GRUPO DE TRENZAS Y SU APLICACIÓN EN CRIPTOGRAFÍA

Fernando de la Hoz Moreno

Responsables de tutorización:
Pedro A. García Sánchez
Departamento de Álgebra
Facultad de Ciencias

Jesús García Miranda
Departamento de Álgebra
ETSIIT

UNIVERSIDAD DE GRANADA

Doble Grado en Ingeniería Informática y Matemáticas
ETSIIT – Facultad de Ciencias
Universidad de Granada
Curso 2021-2022

Fernando de la Hoz Moreno *Grupo de trenzas y su aplicación en criptografía.*
Trabajo de fin de Grado. Curso académico Curso 2021-2022.

Responsable de tutorización

Pedro A. García Sánchez
Departamento de Álgebra
Facultad de Ciencias

Jesús García Miranda
Departamento de Álgebra
ETSIIT

Doble Grado en
Ingeniería Informática y
Matemáticas

Universidad de Granada

Declaración de originalidad

D./Dña. Fernando de la Hoz Moreno

Declaro explícitamente que el trabajo presentado como Trabajo de Fin de Grado (TFG), correspondiente al curso académico Curso 2021-2022, es original, entendida esta, en el sentido de que no ha utilizado para la elaboración del trabajo fuentes sin citarlas debidamente.

En Granada a 18 de noviembre de 2021

Fdo: Fernando de la Hoz Moreno

Agradecimientos

A Pedro y a Jesús, por aceptar tutorizarme este trabajo, aconsejarme y corregirme, estando siempre disponibles para cualquier duda que tuviese. A mis padres y hermana por apoyarme a lo largo de la carrera y ayudarme a no perder el impulso que me ha llevado hasta aquí. A mi familia. A mis compañeros y amigos que me han acompañado y ayudado en todo este proceso. Y por último, a los profesores por su dedicación y por todos los conocimientos aprendidos de ellos.

Índice general

Resumen

El trabajo tiene como objetivo, por un lado, comprender el grupo de trenzas y cómo poder trabajar con él, y por otro, ver cómo se aplica en métodos criptográficos y evaluar su seguridad. En un primer lugar trataremos de realizar un estudio detallado del grupo de trenzas. Veremos una serie de conceptos y resultados sobre grupos y monoides. Estos nos servirán para realizar la construcción del grupo de trenzas, la cual haremos desde un punto de vista tanto algebraico como topológico, y deduciremos que ambos son equivalentes. Por último aplicaremos las herramientas vistas sobre grupos y monoides a las trenzas para obtener propiedades que nos facilitará el trabajar con estas. Además, esta parte servirá de fundamentación matemática para la aplicación del grupo de trenzas al área de la criptografía.

En segundo lugar nos centraremos en la criptografía. Introduciremos en qué consiste la criptografía dando la definición de conceptos básicos. Esto dará pie a ver cómo podemos aplicar el grupo de trenzas a la criptografía de manera que podamos obtener distintas herramientas criptográficas. Por último veremos algunas de las debilidades que se pueden encontrar en el cifrado basado en el grupo de trenzas, e implementaremos un posible ataque haciendo uso de una heurística.

Palabras clave: Grupo de trenzas, Monoide de Garside, problema del conjugado, criptografía, esquema Anshel-Anshel-Goldfeld, esquema Diffie-Hellman, criptoanálisis.

Abstract

This project develops mainly in the area of group theory, monoid theory and cryptography. We will review both the construction of ()braids and ()groups of braids and cryptographic methods which base their security on issues related to the group of braids. Emphasis will be placed on the study of Garside monoids which will allow us to make a practical use of braids so that we can work with them from a computational point of view, making it possible to create algorithms that involve them. They will also provide us with solutions to both the conjugacy problem and the word problem.

We begin the project by reviewing basic concepts and results about groups which are necessary for the construction of the braid group, such as quotient groups or group presentations.

We introduce the definition of monoid together with two kinds of monoids, free monoids and atomic monoids, that satisfy some characteristics. These monoids allow us to work with different monoid presentations. Under the hypothesis of working with a monoid that has a weighted presentation, it provides us a solution to the word problem and the divisibility problem.

Starting from an atomic monoid M and its subset $\Sigma \in M$, we define the monoid M_Σ. With this definition we can state a theorem of great relevance, since under the hypothesis that it presupposes we obtain in M_Σ a normal form, the property of cancellation, a solution to the word problem (inherited from the one obtained with the presentations) and a solution to the conjugacy problem. Furthermore, having the concept of an exhaustive set we are able to identify the monoid M_Σ with the monoid M from which it derives.

With monoids we get a set of highly useful results. Ultimately we are going to work with the braid group and we are interested in seeing how results on groups can be obtained from those ()on monoids. Fraction groups and pre-Garside monoids are introduced for this purpose. A pre-Garside monoid is defined as a pair (M, Δ) where M is a monoid and Δ is an element of M that fulfills certain properties. This element is called Garside element. A monomorphism is established between a pre-Garside monoid and its group of fractions, so that we can identify the pre-Garside monoid as a subset of the group of fractions. Thanks to this we get a normal form for the group of fractions as

a product of a power of the Garside element and elements of the monoid. This normal form that we call the greedy normal form is what we use to work with the group of braids. Furthermore, the conjugacy problem in the fraction group can be reduced to a equivalent conjugacy problem in the monoid.

A Garside monoid is a pre-Garside monoid (M, Δ), where M is atomic and Σ is the set of divisors of Δ, fulfilling that for any two atoms s, t of M, the set

$$\{a \in \Sigma \mid s \preceq a,\ t \preceq a\}$$

has a minimal element $\Delta_{s,t}$ (with regard to $\preceq$). A Garside monoid is exhaustive when $1 \in \Sigma$ and M has a presentation whose generators and relations belong to Σ. The exhaustive Garside monoid manages to gather the characteristics obtained from both the M_Σ monoid, and ()the pre-Garside monoids. This way, with the exhaustive Garside monoids, we obtain what we are searching, enabling us to work with the group of braids, a normal form in the group of fractions, which in turn solves the word problem and a solution to the conjugate problem in the group of fractions. We also achieve other desirable properties such as cancellations or the existence and uniqueness of the greatest common divisor and least common multiple in M.

The terms braid and braid group were coined by the mathematician Emil Artin in the first half of the 20th century. He first gave a topological definition of the braid group and it was not until years later that he obtained an equivalent algebraic definition of the group of braids from an explicit group presentation. The algebraic definition is known as Artin's braid group and is the group generated by $n-1$ elements $\sigma_1, \sigma_2, ..., \sigma_{n-1}$ and the braid relations

$$\sigma_i \sigma_j = \sigma_j \sigma_i,$$

for all $i, j \in \{1, 2, ..., n-1\}$ con $|i - j| \geq 2$ y

$$\sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1},$$

for $i \in \{1, 2, ..., n-2\}$.

The topological definition of braid is more intuitive and is developed from the concept of geometric braid of n strands. A geometric braid of n strands, with $n \geq 1$, is a subset $\mathcal{B} \subset \mathbb{R}^2 \times I$ formed by n topological intervals (subsets of $\mathbb{R}^2 \times I$ homeomorphs to the interval I, being I the interval $[0, 1]$) disjoints called strands in such a way that the projection $\mathbb{R}^2 \times I \to I$ establishes a homeomorphism of each thread in I and

$$\mathcal{B} \cap (\mathbb{R}^2 \times \{0\}) = \{(1, 0, 0), (2, 0, 0), ..., (n, 0, 0)\},$$

$$\mathcal{B} \cap (\mathbb{R}^2 \times \{1\}) = \{(1, 0, 1), (2, 0, 1), ..., (n, 0, 1)\}.$$

Each strand of $\mathcal{B}$ intersects the plane $\mathbb{R}^2 \times \{t\}$ with $t \in I$ at a single point and connects a point $(i, 0, 0)$ with a point $(s(i), 0, 1)$ where $i, s(i) \in \{1, 2, ..., n\}$. The sequence $(s(1), s(2), ..., s(n))$ is a permutation of the set $\{1, 2, ..., n\}$ called the underlying permutation of $\mathcal{B}$.

Through isotopies that are informally simply continuous deformations of one geometric braid in another, an equivalence relation is established. The set of equivalence classes associated with this relation are called braids of n strand.

Geometric braids can be represented by braid diagrams which are projections on the plane. Through the concatenation of the braid diagrams the product of braids is defined. The set of braids of n strands plus this operation form a group. Finally it is shown that this group is equivalent to Artin's group of braids.

We build the braid monoid using the braid group presentation as the monoid presentation. The braids of this monoid are called positive braids. The pair formed by the braid monoid and the Garside element Δ_n constitutes an exhaustive Garside monoid and the associated group of fractions is the braid group, being possible to use all the results and characteristics mentioned above.

We review the basic concepts of cryptography and what the main cryptographic methods consist of, such as symmetric and asymmetric encryption, key exchange, digital signature or authentication.

In order to apply the use of the braid group in cryptography, we use cryptographic platforms, cryptographic protocols based on algebraic objects. If the algebraic object used is a group, this is called a platform group. The security of a cryptographic protocol then depends on the computational or theoretical difficulty of solving a group theory problem in the platform group. For a platform group G to be suitable for a group-based cryptographic protocol, G must possess certain properties that make the protocol both efficient to implement and secure. It must have a finite presentation. It also has to have a normal form to represent the elements in a unique way. It is desirable that the set of elements that can be represented with a certain maximum length grows exponentially with it. It is important that it presents a good diffusion when determining the normal forms of the products, in other words, that when the normal form of a product is found it is computationally hard to calculate the factors. Finally, the group must have a group problem complex enough that it cannot be solved computationally.

We review some cryptographic methods based on the braid group and the conjugacy problem or its variants, such as their algorithms. Among them are the Anshel-Anshel-Goldfeld and Ko, Lee et al key exchange schemes, which were the first attempts to use non-abelian groups as a platform group. An implementation of these is carried out and a study of the execution times is made by varying the parameters of the algorithms.

Finally, a section is dedicated to cryptanalysis of this type of methods. In essence, to break these methods it is enough to solve the conjugacy problem. We are talking about

various techniques to solve this problem computationally, since the solution provided by the Garside monoids is theoretical and cannot be calculated in practice. These are the calculation of the super summit sets, the Embeddability length-based attacks, and the theoretical representation attacks. We also show a heuristic to solve the conjugate problem of which we carry out an implementation. We perform different runs of the algorithm for different conjugacy problems and show their exit rate. The results obtained by the original author's implementation are also shown.

Key words: Group, subgroup, symmetric group, quotient group, free group, group presentation, monoid, divisibility, equivalence, congruence, free monoid, atomic monoid, monoid presentation, weighted presentation, word problem, divisibility problem, normal form, conjugacy problem, cancelativity, comprehensive set, group of fractions, pre-Garside monoid, embeddability, Garside monoid, Artins braid group, geometric braid, brand, isotopy, braid diagram, pure braid, cryptography, symmetric encryptation, asymmetric encryption, key exchange, digital signature, authentication, hash function, cryptographic hash function, Merkle-Damgard construction, Anshel-Anshel-Goldfeld scheme, Diffie-Hellman scheme, cryptanalysis, super summit sets, length-based attacks, theoretical representation attacks, heuristic.

Capítulo 1

Introducción

La criptografía es básicamente la ciencia de cifrar y descifrar mensajes secretos, y se relaciona con la tarea de romper o descubrir estos mensajes secretos. En la criptografía se hace uso de muchas otras disciplinas entre las que se encuentran las matemáticas, la ciencia de la computación y la ingeniería.

Esta disciplina ha sido principalmente utilizada en situaciones militares y de espionaje. Ya Julio Cesar enviaba mensajes cifrados en sus campañas militares. Su método, conocido como Cifrado César, consistía en desplazar cada letra una cierta cantidad de posiciones en el abecedario. Este cifrado volvería a ser utilizado más tarde en la Guerra Civil Americana.

La mayoría de los métodos de cifrado revelaban información estadística del texto plano. Esta información podía ser usada para romperlos. Fue en el siglo IX cuando el matemático árabe Al-Kindi el análisis de frecuencia que volvía a los métodos de sustitución vulnerables.

La utilización de la criptografía en Europa durante la edad media se encontró predominantemente en la diplomacia. En 1470, Leon Alberti desarrolló un método que intentó frustrar el análisis estadístico. Su innovación fue la de usar un cifrado polialfabético, donde diferentes partes del mensaje se cifraban con diferentes alfabetos. Este método fue equivocadamente atribuido al criptógrafo francés Blaise de Vigenére un siglo después, conociéndose como el Cifrado de Vigenére. Finalmente también se demostró que era vulnerable a ataques estadísticos.

Hubo varios dispositivos físicos construidos con la función de cifrar mensajes. Antes del uso generalizado de ordenadores, el dispositivo más famoso fue la Máquina Enigma, desarrollada y utilizada por el ejercito alemán durante la Segunda Guerra Mundial. Era una máquina que disponía de varios rotores para realizar un cifrado polialfabético. Criptógrafos británicos liderados por el matemático Alan Turing consiguieron romperlo, lo que tuvo un gran impacto en la guerra.

Con la llegada de los ordenadores se requirió desarrollar métodos de cifrado más complejos, ya que la capacidad de criptoanálisis también aumentó. En 1976, Diffie y Hellman desarrollaron el primer protocolo de intercambio de llave pública. Un año después, Rivest, Adelman y Shamir desarrollaron el algoritmo RSA, un segundo protocolo de llave pública.

Tanto Diffie-Hellman como RSA necesitaban espacios de llaves muy largos. Con la idea de reducir el espacio de llaves se propuso que Diffie-Hellman fuese aplicado a otros grupos abelianos.

Las aplicaciones criptográficas se volvieron tan predominantes que se hizo necesaria la creación de un cifrado estándar. Así surgió en el año 1976 *Data Encryption Standard* (DES). Este se demostró inseguro por primera vez en 1999, por lo que tuvo que sustituirse por *Advanced Encryption Standard*(AES).

Finalmente, en los últimos años se ha visto que las ideas de algoritmos cuánticos y la posibilidad de ordenadores cuánticos funcionales presentan un riesgo para algunos métodos criptográficos basados en grupos abelianos. Esto llevó a considerar métodos de llave pública usando grupos no abelianos. En 1999 Ko, Lee *et al.* y Anshel-Anshel-Goldfeld introdujeron métodos de intercambio de llave pública involucrando al grupo de trenzas como plataformas, una linea en la que se ha estado investigando desde entonces.

Este trabajo se desarrolla mayoritariamente en las áreas de la teoría de grupos, teoría de monoides y criptografía. Los objetivos que deseamos abarcar son, por una parte, conocer en profundidad el grupo de trenzas y cómo trabajar con él, y por otra parte, dar ejemplos de su aplicación en criptografía y evaluar su seguridad.

Con estos propósitos comenzaremos repasando conceptos y resultados básicos sobre grupos necesarios para la construcción del grupo de trenzas, como pueden ser los grupos cocientes y las presentaciones de grupos.

Luego procederemos a la construcción del grupo de trenzas, tanto desde el punto de vista algebraico como topológico y demostraremos su equivalencia haciendo uso de los conceptos básicos vistos previamente.

También realizaremos un profundo estudio de los monoides de Garside a partir de los cuales obtendremos un conjunto de propiedades necesarias para trabajar tanto con el monoide como con el grupo de fracciones asociado. Entre estas propiedades se encuentran la cancelatividad, soluciones al problema de palabra y al problema del conjugado, y una forma normal.

A partir de la definición del monoide de trenzas demostraremos que es un monoide de Garside. Veremos que su grupo de fracciones es el grupo de trenzas inicialmente definido, obteniendo las propiedades necesarias para trabajar con él.

Repasaremos los principales métodos criptográficos, viendo algoritmos que los llevan a cabo haciendo uso de las trenzas y del problema del conjugado o variantes suyos como base de su seguridad. También se realizará una implementación de los algoritmos para

el intercambio de llaves en los que se medirán los tiempos de ejecución para diferentes valores de los parámetros del algoritmo.

Por último veremos algunas técnicas de criptoanálisis que se basan en resolver el problema del conjugado en el grupo de trenzas para romper la seguridad los métodos criptográficos vistos. Se llevará a cabo la implementación de una heurística cuya intención es resolver el problema del conjugado y se mostrarán sus resultados.

Capítulo 2

Generalidades sobre grupos

2.1. Grupos y subgrupos

En este capítulo repasaremos algunos aspectos de la teoría de grupos que en su mayoría se han visto a lo largo de la carrera [6, 7, 8, 9]. Serán necesarios para desarrollar tanto el grupo de trenzas, como la parte de criptografía y criptoanálisis. En primer lugar vamos a comenzar con algunos conceptos básicos sobre grupos.

Definición 2.1.1 (Grupo)**.** Un grupo es una estructura algebraica formada por un conjunto $G \neq \emptyset$ y una operación interna $G \times G \to G$ a la que llamaremos *producto*. Esta operación asigna a cada pareja (x, y) el elemento xy y verifica las siguientes propiedades:

1. Propiedad asociativa: $x(yz) = (xy)z$ para todo $x, y, z \in G$.
2. Existencia de elemento neutro: Existe $1 \in G$ tal que $x1 = x = 1x$ para todo $x \in G$.
3. Existencia de inverso: Para cada $x \in G$ existe $y \in G$ tal que $xy = yx = 1$. El elemento y es único y se dice que es el *inverso* de x. Lo denotaremos por x^{-1}.

Si además cumple la siguiente propiedad se dice que G es un *grupo abeliano*.

4. Propiedad conmutativa: $xy = yx$ para todo $x, y \in G$.

Definición 2.1.2 (Subgrupo)**.** Sea G un grupo. Un subgrupo de G es un subconjunto $H \subseteq G$, $H \neq \emptyset$ que verifica

1. $xy \in H$,
2. $1 \in H$,

3. $x^{-1} \in H$,

para todo $x, y \in H$.

Si H es un subgrupo de G, lo denotaremos $H \leq G$. Definimos $\mathrm{Sub}(G)$ como el conjunto formado por todos los subgrupos de G, $\mathrm{Sub}(G) = \{H : H \leq G\}$. La siguiente proposición es una caracterización de los subgrupos más práctica.

Proposición 2.1.1. Sea H un subconjunto no vacío de un grupo G, entonces $H \leq G$ si y solo si para todo $x, y \in H$ se tiene que $xy^{-1} \in H$.

Tenemos que la intersección de subgrupos es una operación cerrada en $\mathrm{Sub}(G)$.

Proposición 2.1.2. Sean $H, K \leq G$, entonces $H \cap K \leq G$.

Definición 2.1.3 (Subgrupo generado)**.** Sea G un grupo y $X \subseteq G$ un subconjunto. Definimos el subgrupo generado por X, que denotaremos $\langle X \rangle$, como el menor subgrupo de G que contiene al conjunto X. Se tiene que

$$\langle X \rangle \quad = \bigcap_{X \subseteq K \in \mathrm{Sub}(G)} K.$$

Proposición 2.1.3. Para $X \subseteq G$ siendo $X \neq \emptyset$ se tiene que

$$\langle X \rangle = \{x_1^{n_1} x_2^{n_2} \cdots x_r^{n_r} : \ x_i \in X, \ n_i \in \mathbb{Z}\}.$$

Si $X = \{x_1, x_2, ..., x_k\} \subseteq G$ es un subconjunto finito, entonces escribimos $\langle x_1, x_2, ..., x_k \rangle = \langle X \rangle$.

Si $\langle X \rangle = G$, diremos que X es un conjunto de generadores de G. Diremos que G es finitamente generado si existe $X = \{x_1, x_2, ..., x_k\} \subseteq G$, con $G = \langle X \rangle$.

2.2. Grupo simétrico S_n

A continuación vamos a definir los grupos simétricos con los que más adelante estableceremos una relación con los grupos de trenzas. También vamos a ver algunas de sus propiedades que nos serán de utilidad.

Definición 2.2.1 (Grupo simétrico)**.** Sea $X \neq \emptyset$. Denominamos a una función biyectiva de X en sí misma una *permutación* de X. El conjunto S_X de todas las permutaciones de X es un grupo bajo la composición como operación, y se llama *grupo simétrico* o *grupo de permutaciones*.

El orden de S_X es $|X|!$ cuando X es finito. El grupo S_X es no abeliano para $|X| \geq 3$. El grupo simétrico del conjunto $I_n = \{1, \ldots, n\}$ se denota por S_n y tiene orden $n!$.

Si X es finito, $X = \{a_1, \ldots, a_n\}$ podemos expresar $\sigma \in S_X$ de forma matricial

$$\sigma = \begin{pmatrix} a_1 & a_2 & \cdots & a_n \\ a_{i_1} & a_{i_2} & \cdots & a_{i_n} \end{pmatrix},$$

donde $\sigma(a_j) = a_{i_j}$ para todo $j \in \{1, \ldots, n\}$.

Si $a_1, \ldots, a_k$, con $k \geq 2$, son elementos distintos de X, la expresión

$$(a_1 \cdots a_k)$$

denota la permutación que envía a_i a a_{i+1} para $i \in \{1, \ldots, k-1\}$ y el último elemento a_k al primer elemento a_1. Los demás elementos de X se mantienen fijos. A esta permutación se le llama *ciclo de longitud* k en S_X. Llamamos a un ciclo de longitud 2 $(a\ b)$ transposición, ya que simplemente transpone a y b, dejando al resto de elementos de X fijos.

Dados dos ciclos $(a_1 \cdots a_k)$ y $(b_1 \cdots b_h)$ decimos que son disjuntos si $a_i \neq b_j$ para todo $i \in \{1, \ldots, k\}$, $j \in \{1, \ldots, h\}$.

Teorema 2.2.1. Sea S_X un grupo simétrico, se cumple lo siguiente:

1. Los ciclos disjuntos en S_X conmutan.

2. Cada permutación en S_X es un producto de ciclos disjuntos, siendo el producto único excepto por el orden de los factores.

Una representación de σ como producto de ciclos disjuntos se le llama *representación cíclica* o *descomposición cíclica* de σ. La *estructura cíclica* de una permutación σ es la sucesión de longitudes de ciclo de una descomposición cíclica de σ, o equivalentemente, el número de ciclos de cada longitud en una descomposición cíclica de σ.

Proposición 2.2.1. Todo ciclo puede descomponerse como producto de transposiciones.

Demostración. Basta con comprobar la siguiente igualdad

$$(i_1\ i_2\ i_3 \cdots i_{r-1}\ i_r) = (i_1\ i_r)(i_1\ i_{r-1}) \cdots (i_1\ i_3)(i_1\ i_2),$$

recordando que la composición se realiza de derecha a izquierda. □

Definición 2.2.2. Sean $\alpha, \beta \in S_X$, decimos que son conjugados si existe $\sigma \in S_X$ tal que

$$\alpha = \sigma^{-1}\beta\sigma.$$

Denominamos a α el conjugado de β por σ y lo denotamos por β^σ.

Teorema 2.2.2.

1. Sea $\sigma \in S_n$. Para cualquier ciclo $(a_1 \cdots a_k)$,

   $$(a_1 \cdots a_k)^\sigma = \sigma^{-1}(a_1 \cdots a_k)\sigma = (\sigma a_1 \cdots \sigma a_k).$$

   Por tanto, si $\tau = c_1 \cdots c_k$ es una descomposición cíclica de τ, entonces

   $$\tau^\sigma = c_1^\sigma \cdots c_k^\sigma$$

   es una descomposición cíclica de τ^σ.

2. Dos permutaciones son conjugadas si y solo si tienen la misma estructura cíclica.

Denotamos por s_i con $i \in \{1, \ldots, n-1\}$ a la transposición $(i \;\; i+1) \in S_n$. Veamos que el conjunto formado por estas transposiciones es un conjunto de generadores de S_n.

Proposición 2.2.2. El conjunto $\{s_1, \ldots, s_n\}$ es un conjunto de generadores de S_n.

Demostración. Para ver que $\{s_1, \ldots, s_n\}$ es un conjunto de generadores de S_n hay que demostrar que cualquier elemento de S_n se puede escribir como producto de estos elementos. Por El Teorema 2.2.1 sabemos que una permutación es un producto de ciclos disjuntos y hemos visto que todo ciclo se puede poner como producto de transposiciones (Proposición 2.2.2). Por último nos haría falta probar que una transposición cualquiera (i, j) con $i, j \in \{1, \ldots, n\}$, $j - i \geq 2$ es obtenida como producto de elementos de $\{s_1, \ldots, s_n\}$. Utilizando la conjugación encontradomos fácilmente el producto que buscabamos,

$$(i \;\; i{+}1)^{(i+1 \;\; i+2)\cdots(j-1 \;\; j)} = (j{-}1 \; j)^{-1} \cdots (i{+}1 \; i{+}2)^{-1}(i \;\; i{+}1)(i{+}1 \; i{+}2) \cdots (j{-}1 \; j) = (i \;\; j).$$

□

Es fácil ver que el conjunto $\{s_1, \ldots, s_n\}$ es un conjunto de generadores de S_n, ya que por el Teorema 2.2.1 cualquier permutación se puede escribir como producto de ciclos disjuntos y por la Proposición 2.2.2 cada ciclo se puede escribir como producto de transposiciones, obteniendo así que toda permutación se puede escribir como producto de transposiciones. Ya que $s_i^{-1} = s_i$ para $i \in \{1, \ldots, n-1\}$ y $s_1, \ldots, s_n$ son generadores de

S_n, cualquier permutación de S_n se puede expresar como un producto $\sigma = s_{i_1}s_{i_2}\cdots s_{i_r}$ con $s_{i_1}s_{i_2}\ldots s_{i_r} \in \{1, 2, \ldots, n-1\}$. Si r es mínimo para todas las expresiones de σ, entonces decimos que $s_{i_1}s_{i_2}\cdots s_{i_r}$ es una *expresión reducida* de σ y que $s_{i_1}s_{i_2}\cdots s_{i_r}$ es una *palabra reducida*. Una permutación puede tener diferentes palabras reducidas. Definimos como *longitud* $\lambda(w)$ de una permutación σ como la longitud r de una expresión reducida $s_{i_1}s_{i_2}\cdots s_{i_r}$ para σ.

Sea $w_0 \in S_n$ la permutación $i \mapsto n-1+i$ para todo $i \in \{1, \ldots, n-1\}$:

$$w_0 = \begin{pmatrix} 1 & 2 & \cdots & n-1 & n \\ n & n-1 & \cdots & 2 & 1 \end{pmatrix}.$$

Esta permutación cumple que $\lambda(w_0) = n(n-1)/2$ y se da que $\lambda(\sigma) < n(n-1)/2$ para todo $\sigma \in S_n$ con $\sigma \neq w_0$. Por eso se conoce a w_0 como el *elemento más largo* de S_n.

Lema 2.2.1. Para cualesquiera $\alpha, \beta \in S_n$, tales que $\alpha\beta = w_0$,

$$\lambda(\alpha) + \lambda(\beta) = \lambda(w_0).$$

Las justificaciones de estas afirmaciones sobre w_0 se encuentran en [11].

Definición 2.2.3 (Homomorfismo de grupo)**.** Sean G y G' grupos. Una función $f : G \to G'$ se denomina *homomorfismo de grupos* (o simplemente *homomorfismo*) si

$$f(x, y) = f(x)f(y)$$

para todo $x, y \in G$. Un homomorfismo inyectivo es un *monomorfismo*, un homomorfismo sobreyectivo es un *epimorfismo* y un homomorfismo biyectivo es un *isomorfismo*. Cuando f es un isomorfismo, decimos que G y G' son *isomorfos* y lo denotamos por $G \cong G'$. Definimos la imagen del homomorfismo f como el conjunto

$$\mathrm{Img}(f) := \{f(x) \in G' : x \in G\}.$$

Entonces se tiene que $\mathrm{Img}(f)$ es un subgrupo de G'.

Definición 2.2.4. Sea $f : G \to G'$ un homomorfismo de grupos. Consideramos el conjunto

$$\ker(f) := \{x \in G : f(x) = 1\}.$$

A este conjunto se le denomina el *núcleo* de f.

2.3. Grupo cociente

Definición 2.3.1 (Equivalencia módulo un subgrupo)**.** Sea G un grupo y H un subgrupo de G. Dados $x, y \in G$, decimos que x es congruente con y módulo H si se cumple que $y^{-1}x \in H$. Denotaremos la relación por $x \sim_H y$. A continuación vamos a utilizar el concepto de relación de equivalencia que se ve ampliamente en la Sección 3.1.4.

Proposición 2.3.1. Sea G un grupo y H un subgrupo de G. La relación $\sim_H$ es una relación de equivalencia en G. La clase de equivalencia de $x \in G$ es $xH = \{xh : h \in H\}$.

Como, en general, xH no coincide con Hx, podemos considerar otra relación de equivalencia de modo que las clases de equivalencia sean Hx. La relación será tal que $x_H \sim y$ si $xy^{-1} \in H$. Se tiene que la relción $_H \sim$ es también una relación de equivalencia.

Definición 2.3.2 (Clases laterales)**.** Sea G un grupo, H un subgrupo de G y x un elemento de G. Llamamos *clase lateral a izquierda* de x respecto de H al conjunto $xH = \{xh : h \in H\}$. Análogamente, llamamos *clase lateral por la derecha* de x respecto de H al conjunto $Hx = \{hx : h \in H\}$.

Definición 2.3.3 (Conjunto cociente)**.** Simbolizamos por G/H al conjunto determinado por todas las clases laterales a izquierda de H en G y diremos que se trata del *conjunto cociente* de G por H por la izquierda. El correspondiente conjunto de clases laterales por la derecha, se simbolizará por $H\backslash G$ (conjunto cociente por la derecha).

A continuación vamos a ver cuando es posible establecer una estructura de grupo en el conjunto G/H.

Sea G un grupo, H un subgrupo de G y G/H el conjunto cociente de G por la partición definida por las clases laterales a izquierda de H. Lo que queremos es definir una operación en G/H tal que

$$xH \cdot xH = xyH$$

para todo $x, y \in G$. Es fácil de ver que la condición de que las clases laterales por la izquierda y por la derecha coincidan, es decir, que

$$xH = Hx$$

para todo $x \in G$, es suficiente para que la operación en G/H este bien definida. En efecto, se cumplirá que

$$xHyH = x(Hy)H = x(yH)H = xyHH = xyH$$

para todo $x, y \in G$. Como esta condición no se cumple en todos los subgrupos, se les nombra de forma especial.

Definición 2.3.4 (Subgrupo normal)**.** Sea G un grupo. Diremos que un subgrupo H de G es *normal* en G, y lo dentaremos por $H \trianglelefteq G$, si cumple una de las siguientes afirmaciones (son equivalentes):

1. $xH = Hx$ para todo $x \in G$.
2. $xHx^{-1} = H$ para todo $x \in G$.
3. $xhx^{-1} \in H$ para todo $h \in H, x \in G$.

Definición 2.3.5 (Subgrupo normal generado)**.** Sea G un grupo y $X \subseteq G$ un subconjunto. Definimos el subgrupo normal generado por X, que denotaremos $\langle X \rangle_N$, como el menor subgrupo normal de G que contiene al conjunto X. Se tiene que

$$\langle X \rangle_N = \bigcap_{X \subseteq K, K \trianglelefteq G} K.$$

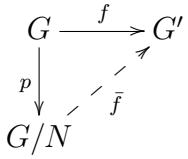
Proposición 2.3.2. Sea G un grupo y H un subgrupo normal de G. Entonces, el conjunto de las clases laterales a izquierda (que coincide con el conjunto de las clases laterales por la derecha) tiene una estructura de grupo respecto a la operación definida por

$$xH \cdot yH := xyH.$$

Denotaremos a este grupo por G/H y lo llamaremos *grupo cociente* de G por H. La función $x \mapsto xH$ es un homomorfismo sobreyectivo al que denominaremos *proyección canónica*.

Proposición 2.3.3. Sea $f : G \to G'$ un homomorfismo de grupos. Entonces, el núcleo del f es un subgrupo de G y además es normal en G.

Teorema 2.3.1 (Teorema de Factorización)**.** Sea $f : G \to G'$ un homomorfismo de grupos y sea N un subgrupo normal de G con $N \leq \ker(f)$, entonces existe un único homomorfismo $\bar{f} : G/N \to G'$ tal que $\bar{f} \circ p = f$.



Además:

1. $\bar{f}$ es epimorfismo si y solo si f es epimorfismo,

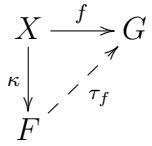
2. $\bar{f}$ es monomorfismo si y solo si $N = \ker(f)$.

Teorema 2.3.2 (Primer Teorema de Isomorfía)**.** Sea $f : G \to G'$ un homomorfismo de grupos y sea K el núcleo de f. Entonces, $\mathrm{Img}(f) \cong G/K$. El isomorfismo viene dado por la correspondencia $\bar{f}$ que a cada xK de G/K le asocia el elemento $f(x)$ de $\mathrm{Img}(f)$.

Demostración. Aplicamos el teorema de factorización al epimorfismo $f' : G \to \mathrm{Img}(f)$, que es la misma función que f, solo que en vez de ir de G a G', va de G a $\mathrm{Img}(f)$. Teniendo en cuenta que $N = \ker(f) = \ker(f') \trianglelefteq G$ y se obtiene que la aplicación $\bar{f} : G/\ker f \to \mathrm{Img}(f)$ tal que $\bar{f}(a\ker(f)) = f(a)$ es un isomorfismo y es único. □

2.4. Grupo libre

Definición 2.4.1 (Grupo libre)**.** Sea X un conjunto y G un grupo. Un par $(F, \kappa : X \to F)$ donde F es un grupo se dice *universal* para X si para cualquier función $f : X \to G$ de X al grupo G hay un único homomorfismo de grupos $\tau_f : F \to G$ tal que $\tau_f \circ \kappa = f$.



Al grupo F se le llama *grupo libre* en X y a X se le denomina un conjunto de *generadores libres* de F. La función κ se llama función universal para el par (F, κ). Usamos la notación F_X para denotar el grupo libre de X, que es único salvo isomorfismo.

La noción de grupo libre puede ser definida constructivamente. Sea X un conjunto no vacío y X' un conjunto en biyección con X pero disjunto. Bajo una biyección fija, escibimos el elemento de X' correspondiente a un elemento $x \in X$ como x^{-1}. Definimos como una *palabra no vacía* en X a una expresión de la forma $x_1^{\epsilon_1} \cdots x_r^{\epsilon_r}$ donde $x_i \in X$, $\epsilon_i \in \{1, -1\}$ y $r > 0$. Si $r = 0$ la denominamos la *palabra vacía* en X y lo denotamos por 1. Llamamos a r la *longitud de la palabra*. Consideramos a dos expresiones la misma palabra si tienen los mismos elementos en la misma posición. Definimos el producto de dos palabras no vacías $x_1^{\epsilon_1} \cdots x_r^{\epsilon_r}$ y $y_1^{\epsilon_1} \cdots y_s^{\epsilon_s}$ como la yuxtaposición de ambas $x_1^{\epsilon_1} \cdots x_r^{\epsilon_r} y_1^{\epsilon_1} \cdots y_s^{\epsilon_s}$. Definimos el producto de cualquier palabra w con 1 por ambos lados ella misma. Llamamos a $x_r^{-\epsilon_r} \cdots x_1^{-\epsilon_1}$ el inverso de la palabra no vacía $x_1^{\epsilon_1} \cdots x_r^{\epsilon_r}$. Escribimos $(x_1^{\epsilon_1} \cdots x_r^{\epsilon_r})^{-1} = x_r^{-\epsilon_r} \cdots x_1^{-\epsilon_1}$ y $1^{-1} = 1$.

Definimos la siguiente relación en el conjunto de palabras visto. Dadas w_1 y w_2, dos palabras en X, decimos que $w_1 \sim w_2$ si w_2 es obtenido de w_1 a través de aplicar una sucesión finita de inserciones o sustracciones de expresiones del tipo xx^{-1} o $x^{-1}x$ para $x \in X$. Esta es una relación de equivalencia y el conjunto F de clases de equivalencia

$[w]$ nos da una estructura de grupo de forma obvia. Definiendo el producto $[w_1][w_2] = [w_1 w_2]$, es fácil ver que F tiene una estructura de grupo bien definida. El grupo F es libre en X, la demostración se puede ver en [9].

2.5. Presentación de grupo

Definición 2.5.1. Una *relación de grupo* entre elementos de un conjunto X es un par ordenado (u, v) de elementos de F_X. Estas relaciones normalmente se escriben como $u = v$.

Una relación de grupo entre elementos de un conjunto X se dice que se *mantiene* en G por una función f de X en G cuando $\varphi(u) = \varphi(v)$, donde $\varphi : F_X \to G$ es el único homomorfismo que extiende f.

Definición 2.5.2. Dado un conjunto X y un conjunto R de relaciones de grupo entre elementos de X, el grupo $\langle X \mid R\rangle$ es el cociente del grupo libre F_X por el subgrupo normal N de F_X generado por todos los elementos uv^{-1} con $(u, v) \in R$.

El grupo $\langle X \mid R\rangle = F_X/N$ tiene una *función canónica* $\iota : X \to \langle X \mid R\rangle$, la composición $\iota = \pi \circ \eta$ de la función inclusión $\eta : X \to F_X$ y la proyección canónica $\pi : F_X \to F_X/N$.

Proposición 2.5.1. Sea R un conjunto de relaciones de grupo entre elementos de un conjunto X. Cada relación $(u, v) \in R$ se mantiene en $\langle X \mid R\rangle$ por la función canónica $\iota : X \to \langle X \mid R\rangle$. Además, $\langle X \mid R\rangle$ es generado por $\iota(X)$.

Definición 2.5.3. Una *presentación* de un grupo G es un isomorfismo de un grupo $\langle X \mid R\rangle$ en G. A los elementos de X se les llama *generadores* y a los elementos de R *relaciones* de la presentación.

Teorema 2.5.1. Todo grupo G tiene una presentación $\langle X \mid R\rangle$.

Las demostraciones de estos dos resultados se pueden encontrar en [6, 8].

Capítulo 3

Monoides de Garside

Vamos a definir una estructura algebraica denominada monoide de Garside, una herramienta muy potente que nos permitirá trabajar con las trenzas. Entre otras cosas nos será posible dar soluciones al problema del conjugado y el problema de palabra definidos más adelante, y dispondremos de una forma normal con la que poder trabajar directamente con las trenzas. Todo esto nos será indispensable para poder construir métodos criptográficos basados en el grupo de trenzas. Lo referente a monoides que se habla en esta sección aparece en [11]. Comenzamos con la definición de monoide.

3.1. Monoides

Un monoide es un conjunto que cuenta con una operación binaria (multiplicación) $M \times M \to M$ la cual es asociativa y tiene un elemento neutro que denotaremos por 1.

Un monoide M es *cancelativo por la izquierda* (respectivamente *por la derecha*) si para todo $a, b, c \in M$,

$$ab = ac \text{ implica } b = c \quad \text{(respectivamente } ba = ca \text{ implica } b = c).$$

Un elemento a de un monoide M es *invertible* si existe $b \in M$ tal que $ab = ba = 1$. De esta forma, un grupo es un monoide en el que todo elemento es invertible.

Definición 3.1.1. Dados M y M' dos monoides, se dice que una función $f : M \to M'$ es un homomorfismo de monoides si $f(xy) = f(x)f(y)$ para todo $x, y \in M$ y $f(1) = 1$. Un homomorfismo inyectivo es un *monomorfismo*, un homomorfismo sobreyectivo es un *epimorfismo* y un homomorfismo biyectivo es un *isomorfismo*. Cuando f es un isomorfismo, decimos que M y M' son *isomorfos* y lo denotamos por $M \cong M'$.

Definición 3.1.2 (Centro)**.** Sea M un monoide, llamamos *centro de* M al conjunto

$$Z(M) := \{x \in M : xa = ax \ \forall a \in M\}.$$

A un elemento $b \in Z(M)$ se le dice que es central en M.

3.1.1. Divisibilidad en monoides

Si $a = bc$, donde a, b, c son elementos de un monoide M, entonces decimos que b es un *divisor por la izquierda* de a y c es un *divisor por la derecha*. También decimos que a es un *múltiplo por la derecha* de b y que a es *múltiplo por la izquierda* de c. Lo denotaremos por $b \preceq a$ y $a \succeq c$. Por ejemplo, $1 \preceq a$ y $a \succeq 1$ para todo $a \in M$, ya que $a = 1a = a1$.

Lema 3.1.1. Las relaciones $\preceq$ y $\succeq$ en un monoide son reflexivas y transitivas.

Demostración. La reflexividad de $\preceq$ viene de la identidad $a = a1$. La transitividad viene dada por la asociatividad de la multiplicación. Se procede de igual manera para $\succeq$. □

3.1.2. Monoide libre

El *monoide libre* sobre un conjunto X es el monoide X^* que contiene X como un subconjunto y tal que cualquier función de X a M, donde M es un monoide, se extiende de forma única a un homomorfismo de monoides $X^* \to M$. Esta propiedad se define bajo isomorfismo de monoides sobre X^*.

Una definición equivalente de monoide libre es la siguiente. Dado un conjunto X no vacío, el conjunto de palabras en X, tomando las definiciones de palabras vistas en la Sección 2.4, considerando solo elementos con exponente positivo, junto con la operación de yuxtaposición es monoide libre en X. La demostración se puede ver en [10].

Cualquier elemento w de X^* puede ser expresado como una palabra de X, es decir, como un producto de elementos de X. El número de elementos de X de esta expresión de $w \in X^*$ se denomina *longitud* de w y se denota por $l(w)$. Se tiene que $l(1) = 0,\ l(x) = 1$ para $x \in X$ y que $l(ww') = l(w) + l(w')$ con $w, w' \in X^*$.

3.1.3. Monoides atómicos

Para cualquier elemento $a \neq 1$ de un monoide M, definimos

$$\|a\| = \sup\{r \geq 1 \mid a = a_1 \cdots a_r, a_1, ..., a_r \in M \backslash \{1\}\}.$$

Según esta definición $\|a\| \in \mathbb{N}^* \cup \{\infty\}$. Establecemos que $\|1\| = 0$. Es fácil de ver que para todo $a, b \in M$,

$$\|ab\| \geq \|a\| + \|b\|.$$

Observar que $\|a\| = 0$ si y solo si $a = 1$. A un elemento $a \in M$ se le llama átomo si $\|a\| = 1$. En otras palabras, $a \in M$ es un átomo si $a \neq 1$ y $a = a_1 \cdots a_r$ implica que $a_i = 1$ para todo i excepto uno. Cualquier $a \in M$ con $\|a\| < \infty$ tiene una expansión $a = a_1 \cdots a_r$, donde $\|a\| = r$ y $a_1, ..., a_r$ son átomos. Esto justifica la siguiente definición. Un monoide M es *atómico* si $\|a\| < \infty$ para todo $a \in M$.

Lema 3.1.2. Si dos elementos a, b de un monoide atómico M satisfacen $a \preceq b$ y $b \preceq a$, entonces $a = b$. De igual manera, si $a \succeq b$ y $b \succeq a$, entonces $a = b$.

Demostración. Ya que $a \preceq b \preceq a$, significa que existen $u, v \in M$ tal que $b = au$ y $a = bv$. Entonces $a = auv$ y

$$\|a\| = \|auv\| \geq \|a\| + \|u\| + \|v\| \geq \|a\|.$$

Esto implica que $\|u\| = \|v\| = 0$. Por tanto $u = v = 1$ y $a = b$. La relación $\succeq$ es tratada de forma similar. □

Los Lemas 3.1.1 y 3.1.2 implican que las relaciones $\preceq$ y $\succeq$ en un monoide atómico son órdenes (parciales). Dado un subconjunto E de un monoide atómico M, decimos que un elemento $a \in E$ es *máximo* (respectivamente *mínimo*) con respecto a $\preceq$, si $b \preceq a$ (respectivamente $a \preceq b$) para todo $b \in E$. Un elemento máximo (respectivamente mínimo) de E podría no existir, pero si existe, es único por el Lema 3.1.2. Definiciones equivalentes se obtienen con la relación $\succeq$.

La ecuación $ab = 1$ en un monoide atómico M tiene como única solución $a = 1, b = 1$. De hecho, si $ab = 1$ para $a, b \in M$, entonces $1 \preceq a \preceq 1$, por lo tanto $a = 1$ y $b = 1$. En particular, el elemento neutro es el único elemento invertible de M.

3.1.4. Presentaciones de un monoide

Para poder definir con propiedad que es la presentación de un monoide, primero vamos a tener que ver algunos aspectos sobre relaciones y congruencias. Las definiciones y demostraciones de las afirmaciones referentes a estos se pueden encontrar en [10].

Una *relación (binaria)* en un conjunto X es básicamente un subconjunto ρ del producto cartesiano $X \times X$. Diremos que dos elementos $x, y \in X$ están *relacionados* si $(x, y) \in \rho$ y se suele denotar por $x \sim y$. Para cada $x \in X$ definimos el conjunto $x\rho = \{y \in X : (x, y) \in \rho)\}$.

Una *relación de equivalencia* en X, ρ, es una relación que cumple las siguientes propiedades:

1. Reflexividad. $(x, x) \in \rho$ para todo $x \in X$.

2. Simetría. Si $(x, y) \in \rho$ entonces $(y, x) \in \rho$.

3. Transitividad. Si $(x, y), (y, z) \in \rho$ entonces $(x, z) \in \rho$.

Proposición 3.1.1. Sea ρ una relación de equivalencia en un conjunto X. Entonces la familia

$$\Phi(\rho) = \{x\rho : x \in X\}$$

de subconjuntos de X es una partición de X.

Los conjuntos $x\rho$ que forman la partición asociada a la relación de equivalencia las denominamos $\rho-$*clases* o *clases de equivalencia*. Al conjunto de $\rho-$clases, cuyos elementos son los conjuntos $x\rho$, se le llama *conjunto cociente* de X por ρ, y se denota por X/ρ.

Proposición 3.1.2. Dada $\{\rho_i : i \in I\}$, una familia no vacía de relaciones de equivalencia en un conjunto X, entonces $\cap\{\rho_i : i \in I\}$ es una relación de equivalencia.

Si R es cualquier relación en X, entonces la familia de relaciones de equivalencia conteniendo a R es no vacía, puesto que $X \times X$ es una. Por tanto, la intersección de todas las relaciones de equivalencia conteniendo R es una relación de equivalencia, la mínima relación de equivalencia en X conteniendo a R. La llamaremos relación de equivalencia *generada* por R, y lo denotaremos por R^e.

Definición 3.1.3. Sea M un monoide. Una relación R en el conjunto M se dice *compatible por la izquierda* si $(s, t) \in R$ implica que $(as, at) \in R$ para cualquier $s, t, a \in M$ y *compatible por la derecha* si $(s, t) \in R$ implica que $(sa, ta) \in R$ para cualquier $s, t, a \in M$. Se dice *compatible* si $(s, t), (s't') \in R$ implica que $(ss', tt') \in R$.

Una relación de equivalencia compatible por la izquierda (respectivamente por la derecha) la llamamos *congruencia por la izquierda* (respectivamente *por la derecha*). Una relación de equivalencia compatible la llamamos *congruencia*.

Proposición 3.1.3. Una relación ρ en un monoide M es una congruencia si y solo si es una congruencia por la izquierda y por la derecha.

Sea ρ una congruencia en un monoide M, entonces podemos definir una operación binaria en el conjunto cociente M/ρ de una forma natural,

$$(x\rho)(y\rho) = (xy)\rho$$

para $x, y \in M$. Esta operación está bien definida precisamente porque ρ es compatible.

Teorema 3.1.1. Sea M un monoide y ρ una congruencia en M. Entonces M/ρ es un monoide con la operación que acabamos de definir y la función $\pi : M \to M/\rho$ con

$\pi(x) = x\rho$ para $x \in M$ es un homomorfismo. Sea M' un monoide y $\phi : M \to M'$ un homomorfismo de monoides. Entonces la relación

$$\ker(\phi) = \{(x, y) \in M \times M : \phi(x) = \phi(y)\}$$

es una congruencia en M, y hay un único monomorfismo $\alpha : M/\ker(\phi) \to M'$ tal que $\phi = \alpha \circ \pi$.

La intersección de una familia no vacía de congruencias en un monoide M es una congruencia en M. Procediendo como lo hemos hecho antes con las relaciones de equivalencia, deducimos que para cada relación R en M existe la congruencia más pequeña $R^{\#}$ en M conteniendo a R, la cual es única y se obtiene de la intersección de la familia de todas las congruencias en M que contienen a R.

Para un monoide M y una relación R cualquiera, definimos el conjunto

$$R^c = \{(xay, xby) : x, y \in M,\ (a, c \in (R))\},$$

que es el conjunto más pequeño de relaciones compatibles conteniendo a R.

Proposición 3.1.4. Para cualquier relación R en un monoide M, $R^{\#} = (R^c)^e$.

Definición 3.1.4. Dado un conjunto X, consideramos el monoide libre X^*. Sea R una relación en X^*, el monoide $\langle X \mid R\rangle$ es el cociente $X^*/R^{\#}$.

Definición 3.1.5. Una presentación de un monoide M es un isomorfismo de M en un monoide $\langle X \mid R\rangle$. Se dice que X es el conjunto de generadores, R es el conjunto de relaciones y un elemento $(x, y) \in R$ es una relación que se suele denotar por $x = y$.

Introducimos varios tipos de presentaciones de monoides que nos serán de utilidad. Una presentación de monoide $\langle X \mid R\rangle$ es finita si X y R son finitos. Una presentación $\langle X \mid R\rangle$ de un monoide M se dice *ponderada* si existe un homomorfismo de monoides $\ell : M \to \mathbb{N}$ tal que $\ell(x) \geq 1$ para todo $x \in X$. El homomorfismo ℓ se llama *peso*. Consideramos que $\ell : M \to \mathbb{N}$ es un *peso canónico* si cumple que $\ell(x) = 1$ para todo $x \in X$.

Una presentación $\langle X \mid R\rangle$ de un monoide M se dice *equilibrada en longitud* si $l(r) = l(r')$ para todo $(r, r') \in R$, donde l es la función longitud en X^* introducida en la Sección 3.1.2 y existe un peso canónico para esta. En particular, una presentación equilibrada en longitud es una presentación ponderada.

Lema 3.1.3. Si un monoide M tiene una presentación ponderada $\langle X \mid R\rangle$, entonces M es atómico y todos sus átomos están contenidos en el conjunto X de generadores. Si M tiene una presentación equilibrada en longitud $\langle X \mid R\rangle$, entonces el conjunto de átomos de M coincide con X y $\|a\| = \ell(a)$ para todo $a \in M$, donde ℓ es el peso canónico en M.

Demostración. Sea $\ell : M \to \mathbb{N}$ un homomorfismo de monoides tal que $\ell(x) \geq 1$ para todo generador $x \in X$, que sabemos que existe por tener M una presentación ponderada $\langle X \mid R \rangle$. Entonces $\ell(a) \geq 1$ para todo $a \in M \backslash \{1\}$. Si $a \in M$ se expande como el producto $a_1 \cdots a_r$ con $a = a_1, ..., a_r \in M \backslash \{1\}$, entonces $\ell(a) = \ell(a_1) + \cdots + \ell(a_r) \geq r$. Por tanto $\ell(a) \geq \|a\|$. Que todos los átomos de M pertenecen a X viene del hecho de que cualquier conjunto que genere un monoide debe contener a sus átomos, ya que estos no se pueden obtener como producto de otros elementos y forman parte del monoide.

Para la segunda afirmación, al tener una presentación equilibrada en longitud, en particular, tiene una presentación ponderada, por lo que sabemos que M es atómico y el conjunto de átomos de M está incluido en X. Para comprobar la igualdad, tomamos un elemento $x \in X$ y lo expresamos como producto de elementos de $M \backslash \{1\}$, $x = a_1 \cdots a_r$ con $a_1, ..., a_r \in M \backslash \{1\}$. Como ℓ es un peso canónico tenemos que $1 = \ell(x) = \ell(a_1) + \cdots + \ell(a_r) \geq r \geq 1$. Por tanto, $\|x\| = 1$ para todo $x \in X$, es decir, todo elemento de X es un átomo. Por último, al ser M atómico, podemos expresar $a \in M$ como producto de átomos $a = a_1 \cdots a_r$ con $a_1, ..., a_r \in M \backslash \{1\}$, $\|a_i\| = 1$ para $i \in \{1, \ldots, r\}$. Entonces $\ell(a) = \ell(a_1) + \cdots + \ell(a_r) = r = \|a\|$ por ser ℓ peso canónico y $\|a_i\| = 1$ para $i \in \{1, \ldots, r\}$. □

3.1.5. El problema de palabra y el problema de divisibilidad

El *problema de palabra* para una representación $\langle X \mid R \rangle$ de un monoide M consiste en que dadas dos palabras $\omega, \omega' \in X^*$ representando a los elementos $a, a' \in M$, determinar si $a = a'$. El *problema de divisibilidad por la izquierda* (respectivamente *derecha*) está estrechamente relacionado y consiste en que dadas dos palabras $\omega, \omega' \in X^*$ representando a los elementos $a, a' \in M$, determinar si $a \preceq a'$ (respectivamente $a' \succeq a$).

Tanto el problema de palabra, como el problema de divisibilidad pueden ser fácilmente resueltos con una representación finita $\langle X \mid R \rangle$ de M. Sea $\ell : M \to \mathbb{N}$ un peso, de manera que $\ell(x) \geq 1$ para todo $x \in X$. Observar que el valor de ℓ para cualquier $a \in M$ representado por cualquier palabra no vacía $\omega \in X^*$ es mayor o igual que la longitud de ω. Sea $W(a) \subset X^*$ el conjunto de palabras representando a a. Todas esas palabras tienen longitud menor que $\ell(a)$. Ya que X es finito, el número de palabras de menor longitud que $\ell(a)$ es finito y por tanto también lo es el conjunto $W(a)$. Para listar todos los posibles elementos de $W(a)$, uno comienza con la palabra que representa a y consecutivamente aplica todas las posibles substituciones del tipo

$$\omega_1 r \omega_2 \leftrightarrow \omega_1 r' \omega_2 \quad (r, r') \in R$$

a cualquier elemento de $W(a)$ ya encontrado. Ya que R es finito, este procedimiento también es finito. Esto da una solución al problema de palabra. Dadas dos palabras

$\omega, \omega' \in X^*$ representando a $a, a' \in M$ respectivamente, sabemos que $a = a'$ si $W(a) = W(a')$.

También obtenemos una solución para el problema de divisibilidad por la izquierda y por la derecha. Se tiene que $a \preceq a'$ si y solo si algún prefijo (segmento inicial) de una palabra en $W(a')$ pertenece a $W(a)$. De igual manera, $a' \succeq a$ si y solo si algún sufijo (segmento final) de una palabra en $W(a')$ pertenece a $W(a)$.

Hemos introducido la definición de monoide y varios tipos de monoides que cumplen características específicas cada uno. Estos monoides nos han servido para poder trabajar con diferentes presentaciones de monoide, con las cuales, bajo las hipótesis de trabajar con un monoide que posea una presentación ponderada nos han aportado una solución al problema de palabra y al problema de divisibilidad. Además los conceptos vistos en esta sección nos serán de utilidad en nuestro camino a construir los monoides de Garside.

3.2. Formas normales y el problema del conjugado

Ahora vamos a estudiar el monoide M_Σ derivado de un subconjunto Σ de cierto monoide M. Bajo la presunción de hipótesis favorables vamos a obtener una forma normal para los elementos de M_Σ y resolvemos el problema del conjugado en M_Σ.

3.2.1. El monoide M_Σ

Sea M un monoide y Σ un subconjunto de M que contiene al elemento neutro 1. Sea M_Σ un monoide generado por los simbolos $[a]$, donde $a \in \Sigma$, partiendo de la definición de las relaciones $[1] = 1$ y $[a][b] = [ab]$, siempre que $a, b, ab \in \Sigma$. Existe un homomorfismo de monoides $p : M_\Sigma \to M$ definido por $p([a]) = a$ para todo $a \in \Sigma$.

Otra definición equivalente de M_Σ puede hacerse identificando el producto $[a_1]\cdots[a_r]$ en M_Σ (donde $a_,...,a_r \in \Sigma$) con la sucesión $(a_1, ..., a_r)$. Entonces M_Σ es el conjunto de clases de equivalencia de sucesiones finitas $(a_1, ..., a_r)$ de elementos de Σ bajo la equivalencia generada por

$$(a_1, ..., a_{i-1}, a_i'a_i'', a_{i+1}, ..., a_r) \sim (a_1, ..., a_{i-1}, a_i', a_i'', a_{i+1}, ..., a_r),$$

siempre que $a_i'a_i'' \in \Sigma$. La sucesión vacía es equivalente a la sucesión de un elemento (1), donde $1 \in \Sigma$. El producto en M_Σ está inducido por la concatenación de sucesiones.

Teorema 3.2.1. Sea M un monoide atómico y Σ un subconjunto de M tal que $1 \in \Sigma$ y cumple las siguientes condiciones:

1. Todos los divisores por la izquierda y por la derecha de elementos de Σ pertenecen a Σ. (Cerrado para divisores a izquierda y a derecha.)

2. Para cualesquiera $a, b, c \in \Sigma$, si $ab = ac$ o $ba = ca$, entonces b=c. (Cancelativo.)

3. Para cualesquiera $a, b \in \Sigma$, el conjunto $\{x \in \Sigma : x \preceq b,\ ax \in \Sigma\}$ tiene un elemento máximo (con respecto a $\preceq$.)

Entonces para cualquier $\xi \in M_\Sigma$, hay un único $\alpha(\xi) \in \Sigma$ tal que $[\alpha(\xi)]$ es un divisor por la izquierda de ξ que es el máximo de entre todos los divisores por la izquierda de ξ que se encuentra en el conjunto $\{[a]\}_{a\in\Sigma} \subset M_\Sigma$. Además, hay un único $\omega(\xi) \in M_\Sigma$ tal que $\xi = [\alpha(\xi)]\omega(\xi)$.

Demostración. La prueba consiste de cinco pasos.

Paso 1. Por la tercera hipótesis, para cualquier $a, b \in \Sigma$, el conjunto $\{x \in \Sigma : x \preceq b,\ ax \in \Sigma\}$ tiene un elemento máximo $c \in \Sigma$. Entonces $b = cd$ para algún $d \in M$. Por la primera hipótesis, $d \in \Sigma$, y por la segunda, d es único. Sea $\alpha_2(a, b) = ac \in \Sigma$ y $\omega_2(a, b) = d \in \Sigma$. Claramente se tiene

$$\alpha_2(a, b)\omega_2(a, b) = ab. \tag{3.1}$$

Nuestro objetivo en este paso va a ser probar las siguientes igualdades. Dados $a, b, c \in \Sigma$ tales que $a, b \in \Sigma$

$$\alpha_2(ab, c) = \alpha_2(a, \alpha_2(b, c)), \tag{3.2}$$

$$\omega_2(ab, c) = \omega_2(a, \alpha_2(a, b))\omega_2(b, c). \tag{3.3}$$

Podemos usar la siguiente observación. Si $a, b, c \in M$ satisfacen que $ac \in \Sigma$ y $ab \preceq ac$, entonces $b \preceq c$. De hecho, si $ac = abd$ con $d \in M$, entonces la presunción de que $ac \in \Sigma$ junto con la primera hipótesis implica que $a, c, bd \in \Sigma$. Por la segunda hipótesis tenemos que $c = bd$.

Por definición, $\alpha_2(b, c) = bd$, donde d es máximo tal que $d \preceq c$ y $bd \in \Sigma$. Entonces

$$\alpha_2(a, \alpha_2(b, c)) = \alpha_2(a, bd) = ad',$$

donde d' es máximo tal que $d' \preceq bd$ y $ad' \in \Sigma$. Ya que $b \preceq bd$ y $ab \in \Sigma$ (por hipótesis), al ser d' máximo se tiene que $b \preceq d'$. Escribiendo $d' = be$ con $e \in \Sigma$, obtenemos que $\alpha_2(a, bd) = abe$, con $be \preceq bd \in \Sigma$ y $abe \in \Sigma$. Por la observación hecha previamente vemos que $e \preceq d \preceq c$, así que $e \preceq c$. Ahora $\alpha_2(ab, c) = abf$, donde f es máximo tal que $f \preceq c$ y $abf \in \Sigma$. Por tanto, $e \preceq f$. Por otra parte, $f \preceq c$ y $bf \in \Sigma$. Esto y la inclusión de $abf \in \Sigma$ implica que $bf \preceq d' = be$. Por tanto, $f \preceq e$. Por el Lema 3.1.2 $e = f$ y

$$\alpha_2(ab, c) = abf = abe = ad' = \alpha_2(a, \alpha_2(b, c)).$$

Esto prueba la igualdad (3.2). Para probar la igualdad (3.3) observar (3.1) y (3.2),

$$\alpha_2(ab,c)\omega_2(a,\alpha_2(b,c))\omega_2(b,c) = \alpha_2(a,\alpha_2(b,c))\omega_2(a,\alpha_2(b,c))\omega_2(b,c)$$

$$= a\alpha_2(b,c)\omega_2(b,c) = abc = \alpha_2(ab,c)\omega_2(ab,c).$$

Por la segunda condición, para deducir (3.3) es suficiente probar que el producto $\omega_2(a,\alpha_2(b,c))\omega_2(b,c)$ pertenece a Σ. Por definición, existe $d \in \Sigma$ tal que $\alpha_2(b,c) = bd \in \Sigma$ y $c = d\omega_2(b,c)$. Sea f el elemento máximo de Σ tal que $f \preceq bd$ y $af \in \Sigma$. Ya que $b \preceq bd$ y $ab \in \Sigma$, existe $e \in \Sigma$ tal que $f = be$. Entonces

$$bd = f\omega_2(a,bd) = be\omega_2(a,bd).$$

Por la segunda condición, $d = e\omega_2(a,bd)$ y

$$e\omega_2(a,\alpha_2(b,c))\omega_2(b,c) = e\omega_2(a,bd)\omega_2(b,c) = d\omega_2(b,c) = c.$$

Esto demuestra que $\omega_2(a,\alpha_2(b,c))\omega_2(b,c)$ es divisor por la derecha de $c \in \Sigma$, por tanto es un elemento de Σ.

Paso 2. En este paso probaremos la siguiente afirmación. Existe una única función $\alpha : M_\Sigma \to \Sigma$ tal que

I. $\alpha(1) = 1$,

II. $\alpha([a]\eta) = \alpha_2(a,\alpha(\eta))$ para todo $a \in \Sigma$, $\eta \in M_\Sigma$.

Recordamos el homomorfismo de monoides $p : M_\Sigma \to M$. Para cualquier $\xi \in M_\Sigma$ definimos $H(\xi) = \|p(\xi)\| \geq 0$. Llamamos a $H(\xi)$ el peso de ξ. Está claro que

$$H(\xi\xi') \geq H(\xi) + H(\xi'),$$

para cualesquiera $\xi\xi' \in M_\Sigma$. Observar que $H(\xi) = 0$ si y solo si $\xi = 1$, y $H(\xi) = 1$ si y solo si $\xi = [a]$, donde a es un átomo de M perteneciente a Σ. Para ver esto, tomamos una expansión $\xi = [a_1]\cdots[a_r]$ con $a_1, ..., a_r \in \Sigma$. Entonces

$$H(\xi) = \|p(\xi)\| \geq \|a_1\| + \cdots \|a_r\|.$$

Si $H(\xi) = 0$, entonces $a_1 = \cdots = a_r = 1$ y $\xi = 1$. Si $H(\xi) = 1$, entonces todos los elementos $a_1, ..., a_r \in \Sigma$ son iguales a 1 excepto un elemento, el cual es un átomo.

Dado $\xi \in M_\Sigma$, definimos $\alpha(\xi)$ por inducción sobre el peso de ξ. Para $\xi = 1$ establecemos $\alpha(\xi) = 1 \in \Sigma$. Si $H(\xi) = 1$, entonces $\xi = [a] = [a]1$ para algún átomo $a \in \Sigma$, y para satisfacer (ii) establecemos que $\alpha(\xi) = \alpha_2(a,1) = a$.

Tomamos un entero $k \geq 1$ y suponemos que $\alpha(\xi)$ está definido para todo ξ de peso menor o igual que k de manera que las condiciones (i), (ii) se satisfacen siempre que $H([a]\eta) \leq k$. Sea ξ un elemento de M_Σ de peso $k+1$. Podemos expandir $\xi = [a]\eta$, donde $a \in \Sigma$, $a \neq 1$ y $\eta \in M_\Sigma$. Entonces $H([a]) \geq 1$ y $H([\eta]) < H([a]\eta)$, así que $\alpha(\eta)$ está ya definido. Para satisfacer (ii) establecemos $\alpha(\xi) = \alpha_2(a, \alpha(\eta))$. Debemos comprobar que $\alpha_2(a, \alpha(\eta))$ no depende de la elección de la expansión $\xi = [a]\eta$. Por definición de M_Σ y por la hipótesis de inducción, es suficiente con comprobar que

$$\alpha_2(a, \alpha(\eta)) = \alpha_2(a', \alpha([a'']\eta)),$$

donde $a = a'a''$ con $a', a'' \in \Sigma\backslash\{1\}$. Ya que

$$H([a'']\eta) \leq H([a]\eta) - H([a']) < H([a]\eta),$$

la hipótesis de inducción nos da que $\alpha([a'']) = \alpha_2(a'', \alpha(\eta))$. Por la igualdad (3.2)

$$\alpha_2(a', \alpha([a'']\eta)) = \alpha_2(a', \alpha_2(a'', \alpha(\eta)))$$

$$= \alpha_2(a'a'', \alpha(\eta)) = \alpha_2(a, \alpha(\eta)).$$

Por tanto α está bien definido para los elementos de M_Σ de peso menor o igual que $k+1$ y satisface las condiciones (i) y (ii). Esto completa la inducción y prueba nuestra afirmación.

Paso 3. Ahora comprobamos que para cualquier $\xi \in M_\Sigma$, el elemento $[\alpha(\xi)] \in M_\Sigma$ es un divisor por la izquierda de ξ que es máximo entre todos los divisores por la izquierda de ξ que se encuentrar en el conjunto $\{[a]\}_{a\in\Sigma} \subset M_\Sigma$.

Utilizando la proyección $p : M_\Sigma \to M$, es fácil demostrar que todos los divisores de $1 \in M_\Sigma$ son iguales a 1. Por tanto si $H(\xi) = 0$, entonces $[\alpha(\xi)] = \xi = 1$ es el único divisor por la izquierda de $1 \in \Sigma$. Si $H(\xi) \geq 1$, escribimos $\xi = [a]\eta$ para algún $a \in \Sigma\backslash\{1\}$ y $\eta \in M_\Sigma$. Entonces $H(\eta) < H(\xi)$ y $\alpha(\xi) = \alpha_2(a, \alpha(\eta))$. Por tanto, $\alpha(\xi) = ab$ para algún $b \in \Sigma$ tal que $b \preceq \alpha(eta)$. Por el supuesto de inducción, $[\alpha(\eta)]$ es un divisor por la izquierda de η. Por tanto,

$$[\alpha(\xi)] = [ab] = [a][b] \preceq [a][\alpha(\eta)] \preceq [a]\eta = \xi.$$

Esto demuestra que $[\alpha(\xi)]$ es divisor por la izquierda de ξ perteneciente a $\{[a]\}_{a\in\Sigma}$. Veamos que $[\alpha(\xi)]$ es máximo. Supongamos que $\xi = [a']\eta'$ para algún $a' \in \Sigma$, $\eta' \in M_\Sigma$. Entonces $\alpha(\xi) = \alpha_2(a', \alpha(\eta')) = a'b'$ para algún $b' \in \Sigma$. Por tanto, $a' \preceq \alpha(\xi)$ y $[a'] \preceq [\alpha(\xi)]$.

Paso 4. Existe una única función $\omega : M_\Sigma \to \Sigma$ tal que

I. $\omega(1) = 1$,

II. $\omega([a]\eta) = [\omega_2(a, \alpha(\eta))]\omega(\eta)$ para todo $a \in \Sigma$, $\eta \in M_\Sigma$.

El valor de ω para cualquier $\xi \in M_\Sigma$ está definido por inducción sobre el peso de ξ. Establecemos que $\omega(1) = 1$. Tomamos un entero $k \geq 1$ y suponemos que $\omega(\xi)$ está definido para todo ξ de peso menor o igual que k, de manera que las condiciones (i) y (ii) se satisfacen siempre que $H([a]\eta) \leq k$. Sea $\xi = [a]\eta$ un elemento de M_Σ de peso $k+1$ con un $q \in \Sigma\backslash\{1\}$ y $\eta \in M_\Sigma$. Entonces $H(\eta) < H(\xi)$ y $\omega(\eta)$ está definido. Para satisfacer (ii), establecemos que

$$\omega(\xi) = [\omega_2(a, \alpha(\eta))]\omega(\eta).$$

Debemos de comprobar que $\omega(\xi)$ es independiente de la elección de expansión $\xi = [a]\eta$. Por definición de M_Σ y la inducción de hipótesis, es suficiente con comprobar que

$$\omega_2(a, \alpha(\eta))\omega(\eta) = \omega_2(a', \alpha([a'']\eta))\omega([a'']\eta),$$

donde $a = a'a''$ con $a', a'' \in \Sigma\backslash\{1\}$. Sabemos que $\alpha([a'']\eta) = \alpha_2(a'', \alpha(\eta))$. Ya que $H([a'']\eta) < H([a]\eta)$, la hipótesis de inducción nos da la igualdad $\omega([a'']\eta) = [\omega_2(a'', \alpha(\eta))]\omega(\eta)$. Por la igualdad (3.3)

$$\omega_2(a', \alpha([a'']\eta))\omega([a'']\eta) = \omega_2(a', \alpha_2(a'', \alpha(\eta)))\omega([a'']\eta)$$

$$= \omega_2(a', \alpha_2(a'', \alpha(\eta)))[\omega_2(a'', \alpha(\eta))]\omega(\eta)$$

$$= \omega_2(a'a'', \alpha(\eta))\omega(\eta) = \omega_2(a, \alpha(\eta))\omega(\eta).$$

Por tanto ω está bien definido para los elementos de M_Σ de peso menor o igual que $k+1$ y satisface las condiciones (i) y (ii). Esto completa la inducción y prueba nuestra afirmación.

Paso 5. Para completar la prueba, tenemos que demostrar que para cualquier $\xi \in M_\Sigma$, el elemento $\eta = \omega(xi)$ es el único elemento de M_Σ tal que $\xi = [\alpha(\xi)]\eta$. Procedemos por inducción sobre $H(\xi)$. Si $H(\xi) = 0$, entonces $\xi = 1$, $\alpha(\xi) = 1$, $\omega(\xi) = 1$, y la afirmación es obvia. Supongamos que la afirmación se da para todo ξ de altura menor o igual a k para cualquier entero $k \geq 1$. Sea ξ un elemento de M_Σ de altura $k+1$. Por el Paso 3, $\xi = [\alpha(\xi)]\eta$ con $\eta \in M_\Sigma$. Claramente, $\alpha(\xi) \neq 1$ y por tanto $H(\eta) < H(\xi)$.

Establecemos $\theta = [\alpha(\xi)][\alpha(\eta)]$. Por definición de la función α,

$$\alpha(\theta) = \alpha_2(\alpha(\xi), \alpha(\eta)) = \alpha(\xi)b$$

para algún $b \in \Sigma$ tal que $[b] \preceq [\alpha(\eta)] \preceq \eta$. Por tanto $\alpha(\xi) \preceq \alpha(\theta)$ y

$$[\alpha(\theta)] = [\alpha(\xi)][b] \preceq [\alpha(\xi)]\eta = \xi.$$

Ya que $\alpha(\xi)$ es el máximo divisor por la izquierda de ξ en el conjunto $\{[a]\}_{a\in\Sigma}$, tenemos $[\alpha(\xi)] = [\alpha(\theta)]$. Proyectando a M, concluimos que

$$\alpha(\xi) = \alpha(\theta) = \alpha_2(\alpha(\xi), \alpha(\eta)).$$

Por tanto, $\alpha(\xi)\alpha(\theta) = \alpha_2(\alpha(\xi), \alpha(\eta))\alpha(\eta)$. Por otro lado, por (3.1),

$$\alpha(\xi)\alpha(\theta) = \alpha_2(\alpha(\xi), \alpha(\eta))\omega_2(\alpha(\xi), \alpha(\eta)).$$

Combinando estas ecuaciones obtenemos

$$\alpha_2(\alpha(\xi), \alpha(\eta))\alpha(\eta) = \alpha_2(\alpha(\xi), \alpha(\eta))\omega_2(\alpha(\xi), \alpha(\eta)).$$

Por la segunda condición del teorema, podemos cancelar $\alpha_2(\alpha(\xi), \alpha(\eta))$. Así que $\alpha(\eta) = \omega_2(\alpha(\xi), \alpha(\eta))$ y

$$\omega(\xi) = \omega([\alpha(\xi)]\eta) = [\omega_2(\alpha(\xi), \alpha(\eta))]\omega(\eta) = [\alpha(\eta)]\omega(\eta) = \eta,$$

donde la última igualdad viene de la hipótesis de inducción. Esto demuestra que $\xi = [\alpha(\xi)\omega(\xi)]$ y que cualquier $\eta \in M_\Sigma$ satisfaciendo $\xi = [\alpha(\xi)]\eta$ es igual a $\omega(\xi)$. □

3.2.2. Forma nomal en M_Σ

Bajo las hipótesis del Teorema 3.2.1 , cualquier ξ puede ser expandido inductivamente

$$\xi = [\alpha(\xi)]\omega(\xi) = [\alpha(\xi)][\alpha(\omega(\xi))]\omega^2(\xi)$$
$$= [\alpha(\xi)][\alpha(\omega(\xi))][\alpha(\omega^2(\xi))]\omega^3(\xi) = \cdots .$$

Este proceso de expansión puede ser parado en el r-ésimo paso, donde r el el mínimo entero tal que $\omega^{r+1}(\xi) = 1$. Tal r existe ya que una expansión de ξ como producto de generadores de $[a]$ con $a \in \Sigma$ tiene como mucho un cierto k (visto en la demostración del Teorema 3.2.1) de generadores distintos de 1.

Esta observación nos lleva a una forma normal para cada elemento de M_Σ. Una *forma normal* para $\xi \in M_\Sigma$ es una sucesión $(a_1, a_2, ..., a_r)$ de elementos de Σ, todos diferentes de 1, tal que $\xi = [a_1][a_2]\cdots[a_r]$ y

$$a_i = \alpha([a_i][a_{i+1}]\cdots[a_r])$$

para todo $i \in \{1, 2, ..., r\}$. La unicidad en la última afirmación del Teorema 3.2.1 implica la unicidad de la forma normal.

3.2.3. La cancelatividad de M_Σ

Usamos el Teorema 3.2.1 y la función $\alpha_2 : \Sigma \times \Sigma \to \Sigma$ introducida en la prueba para establecer la cancelatividad por la izquierda de M_Σ.

Lema 3.2.1. Bajo las hipótesis del Teorema 3.2.1, el monoide M_Σ es cancelativo por la izquierda.

Demostración. Necesitamos demostrar que $\xi\eta = \xi\theta$ implica $\eta = \theta$ para todo $\xi, \eta, \theta \in M_\Sigma$. Suponemos primero que $\xi = [a]$ para algún $a \in \Sigma$. Entonces $\alpha(\xi\eta)) = \alpha_2(a, \alpha(\eta)) = ab \in \Sigma$ para algún $b \in \Sigma$ tal que $b \preceq \alpha(\eta)$. Así

$$ab = \alpha(\xi\eta) = \alpha(\xi\theta) = \alpha([a]\theta) = \alpha_2(a, \alpha(\theta)) = ac$$

para algún $c \preceq \alpha(\theta)$ implica que $b = c \preceq \alpha(\theta)$. Entonces existen $\eta', \theta' \in M_\Sigma$ tales que $[b]\eta' = \eta$ y $[b]\theta' = \theta$. Como ya sabemos, $\omega(\xi\eta)$ es el único elemento $x \in M_\Sigma$ tal que $\xi\eta = [\alpha(\xi\eta)]x = [ab]x$. Ya que

$$\xi\eta = [a][b]\eta' = [ab]\eta',$$

tenemos que $\omega(\xi\eta) = \eta'$. De igual manera, $\omega(\xi\theta) = \theta'$. Por tanto,

$$\eta' = \omega(\xi\eta) = \omega(\xi\theta) = \theta',$$

$$\eta = [b]\eta' = [b]\theta' = \theta.$$

En general, $\xi = [a_1][a_2] \cdots [a_r]$ con $a_1, a_2, ..., a_r \in \Sigma$. Como ya sabemos, $[a_1][a_2] \cdots [a_r]\eta = [a_1][a_2] \cdots [a_r]\theta$ implica que $[a_2] \cdots [a_r]\eta = [a_2] \cdots [a_r]\theta$. Repitiendo este proceso, obtenemos que $\eta = \theta$. □

3.2.4. El problema de palabra en M_Σ

Decimos que el conjunto $\Sigma \subset M$ está ponderado si existe una función $\ell : \Sigma \to \mathbb{N}$ tal que $\ell(1) = 0$, $\ell(a) \geq 1$ para $a \neq 1$, y $\ell(ab) = \ell(a) + \ell(b)$ siempre que $a, b, ab \in \Sigma$. La función ℓ se extiende al homomorfismo de monoides $M_\Sigma \to \mathbb{N}$ que convierte la presentación de M_Σ en una presentación ponderada. Además, si Σ es finito, ya hemos visto una solución al problema de palabra y al problema de divisibilidad en M_Σ.

3.2.5. El problema del conjugado en M_Σ

El *problema del conjugado en un grupo G* consiste en encontrar un procedimiento que nos permita, dados $\alpha, \beta \in G$, decidir si existe $\gamma \in G$ tal que $\alpha = \gamma\beta\gamma^{-1}$ o, equivalentemente, $\alpha\gamma = \gamma\beta$. Por extensión, el *problema del conjugado en un monoide M* consiste en encontrar un procedimiento que nos permita, dados $a, b \in M$, decidir si existe $c \in M$ tal que $ac = cb$. El siguiente lema es la clave para el problema del conjugado en M_Σ.

Lema 3.2.2. Sea $M, \Sigma \subset M$ satisfaciendo las hipótesis del Teorema 3.2.1. Dados $a, b \in M_\Sigma$, existe $c \in M_\Sigma$ tal que $ac = cb$ si y solo si existe una una sucesión $a = a_0, a_1, \ldots, a_r = b$ de elementos de M_Σ y una sucesión $c_1, \ldots, c_r$ de elementos de Σ tal que

$$a_{i-1}[c_i] = [c_i]a_i$$

para todo $i \in \{1, \ldots, r\}$.

Demostración. Si nosotros tenemos tales sucesiones, entonces $ac = cb$ para $c = [c_1][c_2]\cdots[c_r]$. En la otra implicación tenemos que $c \in M_\Sigma$ tal que $ac = cb$. Demostramos la afirmación por inducción en la longitud r de la forma normal $(c_1, \ldots, c_r)$ de c. Si $r = 1$, entonces $c = [c_1]$ y queda probado. Supongamos que $r \geq 2$. Ya que

$$[c_1] = [\alpha(c)] \preceq c \preceq cb = ac,$$

tenemos que

$$c_1 \preceq \alpha(ac) = \alpha_2(a, \alpha(c)) = \alpha_2(a, c_1).$$

Por tanto,

$$[c_1] \preceq [\alpha_2(a, c_1)] \preceq ac_1.$$

Existe $a_1 \in M_\Sigma$ tal que

$$[c_1]a_1 = a[c_1],$$

por lo que obtenemos que

$$[c_1]a_1[c_2]\cdots[c_r] = a[c_1][c_2]\cdots[c_r] = ac = cb = [c_1][c_2]\cdots[c_r]b.$$

Por el Lema 3.2.1, podemos dividir por la izquierda por $[c_1]$. Esto nos da que $a_1c' = c'b$, donde $c' = [c_2]\cdots[c_r]$ tiene una forma normal de longitud $r - 1$ a la que se le puede aplicar la hipótesis de inducción. □

El Lema 3.2.2 nos da una solución al problema del conjugado en M_Σ. Suponemos que M, Σ satisfacen las condiciones del Teorema 3.2.1 y que Σ es finito. Suponemos también que M_Σ admite una presentación ponderada finita de manera que el problema de palabra tiene solución. Para determinar si dos elementos $a, b \in M_\Sigma$ son conjugados, primero hay que observar que elementos conjugados de M_Σ tienen el mismo peso. Ya que solo hay un número finito de elementos de M_Σ para un peso específico, a tiene un número finito de conjugados. El Lema 3.2.2 nos dice que, con el objetivo de encontrarlos todos, es suficiente aplicar todas las posibles conjugaciones por elementos de Σ para conocer todos los conjugados de a hasta que no se encuentren nuevos elementos. Así obtenemos una lista finita $a_1, \ldots, a_s$ de conjugados de a en M_Σ. Si $b = a_i$ para algún i, entonces b es conjugado de a. Si no es así, entonces b no es conjugado de a.

3.2.6. Conjuntos exhaustivos

Un conjunto $\Sigma \subset M$ es *exhaustivo* si $1 \in \Sigma$ y M tiene una presentación cuyos generadores y relaciones pertenecen a Σ.

Lema 3.2.3. Si Σ es un subconjunto exhaustivo de un monoide M tal que todos los divisores por la izquierda de los elementos de Σ pertenecen a Σ, entonces el homomorfismo de monoides $p : M_\Sigma \to M$ es un isomorfismo.

Demostración. Ya que Σ contiene un conjunto de generadores de M, el homomorfismo p es sobreyectivo. Solo necesitamos probar que es inyectivo. Observamos primero que si $a_1, a_2, \ldots, a_n$ son elementos de Σ con $n \geq 2$ tales que $a = a_1 a_2 \cdots a_n \in \Sigma$, entonces $[a] = [a_1][a_2] \cdots [a_n]$. De hecho, $a_1 a_2$ es divisor por la izquierda de a y por tanto $a_1 a_2 \in \Sigma$. Por definición de M_Σ, tenemos que $[a_1][a_2] = [a_1 a_2]$. Continuando por inducción obtenemos que $[a_1][a_2] \cdots [a_n] = [a]$.

Cosideramos ahora una presentación $\langle X \mid R \rangle$ de M por generadores y relaciones y sea $P : X^* \to M$ la proyección natural. Asumimos que $P(x) \in \Sigma$ para todo $x \in X$ y $P(r) = P(r') \in \Sigma$ para todo $(r, r') \in R$. Definimos un homomorfismo de monoides $Q : X^* \to M_\Sigma$ por $Q(x) = [P(x)]$ para $x \in X$. La observación anterior implica que para cualquier $r \in X^*$ con $P(r) \in \Sigma$, tenemos que $Q(r) = [P(r)]$. Por tanto para cualquier relación $(r, r') \in R$ tenemos que

$$Q(r) = [P(r)] = [P(r')] = Q(r').$$

Esto implica que existe un homomorfismo de monoides $q : M \to M_\Sigma$ tal que $Q = qP$. Entonces

$$qp([P(x)]) = q(P(x)) = Q(x) = [P(x)]$$

para todo $x \in X$. Ya que el conjunto $P(X)$ genera M, tenemos $qp = id$. Por tanto p es inyectiva. □

El Lema 3.2.3 nos muestra que bajo las hipótesis apropiadas sobre Σ tenemos que $M_\Sigma \cong M$, así que todas las propiedades de M_Σ se transladan a M.

Los conceptos de monoide atómico y del monoide M_Σ nos han permitido enunciar el Teorema 3.2.1. Este teorema posee gran relevancia, ya que bajo las hipótesis que presupone hemos obtenido en M_Σ una forma normal, la propiedad de cancelatividad, una solución al problema de palabra (heredada de la solución de la Sección 3.1.5) y una solución al problema del conjugado. Además, la definición de conjunto exhaustivo nos proporciona una forma de saber cuando podemos identificar el monoide M_Σ con el monoide M del que deriva.

3.3. Grupo de fracciones y monoides pre-Garside

Hasta ahora los resultados obtenidos han sido para monoides. En última instancia nosotros vamos a trabajar con el grupo de trenzas y nos interesa ver cómo podemos obtener resultados sobre grupos a partir de los ya obtenidos sobre monoides. Con este fin a continuación vamos a introducir los grupos de fracciones y los monoides pre-Garside.

3.3.1. Grupo de fracciones

Un homomorfismo de monoides $i : M \to G$ se dice que es *universal* si G es un grupo y para cualquier homomorfismo de monoides f de M a un grupo arbitrario G', existe un único homomorfismo de grupos $g : G \to G'$ tal que $f = gi$. Cada monoide M admite un homomorfismo universal a un grupo. Para ver esto, tomamos una representación arbitraria $\langle X \mid R\rangle$ de M por generadores y relaciones, y consideramos el grupo G definido por $\langle X \mid R\rangle$ visto como una presentación de grupo. La función identidad $id_X : X \to X$ extiende a un homomorfismo de monoides $M \to G$, el cual es fácil ver que es universal. La definición de un homomorfismo universal $i : M \to G$ implica que es único bajo composición con un isomorfismo de grupos. En particular, el grupo G extendido del monoide M está bien definido bajo isomorfismo. Este grupo llamado *grupo de fracciones* de M es denotado por G_M. Una presentación de G_M por generadores y relaciones se puede obtener tomando una representación del monoide M arbitraria de generadores y relaciones y verla como una representación de grupo.

Un monoide M es *embebible* si existe un homomorfismo de monoides inyectivo de M a un grupo. Es claro que M es embebible si y solo si el homomorfismo universal de $M \to G_M$ es inyectivo. Por ejemplo, la inclusión $\mathbb{N} \hookrightarrow \mathbb{Z}$ demuestra que $\mathbb{N}$ es embebible. Es fácil ver que la inclusión es un homomorfismo universal tal que $G_{\mathbb{N}} = \mathbb{Z}$.

Claramente, los monoides embebibles son cancelativos por la izquierda y por la derecha. Por ejemplo, el monoide $\{1, x\}$, con $xx = x$ no es cancelativo por la izquierda. El grupo de fracciones de este monoide es el grupo trivial.

3.3.2. Monoides pre-Garside

Definición 3.3.1. Un monoide pre-Garside es un par (M, Δ) que consiste de un monoide M y un elemento Δ de M tal que el conjunto $\Sigma = \Sigma_\Delta$ de divisores por la izquierda de Δ satisface las siguientes condiciones:

1. Σ es finito, genera M y coincide con el conjunto de divisores por la derecha de Δ.
2. Si $a, b \in \Sigma$ cumplen que $\Delta a = \Delta b$ o $a\Delta = b\Delta$, entonces $a = b$.

El elemento $\Delta \in M$ se llama *elemento de Garside* de M. Observar que el conjunto Σ de los divisores de Δ es cerrado bajo la divisibilidad por la izquierda y por la derecha, es decir, todos los divisores por la izquierda y por la derecha de elementos de Σ pertenecen a Σ. Claramente, $1 \in \Sigma$ y $\Delta \in \Sigma$.

Lema 3.3.1. Sea (M, Δ) un monoide pre-Garside y sea Σ el conjunto de divisores de Δ.

(I). Para todo $a, b, c \in \Sigma$, si $ac = bc$ o $ca = cb$, entonces $a = b$.

(II). Existe una biyección $\delta : \Sigma \to \Sigma$ tal que $\Delta a = \delta(a)\Delta$ para todo $a \in \Sigma$.

(III). Si N es el orden de δ (es decir, el menor entero positivo tal que $\delta^N = id$), entonces $\Delta^N a = a\Delta^N$ para todo $a \in M$.

(IV). Para cualquier $a \in M$, existe un entero $r \geq 1$ tal que $a \preceq \Delta^r$ y $\Delta^r \succeq a$.

Demostración. (i) Ya que $c \in \Sigma$, existe un $d \in M$ tal que $cd = \Delta$. Entonces $ac = bc$ implica $a\Delta = acd = bcd = b\Delta$. Por tanto $a = b$ por la segunda condición de la Definición 3.3.1. Para la implicación $ca = cb$ implica $a = b$ se procede de igual forma.

(ii) Ya que cualquier divisor por la izquierda de Δ es también un divisor por la derecha y viceversa, para cualquier $a \in \Sigma$, existe $a', \delta(a) \in \Sigma$ tal que $\Delta = a'a$ y $\Delta = \delta(a)a'$. Por la afirmación (i), a' y $\delta(a)$ están definidas unívocamente. Tenemos que

$$\Delta a = \delta(a)a'a = \delta(a)\Delta. \tag{3.4}$$

Ya que Σ es finito, con el objetivo de probar que la función $\delta : \Sigma \to \Sigma$ es biyectiva, es suficiente probar que es inyectiva. La igualdad $\delta(a) = \delta(b)$ nos da que

$$\Delta a = \delta(a)\Delta = \delta(b)\Delta = \Delta b.$$

Esto implica que $a = b$ por la segunda condición de la Definición 3.3.1.

(iii) Por inducción sobre n obtenemos de (3.4) que $\Delta^n a = \delta^n(a)\Delta^n$ para todo $a \in \Sigma$. Ya que $\delta^N = id$, tenemos que $\Delta^N a = \delta^N(a)\Delta^n = a\Delta^n$. En otras palabras, Δ^N conmuta con todos los elementos del conjunto Σ. Como este conjunto genera el monoide M, podemos concluir que Δ^N conmuta con todos los elementos de M.

(iv) Escribimos a como producto $a = a_1 \cdots a_r$ de r elementos de Σ. Para cada a_i

hay un $b_i \in \Sigma$ tal que $b_i a_i = \Delta$. Sea $b = \delta^{r-1}(b_r) \cdots \delta(b_2) b_1$. Afirmamos que $ba = \Delta^r$, lo cual prueba que $\Delta^r \succeq a$.

$$\begin{aligned} ba &= \delta^{r-1}(b_r) \cdots \delta(b_2) b_1 a_1 \cdots a_r \\ &= \delta^{r-1}(b_r) \cdots \delta(b_2) \Delta a_2 \cdots a_r \\ &= \delta^{r-1}(b_r) \cdots \Delta b_2 a_2 \cdots a_r \\ &= \delta^{r-1}(b_r) \cdots \delta^2(b_3) \Delta^2 a_3 \cdots a_r \\ &= \delta^{r-1}(b_r) \cdots \Delta^2 b_3 a_3 \cdots a_r \\ &= \cdots = \Delta^{r-1} b_r a_r = \Delta^r. \end{aligned}$$

De forma similar se prueba que si $a_i c_i = \Delta$ para $i \in \{1, \ldots, r\}$ y

$$c = c_r \delta^{-1}(c_{r-1}) \cdots \delta^{-(r-1)}(c_1)$$

entonces $ac = \Delta^r$ y por tanto $a \preceq \Delta^r$. □

3.3.3. Embebilidad de los monoides pre-Garside

Sea (M, Δ) un monoide pre-Garside. Bajo la hipótesis de que M es cancelativo por la izquierda, damos una construcción explicita del grupo de fracciones de M. La construcción implicará que M es embebible.

Sea $N \geq 1$ el orden de $\delta : \Sigma \to \Sigma$. Por el Lema 3.3.1, Δ^N es central en M. Consideramos el producto $H = M \times \mathbb{N}$ de los monoides M y $\mathbb{N}$ con la correspondiente multiplicación

$$(a, p)(b, q) = (ab, p + q)$$

para todo $a, b \in M$ y $p, q \in \mathbb{N}$. El elemento neutro de H es $(1, 0)$.

Definimos la relación $\sim$ en H por $(a, p) \sim (b, q)$ si $\Delta^{qN} a = \Delta^{pN} b$. Por ejemplo, $(\Delta^N, 1) \sim (1, 0)$. Vamos a demostrar que $\sim$ es una relación de equivalencia. La reflexividad y la simetría son obvias. Comprobamos la transitividad. Supongamos que $(a, p) \sim (b, q) \sim (c, r)$. Entonces $\Delta^{qN} a = \Delta^{pN} b$ y $\Delta^{rN} b = \Delta^{qN} c$. Por tanto,

$$\begin{aligned} \Delta^{qN} \Delta^{rN} a &= \Delta^{rN} \Delta^{qN} a = \Delta^{rN} \Delta^{pN} b \\ &= \Delta^{pN} \Delta^{rN} b = \Delta^{pN} \Delta^{qN} c = \Delta^{qN} \Delta^{pN} c. \end{aligned}$$

Ya que M es cancelativo por la izquierda, obtenemos que $\Delta^{rN} a = \Delta^{pN} c$. Esto nos da $(a, p) \sim (c, r)$.

Sea $G = H/\sim$ el conjunto de clases de equivalencia y sea $\pi : H \to G$ la proyección. Ya que Δ^N es central en M, el conjunto G tiene una única estructura de monoide tal que π es un homomorfismo de monoides. Definimos un homomorfismo de monoides $i : M \to G$ por $i(a) = \pi(a, 0)$ para $a \in M$.

Teorema 3.3.1. Sea (M, Δ) un monoide pre-Garside tal que M es cancelativo por la izquierda.

(I) El monoide G construido anteriormente es un grupo y el homomorfismo $i : M \to G$ es inyectivo.

(II) Cualquier elemento de G puede ser escrito de la forma $i(\Delta)^s i(a)$, donde $s \in \mathbb{Z}, a \in M$.

(III) El homomorfismo de monoides $i : M \to G$ es universal y por tanto G es el grupo de fracciones G_M de M.

Demostración. (i) Si $i(a) = i(b)$ para $a, b \in M$, entonces $(a, 0) = (b, 0)$ en H. Esto implica que $a = \Delta^0 a = \Delta^0 b = b$ quedando demostrada la inyectividad de i.

Cualquier elemento $g \in G$ tiene la forma $\pi(a, p)$ para unos ciertos $a \in M$ y $p \in \mathbb{N}$. Vamos a comprobar que $g = \pi(a, p)$ es invertible. Por el Lema 3.3.1(iv) existe un $b \in M$ y un entero $r \geq 1$ tal que $ab = \Delta^r$. Multiplicando b en la derecha por una potencia de Δ, podemos asumir que $r = qN$ para un entero $q \geq p$. Entonces

$$(a, p)(b, q - p) = (ab, q). \tag{3.5}$$

Ya que $\Delta^0 ab = \Delta^r = \Delta^{qN} 1$, tenemos que $(ab, q) \sim (1, 0)$ y $\pi(ab, q) = \pi(1, 0) = 1$. Esto prueba que $g = \pi(a, p)$ tiene inverso por la derecha, que denotamos por g'. Aplicando el mismo razonamiento, g' también tiene un inverso por la derecha g'' y

$$g = g(g'g'') = (gg')g'' = g''.$$

En otras palabras, g' es también el inverso de g por la izquierda. Esto demuestra que G es un grupo.

(ii) Sea ξ el elemento central $(1, 1) \in H = M \times \mathbb{N}$. Si establecemos $a = 1, b = \Delta^N$ y $p = q = 1$ en (3.5), obtenemos $\pi(\xi) i(\Delta)^N = 1$. Por tanto $\pi(\xi) = i(\Delta)^{-N}$. Cualquier elemento de H es de la forma $(a, p) = \xi^p (a, 0)$ para algún $a \in M$ y $p \in \mathbb{N}$. Por tanto cualquier elemento de G puede ser escrito de la forma $\pi(\xi)^p i(a) = i(\Delta)^{-pN} i(a)$ donde $a \in M$ y $p \in \mathbb{N}$.

(iii) Dado un homomorfismo de monoides f de M a un grupo G', consideramos la función $H = M \times \mathbb{N} \to G'$ que envía cualquier par $(a, p) \in M \times \mathbb{N}$ a $f(\Delta)^{-pN} f(a)$. Esta función es constante en las clases de equivalencia de H por $\sim$ e induce un homomorfismo de grupos $H/\sim = G \to G'$. La composición de esta última función con $i : M \to G$ es igual a f. La unicidad del homomorfismo de grupos $G \to G'$ cuya composición con i es igual a f viene del hecho de que $i(M)$ genera G como grupo. □

Colorario 3.3.1. Los monoides pre-Garside cancelativos por la izquierda son embebibles.

Este colorario muestra en particular que para monoides pre-Garside, la cancelatividad por la izquierda implica la cancelatividad por la derecha.

A continuación identificaremos los elementos cancelativos por la izquierda de un monoide pre-Garside M con sus imágenes en G_M, de manera que M se convierte en un subconjunto de G_M.

3.3.4. El problema del conjugado en el grupo de fracciones

Sea (M, Δ) un monoide pre-Garside cancelativo por la izquierda. El problema del conjugado en su grupo de fracciones $G = G_M$ puede ser reducido al problema del conjugado en M de la siguiente manera. Como ya sabemos, para cualquier $\alpha, \beta \in G$, existen $a, b \in M \subset G$ y $s, t \in \mathbb{Z}$ tal que $\alpha = \Delta^s a$ y $\beta = \Delta^t b$ (Teorema 3.3.1 (ii)). Cogemos un entero u tal que $u \leq \text{mín}(s, t)$ y u es divisible por N (Lema 3.3.1(iii)). Sea $a' = \Delta^{s-u} a \in M$ y $b' = \Delta^{t-u} b \in M$. Claramente, $\alpha = \Delta^u a'$ y $\beta = \Delta^u b'$. Afirmamos que α es conjugado de β en G si y solo si a' es conjugado de b' en M. Supongamos que $a'c = cb'$ para algún $c \in M$. Ya que Δ^u es una potencia de Δ^N y es por tanto central en M,

$$\alpha c = \Delta^u a' c = \Delta^u c b' = c \Delta^u b' = c\beta.$$

En el otro sentido, si $\alpha\gamma = \gamma\beta$ con $\gamma \in G$, entonces $\gamma = \Delta^v c$ para algún $c \in M$ y algún entero v divisible por N. Reemplazando $\alpha\beta\gamma$ en la fórmula $\alpha\gamma = \gamma\beta$ por sus expansiones con a', b', c' y usando la centralidad de Δ^u, Δ^v, obtenemos

$$\Delta^{u+v} a' c = \Delta^{u+v} c b'.$$

Dividimos entre Δ^{u+v} y concluimos que $a'c = cb'$.

3.3.5. El caso de M atómico

Para M atómico, la afirmación (ii) del Teorema 3.3.1 admite el siguiente refinamiento.

Teorema 3.3.2. Sea (M, Δ) un monoide pre-Garside tal que M es no trivial, cancelativo por la izquierda y atómico. Entonces cualquier elemento de $G = G_M \supset M$ puede ser escrito unicamente de la forma $\Delta^s b$, donde $s \in \mathbb{Z}$ y b es un elemento de M que no es múltiplo por la derecha de Δ. Esta forma la denominaremos *forma normal greedy*.

Demostración. Observamos primero que $\|\Delta\| > 0$. En efecto, si $\|\Delta\| = 0$, entonces $\Delta = 1$. Ya que M es atómico. Eso implica que $\Sigma_\Delta = \{1\}$. Ya que Σ_Δ genera M, tenemos que $M = \{1\}$. Esto contradice la no trivialidad de M.

Por el Teorema 3.3.1, cualquier elemento de G tiene la forma $\Delta^s a$ con $s \in \mathbb{Z}$ y $a \in M$. Sea t el mayor entero no negativo tal que $\Delta^t \preceq a$ en M. Tal t existe porque la relación $\Delta^t \preceq a$ implica que

$$t\|\Delta\| \leq \|\Delta^t\| \leq \|a\| < \infty.$$

Entonces $a = \Delta^t b$ para algún $b \in M$ tal que $\Delta \not\preceq b$ y $\Delta^s a = \Delta^{s+t} b$. Esto prueba la existencia de la forma declarada.

Supongamos que $\Delta^s b = \Delta^{s'} b'$ para $s, s' \in \mathbb{Z}$ y $b, b' \in M$ tal que $\Delta \not\preceq b$ y $\Delta \not\preceq b'$. Podemos asumir que $s \geq s'$. Dividiendo por $\Delta^{s'}$, obtenemos que $\Delta^{s-s'} b = b'$. Ya que b' no es múltiplo por la derecha de Δ en M, tenemos que $s - s' = 0$. Por tanto, $s = s'$ y $b = b'$, lo que prueba la unicidad. □

Hemos podido establecer un monomorfismo entre un monoide pre-Garside y su grupo de fracciones. De esta forma podemos identificar el monoide pre-Garside como un subconjunto del grupo de fracciones. Por el Teorema 3.3.2 hemos obtenido una forma normal para el grupo de fracciones como producto de una potencia del elemento de Garside y elementos del monoide. Esta forma normal greedy será la que finalmente utilizaremos para trabajar con el grupo de trenzas. Además, podemos reducir el problema del conjugado en el grupo de fracciones a un problema del conjugado en el monoide equivalente.

3.4. Monoides de Garside

Hemos llegado al punto en el que vamos a definir los monoides de Garside, la estructura que vamos buscando desde el inicio de la sección y que, concretamente los monoides de Garside exhaustivos, reunirán todas las propiedades deseables que hemos ido obteniendo.

3.4.1. Definiciones y lemas

Sea (M, Δ) un monoide pre-Garside y sea Σ el conjunto de divisores por la izquierda (y por la derecha) de Δ. Observamos que al generar Σ a M, todos los átomos de M pertenecen a Σ. En otras palabras, todos los átomos de M son necesariamente divisores por la izquierda de Δ.

Definición 3.4.1. El par (M, Δ) es un *monoide de Garside* si M es atómico y para cualesquiera dos átomos s, t de M, el conjunto

$$\{a \in \Sigma \mid s \preceq a,\ t \preceq a\}$$

tiene un elemento mínimo $\Delta_{s,t}$ (con respecto a $\preceq$).

Por el Lema 3.1.2, el elemento mínimo $\Delta_{s,t}$ es único. Se observa que $\Delta_{s,t} = \Delta_{t,s} \in \Sigma,\ s \preceq \Delta_{s,t},\ t \preceq \Delta_{s,t}$ y

$$\{a \in \Sigma \mid s \preceq a,\ t \preceq a\} = \{a \in \Sigma \mid \Delta_{s,t} \preceq a\}.$$

Cualquier átomo $s \in M$ es un elemento mínimo del conjunto $\{a \in \Sigma \mid s \preceq a\}$, así que $\Delta_{s,s} = s$.

Lema 3.4.1. Si (M, Δ) es un monoide de Garside, entonces el conjunto Σ satisface todas las condiciones del Teorema 3.2.1.

Este lema es clave para permitirnos aplicar los resultados vistos en la sección de "Formas Normales y problema del conjugado" a los monoides de Garside. El resto de esta subsección irá enfocado a la prueba del Lema 3.4.1. Necesitamos verificar que Σ satisface las condiciones (1.) a (3.) del Teorema 3.2.1. La condición (1.) la tenemos de forma directa por la definición de monoide pre-Garside. La condición (2.) fue verificada en el Lema 3.3.1. La parte difícil es la verificación de (3.). Comenzamos con dos lemas. En ambos lemas, asumimos que (M, Δ) es un monoide de Garside, Σ es el conjunto de divisores por la izquierda (y derecha) de Δ, y $S \subset \Sigma$ es el conjunto de átomos de M.

Lema 3.4.2. Sea E un subconjunto finito no vacío de M satisfaciendo las siguientes dos condiciones:

(I). Si $a \in M$ y $b \in E$ con $a \preceq b$, entonces $a \in E$.

(II). Si $a \in E$, $s, t \in S$ son tales que $as, at \in E$, entonces $a\Delta_{s,t} \in E$.

Entonces E tiene un elemento máximo (con respecto a $\preceq$).

Demostración. Sea c un elemento de E tal que $\|c\|$ es máximo (decimos que c es de altura máxima en E). Deseamos demostrar que $E = \{a \in M \mid a \preceq c\}$. Por la condición (i), $\{a \in M \mid a \preceq c\} \subset E$. Vamos a probar la inclusión contraria. Supongamos que no está incluido. Entonces existe $b \in E$ tal que $b \not\preceq c$. Expandimos b como producto de átomos $b = s_1 \cdots s_n$ para algún n y $s_1, \ldots, s_n \in S$. Establecemos $a = s_1 \cdots s_k$ donde $k < n$ es el entero máximo tal que $a \preceq c$ (posiblemente $k = 0$, en cuyo caso $a = 1$). Está claro que $a \in E$ y que existe un átomo $s \in S$ (de hecho $s = s_{k+1}$) tal que $as \in E$ y

$as \not\preceq c$. Consideramos entonces que $a \in E$ es de altura máxima verificando la condición previa. Ya que c es de altura máxima en E, tenemos $\|a\| < \|as\| \leq \|c\|$. Esto y la relación $a \preceq c$ implica que existe $t \in S$ tal que $at \preceq c$. Entonces, necesariamente, $t \neq s$. Ahora tenemos $a, as, at \in E$. Por la condición (ii) $a\Delta_{s,t} \in E$. La relación $as \preceq a\Delta_{s,t}$ y $as \not\preceq c$ implica que $a\Delta_{s,t} \not\preceq c$. Podemos expandir

$$\Delta_{s,t} = ts_1s_2 \cdots s_m,$$

donde $s_1, \ldots, s_m \in S$. Existe $i \in \{1, \ldots, m\}$ tal que $ats_1s_2 \cdots s_{i-1} \preceq c$ y $ats_1s_2 \cdots s_i \not\preceq c$. Sea $a' = ats_1s_2 \cdots s_{i-1}$. El hecho de que $a\Delta_{s,t} \in E$ implica que $a', a's_i \in E$. Por la elección de i tenemos que $a's_i \not\preceq c$. Por lo tanto, a' satisface las mismas condiciones de a, pero $\|a'\| \geq \|at\| > \|a\|$. Tenemos una contradicción con la elección de a. □

Lema 3.4.3. Para cualquier $a, b \in \Sigma$, el conjunto

$$E = \{x \in M \mid x \preceq a,\ x \preceq b\} \subset \Sigma$$

tiene un elemento máximo (con respecto a $\preceq$).

Demostración. Ya que Σ es finito, también lo es E. Claramente, $1 \in E$ y por tanto el conjunto E es no vacío. Obviamente se satisface la condición (i) del Lema 3.4.2. Vamos a comprobar la condición (ii). Tenemos que probar que si xs y xt son divisores por la izquierda de a y b para algún $s, t \in S$, entonces $x\Delta_{s,t}$ lo es también. Sea $y \in \Sigma$ tal que $xy = a$. Por hipótesis, $xs \preceq a = xy$ y $xt \preceq a = xy$. Por el Lema 3.3.1 (i), esto implica que $s \preceq y$ y $t \preceq y$. Por la Definición 3.4.1, $\Delta_{s,t} \preceq y$, y por tanto $x\Delta_{s,t} \preceq xy = a$. De igual manera $x\Delta_{s,t} \preceq b$. De esta forma tenemos que $x\Delta_{s,t} \in E$ dandose (ii). Ahora el Lema 3.4.2 implica que E tiene elemento máximo. □

Ahora estamos en condiciones de verificar la condición (3.) del Teorema 3.2.1. Tomamos cualesquiera $a, b \in \Sigma$. Ya que $a \preceq \Delta$, tenemos que $\Delta = aa'$ para algún $a' \in \Sigma$. Por el Lema 3.4.3, el conjunto

$$\{x \in M \mid x \preceq a',\ x \preceq b\} \subset \Sigma$$

tiene un elemento máximo c con respecto a $\preceq$. Afirmamos que c es máximo en

$$\{x \in \Sigma \mid x \preceq b,\ ax \in \Sigma\}.$$

En efecto, por definición, $c \preceq a'$, por lo cual $ac \preceq aa' = \Delta$ y $ac \in \Sigma$. Sea $d \in \Sigma$ tal que $d \preceq b$ y $ad \in \Sigma$. Entonces $ad \preceq \Delta = aa'$ y en consecuencia, por el Lema 3.3.1 (i) (cancelación por la izquierda en Σ) implica que $d \preceq a'$. Por lo tanto, $d \preceq c$.

3.4.2. Monoides de Garside exhaustivos

Un monoide de Garside (M, Δ) es *exhaustivo* si el conjunto $\Sigma \subset M$ de divisores de Δ es exhaustivo en el sentido en el que fue definido anteriormente, es decir, que $1 \in \Sigma$ y que M posee una presentación cuyos generadores y relaciones pertenecen a Σ. Los resultados previos nos dan como consecuencia las siguientes propiedades que cumple un monoide de Garside exhaustivo (M, Δ).

(1) Tenemos que $M \cong M_\Sigma$ (Lema 3.2.3). En otras palabras, M tiene una presentación con generadores $[a]$, donde a se mueve en Σ, y relaciones $[1] = 1$ y $[a][b] = [ab]$, donde $a, b \in \Sigma$ satisface $ab \in \Sigma$.

(2) Para cualquier $a \in M$, existe un único divisor por la izquierda $\alpha(a) \in \Sigma$ de a que es máximo entre todos los divisores por la izquierda de a pertenecientes a Σ (Teorema 3.2.1).

(3) Cualquier $a \in M$ se expande de forma única como un producto $a = a_1 a_2 \cdots a_r$ de ciertos $a_1, a_2, \ldots, a_r \in \Sigma \backslash \{1\}$ con $r \geq 0$ tal que $a_i = \alpha(a_i a_{i+1} \cdots a_r)$ para todo $i \in \{1, \ldots, r\}$ (Sección 3.2.2).

(4) El homomorfismo de monoides natural de M en su grupo de fracciones G_M es inyectivo (Lema 3.2.1 y Colorario 3.3.1). En particular, M es cancelativo por la izquierda y por la derecha. Podemos identificar M como una copia suya dentro de G_M.

(5) Si $M \neq \{1\}$, entonces cualquier elemento de G_M puede ser escrito de la forma $\Delta^s b$ con unicidad (forma normal greedy), donde $s \in \mathbb{Z}$ y $b \in M$ no es múltiplo por la derecha de Δ (Teorema 3.3.2).

(6) El problema del conjugado G_M es equivalente al problema del conjugado en M (Sección 3.3.4). Este último es soluble si M admite una presentación ponderada (Sección 3.2.5).

3.4.3. Divisores y múltiplos comunes en monoides de Garside

Dados $k \geq 2$ elementos $a_1, \ldots, a_k$ de un monoide M, decimos que $d \in M$ es un *máximo común divisor por la izquierda* (mcd) de $a_1, \ldots, a_k$ si $d \preceq a_i$ para todo $i \in \{1, \ldots, k\}$, y $d' \preceq d$ para cualquier $d' \in M$ tal que $d' \preceq a_i$ ara todo $i \in \{1, \ldots, k\}$. Reemplazando $\preceq$ por $\succeq$, obtenemos análogamente la noción de mcd por la derecha.

Decimos que $m \in M$ es un *mínimo común múltiplo* (mcm) de $a_1, \ldots, a_k$ si $a_i \preceq m$ para todo $i \in \{1, \ldots, k\}$, y $m \preceq m'$ para cualquier $m' \in M$ tal que $a_i \preceq m'$ ara

todo $i \in \{1, \ldots, k\}$. Existe la noción análoga de mcm por la izquierda. Si M es atómico, entonces el mcd y el mcm son únicos siempre que existan. La propiedad (2) de la sección anterior puede ser reformulada diciendo que Δ y cualquier $a \in M$ tienen mcd por la izquierda. Esas propiedades de los monoides de Garside pueden ser generalizadas de la siguiente manera.

Teorema 3.4.1. Sea (M, Δ) un monoide de Garside exhaustivo. Entonces cualquier familia finita de elementos de M tiene un único mcd por la izquierda y un único mcm por la derecha.

Demostración. Sean $b, c \in M$. Consideramos el conjunto

$$E = \{a \in M \mid a \preceq b,\ a \preceq c\}.$$

Con el objetivo de probar que b y c tienen un mcd por la izquierda en M, es suficiente con comprobar que E satisface las condiciones del Lema 3.4.2. El conjunto E claramente satisface la condición (i). El conjunto E es finito porque $\|a\| < \|b\|$ para cualquier $a \in E$, así que a es el producto de como mucho $\|b\|$ átomos de M, y el conjunto de átomos de M, siendo un subconjunto de Σ, es finito.

Vamos a comprobar la condición (ii). Supongamos que tenemos $a \in E$ y los átomos $s, t \in M$ tal que $as, at \in E$. Escribimos $b = ab_1$ con $b_1 \in M$. Ya que M es cancelativo por la izquierda, $as \preceq b = ab_1$ implica $s \preceq b_1$ y $at \preceq b = ab_1$ implica $t \preceq b_1$. Consideramos el divisor por la izquierda máximo $\alpha(b_1)$ de b_1 en Σ. Tenemos que $s \preceq \alpha(b_1)$ y $t \preceq \alpha(b_1)$. Por tanto $\alpha_{s,t} \preceq \alpha(b_1) \preceq b_1$. Por eso $\Delta_{s,t} \preceq b_1$ y $a\Delta_{s,t} \preceq ab_1 = b$. De igual forma $a\Delta_{s,t} \preceq c$. Esto prueba que $a\Delta_{s,t} \in E$.

Que cualquier familia finita de elementos de M tiene un mcd por la izquierda se sigue fácilmente por inducción sobre la cardinalidad de la familia.

Vamos a probar la existencia de mcm por la derecha. Sea $a_1, \ldots, a_k \in M$. En vista al Lema 3.3.1 (iv), existe $r \geq 1$ tal que $a_i \preceq \Delta^r$ para todo $i \in \{1, \ldots, k\}$. Consideramos el conjunto

$$X = \{x \in M \ \mid \ a_i \preceq c \preceq \Delta^r \ \ \forall i \in \{1, \ldots, k\}\}.$$

Ya que el conjunto de átomos de M es finito y todos los divisores por la izquierda de Δ^r se expanden como producto de a lo máximo $r\|\Delta\|$ átomos, el conjunto de divisores por la izquierda de Δ^r es finito. Dado que contiene a X, este es finito también. Sea m el máximo común divisor por la izquierda de los elementos de X. Afirmamos que m es mínimo común múltiplo por la derecha de $a_1, \ldots, a_k$. En efecto, $a_1, \ldots, a_k$ son divisores por la izquierda de todos los elementos de X. Por lo tanto, son divisores por la izquierda de m. Esto demuestra que m es múltiplo común por la derecha de $a_1, \ldots, a_k$.

Sea m' otro común múltiplo por la derecha de $a_1, \ldots, a_k$. Denotamos por m'' un máximo común divisor por la izquierda de m' y Δ^r. Comprobamos que $m'' \in X$. Primero,

$m'' \preceq \Delta^r$. Ya que a_i es un divisor por la izquierda de m' y de Δ^r, es un divisor por la izquierda de m''. Esto prueba que $m'' \in X$. Por definición de m, tenemos que $m \preceq m''$. Dado que $m'' \preceq m'$, obtenemos $m \preceq m'$. Esto prueba nuestra afirmación. □

El monoide de Garside exhaustivo ha conseguido reunir las características obtenidas, por una parte, del monoide M_Σ, y por otra, de los monoides pre-Garside. De esta forma los monoides de Garside exhaustivos consiguen lo que llevamos buscando desde un inicio, una forma normal en el grupo de fracciones, que a su vez resuelve el problema de palabra y una solución al problema del conjugado en el grupo de fracciones entre otras propiedades. Más adelante construiremos un monoide de trenzas, el cual veremos que es un monoide de Garside exhaustivo y cuyo grupo de fracciones será el grupo de trenzas.

Capítulo 4

Construcción del grupo de trenzas

En primer lugar vamos a introducir un poco quién fue Emil Artin y así poder contextualizar mejor el origen del grupo de trenzas.

Emil Artin fue un matemático austriaco con orígenes armenios, nacido en Viena en el año 1898. Inició su carrera en Alemania, en la Universidad de Gotinga, y en 1923 se traslado a la Universidad de Hamburgo. La amenaza de la inminente Segunda Guerra Mundial lo obligó a emigrar a Estados Unidos en 1937 donde estuvo en la Universidad de Indiana (1938-1946) y en la Universidad de Princeton (1946-1958). Fue uno de los mejores y más influyentes algebristas del del siglo XX, llegando a solucionar el problema 17 de la lista de problemas de Hilbert. Trabajó en la teoría de números, contribuyó a la teoría algebraica de anillos asociativos y los números hipercomplejos. Finalmente falleció en Hamburgo, Alemania, en el año 1962.

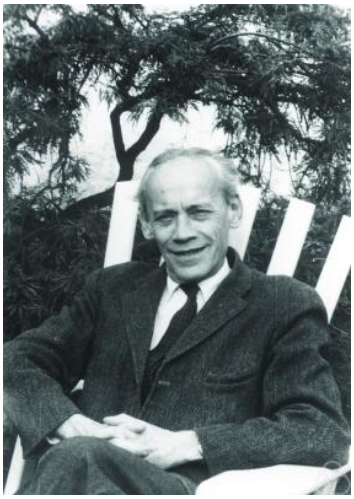


Figura 4.1: Fotografía de Emil Artin [19]

Los términos *trenza* y *grupo de trenzas* fueron acuñados por primera vez por Emil Artin en [1], publicación del año 1925. En esta, las trenzas aparecen como objetos topológicos. No fue hasta el año 1947 en el que Artin dio una presentación explícita de este grupo en la revista *Annals of mathematics* [2]. A continuación vamos a definir los grupos de trenzas tanto desde un punto de vista algebraico, como topológico, las cuales se verán que son equivalentes. También estudiaremos algunas de sus propiedades como grupo. La construcción de este grupo ha sido tomada de [11] y [12].

4.1. El grupo de trenzas de Artin

Ahora vamos a dar la definición algebraica del grupo de trenzas B_n para cualquier $n > 0$. La definición está formulada en términos de presentación de un grupo por generadores y relaciones.

Definición 4.1.1. El grupo de trenzas de Artin B_n es el grupo generado por $n-1$ elementos $\sigma_1, \sigma_2, ..., \sigma_{n-1}$ y las relaciones de trenza

$$\sigma_i \sigma_j = \sigma_j \sigma_i,$$

para todo $i, j \in \{1, 2, ..., n-1\}$ con $|i-j| \geq 2$ y

$$\sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1},$$

para todo $i \in \{1, 2, ..., n-2\}$.

El siguiente lema nos relaciona el grupo de trenzas con otros grupos en los que se cumplan las condiciones de la definición previa para ciertos elementos.

Lema 4.1.1. Si $s_1, ..., s_{n-1}$ son elementos de un grupo G que satisfacen las relaciones de trenza, entonces existe un único homomorfismo de grupos $f : B_n \to G$ tal que $s_i = f(\sigma_i)$ para todo $i \in \{1, 2, ..., n-1\}$.

Demostración. Sea F_n el grupo libre generado por el conjunto $\{\sigma_1, ..., \sigma_{n-1}\}$. Hay un único homomorfismo $g : F_n \to G$ tal que $f(\sigma_i) = s_i$ para todo $i \in \{1, ..., n-1\}$. Por la demostración del Primer Teorema de Isomorfía obtenemos otro homomorfismo $\bar{g} : F_n/K \to G$ donde $K = \ker(g)$ (de hecho, es un monomorfismo). También tenemos un homomorfismo $h : F_n \to B_n$ con $h(\sigma_i) = \sigma_i$ del cual consideramos su núcleo, $H_n = \ker(h)$. Llamamos $k : F_n/H_n \to B_n$ al isomorfismo en cuestión. Si tomamos el subgrupo normal N de B_n al que nos referimos en la Definición 2.5.2, es fácil ver haciendo uso del Teorema de Factorización que $H_n = N$. El que los elementos de s_i de G satisfagan las relaciones de trenza implica que $R \subset R'$, donde R es el conjunto de relaciones de

B_n y R' es el conjunto de relaciones de G y que por tanto $H_n \subset K$. Por tanto el homomorfismo $\phi : F_n/H_n \to F_n/K$ con $\phi(xH_n) = \phi(xK)$ está bien definido. De esta forma, componiendo $f = k^{-1} \circ \phi \circ \bar{g}$ obtenemos nuestro homomorfismo deseado. □

4.2. Trenzas y sus diagramas

Procedemos a realizar la construcción topológica del grupo de trenzas. Para ello comenzamos definiendo lo que es una trenza geométrica, viendo cómo podemos representarla y estableciendo equivalencias entre distintas trenzas. De aquí en adelante consideraremos el intervalo I como el intervalo $[0, 1]$.

Definición 4.2.1 (Trenza geométrica)**.** Una trenza geométrica de n hebras, con $n \geq 1$, es un subconjunto $\mathcal{B} \subset \mathbb{R}^2 \times I$ formado por n intervalos topológicos (subconjuntos de $\mathbb{R}^2 \times I$ homeomorfos al intervalo $[0, 1]$) disjuntos llamados hebras de tal manera que la proyección $\mathbb{R}^2 \times I \to I$ establezca un homeomorfismo de cada hebra en I y

$$\mathcal{B} \cap (\mathbb{R}^2 \times \{0\}) = \{(1, 0, 0), (2, 0, 0), ..., (n, 0, 0)\},$$

$$\mathcal{B} \cap (\mathbb{R}^2 \times \{1\}) = \{(1, 0, 1), (2, 0, 1), ..., (n, 0, 1)\}.$$

Cada hebra de $\mathcal{B}$ interseca con el plano $\mathbb{R}^2 \times \{t\}$ con $t \in I$ en un único punto y conecta un punto $(i, 0, 0)$ con un punto $(s(i), 0, 1)$ donde $i, s(i) \in \{1, 2, ..., n\}$. La sucesión $(s(1), s(2), ..., s(n))$ es una permutación del conjunto $\{1, 2, ..., n\}$ llamada permutación subyacente de $\mathcal{B}$.

Decimos que dos trenzas geométricas $\mathcal{B}$ y $\mathcal{B}'$ son isotópicas si podemos deformar de manera continua $\mathcal{B}$ en $\mathcal{B}'$. Más formalmente, $\mathcal{B}$ y $\mathcal{B}'$ son isotópicas si existe una función continua $F : \mathcal{B} \times I \to \mathbb{R}^2 \times I$ de tal manera qué para cada $s \in I$, la función $F_s : \mathcal{B} \to \mathbb{R}^2 \times I$ con $F_s(x) = F(x, s)$ es un embebimiento cuya imagen es una trenza geométrica de n hebras, cumpliendo que $F_0 = id_{\mathcal{B}} : \mathcal{B} \to \mathcal{B}$ y $F_1(\mathcal{B}) = \mathcal{B}'$. Debido a que la imagen de F_s es una trenza geométrica y F es una función continua tenemos automáticamente que la imagen del punto final de cada hebra por F_s es el mismo.

Tanto la función F como la familia de trenzas geométricas $\{F_s(\mathcal{B})\}_{s \in I}$ se llaman *isotopía* de $\mathcal{B}$ en $\mathcal{B}'$. Se ve fácilmente que la relación de isotopía es una relación de equivalencia. Al conjunto de clases de equivalencia correspondientes se les denomina *trenzas de n hebras* y lo denotamos por $\mathcal{B}_n$.

Para poder especificar una trenza geométrica se puede utilizar la proyección en $\mathbb{R} \times \{0\} \times I$ e indicar de alguna manera qué hebra está por encima de otra cuando se crucen. Solo se podrá aplicar esta solución en trenzas geométricas cuya proyección solo tenga puntos donde se crucen a lo más dos hebras. Pasamos a definir formalmente un diagrama de trenza.

Definición 4.2.2 (Diagrama de trenza)**.** Un diagrama de trenza de n hebras es un conjunto $\mathcal{D} \subset \mathbb{R} \times I$ formado por la unión de n intervalos topológicos llamados hebras y que cumplen las siguientes condiciones:

1. La proyección $\mathbb{R} \times I \to I$ de cada hebra en I es un homeomorfismo.
2. Cada punto de $\{1, 2, ..., n\} \times \{0, 1\}$ es un punto final de la hebra.
3. Cada punto de $\mathbb{R} \times I$ pertenece como mucho a dos hebras. En cada punto donde se intersequen dos hebras, estas se cruzaran transversalmente.

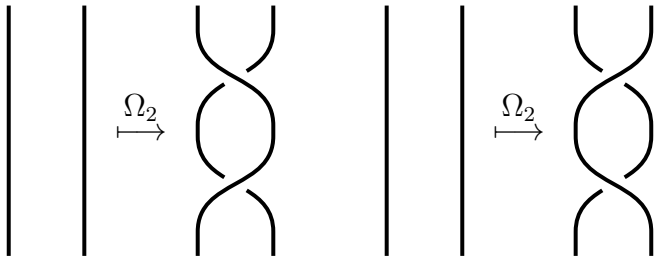
Cada trenza puede ser representada por un diagrama de trenza y cada diagrama tiene asociada una trenza. Dado $\mathcal{D}$ un diagrama, denotaremos como $\beta(\mathcal{D})$ a la trenza de n hebras asociada. La transversalidad de los cruces y la compacidad de las hebras nos asegura que el número de cruces es finito. En un cruce, entendemos que una hebra pasa por encima de otra cuando la preimagen del punto de intersección de la hebra en la trenza geométrica tiene segunda coordenada mayor que la de la preimagen de la otra hebra, que pasa por debajo. Gráficamente la hebra que pasa por debajo en un cruce se representará con una discontinuidad y la que pasa por encima con una línea continua.

Dos diagramas de trenzas $\mathcal{D}, \mathcal{D}'$ de n hebras se dicen que son isotópicos si se puede establecer una función continua entre $F : \mathcal{D} \times I \to \mathbb{R} \times I$ tal que para cada $s \in I$ el conjunto $\mathcal{D}_s = F(\mathcal{D} \times s) \subset \mathbb{R} \times I$ es un diagrama de n hebras, con $\mathcal{D}_0 = \mathcal{D}$ y $\mathcal{D}_1 = \mathcal{D}'$. Se entiende que F lleva los cruces de $\mathcal{D}$ a los cruces de $\mathcal{D}_s$ para todo $s \in I$ preservando si los cruces son por debajo o por encima. A la familia de diagramas $\{\mathcal{D}_s\}_{s \in I}$ se le llama *isotopía* de $\mathcal{D}_0 = \mathcal{D}$ en $\mathcal{D}_1 = \mathcal{D}'$.

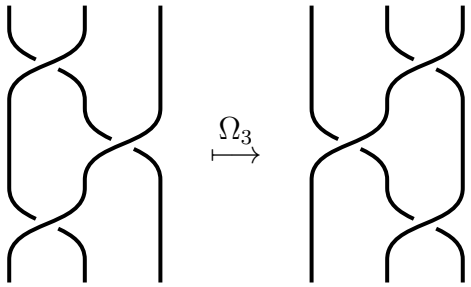
Figura 4.2: Diagramas isótopicos

Definición 4.2.3 (Movimientos de Reidemeister)**.** Los movimientos de Reidemester son transformaciones locales (afectan a la posición de las hebras dentro de un disco en $\mathbb{R} \times I$) en un diagrama de trenzas. Estos están formados por Ω_2, Ω_3 (Figura 4.3) y sus transformaciones inversas Ω_2^{-1}, Ω_3^{-1}.

Lo que estamos intentando hacer es establecer una relación de equivalencia en los diagramas de trenzas semejante a la que se establece con las isotopías en las trenzas



(a)



(b)

Figura 4.3: Movimientos de Reidemeister

geométricas. Para ello se intenta definir el concepto análogo de isotopía en los diagramas de trenzas. Pero hay isotopías en las trenzas geométricas que no tienen su equivalente directo como isotopía de diagramas de trenzas, puesto que en puntos concretos de la deformación del diagrama este deja de cumplir con la definición de diagrama. Con este motivo se definen los movimientos de Reidemeister, para saltar estos puntos problemáticos y así obtener la relación de equivalencia buscada.

Estos movimientos preservan la correspondencia entre diagrama de trenza y trenza. Decimos que dos diagramas de trenzas $\mathcal{D}, \mathcal{D}'$ son $\mathcal{R}$-*equivalentes* si $\mathcal{D}$ puede ser transformado en $\mathcal{D}'$ mediante una sucesión finita de isotopías y movimientos de Reidemeister. Está claro que si $\mathcal{D}, \mathcal{D}'$ son $\mathcal{R}$-*equivalentes*, entonces $\beta(\mathcal{D}) = \beta(\mathcal{D}')$.

Hemos construido dos equivalencias, una para trenzas geométricas y otra para diagramas de trenzas. El Teorema 4.2.1 nos permite trabajar indiferentemente con unas o con otras utilizando ambas equivalencias. También nos garantiza que los diagramas $\mathcal{R}$-*equivalentes* presentan la misma trenza. La prueba se puede ver en [11].

Teorema 4.2.1. Dos diagramas de trenzas representan trenzas geométricas isotópicas si y solo si son $\mathcal{R}$-*equivalentes*.

4.3. Estructura de grupo

Una vez que hemos definido las trenzas y sabiendo cómo podemos representarlas vamos a ver que estas cumplen las propiedades para considerar al conjunto de trenzas como un grupo. Para ello comenzamos definiendo la operación producto.

Definición 4.3.1 (Producto de trenzas)**.** Dado dos diagrama de trenzas $\mathcal{D}_1$, $\mathcal{D}_2$, representando las trenzas de n hebras β_1 y β_2 respectivamente, definimos el producto de diagramas $\mathcal{D}_1\mathcal{D}_2$ concatenando $\mathcal{D}_1$ en la parte superior de $\mathcal{D}_2$ y reduciendo el diagrama resultante en $\mathbb{R} \times I$. Entonces el producto de trenzas $\beta_1\beta_2$ es la trenza asociada al diagrama producto $\mathcal{D}_1\mathcal{D}_2$.

Ejemplo 4.3.1. Sean $\beta_1, \beta_2 \in \mathcal{B}_3$, cuyos diagramas se pueden ver en la Figura 4.4 (a), el resultado del producto $\beta_1\beta_2$ se representa en la Figura 4.4 (b).

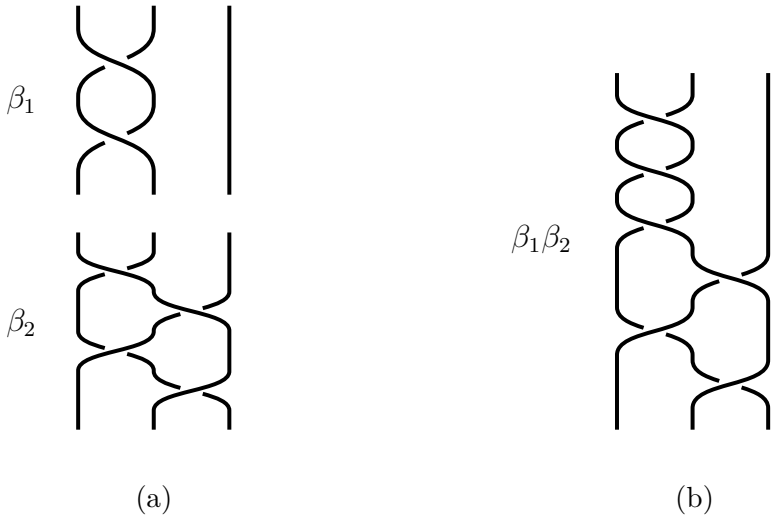


Figura 4.4: Producto de dos hebras

Nota: Formalmente la longitud de los diagramas debería ser constante, pero para una mejor visualización de estos no se ha tenido en cuenta.

De forma clara se ve que el elemento neutro para el producto de trenzas, por ambos lados, es la trenza con el diagrama cuyas hebras se encuentran paralelas entre sí (Figura 4.5).

Trivialmente también se tiene la asociatividad del producto de trenzas como se ilustra en el Ejemplo 4.3.2.

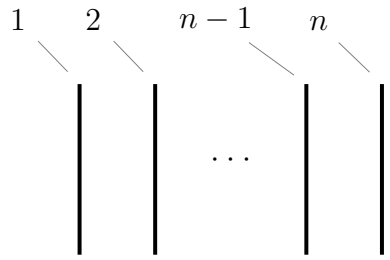


Figura 4.5: Elemento neutro

Por último probamos el siguiente lema el cual implica que $\mathcal{B}_n$ es un grupo.

Definición 4.3.2 (Trenzas elementales)**.** Una trenza elemental es una trenza de $\mathcal{B}_n$ en la que solo hay un cruce de hebras. Denotaremos por σ_i^+ la trenza que cruza la hebra $i+1$-ésima sobre la i-ésima trenza y por σ_i^- a la trenza que cruza la i-ésima trenza sobre la $i+1$-ésima con $i \in \{1, \ldots, n-1\}$.

Los diagramas de las trenzas elementales quedan ilustrados en la Figura 4.7.

Lema 4.3.1. El conjunto de trenzas elementales genera $\mathcal{B}_n$ como monoide.

Demostración. Tenemos que probar que las trenzas $\sigma_1^+, ..., \sigma_{n-1}^+, \sigma_1^-, ..., \sigma_{n-1}^- \in \mathcal{B}_n$ generan $\mathcal{B}_n$ como un monoide. Para ver esto, consideramos una trenza β de n hebras representada por un diagrama de trenza $\mathcal{D}$. A través de una ligera deformación de $\mathcal{D} \subset \mathbb{R} \times I$ en un vecindario de sus puntos de cruce, podemos hacer que los cruces de $\mathcal{D}$ tengan distinta su segunda coordenada. Por tanto podemos establecer una partición

$$0 = t_0, t_1, ..., t_{k_-1}, t_k = 1$$

tal que la intersección de $\mathcal{D}$ con cada banda $\mathbb{R} \times [t_j, t_{j-1}]$ tiene exactamente un cruce dentro. Esta intersección es en realidad un diagrama de σ_i^+ o σ_i^- para algún $i \in \{1, 2, ..., n-1\}$. El resultado de la división de $\mathcal{D}$ como el producto de k diagrama de hebras muestra que

$$\beta = \beta(\mathcal{D}) = \sigma_{i_1}^{\varepsilon_1} \sigma_{i_2}^{\varepsilon_2} \cdots \sigma_{i_k}^{\varepsilon_k},$$

donde cada ε_j es o $+$ o $-$ y $i_1, ..., i_k \in 1, 2, ..., n-1$.

□

Colorario 4.3.1. Sea $\beta \in \mathcal{B}_n$, existe un elemento $\beta^{-1} \in \mathcal{B}_n$ que es el inverso por ambos lados de β.

Ejemplo 4.3.2 (Asociatibidad)**.**

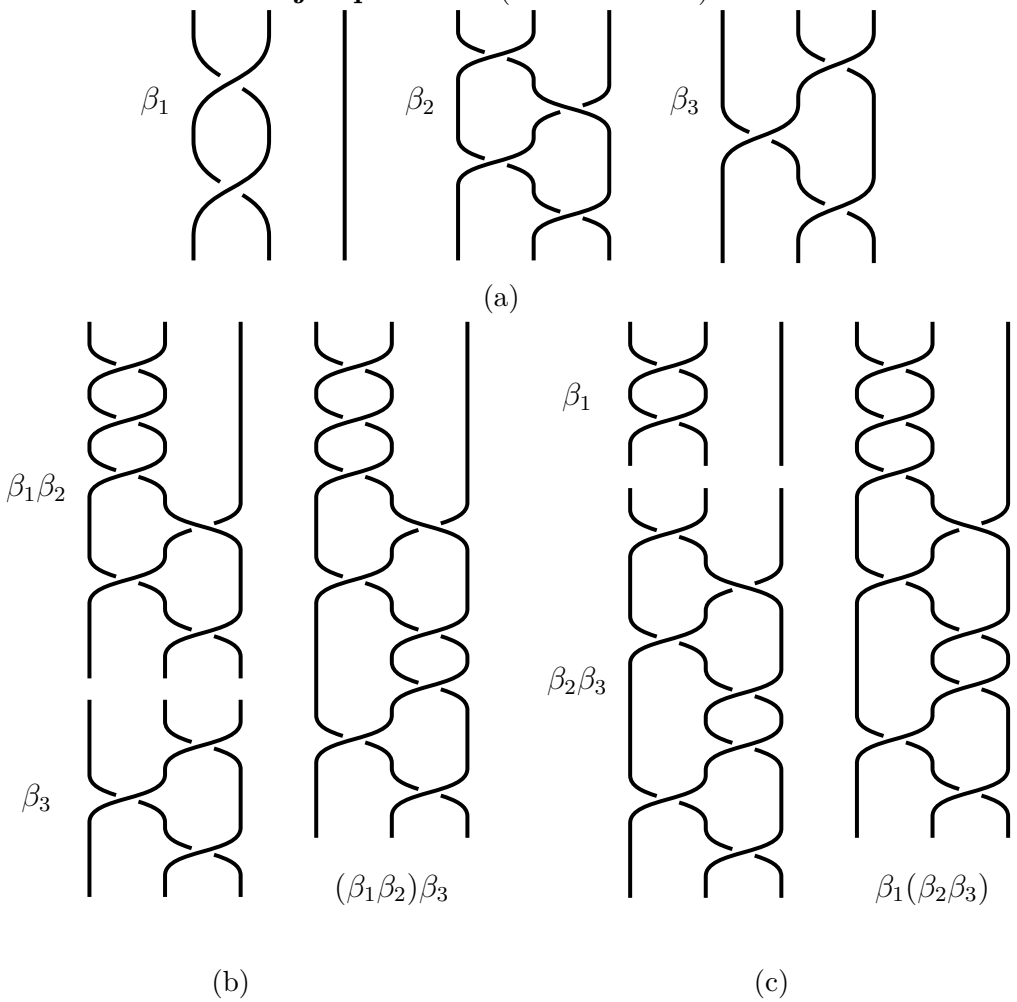


Figura 4.6: Asociatividad

Demostración. Claramente, $\sigma_i^+\sigma_i^- = \sigma_i^-\sigma_i^+ = 1$ para todo $i \in \{1, \ldots, n-1\}$. Por tanto, escribiendo β como producto de trenzas elementales $\beta = \sigma_{i_1}^{\varepsilon_1}\sigma_{i_2}^{\varepsilon_2}\cdots\sigma_{i_k}^{\varepsilon_k}$, se tiene que el inverso de β es $\beta^{-1} = \sigma_{i_k}^{-\varepsilon_k}\cdots\sigma_{i_2}^{-\varepsilon_2}\sigma_{i_k}^{-\varepsilon_k}$ (utilizamos el convenio "$--=+$, $+-=-$, $-+=-$"). □

Lema 4.3.2. Los elementos $\sigma_1^+, ..., \sigma_{n-1}^+ \in \mathcal{B}_n$ satisfacen las relaciones de trenza, las cuales son, $\sigma_i\sigma_j = \sigma_j\sigma_i$ para todo $i, j \in \{1, 2, ..., n-1\}$ con $|i-j| \geq 2$, y $\sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1}$ para todo $i \in \{1, 2, ..., n-2\}$.

Demostración. La primera relación viene del hecho de que ambas partes representan

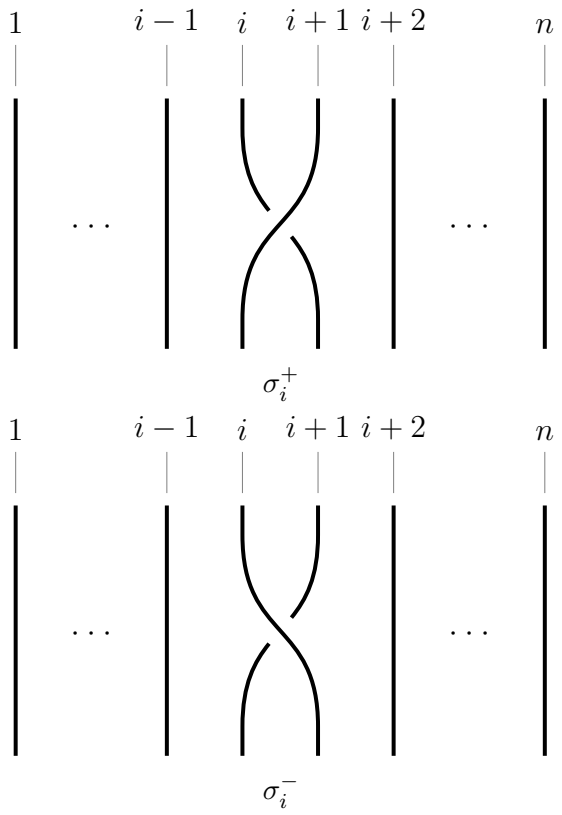


Figura 4.7: Trenzas elementales σ_i^+ y σ_i^-

diagramas isotópicos. Los diagramas que representan ambos lados de la segunda relación se diferencian por un movimiento de Reidmeister Ω_3. □

Teorema 4.3.1. Para $\varepsilon \in \{+, -\}$, existe un único homomorfismo $\varphi_\varepsilon : B_n \to \mathcal{B}_n$ tal que $\varphi_\varepsilon(\sigma_i) = \sigma_i^\varepsilon$ para cada $i \in \{1, 2, ..., n-1\}$. El homomorfismo φ_ε es un isomorfismo.

Demostración. Lo probamos para $\varepsilon = +$, el caso $\varepsilon = -$ es similar. La existencia y unicidad de φ_+ vienen proporcionadas por los Lemas 4.1.1 y 4.3.2. En la prueba del Lema 4.3.1 se demuestra que $\sigma_1^+, \sigma_2^+, ..., \sigma_n^+$ son generadores de $\mathcal{B}_n$. Estos generadores pertenecen a la imagen de φ_+. Por tanto σ_+ es sobreyectiva.

Ahora tomamos la función $\psi : \mathcal{B}_n \to B_n$ tal que $\psi \circ \varphi = id$. Esto implica que φ_+ es inyectiva. Como en la prueba del Lema 4.3.1, representamos cualquier $\beta \in \mathcal{B}_n$ por el diagrama $\mathcal{D}$ en el cual los cruces tienen la segunda coordenada diferente. Esto lleva a una expansión de la forma

$$\psi(\mathcal{D}) = (\sigma_{i_1})^{\varepsilon_1}(\sigma_{i_2})^{\varepsilon_2} \cdots (\sigma_{i_k})^{\varepsilon_k} \in B_n, \tag{4.1}$$

donde $(\sigma_i)^+ = \sigma_i$ y $(\sigma_i)^- = \sigma_i^{-1}$. Se tiene que $\psi(\mathcal{D})$ depende exclusivamente de β. Por el Teorema 4.2.1 solo nos hace falta verificar que $\psi(\mathcal{D})$ es invariable bajo isotopías y movimientos de Reidmeister sobre $\mathcal{D}$. Isotopías de $\mathcal{D}$ que conservan el orden de los puntos dobles de $\mathcal{D}$ con respecto a la segunda coordenada mantienen la expansión (4.1) y por tanto preserva $\psi(\mathcal{D})$. Una isotopía que cambia el orden de dos dobles puntos en $\mathcal{D}$ (como en la Figura 4.2) reemplaza el termino $\sigma_i^{\varepsilon_i}\sigma_j^{\varepsilon_j}$ en (4.1) por $\sigma_j^{\varepsilon_j}\sigma_i^{\varepsilon_i}$ para algún $i, j \in \{1, 2, ..., n-1\}$ con $|i-j| \geq 2$. Bajo ψ, a estas expresiones les corresponde el mismo elemento de B_n debido a la primera relación de trenza en Definición 4.1.1.

El movimiento Ω_2 (respectivamente ω_2^{-1}) en $\mathcal{D}$ inserta (respectivamente elimina) en la expansión (4.1) un término $\sigma_i^+\sigma_i^-$ o $\sigma_i^-\sigma_i^+$. Claramente esto preserva $\psi(\mathcal{D})$.

El movimiento Ω_3 en $\mathcal{D}$ reemplaza la secuencia $\sigma_i\sigma_{i+1}\sigma_i$ en (4.1) por $\sigma_{i+1}\sigma_i\sigma_{i+1}$. Bajo ψ, esta expresión corresponde al mismo elemento de B_n debido a la segunda relación de trenza en Definición 4.1.1. El movimiento ω_3^{-1} se ve de forma similar.

Esto demuestra que ψ es una función bien definida de $\mathcal{B}_n \to B_n$. Por construcción $\psi \circ \varphi_+ = id$. Por tanto φ_+ es sobreyectiva e inyectiva. □

Llegados a este punto ya hemos probado que las trenzas geométricas forman un grupo y que este coincide con el grupo de trenzas de Artin como habíamos dicho al principio. A continuación vamos a ver cómo podemos relacionar el grupo de trenzas con el grupo de permutaciones, para poder así definir un tipo de trenzas denominado trenzas puras.

Sea $\beta \in B_n$ una trenza de n hebras, consideramos una trenza geométrica que la represente. Denotamos por d_i a la hebra que une el punto $(i, 0, 0)$ con el punto $(j(i), 0, 1)$ con $i \in \{1, \ldots, n\}$, siendo j una permutación que nos lleva la primera coordenada del extremo superior de una trenza a la primera coordenada del extremo inferior de la misma. Definimos la función $\pi : B_n \to S_n$ del grupo de trenzas al grupo simétrico como

$$\pi(\beta) = \begin{pmatrix} 1 & 2 & \cdots & n \\ j(1) & j(2) & \cdots & j(n) \end{pmatrix}.$$

Para construir $\pi(\beta)$ a partir de β escribimos este como producto de trenzas elementales, $\beta = \sigma_{i_1}^{\varepsilon_1}\sigma_{i_2}^{\varepsilon_2} \cdots \sigma_{i_k}^{\varepsilon_k}$, como se hizo en la demostración del Lema 4.3.1. Tenemos $\pi(\sigma_{i_h}^{\varepsilon_h}) = s_{i_h}$ para $h \in \{1, \ldots, k\}$ (s_{i_h} transposición definida para la Proposición 2.2.2), como se muestra en la Figura 4.7 y también es fácil de ver que, dados $\alpha, \gamma \in B_n$, entonces $\pi(\alpha\gamma) = \pi(\alpha)\pi(\gamma)$, donde la composición se realiza de izquierda a derecha. Por tanto se tiene que $\pi(\beta) = \pi(\sigma_{i_1}^{\varepsilon_1}\sigma_{i_2}^{\varepsilon_2} \cdots \sigma_{i_k}^{\varepsilon_k}) = \pi(\sigma_{i_1}^{\varepsilon_1})\pi(\sigma_{i_2}^{\varepsilon_2}) \cdots \pi(\sigma_{i_k}^{\varepsilon_k}) = s_{i_1}s_{i_2} \cdots s_{i_k}$. Da igual la trenza geométrica tomada como representante o que descomposición como producto de trenzas simples hemos escogido puesto que todas tienen el mismo punto de inicio y fin para todas las hebras. Podemos decir que π es un invariante para las trenzas geométricas y un homomorfismo de grupos. Como consecuencia se tiene que el

conjunto de generadores $s_1, \ldots, s_n$ de S_n cumplen las relaciones que definen al grupo de trenzas. Normalmente, a la permutación obtenida a partir de una trenza se le denomina *permutación de hebra* y se denota por $\pi(\beta)$.

Por último definimos un tipo de trenzas a las que se hará mención más adelante en la sección de criptoanálisis.

Definición 4.3.3 (Trenzas puras)**.** El núcleo del homomorfismo $\pi : B_n \to S_n$ es llamado *grupo de trenzas puro* y es denotado por P_n

$$P_n = Ker(\pi).$$

Los elementos de P_n se llaman trenzas puras de n hebras. Una trenza geométrica representa a un elemento de P_n si y solo si la hebra que parte de $(i, 0, 0)$ termina en $(i, 0, 1)$, con $i \in \{1, \ldots, n\}$.

En esta sección hemos visto una definición algebraica del grupo de trenzas, dando una presentación de grupo de esta. También hemos dado una definición topológica de las trenzas y una forma de representarlas a través de diagramas. Junto con la operación concatenación hemos demostrado que las trenzas tienen una estructura de grupo y que es equivalente al grupo de trenzas definido de forma algebraica. En el siguiente capítulo nos centraremos en cómo podemos trabajar con el grupo de trenzas.

Capítulo 5

Monoide de trenzas

Ahora aplicaremos la potencia que nos dan los monoides de Garside a los grupos de trenzas que acabamos de construir en la sección previa. Construyendo el monoide de trenzas vamos a disponer de una forma normal en el grupo de trenzas que nos permitirá resolver el problema de palabra y expresar de forma única cada una de las trenzas. Esto es fundamental a la hora de trabajar en protocolos criptográficos que hacen uso de los grupos de trenzas, de los cuales veremos algunos en el siguiente capítulo. También veremos que disponemos de una solución para el problema del conjugado entre otras propiedades.

5.1. Una presentación por generadores y relaciones

Para cualquier $n \geq 1$, denotamos por B_n^+ el monoide que tiene como presentación la formada por $n-1$ generadores $\sigma_1, \sigma_2, \ldots, \sigma_{n-1}$ y las relaciones

$$\sigma_i \sigma_j = \sigma_j \sigma_i \quad si\ |i-j| \geq 2,$$

$$\sigma_i \sigma_j \sigma_i = \sigma_j \sigma_i \sigma_i \quad si\ |i-j| = 1,$$

donde $i, j \in \{1, 2, \ldots, n-1\}$. El monoide B_n^+ se llama *monoide de trenzas* de n hebras. A los elementos de B_n^+ se les llama *trenzas positivas* de n hebras. Por definición, B_1^+ es el monoide trivial. El monoide B_2^+ está generado por un único generador y un conjunto vacío de relaciones. Es isomorfo al monoide $\mathbb{N}$ de los números naturales.

La presentación de B_n^+ dada previamente es finita y equilibrada en longitud en el sentido ya visto, lo que nos proporciona una solución para el problema de palabra para B_n^+. Además, el Lema 3.1.3 implica que el monoide B_n^+ es atómico cuyos átomos son $\sigma_1, \sigma_2, \ldots, \sigma_{n-1}$ y $\|a\| = \ell(a)$ para todo $a \in B_n^+$, donde $\ell : B_n^+ \to \mathbb{N}$ es el homomorfismo de monoides definido por $\ell(\sigma_i) = 1$ para $i \in \{1, \ldots, n-1\}$.

Establecemos

$$\Delta_n = (\sigma_1 \cdots \sigma_{n-2}\sigma_{n-1})(\sigma_1 \cdots \sigma_{n-2}) \cdots (\sigma_1\sigma_2)\sigma_1 \in B_n^+.$$

5.2. Trenzas reducidas

Consideramos el grupo simétrico S_n que está formado por todas las permutaciones del conjunto $\{1, \ldots, n\}$. Definimos la función $\rho : S_n \to B_n^+$ como sigue. Consideramos las transposiciones simples $s_1, \ldots, s_{n-1} \in S_n$, donde s_i permuta i y $i+1$ y deja el resto de elementos fijos. Las transposiciones simples generan S_n, así que cada elemento $\omega \in S_n$ puede ser expresado como una palabra $\omega = s_{i_1}s_{i_2} \cdots s_{i_r}$ con $i_1, i_2, \ldots, i_r \in \{1, 2, \ldots, n-1\}$. Si r es mínimo, entonces esta es una expresión reducida y establecemos que $\rho(\omega) = \sigma_{i_1}\sigma_{i_2} \cdots \sigma_{i_r}$. Como ya vimos en la Sección 2.1, w puede tener varias expresiones reducidas diferentes. En [11] queda demostrado que ρ está bien definido, de manera que la imagen por ρ de distintas expresiones reducidas de w dan como imagen la misma trenza. Sea

$$\pi : B_n^+ \to S_n$$

el homomorfismo de monoides definido por $\pi(\sigma_i) = s_i$ para todo $i \in \{1, \ldots, n-1\}$ es la restricción a B_n^+ del homomorfismo de monoides π definido al final de la Sección 4. Está claro que $\pi \circ \rho = id$, lo que implica que ρ es inyectiva.

Establecemos $B_n^{red} = \rho(S_n) \subset B_n^+$. Este es un conjunto finito de cardinalidad $n!$ y el homomorfismo $\pi : B_n^+ \to S_n$ es una biyección cuando se restringe a B_n^{red}. Decimos que un elemento de B_n^+ es reducido si se encuentra en B_n^{red}. Los átomos $\sigma_1, \ldots, \sigma_{n-1}$ de B_n^+ son reducidos, ya que $\sigma_i = \rho(s_i)$ para $i \in \{1, \ldots, n-1\}$. Recordamos la longitud $\lambda(w)$ para $w \in S_n$ vista en la Sección 2.1. Por definición de λ está claro que

$$\lambda(\pi(a)) = \ell(a)$$

para todo $a \in B_n^+$. El siguiente lema es una útil caracterización algebraica de B_n^{red}.

Lema 5.2.1. Un elemento de B_n^+ es reducido si y solo si $\lambda(\pi(a)) = \ell(a)$.

Demostración. Si $a = \rho(\omega)$ para algún $\omega \in S_n$, entonces $\ell(a) = \lambda(\omega) = \lambda(\pi(a))$. Por el contrario, sea $a = \sigma_{i_1}\sigma_{i_2} \cdots \sigma_{i_r} \in B_n^+$ con $r = \ell(a) = \lambda(\pi(a))$. Entonces $\pi(a) = s_{i_1}s_{i_2} \cdots s_{i_r}$ es una expresión reducida de S_n y $a = \rho(\pi(a)) \in B_n^{red}$. □

Lema 5.2.2. Un divisor por la izquierda o por la derecha de un elemento reducido de B_n^+ es reducido.

Demostración. Si $a, b \in B_n$ y $ab \in B_n^{red}$, entonces

$$\ell(a) + \ell(b) = \ell(ab) = \lambda(\pi(ab)) = \lambda(\pi(a)\pi(b)) \leq \lambda(\pi(a)) + \lambda(\pi(a)).$$

Ya que $\ell(a) \geq \lambda(\pi(a))$ y $\ell(b) \geq \lambda(\pi(b))$, estas desigualdades son realmente igualdades. Por el Lema 5.2.1, se sigue que $a, b \in B_n^{red}$. □

Lema 5.2.3. Para $u, v \in S_n$, tenemos que $\rho(u)\rho(v) = \rho(uv)$ si y solo si $\lambda(u) + \lambda(v) = \lambda(uv)$.

Demostración. Sea $a = \rho(u)\rho(v) \in B_n^+$. Tenemos que $\pi(a) = uv$ y

$$\lambda(uv) = \lambda(\pi(a)) \leq \ell(a) = \ell(\rho(u)) + \ell(\rho(v)) = \lambda(u) + \lambda(v).$$

Por tanto, por el Lema 5.2.1, $a \in B_n^{red}$ si y solo si $\lambda(uv) = \lambda(u) + \lambda(v)$. Por otra parte, $a \in B_n^{red}$ si y solo si $a = \rho(\pi(a)) = \rho(uv)$. □

Recordamos la permutación $\omega_0 = (n, n-1, \ldots, 2, 1)$ de la Sección 2.1. Es el único elemento de $\omega_0 \in S_n$ de longitud máxima. Es fácil comprobar que

$$\omega_0 = (s_1 \cdots s_{n-2}s_{n-1})(s_1 \cdots s_{n-2}) \cdots (s_1 s_2)s_1.$$

Ya que la palabra en la parte derecha tiene longitud $\lambda(\omega_0) = n(n-1)/2$, es reducido. Por lo tanto

$$\rho(\omega_0) = (\sigma_1 \cdots \sigma_{n-2}\sigma_{n-1})(\sigma_1 \cdots \sigma_{n-2}) \cdots (\sigma_1\sigma_2)\sigma_1 = \Delta_n.$$

Esto demuestra que Δ_n es reducido.

Lema 5.2.4. Un elemento de B_n^+ es reducido si y solo si es un divisor por la izquierda de Δ_n.

Demostración. Ya que Δ_n es reducido, todos sus divisores por la izquierda y por la derecha son reducidos por el Lema 5.2.2.

Por el contrario, sea $a = \rho(\pi(a)) \in B_n^{red}$. Definimos $b = \rho(\pi(a)^{-1}\omega_0) \in B_n^{red}$, $u = \pi(a)$ y $v = \pi(b) = \pi(a)^{-1}\omega_0$. Tenemos $uv = \omega_0$. Por tanto, por el Lema 2.2.1

$$\lambda(u) + \lambda(v) = \lambda(\omega_0).$$

Esta igualdad, por el Lema 5.2.3, implica que $\Delta_n = \rho(w_0) = \rho(uv) = \rho(u)\rho(v) = ab$. Por tanto, a es un divisor de Δ_n. Un argumento simiar prueba que a es un divisor por la derecha de Δ_n. □

Teorema 5.2.1. Para todo $n \geq 1$, el par (B_n^+, Δ_n) es un monoide de Garside exhaustivo.

Demostración. Ya observamos que B_n^+ es atómico con átomos $\sigma_1, \ldots, \sigma_{n-1}$. Vamos a probar que (B_n^+, Δ_n) es un monoide pre-Garside comprobando las condiciones (a) y (b) de la Definición 3.3.1.

Por el Lema 5.2.4, el conjunto de divisores por la izquierda de Δ_n coincide con el conjunto de divisores por la derecha de Δ_n y coincide con el conjunto B_n^{red}. Este último es finito y coincide con los generadores $\sigma_1, \ldots, \sigma_{n-1}$ de B_n^+. Esto verifica la condición (a).

Condición (b): vamos a probar que $\Delta_n a = \Delta_n b$ implica $a = b$ para $a, b \in B_n^{red}$. Aplicando el homomorfismo de monoides $\pi : B_n^+ \to S_n$, obtenemos

$$\pi(\Delta_n)\pi(a) = \pi(\Delta_n a) = \pi(\Delta_n b) = \pi(\Delta_n)\pi(b) \in S_n.$$

Ya que S_n es un grupo, $\pi(a) = \pi(b)$. Esto implica que

$$a = \rho(\pi(a)) = \rho(\pi(b)) = b.$$

Para probar que $a\Delta_n = b\Delta_n$ implica $a = b$ se hace de igual forma. Para cualesquiera $i, j \in \{1, \ldots, n-1\}$, establecemos

$$\sigma_{i,j} = \begin{cases} \sigma_i & si \quad i = j, \\ \sigma_i\sigma_j\sigma_i = \sigma_i\sigma_j\sigma_i & si \quad |i-j| = 1, \\ \sigma_i\sigma_j = \sigma_j\sigma_i & si \quad |i-j| \geq 2. \end{cases}$$

Establecemos $s_{i,j} = \pi(\sigma_{i,j}) \in S_n$. Es fácil comprobar que $s_{i,j} = s_i s_j$ es una expresión reducida cuando $|i-j| \geq 2$, y $s_{i,j} = s_i s_j s_i \in S_n$ es una expresión reducida cuando $|i-j| = 1$. Entonces $\sigma_{i,j} = \rho(s_{i,j}) \in B_n^{red}$ para todo i, j. Por lo tanto el conjunto B_n^{red} es exhaustivo.

Para completar la prueba, hay que comprobar la condición de la Definición 3.4.1 . Observamos que $\sigma_i \preceq \sigma_{i,j}$ y $\sigma_j \preceq \sigma_{i,j}$ para todo i, j. Afirmamos que $\sigma_{i,j}$ es el elemento mínimo del conjunto

$$\{a \in B_n^{red} \mid \sigma_i \preceq a,\ \sigma_j \preceq a\}.$$

Debemos demostrar que para cualquier $a \in B_n^{red}$, tal que $\sigma_i \preceq a$ y $\sigma_j \preceq a$, se da que $\sigma_{i,j} \preceq a$. El caso $i = j$ es trivial, por lo que consideraremos el caso $i \neq j$. Ya que los elementos de B_n^{red} están en biyección con los elementos de S_n a través de la función $\pi : B_n^+ \to S_n$, es suficiente establecer que si

$$w = \pi(a) = s_i u = s_j v$$

para algún $u, v \in S_n$ con $\lambda(u) = \lambda(v) = \lambda(w) - 1$, entonces existe $w' \in S_n$ tal que $w = s_{i,j}w'$ y $\lambda(w') = \lambda(w) - \lambda(s_{i,j})$.

Demostramos la última afirmación. Primero observamos que $u \neq v$, ya que $s_i \neq s_j$. Sea $s_{i_1}s_{i_2}\cdots s_{i_r}$ una expresión reducida para v, donde $r = \lambda(w) - 1$. Tenemos que

$$u = s_i w = s_i s_j v = s_i s_j s_{i_1} s_{i_2} \cdots s_{i_r}.$$

Ya que $\lambda(u) < \lambda(w)$, u se obtiene de $s_j s_{i_1} s_{i_2} \cdots s_{i_r}$ por eliminación de uno de los generadores. Si eliminamos el generador más a la izquierda s_j, entonces $u = s_{i_1} s_{i_2} \cdots s_{i_r} = v$, lo que es imposible. Por tanto, $u = s_j w'$ con

$$w' = s_{i_1} \cdots \widehat{s_{i_p}} \cdots s_{i_r},$$

donde algún s_{i_p} es eliminado. Por lo tanto, $\lambda(w') \leq r - 1 = \lambda(w) - 2$. Ya que

$$w = s_i u = s_i s_j w',$$

debemos tener $\lambda(w') = \lambda(w) - 2$. Esto prueba la afirmación cuando $|i - j| \geq 2$, es decir, cuando $s_{i,j} = s_i s_j$.

Consideramos el caso $|i - j| = 1$. Por el calculo previo,

$$v = s_j w = s_j s_i u = s_j s_i s_j w' = s_j s_i s_j s_{i_1} \cdots \widehat{s_{i_p}} \cdots s_{i_r}.$$

Ya que $\lambda(v) < \lambda(w) = r + 1$, u se obtiene de $s_i s_j s_{i_1} s_{i_2} \cdots s_{i_r}$ por eliminación de uno de los generadores. Si eliminamos el generador más a la izquierda s_i, entonces

$$v = s_j s_{i_1} s_{i_2} \cdots s_{i_r} = s_j w' = u,$$

lo que es imposible. Si el generador eliminado es s_j en la segunda posición, entonces

$$v = s_i s_{i_1} s_{i_2} \cdots s_{i_r} = s_j w'.$$

Obtenemos $s_i s_j w' = w = s_j v = s_j s_i w'$, lo que implica $s_i s_j = s_j s_i$. Esto es imposible ya que $|i - j| = 1$. Por tanto $v = s_i s_j w''$, donde w'' se obtiene por borrar un generador de $w' = s_{i_1} s_{i_2} \cdots s_{i_r}$. En consecuencia $\lambda(w'') \leq r - 2 = \lambda(w) - 3$ y

$$w = s_j v = s_j s_i s_j w'' = s_{i,j} w''.$$

Entonces $\lambda(w'') = \lambda(w) - 3$. □

Por el Teorema 5.2.1, el par (B_n^+, Δ_n), comparte todas las propiedades de los monoides de Garside exhaustivos. Hacemos un resumen de esas propiedades.

(1) El monoide B_n^+ tiene una presentación con generadores $[a]$, donde a se encuentra en B_n^{red}, y relaciones $[1] = 1$ y $[a][b] = [ab]$ siempre que $a, b \in B_n^{red}$ cumplan $ab \in B_n^{red}$. Usando la biyección $\rho : S_n \to B_n^{red}$ y el Lema 5.2.3, concluimos que B_n^+ tiene una presentación con generadores $[u]$, donde u se encuentra en S_n, y relaciones $[1] = 1$ y $[u][v] = [uv]$ siempre que $u, v \in S_n$ cumplan $\lambda(u) + \lambda(v) = \lambda(uv)$.

(2) Cualquier familia finita de elementos de B_n^+ tiene un único mcd por la izquierda y un único mcm por la derecha.

(3) Cualquier $a \in B_n^+$ tiene una *forma normal* $(a_1, a_2, \ldots, a_r)$ con $r \geq 0$, donde $a_1, a_2, \ldots, a_r$ son elementos únicos de $B_n^{red} \backslash \{1\}$ tales que $a = a_1 a_2 \cdots a_r$ y a_i es el mcd por la izquierda de $a_i a_{i+1} \cdots a_r$ y Δ_n para todo $i \in \{1, 2, \ldots, r\}$.

(4) El monoide B_n^+ se embebe en su grupo de fracciones. Por definición el grupo de fracciones de B_n^+ tiene la misma presentación que B_n^+ y no es más que el grupo de trenzas B_n. Esto hace que el homomorfismo de monoides $B_n^+ \to B_n$ que envía $\sigma_i \in B_n^+$ a $\sigma_i \in B_n$ para $i \in \{1, \ldots, n-1\}$ es inyectivo. En la práctica identificaremos el monoide B_n^+ con su imagen en B_n.

(5) Para $n \geq 2$, cualquier $\beta \in B_n$ puede ser escrito únicamente en la forma $\beta = \Delta_n^s b$, donde $s \in \mathbb{Z}$ y $b \in B_n^+ \subset B_n$ no es un múltiplo por la derecha de Δ_n (forma normal greedy).

(6) El problema del conjugado en B_n es equivalente al problema del conjugado en B_n^+ y puede ser resuelto como ya se ha visto.

Completamos la lista con el siguiente teorema.

Teorema 5.2.2. Cualquier familia finita de elementos de B_n^+ tiene un único mcd por la derecha y un único mcm por la izquierda.

Demostración. Consideramos la función $rev : B_n^+ \to B_n^+$ obtenida al leer las palabras de generadores $\sigma_1, \ldots, \sigma_{n-1}$ de derecha a izquierda:

$$rev(s_{i_1} s_{i_2} \cdots s_{i_{r-1}} s_{i_r}) = s_{i_r} s_{i_{r-1}} \cdots s_{i_2} s_{i_1}.$$

Esta función está bien definida, ya que las relaciones de la definición de B_n^+, siendo leidas de derecha a izquierda, dan las mismas relaciones. La función rev es un antiautomorfismo involutivo de B_n^+ en el sentido en que $rev^2 = id$, $rev(1) = 1$ y $rev(ab) = rev(a)rev(b)$ para todo $a, b \in B_n^+$. Está claro que $a \preceq b$ si y solo si $rev(a) \succeq rev(b)$ para $a, b \in B_n^+$. Usando estos hechos, es fácil deducir la existencia de mcd por la derecha y mcm por la izquierda a partir de la existencia de mcd por la izquierda y mcm por la derecha. La unicidad se obtiene del Lema 3.1.2. □

5.3. Cálculos

Procedemos a explicar cómo calcular la forma normal greedy $\beta = \Delta^s b$ de una trenza $\beta \in B_n$ que nos da la propiedad (5) de la sección previa. Primero vamos a representar β por una palabra de generadores $\sigma_1, \ldots, \sigma_{n-1}$ y sus inversos. Definimos $\nu_i \in B_n^+$ por $\nu_i \sigma_i = \Delta_n$. Entonces $\sigma_i^{-1} = \Delta_n^{-1} \nu_i$. En la palabra representando β reemplazamos todas las ocurrencias de σ_i^{-1} por $\Delta_n^{-1}\nu_i$ y expandimos todo ν_i en términos de $\sigma_1, \ldots, \sigma_{n-1}$. En la palabra resultante tenemos solo los generadores σ_i (no sus inversos) y potencias negativas de Δ_n. Haremos uso de la identidad

$$\sigma_i \Delta_n = \Delta_n \sigma_{n-i}, \tag{5.1}$$

donde $i \in \{1, \ldots, n-1\}$. Su demostración puede encotrarse en [11]. Con esta identidad podemos mover todas las potencias de Δ_n a la izquierda. De esta forma obtenemos una expansión $\beta = \Delta_n^s b$ con $s \in \mathbb{Z}$ y $b \in B_n^+$. Si b no es un múltiplo por la derecha de Δ_n, entonces tenemos la deseada expansión de β. Si $\Delta_n \preceq b$, entonces $b = \Delta_n b'$ con $b' \in B_n^+$ y $\beta = \Delta_n^{s+1} b'$. Observar que comprobar si $\Delta_n \preceq b$, es suficiente calcular $\alpha(b) \in B_n^{red}$ y ver si $\alpha(b) = \Delta_n$. Después comprobamos si b' es múltiplo por la derecha de Δ_n y así sucesivamente. El proceso se para antes de $2\ell(b)/(n(n-1))$ pasos.

Damos un ejemplo aplicando este procedimiento a

$$\beta = \sigma_1^{-1}\sigma_2\sigma_1\sigma_2^{-2} \in B_4.$$

Denotamos por $W(a)$ al conjunto de palabras de $\sigma_1, \ldots, \sigma_{n-1}$ representado a un elemento $a \in B_n^+$. De

$$\Delta_4 = \sigma_1\sigma_2\sigma_3\sigma_1\sigma_2\sigma_1 = \sigma_1\sigma_2\sigma_3\sigma_2\sigma_1\sigma_2$$

derivamos que $\sigma_1^{-1} = \Delta_4^{-1}\sigma_1\sigma_2\sigma_3\sigma_1\sigma_2$ y $\sigma_2^{-1} = \Delta_4^{-1}\sigma_1\sigma_2\sigma_3\sigma_2\sigma_1$. En consecuencia

$$\begin{aligned}\beta &= (\Delta_4^{-1}\sigma_1\sigma_2\sigma_3\sigma_1\sigma_2)(\sigma_2\sigma_1)(\Delta_4^{-1}\sigma_1\sigma_2\sigma_3\sigma_2\sigma_1)(\Delta_4^{-1}\sigma_1\sigma_2\sigma_3\sigma_2\sigma_1)\\ &= (\Delta_4^{-1}\sigma_1\sigma_2\sigma_3\sigma_1\sigma_2)(\sigma_2\sigma_1)(\Delta_4^{-1}\sigma_3\sigma_2\sigma_1\sigma_2\sigma_3)(\sigma_1\sigma_2\sigma_3\sigma_2\sigma_1)\\ &= \Delta_4^{-3}abc^2\end{aligned}$$

donde

$$a = \sigma_1\sigma_2\sigma_3\sigma_1\sigma_2 \quad b = \sigma_2\sigma_1 \quad c = \sigma_3\sigma_2\sigma_1\sigma_2\sigma_3 = \sigma_1\sigma_2\sigma_3\sigma_2\sigma_1$$

Vamos a calcular $\alpha(abc^2)$ con el objetivo de ver si abc^2 es un múltiplo por la derecha de Δ_4. Observamos que a, b y c son trenzas reducidas. Ya que $c\alpha_2 = \Delta_4$, tenemos que

$\alpha(c^2) = \alpha_2(c, c) = cc'$, donde c' es el mcd por la izquierda de σ_2 y c. Ahora $W(c)$ consiste de seis palabras.

$$\sigma_3\sigma_2\sigma_1\sigma_2\sigma_3 \quad \sigma_1\sigma_2\sigma_3\sigma_2\sigma_1 \quad \sigma_1\sigma_3\sigma_2\sigma_3\sigma_1$$

$$\sigma_1\sigma_3\sigma_2\sigma_1\sigma_3 \quad \sigma_3\sigma_1\sigma_2\sigma_3\sigma_1 \quad \sigma_3\sigma_1\sigma_2\sigma_1\sigma_3$$

Por lo tanto, $c' = 1$ y $\alpha(c^2) = c$. De $b(\sigma_2\sigma_3\sigma_2\sigma_1) = \Delta_4$, obtenemos

$$\alpha(bc^2) = \alpha_2(b, \alpha(c^2)) = \alpha_2(b, c) = bb',$$

donde b' es el mcd por la izquierda de $\sigma_2\sigma_3\sigma_2\sigma_1$ y c. Ahora

$$W(\sigma_2\sigma_3\sigma_2\sigma_1) = \{\sigma_2\sigma_3\sigma_2\sigma_1, \sigma_3\sigma_2\sigma_3\sigma_1, \sigma_3\sigma_2\sigma_1\sigma_3\}.$$

Comparando con $W(c)$, obtenemos $b' = \sigma_3\sigma_2\sigma_1$. Por tanto, $\alpha(bc^2) = d$, donde $d = \sigma_2\sigma_1\sigma_3\sigma_2\sigma_1$. Finalmente, $a\sigma_1 = \Delta_4$, implica

$$\alpha(abc^2) = \alpha_2(a, \alpha(bc^2)) = \alpha_2(a, d) = aa'$$

donde a' es el mcd por la izquierda de σ_1 y d. La lista

$$W(d) = \{\sigma_2\sigma_1\sigma_3\sigma_2\sigma_1, \sigma_2\sigma_3\sigma_1\sigma_2\sigma_1, \sigma_2\sigma_3\sigma_2\sigma_1\sigma_2, \sigma_3\sigma_2\sigma_3\sigma_1\sigma_2, \sigma_3\sigma_2\sigma_1\sigma_3\sigma_2\}$$

demuestra que $a' = 1$. Por tanto, $\alpha(abc^2) = a \neq \Delta_4$. Por lo tanto, abc^2 no es múltiplo por la derecha de Δ_4 y

$$\beta = \Delta_4^{-3}abc^2$$

es la expansión requerida de β.

5.4. Problema del conjugado en B_n

Ya hemos visto una solución para el problema del conjugado en el monoide M_Σ y en el grupo de fracciones. Sin embargo estas soluciones a priori no son computables ya que el cardinal del conjunto de conjugados de un elemento puede ser infinito. Por tanto vamos a ver algunas definiciones y resultados extraídos de [5] que nos serán útiles en el apartado de criptoanálisis para resolver el problema del conjugado algorítmicamente.

El orden parcial $\leq$ entre elementos de B_n se define de la siguiente manera:

Definición 5.4.1. Sean $v, w \in B_n$, entonces $v \leq w$, si y solo si, existen trenzas positivas $\alpha, \beta \in B_n^+$ tal que $w = \alpha v \beta$.

Como ya hemos visto, dada una trenza $w \in B_n$, podemos escribir esta de forma única como

$$w = \Delta_n^r W_1 \cdots W_s,$$

donde $W_1, \ldots, W_s \in \{\sigma_1, \ldots, \sigma_n\}$. Definiremos como $\text{ínf}(w) = r$ y $\sup(w) = s + r$. Las definiciones originales de ínfimo y supremo se encuentran en [4], pero nosotros utilizaremos estas, las cuales son equivalentes y en la práctica serán las que necesitaremos.

Definición 5.4.2. Decimos que $\gamma \in B_n^+$ es una *cola* de la trenza $\alpha \in B_n^+$, si y solo si, existe una factorización $\alpha = \beta\gamma$ con $\beta \in B_n^+$.

También definimos el automorfismo τ de B_n a través de $\tau(w) := \Delta_n^{-1} w \Delta_n$. Ahora estamos en condiciones de ver el lema en el que se basará nuestra resolución algorítmica del problema del conjugado.

Lema 5.4.1. Sean $v, w \in B_n$ dos trenzas conjugadas por una trenza positiva tales que $w = \alpha^{-1} v \alpha$ con $\alpha \in B_n^+$. Sea $\Delta_n^r W_1 \cdots W_s$ la forma normal de w. Entonces se cumplen las siguientes relaciones:

1. Si $\text{ínf}\, w < \text{ínf}\, v$, entonces $\Delta_n \tau^r(W_1^{-1})$ es una cola de α.
2. Si $\sup w > \sup v$, entonces W_s es una cola de α.

La prueba de este lema se puede ver en [4].

Capítulo 6

Criptografía

Procedemos a explicar que es la criptografía y sus conceptos básicos para entender bien en que consiste y cómo podemos aplicar el grupo de trenzas a este campo [13].

La *criptografía* es la ciencia de guardar secretos a salvo. Se asume que un emisor, al que nos referiremos aquí como *Alicia*, quiere enviar un mensaje m a un receptor, al que llamaremos *Bruno*. Ella usa un canal de comunicación inseguro, como puede ser una red de ordenadores o una linea de teléfono. El problema está cuando la información que se transmite es confidencial. El mensaje puede ser interceptado y leído o incluso modificado por un agente externo al que nos referiremos como *Eva*. El objetivo de la criptografía es proveer de métodos para evitar este tipo de ataques.

6.1. Cifrado

La tarea clásica y fundamental de la criptografía es la de proporcionar *confidencialidad* a través de los métodos de cifrado. El mensaje a transmitir se le llama *texto plano*. Alicia *cifra* el texto plano m y obtiene un *texto cifrado* c. El texto cifrado c es transmitido a Bruno. Bruno *descifrando* obtiene el texto plano a partir del texto cifrado. Para *descifrar*, Bruno necesita una información secreta, la llave de descifrado. La adversaria Eva todavía podría interceptar el mensaje cifrado. Sin embargo, el cifrado del texto plano debería garantizar la confidencialidad y prevenir que pueda extraer alguna información del texto plano a partir del texto cifrado.

El cifrado es muy antiguo. Por ejemplo, el *cifrado César* fue introducido hace más de 2000 años. Cada método de cifrado provee un *algoritmo de cifrado* E y un *algoritmo de descifrado* D. En los esquemas de cifrado clásicos, ambos algoritmos dependen de la misma llave secreta k. Esta llave k se usa para cifrar y descifrar. A estos métodos de

cifrado se les llama *simétricos*,

$$D(k, E(k, m)) = m.$$

En 1976, Diffie y Hellman introdujeron el revolucionario concepto de *criptografía de llave pública*. Proporcionaron una solución al problema del intercambio de llaves y señalaron el camino a las firmas digitales. A los métodos de *cifrado de llave pública* se les llama *asimétricos*. Cada destinatario de mensajes tiene una llave personal $k = (pk, sk)$ que se compone de dos partes: pk es la llave de cifrado y esta se hace pública, y sk es la llave de descifrado y se mantiene en secreto. Si Alicia quiere enviar un mensaje M a Bruno, ella cifra m usando la llave de cifrado de Bruno pk, la cual es pública. Bruno descifra el texto cifrado usando su llave de descifrado sk que solo conoce él,

$$D(sk, E(pk, m)) = m.$$

Cualquiera puede cifrar fácilmente el texto plano usando la llave pública pk, pero la otra dirección es complicada. Es prácticamente imposible deducir el texto plano a partir del texto cifrado sin saber la llave secreta sk.

Los métodos de cifrado de llave pública requieren cálculos más complejos y son menos eficientes que los métodos clásicos de cifrado simétrico. Por eso los métodos simétricos son usados para el cifrado de grandes cantidades de datos. Antes de aplicar el cifrado simétrico, el emisor y el receptor acuerdan una llave. Para mantener la llave en secreto necesitan un canal de comunicación seguro. Lo habitual es utilizar el cifrado de llave pública para este propósito.

6.2. Objetivos de la criptografía

Proporcionar **confidencialidad** no es el único objetivo de la criptografía. A su vez esta intenta dar solución a otros problemas. Estos son:

- **Integridad de los datos:** El receptor de los datos debería ser capaz de comprobar si el mensaje ha sido modificado durante la transmisión, ya sea accidentalmente o deliberadamente. Nadie debería ser capaz de alterar total o parcialmente el mensaje original.
- **Autenticación:** El receptor del mensaje debería ser capaz de verificar su origen. Nadie debería ser capaz de enviar un mensaje a Bruno haciéndose pasar por Alicia (*autenticación del origen de los datos*). Cuando se inicia la comunicación, Alicia y Bruno deberían ser capaces de identificar el uno al otro (*autenticación de identidad*).

- **No repudio:** El emisor debería no ser capaz negar que envió el mensaje.

Hay tanto métodos simétricos como de llave pública para asegurar la integridad de los mensajes. Los métodos simétricos clásicos requieren una llave secreta k que es compartida por el emisor y el receptor. El mensaje m se aumenta con el *código de autenticación de mensaje* (MAC acrónimo en inglés). El código es generado por un algoritmo y depende de una llave secreta. el mensaje aumentado $(m, MAC(k, m))$ está protegido contra modificaciones. El receptor del mensaje puede comprobar la integridad del mensaje $(m, \overline{m})$ comprobando que

$$MAC(k, m) = \overline{m}.$$

Los códigos de autenticación de mensaje se implementan mediante funciones hash con llave.

Las *firmas digitales* requieren métodos de llave pública. Como las clásicas firmas a mano, tienen la intención de proveer de no repudio. Las firmas digitales dependen de la clave secreta del firmante (solo pueden ser generadas por él). Por otra parte cualquier persona puede comprobar la validez de la firma utilizando un algoritmo de verificación público *Verify* el cual depende de la llave pública del firmante. Si Alicia quiere firmar un mensaje m, ella utiliza el algoritmo *Sign* con su llave secreta sk y consigue su firma $Sign(sk, m)$. Bruno recibe una firma s para el mensaje m y puede comprobar la firma viendo si

$$Verify(pk, s, m) = ok$$

con la llave pública de Alicia pk.

No es común firmar el mensaje en sí mismo, sino usar primero una *función hash criptográfica* y después firmar el valor hash. Las firmas digitales dependen del mensaje. Mensajes distintos producen firmas distintas. Así que, como los códigos de autenticación de mensaje, las firmas digitales pueden ser utilizadas para garantizar la integridad de los datos.

6.3. Protocolos criptográficos

Los algoritmos de cifrado y descifrado, funciones hash criptográficas y los generadores peusoaleatorios son básicamente los bloques de construcción (también llamados primitivas criptográficas) utilizados para resolver problemas que involucran confidencialidad, autenticación o integridad de los datos.

En muchos casos un único bloque de construcción no puede resolver un problema, sino que se necesita combinar varias primitivas. Una serie de pasos deben ser ejecutados para completar la tarea. A la sucesión de pasos bien definidos se le denomina *protocolo*

criptográfico. Es habitual añadir una condición a esta definición y es la de que dos o más partes estén involucradas. Solo usaremos el término "protocolo" si se requiere al menos dos personas para completar la tarea.

Como contraejemplo echamos un vistazo al esquema de firma digital. Un esquema típico para generar una firma digital es primero aplicar una función hash criptográfica h al mensaje m y después, en un segundo paso, calcular la firma utilizando el algoritmo de descifrado de llave pública sobre el valor hash $h(m)$. Ambos pasos están hechos por una única persona, por lo que no se considera un protocolo.

6.4. Intercambio de llaves

Los criptosistemas de llave pública y secreta asumen que los participantes tienen acceso a las llaves. En la práctica, uno puede utilizar estos sistemas si el problema de la distribución de llaves está resuelto.

El concepto de seguridad para llaves tiene dos niveles. El primer nivel emplea llaves secretas de larga duración, denominadas *llaves maestras*. Las llaves del segundo nivel están asociadas con la sesión, por lo que son llamadas *llaves de sesión*. Una llave de sesión es solo válida durante el breve tiempo de duración de una sesión. Las llaves maestras son normalmente las llaves empleadas en un criptosistema de llave pública.

Hay dos razones principales para el concepto a dos niveles. La primera es que el cifrado por llave simétrica es más eficiente que el cifrado de llave pública. por tanto, las llaves de sesión suelen ser llaves de criptosistemas simétricos, y esas deben ser intercambiadas de manera segura, usando otras llaves. La segunda razón y probablemente más importante es que el concepto a dos niveles proporciona más seguridad. Si la seguridad de una sesión se ve comprometida, solo afecta a esa sesión. Dada una llave de sesión, el número de textos cifrados disponibles para el criptoanálisis es limitado. Las llaves de sesión son generadas cuando se necesitan utilizar y se desechan después de ser usadas.

La llave maestra se utiliza para generar las llaves de sesión. Por tanto es especialmente importante prevenir los ataques sobre la llave maestra. Por eso el acceso a la llave maestra está estrictamente limitado.

6.5. Plataformas criptográficas y grupos de plataforma

Si un protocolo criptográfico está basado en un objeto algebraico cómo puede ser un grupo, un anillo u otros, entonces es llamado *plataforma criptográfica* o *plataforma*. En criptografía basada en un grupo, a este se le llama *grupo de plataforma* para el

protocolo criptográfico. La seguridad de un protocolo criptográfico depende entonces de la dificultad, computacional o teórica, de resolver un problema de teoría de grupos en el grupo de plataforma [14].

Para que un grupo de plataforma G sea adecuado para un protocolo criptográfico basado en grupos, G debe poseer ciertas propiedades que hagan el protocolo tanto eficiente de implementar como seguro. Se asume que el grupo posee una representación finita,

$$G = \langle X \mid R \rangle = \langle x_1, ..., x_n \mid R_1 = \cdots = R_m = 1 \rangle,$$

y que el protocolo de seguridad está basado en un problema de teoría de grupos que denotamos por $\mathcal{P}$. La primera necesidad es que haya una forma eficiente de representar únicamente y multiplicar los elementos de G. En la mayoría de casos esto requiere una *forma normal* para los elementos $g \in G$ en términos de unos generadores $\{x_1, ..., x_n\}$. Una forma normal es, para cada $g \in G$, una única representación en términos de los generadores. Las formas normales proporcionan un método efectivo para distinguir los elementos del grupo. La existencia de una forma normal en un grupo implica que el problema de palabra tiene solución para este, lo que es esencial para estos protocolos. Para $g \in G$ denotaremos su forma normal, en términos del conjunto de generadores X, por $NF_X(g)$. Para que sea útil en criptografía, dado $g \in G$ expresado como una palabra en $x_1, ..., x_n$, el proceso de pasar de la palabra a la forma normal única debe ser computacionalmente eficiente.

Además de que el grupo de plataforma tenga forma normal, idealmente debería presentar un crecimiento exponencial. Es decir, que la función crecimiento para G,

$$\gamma : \mathbb{N} \to \mathbb{R},$$

$$\gamma(n) = \#\{w \in G : l(w) \leq n\},$$

tiene un ratio de crecimiento exponencial. En la definición $l(w)$ se entiende como el número de letras mínimo necesitado para expresar w como una palabra en $x_1, ..., x_n$. El crecimiento exponencial es necesario, ya que esto asegura tener un gran espacio de llaves, haciendo que una búsqueda de fuerza bruta para la llave secreta sea un algoritmo no factible.

Aparte la forma normal debe presentar buena difusión al determinar las formas normales de los productos. Esto quiere decir que al encontrar la forma normal de un producto es difícil computacionalmente descubrir los factores. Esto es formalmente que si nosotros conocemos $NF_X(g_1 g_2)$ es difícil computacionalmente encontrar g_1, g_2 o $NF_X(g_1)$, $NF_X(g_2)$.

Otras necesidades para el grupo plataforma dependen del protocolo en particular. Si la seguridad está basada en el problema de grupo $\mathcal{P}$, tales como el problema de palabra o el problema del conjugado, se tiene que asumir que en G, la solución a $\mathcal{P}$ es

computacionalmente compleja (NP-Complejo) o irresoluble. Sin embargo, lo que realmente queremos es que sea *generalmente complejo*, es decir, complejo para la mayoría de entradas. La solución de $\mathcal{P}$ podría ser irresoluble pero tener una complejidad en el caso promedio polinomial. En este caso, si no tenemos cuidado al elegir las entradas, la solución de $\mathcal{P}$ es fácil y el protocolo criptográfico está roto. Esto no elimina a G como un posible grupo de plataforma pero indica que hay que tener cuidado al escoger las entradas.

Entre los primeros intentos de usar grupos no abelianos como plataformas para criptosistemas de llave pública estuvieron los esquemas de Anshel-Anshel-Goldfeld y Ko, Lee et al que veremos luego.

6.6. Funciones hash criptográficas

Vamos a hablar sobre algunos aspectos de las funciones hash, que serán utilizadas en ciertos esquemas que vamos a ver más adelante [13].

Una *función hash* es una función que toma como entrada una cadena de bits de longitud arbitraria y da como salida una cadena de bits de una longitud fija n. Matemáticamente, una función hash es una función

$$h : \{0,1\}^* \to \{0,1\}^n,\ m \longmapsto h(m).$$

Una *función hash criptográfica* es una función hash que cumple dos propiedades. Es una *función unidireccional*, es decir, que dado un valor $y \in \{0,1\}^n$, es computacionalmente inviable encontrar un $m \in \{0,1\}^*$ tal que $h(m) = y$. Además, una funcion hash criptográfica debe ser *resistente a colisiones*. Ser resistente a colisiones significa que es computacionalmente inviable encontrar $m, m' \in \{0,1\}^*$ tal que $h(m) = h(m')$. El dominio y el codominio de una función hash criptográfica puede cambiar fácilmente, ya que una cadena de bits puede codificar a elementos de otros conjuntos. A menudo, las funciones hash criptográficas son obtenidas utilizando la construcción de Merkle-Damgård explicada a continuación.

6.6.1. Construcción de Merkle-Damgård

Esta construcción es un método que reduce el problema de diseñar una función hash resistente a colisiones $h : \{0,1\}^* \to \{0,1\}^n$ al problema de construir una función resistente a colisiones

$$f : \{0,1\}^{n+r} \to \{0,1\}^n \quad (r \in \mathbb{N}, r > 0)$$

con dominio finito $\{0,1\}^{n+r}$. A esta función se le denomina *función de compresión*. Una función de compresión lleva un mensaje m de longitud $n+r$ a un mensaje $f(m)$ de longitud n. Llamamos a r el *ratio de compresión*. Procedemos a explicar en que consiste la construcción de Merkle-Damgård.

Sea $f : \{0,1\}^{n+r} \to \{0,1\}^n$ una función de compresión con ratio de compresión r. Usando f vamos a definir una función hash

$$h : \{0,1\}^* \to \{0,1\}^n.$$

Sea $m \in \{0,1\}^*$ un mensaje de longitud arbitraria. La función hash h funciona iterativamente. Para calcular el valor hash $h(m)$, comenzamos inicializando el valor hash $v = v_0$ (el mismo para todos los mensajes). El mensaje m es subdividido en bloques de longitud r. Un bloque tras otro se cogen de m, concatenados con el valor de v y comprimidos por f dan lugar a un nuevo v. El v final es el valor hash $h(m)$.

Proposición 6.6.1. Sea f una función de compresión resistente a colisones. La función hash construida con el método de Merkle-Damgård es también resistente a colisiones.

La demostración se puede ver en [13].

6.7. Esquemas criptográficos basados en trenzas

En esta parte vamos a ver diferentes algoritmos que llevan a cabo varios esquemas criptográficos utilizando los grupos de trenzas, pero solo veremos en profundidad el intercambio de llaves que es el que se utiliza en la práctica. Estos algoritmos se pueden ver en [3].

6.7.1. Intercambio de llaves

Alicia y Bruno desean compartir una llave secreta, de tal manera que un intruso que esté observando la comunicación no pueda deducir ninguna información útil sobre un secreto en común.

Esquema Anshel-Anshel-Goldfeld

Propuesto por Anshel & al. [15] este esquema es una plataforma criptográfica en el que se puede utilizar cualquier grupo de plataforma en el que el Problema de Búsqueda del Conjugado sea lo suficientemente complicado.

La llave pública consiste en dos conjuntos de trenzas $\{p_1, ..., p_l\}, \{q_1, ..., q_m\} \subset B_n$. La llave secreta de Alicia es una palabra u formada por l letras y sus inversas y la llave

secreta de Bruno es una palabra v formada por m letras y sus inversas. El intercambio de llave está compuesto de los siguientes pasos:

1. Alicia calcula la trenza $s = u(p_1, ..., p_l)$ y lo usa para calcular los conjugados $q_1' = sq_1s^{-1}, ..., q_m' = sq_ms^{-1}$. Ella envía $q_1', ..., q_m'$.

2. Bruno calcula la trenza $r = v(q_1, ..., q_m)$ y lo usa para calcular los conjugados $p_1' = rp_1r^{-1}, ..., p_l' = rp_lr^{-1}$. Ella envía $p_1', ..., p_l'$.

3. Alicia calcula $t_A = su(p_1', ..., p_l')^{-1}$.

4. Bruno calcula $t_B = v(q_1', ..., q_m')r^{-1}$.

La llave compartida es $t_A = t_B$. Sabiendo que

$$u(p_1', ..., p_l') = r\ u(p_1, ..., p_l)r^{-1},$$

$$v(q_1', ..., q_m') = s\ v(q_1, ..., q_m)s^{-1},$$

es fácil ver que

$$t_A = s\ u(p_1', ..., p_l')^{-1} = s\ r\ u(p_1, ..., p_l)^{-1}r^{-1}$$
$$= s\ r\ s^{-1}r^{-1} = s\ v(q_1, ..., q_m)\ s^{-1}r^{-1} = v(q_1', ..., q_m')r^{-1} = t_B.$$

La seguridad está basada en la dificultad de resolver una variante del Problema de Búsqueda del Conjugado en B_n, que es llamado el Problema de Búsqueda del conjugado Múltiple, en el cual se intenta encontrar una trenza conjugadora no de un único par de trenzas conjugadas (p, p') sino de una familia finita de pares de trenzas $(p_1, p_1'), ..., (p_l, p_l')$. Nótese que el Problema de Búsqueda del Conjugado Múltiple es más fácil que el Problema de Búsqueda del Conjugado, puesto que se tiene más información de los conjugadores. Sin embargo, es lo suficientemente complejo como para que el intruso sea incapaz de encontrar la llave secreta r conociendo los pares $(p_1, p_1'), ..., (p_l, p_l')$. De igual manera, no será capaz de encontrar la llave secreta s conociendo los pares $(q_1, q_1'), ..., (q_m, q_m')$.

Ejemplo 6.7.1. Vamos a realizar un ejemplo de intercambio de llaves utilizando el esquema Anshel-Anshel-Goldfeld. Para ello escogeremos como grupo de trabajo el grupo de trenza B_6. Tomaremos como llaves públicas de Alicia y Bruno las trenzas $p_1 = \sigma_1, p_2 = \sigma_3, p_3 = \sigma_5^{-1}$ y $q_1 = \sigma_4, q_2 = \sigma_2^{-1}, q_3 = \sigma_3^{-1}, q_4 = \sigma_1$ respectivamente. Obtenemos las llaves privadas utilizando

$$u(p_1, p_2, p_3) = p_3p_1p_2p_1p_2,$$

$$v(q_1, q_2, q_3, q_4) = q_4 q_1 q_3 q_2,$$

de manera que la llave privada de Alicia es $s = \sigma_5^{-1}\sigma_1\sigma_3\sigma_1\sigma_3$ y la llave privada de Bruno es $r = \sigma_1\sigma_4\sigma_3^{-1}\sigma_2^{-1}$. Calculamos los conjugados a partir de las llaves públicas y privadas.

$$\begin{aligned}
q_1' &= \sigma_5^{-1}\sigma_1\sigma_3\sigma_1\sigma_3\sigma_4\sigma_3^{-1}\sigma_1^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_5,\\
q_2' &= \sigma_5^{-1}\sigma_1\sigma_3\sigma_1\sigma_3\sigma_2^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_5,\\
q_3' &= \sigma_5^{-1}\sigma_1\sigma_3\sigma_1\sigma_3\sigma_3^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_5,\\
q_4' &= \sigma_5^{-1}\sigma_1\sigma_3\sigma_1\sigma_3\sigma_1\sigma_3^{-1}\sigma_1^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_5,
\end{aligned}$$

$$\begin{aligned}
p_1' &= \sigma_1\sigma_4\sigma_3^{-1}\sigma_2^{-1}\sigma_1\sigma_2\sigma_3\sigma_4^{-1}\sigma_1^{-1},\\
p_2' &= \sigma_1\sigma_4\sigma_3^{-1}\sigma_2^{-1}\sigma_3\sigma_2\sigma_3\sigma_4^{-1}\sigma_1^{-1},\\
p_3' &= \sigma_1\sigma_4\sigma_3^{-1}\sigma_2^{-1}\sigma_5^{-1}\sigma_2\sigma_3\sigma_4^{-1}\sigma_1^{-1}.
\end{aligned}$$

A la hora de enviar los conjugados es obvio que estos se tienen que normalizar antes, ya que si no se hace esto es fácilmente visible la llave secreta. Nosotros no lo haremos en este caso para facilitar los cálculos. Comprobamos que la llave común calculada por ambas partes coincide:

$$t_A = s\ u(p_1', ..., p_l')^{-1} = \sigma_5^{-1}\sigma_1\sigma_3\sigma_1\sigma_3 \cdot \sigma_1\sigma_4\sigma_3^{-1}\sigma_2^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_5\sigma_2\sigma_3\sigma_4^{-1}\sigma_1^{-1}$$
$$= \sigma_5^{-1}\sigma_1\sigma_3\sigma_1\sigma_3\sigma_1\sigma_4\sigma_3^{-1}\sigma_2^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_5 \cdot \sigma_2\sigma_3\sigma_4^{-1}\sigma_1^{-1} = v(q_1', ..., q_m')r^{-1} = t_B.$$

Se ha realizado una implementación en SageMath de este esquema, la cual puede verse en la Figura 6.1. Para esta se ha utilizado la clase `BraidGroup`, que viene incorporada con SageMath y nos permite trabajar de forma directa con las trenzas.

Esquema Diffie-Hellman

Aunque los grupos de trenzas no son conmutativos, estos contienen grandes subgrupos donde para cada elemento de un primer subgrupo conmuta con cada elemento de un segundo subgrupo. Por ejemplo, las trenzas que involucran conjuntos distintos de hebras que conmutan. Si denotamos por LB_n (respectivamente UB_n) al subgrupo de B_n generado por $\sigma_1, ..., \sigma_{m-1}$ (respectivamente $\sigma_{m+1}, ..., \sigma_{n-1}$) con $m = [n/2]$, cada trenza en LB_n conmuta con cada trenza en UB_n.

Esta observación es utilizada por el esquema de intercambio de llave Diffie-Hellman propuesto por Ko & al [16]. En este, la llave pública consiste en una trenza p en B_n. La llave secreta de Alicia es una trenza s en LB_n. La llave secreta de Bruno es una trenza r en UB_n. El intercambio está formado por los siguientes pasos:

```sage
class AnshelAnshelGoldfeld:

    def __init__(self, n, keyLength):
        self.n = n
        self.Bn = BraidGroup(n)
        self.keyLength = keyLength

        index_public_key = list(range(-1*(self.keyLength), self.keyLength+1))
        index_public_key.remove(0)
        self.func = random.choices(index_public_key, k = random.randint(5,10))

        self.generatePublicKey()
        self.generatePrivateKey()



    def generatePublicKey(self):
        self.publicKey = []
        index_generators = list(range(-1*(self.n), self.n+1))
        index_generators.remove(0)

        for i in range(self.keyLength):
            randomList = random.choices(index_generators, k = random.randint(5,10))
            self.publicKey.append(self.Bn(randomList))


    def keyFunction(self, f, l):
        key = self.Bn([1,-1])

        for i in f:
            if i > 0:
                aux = l[i-1]
            else:
                aux = (l[-1*i-1]^-1)

            key = key * aux

        return key

    def generatePrivateKey(self):
        self.privateKey = self.keyFunction(self.func, self.publicKey)

    def messages(self, publicKeyReceiver):
        message = []

        for t in publicKeyReceiver:
            message.append(self.lnf(self.privateKey*t*(self.privateKey^-1)))

        return message

    def commonKey(self, message, participant):
        if(participant == 0):
            key = self.privateKey*(self.keyFunction(self.func, message)^-1)
        else:
            key = self.keyFunction(self.func, message)*(self.privateKey^-1)

        return self.lnf(key)

    def lnf(self,braid):
        normal = self.Bn([1,-1])
        tup = braid.left_normal_form()

        for i in range(len(tup)):
            normal = normal*tup[i]

        return normal
```

Figura 6.1: Implementación de Anshel-Anshel-Goldfeld

1. Alicia calcula el conjugado $p' = sps^{-1}$, y se lo envia a Bruno.
2. Bruno calcula el conjugado $p'' = rpr^{-1}$, y se lo envia a Alicia.
3. Alicia calcula $t_A = sp''s^{-1}$.
4. Bruno calcula $t_B = rp'r^{-1}$.

La llave compartida es $t_A = t_B$. Esto es debido a que s y r conmutan:

$$t_A = s\ p''s^{-1} = s\ r\ p\ r^{-1}s^{-1} = r\ s\ p\ s^{-1}r^{-1} = r\ p'r^{-1} = t_B.$$

Aquí la seguridad también está basada en la dificultad de resolver una variante del Problema de Búsqueda del Conjugado donde dada una trenza p de B_n y las trenzas $p' = sps^{-1}$ y $p'' = rpr^{-1}$, con s y r pertenecientes a LB_n y UB_n respectivamente, hay que encontrar la trenza $rp'r^{-1}$ o lo que es igual $sp''s^{-1}$. El problema es lo bastante complejo como para no poder encontrar $rp'r^{-1} = sp''s^{-1}$ conociendo los pares (p, p') y (p, p'').

Ejemplo 6.7.2. Procedemos a realizar un intercambio de llave utilizando el esquema Diffie-Hellman. Para ello utilizaremos el grupo de trenza B_{10}. De esta manera Alicia utilizará el subgrupo LB_{10} generado por $\sigma_1, \sigma_2, \sigma_3, \sigma_4$ y Bruno el subgrupo RB_{10} generado por $\sigma_6, \sigma_7, \sigma_8, \sigma_8$ para obtener la llave privada. Establecemos como llave pública $p = \sigma_7^{-1}\sigma_5\sigma_1^{-1}\sigma_9\sigma_4$ y como llaves privadas de Alicia y Bruno $s = \sigma_3^{-1}\sigma_1^{-1}\sigma_4\sigma_3\sigma_2$ y $r = \sigma_7\sigma_8\sigma_7\sigma_6^{-1}\sigma_9^{-1}\sigma_8$ respectivamente. Cada uno de ellos calcula el conjugado con su llave privada y pública:

$$p' = sps^{-1} = \sigma_3^{-1}\sigma_1^{-1}\sigma_4\sigma_3\sigma_2\sigma_7^{-1}\sigma_5\sigma_1^{-1}\sigma_9\sigma_4\sigma_2^{-1}\sigma_3^{-1}\sigma_4^{-1}\sigma_1\sigma_3,$$

$$p'' = rpr^{-1} = \sigma_7\sigma_8\sigma_7\sigma_6^{-1}\sigma_9^{-1}\sigma_8\sigma_7^{-1}\sigma_5\sigma_1^{-1}\sigma_9\sigma_4\sigma_8^{-1}\sigma_9\sigma_6\sigma_7^{-1}\sigma_8^{-1}\sigma_7^{-1}.$$

Observamos que debido a la conmutatividad ambos pueden calcular la llave común de forma que coincide:

$$\begin{aligned} t_A &= s\ p''s^{-1} \\ &= \sigma_3^{-1}\sigma_1^{-1}\sigma_4\sigma_3\sigma_2 \cdot \sigma_7\sigma_8\sigma_7\sigma_6^{-1}\sigma_9^{-1}\sigma_8\sigma_7^{-1}\sigma_5\sigma_1^{-1}\sigma_9\sigma_4\sigma_8^{-1}\sigma_9\sigma_6\sigma_7^{-1}\sigma_8^{-1}\sigma_7^{-1} \cdot \sigma_2^{-1}\sigma_3^{-1}\sigma_4^{-1}\sigma_1\sigma_3 \\ &= \sigma_7\sigma_8\sigma_7\sigma_6^{-1}\sigma_9^{-1}\sigma_8 \cdot \sigma_3^{-1}\sigma_1^{-1}\sigma_4\sigma_3\sigma_2\sigma_7^{-1}\sigma_5\sigma_1^{-1}\sigma_9\sigma_4\sigma_2^{-1}\sigma_3^{-1}\sigma_4^{-1}\sigma_1\sigma_3 \cdot \sigma_8^{-1}\sigma_9\sigma_6\sigma_7^{-1}\sigma_8^{-1}\sigma_7^{-1} \\ &= rp'r^{-1} = t_B. \end{aligned}$$

□

Se ha realizado una implementación en SageMath de este esquema, la cual puede verse en la Figura 6.2. Para esta se ha utilizado la clase `BraidGroup`, que viene incorporada con SageMath y nos permite trabajar de forma directa con las trenzas.

```sage
class KoLee:
    def __init__(self, n, public_key):
        self.n = n
        self.Bn = BraidGroup(n)
        self.public_key = public_key

    @classmethod
    def generate_public_key(cls, n):
        Bn = BraidGroup(n)
        index_generators = list(range(-1*(n)+1, n))
        index_generators.remove(0)
        return Bn(random.choices(index_generators, k = random.randint(5,10)))

    def generate_private_key(self, member):
        m = self.n//2
        index_generators_sub = [];

        if member == 0:
            index_generators_sub = list(range(1, m)) + list(range(-m+1, 0))
        else:
            index_generators_sub = list(range(m+1, self.n)) + list(range(-self.n+1, -m))

        self.private_key = self.Bn(random.choices(index_generators_sub, k = random.randint(5,10)))

    def message(self):
        return self.lnf(self.private_key*self.public_key*(self.private_key^-1))

    def commonKey(self, message):
        return self.lnf(self.private_key*message*(self.private_key^-1))

    def lnf(self,braid):
        normal = self.Bn([1,-1])
        tup = braid.left_normal_form()

        for i in range(len(tup)):
            normal = normal*tup[i]

        return normal

```

Figura 6.2: Implementación del Esquema Diffie-Hellman

6.7.2. Cifrado

Aquí el problema es que Bruno desea enviar un mensaje m a Alicia, y el puede usar la llave pública de Alicia para cifrar su mensaje. Alicia debe ser capaz de recuperar el mensaje original de Bruno usando su llave privada, de manera que un intruso que observe la comunicación no pueda.

Esquema Ko & al.

Este esquema fue propuesto por Ko & al. en [16]. La notación será la empleada en el esquema de intercambio de llaves previo. Además, asumimos que h es una función hash criptográfica de B_n a $\{0,1\}^N$.

Comenzamos con $p \in B_n$ y $s \in LB_n$. La llave pública de Alicia es el par (p, p'), con $p' = sps^{-1}$. La llave privada de Alicia es s. Con el fin de enviar el mensaje m_B, el cual asumimos que se encuentra en $\{0,1\}^N$, usando $\oplus$ para la operación booleana "or-exclusivo"(es decir, la suma en $\mathbb{Z}/2\mathbb{Z}$):

1. Bruno elige una trenza aleatoria r en UB_n y envía el texto cifrado $m'' = m_B \oplus h(rp'r^{-1})$ junto con el dato auxiliar $p'' = rpr^{-1}$.

2. Alicia calcula $m_A = m'' \oplus h(sp''s^{-1})$.

Entonces tenemos que $m_A = m_B$ es decir, Alicia recupera el mensaje original de Bruno. Esto se debe a que las trenzas r y s conmutan,

$$sp''s^{-1} = s\ r\ p\ r^{-1}s^{-1} = r\ s\ p\ s^{-1}r^{-1},$$

y entonces tenemos que

$$m_A = m_B \oplus h(rp'r^{-1}) \oplus h(sp''s^{-1}) = m_B.$$

Aquí la seguridad está basada en la dificultad del Problema del Conjugado inspirado en Diffie-Hellman para B_n. Bajo las hipótesis sobre h, ser capaz de romper el sistema implica tener que recuperar los valores comunes $rp'r^{-1}$ y $sp''s^{-1}$ a partir de los pares (p, p') y (p, p'').

6.7.3. Autenticación de identidad

Ahora el problema es que Alicia desea probar su identidad a Bruno, es decir, ella desea probar que conoce la llave privada sin permitir que un intruso que observe la comunicación deduzca nada sobre su llave privada.

Esquema inspirado en Diffie-Hellman

El siguiente esquema desafío-respuesta [18] está inspirado en el esquema previo de Ko & al. La llave pública es un par de trenzas conjugadas (p, p') en B_n con $p' = sps^{-1}$, mientras que la llave privada de Alicia es la trenza s utilizada para conjugar p y p'. Asumimos que $s \in LB_n$. También usaremos h como función hash criptográfica en B_n. Entonces para realizar la autenticación se hace el siguiente intercambio:

1. Bruno elige una trenza aleatoria $r \in UB_n$, y le envía el desafío $p'' = rpr^{-1}$ a Alicia.

2. Alicia le envía la respuesta $y = h(sp''s^{-1})$.

3. Bruno comprueba que $y = h(rp'r^{-1})$.

Una respuesta correcta por parte de Alicia implica la aceptación por parte de Bruno, debido a que las trenzas r y s conmutan, y por tanto, tenemos que $rp'r^{-1} = sp''s^{-1}$. Por otra parte, la aceptación ocurre solo si la respuesta de y de A satisface $y = h(rp'r^{-1})$, por tanto, bajo las hipótesis de h, solo si Alicia ha sido capaz de resolver el Problema del Conjugado inspirado en Diffie-Helman para B_n. Una vez más, la seguridad recae en la dificultad de resolver este problema.

Esquema inspirado en Fiat-Shamir

Este esquema de autenticación es propuesto por Sibert & al [18]. Este esquema hereda del esquema Fiat-Shamir lo que implica repetir el desafío-respuesta en tres pasos. Como antes, las llaves públicas son un par de trenzas conjugadas (p, p') con $p' = sps^{-1}$, mientras s, la trenza conjugadora, es la llave privada de Alicia. En contraste al esquema previo $p, s \in B_n$, es decir, no asumimos que pertenezcan a un subgrupo en particular como LB_n o UB_n. Asumimos que h es una función hash unidireccional libre de colisiones en B_n. El procedimiento de autenticación consiste en repetir k veces los siguientes tres pasos:

1. Alicia elige una trenza aleatoria $r \in B_n$ el *compromiso* $x = h(rp'r^{-1})$.

2. Bruno elige un bit aleatorio c y se lo envia a Alicia.

3. Si $c = 0$, Alicia envía $y = r$, y Bruno comprueba que $x = h(yp'y^{-1})$.

4. Si $c = 1$, Alicia envía $y = rs$, y Bruno comprueba que $x = h(yp'y^{-1})$.

Si Alicia conoce s y responde correctamente, ella es aceptada por Bruno. Para $c = 0$, tenemos directamente $x = h(yp'y^{-1})$, mientras que para $c = 1$, tenemos $ypy^{-1} = (rs)p(rs)^{-1} = rp'r^{-1}$, por tanto $x = h(ypy^{-1})$. Por otra parte, si Alicia es deshonesta, ella puede hacer trampas y enviar la respuesta correcta en ambos casos. En el caso $c = 0$, basta con que Alicia mantenga un registro de r y enviar respuestas coherentes. En el caso $c = 1$ es suficiente con que el compromiso x se elige para anticipar la igualdad $x = h(ypy^{-1})$ que también es fácil. Pero un tramposo no puede elegir su compromiso de tal manera que acierte correctamente ambos casos. Si anticipa $c = 0$, la probabilidad de responder correctamente para $c = 1$ es despreciable y viceversa. Un tramposo nunca tendrá más de 0,5 de probabilidad de acertar. Así que si repetimos el intercambio k veces, la probabilidad del tramposo de acertar sera de $1/2^k$.

La seguridad de este esquema radica en la dificultad del Problema de Búsqueda del Conjugado original en B_n. Tener más de un 0.5 de probabilidad de ser aceptado significa ser capaz de responder para $c = 0$ y $c = 1$, es decir, conocer y, y' que cumplan $x = h(ypy^{-1})$ y $x = h(y'p'y'^{-1})$. Como se supone que h es libre de colisión, es decir, es virtualmente inyectiva, se tendría que $ypy^{-1} = y'p'y'^{-1}$, que es equivalente a $(y'^{-1}y)p(y'^{-1}y)^{-1} = p'$. Esto implicaría que para que suceda esto se debe solucionar el Problema de Búsqueda del Conjugado para (p, p').

6.7.4. Firma

El problema consiste en que Alicia quiere enviar a Bruno un mensaje (texto plano o cifrado) junto con una firma que pruebe el origen del mensaje. Observar que con la firma se puede implementar la autenticación. Esto se haría enviando Bruno a Alicia un mensaje y que esta se lo devolviera firmado.

Vamos a ver dos esquemas de firma basados en trenzas propuestos por Ko & al [17]. El primer esquema es más simple y legible mientras que el segundo es el esquema recomendado. Estos esquemas usan la supuesta diferencia de complejidad entre el Problema del Conjugado y el Problema de Búsqueda del Conjugado. Para valores medios de n (típicamente $n = 20$), al usar una representación lineal de las trenzas hace posible decidir si dos trenzas de B_n son conjugadas sin ser capaz de determinar la trenza conjugada cuando existe.

Esquema simple

Las llaves públicas son un par de trenzas conjugadas (p, p') con $p' = sps^{-1}$, mientras que s, que es la trenza conjugadora, es la llave privada de Alicia. Las trenzas p y s pertenecen a B_n. Usamos H como función hash criptográfica de $\{0, 1\}^*$ a B_n. Usamos $\sim$ para la conjugación en B_n. El esquema es como sigue:

1. Alicia firma el mensaje m con $q' = sqs^{-1}$, donde $q = H(m)$.

2. Bruno comprueba que $q' \sim q$ y que $p'q' \sim pq$.

Si Alicia usa la llave secreta s, tenemos que $q' = sqs^{-1}$ y que $p'q' = spqs^{-1}$, así que la firma es aceptada. La seguridad de este esquema se encuentra en el Problema de Búsqueda del Conjugado Coincidente. Este consiste en que teniendo $p' \sim p$ y q en B_n, hay que encontrar q' que satisfaga que $q' \sim q$ y $p'q' \sim pq$. Está demostrado que este problema tiene al menos la misma complejidad que el Problema de Búsqueda del Conjugado para el par (p, p').

Esquema recomendando

Una posible debilidad del esquema previo está en que un uso repetido de este revela muchos pares (q_i, q_i') asociados a un conjugador s en común. Para abordar esto se modifica el esquema incorporando una trenza aleatoria. Aquí de nuevo usaremos h como función hash criptográfica de B_n a $\{0,1\}^*$:

1. Alicia elige una trenza aleatoria $r \in B_n$.

2. Alicia firma el mensaje m con la 3-tupla (p'', q'', q'), donde $p'' = rpr^{-1}$, $q = H(mh(p''))$, $q'' = rqr^{-1}$ y $q' = rs^{-1}qsr^{-1}$.

3. Bruno comprueba que $p'' \sim p$, $q'' \sim q'$, $p''q'' \sim pq$ y $p''q' \sim p'q$.

El análisis de seguridad es similar al del esquema previo.

6.8. Resultados

Se han ejecutado las implementaciones realizadas de los protocolos de intercambio de llave, en concreto, los esquemas Anshel-Anshel-Goldfeld y Diffie-Hellman. Estas ejecuciones se han realizado en una máquina con un procesador Intel(R) Core(TM) i7-4720HQ CPU @ 2.60GHz. Se ha medido el tiempo de ejecución variando el número de hebras de las trenzas con las que se trabaja, n, y la longitud de palabra de las llaves utilizadas, l. Se ha intentado realizar tres muestras para que medición fuese más representativa, pero cuando los tiempos exceden un limite se reduce el número de muestras. En las Tablas 6.1 y 6.2 se muestran los tiempos obtenidos.

Muestras	n	l	Tiempo (s)	Desviación Estándar (s)	Mínimo (s)	Máximo (s)
3	5	5	6.28	0.67	5.77	7.05
3	5	10	14.77	14.62	2.81	31.07
3	10	5	192.54	199.27	42.88	418.74
1	10	10	658.55	0	658.55	658.55
2	15	5	416.68	465.42	87.58	745.79
1	15	10	423.07	0	423.07	423.07
3	20	5	235.87	165.23	81.02	409.83
1	20	10	1246.68	0	1246.68	1246.68
1	25	5	2350.10	0	2350.10	2350.10

Cuadro 6.1: Tiempos de ejecución del esquema Anshel-Anshel-Goldfeld

Muestras	n	l	Tiempo (s)	Desviación Estándar (s)	Mínimo (s)	Máximo (s)
3	5	5	1.94	0.10	1.83	2.03
3	5	10	6.46	3.26	4.06	10.17
3	10	5	28.66	11.48	15.40	35.52
3	10	10	180.22	56.13	117.33	225.24
3	15	5	105.46	26.22	76.81	128.27
1	15	10	643.59	0	643.59	643.59
1	20	5	477.95	0	477.95	477.95
1	20	10	1318.74	0	1318.74	1318.74
1	25	5	665.31	0	665.31	665.31

Cuadro 6.2: Tiempos de ejecución del esquema Diffie-Hellman

Capítulo 7

Criptoanálisis

Cuando se introdujeron los criptosistemas basados en grupos de trenzas hubo una gran expectación. La dificultad de los principales problemas de teoría de grupos, el problema del conjugado y el problema de la búsqueda del conjugado, parecían lo suficientemente complejos en B_n como para ser seguro. Sin embargo las investigaciones que se han realizado sobre el grupo de trenzas han hecho evidente que, sin consideraciones especiales, los criptosistemas basados en estos son inseguros. A continuación vamos a ver algunos de los posibles ataques que se han visto [14].

Estos ataques se pueden clasificar según diferentes categorías. Ataques basados directamente en la solución del problema del conjugado vía los llamados conjuntos de cumbre, ataques basados en longitud los cuales utilizan la longitud de las trenzas, ataques teóricos de representación, y finalmente ataques en la difusión de multiplicación de elementos de grupo por variación de formas normales.

7.1. Ataques al Problema de Búsqueda del Conjugado

Ya que la seguridad de los criptosistemas de grupos de trenzas iniciales estaba basada en la dificultad en resolver el problema del conjugado, el problema de búsqueda del conjugado y el problema de búsqueda del conjugador, la mayoría de los ataques han ido enfocados a estos problemas. El problema del conjugado para B_n fue resuelto originalmente por Garside como ya hemos visto y se asume que es complejo. Por lo tanto pareció que usar el problema del conjugado o el problema de búsqueda del conjugador era suficientemente seguro para propósitos criptográficos. Recientemente ha habido investigaciones sobre la complejidad de la solución para varios problemas de búsqueda del conjugado y se han expuesto debilidades en los criptosistemas basados en estos.

La solución para el problema del conjugado en B_n usa lo que llamamos *conjuntos de cumbre*. Para cada $b \in B_n$ su conjunto de cumbre, el cual denotaremos por $SS(b)$ consiste de un conjunto finito de conjugados de b. El conjunto de cumbre contiene un subconjunto llamado super conjunto de cumbre denotado por $SSS(b)$ que es algorítmicamente computable. El-Rifai y Morton demostraron cómo calcular para cada $b \in B_n$ el super conjunto de cumbre $SSS(b)$. Es más, dos trenzas $b, b' \in B_n$ son conjugadas si y solo si sus super conjuntos de cumbre coinciden, o equivalentemente, si y solo si se intersecan.

Teorema 7.1.1. Para cada $b \in B_n$ el conjunto $SSS(b)$ es finito y computacionalmente computable. Si $b, b' \in B_n$ entonces b y b' son conjugados si y solo si $SSS(b) \cap SSS(b') \neq \emptyset$.

Por lo tanto para resolver el problema del conjugado, y en esencia romper los protocolos de tipo Ko-Lee, hay que encontrar el super conjunto de cumbre de los elementos. Recientemente se ha encontrado un refinamiento del super conjunto de cumbre llamado ultra conjunto de cumbre $USS(b)$ introducido por Gebhardt. El tamaño de $USS(b)$ es a menudo mucho más pequeño que $SSS(b)$.

El problema es que la complejidad de los algoritmos para calcular estos conjuntos no está clara ya que no se conoce su cardinalidad, pero el de este último tiene complejidad polinomial en media.

7.1.1. Ataques basados en longitud

El protocolo Anshel-Anshel-Goldfeld depende del problema de búsqueda del conjugador. Sin embargo, para atacar el protocolo no es necesario encontrar el conjugador exacto. Si b, b' son conjugados entonces conociendo cualquier conjugador, es decir, un elemento w tal que $wbw^{-1} = b'$ es suficiente.

Los ataques basados en longitud han sido usados efectivamente en ataques al problema de búsqueda del conjugador. Recordamos que usando la forma normal de Garside cada trenza tiene una longitud bien definida. Esto nos da una función de longitud bien definida en el grupo de trenzas B_n.

En los protocolos del tipo Anshel-Anshel-Goldfeld que dependen del problema del conjugador un atacante tiene acceso a múltiples conjuntos de pares (a, axa^{-1}) para un valor secreto a tomado de un conjunto con un número finito de generadores conocidos. El principio común en los ataques basados en longitud es intentar recuperar un conjugador para el par (b, b') el cual se supone se deriva a la forma de b conjugando iterativamente b' en un nuevo elemento $tb't^{-1}$ de manera que la longitud sea mínima. Si definimos la distancia entre dos trenzas a, b como $d(a, b) = l(ab^{-1})$, esto nos da una verdadera métrica en el grupo de trenzas la cual satisface la desigualdad triangular. La idea en

un ataque basado en longitud es que no existe demasiada cancelación multiplicando por generadores σ_i y por lo tanto si un generador σ_i no es un segmento inicial de un conjugador a entonces con una probabilidad distinta de cero la longitud $\sigma(at_ia^{-1})\sigma^{-1}$ es mayor que la longitud de at_ia^{-1}. Entonces aplicar cada generador σ_i hasta que la longitud se reduzca y este generador puede ser despegado del conjugador.

Ha habido un considerable éxito con esta técnica. Parte del éxito sin embargo es debido al hecho de que los grupos de trenzas tienen la propiedad de grupo libre genérica. Hacer reducción en longitud es realmente hacer reducción de grupo libre. De hecho, muchos criptosistemas basados en grupos de trenzas han sido rotos porque los grupos de trenzas tenían la propiedad de grupo libre genérica.

7.1.2. Ataques teóricos de representación

Muchos ataques con éxito en los protocolos criptográficos de grupo de trenzas han usado varias representaciones lineales. Burau, Burau coloreado y Lawrence-Krammer. En muchos ataques encontrar la llave secreta es reducido, usando estas representaciones, a álgebra lineal sobre campos finitos. Recordamos cómo procedía el protocolo de Ko-Lee:

Bruno y Alicia eligen un subgrupo dentro de B_n con elementos conmutativos. A es el subgrupo de Alicia y B el subgrupo de Bruno. Alicia elige aleatoriamente un $a \in A$. Este elemento a será su llave secreta. Su llave pública será (g, g^a) donde $g^a = aga^{-1}$ es el conjugado de g por su llave secreta a. Toda información pública y comunicación está en términos de las formas normales de estos elementos.

De igual manera Bruno elige aleatoriamente un elemento $b \in B$. Este elemento b será su llave secreta. Su llave pública es (g, g^b) donde $g^b = b^{-1}gb$ es el conjugado de g por su llave secreta b. Como Alicia, toda su información publica y comunicación está en términos de la forma normal de estos elementos.

La llave secreta compartida es g^{ab}. Cheon y Jun demostraron cómo usar la representación de Lawrence-Krammer para encontrar la llave secreta conociendo solo g^a y g^b. Supongamos que $Y_a = \rho(g^a)$ y $Y_b = \rho(g^b)$ son las imágenes de g^a y g^b bajo la representación de Lawrence-Krammer ρ. Cheon y Kim trabajando módulo un primo p y ciertos polinomios irreducibles en t y q solucionan para la matriz M en las ecuaciones

$$Y_aM = MY_b,$$

$$\rho(\sigma_i)M = M\rho(\sigma_i) \quad \sigma_i \in B_{2n}.$$

En este ataque incluso aunque la solución a las ecuaciones lineales pueda no ser $\rho(a)$la conmutatividad de las ecuaciones nos da

$$MY_bM^{-1} = \rho(b)Y_a\rho(b)^{-1} = \rho(abxb^{-1}a^{-1}),$$

el cual puede ser elevado a la trenza necesaria. Este ataque es efectivo porque las matrices provenientes de ρ cumplen unos limites muy restrictivos. La efectividad de este método demuestra que la representación de lawrence-Krammer permite una solución al problema de Diffie-Hellman en grupos de trenzas en tiempo polinomial.

7.2. Aproximación heurística al problema del conjugado

Vamos a ver una heurística que nos permitirá, de forma eficiente, resolver en ocasiones el problema del conjugado devolviendonos un conjugador. De esta manera también resolvemos el problema de búsqueda del conjugado. Esta heurística entraría dentro del conjunto de ataques basados en longitud. El algoritmo ha sido extraído de [5]. Aquí mostramos el pseudocódigo del algoritmo.

- Entrada: $v, w \in B_n$ con $w = x^{-1}vx$ para algún $x \in B_n^+$ con $\inf x = 0$
- Salida: $\alpha \in B_n^+$ con $w = \alpha^{-1}v\alpha$ o 'failed'.

1. Inicializamos α como una palabra vacia ε.
2. Ponemos v y w en forma normal, asi que $\Delta_n^r W_1 \cdots W_s$.
3. While $\inf w < inf v$ do
   Sea $\gamma := \Delta\tau^r(W_1^{-1})$, $\alpha := \gamma\alpha$, $w := \gamma w \gamma^{-1}$.
   Ponemos w en forma normal.
4. While $\sup w > \sup v$ do
   Sea $\gamma := W_s$, $\alpha := \gamma\alpha$, $w := \gamma w \gamma^{-1}$.
   Ponemos w en forma normal.
5. Sea $\mu :=$ GuessPermutation(v, w), $\alpha := \mu\alpha$, $w := \mu w \mu^{-1}$.
6. If $v = w$, then
   Return α.

else
   Return 'failed'.

Como se puede ver, el algoritmo restringe la entrada para trenzas conjugadas positivamente. También requiere que la trenza conjugadora $x = \Delta_n^r X_1 \cdots X_s$ cumpla que $\inf x = r = 0$. Estas restricciones no son relevantes, puesto que si tenemos $w = x^{-1}vx$, entonces se tiene también que $w = \tilde{x}^{-1}v\tilde{x}$ para $\tilde{x} := \Delta_n^{r \bmod 2} X_1 \cdots X_s$. Por tanto o $\tilde{x}$ o

$\Delta_n^{-1}\tilde{x}$ es una trenza positiva con ínfimo 0. Así que en el caso general habrá que ejecutar el algoritmo para las entradas v, w y $\Delta_n^{-1}v\Delta_n, w$.

Pasamos a explicar el algoritmo. En los pasos 1 y 2 inicializamos α como una palabra vacía para calcular el conjugador x y preparamos $v, w \in B_n$ normalizándolos. En los pasos 3 y 4 utilizamos el Lema 5.4.1, de forma que al completar estos pasos tenemos un nuevo problema del conjugado donde la longitud canónica de w es menor que la de v. Durante estos pasos se va construyendo α, de manera que en virtud del Lema 5.4.1, al principio del paso 5 tenemos que $w = x'^{-1}vx'$ con $x = x'\alpha$ y $x \in B_n^+$. Asumimos en esta situación que si todavía no se ha descubierto la parte x' de la palabra conjugadora x, está esencialmente determinada por la permutación $\pi(x')$. Consecuentemente, en este paso, adivinamos la permutación $\pi(x')$, es decir, la función GuessPermutation(v,w) nos devuelve un factor canónico μ tal que $\pi(\mu) = \pi(x')$. Encontrar dicha palabra de trenza μ en el caso general consistirá en resolver el problema del conjugado $\pi(w) = \pi(x')^{-1}\pi(v)\pi(x')$ en el grupo simétrico S_n. En el caso de trenzas puras esto no funcionaría y habría que utilizar otro método mencionado en [5]. Finalmente si w y v coinciden, esto significa que las trenzas originales de la entrada son conjugadas y que el conjugador es el α devuelto. Si devuelve 'failed' no se obtiene ninguna conclusión.

En la Figura 7.2 vemos la implementación realizada del algoritmo. La Figura 7.3 muestra la función GuessPermutation, la cual nos devuelve una trenza reducida cuya permutación asociada nos da la solución al problema del conjugado en el grupo simétrico que se obtiene a partir de nuestras trenzas. Por último, En la Figura 7.1 aparece una función en la que, dada una permutación, nos devuelve la trenza reducida asociada. Para la implementación de estas dos funciones ha sido necesaria la utilización de las propiedades vistas sobre las permutaciones. Han sido realizadas en SageMath y se han utilizado las clases `BraidGroup` y `Permutation`.

Se ha probado la heurística con el objetivo de ver la efectividad de la misma. Las pruebas realizadas han utilizado trenzas cuya longitud de palabra l y número de trenzas n hemos ido variando. Para cada valor de estas variables hemos repetido el experimento en total de 10 veces y se ha tomado la tasa de acierto que queda reflejada en la Tabla 7.1.

En el artículo original [5], los resultados que obtiene la implementación de la heurística que realizan son los que aparecen en la Tabla 7.2.

```python
# Se le pasa una lista que define la permutación
def pi_inverse(l):
    n = len(l)
    Bn = BraidGroup(n)
    p = PermutationGroupElement(l)
    ciclos = p.cycle_tuples()
    b = Bn([1,-1])


    # Descomponemos en transposiciones los ciclos
    desc_trans = []
    for i in range(len(ciclos)):
        for j in range(len(ciclos[i])-1,0, -1):
            desc_trans.append((ciclos[i][0], ciclos[i][j]))

    desc_trans_sim = []
    for i in range(len(desc_trans)):
        d = [(desc_trans[i][0],desc_trans[i][0]+1)]
        for j in range(desc_trans[i][0]+1,desc_trans[i][1]):
            d = [(j,j+1)] + d + [(j,j+1)]
        desc_trans_sim = desc_trans_sim + d

    for i in range(len(desc_trans_sim)-1,-1,-1):
                b = b * Bn([desc_trans_sim[i][0]])


    return b

```

Figura 7.1: Implementación de la función π^{-1}

n	l	Muestras	Tasa de éxito
5	5	10	50.0 %
5	10	10	10.0 %
10	5	10	90.0 %
10	10	10	0.0 %
15	5	10	70.0 %
15	10	10	30.0 %
20	5	10	80.0 %
20	10	10	50.0 %
25	5	10	70.0 %

Cuadro 7.1: Tasa de éxito de la implementación realizada.

n	l	Muestras	Tasa de éxito
45	3	1000	78.1 %
50	5	1000	79.1 %
70	7	1000	79.1 %
90	12	1000	80.0 %

Cuadro 7.2: Tasa de éxito del artículo original [5].

```
def breakConjugate(v,w,n):
    Bn = BraidGroup(n)
    delta = Bn.delta()
    alpha = Bn([1,-1])
    v_normal = v.left_normal_form()
    w_normal = w.left_normal_form()
    iteraciones = 0
    max_iteraciones = 100



    rw = inf(w_normal,n)
    rv = inf(v_normal,n)


    while rw < rv and iteraciones < max_iteraciones:
        gamma = delta*tauC(w_normal[1],rw,n)
        alpha = gamma*alpha
        w = gamma*compact(w_normal,n)*(gamma^-1)
        w_normal = w.left_normal_form()
        rw = inf(w_normal,n)
        iteraciones += 1



    sw = len(w_normal)-1 + inf(w_normal,n)
    sv = len(v_normal)-1 + inf(v_normal,n)
    while sw > sv and iteraciones < max_iteraciones:
        gamma = w_normal[len(w_normal)-1]
        alpha = gamma*alpha
        w = gamma*compact(w_normal,n)*(gamma^-1)
        w_normal = w.left_normal_form()
        sw = len(w_normal)-1 + inf(w_normal,n)
        iteraciones += 1

    v = compact(v_normal,n)
    w = compact(w_normal,n)
    mu = GuessPermutation(v,w,n)
    alpha = mu*alpha
    w = mu*w*(mu^-1)

    if v == w and iteraciones < max_iteraciones:
        return alpha
    else:
        return "failed"

```

Figura 7.2: Heurística del problema del conjugado

```sage
def GuessPermutation(v,w,n):
    vp = PermutationGroupElement(v.permutation()).cycle_tuples(singletons=True)
    wp = PermutationGroupElement(w.permutation()).cycle_tuples(singletons=True)

    vp = order_cycles(vp)
    wp = order_cycles(wp)


    tau = [i for i in range(n)]

    for i in range(len(wp)):
        for j in range(len(wp[i])):
            tau[vp[i][j]-1] = wp[i][j]
    #print(tau)
    return pi_inverse(tau)

```

Figura 7.3: Función GuessPermutation

Capítulo 8

Conclusión y trabajos futuros

En este trabajo hemos estudiado la aplicación el grupo de trenzas y su aplicación en el área de la criptografía, desde los fundamentos teóricos hasta su uso en métodos criptográficos y el estudio algunas de sus vulnerabilidades.

Como punto de partida tomamos la construcción del grupo de trenzas tanto desde el punto de vista algebraico como topológico y demostramos su equivalencia haciendo uso de los conceptos básicos vistos previamente.

Realizamos un profundo estudio de los monoides de Garside a partir de los cuales obtenemos un conjunto de propiedades necesarias para trabajar tanto con el monoide como con el grupo de fracciones asociado. Entre estas propiedades se encuentran la cancelatividad, soluciones al problema de palabra y al problema del conjugado, y una forma normal.

A partir de la definición del monoide de trenzas hemos demostrado que este es un monoide de Garside. Hemos visto que su grupo de fracciones es el grupo de trenzas inicialmente definido, obteniendo las propiedades necesarias para trabajar con él.

Hemos repasado los principales métodos criptográficos, mostrando varios algoritmos que los llevan a cabo haciendo uso de las trenzas y del problema del conjugado o variantes suyos como base de su seguridad. También se ha realizado una implementación de los algoritmos para el intercambio de llaves en los que se han realizado mediciones de los tiempos de ejecución para diferentes valores de los parámetros del algoritmo.

Por último hemos vistos algunas técnicas de criptoanálisis que se basan en resolver el problema del conjugado en el grupo de trenzas para romper la seguridad de los métodos vistos. Se ha llevado a cabo la implementación de una heurística que resuelve el problema del conjugado y se ha mostrado la tasa de acierto en conjuntos de experimentos con diferentes parámetros.

Los ataques que hemos visto muestran un riesgo real para los métodos criptográficos basados en el problema del conjugado del grupo de trenzas y sus variantes. Si se au-

mentase la complejidad, incrementando el tamaño de los parámetros de los algoritmos criptográficos para volverlos más seguros, los tiempos de ejecución de estos se volverían muy largos.

Algunas de las futuras lineas de trabajo para mejorar la seguridad de estos métodos consisten en cambiar el problema en el que se basan reemplazando al problema del conjugado [3]. Una posible vía son los problemas de raíz. El grupo de trenzas es un grupo libre de torsión, es decir, si b es una trenza no trivial, entonces b^e para $e \geq 2$, es no trivial. De esta propiedad surgen de manera natural dos problemas. El primero es el problema de existencia de raíz para un exponente e que consiste en que dada una trenza b de B_n hay que determinar si existe un $c \in B_n$ satisfaciendo $c^e = b$. El segundo es el problema de extracción de la raíz para un exponente e y que, dadas las condiciones del anterior problema, hay que encontrar el c que satisface $c^e = b$.

La mayoría de ataques que se implementan son basados en longitud. Estos ataques aprovechan la mala difusión de la forma normal para trenzas generadas aleatoriamente. Una posible solución para solventar este problema es investigar la generación de llaves que tengan una buena difusión.

Otro camino a explorar es el problema de longitud mínima en el que hay que encontrar la palabra de longitud mínima representando una trenza. No es estrictamente un problema dentro del grupo B_n, sino más bien un problema para B_n junto con la elección de una familia de generadores concreta.

También podemos escoger operaciones más exóticas. No es necesario elegir únicamente operaciones de grupo de B_n. La utilización de otros operaciones con propiedades deseables nos puede permitir modificar el problema del conjugado para que este sea más seguro.

Bibliografía

[1] Emil Artin, *Theorie der zöpfe*, 1925.

[2] Emil Artin, *Theory of braids*, Vol. 48, No. 1, January, 1947.

[3] Patrick Dehornoy, *Braid-based cryptography.*

[4] Elsayed A. Elrifai,Hugh R. Morton, *Algorithms for positive braids.*

[5] Dennis Hofheinz,Rainer Steinwandt, *A Practical Attack on Some Braid Group Based Cryptographic Primitives.*

[6] Stevev Roman, *Fundamentals of group theory. An advance aproach.*

[7] *Algebra abstracta*
https://es.wikibooks.org/wiki/Matemáticas_Universitarias/Álgebra_Abstracta

[8] Pierre Antoine Brillet, *Abstract algebra.* 2007.

[9] B. Suri, *Free groups - basics.* 2010.

[10] John M. Howie, *Fundamentals of semigroup theory.*

[11] Christian Kassel, Vladimir Turaev, *Braid groups.*

[12] Kunio Murasugi, Bohdan l. Kurpita, *A study of braids.*

[13] Hans Delfs, Helmut Knebl, *Introduction to cryptography. Principles and Applications.*

[14] Gilbert Baumslag, Benjamin Fine, Martin Kreuzer, Gerhard Rosenberger,*A course in mathematical cryptography.*

[15] I. Anshel, M. Anshel, D. Goldfeld, *New key agreement protocols in braid group cryptography.* 2001.

[16] K.H. Ko, S.J. Lee, J.H. Cheon, J.W. Hang, J.S. Kang, C. Park, *New public-key cryptosystem using braid groups*. 2000.

[17] K.H. Ko, P. Dehornoy, M. Girault, J.W. Lee, *New signature scheme using conjugacy problem, Preprint*. 2002.

[18] H. Sibert, D.H. Choi, M.S. Cho, J.W. Lee, *Entity authentication schemes using braid word reduction*. 2003.

[19] *Imagen de Emil Artin*. De Konrad Jacobs, Erlangen - Mathematisches Forschungsinstitut Oberwolfach, https://opc.mfo.de/detail?photoID=116, CC BY-SA 2.0 de, https://commons.wikimedia.org/w/index.php?curid=3898471

[20] *Manual Sage. Grupo de trenzas.* https://doc.sagemath.org/html/en/reference/groups/sage/groups/braid.html

[21] *Manual Sage. Grupo de permutaciones.* https://doc.sagemath.org/html/en/reference/groups/sage/groups/perm_gps/permgroup_element.html