

MATEMÁTICA DISCRETA

Teoría de Grafos

Grado en Matemáticas

2021 - 2022

3. BLOQUE III: TEORÍA DE GRAFOS

3.1. Tema 6: Grafos y su representación. Isomorfismo. Caminos y ciclos. Grafos eulerianos y hamiltonianos.

El término grafo fue introducido en 1887 por Sylvester, en el contexto de análisis algebraico de estructuras moleculares. No obstante, el origen de la teoría de grafos puede remontarse unos 150 años cuando Euler en 1736, publicó un trabajo en el cual resolvía el que ha venido a llamarse problema de los puentes de Königsberg. La ciudad de Königsberg (Kaliningrado, durante la época soviética) estaba dividida por el río Pregel en cuatro zonas: las dos orillas, la isla llamada Kneiphof y la parte comprendida entre las dos bifurcaciones del río. En aquel entonces existían siete puentes comunicando las distintas zonas tal y como se observa en la figura 2.

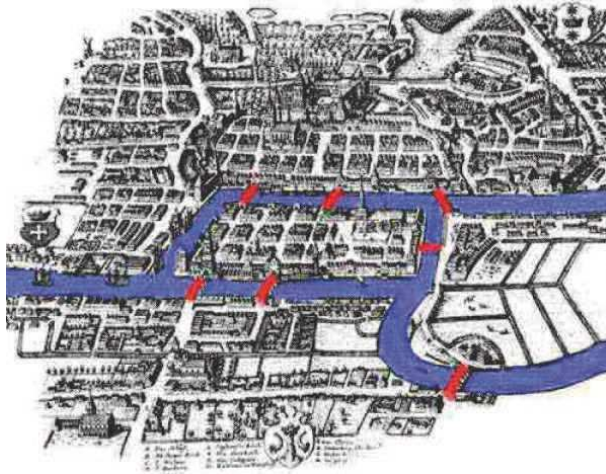


Figura 2: Los 7 puentes de Königsberg

Parece ser que uno de los pasatiempos de los ciudadanos de Königsberg consistía en dar un paseo que cruzara cada puente exactamente una vez. Se pensaba que tal paseo no era posible habida cuenta de los múltiples intentos fallidos pero nadie había podido demostrar tal imposibilidad. Fue Euler quien demostró este hecho. Euler reemplazó el mapa de la ciudad por un diagrama más simple, donde se incluye la información relevante del problema. No llegó a dibujar el diagrama de puntos y líneas que hoy se conoce como grafo y enunció, pero no demostró, que la condición necesaria dada por él también

era suficiente. Su demostración se debe a Hierholzer en 1871.

Desde entonces, los resultados y aplicaciones de la teoría de grafos han ido aumentando y los grafos se utilizan en contextos muy diferentes, construir redes de comunicaciones, implementación de circuitos en el plano, distinción de compuestos químicos con igual fórmula molecular, redes de transporte, asignación de horarios, diseño de estructuras de datos, etc.

Grafos y su representación.

Definición 6.29 *Un grafo (finito) es una terna $G = (V, A, \varphi)$ donde $V = V(G)$ es un conjunto finito (cuyos elementos se denominan vértices o nodos) no vacío, $A = A(G)$ es un conjunto finito cuyos elementos se llaman aristas, y $\varphi : A \rightarrow P(V)$ es una aplicación que asocia a cada arista un par de vértices. Es decir $\varphi(a) = \{x, y\}$ con $a \in A$ y $x, y \in V$.*

En estas condiciones se dice que la arista a une x con y , o que los extremos de a son x e y , o que a es incidente con x e y , o incluso que x e y son adyacentes por a .

Se dice que la arista a es un lazo si $\varphi(a) = \{x, x\}$ para un cierto $x \in V$. Se dice que un vértice es aislado si no existen aristas incidentes con él. Se dice que dos aristas son adyacentes o contiguas si tienen un extremo en común.

Ejemplo 6.30 *Algunos grafos particulares tienen características especiales y reciben un nombre en concreto. A continuación se detallan algunos de ellos cuya representación gráfica se muestra en la figura 3.*

i.) *El grafo completo: K_n .*

Vértices: $V = \{x_1, x_2, \dots, x_n\}$,

Aristas: $A = \{a_{ij}, 1 \leq i < j \leq n$ tal que $\varphi(a_{ij}) = \{x_i, x_j\}$.

ii.) *El grafo lineal: L_n .*

Vértices: $V = \{x_1, x_2, \dots, x_n\}$,

Aristas: $A = \{a_i, 1 \leq i \leq n - 1$ tal que $\varphi(a_i) = \{x_i, x_{i+1}\}$.

iii.) *El grafo ciclo: C_n .*

Vértices: $V = \{x_1, x_2, \dots, x_n\}$,

Aristas: $A = \{a_i, 1 \leq i \leq n$ tal que $\varphi(a_i) = \{x_i, x_{i+1}\}$ si $i < n$ y $\varphi(a_n) = \{x_n, x_1\}$.

iv.) *Grafo rueda: R_n .*

Vértices: $\{x_0, x_1, x_2, \dots, x_n\}$,

Aristas: $A = \{a_i, b_j\}, 1 \leq i, j \leq n$ tal que a_i es como en el grafo ciclo y $\varphi(b_j) = \{x_o, x_j\}$.

v.) El grafo bipartito completo: $K_{n,m}$.

Vértices: $\{x_1, x_2, \dots, x_n, y_1, y_2 \dots y_m\}$,

Aristas: $A = \{a_{ij}\}, 1 \leq i \leq n, 1 \leq j \leq m$ tal que $\varphi(a_{ij}) = \{x_i, y_j\}$.

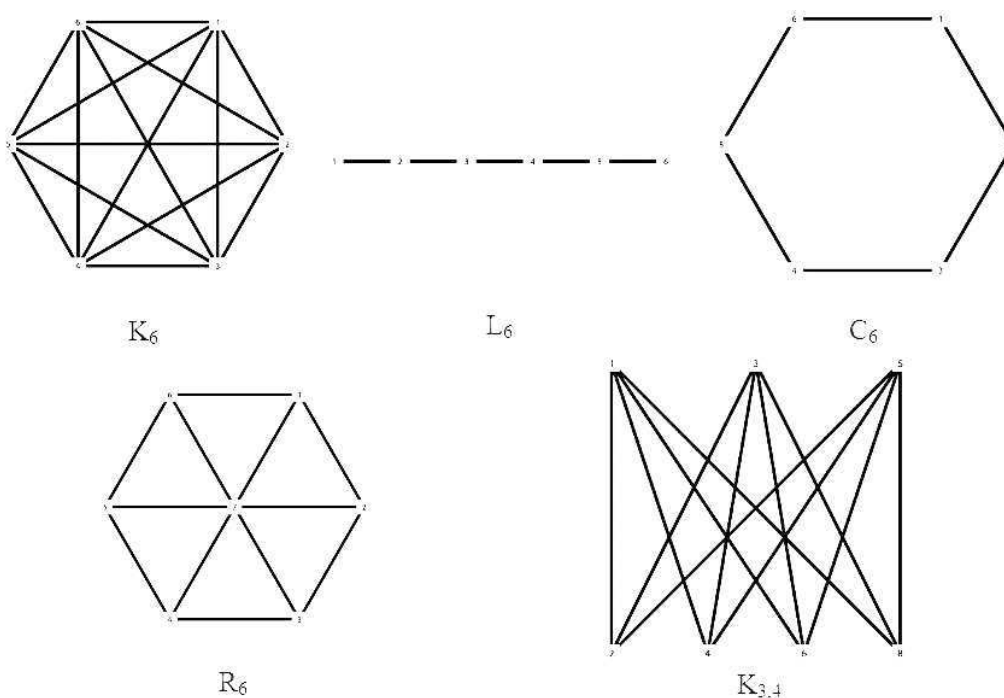


Figura 3: Algunos grafos notables

En general, un grafo G se dice bipartito si existe una partición de V , $V = V_1 \oplus V_2$ (es decir $V_1 \cap V_2 = \emptyset$ y $V_1 \cup V_2 = V$) tal que todas las aristas de G unen un vértice de V_1 con un vértice de V_2 .

Definición 6.31 Diremos que un grafo es simple si no hay lazos y dos vértices están unidos a lo sumo por una única arista.

Llamamos aristas múltiples a aquellas aristas repetidas, es decir existen $a_1, a_2, \dots, a_k \in A$ tal que $\varphi(a_1) = \varphi(a_2) = \dots = \varphi(a_k)$.

En algunas situaciones reales en las que se usan grafos, las aristas además tienen un sentido o dirección; es decir, salen de un vértice y llegan a otro (algo así como el sentido de circulación).

Definición 6.32 Un grafo dirigido o digrafo es una terna $G = (V, A, \varphi)$ donde V, A son como en la definición de grafo y $\varphi : A \rightarrow V \times V$ es una aplicación que asocia a cada arista un par de vértices ordenados. Es decir $\varphi(a) = (x, y)$ con $a \in A$ y $x, y \in V$. Es decir una arista que empieza en x y acaba en y .

En este sentido, las aristas de un grafo dirigido representan una dirección. Por tanto, se ha de distinguir entre dos aristas a, b incidentes a los mismos dos vértices, x, y pero tal que $\varphi(a) = (x, y)$ y $\varphi(b) = (y, x)$.

Existen diversas maneras de representar o determinar un grafo. En primer lugar, un grafo puede ser visto como la terna (V, A, φ) vista en la definición 6.29, o en la definición 6.32 si se trata de un grafo dirigido. Sin embargo, ésta no es la forma mas adecuada generalmente para trabajar con estas estructuras. A continuación se enumeran tres maneras de expresar un grafo.

i.) Representación Grafica.

Los grafos se pueden representar mediante un dibujo. Cada vértice del grafo se representa por un punto y se unen dos vértices mediante una línea/arco si hay una arista incidente a ambos. En general esta representación se puede realizar en \mathbb{R}^n y a veces en \mathbb{R}^2 como se verá mas adelante.

Esta representación es la más natural de todas ellas, sin embargo no hay que confundir el grafo con el dibujo; un mismo grafo puede tener dos representaciones distintas como se muestra en la figura 4.

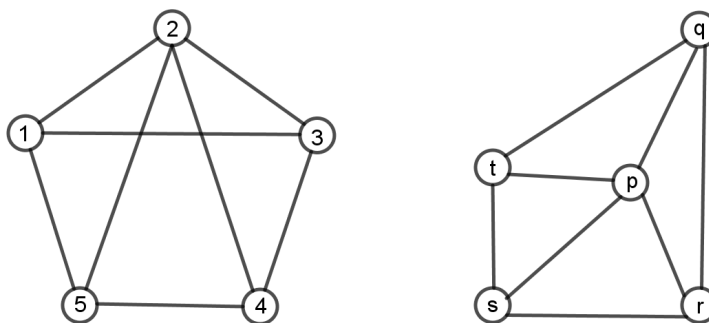


Figura 4: Dos representaciones del mismo grafo

ii.) Matriz de adyacencia.

Sea $G = (V, A, \varphi)$ un grafo. Se construye una matriz de tamaño $n \times n$ para el grafo. La entrada (i, j) , $i \neq j$, de la matriz es el número de aristas que unen el vértice i con el vértice j . Si $i = j$, el término correspondiente de la diagonal principal es dos veces el número de lazos del vértice i .

En un grafo no dirigido esta matriz es siempre simétrica.

iii.) Matriz de incidencia.

Sea $G = (V, A, \varphi)$ un grafo. Se construye una matriz $n \times m$ con $n = |V|$ y $m = |A|$. La entrada (i, j) de la matriz vale 1 si la arista i es incidente al vértice j . Vale 2 si la arista i es un lazo en el vértice j y 0 en caso contrario.

Existe una relación entre ambas matrices que viene dada por el siguiente teorema.

Teorema 6.33 *Sea G un grafo y A_G, B_G sus matrices de adyacencia e incidencia respectivamente. Entonces*

$$B_G^t B_G = A_G + \begin{pmatrix} d_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & d_n \end{pmatrix}$$

donde d_i representa el número de aristas incidentes en el vértice i -ésimo.

Estos valores d_i juegan un papel importante en un grafo como se verá por ejemplo en la determinación de ciclos y/o caminos eulerianos de un grafo.

Definición 6.34 *Sea G un grafo y $x \in V$ un vértice del grafo. Se denomina grado del vértice x , y se denota por $d(x)$, al número de aristas incidentes a x , entendiendo que un lazo aporta 2 al grado. Para digrafos se consideran el grado de entrada, $d_-(x)$, como el número de aristas que llegan a x , y el grado de salida, $d_+(x)$, el número de aristas que salen de x .*

Teorema 6.35 (del apretón de manos) *La suma de todos los grados de los vértices de un grafo es igual al doble del número de aristas del grafo, es decir:*

$$\sum_{x \in V} d(x) = 2|A|$$

Corolario 6.36 *En un grafo hay siempre un número par de vértices de grado impar (considerando el 0 como par).*

Teorema 6.37 *Para un digrafo se tiene que (las que entran por las que salen)*

$$\sum_{x \in V} d_-(x) = \sum_{x \in V} d_+(x)$$

Lema 6.38 *En todo grafo simple con más de un vértice hay al menos dos vértices del mismo grado.*

De los resultados anteriores se deduce fácilmente que no todas las diferentes combinaciones de grados son posibles para un grafo. El teorema siguiente proporciona una condición necesaria y suficiente para que una combinación de enteros no negativos sea la correspondiente a un grafo. Asimismo, esta caracterización permite construir el grafo correspondiente.

Teorema 6.39 (Havel-Hakimi) *Una secuencia de enteros no negativos $d_1 \leq d_2 \leq \dots \leq d_n = \delta$ es la secuencia de grados correspondiente a un grafo si y solo si la secuencia siguiente*

$$d_1, d_2, \dots, d_{n-\delta} - 1, \dots, d_{n-2} - 1, d_{n-1} - 1$$

es la secuencia de grados de algún otro grafo.

Definición 6.40 *Un grafo se dice regular si todos los vértices tienen el mismo grado.*

A partir de un grafo es posible obtener más grafos mediante diversas construcciones como se muestra a continuación

Definición 6.41 *Dados dos grafos $G_1 = (V_1, A_1, \varphi_1)$ y $G_2 = (V_2, A_2, \varphi_2)$ se dice que G_2 es subgrafo de G_1 si $V_2 \subset V_1$, $A_2 \subset A_1$ y $\varphi_1|_{A_2} = \varphi_2$.*

Ejemplos

- i.) Dados G un grafo y $v \in V(G)$ denotamos por $G - \{v\}$ al grafo obtenido a partir de G al eliminar el vértice v y las aristas incidentes a él.
- ii.) Dados G un grafo y $a \in A(G)$ denotamos por $G - \{a\}$ al grafo obtenido a partir de G al eliminar la arista a (solo la arista).
- iii.) Dado G un grafo y $V' \subset V(G)$ un subconjunto de vértices de él, el subgrafo generado por V' es el grafo con vértices los de V' y aristas aquellas de G con ambos extremos en V' . Se dice que H es un subgrafo generador de G si $V(H) = V(G)$.

- iv.) Dado un grafo G , llamamos grafo complementario de G , G^c al grafo que posee los mismos vértices de G y cuyo conjunto de aristas es el formado por las aristas que no tiene G . Es decir, si dos vértices no son adyacentes en G si lo son en G^c y recíprocamente.
- v.) Dados dos grafos G y G' con vértices disjuntos, el grafo union es aquel que posee como vértices los de ambos grafos y aristas las de los dos. El grafo suma es un grafo con vértices los vértices de G y los de G' y como aristas además de las de ambos grafos, también tiene todas las aristas que unen vértices de G con vértices de G' .

Es fácil ver por ejemplo que $K_n + K_m = K_{n+m}$.

Isomorfismo de grafos.

Como se ha visto con anterioridad, el mismo grafo puede representarse de muchas formas. Por tanto, es natural preguntarse cuando dos representaciones corresponden al mismo grafo, o más concretamente, cuando se trata del mismo grafo.

Definición 6.42 Sean $G_1 = (V_1, A_1, \varphi_1)$ y $G_2 = (V_2, A_2, \varphi_2)$ dos grafos, se dice que son isomorfos si existen biyecciones $\phi : V_1 \rightarrow V_2$ y $\tilde{\phi} : A_1 \rightarrow A_2$ tal que si a es una arista de G_1 que une x con y entonces $\tilde{\phi}(a)$ es una arista de G_2 que une $\phi(x)$ con $\phi(y)$.

Determinar si dos grafos (o representaciones) son isomorfos no es un problema que se pueda resolver de forma rápida. No obstante, se tiene necesariamente que cumplir ciertas condiciones para que dos grafos sean isomorfos.

Teorema 6.43 El número de vértices, número de aristas, secuencia de grados y el número de vértices con un grado determinado son invariantes de un grafo.

No se conoce un sistema completo de invariantes de un grafo; es decir una colección de datos de un grafo que permite conocer si dos grafos son isomorfos.

El siguiente resultado puede ayudar a determinar si dos grafos son isomorfos.

Proposición 6.44 Sean $G_1 = (V_1, A_1, \varphi_1)$ y $G_2 = (V_2, A_2, \varphi_2)$ dos grafos isomorfos.

- i.) Para todo subgrafo H_1 de G_1 existe un subgrafo H_2 de G_2 isomorfo a H_1 .

- ii.) El subgrafo de G_1 inducido por un conjunto de vértices V' es isomorfo al subgrafo de G_2 inducido por el conjunto de vértices $\phi(V')$.
- iii.) $d_{G_1}(x) = d_{G_2}(\phi(x))$.
- iv.) El subgrafo de G_1 inducido por el conjunto de vértices de un grado fijo es isomorfo al subgrafo de G_2 inducido por los vértices de ese grado.
- v.) Si $x \in V_1$ entonces $G_1 - \{x\}$ es isomorfo a $G_2 - \{\phi(x)\}$.
- vi.) Si $a \in A_1$ entonces $G_1 - \{a\}$ es isomorfo a $G_2 - \{\tilde{\phi}(a)\}$.

Caminos y ciclos

Como ya se ha indicado anteriormente, la noción de grafo y sus primeros resultados surgieron a la hora de resolver el problema de los puentes de Königsberg. Es decir, determinar si era posible recorrer cada uno de los puentes sobre el río Pregel una sola vez y volver al punto de partida. La figura 5 muestra el grafo correspondiente a este problema.

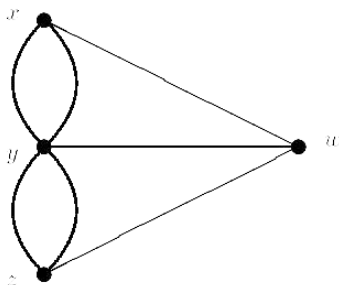


Figura 5: El grafo correspondiente a los puentes de Königsberg

Es necesario introducir para ello las nociones necesarias sobre recorridos en grafos.

Definición 6.45 Sea $G = (V, A, \varphi)$ un grafo.

- i.) Un recorrido en G es una sucesión $x_0 a_1 x_1 a_2 x_2 \cdots x_{k-1} a_k x_k$ donde $x_i \in V$ y a_i es una arista incidente a x_{i-1} y a x_i .
- ii.) Un recorrido se dice camino si todas las aristas son distintas.
- iii.) Se dice cerrado si $x_0 = x_k$ (acaba donde empieza).

- iv.) Se dice simple si no hay vértices repetidos salvo quizás el primero y el último.
- v.) Se denomina circuito si es un camino cerrado.
- vi.) Se denomina ciclo si es un camino simple cerrado no trivial (no es lazo).

Un circuito no es un ciclo, en el circuito se pueden repetir vértices mientras que en el ciclo no (aunque ambos parten y llegan al mismo sitio).

La longitud de un recorrido viene dada por el número de aristas del recorrido. Lógicamente, la matriz de adyacencia de un grafo muestra los recorridos de longitud 1. Recorridos de mayor longitud también están ligados a la matriz de adyacencia como se muestra a continuación.

Teorema 6.46 Sea $G = (V, A, \phi)$ un grafo y M su matriz de adyacencia. El elemento (i, j) de la matriz M^k , $k \in \mathbb{N}$ indica el número de recorridos de longitud k en el grafo G del vértice i al vértice j .

Definición 6.47 Se dice que un grafo es conexo si dos vértices cualesquiera se pueden unir por un camino.

La noción de conexión de un grafo se muestra, lógicamente, de manera muy visual en la representación gráfica, un grafo es conexo si no está dividido en trozos. No obstante, no siempre es fácil descubrirlo pues depende de la posición de los vértices en la representación.

Corolario 6.48 Sea G un grafo con n vértices y M su matriz de adyacencia. G es conexo si y solo si la matriz $Id_n + M + M^2 + \dots + M^{n-1}$ no tiene ceros.

Dado un grafo G y dos vértices $x, y \in G$, la relación $x \sim y$ si y solo si existe un recorrido de x a y es una relación de equivalencia. Los subgrafos de G generados por los vértices de las diferentes clases de equivalencia son precisamente las componentes conexas de un grafo. El número de componentes conexas de un grafo está relacionado con el número de vértices y aristas como se muestra a continuación:

Teorema 6.49 Sea G un grafo con n vértices, m aristas y k componentes conexas. Entonces $m \geq n - k$.

Grafos eulerianos

Volviendo al problema de recorrer las aristas de un grafo, ¿cuando existe un circuito que recorre todas las aristas?

Definición 6.50 *Un circuito euleriano en un grafo G es un circuito que pasa por todas las aristas de G , una sola vez por cada una.*

Un camino euleriano en un grafo es un camino que pasa por todas las aristas. Al ser camino el origen y el final no tienen porque coincidir.

Definición 6.51 *Un grafo es euleriano si tiene un circuito euleriano.*

El problema de determinar un circuito euleriano en un grafo tiene una solución completa, basada en los grados de los vértices del grafo. Al recorrer un grafo, cada vez que se llega a un vértice, si no se ha completado el circuito, se vuelve a salir. Es obvio entonces que si un circuito recorre todas las aristas de un grafo, y cada arista se recorre una sola vez, entonces el grado de cada vértice debe ser par. Esta condición también es suficiente como muestra el siguiente teorema.

Teorema 6.52 *Sea G un grafo conexo.*

- *Existe un circuito euleriano en G si y solo si todos sus vértices tienen grado par.*
- *Existe un camino euleriano en G si y solo si todos sus vértices excepto a lo sumo dos de ellos tienen grado par (o hay 0 o hay dos de grado impar).*

Además, en caminos eulerianos si hay vértices de grado impar, el camino empieza en uno de ellos y acaba en el otro.

También existe un resultado análogo para digrafos.

Teorema 6.53 *Sea G un digrafo conexo.*

- *Existe un circuito euleriano en G si y solo si para todos sus vértices x se cumple que $d_-(x) = d_+(x)$.*
- *Existe un camino euleriano en G si y solo para cualquier vértice x excepto a lo sumo dos de ellos se cumple que $d_-(x) = d_+(x)$ y para los dos que no lo cumplen, se tiene que*

$$|d_-(y_0) - d_+(y_0)| = 1, |d_-(y_1) - d_+(y_1)| = 1.$$

La demostración de ambos resultados es constructiva, eligiendo aristas hasta completar el recorrido. Existen diversos algoritmos para tal fin, a continuación se señalan dos de ellos.

Algoritmo 6.54 *Algoritmo de Hierholzer. Solo para circuitos eulerianos.*

Sea G un grafo euleriano.

Paso 1 Se elige un vértice de inicio y fin cualquiera x .

Paso 2 Construimos un circuito simple con salida y llegada en x .

Paso 3 Eliminamos del grafo de partida las aristas elegidas en el circuito anterior y los vértices que queden desconectados.

Paso 4 Mientras nos queden vértices en el grafo, escogemos otro circuito simple con origen un vértice que no sea el inicial pero que si esté en el circuito construido.

a) borramos del grafo las aristas de este nuevo circuito y los vértices desconectados.

b) añadimos al circuito del paso 2, en el lugar adecuado, el nuevo circuito.

Cuando el algoritmo acaba no hay más aristas y tenemos un circuito euleriano.

El siguiente algoritmo construye el camino de una sola vez sin tener que reordenar las aristas como se hacía en el paso 4b. Sin embargo, es más costoso que el anterior. Utiliza el concepto de arista puente que se define de la forma siguiente:

Definición 6.55 *Una arista puente en un grafo es aquella arista a tal que $G - \{a\}$ tiene más componentes conexas que G .*

Algoritmo 6.56 *Algoritmo de Fleury.*

Sea G un grafo euleriano y $x_0 \in V(G)$ un vértice.

Etapa 1 Se elige cualquier arista que incida en x_0 , por ejemplo a_1 y sea x_1 su otro extremo. Sea H_1 el subgrafo de G obtenido al eliminar a_1 y x_0 si este vértice queda aislado.

Etapa k Se supone construido un camino $x_0 a_1 x_1 \dots a_{k-1} x_{k-1}$ y el subgrafo H_{k-1} . Si hay más aristas incidentes en x_{k-1} , se elige una al azar con la condición de que solo se escoge una arista puente si no hay mas opciones. Se construye entonces otro camino de longitud una unidad más, es decir, $x_0 a_1 x_1 \dots a_{k-1} x_{k-1} a_k x_k$.

El algoritmo para cuando se llega a un x_k en el que no hay más aristas. Puesto que el grado de todos es par, $x_0 = x_k$.

La salida del algoritmo es un circuito euleriano. Pero, también sirve para calcular caminos eulerianos.

Grafos hamiltonianos

Otro problema similar al de circuitos eulerianos es el de circuitos hamiltonianos. En este caso, no se trata de recorrer todas las aristas del grafo sino que hay que pasar por todos los vértices una única vez.

Definición 6.57 *Sea G un grafo. Un circuito hamiltoniano en G es un ciclo (camino cerrado simple) que pasa por todos los vértices. Se dice que G es un grafo hamiltoniano si tiene un circuito hamiltoniano.*

Un camino hamiltoniano en un grafo es un camino simple que pasa por todos los vértices.

De la analogía entre las definiciones de grafos eulerianos y grafos hamiltonianos, se puede pensar que entre estos dos conceptos hay una estrecha relación y que se dispone de un criterio sencillo para determinar si un grafo es hamiltoniano. Pero este no es el caso, ya que el problema de encontrar una caracterización de los grafos hamiltonianos es un problema abierto en teoría de grafos. Hay que señalar también que los conceptos de grafo euleriano y grafo hamiltoniano son independientes como se muestra en la figura 6.

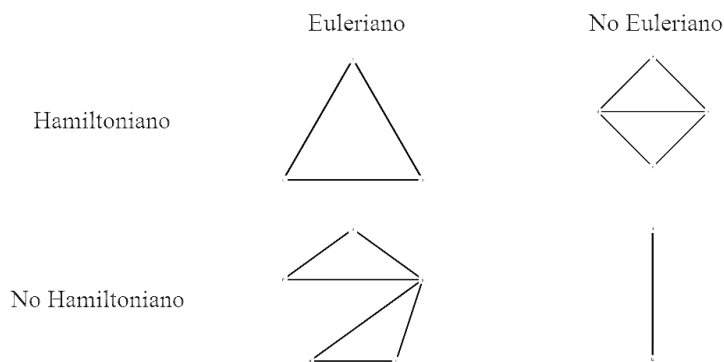


Figura 6: Grafos eulerianos vs. grafos hamiltonianos

A pesar de lo comentado, es posible dar algunas características de los grafos hamiltonianos.

Proposición 6.58 *Sea G un grafo hamiltoniano. Entonces:*

- i.) G no tiene aristas puente, en particular no hay vértices de grado 1.*
- ii.) G no tiene puntos de corte. Es decir para todo vértice $x \in V(G)$ el número de componentes conexas de G es el mismo que el de $G - \{x\}$.*
- iii.) Si x es un vértice de grado 2, las dos aristas de G incidentes con x deben estar en cualquier circuito hamiltoniano.*

Esta proposición nos da condiciones sobre un grafo para que sea hamiltoniano, pero solo condiciones necesarias y no suficientes. No obstante, se conocen algunas condiciones suficientes para que un grafo sea hamiltoniano, pero ninguna de ellas es necesaria.

Teorema 6.59 (Ore) *Sea G un grafo simple con $n \geq 3$ vértices. Si para todo par de vértices $x, y \in V(G)$ no adyacentes se cumple que $d(x) + d(y) \geq n$, entonces G es hamiltoniano.*

Si en el teorema se sustituye n por $n - 1$ tendremos un camino hamiltoniano.

Corolario 6.60 (Dirac) *Sea G un grafo simple con $n \geq 3$ vértices tal que para todo vértice del grafo se cumple que $d(x) \geq n/2$. Entonces G es hamiltoniano.*

Los resultados anteriores (Ore y Dirac) proporcionan condiciones suficientes para que un grafo disponga de un circuito hamiltoniano. No obstante, estas condiciones no son necesarias, como puede verse en el grafo ciclo C_n , $n \geq 5$ que es claramente hamiltoniano pero en cambio no satisface ninguna de las hipótesis.

Proposición 6.61 *Sea G un grafo bipartito con conjunto de vértices $V = V_1 \cup V_2$, es decir las aristas de G unen vértices de V_1 con vértices de V_2 . Si $|V_1| \neq |V_2|$ entonces G no admite un circuito hamiltoniano.*

3.2. Tema 7. Caminos mínimos. Algoritmos de Dijkstra y Floyd-Warshall. Árboles generadores. Búsqueda en anchura y profundidad. Algoritmos de Kruskal y Prim.

Cuando se ha definido anteriormente la noción de recorrido en un grafo, se definía su longitud como el número de aristas de dicho recorrido. Dado un grafo cualquiera G y dos vértices x, y en G , puede ocurrir que existan diversos caminos entre estos vértices. Por tanto, es natural preguntarse que recorrido tiene longitud mínima. No obstante, con las definiciones anteriores, todas las aristas medían o pesaban lo mismo, es decir, aportaban uno a la longitud del recorrido. Cuando se modeliza un problema, las aristas no tienen que ser siempre de la misma longitud, por ejemplo si representan distancias entre ciudades, coste de transporte, etc.

Definición 7.1 *Un grafo pesado o ponderado en aristas es un grafo $G = (V, A, \phi)$ junto con una aplicación $w : A \rightarrow \mathbb{R}_+$. w es la aplicación peso que asigna a cada arista una longitud, peso, tamaño,...*

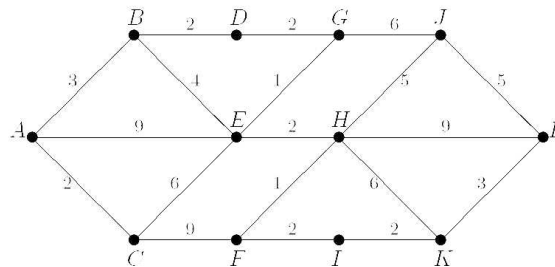


Figura 7: Grafo pesado

Dada una trayectoria en un grafo pesado $x_0 a_1 x_1 \dots a_k x_k$, se define su peso como la suma de todos los pesos de las aristas que están en la trayectoria, $\sum w(a_i)$. Si los pesos se interpretan como tiempos de recorrido, el peso de una trayectoria es lo que se tarda en ir de x_0 a x_k .

Camino mínimo

Definición 7.2 *Dados dos vértices x, y en un grafo pesado, llamaremos distancia mínima entre x e y , $d(x, y)$, al mínimo de los pesos de caminos que unen x con y . Un camino $x_0 a_1 x_1 \dots a_k x_k$ es de peso mínimo, o de distancia mínima, si su peso es $d(x_0, x_k)$.*

Por definición si x, y no están en la misma componente conexa (no hay un camino entre ellos) diremos que están a distancia infinita, $d(x, y) = \infty$. Si G no es simple, para calcular caminos mínimos o distancias en el grafo, no hay problema en suprimir los lazos, y cuando entre dos vértices adyacentes haya varias aristas, eliminamos las que tengan más peso quedándonos únicamente con la de peso mínimo.

Existen diversos algoritmos para tal fin dependiendo de cual sea el objetivo que se desee alcanzar. El siguiente algoritmo determina, fijado un vértice inicial, tanto las distancias mínimas al resto de los vértices como un camino en el grafo que tiene esa distancia.

Algoritmo 7.3 *Algoritmo de Dijkstra.*

Sea G un grafo simple pesado. Sean x_1, x_2, \dots, x_n sus vértices y sea w su función de peso. Definimos $w_{ij} = d(x_i, x_j)$ si x_i y x_j son adyacentes y $w_{ij} = \infty$ en caso contrario. Es claro que w define una matriz simétrica con $w_{ii} = \infty$.

Se fija un vértice, supongamos x_1 . El algoritmo nos va a permitir calcular la distancia del resto de los vértices con x_1 además de dar el camino explícito con esa distancia.

Etapa 1 Se inicializan las variables:

$$\begin{array}{l} P_1 = \{x_1\} \mid T_1 = V - \{x_1\} \\ L(x_1) = 0 \mid \forall x_j \in T_1, L'_1(x_j) = w_{1j} \end{array}$$

c_1 camino trivial: x_1 .

Etapa t Recursivamente.

Sea x_{i_t} el vértice tal que $L'_{t-1}(x_{i_t}) = \min\{L'_{t-1}(x_j) \mid x_j \in T_{t-1}\}$. Es decir, el vértice que en el paso anterior tiene peso mínimo con la función peso L' .

Reasignando variables:

$$\begin{array}{l} P_t = P_{t-1} \cup \{x_{i_t}\} \mid T_t = V - P_t \\ L(x_{i_t}) = L'_{t-1}(x_{i_t}) \mid \forall x_j \in T_t, L'_t(x_j) = \min\{L'_{t-1}(x_j), L(x_{i_t}) + w_{i_t j}\} \end{array}$$

Sea k tal que $L'_k(x_{i_t}) = L(x_{i_t})$ y $L'_{k-1}(x_{i_t}) > L(x_{i_t})$ (puede ser $k = 1$). Entonces $c_t = c_k, x_{i_t}$.

c_t es un camino mínimo de x_1 a x_{i_t} .

El algoritmo finaliza cuando $P_t = V(G)$

Este algoritmo se puede organizar en la práctica mediante una tabla con una columna por cada vértice. Cada fila corresponderá a una etapa de ejecución del algoritmo y representará el camino mínimo hasta el vértice correspondiente x_{i_t}

El algoritmo de Dijkstra puede englobarse dentro de una familia de algoritmos en grafos que se conoce como búsqueda en anchura que se verá más adelante.

Si lo que se desea es obtener únicamente las distancias mínimas entre todos los pares de vértices del grafo, se puede usar el siguiente algoritmo.

Algoritmo 7.4 *Algoritmo de Floyd-Warshall.*

Dado un grafo conexo pesado de n vértices queremos encontrar la distancia entre todos sus vértices.

Etapa 1 Sea $M^0 = (m_{ij})$ su matriz de adyacencia con pesos en aristas. Es decir $m_{ij} = w_{ij}$ el peso de la arista del vértice x_i al vértice x_j y ∞ en caso contrario.

Etapa t Para cada $1 \leq t \leq n$ se define recursivamente la matriz $M^t = (m_{ij}^t)$ donde

$$m_{ij}^t = \min\{m_{ij}^{t-1}, m_{it}^{t-1} + m_{tj}^{t-1}\}$$

Es decir, el mínimo entre lo que había antes y la suma del elemento i de la columna t con el elemento j de la fila t .

Nota: En la construcción de M^t la fila y la columna t no cambian.

Si $i \neq j$ entonces $m_{ij}^n = d(x_i, x_j)$.

En este algoritmo el elemento (i, j) de la matriz M^k indica la longitud del camino mínimo entre los vértices x_i y x_j utilizando como vértices interiores del camino los del conjunto x_1, x_2, \dots, x_k .

La diferencia principal con el algoritmo de Dijkstra es que en este se calculan todas las distancias entre pares de vértices y no se calcula ningún camino mínimo. Tanto el algoritmo de Dijkstra como el de Floyd-Warshall pueden aplicarse a grafos dirigidos.

Árboles

La propiedad de que un grafo sea conexo o no, tiene implicaciones en muchos campos, por ejemplo en telecomunicaciones, redes de transporte, etc. Si se desea diseñar una red que conecta un número de nodos con el mínimo número de enlaces. El grafo que modeliza esta red debe ser conexo, para que toda pareja de nodos pueda comunicarse y además no puede disponer de ciclos ya que en caso contrario se podría eliminar una arista contenida en el ciclo manteniendo aún la comunicación.

A este tipo de grafos se los denomina árboles.

En ciencias de la computación los árboles se utilizan en numerosas situaciones, en la representación de datos jerárquicos, en algoritmos de clasificación, y búsqueda, algoritmos de ordenación, codificación de datos, etc.

Definición 7.5 *Un árbol es un grafo conexo que no contiene ciclos.*

Por analogía, se puede definir un bosque como un conjunto de árboles, es decir, un grafo sin ciclos (cada componente conexa es un árbol).

Ser conexo exige tener bastantes aristas, para así poder conectar todos los vértices, mientras que no tener ciclos supone, en principio, que haya pocas, para que no se formen ciclos.

Caracterizaciones diferentes de árbol vienen dadas por el siguiente resultado que muestra además diversas propiedades de un árbol.

Teorema 7.6 *Sea G un grafo con n vértices, son equivalentes:*

- i.) G es un árbol.*
- ii.) G no tiene ciclos y tiene $n-1$ aristas.*
- iii.) G es conexo y tiene $n-1$ aristas.*
- iv.) G es conexo y toda arista es puente.*
- v.) Dos vértices cualesquiera se pueden unir exactamente por un único camino simple.*
- vi.) G no tiene ciclos y si a G se le añade una arista aparece un circuito.*

Como el número de aristas de un árbol de n vértices es siempre $n - 1$, se tiene que cumplir la igualdad

$$\sum_{x \in V} d(x) = 2|V| - 2,$$

lo que implica que los vértices de un árbol tienen más limitaciones con respecto a su grado. De hecho, se tiene la siguiente propiedad:

Proposición 7.7 *Todo árbol con más de 1 vértice tiene al menos dos vértices de grado 1. De hecho el número de vértices de grado 1 es $2 + \sum_{d(x) \geq 3} (d(x) - 2)$.*

Dado un grafo G , existen diferentes subgrafos de él. Algunos de estos subgrafos son además árboles. Son de especial interés aquellos subgrafos, que además de ser árboles, son árboles generadores del grafo G ; es decir tienen todos los vértices del grafo original y únicamente las aristas necesarias para conectar todos los vértices entre sí.

Los árboles generadores están fuertemente relacionados con la conexión en grafos.

Proposición 7.8 *Un grafo G es conexo si y solo si existe un árbol generador T de G .*

Si se dispone de algoritmos eficaces para hallar árboles generadores de un grafo (o para determinar que no los hay), se tendría una manera de determinar si un grafo es conexo o no. Si un grafo G es conexo y $|A(G)| + 1 = |V(G)| + k$, con $k \geq 0$, entonces eliminando k aristas adecuadas se obtiene un árbol generador. Sin embargo, existen algoritmos más eficaces para construir un árbol generador en un grafo (o para concluir que no los hay). En lugar de ir eliminando aristas, se va haciendo crecer el árbol en pasos sucesivos hasta obtener uno generador.

Estos algoritmos están diseñados, en principio, para buscar y etiquetar los vértices de un grafo, y se pueden adaptar a los problemas que interesen en cada momento; búsqueda de caminos mínimos (como ya se mencionó en el algoritmo de Dijkstra), árboles generadores de peso mínimo, etc. Pero además, sirven para dar una respuesta (computacionalmente sencilla) a diversas cuestiones como son las de determinar si un grafo es conexo (o, en caso contrario, identificar componentes conexas), localizar ciclos, decidir si una arista es puente, etc.

Busqueda en anchura (BFS) y profundidad (DFS).

Sea un grafo que se desea estudiar de alguna forma, pero que no se conoce de manera global sino únicamente local, es decir, se puede saber que pasa alrededor de un vértice dado pero no de todos a la vez. Por ejemplo, supóngase que se desea conocer si el grafo es conexo. La manera más cómoda de conseguirlo con seguridad es construir un árbol generador; de esta forma se habrá recorrido los vértices del grafo que están en la misma componente conexa que el vértice de partida sin usar todas las aristas.

Algoritmo 7.9 *Busqueda en anchura: Breadth first search*

Se dispone de un grafo G y se va a construir un árbol a partir de un vértice inicial x_1 cuyos vértices están agrupados por generaciones en función de su distancia al vértice original.

Etapa 1 $R = \emptyset$. T árbol sin vértices ni aristas.

Etapa 2 $R = \{x_i : x_i \text{ es adyacente a } x_1 \text{ y no está en } R\}$.
Vértices(T)= $\{x_1\} \cup R$, Aristas(T)= $\{x_1x_i\}_{x_i \in R}$.

Etapa 3 Mientras $R \neq \emptyset$, para cada vértice y en R :
 $R2 := \{x : x \text{ es adyacente a } y \text{ y no está en } R\}$, $R = [R \setminus y, R2]$
 $V(T) = V(T) \cup R2$, $A(T) = A(T) \cup \{yx\}_{x \in R2}$

T es un árbol generador de la componente conexa de x_1

La idea básica es la siguiente; fijado un vértice primero se tratan los vértices adyacentes, luego los adyacentes a estos que no se hayan visitado y así sucesivamente hasta que no queden más vértices. Es importante destacar que en ningún momento se exige conocer el número de vértices del grafo original.

Otra forma diferente de recorrer el grafo se basa en la estrategia de recorrer un camino hasta el final. Es decir, a partir de un vértice inicial, ir construyendo un camino en el grafo añadiendo vértices hasta que se llegue a un vértice para el cual todos los adyacentes a este ya estaban visitados. Cuando se llegue a este caso, se regresa por el ultimo camino hasta encontrar un vértice que posea vértices adyacentes no visitados y se repite el proceso.

Algoritmo 7.10 *Busqueda en profundidad: Depth first search*

Sea G un grafo y un vértice inicial x_1

Paso 1 $R = \emptyset$. T árbol sin vértices ni aristas.

Paso 2 Sea y un vértice adyacente a x_1 . $R = \{y\}$
Vértices(T)= $\{x_1, y\}$, Aristas(T)= $\{x_1y\}$.

Paso 3 Mientras $R \neq \emptyset$, sea y el último elemento de R .
Si y tiene un vértice adyacente z que no está en R entonces $R = [R, z]$.
 $V(T) = V(T) \cup \{z\}$, $A(T) = A(T) \cup \{yz\}$.
En caso contrario, $R = R \setminus y$.

T es un árbol generador de la componente conexa de x_1

Ambos algoritmos tienen sus ventajas y sus desventajas, en función del problema que se esté tratando. Ambos permiten resolver de manera sencilla determinar si un cierto grafo G es conexo o no comprobando si el árbol que produce el algoritmo incluye todos los vértices del grafo o no.

Con ligeras variaciones, permiten también resolver las siguientes cuestiones:

- determinar las componentes conexas de un grafo G
- determinar si G tiene o no ciclos (localizando las componentes conexas y contando las aristas de cada una de ellas; esto es, comprobando si son árboles)
- decidir si una cierta arista es puente o no del grafo (comparando el número de componentes conexas antes y después de eliminar la arista en cuestión)
- determinar si un grafo G es bipartito o no (ya que un grafo es bipartito si y solo si no contiene ciclos de longitud impar)

El número de árboles generadores de un grafo viene determinado por el siguiente teorema:

Teorema 7.11 (Kirchhoff) *Sea G un grafo con n vértices. Sea $L = (l_{ij})$ la matriz definida de la forma siguiente:*

$$l_{ij} = \begin{cases} d(x_i), & i=j; \\ -1, & \text{si } x_i x_j \text{ es arista,} \\ 0, & \text{en caso contrario.} \end{cases}$$

Sea L^- la matriz resultante al eliminar la última fila y la última columna de L . El número de árboles generadores de G es $|L^-|$.

Árboles generadores de peso mínimo

Dado un grafo cualquiera el árbol generador no es único; por ejemplo para el grafo completo K_n existen n^{n-2} árboles generadores distintos. En el caso de grafos pesados, es de interés conocer árboles generadores de peso mínimo; es decir, entre todos los árboles generadores posibles, determinar uno tal que la suma de los pesos de todas sus aristas sea la más pequeña posible. Por ejemplo, si se desea conectar un conjunto de ciudades mediante líneas telefónicas de forma que el coste total sea mínimo teniendo en cuenta que conectar cada par de ciudades tiene un coste determinado.

Existen diversos algoritmos que resuelven el problema.

Algoritmo 7.12 *Algoritmo de Kruskal.*

Estrategia: Se eligen aristas de la forma más barata, sin formar ciclos.

Sea G un grafo pesado, $K(G)$ denota el número de componentes conexas de G .

Etapa 0 Sea T_0 un bosque tal que $V(T_0) = V(G)$ y $A(T_0) = \emptyset$.

Etapa 1 Sea e_1 una arista de G (que no sea lazo) de peso mínimo.
 $V(T_1) = V(G)$, $A(T_1) = \{e_1\}$. ($K(T_1) = n - 1$).

Etapa i Sea e_i una arista de G (que no sea lazo) de peso mínimo entre las que conectan componentes conexas de T_{i-1} .
 $V(T_i) = V(G)$, $A(T_i) = A(T_{i-1}) \cup \{e_i\}$. ($K(T_i) = n - i$).

T_{n-1} es un árbol generador de peso mínimo.

Algoritmo 7.13 *Algoritmo de Prim.*

Estrategia: Se parte de un vértice y se van alcanzando los demás, de uno en uno, del modo más económico posible.

Sea G un grafo pesado, y x_0 un vértice cualquiera.

Etapa 0 Sea T_0 un bosque tal que $V(T_0) = \{x_0\}$ y $A(T_0) = \emptyset$.

Etapa 1 Sea e_1 una arista de G (que no sea lazo) incidente con x_0 de peso mínimo entre las incidentes con x_0 y de extremo x_1 .
 $V(T_1) = \{x_0, x_1\}$, $A(T_1) = \{e_1\}$.

Etapa i Se elige una arista e_i de peso mínimo entre todas las que conectan alguno de los vértices $\{x_0, x_1, \dots, x_{i-1}\}$ de T_{i-1} con alguno de los otros.
Sea x_i su otro extremo.
 $V(T_i) = \{x_0, x_1, \dots, x_i, x_i\}$, $A(T_i) = A(T_{i-1}) \cup \{e_i\}$.

Teorema 7.14 T_{n-1} es un árbol generador de peso mínimo.

El árbol generador de peso mínimo no es único, pero si la sucesión de pesos de sus aristas. Cualquier árbol generador se puede obtener como aplicación de los algoritmos de Kruskal o de Prim. Mientras el algoritmo de Kruskal se centra en el peso mínimo, el algoritmo de Prim se centra en la conexión.

Arboles con raíz, árboles m-arios

En algunas aplicaciones de los árboles (problemas de decisión, torneos, ordenación, etc) es necesario distinguir uno de los vértices del árbol.

Definición 7.15 *Un árbol con raíz es un árbol en el que hemos señalado un vértice que se denomina raíz y el resto crece a partir de él.*

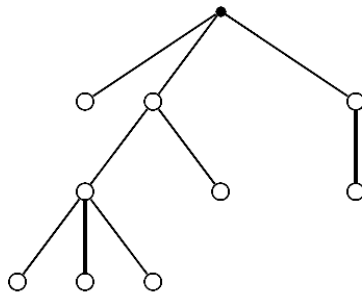


Figura 8: Arbol con raíz

En árboles dirigidos también tiene sentido definir árboles con raíz pero teniendo en cuenta las direcciones, tiene que haber un camino dirigido desde la raíz hasta los demás vértices. De hecho, un árbol con raíz se puede entender como un árbol dirigido tomando en cada arista la dirección correcta, la arista (u, v) tiene el sentido $u \rightarrow v$ si u está más cerca de la raíz que v .

Definición 7.16 *Sea T un árbol con raíz x_0 . Para cada vértice x_k existe un camino único $x_0 - x_1 - x_2 - \dots - x_k$. Se dice que:*

- x_{i-1} es el padre de x_i y x_j es hijo de x_{j-1} .
- Un vértice de grado uno se denomina hoja.
- Los vértices que no son hojas se denominan internos.
- El nivel de un vértice es la longitud del camino único a la raíz.
- La altura $h(T)$ de un árbol con raíz es el más grande de los niveles.
- Si el nivel de cada hoja es $h(T)$ o $h(T) - 1$ se dice que es equilibrado.
- El árbol se dice m -ario ($m \geq 2$) si el grado de salida hacia los descendientes es más pequeño o igual a m . Se dice completo si $d_+(x) = m$ ó $d_+(x) = 0$.

A partir de estas definiciones es posible obtener relaciones entre el número de vértices del árbol y el número de vértices internos o entre la altura de un árbol m -ario completo y su número de vértices (u hojas) como se muestra a continuación.

Teorema 7.17 *Sea T un árbol completo m -ario.*

- *Si T tiene i vértices internos entonces el número de vértices totales es $n = mi + 1$ y el número de hojas es $f = (m - 1)i + 1$.*
- *Si T tiene n vértices entonces el número de hojas es $\frac{n(m - 1) + 1}{m}$.*

Corolario 7.18 *Si T es un árbol m -ario completo y con f hojas entonces T tiene $\frac{f - 1}{m - 1}$ vértices internos y $\frac{mf - 1}{m - 1}$ vértices.*

Teorema 7.19 *Si T es un árbol m -ario con n vértices y altura h entonces*

$$h \geq \log_m \left(\frac{n(m - 1) + 1}{m} \right).$$

En general si T es un árbol m -ario con f hojas entonces $h \geq \lceil \log_m(f) \rceil$ y se da la igualdad si además es completo y equilibrado.

Como aplicación de este resultado se puede deducir el número mínimo de comparaciones que se deben hacer, en el peor de los casos, para ordenar una lista de n elementos. Representando las $n!$ posibles ordenaciones como las hojas de un árbol binario completo y equilibrado, el número mínimo de comparaciones corresponde a la altura del árbol y por tanto es $\lceil \log_2(\frac{n!+1}{2}) \rceil$ (del orden de $n \log_2(n)$).

Otra aplicación de los árboles m -arios son los problemas de decisión. El siguiente ejemplo muestra como aplicarlo.

Ejemplo 7.20 *Se dispone de 8 monedas idénticas de las que se sabe que una de ellas es falsa y pesa un poco más que las demás. Usando una balanza es posible determinar la moneda falsa únicamente en dos pesadas.*

Como se disponen de 8 monedas éstas serán las hojas del árbol. Una pesada en la balanza puede dar tres resultados; ambos platillos pesan igual o alguno de los dos es más pesado. Entonces se tiene un árbol ternario con 8 hojas. La altura del árbol es $h \geq \lceil \log_3(8) \rceil = 2$. Si fuera completo y equilibrado se tendría la igualdad, pero no lo es, pues en ese caso el árbol tendría

$\frac{3 * 8 - 1}{2} = 12,5$ vértices. Pero la balanza siempre nos va a dar uno de los tres resultados posibles (aunque alguno sea incompatible), por lo que podemos suponer que el árbol es completo y que el árbol tiene 4 vértices internos y 13 vértices en total. De ello se puede deducir que la altura es 2. En la figura 9 se puede ver cómo realizarlo donde $\{a, b, \dots, g, h\}$ representan las diferentes monedas.

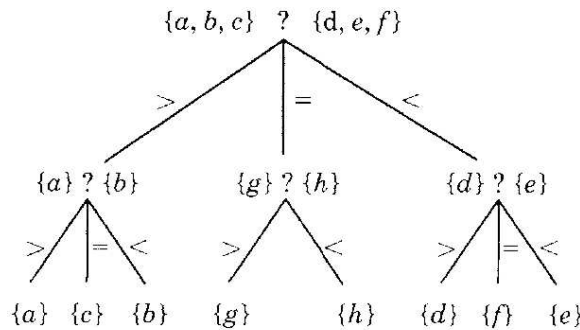


Figura 9: Determinación de la moneda falsa

3.3. Tema 8. Grafos bipartitos. Emparejamientos y emparejamientos maximales. Teorema de Hall.

Grafos bipartitos.

Una relación entre dos conjuntos X e Y es un subconjunto R de $X \times Y$. Por ejemplo

- X alumnos del grado de Matemáticas,
- Y asignaturas del grado de Matemáticas.

$(x, y) \in R \subset X \times Y$ si y solo si el alumno x está matriculado en la asignatura y . Dada una relación entre dos conjuntos, se puede definir un grafo bipartito G cuyos vértices son los elementos de $X \cup Y$ y las aristas conectan elementos relacionados.

En todo grafo bipartito $G = (V_1 \cup V_2, A)$ se tiene que

$$\sum_{x \in V_1} d(x) = \sum_{x \in V_2} d(x) = |A|.$$

Los grafos bipartitos modelizan por tanto relaciones entre dos conjuntos.

Ejemplo 8.1 *Problema del matrimonio.* V_1 y V_2 representan un conjunto de chicos y chicas respectivamente. Las aristas vienen dadas por la relación "querer casarse con".

Ejemplo 8.2 *Problema del alojamiento.* V_1 y V_2 representan el mismo conjunto de personas. Se dispone de un número determinado de habitaciones dobles. Sin embargo algunas parejas son compatibles para dormir en la misma habitación y otras no. Las aristas representan las compatibilidades.

Ejemplo 8.3 *Problema de asignación de tareas.* V_1 representa un conjunto de personas y V_2 un conjunto de tareas a realizar. Las aristas vienen dadas por la capacidad de las personas para realizar dicha tarea.

En los ejemplos anteriores se ve claramente cuál es el propósito que se desea. Se trata de conseguir que se case el mayor número posible de parejas, alojar a la mayor cantidad de personas posible o realizar la mayor parte de los trabajos.

Emparejamientos y emparejamientos maximales.

Definición 8.4 *Un emparejamiento en un grafo G es un conjunto de aristas H de G sin vértices comunes. Es decir, un subgrafo donde todos los vértices tienen grado menor o igual que 1.*

Si $G = (V_1 \cup V_2, E)$ es un grafo bipartito, un emparejamiento es una correspondencia inyectiva entre vértices de V_1 y vértices de V_2 . Gráficamente, en la figura 10 se muestra un emparejamiento en el grafo bipartito donde las aristas más gruesas son precisamente el emparejamiento.

Aunque un emparejamiento se puede dar en cualquier tipo de grafos, el estudio que se va a realizar solo engloba a los grafos bipartitos. Sin pérdida de generalidad se puede restringir al caso de grafos bipartitos $G = (V_1 \cup V_2, A)$ con $|V_1| \leq |V_2|$ ya que en caso contrario existirán vértices de V_1 no emparejados.

Se pueden distinguir varios tipos de emparejamientos:

- máximo: si no hay emparejamientos con más aristas.
- maximal: si no está contenido en otro emparejamiento.
- completo: si todos los vértices de V_1 tienen pareja.

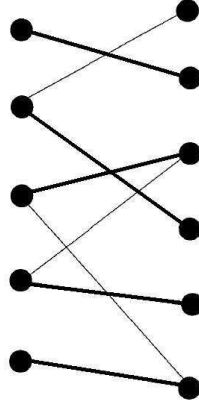


Figura 10: Emparejamiento en un grafo bipartito

- perfecto: si todos los vértices tanto de V_1 como de V_2 están emparejados.

Un emparejamiento máximo es siempre maximal pero el recíproco no es cierto.

Es claro que en el grafo bipartito completo $K_{n,m}$ ($n \leq m$) existe un emparejamiento completo que además es perfecto si $n = m$. Lógicamente si $|V_1| \neq |V_2|$ no existirán emparejamientos perfectos. Es posible conocer el número de emparejamientos perfectos o completos de un grafo bipartito sin tener que calcularlos.

Definición 8.5 Sea $G = (V_1 \cup V_2, A)$ un grafo bipartito, $V_1 = \{x_1, x_2, \dots, x_n\}$, $V_2 = \{y_1, y_2, \dots, y_m\}$. Se define la matriz de adyacencia bipartita como la matriz cuadrada de orden n cuya entrada (i, j) es 1 si $x_i y_j$ es una arista de G y 0 en caso contrario.

Teorema 8.6 Sea $G = (V_1 \cup V_2, A)$ un grafo bipartito con $|V_1| = |V_2|$. El número de emparejamientos perfectos de G es la permanente $(P(G))$ de su matriz de adyacencia bipartita.

Si en cambio $|V_1| < |V_2|$, un emparejamiento completo puede obtenerse a partir de un emparejamiento perfecto de otro grafo de la siguiente forma:

- Se construye el grafo $\tilde{G} = (\tilde{V}_1 \cup V_2, A \cup \tilde{A})$ con

$$\begin{aligned} \tilde{V}_1 &= V_1 \cup \{|V_2| - |V_1| \text{ vértices más}\} \\ \tilde{A} &= \{ \text{áristas de los vértices de } \tilde{V}_1 \setminus V_1 \text{ a cada vértice de } V_2 \} \end{aligned}$$

- Si \tilde{G} tiene un emparejamiento perfecto

$$(x_1, y_1), (x_2, y_2), \dots, (x_{|V_1|}, y_{|V_1|}), (\tilde{x}_1, \tilde{y}_1), \dots, (\tilde{x}_s, \tilde{y}_s)$$

Entonces

$$(x_1, y_1), (x_2, y_2), \dots, (x_{|V_1|}, y_{|V_1|})$$

es un emparejamiento completo de G .

Recíprocamente, a partir de un emparejamiento completo de G se puede obtener un emparejamiento perfecto de \tilde{G} . No obstante dos emparejamientos perfectos distintos de \tilde{G} pueden dar lugar a emparejamientos iguales de G .

Teorema 8.7 *Sea $G = (V_1 \cup V_2, E)$ un grafo bipartito con $|V_1| < |V_2|$. El número de emparejamientos completos de G es $P(\tilde{G})/(|V_2| - |V_1|)!$, donde \tilde{G} es el grafo bipartito definido como antes.*

Teorema de Hall.

Es fácil ver que para que exista un emparejamiento completo en un grafo bipartito $G = (V_1 \cup V_2, A)$, es necesario que cada par de vértices de V_1 sea adyacente a al menos dos vértices de V_2 . Si existiera un par $x_1, x_2 \in V_1$ adyacentes a un único vértice $y_1 \in V_2$, en cualquier emparejamiento al menos uno de los dos no tendría pareja. Esta condición debe darse para cualquier subconjunto de vértices de V_1 .

Definición 8.8 *Sea G un grafo y $X \subseteq V(G)$ un conjunto de vértices de G . Se define el conjunto de vecinos de X , como $n(X) = \{v \in V(G) \setminus X \mid v \text{ es adyacente a algún vértice de } X\}$.*

Teorema 8.9 (Teorema de Hall) *Sea $G = (V_1 \cup V_2, A)$ un grafo bipartito con $|V_1| \leq |V_2|$. G posee algún emparejamiento completo si y solo si para todo $X \subseteq V_1$ se cumple que $|n(X)| \geq |X|$.*

Sin embargo, no todo grafo bipartito posee un emparejamiento completo.

Definición 8.10 *Se llama deficiencia de un grafo G al entero*

$$\delta(G) = \max\{|X| - |n(X)|, X \subseteq V_1\}$$

El siguiente resultado determina el tamaño máximo de un emparejamiento.

Corolario 8.11 *Sea $G = (V_1 \cup V_2, A)$ un grafo bipartito con $|V_1| \leq |V_2|$. El tamaño máximo de un emparejamiento es $|V_1| - \delta(G)$.*

Construir un emparejamiento máximo de un grafo bipartito puede realizarse de forma recursiva. A partir de cualquier emparejamiento no máximo y utilizando un camino aumentador, se construye otro emparejamiento con más aristas que el de partida.

Definición 8.12 Sea $G = (V_1 \cup V_2, A)$ un grafo bipartito y M un emparejamiento (un conjunto de aristas de G , no incidentes 2 a 2).

- Un vértice x se dice libre respecto al emparejamiento M si ninguna arista de M es incidente con x .
- Se denomina camino aumentador para M a un camino en el grafo $x_0y_1x_1y_2 \dots y_{k-1}x_{k-1}y_k$ en el que x_0, y_k son vértices libres y x_jy_j son aristas de M para $j = 1 \dots k - 1$.

Un camino aumentador también se puede denominar alternado pues las aristas que forman el camino pertenecen alternativamente a M y $A \setminus M$.

A partir de la definición de camino aumentador se puede deducir que:

- Un camino aumentador para un emparejamiento M tiene siempre un número impar de aristas.
- El camino aumentador más elemental posible está formado por una arista entre dos vértices libres.
- Si M es un emparejamiento de un grafo G y $x_0y_1x_1y_2 \dots y_{k-1}x_{k-1}y_k$ es un camino aumentador para M , entonces

$$M' = M \setminus \{y_1x_1, y_2x_2, \dots, y_{k-1}x_{k-1}\} \cup \{x_0y_1, x_1y_2, \dots, x_{k-2}y_{k-1}, x_{k-1}y_k\}$$

es un emparejamiento con una arista más que M .

Utilizando las ideas expuestas anteriormente, se puede dar un algoritmo para determinar emparejamientos máximos.

Algoritmo 8.13 Algoritmo para encontrar emparejamientos máximos.

- i.) Se parte de un emparejamiento M_0 (por ejemplo el emparejamiento vacío).
- ii.) Se busca un camino aumentador.
- iii.) Se cambia M_0 por el emparejamiento obtenido en el paso anterior que tiene una pareja más.

Sin embargo, el algoritmo anterior debe su eficiencia a la búsqueda de caminos aumentadores para cada emparejamiento. El siguiente resultado asegura que siempre existe.

Proposición 8.14 *Si M es un emparejamiento que no es máximo, entonces tiene un camino aumentador.*

Para determinar un camino aumentador para un emparejamiento en un grafo se puede usar el siguiente algoritmo.

Algoritmo 8.15 *Algoritmo de etiquetado. Búsqueda de caminos aumentadores.*

Sea $G = (V_1 \cup V_2, E)$ un grafo bipartito y un emparejamiento M .

- i.) Sea $v_1 \in V_1$ un vértice libre para M y se etiqueta como $[-]$. Si no existe ningún vértice libre el emparejamiento es máximo.*
- ii.) Sea $w \in V_2$ un vértice no etiquetado tal que es adyacente a un vértice $v \in V_1$ ya etiquetado y vw no es una arista de M . Se etiqueta w como $[v]$.*
- iii.) Sea $v \in V_1$ un vértice no etiquetado y adyacente a un vértice $w \in V_2$ ya etiquetado tal que vw es una arista de M . Se etiqueta v como $[w]$.*
- iv.) Se repiten los pasos 2 y 3 hasta que*
 - Se etiqueta un vértice $w \in V_2$ libre. En consecuencia se obtiene un camino aumentador.*
 - No se puede etiquetar más y no habrá caminos aumentadores que comiencen en v_1 .*

3.4. Tema 9. Grafos dirigidos. Redes y Flujos. Algoritmo de Ford-Fulkerson.

Grafos dirigidos.

Anteriormente, se definió un grafo dirigido como un grafo cuyas aristas poseen una dirección. Formalmente, un digrafo también representa una relación R entre elementos del mismo conjunto (tal como sucedía con los grafos bipartitos). En lugar de decir que x está relacionado con y mediante la relación R , se dice ahora que el par (x, y) es un arco del digrafo cuyo conjunto

de arcos es R . Si la relación es simétrica, el correspondiente digrafo cumple que sus arcos (excepto los lazos) ocurren en parejas, es decir, (x, y) e (y, x) son arcos, o ninguno de los dos lo es.

La noción de recorrido se extiende de forma natural en un digrafo. Un recorrido dirigido es una sucesión de vértices, $x_0x_1x_2 \dots x_k$ tal que x_ix_{i+1} es un arco del digrafo.

En algunas situaciones es más apropiado el uso de digrafos que el de grafos, como por ejemplo, el plano de una ciudad con calles de dirección única; un grafo que represente las distintas fases de una evolución, un torneo entre varios participantes donde cada competidor se enfrenta al resto y no se admiten empates, etc. Un árbol con raíz o un grafo bipartito pueden verse también como digrafos.

Redes y flujos.

De igual forma que a los grafos, se puede dotar a las aristas de un grafo un peso que represente un coste o distancia.

Definición 9.1 Una red es un par $R = (G, \psi)$ donde G es un grafo simple dirigido y una aplicación $\psi : A(G) \rightarrow \mathbb{R}_+$. Para cada $a \in A(G)$, $\psi(a)$ se llama la capacidad de a . Si $x \in V(G)$, sus capacidades de entrada y salida son respectivamente:

$$\psi_-(x) = \sum_{a \text{ termina en } x} \psi(a), \quad \psi_+(x) = \sum_{a \text{ empieza en } x} \psi(a)$$

En definitiva una red es un grafo dirigido pesado.

En una red existen vértices especiales dependiendo de su grado de entrada o salida.

Definición 9.2 Un vértice x de una red se denomina fuente si no es final de ninguna arista, $\psi_-(x) = 0$. Se dice que es sumidero si no es el comienzo de ninguna arista, $\psi_+(x) = 0$.

Uno de los aspectos que puede tratarse con las redes es calcular caminos mínimos al igual que sucedía en grafos pesados y los algoritmos que se usan a tal fin son similares a los estudiados. Otro ejemplo de redes surge al representar un trabajo dividido en actividades que requieren un tiempo necesario para su realización y que además deban hacerse en un orden establecido. Utilizando una variante de la búsqueda en anchura puede calcularse el tiempo mínimo de ejecución del trabajo.

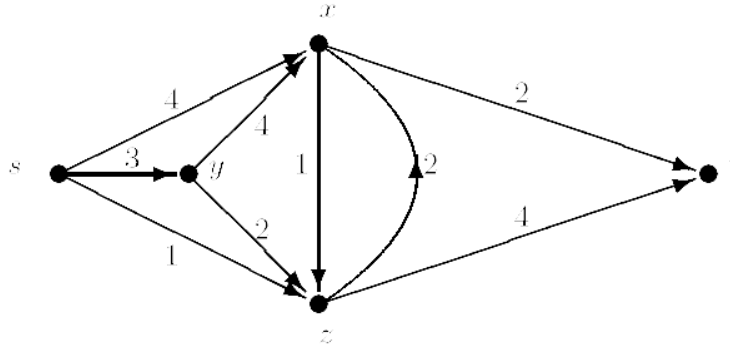


Figura 11: Grafo de una Red

Otra de las situaciones que se pueden modelar con estos digrafos son las redes de transporte. Cada vértice representa una estación, almacén, etc, y los arcos entre vértices tienen como pesos la capacidad que se puede transportar de un vértice a otro. Es posible determinar la cantidad máxima de mercancía que puede transportarse mediante la noción de flujo.

Definición 9.3 Dada una red $R = (G, \psi)$ un flujo en la red es una aplicación $\phi : A(G) \rightarrow \mathbb{R}_+$ tal que

- $\phi(a) \leq \psi(a)$ para toda arista a ,
- $\phi_-(x) = \phi_+(x)$ para todo vértice que no sea fuente o sumidero.

El problema general se reduce fácilmente al caso en que la red tiene y una única fuente y un único sumidero. Como no se permite acumular ninguna mercancía en los vértices intermedios, naturalmente el valor de un flujo en la fuente (f) debe coincidir con el valor del flujo en el sumidero (s), es decir, $\phi_+(f) = \phi_-(s)$. A esta cantidad se la denomina valor del flujo.

Dada una red $R = (G, \psi)$ y un flujo ϕ en ella, una cota máxima para el valor del flujo es $\min\{\psi_+(f), \psi_-(s)\}$ pues lógicamente no se puede transportar más de lo que sale de la fuente ni de lo máximo que puede llegar al sumidero. Esta idea se puede generalizar para un conjunto de aristas de la red.

Definición 9.4 Dada una red $R = (G, \psi)$, se denomina corte en R a todo conjunto de aristas A de G tal que todo camino (no necesariamente dirigido) de la fuente al sumidero pasa por una arista de A .

Todo corte separa el grafo en dos partes:

$$S = \{x \in V(G); x \text{ se puede unir con } f \text{ sin pasar por } A\}$$

$$T = \{x \in V(G); x \text{ NO se puede unir con } f \text{ sin pasar por } A\}$$

La capacidad de un corte es la suma de las capacidades de las aristas que salen de vértices de S y llegan a vértices de T .

Lema 9.5 Sea ϕ un flujo en una red y C un corte que separa los vértices de G en dos partes S y T . Entonces

$$\phi_+(f) = \sum_{x \in S, y \in T} \phi(x, y) - \sum_{x \in S, y \in T} \phi(y, x)$$

Como consecuencia, si C es un corte y ϕ un flujo se tiene que

$$\phi_+(f) \leq \psi(C).$$

Es fácil comprobar que en estas condiciones siempre existe un flujo de valor máximo. De hecho, el valor de este flujo máximo coincide con la capacidad de un corte (pero no de cualquier corte).

Teorema 9.6 (Max-flow min-cut) El valor máximo de un flujo coincide con el mínimo de las capacidades de todos los cortes.

Algoritmo de Ford-Fulkerson.

Determinar un flujo máximo para una red se puede realizar usando una estrategia similar a la propuesta para emparejamientos. A partir de un flujo inicial, se construye un nuevo flujo cuyo valor es mayor.

Definición 9.7 Sea $R = (G, \psi)$ una red y ϕ un flujo. Un camino aumentador para ϕ es un camino simple no necesariamente dirigido de f a s que cumple que:

- Si a es una arista del camino aumentador recorrida en el sentido correcto entonces $\phi(a) < \psi(a)$.
- Si a es una arista del camino aumentador recorrida en el sentido contrario entonces $\phi(a) > 0$.

Dado un camino aumentador, se cambia el flujo ϕ por otro flujo $\tilde{\phi}$ con más valor donde

- Si $a \in A(G)$ no aparece en el camino $\phi(a) = \tilde{\phi}(a)$.
- Si $a \in A(G)$ está en el camino:
 - $\tilde{\phi}(a) = \phi(a) + \epsilon$ si a se recorre en sentido correcto en el camino.
 - $\tilde{\phi}(a) = \phi(a) - \epsilon$ si a se recorre en sentido contrario en el camino.
$$\epsilon = \min\{\{\psi(a) - \phi(a), \text{ a en sentido correcto}\}, \{\phi(a), \text{ a en sentido opuesto}\}\}.$$

Algoritmo 9.8 *Proceso de búsqueda de flujos máximos*

- i.) Se parte de un flujo F_0 (por ejemplo el flujo nulo $\phi(a) = 0$).
- ii.) Se busca un camino aumentador.
- iii.) Se cambia F_0 por el flujo obtenido al usar el camino aumentador.

Caminos aumentadores para un flujo no máximo siempre existen.

Teorema 9.9 *Sea $R = (G, \psi)$ una red y ϕ un flujo. ϕ tiene el valor máximo si y solo si no acepta caminos aumentadores.*

El proceso de encontrar un camino aumentador para un flujo dado puede realizarse mediante el siguiente algoritmo.

Algoritmo 9.10 *Algoritmo de Ford-Fulkerson. Búsqueda de caminos aumentadores.*

Sea $N = (G, \psi)$ una red y ϕ un flujo. Se van a etiquetar los vértices de G en cualquier orden siguiendo unas reglas. La etiqueta de un vértice w es de la forma $[v, c]$ donde v representa el vértice que precede a w en el camino aumentador y $c \in \mathbb{R}_+$ la cantidad en que se puede aumentar (temporalmente) el flujo.

- i.) Se etiqueta f como $[-, \infty]$
- ii.) Sea v etiquetado como $[*, c]$. Si a es una arista que empieza en v y acaba en un vértice no etiquetado w y además $\phi(a) < \psi(a)$. Se etiqueta w como $[v, \min(c, \psi(a) - \phi(a))]$.
- iii.) Sea v etiquetado como $[*, c]$. Si a es una arista que empieza en un vértice no etiquetado w y acaba en v y además $\psi(a) > 0$. Se etiqueta w como $[v, \min(c, \phi(a))]$.

Se continua hasta que no se puede etiquetar más o se etiqueta el sumidero.

Si el algoritmo etiqueta el sumidero s entonces se ha encontrado un camino aumentador y el flujo se puede aumentar. Además, si existe un camino aumentador, el algoritmo lo encuentra.

Este algoritmo puede usarse igualmente para encontrar emparejamientos en grafos bipartitos $G = (V_1 \cup V_2, A)$ convirtiéndolos previamente en una red donde todas las aristas tienen el mismo peso. Es necesario añadir dos vértices que hagan las veces de fuente y sumidero uniendo la fuente con todos los vértices de V_1 y el sumidero con los vértices de V_2 .

Se tratarán también otro tipo de problemas en redes como pueden ser aquellas con vértices limitados en capacidad. El algoritmo para el cálculo de flujos máximos también puede usarse cuando alguno de los vértices interiores tiene limitada la capacidad. Para ello basta con duplicar dicho vértice de forma que en el primero solo llegue flujo y el segundo sea el que lo distribuye añadiendo además una arista entre ambos que es la capacidad del vértice.

3.5. Tema 10. Grafos planos y aplanables. Formula de Euler. Teorema de Kuratowski. Coloreado de grafos. Polinomio cromatico y coloraciones.

Grafos planos y aplanables.

Anteriormente se ha definido la representación gráfica de un grafo como una forma de determinar un grafo. No obstante, fijado un grafo, su representación no tiene porque ser única. Dependerá del espacio donde se quiera dibujar y de la forma del dibujo. En algunas situaciones modelizadas por grafos puede ser necesario exigir además ciertas condiciones al diseño. Por ejemplo: se dispone de tres casas y tres pozos y se desea construir un camino de cada casa a cada pozo. ¿Es posible realizarlo de forma que los caminos no se crucen? Para resolver este tipo de problemas se introduce la noción de planaridad.

Definición 10.1 *Se dice que un grafo G es aplanable si y solo si existe una inmersión en \mathbb{R}^2 . Una representación de un grafo es una inmersión si dos aristas se cortan únicamente en los vértices.*

Definición 10.2 *Se denomina grafo plano G a una representación de un grafo G en \mathbb{R}^2 . Es decir, un conjunto de puntos de \mathbb{R}^2 que representan los vértices junto con un conjunto de curvas (aristas) que unen algunos de los vértices, de forma que dos aristas se cortan a lo sumo en extremos comunes.*

No hay que confundir grafo aplanable con grafo plano. Mientras el primero se refiere a la estructura del grafo con vértices y aristas, la segunda es esencialmente una representación del grafo. Un grafo aplanable puede representarse de este modo por varios grafos planos.

Definición 10.3 Sea G un grafo plano, se llama soporte de G , $sop(G)$ al conjunto de puntos de \mathbb{R}^2 que son vértices de G o pertenecen a aristas de G . Se llaman caras del grafo G a las distintas componentes conexas de $\mathbb{R}^2 - sop(G)$.

- i.) En todo grafo plano existe una cara que se denomina exterior que es no acotada.
- ii.) La frontera (topológica) de cada cara está formada por un cierto número de aristas (y vértices).
- iii.) La frontera exterior de cada cara acotada es un ciclo. En general la frontera de una cara es un ciclo junto con alguna arista más.

Fórmula de Euler.

El número de caras de un grafo plano es independiente de la representación elegida.

Teorema 10.4 (Fórmula de Euler) Sea G un grafo plano y conexo con V vértices, A aristas y C caras, entonces:

$$V - A + C = 2$$

Corolario 10.5 Si G es un grafo plano con k componentes conexas, se tiene que $V - A + C = k + 1$.

Teorema de Kuratowski.

No todos los grafos son aplanables. Una condición necesaria viene dada por el siguiente resultado.

Teorema 10.6 Si G es un grafo simple y aplanable con más de dos vértices se tiene que $A \leq 3V - 6$.

Ejemplos de grafos no aplanables se deducen del resultado anterior.

Teorema 10.7 K_5 y $K_{3,3}$ no son aplanables.

Corolario 10.8 *Si G es un grafo que contiene como subgrafo a K_5 o $K_{3,3}$, entonces no es aplanable.*

Estos dos grafos son bastante importantes en la caracterización de la planaridad de un grafo. De hecho, caracterizan todos los grafos aplanables.

Definición 10.9 *Sea G un grafo, un grafo H es subdivisión de G si se puede obtener a partir de G introduciendo vértices de grado 2 en las aristas de G .*

Teorema 10.10 (Kuratowski) *Un grafo G es aplanable si y solo si no contiene como subgrafo ninguna subdivisión de K_5 o $K_{3,3}$.*

Otra condición necesaria para que un grafo sea aplanable viene dada por el grado de alguno de sus vértices.

Proposición 10.11 *Todo grafo simple aplanable contiene al menos un vértice de grado menor o igual que 5.*

Existen otras caracterizaciones de la planaridad de un grafo.

Definición 10.12 *Sea G un grafo y a una arista de él incidente a los vértices x e y . Se denomina contracción de G por a , que se denota por G_a , al grafo que resulta al eliminar la arista a e identificar los vértices x e y .*

Teorema 10.13 (Wagner) *Un grafo G es aplanable si no contiene subgrafos contractibles a K_5 ni a $K_{3,3}$.*

Sólidos platónicos

Si todas las caras de un grafo plano tienen como frontera ciclos de longitud constante, entonces éstos grafos corresponden a las representaciones de los sólidos platónicos. En la figura 12 se muestran las representaciones en grafos de los sólidos platónicos

Coloreado de Grafos

Uno de los problemas clásicos en teoría de grafos es el de coloreado de mapas. Es decir, determinar el mínimo número de colores que se necesitan para colorear un mapa de forma que dos países con frontera común no tengan el mismo color. Este problema engloba además otras situaciones como pueden ser, por ejemplo, el almacenamiento de productos peligrosos (que son incompatibles entre si), horario de clases (con alumnos comunes), etc. En todos estos problemas se trata de asignar colores a los vértices de un grafo de forma que dos vértices unidos por una arista tengan color diferente. Lógicamente, se supone que los grafos son simples.

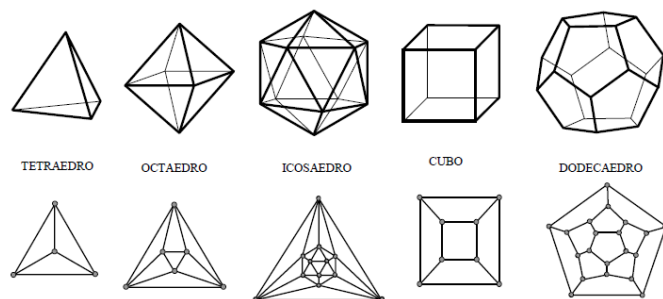


Figura 12: Grafos planos de los sólidos platónicos

Definición 10.14 *Se dice que un grafo G es k -coloreable ($k \geq 1$) si se puede asignar a cada vértice un color de entre k de forma que dos vértices adyacentes tengan colores distintos.*

De la definición anterior se deduce inmediatamente que:

- Si G tiene n vértices entonces G es n -coloreable.
- Si G es k -coloreable, entonces es k' -coloreable para todo $k' \geq k$.

Por tanto existe un número k tal que G es k -coloreable pero no $(k - 1)$ -coloreable. A este número se le denomina número cromático del grafo y se le denota por $\chi(G) = k$.

El número cromático de algunos casos particulares de grafos puede calcularse directamente a partir de la definición de coloreado de grafos utilizando un algoritmo voraz (greedy). El método consiste en asignar colores a los vértices en orden, tal que cada vértice recibe el primer color disponible que no ha sido asignado a ninguno de sus vértices adyacentes. En cada paso se realiza la mejor elección pero sin tener en cuenta lo que ocurra después. Sin embargo, no siempre determina el mínimo número de colores, es decir, el número cromático, en general será mayor.

Las siguientes propiedades y ejemplos pueden probarse fácilmente:

- i.) $\chi(G) = 1$ si y solo si no hay aristas en G .
- ii.) $\chi(G) = 2$ si y solo si G es bipartito (con aristas).
- iii.) $\chi(K_n) = n$.
- iv.) $\chi(C_n) = 2$ si n es par y $\chi(C_n) = 3$ si es impar.

Usando una estrategia voraz, es posible determinar una cota para el número cromático.

Teorema 10.15 *Sea G un grafo y sea ρ tal que para todo vértice x de G , $d(x) \leq \rho$. Entonces:*

- $\chi(G) \leq (\rho + 1)$.
- Si G es conexo y no regular, $\chi(G) \leq \rho$.

La condición de no regularidad del teorema anterior es necesaria pues el grafo completo K_n tiene número cromático n y cada vértice es adyacente a $n - 1$ vértices. Aplicando el teorema anterior a los grafos aplanables es fácil demostrar el teorema de los seis colores que afirma que todo grafo aplanable entonces tiene número cromático menor o igual que 6.

El problema de coloreado de grafos tiene su origen a mediados del siglo XIX. En 1852 Guthrie se interesa por demostrar matemáticamente que son suficientes cuatro colores para colorear cualquier mapa de forma que países con frontera común tengan asignados colores distintos. A partir de un mapa es posible construir un grafo asignando a cada país un vértice y las aristas coinciden con las fronteras. En 1879 Alfred Kempe publicó un artículo en el que afirmaba haber demostrado la conjetura de los cuatro colores. Aunque la demostración de Kempe resultó ser incompleta ésta contenía la mayor parte de las ideas fundamentales, reduciendo el problema a considerar un conjunto de configuraciones. Finalmente, Kenneth Appel y Wolfgang Haken en 1976, con la ayuda del ordenador, comprobaron que la conjetura de los cuatro colores es cierta.

Teorema 10.16 (de los 4 colores) *(Appel-Haken 1976) Un grafo aplanable se puede colorear con 4 colores.*

Utilizando técnicas más básicas es posible demostrar el teorema de los 5-colores; que afirma que todo grafo plano se puede colorear con 5 colores.

Polinomio cromático y coloraciones.

No solo interesa saber si se puede colorear un grafo con k colores, sino también de cuántas formas se puede colorear. El primer problema queda resuelto en cuanto se conoce el número cromático, $\chi \leq k$ si y solo si se podrá colorear el grafo con k . La segunda cuestión determina el cardinal del conjunto siguiente

$$P_G(k) = \#\{\text{coloraciones distintas de } G \text{ usando los colores } \{1, \dots, k\}\}.$$



Figura 13: Coloreado de España con 4 colores

P_G es una función de k , y de hecho es un polinomio en k , que se denomina el polinomio cromático de G , $P_G(k) = \sum_j \alpha_j k^j$. Claramente $P_G(k) = 0$ si y solo si $\chi(G) > k$; de hecho si α es la máxima raíz entera de $P_G(k)$, entonces $\chi(G) = \alpha + 1$.

Es fácil determinar el polinomio cromático de ciertos grafos

- $P_{K_n} = x(x-1)(x-2) \cdots (x-n+1)$.
- $P_{L_n} = x(x-1)^{n-1}$.

El polinomio cromático de un grafo puede descomponerse según las componentes conexas de un grafo.

Proposición 10.17 Sea G un grafo con r componentes conexas, G_1, G_2, \dots, G_r , entonces

$$P_G = P_{G_1} \cdot P_{G_2} \cdot \dots \cdot P_{G_r}.$$

Determinar el polinomio cromático de un grafo puede realizarse de manera recursiva utilizando las siguientes propiedades:

Teorema 10.18 Sea $G = (V, A)$ un grafo. Dada una arista $a \in A$ sea G_a la contracción de G por a . Entonces:

$$P_G = P_{G-\{a\}} - P_{G_a}.$$

Teorema 10.19 Sea $G = (V, A)$ un grafo y dos vértices x, y no adyacentes en G . Sea G_{xy}^+ el grafo que se obtiene de G añadiendo la arista xy y sea G_{xy}^{++} el grafo que resulta al identificar x con y en G . Entonces:

$$P_G = P_{G_{xy}^+} + P_{G_{xy}^{++}}.$$

Teorema 10.20 Sea $G = (V, A)$ un grafo y $G_1 = (V_1, A_1), G_2 = (V_2, A_2)$ subgrafos de G tales que $G = G_1 \cup G_2 = (V_1 \cup V_2, A_1 \cup A_2)$ y $G_1 \cap G_2 = (V_1 \cap V_2, A_1 \cap A_2)$ es un grafo completo. Entonces:

$$P_G = \frac{P_{G_1} \cdot P_{G_2}}{P_{G_1 \cap G_2}}.$$

Utilizando estas técnicas es fácil ver que $P_{C_n} = (x - 1)^n + (-1)^n(x - 1)$.

4. Bibliografía

Muchos de los tópicos de la asignatura se encuentran en casi cualquier libro de Matemática Discreta. Para el desarrollo de la asignatura se recomiendan los siguientes:

- N. L. Biggs, *Discrete Mathematics*. Oxford University Press. 2002.
- R. P. Grimaldi, *Discrete and combinatorial mathematics, an applied introduction*. Addison-Wesley. 1989.
- C. Munuera, J. Tena, *Codificación de la información*. Serv. Pub. U. de Valladolid. 1997.
- K. H. Rosen, *Discrete mathematics and its applications*. McGraw-Hill. 1999.

Además de la bibliografía ya mencionada, otras referencias utilizadas para la elaboración del proyecto docente son:

- R.B.J. Allenby, A. Slomson, *How to count, an introduction to Combinatorics*. CRC Press. 1991.
- M. Bóna, *A walk through combinatorics. An Introduction to Enumeration and Graph Theory*. World Scientific. 2006.
- P. Fernandez-Gallardo, J.L. Fernández Pérez. *Notas de Matemática Discreta*. Disponible en www.uam.es/pablo.fernandez. 2008.
- J. Gimbert, R. Moreno, J.M. Ribó, M. Valls, *Apropament a la teoria de grafs i als seus algorismes*. U. de LLeida. 1998.
- F. Harary, *Graph Theory*. Addison-Wesley. 1969.
- J. Matousek, J. Nešetřil, *Invitation to Discrete Mathematics*. Clarendon Press, Oxford. 1998.