INTERCAMBIO DIFFIE HELLMAN USANDO ISOGENIAS SUPERSINGULARES

LUIS ENRIQUE MANTILLA SANABRIA

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS
ESCUELA DE MATEMÁTICAS
BUCARAMANGA
2023

INTERCAMBIO DIFFIE HELLMAN USANDO ISOGENIAS SUPERSINGULARES

LUIS ENRIQUE MANTILLA SANABRIA

Trabajo de Grado para optar al título de
Matemático

Director
WILSON OLAYA LEÓN
Doctor en Matemáticas
Codirector
JOHN BAYRON BAENA GIRALDO
Doctor en Matemáticas

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS
ESCUELA DE MATEMÁTICAS
BUCARAMANGA
2023

A mi madre quien moldeó
el hombre que soy.

A mi querida hermana
quien amo con mi alma.

A mi padre quien me vio empezar
este proyecto pero no finalizar.

AGRADECIMIENTOS

A mis asesores el Doctor Wilson Olaya León y de especial modo al Doctor John Bayron Baena Giraldo por su consejo, orientación y tiempo dedicado en este trabajo.

Al Doctor Alexander Holguín Villa por su consejo.

A mi compañero de universidad Jhovanny Gutierrez por su amistad.

A todos aquellos que me ayudaron de una o de otra forma en este proyecto.

CONTENIDO

LISTA DE FIGURAS

pág.

RESUMEN

TÍTULO: INTERCAMBIO DIFFIE HELLMAN USANDO ISOGENIAS SUPERSINGULARES [*]

AUTOR: LUIS ENRIQUE MANTILLA SANABRIA [**]

PALABRAS CLAVE: DIFFIE HELLMAN, ISOGENIA, ISOGENIA SUPERSINGULAR, CURVA ELÍPTICA, CRIPTOGRAFÍA, CRIPTOGRAFÍA POSTCUÁNTICA, SIDH.

DESCRIPCIÓN:

En la actualidad nuestra información es salvaguardada por protocolos criptográficos, algunos de los más usados son RSA y Diffie-Hellman sobre curvas elípticas, los cuales buscan un intercambio de llaves de manera segura. Sin embargo estos protocolos ya están amenazados por un algoritmo cuántico, este algoritmo diseñado por Peter Shor garantiza romper protocolos criptográficos como los antes mencionados, siempre y cuando se tenga un computador cuántico "potente", esto no sería un problema si no fuera porque grandes empresas como IBM invierten grandes cantidades de dinero en crear un computador cuántico "potente" y, peor aun, avanzan satisfactoriamente. Es por esto que se crea la rama de la criptografía llamada criptografía postcuántica, la cual busca crear protocolos criptográficos capaces de soportar el ataque de un computador cuántico y que puedan ser implementados en computadores convencionales.

Uno de los primeros protocolos diseñados para un intercambio de llaves de manera segura ante el ataque de un computador cuántico es SIDH (Supersingular Isogeny Diffie-Hellman). Este trabajo se presentan los fundamentos matemáticos del protocolo SIDH, comenzando por definir una curva elíptica y desarrollando la teoría necesaria de las isogenias, luego se presentarán los grafos finitos para ilustrar mejor la aplicación de SIDH y finalmente mostrar la estructura matemática del protocolo SIDH. También mostramos como implementarlo en el lenguaje de programación SageMath.

---

[*] Trabajo de grado

[**] Facultad de Ciencias. Escuela de Matemáticas. Director: Wilson Olaya León, Doctor en Matemáticas. Codirector: John Bayron Baena Giraldo, Doctor en Matemáticas

ABSTRACT

TITLE: DIFFIE HELLMAN KEY EXCHANGE USING SUPERSINGULAR ISOGENIES. *

AUTHOR: LUIS ENRIQUE MANTILLA SANABRIA **

KEYWORDS: DIFFIE-HELLMAN, ISOGENY, SUPERSINGULAR ISOGENY, ELLIPTIC CURVE, CRYPTOGRAPHY, POST-QUANTUM CRYPTOGRAPHY, SIDH.

DESCRIPTION:

Currently, our information is safeguarded by cryptographic protocols, some of the most commonly used being RSA and elliptic curve Diffie-Hellman, which aim to achieve a secure key exchange. However, these protocols are already threatened by a quantum algorithm. This algorithm, designed by Peter Shor, guarantees the breaking of cryptographic protocols such as the ones mentioned above, as long as a "powerful"quantum computer is available. This would not be a problem if it weren't for large companies like IBM investing large amounts of money in creating a "powerful"quantum computer, and worse yet, making satisfactory progress. This is why the field of post-quantum cryptography has been created, which seeks to create cryptographic protocols capable of withstanding a quantum computer attack and that can be implemented on conventional computers.

One of the first protocols designed for secure key exchange in the face of a quantum computer attack is SIDH (Supersingular Isogeny Diffie-Hellman). This work presents the mathematical foundations of the SIDH protocol, beginning by defining an elliptic curve and developing the necessary theory of isogenies. Finite graphs will then be presented to better illustrate the application of SIDH, and finally, the mathematical structure of the SIDH protocol will be shown. We also demonstrate how to implement it in the programming language SageMath.

---

* Bachelor Thesis

** Facultad de Ciencias. Escuela de Matemáticas. Director: Wilson Olaya León, Doctor in Mathematics. Codirector: John Bayron Baena Giraldo, Doctor in Mathematics

INTRODUCCIÓN

En los años 1980, físicos como Richard Feynman[1] o David Deutsch, teorizaron un computador cuántico siguiendo la idea de una *maquina de turing cuántica*, proceso donde el *cúbit* es el análogo al bit en informática. Fue en la década de 1990 cuando se crearon algoritmos cuánticos, o sea algoritmos para ser implementados en computadores cuánticos. Uno de estos es el algoritmo creado por Peter Shor, quien mostró en *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*[2] un algoritmo que factoriza números en tiempo polinomial con el uso de un computador cuántico, cabe decir que no se ha encontrado un algoritmo clásico que haga esto. Lo anterior trae grandes implicaciones a la criptografía moderna, pues la seguridad de algoritmos criptográficos implementados por millones de empresas en el año 2023, depende de la dificultad de factorizar números grandes o resolver el problema del algoritmo discreto como es el caso de RSA o Diffie-Hellman de curvas elípticas respectivamente. Por otro lado, la creación de computadores es una carrera en las que grandes empresas como IBM, QCI (Quantum Circuits) y Microsoft Azure Quantum trabajan y avanzan satisfactoriamente.

El primer computador cuántico con el que se implementó el algoritmo de Shor fue producido por IBM y la Universidad de Stanford en el año 2001, el cual factorizó el número 15 y utilizó 7 cúbits. No fue hasta el nueve de noviembre del 2022 donde IBM en su publicación *IBM Unveils 400 Qubit-Plus Quantum Processor and Next-*

---

[1] Richard P. FEYNMAN. «Simulating physics with computers». En: *International journal of theoretical physics* 21.6-7 (1982), págs. 467-488.

[2] P. W SHORT. «Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer». En: *SIAM review* 41.2 (1999), págs. 303-332.

Generation IBM Quantum System Two[3], expone su nuevo procesador con 433 cúbits, llamado *"Osprey"*, convirtiéndose en el procesador cuántico más potente hasta la fecha. En este mismo artículo periodístico hace mención sobre la dificultad de crear un computador cuántico capaz de romper los sistemas como RSA, pues es necesario una máquina con millones de cúbits.

Dado los riesgos que se enfrenta la seguridad actual ante los computadores cuánticos del futuro, se crea la rama de la criptografía, llamada *criptografía postcuántica*, cuyo propósito es desarrollar sistemas criptográficos capaces de soportar el ataque de un computador cuántico, y que puedan ser implementados en computadores convencionales. Un objetivo claro es implementar estos sistemas criptográficos antes de la aparición de los computadores cuánticos de gran tamaño. Es por esto que la NIST (National Institute of Standards and Technology) en el año 2015 lanza el concurso para la creación y estandarización de protocolos criptográficos postcuanticos, donde tuvieron más de 50 candidatos, quienes se fueron reduciendo a medida que pasaban las rondas establecidas por la NIST, las cuales buscaban ver la vulnerabilidad ante criptoanálisis o su viabilidad. Uno de los finalistas de la cuarta ronda publicada el 5 de julio de 2022 es el protocolo *SIKE* cuyo fundamento teórico era *SIDH*, el cual se basaba en isogenias supersingulares o en inglés *Supersingular Isogeny* (además el primero de su clase), a pesar de que este protocolo fue roto a finales del año 2022, su base teórica fue el inicio de otros protocolos que aún no han sido rotos como *SIDH singatures, CSIDH o OSIDH*.

En esta monografía se desarrollará el trasfondo matemático del protocolo cripto-

---

[3] Chris COLLINS Hugh y NAY. «IBM Unveils 400 Qubit-Plus Quantum Processor and Next-Generation IBM Quantum System Two». En: (2022). https://newsroom.ibm.com/2022-11-09-IBM-Unveils-400-Qubit-Plus-Quantum-Processor-and-Next-Generation-IBM-Quantum-System-Two,

gráfico SIDH, para esto utilizaremos el artículo de Craig Costello[4] y el libro *Elliptic curves: number theory and cryptography*[5] principalmente.

Como preliminar se hablará de conceptos necesarios como la definición de curva elíptica y su ley de grupo, lo cual fue tomado del libro de los autores Decker Wolfram y Pfister Gerhard[6], para luego hablar de teoría de grupos, en particular grupos finitos y de R-modulos, para finalizar hablando de terminología necesaria sobre criptografía y el algoritmo de Diffie-Hellman de curvas elípticas.

En el capítulo 2 desarrollaremos propiedades de las *isogenias*, homomorfismos entre curvas elípticas, y curvas *supersingulares*, es decir, aquellas para las cuales la p-torsión es el conjunto con un solo elemento (el elemento neutro), donde p es la característica del cuerpo en el que se define a la curva elíptica. De los términos supersingular e isogenia se desprenden las dos primeras letras del protocolo SIDH.

En el capítulo 3 introduciremos los términos *grafo de isogenias* y *grafo de ℓ-isogenias*, lo cual nos ayudará en la explicación del protocolo SIDH del capítulo siguiente.

En el capítulo 4 primero se hablará del protocolo SIDH con ayuda del grafo de ℓ-isogenias y mostraremos el significado de las últimas letras de SIDH, las cuales son los apellidos de Diffie y Hellman, y así dar lugar a la seguridad que se creía tener y

---

[4] Craig COSTELLO. «Supersingular Isogeny Key Exchange for Beginners». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020, págs. 21-50.

[5] Lawrence C WASHINGTON. *Elliptic curves: number theory and cryptography*. 2.ª ed. Discrete mathematics and its applications. CRC Press, 2008.

[6] Gerhard DECKER Wolfram y PFISTER. *A First Course in Computational Algebraic Geometry*. New York: Cambridge University Press, 2013.

las últimas noticias acerca de este protocolo. Además, comentaremos algunos protocolos postcuanticos que implementan isogenias supersingulares.

Junto a la parte teórica que se desarrollará, se utilizará el lenguaje de programación SageMath[7], el cual nos ayudará a realizar las cuentas de los ejemplos prácticos a lo largo de todo el trabajo. En particular, la aplicación del protocolo SIDH en el Ejemplo (4.1.2).

[7] *SageMath*. Lenguaje de programación, `https://www.sagemath.org/`.

1. Preliminares

En este capitulo se definirá formalmente *curva elíptica*, el espacio donde vamos a trabajar, algunos resultados sobre teoría de grupos y terminología necesaria sobre criptografía.

1.1. Curvas Elípticas

En esta sección entenderemos el concepto de curva elíptica sobre el plano proyectivo, para así llegar a la forma corta de Weierstrass, la cual se utilizará a lo largo del trabajo. Además se definirá la ley de suma (+), que convierte a una curva elíptica en un grupo Abeliano.

1.1.1. Plano proyectivo Antes de definir curva elíptica es necesario definir el espacio en el que vive, el cual es el **plano proyectivo**.

Definición 1.1.1. *(Espacio afín n-dimensional)*
*Sea K un cuerpo, llamaremos al **espacio afín n-dimensional** sobre el cuerpo K al conjunto*

$$\mathbb{A}^n(K) = \{(a_1, \ldots, a_n) \mid a_1, \ldots, a_n \in K\}$$

Definición 1.1.2. *(Anillo de polinomios) Sea R un anillo, denotaremos como $R[x_1, \cdots, x_n]$ al anillo de los polinomios con las variables $\{x_1, \cdots, x_n\}$ y coeficientes en R.*

Observación 1.1.3. *Claramente $\mathbb{A}^n(K) \subset \mathbb{A}^n(\bar{K})$, donde $\bar{K}$ denota la clausura algebraica del cuerpo K.*

Cada polinomio $f \in K[x_1, \cdots, x_n]$ define una función

$$f: \quad \mathbb{A}^n(K) \quad \mapsto \quad K$$

$$(a_1, \ldots, a_n) \;\mapsto\; f(a_1, \ldots, a_n)$$

la cual llamaremos **función polinomial** sobre $\mathbb{A}^n(K)$. Ahora viendo a f como función podemos hablar de los ceros de f.

Definición 1.1.4. *(Variedad afín)*
*Sea $T \subset K[x_1, \ldots, x_n]$, su lugar **geométrico de ceros** en $\mathbb{A}^n(K)$ es el conjunto*

$$\mathrm{V}(T) = \{p \in \mathbb{A}^n(K) \mid f(p) = 0 \text{ para todo } f \in T\}.$$

*$\mathrm{V}(T)$ se llamará **variedad afín** de T.*

Definición 1.1.5. *(Hipersuperficie)*
*Sea $T = \{f\}$, con f no nulo en $F[x_1, \cdots, x_n]$, decimos que $\mathrm{V}(T)$ es una **hipersuperficie**. Y si $n = 2$, entonces $\mathrm{V}(T)$ se llamará **curva plana**.*

Definición 1.1.6. *(Plano proyectivo)*
*El **plano proyectivo** sobre un cuerpo K, es el conjunto*

$$\mathbb{P}^2(K) \cong \left(\mathbb{A}^3(K) \backslash \{0\}\right) / \sim,$$

donde $\sim$ es una relación de equivalencia sobre $\mathbb{A}^3(K) \setminus \{0\}$. Se dice que dos puntos (x_1, y_1, z_1) y (x_2, y_2, z_2) en $\mathbb{A}^3$ están relacionados bajo $\sim$ si existe $\lambda \in K$ distinto de cero, tal que

$$(x_1, y_1, z_1) = (\lambda x_2, \lambda y_2, \lambda z_2).$$

Y la clase de equivalencia que contiene a (x, y, z) se denota como $[x : y : z]$.

Observación 1.1.7.

- *Si* $[x : y : z]$ *es una clase con* $z \neq 0$*, entonces* $[x : y : z] = [x/z : y/z : 1]$*, estos son llamados puntos "finitos" en* $\mathbb{P}^2(K)$*.*
- *Las clases* $[x : y : z]$ *con* $z = 0$*, son los puntos en el "infinito" en* $\mathbb{P}^2(K)$*.*

A continuación entenderemos la razón de llamar puntos "finitos" e "infinitos".

Observación 1.1.8. *Con la siguiente función*

$$\mathbb{A}^2(K) \hookrightarrow \mathbb{P}^2(K)$$

$$(x, y) \mapsto [x : y : 1],$$

podemos decir que el plano afín se encuentra inmerso en el plano proyectivo, además el plano proyectivo es la unión disyunta del plano afín (puntos "finitos") y los puntos en el "infinito".

Definición 1.1.9. *(Polinomio homogéneo). Sea* $f \in K[x_1, \cdots, x_n]$*, diremos que* f *es* ***homogéneo*** *si para cualquier* $\lambda \in K$ *existe un* $m \in \mathbb{Z}^+$ *tal que* $f(\lambda x_1, \cdots, \lambda x_n) = \lambda^m f(x_1, \cdots, x_n)$*. En otras palabras, Un polinomio es homogéneo si el grado de cada monomio es el mismo para todos.*

Ejemplo 1.1.10. *Sea* $F(x, y, z) = x^2y + z^3 + y^2z$ *un polinomio homogéneo de grado 3 en* $K[x, y, z]$*. Si un polinomio* F *de grado* n *es homogéneo, entonces* $F(\lambda x, \lambda y, \lambda z) = \lambda^n F(x, y, z)$ *para todo* $\lambda \in K$*. De lo anterior, si* F *es homogéneo de grado* n *y* $(x_1, y_1, z_1) \sim (x_2, y_2, z_2)$*, entonces* $F(x_1, y_1, z_1) = 0$ *si y solo si* $F(x_2, y_2, z_2) = 0$*, esto muestra que* F *no depende del representante de la clase de equivalencia para el cero, esto quiere decir que para cada punto del dominio de la función le corresponde una única imagen. O en otras palabras* F *está bien definida sobre el conjunto de ceros sobre* $\mathbb{P}^2(K)$*. Ahora bien, cuando estamos trabajando con un polinomio* $F(x, y, z)$ *arbitrario, no podemos asegurar que los ceros de este polinomio no dependan del*

representante, por ejemplo $G(x, y, z) = x^3 + 2y - 3z$*, observemos que* $G(1, 1, 1) = 0$ *y* $G(2, 2, 2) = 2$ *y sin embargo* $(1, 1, 1) \sim (2, 2, 2)$*, de aquí viene la importancia de trabajar con polinomios homogéneos en* ***variedades proyectivas****.*

Definición 1.1.11. *(Variedades proyectivas)*
Sea $T \subset K[x_0, \ldots, x_n]$ *un conjunto de polinomios homogéneos, su lugar geométrico de ceros* $\mathbb{P}^2(K)$ *es el conjunto*

$$\mathrm{V}_{\mathbb{P}}(T) = \left\{ p \in \mathbb{P}^2(K) \mid f(p) = 0 \text{ para todo } f \in T \right\}.$$

A este tipo de conjuntos los llamaremos ***variedad proyectiva****.*

A continuación definiremos **curva elíptica** usando la ecuación de **Weierstrass**.

Definición 1.1.12. *(Curva elíptica*[8]*) Sea* K *un cuerpo. Una* ***curva elíptica*** E *dada por la forma normal de Weierstrass es la variedad proyectiva del polinomio "no singular"*

$$F(X, Y, Z) = Y^2Z + a_1XYZ + a_3YZ^2 - X^3 - a_2X^2Z - a_4XZ^2 - a_6Z^3, \qquad (1)$$

donde $a_i \in K$*. O en otras palabras, una curva elíptica son las clases de equivalencia de* $\mathbb{P}^2(K)$ *de la siguiente ecuación*

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3. \qquad (2)$$

Observación 1.1.13. *Más adelante hablaremos de lo que significa ser "no singular".*

Ahora, haciendo $Z = 0$ en (2), tenemos la siguiente ecuación

$$0 = X^3.$$

---

[8] Joseph H SILVERMAN. *The Arithmetic of Elliptic Curves*. 2.ª ed. New York, NY: Springer-Verlag, 2009.

Lo anterior implica que $X = 0$, y como Y puede tomar cualquier valor excepto el cero, pues alguna coordenada debe ser distinta de cero, entonces el único punto "infinito" de la curva elíptica, es $[0 : Y : 0] = [0 : 1 : 0]$.
Si queremos ver los puntos "finitos" de la curva elíptica, es sólo hacer $Z = 1$, la cual queda de la siguiente manera

$$f(x, y) = y^2 + a_1xy + a_3y - x^3 - a_2x^2 - a_4x - a_6. \tag{3}$$

Note que los ceros de f se encuentran en $\mathbb{A}^2(K)$, a este proceso se le denomina **deshomogenización** o **forma no homogénea** de (1) y el proceso inverso se le llama **homogenización** o **proyección**, es decir, si tenemos el polinomio (3) y le introducimos la variable Z de tal forma que convierta a f en un polinomio homogéneo, el cual coincidirá con (1).

Ejemplo 1.1.14. *Es fácil ver que cualquier polinomio $f \in K[x, y]$, con K cuerpo. se puede proyectar a $\mathbb{P}^2(K)$, un ejemplo particular, es el siguiente, sea el polinomio $f(x, y) = x - a$, su proyección quedaría de la siguiente manera*

$$F(X, Y, Z) = X - aZ. \tag{4}$$

Veamos que $V(f)$ es precisamente las soluciones de la ecuación $x = a$, las cuales son los puntos (a, y), para cualquier $y \in K$. Por otro lado con $\mathrm{V}_{\mathbb{P}}(F)$, tenemos las clases de equivalencia que satisfacen $X = aZ$, es decir, $\mathrm{V}_{\mathbb{P}}(F)$ son las clases de la forma $[aZ : Y' : Z]$ para cualesquiera $Z, Y' \in K$, no ambos nulos al mismo tiempo. Si $Z \neq 0$ entonces $[aZ : Y' : Z] = [a : Y : 1]$, siendo $Y = Y'/Z$ (Note que estas clases de equivalencia coinciden con la variedad afín). Si $Z = 0$, entonces $[aZ : Y' : Z] = [0 : Y' : 0] = [0 : 1 : 0]$, esto es, la proyección de f tiene un solo punto en el "infinito", el cual es $[0 : 1 : 0]$, además que esta clase de equivalencia también la comparte toda curva elíptica.

Notación 1.1.1. *Gracias a la observación (1.1.8) y el ejemplo anterior, cuando trabajemos con una curva elíptica E, utilizaremos su forma no homogénea. Más aun, los puntos "finitos" de la curva se escribirán como las parejas (x, y) que son raíces de su forma no homogénea.*

Ejemplo 1.1.15. *Tomando de (3) y $f = 0$ tenemos la ecuación*

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6.$$

Si estamos trabajando sobre un cuerpo de característica distinta de 2 entonces podemos hacer la sustitución $y \to \frac{1}{2}(y - a_1x - a_3)$ y así tenemos la ecuación

$$y^2 = 4x^3 + b_2x^2 + 2b_4x + b_6,$$

donde

$$b_2 = a_1^2 + 4a_4, \quad b_4 = 2a_4 + a_1a_3, \quad b_6 = a_3^2 + 4a_6.$$

y si la característica es distinta de 2 y de 3, entonces haciendo la sustitución

$$(x, y) \longmapsto \left(\frac{x - 3b_2}{36}, \frac{y}{108}\right)$$

tenemos la ***forma corta de Weierstrass*** *, y es de la forma*

$$y^2 = x^3 + Ax + B. \tag{5}$$

Nota 1.1.16. *Sea E una curva elíptica de la forma corta de Weierstrass, en lo que sigue de este trabajo, usaremos las siguientes notaciones,*

- *Diremos que E está definida sobre K, si $A, B \in K$.*
- *Cuando una curva E toma valores en $\mathbb{A}^2(K)$ y está definida sobre K, diremos*

que E está sobre K y escribiremos $E(K)$ o si en el contexto está definido el cuerpo K, entonces denotaremos a la curva como E.

Definición 1.1.17. *(Ver[8]) Sea el polinomio $f(x,y) = -y^2 + x^3 + Ax + B$ un polinomio sobre un cuerpo K de característica distinta de 2 y de 3, entonces diremos que f es singular si el discriminate del polinomio $x^3 + Ax + B$ es distinto de cero, es decir, $4A^3 + 27B^2 \neq 0$.*

Notación 1.1.2. *De ahora en adelante utilizaremos cuerpos de característica distinta de 2 y de 3, por ende trabajaremos con la forma corta de Weierstrass. Este tipo de curvas se denotaran como $E : y^2 = x^3 + Ax + B$ o simplemente como E y gracias a que debe de ser no singular, entonces se debe cumplir $4A^3 + 27B^2 \neq 0$. Por lo discutido anteriormente, los puntos "finitos" serán escritos de la forma (x, y) y el único punto en "infinito" será escrito como $\mathcal{O}$. Conjuntivamente podemos decir que*

$$E(K) = \{(x,y) : y^2 = x^3 + Ax + b, 4A^3 + 27B^2 \neq 0, x, y \in K\} \cup \mathcal{O}.$$

Antes de continuar con la ley de grupo sobre la curva E, es necesario hacer la siguiente observación.

El Ejemplo (1.1.14) implica que cualquier recta vertical en el espacio afín $\mathbb{A}^2(K)$, y proyectada en $\mathbb{P}^2(K)$, contiene a la clase $[0:1:0]$, de aquí viene el término "punto en el infinito". Pues si estamos estudiando una curva elíptica E en $\mathbb{A}^2(K)$, no podremos visualizar el punto $[0:1:0]$, sin embargo cualquier recta vertical que veamos en $\mathbb{A}^2(K)$ tendrá una clase de equivalencia en $\mathbb{P}^2(K)$ que comparte con la curva E, el cual será precisamente $[0:1:0] = \mathcal{O}$.

Nota 1.1.18. *Siguiendo el mismo procedimiento que en el Ejemplo 1.1.14 se puede concluir que dos rectas paralelas cualesquiera en el plano afín, en el plano proyectivo compartirán una clase de equivalencia, de aquí viene la afirmación "dos rectas paralelas se cortan en el infinito".*

A continuación se mostrará que el conjunto de puntos que establece la curva elíptica forma un grupo Abeliano aditivo con la operación $(+)$ y que $\mathcal{O}$ será el elemento neutro del grupo.

1.1.2. Operación de grupo De forma general la suma de dos puntos P_1 y P_2 de una curva elíptica E, es trazar una recta que pase por los puntos P_1 y P_2, esta recta cortará en un tercer punto Q el cual será reflejado sobre el eje x, dando el punto P_3 el cual será la suma de P_1 y P_2, ver ejemplo en la Figura 1.

Algoritmo de la adición sobre curvas elípticas $(+)$[5]**:** Sea la curva elíptica $E : Y^2 = X^3 + AX + B$ sobre un cuerpo K de característica distinta de 2 y de 3, con $P_1 = (x_1, y_1)$ y $P_2 = (x_2, y_2)$ puntos en E no nulos. Definamos $P_1 + P_2 = P_3 = (x_3, y_3)$ de la siguiente manera.

- Si $x_1 \neq x_2$, entonces

$$x_3 = m^2 - x_1 - x_2, \quad y_3 = m\,(x_1 - x_3) - y_1, \quad \text{donde } m = \frac{y_2 - y_1}{x_2 - x_1}.$$

- Si $x_1 = x_2$ pero $y_1 \neq y_2$, entonces $P_1 + P_2 = \mathcal{O}$.

- si $P_1 = P_2$ y $y_1 \neq 0$, entonces

$$x_3 = m^2 - 2x_1, \quad y_3 = m\,(x_1 - x_3) - y_1, \quad \text{donde } m = \frac{3x_1^2 + A}{2y_1}.$$

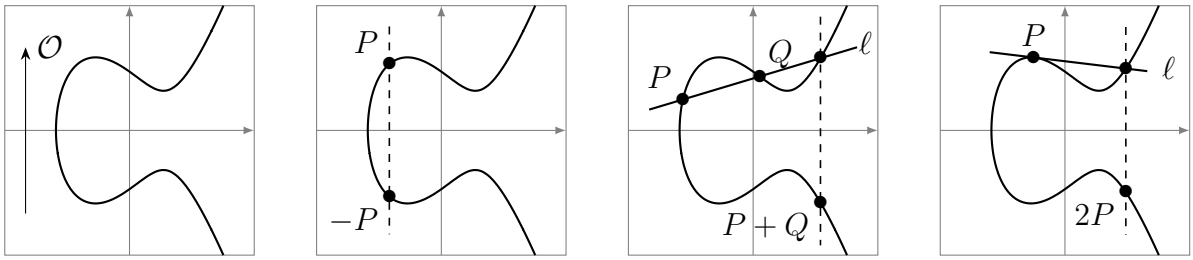
- Si $P_1 = P_2$ y $y_1 = 0$, entonces $P_1 + P_2 = \mathcal{O}$.

- Además, definimos $P + \mathcal{O} = P$ para todo punto P en E.

Denotemos como $-P$ como el inverso aditivo de P, o sea si $P = (x, y)$ entonces $-P = (x, -y)$. Por otra parte, si n es un entero positivo entonces nP es la suma

reiterada de P n-veces y $-nP$ es la suma reiterada de $-P$ n-veces, y si $n = 0$ entonces $0P = \mathcal{O}$.

Observación 1.1.19. *Sea $P = (a, b)$ un punto en la curva elíptica E, luego $-P$ es el punto P reflejado en el eje x. Esto es así gracias a que si trazamos una recta que pasa por P y $-P$, esta recta tendrá la forma $x = a$, lo cual es una recta vertical. Por la discusión de la sección anterior, esta recta comparte con E en el espacio proyectivo el punto $\mathcal{O}$, lo que implica que el corte de $x = a$ y E es precisamente $\mathcal{O}$. O escrito en la forma de suma, se tiene que $P + (-P) = \mathcal{O}$.*



Elemento neutro $\mathcal{O}$ | Elemento inverso $-P$ | $P + Q$ | $2P = P + P$

ℓ:"recta secante" | ℓ: "recta tangente"

Figura 1. Ley de grupo de curvas elípticas, hecho por el autor.

Teorema 1.1.20. *(Curvas elípticas como grupo) La operación suma (+) sobre $E(K)$ con K un cuerpo, hace a $E(K)$ un grupo Abeliano. En particular, si K es finito, $E(K)$ es un grupo finito.*

Demostración. Ver[5]. □

Notación 1.1.3. *De ahora en adelante cuando denotemos una curva elíptica E, se tendrá claro que se trabaja sobre el grupo $(E, +)$ siendo $(+)$ la ley de grupo definida en este trabajo.*

Ejemplo 1.1.21. *($E(\mathbb{F}_{13})$ en SageMath)*

Veamos el grupo $E : Y^2 = X^3 + 3X + 8$ *sobre* $\mathbb{F}_{31}$ *con la operación* $(+)$. *Con ayuda de SageMath definamos la curva* E *con la clase*

$$EllipticCurve(R, [a1, a2, a3, a4, a6])$$

donde R *es el cuerpo que trabajaremos y los valores* $a1, a2, a3, a4$ *y* $a6$ *los coeficientes de la curva elíptica* $Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6$. *Implementando la clase en SageMath tenemos:* `EllipticCurve(GF(31),[3,8])` *donde* $GF(31) = \mathbb{F}_{31}$ *y* $a1 = 3$, $a2 = 8$ *y* $a3 = a4 = 0$. *A continuación veamos cuántos elementos tiene este grupo, para esto usemos el comando* `points()` *el cual nos mostrará una lista de tripletas, las cuales tienen sentido en el espacio proyectivo.*

```
1 E=EllipticCurve(GF(31),[3,8])
2 E.points()
```

```
Salida:
    [(0 : 1 : 0), (0 : 15 : 1), (0 : 16 : 1), (6 : 5 : 1), (6 : 26
↪  : 1), (7 : 0 : 1), (9 : 12 : 1), (9 : 19 : 1), (11 : 15 : 1),
↪  (11 : 16 : 1), (12 : 6 : 1), (12 : 25 : 1), (14 : 2 : 1), (14 :
↪  29 : 1), (15 : 7 : 1), (15 : 24 : 1), (18 : 2 : 1), (18 : 29 :
↪  1), (20 : 15 : 1), (20 : 16 : 1), (21 : 1 : 1), (21 : 30 : 1),
↪  (24 : 4 : 1), (24 : 27 : 1), (27 : 5 : 1), (27 : 26 : 1), (29 :
↪  5 : 1), (29 : 26 : 1), (30 : 2 : 1), (30 : 29 : 1)]
```

Al graficar estos puntos en $\mathbb{R}^2$, *tenemos lo siguiente*

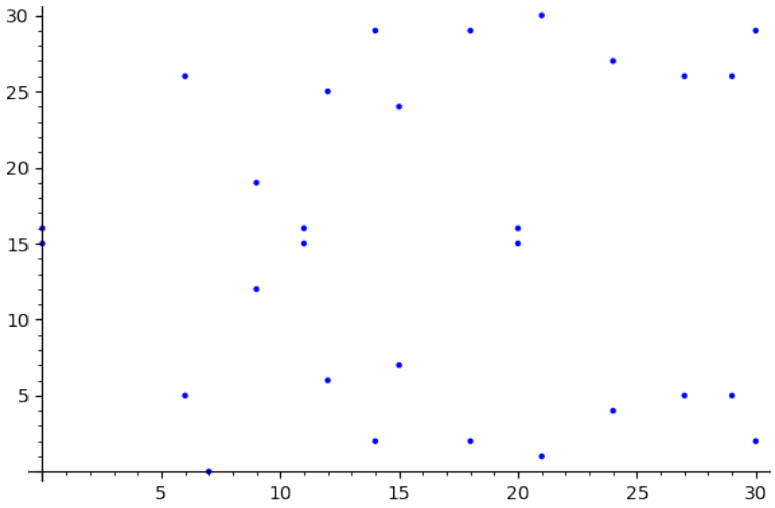


Figura 2. Gráfica de una curva elíptica sobre $\mathbb{F}_{31}$, hecho por el autor en SageMath.

Tomemos las clases $Q = [11 : 15 : 1]$ *y* $P = [27 : 5 : 1]$*, las cuales en el plano afín serían los puntos* $Q = (11, 15)$ *y* $P = (27, 5)$*. Para sumar* $P + Q$ *escribimos*

```
1 print(E(11,15)+E(27,5))


Salida:
    (21 : 30 : 1)
```

este punto en el plano afín es $(21, 30)$*. Podemos observar que en SageMath se puede utilizar la escritura en el plano afín, o sea los puntos de la forma* (x, y)*, sin la necesidad de escribirlo de la forma proyectiva.*

1.2. Teoría de grupos

En esta sección se hablará sobre algunos resultados importantes de teoría de grupos. En lo que sigue del trabajo, todo número denotado por p, será un natural primo.

Notación 1.2.1. *Denotaremos $\mathbb{F}_q$ como el cuerpo finito con q elementos, siendo $q = p^n$.*

Teorema 1.2.1 (Cardinal de un cuerpo finito)**.** *Sea F un cuerpo finito. Entonces F tiene p^n elementos, donde el primo p es la característica de F y n es el grado de F sobre su subcuerpo primo.* [9]

Teorema 1.2.2. *La clausura algebraica del cuerpo con p^n elementos $\mathbb{F}_{p^n}$ tiene característica p.*

Nota 1.2.3. *La característica de un cuerpo es cero o un primo p.*

Proposición 1.2.4. *Cualquier cuerpo algebraicamente cerrado es cuadráticamente cerrado, es decir, si K es un cuerpo algebraicamente cerrado y $a \in K$, entonces existe un $b \in K$, tal que $b \cdot b = a$.*

Proposición 1.2.5. *Sea $\phi : G \to H$ un homomorfismo de grupos Abelianos. Entonces para todo $h \in \phi(G)$ se tiene que*

$$|\phi^{-1}(\{h\})| = |ker(\phi)|.$$

Demostración. Sea $h \in \phi(G)$, entonces existe $g \in G$ tal que $\phi(g) = h$. Demostremos a continuación que $g \cdot ker(\phi) = \phi^{-1}(\{h\})$, siendo $g \cdot ker(\phi) = \{g \cdot x : x \in ker(\phi)\}$. Sea $x \in ker(\phi)$ entonces $\phi(g \cdot x) = \phi(g) \cdot \phi(x) = h \cdot e_H = h$, luego $g \cdot ker(\phi) \subseteq \phi^{-1}(\{h\})$. Ahora tomemos a $g' \in \phi^{-1}(\{h\})$, entonces $\phi(g') = h$, notemos que $g' = g' \cdot g \cdot g^{-1} = g \cdot (g' \cdot g^{-1})$, mostremos que $g' \cdot g^{-1} \in ker(\phi)$, en efecto $\phi(g' \cdot g^{-1}) = \phi(g') \cdot \phi(g^{-1}) = h \cdot \phi(g)^{-1} = e_H$ luego $\phi^{-1}(\{h\}) \subseteq g \cdot ker(\phi)$, entonces $g \cdot ker(\phi) = \phi^{-1}(\{h\})$. Finalmente $|\phi^{-1}(\{h\})| = |g \cdot ker(\phi)| = |ker(\phi)|$. □

---

[9] Harald LIDL Rudolf y NIEDERREITER. *Finite fields*. Cambridge university press, 1997.

Definición 1.2.6. *Sea R un anillo (no necesariamente con identidad). Un R-módulo es un grupo abeliano $(M,+)$ junto con una aplicación $R \times M \to M$, denotada por $(r,x) \mapsto rx$ que satisface para todos $r,s \in R, x,y \in M$,*

- $r(sx) = (rs)x$ *(asociatividad);*
- $(r+s)x = rx + sx$ *(distributiva con respecto a la suma en R);*
  $r(x+y) = rx + ry$ *(distributiva con respecto a la suma en M).*
  Además, si R tiene identidad,
- $1x = x$ *(también se dice que M es unitario)*

Proposición 1.2.7. *Sea n un natural, luego la suma directa $\mathbb{Z}_n \oplus \mathbb{Z}_n$ es un $\mathbb{Z}_n$-módulo.*

Definición 1.2.8. *Sean M un R-módulo y $x \in M$, definimos el anulador de x, como $ann(x) = \{r \in R : rx = 0_M\}$.*

Proposición 1.2.9. *Sean n un natural y x, y enteros y primos relativos con n. Entonces $P = (x,0)$ y $Q = (0,y)$ son base para $\mathbb{Z}_n \oplus \mathbb{Z}_n$, además P y Q tienen orden n y son linealmente independientes.*

Demostración. Claramente P y Q generan a $\mathbb{Z}_n \oplus \mathbb{Z}_n$ y $ann(P) = \{r \in \mathbb{Z}_n : rP = 0\} = \{0\}$ y $ann(Q) = \{0\}$ esto gracias a que x y y son primos relativos con n. □

Proposición 1.2.10. *Sea $r(x) = p(x)/q(x)$ un cociente de polinomios irreducibles sobre un cuerpo K, entonces*

$$\frac{d}{dx}\left(\frac{p(x)}{q(x)}\right) = 0$$

si y sólo si $p'(x) = 0$ y $q'(x) = 0$.

Demostración. Supongamos que $\frac{d}{dx}\left(\frac{p(x)}{q(x)}\right) = 0$ entonces $\frac{p'(x)q(x) - q'(x)p(x)}{(q(x))^2} = 0$, lo que implica que

$$p'(x)q(x) - q'(x)p(x) = 0,$$

$$p'(x)q(x) = q'(x)p(x).$$

Si suponemos que $q'(x) \neq 0$, entonces

$$\frac{p'(x)}{q'(x)} = \frac{p(x)}{q(x)},$$

lo que implica que $deg(p'(x)) \geq deg(p(x))$ y $deg(q'(x)) \geq deg(q(x))$, lo cual es una contradicción, pues $p(x)/q(x)$ es irreducible y el grado de $p'(x)$ es menor al grado de $p(x)$, de manera análoga con $q(x)$. La recíproca es inmediata. □

1.3. Criptografía

Terminología 1.3.1.

- *Criptología: El arte y la ciencia de hacer y romper "códigos secretos"*
- *Criptografía: la elaboración de "códigos secretos "*
- *Criptoanálisis: el romper o descifrar "códigos secretos"*
- *Texto-plano: es el texto que todos pueden entender*
- *Texto-cifrado: es el texto que nadie puede entender, solo las personas con la llave correcta*
- *Cifrar: la acción de pasar del texto-plano al texto-cifrado*
- *Descifrar: la acción de pasar del texto-cifrado al texto-plano*
- *Clave: Es usada para encriptar o descifrar, no necesariamente debe ser la misma para ambas.*
- *Criptosistema: protocolo para cifrar y descifrar*
- *Emisor: el que cifra el texto-plano*

- *Receptor: el que descifra el texto-cifrado*

Definición 1.3.1. *(Criptosistema simétrico y asimétrico) En un cifrado **simétrico**, la misma clave se usa para cifrar y descifrar. En un cifrado **asimétrico** se cifra con una clave l_1 y se descifra con una clave l_2, con $l_1 \neq l_2$ y la clave l_1 la llamaremos **clave pública** y a l_2 **clave privada**.*

Por los fines de este trabajo solo estudiaremos los cifrados asimétricos.

Nota 1.3.2. *Cuando estemos tratando un protocolo criptográfico asimétrico, como es el caso de **Diffie-Hellman de curvas elípticas** o **SIDH** (Supersingular Isogeny Diffie-Hellman), no hablamos de un emisor y un receptor. En estos casos se trabaja como parte A o parte B, algunas veces se llamaran Alice o Bob. Esto se debe a que en un protocolo de intercambio de llaves las dos partes envían y reciben información.*

A continuación se definirá el algoritmo Diffie-Hellman de curvas elípticas.

Algoritmo 1.3.1. *(Diffie-Hellman de curvas elípticas) Sean Alice y Bob las dos partes que quieren compartir una clave simétrica, entonces estas partes establecen una curva $E(\mathbb{F}_q)$, con $q = p^\ell$, y un punto $P_0 \in E(\mathbb{F}_q)$, los cuales deben ser públicos.*

1. *Alice escoge un entero aleatorio a, el cual será su clave privada. Similarmente Bob escoge un entero aleatorio b.*
2. *Alice y Bob calculan aP_0 y bP_0 respectivamente y los hacen públicos, estos serán las claves públicas de cada uno.*
3. *La clave compartida será $a(bP_0) = b(aP_0)$.*

Este protocolo se puede visualizar en el siguiente diagrama.

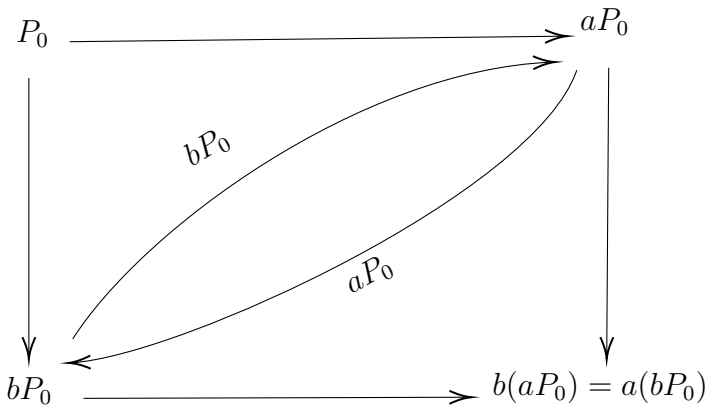


Figura 3. Diagrama del protocolo DH, hecho por el autor.

La seguridad de este algoritmo radica en la dificultad de resolver ***problema del algoritmo discreto sobre curvas elípticas*** *o en su siglas en ingles (ECDLP). A este protocolo se denotará por DH.*

Definición 1.3.3. *(ECDLP) Sean E una curva elíptica sobre el cuerpo $\mathbb{F}_p$, P y Q puntos en $E(\mathbb{F}_p)$. El* ***problema del algoritmo discreto de curvas elípticas (ECDLP)*** *es el problema de hallar un entero positivo n tal que $Q = nP$, este entero n lo denotaremos como*

$$n = \log_P(Q).$$

A continuación realizaremos un ejemplo sencillo del anterior protocolo y utilizaremos SageMath para la cuentas.

Ejemplo 1.3.4. *Tomemos la curva del Ejemplo 1.1.21 y tomemos el punto $P_0 = (30, 29)$. Supongamos que la clave secreta de Alice sea 7 y la clave secreta de Bob sea 3, entonces*

- *Alice calcula $7P_0 = (20, 15)$ y Bob $3P_0 = (29, 5)$.*
- *Alice le pasa a Bob $(20, 15)$ y Bob le pasa a Alice $(29, 5)$, luego Alice calcula $7(29, 5) = (24, 4)$ y Bob $3(20, 15) = (24, 4)$.*

- *La clave compartida entre Alice y Bob es el punto* $(24, 4)$ *de la curva elíptica* E*.*

Haciendo las cuentas en SageMath, quedaría de la siguiente manera

```
E=EllipticCurve(GF(31),[3,8])
P=E(30,29)
A=7
B=3
AP=7*P
BP=3*P
BAP=3*AP
ABP=7*BP
print(f"Clave secreta de Alice: {7}, Clave secreta de Bob: {3}")
print("")
print(f"Clave publica de Alice: {AP}, Clave publica de Bob: {BP}")
print("")
print(f"Secreto compartido de Alice: {ABP}, Secreto compartido
de Bob: {BAP}")
```

```
Salida:
    Clave secreta de Alice: 7, Clave secreta de Bob: 3

    Clave publica de Alice: (20 : 15 : 1), Clave publica de Bob:
↪   (29 : 5 : 1)

    Secreto compartido de Alice: (24 : 4 : 1), Secreto compartido
↪   de Bob: (24 : 4 : 1)
```

Hecho por el autor en Sagemath.

2. Isogenias

En este capítulo hablaremos de algunos resultados sobre curvas elípticas, como homomorfismos entre estas curvas, las cuales son llamadas isogenias. Luego hablaremos sobre algunas isogenias importantes como las isogenias supersingulares, o en inglés *Supersingular Isogeny*, note que sus iniciales son las primeras letras de SIDH (protocolo de estudio de este trabajo) esto no es casualidad, pues la teoría de Supersingular Isogeny es la base matemática de este protocolo.

2.1. j-invariante

Definición 2.1.1. *(j-invariante [5]) Sea $E : y^2 = x^3 + Ax + B$ sobre un cuerpo K de característica distinta de 2 y 3, entonces el j-invariante de E es*

$$j(E) = 1728\frac{4A^3}{4A^3 + 27B^2} \in K.$$

Note que el denominador de la anterior ecuación nunca será cero, pues E debe ser una curva elíptica.

Teorema 2.1.2. *Dos curvas elípticas son isomorfas sobre $\bar{K}$ si y solo si, tienen el mismo j-invariante [8].*

Cuando nos referimos a isomorfismo entre curvas elípticas, hacemos alusión a que existe un isomorfismo de grupo entre los puntos finitos de las curvas elípticas, el cual relacione los ceros de las curvas. Gracias al teorema anterior los j-invariantes crean una clase de equivalencia sobre un cuerpo cerrado algebraicamente, es decir, dos curvas elípticas sobre un cuerpo algebraicamente cerrado están relacionadas si tiene el mismo j-invariante. Sin embargo, si dos curvas elípticas E_1 y E_2 tienen el mismo j-invariante sobre un cuerpo K, el cual no sea cerrado algebraicamente, no

podemos decir nada sobre la existencia de un isomorfismo entre E_1 y E_2 sobre K, ilustremos con un ejemplo. Pero antes veamos cómo deben ser las curvas elípticas si su j-invariantes es 1728 o 0.

Ejemplo 2.1.3.

Sea K un cuerpo de característica distinta de dos y de tres. En este ejemplo trabajaremos con curvas elípticas sobre $\bar{K}$ y definidas en K (es decir, las constantes A y B están en K). Veamos como debe ser una curva E si su j-invariante es 1728 o 0.

- $j = 1728$*: Veamos que cuando el j-invariante de una curva elíptica E sea $j = 1728$, implica que*

$$\frac{4A^3}{4A^3 + 27B^2} = 1.$$

Haciendo operaciones algebraicas tenemos

$$27B^2 = 0,$$

gracias a que K es cuerpo, implica que $B = 0$, entonces la curva E, se puede escribir como

$$y^2 = x^3 + Ax,$$

con $0 \neq A \in \bar{K}$.

- $j = 0$*: Veamos que cuando el j-invariante de una curva elíptica E sea $j = 0$, implica que $4A^3 = 0$ y $A = 0$, entonces la curva E, se puede escribir como*

$$y^2 = x^3 + B$$

con $0 \neq B \in \bar{K}$.

Ejemplo 2.1.4. *Sean $y^2 = x^3 - 25x$ y $y^2 = x^3 - 4x$ dos curvas elípticas sobre el cuerpo de los números racionales $\mathbb{Q}$. Por el ejemplo anterior, el j-invariante de estas*

curvas es $j = 1728$. Con ayuda de SageMath construimos las curvas elípticas sobre $\mathbb{Q}$.

```
Cuerpo=QQ
G=EllipticCurve(Cuerpo,[-25,0])
H=EllipticCurve(Cuerpo,[-4,0])
G.is_isomorphic(H)


Salida:
    False
```

Hecho por el autor en Sagemath.

Para referirnos al cuerpo $\mathbb{Q}$, basta con escribirlo como QQ. El método `is_isomorphic()` nos dice si los grupos son isomorfos, como en este caso la salida es 'False', entonces estas curvas sobre $\mathbb{Q}$ no son isomorfas. Sin embargo, si hacemos el mismo procedimiento sobre la extensión $\mathbb{Q}[\sqrt{10}]$, tenemos lo siguiente

```
Cuerpo=QQ[sqrt(10)]
G=EllipticCurve(Cuerpo,[-25,0])
H=EllipticCurve(Cuerpo,[-4,0])
G.is_isomorphic(H)


Salida:
    True
```

Hecho por el autor en Sagemath.

Este es un claro ejemplo de cuando tomamos dos curvas elípticas sobre un cuerpo

K arbitrario con sus j-invariantes iguales, no se garantiza un isomorfismo entre estas dos curvas en el cuerpo K.

Observación 2.1.5. *Sea j en un cuerpo K con característica distinta de 2 y de 3, es fácil ver que existe una curva elíptica E, tal que su j-invariante es j. Solo es ver que la curva elíptica se puede escribir así*

$$E : y^2 = x^3 + \frac{3j}{1728 - j}x + \frac{2j}{1728 - j}.$$

Note que E está sobre K, o en otras palabras, no es necesario extender el cuerpo K para definir E. Este resultado será utilizado más adelante.

2.2. Endomorfismos

Definición 2.2.1. *(Endomorfismo de E) Un endomorfismo de una curva elíptica $E(\bar{K})$, es un homomorfismo $\alpha : E(\bar{K}) \longmapsto E(\bar{K})$, el cual esta dado por funciones racionales y no nulas. Esto quiere decir que $\alpha(P_1 + P_2) = \alpha(P_1) + \alpha(P_2)$ y existen funciones racionales (cociente de polinomios) $R_1(x, y)$ y $R_2(x, y)$ con coeficientes en $\bar{K}$ tales que*

$$\alpha(x, y) = (R_1(x, y), R_2(x, y))$$

para todo $(x, y) \in E(\bar{K})$.

A continuación hablaremos sobre los puntos donde no están definidas las funciones racionales. Como estamos trabajando con homomorfismos, entonces $\alpha(\mathcal{O}) = \mathcal{O}$.

Ejemplo 2.2.2. *Sea E dada por $y^2 = x^3 + Ax + B$ y sea α el homomorfismo definido por $\alpha(P) = 2P$, para todo $P \in E$. Entonces α es un endomorfismo dado por*

$$\alpha(x, y) = \left(R_1(x, y), R_2(x, y)\right),$$

donde

$$R_1(x,y) = \left(\frac{3x^2+A}{2y}\right)^2 - 2x$$
$$R_2(x,y) = \left(\frac{3x^2+A}{2y}\right)\left(3x - \left(\frac{3x^2+A}{2y}\right)^2\right) - y.$$

Esto último, gracias a la ley de grupo, vea la sección 1.1.2.

Observación 2.2.3. *Estandaricemos las funciones racionales con endomorfismos como se muestra a continuación. Consideremos una curva elíptica escrita de la forma de Weierstrass* $y^2 = x^3 + Ax + B$ *con* $(x,y) \in E(\bar{K})$ *y* $R(x,y)$ *una función racional. Note que cualquier potencia par de* y*, se puede remplazar por un polinomio que depende solo de* x*, es decir,*

$$y^{2n} = (x^3 + Ax + B)^n$$

con n en los naturales. Si la potencia de y *es impar, entonces podemos escribirlo de la siguiente manera*

$$y^{2n+1} = y \cdot (x^3 + Ax + B)^n$$

con n en los naturales. Esto implica que $R(x,y)$ *lo podemos reescribir como*

$$R(x,y) = \frac{p_1(x) + p_2(x)y}{p_3(x) + p_4(x)y},$$

donde p_1, p_2, p_3 *y* p_4 *son polinomios de una sola variable. Además, podemos eliminar la variable* y *del denominador multiplicando el numerador y el denominador por* $p_3(x) - y \cdot p_4(x)$ *y reemplazando* y^2 *por* $y^2 = x^3 + Ax + B$ *, obteniendo así*

$$R(x,y) = \frac{q_1(x) + q_2(x)y}{q_3(x)}. \tag{6}$$

Ahora consideremos el endomorfismo

$$\alpha(x, y) = (R_1(x, y), R_2(x, y)) .$$

Dado que estamos trabajando las curvas elípticas como grupo con las operaciones definidas en la Sección 1.1.2, entonces $\alpha(x, -y) = \alpha(-(x, y))$ *y como* α *es un homomorfismo, entonces*

$$\alpha(x, -y) = -\alpha(x, y).$$

Si escribimos R_1 *y* R_2 *como en (6)*

$$\alpha(x, -y) = (R_1(x, -y), R_2(x, -y))$$

$$-\alpha(x, y) = (R_1(x, y), -R_2(x, y))$$

Esta ultima igualdad es gracias a que estamos en la forma de Weierstrass y como $\alpha(x, -y) = -\alpha(x, y)$ *entonces*

$$(R_1(x, -y), R_2(x, -y)) = (R_1(x, y), -R_2(x, y)) .$$

Y por (6)

$$\frac{q_1(x) - q_2(x)y}{q_3(x)} = \frac{q_1(x) + q_2(x)y}{q_3(x)}.$$

Esto implica que $q_2(x)y = 0$*, y análogamente si* $R_2(x, y)$ *está escrito como (6) de la siguiente manera*

$$R_2(x, y) = \frac{q_4(x) + q_5(x)y}{q_6(x)}.$$

Se tiene que $q_4(x) = 0$*. Finalmente* α *se podría escribir de la siguiente manera*

$$\alpha(x, y) = (r_1(x), r_2(x)y)$$

Siendo r_1 y r_2 funciones racionales. Siguiendo esta idea, ahora podemos hablar de los puntos donde las funciones racionales no están definidas. Escribamos

$$r_1 = \frac{p(x)}{q(x)}$$

como p y q polinomios que no tienen factores en común. Si $q = 0$ para algún punto (x, y), entonces asumiremos que $\alpha(x, y) = \mathcal{O}$. Es fácil probar que si $q_2 \neq 0$ para algún (x, y) entonces r_2 está definida en (x, y). Por lo tanto α tendrá todos los puntos "finitos" bien definidos.

Nota 2.2.4.

- *Note que por lo anterior, podemos concluir que el kernel de α debe ser finito, pues las raíces del polinomio $q(x)$ son finitas.*
- *De ahora en adelante, notaremos a los endomorfsmos de la forma $\alpha(x, y) = (r_1(x), yr_2(x))$, con r_1 y r_2 funciones racionales.*
- *Por los anteriores ítems y si $r_1(x) = p(x)/q(x)$ con p y q contantes, entonces la preimagen de $p(x)/q(x)$ sería infinita, entonces $p(x)$ o $q(x)$ es no constante.*

Definición 2.2.5. *Sea α un endomorfismo escrito de la forma $\alpha(x, y) = (r_1(x), yr_2(y))$, con r_1 y r_2 funciones racionales y siendo $r_1 = \frac{p(x)}{q(x)}$, con p y q polinomios sin factores en común.*

- *El grado de α, se define como*

$$\deg(\alpha) = \mathrm{Max}\{\deg(p(x)), \deg(q(x))\}.$$

- *Un endomorfismo α se dice que es **separable** si la derivada $r_1'(x)$ no es $0_{\overline{K}}$, esto es equivalente a decir que al menos uno de $p'(x)$ o $q'(x)$ son distintas de cero.*

Nota 2.2.6. *Los polinomios con derivada cero, en cuerpos de característica* $p > 0$ *son de la forma* $g(x^p)$*.*

Ejemplo 2.2.7. *Continuando con el Ejemplo 2.2.2, donde* $\alpha(P) = 2P$*. Teníamos*

$$R_1(x, y) = \left(\frac{3x^2 + A}{2y}\right)^2 - 2x.$$

Dado que la curva E *está de la forma* $y^2 = x^3 + Ax + B$ *y con un poco de manipulación algebraica, tenemos que*

$$r_1(x) = \frac{x^4 - 2Ax^2 - 8Bx + A^2}{4\left(x^3 + Ax + B\right)}.$$

Por las definiciones anteriores tenemos:

- *Como* $Max\{deg(x^4 - 2Ax^2 - 8Bx + A^2), deg(4\left(x^3 + Ax + B\right))\} = 4$*, entonces el grado de* α *es 4.*
- *Llamemos* $q(x) = 4(x^3 + Ax + B)$*, como* $q'(x) = 4\left(3x^2 + A\right)$*, entonces* α *es separable.*

Más adelante veremos que este es un ejemplo de una ***isogenia*** *y recibirá el nombre de* ***multiplicación por 2 isogenia****.*

Definición 2.2.8. *(Kernel de un endomorfismo) Sea* α *un endomorfismo no nulo de una curva elíptica* E*. El* ***kernel de*** α *se define como*

$$Ker(\alpha) = \{P \in E(\bar{K}) : \alpha(P) = \mathcal{O}\}.$$

Observación 2.2.9 ($\#Ker(E)$)**.** *De ahora en adelante* $\#Ker(E)$ *es la cantidad de puntos que tiene* $Ker(E)$*.*

Lema 2.2.10. *Sea* E *una curva elíptica sobre* $\mathbb{F}_q$*. Entonces*

$$\phi_q(x, y) = (x^q, y^q),$$

es un endomorfismo de E de grado q, y ϕ_q no es separable.

Demostración. ver [5] □

Proposición 2.2.11. *Sea $\alpha = (r_1(x), yr_2(x))$ un endomorfismo sobre la curva elíptica E, entonces $r_1(x) = p(x)/q(x)$ toma infinitos valores distintos a medida que x recorre $\bar{K}$.*

Demostración. Realicemos la demostración por contradicción, supongamos que $r_1(x)$ toma una cantidad finita de valores, entonces existirá algún w tal que $p(x') - wq(x') = 0$ para una cantidad infinita de $x' \in \bar{K}$. Entonces $p(x) - wq(x) = 0$ para todo $x \in \bar{K}$, luego $p(x)/q(x) = w$ para todo $x \in \bar{K}$. Por la nota en (2.2.4) entonces p o q no es constante. Finalmente p y q deben tener factores en común, y esto es una contradicción. □

Proposición 2.2.12. *Sea α un endomorfismo no nulo y separable de una curva elíptica E, entonces*

$$\deg \alpha = \#\textit{Ker}(\alpha).$$

Si α un endomorfismo no nulo y no es separable, entonces

$$\deg \alpha > \#\textit{Ker}(\alpha).$$

Demostración. Sea α un endomorfismo separable no nulo sobre la curva elíptica E, entonces α se puede escribir como $\alpha(x, y) = (r_1(x), yr_2(x))$ siendo r_1 y r_2 funciones racionales con r_1 escrito de la siguiente forma $r_1 = p(x)/q(x)$. Por la Definición 2.2.5, tenemos que $r_1' \neq 0$, luego $p'q - q'p$ es un polinomio no nulo.
Sea S el conjunto de $x \in \bar{K}$ tal que $(pq' - p'q)(x)q(x) = 0$. Como el polinomio $p'q - q'p$ es no cero y $q(x)$ tampoco, entonces S debe ser finito y la imagen sobre α

debe ser un conjunto finito. Por la Proposición 2.2.11 tenemos que $r_1(x)$ debe tomar infinitos valores a medida que x recorre $\bar{K}$. Ya que $\bar{K}$ es algebraicamente cerrado, entonces es cuadraticamente cerrado, tenemos que para cada $x \in \bar{K}$ existe un punto $(x, y) \in E(\bar{K})$, entonces $\alpha(E(\bar{K}))$ es un conjunto infinito. Gracias a lo anterior, veamos que existe el punto $(a, b) \in E(\bar{K})$, tal que

1. $a \neq 0$, $b \neq 0$ y $(a, b) \neq \mathcal{O}$
2. $(a, b) \in \alpha(E(\bar{K}))$,
3. $a \notin r_1(S)$,
4. $\deg(p(x) - aq(x)) = \mathrm{Max}\{\deg(p), \deg(q)\} = \deg(\alpha)$.

A continuación mostraremos por qué existe dicha pareja (a, b). Como $r_1(S)$ es finito y $\alpha(E(\bar{K}))$ infinitos, entonces hay una cantidad infinita de puntos $(a', b') \in \alpha(E(\bar{K}))$ tales que $a' \notin r_1(S)$. Antes de verificar el item $(4.)$, cabe aclarar que el polinomio $p(x) - a'q(x)$ tendrá el mismo grado que el máximo grado entre p y q, si los grados de p y q son distintos. Supongamos que m es el grado de p y q, y sean k y t los coeficientes del término x^m en p y q respectivamente, entonces la única posibilidad que hay para que $p(x) - a'q(x)$ no tenga grado m, es que $kx^m = a'tx^m$, o sea $k = a't$, despejando a', $a' = kt^{-1}$, esto último es gracias a que estamos trabajando sobre la clausura algebraica de K, más aun como kt^{-1} es único, entonces si nuestro punto tomado (a', b') cumple que $a' = kt^{-1}$, podemos tomar cualquier otro (a, b) tal que $a \neq a'$ y esto es posible, gracias a que anteriormente mostramos que hay infinitos puntos que cumplen las condiciones del 1 al 3. El punto (a, b) será un punto que cumpla los anteriores items, a menos que se mencione lo contrario.

Demostraremos que existe exactamente $deg(\alpha)$ puntos $(x', y') \in E(\bar{K})$ tales que $\alpha(x', y') = (a, b)$. Para cada preimagen (x', y') de (a, b) se tiene que

$$\frac{p\,(x')}{q\,(x')} = a, \quad y' r_2\,(x') = b.$$

Como $(a, b) \neq \mathcal{O}$, entonces $q(x') \neq 0$ y $r_2(x')$ está bien definido. Ya que $b \neq 0$ y $y' r_2(x') = b$, entonces $y' = b/r_2(x')$. Esto implica que y' queda totalmente determinada por x', entonces solo falta ver que valores puede tomar x'. Por (2) , $p(x) - aq(x)$ tiene $deg(\alpha)$ raíces, contando la multiplicidad. Entonces solo falta probar que $p(x) - aq(x)$ no tiene raíces múltiples. Supongamos que x_0 es una raíz múltiple, entonces

$$p(x_0) - aq(x_0) = 0 \quad \text{and} \quad p'(x_0) - aq'(x_0) = 0$$

Multiplicando la ecuación $p = aq$ y $aq' = p'$, tenemos

$$ap(x_0) q'(x_0) = ap'(x_0) q(x_0).$$

Como $a \neq 0$, entonces x_0 es una raíz de $pq' - p'q$, entonces $x_0 \in S$. Luego $a = r_1(x_0) \in r_1(S)$, y esto es contradictorio. Esto concluye existe exactamente $deg(\alpha)$ puntos $(x', y') \in E(\bar{K})$ tales que $\alpha(x', y') = (a, b)$.
Por la Proposición 1.2.5 tenemos que el kernel de α tiene $deg(\alpha)$ elementos y para cualquier $(a, b) \in E(\bar{K})$ hay exactamente $deg(\alpha)$ preimagenes de (a, b).
Supongamos que α no es separable, entonces la demostración se hace similar a la anterior excepto que $p' - aq'$ es el polinomio cero, luego $p(x) - aq(x) = 0$ tiene raíces múltiples entonces tienes menos que $deg(\alpha)$ soluciones. □

Proposición 2.2.13. *Sea E una curva elíptica definida sobre un cuerpo K. Sea $\alpha \neq 0$ un endomorfismo de E. Entonces $\alpha : E(\bar{K}) \to E(\bar{K})$ es sobreyectiva.*

Demostración. Sea $(a, b) \in E(\bar{K})$. Como $\alpha(\mathcal{O}) = \mathcal{O}$, entonces asumiremos que $(a, b) \neq \mathcal{O}$. Como α es un endomorfismo, entonces lo podemos escribir como

$$\alpha(x, y) = (r_1(x), y r_2(x)),$$

con r_1 y r_2 funciones racionales. Entonces podemos reescribir r_1, como $r_1 = \frac{p(x)}{q(x)}$, con p y q polinomios sin raíces en común. La demostración se realizará por casos:
Caso 1: Si $p(x) - aq(x)$ es un polinomio no constante, entonces tendrá una raíz x_0. Note que si $q(x_0) = 0$, entonces $p(x_0) = 0$, esto implicara que q y p tienen una raíz en común, lo cual es contradictorio, luego $q(x_0) \neq 0$. Tomando $y_0 \in \bar{K}$ siendo la raíz cuadrada de $x_0^3 + Ax_0 + B$. Sabemos que y_0 existe, pues todo cuerpo algebraicamente cerrado, también es cuadráticamente cerrado. Entonces $\alpha(x_0, y_0)$ está definido y es igual a $(a, b') \in E(\bar{K})$ para algún b'. Entonces $b'^2 = a^3 + Aa + B = b^2$, luego $b = \pm b'$. Si $b' = b$, hemos terminado. Si $b' = -b$, es solo ver que $\alpha\,(x_0, -y_0) = (a, -b') = (a, b)$.

Caso 2: Veamos el caso cuando $p(x) - aq(x)$ es constante. Por la nota en 2.2.4 p o q no son constantes simultáneamente. Además solo hay a lo mucho una constante a tal que $p - aq$ sea constante, en efecto si a' es otra constante, entonces $(a' - a)\, q = (p - aq) - (p - a'q)$ es constante y $(a' - a)\, p = a'(p - aq) - a\,(p - a'q)$ es constante, lo que implica que p y q serían constantes. Por lo tanto, hay a lo mucho dos puntos (a, b) y $(a, -b)$ para algún b, tales que no estén en la imagen de α (ya que no tendría preimagen estos dos puntos). Podemos elegir (a_1, b_1) tal que $(a_1, b_1) + (a, b) \neq (a, \pm b)$ y exista $P_1 \in E(\bar{K})$ y $\alpha(P_1) = (a_1, b_1)$. Entonces existe P_2 con $\alpha(P_2) = (a_1, b_1) + (a, b)$. Finalmente $\alpha\,(P_2 - P_1) = (a, b)$, y $\alpha\,(P_1 - P_2) = (a, -b)$. Luego α es sobreyectiva. □

2.3. Puntos de torsión

Definición 2.3.1. *Sea E una curva elíptica sobre el cuerpo K. Sea n un entero positivo. A el conjunto*

$$E[n] = \{P \in E(\bar{K}) \mid nP = \mathcal{O}\}$$

*lo llamaremos como el conjunto de **puntos de n-torsión** de E.*

Nota 2.3.2. *Es fácil ver que $E[n]$ es un subgrupo de $E(\bar{K})$.*

Proposición 2.3.3. *Sea E una curva elíptica de la forma $y^2 = x^3 + Ax + B$ sobre un cuerpo K. Recordemos que esta forma es gracias a que K debe tener característica mayor a 3. Entonces*

$$E[2] \cong \mathbb{Z}_2 \oplus \mathbb{Z}_2.$$

Demostración. Por la forma que tomamos la curva elíptica E, podemos decir que existe $e_1, e_2, e_3 \in \overline{K}$, distintos, tales que

$$y^2 = (x - e_1)(x - e_2)(x - e_3).$$

Esto implica que los puntos $(e_1, 0), (e_2, 0)$ y $(e_3, 0)$ pertenecen a la curva $E(\overline{K})$. Gracias a la ley de grupos de las curvas elípticas vista en la sección 1.1.2, tenemos que $2(e_1, 0) = 2(e_2, 0) = 2(e_3, 0) = \mathcal{O}$. Por otro lado, como la característica de K es mayor a 3, deducimos

$$E[2] = \{\mathcal{O}, (e_1, 0), (e_2, 0), (e_3, 0)\}.$$

Viendo a este conjunto como grupo, con la ley de grupo de las curvas elípticas, se tiene que no hay elementos con orden 4, solo tiene elementos de orden 2 y 1, esto implica que $E[2] \cong \mathbb{Z}_2 \oplus \mathbb{Z}_2$ y $E[2]$ tiene 3 subgrupos de orden 2.

□

Observación 2.3.4. *Como $E[2]$ tiene 3 subgrupos de orden 2 y $E[2]$ es un subgrupo de $E(\bar{K})$, entonces podemos decir que $E(\bar{K})$ tiene 3 subgrupos de orden 2.*

Teorema 2.3.5. *Sea E una curva elíptica sobre K y sea n un entero positivo. Si la característica de K no divide a n, o es 0, entonces*

$$E[n] \cong \mathbb{Z}_n \oplus \mathbb{Z}_n.$$

Si la característica p de K es distinta de cero y $p|n$, escribimos $n = p^r n'$ con $p \nmid n'$.

Entonces

$$E[n] \cong \mathbb{Z}_{n'} \oplus \mathbb{Z}_{n'} \quad o \quad \mathbb{Z}_n \oplus \mathbb{Z}_{n'}.$$

Demostración. ver[5] □

Corolario 2.3.6. *Sea E una curva elíptica sobre K y sea ℓ un entero positivo tal que la característica de K no divida a ℓ. Entonces $E(\bar{K})$ tiene $\ell + 1$ subgrupos finitos de orden ℓ.*

Demostración. Por el Teorema 2.3.5 y la nota en 2.3.2. □

Observación 2.3.7. *Observemos que si E es una curva elíptica sobre un cuerpo K de característica p, siguiendo la notación del teorema 2.3.5 y siendo $n = p$, entonces tenemos dos posibilidades*

$$E[n] \cong \mathbb{Z}/n'\mathbb{Z} \oplus \mathbb{Z}/n'\mathbb{Z} \quad o$$

$$E[n] \cong \mathbb{Z}/n\mathbb{Z} \oplus \mathbb{Z}/n'\mathbb{Z}.$$

Como $n' = 1$ y remplazando con p, tenemos

$$E[p] \cong \mathbb{Z}/\mathbb{Z} \oplus \mathbb{Z}/\mathbb{Z} \quad o$$

$$E[p] \cong \mathbb{Z}/p\mathbb{Z} \oplus \mathbb{Z}/\mathbb{Z},$$

por isomorfismos concluimos que

$$E[p] \cong \mathcal{O} \quad o$$

$$E[p] \cong \mathbb{Z}/p\mathbb{Z}.$$

Definición 2.3.8. *Sea E una curva elíptica sobre un cuerpo de característica p. Definimos lo siguiente:*

- *Si $E[p] \cong \mathbb{Z}_p$, entonces llamaremos a E* ***ordinaria****.*
- *Si $E[p] \cong \{\mathcal{O}\}$, entonces llamaremos a E* ***supersingular****.*

Nota 2.3.9. *Note que el término "supersingular" y "singular" (puntos singulares de la curva) no están relacionados.*

Observación 2.3.10. *Las curvas supersingulares serán de principal importancia para el algoritmo SIDH que veremos después.*

Definición 2.3.11. *Sea p un número primo. Denotaremos al cuerpo finito con p elementos, como $\mathbb{F}_p$.*

Definición 2.3.12. *(j-invariante supersingulares) Gracias a que el j-invariante me define una clase de equivalencia, llamaremos al conjunto de las curvas supersingulares que comparten el mismo j-invariante, como la clase de los j-**invariantes supersingulares**.*

Teorema 2.3.13. *Sea E una curva elíptica sobre $\mathbb{F}_q$, con $q = p^m$, y E supersingular, entonces $j(E) \in \mathbb{F}_{p^2}$. Más aun, el número de clases de equivalencia $\mathcal{N}$ sobre $\mathbb{F}_{p^2}$ es $\mathcal{N} = 1$ si p es 2 ó 3. Si $p \neq 2, 3$, entonces $\mathcal{N} = \lfloor p/12 \rfloor + \epsilon_p$ si $p \equiv 1, 5, 7, 11 (\mathrm{mód} 12)$, donde $\epsilon_p = 0, 1, 1, 2$ respectivamente.*

Demostración. ver [10] □

El siguiente ejemplo muestra la aplicación del teorema anterior.

---

[10] S GALBRAITH. «Mathematics of public key cryptography (version 0.6)». En: *Available in: https://www.math.auckland.ac.nz/~sgal018/crypto-book/main.pdf* (2010), págs. 165-203.

Ejemplo 2.3.14. *Calculemos por fuerza de cómputo los j-invariantes supersingulares sobre $\mathbb{F}_{107}$ y $\mathbb{F}_{431}$ utilizando SageMath. El algoritmo de SageMath nos calcula todos los j-invariantes supersingulares de curvas que están sobre $\mathbb{F}_{107}$.*

```
1 S = SupersingularModule(107)
2 L,d = S.supersingular_points()
3 print(L)


   Salida:
       [16, 94, 68*a + 37, 39*a + 95, 81, 26*a + 22, 81*a + 19,
↪  72, 47, 0]
```

Hecho por el autor en Sagemath.

Veamos que la cantidad de j-invariantes supersingulares son 10, lo que coincide si aplicamos el teorema anterior. Pues $\lfloor 107/12 \rfloor = 8$ y $p \equiv 11(\text{mód}12)$, entonces hay 10 j-invariantes supersingulares.
Ahora con el primo 431, y usando el mismo algoritmo tenemos

```
S = SupersingularModule(431)
L,d = S.supersingular_points()
print(L)
```

Salida:

```
[4, 19, 241, 234, 379*a + 144, 52*a + 92, 242, 189, 237*a + 412, 217*a + 172, 214*a + 389, 194*a + 218, 67, 150, 381, 138*a + 357, 336*a + 404, 265*a + 273, 95*a + 309, 166*a + 107, 293*a + 64, 326*a + 141, 105*a + 36, 143, 358, 14*a + 372, 419, 107, 417*a + 386, 319, 422, 125, 102, 61, 316, 356, 0]
```

Hecho por el autor en Sagemath.

Usando el método $len()$, contamos cuantos elementos tiene la lista L, así:

```
len(L)
```

Salida:

```
37
```

Hecho por el autor en Sagemath.

Es fácil comprobar el teorema con este ejemplo, de manera similar que el anterior.

Observación 2.3.15. *Por el teorema 2.3.13 y la observación 2.1.5 tenemos que para cada j-invariante supersingular existe una curva elíptica E sobre $\mathbb{F}_{p^2}$.*

2.4. Isogenias

Definición 2.4.1. *(Isogenia) Sean E_1 y E_2 dos curvas elípticas definidas sobre K un cuerpo. Una isogenia de E_1 a E_2 es un homomorfismo no constante*

$$\psi : E_1(\bar{K}) \longmapsto E_2(\bar{K}),$$

el cual es dado por funciones racionales. Esto significa que

$$\psi(P+Q) = \psi(P) + \psi(Q) \text{ para todo } P, Q \in E_1(\overline{K})$$

y existen R_1 y R_2 funciones racionales (cociente de polinomios) tales que, si $\psi(x_1, y_1) = (x_2, y_2)$, entonces

$$x_2 = R_1(x_1, y_1), \quad y_2 = R_2(x_1, y_1)$$

para todos los puntos excepto para una cantidad finita $(x_1, y_1) \in E(\bar{K})$, similarmente como en la sección de endomorfismos.
*Dos curvas elípticas E_1 y E_2 son llamadas **isógenas** si hay una isogenia ψ desde E_1 a E_2 con $\psi(E_1) \neq \{\mathcal{O}\}$.*

Nota 2.4.2.

- *Con la definición anterior, es fácil ver que un endomorfismo no nulo de una curva elíptica E es una isogenia.*
- *En la sección de endomorfismos mostramos que α puede ser escrito como $(x_2, y_2) = \alpha(x_1, y_1) = (r_1(x_1), y_1 r_2(x_1))$, donde r_1 y r_2 son funciones racionales, siguiendo el mismo razonamiento llegamos a que una isogenia ψ se puede escribir como*

$$\psi(x_1, y_1) = (r_1(x_1), y_1 r_2(x_1)),$$

donde r_1 y r_2 son funciones que dependen solamente de x. Si los coeficientes de r_1 y r_2 están en K, decimos que ψ ***está definida sobre*** *K.*

Definición 2.4.3. *(Grado de una isogenia) Sea ψ una isogenias definida sobre K de E_1 a E_2 escrita de la forma*

$$\phi\left(x_1, y_1\right) = \left(r_1\left(x_1\right), y_1 r_2\left(x_1\right)\right),$$

y siendo

$$r_1(x) = p(x)/q(x),$$

con p y q polinomios sin factores en común. Definimos el ***grado*** *de ψ como:*

$$\deg(\psi) = \mathrm{Max}\{\deg p(x), \deg q(x)\}$$

Además, si la derivada $r_1'(x)$ no es cero, decimos que ψ es ***separable****.*

Proposición 2.4.4. *Sean E_1, E_2 y E_3 curvas elípticas sobre un cuerpo K de característica p. Sean $\psi_1 : E_1 \mapsto E_2$ y $\psi_2 : E_2 \mapsto E_3$ isogenias separables, tales que $\psi_1(x, y) = (r(x), yr_2(x))$ y $\psi_2(x, y) = (t(x), yt_2(x))$ siendo $r(x) = p(x)/q(x)$ y $t(x) = h(x)/g(x)$ cocientes irreducibles. Entonces $\psi_1 \circ \psi_2$ es separable. En otras palabras, la composición finita de isogenias separables es separable.*

Demostración. Por ser ψ_1 y ψ_2 separables, tenemos que $r'(x) \neq 0$ y $t'(x) \neq 0$, lo que implica por la proposición 1.2.10 que $p'(x) \neq 0$ ó $q'(x) \neq 0$. Para que la compuesta sea separable, basta ver que la siguiente derivada no es cero

$$\frac{d}{dx}\left(r(t(x))\right) = r'(t(x))t'(x).$$

Como $t'(x) \neq 0$ entonces solo falta ver que $r'(t(x)) \neq 0$, para llegar a esto es nece-

sario ver que $p'(t(x)) \neq 0$ o $q'(t(x)) \neq 0$. Supongamos que $p'(x) \neq 0$ y mostremos que $p'(t(x)) \neq 0$, gracias a que $p'(x) \neq 0$, entonces existe al menos un término de $p'(x)$ que se puede escribir como ax^n siendo $a \neq p$ y $n > 0$ (si tomamos $n = 0$ ya lo tenemos) ahora remplazando x por $t(x) = h(x)/g(x)$, entonces

$$a\,(h(x)/g(x))^n \neq 0$$

lo que implica que $p'(t(x)) \neq 0$. Similarmente se hace si suponemos $q'(x) \neq 0$. □

Definición 2.4.5. *(ℓ-isógenas) Dos curvas son ℓ-**isógenas** si existe una isogenia de grado ℓ entre ellas. En dicho caso se dice que hay una ℓ-**isogenia** entre ellas.*

Proposición 2.4.6. *Sea $\psi : E_1 \longmapsto E_2$ una isogenia. Si ψ es separable, entonces*

$$\deg \psi = \#\operatorname{Ker}(\psi).$$

Si ψ no es separable, entonces

$$\deg \psi > \#\operatorname{Ker}(\psi).$$

Además, el kernel de una isogenia es un subgrupo finito de $E_1(\overline{K})$.

Demostración. La demostración es similar a la demostración de la proposición 2.2.12 □

Proposición 2.4.7. *Sea $\psi : E_1 \longmapsto E_2$ una isogenia. Entonces $\psi : E_1(\bar{K}) \longmapsto E_2(\bar{K})$ es sobreyectiva.*

Demostración. La demostración es similar a la demostración de la proposición 2.2.13 □

Definición 2.4.8. *(Multiplicación por m) Sean E una curva elíptica sobre un cuerpo K, y $P \in E(\bar{K})$. Para un $m \in \mathbb{Z}$ definamos*

$$[m]_E : E(\bar{K}) \longmapsto E(\bar{K})$$

$$P \longmapsto mP,$$

donde mP es como se definió en (1.1.2). En particular, la identidad es cuando $m = 1$ y nulo con $m = 0$. Llamaremos a esta estructura ***multiplicación por m*** *y se denotará como $[m]_E$.*

Proposición 2.4.9. *Sea $m \in \mathbb{Z}$ y $m \neq 0$, la Multiplicación-por-m y denotado por $[m]_E$, es una isogenia y tiene grado m^2.*

Demostración. ver [10] □

Observación 2.4.10. *Gracias a la definición de $[m]_E$ sobre un cuerpo K es fácil ver que*

$$Ker([m]_E) = \{P \in E(\bar{K}) : mP = \mathcal{O}\} = E[m],$$

Donde $E[m]$ es la m-torsión de E, ver la definición 2.3.1. Si la característica de K es distinta de cero y no divide a m, entonces por el teorema 2.3.5 se tiene que

$$Ker([m]_E) \cong \mathbb{Z}_m \oplus \mathbb{Z}_m.$$

A continuación ejemplificaremos la anterior observación y notaremos cosas a tener en cuenta.

Ejemplo 2.4.11. *Tomemos la curva elíptica E definida como $y^3 = x^3 + 3x + 4$ sobre el cuerpo $\mathbb{F}_{107}$. Con ayuda de SageMath buscaremos los puntos $P \in E(\mathbb{F}_{107})$ tales que $2P = \mathcal{O}$, de la siguiente manera:*

```
1  E=EllipticCurve(GF(107),[3,4])
2  for i in E:
3     if 2*i==E[0]:
4        print(f"2*{i}= {2*i}")
5     #print(f"2*{i}= {2*i}")


    Salida:
        2*(0 : 1 : 0)= (0 : 1 : 0)
        2*(15 : 0 : 1)= (0 : 1 : 0)
        2*(93 : 0 : 1)= (0 : 1 : 0)
        2*(106 : 0 : 1)= (0 : 1 : 0)
```

Hecho por el autor en Sagemath.

Como 107 *no divide a* 2*, entonces por el teorema anterior, tenemos que el cardinal del* $Ker([2]_E)$ *es 4, entonces*

$$Ker([2]_E) = \{(0:1:0), (15:0:1), (93:0:1), (106:0:1)\}.$$

Veamos que $Ker([2]_E) \subset E(\mathbb{F}_{107})$ *pero como veremos a continuación esto no es un comportamiento general. Algo para notar en las anteriores líneas de código es que si se quita el símbolo "#" en la quinta línea, se podrá ver todos los puntos* P *de* E *al ser multiplicados por 2.*

Ahora realicemos el mismo procedimiento en SageMath pero con la curva $E' : y^3 = x^3 + 3x + 3$*.*

```
E=EllipticCurve(GF(107),[3,3])
for i in E:
    if 2*i==E[0]:
        print(f"2*{i}= {2*i}")
    #print(f"2*{i}= {2*i}")


 Salida:
     2*(0 : 1 : 0)= (0 : 1 : 0)
     2*(46 : 0 : 1)= (0 : 1 : 0)
```

Hecho por el autor en Sagemath.

Similarmente como en el anterior caso, se tiene que el kernel de $[2]_{E'}$ debe tener 4 elementos, esto implica que a diferencia del anterior $ker([2]_{E'}) \nsubseteq E'(\mathbb{F}_{107})$. Esto se debe a que $[2]_{E'}$ debe ser evaluado sobre la clausura algebraica del cuerpo, para esto, deberíamos ver los puntos P en $E(\overline{\mathbb{F}}_{107})$, ver la definición 2.4.8.

Teorema 2.4.12. *Sea E_1 y E_2 curvas elípticas sobre un cuerpo K. Sea $\psi : E_1(\bar{K}) \mapsto E_2(\bar{K})$ una función no constante dada por funciones racionales. Si $\psi(\mathcal{O}) = \mathcal{O}$, entonces ψ es un homomorfismo, más aún una isogenia.*

Demostración. ver [5] □

Teorema 2.4.13. *(Teorema de Vélu) Sea E una curva elíptica dada por la ecuación general de Weierstrass*

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,$$

con a_i en un cuerpo K. Sea C un subgrupo finito de $E(\bar{K})$. Entonces existe una única curva elíptica E_2 (salvo isomorfismos) e isogenia separable $\psi : E \mapsto E_2$ tal

que $C = ker(\psi)$ *y escribimos* $E_2 = E/\langle C\rangle$.

Demostración. Ver[5] □

Nota 2.4.14. *Cabe aclarar que la demostración de este teorema se realiza por construcción, lo que implica que tenemos un algoritmo que calcula la curva* $E/\langle C\rangle$.

Ejemplo 2.4.15. *Sea* $E : y^2 = x^3 + Ax + B$ *una curva elíptica sobre el cuerpo* K *de característica mayor a 3 y* $C = \{\mathcal{O}, P = (x', y')\}$ *subgrupo de* $E(\bar{K})$ *de orden 2. Las formulas para calcular la isogenia* $\psi : E \mapsto E/\langle C\rangle$ *son las siguientes. La curva* $E/\langle C\rangle$ *se puede escribir de la forma*

$$E/\langle C\rangle : y^2 = x^3 + \left(A - 5(3x'^2 + A)\right)x + \left(b - 7\left[(-2y')^2 + x'(3x'^2 + A)\right]\right),$$

y la isogenia ψ *se pueden definir por las coordenadas* X *y* Y *de la forma*

$$X = x + \frac{3x'^2 + A}{x - x'} + \frac{(-2y')^2}{(x - x')^2},$$

$$Y = y - \left(-2y'\frac{2y}{(x - x')^3} - 2y'\frac{y - y'}{(x - x')^2} - \frac{(3x^2 + A)(-2y')}{(x - x')^2}\right).$$

Observación 2.4.16. *Gracias al teorema 2.4.13 concluimos que las isogenias separables están completamente determinadas por su kernel.*

Corolario 2.4.17. *Sea* E *una curva elíptica sobre un cuerpo* K *y* ℓ *un natural tal que la característica de* K *no divida a* ℓ. *Entonces existe un conjunto* $\{E_i\}_{i\in\{1,\cdots,\ell+1\}}$ *de curvas elípticas distintas y únicas (salvo isomorfismos) e isogenias separables* $\psi_i : E \mapsto E_i$ *tales que* $ker(\psi_i) = G_i$, *donde* G_i, *es un subgrupo de* $E(\bar{K})$.

Demostración. Por el Corolario (2.3.6), existe un conjunto $G = \{G_{i\in\{1,\cdots,\ell+1\}}\}$, donde $G_{i\in\{1,\cdots,\ell+1\}}$ son subgrupos finitos de orden ℓ y distintos de $E(\bar{K})$. Por el Teorema (2.4.13) existe un conjunto $\{E_i\}_{i\in\{1,\cdots,\ell+1\}}$ de curvas elípticas distintas y únicas (salvo isomorfismos) e isogenias separables $\psi_i : E \mapsto E_i$ tales que $ker(\psi_i) = G_i$. □

Teorema 2.4.18. *Sea $\psi : E_1 \mapsto E_2$ una isogenia entre curvas elípticas sobre un cuerpo de característica p. E_1 es supersingular si y solo si E_2 es supersingular.*

Demostración. Ver[11]. □

[11] Andrew V. SUTHERLAND. «Identifying supersingular elliptic curves». En: *LMS journal of computation and mathematics* 15 (2012), págs. 317-325.

3. Grafos de Isogenias

En este capítulo definiremos el grafo de isogenias, en el cual será muy útil a la hora de ilustrar el protocolo SIDH en el capítulo 5.

3.1. Grafo de isogenias supersingulares

Definición 3.1.1. *(Grafo) Un grafo G es un par ordenado $G = (V, E)$, donde:*

- *V es un conjunto de vértices o nodos,*
- *E es un conjunto de aristas o arcos, que relacionan estos nodos.*

Definición 3.1.2. *Un **grafo de isogenias** es un (multi-)grafo (un vértice puede tener varias aristas) $G = (V, E)$, cuyos nodos son los j-invariantes de curvas elípticas y sus aristas representan isogenias entre las curvas con ese j-invariante.*

Definición 3.1.3. *El grafo de ℓ-isogenias, $G_\ell(K)$, es un subgrafo del grafo de isogenias, definidas sobre K, en el que solo se consideran las aristas definidas por isogenias de grado ℓ.*

Por el Teorema 2.4.18 podemos decir que el grafo de isogenias no es conexo, pues dos curvas E_1 y E_2 donde E_1 supersingular y E_2 ordinaria, no es posible encontrar una isogenia entre ellas. Sin embargo, el teorema a continuación asegura que $G_\ell(\mathbb{F}_q)$ sobre curvas supersingulares es conexo.

Teorema 3.1.4. *El grafo de ℓ-isogenias de curvas elípticas supersingulares sobre $\mathbb{F}_q$ es conexo, $\ell + 1$-regular y tiene la propiedad de Ramanujan.*

Demostración. Ver[12]. □

[12] Luca DE FEO et al. «Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies». En: *Journal of mathematical cryptology* 8.3 (2014), págs. 209-247.

Decir que es $\ell + 1$-regular hace mención a que cada vértice tiene $\ell + 1$ aristas que salen de él, y que tiene la propiedad de Ramanujan hace referencia a la dificultad de encontrar la isogenia entre las clases.

Observación 3.1.5. *Por el Teorema 2.3.13 sabemos que los j-invariantes supersingulares sobre el cuerpo $\mathbb{F}_p$ están en $\mathbb{F}_{p^2}$, luego el grafo ℓ-isogenias es finito y los nodos serán elementos de $\mathbb{F}_{p^2}$.*

Antes de continuar con un ejemplo, es necesario explicar cómo se calcularán las aristas que unan a los nodos. En otras palabras, ¿cómo sabemos qué nodos están directamente conectados por alguna ℓ-isogenia?. La respuesta se presenta en el siguiente teorema.

Teorema 3.1.6. *([5]) Sea ℓ un primo distinto de p, E_1 y E_2 curvas elípticas con su respectivo j-invariante j_1 y j_2. Entonces existe un polinomio $\Phi_\ell(X, Y) \in \mathbb{Z}[X, Y]$ tal que $\Phi_\ell(j_1, j_2) = 0$, si y solo si hay una isogenia de E_1 a E_2 de grado ℓ.*

El polinomio $\Phi_\ell(X, Y)$ es de grado $\ell + 1$ y simétrico con respecto a X y Y. $\Phi_\ell(X, Y)$ es llamado **polinomio modular**. Muchos polinomios de este estilo ya están calculados según el valor de ℓ, por ejemplo la universidad del MIT tiene una lista en su sitio web*Modular polynomials*. página web, `https://math.mit.edu/~drew/ClassicalModPolys.html`. Junto a esta lista tenemos los polinomios $\Phi_2(X, Y)$ y $\Phi_3(X, Y)$ ya calculados, los cuales son:

$$\begin{aligned}\Phi_2(x, y) = {} & x^3 - x^2y^2 + 1488x^2y - 162000x^2 + 1488 * xy^2 + 40773375xy \\ & + 8748000000x + y^3 - 162000 * y^2 + 8748000000y - 157464000000000,\end{aligned}$$

$$\Phi_3(x,y) = x^4 + 36864000x^3 + 452984832000000x^2 + 1855425871872000000000x + y^4$$
$$+ 36864000y^3 + 452984832000000y^2 + 1855425871872000000000y - x^3y^3$$
$$+ 2587918086x^2y^2 - 770845966336000000xy + 2232x^3y^2 - 1069956x^3y$$
$$+ 8900222976000x^2y + 2232y^3x^2 - 1069956y^3x + 8900222976000y^2x.$$

Nota 3.1.7. *La necesidad de hablar de los polinomios $\Phi_2(x,y)$ y $\Phi_3(x,y)$ concretamente se debe a que son los únicos que se utilizarán durante el protocolo criptográfico SIDH.*

A continuación mostraremos un ejemplo de grafo de 2-isogenias y 3-isogenias de curvas elípticas supersingulares sobre $\mathbb{F}_{107}$.

Ejemplo 3.1.8. *Definamos primeramente los polinomios sobre el cuerpo $\mathbb{F}_{p^2}$ con SageMath.*

```
R.<x,y>=PolynomialRing(GF(p^2,'a'));
x, y = R.gens()

Phi2=x^3-x^2*y^2 + 1488*x^2*y- 162000*x^2 + 1488*x*y^2
+40773375*x*y + 8748000000*x + y^3- 162000*y^2 +
8748000000*y-157464000000000;

Phi3=x^4+36864000*x^3+452984832000000*x^2
+1855425871872000000000*x+y^4+36864000*y^3
+452984832000000*y^2+1855425871872000000000*y-x^3*y^3
+2587918086*x^2*y^2-770845966336000000*x*y
+2232*x^3*y^2-1069956*x^3*y+8900222976000*x^2*y
+2232*y^3*x^2-1069956*y^3*x+8900222976000*y^2*x
```

Hecho por el autor en Sagemath.

$Phi2$ y $Phi3$ son los polinomios $\Phi_2(x,y)$ y $\Phi_3(x,y)$ respectivamente. Ahora calculemos el grafo de 2-isogenias de curvas elípticas supersingulares sobre $\mathbb{F}_{107}$ usando SageMath. Gracias al ejemplo 2.3.14, sabemos que hay 10 j-invariantes supersingulares los cuales son $L = \{16,\ 94,\ 68\alpha+37,\ 39\alpha+95,\ 81,\ 26\alpha+22,\ 81\alpha+19,\ 72,\ 47,\ 0\}$, estos j-invariantes supersingulares son los nodos de nuestro grafo, ver Figura (4).

Ahora es necesario calcular las 2-isogenias que une a los nodos. Siendo $p = 107$ y L la lista de los j-invariantes supersingulares, el siguiente código compara todos los elementos de L y si un par j_1 y j_2 de este conjunto satisface $\Phi_2(j_1, j_2) = 0$, entonces, existe una 2-isogenia entre las clases de j_1 y j_2.

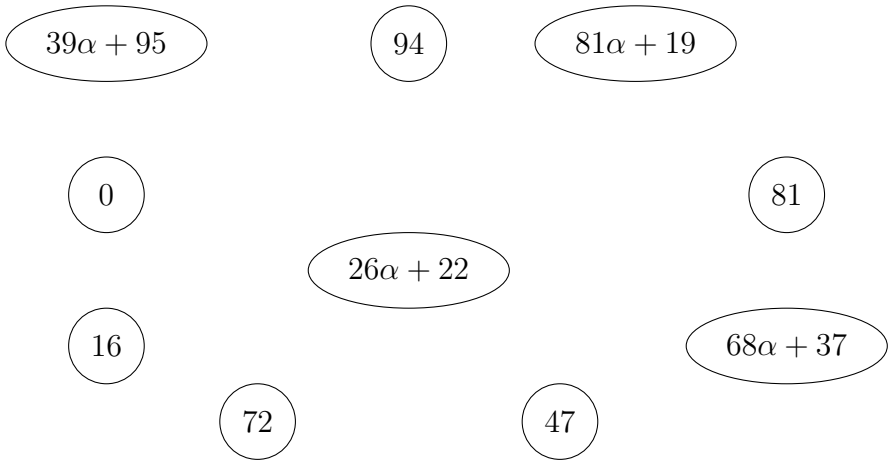


Figura 4. Grafo sin aristas, hecho por el autor.

```sage
Aristas2=[]
s=len(L)
for i in range(s):
   for k in range(i,s):
      if Phi2(L[i],L[k])==0:
         Aristas2.append((L[i],L[k]))
print(' --- Aristas de 2-isogenias ---')
print(Aristas2)
print(' ')
```

```
Salida:
    --- Aristas de 2-isogenias ---
    [(16, 16), (16, 94), (94, 68*a + 37), (94, 39*a + 95),
↪ (68*a + 37, 81), (68*a + 37, 26*a + 22), (39*a + 95, 81), (39*a
↪ + 95, 81*a + 19), (81, 72), (26*a + 22, 81*a + 19), (26*a + 22,
↪ 47), (81*a + 19, 47), (72, 47), (72, 0)]
```

Hecho por el autor en Sagemath.

Vemos que la salida es una lista de tuplas, las cuales representan los nodos que deben estar conectados por una arista (2-isogenia). El grafo quedaría como se muestra en la figura 5.

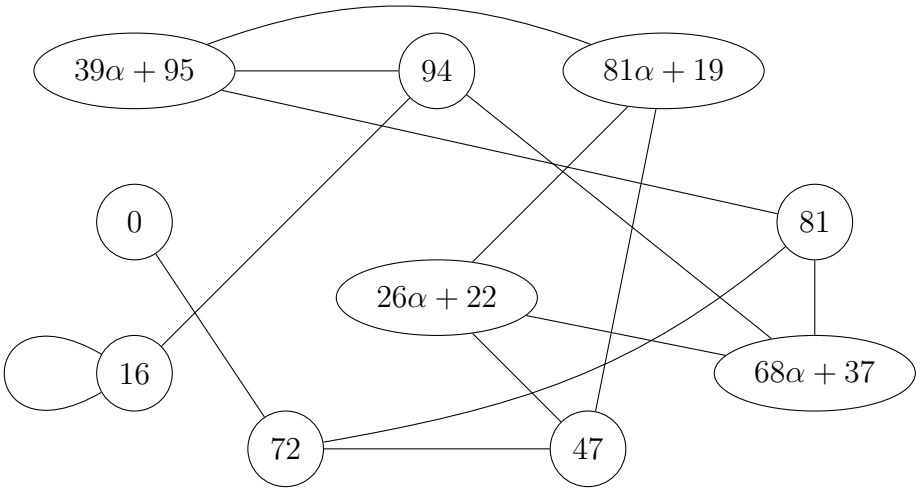


Figura 5. Grafo de 2-isogenias, , hecho por el autor.

Procedemos similarmente para calcular las aristas del grafo de 3-isogenias.

```
Aristas2=[]
s=len(L)
for i in range(s):
    for k in range(i,s):
        if Phi3(L[i],L[k])==0:
            Aristas2.append((L[i],L[k]))
print('--- Aristas de 3-isogenias ---')
print(Aristas2)
print(' ')
```

```
Salida:
--- Aristas de 3-isogenias ---
[(16, 72), (16, 47), (94, 81), (94, 26*a + 22), (94, 81*a +
↪ 19), (94, 0), (68*a + 37, 39*a + 95), (68*a + 37, 81*a + 19),
↪ (68*a + 37, 72), (39*a + 95, 26*a + 22), (39*a + 95, 72), (81,
↪ 81), (81, 47), (26*a + 22, 81*a + 19), (26*a + 22, 47), (81*a +
↪ 19, 47), (72, 72), (0, 0)]
```

Hecho por el autor en Sagemath.

El grafo 3-isogenias quedaría como se muestra en la figura 6.

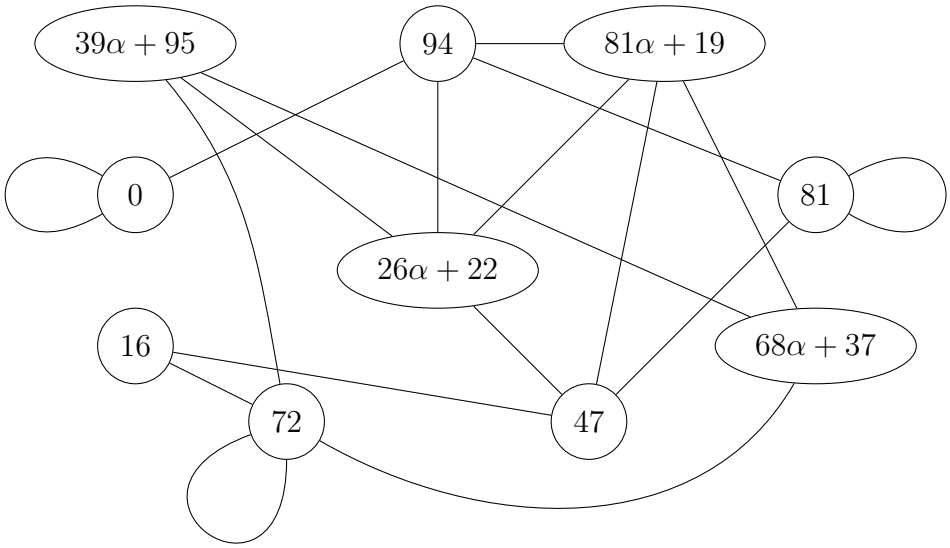


Figura 6. Grafo de 3-isogenias, hecho por el autor.

Observación 3.1.9.

- *Aquí hay nodos en los cuales se observan menos aristas de las que deberían tener, por ejemplo en la figura 6 en el nodo $39\alpha + 95$ en el cual es visible 3 aristas. Sin embargo, por el teorema 3.1.6 cada nodo debe tener 4 aristas. La respuesta a esta observación es que en alguna de las 3 aristas que se muestran en el dibujo debe ser una arista doble. Es por estas observaciones que en los grafos de ℓ-isogenias no se grafica la orientación de las aristas.*

- *Otra observación que se puede hacer, es por ejemplo en el nodo 0 del grafo 2-isogenias en la figura 5, en el cual se observa una sola arista. Esta inconsistencia se debe al isomorfismo de esa clase de equivalencia, sin embargo, en el protocolo SIDH se ignoran estas clases de equivalencia.*

3.2. Caminando sobre del grafo

En esta sección hablaremos sobre cómo "caminar a través del grafo", pero primero un teorema.

Teorema 3.2.1. *([11]) Sea E una curva elíptica supersingular sobre un cuerpo de característica $p > 3$. Entonces se sigue que*

- *Si E está definida sobre $\mathbb{F}_p$ entonces $\#E(\mathbb{F}_p) = p + 1$.*
- $E(\mathbb{F}_{p^2}) \cong (\mathbb{Z}/(p-1)\mathbb{Z})^2$ *o* $E(\mathbb{F}_{p^2}) \cong (\mathbb{Z}/(p+1)\mathbb{Z})^2$.

Hablemos y fundamentemos una pequeña parte del protocolo SIDH el cual ejemplificaremos. Primeramente tomemos una curva elíptica supersingular E sobre $\mathbb{F}_{p^2}$, con $p = 2^a 3^b - 1$ siendo $a, b \in \mathbb{N}$, más adelanta mostraremos la importancia de tomar p de esta manera. Además $E(\mathbb{F}_{p^2})$ debe cumplir $E(\mathbb{F}_{p^2}) \cong (\mathbb{Z}/(p+1)\mathbb{Z})^2$, remplazando $p = 2^a 3^b - 1$ tenemos lo siguiente

$$E(\mathbb{F}_{p^2}) \cong \mathbb{Z}_{2^a 3^b} \oplus \mathbb{Z}_{2^a 3^b}. \tag{7}$$

La siguiente proposición muestra que no necesitamos irnos hasta $E(\overline{\mathbb{F}_p})$ para encontrar la torsión $E[2^a 3^b]$. En particular $E[2^a]$ lo mismo sucede para $E[3^b]$, pues estas son subgrupos de $E[2^a 3^b]$.
Siguiendo esto ultimo, tenemos el siguiente resultado.

Proposición 3.2.2. $E(\mathbb{F}_{p^2}) = E[2^a 3^b]$.

Demostración. Como $2^a 3^b$ no divide a p, entonces $E[2^a 3^b] \cong \mathbb{Z}_{2^a 3^b} \oplus \mathbb{Z}_{2^a 3^b}$, luego $E[2^a 3^b] \cong E(\mathbb{F}_{p^2})$, luego solo falta probar una contenencia, para tener la igualdad. Mostremos que $E(\mathbb{F}_{p^2}) \subseteq E[2^a 3^b]$. Sea $P \in E(\mathbb{F}_{p^2})$, implica por el isomorfismo (7) tenemos que el orden de P debe dividir a $2^a 3^b$, luego $2^a 3^b P = \mathcal{O}$ y $P \in E[2^a 3^b]$, y así tenemos lo que queríamos. □

Continuando con el protocolo. Como 2 ni 3 dividen a p, entones por el teorema 2.3.5, tenemos que

$$E[2^a] \cong \mathbb{Z}_{2^a} \oplus \mathbb{Z}_{2^a},$$

$$E[3^b] \cong \mathbb{Z}_{3^b} \oplus \mathbb{Z}_{3^b}.$$

Ahora, la proposición 1.2.9 implica la existencia de P_1 y Q_1 de orden 2^a, y P_2 y Q_2 de orden 3^b tales que

$$\langle P_1, Q_1 \rangle = E\left[2^a\right] \cong \mathbb{Z}_{2^a} \oplus \mathbb{Z}_{2^a},$$

$$\langle P_2, Q_2 \rangle = E\left[3^b\right] \cong \mathbb{Z}_{3^b} \oplus \mathbb{Z}_{3^b}.$$

Note que P_1, Q_1, P_2 y Q_2 no se salen de los puntos en $E(\mathbb{F}_{p^2})$, el cual es un conjunto finito. Ahora tomando k_1 y k_2 dos enteros menores a 2^a y 3^b respectivamente, podemos calcular puntos S_1 y S_2 de la siguiente manera

$$S_1 = P_1 + k_1 Q_1,$$

$$S_2 = P_2 + k_2 Q_2.$$

Mostremos que S_1 y S_2 tienen orden 2^a y 3^b respectivamente. Sea $\alpha \in \mathbb{Z}_{2^a}$ tal que $\alpha S_1 = 0$ entonces

$$\alpha P_1 + \alpha k_1 Q_1 = 0.$$

Como P_1 y Q_1 son linealmente independientes tenemos que $\alpha = 2^a l$, con l entero, entonces el orden de S_1 es 2^a. Similarmente tenemos que S_2 es de orden 3^b.
Ahora, por el teorema 2.4.13 existe una isogenia separable $\psi_1 : E \to E_1$, donde $E_1 = E/\langle S_1 \rangle$, recordemos que la notación $E/\langle S_1 \rangle$ quiere decir que ψ_1 tiene como kernel al grupo $\langle S_1 \rangle$. Similarmente podemos calcular $\psi_2 : E \to E_2$ siendo $E_2 = E/\langle S_2 \rangle$. Hasta aquí el teorema 2.4.13 nos muestra que estos pasos se pueden hacer. Sin embargo, el tiempo que se demora un computador convencional en calcular ψ_1 y ψ_2

no es corto cuando p es grande, por esta razón es necesario una forma de calcular ψ_1 y ψ_2 más rápidamente. Una manera para lograrlo es por medio de composiciones de isogenias de grado 2 y 3 como lo menciona Craig Costello en [4]. Explicaremos este algoritmo a continuación.

Algoritmo 3.2.1. *Este algoritmo busca una isogenia separable tal que el dominio sea una curva elíptica E sobre un cuerpo K y el kernel sea $\langle R \rangle$, donde $R \in E$ y tiene orden ℓ^e, con ℓ un primo que no divide a la característica de K. El algoritmo es el siguiente*

$$E_{i+1} = E_i / \left\langle \ell^{e-i-1} R_i \right\rangle,$$
$$\psi_i : E_i \to E_{i+1},$$
$$R_{i+1} = \psi\left(R_i\right),$$
$$\text{y } E/\langle R \rangle = E_e.$$

Y así quedaría haciendo la composición de los ψ_i

$$E_0 \xrightarrow{\psi_0} E_1 \xrightarrow{\psi_1} E_2 \xrightarrow{\psi_2} \cdots \quad E_{e-1} \xrightarrow{\psi_{e-1}} E_e.$$

Por la proposición 2.4.4, sabemos que la composición de todas estas isogenias es separable, por otro lado tiene como kernel a $\langle R \rangle$. Esta última afirmación se debe a la siguiente proposición.

Proposición 3.2.3. *Sean E_0, E_1 y E_2 curvas elípticas sobre un cuerpo K de característica p. Sean $\psi_0 : E_0 \to E_1$ y $\psi_1 : E_1 \to E_2$ isogenias separables, tales que $Ker(\psi_0) = \langle P \rangle$ y $Ker(\psi_1) = \langle \psi_0(Q) \rangle$. Entonces $Ker(\Psi = \psi_1 \circ \psi_0) = \langle P, Q \rangle$.*

Demostración. Sea $R \in E_0$, tal que $\Psi(R) = \mathcal{O}_{E_2}$ si y solo si $\psi_0(R) = k\psi_0(Q)$ con k en los enteros, esto es equivalente a decir que

$$\psi_0(R) - \psi_0(kQ) = \psi_0(R - kQ) = \mathcal{O}_{E_1}.$$

Por definición de kernel del ψ_0 se tiene que $R - kQ = tP$ con t en los enteros, despejando R, $R = tP + kQ$ o $R \in \langle P, Q \rangle$. Como solo hemos utilizado equivalencias, se concluye que $Ker(\Psi) = \langle P, Q \rangle$. □

Observación 3.2.4. *Observemos que no hemos asegurado si la curva elíptica E_e, o sea el codominio, sea el mismo que el que se calcula usando el teorema 2.4.13 y eso se debe a que no siempre pasa esto. Sin embargo, es irrelevante que sean exactamente las mismas curvas, pues como veremos más adelante, lo que realmente importa es que sean isomorfas. Esto último sí se garantiza gracias al teorema 2.4.13. Por otro lado, observemos que la composición del anterior algoritmo 3.2.1 tiene grado ℓ^e.*

A continuación realicemos un ejemplo usando $p = 2^2 3^3 - 1 = 107$.

Ejemplo 3.2.5. *(Caminamos a través del grafo de 2-isogenias) Usando $p = 107$ y tomando la curva elíptica $E(\mathbb{F}_{p^2})$, definida de la siguiente manera*

$$E : y^2 = x^3 + (24\alpha + 33)x + (59\alpha + 42).$$

Usando SageMath determinemos si E es supersingular y cual es su j-invariante.

```
E=EllipticCurve(GF(107^2,"a"),["24*a + 33","59*a+42"])

print(f"¿E es supersingular?= {E.is_supersingular()}")
print("")
print(f"¿Cuál es el j-invariante de E?= {E.j_invariant()}")
```

```
Salida:
    ¿E es supersingular?= True

    ¿Cuál es el j-invariante de E?= 81*a + 19
```

Hecho por el autor en Sagemath.

Tomemos dos puntos P y Q tales que sean base de $E[2^2]$. Para esto usaremos algunas librerías en SageMath, las cuales no profundizaremos aquí, pero si comprobaremos que P y Q son de orden 4 y generan a todo el grupo de torsión. Para hacer esto último basta ver que el tamaño del grupo generado por P y Q sea 16.

```
Ta=E.torsion_polynomial(2^2).roots(multiplicities=0)
Tb=[Ta[4],Ta[5]]
E_torsion=[E.lift_x(a) for a in Tb ]
P=E_torsion[0]
Q=E_torsion[1]

print(f"P={P}")
print(f"Q={Q}")
print("")
print(f"Orden de P= {P.order()}")
print(f"Orden de Q= {P.order()}")
```

```
Salida:
    P=(69*a + 52 : 20*a + 3 : 1)
    Q=(51*a + 28 : 77*a + 65 : 1)

    Orden de P= 4
    Orden de Q= 4
```

Hecho por el autor en Sagemath.

Aquí tomamos

$$P = (69\alpha + 52, 20\alpha + 3)$$

$$Q = (51\alpha + 28, 77\alpha + 65)$$

La forma en la que SageMath imprime estos puntos, es la notación que fué explicada en los preliminares. Para la simplicidad de este ejemplo sólo tomaremos los puntos finitos. Ahora, observemos que el tamaño del grupo generado por P y Q es 16. Para

esto usamos el hecho de que 2^2 no es un número muy grande y así hacer todas las posibles combinaciones lineales de P y Q, para decidir si precisamente estos puntos me generan a 16 puntos distintos.

```
v=[]
cont=1
for i in range(2^2):
    for j in range(2^2):
        v.append(i*P+j*Q)
f=set(v)
print(f"¿Cuántos elementos distintos genera P y Q? = {len(set(v))}")


 Salida:

     ¿Cuántos elementos distintos genera P y Q? = 16
```

Hecho por el autor en Sagemath.

Ahora tomemos un k menor que 2^2, para este caso tomemos $k = 1$ y observemos si $S = P + Q = (84\alpha + 7, 35\alpha + 106)$ tiene orden 2^2, así

```
S=P+Q
print(S)
print(F"Orden de S es: {S.order()}")


 Salida:
     (84*a + 7 : 35*a + 106 : 1)
     Orden de S es: 4
```

Hecho por el autor en Sagemath.

Ahora falta calcular la isogenia ψ tal que el dominio sea E y el kernel $\langle S \rangle$. Para esto basta con escribir una línea de código en SageMath, la cual es $Psi = E.isogeny([S])$ y entonces la variable "Psi" es la isogenia ψ que queríamos. Sin embargo, mostraremos con SageMath el algoritmo 3.2.1 antes visto para llegar a esta isogenia. En las siguientes líneas de código observaremos el codominio de la isogenia ψ (el cual debe ser otra curva elíptica) y su j-invariante.

```
Psi = E.isogeny([S])

E_A=Psi.codomain()
j_A=E_A.j_invariant()

print(' La isogenia secreta de Alicia es:')
print(' ---  '"{}".format(Psi))
print("")
print('El j-invariante de la curva de llegada= '"{}".format(j_A))
```

```
Salida:
    La isogenia secreta de Alicia es:
    ---  Isogeny of degree 4 from Elliptic Curve defined by y^2
↪ = x^3 + (24*a+33)*x + (59*a+42) over Finite Field in a of size
↪ 107^2 to Elliptic Curve defined by y^2 = x^3 + (106*a+42)*x +
↪ (13*a+51) over Finite Field in a of size 107^2

El j-invariante de la curva de llegada= 72
```

Hecho por el autor en Sagemath.

Mostremos las imágenes $\psi(P)$ y $\psi(Q)$ bajo la isogenia ψ:

$$\psi(P) = (8\alpha + 1, 83\alpha + 45),$$

$$\psi(Q) = (8\alpha + 1, 24\alpha + 62).$$

```
1 Psi_P=Psi(P)
2 Psi_Q=Psi(Q)
3
4 print(f"imagen de P= {Psi_P}")
5 print(f"imagen de Q= {Psi_Q}")


    Salida:
        imagen de P= (8*a + 1 : 83*a + 45 : 1)
        imagen de Q= (8*a + 1 : 24*a + 62 : 1)
```

Hecho por el autor en Sagemath.

En las siguiente líneas de código aplicaremos el algoritmo 3.2.1 en SageMath. La función definida en la línea 1 del próximo código, recrea el algoritmo mencionado, y podemos ver en la salida los saltos entre las curvas, en particular los j-invariantes de estas curvas, para así llegar al mismo j-invariante que llega SageMath con el método $.isogeny()$. Cabe mostrar que aquí además de llegar a la misma clase del j-invariante, llegamos a la misma curva elíptica. Esto es gracias a que el primo $p = 107$ es pequeño, lo que implica que es muy probable este tipo de situaciones, pero si p fuere muy grande, este comportamiento no tiene por qué repetirse.

```
def camino_isogenias(E0,SA):
    (pA,eA)=factor(SA.order())[0]
    j0= E0.j_invariant()
    Curv=[(E0,j0)]
    for i in range(eA):
        l=len(Curv)
        RA=(pA^(eA-1-i))*SA
        Ef=Curv[l-1][0].isogeny(RA).codomain()
        SA=Curv[l-1][0].isogeny(RA)(SA)
        Curv.append((Ef,Ef.j_invariant()))
    return Curv
print('--- El camino que sigue en el grafo es:')
L2=camino_isogenias(E,S)
cont=0
for i in L2:
    print("")
    print(f"Curva E_{cont}: E={i[0]} y el j-invariante es:{i[1]} " )
```

```
Salida:
--- El camino que sigue en el grafo es:
    Curva E_0: E=Elliptic Curve defined by y^2 = x^3 +
↪ (24*a+33)*x + (59*a+42) over Finite Field in a of size 107^2 y
↪ el j-invariante es:81*a + 19
    Curva E_1: E=Elliptic Curve defined by y^2 = x^3 +
↪ (24*a+62)*x + (24*a+53) over Finite Field in a of size 107^2 y
↪ el j-invariante es:47
    Curva E_2: E=Elliptic Curve defined by y^2 = x^3 +
↪ (106*a+42)*x + (13*a+51) over Finite Field in a of size 107^2 y
↪ el j-invariante es:72
```


Hecho por el autor en Sagemath.

Los saltos se pueden ver de la siguiente manera

$$E_0 \xrightarrow{\psi_0} E_1 \xrightarrow{\psi_1} E_2,$$

donde ψ_0 y ψ_1 son las 2 isógenas, $E_0 = E$ y $E_2 = E/\langle S \rangle$. Para un mejor entendimiento de los saltos de las 2-isógenas a través del grafo de j-invariantes supersingulares, podemos ver en el grafo 7 , donde el nodo verde es el j-invariante donde se empezó y el nodo amarillo el j-invariante donde terminó. Además, el camino que se recorrió son las aristas de color azul siendo ψ_0 la primer 2-isogenia y ψ_1 la segunda. Y así tenemos que la compuesta de ψ_0 y ψ_1 es una isogenia de grado 2^2 separable y el kernel como $\langle S \rangle$.

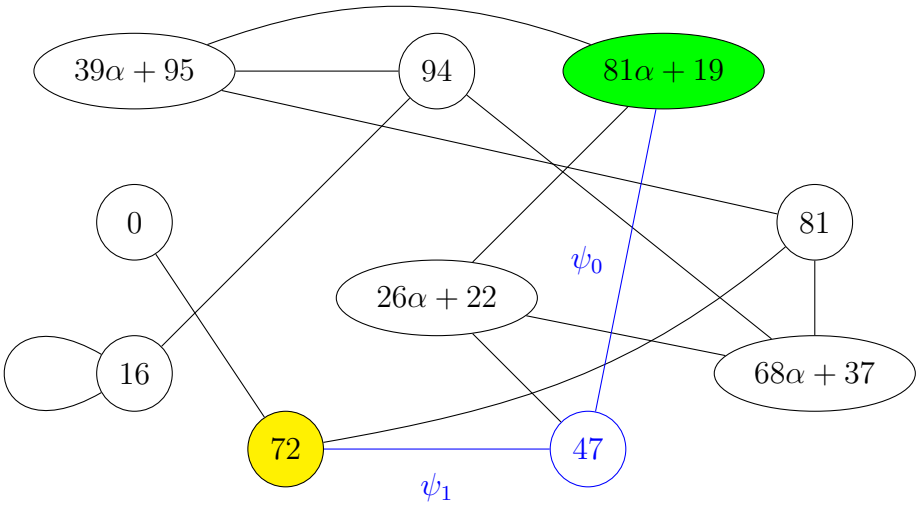


Figura 7. Camino sobre el grafo de 2-isogenias, hecho por le autor.

Y así caminamos a través del grafo de 2-isogenias.

Observación 3.2.6. *Cabe aclarar que si queremos caminar sobre el grafo de 3-isogenias, el procedimiento es similar, lo que va a cambiar es que el P y el Q ahora tienen grado 3^3. Siguiendo el ejemplo anterior, pues $p = 2^2 3^3 - 1 = 107$.*

4. Protocolo SIDH y su seguridad

En este capítulo hablaremos sobre el protocolo SIDH y utilizaremos SageMath para mostrar su aplicación.

Observación 4.0.1. *Antes de empezar es necesario definir cierta terminología en lo que viene del trabajo. Primeramente como fue mencionado y detallado en los preliminares, este protocolo criptográfico es un protocolo de intercambio de llaves, esto quiere decir que no se debe hablar de emisor y receptor, en cambio se debe hablar de dos partes. Para facilitar la escritura hablaremos de Alice y Bob, los cuales serán las dos partes antes mencionadas, la A será la etiqueta para Alice y B la etiqueta de Bobo. A lo largo del capítulo definiremos elementos a cada uno de las partes, . Por ejemplo, si un número primo es de Alice, entonces es posible que se denote como p_A y si es de Bob, puede que sea p_B.*

4.1. SIDH

A continuación hablaremos del protocolo SIDH, el cual es meramente teórico, es decir, el fin de este protocolo no es ser rápido computacionalmente (el protocolo basado en SIDH que busca la rapidez computacional es *Supersingular Isogeny Key Encapsulation (SIKE)*), SIDH se enfoca en asegurar matemáticamente la existencia de algunos puntos o veracidad del procedimiento. La fuente de este protocolo es [4] y el autor es Craig Costello.

Protocolo 4.1.1. *(SIDH [4])*

- ***Parámetros públicos*** (E, P_A, Q_A, P_B, Q_B)*:*
  - *Consideremos un primo $p = 2^{e_A}3^{e_B} - 1$, con $2^{e_A} \approx 3^{e_B}$, e_A y e_B enteros. La notación e_A y e_B, se debe a que el primero lo va a usar Alice y el segundo*

Bob. Este primo p es público.

- *E es una curva elíptica supersingular sobre $\mathbb{F}_{p^2}$ con*

$$E\left(\mathbb{F}_{p^2}\right) \cong (\mathbb{Z}/(p+1)\mathbb{Z})^2 = \mathbb{Z}/(2^{e_A}3^{e_B})\mathbb{Z} \oplus \mathbb{Z}/(2^{e_A}3^{e_B})\mathbb{Z}.$$

Se puede encontrar esta curva elíptica gracias al Teorema 3.2.1, además esta curva cumple que

$$\#E(\mathbb{F}_{p^2}) = (2^{e_A}3^{e_B})^2.$$

- *Los puntos P_A, Q_A, P_B y Q_B, son aquellos tales que*

$$\langle P_A, Q_A \rangle = E\left[2^{e_A}\right],$$
$$\langle P_B, Q_B \rangle = E\left[3^{e_B}\right].$$

Estos puntos existen gracias a lo expuesto en la sección 3.2. Además, P_A y Q_A son elegidos por Alice, y P_B junto a Q_B por Bob.

- ***Generación de claves públicas y secretas:***

  - ***Alice:** Alice escoge un número entero aleatorio k_A menor que 2^{e_A} y calcula*

$$S_A = P_A + k_A Q_A.$$

Por lo expuesto en la sección 3.2, S_A tiene orden 2^{e_A}.

    - *$\langle S_A \rangle$ es llamado subgrupo secreto de Alice.*
    - *La isogenia $\psi_A : E \to E_A$, donde $E_A = E/\langle S_A \rangle$, es llamada la isogenia secreta de Alice.*
    - ***Clave secreta de Alice:** k_A*
    - ***Clave pública de Alice:** $(E_A, \psi_A(P_B), \psi_A(Q_B))$*

- ***Bob:*** *De manera similar a Alice, Bob escoge un número entero aleatorio* k_B *menor que* 3^{e_B} *y calcula*

$$S_B = P_B + k_B Q_B.$$

  S_B *tiene orden* 3^{e_B}*.*

  - $\langle S_B \rangle$ *es llamado subgrupo secreto de Bob.*
  - *La isogenia* $\psi_B : E \to E_B$*, donde* $E_B = E/\langle S_B \rangle$*, es llamada la isogenia secreta de Bob.*
  - ***Clave secreta de Bob:*** k_B
  - ***Clave pública de Bob:*** $(E_B, \psi_B(P_A), \psi_B(Q_A))$

- ***Secreto compartido***

  - ***Alice:*** *Alice calcula* $\psi_B(S_A)$*, así:*

$$\psi_B(S_A) = \psi_B\left(P_A + k_A Q_A\right) = \psi_B\left(P_A\right) + k_A \psi_B\left(Q_A\right).$$

    Esto gracias a que ψ_B *es una isogenia. Además, calcula*

$$\Psi_A : E_B \to E_B/\langle \psi_B(S_A) \rangle := E_{AB}.$$

  - ***Bob:*** *Análogamente Bob calcula* $\psi_A(S_B)$*, así:*

$$\psi_A(S_B) = \psi_A\left(P_B + k_B Q_B\right) = \psi_A\left(P_B\right) + k_B \psi_A\left(Q_B\right)$$

    Esto gracias a que ψ_A *es una isogenia. Además calcula*

$$\Psi_B : E_A \to E_A/\langle \psi_A(S_B) \rangle := E_{BA}.$$

Gracias a que la composición de isogenias separables es separable y a la proposición 3.2.3, tenemos que

$$E_{AB} \cong E_{BA}.$$

Entonces, el secreto compartido es el j-invariante de estas curvas elípticas.

En la figura 8 se puede observar el diagrama del protocolo SIDH. Además, comparándolo con el diagrama de DH 3 es fácil ver similitudes, de aquí viene la explicación de los nombre Diffie y Hellman en el protocolo SIDH.

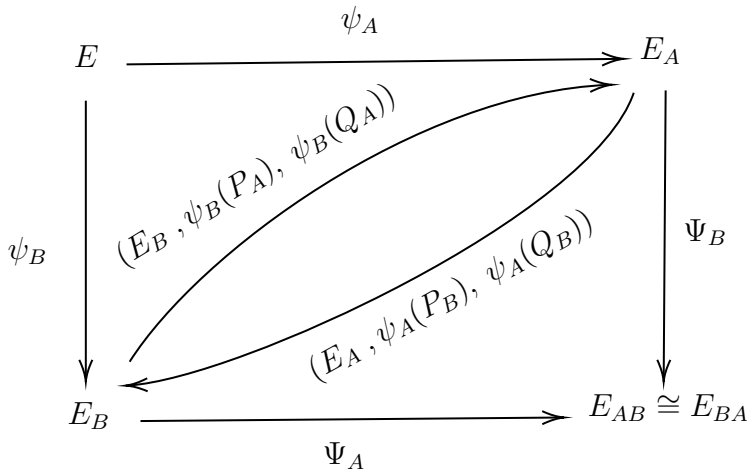


Figura 8. Diagrama del protocolo SIDH, hecho por el autor.

Nota 4.1.1. *Notemos que el protocolo no nos asegura la igualdad entre E_{AB} y E_{BA}, pues los fundamentos matemáticos no lo hacen. Como veremos en el siguiente ejemplo, comprobaremos que estas curvas no tienen porque ser iguales. Los pasos que realicemos a través del grafo, es gracias al algoritmo 3.2.1.*

Ejemplo 4.1.2.

- ***Parámetros públicos:***

- *Tomemos el primo de la forma*

$$p = 2^2 3^3 - 1 = 107,$$

donde $e_A = 2$ *y* $e_B = 3$.

- *Tomemos a* E *como la curva* $E : y^2 = x^3 + (24\alpha + 33)x + (59\alpha + 42)$. *Gracias al ejemplo 3.2.5, sabemos que* E *es supersingular y su* j*-invariante es* $81\alpha + 19$. *A continuación verifiquemos que* $\#E(\mathbb{F}_{p^2}) = (2^2 3^3)^2 = 11664$, *con la siguiente línea de código de SageMath.*

```
print(f"#E= {E.cardinality()}")
```

Salida:

#E= 11664

Hecho por el autor en Sagemath.

- *Ahora Alice y Bob toman los siguientes puntos respectivamente*

$$P_A = (69\alpha + 52, 20\alpha + 3) \quad y \quad Q_A = (51\alpha + 28, 77\alpha + 65),$$

$$P_B = (48\alpha + 8, 55\alpha + 85) \quad y \quad Q_B = (26\alpha + 89, 98\alpha + 77),$$

Para los cuales $E[2^2] = \langle P_A, Q_A \rangle$ *y* $E[3^3] = \langle P_B, Q_B \rangle$.

- ***Generación de claves públicas y secretas:***

- ***Alice:*** *Alice toma* k_A *como* $k_A = 1$, *el cual es menor que* 2^2 *y ahora calcula*

S_A. *Por el ejemplo 3.2.5, tenemos que*

$$S_A = P_A + k_A Q_A = (84\alpha + 7, 35\alpha + 106).$$

Además, se tiene ψ_A *descrita de la siguiente manera*

$$\psi_A : E \to E_A,$$

donde $E_A : y^2 = x^3 + (106\alpha + 42)x + (13\alpha + 51)$*, la cual tiene* j*-invariante* 72*. A continuación calculamos* $\psi_A(P_B)$ *y* $\psi_A(Q_B)$*,*

```
1  Psi_A= E.isogeny([S_A])
2
3  print(f"Psi_A(P_B)= {Psi_A(P_B)}")
4  print(f"Psi_A(Q_B)= {Psi_A(Q_B)}")


Salida:
    Psi_A(P_B)= (16*a + 38 : 20*a + 54 : 1)
    Psi_A(Q_B)= (8*a + 73 : 43*a + 54 : 1)
```

Hecho por el autor en Sagemath.

Obtenemos

$$\psi_A(P_B) = (16\alpha + 38, 20\alpha + 54) \quad \textit{y} \quad \psi_A(Q_B) = (8\alpha + 73, 43\alpha + 54).$$

Así tenemos la clave pública de Alice, la cual debe pasar a Bob:

$$(E_A, (16\alpha + 38, 20\alpha + 54), (8\alpha + 73, 43\alpha + 54)).$$

La clave privada de Alice es $k_A = 1$, y el camino que recorrió Alice en el grafo de 2-isogenias es:

```
Alice_1=camino_isogenias(E,S_A)
print("Saltos de Alice del grafo 2-isogenas : ", end=" ")
for i in Alice_1:
    if i!= Alice_1[-1]:
        print(f"{i[1]}  -->  ", end="")
    else:
        print(f"{i[1]}")


Salida:
    Saltos de Alice del grafo 2-isogenas :  81*a + 19  -->  47  -->
↪  72
```

Hecho por el autor en Sagemath.

Gráficamente se vería así

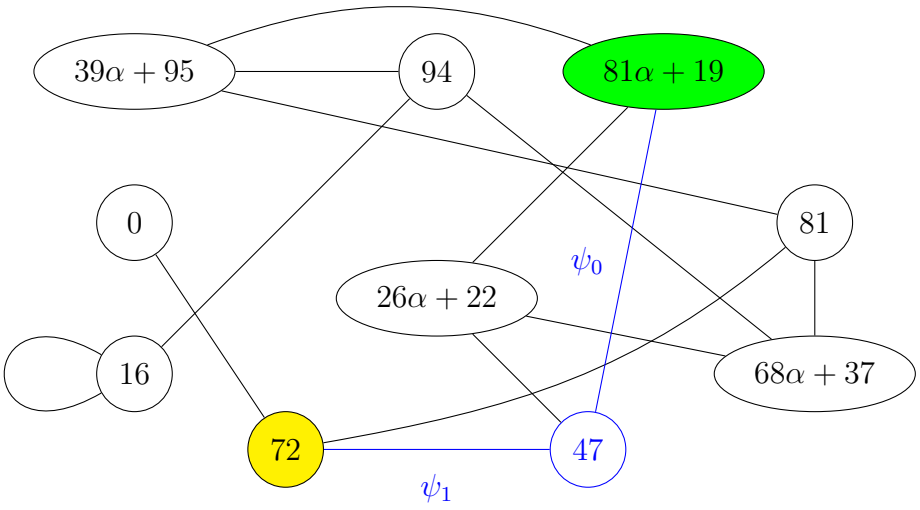


Figura 9. Primer salto de Alice en el grafo, hecho por el autor.

Donde ψ_0 y ψ_1 son las isogenias que hacen el salto.

- ***Bob:*** *Bob toma $k_B = 1$, y calcula S_B. Usando SageMath calcularemos la isogenia ψ_B similarmente como se hizo con Alice, junto a los puntos $\psi_B(P_A)$ y $\psi_B(P_B)$, de la siguiente manera.*

```
S_B=P_B+Q_B

Psi_B= E.isogeny([S_B])
E_B=Psi_B.codomain()
j_B=E_B.j_invariant()

print(f"Codominio de Psi_B: {E_B} ")
print(f"j-invariante de E_B: {j_B}")
print()
print(f"Psi_B(P_A)= {Psi_B(P_A)}")
print(f"Psi_B(Q_A)= {Psi_B(Q_A)}")


Salida:
    Codominio de Psi_B: Elliptic Curve defined by y^2 = x^3 +
↪   (5*a+57)*x over Finite Field in a of size 107^2
    j-invariante de E_B: 16

    Psi_B(P_A)= (56*a + 103 : 32*a + 30 : 1)
    Psi_B(Q_A)= (45*a + 10 : 94*a + 18 : 1)
```

Hecho por el autor en Sagemath.

Entonces tenemos que ψ_B *es*

$$\psi_B : E \to E_B,$$

donde $E_B : y^2 = x^3 + (5\alpha + 57)x$*, la cual tiene como* j*-invariante 16.*

Entonces la clave secreta de Bob es $k_B = 1$ y la clave pública es

$$(E_B, (56\alpha + 103, 32\alpha + 30), (45\alpha + 10, 94\alpha + 18)).$$

Ahora calculamos los saltos de Bob de la siguiente manera:

```
1 Bob_1=camino_isogenias(E,S_B)
2 print("Saltos de Bob del grafo 2-isogenas : ", end=" ")
3 for i in Bob_1:
4     if i!= Bob_1[-1]:
5         print(f"{i[1]}  -->  ", end="")
6     else:
7         print(f"{i[1]}")


Salida:
    Saltos de Bob del grafo 2-isogenas :  81*a + 19  -->  68*a + 37
↪  -->  72  -->  16
```

Hecho por el autor en Sagemath.

Gráficamente se vera así:

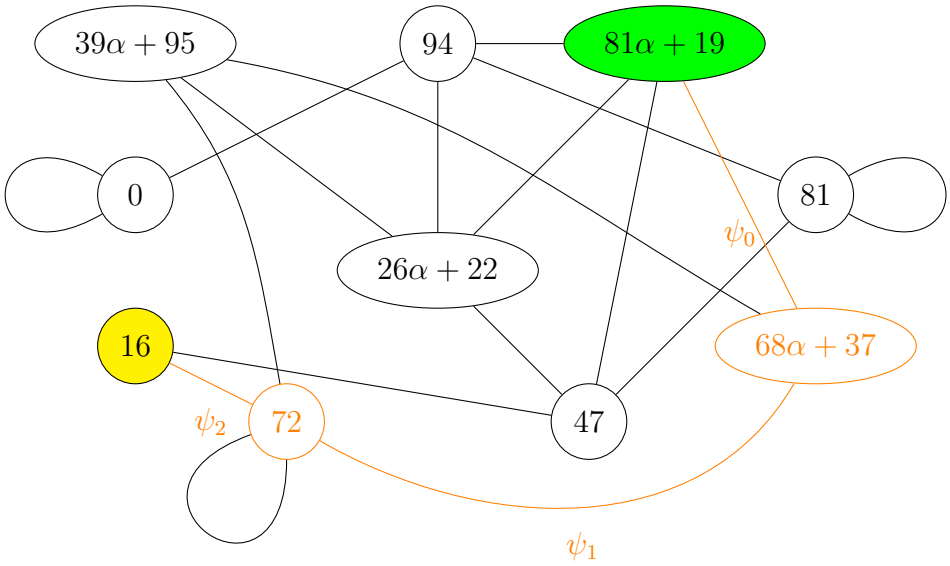


Figura 10. Primer salto de Bob en el grafo, hecho por el autor.

donde ψ_i son las isogenias que hacen el salto.

- ***Secreto Compartido:***

  - ***Alice:*** *Alice calcula $\psi_B(S_A)$,*

$$\psi_B(S_A) = \psi_B(P_A + k_A Q_A) = \psi_B(P_A) + k_A \psi_B(Q_A) = (62\alpha + 97, 17\alpha + 48),$$

y la respectiva isogenia Ψ_A, que estará definida como

$$\Psi_A : E_B \to E_{AB},$$

siendo $E_{AB} : y^2 = x^3 + (73\alpha + 1)x + (73\alpha + 99)$, la cual tiene j-invariante $68\alpha + 37$.

```
Phi_SA=Psi_B(P_A)+Psi_B(Q_A)

Psi_A_A = E_B.isogeny([Phi_SA])
E_AB=Psi_A_A.codomain()
j_A=E_AB.j_invariant()

print(f"La imagen de S_A: {Phi_SA}")
print()
print(Psi_A_A)
print()
print("-----")
print(f"j-invariante del codominio: {j_A}")
```

```
Salida:
    La imagen de S_A: (62*a + 97 : 17*a + 48 : 1)

    Isogeny of degree 4 from Elliptic Curve defined by y^2 = x^3 +
↪   (5*a+57)*x over Finite Field in a of size 107^2 to Elliptic
↪   Curve defined by y^2 = x^3 + (73*a+1)*x + (73*a+99) over Finite
↪   Field in a of size 107^2

    -----
    j-invariante del codominio: 68*a + 37
```

Hecho por el autor en Sagemath.

Veamos los saltos que haría Alice con el algoritmo sobre el grafo de 2-

isogenias,

```
Alice_2=camino_isogenias(E_B,Phi_SA)
print("Saltos de Alice del grafo 2-isogenas : ", end=" ")
for i in Alice_2:
    if i!= Alice_2[-1]:
        print(f"{i[1]}  -->  ", end="")
    else:
        print(f"{i[1]}")


Salida:
    Saltos de Alice del grafo 2-isogenas :  16  -->  94  -->  68*a
↪  + 37
```

Hecho por el autor en Sagemath.

Gráficamente

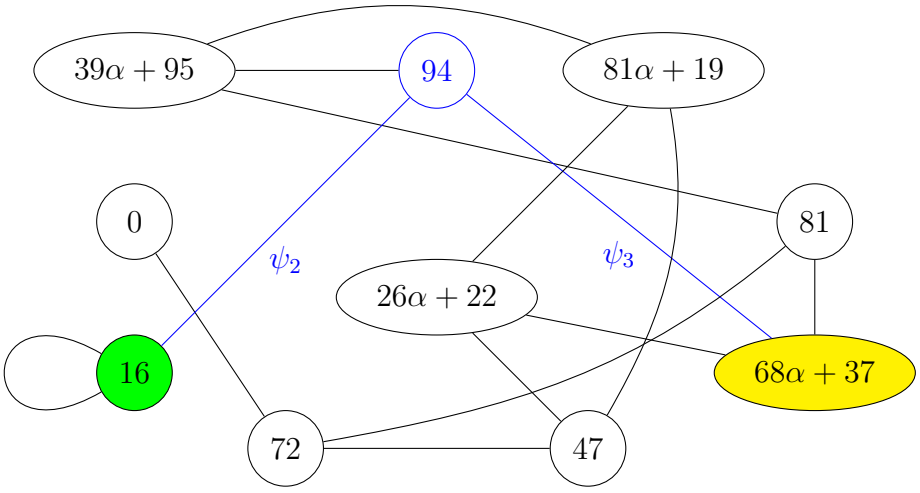


Figura 11. Segundo salto de Alice en el grafo, hecho por el autor.

donde ψ_2 y ψ_3, las isogenias de grado 2 que hacen los saltos.

- ***Bob:*** *Bob calcula $\psi_A(S_B)$,*

$$\psi_A(S_B) = \psi_A(P_B + k_B Q_B) = \psi_A(P_B) + k_B \psi_A(Q_B) = (51\alpha + 79, 85\alpha + 51)$$

y la respectiva isogenia Ψ_B, que estará definida como

$$\Psi_B : E_A \to E_{BA},$$

siendo $E_{BA} : y^2 = x^3 + (73\alpha + 1)x + (73\alpha + 99)$ la cual tiene j-invariante $68\alpha + 37$.

```
Phi_SB=Psi_A(P_B)+Psi_A(Q_B)

Psi_B_B = E_A.isogeny([Phi_SB])

E_BA=Psi_B_B.codomain()
j_B=E_BA.j_invariant()

print(f"La imagen de S_B: {Phi_SB}")
print()
print(Psi_B_B)
print()
print("-----")
print(f"j-invariante del codominio: {j_B}")
```

```
Salida:
    La imagen de S_B: (51*a + 79 : 85*a + 51 : 1)

    Isogeny of degree 27 from Elliptic Curve defined by y^2 = x^3 +
↪   (106*a+42)*x + (13*a+51) over Finite Field in a of size 107^2
↪   to Elliptic Curve defined by y^2 = x^3 + (73*a+1)*x + (73*a+99)
↪   over Finite Field in a of size 107^2

    -----
    j-invariante del codominio: 68*a + 37
```

Hecho por el autor en Sagemath.

Ahora veamos los saltos que daría Bob en el grafo de 3-isogenias,

```
Bob_2=camino_isogenias(E_A,Phi_SB)
print("Saltos de Alice del grafo 2-isogenas : ", end=" ")
for i in Bob_2:
    if i!= Bob_2[-1]:
        print(f"{i[1]}  -->  ", end="")
    else:
        print(f"{i[1]}")


Salida:
    Saltos de Alice del grafo 2-isogenas :  72  -->  72  -->  39*a
↪  + 95  -->  68*a + 37
```

Hecho por el autor en Sagemath.

Gráficamente quedaría así:

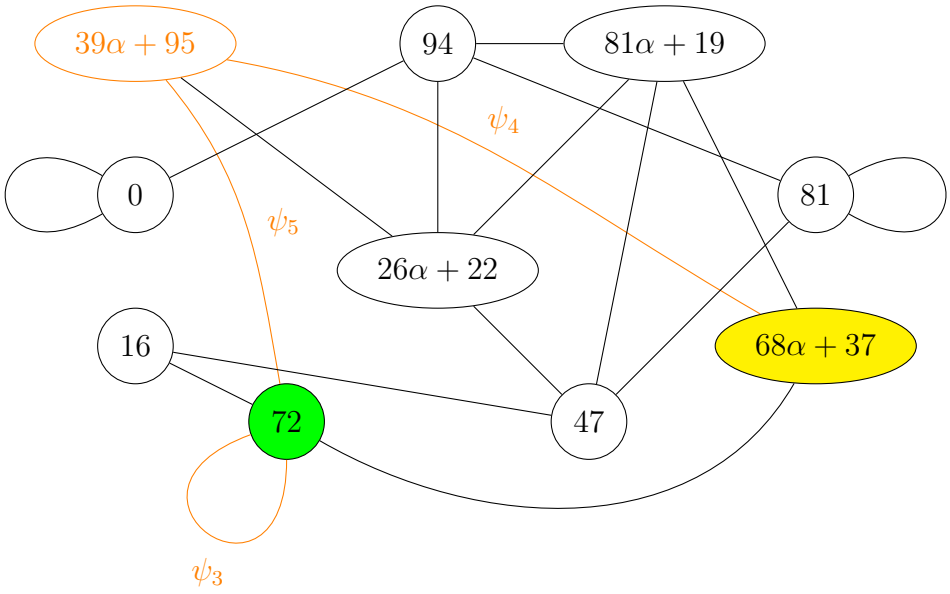


Figura 12. Segundo salto de Bob en el grafo, hecho por el autor.

donde ψ_j son las isogenias de grado 3, sobre el grafo de 3-isogenias.

Finalmente, podemos observar que ambos llegan a una curva (distinta para ambos) tal que su j-invariante es el mismo. O sea, su secreto compartido es $68\alpha + 37$, el cual pueden utilizarlo como clave para algún protocolo criptográfico simétrico.

4.2. Seguridad y criptoanálisis de SIDH

En esta sección discutiremos acerca de la seguridad de SIDH, junto a las últimas noticias sobre este protocolo criptográfico en el año 2022.
El punto clave de un protocolo de intercambio de llaves es que los parámetros públicos no den información acerca de los parámetros privados. En SIDH los datos públicos son los siguientes:

- Parámetros públicos: (E, P_A, Q_A, P_B, Q_B)
- Clave pública de Alice: $(E_A, \psi_A(P_B), \psi_A(Q_B))$

- Clave pública de Bob: $(E_B, \psi_B(P_A), \psi_B(Q_A))$

Las claves privadas son:

- Clave privada de Alice: $(k_A, \psi_A, \langle S_A \rangle)$
- Clave privada de Bob: $(k_B, \psi_B, \langle S_B \rangle)$

Llamemos Cooper a la parte distinta de Alice y Bob, y quien quiere conocer la clave secreta de Alice o la de Bob. Cooper conoce todo lo público entre la comunicación entre Alice y Bob, en particular conoce (E, P_A, Q_A) los cuales hacen parte de los parámetros públicos y la clave pública de Bob $(E_B, \psi_B(P_A), \psi_B(Q_A))$. Entonces, si Cooper es capaz de calcular el kernel de ψ_B de alguna manera, entonces puede calcular ψ_B, es decir, Cooper conocería la clave privada de Bob.

Definición 4.2.1. *Al problema de calcular el kernel de ψ_B, le llamaremos el **problema de SIDH**.*

Observación 4.2.2. *Cabe observar que el problema de SIDH se puede reescribir para el caso de calcular el kernel de ψ_A.*

4.2.1. Lo que se creía: Antes del año 2022, no existía un algoritmo ni clásico, ni cuántico que pudiera resolver el problema de SIDH en tiempo polinomial, incluso se conjeturaba que no existía este algoritmo, como lo mencionan Stiven Galbraith y Frederik Vercauteren en el año 2018 en el artículo [13]. Esta conjetura era cada vez más aceptada gracias a diversos ataques que recibió SIDH, incluso años después del artículo, por ejemplo:

---

[13] Frederik GALBRAITH Steven D. y VERCAUTEREN. «Computational problems in supersingular elliptic curve isogenies». En: *Quantum Information Processing* 17.10 (2018), págs. 1-22.

- Galbraith, Petit, Shani, Ti (2016/859; Asiacrypt 2016) (ver [14]).
- Petit (2017/571; Asiacrypt 2017) (ver [15]).
- Quehen, Kutas, Leonardi, Martindale, Panny, Petit, Stange (2020/633; Crypto 2021) (ver [16]).
- Fouotsa and Petit (2021/1322; CT-RSA 2022) (ver [17]).

Estos trabajos no exponían una vulnerabilidad, en la cual SIKE (protocolo que aplica SIDH) tuviera que hacer grandes cambios en su implementación, o en otras palabras que fuera roto. Además, SIKE logró pasar las primeras 3 rondas que propuso la NIST para estudiar la seguridad de los protocolos post-cuánticos.

4.2.2. Lo que pasó: A finales de julio del año 2022, los matemáticos belgas Wouter Castryck y Thomas Decru, publicaron el artículo *An efficient key recovery attack on SIDH (Preliminay version)*, ver [18]. Este trabajo describe un nuevo ataque al protocolo SIKE usando el teorema de Kani del año 1997, y además explican su implementación en el lenguaje de programación Magma. Pocos días después, un

---

[14] Steven D. GALBRAITH et al. «On the Security of Supersingular Isogeny Cryptosystems». En: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, págs. 63-91.

[15] Christophe PETIT. «Faster Algorithms for Isogeny Problems Using Torsion Point Images». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, págs. 330-353.

[16] Victoria DE QUEHEN et al. «Improved Torsion-Point Attacks on SIDH Variants». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, págs. 432-470.

[17] Christophe FOUOTSA Tako Boris y PETIT. «A New Adaptive Attack on SIDH». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2022, págs. 322-344.

[18] Thomas CASTRYCK Wouter y DECRU. «An efficient key recovery attack on SIDH (preliminary version)». En: *Cryptology ePrint Archive* (2022).

computador de un solo núcleo, logró romper SIKE en cuestión de horas. A este ataque se le llama **The Castryck-Decru attack**. Escribamos el teorema de Kani. Primero la definición de **isogenia diamond de orden** 2^e.

Definición 4.2.3. *Una **isogenia diamond de orden** 2^e es una tupla (ψ, G_1, G_2) con*

1. $\psi : E \to E'$ *una isogenia;*
2. $G_1, G_2 \subset \mathrm{Ker}(\psi)$*;*
3. $G_1 \cap G_2 = \{\mathcal{O}_E\}$*;*
4. $\deg(\psi) = \#G_1 \cdot \#G_2$*;*
5. $2^e = \#G_1 + \#G_2$*.*

Teorema 4.2.4. *(Teorema de Kani [19]) Sea (ψ, H_1, H_2) una isogenia diamond de orden N, sea $d := \gcd(\#H_1, \#H_2)$, $n := N/d$, y $k_i := \#H_i/d$. Entonces $\psi = \psi' \circ [d]$ para algún $\psi' : C \to E$ y existe un único irreducible anti-simétrico $\iota : C[N] \to E[N]$ tal que*

$$\iota(k_1 R_1 + k_2 R_2) = \psi'(R_2 - R_1)$$

para todo $R_i \in [n]^{-1} H_i$, y cada reducible anti-isometrica $C[N] \to E[N]$ surge de esta manera.

A simple vista, pareciera que el Teorema de Kani no tiene relación con el problema de SIDH, pues este teorema habla sobre **variedades Abelianas**, incluso el mismo autor del teorema, el Dr. Ernst Kani, se muestra sorprendido en la conferencia dada por la universidad de Quenn, ver [20]. Sin embargo en el artículo *An efficient key*

---

[19] Ernst KANI. «The number of curves of genus two with elliptic differentials». En: *Journal für die reine und angewandte Mathematik* 1997.485 (2022), págs. 93-122.

[20] Catarina CHAGAS. *Mathematics works in serendipitous ways*. Articulo periodistico, `https://www.queensu.ca/gazette/stories/mathematics-works-serendipitous-ways`. 2022.

recovery attack on SIDH (Preliminay version) los autores muestran que el problema de SIDH se puede "forzar" a que cumpla el Teorema 4.2.4 y una vez hecho esto, poder calcular en tiempo polinomial el kernel de ψ_B. En la conferencia [21] el coautor Thomas Decru del articulo [18] explica la matemática detras de la vulnerabilidad.
En *An efficient key recovery attack on SIDH (Preliminay version)* muestran un algoritmo en Magma que rompe al protocolo SIKEp751 (implementación de SIDH y el más "fuerte") en algo más de 20 horas, sin embargo en el articulo [22] lo hacen entre 1 a 2 horas usando SageMath.
Cabe decir que el protocolo puede ser modificado para no ser vulnerable ante The Castryck-Decru attack, pero los parámetros deben ser demasiado grandes, lo que implica su inviabilidad.

4.3. Futuro de la criptografía basada en isogenias

Aunque es verdad que no hay finalistas en la NIST que se base en isogenias, sí hay sistemas criptográficos basados en isogenias que aún no se han roto y en los cuales se sigue investigando. Una lista de estos son:

- SIDH signatures
- CSIDH, SeaSign, CSI-FiSH
- OSIDH
- SQISign

---

[21] Thomas DECRU. *The Isogeny Club: #1 BREAKING SIKE*. Vídeo conferencian, `https://www.youtube.com/watch?v=mx9qNHm3mco&ab_channel=TheIsogenyClub`. 2022.

[22] Giacomo OUDOMPHENG Rémy Y POPE. «A Note on Reimplementing the Castryck-Decru Attack and Lessons Learned for SageMath». En: *Cryptology ePrint Archive* (2022).

BIBLIOGRAFÍA

CASTRYCK Wouter y DECRU, Thomas. «An efficient key recovery attack on SIDH (preliminary version)». En: *Cryptology ePrint Archive* (2022) (vid. págs. 95, 97).

CHAGAS, Catarina. *Mathematics works in serendipitous ways*. Articulo periodistico, `https://www.queensu.ca/gazette/stories/mathematics-works-serendipitous-ways`. 2022 (vid. pág. 96).

COLLINS Hugh y NAY, Chris. «IBM Unveils 400 Qubit-Plus Quantum Processor and Next-Generation IBM Quantum System Two». En: (2022). `https://newsroom.ibm.com/2022-11-09-IBM-Unveils-400-Qubit-Plus-Quantum-Processor-and-Next-Generation-IBM-Quantum-System-Two`, (vid. pág. 11).

COSTELLO, Craig. «Supersingular Isogeny Key Exchange for Beginners». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020, págs. 21-50 (vid. págs. 12, 67, 77).

DE FEO, Luca et al. «Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies». En: *Journal of mathematical cryptology* 8.3 (2014), págs. 209-247 (vid. pág. 57).

DE QUEHEN, Victoria et al. «Improved Torsion-Point Attacks on SIDH Variants». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, págs. 432-470 (vid. pág. 95).

DECKER Wolfram y PFISTER, Gerhard. *A First Course in Computational Algebraic Geometry*. New York: Cambridge University Press, 2013 (vid. pág. 12).

DECRU, Thomas. *The Isogeny Club: #1 BREAKING SIKE*. Vídeo conferencian, https://www.youtube.com/watch?v=mx9qNHm3mco&ab_channel=TheIsogenyClub. 2022 (vid. pág. 97).

FEYNMAN, Richard P. «Simulating physics with computers». En: *International journal of theoretical physics* 21.6-7 (1982), págs. 467-488 (vid. pág. 10).

FOUOTSA Tako Boris y PETIT, Christophe. «A New Adaptive Attack on SIDH». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2022, págs. 322-344 (vid. pág. 95).

GALBRAITH Steven D. y VERCAUTEREN, Frederik. «Computational problems in supersingular elliptic curve isogenies». En: *Quantum Information Processing* 17.10 (2018), págs. 1-22 (vid. pág. 94).

GALBRAITH, S. «Mathematics of public key cryptography (version 0.6)». En: *Available in: https://www.math.auckland.ac.nz/~sgal018/crypto-book/main.pdf* (2010), págs. 165-203 (vid. págs. 46, 52).

GALBRAITH, Steven D. et al. «On the Security of Supersingular Isogeny Cryptosystems». En: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, págs. 63-91 (vid. pág. 95).

KANI, Ernst. «The number of curves of genus two with elliptic differentials». En: *Journal für die reine und angewandte Mathematik* 1997.485 (2022), págs. 93-122 (vid. pág. 96).

LIDL Rudolf y NIEDERREITER, Harald. *Finite fields*. Cambridge university press, 1997 (vid. pág. 25).

Modular polynomials. página web, https://math.mit.edu/~drew/ClassicalModPolys.html (vid. pág. 58).

OUDOMPHENG Rémy Y POPE, Giacomo. «A Note on Reimplementing the Castryck-Decru Attack and Lessons Learned for SageMath». En: *Cryptology ePrint Archive* (2022) (vid. pág. 97).

PETIT, Christophe. «Faster Algorithms for Isogeny Problems Using Torsion Point Images». En: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, págs. 330-353 (vid. pág. 95).

SageMath. Lenguaje de programación, https://www.sagemath.org/ (vid. pág. 13).

SHORT, P. W. «Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer». En: *SIAM review* 41.2 (1999), págs. 303-332 (vid. pág. 10).

SILVERMAN, Joseph H. *The Arithmetic of Elliptic Curves*. 2.ª ed. New York, NY: Springer-Verlag, 2009 (vid. págs. 17, 20, 32).

SUTHERLAND, Andrew V. «Identifying supersingular elliptic curves». En: *LMS journal of computation and mathematics* 15 (2012), págs. 317-325 (vid. págs. 56, 65).

WASHINGTON, Lawrence C. *Elliptic curves: number theory and cryptography*. 2.ª ed. Discrete mathematics and its applications. CRC Press, 2008 (vid. págs. 12, 21, 22, 32, 40, 45, 54, 55, 58).