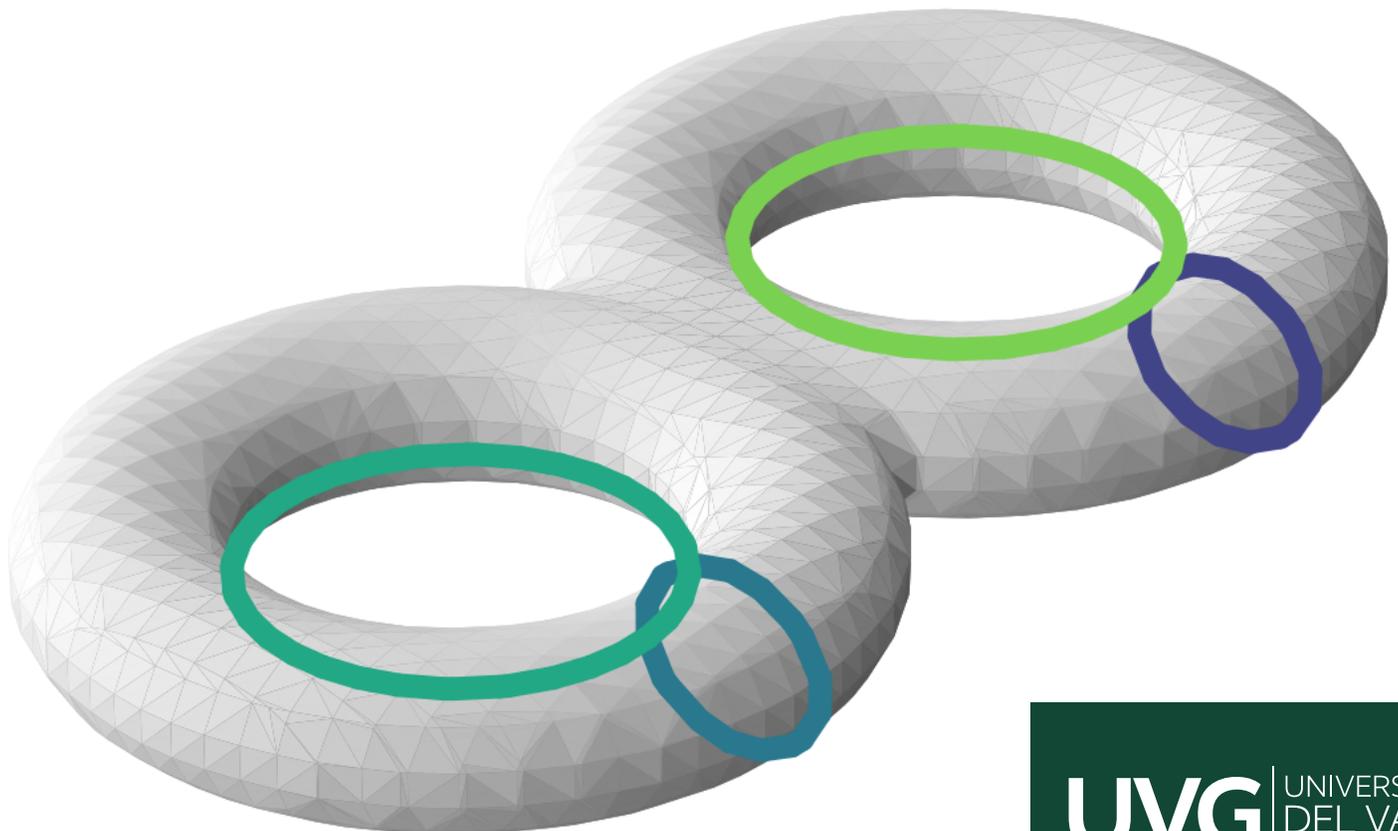

Algoritmos para la Identificación de Grupos de Homología en Complejos Simpliciales

Oscar Leonel Godoy Godoy



UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ciencias y Humanidades



Algoritmos para la Identificación de Grupos de Homología en Complejos Simpliciales

Trabajo de graduación en modalidad de tesis presentado por
Oscar Leonel Godoy Godoy
para optar al grado académico de Licenciado en Matemática Aplicada

Guatemala
2024

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ciencias y Humanidades



Algoritmos para la Identificación de Grupos de Homología en Complejos Simpliciales

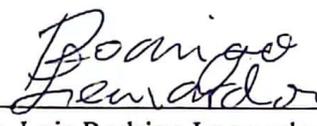
Trabajo de graduación en modalidad de tesis presentado por
Oscar Leonel Godoy Godoy
para optar al grado académico de Licenciado en Matemática Aplicada

Guatemala
2024

Vo. Bo. :

(f) 
MSc. Luis Rodrigo Leonardo Castellanos
Asesor

Tribunal Examinador:

(f) 
MSc. Luis Rodrigo Leonardo Castellanos
Asesor

(f) 
Lic. Dorval José Carias Samayoa

(f) 
Dr. Alan Gerardo Reyes Figueroa

Fecha de aprobación: Guatemala, 23 de febrero de 2024

Índice

Lista de figuras	VII
Lista de cuadros	IX
Resumen	X
1 Introducción	1
1.1 Objetivos	2
1.1.1 Objetivo general	2
1.1.2 Objetivos específicos	2
1.2 Justificación	3
1.3 Antecedentes	3
2 Marco teórico	5
2.1 Grupos libres	5
2.1.1 Construcción	5
2.1.2 Generadores y relaciones	6
2.2 Teoría de grafos	8
2.2.1 Grafos, caminos y ciclos	8
2.2.2 Árboles	10
2.3 Matroides y algoritmos voraces	12
3 El grupo fundamental	16
3.1 Homotopías	16
3.2 Construcción del grupo fundamental	18
3.3 Equivalencia homotópica	21
3.4 Cálculos	23
3.4.1 El círculo y el toro	23
3.4.2 El teorema de Van Kampen	24
3.4.3 La esfera, suma cuña y suma conexa	26
4 Triangulaciones y superficies	28
4.1 Complejos simpliciales	28
4.2 Cálculo de triangulaciones	30
4.2.1 Triangulación por parametrización	30
4.2.2 Cubos marchantes	31
5 Homología simplicial	34
5.1 Grupos de homología	34
5.2 Ejemplo de cálculo	37
5.3 Homología con coeficientes en \mathbb{Z}_2	39
5.4 Bases de homología óptimas	40
6 Algoritmos de identificación de grupos de homología	42

6.1	Estructuras de datos	42
6.2	Algoritmo de base de homología arbitraria	43
6.3	Algoritmo de base de homología óptima ingenua	44
6.3.1	Construcción del matroide	45
6.3.2	Ejecución del algoritmo voraz	47
6.4	Algoritmo de base de homología óptima con anotaciones	48
6.4.1	Anotación de aristas	48
6.4.2	Otra ejecución del algoritmo voraz	49
6.5	Complejidad de tiempo de ejecución	50
7	Resultados	52
7.1	Complejidad de tiempo de ejecución experimental	53
7.2	Cálculos de bases de homología arbitrarias	55
7.3	Cálculos de bases de homología óptimas	57
8	Conclusiones	61
9	Recomendaciones	62
10	Bibliografía	63
11	Anexos	65
11.1	Ejemplos de ejecución de algoritmos	65
11.1.1	Algoritmo HomArb	66
11.1.2	Algoritmo HomOpt1	66
11.1.3	Algoritmo HomOpt2	67

Lista de figuras

1.1	Lazos en un espacio topológico.	1
2.1	Subgrafo de expansión.	8
3.1	Homotopía de línea recta entre dos caminos en el disco unitario.	17
3.2	Reparametrizaciones entre lazos de X de la prueba del teorema 3.1.	19
3.3	Relaciones entre elementos del teorema de Van Kampen.	24
3.4	Cuadrículado del dominio $I \times I$ de H	26
3.5	Modelo del doble toro como un cociente.	27
4.1	Símplices de dimensiones 0, 1, 2 y 3, de izquierda a derecha.	29
4.2	Triangulación del dominio de una parametrización f de \mathbb{T}^2 y su imagen bajo f	31
4.3	Las 15 posibles intersecciones de una superficie con cada cubo [18].	32
4.4	Triangulaciones de un toro.	32
4.4a	Por parametrización.	32
4.4b	Por cubos marchantes.	32
5.1	Complejo simplicial K	37
5.2	Generadores de los grupos $Z_1(K)$ (azul) y $B_1(K)$ (rojo).	38
5.3	Tipos de ciclos en un complejo.	40
6.1	Árbol de distancia más corta en la triangulación de un doble toro.	45
6.2	Árbol de distancia más corta con raíz en v_0 en un grafo G_K	47
7.1	Tiempos de ejecución de algoritmos de cálculo de homología	53
7.2	Tiempos de ejecución de algoritmos de cálculo de homología (log-log).	54
7.3	Máxima cantidad de aristas en un ciclo construido por ConstruCiclos.	55
7.4	Bases de homología arbitrarias de un toro y un doble toro.	56
7.4a	Toro	56
7.4b	Doble toro	56
7.5	Más bases de homología arbitrarias.	56
7.5a	Taza de té de Utah	56
7.5b	Superficie de género 3.	56
7.6	Bases de homología óptimas.	57
7.6a	Cicloide de Dupin.	57
7.6b	Doble toro.	57
7.6c	Toro pinchado dos veces.	57
7.6d	Superficie de género 2.	57
7.6e	Superficie de género 3.	57
7.6f	Botella de Klein.	57
7.6g	Superficie P de Schwarz.	57
7.6h	Giroide de Schoen.	57
7.7	Bases de homología óptimas de objetos varios.	59
7.7a	Botijo.	59

7.7b	El pensador.	59
7.7c	Taza de té de Utah.	59
7.7d	3DBenchy.	59
7.7e	Componente mecánico.	59
7.7f	Discóbolo.	59
11.1	Complejo simplicial K	65
11.2	Árboles de distancia más corta del complejo K	67

Lista de cuadros

6.1	Complejidad de tiempos de ejecución de algoritmos de cálculo de homología.	51
7.1	Tiempos de ejecución medios de algoritmos de cálculo de homología.	53
7.1a	HomArb	53
7.1b	HomOpt1	53
7.1c	HomOpt2	53
7.2	Coefficientes de regresiones lineales.	54
7.3	Cantidad de generadores en los grupos de homología.	60

La presente tesis busca elaborar herramientas computacionales que favorezcan el entendimiento de la estructura homológica de complejos simpliciales, y en específico, de superficies triangulables. Esta estructura se estudia a través de los grupos de homología, invariantes topológicas que, en el estudio de superficies y otros espacios, comunican información geométrica de interés como la presencia de agujeros. En búsqueda de este fin, el escrito desarrolla las bases teóricas de la topología algebraica necesarias para dicha discusión, tomando como punto de partida el grupo fundamental, luego recorriendo conceptos básicos de homología y finalmente describiendo un problema de optimización respecto a las bases de homología. El conocimiento adquirido es luego traducido a algoritmos, los cuales son capaces de calcular computacionalmente los rangos y generadores de los grupos de homología de una superficie triangulada u otro complejo simplicial. Como evidencia visual, se tangibilizan las salidas de estos algoritmos en figuras, las cuales muestran los ciclos homológamente independientes de un diverso catálogo de superficies.

La topología algebraica puede resumirse como el estudio de espacios topológicos a través de estructuras algebraicas asociadas a los espacios. Las estructuras principales analizadas son el grupo fundamental y los grupos de homología. En conjunto, estas estructuras permiten distinguir entre espacios topológicos y, en el estudio de superficies y otros espacios particulares, comunican información geométrica de interés como la presencia de agujeros. Es más, es en el contexto de estas estructuras algebraicas que se matematiza el concepto de agujero de un espacio topológico. A grandes rasgos, desde la visión del grupo fundamental, un agujero se representa como la clase de equivalencia no nula de un lazo en un espacio topológico, bajo una relación que agrupa lazos que pueden deformarse continuamente el uno en el otro. En la Figura 1.1 mostramos un espacio X y dos lazos en X con clases de equivalencia distintas bajo esta relación.

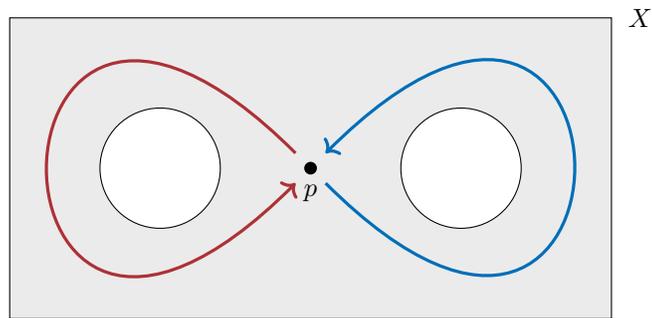


Figura 1.1: Lazos en un espacio topológico.

Por un lado, el grupo fundamental logra describir el concepto de agujero exitosamente para cualquier espacio topológico a un nivel teórico. No obstante, este cálculo no es trivial, no se encuentra sistematizado y resulta retador incluso para espacios relativamente simples. En contraste, la homología simplicial ofrece una alternativa semejante, con cálculos estructurados, aunque pierde un poco de generalidad en las definiciones. Estas estructuras se encuentran definidas solo para una colección restringida de espacios, denominada los espacios triangulables, definida en el lenguaje de los complejos simpliciales. A pesar de que los espacios triangulables son una minoría ante la gran familia de espacios topológicos generales, resulta que múltiples espacios de interés son triangulables.

En esta tesis, elaboramos herramientas computacionales que favorezcan el entendimiento de la estructura homológica de un complejo simplicial. Discutimos las bases teóricas de la topología algebraica, incluyendo el grupo fundamental y los grupos de homología. Luego, desarrollamos e implementamos algoritmos para la identificación de los grupos de homología en complejos simpliciales. En especial, tratamos un problema de optimización que busca identificar grupos de homología con ciclos de longitud mínima. Como resultado, contamos con tres algoritmos que identifican grupos de homología, implementados en el lenguaje de programación Python.

Estructuramos el contenido del documento de la siguiente forma. En el resto de este capítulo exponemos los objetivos, justificación y antecedentes de esta tesis. Luego, en el capítulo 2, discutimos la estructura de grupo libre, útil a lo largo del tratamiento de la topología algebraica. A su vez, introducimos lenguaje y resultados de la teoría de grafos, matroides y algoritmos voraces, todos esenciales para la discusión de los algoritmos. El capítulo 3 inicia el estudio de la topología algebraica con el grupo fundamental. Se listan sus propiedades relevantes, su invarianza entre espacios homeomorfos y se realizan cálculos de espacios conocidos.

En el capítulo 4 se redirige el estudio a la homología, primero definiendo objetos útiles como los complejos simpliciales. También se invierte un poco de tiempo en estudiar el algoritmo de cubos marchantes, el cual permite extraer un complejo simplicial asociado a una superficie implícita en \mathbb{R}^3 . Las definiciones y resultados básicos de la teoría de homología se muestran en el capítulo 5. Este es el último capítulo con propósito completamente teórico del texto, y es acá en donde se introduce el problema de optimización asociado a los ciclos en los grupos de homología.

Finalmente, los capítulos 6 y 7 desarrollan los algoritmos mencionados y presentan sus resultados. Analizamos las complejidades de tiempo de ejecución de los algoritmos y las verificamos experimentalmente. A su vez, ponemos a prueba los algoritmos con un diverso catálogo de superficies, generando una variedad de visualizaciones donde se identifican los grupos de homología. Para cada ejemplo, corroboramos la veracidad de las salidas con cálculos teóricos.

1.1. Objetivos

1.1.1. Objetivo general

Elaborar herramientas computacionales que favorezcan el entendimiento de la estructura homológica de un complejo simplicial.

1.1.2. Objetivos específicos

- Exponer las bases teóricas de la topología algebraica necesarias para la discusión del cálculo computacional de grupos de homología.
- Desarrollar e implementar algoritmos en un lenguaje de programación para la identificación de rangos y generadores de grupos de homología de complejos simpliciales.
- Generar visualizaciones de complejos simpliciales donde se identifiquen los ciclos en los grupos de homología calculados computacionalmente.

1.2. Justificación

La presente tesis ofrece una herramienta computacional para apoyar el estudio de una de las ramas centrales de la matemática moderna, que es la topología algebraica. En conjunto, la topología algebraica ha mostrado grandes aplicaciones teóricas desde su concepción en el siglo XX, incluyendo demostraciones de famosos teoremas como el teorema de punto fijo de Brouwer y el teorema de Borsuk-Ulam. Hoy en día la rama es fuente de publicaciones teóricas y aplicadas que generan valor dentro de la comunidad matemática.

Más allá de la teoría, los conceptos y algoritmos a discutir tienen aplicaciones en las ciencias de la computación y la ingeniería. Un ejemplo inmediato de estas se da en la generación de gráficos por computadora. Al momento de parametrizar una superficie, esta suele cortarse a través de un conjunto de ciclos para aplanarla, y resulta beneficioso que estos ciclos sean de longitud mínima, es decir, que pertenezcan a una base de homología óptima, para producir mejores renderizaciones [7]. El tópico de esta tesis puede considerarse dentro de la matemática aplicada denominada análisis topológico de datos, la cual observa usos en múltiples otras disciplinas, incluyendo química, biología y medicina. Para el lector interesado, “Database of Original & Non-Theoretical Uses of Topology”, abreviada como DONUT, es una base de datos presentada en la revista “Notices of the American Mathematical Society” en 2023 que busca compilar artículos de esta índole [12]. Al momento de redactar esta tesis, la base de datos cuenta con cerca de 500 documentos.

A nivel institucional y personal, contar con una herramienta personalizada para el cálculo y visualización de grupos de homología es de beneficio para el mejor entendimiento del tema. La homología guarda mucho potencial para generar visualizaciones interesantes, pero este se ve obstaculizado por la dificultad de los cálculos involucrados y el nivel de abstracción manejado. Expresar el estudio de este campo de forma visual permite tangibilizar un tópico usualmente abstracto y abre las oportunidades para las aplicaciones.

1.3. Antecedentes

El trabajo emplea como fundamento teórico principal la topología algebraica, una de las principales ramas de la matemática actual, introducida por Poincaré en 1895, y con grandes avances en el último siglo [1]. Naturalmente, para su estudio es necesario un entendimiento de topología general y álgebra abstracta, especialmente en teoría de grupos. Esta última requiere de temas adicionales no cubiertos en un curso tradicional del área, como grupos libres y representación de grupos con generadores y relaciones.

Una introducción a la topología algebraica inicia dentro de la rama de la homotopía, con la construcción del grupo fundamental. Luego, se exhiben sus propiedades básicas y su invariabilidad topológica. Es más, se discute que el grupo fundamental es invariante entre espacios homotópicamente equivalentes, una relación más general que el homeomorfismo. Se calculan algunos grupos fundamentales para exhibir su utilidad para categorizar espacios topológicos. No obstante, dada la complejidad de cálculo, el enfoque de la teoría se traslada a analizar otras estructuras algebraicas en espacios más restrictos y manejables. Los nuevos espacios de interés son los espacios triangulables, desarrollados a través de la estructura de los complejos simpliciales. Identificamos espacios comúnmente utilizados, como las superficies incrustables en \mathbb{R}^3 , dentro de esta categoría, y discutimos algoritmos que generen las triangulaciones.

Luego de estudiar los complejos simpliciales se describe en ellos la homología simplicial. Esto trae la ventaja de que pueden definirse cómodamente múltiples grupos de homología de distinta dimensionalidad. En particular, el primero de estos grupos de homología está altamente relacionado con el grupo fundamental. La literatura en estos temas es amplia y en gran parte es material estándar en un curso de posgrado del tópico [2, 13, 19].

El cálculo algorítmico de grupos de homología, en específico de aquellos con generadores óptimos, ha sido un tema popular de investigación en las últimas dos décadas. Las bases teóricas que sustentan los algoritmos de optimización presentados en este trabajo datan de 2005 y 2012, aunque las primeras ideas se remontan a 1987 en el contexto de la teoría de grafos [3, 10, 14]. Por otra parte, cuando solo se posee información sobre los puntos en un complejo y no de sus aristas y triángulos, se han producido algoritmos más sofisticados para aproximar soluciones [6]. Este y otros algoritmos similares atacan el problema desde una perspectiva distinta, llamada homología persistente.

En este capítulo, presentamos algunas nociones preliminares no son tan comunes de cubrir a un nivel de pregrado pero necesarias para estudiar topología algebraica y describir los algoritmos de los capítulos finales. En específico, desarrollamos el tópico de grupos libres dentro de la teoría de grupos, así como los elementos básicos de la teoría de grafos. Por otra parte, estudiamos la rama de la combinatoria denominada teoría de matroides. A lo largo del texto, asumimos que el lector tiene conocimiento de teoría de grupos, topología y álgebra lineal. Empleamos lenguaje y resultados estándar en estas áreas.

2.1. Grupos libres

El estudio de grupos libres suele considerarse un tema adicional de la teoría de grupos, y es frecuente que un texto en álgebra incluso omita su tratamiento. No obstante, la estructura de grupo libre es de gran utilidad para describir objetos de la topología algebraica, como se muestra más adelante en el capítulo 3. Resaltamos las construcciones y resultados necesarios para su discusión, y referimos al texto de Gallian para un tratamiento más a detalle [11].

2.1.1. Construcción

Dado un conjunto no vacío S , definimos un conjunto de símbolos $S^{-1} = \{x^{-1} : x \in S\}$ disjunto de S . A pesar de la notación empleada, por el momento no existe una relación algebraica entre elementos de S y S^{-1} . Una palabra de S es una cadena formal finita de la forma $x_1x_2 \cdots x_n$, donde $x_i \in S \cup S^{-1}$. Consideramos también como palabra de S a la cadena vacía, denotada por e . Llamamos $W(S)$ al conjunto de palabras de S . Con el ánimo de hacer de $W(S)$ un grupo, definimos en $W(S)$ la operación binaria de concatenación, dada por

$$(x_1x_2 \cdots x_n) \cdot (y_1y_2 \cdots y_n) = x_1x_2 \cdots x_ny_1y_2 \cdots y_n.$$

Notamos que la operación es asociativa y que e sirve de elemento neutro. No obstante, esta operación no permite la existencia de inversos en $W(S)$, pues la cadena vacía e nunca es la concatenación de cadenas no vacías. Dada la naturaleza simbólica de los elementos de S^{-1} , una cadena como xx^{-1} es distinta de e en $W(S)$. Para hacer identificaciones entre elementos como xx^{-1} y e en $W(S)$, introducimos una relación de equivalencia en $W(S)$.

Decimos que $u, v \in W(S)$ se encuentran relacionados bajo \sim si v puede obtenerse a partir de u vía una secuencia finita de eliminaciones o inserciones de palabras de la forma xx^{-1} y $x^{-1}x$, donde $x \in S$. Como un ejemplo, para $a, b \in S$ tenemos $ab^{-1}ba^{-1} \sim e$ en $W(S)$ gracias a la secuencia de eliminaciones $ab^{-1}ba^{-1} \rightarrow aa^{-1} \rightarrow e$. Mostramos que \sim es de equivalencia.

- Sean $u, v, w \in W(S)$. Si permitimos que dicha secuencia de eliminaciones o inserciones sea vacía, tenemos que \sim es reflexiva.
- Por otra parte, una secuencia de eliminaciones o inserciones que convierte a u en v puede leerse en sentido contrario, intercambiando eliminaciones por inserciones y viceversa, para obtener una secuencia que convierte a v en u . Por lo tanto, \sim es simétrica.
- Finalmente, una secuencia que convierte a u en v puede componerse con una secuencia que convierte a v en w para obtener una secuencia que convierte a u en w . Con esto, \sim es transitiva, y por lo tanto, de equivalencia.

Para cada $x \in W(S)$ denotamos por $[x] \in W/\sim$ a la clase de equivalencia de x en $W(S)$. Dotamos a $W(S)/\sim$ de la operación binaria inducida por la operación en $W(S)$, es decir

$$[u] \cdot [v] = [u \cdot v]$$

En efecto, la operación está bien definida. Si $[u_1] = [u_2]$ y $[v_1] = [v_2]$, la secuencia que convierte a u_1 en u_2 puede emplearse para convertir a $u_1 \cdot v_1$ en $u_2 \cdot v_1$, con lo cual $[u_1 \cdot v_1] = [u_2 \cdot v_1]$. Del mismo modo, la secuencia que convierte a v_1 en v_2 puede emplearse para convertir a $u_2 \cdot v_1$ en $u_2 \cdot v_2$, implicando $[u_2 \cdot v_1] = [u_2 \cdot v_2]$. Por la transitividad de \sim , tenemos $[u_1 \cdot v_1] = [u_2 \cdot v_2]$.

Por otra parte, vemos que la operación en $W(S)/\sim$ hereda su asociatividad de la operación en $W(S)$. Además, $[e]$ sirve como el nuevo elemento neutro. Finalmente, notamos que todo elemento $[u] \in W(S)/\sim$ posee como elemento inverso a $[u^{-1}]$, pues tenemos $[u][u^{-1}] = [u \cdot u^{-1}] = [e]$ y $[u^{-1}][u] = [u^{-1} \cdot u] = [e]$. Por lo tanto, $W(S)/\sim$ forma un grupo.

Definición 2.1. *El grupo $W(S)/\sim$ con la operación anterior es llamado el grupo libre generado por S , y es denotado por $F(S)$.*

En efecto, vemos que S es un conjunto de generadores de $F(S)$. El grupo libre generado por un conjunto S unitario es isomorfo al grupo cíclico infinito \mathbb{Z} . Cuando S posee dos o más elementos, $F(S)$ es un grupo no abeliano. Si $s_1, s_2 \in S$ son generadores distintos, las palabras s_1s_2 y s_2s_1 en S no están relacionadas bajo \sim . Entonces, es de interés definir al grupo libre abeliano generado por S . Como lo indica su nombre, este se obtiene de abelianizar a $F(S)$, efectuando el cociente $F(S)/N$, donde N es el subgrupo conmutador de $F(S)$.

2.1.2. Generadores y relaciones

Uno de los beneficios de discutir grupos libres es que estos ofrecen una alternativa de representación para cualquier otro grupo. Esto se respalda en el siguiente resultado

Teorema 2.1. *Todo grupo es isomorfo al grupo cociente de un grupo libre.*

Demostración. Sea G un grupo arbitrario y S un conjunto de generadores de G . Consideramos la función $\phi : F(S) \rightarrow G$ que envía a $[x_1x_2 \cdots x_n] \in F(S)$ al producto en G dado por $x_1x_2 \cdots x_n$. Para diferenciar entre los productos de $W(S)$ y G , empleamos la notación $(x_1 \cdots x_n)_W$ y $(x_1 \cdots x_n)_G$. Con esta notación, ϕ está dado por $[(x_1x_2 \cdots x_n)_W] \mapsto (x_1x_2 \cdots x_n)_G$. Vemos que ϕ está bien definida, pues la

inserción o eliminación de palabras de la forma $(xx^{-1})_W$ en $W(S)$ corresponde a simplificar productos de G . De la asociatividad en G se sigue que ϕ es un homomorfismo, pues para todos $(x_1 \cdots x_n)_W$ y $(y_1 \cdots y_m)_W$ elementos de $W(S)$ tenemos $(x_1 \cdots x_n y_1 \cdots y_m)_G = (x_1 \cdots x_n)_G (y_1 \cdots y_m)_G$.

Vemos que ϕ es sobreyectiva. Por lo tanto, por el teorema fundamental de homomorfismos, G es isomorfo al cociente $F(S)/\ker \phi$. \square

Una forma usual de representar un grupo es mediante las relaciones que sus elementos satisfacen. Por ejemplo, el grupo dihedro D_n suele definirse como el grupo con $2n$ elementos dado por dos generadores a, b y las relaciones $a^2 = b^n = e$ y $ab = b^{-1}a$. Sin embargo, no es claro cómo estas relaciones definen un grupo, o la unicidad del mismo. Formalmente, usamos grupos libres para definir la presentación de un grupo mediante generadores y relaciones.

Definición 2.2. Sea G un grupo, $S = \{s_1, \dots, s_n\}$ un conjunto generador de G y $F(S)$ el grupo libre generado por S . Además, sea $R = \{r_1, \dots, r_m\}$ un subconjunto de $F(S)$ y N_R el subgrupo normal generado por R . Decimos que G está dado por los generadores s_1, s_2, \dots, s_n y las relaciones $r_1 = r_2 = \dots = r_m = e$ si existe un isomorfismo entre $F(S)/N_R$ y G tal que $N_R s_i \mapsto s_i$. Denotamos esto por $G = \langle S \mid R \rangle$.

Si G está dado por los generadores s_1, s_2, \dots, s_n y las relaciones $r_1 = r_2 = \dots = r_m = e$, también escribimos

$$G = \langle s_1, \dots, s_n \mid r_1 = \dots = r_m = e \rangle.$$

Con esta notación, puede mostrarse que en efecto tenemos $D_n = \langle a, b \mid a^2 = b^n = (ab)^2 = e \rangle$. Notamos que un grupo libre sobre $S = \{s_1, \dots, s_n\}$ no posee relaciones no triviales, con lo cual tenemos $F(S) \simeq \langle s_1, \dots, s_n \rangle$.

Si tenemos que un grupo está dado por generadores S y relaciones R , adicionar relaciones en R produce un grupo con una estructura más simple a la del grupo original.

Teorema 2.2. Sea G un grupo dado por generadores en S y relaciones en R . Si G' es otro grupo dado por generadores en S y relaciones en R' , con $R \subset R'$, entonces G' es imagen homomorfa de G .

Demostración. Sean N y N' los subgrupos normales en $F(S)$ generados por R y R' , respectivamente. Luego, tenemos $G \simeq F(S)/N$ y $G' \simeq F(S)/N'$. Definimos el homomorfismo $\phi : F(S)/N \rightarrow F(S)/N'$ dado por $\phi(Ng) = N'g$. En efecto, ϕ es un homomorfismo, y como $N \subset N'$, ϕ es sobreyectivo. Por lo tanto, G' es imagen homomorfa de G . \square

La presentación de un grupo en generadores y relaciones es útil para definir homomorfismos entre grupos. En particular, nos permite extender una función cualquiera definida sobre los generadores de un grupo a un homomorfismo, siempre que tal función respete las relaciones en el dominio.

Teorema 2.3. Sean $G = \langle S \mid R \rangle$ y G' dos grupos, y $\phi : S \rightarrow G'$ una función cualquiera. Además, sea $\bar{\phi} : F(S) \rightarrow G'$ la extensión de ϕ a $F(S)$. Entonces, $\bar{\phi}$ puede extenderse a un homomorfismo de G a G' si $\bar{\phi}(r)$ es la identidad en G' para todo $r \in R$.

Demostración. Primero, vemos que $\bar{\phi}$ posee una única extensión homomorfa $\bar{\phi} : F(S) \rightarrow G'$ al grupo libre en S , dada por $\bar{\phi}(s_1 \cdots s_n) = \bar{\phi}(s_1) \cdots \bar{\phi}(s_n)$. Sea N el subgrupo normal en $F(S)$ generado por R . Como $\bar{\phi}(r)$ es la identidad en G' para todo $r \in R$, se sigue $R \subset \ker \bar{\phi}$. Luego, $N \subset \ker \bar{\phi}$. Entonces, $\bar{\phi}$ induce un homomorfismo canónico $\phi : F(S)/N \rightarrow G'$, donde abusamos notación y reciclamos el símbolo ϕ . Dado que $G \simeq F(S)/N$, este homomorfismo es una extensión de la función ϕ original. \square

La representación de grupos también facilita la descripción de una nueva operación entre grupos, llamada el producto libre. Dados dos grupos G, H con representaciones $G = \langle S \mid R \rangle$ y $H = \langle S' \mid R' \rangle$, definimos el grupo $G * H$ por la representación $G * H = \langle S \cup S' \mid R \cup R' \rangle$.

2.2. Teoría de grafos

Introducimos lenguaje de la teoría de grafos, útil para desarrollar los algoritmos del capítulo 6. Tratamos los conceptos iniciales de caminos, ciclos y árboles. Para un tratamiento más amplio, referenciamos el libro de texto de Diestel [8].

2.2.1. Grafos, caminos y ciclos

Definición 2.3. Un grafo es un par ordenado $G = (V, E)$ de conjuntos donde E es una colección de subconjuntos de dos elementos de V . Decimos V es el conjunto de vértices de G , mientras que E es el conjunto de aristas de G . Nos restringimos a grafos donde $0 < |V| < \infty$.

Si G es un grafo, escribimos $V(G)$ y $E(G)$ para referirnos a los conjuntos de vértices y aristas de G , respectivamente. Además, usualmente escribimos uv para representar una arista $\{u, v\} \in E(G)$. Por otra parte, definimos el orden de un grafo G como la cantidad de vértices en $V(G)$, y lo denotamos por $|G|$.

Definición 2.4. Dos vértices u, v de G son adyacentes si uv es una arista de G . Además, un vértice v de G es incidente a una arista e de G si $v \in e$. El grado de un vértice v en G es la cantidad de vértices de G adyacentes a él.

En ocasiones queremos discutir relaciones entre grafos como la contención, o deseamos combinar grafos para obtener un grafo más grande que contenga todas las aristas y vértices de sus grafos constituyentes. Para ello, introducimos el siguiente lenguaje.

Definición 2.5. Sean $G = (V, E)$ y $G' = (V', E')$ dos grafos. La unión de G y G' es el grafo dado por $G \cup G' = (V \cup V', E \cup E')$. Por otra parte, si $V' \subset V$ y $E' \subset E$, decimos que G' es subgrafo de G o que G' está contenido en G .

Si $G' = (V', E')$ es subgrafo de $G = (V, E)$, pero $V' \neq V$ o $E' \neq E$, decimos que G' es subgrafo propio de G .

Cuando hablamos de unión o contención de grafos estamos empleando la convención de evitar el significado literal de la contención o unión de la teoría de conjuntos. Tratamos a los grafos como pares ordenados (V, E) , y solo es de nuestro interés hablar de las operaciones y relaciones entre ellos a través de sus componentes.

Definición 2.6. Si G' es un subgrafo de G , decimos que G' es de expansión si $V(G') = V(G)$.

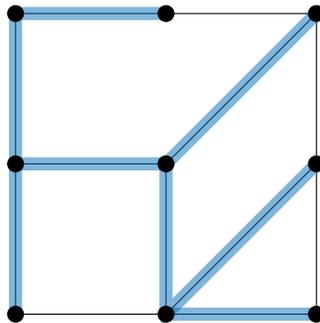


Figura 2.1: Subgrafo de expansión.

La Figura 2.1 muestra un subgrafo de expansión, resaltado en azul, de un grafo. Aunque el subgrafo de expansión cuenta con 8 aristas mientras el grafo original cuenta con 12, ambos grafos tienen 9 vértices.

Ahora, introducimos los conceptos de caminos y ciclos en un grafo.

Definición 2.7. Dado un grafo G , un camino en G es una lista ordenada (v_0, v_1, \dots, v_k) no vacía de vértices de G tal que $v_i v_{i+1}$ es una arista de G para todo $i = 0, \dots, k - 1$.

Representamos el camino (v_0, v_1, \dots, v_k) por $v_0 v_1 \cdots v_k$, y decimos que v_0 es el vértice de inicio y v_k el vértice de fin del camino. Cuando $k > 0$, decimos que el camino $v_0 v_1 \cdots v_k$ es “de v_0 a v_k ”. Notamos que un camino $v_0 v_1 \cdots v_k$ define un conjunto de vértices $\{v_i\}$ y un conjunto de aristas $\{v_i v_{i+1}\}$ que produce un subgrafo de G . Es en este sentido que decimos que un camino en G está contenido en G . Usualmente nos referimos indistintamente a los caminos de G y sus subgrafos de G asociados. Como los caminos $v_0 v_1 \cdots v_k$ y $v_k \cdots v_1 v_0$ definen un mismo subgrafo asociado, tampoco hacemos distinción entre estos caminos, y decimos que $v_0 v_1 \cdots v_k$ es de v_0 a v_k , o bien, de v_k a v_0 .

Decimos que un camino es *cerrado* si los vértices de inicio y fin coinciden. Así, un camino cerrado es de la forma $v_0 v_1 \cdots v_k v_0$. Similar a la discusión anterior, vemos que el camino cerrado $v_0 v_1 \cdots v_k v_0$ produce el mismo subgrafo asociado que $v_1 v_2 \cdots v_k v_0 v_1$. Por lo tanto, tampoco distinguimos entre estos caminos cerrados.

Definición 2.8. Dado un grafo G , un ciclo en G es un camino cerrado en G que no repite vértices, a excepción del los vértices de inicio y fin.

Decimos que un camino $v_0 v_1 \cdots v_k$ es *simple* si todos sus vértices son distintos entre sí. Notamos que, si para $u, v \in V(G)$ existe un camino en G de u a v , puede recorrerse este camino iniciando en u y removiendo vértices repetidos hasta llegar a v para construir un camino simple en G de u a v .

Con una noción de caminos en un grafo que unen pares de vértices, discutimos el tema de la conexidad de un grafo.

Definición 2.9. Un grafo G es *conexo* si, para todo par de vértices u, v de G , existe un camino en G de u a v .

Definición 2.10. Una *componente conexa* de un grafo G es un subgrafo conexo G' de G que es *maximal*, es decir, donde G' no es subgrafo propio de ningún subgrafo conexo de G .

Por la propiedad anterior, alternativamente podemos definir un grafo conexo como un grafo donde, para cada par de vértices u, v de G , existe un camino simple en G de u a v , en lugar de pedir la existencia solo de un camino general.

Ahora, discutimos uno de los resultados más clásicos de la teoría de grafos, referente a la existencia de cierto tipo de camino en un grafo, uno que visite todas las aristas de un grafo exactamente una vez. El teorema caracteriza cuáles grafos poseen un camino con esta propiedad, y su discusión se remonta a 1735, cuando Euler ofreció su solución al problema de los puentes de Königsberg.

Definición 2.11. Dado un grafo G , un camino en G es *euleriano* si recorre todas las aristas de G exactamente una vez.

Teorema 2.4. Un grafo conexo posee un camino cerrado euleriano si, y solo si, todos sus vértices tienen grado par.

Demostración. Sea G un grafo conexo que posee un camino cerrado euleriano $w = v_0 v_1 \cdots v_k v_0$. Notamos que el grafo asociado a w es precisamente G . Luego, el grado de un $v_i \neq v_0$ en G es igual al doble del número de veces que v_i aparece en $v_0 v_1 \cdots v_k v_0$, pues v_i es adyacente a v_{i-1} y v_{i+1} . Se

verifica que el grado de v_0 en G es el doble del número de veces que v_0 aparece en $v_0v_1 \cdots v_k$. En cualquier caso, el grado de los vértices de G es siempre par.

Ahora, sea G un grafo conexo, donde todos sus vértices tienen grado par. Sea $w = v_0v_1 \cdots v_k$ un camino en G que no recorre una misma arista dos veces y que maximiza la cantidad de aristas que visita. Vemos que todas las aristas incidentes a v_k están en w por la maximalidad de w , con lo cual el grado de v_k en w es igual al grado de v_k en G . Cada aparición de v_0 en $v_1 \cdots v_{k-1}$ aporta dos aristas al grado de v_0 . Se sigue que $v_0 = v_k$, pues de otro modo v_0 tendría grado impar.

Supóngase por contradicción que w no es euleriano. Luego, existe una arista de G que no está en w . Dada la conexidad de G , escogemos una arista $uv_i \in E(G) - E(w)$ que sea incidente a un vértice de w . Entonces, el camino $w' = uv_iv_{i+1} \cdots v_{k-1}v_0v_1 \cdots v_{i-1}$ es un camino en G que no recorre una misma arista dos veces, pero que visita más aristas que w , una contradicción. Por lo tanto, w sí es euleriano. \square

Corolario 2.1. *Un grafo conexo que solo posee vértices de grado par puede descomponerse como la unión de ciclos que no comparten aristas a pares.*

Demostración. Por el teorema anterior, dicho grafo contiene un camino cerrado euleriano $w = v_0v_1 \cdots v_kv_0$. Para extraer ciclos de este camino cerrado, replicamos la técnica de extracción de un camino simple de un camino arbitrario. Recorremos la lista w iniciando en v_0 y removemos vértices repetidos, así como los caminos cerrados entre ellos en w . Notamos que estos caminos cerrados definen ciclos. Tras recorrer la totalidad de w , tenemos un listado de ciclos que no comparten aristas y cuya unión es w . \square

2.2.2. Árboles

Definición 2.12. *Un árbol es un grafo conexo que no contiene ciclos.*

Una primera propiedad de los árboles es la relación entre su número de vértices y aristas. Como veremos, esta relación incluso caracteriza a los árboles dentro de los grafos conexos. Un resultado previo para esta prueba es la existencia de vértices de grado 1 en un árbol, los cuales llamamos *hojas*.

Si T es un árbol con dos vértices o más, consideramos un camino simple $w = v_0v_1 \cdots v_k$ en T que maximice la cantidad de aristas que visita. Luego, todas las aristas incidentes a v_k en T están en w . Dado que T no contiene ciclos, v_k no es incidente a ningún otro vértice de w excepto v_{k-1} . Por lo tanto, v_k tiene grado 1 en w y en T , por lo que v_k es una hoja de T . Un argumento análogo muestra que v_0 también es una hoja de T .

Teorema 2.5. *Un grafo T es un árbol si, y solo si, T es conexo y satisface $|V(T)| = |E(T)| + 1$.*

Demostración. Sea T un árbol. Conocemos que es T es conexo, así que probamos que satisface $|V(T)| = |E(T)| + 1$. Procedemos por inducción en $|V(T)|$. Si T posee un único vértice, vemos que T no posee aristas, y la propiedad se cumple.

Ahora, sea T un árbol con más de un vértice, y supóngase por inducción que todo árbol con menos de $|V(T)|$ vértices cumple la propiedad. Sea v_0 una hoja de T , y considérese el subgrafo T' de T obtenido de eliminar a v_0 y a la única arista incidente a v_0 en T . Entonces, tenemos

$$|V(T')| = |V(T)| - 1 \quad \text{y} \quad |E(T')| = |E(T)| - 1$$

Notamos que T' también es un árbol, con lo cual $|V(T')| = |E(T')| + 1$. Se sigue de inmediato que $|V(T)| = |E(T)| + 1$.

Por otra parte, supóngase que T es un grafo conexo tal que $|V(T)| = |E(T)| + 1$. Por contradicción, supóngase que T no es un árbol, con lo cual T posee al menos un ciclo. Eliminamos aristas de los ciclos de T hasta obtener un subgrafo T' de T sin ciclos, que preserva la conexidad de T . Luego, T' es un árbol. Sin embargo, tenemos $|V(T')| = |V(T)|$ y $|E(T')| < |E(T)|$, con lo cual $|V(T')| < |E(T')| + 1$, una contradicción. Por lo tanto, T sí es un árbol. \square

Los árboles son grafos ricos en propiedades, de las cuales resaltamos dos en el siguiente teorema. Más que propiedades, estas son en realidad caracterizaciones de árboles.

Teorema 2.6. *Para un grafo T , los siguientes enunciados son equivalentes:*

- I. T es un árbol.
- II. Para cada par de vértices u, v de T , existe un único camino simple en T de u a v .
- III. T no contiene ciclos, pero para cada par de vértices u, v de T no adyacentes, el grafo $T \cup \{uv\}$, obtenido de añadir la arista uv a T sí contiene ciclos.

Demostración. Primero, mostramos (I) implica (II). Sea T un árbol, y sean u, v vértices de T . Como T es conexo, existe un camino simple w de u a v en T . Si w' fuera otro camino de u a v , tendríamos un ciclo en T dado por la unión de w y w' , lo cual no es posible en el árbol T . Por lo tanto, w es único.

Ahora, mostramos (II) implica (III). Sea T un grafo donde existe un único camino simple en T entre cada par de vértices, denotado por $T[u, v]$. Dado que estos caminos son únicos, T no posee ciclos. Además, si u, v son vértices de T no adyacentes, el grafo $T \cup \{uv\}$ contiene al ciclo obtenido de agregar la arista uv al camino de u a v en T .

Finalmente, vemos (III) implica (I). Sea T un grafo que cumple (III). De entrada tenemos que T no posee ciclos. Sean u, v vértices de T . Si $uv \in E(T)$, uv es un camino de u a v en T . Por otro lado, si $uv \notin E(T)$, $T \cup \{uv\}$ contiene un ciclo que emplea la arista uv . El resto de este ciclo resulta en un camino de u a v en T . En cualquier caso, vemos que podemos unir u y v con un camino en T , con lo cual T es conexo. Por lo tanto, T es un árbol. \square

Para nuestros fines nos interesa la equivalencia entre las propiedades (1) y (3). Esto permite construir un árbol de expansión en un grafo conexo arbitrario a través de un grafo sin ciclos maximal. En la elaboración de algoritmos de teoría de grafos es de fundamental relevancia la existencia de estos árboles.

Corolario 2.2. *Todo grafo conexo contiene un árbol de expansión.*

Demostración. Sea G un grafo conexo, y consideremos un subgrafo T de G sin ciclos que maximice la cantidad de aristas que contiene. Del teorema anterior, T es un árbol de G . Dada la maximalidad de T , este debe contener cada vértice de G . Luego, T es el árbol de expansión de G buscado. \square

La Figura 2.1 evidencia un árbol de expansión de un grafo. Ahora que garantizamos la existencia de árboles de expansión de grafos conexos, nos interesan dos tipos de estos árboles, cada uno de los cuales minimiza una propiedad de interés. Para ello, introducimos el concepto de grafo ponderado, que permite asociar un real positivo a cada arista de un grafo, o bien, una noción de distancia entre vértices adyacentes.

Definición 2.13. *Un grafo ponderado es un grafo G junto con una función asociada $\ell : E(G) \rightarrow \mathbb{R}^+$ de valores reales positivos, la cual llamamos función longitud. Para todo subgrafo G' de G , extendemos ℓ a G' vía*

$$\ell(G') = \sum_{e \in E(G')} \ell(e).$$

Con esta función longitud consideramos dos árboles de expansión notables, denominados los árboles de expansión mínimo y de distancia más corta respecto a un vértice. Ambos de estos árboles serán utilizados para los algoritmos del capítulo 6.

Definición 2.14. *Dado un grafo conexo G ponderado por ℓ , un árbol de expansión mínimo T de G es un árbol de expansión de G tal que $\ell(T)$ es mínimo.*

Definición 2.15. *Sea G un grafo conexo ponderado por ℓ y v un vértice de G . Un árbol de distancia más corta de G con raíz en v es un árbol de expansión T tal que, para todo $v' \in V(G)$, el camino único de v a v' en T es de longitud mínima en G .*

Por el corolario 2.2 y por la finitud de nuestros grafos, está garantizada la existencia de un árbol de expansión mínimo en un grafo conexo. A su vez, la definición 2.15 indica un método de construcción de un árbol de expansión en un grafo conexo, el cual resulta conexo y sin ciclos por la condición de minimalidad impuesta. Existen múltiples algoritmos clásicos para identificar estos árboles. Un método para obtener un árbol de expansión mínimo es el algoritmo de Kruskal [15]. Por otra parte, para encontrar árboles de distancia más corta contamos con el algoritmo de Dijkstra [9].

2.3. Matroides y algoritmos voraces

Para múltiples problemas de optimización, existe una estrategia para encontrar soluciones denominada el algoritmo voraz. Los matroides son una estructura que busca generalizar las situaciones donde este algoritmo produce una solución óptima. En este apartado, definimos esta estructura y demostramos un teorema que garantiza que el algoritmo voraz en realidad produce soluciones óptimas en matroides. Seguimos el esquema presentado por el libro de texto en algoritmos de Cormen [5].

En el capítulo 6 identificamos un problema de optimización que posee una estructura de matroide ponderado. Para su solución, citamos el resultado al final de esta sección.

Definición 2.16. *Dado un conjunto finito S y una colección \mathcal{I} no vacía de subconjuntos de S , un matroide es un par ordenado $M = (S, \mathcal{I})$ que cumple las siguientes propiedades:*

- I. *Si $A \in \mathcal{I}$ y $B \subset A$, entonces $B \in \mathcal{I}$. (propiedad de herencia)*
- II. *Si $A, B \in \mathcal{I}$ y $|A| < |B|$, entonces existe un $s \in B - A$ tal que $A \cup \{s\} \in \mathcal{I}$. (propiedad de intercambio)*

Los elementos de \mathcal{I} son llamados subconjuntos independientes de S . Notamos que $\emptyset \in \mathcal{I}$, gracias a que \mathcal{I} es un conjunto no vacío que posee la propiedad de herencia. Los matroides generalizan la noción de independencia lineal del álgebra lineal a otras estructuras que no son espacios vectoriales. Vemos que, si S es un subconjunto finito en un espacio vectorial e \mathcal{I} es la colección de subconjuntos de S cuyos elementos son linealmente independientes, (S, \mathcal{I}) forma un matroide.

Definición 2.17. *Dado un matroide $M = (S, \mathcal{I})$ y $A \in \mathcal{I}$, decimos que $s \in S$ es una extensión de A si $A \cup \{s\} \in \mathcal{I}$. Además, $A \in \mathcal{I}$ es maximal si A no tiene extensiones.*

Lema 2.1. *Todos los subconjuntos maximales de un matroide $M = (S, \mathcal{I})$ poseen la misma cardinalidad.*

Demostración. Por contradicción, supóngase que $A, B \in \mathcal{I}$ son subconjuntos maximales tales que $|A| < |B|$. Por la propiedad de intercambio de \mathcal{I} , existe un $s \in B - A$ tal que $A \cup \{s\} \in \mathcal{I}$. No obstante, esto contradice la maximalidad de A . \square

Definición 2.18. *Sea $M = (S, \mathcal{I})$ un matroide. Decimos que M es ponderado si M tiene asociado una función $w : S \rightarrow \mathbb{R}^+$ de valores reales positivos, la cual llamamos la función peso de M . Además, extendemos a w para todo $A \in \mathcal{I}$ vía el peso de sus elementos, es decir,*

$$w(A) = \sum_{a \in A} w(a).$$

En un matroide ponderado $M = (S, \mathcal{I})$ con función peso w pudieramos discutir una noción de maximalidad respecto a w . Esto es porque w induce un preorden en \mathcal{I} dado por $A \leq B \Leftrightarrow w(A) \leq w(B)$, donde por preorden nos referimos a una relación reflexiva y transitiva en \mathcal{I} . Un elemento maximal en este preorden es un subconjunto independiente que maximiza a $w : \mathcal{I} \rightarrow \mathbb{R}^+$. Decimos que un subconjunto independiente es óptimo en M si es maximal respecto a este preorden inducido por w . Reservamos el calificativo maximal como es descrito en la definición 2.17.

Contrastamos las nociones de subconjunto maximal y óptimo. Por un lado, vemos que todo subconjunto óptimo debe ser maximal. En efecto, si A no es maximal, entonces existe una extensión $s \in S$ de A , lo cual genera un $A \cup \{s\} \in \mathcal{I}$ tal que

$$w(A \cup \{s\}) = w(A) + w(s) > w(A).$$

por lo que $A \cup \{s\} > A$ y A no es óptimo. Por otro lado, no todo subconjunto maximal debe ser óptimo. Si $S = \{1, 2\}$, $\mathcal{I} = \{\emptyset, \{1\}, \{2\}\}$ y w es la identidad en S , se verifica que (S, \mathcal{I}) es un matroide ponderado donde $\{1\}$ y $\{2\}$ son maximales pero $\{1\}$ no es óptimo.

De interés para nosotros es resolver el problema de optimización de identificar subconjuntos óptimos en un matroide ponderado. Para ello consideramos el algoritmo siguiente, denominado el algoritmo voraz en el matroide ponderado $M = (S, \mathcal{I})$ con función peso w . Este algoritmo propone un candidato a solución $A \in \mathcal{I}$, el cual se construye de forma iterativa, iniciando con $A_0 = \emptyset$. Para esta iteración, enumeramos los elementos de S según el orden impuesto por w de forma decreciente, de modo que obtenemos una lista ordenada $s_1, s_2, \dots, s_n \in S$ donde $w(s_1) \geq w(s_2) \geq \dots \geq w(s_n)$. Recorremos la lista de izquierda a derecha, y para cada $s_i \in S$, definimos

$$A_i = \begin{cases} A_{i-1} \cup \{s_i\}, & A_{i-1} \cup \{s_i\} \in \mathcal{I}, \\ A_{i-1}, & \text{en otro caso.} \end{cases}$$

Al finalizar la iteración, la salida del algoritmo es $A = A_n$. La lista A_0, A_1, \dots, A_n es anidada, y cada A_i difiere de A_{i-1} a lo máximo por un elemento. Interpretamos cada A_i en esta lista como una actualización de A_{i-1} , a medida que el algoritmo traversa y escoge elementos de s_1, \dots, s_n para formar parte de A . Al escribir pseudocódigo de este algoritmo, evitamos almacenar cada A_i , y en su lugar, actualizamos un mismo A inicial, como se ve en el algoritmo 1.

Dado que iniciamos con un subconjunto independiente A_0 , la definición de A_i garantiza que cada elemento de la lista A_1, \dots, A_n es también un subconjunto independiente. En particular, la salida del algoritmo voraz también es un subconjunto independiente. Como w asume valores reales positivos y $A_{i-1} \subset A_i$, se sigue $w(A_{i-1}) < w(A_i)$. Luego, el algoritmo mejora el candidato a solución A_i en cada iteración, en búsqueda de un subconjunto óptimo A . El algoritmo recibe el calificativo voraz porque, en cada actualización, este elige al mejor candidato inmediato, sin planear una elección de elementos de A a futuro. En lo que queda de nuestra discusión de matroides, mostramos que el algoritmo voraz en efecto produce un subconjunto óptimo.

Algoritmo 1 Voraz

Entrada: Matroide ponderado $M = (S, \mathcal{I})$ con función peso w .

Salida: Subconjunto óptimo A de M .

- 1: $A \leftarrow \emptyset$.
 - 2: Crear lista ordenada $s_1, \dots, s_n \in S$, donde $w(s_1) \geq \dots \geq w(s_n)$.
 - 3: **para cada** s_i **hacer**
 - 4: **si** $A \cup \{s_i\} \in \mathcal{I}$ **entonces**
 - 5: $A \leftarrow A \cup \{s_i\}$
 - 6: **retornar** A
-

Lema 2.2. Sea $M = (S, \mathcal{I})$ un matroide ponderado con función peso w . Además, sea s_1, \dots, s_n la lista ordenada de elementos de S según el orden de iteración del algoritmo voraz, y s el primer elemento de esta lista que escoge el algoritmo, en caso escoja alguno. Entonces, existe un conjunto óptimo A de M tal que $s \in A$.

Demostración. Sea B un subconjunto óptimo. Si $s \in B$, el lema se cumple. Por lo tanto, supóngase que $s \notin B$. Construimos un subconjunto independiente A a partir de B tal que $s \in A$. Iniciamos con $A = \{s\}$ y, de forma iterativa, agregamos elementos de B a A según la propiedad de intercambio de M . Agregamos elementos hasta que $|A| = |B|$, en cuyo caso A y B son conjuntos independientes que difieren solo por los elementos s y t , donde $B - A = \{t\}$. Vemos que, al cumplirse $t \in B$, por la propiedad de herencia de \mathcal{I} tenemos $B \in \mathcal{I} \implies \{t\} \in \mathcal{I}$, con lo cual t aparece luego de s en s_1, \dots, s_n y $w(s) \geq w(t)$. Con esto, tenemos

$$w(A) = w(B \cup \{s\} - \{t\}) = w(B) + w(s) - w(t) \geq w(B).$$

Como B es óptimo, entonces A también es óptimo, como se deseaba. \square

El anterior resultado evidencia que, después de que el algoritmo voraz escoge un elemento s , hay esperanza de que el algoritmo produzca un subconjunto óptimo, siempre que el resto de elementos se escojan adecuadamente. El lema a continuación permita decir que este es el caso.

Lema 2.3. Sean $M = (S, \mathcal{I})$ un matroide ponderado con función peso w y s el primer elemento de S que el algoritmo voraz escoge, en caso escoja alguno. Definimos un matroide ponderado $M' = (S', \mathcal{I}')$, llamado la *contracción de M por s* , dado por

$$\begin{aligned} S' &= \{s' \in S : \{s, s'\} \in \mathcal{I}\}, \\ \mathcal{I}' &= \{A' \in \mathcal{I} : A' \cup \{s\} \in \mathcal{I}, s \notin A'\} \end{aligned}$$

cuya función peso es la restricción de w a S' . Luego, un subconjunto A' es óptimo en M' si, y solo si, $A' \cup \{s\}$ es óptimo en M .

Demostración. Primero, comprobamos que $M' = (S', \mathcal{I}')$ es un matroide.

Sean $A' \in \mathcal{I}'$ y $B' \subset A'$. Luego, $B' \cup \{s\} \subset A' \cup \{s\}$, donde $A' \cup \{s\} \in \mathcal{I}$. Por la propiedad de herencia en \mathcal{I} , se sigue $B' \cup \{s\} \in \mathcal{I}$. En vista de que $s \notin B'$, tenemos $B' \in \mathcal{I}'$, con lo cual \mathcal{I}' también goza la propiedad de herencia. Por otra parte, sean $A', B' \in \mathcal{I}'$ tales que $|A'| < |B'|$. En particular tenemos $A', B' \in \mathcal{I}$, así que, por lo que la propiedad de intercambio en \mathcal{I} , existe un $t \in B' - A'$ tal que $A' \cup \{t\} \in \mathcal{I}$. Puesto que $s \notin A'$, tenemos $A' \cup \{t\} \in \mathcal{I}'$. Entonces, \mathcal{I}' también posee la propiedad de intercambio.

Ahora, sea $A' \in \mathcal{I}'$. Mostramos los contrapositivos de las implicaciones del lema. Si A' no es óptimo en M' , entonces existe un $B' \in \mathcal{I}'$ tal que $w(B') > w(A')$. Luego, $A' \cup \{s\}$ no es óptimo en M , pues $B' \cup \{s\} \in \mathcal{I}$ cumple

$$w(B' \cup \{s\}) = w(B') + w(s) > w(A') + w(s) = w(A' \cup \{s\}).$$

Por otra lado, si $A' \cup \{s\}$ no es óptimo en M , existe un $B \in \mathcal{I}$ tal que $w(B) > w(A' \cup \{s\})$. De acuerdo con el lema 2.2, escogemos dicho B tal que $s \in B$. Consideramos a $B' = B - \{s\}$, donde $B' \in \mathcal{I}'$. Se sigue que A' no es óptimo en M' , pues tenemos

$$w(B') = w(B) - w(s) > w(A' \cup \{s\}) - w(s) = w(A').$$

□

Finalizamos presentando el teorema anticipado, que muestra que el algoritmo voraz resuelve nuestro problema de optimización.

Teorema 2.7. *El algoritmo voraz en un matroide $M = (S, \mathcal{I})$ con función peso w produce un subconjunto óptimo.*

Demostración. Procedemos por inducción en el número de elementos de un subconjunto óptimo de M . Este es en efecto constante en M , pues cada subconjunto óptimo es maximal, y por el lema 2.1, cada subconjunto maximal comparte la misma cardinalidad. Llamamos A a la salida del algoritmo voraz en M . Vemos que, si el algoritmo no escoge ningún elemento de S , entonces M no posee subconjuntos independientes y $A = \emptyset$. Por lo tanto, suponemos que el algoritmo escoge al menos un elemento de S , y llamamos s al primero de estos.

Si los subconjuntos óptimos de M poseen un único elemento, entonces $A = \{s\}$. Por definición, $\{s\}$ es un conjunto independiente que maximiza a w entre los subconjuntos unitarios. Por lo tanto, A es óptimo, y el algoritmo encuentra subconjuntos óptimos de un elemento.

Por hipótesis de inducción, supóngase que el algoritmo voraz encuentra exitosamente subconjuntos óptimos de $n \in \mathbb{Z}^+$ elementos, y que M posee subconjuntos óptimos de $n + 1$ elementos. Cuando el algoritmo escoge a su primer elemento $s \in S$, vemos que $A - \{s\}$ es igual a la salida A' del algoritmo voraz en la contracción de M por s , dada por $M' = (S', \mathcal{I}')$. Esto se debe a que, para un subconjunto B' que no contiene a s , tenemos $B' \in \mathcal{I}'$ si, y solo si, $B' \cup \{s\} \in \mathcal{I}$. Por hipótesis de inducción, A' es óptimo en M' . Finalmente, por el lema 2.3, $A = A' \cup \{s\}$ es óptimo en M . Por lo tanto, el algoritmo voraz encuentra subconjuntos óptimos de $n + 1$ elementos.

En conclusión, la salida A del algoritmo voraz en M es un subconjunto óptimo. □

El grupo fundamental

En este capítulo introducimos el primer objeto de la topología algebraica, el denominado grupo fundamental. Como anticipamos en la introducción de esta tesis, el grupo fundamental es una estructura algebraica asociada a los espacios topológicos. Estos grupos permiten distinguir entre espacios y, en el estudio de superficies y otros espacios particulares, comunican información geométrica de interés como la presencia de agujeros.

Iniciamos con un tratamiento general de las homotopías en la primera sección, para proceder con la construcción del grupo en la siguiente. Luego, en la tercera sección, mostramos la invarianza de esta estructura entre espacios homotópicamente equivalentes, una relación entre espacios topológicos más general al homeomorfismo. Finalmente, calculamos los grupos fundamentales de un catálogo de espacios, sirviéndonos de teoremas célebres de la teoría como el teorema de Van Kampen.

El desarrollo de este capítulo es de carácter teórico, es estándar de un primer curso de posgrado en el tema y puede encontrarse a mayor detalle en otros textos [2, 13, 20]. Resumimos acá los enunciados más relevantes de la teoría para nuestra discusión. En la sección de resultados de esta tesis, que tiene un enfoque más aplicado, calificamos la veracidad de las salidas de los algoritmos producidos con la teoría generada en este capítulo.

3.1. Homotopías

Sea X un espacio topológico. Un camino en X es un mapeo continuo $\alpha : I \rightarrow X$, donde $I = [0, 1]$. Decimos que un camino α en X tiene inicio en $\alpha(0)$ y fin en $\alpha(1)$. Si α, β con caminos en X con inicio en x_0 y fin en x_1 , buscamos describir una deformación continua de α a β . Definimos esto en un contexto más general, vía las homotopías.

Definición 3.1. Sean $f, g : X \rightarrow Y$ mapeos continuos. Decimos que f es homotópico a g si existe una mapeo continuo $H : X \times I \rightarrow Y$ tal que $H(x, 0) = f(x)$ y $H(x, 1) = g(x)$ para todo $x \in X$. Llamamos a H una homotopía de f a g . Denotamos que f es homotópico a g por $f \simeq g$.

Un ejemplo útil de una homotopía es el siguiente. Sea X un subconjunto convexo de \mathbb{R}^n y sean

$\alpha, \beta : I \rightarrow X$ dos caminos en X . Luego, podemos definir un mapeo continuo $H : I \times I \rightarrow X$ por $H(s, t) = (1-t)\alpha(s) + t\beta(s)$. Para un $s \in I$ fijo, $H(s, t)$ une con un segmento de recta a $\alpha(s)$ y $\beta(s)$. Por la convexidad de X , $H(s, t)$ siempre se encuentra en X y es una homotopía por caminos de α a β . Por lo tanto, cualquier par de caminos en X son homotópicos entre sí. A H se le conoce como la homotopía por caminos de línea recta. Un ejemplo de esta homotopía entre dos caminos en el disco unitario se observa en la Figura 3.1.

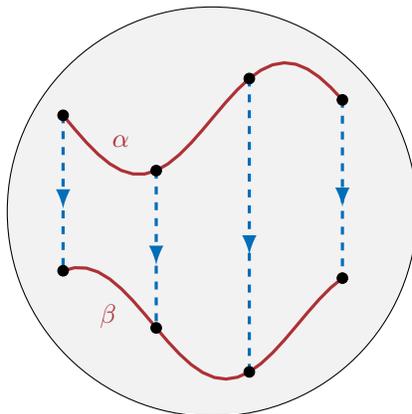


Figura 3.1: Homotopía de línea recta entre dos caminos en el disco unitario.

Ahora, sean α, β caminos en X con inicio en x_0 y fin en x_1 , homotópicos vía H . Fijamos un $0 \leq t \leq 1$ y consideramos la restricción de H a $X \times \{t\}$, lo que genera un mapeo continuo $h_t(x) = H(x, t)$. Pensamos en h_t como un camino intermedio en la deformación continua de α a β . Vemos que h_t es un camino en X , aunque no necesariamente tiene inicio x_0 y fin x_1 . Para forzar esta propiedad en los h_t , extendemos la definición 3.1.

Definición 3.2. Sean $\alpha, \beta : I \rightarrow X$ dos caminos en X con inicio en x_0 y fin en x_1 . Decimos que α es homotópico por caminos a β si existe una homotopía $H : I \times I \rightarrow X$ de α a β tal que $H(0, t) = x_0$ y $H(1, t) = x_1$ para todo $0 \leq t \leq 1$. Llamamos a H una homotopía por caminos de α a β . Denotamos que α es homotópico por caminos a β por $\alpha \simeq_p \beta$.

Nos interesa mostrar que las relaciones introducidas en las definiciones 3.1 y 3.2 son de equivalencia. Previo a esto, empleamos un lema de la topología que facilita mostrar que las funciones con las que lidiamos son en realidad continuas.

Lema 3.1 (del pegado). Sean X, Y espacios topológicos, $A, B \subset X$ y supóngase que A y B son cerrados de $A \cup B$. Además, sean $f : A \rightarrow Y$ y $g : B \rightarrow Y$ mapeos que concuerdan en $A \cap B$. Definimos a $f \cup g : A \cup B \rightarrow Y$ por $(f \cup g)(a) = f(a)$ si $a \in A$, o bien, $(f \cup g)(b) = g(b)$ si $b \in B$. Luego, si f y g son continuos, entonces $f \cup g$ también es continuo.

Demostración. Dotamos a A, B y $A \cup B$ de las topologías de subespacio. Sea U un cerrado de Y . Por la continuidad de f , $f^{-1}(U)$ es cerrado en A . Dado que A es cerrado en $A \cup B$, $f^{-1}(U)$ también es cerrado en $A \cup B$. De forma análoga, $g^{-1}(U)$ es cerrado en $A \cup B$. Ahora, como $(f \cup g)^{-1}(U) = f^{-1}(U) \cup g^{-1}(U)$, tenemos que $(f \cup g)^{-1}(U)$ es cerrado en $A \cup B$. Por lo tanto, $f \cup g$ es continuo. \square

Lema 3.2. Las relaciones de homotopía y homotopía por caminos son de equivalencia.

Demostración. Mostramos que \simeq es de equivalencia. Sean f, g, h mapeos continuos de X en Y . Vemos que $f \simeq f$ vía la homotopía $F(x, t) = f(x)$, con lo cual \simeq es reflexiva. Por otra parte, si

$f \simeq g$ vía $F(x, t)$, entonces $g \simeq f$ vía $G(x, t) = F(x, 1 - t)$, por lo que \simeq es simétrica. Finalmente, si $f \simeq g$ vía $F(x, t)$ y $g \simeq h$ vía $G(x, t)$, construimos a

$$H(x, t) = \begin{cases} F(x, 2t), & t \in [0, \frac{1}{2}], \\ G(x, 2t - 1), & t \in [\frac{1}{2}, 1]. \end{cases}$$

donde $H(x, 0) = f(x)$, $H(x, \frac{1}{2}) = g(x)$ y $H(x, 1) = h(x)$. Vemos que $F(x, 2t)$ y $G(x, 2t - 1)$, restringidos a $X \times [0, \frac{1}{2}]$ y $X \times [\frac{1}{2}, 1]$ respectivamente, son mapeos continuos que concuerdan a $g(x)$ en la intersección $X \times \{\frac{1}{2}\}$ de sus dominios. Luego, por el lema del pegado, H es continuo, con lo cual $f \simeq h$. Entonces, \simeq es transitiva, y por lo tanto, de equivalencia.

Se prueba que \simeq_p es de equivalencia utilizando las mismas construcciones. Cuando los mapeos continuos f , g y h son caminos con inicio y fin comunes, las homotopías F , G y H descritas arriba son homotopías por caminos. \square

Cerramos esta sección introductoria con un lema que describe como se comportan las homotopías con los mapeos continuos.

Lema 3.3. *Las relaciones de homotopía y homotopía por caminos se preservan bajo la composición de mapeos continuos.*

Demostración. Mostramos el enunciado para la homotopía. Sean $f, g : X \rightarrow Y$ mapeos continuos tales que $f \simeq g$ vía F . Además, sean $h : W \rightarrow X$ y $h' : Y \rightarrow Z$ otros mapeos continuos. Tenemos el siguiente diagrama.

$$\begin{array}{ccccc} W & \xrightarrow{h} & X & \begin{array}{c} \xrightarrow{f} \\ \xrightarrow{g} \end{array} & Y & \xrightarrow{h'} & Z \end{array}$$

Vemos que, tanto $f \circ h, g \circ h : W \rightarrow Y$ como $h' \circ f, h' \circ g : X \rightarrow Z$, son homotópicos a pares. Por un lado tenemos $f \circ h \simeq g \circ h$ vía G , donde $G(w, t) = F(h(w), t)$. Por otro, se sigue $h' \circ f \simeq h' \circ g$ vía G' , donde $G'(x, t) = h(F(x, t))$.

De nuevo, cuando los mapeos continuos f y g son caminos con inicio común y fin común, las homotopías G y G' son homotopías por caminos. \square

3.2. Construcción del grupo fundamental

Sea X un espacio topológico. Un lazo en X es un camino α en X tal que $\alpha(0) = \alpha(1)$. Se dice que un lazo tiene punto base en $\alpha(0)$. Para un $x_0 \in X$, definimos un producto en el conjunto de lazos en X con punto base en x_0 , dado por

$$(\alpha \cdot \beta)(s) = \begin{cases} \alpha(2s), & s \in [0, \frac{1}{2}], \\ \beta(2s - 1), & s \in [\frac{1}{2}, 1]. \end{cases} \quad (3.1)$$

Vemos que $\alpha(2s)$ y $\beta(2s - 1)$, restringidos a $[0, \frac{1}{2}]$ y $[\frac{1}{2}, 1]$ respectivamente, son mapeos continuos que concuerdan a x_0 en $t = \frac{1}{2}$, la intersección de sus dominios. Por el lema del pegado, $\alpha \cdot \beta$ es continuo y en efecto un lazo en X . Podríamos definir el producto anterior para lazos en general, pero vemos que el requisito de punto base común es esencial para garantizar la continuidad del producto. Notamos que el lazo $\alpha \cdot \beta$ recorre primero la imagen de α y luego la de β , al doble de “velocidad” cada tramo.

El producto de lazos no define un grupo en el conjunto de lazos en X con un punto base común. Si α, β, γ son lazos en X con punto base común, las expresiones para $(\alpha \cdot \beta) \cdot \gamma$ y $\alpha \cdot (\beta \cdot \gamma)$ son

$$(\alpha \cdot \beta) \cdot \gamma = \begin{cases} \alpha(4s), & s \in [0, \frac{1}{4}], \\ \beta(4s - 1), & s \in [\frac{1}{4}, \frac{1}{2}], \\ \gamma(2s - 1), & s \in [\frac{1}{2}, 1]. \end{cases} \quad \text{y} \quad \alpha \cdot (\beta \cdot \gamma) = \begin{cases} \alpha(2s), & s \in [0, \frac{1}{2}], \\ \beta(4s - 2), & s \in [\frac{1}{2}, \frac{3}{4}], \\ \gamma(4s - 3), & s \in [\frac{3}{4}, 1]. \end{cases} \quad (3.2)$$

Aunque similares, estos lazos no son iguales como funciones. No obstante, sus clases de equivalencia bajo la relación de homotopía por caminos sí coinciden. Llamamos al conjunto de estas clases $\pi_1(X, x_0)$, y es sobre este conjunto que construimos el llamado grupo fundamental de X con punto base en x_0 .

Teorema 3.1. *El conjunto $\pi_1(X, p)$ de todas las clases de equivalencia de lazos en X con punto base en x_0 bajo la relación de homotopía por caminos forma un grupo con el producto $[\alpha] \cdot [\beta] = [\alpha \cdot \beta]$.*

Demostración. Primero, vemos que este producto se encuentra bien definido. Si $\alpha \simeq_p \alpha'$ vía F y $\beta \simeq_p \beta'$ vía G , tenemos que $\alpha \cdot \beta \simeq_p \alpha' \cdot \beta'$ vía H , donde H es la homotopía por caminos dada por

$$H(s, t) = \begin{cases} F(2s, t), & s \in [0, \frac{1}{2}], \\ G(2s - 1, t), & s \in [\frac{1}{2}, 1]. \end{cases}$$

Al igual que otros mapeos anteriores, se garantiza la continuidad de H por el lema del pegado.

• *Asociatividad:* Para mostrar que el producto en $\pi_1(X, x_0)$ es asociativo, argumentamos que los lazos $(\alpha \cdot \beta) \cdot \gamma$ y $\alpha \cdot (\beta \cdot \gamma)$, vistos en la expresión 3.2, son homotópicos por caminos. Para ello, primero buscamos un mapeo $u : I \rightarrow I$ que reparametrice los lazos, de tal forma que $((\alpha \cdot \beta) \cdot \gamma) \circ u = \alpha \cdot (\beta \cdot \gamma)$.

El producto $\alpha \cdot (\beta \cdot \gamma)$ produce la partición $[0, \frac{1}{2}] \cup [\frac{1}{2}, \frac{3}{4}] \cup [\frac{3}{4}, 1]$ de I dada por los dominios de α , β y γ en el producto. Similarmente, $(\alpha \cdot \beta) \cdot \gamma$ produce la partición $[0, \frac{1}{4}] \cup [\frac{1}{4}, \frac{1}{2}] \cup [\frac{1}{2}, 1]$ de I . Construimos un mapeo u que envíe el i -ésimo intervalo de la partición de I producida por $\alpha \cdot (\beta \cdot \gamma)$ al i -ésimo intervalo de la partición de I producida por $(\alpha \cdot \beta) \cdot \gamma$. Este mapeo u está dado por la primera gráfica de la Figura 3.2. Se verifica que $((\alpha \cdot \beta) \cdot \gamma) \circ u = \alpha \cdot (\beta \cdot \gamma)$.

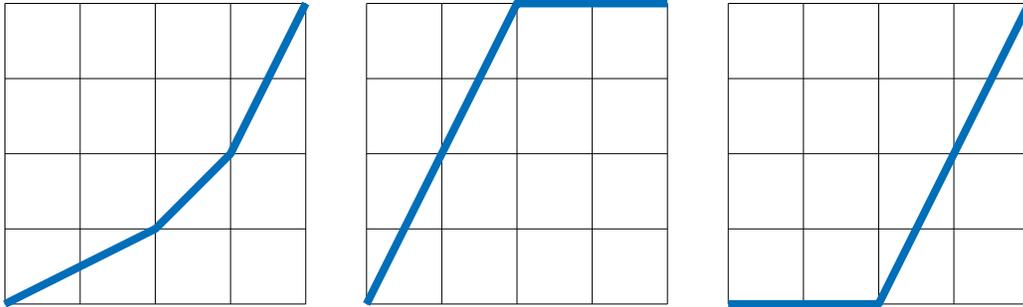


Figura 3.2: Reparametrizaciones entre lazos de X de la prueba del teorema 3.1.

Notamos que u y el mapeo identidad $\mathbb{1}_I$ en I son caminos en I con inicio en 0 y fin en 1. Dado que I es convexo, por la homotopía por caminos de línea recta, tenemos $u \simeq_p \mathbb{1}_I$. Luego, por el lema 3.3, se sigue $((\alpha \cdot \beta) \cdot \gamma) \circ u \simeq_p ((\alpha \cdot \beta) \cdot \gamma) \circ \mathbb{1}_I$. En resumen, el producto de clases de lazos es asociativo, pues tenemos

$$\alpha \cdot (\beta \cdot \gamma) = ((\alpha \cdot \beta) \cdot \gamma) \circ u \simeq_p ((\alpha \cdot \beta) \cdot \gamma) \circ \mathbb{1}_I = (\alpha \cdot \beta) \cdot \gamma.$$

• *Elemento neutro:* Por otra parte, proponemos como elemento neutro en $\pi_1(X, x_0)$ a la clase del lazo constante $e : I \rightarrow X$, dado por $e(s) = x_0$ para todo $0 \leq s \leq 1$. Si α es un lazo en X con punto

base en x_0 , mostramos que $\alpha \cdot e \simeq_p \alpha$. De nuevo, creamos un mapeo $u : I \rightarrow I$ que reparametrice los lazos de tal forma que $\alpha \cdot e = \alpha \circ u$. La segunda gráfica de la Figura 3.2 muestra tal mapeo. Como en la discusión anterior, tenemos $u \simeq_p \mathbb{1}_I$, como caminos en I con inicio en 0 y fin en 1. Por lo tanto,

$$\alpha \cdot e = \alpha \circ u \simeq_p \alpha \circ \mathbb{1}_I = \alpha$$

De forma análoga se comprueba que $e \cdot \alpha \simeq_p \alpha$, utilizando la reparametrización vista en la tercera gráfica de la Figura 3.2. Por lo tanto $[e]$ es un elemento neutro en $\pi_1(X, x_0)$.

• *Inversos:* Finalmente, para todo lazo α en X con punto base en x_0 proponemos como inverso de $[\alpha]$ en $\pi_1(X, p)$ a $[\alpha^{-1}]$, donde $\alpha^{-1}(s) = \alpha(1 - s)$. Con esta notación, vemos que $\alpha^{-1} = \alpha \circ \mathbb{1}_I^{-1}$, donde pensamos en $\mathbb{1}_I$ como un camino en I con inicio en 0 y fin en 1. Este inverso está bien definido, pues si $\alpha \simeq_p \beta$ vía F entonces $\alpha^{-1} \simeq_p \beta^{-1}$ vía $F(1 - s, t)$. Mostramos que $\alpha \cdot \alpha^{-1} \simeq_p e$. Primero, observamos que $\mathbb{1}_I \cdot \mathbb{1}_I^{-1}$ es un lazo en I con punto base en 0, al igual que el lazo constante de I en 0. Llamemos por el momento a este lazo constante $\bar{0}$. Por la homotopía por caminos de línea recta, se sigue $\mathbb{1}_I \cdot \mathbb{1}_I^{-1} \simeq_p \bar{0}$. Por el lema 3.3, tenemos

$$\alpha \cdot \alpha^{-1} = (\alpha \circ \mathbb{1}_I) \cdot (\alpha \circ \mathbb{1}_I^{-1}) = \alpha \circ (\mathbb{1}_I \cdot \mathbb{1}_I^{-1}) \simeq_p \alpha \circ \bar{0} = e.$$

De manera análoga se muestra que $\alpha^{-1} \cdot \alpha \simeq_p e$. Entonces, el inverso de $[\alpha]$ en $\pi_1(X, x_0)$ es $[\alpha^{-1}]$. En conclusión, $\pi_1(X, x_0)$ cumple con ser un grupo. \square

El grupo fundamental de un espacio topológico es una propiedad determinada por el espacio y un punto base particular. Sin embargo, en el caso donde X es conexo por caminos, podemos identificar los grupos fundamentales de X con puntos base distintos por un isomorfismo de grupos.

Teorema 3.2. *Si X es conexo por caminos, existe un isomorfismo entre $\pi_1(X, x_0)$ y $\pi_1(X, x_1)$ para todo $x_0, x_1 \in X$.*

Demostración. Antes de proceder, generalizamos el producto de lazos en X a caminos cuyos puntos de inicio y fin sean compatibles. Si α y β son caminos en X con $\alpha(1) = \beta(0)$, definimos $\alpha \cdot \beta$ de igual manera que en la expresión 3.1 para el caso de lazos. El lema del pegado garantiza que $\alpha \cdot \beta$ es un camino en X de $\alpha(0)$ a $\beta(1)$ gracias a que $\alpha(1) = \beta(0)$. En contraste con los lazos, los caminos en X no producen una estructura tan dotada de propiedades como el grupo fundamental. Sin embargo, podemos extender algunas propiedades vistas en la prueba del teorema 3.1 a los caminos de X .

Si α, β, γ son caminos en X tales que $\alpha(1) = \beta(0)$ y $\beta(1) = \gamma(0)$, es decir, con inicios y fines compatibles, se sigue $(\alpha \cdot \beta) \cdot \gamma \simeq_p \alpha \cdot (\beta \cdot \gamma)$. Además, para todo camino α en X tenemos $e_0 \cdot \alpha \simeq_p \alpha$ y $\alpha \cdot e_1 \simeq_p \alpha$, donde e_0 y e_1 son los lazos constantes de X en $\alpha(0)$ y $\alpha(1)$. Por último, para todo camino α en X también definimos a α^{-1} como $\alpha^{-1}(s) = \alpha(1 - s)$. Vemos que si α es un camino de x_0 a x_1 , entonces α^{-1} es un camino de x_1 a x_0 . Análogo a lo visto en el teorema 3.1, se sigue que $\alpha \cdot \alpha^{-1} \simeq_p e_0$ y $\alpha^{-1} \cdot \alpha \simeq_p e_1$.

Ahora, sean $x_0, x_1 \in X$. Luego, existe un camino γ en X con inicio x_0 y fin x_1 . Definimos una función $\gamma_* : \pi_1(X, x_0) \rightarrow \pi_1(X, x_1)$ dada por

$$\gamma_*([\alpha]) = [\gamma^{-1} \cdot \alpha \cdot \gamma].$$

Se entiende por contexto si los corchetes simbolizan elementos en $\pi_1(X, x_0)$ o $\pi_1(X, x_1)$. Como en la construcción del grupo fundamental, se verifica que γ_* está bien definida. Por otra parte, si $[\alpha_1], [\alpha_2] \in \pi_1(X, x_0)$, de las propiedades anteriores sobre el producto de caminos se sigue que

$$\begin{aligned} \gamma^{-1} \cdot (\alpha_1 \cdot \alpha_2) \cdot \gamma &\simeq_p (\gamma^{-1} \cdot \alpha_1) \cdot (\alpha_2 \cdot \gamma) \\ &\simeq_p (\gamma^{-1} \cdot \alpha_1) \cdot (\gamma \cdot \gamma^{-1}) \cdot (\alpha_2 \cdot \gamma) \\ &\simeq_p (\gamma^{-1} \cdot \alpha_1 \cdot \gamma) \cdot (\gamma^{-1} \cdot \alpha_2 \cdot \gamma). \end{aligned}$$

Por lo tanto, $\gamma_*([\alpha_1] \cdot [\alpha_2]) = \gamma_*([\alpha_1]) \cdot \gamma_*([\alpha_2])$ y γ_* es un homomorfismo. Finalmente, consideramos a $(\gamma^{-1})_* : \pi_1(X, x_1) \rightarrow \pi_1(X, x_0)$, dada por $(\gamma^{-1})_*([\alpha]) = [\gamma \cdot \alpha \cdot \gamma^{-1}]$ para todo $[\alpha] \in \pi_1(X, x_1)$. De manera análoga, se verifica que $(\gamma^{-1})_*$ también es un homomorfismo. Es más, $(\gamma^{-1})_*$ es función inversa de γ_* , pues para todo $[\alpha] \in \pi_1(X, x_0)$ tenemos

$$\gamma \cdot (\gamma^{-1} \cdot \alpha \cdot \gamma) \cdot \gamma^{-1} \simeq_p (\gamma \cdot \gamma^{-1}) \cdot \alpha \cdot (\gamma \cdot \gamma^{-1}) \simeq_p \alpha.$$

De igual forma, γ_* es un inverso para $(\gamma^{-1})_*$, con lo cual ambos homomorfismos son isomorfismos. En particular, $\pi_1(X, x_0)$ es isomorfo a $\pi_1(X, x_1)$. \square

Con este resultado, en espacios topológicos conexos por caminos podemos referirnos a un único grupo fundamental $\pi_1(X)$, independiente del punto base.

Finalmente, discutimos la idea de heredar el grupo fundamental de un espacio a otro vía un mapeo continuo. Sean $h : X \rightarrow Y$ un mapeo continuo, $x_0 \in X$ y $y_0 = h(x_0) \in Y$. Definimos a $h_* : \pi_1(X, x_0) \rightarrow \pi_1(Y, y_0)$ por $h_*([\alpha]) = [h \circ \alpha]$. De la propiedad $h \circ (\alpha \cdot \beta) = (h \circ \alpha) \cdot (h \circ \beta)$ vemos que h_* es un homomorfismo. Decimos que h_* es el homomorfismo inducido por el mapeo continuo h . Ahora, cuando $h : X \rightarrow Y$ es un homeomorfismo, tenemos un resultado más fuerte, pues h_* resulta ser un isomorfismo.

Antes de continuar con la prueba de este hecho, vemos una breve propiedad. Sean $f : X \rightarrow Y$ y $g : Y \rightarrow Z$ mapeos continuos, con $x_0 \in X$, $y_0 = f(x_0) \in Y$ y $z_0 = g(y_0) \in Z$. Entonces, la función $g_* \circ f_*$ ilustrada por el diagrama

$$\pi_1(X, x_0) \xrightarrow{f_*} \pi_1(Y, y_0) \xrightarrow{g_*} \pi_1(Z, z_0)$$

está dada por $(g_* \circ f_*)([\alpha]) = [g \circ (f \circ \alpha)]$. Gracias a la asociatividad de la composición de funciones, el homomorfismo $(g \circ f)_*$ dado por $(g \circ f)_*([\alpha]) = [(g \circ f) \circ \alpha]$ es idéntico a $g_* \circ f_*$.

Teorema 3.3. *Sean X y Y espacios topológicos y $h : X \rightarrow Y$ un homeomorfismo. Para $x_0 \in X$ y $y_0 = h(x_0) \in Y$, los grupos fundamentales $\pi_1(X, x_0)$ y $\pi_1(Y, y_0)$ son isomorfos.*

Demostración. Conocemos que h_* es un homomorfismo, así que mostramos que h_* es biyectivo. De la propiedad anterior conocemos

$$h_* \circ (h^{-1})_* = (h \circ h^{-1})_* = (\mathbb{1}_X)_* \quad \text{y} \quad (h^{-1})_* \circ h_* = (h^{-1} \circ h)_* = (\mathbb{1}_Y)_*.$$

Dado que $(\mathbb{1}_X)_*$ y $(\mathbb{1}_Y)_*$ son los isomorfismos identidad sobre $\pi_1(X, x_0)$ y $\pi_1(Y, y_0)$, h_* es invertible. Por lo tanto, h_* es un isomorfismo entre $\pi_1(X, x_0)$ y $\pi_1(Y, y_0)$. \square

Si X y Y son espacios homeomorfos conexos por caminos, el resultado en el teorema 3.3 puede independizarse de los puntos base de los grupos fundamentales gracias al teorema 3.2. Por lo tanto, tenemos el siguiente resultado.

Teorema 3.4. *El grupo fundamental, salvo isomorfismo, es un invariante topológico entre espacios conexos por caminos.*

3.3. Equivalencia homotópica

Buscamos expandir la colección de espacios que comparten un mismo grupo fundamental. Por el teorema 3.4, sabemos que espacios homeomorfos y conexos por caminos poseen grupos fundamentales isomorfos. Consideramos una relación entre espacios más general que el homeomorfismo, llamada la equivalencia homotópica.

Definición 3.3. Sean X y Y espacios topológicos. Decimos que X es homotópicamente equivalente a Y si existen mapeos continuos $f : X \rightarrow Y$ y $g : Y \rightarrow X$ tales que $g \circ f \simeq \mathbb{1}_X$ y $f \circ g \simeq \mathbb{1}_Y$. Llamamos a f y g equivalencias homotópicas, inversas entre sí. Denotamos que X es homotópicamente equivalente a Y por $X \simeq Y$.

Se sigue que si $h : X \rightarrow Y$ es un homeomorfismo, entonces h^{-1} es un mapeo continuo tal que $h^{-1} \circ h = \mathbb{1}_X$ y $h \circ h^{-1} = \mathbb{1}_Y$, con lo cual h es una equivalencia homotópica de X a Y . Al igual que la relación de homeomorfismo, vemos que \simeq también es de equivalencia.

Lema 3.4. La relación de equivalencia homotópica en la colección de espacios topológicos es de equivalencia.

Demostración. Para todo espacio X , vemos que el mapeo identidad en X es una equivalencia homotópica de X en X , con lo cual \simeq es reflexiva. De la definición, se sigue que \simeq es simétrica. Ahora, si $X \simeq Y$ y $Y \simeq Z$, tenemos las equivalencias homotópicas del siguiente diagrama

$$\begin{array}{ccccc} X & \xrightarrow{f} & Y & \xrightarrow{f'} & Z \\ & \searrow g & & \swarrow g' & \\ & & X & & \end{array}$$

Notamos que $f' \circ f : X \rightarrow Z$ es una equivalencia homotópica de X a Z con inversa $g \circ g' : Z \rightarrow X$. Vemos que, gracias al lema 3.3, estos mapeos cumplen con

$$\begin{aligned} (g \circ g') \circ (f' \circ f) &= g \circ (g' \circ f') \circ f \simeq g \circ \mathbb{1}_Y \circ f = g \circ f \simeq \mathbb{1}_X, \\ (f' \circ f) \circ (g \circ g') &= f' \circ (f \circ g) \circ g' \simeq f' \circ \mathbb{1}_Y \circ g' = f' \circ g' \simeq \mathbb{1}_Z. \end{aligned}$$

Por lo tanto, $X \simeq Z$ y \simeq es transitiva. En conclusión la equivalencia homotópica es una relación de equivalencia. \square

En paralelo con la prueba del teorema 3.3, donde un homeomorfismo $h : X \rightarrow Y$ inducía un isomorfismo $h_* : \pi_1(X, x_0) \rightarrow \pi_1(Y, h(x_0))$, ahora producimos un isomorfismo de grupos vía una equivalencia homotópica. Como vemos abajo, la estrategia de demostración es similar.

Teorema 3.5. Sean X y Y espacios topológicos y $f : X \rightarrow Y$ una equivalencia homotópica. Para $x_0 \in X$ y $y_0 = f(x_0) \in Y$, los grupos fundamentales $\pi_1(X, x_0)$ y $\pi_1(Y, y_0)$ son isomorfos.

Demostración. Dado que $X \simeq Y$, tenemos mapeos continuos $f : X \rightarrow Y$ y $g : Y \rightarrow X$ tales que $g \circ f \simeq \mathbb{1}_X$ y $f \circ g \simeq \mathbb{1}_Y$. Buscamos mostrar que el homomorfismo $f_* : \pi_1(X, x_0) \rightarrow \pi_1(Y, y_0)$ inducido por f , dado por $f_*([\alpha]) = [f \circ \alpha]$, es un isomorfismo.

Sea $F : X \times I \rightarrow X$ la homotopía de $\mathbb{1}_X$ a $g \circ f$, y sea γ el camino en X con inicio en x_0 y fin en $x_1 = (g \circ f)(x_0)$ dado por $\gamma(t) = F(x_0, t)$. Resaltamos una propiedad de utilidad, que indica que

$$(g \circ f)_* = \gamma_* : \pi_1(X, x_0) \rightarrow \pi_1(X, x_1),$$

con γ_* definido por $\gamma_*([\alpha]) = [\gamma^{-1} \cdot \alpha \cdot \gamma]$, como en la demostración del teorema 3.2. Para mostrar esto, consideremos a $\gamma_t(s) = \gamma^{-1}((1-s)(1-t))$, un camino en X con inicio en $\gamma(t)$ y fin en x_1 . Notamos que todo lazo α en X con punto base en x_0 cumple con $(g \circ f) \circ \alpha \simeq_p \gamma^{-1} \cdot \alpha \cdot \gamma$, gracias a la homotopía por caminos

$$H(s, t) = \gamma_t^{-1}(s) \cdot F(\alpha(s), t) \cdot \gamma_t(s).$$

Para un t dado, $\gamma_t^{-1}(s)$ es un camino de x_1 a $\gamma(t)$, $F(\alpha(s), t)$ es un lazo con base en $\gamma(t)$ y $\gamma_t(s)$ es un camino de $\gamma(t)$ a x_1 . Se sigue que H es una homotopía por caminos de $\gamma^{-1} \cdot \alpha \cdot \gamma$ a $(g \circ f) \circ \alpha$.

Un tratamiento análogo de la homotopía $G : Y \times I \rightarrow Y$ de $\mathbb{1}_Y$ a $f \circ g$, así como del camino δ en Y de y_0 a $y_1 = (f \circ g)(y_0)$ dado por $\delta(t) = G(y_0, t)$, muestra que

$$(f \circ g)_* = \delta_* : \pi_1(Y, y_0) \rightarrow \pi_1(Y, y_1).$$

Con estas propiedades, el teorema sigue de inmediato. Vemos que $g_* \circ f_* = (g \circ f)_* = \gamma_*$, donde γ_* es un isomorfismo por el teorema 3.2. Luego, f_* es inyectivo. De igual manera, notamos que $f_* \circ g_* = (f \circ g)_* = \delta_*$, donde δ_* también es un isomorfismo. Por lo tanto, f_* es sobreyectivo. Concluimos que f_* es un isomorfismo. \square

De nuevo, combinando este resultado con el teorema 3.2, en espacios conexos por caminos, se obtiene el siguiente resultado.

Teorema 3.6. *El grupo fundamental, salvo isomorfismo, es invariante entre espacios homotópicamente equivalentes y conexos por caminos.*

Un caso especial de espacios homotópicamente equivalentes es el siguiente. Sean X un espacio topológico y A un subespacio de X . Decimos que una homotopía $H : X \times I \rightarrow X$ es una retracción de X a A si $H(x, 0) = x$ y $H(x, 1) \in A$ para todo $x \in X$, y además $H(a, t) = a$ para todo $a \in A$ y $0 \leq t \leq 1$. Si existe tal función, decimos que A es una deformación por retracción de X , o que X se deforma por retracción a A . En este caso, se sigue que X y A son homotópicamente equivalentes. Para ello, podemos considerar los mapeos continuos $f : X \rightarrow A$ y $g : A \rightarrow X$ dados por $f(x) = H(x, 1)$ y $g(a) = a$, los cuales se verifica que son equivalencias homotópicas inversas entre sí. Como ejemplos rápidos, vemos que un plano pinchado y una banda de Möbius se deforman por retracción a S^1 .

3.4. Cálculos

3.4.1. El círculo y el toro

Hasta el momento, en este escrito no hemos calculado ningún grupo fundamental no trivial. En la literatura, el primero de estos cálculos suele ser $\pi_1(S^1) = \mathbb{Z}$. Esta identificación se alcanza a través del isomorfismo que asigna cada entero n al lazo γ_n en $S^1 \subset \mathbb{C}$ dado por $\gamma_n(t) = e^{2\pi i n t}$. En palabras, el lazo γ_n da n vueltas en sentido antihorario a S^1 . Este resultado comunica que S^1 cuenta con un agujero, lo que hace sentido intuitivo. No obstante, mostrar que esta asignación es en verdad un isomorfismo demanda múltiples lemas y páginas de desarrollo. Esta prueba puede encontrarse en cualquier libro de texto que cubra el grupo fundamental [2, 13, 20].

Para facilitar el cálculo de grupos fundamentales, discutimos dos propiedades relevantes de la teoría de homotopía. La primera describe el grupo fundamental de un espacio producto, mientras que el segundo describe el grupo fundamental de un espacio que puede descomponerse como unión de abiertos conexos por caminos. Este segundo resultado es más profundo y se conoce como el teorema de Van Kampen.

Teorema 3.7. *Si X y Y son espacios conexos por caminos, entonces $\pi_1(X \times Y) \simeq \pi_1(X) \times \pi_1(Y)$.*

Demostración. De la topología se conoce que el producto de dos espacios topológicos conexos por caminos también es conexo por caminos. Basamos todos los lazos en estos espacios en puntos fijos $x_0 \in X$, $y_0 \in Y$ y $(x_0, y_0) \in X \times Y$. Llamamos e_X, e_Y y $e_{X \times Y}$ a los lazos constantes en estos puntos.

Sean $p_1 : X \times Y \rightarrow X$ y $p_2 : X \times Y \rightarrow Y$ las proyecciones desde el espacio producto. Al ser p_1 y p_2 mapeos continuos, contamos con homomorfismos inducidos p_{1*} y p_{2*} . Esto motiva a construir

el homomorfismo ϕ de $\pi_1(X \times Y)$ a $\pi_1(X) \times \pi_1(Y)$ cuyas entradas están dadas por p_{1*} y p_{2*} . En concreto, tenemos

$$\phi : \pi_1(X \times Y) \rightarrow \pi_1(X) \times \pi_1(Y), \quad \phi([\alpha]) = ([p_1 \circ \alpha], [p_2 \circ \alpha]).$$

Mostramos que ϕ es un isomorfismo. Sea $[\alpha] \in \pi_1(X \times Y)$ tal que $\phi([\alpha]) = ([e_{x_0}], [e_{y_0}])$. Luego, $p_1 \circ \alpha \simeq_p e_X$ vía una homotopía por caminos F y $p_2 \circ \alpha \simeq_p e_Y$ vía una homotopía por caminos G . Se sigue que $H(s, t) = (F(s, t), G(s, t))$ es una homotopía por caminos entre α y $e_{X \times Y}$. Por lo tanto, únicamente $[e_{X \times Y}]$ tiene imagen nula bajo ϕ , de modo que ϕ es inyectivo.

Por otra parte, sea $([\alpha_1], [\alpha_2]) \in \pi_1(X) \times \pi_1(Y)$. Consideramos el lazo $\alpha(s) = (\alpha_1(s), \alpha_2(s))$ en $X \times Y$, el cual vemos que cumple $\phi([\alpha]) = ([\alpha_1], [\alpha_2])$. Con esto, ϕ es sobreyectivo. En resumen, ϕ es un isomorfismo. \square

Con este resultado en mano tenemos acceso al cálculo de más grupos fundamentales. En particular, para el toro \mathbb{T}^2 tenemos $\pi_1(\mathbb{T}^2) = \pi_1(S^1 \times S^1) = \pi_1(S^1) \times \pi_1(S^1) = \mathbb{Z} \times \mathbb{Z}$.

3.4.2. El teorema de Van Kampen

Ahora, procedemos con el enunciado del teorema de Van Kampen. Abarcamos la prueba en el caso especial donde hay dos abiertos U_1 y U_2 involucrados. La interesante prueba del caso general, donde se agrega una condición adicional, puede leerse en el libro de texto de topología algebraica de Hatcher [13]. Además, puede verse la sección 2.1 de esta tesis para un tratamiento de grupos y productos libres, útil para entender este teorema.

Teorema 3.8 (Van Kampen). *Sea X un espacio conexo por caminos con abiertos U_1 y U_2 no disjuntos tales que $X = U_1 \cup U_2$. Supóngase que U_1 , U_2 y $U_1 \cap U_2$ también son conexos por caminos. Sean i_1, i_2, j_1, j_2 los mapeos inclusión del primer diagrama en la Figura 3.3.*

*Luego, la extensión homomorfa $\phi : \pi_1(U_1) * \pi_1(U_2) \rightarrow \pi_1(X)$ de i_{1*} y i_{2*} es sobreyectiva. En adición, $\ker \phi$ es generado por los elementos de la forma $j_{1*}([\alpha])j_{2*}([\alpha])^{-1}$ con $[\alpha] \in \pi_1(U_1 \cap U_2)$. Con esto, tenemos el isomorfismo $\pi_1(X) \simeq \pi_1(U_1) * \pi_1(U_2) / \ker \phi$.*

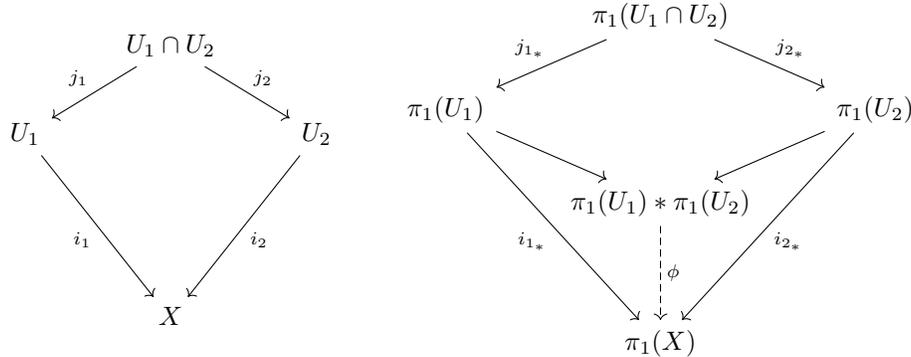


Figura 3.3: Relaciones entre elementos del teorema de Van Kampen.

Demostración. Ilustramos la relación entre los homomorfismos mencionados en el segundo diagrama de la Figura 3.3. A lo largo de esta discusión fijamos un $x_0 \in U_1 \cap U_2 \subset U_1, U_2 \subset X$ que es punto base para todos los lazos discutidos en estos espacios. Primero, mostramos la sobreyectividad de ϕ .

- *Sobreyectividad de ϕ :* Sea α un lazo en X . Construimos lazos $\tilde{\alpha}_0, \dots, \tilde{\alpha}_n$, cada uno contenido completamente en U_1 o U_2 , tales que $\alpha \simeq_p \tilde{\alpha}_0 \cdots \tilde{\alpha}_n$, donde por $\tilde{\alpha}_0 \cdots \tilde{\alpha}_n$ nos referimos al producto

de lazos definido en $\pi_1(X)$. Con esto, tendremos $\phi([\tilde{\alpha}_0]_U \cdots [\tilde{\alpha}_n]_U) = [\alpha]_X$. Empleamos la notación $[\cdot]_X$ para referirnos a la clase de un lazo en $\pi_1(X)$, $[\cdot]_{U_i}$ para la clase de un lazo en $\pi_1(U_i)$ y $[\cdot]_U$ cuando la clase es sobre U_1 o U_2 , sin especificar alguno.

Primero, notamos que $\alpha(s)$ se encuentra en U_1 o U_2 para cada $s \in I$. Como U_1 y U_2 son abiertos y α es continuo, cada s está contenido en un intervalo $[s - \epsilon_s, s + \epsilon_s]$ cuya imagen bajo α también se encuentra en U_1 o U_2 . La unión de todos los $(s - \epsilon_s, s + \epsilon_s)$ forma una cubierta abierta de I , la cual posee una subcubierta finita por la compacidad de I . Seleccionamos los límites de los intervalos de esta subcubierta como los miembros de una partición $0 = s_0 < s_1 < \cdots < s_{m+1} = 1$ de I . De nuestra construcción, se sigue que cada $\alpha([s_i, s_{i+1}])$ está completamente contenido en U_1 o U_2 . Incluso, si eliminamos a los s_i en nuestra partición cuya imagen bajo α no esté en $U \cap V$, aún cumplimos con esta propiedad. Por ejemplo, si $\alpha(s_i) \in U_1 - U_2$, se necesita que $\alpha([s_{i-1}, s_i])$ y $\alpha([s_i, s_{i+1}])$ estén contenidos en U_1 , con lo cual $\alpha([s_{i-1}, s_{i+1}]) \subset U_1$. Escogemos a los s_i de este modo.

Denotamos por α_i a la restricción de α en $[s_i, s_{i+1}]$, y notamos que $\alpha \simeq_p \alpha_0 \cdots \alpha_m$, como producto de caminos. Sin embargo, buscamos expresar a α como producto de lazos, no de caminos, con lo cual agregamos términos a este producto. Para cada $s_i \in U_1 \cap U_2$ existe un camino γ_i de x_0 a $\alpha(s_i)$ debido a la conexidad por caminos de $U_1 \cap U_2$. Definimos un lazo β en X como

$$\beta = (\alpha_0 \cdot \gamma_1^{-1}) \cdot (\gamma_1 \cdot \alpha_1 \cdot \gamma_2^{-1}) \cdots (\gamma_{m-1} \cdot \alpha_{m-1} \cdot \gamma_m^{-1}) \cdot (\gamma_m \cdot \alpha_m). \quad (3.3)$$

Cada camino en paréntesis es un lazo con base en x_0 . Los γ_i agregados se cancelan al discutir homotopía, con lo cual $\beta \simeq_p \alpha_0 \cdots \alpha_m \simeq_p \alpha$. Como cada uno de estos lazos está contenido en U_1 o U_2 , esta es la representación deseada de α . Por lo tanto, ϕ es sobreyectivo.

• *Identificación de $\ker \phi$* : Procedemos con la segunda parte de la prueba. Sea N el subgrupo normal de $\pi_1(U_1) * \pi_1(U_2)$ generado por todos los $j_{1*}([\alpha]j_{2*}([\alpha])^{-1})$ con $[\alpha] \in \pi_1(U_1 \cap U_2)$. Además, sea $\bar{\phi}: \pi_1(U_1) * \pi_1(U_2)/N \rightarrow \pi_1(X)$ el homomorfismo canónico inducido por ϕ en el grupo cociente. Probamos que $\bar{\phi}$ es inyectivo. Para este fin, sea $[\alpha_0]_U \cdots [\alpha_k]_U$ un elemento en $\ker \bar{\phi}$. Seguimos usando la notación $[\cdot]_U$ para elementos del cociente, aunque ahora contamos con las relaciones $[\alpha]_{U_1} = [\alpha]_{U_2}$ cuando $[\alpha] \in \pi_1(U \cap V)$.

Queremos mostrar que $[\alpha_0]_U \cdots [\alpha_k]_U$ se anula en $\pi_1(U_1) * \pi_1(U_2)/N$. Como $[\alpha_0]_U \cdots [\alpha_k]_U \in \ker \bar{\phi}$, tenemos que $\alpha = \alpha_0 \cdots \alpha_k \simeq_p e$ como lazos en X , donde e es el lazo constante en x_0 . Sea H una homotopía por caminos en X de $\alpha_0 \cdots \alpha_k$ a e . Por un argumento similar al visto en la sobreyectividad de ϕ , existen particiones $0 = s_0 < s_1 < \cdots < s_{m+1} = 1$ y $0 = t_0 < t_1 < \cdots < t_{n+1} = 1$ de I tales que, para cada rectángulo $R_{ij} = [s_i, s_{i+1}] \times [t_j, t_{j+1}]$, tenemos que $H(R_{ij})$ está contenido en U_1 o U_2 . Añadimos los puntos límite de los dominios de los α_i en α a la partición de los s_i . Además, renombramos los α_i a las restricciones de α a los $[s_i, s_{i+1}]$, notando que el producto original $[\alpha_0]_U \cdots [\alpha_k]_U$ en $\pi_1(U_1) * \pi_1(U_2)/N$ no cambia.

Sea μ_{ij} el camino en X dado por la composición de la línea recta en $I \times I$ de (s_i, t_j) a (s_{i+1}, t_j) con H . Del mismo modo, sea ν_{ij} el camino en X dado por la composición de la línea recta en $I \times I$ de (s_i, t_j) a (s_i, t_{j+1}) con H . Dado que $\alpha(s) = H(s, 0)$, se sigue que α es el producto de los caminos $\mu_{i,0}$ en X . Buscando expresar a α como producto de lazos en X , añadimos caminos adicionales a este producto. Por la conexidad por caminos de U_1 y U_2 existe un camino γ_{ij} de x_0 a $H(s_i, t_j)$ contenido en U_1 o U_2 , dependiendo donde esté $H(s_i, t_j)$. Es más, por la conexidad por caminos de $U_1 \cap U_2$, escogemos este γ_{ij} contenido en $U_1 \cap U_2$ si $H(s_i, t_j) \in U_1 \cap U_2$. Como en la expresión 3.3, notamos que podemos construir lazos $\tilde{\mu}_{ij} = \gamma_{ij} \cdot \mu_{ij} \cdot \gamma_{i+1,j}^{-1}$ y $\tilde{\nu}_{ij} = \gamma_{ij} \cdot \nu_{ij} \cdot \gamma_{i,j+1}^{-1}$ contenidos en U_1 o U_2 . Con esto, tenemos que α es el producto de los lazos $\tilde{\mu}_{i,0}$, cada uno contenido en U_1 o U_2 .

Probamos que el producto de los $[\tilde{\mu}_{i,0}]_U$ se anula en $\pi_1(U_1) * \pi_1(U_2)/N$. Como lazos en X , vemos que $\tilde{\mu}_{i,0} \simeq_p \tilde{\nu}_{i,0} \cdot \tilde{\mu}_{i,1} \cdot \tilde{\nu}_{i+1,0}^{-1}$, pues $\mu_{i,0}$ es homotópico a $\nu_{i,0} \cdot \mu_{i,1} \cdot \nu_{i+1,0}^{-1}$ vía la restricción de H en $R_{i,0}$. Si $\tilde{\mu}_{i,0}$ y $H(R_{ij})$ se encuentran ambos en U_1 , tenemos $[\tilde{\mu}_{i,0}]_{U_1} = [\tilde{\nu}_{i,0}]_{U_1} [\tilde{\mu}_{i,1}]_{U_1} [\tilde{\nu}_{i+1,0}^{-1}]_{U_1}$. Ahora, si $\tilde{\mu}_{i,0}$ se encuentra en U_1 y $H(R_{ij})$ en U_2 , entonces $\tilde{\mu}_{i,0}$ está contenido en $U \cap V$. Dado que estamos trabajando en el cociente $\pi_1(U_1) * \pi_1(U_2)/N$, tenemos la relación $[\tilde{\mu}_{i,0}]_{U_1} = [\tilde{\mu}_{i,0}]_{U_2}$, con lo cual $[\tilde{\mu}_{i,0}]_{U_1} = [\tilde{\mu}_{i,0}]_{U_2} = [\tilde{\nu}_{i,0}]_{U_2} [\tilde{\mu}_{i,1}]_{U_2} [\tilde{\nu}_{i+1,0}^{-1}]_{U_2}$.

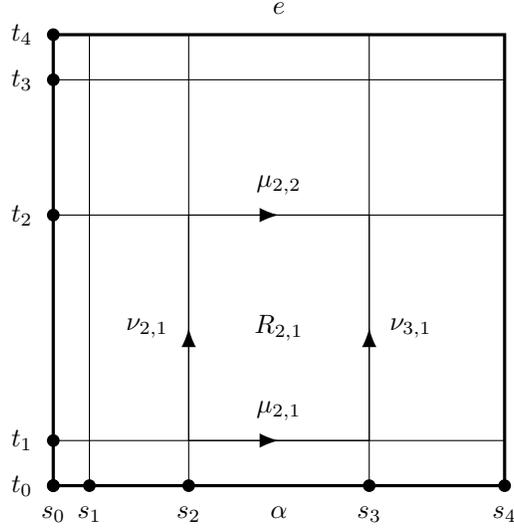


Figura 3.4: Cuadrículado del dominio $I \times I$ de H .

Aplicando estas relaciones, vemos que el producto de los $[\tilde{\mu}_{i,0}]$ que define a $[\alpha_1]_U \cdots [\alpha_k]_U$ es igual al producto de los $[\tilde{\mu}_{i,1}]$, pues todos los $[\tilde{\nu}_{i,0}]_U$ y $[\tilde{\nu}_{i,0}^{-1}]_U$ se cancelan entre sí. Realizando el mismo procedimiento con la segunda fila y continuando con las demás, eventualmente llegamos a la última fila. Con esto, el producto $[\alpha_1]_U \cdots [\alpha_k]_U$ es igual a un producto de ciclos $[\tilde{\mu}_{i,n+1}]_U$, todos nulos en $\pi_1(U_1) * \pi_1(U_2)/N$, con lo cual $[\alpha_1]_U \cdots [\alpha_k]_U$ también es nulo. Esto concluye que $\tilde{\phi}$ es inyectivo. \square

3.4.3. La esfera, suma cuña y suma conexa

Con el teorema de Van Kampen tenemos acceso a otra variedad de grupos fundamentales de otros espacios. Un espacio topológico que resulta especialmente fácil de enfrentar es la esfera S^2 . Podemos tomar como abiertos U y V de S^2 a la esfera menos su polo norte y la esfera menos su polo sur, y notar que esto satisface las condiciones del teorema. Se sigue que $\pi_1(S^2)$ puede obtenerse como un cociente de $\pi_1(U) * \pi_1(V)$. Sin embargo, tanto U como V son homotópicamente equivalentes a un punto, con lo cual sus grupos fundamentales y su producto libre son triviales. En resumen, encontramos $\pi_1(S^2) = \{e\}$.

Otro espacio de interés la suma cuña de espacios, definido del siguiente modo. Para dos espacios X y Y y dos puntos $x_0 \in X$ y $y_0 \in Y$, definimos la suma cuña de X y Y , denotada por $X \vee Y$, como el cociente en la unión disjunta $X \sqcup Y$ que identifica a x_0 con y_0 .

Supongamos que X y Y son espacios conexos por caminos, A un abierto de X que se deforma por retracción a $x_0 \in X$ y B un abierto de Y que se deforma por retracción a $y_0 \in Y$. Luego, X y Y son deformaciones por retracción de $U = X \vee B$ y $V = A \vee Y$, respectivamente. Además, vemos que $U \cap V = A \vee B$, el cual se deforma por retracción a x_0 . Esto implica que $U \cap V$ es conexo por caminos, lo cual termina de corroborar las hipótesis del teorema de Van Kampen. Dado que $\pi_1(U \cap V)$ es trivial, el homomorfismo $\phi : \pi_1(U) * \pi_1(V) \rightarrow \pi_1(X \vee Y)$ es en realidad un isomorfismo.

Un ejemplo de la suma cuña anterior es el espacio $S^1 \vee S^1$, el cual es homeomorfo a la figura “8”. Este posee grupo fundamental $\pi_1(S^1 \vee S^1) = \pi_1(S^1) * \pi_1(S^1) = \mathbb{Z} * \mathbb{Z}$, el grupo libre con dos generadores. En general, tenemos que la suma cuña de n círculos S^1 tiene como grupo fundamental el producto libre de n copias de \mathbb{Z} .

Presentamos un último cálculo. La suma conexa de dos toros, también llamada doble toro y

denotada por $\mathbb{T}^2 \# \mathbb{T}^2$, es una superficie obtenida de remover el interior de un disco en dos toros e identificar los bordes de estos discos. Un modelo útil de este espacio como un cociente, similar al modelo del toro, es un octágono con 4 pares de lados identificados, como en la Figura 3.5. Ubicamos un abierto U en este modelo, señalado con rojo, un abierto V , señalado con azul, y un punto $x_0 \in U \cap V$. Notamos que U , V y $U \cap V$ cumplen las condiciones del Teorema de Van Kampen. Por un lado, U y $U \cap V$ se deforman por retracción a $\{x_0\}$ y S^1 , respectivamente, con lo cual $\pi_1(U) = \{e\}$ y $\pi_1(U \cap V) = \mathbb{Z}$, donde e es el lazo constante en x_0 . Por otra parte, V se deforma por retracción al borde del octágono, el cual es homeomorfo a $S^1 \vee S^1 \vee S^1 \vee S^1$. Por lo tanto, tenemos $\pi_1(V) = \mathbb{Z} * \mathbb{Z} * \mathbb{Z} * \mathbb{Z}$. Entonces, $\pi_1(U) * \pi_1(V)$ es el grupo libre con cuatro generadores.

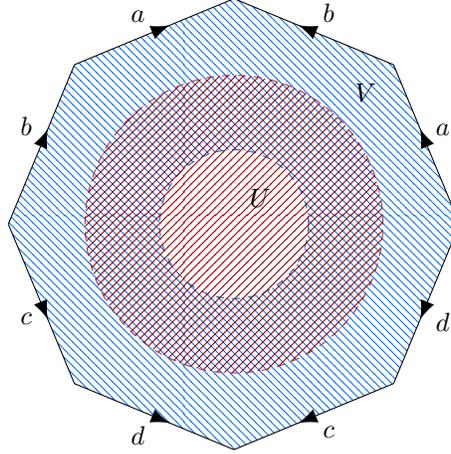


Figura 3.5: Modelo del doble toro como un cociente.

De acuerdo con el teorema de Van Kampen, obtenemos $\pi_1(U \cup V)$ al cocientar $\pi_1(U) * \pi_1(V)$ con su subgrupo normal N generado por los $[\alpha]_U[\alpha]_V^{-1}$, con $[\alpha] \in \pi_1(U \cap V)$. Si $[\alpha] \in \pi_1(U \cap V)$, tenemos que $[\alpha]_U$ es nulo porque $\pi_1(U)$ es trivial. Por otro lado, si $[\alpha]$ genera a $\pi_1(V)$, entonces α es homotópico al borde del octágono en $\pi_1(V)$. Luego, tenemos $[\alpha]_V = [aba^{-1}b^{-1}cdc^{-1}d^{-1}]_V$. Por lo tanto, en notación de generadores y relaciones, el grupo fundamental del toro está dado por

$$\pi_1(\mathbb{T}^2 \# \mathbb{T}^2) = (\mathbb{Z} * \mathbb{Z} * \mathbb{Z} * \mathbb{Z}) / N = \langle a, b, c, d \mid aba^{-1}b^{-1}cdc^{-1}d^{-1} = 1 \rangle$$

Un argumento similar puede darse para la suma conexas de n toros, también llamada una superficie de género n . En este caso, vemos que el grupo fundamental de una superficie de género n es un cociente del grupo libre con $2n$ generadores $a_1, b_1, \dots, a_n, b_n$. En términos de generadores y relaciones, este grupo es

$$\pi_1(\underbrace{\mathbb{T}^2 \# \dots \# \mathbb{T}^2}_{n \text{ toros}}) = \langle a_1, b_1, \dots, a_n, b_n \mid a_1 b_1 a_1^{-1} b_1^{-1} \dots a_n b_n a_n^{-1} b_n^{-1} = 1 \rangle.$$

Triangulaciones y superficies

En este capítulo hacemos un breve tratamiento teórico de triangulaciones con el fin de encaminarnos a estudiar otras estructuras de la topología algebraica, los denominados grupos de homología. Posteriormente, analizamos algunos algoritmos para la generación de triangulaciones. Un texto que expande en el tópico de triangulaciones es el presentado por De Loera, Rambau y Santos [17].

4.1. Complejos simpliciales

El capítulo pasado estudiamos el grupo fundamental, una estructura algebraica derivada de un espacio topológico. Este grupo tiene la ventaja de estar definido sobre cualquier espacio arbitrario, que además resulta ser un invariante no solo bajo homeomorfismos, sino que también sobre equivalencia homotópica de espacios conexos por caminos. No obstante, el grupo fundamental viene con algunos inconvenientes.

En primer lugar, no contamos con un procedimiento estandarizado para calcular el grupo fundamental de un espacio arbitrario X . Dicho procedimiento requeriría describir con más detalle la colección de lazos en X con punto base común, la cual no posee una clasificación clara y fácilmente puede ser de cardinal no contable dependiendo de la topología en X . Además, requerimos de un algoritmo de decisión que identifique si dos lazos son homotópicos por caminos entre sí. Sin embargo, la definición de homotopía por caminos descansa en la existencia de cierto mapeo $H : I \times I \rightarrow X$, la cual por el momento no hay esperanza de determinar algorítmicamente.

Hasta el momento hemos calculado únicamente un par de grupos fundamentales no triviales, lo cual requirió herramienta sofisticada como el teorema de Van Kampen. No obstante, esta herramienta queda obsoleta cuando estudiamos espacios más generales que no podemos describir inmediatamente, como sumas conexas o productos de espacios. Como vía alterna, nos encaminamos a estudiar otra estructura algebraica, los grupos de homología, que gozan de una mayor facilidad de cálculo.

Con este objetivo en mente, restringimos nuestro enfoque a espacios con una estructura particular, los llamados espacios triangulables. Describimos estos espacios en el lenguaje de los símlices, como vemos a continuación.

Definición 4.1. Una combinación convexa de un conjunto de puntos $v_0, \dots, v_k \in \mathbb{R}^n$ es una combinación lineal $v = \sum_{i=0}^k \lambda_i v_i$, donde cada coeficiente λ_i es no negativo y $\sum_{i=0}^k \lambda_i = 1$. Cuando no se exige la no negatividad de los λ_i , se dice que la combinación lineal es afín. El casco convexo (afín) de un conjunto de puntos es el conjunto de todas sus combinaciones convexas (afines).

Definición 4.2. Un conjunto de puntos $v_0, \dots, v_k \in \mathbb{R}^n$ es afínmente independiente si todo subconjunto propio de v_0, \dots, v_k produce un casco afín estrictamente contenido en el casco afín de v_0, \dots, v_k .

Equivalentemente, se dice que un conjunto de puntos v_0, \dots, v_k es afínmente independiente si, y solo si, el conjunto $v_1 - v_0, v_2 - v_0, \dots, v_k - v_0$ es linealmente independiente.

Definición 4.3. Un k -símplex es el casco convexo de un conjunto de $k + 1$ puntos afínmente independientes, y se dice que un k -símplex tiene dimensión k .

Usamos el término *símplex* para referirnos a un k -símplex sin una dimensión específica, y *símplices* para su plural. De particular interés serán los símplices de dimensiones 0, 1 y 2, los cuales llamamos *vértices*, *aristas* y *triángulos*, respectivamente. Incluimos abajo ilustraciones de algunos símplices de dimensiones 0 a 3.

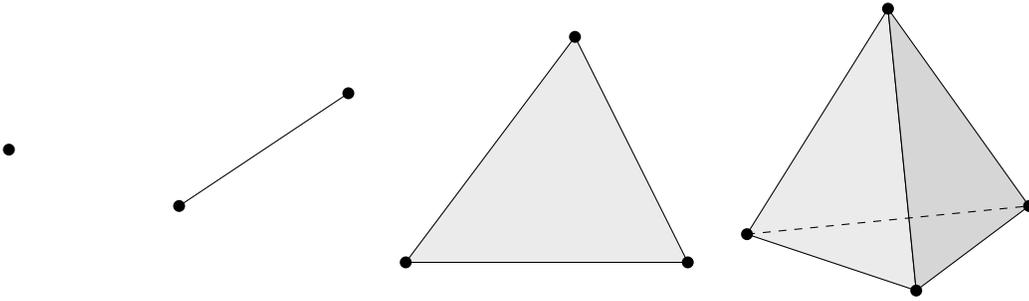


Figura 4.1: Símplices de dimensiones 0, 1, 2 y 3, de izquierda a derecha.

Definición 4.4. Una cara de un k -símplex con vértices v_0, \dots, v_k es un k' -símplex, con $k' \leq k$, definido por un subconjunto de los vértices v_0, \dots, v_k .

Los símplices forman los bloques de construcción de los espacios topológicos a los que restringimos nuestro enfoque. Intuitivamente, “pegamos” un conjunto de símplices en algún espacio euclidiano, cara con cara, para formar una estructura más grande. Formalizamos esto con la definición de *complejo simplicial*.

Definición 4.5. Un *complejo simplicial* es una colección finita K de símplices que cumple

- Si $\sigma \in K$, entonces K contiene todas las caras de σ .
- Si $\sigma, \sigma' \in K$, entonces σ y σ' son disjuntos o se intersecan en una cara común.

En ocasiones usamos el término *complejo* para hablar de complejos simpliciales. Definimos la dimensión de un complejo simplicial como la máxima dimensión de sus símplices.

Dado un complejo simplicial K , se emplea la notación $|K|$ cuando se desea tratar al complejo en su calidad de espacio topológico. Dotamos a $|K|$ de la topología de subespacio, heredada del espacio euclidiano donde habita K . Al espacio $|K|$ se le denomina *polihedro*.

Definición 4.6. Una *triangulación* de un espacio topológico X es un complejo simplicial K junto con un homeomorfismo $h : |K| \rightarrow X$. Si dicha triangulación existe, el espacio X se dice *triangulable*.

Se sigue que “ser triangulable” es un invariante topológico.

Como anticipábamos, la colección de espacios triangulables es restricta. En primer lugar, todo polihedro proviene de un espacio euclidiano y por lo tanto es metrizable. Segundo, todo polihedro puede descomponerse como unión finita de cascós convexos, por lo que $|K|$ es cerrado en un espacio euclidiano y, por lo tanto, compacto. Entonces, cualquier espacio triangulable, el cual debe ser homeomorfo a algún polihedro, también debe ser metrizable y compacto. Esto puede parecer una concesión fuerte, pero múltiples espacios de interés cumplen con esta descripción, como vemos en la siguiente sección.

Antes de continuar con las triangulaciones, mencionamos algunos conceptos en relación a la orientación de un simplex.

Definición 4.7. *Un k -simplex σ con vértices v_0, \dots, v_k es orientado si sus vértices tienen asociado un orden (v_0, \dots, v_k) . Decimos que dos órdenes son equivalentes si estos difieren por una permutación par. La orientación de σ es la clase de equivalencia del orden de sus vértices bajo esta relación.*

Usamos la notación $v_0 \cdots v_k$ para referirnos al k -simplex orientado generado por los vértices v_0, \dots, v_k con ese orden. Cuando $k > 0$, vemos que un k -simplex tiene exactamente dos posibles orientaciones, cada una correspondiente a una de las clases laterales del subgrupo de permutaciones pares en el grupo de permutaciones de $k + 1$ elementos. Un 0-simplex posee una única orientación.

Finalmente, como referencia futura que será útil al momento de estudiar homología, definimos la orientación que un simplex induce en sus caras.

Definición 4.8. *Sea σ el k -simplex orientado $v_0 \cdots v_k$, y sea σ_i la cara de σ obtenida de eliminar el vértice v_i de σ . El orden $v_0 \cdots v_{i-1} v_{i+1} \cdots v_k$ define una orientación para σ_i . Si i es par, dotamos a σ_i de esta orientación, y si i es impar, dotamos a σ_i de la orientación opuesta.*

4.2. Cálculo de triangulaciones

Aunque los espacios triangulables forman una colección de espacios muy restringidos ante la gran familia de espacios topológicos generales, resulta que múltiples espacios topológicos de interés admiten una triangulación. En específico, nos referimos a superficies compactas. Un resultado por Radó garantiza que cada superficie compacta es triangulable [21]. Ahora, buscamos métodos de construcción de triangulaciones.

4.2.1. Triangulación por parametrización

En primer lugar, notamos que una superficie parametrizada como $f : [a, b] \times [c, d] \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3$ es fácilmente triangulable, pues su dominio admite una triangulación. Como ejemplo, consideramos el problema de triangular el toro \mathbb{T}^2 . Si \mathbb{T}^2 tiene radio mayor R y radio menor r , una parametrización $f : [0, 2\pi] \times [0, 2\pi] \rightarrow \mathbb{R}^3$ del toro está dada por

$$f(u, v) = ((R + r \cos v) \cos u, (R + r \cos v) \sin u, r \sin v). \quad (4.1)$$

Vemos que el dominio $[0, 2\pi] \times [0, 2\pi]$ del toro es una superficie compacta, con lo cual admite una triangulación por el teorema de Radó. Podemos construir explícitamente un complejo simplicial de dimensión 2 como en la Figura 4.2 que triangule a $[0, 2\pi] \times [0, 2\pi]$. Ahora, si consideramos la colección de imágenes de los simplices de este complejo bajo f , obtenemos otro complejo simplicial en \mathbb{R}^3 , también de dimensión 2, que es homeomorfo al toro. Ya que identificamos los lados opuestos del cuadrado $[0, 2\pi] \times [0, 2\pi]$ al formar a \mathbb{T}^2 , nuestra triangulación del toro utiliza 9 vértices, 27 aristas y 18 triángulos. Mostramos este complejo también en la Figura 4.2.

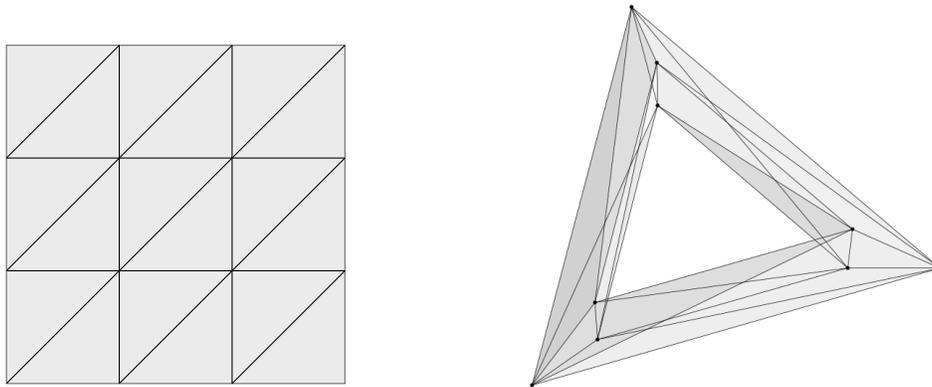


Figura 4.2: Triangulación del dominio de una parametrización f de \mathbb{T}^2 y su imagen bajo f .

Si partimos el dominio de f en triángulos más finos, obtenemos un polihedro que conserva el homeomorfismo con el toro y que se asemeja más a la imagen suave de f . Al triangular espacios, nos interesa que los complejos sean lo más simples posibles, pero por fines visuales exploramos la visualización de complejos más grandes.

Cuando lidiamos con un alto número de triángulos en \mathbb{R}^3 , optamos por sombrear estos para comunicar profundidad. En el caso donde tratamos con una superficie suave orientable, generada por una parametrización $f(u, v)$, calculamos un campo normal unitario a la superficie, dado por la normalización de $\partial_u f \times \partial_v f$, y lo evaluamos en todos los vértices del complejo. Luego, aproximamos el vector normal a cada triángulo como el promedio de los vectores normales unitarios de sus vértices. Si definimos un vector constante que represente una dirección de iluminación, el producto interno de este vector con el campo normal indica el grado de iluminación que se debe asignar a cada triángulo.

4.2.2. Cubos marchantes

A diferencia del toro de la sección anterior, no todas las superficies que deseamos triangular poseen una parametrización tan agradable. Una representación común de superficies en \mathbb{R}^3 puede darse de forma implícita como el conjunto de ceros de una función $F : \mathbb{R}^3 \rightarrow \mathbb{R}$. Como ejemplo, el toro visto arriba queda determinado por los ceros de

$$F(x, y, z) = (x^2 + y^2 + z^2 + R^2 - r^2)^2 - 4R^2(x^2 + y^2). \quad (4.2)$$

Discutimos un algoritmo, denominado cubos marchantes, que ataca el problema de encontrar una triangulación de una superficie implícita compacta. Este algoritmo, inicialmente descrito por Lorenson en 1987, surgió en el contexto de procesamiento de imágenes a partir de datos médicos en tres dimensiones [18].

Dada una superficie S definida implícitamente por los ceros de $F : \mathbb{R}^3 \rightarrow \mathbb{R}$, encontramos un $R > 0$ suficientemente grande tal que la caja $[-R, R]^3 \subset \mathbb{R}^3$ contenga a S . Consideramos una partición de $[-R, R]$ en n segmentos de igual longitud, la cual induce un reticulado de n^3 cubos de igual tamaño en $[-R, R]^3$. El algoritmo procede por un esquema de dividir y conquistar, triangulando la intersección de S con cada uno de los n^3 cubos anteriores.

Para cada cubo, evaluamos F en sus vértices. Etiquetamos un vértice con un 1 si su imagen bajo F es positiva, o con un 0 en caso contrario. Notamos que las aristas del cubo que conectan vértices con etiquetas distintas indican dónde la superficie interseca al cubo. Como tenemos 8 vértices en cada cubo y dos posibles etiquetas por vértice, existen $2^8 = 256$ posibles formas en las que la superficie

interseca un cubo. Generamos una tabla que indique como construir triángulos en el cubo de acuerdo con cada caso. A su vez, asociamos un vector normal unitario a cada triángulo generado vía una estimación del gradiente de F , obtenido con el método de diferencia finita.

En realidad, muchos de los 256 casos en los que S interseca un cubo resultan equivalentes gracias a las simetrías del cubo. Lorensen define una lista con 15 configuraciones fundamentales que resumen estos, la cual se muestra en la Figura 4.3

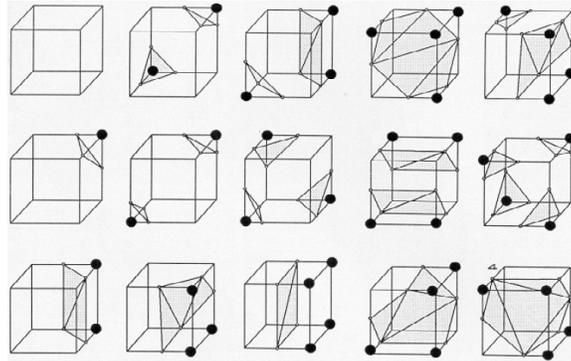
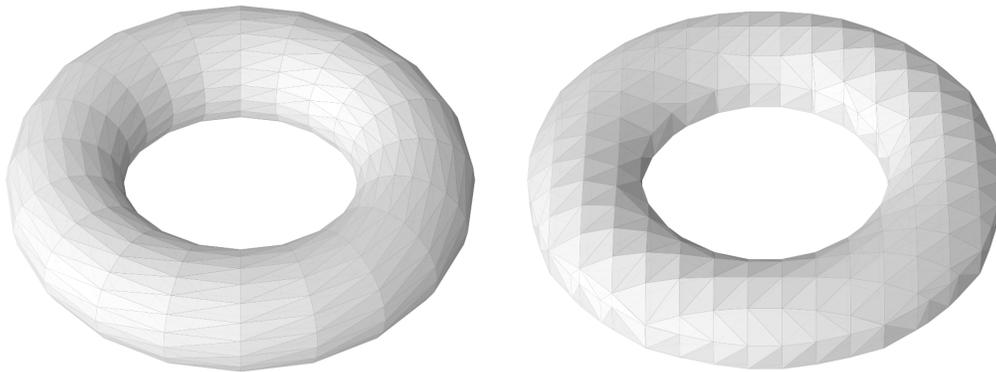


Figura 4.3: Las 15 posibles intersecciones de una superficie con cada cubo [18].

Desde su concepción, el algoritmo ha visto mejoras de rendimiento y precisión. En este trabajo se emplea la versión de Lewiner del algoritmo de cubos marchantes cuando se busca triangular una superficie en \mathbb{R}^3 dada de forma implícita. Esta versión ofrece un método más eficiente al original propuesto por Lorensen, que además garantiza que el polihedro resultante es homeomorfo a la superficie original [16].



(a) Por parametrización.

(b) Por cubos marchantes.

Figura 4.4: Triangulaciones de un toro.

En la Figura 4.4, mostramos dos triangulaciones de un toro obtenidas por los métodos discutidos. La triangulación en la izquierda fue obtenida a partir de la parametrización vista en 4.1. La triangulación en la derecha se dedujo del algoritmo de cubos marchantes según la expresión 4.2. Empleamos la implementación de cubos marchantes de la librería `scikit` del lenguaje de programación Python, la cual emplea la versión de Lewiner. Como puede apreciarse, la segunda triangulación guarda la apariencia del reticulado cúbico utilizado por el algoritmo de cubos marchantes, en contraste con la triangulación más suave en la izquierda.

La triangulación vía parametrización de la Figura 4.4 emplea 800 triángulos, mientras que la triangulación vía cubos marchantes de la misma figura utiliza 832 triángulos. Vemos que la primera triangulación ofrece una imagen más suave que la segunda triangulación, a pesar de que la primera cuenta con menos símlices. Esto es de esperarse, pues el método de cubos marchantes no busca optimizar el tamaño o posición de los triángulos. El algoritmo, al proceder con un esquema de dividir y conquistar, no conoce la geometría global de la superficie ni distribuye los triángulos de acuerdo a esta. Con frecuencia el algoritmo genera triángulos pequeños o con ángulos muy agudos que no aportan mucho a la suavidad de la imagen, en comparación a los triángulos generados por una parametrización. Sin embargo, cubos marchantes nos permite triangular algorítmicamente superficies difíciles de parametrizar.

Con los conceptos de complejos simpliciales y triangulaciones introducidos en el capítulo 4, procedemos a estudiar los grupos de homología. Estos grupos, junto con el grupo fundamental, son las estructuras algebraicas principales que utiliza la topología algebraica para estudiar espacios topológicos. Destacamos que estos grupos a estudiar en este capítulo gozan de una descripción más discreta que el grupo fundamental, con lo cual son más simples de lidiar computacionalmente.

La primera sección expone las definiciones y enunciados básicos de la teoría, mientras que la segunda ofrece un ejemplo de cálculo de los grupos de homología. La tercera sección brevemente describe una variante binaria de estos grupos que enriquece a las estructuras, y la cuarta sección introduce un problema de optimización respecto a las mismas. Como en el capítulo 3, puede encontrarse un desarrollo más amplio de esta teoría en un libro de texto de topología algebraica [2, 13, 19].

5.1. Grupos de homología

Los grupos de homología se construyen en el lenguaje de los grupos libres. Un breve tratamiento de este y otros conceptos relacionados se encuentra en la sección 2.1.

Definición 5.1. *Sea K un complejo simplicial. El grupo de q -cadenas de K , denotado por $C_q(K)$, es el grupo libre abeliano generado por los q -símplices orientados de K , donde tenemos las relaciones $\sigma + \sigma' = 0$ cuando σ, σ' representan al mismo símplex con orientaciones opuestas.*

Empleamos la notación aditiva para la operación del grupo libre para enfatizar que este es abeliano. Notamos que el conjunto de q -símplices en K , con alguna orientación, forma un conjunto generador de $C_q(K)$. Si $\sigma_1, \dots, \sigma_k$ es un conjunto generador de $C_q(K)$, un elemento arbitrario en $C_q(K)$ es una suma formal de la forma $c_1\sigma_1 + \dots + c_k\sigma_k$, donde c_1, \dots, c_k son números enteros. Por conveniencia, definimos $C_{-1}(K) = \{0\}$.

Buscamos definir un homomorfismo entre los grupos de cadenas, lo cual hacemos a través de sus generadores. Definimos una función $\partial_q : C_q(K) \rightarrow C_{q-1}(K)$, primero solo sobre los generadores de $C_q(K)$. Luego, extendemos ∂_q a todo $C_q(K)$ vía $\partial_q(c_1\sigma_1 + \dots + c_k\sigma_k) = c_1\partial_q(\sigma_1) + \dots + c_k\partial_q(\sigma_k)$,

verificando que las relaciones en $C_q(K)$ se mantengan bajo ∂_q .

Formalmente, estamos empleando la presentación de $C_q(K)$ en generadores y relaciones para definir un homomorfismo de $C_q(K)$ a $C_{q-1}(K)$, como en el teorema 2.3. Dado que ambos grupos son abelianos, solo debemos verificar que las relaciones adicionales $\sigma + \sigma' = 0$ en la definición 5.1 satisfacen $\partial_q(\sigma + \sigma') = 0$.

Definición 5.2. *El operador borde $\partial_q : C_q(K) \rightarrow C_{q-1}(K)$ es el homomorfismo tal que, para todo generador $v_0 \cdots v_q$ de $C_q(K)$, se tiene*

$$\partial_q(v_0 \cdots v_q) = \sum_{i=0}^q (-1)^i (v_0 \cdots \hat{v}_i \cdots v_q), \quad (5.1)$$

donde $v_0 \cdots \hat{v}_i \cdots v_q$ es la i -ésima cara de $v_0 \cdots v_q$, obtenida de eliminar al vértice v_i . Decimos que $\partial(\sigma)$ es el borde del simplex σ .

Usamos la notación ∂ para discutir operadores borde sin necesidad de especificar una dimensión. Vemos que $(-1)^i (v_0, \dots, \hat{v}_i, \dots, v_q)$ cuenta con la orientación inducida por el simplex original, como en la definición 4.8. En lo que sigue, denotamos a esta cara orientada por σ_i . Con esta notación, tenemos $\partial(\sigma) = \sum \sigma_i$.

Ahora, verificamos que el operador frontera satisface las relaciones impuestas en el grupo libre abeliano $C_q(K)$. Sean $\sigma, \sigma' \in C_q(K)$ simplices con orientaciones opuestas. Mostramos que las caras σ_i y σ'_i también tienen orientaciones opuestas. Si $\sigma = v_0 v_1 v_2 \cdots v_q$, representamos a σ' por el mismo orden de σ pero adicionamos una transposición entre v_0 y v_1 , con lo cual $\sigma' = v_1 v_0 v_2 \cdots v_q$. Estos ordenes difieren por una permutación impar, y por lo tanto representan orientaciones opuestas de un mismo simplex. Si $i \geq 2$, entonces $\sigma_i = (-1)^i (v_0 v_1 v_2 \cdots \hat{v}_i \cdots v_q)$ y $\sigma'_i = (-1)^i (v_1 v_0 v_2 \cdots \hat{v}_i \cdots v_q)$ también difieren por la transposición entre v_0 y v_1 . Si $i = 0$, tenemos $\sigma_0 = v_1 v_2 \cdots v_q$ y $\sigma'_0 = -v_1 v_2 \cdots v_q$, y si $i = 1$, tenemos $\sigma_1 = -v_0 v_2 \cdots v_q$ y $\sigma'_1 = v_0 v_2 \cdots v_q$. En cualquier caso, σ_i y σ'_i también tienen orientaciones opuestas. Con esto, se cumple

$$\partial_q(\sigma + \sigma') = \partial_q(\sigma) + \partial_q(\sigma') = \sum \sigma_i + \sum \sigma'_i = \sum (\sigma_i + \sigma'_i) = \sum 0 = 0.$$

Ya que ∂_q respeta las relaciones en $C_q(K)$, en efecto podemos extender ∂_q a un homomorfismo sobre todo $C_q(K)$. Con el operador borde bien definido, procedemos a mostrar uno de los resultados centrales de la homología simplicial.

Teorema 5.1. *La composición $\partial_q \circ \partial_{q+1} : C_{q+1}(K) \rightarrow C_{q-1}(K)$ es el homomorfismo nulo.*

Demostración. Sea $\sigma = v_0 \cdots v_{q+1} \in C_{q+1}(K)$ un generador. Mostramos que $\partial_q \circ \partial_{q+1}$ anula a σ . De la definición 5.2, tenemos

$$\partial_q(\partial_{q+1}(\sigma)) = \partial_q \left(\sum_{i=0}^{q+1} \sigma_i \right) = \sum_{i=0}^{q+1} \partial_q(\sigma_i) = \sum_{i=0}^{q+1} \sum_{\substack{j \neq i \\ 0 \leq j \leq q+1}} \sigma_{ij}, \quad (5.2)$$

donde σ_{ij} es la cara orientada de σ obtenida de remover los vértices v_i y v_j , en ese orden. Analizamos con más detalle cada una de las $q \cdot (q+1)$ caras orientadas σ_{ij} de la ecuación 5.2, viendo los casos donde $i > j$ y $i < j$ por separado. Vemos que

$$\begin{aligned} \text{Si } i > j, & \quad \sigma_{ij} = (-1)^{i+j} (v_0 \cdots \hat{v}_j \cdots \hat{v}_i \cdots v_q), \\ \text{Si } i < j, & \quad \sigma_{ij} = (-1)^{i+j-1} (v_0 \cdots \hat{v}_i \cdots \hat{v}_j \cdots v_q). \end{aligned}$$

Si $i > j$, v_j se encuentra en la j -ésima entrada, tanto de σ como de σ_i . Luego, el signo de σ_{ij} en este caso es $(-1)^{i+j}$. Por otra parte, si $i < j$, notamos que v_j , que usualmente se encuentra en la j -ésima

entrada de σ , se ubica en la $(j - 1)$ -ésima entrada de σ_i . Por lo tanto, el signo que acompaña a σ_{ij} es $(-1)^{i+j-1}$.

Con esto vemos que, para $i \neq j$, las caras σ_{ij} y σ_{ji} en la suma 5.2 poseen orientaciones opuestas y se cancelan. Dado que esto ocurre para los $q \cdot (q + 1)/2$ pares de índices $i \neq j$, la suma 5.2 se anula en su totalidad. Por lo tanto, tenemos $(\partial_q \circ \partial_{q+1})(\sigma) = 0$. \square

Este teorema nos dice que el borde de un simplex siempre posee borde nulo. Para reinterpretar este resultado, introducimos una notación para aquellas cadenas con borde nulo, así como aquellas cadenas que son un borde.

Definición 5.3. *Llamamos al kernel de ∂_q el subgrupo de q -ciclos de $C_q(K)$, y lo denotamos por $Z_q(K)$. A su vez, llamamos a la imagen de ∂_{q+1} el subgrupo de q -bordes de $C_q(K)$, y lo denotamos por $B_q(K)$.*

Con esta simbología, el teorema 5.1 nos dice que $B_q(K)$ es un subgrupo de $Z_q(K)$. Como ambos grupos son abelianos, esto motiva a efectuar el cociente entre ellos.

Definición 5.4. *El q -ésimo grupo de homología de K , denotado por $H_q(K)$, es*

$$H_q(K) = \frac{Z_q(K)}{B_q(K)}.$$

Para cada q -ciclo $z \in Z_q(K)$, llamamos a la clase de z en $H_q(K)$ su *clase de homología*, y la denotamos por $[z]$. Cuando dos ciclos comparten clase de homología, les llamamos ciclos *homólogos*.

Notamos que, siendo $Z_q(K)$, $B_q(K)$ y $H_q(K)$ grupos abelianos, podemos ver estas estructuras como \mathbb{Z} -módulos. De hecho, un estudio más general de la homología construye a los grupos de homología como R -módulos, donde R puede ser otro anillo de interés.

Dado que $C_q(K)$ proviene de un complejo simplicial K , tanto $C_q(K)$ como sus subgrupos $Z_q(K)$ y $B_q(K)$ son finitamente generados. Luego, $H_q(K)$ también es un grupo finitamente generado. Por el teorema fundamental de los grupos abelianos finitamente generados, tenemos el isomorfismo $H_q(K) \cong \mathbb{Z}^{\beta_q} \oplus T$, donde T es un grupo abeliano finito de torsión y β_q es un entero no negativo. Al entero β_q se le llama el q -ésimo número de Betti de K .

Los grupos de homología de un complejo simplicial resultan ser invariantes bajo la equivalencia homotópica de su polihedro asociado. Esto puede probarse siguiendo un esquema similar al de la prueba del teorema 3.6, el cual muestra que el grupo fundamental es invariante entre espacios topológicos homotópicamente equivalentes y conexos por caminos. Dados dos complejos simpliciales K y L y dos equivalencias homotópicas $f : |K| \rightarrow |L|$ y $g : |L| \rightarrow |K|$ inversas entre sí, se definen homomorfismos inducidos $f_{q*} : H_q(K) \rightarrow H_q(L)$ y $g_{q*} : H_q(L) \rightarrow H_q(K)$ entre los grupos de homología de K y L . Luego, se muestra que estos son en realidad isomorfismos. Sin embargo, estas demostraciones demandan un trabajo considerablemente mayor que la prueba del teorema 3.6, dada la estructura adicional de los complejos simpliciales y la libertad de elección de una triangulación para representar un polihedro. Una prueba completa siguiendo este esquema puede encontrarse, por ejemplo, en el libro de texto de Armstrong [2].

La invarianza de los grupos de homología entre polihedros con homotópicamente equivalentes permite dotarle grupos de homología a espacios topológicos triangulables. Si X es un espacio triangulable, existe un complejo simplicial K cuyo polihedro $|K|$ es homeomorfo a X . Con esto, definimos el q -ésimo grupo de homología de X como $H_q(X) = H_q(K)$.

Finalmente, un resultado importante de la teoría de homología es la relación entre el primer grupo de homología de un polihedro con su grupo fundamental. Sin entrar a los detalles de la prueba puede mostrarse que, para un complejo K tal que el polihedro $|K|$ es conexo por caminos, el primer grupo

de homología $H_1(K)$ es la abelianización del grupo fundamental $\pi_1(|K|)$. Al igual que la prueba de la invarianza de los grupos de homología, esta prueba exige herramienta previa que estudie el comportamiento de funciones continuas entre polihedros. Resaltamos que este resultado también puede deducirse desde un contexto más general, utilizando el concepto de homología singular [13].

5.2. Ejemplo de cálculo

Ahora que construimos los grupos de homología, presentamos un ejemplo de su cálculo en un complejo simplicial particular. Realizamos los cálculos a mano, preparando un esquema para computarizar nuestro procedimiento en el capítulo siguiente. Sea K el complejo simplicial 2-dimensional mostrado en la Figura 5.1. Acá, la región sombreada indica la presencia de un 2-símplex. Así, K cuenta con 1 triángulo, 7 aristas y 5 vértices, y es homotópicamente equivalente a $S_1 \vee S_1$.

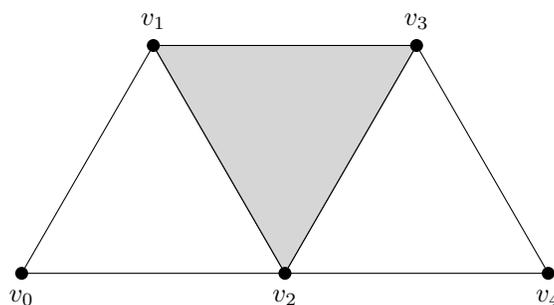


Figura 5.1: Complejo simplicial K .

Utilizamos estos símplexes como los generadores de los primeros tres grupos de cadenas de K , orientados de acuerdo al orden de los índices de sus vértices. Explícitamente, los generadores son

$$\begin{aligned} \text{vértices de } K &: v_0, v_1, v_2, v_3, v_4, \\ \text{aristas de } K &: v_0v_1, v_0v_2, v_1v_2, v_1v_3, v_2v_3, v_2v_4, v_3v_4, \\ \text{triángulos de } K &: v_1v_2v_3. \end{aligned}$$

Buscamos identificar los grupos de homología de K a partir de estos generadores. En primer lugar, vemos que $C_q(K)$ es trivial para todo $q > 2$, pues el complejo K es 2-dimensional. Entonces $Z_q(K)$, y por lo tanto $H_q(K)$, también son triviales para todo $q > 2$.

Luego, calculamos $H_0(K)$. Conocemos que $Z_0(K) = C_0(K)$ ya que $C_{-1}(K)$ es trivial. Además, vemos que $B_0(K)$ es generado por los $\partial_1(\sigma)$, con σ un generador de $C_1(K)$. Estos bordes son de la forma $\partial_1(v_iv_j) = v_j - v_i$ para cada arista v_iv_j . Ahora, si v_i, v_j son vértices cualquiera de K , notamos que existe otro vértice v_k tal que $\partial_1(v_iv_k + v_kv_j) = \partial_1(v_iv_k) + \partial_1(v_kv_j) = (v_k - v_i) + (v_j - v_k) = v_j - v_i$. Entonces, todos los vértices de K son homólogos entre sí, es decir, $H_0(K) = \mathbb{Z}$.

Ahora, calculamos $H_1(K)$. De inmediato tenemos que $B_1(K) = \text{Im } \partial_2 = \langle b_1 \rangle$, donde $b_1 = \partial_2(v_1v_2v_3) = v_2v_3 - v_1v_3 + v_2v_3$. Calcular $Z_1(K)$ requiere un poco más de trabajo. Consideramos un elemento arbitrario σ en $C_1(K)$, dado por

$$\sigma = c_0(v_0v_1) + c_1(v_0v_2) + c_2(v_1v_2) + c_3(v_1v_3) + c_4(v_2v_3) + c_5(v_2v_4) + c_6(v_3v_4).$$

Buscamos condiciones sobre los c_i que equivalgan a $\partial_1(\sigma) = 0$. Aplicando ∂_1 en esta igualdad y empleando la propiedad $\partial_1(v_iv_j) = v_j - v_i$ para cada arista v_iv_j , se sigue

$$\begin{aligned} 0 &= (c_0(v_1) - c_0(v_0)) + (c_1(v_2) - c_1(v_0)) + (c_2(v_2) - c_2(v_1)) + (c_3(v_3) - c_3(v_1)) \\ &\quad + (c_4(v_3) - c_4(v_2)) + (c_5(v_4) - c_5(v_2)) + (c_6(v_4) - c_6(v_3)). \end{aligned}$$

o bien, agrupando términos,

$$0 = (-c_0 - c_1)(v_0) + (c_0 - c_2 - c_3)(v_1) + (c_1 + c_2 - c_4 - c_5)(v_2) \\ + (c_3 + c_4 - c_6)(v_3) + (c_5 + c_6)(v_4).$$

Dado que los generadores en $C_0(K)$ no poseen relaciones entre sí, la igualdad anterior se cumple únicamente cuando los coeficientes que acompañan a los 0-símplices son nulos. Esto produce un sistema de 5 ecuaciones, una por cada vértice, en las 7 variables c_i 's, una por cada arista. En términos de matrices, buscamos el kernel de la matriz

$$\begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Notamos que esta es la matriz de adyacencia de un grafo dirigido asociado al complejo K , con la dirección de las aristas dada por su orientación. Tras encontrar este kernel con eliminación gaussiana, obtenemos un conjunto de generadores para $Z_1(K) = \ker \partial_1$, con 3 1-ciclos dados por

$$z_1 = v_0v_1 - v_0v_2 + v_1v_2, \\ z_2 = -v_0v_1 + v_0v_2 - v_1v_3 + v_2, v_3, \\ z_3 = v_0v_1 - v_0v_2 + v_1v_3 - v_2v_4 + v_3v_4,$$

con lo cual $Z_1(K) = \langle z_1, z_2, z_3 \rangle$. Por lo tanto, tenemos $H_1(K) = Z_1(K)/B_1(K) = \langle z_1, z_2, z_3 \rangle / \langle b_1 \rangle$. Finalmente, se corrobora que $b_1 = -z_1 + z_2 = v_1v_2 + v_2v_3 + v_3v_1$, con lo cual $H_1(K) = \langle [z_1], [z_3] \rangle \simeq \mathbb{Z} \oplus \mathbb{Z}$. La Figura 5.2 ilustra la interpretación geométrica de los generadores de $Z_1(K)$ y $B_1(K)$, y facilita la visualización de sus relaciones algebraicas.

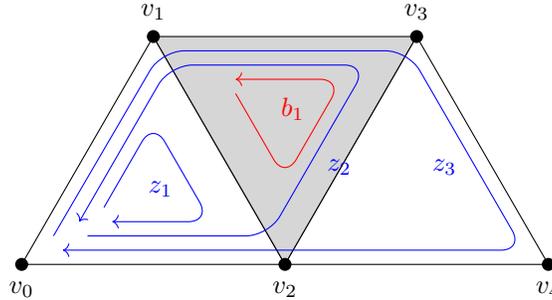


Figura 5.2: Generadores de los grupos $Z_1(K)$ (azul) y $B_1(K)$ (rojo).

Finalmente, para calcular $H_2(K)$, vemos que el único generador de $C_2(K)$ no es un ciclo, con lo cual $Z_2(K)$ es trivial. En consecuencia, $H_2(K)$ también es trivial. Notamos que cada grupo de homología de K se descompone como la suma de copias de \mathbb{Z} , sin necesidad de añadir subgrupos de torsión. Los q -ésimos números de Betti de K encontrados fueron $\beta_0 = 1$, $\beta_1 = 2$ y $\beta_q = 0$ para $q \geq 2$.

A diferencia del grupo fundamental de un espacio topológico, el ejemplo anterior ilustra que los grupos de homología de un complejo simplicial pueden calcularse de forma sistemática. Sin embargo, el cálculo anterior resulta engorroso y su desarrollo a mano puede escalar con facilidad para complejos ligeramente más grandes. Por lo tanto, esto motiva a implementar un algoritmo computacional para calcular grupos de homología. El método visto en esta sección resume la esencia del primer algoritmo presentado en el 6.

5.3. Homología con coeficientes en \mathbb{Z}_2

El grupo de q -cadenas $C_q(K)$ de un complejo simplicial K fue introducido al inicio del presente capítulo como un grupo libre abeliano generado por los q -símplices orientados de K . En este grupo, cada q -cadena es una suma formal de la forma $c_1\sigma_1 + \cdots + c_k\sigma_k$, donde $\sigma_1, \dots, \sigma_k$ son q -símplices de K y c_1, \dots, c_k son números enteros. Este concepto puede generalizarse de tal forma que los coeficientes c_i que acompañan a los símplices ya no estén restringidos a ser enteros, si no que puedan provenir de cualquier anillo R . De interés para nosotros es el caso donde R es el campo \mathbb{Z}_2 .

Sea K un complejo simplicial. Consideramos al grupo abeliano $C_q(K, \mathbb{Z}_2)$ dado por todas las sumas formales $c_1\sigma_1 + \cdots + c_k\sigma_k$, donde $\sigma_1, \dots, \sigma_k$ son q -símplices orientados de K y c_1, \dots, c_n son elementos de \mathbb{Z}_2 . Como en el caso de $C_q(K)$, agregamos las relaciones $\sigma + \sigma' = 0$ en $C_q(K, \mathbb{Z}_2)$ cuando σ, σ' representan al mismo símplex con orientaciones opuestas. Llamamos a $C_q(K, \mathbb{Z}_2)$ el grupo de q -cadenas de K con coeficientes en \mathbb{Z}_2 . Notamos que $C_q(K, \mathbb{Z}_2)$ es una imagen homomorfa de $C_q(K)$, pues estamos conservando su estructura pero introducimos las relaciones $\sigma + \sigma = 0$ para todo $\sigma \in C_q(K)$, lo cual obliga a los coeficientes de las cadenas a ser binarios.

El resto de objetos introducidos en este capítulo pueden traducirse al trato de q -cadenas con coeficientes en \mathbb{Z}_2 . En particular, el operador borde de la definición 5.2 se traduce al operador borde $\partial_q : C_q(K, \mathbb{Z}_2) \rightarrow C_{q-1}(K, \mathbb{Z}_2)$ dado por

$$\partial_q(v_0 \cdots v_q) = \sum_{i=0}^q (v_0 \cdots \hat{v}_i \cdots v_q). \quad (5.3)$$

Podemos ignorar los signos de los coeficientes de la cadena porque estamos trabajando en \mathbb{Z}_2 . Respecto a este operador, definimos los correspondientes *subgrupos de ciclos y bordes de K con coeficientes en \mathbb{Z}_2* , y los denotamos por $Z_q(K, \mathbb{Z}_2)$ y $B_q(K, \mathbb{Z}_2)$. Llamamos al cociente $H_q(K, \mathbb{Z}_2) = Z_q(K, \mathbb{Z}_2)/B_q(K, \mathbb{Z}_2)$ el q -ésimo grupo de homología de K con coeficientes en \mathbb{Z}_2 . Puede mostrarse que $H_q(K, \mathbb{Z}_2)$ también es un invariante bajo la equivalencia homotópica de $|K|$.

Al considerar homología con coeficientes en \mathbb{Z}_2 , perdemos información que brindan los grupos de homología con coeficientes enteros sobre el espacio $|K|$. En particular, ya no contamos con una noción de orientación de cadenas. A cambio de esta concesión, hemos dotado a los grupos de cadenas, ciclos, bordes y de homología de una estructura de espacio vectorial sobre \mathbb{Z}_2 , lo cual no era posible considerando cadenas de coeficientes enteros. La relación entre $H_q(K, \mathbb{Z}_2)$ y $H_q(K)$ puede hacerse precisa con el teorema de coeficientes universales de homología, un resultado más avanzado de la teoría de homología [13]. En particular notamos que, cuando los grupos de homología no cuentan con subgrupos de torsión, tenemos el isomorfismo de espacios vectoriales $H_q(K, \mathbb{Z}_2) \cong \mathbb{Z}_2^{\beta_q}$, donde β_q es el rango de $H_q(K)$. Notamos además que, gracias a que los coeficientes de este espacio pertenecen a un campo finito, $H_q(K, \mathbb{Z}_2)$ también es finito y posee 2^{β_q} elementos.

Dada esta estructura de espacio vectorial, podemos introducir nuevo lenguaje que no era posible dos secciones atrás. Recordamos que dos q -ciclos que comparten clases de homología son llamados ciclos homólogos. Ahora, si z_1, \dots, z_k son q -ciclos con coeficientes en \mathbb{Z}_2 cuyas clases $[z_1], \dots, [z_k]$ son linealmente independientes en el espacio vectorial $H_1(K, \mathbb{Z}_2)$, decimos que el conjunto de ciclos es *homológicamente independiente*. Notamos que, como consecuencia del uso de coeficientes binarios, dos ciclos $z_1, z_2 \in Z_q(K, \mathbb{Z}_2)$ son homológicamente independientes si los vectores coordenados de sus clases de homología son distintos. Entonces, dos ciclos son homólogos u homológicamente independientes.

Aprovechamos la estructura de espacio vectorial de los grupos de homología con coeficientes en \mathbb{Z}_2 para facilitar su cálculo, empleando las técnicas del álgebra lineal, que veremos en el capítulo 6.

5.4. Bases de homología óptimas

Un conjunto de q -ciclos z_1, \dots, z_{β_q} es una base de homología de $H_q(K, \mathbb{Z}_2)$ si $[z_1], \dots, [z_{\beta_q}]$ es una base de $H_q(K, \mathbb{Z}_2)$. Podemos hablar de bases de $H_q(K, \mathbb{Z}_2)$ ya que estamos seguros de que este es un espacio vectorial. Buscamos optimizar estas bases de acuerdo a una función objetivo.

Como todo complejo simplicial K se encuentra incrustado en un espacio euclidiano, a cada q -simplex σ de K podemos asociarle un real positivo $\ell(\sigma)$ dado por el volumen q -dimensional que encierra σ . Además, podemos extender este volumen a una función $\ell : C_q(K, \mathbb{Z}_2) \rightarrow \mathbb{R}^+$ dada por $\ell(\sigma_1 + \dots + \sigma_k) = \ell(\sigma_1) + \dots + \ell(\sigma_k)$ para todo $\sigma_1 + \dots + \sigma_k \in C_q(K, \mathbb{Z}_2)$.

Definición 5.5. Una base de homología z_1, \dots, z_{β_q} de $H_q(K, \mathbb{Z}_2)$ es óptima si $\sum \ell(z_i)$ es mínimo entre las bases de homología de $H_q(K, \mathbb{Z}_2)$.

Se ha mostrado que el problema de encontrar una base de homología óptima en el caso $q > 1$ es NP-difícil, así que nos concentramos en el caso $q = 1$ [4]. Acá, nos interesa una descripción de los elementos de una base de homología óptima de $H_1(K, \mathbb{Z}_2)$. Aprovechamos esta propiedad para los algoritmos del siguiente capítulo.

Primero, notamos que la colecciones de 0-símplices y 1-símplices de K definen los vértices V_K y aristas E_K de un grafo $G_K = (V_K, E_K)$. De este modo, existe una correspondencia biyectiva entre 1-cadenas de $C_1(K, \mathbb{Z}_2)$ y subgrafos de G_K . Podemos ponderar al grafo G_K con la misma función longitud ℓ anterior, y discutir nuestro problema de optimización empleando la herramienta de teoría de grafos. Usamos la terminología discutida en la sección 2.2.

Empleamos el término ciclo con precaución, pues podemos referirnos a un elemento de $Z_1(K, \mathbb{Z}_2)$ o un subgrafo de G_K . Sin embargo, existe cierto paralelo entre los ciclos de $Z_1(K, \mathbb{Z}_2)$ y los ciclos de G_K . Si $z = x_0x_1 \cdots x_kx_0$ es un ciclo en G_K , su 1-cadena asociada en $C_1(K, \mathbb{Z}_2)$ es un 1-ciclo, pues

$$\begin{aligned} \partial(z) &= \partial \left(x_kx_0 + \sum_{i=0}^{k-1} x_ix_{i+1} \right) = \partial(x_kx_0) + \sum_{i=0}^{k-1} \partial(x_ix_{i+1}) = (x_k + x_0) + \sum_{i=0}^{k-1} (x_i + x_{i+1}) \\ &= (x_k + x_0) + (x_0 + x_k) = 0. \end{aligned}$$

Sin embargo, el converso no es siempre cierto. Un ciclo en $Z_1(K, \mathbb{Z}_2)$ no necesariamente corresponde a un ciclo en G_K . Es más, un ciclo en $Z_1(K, \mathbb{Z}_2)$ no debe corresponder a un camino cerrado en G_K , pues un ciclo en $Z_1(K, \mathbb{Z}_2)$ puede tener dos componentes conexas en G_K .

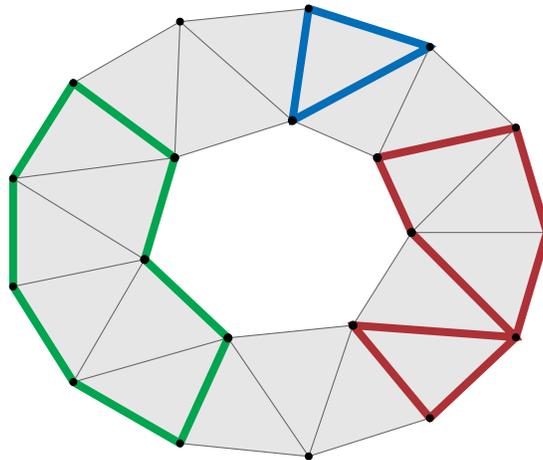


Figura 5.3: Tipos de ciclos en un complejo.

La Figura 5.3 ejemplifica los distintos tipos de ciclos con los que trabajamos. Se presenta un complejo 2-dimensional, donde las regiones grises simbolizan presencia de triángulos. El complejo es homotópicamente equivalente a S^1 , por lo que $H_1(K, \mathbb{Z}) = \mathbb{Z}$. En rojo, se muestra un ciclo de $Z_1(K, \mathbb{Z}_2)$ que no es un ciclo de G_K , ya que contiene un vértice que se visita más de 2 veces. En verde se muestra un ciclo de G_K que no es un ciclo ajustado. Vemos que un requisito para ser un ciclo ajustado es contener todas las aristas del grafo original que unen vértices del ciclo, pues estas aristas son caminos de distancia mínima entre pares de vértices adyacentes. Finalmente, en azul se muestra un ciclo ajustado que no pertenece a una base de homología, pues este ciclo es el borde de un triángulo.

Un ciclo de G_K que puede pertenecer a una base de homología es el ciclo compuesto por los 13 vértices externos del grafo. Como $H_1(K, \mathbb{Z}) = \mathbb{Z}$, estas bases cuentan con un único ciclo. Existen múltiples ciclos que forman una base de homología, pero únicamente el ciclo compuesto por los 7 vértices internos de G_K forma una base de homología óptima.

Buscamos mostrar que los ciclos de una base de homología óptima sí corresponden a ciclos de G_K , que además cumplen con la siguiente característica. Esta propiedad fue mostrada por Erickson y Whittlesey como parte de uno de los algoritmos que mostramos en el capítulo 6 [10].

Definición 5.6. *Un ciclo en G_K es ajustado si, para todo par de vértices u, v del ciclo, este contiene un camino de distancia mínima en G_K de u a v .*

Lema 5.1. *Todo ciclo de una base de homología óptima de $H_1(K, \mathbb{Z}_2)$ es un ciclo ajustado de G_K .*

Demostración. Sea $z_1, z_2, \dots, z_{\beta_1}$ una base de homología óptima de $H_1(K, \mathbb{Z}_2)$. Iniciamos mostrando que cada z_i corresponde a un subgrafo conexo de G_K . Por el absurdo, supóngase que algún z_i no cumple este requisito, digamos, z_1 . Sean w_1, \dots, w_k las componentes conexas de z_1 en G_K , donde notamos que $z_1 = \sum w_i$. Como los w_i no comparten vértices, tenemos $\partial(w_i) = 0$ para cada w_i , con lo cual los w_i son ciclos. Vemos que al menos un w_i es homológicamente independiente a z_2, \dots, z_k , pues de otro modo $[z_1] = [w_1 + \dots + w_k] = [w_1] + \dots + [w_k]$ no sería linealmente independiente de $[z_2], \dots, [z_{\beta_1}]$. Sin pérdida de generalidad, sea w_1 homológicamente independiente a z_2, \dots, z_k . Se sigue que w_1, z_2, \dots, z_k es una base de homología de $H_1(K, \mathbb{Z}_2)$ con longitud menor que $z_1, z_2, \dots, z_{\beta_1}$, una contradicción. Por lo tanto, cada ciclo en $z_1, z_2, \dots, z_{\beta_1}$ es un subgrafo conexo de G_K .

Para un z_i fijo, notamos que cada vértice de z_i debe ser incidente a un número par de aristas distintas en z_i , de modo que la suma de los bordes de las aristas de z_i se anulen y tengamos $\partial(z_i) = 0$. Luego, todos los vértices de z_i poseen grado par en z_i . Por el corolario 2.1, cada z_i es la unión de ciclos de G_K que no comparten aristas a pares. En el lenguaje de 1-cadenas de K , z_i es suma de ciclos de G_K .

Ahora, suponemos por contradicción que algún z_i no es un ciclo de G_K , digamos, z_1 de nuevo. Entonces, z_1 se descompone como la suma de dos o más ciclos w_1, \dots, w_k de G_K que no comparten aristas a pares. De nuevo, podemos asumir que w_1 es homológicamente independiente a z_2, \dots, z_k , pues de otro modo contradecimos que z_1, \dots, z_k es una base de homología. Obtenemos que w_1, z_2, \dots, z_k es una base de homología de longitud menor que z_1, \dots, z_k , un absurdo. Entonces, cada z_i es un ciclo de G_K .

Finalmente, supóngase que z_1 no es ajustado. Entonces, existen dos vértices u y v de z_1 tales que z_1 no contiene el camino más corto γ de u a v en G_K . Notamos que u y v dividen a z_1 en dos caminos simples α y β tales que $z_1 = \alpha + \beta$. Consideramos a los ciclos $z'_1 = \alpha + \gamma$ y $z''_1 = \beta + \gamma$ de $Z_1(K, \mathbb{Z}_2)$, donde vemos que $z_1 = z'_1 + z''_1$. Como en las pruebas anteriores, esto produce una contradicción. Por lo tanto, cada z_i es un ciclo ajustado de G_K . \square

Algoritmos de identificación de grupos de homología

En este capítulo, describimos algoritmos para la identificación de grupos de homología de un complejo simplicial, los cuales describimos en el capítulo 5. El primero de estos algoritmos busca una base de homología arbitraria, para q -ciclos de cualquier dimensión. Por otra parte, el segundo y tercer algoritmo presentan una base de homología óptima en el caso 1-dimensional, donde el complejo estudiado es conexo. Para describir estos procesos, nos apoyamos tanto de la teoría vista en el capítulo 5 como de otras áreas de la matemática. De especial uso son el álgebra lineal, la teoría de grafos y la teoría de matroides.

Iniciamos nuestra discusión en la primera sección describiendo las estructuras de datos utilizadas para almacenar símplexes y cadenas, entre otros objetos. Luego, la segunda, tercera y cuarta sección detallan los algoritmos mencionados. Por último, la quinta sección compara la complejidad de tiempo de ejecución de estos algoritmos.

6.1. Estructuras de datos

Describimos las estructuras de datos que utilizamos para representar objetos de interés dentro de los algoritmos. En específico, dado un complejo simplicial K , deseamos almacenar sus símplexes, sus cadenas, los vectores coordenados asociados a estas cadenas y los operadores borde.

- *Símplexes*: Iniciamos etiquetando los vértices de K con números enteros de 0 a $n_0 - 1$. Para futura referencia, llamamos n_q a la cantidad de q -símplexes de K . Representamos a cualquier q -símplex de K a través de una lista ordenada (v_0, \dots, v_q) de longitud $q + 1$, cuyas entradas constituyen las etiquetas de los vértices que definen al q -símplex. Dado que la orientación de un símplex no es relevante al tratar homología con coeficientes en \mathbb{Z}_2 , no nos preocupamos por el orden de esta lista. Como convención, ordenamos las etiquetas de los vértices de un símplex según su orden como números enteros.
- *Cadenas*: Recordamos que una q -cadena es una suma formal de q -símplexes. Denotamos el espacio de las q -cadenas de K con coeficientes en \mathbb{Z}_2 por $C_q(K, \mathbb{Z}_2)$. Notamos que una cadena

$\sigma_1 + \dots + \sigma_k$ en $C_q(K, \mathbb{Z}_2)$ puede asociarse a un único subconjunto $\{\sigma_1, \dots, \sigma_k\}$ del conjunto de q -símplices de K , gracias a los coeficientes binarios y la conmutatividad en $C_q(K, \mathbb{Z}_2)$. Por lo tanto, representamos una q -cadena $\sigma_1 + \dots + \sigma_k$ como una lista ordenada de longitud k , cuyas entradas son q -símplices en su representación de listas ordenadas de longitud $q + 1$. De nuevo, no nos preocupamos por el orden de estos q -símplices en la lista de longitud k .

En la práctica, en lugar de representar una q -cadena $\sigma_1 + \dots + \sigma_k$ con una lista de k sub-listas de $q + 1$ entradas cada una, empaquetamos esta lista de listas en una matriz de $k \times (q + 1)$. Por ejemplo, si deseamos almacenar el borde del triángulo $v_0v_1v_2$, dado por $v_1v_2 + v_0v_2 + v_0v_1$, empleamos la matriz

$$\partial_2(v_0, v_1, v_2) = \begin{pmatrix} v_1 & v_2 \\ v_0 & v_2 \\ v_0 & v_1 \end{pmatrix}.$$

- *Vectores coordenados*: Como alternativa a la estructura anterior, podemos representar cadenas en $C_q(K, \mathbb{Z}_2)$ con sus vectores coordenados gracias a que $C_q(K, \mathbb{Z}_2)$ es finito-dimensional. Utilizamos como base canónica de $C_q(K, \mathbb{Z}_2)$ el listado ordenado $\sigma_1, \dots, \sigma_{n_q}$ de q -símplices de K . Vemos que $\sigma \in C_q(K, \mathbb{Z}_2)$ tiene vector coordenado $(c_1, \dots, c_{n_q}) \in \mathbb{Z}_2^{n_q}$ si, y solo si,

$$\sigma = \sum_{i=1}^{n_q} c_i \sigma_i,$$

Obtenemos el vector coordenado de una q -cadena σ a partir de la matriz que la almacena. Iteramos sobre la base canónica $\sigma_1, \dots, \sigma_{n_q}$ y hacemos $c_i = 1$ si σ_i se encuentra en una de las filas de la matriz que almacena a σ , y $c_i = 0$ en caso contrario. Esta revisión se beneficia de que los símplices se almacenan con una estructura ordenada. Una implementación más eficiente para el cálculo de vectores coordenados involucra una tabla hash, una estructura de datos que asocia cada elemento de la base de $C_q(K, \mathbb{Z}_2)$ con su posición en esta base.

- *Operadores borde*: Estos operadores son transformaciones lineales entre espacios vectorial finito-dimensionales, por lo que podemos representarlos con matrices. En concreto, si n_q y n_{q-1} son las dimensiones de $C_q(K, \mathbb{Z}_2)$ y $C_{q-1}(K, \mathbb{Z}_2)$, el operador borde $\partial_q : C_q(K, \mathbb{Z}_2) \rightarrow C_{q-1}(K, \mathbb{Z}_2)$ se representa con una matriz $\mathbf{\partial}_q$ de $n_{q-1} \times n_q$. Una vez contamos con bases canónicas para los espacios de cadenas, construimos $\mathbf{\partial}_q$ colocando en su j -ésima columna el vector coordenado de la $(q - 1)$ -cadena $\partial_q(\sigma_j)$, donde σ_j es el j -ésimo elemento de la base de $C_q(K, \mathbb{Z}_2)$.

6.2. Algoritmo de base de homología arbitraria

Este algoritmo calcula una base de homología de $H_q(K, \mathbb{Z}_2)$ de un complejo simplicial K . Como describimos en la sección 5.4, un conjunto de q -ciclos en $Z_q(K, \mathbb{Z}_2)$ es una base de homología si el conjuntos de sus clases de equivalencia en $H_q(K, \mathbb{Z}_2) = Z_q(K, \mathbb{Z}_2)/B_q(K, \mathbb{Z}_2)$ es una base del espacio cociente. La entrada del algoritmo son tres listas con los símplices de K de dimensiones $q + 1$, q y $q - 1$. Empaquetamos las listas de entrada en las matrices \mathbf{S}_{q+1} , \mathbf{S}_q y \mathbf{S}_{q-1} , al estilo de la estructura de datos de cadenas descrita en la sección anterior. Alternativamente, la entrada del algoritmo puede ser el complejo simplicial K en su totalidad, en cuyo caso pueden calcularse todos los grupos de homología de K .

La estrategia del algoritmo simula la construcción típica de una base de un espacio vectorial cociente, como se presenta en un texto de álgebra lineal. Conocemos que el q -ésimo grupo de homología $H_q(K, \mathbb{Z}_2)$ resulta ser un espacio vectorial, dado por el cociente del espacio de q -ciclos $Z_q(K, \mathbb{Z}_2) = \ker \partial_q$ entre el subespacio de q -bordes $B_q(K, \mathbb{Z}_2) = \text{Im } \partial_{q+1}$. Entonces, el cálculo de $H_q(K, \mathbb{Z}_2)$ se remonta al estudio de tres espacios de cadenas y dos operadores borde entre ellos, dados por

$$\partial_{q+1} : C_{q+1}(K, \mathbb{Z}_2) \rightarrow C_q(K, \mathbb{Z}_2) \quad \text{y} \quad \partial_q : C_q(K, \mathbb{Z}_2) \rightarrow C_{q-1}(K, \mathbb{Z}_2).$$

Construimos conjuntos generadores de $Z_q(K, \mathbb{Z}_2)$ y $B_q(K, \mathbb{Z}_2)$ a partir de las representaciones matriciales ∂_{q+1} y ∂_q de estos operadores. De inmediato vemos que las columnas de ∂_{q+1} generan a $\text{Im } \partial_{q+1}$. Para encontrar una base de $\ker \partial_q$, calculamos la forma reducida escalonada por renglones de ∂_q utilizando eliminación gaussiana. Empaquetamos los vectores coordenados de la base de $\ker \partial_q$ encontrada en las columnas de una matriz \mathbf{Z} .

Ahora procedemos con el cálculo de la base de homología. Del álgebra lineal, conocemos que una base del cociente de interés $Z_q(K, \mathbb{Z}_2)/B_q(K, \mathbb{Z}_2)$ puede obtenerse de los generadores de $Z_q(K, \mathbb{Z}_2)$ y $B_q(K, \mathbb{Z}_2)$ según el teorema de rango y nulidad. Construimos la matriz aumentada $[\partial_{q+1} \mid \mathbf{Z}]$, donde recordamos que las columnas de ∂_{q+1} generan a $B_q(K, \mathbb{Z}_2)$. Leemos las columnas de $[\partial_{q+1} \mid \mathbf{Z}]$ de izquierda a derecha, y anotamos los índices de aquellas columnas que son linealmente independientes a las anteriores. Las columnas anotadas en ∂_{q+1} forman una base de $B_q(K, \mathbb{Z}_2)$, mientras que las columnas anotadas en \mathbf{Z} forman una base del cociente. Estas columnas de \mathbf{Z} representan la base de homología buscada.

En la práctica, verificar en cada iteración que cada columna en $[\partial_{q+1} \mid \mathbf{Z}]$ es independiente de las columnas a su izquierda es costoso. Resulta más económico procesar todas las columnas al mismo tiempo, empleando eliminación gaussiana. Sea $[\partial'_{q+1} \mid \mathbf{Z}']$ la forma reducida escalonada por renglones de $[\partial_{q+1} \mid \mathbf{Z}]$. Entonces, las columnas con pivote de $[\partial'_{q+1} \mid \mathbf{Z}']$ son precisamente aquellas que son linealmente independientes a las columnas a su izquierda, tanto en la matriz original como en la reducida. Entonces, la salida del algoritmo se traduce a aquellas columnas de \mathbf{Z} donde la columna correspondiente en \mathbf{Z}' tiene un pivote.

Resumimos los pasos discutidos en el siguiente algoritmo.

Algoritmo 2 HomArb

Entrada: Listas \mathbf{S}_{q+1} , \mathbf{S}_q y \mathbf{S}_{q-1} de símlices de K de dimensiones $q+1$, q y $q-1$.

Salida: Vectores coordenados de ciclos de una base de homología de $H_q(K, \mathbb{Z}_2)$.

- 1: Construir las matrices ∂_{q+1} y ∂_q .
 - 2: Construir la matriz \mathbf{Z} cuyas columnas forman una base de $\ker \partial_q$.
 - 3: Aplicar eliminación gaussiana a $[\partial_{q+1} \mid \mathbf{Z}]$ para obtener $[\partial'_{q+1} \mid \mathbf{Z}']$.
 - 4: **retornar** Columnas de \mathbf{Z} que tienen pivote en \mathbf{Z}' .
-

6.3. Algoritmo de base de homología óptima ingenua

Este algoritmo calcula una base de homología óptima para $H_1(K, \mathbb{Z}_2)$. Como en la sección 5.4, una base de homología es óptima si la suma de sus longitudes respecto a una función $\ell : C_q(K, \mathbb{Z}_2) \rightarrow \mathbb{R}^+$ es mínima. Nos restringimos al caso 1-dimensional por la dificultad de este problema en dimensiones más altas [4]. A su vez, nos restringimos a complejos conexos, pues el proceso emplea extensivamente árboles de expansión. Este algoritmo surgió en el contexto de la teoría de grafos y origina de una publicación por Horton [14]. Posteriormente, Erickson y Whittlesey adaptaron dicho resultado al lenguaje de la homología [10].

La entrada de este algoritmo también está compuesta por tres listas de símlices de K , dadas por los triángulos, aristas y vértices de K . Sin embargo, podemos deducir la lista de vértices a partir de la lista de aristas conociendo que K es conexo. Empaquetamos las listas de triángulos y aristas en \mathbf{F}_K y \mathbf{E}_K como en el algoritmo HomArb. Por último, requerimos de una lista con las longitudes de las aristas de K en \mathbb{R}^3 .

La idea principal detrás de este método se encuentra en identificar nuestro problema de optimización como un matroide ponderado y aplicar una estrategia voraz. Puede revisarse la sección 2.3 para un tratamiento detallado de los elementos básicos de la teoría de matroides. Para hacer esta identificación, primero construimos una nueva función w que nos permita transformar nuestro

problema de minimización en uno de maximización. Sea $\ell_{\text{máx}}$ una constante mayor a la máxima longitud de una arista de K . Para toda arista σ de K , definimos a la función de valores reales positivos $w(\sigma) = \ell_{\text{máx}} - \ell(\sigma)$. Extendemos w a $C_1(K, \mathbb{Z}_2)$ vía $w(\sigma_1 + \dots + \sigma_k) = w(\sigma_1) + \dots + w(\sigma_k)$ para todo $\sigma_1 + \dots + \sigma_k \in C_1(K, \mathbb{Z}_2)$. Con esta función, caracterizamos a las bases de homología óptimas como bases de homología z_1, \dots, z_{β_1} tales que $w(z_1) + \dots + w(z_{\beta_1})$ es máximo.

Ahora, identificamos el problema de encontrar una base de homología óptima como el problema de encontrar un subconjunto independiente óptimo en un matroide ponderado $M = (\Pi, \mathcal{I})$. Sea $\Pi = Z_1(K, \mathbb{Z}_2)$ e \mathcal{I} la colección de subconjuntos de Π dados por ciclos homológicamente independientes entre sí. Es decir, $\{z_1, \dots, z_k\} \in \mathcal{I}$ si, y solo si, $\{[z_1], \dots, [z_k]\}$ es linealmente independiente en $H_1(K, \mathbb{Z}_2)$. Dado que $H_1(K, \mathbb{Z}_2)$ es un espacio vectorial, puede verificarse que $M = (\Pi, \mathcal{I})$ satisface la definición 2.16 de matroide. Si dotamos a M de la función peso w , obtenemos un matroide ponderado cuyos subconjuntos independientes óptimos corresponden a bases de homología óptimas. Por el teorema 2.7, conocemos que el algoritmo voraz en M ofrece una solución al problema.

6.3.1. Construcción del matroide

Analizamos al problema de listar todos los elementos de $\Pi = Z_1(K, \mathbb{Z}_2)$. Ingenuamente, pudiéramos crear una lista de todas las 1-cadenas de K y verificar cuáles son 1-ciclos, calculando el borde de cada una de ellas. No obstante, de inmediato tenemos un inconveniente, pues existen 2^{n_1} 1-cadenas en K . Aunque podamos generar un listado de todos los 1-ciclos de K de otro modo, la longitud de tal listado sería de orden exponencial respecto a $\dim Z_1(K, \mathbb{Z}_2)$. En búsqueda de un algoritmo con complejidad de tiempo polinomial, restringimos Π a un subconjunto propio de $Z_1(K, \mathbb{Z}_2)$, pero que aún contenga al menos una base de homología óptima.

Nos apoyamos de la teoría de grafos para esta construcción. Puede revisarse la sección 2.2 para un cubrimiento básico del tópico. Sea $G_K = (V_K, E_K)$ el grafo conexo dado por los vértices y aristas de K . Para cada par de vértices $u, v \in V_K$, definimos a $G_K[u, v]$ como el camino más corto en G_K de u a v según la función longitud ℓ . En caso de que exista más de un camino de longitud mínima de u a v , seleccionamos como $G_K[u, v]$ al menor camino según el orden lexicográfico de sus listas de vértices.

Primero, para cada vértice $v \in V_K$, construimos un conjunto Π_v de ciclos de G_K del siguiente modo. Consideramos al grafo T_v cuyas aristas están dadas por todos los caminos $G_K[v, v']$ con $v' \in V_K$. Se sigue que T_v es un árbol de distancia más corta con raíz en v , el cual podemos construir con el algoritmo de Dijkstra. En la Figura 6.1, incluimos una ilustración de uno de estos árboles, donde K es una triangulación del doble toro y v es el vértice resaltado.

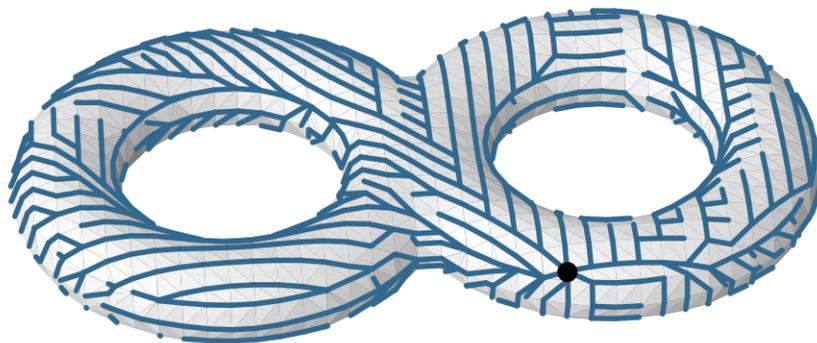


Figura 6.1: Árbol de distancia más corta en la triangulación de un doble toro.

Segundo, para cada arista $e \in E_K - E(T_v)$, consideramos el ciclo $\varphi(T_v, e)$ de G_K dado por la suma de e con el único camino simple en T_v que une los vértices de e . Este camino existe y es único dado que T_v es un árbol de expansión. Si $e = u_1u_2$, obtenemos este camino vía $G_K[v, u_1] + G_K[v, u_2]$. Definimos a Π_v como el conjunto de estos ciclos, es decir,

$$\Pi_v = \{\varphi(T_v, e) : e \in E_K - E(T_v)\}, \quad \text{para todo } v \in V_K.$$

Finalmente, definimos $\Pi = \bigcup_{v \in V_K} \Pi_v$. Argumentamos que Π es una colección de subconjuntos de $Z_1(K, \mathbb{Z}_2)$ que contiene una base de homología óptima de $H_1(K, \mathbb{Z}_2)$. Es más, mostramos que Π contiene todas estas bases.

Teorema 6.1. *El conjunto de ciclos Π contiene todas las bases de homología óptimas de $H_1(K, \mathbb{Z}_2)$.*

Demostración. Mostramos que Π contiene todos los ciclos ajustados de G_K . Esto es suficiente para obtener el resultado, pues el lema 5.1 garantiza que todo ciclo en una base de homología óptima es un ciclo ajustado de G_K .

Sea $z = v_0v_1 \cdots v_kv_0$ un ciclo ajustado de G_K . Notamos que, para cada v_i en z , existen únicamente dos caminos simples de v_0 a v_i en z , dados por $v_0v_1 \cdots v_{i-1}v_i$ y $v_0v_k \cdots v_{i+1}v_i$. Entonces,

$$G_K[v_0, v_i] = v_0v_1 \cdots v_{i-1}v_i \quad \text{o} \quad G_K[v_0, v_i] = v_0v_k \cdots v_{i+1}v_i.$$

Sea v_j el vértice de índice máximo en z tal que $G_K[v_0, v_j] = v_0v_1 \cdots v_j$. Luego, $G_K[v_0, v_{j+1}] = v_0v_k \cdots v_{j+1}$, con lo cual v_jv_{j+1} no pertenece a T_{v_0} . Entonces, tenemos

$$z = G_K[v_0, v_j] + G_K[v_0, v_{j+1}] + v_jv_{j+1} = \varphi(T_{v_0}, v_jv_{j+1}) \in \Pi_{v_0} \subset \Pi.$$

por lo que $z \in \Pi$. Entonces, todo ciclo ajustado de G_K pertenece a Π . □

Resumimos la construcción de Π en el siguiente algoritmo.

Algoritmo 3 ConstruCiclos

Entrada: Lista E_K de aristas de K .

Salida: Conjunto Π con todos los ciclos ajustados de G_K .

- 1: $\Pi \leftarrow \emptyset$.
 - 2: **para cada** $v \in V_K$ **hacer**
 - 3: Construir árbol de distancia más corta T_v de G_K con raíz en v .
 - 4: **para cada** $v_iv_j \in E_K - E(T_v)$ **hacer**
 - 5: $z = G_K[v, v_i] + G_K[v, v_j] + v_iv_j$.
 - 6: $\Pi \leftarrow \Pi \cup \{z\}$.
 - 7: **retornar** Π .
-

El conjunto Π contiene todos los ciclos ajustados de G_K , aunque contiene otra variedad de ciclos. Para ver esto, consideramos el complejo simplicial de la Figura 6.2. El complejo cuenta con 5 vértices y 6 aristas y habita en \mathbb{R}^2 . En el grafo asociado G_K , ponderado con la función longitud heredada de \mathbb{R}^2 , trazamos el árbol de distancia más corta T_{v_0} . Vemos que $\varphi(T_{v_0}, v_3v_4)$ genera un ciclo en $\Pi_{v_0} \subset \Pi$, dado por $v_3v_1v_0v_2v_4v_3$. No obstante, este ciclo no es ajustado, pues no contiene a $G_K[v_1, v_2] = v_1v_2$.

A pesar de que Π contiene ciclos que nunca pudieran formar parte de una base de homología, Π es mucho más pequeño que la totalidad de $Z_1(K, \mathbb{Z}_2)$. Notablemente, el tamaño de Π es polinomial respecto al tamaño de G_K . Para estimar el tamaño de Π , vemos que cada árbol de expansión T_v debe contener los n_0 vértices de K , con lo cual T_v siempre posee $n_0 - 1$ aristas. Luego, para cada $v \in V_K$ existen $n_1 - (n_0 - 1)$ aristas en E_K que no pertenecen a $E(T_v)$. Entonces, cada Π_v posee $n_1 - n_0 + 1$ ciclos distintos. Como contamos con n_0 conjuntos Π_v , se sigue

$$|\Pi| \leq n_0|\Pi_v| = n_0(n_1 - n_0 + 1). \tag{6.1}$$

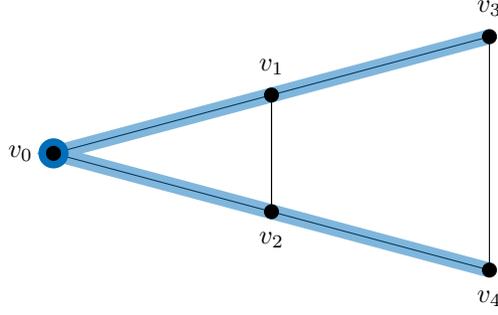


Figura 6.2: Árbol de distancia más corta con raíz en v_0 en un grafo G_K .

6.3.2. Ejecución del algoritmo voraz

Definimos en Π el matroide ponderado $M = (\Pi, \mathcal{I})$, donde \mathcal{I} es la colección de subconjuntos de ciclos en Π homológamente independientes entre sí. Empleamos como función peso la función w discutida antes. Procedemos a aplicar una versión del algoritmo voraz en Π .

Como en el algoritmo HomArb, le delegamos la prueba de independencia del algoritmo voraz a la eliminación gaussiana. Construimos ∂_2 y una matriz $\mathbf{\Pi}$ cuyas columnas son los vectores coordenados de los ciclos en Π . Como en el algoritmo voraz, ordenamos estas columnas de izquierda a derecha según la longitud de los ciclos que representan, ubicando en la izquierda los ciclos que minimizan a ℓ , o bien, que maximizan a w . Notamos que la longitud de cada ciclo $\varphi(T_v, u_1u_2) \in \Pi$ puede calcularse durante su construcción en el algoritmo ConstruCiclos vía

$$\ell(\varphi(T_v, e)) = \ell(G_K[v, u_1]) + \ell(G_K[v, u_2]) - 2\ell(G_K[v, u]) + \ell(u_1u_2), \quad (6.2)$$

donde u es último vértice común en los caminos $G_K[v, u_1]$ y $G_K[v, u_2]$. Esto optimiza el cálculo de las longitudes de ciclos, pues ya conocemos las longitudes de $G_K[v, u_1]$, $G_K[v, u_2]$ y $G_K[v, u]$ por el algoritmo de Dijkstra.

Consideramos la matriz aumentada $[\partial_2 \mid \mathbf{\Pi}]$ y encontramos su forma reducida escalonada por renglones $[\partial'_2 \mid \mathbf{\Pi}']$. Las columnas con pivote en la matriz reducida vuelven a representar aquellas que son linealmente independientes a las columnas a su izquierda. Además, dada la inclusión de la submatriz ∂_2 , las columnas con pivote en $\mathbf{\Pi}'$ representan aquellas cuyas clases de homología son linealmente independientes a las clases de homología de las columnas a su izquierda. Estos son precisamente los ciclos que debe elegir el algoritmo voraz. Por el teorema 2.7, estos ciclos forman una base de homología óptima. Entonces, la salida del algoritmo son las columnas de $\mathbf{\Pi}$ cuyas columnas correspondientes en $\mathbf{\Pi}'$ tiene un pivote.

El pseudocódigo de este algoritmo es el siguiente.

Algoritmo 4 HomOpt1

Entrada: Listas F_K y E_K de triángulos y aristas de K . Lista de longitudes de aristas de K .

Salida: Vectores coordenados de ciclos de una base de homología óptima de $H_1(K, \mathbb{Z}_2)$.

- 1: Construir la matriz ∂_2 .
 - 2: Construir Π con el algoritmo ConstruCiclos.
 - 3: Construir la matriz $\mathbf{\Pi}$ y ordenar sus columnas según ℓ .
 - 4: Aplicar eliminación gaussiana a $[\partial_2 \mid \mathbf{\Pi}]$ para obtener $[\partial'_2 \mid \mathbf{\Pi}']$.
 - 5: **retornar** Columnas de $\mathbf{\Pi}$ que tienen pivote en $\mathbf{\Pi}'$.
-

6.4. Algoritmo de base de homología óptima con anotaciones

En esta sección presentamos una versión más eficiente del algoritmo HomOpt1 para el cálculo de una base de homología óptima de $H_1(K, \mathbb{Z}_2)$. Volvemos a restringirnos a ciclos 1-dimensionales en complejos conexos. En nuestro tratamiento del algoritmo HomOpt1, vimos que el problema de encontrar una base de homología óptima de $H_1(K, \mathbb{Z}_2)$ equivale a encontrar un subconjunto óptimo en un matroide ponderado, lo cual logramos a través de un algoritmo voraz. Acá procedemos con la misma estrategia, utilizando el mismo matroide proveído por el algoritmo ConstruCiclos. La novedad en esta sección consiste en optimizar la revisión de independencia entre ciclos que requiere el algoritmo voraz. La publicación original de este algoritmo se encuentra en [3].

En el algoritmo HomOpt1 aplicamos eliminación gaussiana a una matriz $[\partial_2 | \mathbf{\Pi}]$ para evaluar independencia de ciclos. Por la ecuación 6.1, sabemos que $\mathbf{\Pi}$ cuenta con $n_0(n_1 - n_0 + 1)$ columnas, con posibles repeticiones. Luego, $\mathbf{\Pi}$ es de $n_1 \times n_0(n_1 - n_0 + 1)$. La eliminación gaussiana de esta matriz resulta en el paso más costoso del algoritmo, por lo que buscamos una alternativa. Con este fin, introducimos el concepto de anotación de aristas.

Definición 6.1. *Sea K un complejo simplicial y H una base ordenada de $H_1(K, \mathbb{Z}_2)$. Una anotación de las aristas de K respecto a H es una función $\mathbf{a} : E_K \rightarrow (\mathbb{Z}_2)^{\beta_1}$ tal que, para todo 1-ciclo z , tenemos que $\sum_{e \in z} \mathbf{a}(e)$ es el vector coordenado de $[z]$ respecto a H . Decimos que $\mathbf{a}(e)$ es la anotación de la arista e , y definimos $\mathbf{a}(z) = \sum_{e \in z} \mathbf{a}(e)$ como la anotación del 1-ciclo z .*

En este algoritmo, optimizamos la revisión de independencia entre ciclos calculando primero los vectores coordenados de sus clases de homología. Los vectores coordenados de las clases de homología ofrecen una ventaja computacional sobre los vectores de los ciclos, pues en la práctica estos son mucho menores en longitud. Como la definición 6.1 sugiere, podemos realizar esta tarea a través de una anotación de las aristas de K . Suponiendo que invertimos una pequeña cantidad de tiempo en este proceso, relativo al tiempo que requiere reducir la matriz $[\partial_2 | \mathbf{\Pi}]$, podemos acelerar la identificación de ciclos homológamente independientes. Entonces, procedemos a calcular una anotación de K .

6.4.1. Anotación de aristas

Primero, fijamos un árbol de expansión T en G_K , por ejemplo, un árbol de expansión mínimo. Como en el algoritmo HomOpt1, para toda arista $e \in E_K - E(T)$ consideramos el ciclo $\varphi(T, e)$ dado por la suma de e con el único camino simple en T que une los vértices de e . Cuando $e \in E(T)$ definimos $\varphi(T, e) = 0$. Mostramos que $Z = \{\varphi(T, e) : e \in E_K - E(T)\}$ es una base de $Z_1(K, \mathbb{Z}_2)$ que goza de cierta facilidad para representar ciclos.

Teorema 6.2. *Sea K un complejo simplicial y T un árbol de expansión en G_K . El conjunto de ciclos $Z = \{\varphi(T, e) : e \in E_K - E(T)\}$ es una base de $Z_1(K, \mathbb{Z}_2)$. Además, para todo ciclo $z \in Z_1(K, \mathbb{Z}_2)$ tenemos*

$$z = \sum_{e \in z} \varphi(T, e).$$

Demostración. Probamos que Z es un conjunto linealmente independiente en $Z_1(K, \mathbb{Z}_2)$. Vemos que cada arista $e \in E_K - E(T)$ aparece únicamente en el ciclo $\varphi(T, e)$ de Z . Luego, $\varphi(T, e)$ nunca es combinación lineal del resto de ciclos en Z . Repitiendo este argumento para el resto de ciclos en z , tenemos que Z es un conjunto linealmente independiente en $Z_1(K, \mathbb{Z}_2)$.

Ahora, mostramos que Z genera a $Z_1(K, \mathbb{Z})$. Sea $z \in Z_1(K, \mathbb{Z}_2)$. Vemos que, como $\partial(z) = 0$, cada vértice de z es adyacente a un número par de vértices de z . Entonces, si v es un vértice fijo de V_K , la suma de los $T[v, u_1] + T[v, u_2]$ sobre todos los $u_1 u_2 \in z$ se anula. Por lo tanto,

$$z = \sum_{u_1 u_2 \in z} u_1 u_2 = \sum_{u_1 u_2 \in z} (T[v, u_1] + T[v, u_2] + u_1 u_2) = \sum_{u_1 u_2 \in z} \varphi(T, u_1 u_2).$$

Por lo tanto, los ciclos en Z son un conjunto generador de $Z_1(K, \mathbb{Z}_2)$. En conclusión, Z es una base de $H_1(K, \mathbb{Z}_2)$. En particular, vemos que $\dim Z_1(K, \mathbb{Z}_2) = |Z| = n_1 - n_0 + 1$. \square

Segundo, una vez contamos con Z , una base de $Z_1(K, \mathbb{Z}_2)$, identificamos en ella una base de homología como en los algoritmos anteriores. Respecto a esta base de homología es que definiremos la anotación de aristas. Construimos las matrices \mathcal{D}_2 y \mathbf{Z} , donde las columnas de \mathbf{Z} son vectores coordenados de ciclos en Z . Consideramos la matriz $[\mathcal{D}_2 \mid \mathbf{Z}]$ y su forma reducida escalonada por renglones $[\mathcal{D}'_2 \mid \mathbf{Z}']$. Las columnas de \mathcal{D}_2 que resultan con pivote en \mathcal{D}'_2 forman una base de $B_1(K, \mathbb{Z}_2)$, mientras que las columnas de \mathbf{Z} que resultan con pivote en \mathbf{Z}' forman una base de homología H .

Tercero, para cada $e \in E_K$, definimos la anotación $\mathbf{a}(e)$ como el vector coordenado de $[\varphi(T, e)]$ respecto a H . Si $e \in E(T)$, vemos que $\varphi(T, e) = 0$ y $\mathbf{a}(e)$ es el vector nulo. Para ciclos en Z es claro que la definición 6.1 de anotación se cumple, pues $\mathbf{a}(\varphi(T, e)) = \mathbf{a}(e)$ gracias a que el resto de aristas de $\varphi(T, e)$ tienen anotación nula. Ahora, si z es un ciclo cualquiera de $Z_1(K, \mathbb{Z}_2)$ tenemos $\mathbf{a}(z) = \sum_{e \in z} \mathbf{a}(e) = \sum_{e \in z} \mathbf{a}(\varphi(T, e))$. Por el teorema 6.2, se sigue que $\mathbf{a}(z)$ es el vector coordenado de $[z]$ respecto a H . Por lo tanto, \mathbf{a} es una anotación de aristas.

Para calcular estas anotaciones nos asistimos de la forma reducida escalonada por renglones $[\mathcal{D}'_2 \mid \mathbf{Z}']$ empleada para identificar H . Las primeras $|Z|$ filas de \mathbf{Z}' indican los vectores coordenados de los ciclos de Z , pero esta vez respecto a la base de $Z_1(K, \mathbb{Z}_2)$ definida por las columnas con pivote de $[\mathcal{D}'_2 \mid \mathbf{Z}']$. Entonces, las filas $|Z| - \beta_1, \dots, |Z|$ de \mathbf{Z}' indican los vectores coordenados de las clases de homología de los ciclos de Z respecto a H . Esta sub-matriz de $\beta_1 \times |Z|$, la cual llamamos \mathbf{A} , contiene las anotaciones de las aristas $e \in E_K - E(T)$ empaquetadas en sus columnas. Insertamos columnas nulas en \mathbf{A} en posiciones donde $e \in E(T)$ hasta que \mathbf{A} sea de $\beta_1 \times n_1$.

El algoritmo resumido para calcular anotaciones de aristas de K es el siguiente:

Algoritmo 5 AnotAristas

Entrada: Listas \mathbf{F}_K y \mathbf{E}_K de triángulos y aristas de K .

Salida: Anotaciones de aristas de K , como lista de n_1 vectores en $\mathbb{Z}_2^{\beta_1}$.

- 1: Construir la matriz \mathcal{D}_2 .
 - 2: Construir árbol de expansión mínimo T de G_K .
 - 3: Construir la matriz \mathbf{Z} cuyas columnas son los ciclos $\varphi(T, e)$ con $e \in E_K - E(T)$.
 - 4: Aplicar eliminación gaussiana a $[\mathcal{D}_2 \mid \mathbf{Z}]$ para obtener $[\mathcal{D}'_2 \mid \mathbf{Z}']$.
 - 5: Asignar las filas $|Z| - \beta_1, \dots, |Z|$ de \mathbf{Z}' a la matriz \mathbf{A} .
 - 6: Insertar columnas nulas en \mathbf{A} en posiciones donde $e \in E(T)$.
 - 7: **retornar** \mathbf{A} .
-

6.4.2. Otra ejecución del algoritmo voraz

Como en el algoritmo HomOpt1, construimos un conjunto de ciclos Π que contiene una base de homología con el algoritmo ConstruCiclos. Definimos en Π el mismo matroide ponderado $M = (\Pi, \mathcal{I})$ con función peso w , donde \mathcal{I} es la colección de ciclos en Π homológamente independientes entre sí. Volvemos a aplicar el algoritmo voraz en este matroide, pero modificamos la prueba de independencia entre ciclos aprovechándonos de la anotación de aristas \mathbf{a} calculada.

Para cada ciclo $z \in \Pi$, obtenemos el vector coordenado de $[z]$ respecto a una base de homología común calculando la anotación $\mathbf{a}(z)$ de cada ciclo. Colocamos estos vectores en las columnas de una matriz \mathbf{B} de $\beta_1 \times n_0(n_1 - n_0 + 1)$. Siguiendo la guía del algoritmo voraz, ordenamos estas columnas de izquierda a derecha según la longitud de los ciclos que representan, ubicando en la izquierda los ciclos que maximizan a w . Reciclamos el método visto en el algoritmo HomOpt1 para medir la longitud de estos ciclos de forma más eficiente, usando la expresión 6.2.

Aplicamos eliminación gaussiana a \mathbf{B} para obtener su forma reducida \mathbf{B}' . Las columnas de \mathbf{B}' con pivotes indican los ciclos cuyas clases de homología son linealmente independientes a las clases de homología de los ciclos a su izquierda. Estos son los ciclos que elige el algoritmo voraz, los cuales forman una base de homología óptima por el teorema 2.7. Entonces, la salida del algoritmo son los ciclos de Π cuyas columnas correspondientes en \mathbf{B}' tienen un pivote.

En resumen, el algoritmo de identificación de bases de homología óptimas empleando anotaciones es el siguiente:

Algoritmo 6 HomOpt2

Entrada: Listas \mathbf{F}_K y \mathbf{E}_K de triángulos y aristas de K . Lista de longitudes de aristas de K .

Salida: Vectores coordenados de ciclos de una base de homología óptima de $H_1(K, \mathbb{Z}_2)$.

- 1: Construir matriz \mathbf{A} de anotaciones de aristas con el algoritmo AnotAristas.
 - 2: Construir Π con el algoritmo ConstruCiclos.
 - 3: Construir matriz \mathbf{B} de anotaciones de ciclos de Π y ordenar sus columnas según w .
 - 4: Aplicar eliminación gaussiana a \mathbf{B} para obtener \mathbf{B}' .
 - 5: **retornar** Ciclos de Π que tienen pivote en \mathbf{B}' .
-

Por el momento, calculamos la anotación de cada ciclo en Π individualmente, mediante la suma de las anotaciones de sus aristas. No obstante, podemos usar un procedimiento análogo al empleado para medir longitudes de ciclos, ilustrado con la expresión 6.2. De este modo, podemos calcular simultáneamente todas las anotaciones de los ciclos $\varphi(T_v, u_1u_2)$ para un vértice v dado, a medida que los construimos con el algoritmo ConstruCiclos.

Sea $v \in V_K$ un vértice fijo. Para cada $u \in V_K$, definimos $\mathbf{a}_v(u)$ como la suma de las anotaciones de las aristas en $G_K[v, u]$, donde $\mathbf{a}_v(v) = 0$. Notamos que, si u' es el padre de u en T_v , dado por el vértice adyacente a u en $G_K[v, u]$, entonces $\mathbf{a}_v(u) = \mathbf{a}_v(u') + \mathbf{a}(uu')$. Así, podemos calcular recursivamente los $\mathbf{a}_v(u)$ para todo $u \in V_K$, procurando que siempre visitemos a u' antes que u . Garantizamos esto visitando los vértices en un orden de búsqueda por amplitud. Con los $\mathbf{a}_v(u)$ calculados, la anotación del ciclo $\varphi(T_v, u_1u_2)$ está dada por

$$\mathbf{a}(\varphi(T_v, u_1u_2)) = \mathbf{a}(G_K[v, u_1]) + \mathbf{a}(G_K[v, u_2]) + \mathbf{a}(u_1u_2) = \mathbf{a}_v(u_1) + \mathbf{a}_v(u_2) + \mathbf{a}(u_1u_2). \quad (6.3)$$

6.5. Complejidad de tiempo de ejecución

Los algoritmos de este capítulo ofrecen tres métodos para identificar grupos de homología simplicial. Es más, los últimos dos ofrecen bases de homología óptima, donde se optimiza la longitud de los ciclos que representan una base de $H_1(K, \mathbb{Z}_2)$. En esta sección comparamos la complejidad de tiempo de ejecución de cada uno de estos algoritmos.

Empleamos la notación “O-grande” usual para clasificar esta complejidad de tiempo y la denotamos por \mathcal{O} . Comparamos la complejidad con el tamaño de las listas de entrada de los algoritmos. En el caso de HomArb, estas listas poseen longitud n_{q+1} , n_q y n_{q-1} . El resto de algoritmos pide listas de tamaño n_2 y n_1 , o en el caso del algoritmo ConstruCiclos, solamente una lista de tamaño n_1 . Para facilitar el lenguaje, utilizamos un único n que mayormente la suma de las longitudes de las listas de entrada.

- *Eliminación gaussiana:* Sea M una matriz de $m_1 \times m_2$. Notamos que, como máximo, contamos con $\min(m_1, m_2)$ pivotes en esta matriz. Si un pivote se encuentra en la entrada (i_0, j_0) de M , se requiere modificar todas las entradas (i, j) de M fuera de la fila i_0 donde la entrada (i, j_0) sea distinta de 0. Como máximo, deben realizarse $(m_1 - 1)m_2$ de estas modificaciones. Por lo tanto, la eliminación gaussiana de M tiene complejidad $\mathcal{O}(m_1m_2 \min(m_1, m_2))$.

- *HomArb*: Iniciamos calculando las matrices borde \mathfrak{d}_{q+1} y \mathfrak{d}_q , de dimensiones $n_q \times n_{q+1}$ y $n_{q-1} \times n_q$. Cada vector coordenado en las columnas de estas matrices se calcula en $\mathcal{O}(n)$, con lo cual ambas matrices se generan en $\mathcal{O}(n^2)$. El resto del algoritmo se resume en aplicar eliminación gaussiana a las matrices \mathfrak{d}_q y $[\mathfrak{d}_{q+1} \mid \mathbf{Z}]$. Dado que estas matrices tienen dimensiones del orden $n \times n$, este paso se logra en $\mathcal{O}(n^3)$. Entonces, el algoritmo es $\mathcal{O}(n^3)$.
- *ConstruCiclos*: El bucle exterior de este algoritmo se ejecuta n_0 veces, una por cada vértice $v \in V_K$. En cada iteración, el árbol de expansión T_v se construye en $\mathcal{O}(n_1 \log n_0)$ con el algoritmo de Dijkstra [5]. Por otra parte, el bucle interior se efectúa $n_1 - n_0 + 1$ veces, una por cada arista $u_i u_j \in E_K - E(T_v)$. Para cada una de estas aristas se calcula el camino simple en T_v que une u_i con u_j . A lo sumo, este camino posee n_1 aristas y se encuentra en $\mathcal{O}(n_1)$. En resumen, el algoritmo se ejecuta en $\mathcal{O}(n_0(n_1 \log n_0 + (n_1 - n_0 + 1)n_1))$, o bien, $\mathcal{O}(n^3)$.
- *HomOpt1*: Construimos a \mathfrak{d}_2 en $\mathcal{O}(n^2)$ como en el algoritmo HomArb y a Π en $\mathcal{O}(n^3)$ con el algoritmo ConstruCiclos. Notamos que Π posee $n_0(n_1 - n_0 + 1)$ elementos, cantidad del orden n^2 . Luego, los vectores coordenados de los ciclos en Π se calculan en $\mathcal{O}(n^3)$. Las longitudes de los ciclos se calculan en tiempo constante durante su construcción, y el orden de los mismos se da en $\mathcal{O}(n^2 \log n)$. Finalmente, se aplica eliminación gaussiana a la matriz $[\mathfrak{d}_2 \mid \Pi]$ de $n_1 \times (n_2 + n_0(n_1 - n_0 + 1))$. Las dimensiones de la matriz aumentada son del orden $n \times n^2$, lo que implica que su eliminación gaussiana se logra en $\mathcal{O}(n^4)$. Siendo este el paso más costoso del algoritmo, el proceso en su totalidad se ejecuta en $\mathcal{O}(n^4)$.
- *AnotAristas*: Calculamos \mathfrak{d}_2 en $\mathcal{O}(n^2)$ y obtenemos un árbol de expansión mínimo con el algoritmo de Kruskal en $\mathcal{O}(n \log n)$ [5]. Los $n_1 - n_0 + 1$ ciclos de Z se encuentran en $\mathcal{O}(n^2)$, pues cada ciclo es de la forma $T[u_1, u_2] + u_1 u_2$ y cada camino $T[u_1, u_2]$ se obtiene en $\mathcal{O}(n)$. Luego, formamos la matriz \mathbf{Z} en $\mathcal{O}(n^2)$. El último paso costoso consiste en aplicar eliminación gaussiana a $[\mathfrak{d}_2 \mid \mathbf{Z}]$. Dado que esta matriz es de $n_1 \times (n_2 + n_1 - n_0 + 1)$, se alcanza su forma reducida escalonada por renglones en $\mathcal{O}(n^3)$. Esta es también la complejidad del algoritmo.
- *HomOpt2*: Ejecutamos los algoritmos AnotAristas y ConstruCiclos en $\mathcal{O}(n^3)$. Calculamos las anotaciones de los ciclos de Π como en la expresión 6.3. Para cada $v \in V_K$ encontramos los $\mathbf{a}_v(u)$ en $\mathcal{O}(\beta_1 n)$, lo cual no modifica la complejidad de ConstruCiclos. Con esto obtenemos las anotaciones de los ciclos en tiempo constante durante su construcción, al igual que sus longitudes. Luego, ordenamos los ciclos de Π en $\mathcal{O}(n^2 \log n)$. Por último, aplicamos eliminación gaussiana a la matriz de anotaciones de ciclos \mathbf{B} , cuyas dimensiones son del orden $\beta_1 \times n^2$. Tomando β_1 como pequeño en relación a n , esto se alcanza en $\mathcal{O}(\beta_1^2 n^2)$. En conclusión, el algoritmo es $\mathcal{O}(n^3)$.

Resumimos la complejidad de los tres algoritmos principales para el cálculo de bases de homología, HomArb, HomOpt1 y HomOpt2, en el Cuadro 6.1.

Algoritmo	Complejidad
HomArb	$\mathcal{O}(n^3)$
HomOpt1	$\mathcal{O}(n^4)$
HomOpt2	$\mathcal{O}(n^3)$

Cuadro 6.1: Complejidad de tiempos de ejecución de algoritmos de cálculo de homología.

Implementamos todos los algoritmos vistos en el capítulo pasado en el lenguaje de programación Python, versión 3.11. Para las tareas de reducción a forma escalonada por renglones y cálculo de kernels se desarrolló una versión de eliminación gaussiana en \mathbb{Z}_2 . Esto se decidió porque las librerías estándar disponibles no operan en campos finitos, si no que en \mathbb{R} . Por otro lado, en lugar de implementar “desde cero” los algoritmos de Dijkstra y Kruskal, importamos las respectivas funciones de la librería `scipy`. Las principales librerías usadas y sus roles fueron:

- `numpy`: representación de matrices y funciones matemáticas básicas.
- `scipy`: algoritmos de Dijkstra y Kruskal.
- `scikit-image`: algoritmo de cubos marchantes.
- `matplotlib` y `seaborn`: visualización de figuras 3D.
- `numba`: compilación de código.

El paquete `numba` fue especialmente útil para acelerar operaciones clave de los algoritmos, como la eliminación gaussiana y el algoritmo `ConstruCiclos`. Python es un lenguaje interpretado, por lo que compilar secciones específicas de código permite alcanzar mejoras de rendimiento. Otra librería útil fue `numpy-stl`, la cual permite leer y escribir archivos `.stl`. Este formato de archivo es ampliamente usado para representar triangulaciones en tres dimensiones, sobre todo en el ámbito de impresión 3D. Finalmente, la librería `time` permitió medir el tiempo de ejecución de los algoritmos durante experimentos.

Resumiendo, contamos con tres implementaciones de algoritmos de cálculo de homología. Estos algoritmos son `HomArb`, `HomOpt1` y `HomOpt2`. El primero de estos calcula una base de homología arbitraria de un complejo simplicial, mientras que los últimos dos buscan una base de homología óptima. De acuerdo con el Cuadro 6.1, los algoritmos `HomArb` y `HomOpt2` tienen complejidad de tiempo de ejecución $\mathcal{O}(n^3)$, mientras que `HomOpt1` tiene complejidad $\mathcal{O}(n^4)$. Aquí, n es una cantidad que mayormente la suma de los tamaños de las listas de entrada. A continuación, corroboramos experimentalmente la complejidad de los tiempos de ejecución.

7.1. Complejidad de tiempo de ejecución experimental

Como diseño experimental, generamos triangulaciones del toro con distintas cantidades de triángulos a partir de una parametrización de la superficie, al estilo de la Figura 4.2. La cantidad de triángulos aumenta en intervalos aproximados de 250 para el algoritmo HomArb, 1000 para el algoritmo HomOpt2 y 5000 para el algoritmo HomOpt1. Procurando que ningún algoritmo demore más de 10 minutos en ejecutarse, estos intervalos generan cerca de 10 valores de medición por algoritmo.

Medimos el tiempo de ejecución de los algoritmos desde una computadora con un procesador AMD Ryzen 7 5700U, RAM de 16 GB y sistema operativo Windows 11. Como adelantamos, realizamos estas mediciones con la librería `time` de Python. En ordenadores Windows, esta librería ofrece una precisión de 16 milisegundos. Corremos 5 veces cada uno de los algoritmos en cada uno de sus complejos simpliciales asociados y registramos los tiempos.

n_2	Δt (s)	n_2	Δt (s)	n_2	Δt (s)
5 000	0.8	242	0.3	1 058	3.7
10 082	5.6	512	2.3	2 048	14.6
15 138	11.3	722	6.5	3 042	32.7
20 000	26.2	1 058	20.8	4 050	61.3
25 088	58.4	1 250	36.6	5 000	102.3
30 258	104.8	1 568	83.3	6 050	158.4
35 378	170.9	1 800	141.0	7 200	240.5
40 328	269.2	2 048	290.3	8 192	318.1
45 000	396.3	2 312	469.2	9 248	443.5
50 562	579.5	–	–	10 082	496.7

(a) HomArb
(b) HomOpt1
(c) HomOpt2

Cuadro 7.1: Tiempos de ejecución medios de algoritmos de cálculo de homología.

En el Cuadro 7.1 presentamos el tiempo medio de ejecución en segundos para cada algoritmo, junto con la cantidad de triángulos en cada experimento. A su vez, la Figura 7.1 resume estos datos en diagramas de dispersión. Si una triangulación del Cuadro 7.1 cuenta con n_2 triángulos, se deduce que el complejo contiene $3n_2/2$ aristas y $n_2/2$ vértices. Esto se debe a que, en las triangulaciones trabajadas, cada arista es parte de 2 triángulos y cada vértice es parte de 6 triángulos. Omitimos dar medidas de error, ya que el coeficiente de variación máximo para un algoritmo y triangulación dados fue 3.7%.

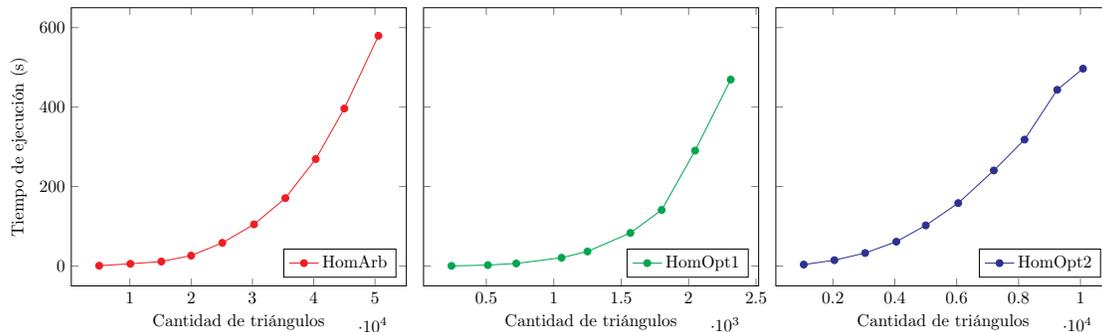


Figura 7.1: Tiempos de ejecución de algoritmos de cálculo de homología

En el caso del algoritmo HomOpt1, aunque se deseara hacer pruebas más largas, nos encontramos con dificultades con el uso de almacenamiento. El complejo más grande en el que se probó este algoritmo posee 2592 triángulos, 3888 aristas y 1296 vértices. Dado que la última matriz involucrada

en este algoritmo tiene dimensiones de $n_1 \times (n_2 + n_0(n_1 - n_0 + 1))$ y usamos un byte de memoria para almacenar cada entrada de esta, la matriz ocupa más de 13 GB de RAM.

Como era de esperarse, HomArb es el algoritmo más rápido entre los tres principales, seguido de HomOpt2 y finalmente de HomOpt1. Para precisar el orden de complejidad de estas nubes de puntos, aplicamos una regresión de potencia para cada algoritmo, dada entre el número de triángulos y el tiempo de ejecución medio. En concreto, si $\{(x_i, y_i)\}_{i=1}^k$ es uno de los conjuntos de datos del Cuadro 7.1, buscamos números reales α, β que minimicen $\sum_{i=1}^k (y_i - \beta x_i^\alpha)^2$. Una aproximación usual para α, β es resolver en su lugar el problema de mínimos cuadrados lineal para $\{(\log x_i, \log y_i)\}_{i=1}^k$. Esto se justifica en que, si $y_i = \beta x_i^\alpha$, aplicar logaritmos produce $\log y_i = \alpha \log x_i + \log \beta$, una relación lineal entre $\log x_i$ y $\log y_i$. Notamos que logaritmos de bases distintas producen los mismos coeficientes α, β .

Aplicamos el método anterior en nuestros conjuntos de datos, una vez para cada uno de los tres algoritmos. En la Figura 7.2 transformamos los ejes de las gráficas de la Figura 7.1 en escala logarítmica base 10 y aplicamos las regresiones lineales discutidas. Los coeficientes α y β obtenidos, junto con los coeficientes de determinación r^2 , se muestran en el Cuadro 7.2.

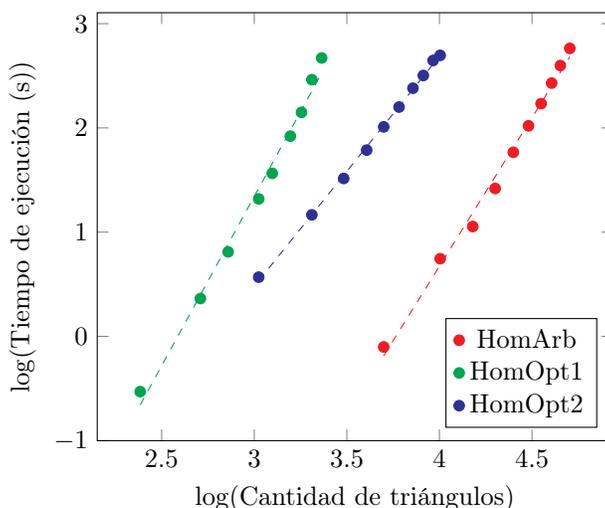


Figura 7.2: Tiempos de ejecución de algoritmos de cálculo de homología (log-log).

Los coeficientes α del Cuadro 7.2 representan un exponente experimental de la complejidad de tiempo de ejecución de los algoritmos. Notamos que cada algoritmo evidencia un exponente menor al teórico, pues se esperaba que HomArb, HomOpt1 y HomOpt2 tuvieran órdenes de complejidad 3, 4 y 3, respectivamente. En su lugar, se obtuvieron los exponentes 2.85, 3.25 y 2.21, donde notamos que los algoritmos de homología óptima son casi un orden menor a la cota teórica propuesta. Una primera justificación de esta diferencia es que la complejidad de la eliminación gaussiana se ve reducida al trabajar con matrices esparsas, es decir, matrices con gran porcentaje de entradas nulas. Las matrices con las que trabajamos tienden a ser esparsas, pues provienen de grafos donde $n_1 \ll n_0^2$. Este factor afecta a los tres algoritmos, pues todos dependen de eliminación gaussiana de uno u otro modo.

Algoritmo	α	β	r^2
HomArb	2.853	-10.739	0.993
HomOpt1	3.248	-8.401	0.991
HomOpt2	2.206	-6.138	0.999

Cuadro 7.2: Coeficientes de regresiones lineales.

Respecto a los algoritmos HomOpt1 y HomOpt2, otra explicación se encuentra en la holgada cota de complejidad asignada a ConstruCiclos. Se estableció que este algoritmo es $\mathcal{O}(n^3)$, donde un factor n de n^3 proviene de asumir que la cantidad máxima de aristas en uno de estos ciclos es n_1 . Sin embargo, para las triangulaciones del toro empleadas en los experimentos, esta cota nunca se alcanza. Por el contrario, es frecuente construir ciclos con pocas aristas, incluso con solo 3 vértices.

La Figura 7.3 muestra la máxima cantidad de aristas de un ciclo construido por ConstruCiclos al aplicar el algoritmo en triangulaciones del toro. Una regresión de potencia contra la cantidad de triángulos en el complejo, la cual adjuntamos en la Figura 7.3, indica que esta cota es del orden $\mathcal{O}(n^{0.516})$. Entonces, para los algoritmos HomOpt1 y HomOpt2, esta cota explica una porción de la diferencia entre las complejidades de los cuadros 6.1 y 7.2.

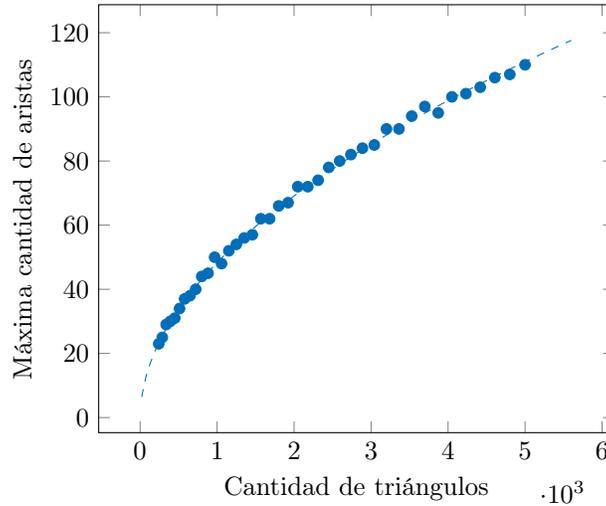


Figura 7.3: Máxima cantidad de aristas en un ciclo construido por ConstruCiclos.

7.2. Cálculos de bases de homología arbitrarias

Ahora, presentamos las salidas de los algoritmos en algunos complejos particulares. Iniciamos con el algoritmo HomArb. En la Figura 7.4 se muestran generadores del primer grupo de homología con coeficientes en \mathbb{Z}_2 de un toro y un doble toro encontrados por este algoritmo. De parte del toro, tenemos 2 generadores, mientras que el doble toro posee 4. Esto se alinea con los resultados de la teoría algebraica. Como demostramos con el teorema de Van Kampen, los grupos fundamentales de estos espacios son $\pi_1(\mathbb{T}) = \mathbb{Z}_2 \times \mathbb{Z}_2$ y $\pi_1(\mathbb{T}^2) = \langle a, b, c, d \mid aba^{-1}b^{-1}cdc^{-1}d^{-1} \rangle$. Abelianizar estos grupos e introducir coeficientes en \mathbb{Z}_2 en efecto devuelve los grupos de homología $H_1(\mathbb{T}^2, \mathbb{Z}_2) = \mathbb{Z}_2^2$ y $H_1(\mathbb{T}^2 \# \mathbb{T}^2, \mathbb{Z}_2) = \mathbb{Z}_2^4$.

En la Figura 7.5 presentamos un par de cálculos más de bases de homología encontradas por HomArb. La Figura 7.5a muestra un modelo famoso en el campo de diseño 3D en computadora, denominado la taza de Utah. Topológicamente, la taza de Utah es homeomorfa a un doble toro. Este modelo data de 1975 y fue desarrollado por Martin Newell en la universidad mencionada. La triangulación empleada, así como muchas otras a continuación, se obtuvieron del sitio web Thingiverse, un foro gratuito para compartir modelos diseñados para impresión 3D.

Por otra parte, la Figura 7.5b muestra una superficie de género 3. Abordamos esta superficie en la siguiente sección, pues esta visualización evidencia claramente un problema recurrente en las bases de homología encontradas al momento. Los ciclos de las bases de homología vistas resultan con múltiples aristas compartidas entre sí. En primera instancia, ya es desafiante identificar los

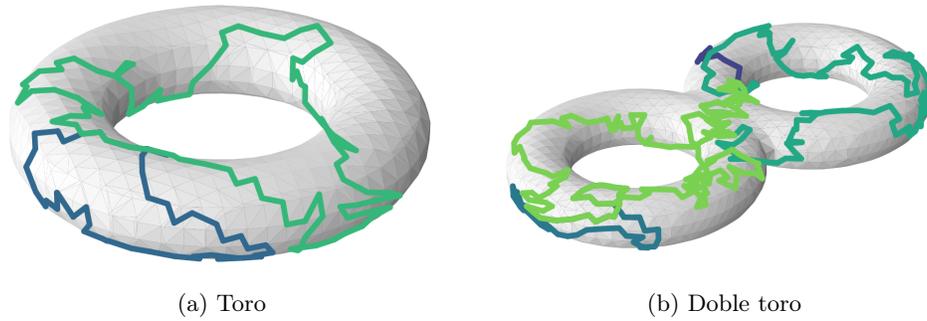


Figura 7.4: Bases de homología arbitrarias de un toro y un doble toro.

ciclos dada su falta de optimalidad, pero este problema se agrava cuando no pueden observarse la totalidad de los ciclos simultáneamente porque ocupan un mismo puesto. En consecuencia, los ciclos en la Figura 7.5b se ven opacados por el último ciclo trazado, en este caso el visto en verde claro. Nos abstenemos de presentar más ejemplos con este algoritmo, pues este problema es más serio en complejos más grandes con más agujeros.

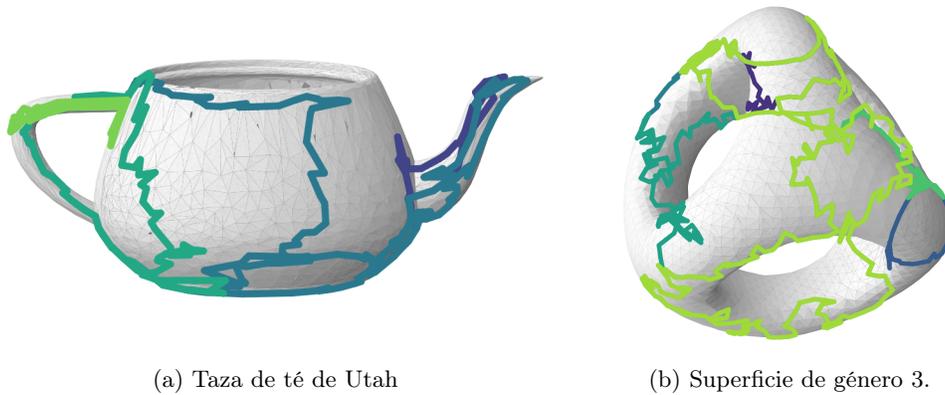


Figura 7.5: Más bases de homología arbitrarias.

Para explicar este fenómeno, analizamos la esparcidad de las matrices que participan en el algoritmo HomArb. Definimos la esparcidad de una matriz como el cociente entre la cantidad de entradas no nulas y la cantidad de entradas totales de la matriz. Un breve cálculo muestra que ∂_2 y ∂_1 , de dimensiones $n_1 \times n_2$ y $n_0 \times n_1$, tienen esparcidades $3/n_1$ y $2/n_0$. En la triangulación del toro con 5000 triángulos, estas esparcidades son de 4.0×10^{-4} y 8.0×10^{-4} . Sin embargo, para esta misma triangulación, las primeras 1000 filas de \mathbf{Z} poseen esparcidad 1.0×10^{-2} . Esta esparcidad se encuentra dos ordenes de magnitud por encima de las primeras dos, y se debe a que la eliminación gaussiana no necesariamente preserva la esparcidad de las matrices. La alta esparcidad de \mathbf{Z} propicia que sus columnas seleccionadas para formar parte de la base de homología compartan entradas no nulas en las mismas posiciones, generando aristas compartidas.

Aunque este algoritmo no se luce en la calidad visual de las bases de homología encontradas, HomArb ofrece el cálculo de la dimensión del grupo de homología en tiempo récord, el cual por su cuenta transmite conocimiento sobre el complejo a analizar. Además, HomArb verifica los resultados de los siguientes algoritmos. En los complejos de la siguiente sección, HomArb se ejecutó para cada triangulación antes de ejecutar los otros algoritmos, pues HomArb ofrece una vista previa para determinar si los resultados experimentales son teóricamente correctos, sin necesidad de invertir mucho tiempo.

7.3. Cálculos de bases de homología óptimas

Evaluamos el desempeño de los algoritmos HomOpt1 y HomOpt2 calculando una variedad de ejemplos. En realidad, las bases de homología encontradas son ambas óptimas y en ocasiones muy similares. Por lo tanto, dada la superior eficiencia del algoritmo HomOpt2, le delegamos a este el cálculo de homología de los ejemplos a seguir. En primer lugar, calculamos la homología de algunas superficies conocidas provenientes de ecuaciones implícitas, para luego comparar los resultados con la teoría, similar al tratamiento dado al algoritmo HomArb. Después, calculamos la homología de objetos más exóticos, provenientes de modelos para impresión 3D.

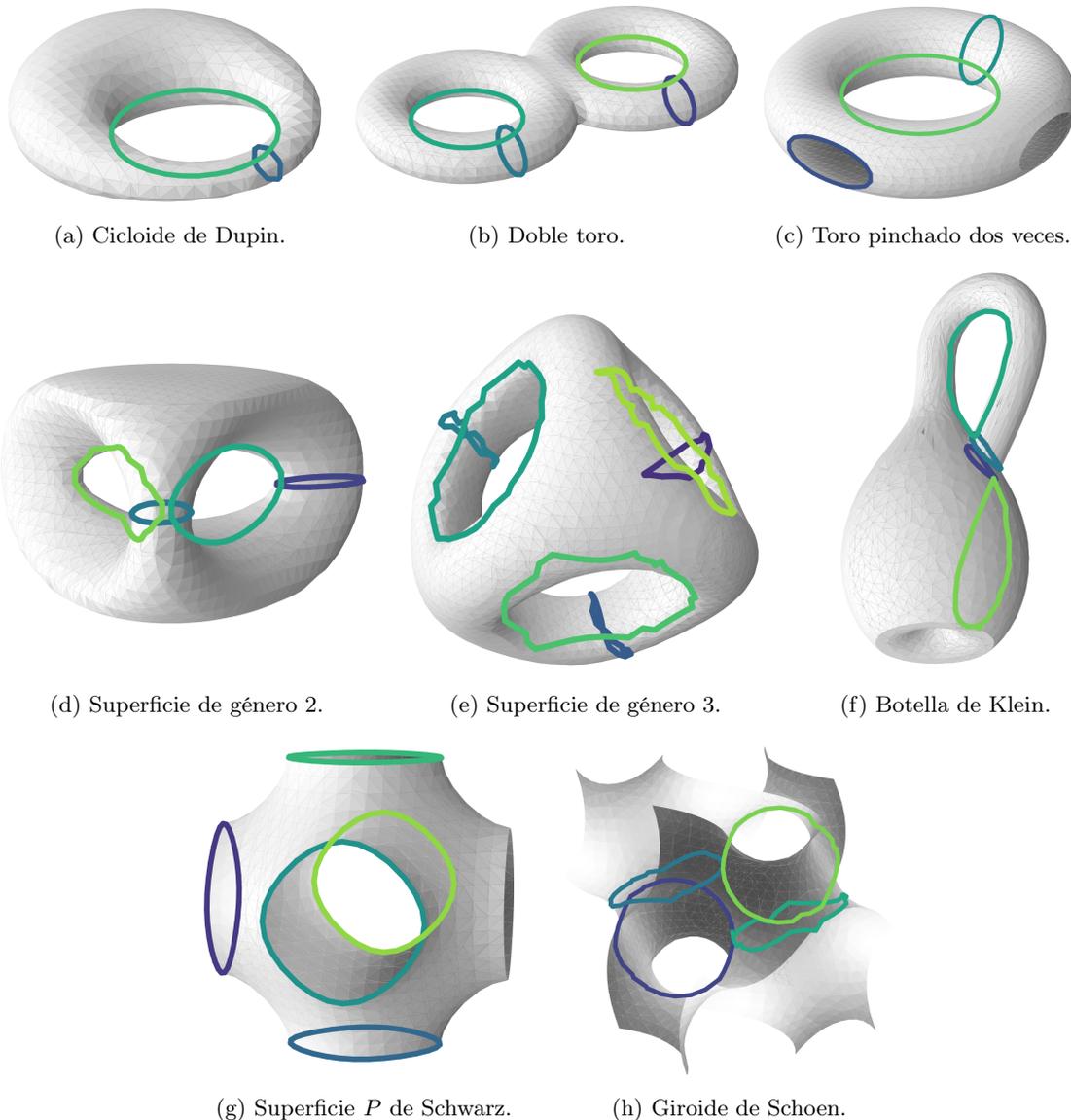


Figura 7.6: Bases de homología óptimas.

En la Figura 7.6 presentamos 8 ejemplos de complejos con sus bases de homología óptimas. En cada figura, el color de los ciclos se determina por el orden de sus longitudes, donde los ciclos cortos utilizan matices más violetas y azules que los ciclos largos, los cuales se representan con matices amarillos y verdes. Vemos que, en contraste con el algoritmo HomArb visto en la sección anterior,

los ciclos de las bases encontradas por HomOpt2 raramente comparten aristas, con lo cual son fácilmente distinguibles entre sí. Además, la optimalidad de los ciclos comunica mejor al estructura homológica de los complejos en comparación a los ciclos arbitrarios anteriores.

En primer lugar, en la Figura 7.6a contamos con un cicloide de Duplin, descrito a través de la ecuación

$$(x^2 + y^2 + z^2 + b^2 - d^2)^2 - 4(ax - cd)^2 - 4b^2y^2 = 0. \quad (7.1)$$

para algunas constantes a, b, c, d . En este ejemplo, se emplearon las constantes $a = 1$, $b = 0.97$, $c = 0.2$ y $d = 0.3$. Como puede apreciarse, el cicloide de Duplin es homeomorfo al toro, con lo cual $H_1(K, \mathbb{Z}_2)$ también posee dos generadores. En la Figura 7.6b contamos con un doble toro y de nuevo vemos 4 ciclos generadores, como en la Figura 7.4a.

La Figura 7.6c muestra un toro con lo que parece ser dos agujeros adicionales. Este complejo es homotópicamente equivalente a un toro pinchado dos veces. Puede mostrarse que el toro pinchado una vez es homotópicamente equivalente a $S^1 \vee S^1$, mientras que el toro pinchado dos veces es homotópicamente equivalente a $S^1 \vee S^1 \vee S^1$. Entonces, por el teorema de Van Kampen se sigue que el toro pinchado dos veces posee grupo fundamental $\pi_1(S^1 \vee S^1 \vee S^1) = \mathbb{Z} * \mathbb{Z} * \mathbb{Z}$. La abelianización de este grupo fundamental produce un primer grupo de homología con 3 generadores.

La figuras 7.6d y 7.6e muestran superficies de género 2 y 3, homeomorfas a la suma conexas de dos y tres toros, respectivamente. Estas superficies fueron obtenidas de las siguientes ecuaciones implícitas. Como se esperaba, sus bases de homología cuentan con 4 y 6 ciclos generadores.

$$\text{Figura 7.6d: } 2y(y^2 - 3x^2)(1 - z^2) + (x^2 + y^2)^2 - (9z^2 - 1)(1 - z^2) = 0. \quad (7.2)$$

$$\begin{aligned} \text{Figura 7.6e: } & (x - 2)^2(x + 2)^2 + (y - 2)^2(y + 2)^2 + (z - 2)^2(z + 2)^2 \\ & + 3(x^2y^2 + x^2z^2 + y^2z^2) + 6xyz - 10(x^2 + y^2 + z^2) + 22 = 0. \end{aligned} \quad (7.3)$$

La Figura 7.6f es distinta, pues muestra una inmersión de la botella de Klein en \mathbb{R}^3 como una superficie con auto intersecciones. La homología de esta inmersión no necesariamente refleja la homología de la botella de Klein, pues simplemente estamos calculando la base de homología de un complejo utilizado para representar este objeto en tres dimensiones. Dicho esto, coincidentemente puede mostrarse que el primer grupo de homología con coeficientes en \mathbb{Z}_2 de la botella de Klein sí cuenta con cuatro generadores. El modelo empleado en esta figura, al igual que la taza de Utah vista en la Figura 7.5a, fue diseñado para ser impreso en 3D y se extrajo del mismo sitio.

Las últimas dos figuras, 7.6g y 7.6h, muestran superficies mínimas estudiadas en geometría diferencial. En realidad, se presenta solo una celda de las superficies totales, pues ambas son superficies triple periódicas en \mathbb{R}^3 . La primera lleva el nombre de superficie P de Schwarz, mientras que la segunda se conoce como el giroide de Schoen. Ambas de estas superficies se obtuvieron bajo aproximaciones con ecuaciones trigonométricas, las cuales están dadas por

$$\text{Figura 7.6g: } \cos x + \cos y + \cos z = 0. \quad (7.4)$$

$$\text{Figura 7.6h: } \sin x \cos y + \sin y \cos z + \sin z \cos x = 0. \quad (7.5)$$

La superficie P de Schwarz es homotópicamente equivalente a una esfera pinchada 6 veces. Dado que la esfera pinchada es homeomorfa al plano, la esfera pinchada 6 veces es homeomorfa al plano pinchado 5 veces. A su vez, este plano 5 veces pinchado es homotópicamente equivalente a la suma cuña de 5 círculos S^1 , la cual posee como grupo fundamental el grupo libre en 5 generadores. Por lo tanto, el primer grupo de homología de la superficie P de Schwarz también tiene 5 generadores.

Por otra parte, para el giroide de Schoen no es claro cómo extraerle de forma teórica su primer grupo de homología. Esta celda de la superficie guarda la apariencia de un sistema de cuevas y es visualmente más compleja que el resto de objetos vistos hasta el momento. En casos como este es especialmente útil el cálculo computacional de grupos de homología, pues un algoritmo no se ve intimidado por la apariencia de una superficie.

Finalizamos el capítulo con otro cúmulo de complejos interesantes junto con sus grupos de homología, pero esta vez exclusivamente de modelos 3D como la taza de Utah. Todos los diseños fueron extraídos de los sitios web Thingiverse y Printables, e inicialmente fueron creados para su impresión 3D. En ocasiones se redujo la resolución original de estos modelos en el software 3D Builder de Microsoft antes de alimentarlos al algoritmo HomOpt1, pues algunos de estos inicialmente contaban con más de 100 000 triángulos.

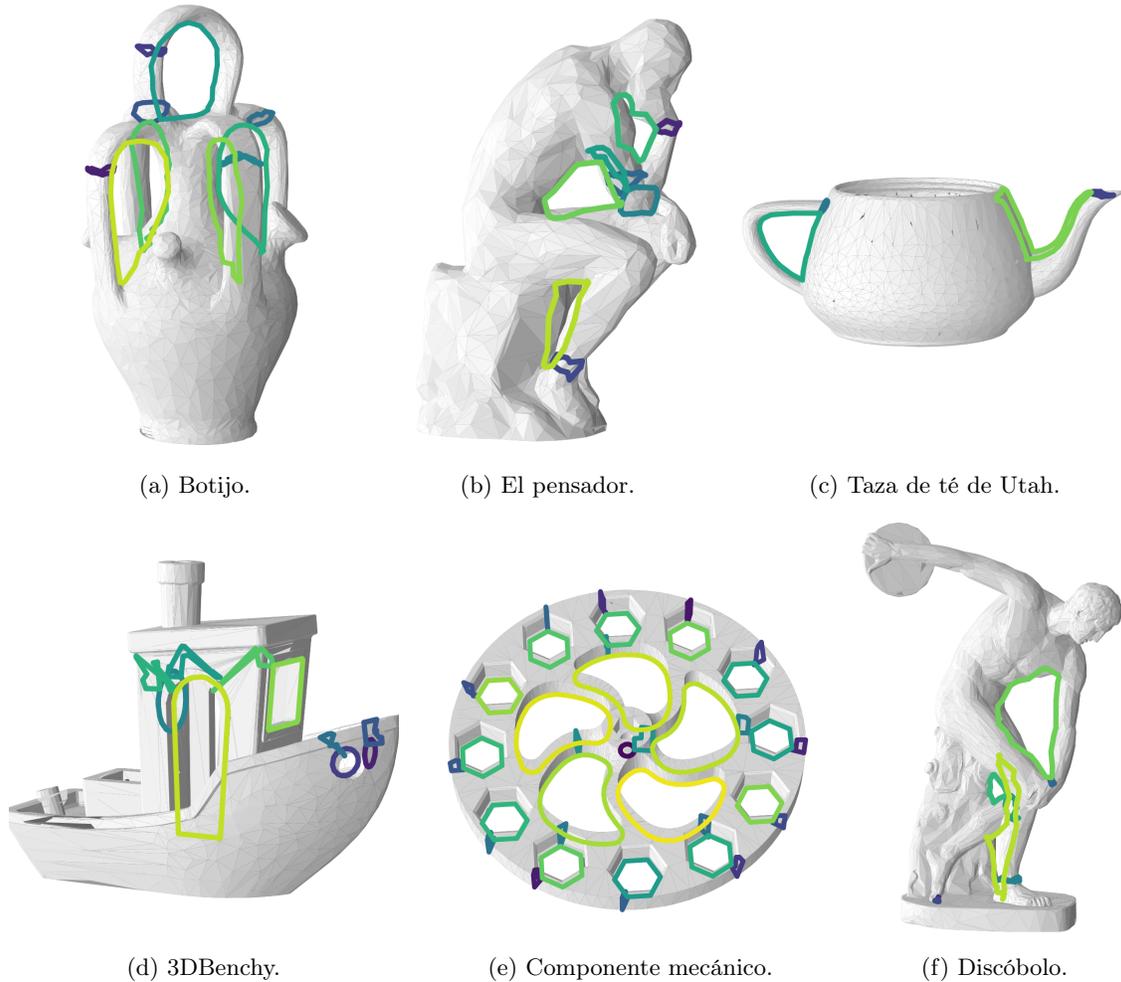


Figura 7.7: Bases de homología óptimas de objetos varios.

La Figura 7.7a muestra un botijo, otro modelo icónico en el contexto de diseño de gráficos por computadora. Por otro lado, las figuras 7.7b y 7.7f muestran dos esculturas célebres: El Pensador y El Discóbolo. La Figura 7.7d muestra un barco, conocido en la ingeniería como “3DBenchy”, y es el modelo estándar para calibrar y examinar el buen funcionamiento de una impresora 3D. Finalmente, la Figura 7.7e muestra un componente mecánico parecido a una rueda, en la cual se identificaron 36 ciclos en la base de homología.

Concluimos con un resumen de la cantidad de ciclos generadores encontrados en las bases de homología de los complejos anteriores. Presentamos estos datos en el Cuadro 7.3.

K	No. de figura	$\dim H_1(K, \mathbb{Z}_2)$
Toro	7.4a	2
Cicloide de Dupin	7.6a	2
Doble Toro	7.6b	4
Toro pinchado dos veces	7.6c	3
Superficie de género 2	7.6d	4
Superficie de género 3	7.6e	6
Botella de Klein	7.6f	4
Superficie P de Schwarz	7.6g	5
Giroide de Schoen	7.6h	4
Botijo	7.7a	10
El Pensador	7.7b	8
Taza de té de Utah	7.7c	4
3DBenchy	7.7d	10
Componente mecánico	7.7e	36
Discóbolo	7.7f	8

Cuadro 7.3: Cantidad de generadores en los grupos de homología.

- En este texto se discutieron las bases teóricas de la topología algebraica, incluyendo el grupo fundamental y los grupos de homología simplicial. En más detalle, se describió que los grupos de homología simplicial sobre un campo poseen una estructura de espacio vectorial, lo cual se aprovechó al implementar algoritmos para el cálculo computacional de grupos de homología. Con este lenguaje, se abarcó un problema de optimización respecto a la longitud de los ciclos en estos grupos de homología.
- Se desarrollaron e implementaron algoritmos en un lenguaje de programación para la identificación de bases de grupos de homología en complejos simpliciales con coeficientes en \mathbb{Z}_2 , vistos como espacios vectoriales. Los algoritmos principales son HomArb, que busca una base arbitraria de ciclos de cualquier dimensión, y HomOpt1 y HomOpt2, que buscan bases óptimas de ciclos unidimensionales.
- Los algoritmos HomArb, HomOpt1 y HomOpt2 generaron resultados teóricamente correctos para el cálculo de grupos de homología con coeficientes en \mathbb{Z}_2 . Esto se verificó para una docena de complejos simpliciales con apoyo de herramienta de la topología algebraica como el teorema de Van Kampen. A su vez, se generaron visualizaciones de estos complejos para identificar gráficamente los ciclos de los grupos de homología calculados computacionalmente.
- Los algoritmos HomArb, HomOpt1 y HomOpt2 poseen complejidades de tiempo de ejecución polinomiales. En específico, estas complejidades son $\mathcal{O}(n^3)$, $\mathcal{O}(n^4)$ y $\mathcal{O}(n^3)$ respectivamente, donde n es una cantidad que mayor a la suma de las longitudes de las listas de entrada de los algoritmos. En la práctica se confirmaron estas cifras, pues se obtuvieron complejidades experimentales de $\mathcal{O}(n^{2.85})$, $\mathcal{O}(n^{3.25})$ y $\mathcal{O}(n^{2.21})$ al aplicar los algoritmos en distintas triangulaciones del toro.
- El algoritmo HomArb ofrece el cálculo de la dimensión de un grupo de homología en tiempo récord en relación a los algoritmos HomOpt1 y HomOpt2, el cual por su cuenta transmite conocimiento sobre el complejo a analizar. Sin embargo, HomArb no se luce en la calidad visual de las bases de homología que produce, culpa de la falta de optimalidad de sus ciclos y las colisiones entre sus aristas.
- Los algoritmos HomOpt1 y HomOpt2 producen bases de homología cuyos ciclos raramente comparten aristas, con lo cual son fácilmente distinguibles entre sí. En las visualizaciones, la optimalidad de los ciclos que retornan estos algoritmos comunica mejor la estructura homológica de los complejos en comparación a los ciclos arbitrarios generados por el algoritmo HomArb.

Recomendaciones

- Al lector interesado en perseguir una línea de estudio similar, se recomienda investigar sobre un problema de optimización distinto al de encontrar una base de homología óptima, aunque fuertemente relacionado a este. El problema consiste en, dado un q -ciclo z en un complejo simplicial, encontrar un ciclo de longitud mínima en la clase de homología de z . En contraste con el problema discutido en esta tesis, el cual es NP-difícil para $q > 1$ pero posee una solución polinomial para $q = 1$, este problema es NP-difícil incluso para $q = 1$ [4]. Para un planteamiento de una solución utilizando programación lineal, puede verse [7].
- Aunque los algoritmos descritos en este texto son polinomiales, el menor grado de complejidad de estos es cúbico. Dado que no es inusual que el tamaño de los complejos simpliciales de interés posean un número de vértices en el orden de los millones, sobre todo en aplicaciones como la generación de gráficos por computadora, este grado de complejidad puede resultar muy alto. Por lo tanto, puede haber interés en buscar aproximaciones a bases de homología óptimas, con un tiempo de ejecución menor. Existen algoritmos, siempre concentrados en dimensión 1, que atacan el problema desde esta perspectiva, que pueden interesarle al lector [6]. Las estrategias en estos algoritmos emplean otras estructuras de la topología algebraica como los complejos de Vietoris–Rips, e incluso incorporan tópicos de la geometría diferencial.

-
- [1] Adhikari, M.R.: *Brief History of Algebraic Topology. Basic Topology 3*. Springer Singapore, 2022.
- [2] Armstrong, M.A.: *Basic Topology*. Springer New York, 1ª edición, 1983.
- [3] Busaryev, Oleksiy, Sergio Cabello, Chao Chen, Tamal Dey y Yusu Wang: *Annotating Simplices with a Homology Basis and Its Applications*. En *Algorithm Theory – SWAT 2012*, páginas 189–200. Springer Berlin, 2012.
- [4] Chen, Chao y Daniel Freedman: *Hardness Results for Homology Localization*. *Discrete & Computational Geometry*, 45:425–448, 2011.
- [5] Cormen, Thomas, Charles Leiserson, Ronald Rivest y Stein Clifford: *Introduction to Algorithms*. The MIT Press, 3ª edición, 2009.
- [6] Dey, Tamal, Jian Sun y Yusu Wang: *Approximating loops in a shortest homology basis from point data*. En *SoCG '10: Proceedings of the twenty-sixth annual symposium on computational geometry*, página 166–175. ACM, 2010.
- [7] Dey, Tamal y Yusu Wang: *Computational Topology for Data Analysis*. Cambridge University Press, 2022.
- [8] Diestel, Reinhard: *Graph Theory*. Springer Berlin, 5ª edición, 2017.
- [9] Dijkstra, Edsger: *A Note on Two Problems in Connexion with Graphs*. *Numerische mathematik*, 1:269–271, 1959.
- [10] Erickson, Jeff y Kim Whittlesey: *Greedy Optimal Homotopy and Homology Generators*. En *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on discrete algorithms*, página 1038–1046. ACM-SIAM, 2005.
- [11] Gallian, Joseph: *Contemporary Abstract Algebra*. Cengage Learning, 8ª edición, 2012.
- [12] Giunti, Barbara, Jānis Lazovskis y Bastian Rieck: *DONUT: Creation, Development, and Opportunities of a Database*. *Notices of the American Mathematical Society*, 70(10):1640–1644, 2023.
- [13] Hatcher, Allen: *Algebraic Topology*. Cambridge University Press, 1ª edición, 2001.

- [14] Horton, Joseph: *A Polynomial-Time Algorithm to Find the Shortest Cycle Basis of a Graph*. SIAM Journal on Computing, 16(2):358–366, 1987.
- [15] Kruskal, Joseph B.: *On the shortest spanning subtree of a graph and the traveling salesman problem*. Proceedings of the American Mathematical society, 7(1):48–50, 1956.
- [16] Lewiner, Thomas, Hélio Lopes, Antônio Wilson y Geovan Tavares: *Efficient Implementation of Marching Cubes' Cases with Topological Guarantees*. Journal of Graphics Tools, 8(2):1–15, 2003.
- [17] Loera, Jesús A., Jörg Rambau y Francisco Santos: *Triangulations: Structures for Algorithms and Applications*. Springer Singapore, 2010.
- [18] Lorensen, William y Harvey Cline: *Marching Cubes: A High Resolution 3D Surface Construction Algorithm*. ACM SIGGRAPH Computer Graphics, 21(4):163–169, 1987.
- [19] Munkres, James: *Elements of Algebraic Topology*. Addison Wesley Publishing Company, 1984.
- [20] Munkres, James: *Topology, Pearson New International Edition*. Pearson, 2ª edición, 2013.
- [21] Radó, Tibor: *Über den Begriff der Riemannschen Fläche*. Acta Litt. Sci. Szeged, 2:101–121, 1925.

11.1. Ejemplos de ejecución de algoritmos

En este anexo, ejemplificamos los tres algoritmos del capítulo 6 para la identificación de grupos de homología en complejos simpliciales. En concreto, revisamos los algoritmos HomArb, HomOpt1 y HomOpt2. Aplicamos estos algoritmos al complejo simplicial K visto en la Figura 11.1. Dado que K es homotópicamente equivalente a $S^1 \vee S^1$, por el teorema de Van Kampen conocemos que su grupo fundamental es $\pi_1(K) = \mathbb{Z} * \mathbb{Z}$. Entonces, esperamos que las salidas de los tres algoritmos consistan de dos ciclos homológicamente independientes.

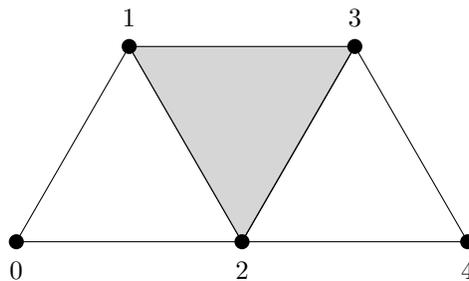


Figura 11.1: Complejo simplicial K .

El complejo K cuenta con 1 triángulo, 7 aristas y 5 vértices. Etiquetando los vértices de K como en la Figura 11.1, se sigue que las matrices de entrada de los algoritmos que describen los símlices de K son

$$S_2 = (1 \quad 2 \quad 3), \quad S_1 = \begin{pmatrix} 0 & 1 \\ 0 & 2 \\ 1 & 2 \\ 1 & 3 \\ 2 & 3 \\ 2 & 4 \\ 3 & 4 \end{pmatrix}, \quad S_0 = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}.$$

Estos símlices definen las tres bases ordenadas canónicas de los espacios de cadenas $C_2(K, \mathbb{Z}_2)$, $C_1(K, \mathbb{Z}_2)$ y $C_0(K, \mathbb{Z}_2)$, respecto a las cuales calculamos vectores coordenados.

11.1.1. Algoritmo HomArb

Iniciamos con la construcción de las matrices ∂_2 y ∂_1 . Vemos que ∂_2 está formado por el vector coordenado de $\partial_2(123) = 12 + 13 + 23$, el borde del único triángulo de K . Por otra parte, la j -ésima columna de ∂_1 es el vector coordenado del borde de la j -ésima arista de K .

$$\partial_2 = \begin{pmatrix} \cdot \\ \cdot \\ 1 \\ 1 \\ 1 \\ \cdot \\ \cdot \end{pmatrix}, \quad \partial_1 = \begin{pmatrix} 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot \\ \cdot & 1 & 1 & \cdot & 1 & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 \end{pmatrix}.$$

Luego, calculamos una base de $\ker \partial_1$. Para ello, aplicamos eliminación gaussiana a ∂_1 para obtener la matriz ∂'_1 . Como puede apreciarse, ∂'_1 cuenta con 3 columnas sin pivote, con lo cual $\ker \partial_1$ cuenta con 3 elementos. Los vectores coordenados de estos 3 ciclos se empaquetan en las columnas de \mathbf{Z} .

$$\partial'_1 = \begin{pmatrix} \mathbf{1} & \cdot & 1 & \cdot & 1 & \cdot & 1 \\ \cdot & \mathbf{1} & 1 & \cdot & 1 & \cdot & 1 \\ \cdot & \cdot & \cdot & \mathbf{1} & 1 & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{1} & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix}, \quad \mathbf{Z} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & \cdot & \cdot \\ \cdot & 1 & 1 \\ \cdot & 1 & \cdot \\ \cdot & \cdot & 1 \\ \cdot & \cdot & 1 \end{pmatrix}.$$

Por último, aplicamos eliminación gaussiana a $[\partial_2 \mid \mathbf{Z}]$ para obtener $[\partial'_2 \mid \mathbf{Z}']$.

$$[\partial_2 \mid \mathbf{Z}] = \left(\begin{array}{c|ccc} \cdot & 1 & 1 & 1 \\ \cdot & 1 & 1 & 1 \\ 1 & 1 & \cdot & \cdot \\ 1 & \cdot & 1 & 1 \\ 1 & \cdot & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 \\ \cdot & \cdot & \cdot & 1 \end{array} \right) \xrightarrow{\text{FRER}} [\partial'_2 \mid \mathbf{Z}'] = \left(\begin{array}{c|ccc} \mathbf{1} & \cdot & 1 & \cdot \\ \cdot & \mathbf{1} & 1 & \cdot \\ \cdot & \cdot & \cdot & \mathbf{1} \\ \cdot & \cdot & \cdot & \cdot \end{array} \right).$$

La salida del algoritmo son los ciclos de las columnas 1 y 3 en \mathbf{Z} , pues estas columnas tienen pivotes en \mathbf{Z}' . En concreto, la base de homología encontrada está dada por los ciclos $01 + 02 + 12$ y $01 + 02 + 13 + 24 + 34$.

11.1.2. Algoritmo HomOpt1

Iniciamos siguiendo el procedimiento indicado por el algoritmo ConstruCiclos. Primero, listamos los 5 árboles de distancia más corta T_v para cada uno de los 5 vértices v que tiene K . Cuando existen múltiples caminos de distancia mínima entre dos vértices, se selecciona el camino más pequeño según el orden lexicográfico de sus vértices. Estos árboles se muestran en la Figura 11.2. Cada uno de estos árboles T_v aporta tres ciclos de la forma $\varphi(T_v, e) = G_K[v, u_i] + G_K[v, u_j] + u_i u_j$ al conjunto de ciclos Π , con lo cual Π tiene 15 elementos. Estos ciclos son los siguientes:

$$\begin{array}{lll} T_0 : & 01 + 02 + 12, & 01 + 02 + 13 + 23, & 01 + 02 + 13 + 24 + 34. \\ T_1 : & 01 + 02 + 12, & 12 + 13 + 23, & 12 + 13 + 24 + 34. \\ T_2 : & 01 + 02 + 12, & 12 + 13 + 23, & 23 + 24 + 34. \\ T_3 : & 01 + 02 + 13 + 23, & 12 + 13 + 23, & 23 + 24 + 34. \\ T_4 : & 01 + 02 + 12, & 12 + 13 + 24 + 34, & 23 + 24 + 34. \end{array} \quad (11.1)$$

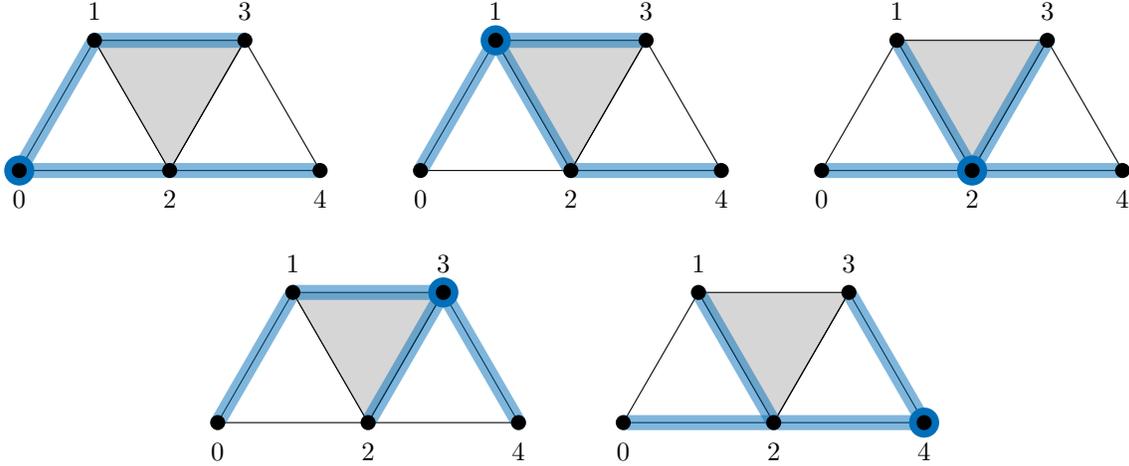


Figura 11.2: Árboles de distancia más corta del complejo K .

Luego, construimos la matriz $\mathbf{\Pi}$ cuyas columnas almacenan los vectores coordenados de estos 15 ciclos. Ordenamos las columnas de izquierda a derecha según su longitud de sus ciclos, colocando en la izquierda los ciclos más pequeños.

$$[\partial_2 \mid \mathbf{\Pi}] = \left(\begin{array}{c|cccccccccccccccc} \cdot & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 \\ \cdot & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot \\ 1 & \cdot & \cdot & \cdot & \cdot & 1 & 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 & 1 & \cdot \\ 1 & \cdot & \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot \\ \cdot & 1 & 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 \\ \cdot & 1 & 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 \end{array} \right). \quad (11.2)$$

Por último, aplicamos eliminación gaussiana a $[\partial_2 \mid \mathbf{\Pi}]$ para obtener $[\partial'_2 \mid \mathbf{\Pi}']$.

$$[\partial'_2 \mid \mathbf{\Pi}'] = \left(\begin{array}{c|cccccccccccccccc} 1 & \cdot & \cdot & \cdot & \cdot & 1 & 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 & 1 & 1 \\ \cdot & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot \\ \cdot & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{array} \right).$$

La salida del algoritmo son los ciclos en las columnas 1 y 8 de $\mathbf{\Pi}$, pues estas columnas tienen pivotes en $\mathbf{\Pi}'$. Por lo tanto, la base de homología óptima encontrada está dada por los ciclos $01 + 02 + 12$ y $23 + 24 + 34$.

11.1.3. Algoritmo HomOpt2

Primero calculamos una anotación de las aristas de K como en el algoritmo AnotAristas. Empleamos como árbol de expansión mínimo el árbol T_0 visto en la Figura 11.2. Como vimos en la primera fila de la ecuación 11.1, este árbol produce los ciclos $01 + 02 + 12$, $01 + 02 + 13 + 23$ y $01 + 02 + 13 + 24 + 34$. Definimos la matriz \mathbf{Z} cuyas columnas cuentan con los vectores coordenados

de estos 3 ciclos.

$$\mathbf{Z} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & \cdot & \cdot \\ \cdot & 1 & 1 \\ \cdot & 1 & \cdot \\ \cdot & \cdot & 1 \\ \cdot & \cdot & 1 \end{pmatrix}.$$

Aplicamos eliminación gaussiana a $[\partial_2 \mid \mathbf{Z}]$ para obtener $[\partial'_2 \mid \mathbf{Z}']$. Asignamos las filas 1 y 2 de \mathbf{Z}' a una matriz \mathbf{A} . La matriz \mathbf{A} guarda las anotaciones de las aristas que no pertenecen a T_0 , dadas por $\mathbf{a}(12) = (1, 0)$, $\mathbf{a}(23) = (1, 0)$ y $\mathbf{a}(34) = (0, 1)$. Fijamos $\mathbf{a}(e)$ nulo para el resto de aristas de K .

$$[\partial_2 \mid \mathbf{Z}] = \left(\begin{array}{c|cccc} \cdot & 1 & 1 & 1 \\ \cdot & 1 & 1 & 1 \\ 1 & 1 & \cdot & \cdot \\ 1 & \cdot & 1 & 1 \\ 1 & \cdot & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 \\ \cdot & \cdot & \cdot & 1 \end{array} \right) \xrightarrow{\text{FRER}} [\partial'_2 \mid \mathbf{Z}'] = \left(\begin{array}{c|cccc} 1 & \cdot & 1 & \cdot \\ \cdot & 1 & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot \end{array} \right) \quad \mathbf{A}$$

De igual forma que en el algoritmo anterior, construimos el conjunto de ciclos Π evidenciado en la ecuación 11.2. Para cada uno de los 15 ciclos encontrados, calculamos su anotación vía la suma de las anotaciones de sus aristas.

$$\begin{aligned} \mathbf{a}(01 + 02 + 12) &= (1, 0), \\ \mathbf{a}(12 + 13 + 23) &= (0, 0), \\ \mathbf{a}(23 + 24 + 34) &= (1, 1), \\ \mathbf{a}(01 + 02 + 13 + 23) &= (1, 0), \\ \mathbf{a}(12 + 13 + 23 + 34) &= (1, 1), \\ \mathbf{a}(01 + 02 + 13 + 24 + 34) &= (0, 1). \end{aligned}$$

Construimos la matriz \mathbf{B} que guarda en sus columnas estas anotaciones. Ordenamos las columnas de izquierda a derecha según la longitud de los ciclos, colocando en la izquierda los ciclos más pequeños. Este es el mismo orden visto en la ecuación 11.2.

$$\mathbf{B} = \begin{pmatrix} 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 & 1 & 1 & \cdot \\ \cdot & 1 & 1 & 1 & \cdot & \cdot & 1 & 1 & 1 \end{pmatrix}$$

Luego, encontramos la forma reducida escalonada por renglones de \mathbf{B} , dada por \mathbf{B}' .

$$\mathbf{B}' = \begin{pmatrix} 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 \\ \cdot & 1 & 1 & \cdot & \cdot & 1 & 1 & 1 \end{pmatrix}$$

La salida del algoritmo son los ciclos 1 y 8 de Π , pues estas columnas poseen pivote en \mathbf{B}' . Por lo tanto, la base de homología óptima encontrada está dada por los ciclos $01 + 02 + 12$ y $23 + 24 + 34$. Vemos que, en esta ocasión, las salidas de los algoritmos HomOpt1 y HomOpt2 coinciden.