E);
16  //Forzando a P a tener orden r
17  P := H * P;
18  //================================================
19
20
21  //Construimos el campo F_r = Z_r
22  F_r := GF(r);
23  //Generamos el anillo F_r[l]
24  F_rP<l> := PolynomialRing(F_r);
```

```
25
26 //Definimos el endomorfismo GLS (psi)
27 GLS := map<E -> E | P :-> E ! [P[1]^(q), P[2]^(q) + u*P[1]^(q), P[3]]>
28
29 //Calculamos las raices del polinomio característico de GLS
30 RootsGLS := Roots(l^2 + 1);
31 ;
32 //Comprobamos cuál de las raices satisface la ecuación
33 //GLS(P) = kP, donde k es una raiz
34 for roots in RootsEndo do
35     if Integers()!roots[1] * P eq GLS(P) then
36         Lambda0 := Integers()!roots[1];
37     end if;
38 end for;
```

Código B.5: 3-Descomposición

```
1  //Elegimos escalar k aleatorio
2  k := Random(r);
3  //Representamos a k como una matriz de 3x1
4  Matrixk := Matrix(RationalField(),3,1, [k, 0,0]);
5  //Construimos la base
6  //Lambda0 := Random(r);
7  //Lambda1 := Random(r);
8  v0 := Matrix(Integers(),3,1, [r, 0,0]);
9  v1 := Matrix(Integers(),3,1, [Lambda0, -1,0]);
10 v2 := Matrix(Integers(),3,1, [Lambda1, 0,-1]);
11 //Representamos la base en forma de Matriz
12 M := Matrix(RationalField(),[[Lambda0, -1, 0],[r,0,0],
13 [Lambda1, 0,-1]]);
14 //Aplicamos el algoritmo LLL a la matriz M
15 B := LLL(M);
16 //Calculamos los escalares beta_i
17 beta := B^-1 * Matrixk;
18 //Aplicamos el redondeo a los beta_i
19 b := [Floor(beta[i][1] + 0.5) : i in [1,2,3]];
20 //calculamos los ki
21 ki := Matrixk - (b[1]*v0 + b[2]*v1 + b[3]*v2);
22 ki := [(Integers()!ki[i][1]) mod r : i in [1,2,3]];
```

Apéndice C

3-Decomposición

C.1. 3-GLV

Zhenghua Zhou, Zhi Hu, Maozhi Xu y Wangan Song[34] nos dicen que es posible extender de manera natural el método GLV usando un segundo endomorfismo para producir un 3-GLV. En este trabajo nos dan condiciones relacionadas a los endomorfismos y una forma de obtener una base para el retículo formado por los endomorfismos. A contianuación se presenta el método que nos proporcionan y el respectivo analisis del costo computacional[1].

Las condiciones que son necesarias para obtener una 3-descomposición son las siguientes.

- Contar con un par de endomorfismos, f_0 y f_1.
- Los endomorfismos sean tales que $f_0(P) = \lambda_0 P, f_1(P) = \lambda_1 P$ y $\lambda_0 \neq \pm\lambda_2$.

[1] Únicamente se considera el costo de la multiplicación escalar y no de la descomposición.

En nuestro caso contamos con φ_d y ψ. Sabemos que $\varphi^2 = [-d^2]$ y que $\psi^2 = [-1]$, por lo tando si consideramos $d \neq 1$, φ_d y ψ satisfacen la segunda condición. Zhenghua *et al.* garantizan una 3-descomposición y que es posible aplicar un 3-GLV como se describe a continuación.

- Considerar la función $f(x, y, z) : x + y\lambda_0 + z\lambda_1 \equiv 0(\text{mód } r)$.
- Encontrar una base $\{v_0, v_1, v_2\}$para $Ker(f)$ tal que $||v_i|| \approx \sqrt[3]{r}$.
- Encontrar números racionales tales que $(k, 0, 0) = \beta_0 v_0 + \beta_1 v_1 + \beta_2 v_2$.
- Usar la aproximación de Babai (algoritmo 2.3.2) para obtener enteros n_0, n_1, b_2 tales que $(k, 0, 0) = b_0 v_0 + b_1 v_1 + b_2 v_2$.
- Obtener los enteros $(k_0, k_1, k_2) = (k, 0, 0) - (b_0 v_0 + b_1 v_1 + b_2 v_2)$.

La base propuesta por Zhenghua *et al.* es

$$v_0 = (r, 0, 0) \qquad v_1 = (\lambda_0, -1, 0) \qquad v_2 = (\lambda_1, 0, -1).$$

Como resultado experimental tienen que, escogiendo aleatoriamente $\lambda_0, \lambda_1 \in \mathbb{Z}_r$ y aplicando el algoritmo LLL (algoritmo 2.3.4) a los vectores v_i obtienen tres vectores linealmente independientes tales que $||v_i|| \approx \sqrt[3]{r}$.

C.1.1. Ejemplo pequeño

A continuación se presenta un ejemplo de esta descomposición usando el campo y curva de la sección 4.2.3. La metodología es,

- Elegir un entero positivo aleatorio no mayor que r.
- Usar la base propuesta por Zhenghua *et al.*.
- Reducir la base usando el algoritmo LLL.
- Encontrar los escalares.

Haremos uso de Magma para efectuar los cálculos. El código utilizado es el código B.5.

- Elegimos $k = 725$
- Sabemos que $r = 1613$, $\psi = [1486]$ y $\varphi_3 = [1232]$.
- Obtenemos la base

$$\begin{pmatrix} 0 & 3 & -1 \\ 13 & -5 & -11 \\ 25 & 6 & 15 \end{pmatrix}$$

- Encontramos los escalares $k_0 = 1, k_1 = -2$ y $k_2 = 3$.

....

Con base en esto tenemos que para calcular kP necesitamos 1 doblado y 3 sumas, más el costo de calcular φ_3. Calculamos NAF(k) = [1, 0, -1, 0, -1, 0, 1, 0, 1, 0, 1]. Con base en NAF(k) podemos ver que el algoritmo 3.3.3 requiere 10 doblados y 6 sumas. Por otro lado si utilizamos el algoritmo 3.4.1 con $\lambda = 1486$ obtenemos los escalares 1 y 7 para los cuales necesitariamos calcular 3 doblados y 2 sumas.