

# NUDOS COMBINATORIOS Y MARIPOSAS\*

por

Margarita María Toro Villegas <sup>1</sup>

Dedicado a la memoria del profesor Jairo Charris C.

## Resumen

**Toro Villegas, Margarita María:** Nudos combinatorios y mariposas. Rev. Acad. Colomb. Cienc. **28** (106): 79–86, 2004. ISSN 0370-3908.

En este artículo mostraremos una aplicación del concepto de nudo combinatorio para obtener la  $n$ -mariposa asociada a un diagrama de  $n$  cruces de un nudo  $K$ . Se describirán los algoritmos implementados en el programa *Mathematica*.

**Palabras clave:** Nudos, enlaces, combinatoria, 3-variedades.

## Abstract

In this paper we use the concept of combinatorial knot in order to construct a  $n$ -butterfly associated to a knot diagram with  $n$  crosses. We describe an algorithm and its implementation in the software *Mathematica*.

**Key words:** Knots, links, combinatorial, 3-manifold.

## 1. Introducción

Este artículo tiene un carácter teórico-computacional. Nuestro resultado central, *todo nudo admite una presentación como una  $n$ -mariposa*, tiene una prueba algorítmica. Presentamos dicha prueba y desarrollamos e implementamos el algoritmo, utilizando el programa *Mathematica*. Para la implementación de dicho algoritmo, se utiliza el concepto de nudo combinatorio, desarrollado como herramienta computacional, ver [10] y

[11]. Se muestran mediante un ejemplo concreto, el nudo  $4_1$  o nudo del ocho, los resultados que se obtienen con nuestro programa. Este trabajo está estrechamente relacionado con el artículo *Mariposas y 3-Variedades*, ver [4], que aparece en este número de la Revista de la Academia Colombiana de Ciencias.

El artículo está organizado de la siguiente forma: en la primera sección definimos el concepto de nudo combinatorio. Como el trabajo está pensado para un público

---

<sup>1</sup>Universidad Nacional de Colombia-Sede Medellín. e-mail: mmtoro@perseus.unalmed.edu.co

\* Proyecto financiado parcialmente por COLCIENCIAS, proyecto 1118-05-13631, y por DINAIN, proyecto DI00C1236  
AMS 2000 Mathematics Subject Classification: Primary 57M25, Secondary: 57M15, 05C99.

no especializado en los conceptos de la teoría de nudos, en esta sección damos algunas definiciones básicas de la teoría de nudos. En la segunda sección mostramos unos ejemplos motivadores del concepto de  $n$ -mariposa. En la tercera sección damos una definición intuitiva del concepto de  $n$ -mariposa, (para un desarrollo formal ver [4], [5]) y probamos el resultado teórico central. En la siguiente sección describimos el algoritmo que nos permite construir una  $n$ -mariposa a partir de un nudo, y desarrollamos su implementación en *Mathematica*. Se muestran unos cómputos concretos, para el nudo  $4_1$ . En la última sección se hacen comentarios y se establecen algunas preguntas interesantes relacionadas con el tema, que muestran la dirección actual de nuestra investigación.

## 2. Nudos Combinatorios

En esta sección definimos el concepto de nudo combinatorio, como una extensión del concepto de nudo. Para ello recordaremos algunos conceptos básicos; recomendamos el libro de Adams, [1], para una introducción general e intuitiva a la teoría de nudos. Para un tratamiento formal, recomendamos [2], [6] y [7].

**Definición 1.**  $K \subset \mathbb{R}^3$  es un nudo si existe un homeomorfismo  $\phi : S^1 \hookrightarrow K \subset \mathbb{R}^3$ .

Un enlace es la imagen de una unión disjunta de copias de  $S^1$ . En este artículo trataremos especialmente nudos, pero toda la teoría se puede aplicar a enlaces. Esto se hará en detalle en un trabajo posterior.

**Definición 2.** Dos nudos  $K$  y  $K'$  son equivalentes si existe un homeomorfismo  $h : \mathbb{R}^3 \rightarrow \mathbb{R}^3$  tal que  $h(K) = K'$

Como una herramienta de trabajo para estudiar nudos se acostumbra utilizar la *proyección regular* del nudo en un plano, es decir, una proyección sobre un plano de tal forma que los segmentos que forman la proyección se intersecten transversalmente y la gráfica sólo tenga puntos dobles. Los siguientes son ejemplos de proyecciones de nudos.

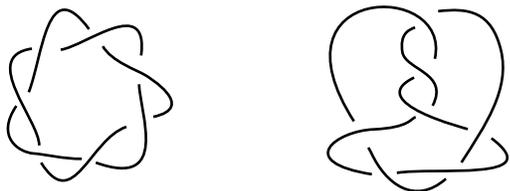


Figura 1

Dibujamos la proyección de tal manera, que en los puntos dobles, que llamaremos *cruces*, se indique, mediante una línea cortada, cuál segmento va por encima y cuál va por debajo. A los segmentos continuos, sin interrupción, se les acostumbra llamar *arcos*.

Los nudos se clasifican como “mansos” y “salvajes”. Los nudos mansos, o poligonales, son aquellos que admiten una proyección con un número finito de puntos doble. Nosotros estudiaremos sólo nudos mansos o poligonales. Es posible considerar nudos con o sin orientación, es decir, podemos asignarle al nudo un sentido en el cual recorrerlo. En ese caso, decimos que el nudo es orientado.

Al utilizar proyecciones de nudos, que se llaman también diagramas de nudos, cambiamos un poco el problema, y necesitamos garantizar que el estudio de los diagramas de nudos en efecto permite estudiar los nudos mismos. Afortunadamente tenemos el siguiente teorema, para la prueba ver [2].

**Teorema 3.** *Cualquier nudo poligonal  $K$  es equivalente, bajo una rotación arbitrariamente pequeña, a un nudo poligonal que admite una proyección regular.*

Estudiar los nudos según su proyección presenta una desventaja, que es fuente a su vez de muchas preguntas de la teoría: *Si dos nudos tienen la misma proyección, ellos son equivalentes, pero el recíproco no es cierto.* Por ejemplo, los dos diagramas siguientes corresponden al mismo nudo, que se acostumbra llamar *trivial* o *suelto*.



Figura 2.

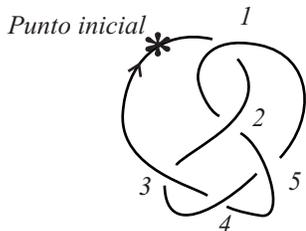
Además, Reidemeister resolvió el problema de la falta de unicidad al introducir los movimientos, conocidos hoy con su nombre, y probar el teorema: *Dos diagramas representan nudos equivalentes si y sólo si uno de ellos se puede deformar en el otro mediante una sucesión finita de movimientos de Reidemeister*, (ver [2]). Claro que éste es un resultado de carácter teórico, y en la práctica, dados dos diagramas no existe un algoritmo efectivo, que siempre permita decidir si los nudos que representan son o no equivalentes.

Este es uno de los problemas centrales de la teoría de nudos, que, aunque ha tenido muchos progresos en los últimos años, aún no está completamente resuelto.

A pesar de la dificultad que se presenta por la falta de unicidad al representar un nudo mediante un diagrama, históricamente los diagramas se han usado extensivamente para estudiar nudos.

Nosotros queremos desarrollar una codificación para los diagramas, que nos permita poder manipularlos en el computador y efectuar los cálculos y operaciones que están definidos para los diagramas de nudos.

Nuestra codificación consiste en tomar un punto inicial en el diagrama del nudo orientado y enumerar los cruces del nudo con enteros positivos  $\{a_1, a_2, \dots, a_n\}$ , típicamente usamos  $\{1, 2, \dots, n\}$ . Luego recorremos el nudo y escribimos una lista con los números de los cruces en el orden de recorrido, positivo si el cruce se recorre por encima y negativo si el cruce se recorre por debajo. Se termina al llegar de nuevo al punto inicial. De esta forma cada cruce se recorre dos veces y su correspondiente índice se escribe una vez con signo positivo y otra con signo negativo.



Cruces =  $\{-1, 2, -3, 4, -5, 1, -2, 5, -4, 3\}$

Figura 3.

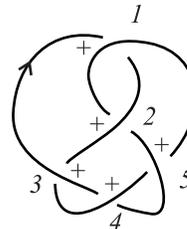
Como queremos que nuestra codificación determine orientación, adicionalmente a cada cruce se le asigna un signo, que está determinado por su orientación, siguiendo la regla de la mano derecha, de la forma indicada en la figura:



Figura 4.

La información sobre los signos de los cruces se almacena en una segunda lista, en la que cada posición representa el número del cruce. En la posición  $i$ -ésima se escribe un 1 si el cruce  $i$ -ésimo es positivo y un  $-1$

si el cruce es negativo. Así, el nudo anterior tiene como codificación:



$\{-1, 2, -3, 4, -5, 1, -2, 5, -4, 3\}, \{1, 1, 1, 1, 1\}$

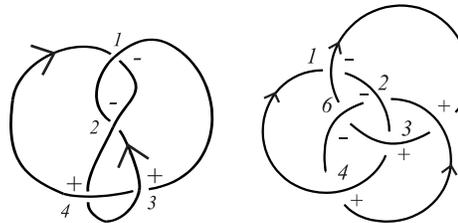
Figura 5.

**Definición 4.** La codificación de un diagrama de un nudo  $N$  es la asignación de una lista de listas  $\{\{i_1, i_2, \dots, i_{2n}\}, \{\epsilon_1, \epsilon_2, \dots, \epsilon_m\}\}$  tal que, como conjuntos,

$$\begin{aligned} \{i_1, i_2, \dots, i_{2n}\} &= \{a_1, a_2, \dots, a_n\} \cup \\ &\quad \{-a_1, -a_2, \dots, -a_n\}, \\ a_i \text{ un entero}, \epsilon_j &\in \{1, -1\}, \end{aligned}$$

A este tipo de listas las llamamos *nudos combinatorios*. Estos nudos combinatorios, aparte de permitirnos hacer muchos cálculos de operaciones e invariante de nudos, se han convertido en un objeto de estudio en sí mismos y son la fuente de preguntas muy interesantes, ver [10].

Los siguientes ejemplos muestran el nudo del ocho, o  $N_1$ , y los anillos de Borromeo. Este último es un enlace, y lo que hacemos con nuestra codificación es que la lista de cruces a su vez está dividida en sub-listas, una por cada una de las componentes del enlace.



$N_1 =$  Nudo del ocho       $N_2 =$  Anillos de Borromeo

Figura 6

Los respectivos nudos combinatorios son:

$$\begin{aligned} N_1 &= \{\{-1, 2, -4, 3, -2, 1, -3, 4\}, \{-1, -1, 1, 1\}\}. \\ N_2 &= \{\{\{-1, 2, -3, 4\}, \{1, -5, 3, -6\}, \{5, -2, 6, -4\}\}, \\ &\quad \{-1, -1, 1, 1, 1, -1\}\} \end{aligned}$$

La información referente a la orientación de cada cruce es altamente redundante, pero esta redundancia ayuda como forma de autocontrol. Note que como permitimos que el conjunto de índices para los cruces sea un subconjunto cualquiera  $S$  de  $\mathbb{N}$ , hay que tener cuidado con la sucesión que representa la orientación de los cruces, que tiene un sentido posicional: la orientación del cruce  $i$  se escribe en el puesto  $i$ . Por lo tanto, la lista de la orientación debe tener longitud igual a  $Max\{S\}$ , y contendrá información acerca de un conjunto de cruces ficticios. Esto no constituye ningún obstáculo en nuestro trabajo, al contrario, la posibilidad de trabajar

con cruces ficticios nos da una gran versatilidad para implementar conceptos de la teoría de nudos, como los movimientos de Reidemeister, ver [10].

Un aspecto fundamental que debemos considerar es que, dado un nudo, la codificación depende del punto inicial, de la forma de enumeración y, muy especialmente, del diagrama del nudo. Por ejemplo, el nudo  $N1$  de la Figura 4, se puede codificar como  $\{\{3, -2, 1, -3, 4, -1, 2, -4\}, \{-1, -1, 1, 1\}\}$ . El problema con el punto de origen se resuelve definiendo clases de equivalencia en el conjunto de todos los nudos combinatorios, de la siguiente forma

$$\begin{aligned} \{\{i_1, i_2, \dots, i_{2n}\}, \{\epsilon_1, \epsilon_2, \dots, \epsilon_m\}\} &\approx \{\{i_2, \dots, i_{2n}, i_1\}, \{\epsilon_1, \epsilon_2, \dots, \epsilon_m\}\} \\ &\approx \dots \\ &\approx \{\{i_{2n}, \dots, i_1, i_2\}, \{\epsilon_1, \epsilon_2, \dots, \epsilon_m\}\}. \end{aligned}$$

Esta equivalencia corresponde a la idea de lista circular, que es muy natural en *Mathematica*.

El problema de la forma de enumeración de los cruces, se resuelve mediante la relación de equivalencia en el conjunto de las codificaciones, determinada por:

$$\{\{i_1, i_2, \dots, i_{2n}\}, \{\epsilon_1, \epsilon_2, \dots, \epsilon_m\}\} \approx \{\{j_1, j_2, \dots, j_{2n}\}, \{\delta_1, \delta_2, \dots, \delta_s\}\} \tag{1}$$

si y sólo si existe una biyección  $\sigma : \{i_1, i_2, \dots, i_{2n}\} \rightarrow \{j_1, j_2, \dots, j_{2n}\}$  tal que, para todo  $k, 1 \leq k \leq 2n$

$$\begin{aligned} \sigma(-i_k) &= -\sigma(i_k) \\ \epsilon_{|i_k|} &= \delta_{|\sigma(i_k)|}. \end{aligned} \tag{2}$$

Es decir, que exista una reenumeración de índices.

Tomando el conjunto de nudos combinatorios y partiéndolo por las relaciones de equivalencia definidas, hemos resuelto el problema de la selección de índices y del punto de origen. De ahora en adelante usamos el término nudo combinatorio para referirnos a su clase de equivalencia bajo estas relaciones.

La dependencia del diagrama específico del nudo es un problema que depende de la teoría de nudos, y no de la forma de codificación, y es el mismo con el que se encontraron los matemáticos a principios del siglo XX al querer estudiar los nudos vía su proyección en un plano. Resolvemos este problema utilizando una adaptación de los movimientos de Reidemeister.

**2.1. Ejemplos Motivadores de Mariposas**

Considere el enlace  $N2$  de la Figura 4, que se conoce en la literatura con el nombre de anillos de Borromeo. Thurston probó que se puede obtener a partir de una poligonalización de la superficie de una bola, identificando

caras, mediante reflexiones a lo largo de aristas. En este caso tomamos un cubo y cada una de las caras las dividimos en dos, quedándonos así 12 caras. Hacemos identificación de las caras según lo indican las flechas,

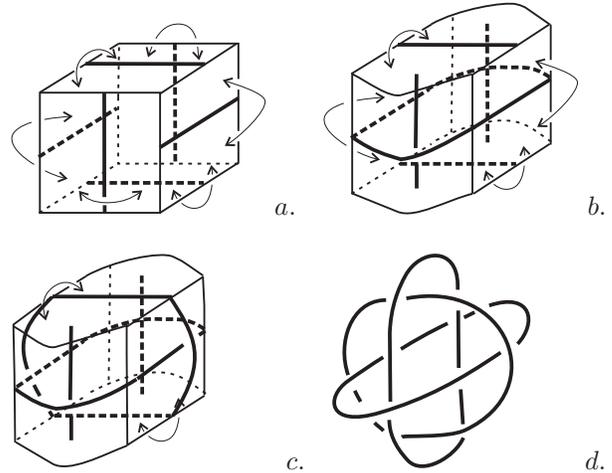


Figura 7

De la misma forma, todo nudo racional se puede obtener mediante un proceso similar, identificando caras, mediante reflexión a lo largo de aristas comunes, de la poligonalización de la superficie de una bola, esta vez mediante 4 hexágonos.

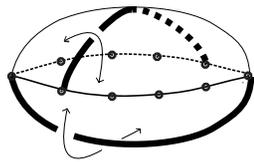


Figura 8

Si nombramos los vértices de los polígonos, tenemos un algoritmo que nos permite, mediante el uso de nudos combinatorios, producir los nudos o enlaces a partir de la poligonalización de la 3-bola y la identificación de las caras. Este algoritmo será presentado en un trabajo posterior, por ahora, nos interesa dedicarnos a la existencia de  $n$ -mariposas.

## 2.2. Resultado Central

Sin entrar en detalles formales, diremos que una  $n$ -mariposa es una 3-bola cuya frontera  $S$  está poligonalizada en  $2n$  polígonos, que se van a identificar por parejas, mediante reflexiones a lo largo de la arista en común. Los polígonos de la mariposa los llamamos *alas*, la unión de las aristas a lo largo de las cuales se hace la identificación la llamamos *tronco*.

En este sentido, los ejemplos anteriores dan lugar a los resultados:

- Todo nudo o enlace racional admite una representación como una 2-mariposa.
- Los anillos de Borromeo también admiten una representación como una 6-mariposa.

El resultado teórico central de éste artículo es probar que todo nudo o enlace admite una presentación como una  $n$ -mariposa, para algún  $n \in \mathbb{N}$ .

La prueba es constructiva, y provee un algoritmo para construir la mariposa a partir de un diagrama  $D$  del nudo. Por ahora, trabajamos sólo con nudos, pero el trabajo se puede extender a enlaces, y probablemente a grafos.

**Teorema 5.** *Todo nudo admite una representación de  $n$ -mariposa, para algún  $n \in \mathbb{N}$ .*

*Demostración:* Sea  $K$  un nudo o enlace en  $S^3 = \mathbb{R}^3 \cup \{\infty\}$ , de tal manera que su proyección regular esté

totalmente contenida en un plano  $P$  unido con un número finito de pequeñas bolas, que son vecindades de sus cruces. Tomamos ahora un punto  $B$  que no pertenezca a la unión del plano  $P$  con las bolas alrededor de los cruces, y formamos el cono del nudo con vértice  $B$ .

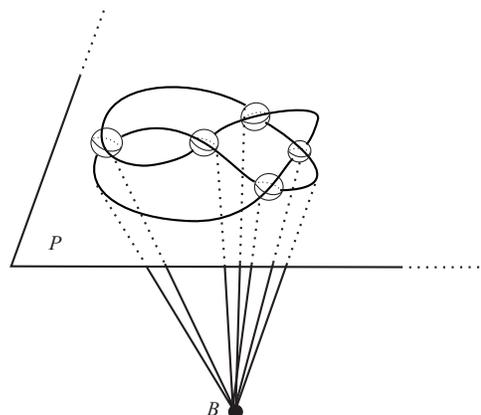


Figura 9

Al cortar a lo largo del cono, se obtiene precisamente una bola con caras, que al ser identificadas por reflexiones a lo largo de los arcos provenientes del diagrama del nudo, define la mariposa.

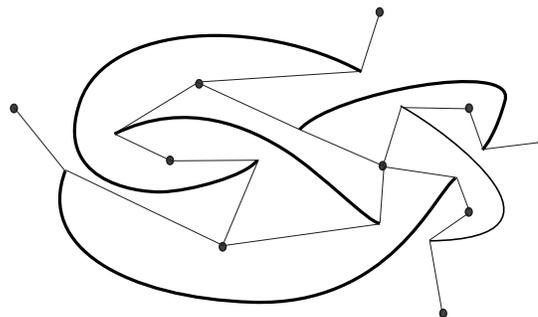


Figura 10

**Nota:** Esta prueba puede aplicarse también a enlaces, pero hay que hacer restricciones cuando la proyección del enlace no es conexa.

**Corolario 6.** *Un nudo no trivial  $K$  con  $n$ -cruces admite una representación de  $n$ -mariposa.*

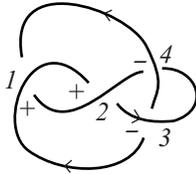
La prueba de este resultado es constructiva, y permite dar un algoritmo para construir la mariposa a partir de un diagrama  $D_K$  del nudo  $K$  orientado. El número de mariposa coincide con el número de cruces del diagrama.

Este algoritmo es el que queremos implementar en *Mathematica*. Lo interesante es que al implementar el algoritmo, se van aclarando algunos puntos teóricos y surgen preguntas nuevas.

### 3. Implementación del algoritmo

Simplemente reproducimos la prueba del teorema anterior, en forma algorítmica. Para cada uno de los pasos del algoritmo describimos cómo se hizo la implementación. No mostramos códigos, pero sí los resultados obtenidos por los programas para el nudo del ocho, con el que se ilustra el algoritmo.

**Paso 1.** Tomamos un diagrama  $D_K$  del nudo  $K$ , que es una colección de arcos disjuntos orientados. Para implementarlo en el computador, tomamos el nudo combinatorio.



$$\text{ocho} = \{\{1, -2, 3, -4, 2, -1, 4, -3\}, \{1, 1, -1, -1\}\}$$

Figura 11

Note que el nudo que estamos usando para los ejemplos, es la imagen espejo del que usamos en la Figura 5.

**Paso 2.** Tomamos las regiones  $R_i$ , determinadas por la proyección del nudo en el plano, elegimos un punto en cada región, que denotamos  $B[i]$ . En la región no acotada, tomamos el punto en el infinito.

Para determinar las regiones del plano, la información dada por el nudo combinatorio no es suficiente y, por lo tanto, debemos hacer algunos ajustes en la codificación.

**Cambio de la codificación:** Reemplazamos el cruce  $i$  por  $c[i]$ , e introducimos dos nuevos vértices:  $a[i,-1]$  y  $a[i,1]$ , que son puntos correspondientes al cruce  $i$  cuando se recorre por debajo, con la convención que  $a[i,-1]$  se recorre antes que  $a[i,1]$ . Al identificar las alas, estos puntos quedan identificados. Al crear la nueva lista, incorporamos la información de los signos de los cruces en el signo que le asignamos a  $c[i]$ . No se crea confusión, ya que la información de cuándo se pasa por encima y cuándo por debajo ya está consignada en los vértices de tipo  $a$ .

Creamos una nueva lista en la cual efectuamos los reemplazos

$$i \rightarrow -c[i]*\text{signocruce}(i)$$

$$-i \rightarrow (a[i,-1],c[i]*\text{signocruce}(i),a[i,1])$$

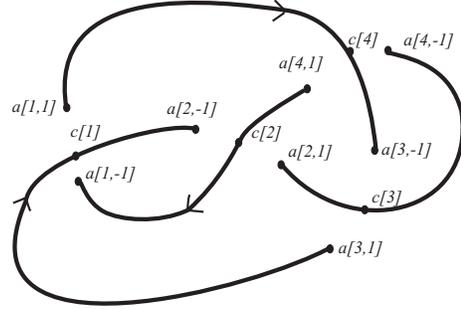


Figura 12

El programa que hace ésto ya está implementado para trabajar con enlaces. El resultado en el caso del nudo del ocho es:

$$\text{cambiarnotacion}[\text{ocho}] = \{-c[1], a[2, -1], c[2], a[2, 1], c[3], a[4, -1], -c[4], a[4, 1], -c[2], a[1, -1], c[1], a[1, 1], c[4], a[3, -1], -c[3], a[3, 1]\}$$

Para producir las regiones en las que la proyección del nudo divide el plano, construimos un complejo simplicial, en el cual los vértices son todos los vértices de la codificación.

Los lados son las parejas de vértices consecutivos, recordando que la lista es circular. En nuestro caso tenemos:

$$\text{lados} = \{\{-c[1], a[2, -1]\}, \{a[2, -1], c[2]\}, \{c[2], a[2, 1]\}, \{a[2, 1], c[3]\}, \{c[3], a[4, -1]\}, \{a[4, -1], -c[4]\}, \{-c[4], a[4, 1]\}, \{a[4, 1], -c[2]\}, \{-c[2], a[1, -1]\}, \{a[1, -1], c[1]\}, \{c[1], a[1, 1]\}, \{a[1, 1], c[4]\}, \{c[4], a[3, -1]\}, \{a[3, -1], -c[3]\}, \{-c[3], a[3, 1]\}, \{a[3, 1], -c[1]\}\}$$

Las caras del complejo simplicial se construyen de la siguiente forma: Empezamos a recorrer los lados, en sentido positivo, si se llega a un vértice de tipo  $a[i,\pm 1]$  continuamos con el vértice que sigue en la codificación, pero si llegamos a un vértice de tipo  $c[i]$  “viramos a la izquierda”, como se muestra en la figura 13.

Repetimos este proceso hasta haber recorrido todos los lados 2 veces, una en sentido positivo y otra en sentido negativo. De esta forma hemos construido una partición del plano en  $n+2$  regiones.

Implementamos este proceso en un programa, llamado **producirregionesmar**, que toma el nudo combinatorio y produce las regiones.

El programa **producirregionesmar** de hecho está diseñado para trabajar con cualquier nudo (o enlace) combinatorio, y calcula una poligonalización de una superficie orientable de genero mínimo, sobre la cual se puede proyectar el nudo combinatorio.

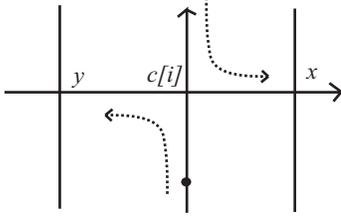


Figura 13

En nuestro caso tenemos

$$\begin{aligned} \text{regiones} = & \{ \{a[2, -1], c[2], a[4, 1], c[4], a[1, 1], c[1]\}, \\ & \{a[1, -1], c[2], a[2, -1], c[1]\}, \\ & \{c[1], a[3, 1], c[3], a[2, 1], c[2], a[1, -1]\}, \\ & \{c[4], a[4, -1], c[3], a[3, 1], c[1], a[1, 1]\}, \\ & \{c[3], a[3, -1], c[4], a[4, 1], c[2], a[2, 1]\}, \\ & \{c[3], a[4, -1], c[4], a[3, -1]\} \end{aligned}$$

Nótese que son 6 regiones.

Por cada una de estas regiones tomamos un vértice  $b[i]$ . No es relevante saber cual corresponde a la región no acotada. Para simplicidad del diagrama, repetimos este vértice.

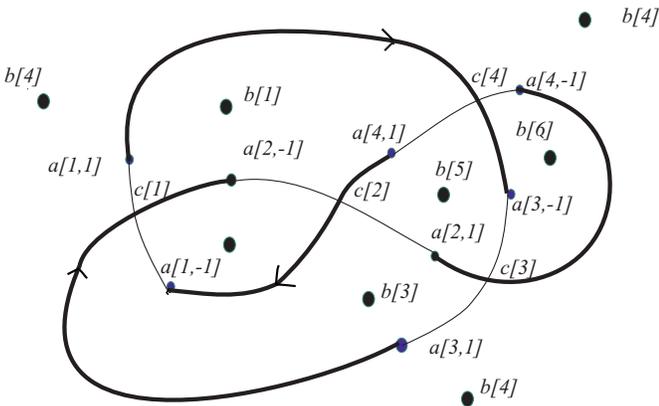


Figura 14

**Paso 3.** Los extremos de cada uno de los arcos del diagrama, que llamamos vértices de tipo  $A$ , los unimos con los puntos  $B(i)$ , de las dos regiones adyacentes al arco. Así el plano está poligonalizado. Este proceso lo implementamos en dos procesos independientes:

**a.** Producir los arcos, que serán el Tronco de la mariposa. Se usa el programa **prodarcos**, que a partir de un nudo combinatorio produce los arcos.

Los arcos se determinan por sus extremos, pero para poder construir los polígonos, debemos tener en cuenta sobre cuales cruces pasa cada arco. En este caso no importa el signo de los vértices de tipo  $c[i]$ .

$$\text{arcos} = \{ \{a[2, 1], c[3], a[4, -1]\}, \{a[4, 1], c[2], a[1, -1]\}, \{a[1, 1], c[4], a[3, -1]\}, \{a[3, 1], c[1], a[2, -1]\} \}$$

**b.** Construir las alas. Se usa un programa, que requiere como parámetros de entrada las regiones y los arcos, escritos con la convención indicada.

Eliminamos de las regiones los vértices de la forma  $c[i]$ , y se construye la poligonalización de la superficie uniendo, para cada región  $j$ , el punto  $b[j]$  con los puntos extremos de los lados que forman la región. Estos son los puntos extremos de los arcos, que son del tipo  $a[i, \pm 1]$ . De esta forma se construyen los lados de los polígonos. El tronco está formado por los arcos, al eliminar los vértices de la forma  $c[i]$ .

En nuestro caso tenemos la 4-mariposa dada por:

**mariposa:**

$$\begin{aligned} & \{ \{a[2, 1], a[4, -1]\}, \{b[3], a[2, 1]\}, \\ & \{b[3], a[3, 1]\}, \{b[4], a[3, 1]\}, \{b[4], a[4, -1]\} \}, \\ & \{ \{a[2, 1], a[4, -1]\}, \{b[5], a[2, 1]\}, \\ & \{b[5], a[3, -1]\}, \{b[6], a[3, -1]\}, \{b[6], a[4, -1]\} \}, \\ & \{ \{a[4, 1], a[1, -1]\}, \{b[1], a[4, 1]\}, \{b[1], a[2, -1]\}, \\ & \{b[2], a[2, -1]\}, \{b[2], a[1, -1]\} \}, \\ & \{ \{a[4, 1], a[1, -1]\}, \{b[5], a[4, 1]\}, \{b[5], a[2, 1]\}, \\ & \{b[3], a[2, 1]\}, \{b[3], a[1, -1]\} \}, \\ & \{ \{a[1, 1], a[3, -1]\}, \{b[1], a[1, 1]\}, \{b[1], a[4, 1]\}, \\ & \{b[5], a[4, 1]\}, \{b[5], a[3, -1]\} \}, \\ & \{ \{a[1, 1], a[3, -1]\}, \{b[4], a[1, 1]\}, \{b[4], a[4, -1]\}, \\ & \{b[6], a[4, -1]\}, \{b[6], a[3, -1]\} \}, \\ & \{ \{a[3, 1], a[2, -1]\}, \{b[3], a[3, 1]\}, \{b[3], a[1, -1]\}, \\ & \{b[2], a[1, -1]\}, \{b[2], a[2, -1]\} \}, \\ & \{ \{a[3, 1], a[2, -1]\}, \{b[4], a[3, 1]\}, \{b[4], a[1, 1]\}, \\ & \{b[1], a[1, 1]\}, \{b[1], a[2, -1]\} \} \end{aligned}$$

$$\text{tronco} = \{ \{a[2, 1], a[4, -1]\}, \{a[4, 1], a[1, -1]\}, \{a[1, 1], a[3, -1]\}, \{a[3, 1], a[2, -1]\} \}$$

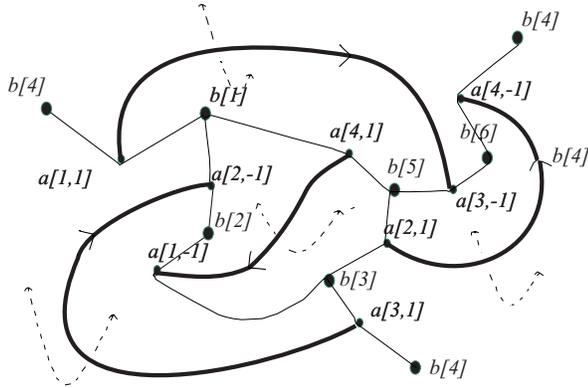


Figura 15

**Paso 4.** Sobre el plano  $P$  suponemos que hay una 3-bola cuya frontera es el plano poligonizado.

**Paso 5.** Las caras adyacentes a cada arco  $\alpha$  se identifican por una reflexión a lo largo de  $\alpha$ , para cada arco  $\alpha$  que se encuentre en el tronco. Para construir la mariposa, se identifican las alas simétricas a lo largo del arco común.

**Fin del algoritmo**

#### 4. Comentarios

Este trabajo presenta muchas ideas nuevas que aún se están desarrollando. Se presentan los conceptos de una forma intuitiva, pero todos pueden ser formalizados, ver [5]. Se tienen una gran cantidad de preguntas sobre las cuales estamos trabajando. Por ejemplo:

¿Cómo reducir el número de alas de una  $n$ -mariposa?

¿Qué variaciones hay al extender los resultados a enlaces y a grafos?

Vimos en los ejemplos que los vértices de la  $n$ -mariposa tienen distinto comportamiento. ¿Cómo clasificar los vértices?

Se sabe que todo diagrama de un nudo da lugar a un nudo combinatorio, pero existen nudos combinatorios que no provienen de diagramas. ¿Para este tipo de nudos combinatorios, qué sentido tiene aplicar los algoritmos descritos en este trabajo?

#### Referencias

- [1] **Adams, C. C.** *The Knot Book: An elementary introduction to the Mathematical Theory of Knots*. W. H. Freeman and Company-New York, 2001
- [2] **Burde, G. and Zieschang, H.** *Knots*, Walter de Gruyter, Berlin-New York, 1985.
- [3] **Cairns, G. and Elton, D.** *The Planarity Problem for Signed Gauss Words*, Journal of Knot Theory and its Ramifications, **Vol. 2**, No. 4 (1993) 359-367.
- [4] **Hilden, M., Montesinos, J., Tejada, D. y Toro, M.**, *Mariposas y 3-Variedades*, Festival Académico, dedicado a la memoria del Profesor J. Charris, Universidad Nacional de Colombia, Bogotá, agosto, 2003.
- [5] **Hilden, M., Montesinos, J., Tejada, D. y Toro, M.**, *Butterflies*, Preprint, 2003
- [6] **Murasugi, K.** *Knot Theory and its Applications*, Birkhäuser, Boston-Basel-Berlin, 1996.
- [7] **Rolfsen, D.** *Knot and Links*, Publish or perish, Inc., 1976.
- [8] **Tejada, D.** *Variedades, Triangulaciones y Representaciones*, Preprint, 2003.
- [9] **Thurston, W.** *Three Dimensional Geometry and Topology*, Princeton, NJ, 1997.
- [10] **Toro, M.** *Programación en Mathematica con aplicaciones a la Teoría de Nudos*, Preprint, 2003.
- [11] **Toro, M.** *Clasificación de nudos: Una aproximación computacional*. Preprint, 2002.