

Introducción a la Teoría de Gráficas

César Hernández Cruz

16 de agosto de 2022

Índice general

1. Introducción	5
1.1. Definiciones básicas	5
1.2. Algunas familias de gráficas	8
1.2.1. Introducción	8
1.2.2. Gráficas de Intersección	9
1.3. Representaciones	10
1.4. Isomorfismo	12
1.5. Subgráficas	15
1.6. Caminos y conexidad	16
1.6.1. Caminos	16
1.6.2. Ciclos y trayectorias	16
1.6.3. Conexidad	17
1.7. Propiedades hereditarias	19
1.8. Operaciones en gráficas	22
1.8.1. Introducción	22
1.8.2. Productos	23
1.8.3. Complemento	24
1.8.4. Otras operaciones binarias	25
1.9. Digráficas	25
2. Árboles	29
2.1. Introducción	29
2.2. Caracterizaciones	31
2.3. Vértices de corte	35
2.4. Búsqueda en gráficas	36
2.5. BFS	39
2.6. DFS	42
2.7. Árboles de peso mínimo	44
2.8. Isomorfismo de árboles	48
3. Conexidad	49
3.1. Introducción	49

3.2. Bloques	50
3.3. Descomposiciones por orejas	53
3.4. Teorema de Menger	54
3.5. Conexidad por aristas	59
4. Gráficas eulerianas y hamiltonianas	63
4.1. Gráficas eulerianas	63
4.2. El algoritmo de Hierholzer	65
4.3. Gráficas hamiltonianas	70
5. Apareamientos	75
5.1. Apareamientos	75
5.2. Apareamientos en gráficas bipartitas	76
5.3. Apareamientos perfectos	79
5.4. El algoritmo de Egerváry	81
6. Coloraciones	83
6.1. Coloración por aristas	83
6.2. Coloración por vértices	86
7. Gráficas Planas	95
7.1. Gráficas planas y aplanables	95
7.2. Gráficas duales	98
7.3. La fórmula de Euler	100
7.4. El Teorema de los Cinco Colores	103
Index	107

Capítulo 1

Introducción

1.1. Definiciones básicas

Una **gráfica** G es una terna ordenada (V_G, E_G, ψ_G) . Llamaremos **vértices** a los *gráfica* elementos de V_G . El conjunto E_G es tal que $V_G \cap E_G = \emptyset$, y sus elementos son llamados **aristas**. La función ψ_G es la **función de incidencia** de G , que asocia a cada arista una pareja no ordenada de vértices (no necesariamente distintos) de G . En otras palabras

$$\psi_G: E_G \rightarrow \binom{V_G}{1} \cup \binom{V_G}{2}.$$

Toda gráfica puede ser representada mediante un diagrama, dibujando un punto (o círculo pequeño) por cada vértice, y una línea (no necesariamente recta) por cada arista, de tal forma que los extremos de la línea que representa a la arista e sean los puntos correspondientes a los vértices en el conjunto $\psi(e)$. Consideremos por ejemplo a la gráfica G con conjunto de vértices $\{v_i\}_{i=1}^{10}$, con conjunto de aristas $\{e_i\}_{i=1}^{17}$, y tal que ψ_G está dada por

$\psi_G(e_1) = \{v_1\},$	$\psi_G(e_7) = \{v_4, v_5\},$	$\psi_G(e_{13}) = \{v_7, v_8\},$
$\psi_G(e_2) = \{v_1\},$	$\psi_G(e_8) = \{v_2, v_5\},$	$\psi_G(e_{14}) = \{v_6, v_8\},$
$\psi_G(e_3) = \{v_1, v_2\},$	$\psi_G(e_9) = \{v_3\},$	$\psi_G(e_{15}) = \{v_6, v_8\},$
$\psi_G(e_4) = \{v_2, v_3\},$	$\psi_G(e_{10}) = \{v_6, v_7\},$	$\psi_G(e_{16}) = \{v_6, v_9\},$
$\psi_G(e_5) = \{v_3, v_4\},$	$\psi_G(e_{11}) = \{v_6, v_7\},$	$\psi_G(e_{17}) = \{v_6, v_{10}\}.$
$\psi_G(e_6) = \{v_4, v_5\},$	$\psi_G(e_{12}) = \{v_7, v_8\},$	

Entonces, un posible diagrama para G se muestra en la Figura 1.1. Notemos que hay una infinidad de posibles diagramas para una misma gráfica, por lo que siempre haremos énfasis en que la gráfica es un objeto matemático, en este caso una terna de la forma (V_G, E_G, ψ_G) , y no uno de los posibles diagramas para representarla.

Si $e \in E_G$ y $u, v \in V_G$ son tales que $\psi_G(e) = \{u, v\}$, diremos que u y v son los **extremos** de la arista e ; diremos también que u y v son **adyacentes**, que u es

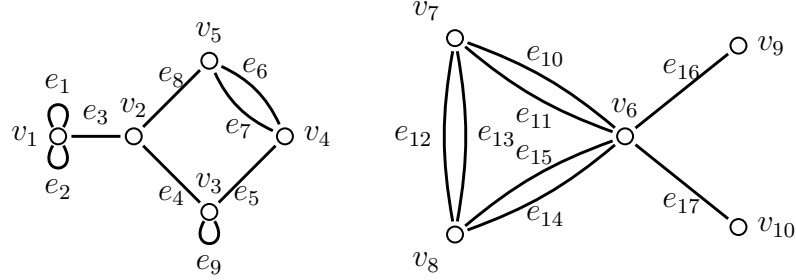


Figura 1.1: El diagrama de una gráfica con lazos y aristas múltiples.

vecindad

vecino de v , que u **domina** a v , que e **cubre** a u y a v , y que e es **incidente** en u y en v , y viceversa. El conjunto de vecinos de un vértice v es llamado **vecindad de v** y se denota por $N_G(v)$. Usualmente G denotará a una gráfica. Cuando no haya posibilidad de confusión, omitiremos el subíndice G , y escribiremos, por ejemplo $G = (V, E, \psi)$.

Una arista con extremos iguales es un **lazo** y una arista con extremos distintos es un **eslabón**. Dos o más aristas con los mismos extremos son **aristas paralelas** o **aristas múltiples**. Si dos aristas comparten un extremo diremos que son **adyacentes**. En la Figura 1.1, las aristas e_1, e_2 y e_9 son lazos. Las aristas e_1 y e_2 , e_6 y e_7 , e_{10} y e_{11} , e_{12} y e_{13} , y e_{14} y e_{15} son aristas múltiples. Debido a la existencia de los lazos, los únicos vértices adyacentes a sí mismos son v_1 y v_3 ; el vértice v_2 es adyacente a los vértices v_1, v_3 y v_5 , pero no es adyacente, por ejemplo, a v_4 .

gráfica simple

Al observar la Figura 1.1, podemos notar la importancia de etiquetar cada una de las aristas en la gráfica, pues al haber más de una arista con el mismo conjunto de extremos, sería imposible distinguirlas de no hacerlo. Sin embargo, podemos observar que este fenómeno ocurre únicamente debido a la existencia de las aristas múltiples. Es posible considerar a una gráfica sin aristas múltiples G como una pareja ordenada $G = (V_G, E_G)$, donde V_G es un conjunto, y E_G es un subconjunto (posiblemente vacío) de $\binom{V_G}{1} \cup \binom{V_G}{2}$, ya que entre cualesquiera dos vértices existe una única arista. Una **gráfica simple** G es una gráfica sin lazos ni aristas múltiples, por lo que en una gráfica simple podemos suponer adicionalmente que E_G es un subconjunto de $\binom{V_G}{2}$. Para la mayor parte de los temas que serán tratados en estas notas, será suficiente considerar gráficas simples, e incluso en los temas donde las gráficas con aristas múltiples tienen sentido (como gráficas eulerianas o gráficas planas), teniendo suficiente cuidado podemos evitar lidiar con la función de incidencia. De este modo, usualmente consideraremos que las gráficas consideradas en este texto serán gráficas simples. Notemos que para una gráfica G posiblemente con lazos y aristas múltiples, existe una gráfica simple que tiene una estructura muy semejante a la de G (tiene, salvo las auto-adyacencias, el mismo patrón de adyacencias que G), llamada la **gráfica subyacente de G** , U_G . El conjunto de vértices de U_G es el mismo que el de G , y dos vértices distintos en U_G son adyacentes si y sólo si son adyacentes en G . La

gráfica subyacente de la gráfica de la Figura 1.1 se muestra en la Figura 1.2.

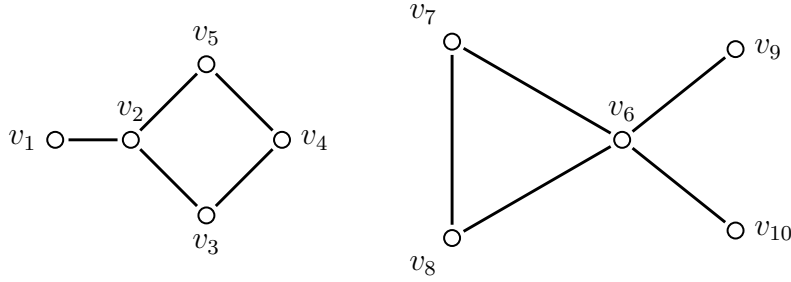


Figura 1.2: La gráfica (simple) subyacente de la gráfica de la Figura 1.1.

El **grado** $d_G(v)$ del vértice v es el número de aristas incidentes en v , donde los lazos se cuentan dos veces. Claramente, si G es simple, $d_G(v) = |N_G(v)|$. El grado máximo de G se denota por Δ_G , y el grado mínimo de G se denota por δ_G ; cuando no haya lugar a confusión, usualmente se omitirá el subíndice. Si G tiene vértices v_1, \dots, v_n , la sucesión $(d(v_1), \dots, d(v_n))$ se conoce como la **sucesión de grados** de G ; la sucesión de grados de una gráfica usualmente se escribe en forma no creciente; decimos que la sucesión $d = (d_1, \dots, d_n)$ es **gráfica** si existe una gráfica simple con sucesión de grados d . Una gráfica G es **k -regular** si $d(v) = k$ para cada $v \in V$; una gráfica es **regular** si es k -regular para algún natural k . Un vértice de grado 0 es un **vértice aislado**. En la gráfica de la Figura 1.1, el vértice v_1 tiene grado 5, y su vecindad es el conjunto $\{v_1, v_2\}$, mientras que en su gráfica subyacente (en la Figura 1.2), el vértice v_1 tiene grado 1, y su vecindad es el conjunto $\{v_2\}$. Análogamente, en la gráfica de la Figura 1.1, el vértice v_6 tiene grado 6, y su vecindad es $\{v_7, v_8, v_9, v_{10}\}$, mientras que en la gráfica de la Figura 1.2, el mismo vértice tiene grado 4, y la misma vecindad.

Si G es una gráfica y S es un subconjunto de V , decimos que S es **independiente** o **estable** si cualesquiera dos vértices distintos en S son no adyacentes, y decimos que S es un **clan** si cualesquiera dos vértices distintos en S son adyacentes. En la gráfica de la Figura 1.2, el conjunto $I = \{v_2, v_4, v_6\}$ es un conjunto independiente, y de hecho, es fácil observar que no es posible añadir vertice alguno a I de tal manera que el conjunto obtenido siga siendo independiente. En este caso, decimos que el conjunto I es máximo por contención con la propiedad de ser independiente, o que es un *conjunto independiente máximo por contención* pues no está contenido propiamente en conjunto independiente alguno. Sin embargo, este no es el conjunto independiente de mayor cardinalidad que puede encontrarse en esta gráfica. Por ejemplo, el conjunto $J = \{v_1, v_3, v_5, v_7, v_9, v_{10}\}$ es un conjunto independiente con más elementos que I . En este caso, es fácil verificar mediante una exploración exhaustiva que no hay un conjunto independiente con más elementos que J , por lo que diremos que J es un *conjunto independiente máximo*. El **número de independencia** de una gráfica G , denotado por α_G , se define como la cardinalidad de un conjunto independinete

máximo en G . Análogamente, podemos observar que el conjunto $\{v_2, v_5\}$ es un clan máximo por contención en la gráfica de la Figura 1.2 aunque no es un clan máximo, pero el conjunto $\{v_6, v_7, v_8\}$ es un clan máximo en la misma gráfica. Para una gráfica G , el **número de clan** de G se define como la cardinalidad de un clan máximo en G , y se denota por ω_G .

número de clan

conexa

inconexa

Una gráfica es **conexa** si para cualquier partición de su conjunto de vértices en dos conjuntos no vacíos X y Y , existe al menos una arista con un extremo en X y el otro en Y ; de otra forma, la gráfica es **inconexa**. En matemáticas, usualmente un objeto conexo es aquel en el que cualesquiera dos puntos pueden “alcanzarse”, bajo alguna noción de alcanzabilidad dependiente del contexto. Quizá esta primera definición de conexidad resulta contraria a la intuición; más adelante se introducirá una equivalencia del concepto de conexidad más acorde a las definiciones que suelen encontrarse en otras áreas de las matemáticas.

1.2. Algunas familias de gráficas

1.2.1. Introducción

Muchos problemas en teoría de gráficas son difíciles de atacar en el caso general; por este motivo, es importante contar con familias de gráficas con una estructura bien determinada que nos permita llevar a cabo exploraciones iniciales. A veces será posible determinar, con menos esfuerzo, que una propiedad que se desea demostrar para todas las gráficas es cierta en alguna de estas familias infinitas. Esto tiene varias consecuencias deseables, por ejemplo, al demostrar que la propiedad \mathcal{P} se cumple para todas las gráficas en la familia \mathcal{F} , una potencial demostración de que todas las gráficas tienen la propiedad \mathcal{P} puede iniciar suponiendo que la gráfica considerada no está en la familia \mathcal{F} , lo que representa una hipótesis adicional. En otras ocasiones, será posible “descomponer” a una gráfica arbitraria en subgráficas más pequeñas, tales que cada una de éstas se encuentre en la familia \mathcal{F} y, a veces, saber si cada una de estas subgráficas tiene la propiedad \mathcal{P} será de ayuda para determinar si la gráfica original también la tiene. Por otro lado, si es posible determinar que ninguna gráfica de la familia \mathcal{F} tiene la propiedad \mathcal{P} , esta información puede ser utilizada como un punto de partida para caracterizar a las gráficas que cumplen la propiedad. Por este motivo, en esta sección presentamos algunas de las familias de gráficas más estudiadas, y por lo tanto mejor comprendidas.

bipartita

Existe una única gráfica G con $V_G = \emptyset$, a la que llamamos la **gráfica nula**. Una gráfica con un único vértice es **trivial**, y cualquier otra gráfica es **no trivial**. Una gráfica es **vacía** si su conjunto de aristas es vacío y, si G es simple, diremos que es **completa** si su conjunto de aristas es igual a $\binom{V}{2}$. Una gráfica G es **bipartita** si su conjunto de vértices V admite una partición $V = (X, Y)$ de tal forma que X y Y son conjuntos independientes. Una gráfica bipartita G con bipartición (X, Y) será denotada por $G[X, Y]$. Si $G[X, Y]$ es simple y cada vértice de X es adyacente a

cada vértice de Y , entonces diremos que G es una gráfica **bipartita completa**. Una *bipartita completa* **estrella** es una gráfica bipartita completa $G[X, Y]$ donde $|X| = 1$ o $|Y| = 1$. Una gráfica es **escindible** si su conjunto de vértices admite una partición $V = (X, Y)$ de *escindible* tal forma que X es un conjunto independiente y Y es un clan.

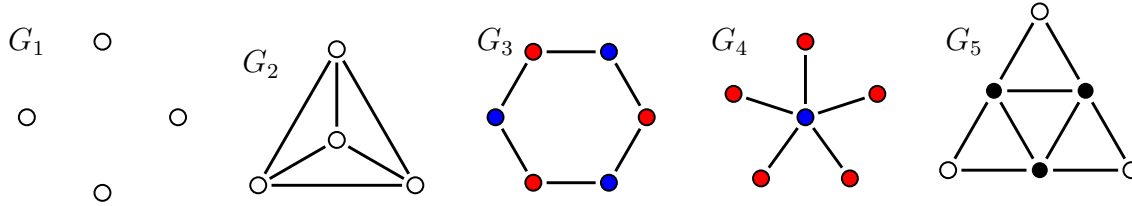


Figura 1.3: Ejemplos de gráficas vacía, completa, bipartita, bipartita completa y escindible.

Dado un entero positivo k , una gráfica G es k -partita si su conjunto de vértices admite una partición en a lo más k conjuntos independientes. Una gráfica es **multi-partita** si es k -partita para algún entero positivo k . Es fácil observar que las gráficas *multipartita* 1-partitas son las gráficas vacías, y las gráficas 2-partitas coinciden con las gráficas bipartitas. Así como existen las gráficas bipartitas completas, también podemos definir a las gráficas k -partitas completas, como aquellas gráficas k -partitas donde cada vértice es adyacente a cada vértice en una parte distinta a la suya. Una gráfica es **multipartita completa** si es k -partita completa para algún entero positivo k . Una *multipartita completa* familia relacionada a las k -partitas completas es la de los k -clusters. Nuevamente, para un entero positivo k , decimos que una gráfica es un k -cluster si su conjunto de vértices admite una partición en a lo más k clanes, y no hay aristas adicionales en la gráfica, i.e., no hay aristas entre clanes distintos. Una gráfica es un **cluster** si es *cluster* un k -cluster para algún entero positivo k .

Una **trayectoria** es una gráfica simple en la que su conjunto de vértices admite un *trayectoria* orden lineal de tal forma que dos vértices son adyacentes si y sólo si son consecutivos en el orden. Un **ciclo** de tres o más vértices es una gráfica simple en la que su conjunto *ciclo* de vértices admite un orden cíclico de tal forma que dos vértices son adyacentes si y sólo si son consecutivos en el orden; un vértice con un lazo es un ciclo de un vértice, y un par de vértices unidos por un par de aristas paralelas es un ciclo de dos vértices. La **longitud** de una trayectoria o un ciclo es su número de aristas. Una trayectoria o ciclo de longitud k es una k -trayectoria o k -ciclo, respectivamente. Una trayectoria o ciclo es **par** o **impar** dependiendo de la paridad de su longitud.

1.2.2. Gráficas de Intersección

Dada una familia de conjuntos \mathcal{F} , la **gráfica de intersección** asociada a \mathcal{F} es *gráfica de in-* la gráfica G cuyo conjunto de vértices es \mathcal{F} , y tal que dos vértices $x, y \in V_G$ son *tersección* adyacentes si y sólo si su intersección es no vacía.

gráfica de intervalos

Quizá el ejemplo mejor conocido de las gráficas de intersección es el de las gráficas de intervalos. Una gráfica G es una **gráfica de intervalos** si es la gráfica de intersección de una familia de intervalos en \mathbb{R} . Si G es la gráfica de intersección de un conjunto de intervalos \mathcal{I} sobre la recta real, diremos que \mathcal{I} es una representación en intervalos de G . Una pregunta que surge naturalmente al ver esta definición es, ¿cualquier gráfica G tiene una representación en intervalos? Quizá incluso valga la pena regresar a la pregunta más general que se puede hacer en este contexto, ¿cualquier gráfica es una gráfica de intersección?, o, en otras palabras, ¿para cualquier gráfica G existe una familia de conjuntos \mathcal{F} tal que G es la gráfica de intersección asociada a \mathcal{F} ? La siguiente proposición responde negativamente a la primera de estas dos preguntas.

Proposición 1.2.1. *Si un ciclo es una gráfica de intervalos, entonces es un 3-ciclo.*

De manera quizá un poco sorprendente, después del resultado de la Proposición 1.2.1, se tiene que, por otro lado, la respuesta a la segunda pregunta antes planteada es afirmativa. Cualquier gráfica es la gráfica de intersección de algún conjunto. Si se piensa con un poco de detenimiento, esto no debería de sorprendernos, a fin de cuentas, el conjunto de todas las gráficas de intervalos es un subconjunto (bastante pequeño) del conjunto de todas las gráficas de intersección.

Proposición 1.2.2. *Toda gráfica es una gráfica de intersección.*

gráfica de líneas

Una idea que surge naturalmente es considerar gráficas de intersección de familias de subconjuntos de vértices de una gráfica. Al considerar todos los subconjuntos de vértices de una gráfica con una propiedad particular, y tomar la gráfica de intersección de dichos subconjuntos, obtenemos una operación 1-aria, que a cada gráfica le asocia una gráfica de intersección. El ejemplo arquetípico de estos operadores es la gráfica de líneas de una gráfica G , que es la gráfica de intersección del conjunto de E_G . La definición que suele encontrarse para esta familia es la siguiente. Dada una gráfica G , la **gráfica de líneas** de G , L_G , es la gráfica cuyo conjunto de vértices es el conjunto de aristas de G , y tal que dos vértices e y f de L_G (dos aristas de G) son adyacentes, si y sólo si, e y f tienen un extremo común en G (es decir, si la intersección de e y f es no vacía).

1.3. Representaciones

Además de la definición formal, hay varias maneras de representar a una gráfica G con n vértices $\{v_1, \dots, v_n\}$ y m aristas $\{e_1, \dots, e_m\}$. Dependiendo del uso que vayamos a darle, hay algunas que resultan más eficientes que otras. En esta sección sólo consideraremos las tres más comunes. La primera es la representación mediante **listas de adyacencias**, que como su nombre lo indica, consiste en un conjunto $\{A_1, \dots, A_n\}$ donde A_i es la lista de vértices adyacentes a v_i . El uso de listas es fundamental para poder representar aristas paralelas, pues permiten elementos repetidos.

A cualquier gráfica G con n vértices le corresponde una matriz de $n \times n$, A_G , llamada la **matriz de adyacencia** de G , con $A_G = (a_{ij})$, donde a_{ij} es el número de aristas que hay en G con extremos v_i y v_j . En el caso en el que no hay aristas múltiples en G tenemos que

$$a_{ij} = \begin{cases} 1 & \text{si } v_i v_j \in E, \\ 0 & \text{si } v_i v_j \notin E. \end{cases}$$

Si D tiene m aristas, también podemos considerar la **matriz de incidencia**, M_G , que es una matriz de $n \times m$ dada por $M_G = (m_{ij})$, donde m_{ij} es el número de veces que v_i y e_j son incidentes. En particular, si el vértice v_i tiene un lazo e_j , entonces $m_{ij} = 2$.

Cuando G es una gráfica bipartita, tenemos una representación más compacta. Supongamos que $G[X, Y]$ es una gráfica bipartita con $X = \{x_1, \dots, x_r\}$ y $Y = \{y_1, \dots, y_s\}$. La **matriz de adyacencia bipartita** de G es la matriz B de $r \times s$, dada por $B = (b_{ij})$ donde b_{ij} es el número de aristas que unen a x_i con y_j .

Cada una de las representaciones tiene ventajas y desventajas. La ventaja obvia de la representación en listas de adyacencias en contraposición con ambas matrices es que optimiza el espacio necesario para codificar a una gráfica, pues no es necesario guardar las no-adyacencias (cada entrada igual a 0 en las matrices). Quizá la principal utilidad de la matriz de adyacencia es la implementación del algoritmo de Floyd-Warshall, gracias a un resultado simple que relaciona las entradas de la k -ésima potencia de la matriz de adyacencia con el número de caminos entre los vértices de una gráfica. La matriz de incidencia no tiene muchas aplicaciones o ventajas, sin embargo, es posible utilizarla para realizar demostraciones formales y compactas como la siguiente.

Proposición 1.3.1. *Para toda gráfica G , se cumple*

$$\sum_{v \in V} d(v) = 2|E|.$$

Demostración. Sea M la matriz de incidencia de G , y supongamos que $V = \{v_1, \dots, v_n\}$, donde v_i corresponde al i -ésimo renglón de M . Sabemos que las entradas de cada columna de M suman 2, pues o bien la j -ésima columna corresponde a un lazo, y hay una única entrada distinta de 0, que es igual a 2, o bien corresponde a un eslabón, y hay exactamente dos entradas iguales a 1, una por cada extremo de la arista. Por otro lado, las entradas del i -ésimo renglón de M suman $d(v_i)$, pues precisamente las entradas distintas de 0 corresponden a todas las aristas de las que v_i es extremo. De este modo, sumando de dos formas distintas todas las entradas de

M , tenemos que:

$$\begin{aligned}
 2|E| &= \sum_{j=1}^m 2 \\
 &= \sum_{j=1}^m \sum_{i=1}^n M_{ij} \\
 &= \sum_{i=1}^n \sum_{j=1}^m M_{ij} \\
 &= \sum_{i=1}^n d(v_i) \\
 &= \sum_{v \in V} d(v).
 \end{aligned}$$

□

El siguiente corolario, aunque bastante sencillo, es de gran utilidad en distintos contextos dentro de la teoría de gráficas.

Corolario 1.3.2. *En cualquier gráfica, el número de vértices de grado impar es par.*

Demostración. Sea G una gráfica y sea X el subconjunto de V que contiene a todos los vértices de grado par en G . Claramente, si $Y = V - X$, entonces Y consta de los vértices de G de grado impar, y (X, Y) es una partición de V .

Entonces,

$$\begin{aligned}
 0 &\stackrel{2}{\equiv} 2|E| \\
 &= \sum_{v \in V} d(v) \\
 &= \sum_{v \in X} d(v) + \sum_{v \in Y} d(v) \\
 &\stackrel{2}{\equiv} \sum_{v \in Y} d(v)
 \end{aligned}$$

Es decir, la suma de todos los grados de los vértices en Y es par. Pero Y consta de todos los vértices de G de grado impar, y sabemos que una suma de impares es par si y sólo si hay un número par de sumandos. Por lo tanto, $|Y|$ es par. □

1.4. Isomorfismo

Una pregunta fundamental en matemáticas es ¿cuándo dos objetos son iguales? Aunque formalmente, la respuesta a esta pregunta siempre es la misma: dos objetos

son iguales si y sólo si son el mismo objeto. Entonces, la pregunta que realmente queremos hacer es ¿cuándo dos objetos son esencialmente iguales? Es decir, cuándo, desde el punto de vista de la teoría en la que estamos, podemos tratar a dos objetos como si fueran el mismo. Como un ejemplo muy sencillo, consideremos a la teoría de grupos. Si $\langle G_1, \circ \rangle$ y $\langle G_2, * \rangle$ son grupos, cada uno con dos elementos, digamos $G_1 = \{a, b\}$ y $G_2 = \{c, d\}$, entonces, sabemos que, forzosamente, uno de los dos elementos en cada grupo funge el papel de la identidad, y el otro, dado que debe tener un inverso, cumple que al operarlo consigo mismo debe de resultar en la identidad. Si para nuestro ejemplo suponemos que a es la identidad en G_1 y c es la identidad en G_2 , entonces se tiene que $b \circ b = a$ y que $d * d = c$. En estas condiciones, si estamos trabajando con G_1 , podemos hacer una sustitución de cada ocurrencia de a por c , cada ocurrencia de b por d , y cada ocurrencia de \circ por $*$, y todo lo que demostremos seguirá siendo cierto. En otras palabras, G_1 y G_2 se comportan de la misma forma, lo único que los distingue es un “cambio de nombre”, que puede interpretarse como una función biyectiva φ entre G_1 y G_2 que tiene una propiedad adicional, que en este caso es preservar el comportamiento de la operación, es decir, para cualesquiera x y y en G_1 , $\varphi(x \circ y) = \varphi(x) * \varphi(y)$.

La última expresión del párrafo anterior es la propiedad que define al isomorfismo de grupos; cada rama de las matemáticas tiene su propia noción de isomorfismo, dependiendo de qué es aquello que caracteriza a los objetos que estamos estudiando. En el caso de la teoría de gráficas, la estructura que nos interesa es la de adyacencia. Intuitivamente, dos gráficas son esencialmente la misma si podemos “dibujarlas igual”, es decir, si podemos asociarles el mismo diagrama, más aún, si a dos gráficas distintas se les puede asociar el mismo diagrama, éste nos ayudará de una manera obvia a encontrar una biyección entre sus conjuntos de vértices. Entonces, formalmente, ¿qué se necesita para que a dos gráficas se les pueda asociar el mismo diagrama? Basta con que los números de vértices y de aristas coincidan para ambas gráficas, y que la relación de adyacencia y no adyacencia sea la misma.

Dos gráficas G y H son **iguales** o **idénticas** (denotado $G = H$) si $V_G = V_H$, $E_G = E_H$ y $\psi_G = \psi_H$. Decimos que G y H son **isomorfas**, denotado $G \cong H$, si existen biyecciones $\theta: V_G \rightarrow V_H$ y $\phi: E_G \rightarrow E_H$ tales que $\psi_G(e) = uv$ si y sólo si $\psi_H(\phi(e)) = \theta(u)\theta(v)$; el par (θ, ϕ) es un isomorfismo entre G y H .

Si ni G ni H tienen aristas múltiples y tenemos que $G = (V_G, E_G)$ y $H = (V_H, E_H)$, entonces podemos simplificar esta condición y decir que G es isomorfa a H si existe una biyección $\varphi: V_G \rightarrow V_H$ tal que para cualesquiera vértices $u, v \in V_G$, se tiene que $uv \in E_G$ si y sólo si $\varphi(u)\varphi(v) \in E_H$; la biyección φ es un **isomorfismo** entre G y H . Un **automorfismo** es un isomorfismo $\varphi: G \rightarrow G$.

Si G es una gráfica, con $u, v \in V$, decimos que u y v son **vértices similares** si existe un automorfismo α de G tal que $\alpha(u) = v$. Las gráficas donde todos los vértices son similares son llamadas **vértice-transitivas**. Las gráficas donde no hay pares de vértices similares, son llamadas **asimétricas**; estas gráficas tienen grupo de automorfismos igual a $\{1_V\}$. El grupo de automorfismos de una gráfica se denota por

$\text{Aut}(G)$, y su cardinalidad por $\text{aut}(G)$.

Retomando el ejemplo con el que iniciamos esta sección, en teoría de grupos es fácil demostrar que todos los grupos con dos elementos son isomorfos; usualmente se dice que, *salvo isomorfismo*, existe un único grupo con dos elementos. En teoría de gráficas sucede un fenómeno similar con algunas gráficas. Fijemos un entero positivo n , y consideremos a las gráficas completas de orden n . Hay una infinidad de gráficas completas de orden n , pues para cualquier conjunto S de n elementos, podemos considerar a la gráfica G que tiene como conjunto de vértices a S , y tal que su conjunto de aristas es $\binom{S}{2}$. Sin embargo, a todas las gráficas antes referidas se les puede asociar el mismo diagrama, i.e., todas las gráficas completas de orden n son isomorfas. Luego, para cada $n \in \mathbb{Z}^+$ existe una única gráfica completa, salvo isomorfismo. Al ser única, es posible hablar de “la” gráfica completa de orden n , y asociarle una notación para identificarla; la gráfica completa de orden n se denota por K_n . Las gráficas completas no son las únicas que tienen esta propiedad; el siguiente resultado menciona algunas gráficas clásicas que son únicas salvo isomorfismo, e introduce la notación estándar para las mismas.

Proposición 1.4.1. *Sean n, k, n_1, \dots, n_k enteros positivos. Las siguientes gráficas son únicas salvo isomorfismo.*

1. La gráfica completa de orden n , y se denota por K_n .
2. La gráfica vacía de orden n , y se denota por O_n .
3. La trayectoria de orden n , y se denota por P_n .
4. El ciclo de orden n , y se denota por C_n .
5. La gráfica k -partita completa, donde la i -ésima parte tiene cardinalidad n_i , para cada $i \in \{1, \dots, k\}$, y se denota por K_{n_1, \dots, n_k} .

Demostración. Demostraremos el primer y tercer incisos, las demás demostraciones son análogas y se dejan como ejercicio.

Sean G y H gráficas completas de orden n . Al tener la misma cardinalidad, existe una biyección φ entre V_G y V_H ; afirmamos que φ es un isomorfismo. Ya que G y H son completas, para cualesquiera vértices u y v en V_G , tenemos que las proposiciones $uv \in E_G$ y $\varphi(u)\varphi(v) \in E_H$ son verdaderas, por lo que su bicondicional, que es la definición de isomorfismo entre G y H , también lo es.

Sean G y H trayectorias de orden n . Si $V_G = \{v_1, \dots, v_n\}$ y $V_H = \{u_1, \dots, u_n\}$, por la definición de trayectoria podemos suponer que dos vértices distintos v_i y v_j de G son adyacentes si y sólo si $j = i + 1$ o $i = j + 1$, y análogamente para H . Así, proponemos a la función $\varphi: V_G \rightarrow V_H$ dada por $\varphi(v_i) = u_i$ para cada $i \in \{1, \dots, n\}$. Claramente φ es una biyección, veamos que es un isomorfismo. Si $v_i v_j \in E_G$, entonces podemos suponer que $j = i + 1$ (el caso cuando $i = j + 1$ es análogo). Por la definición de φ , tenemos que $\varphi(v_i) = u_i$ y $\varphi(v_j) = u_j = u_{i+1}$. Como $u_i u_{i+1} \in E_H$,

concluimos que $\varphi(v_i)\varphi(v_j) \in E_H$. Recíprocamente, si $v_iv_j \notin E_G$, entonces $i \neq j+1$ y $j \neq i+1$. Se sigue que $\varphi(v_i)\varphi(v_j) = u_iu_j \notin E_H$. \square

1.5. Subgráficas

Dadas dos gráficas (simples) G y H , si $V_H \subseteq V_G$ y $E_H \subseteq E_G$, entonces decimos que H es una **subgráfica** de G , y que G es una supergráfica de H , denotado $H \subseteq G$. Si $H \subseteq G$ y $H \neq G$, decimos que H es una **subgráfica propia** de G . En caso de que $V_H = V_G$, decimos que H es una **subgráfica generadora** de G , y que G es una **supergráfica generadora** de H . *subgráfica generadora*

Si V' es un subconjunto de V , la subgráfica de G cuyo conjunto de vértices es V' , y cuyas aristas son exactamente aquellas aristas de G que tienen ambos extremos en V' , es llamada la subgráfica de G **inducida** por V' , y es denotada por $G[V']$; decimos que $G[V']$ es una **subgráfica inducida** de G . La subgráfica inducida $G[V - V']$ se denota por $G - V'$, y es exactamente la subgráfica obtenida de G al borrar los vértices de V' junto con todas las aristas que inciden en ellos. Si $V' = \{v\}$, escribiremos $G - v$ en lugar de $G - \{v\}$. Bajo esta definición, dad una gráfica G , un subconjunto S de V es independiente si $G[S]$ es una gráfica vacía, y es un clan si $G[S]$ es una gráfica completa. *subgráfica inducida*

Si G y H son gráficas, y G contiene una subgráfica isomorfa a H , diremos que G contiene a H (aunque formalmente no sea el caso, pues sólo contiene una copia isomorfa de H). Si G contiene una subgráfica inducida isomorfa a H , diremos que G contiene a H como subgráfica inducida, o que G contiene una H inducida. Si G no contiene a H como subgráfica inducida, diremos que G es **libre de H** o **H -libre**. Si \mathcal{F} es un conjunto de gráficas, y para cada $F \in \mathcal{F}$ tenemos que G es F -libre, diremos que G es **\mathcal{F} -libre**. *H -libre*

Análogamente, si E' es un subconjunto no vacío de E , la subgráfica de G cuyo conjunto de vértices es el conjunto de extremos de aristas en E' , y cuyo conjunto de aristas es E' es la subgráfica de G **inducida** por E' , y se denota por $G[E']$. La subgráfica generadora de G con conjunto de aristas $E - E'$ se escribe $G - E'$; es la subgráfica obtenida de G al borrar las aristas en E' . De manera similar, la supergráfica obtenida de G al añadir un conjunto de aristas E' se denota por $G + E'$. Si $E' = \{e\}$ escribimos $G - e$ y $G + e$ en lugar de $G - \{e\}$ y $G + \{e\}$, respectivamente.

Dado un número natural k , un **k -factor** de la gráfica G es una subgráfica generadora y k -regular de G . La gráfica G es **k -factorizable** si existe una familia \mathcal{F} de k -factores de G , ajenos por aristas dos a dos, y tales que $\bigcup_{F \in \mathcal{F}} E_F = E_G$. *k -factor
 k -factorizable*

1.6. Caminos y conexidad

1.6.1. Caminos

camino

Un **camino** W en una gráfica G es una sucesión alternante de vértices y aristas de la forma

$$W = (v_0, e_1, v_1, e_2, \dots, e_{k-1}, v_{k-1}, e_k, v_k),$$

donde v_i es un vértice de G para cada $i \in \{0, \dots, k\}$ y e_i es una arista de G tal que $\psi(e_i) = v_{i-1}v_i$ para cada $i \in \{1, \dots, k\}$. La longitud de W , $\ell(W)$, es k . Si los vértices inicial y final del camino W son u y v , respectivamente, decimos que W es un uv -camino. Cuando G es una gráfica simple se tiene que entre cualesquiera dos vértices hay a lo más una arista, por lo que es redundante indicar qué aristas son utilizadas en un camino W y éste puede ser descrito únicamente como una sucesión de vértices de la forma

$$W = (v_0, v_1, \dots, v_{k-1}, v_k),$$

donde $v_{i-1}v_i \in E$ para cada $i \in \{1, \dots, k\}$.

Si W_1 y W_2 son caminos, $W_1 = (u_0, \dots, u_k)$, $W_2 = (v_0, \dots, v_l)$, tales que $u_k = v_0$, y tenemos que i, j cumplen $0 \leq i < j \leq k$, entonces definimos a los caminos W_1W_2 , $u_iW_1u_j$ y W_1^{-1} como

- $W_1W_2 = (u_0, \dots, u_k, v_1, \dots, v_l)$,
- $u_iW_1u_j = (u_i, u_{i+1}, \dots, u_{j-1}, u_j)$, y
- $W_1^{-1} = (u_k, u_{k-1}, \dots, u_1, u_0)$.

En algunas ocasiones, al utilizar la notación $u_iW_1u_j$, si $i = 0$ escribiremos únicamente W_1u_j , y si $j = k$, escribiremos únicamente u_iW_1 . Si X y Y son subconjuntos de V , un XY -camino se define como un camino en el que el vértice inicial está en X , el vértice final está en Y , y ningún vértice intermedio está en $X \cup Y$.

1.6.2. Ciclos y trayectorias

paseo

trayectoria

Un camino que no repite aristas es un **paseo** y un camino que no repite vértices es una **trayectoria**. Un camino es **cerrado** si su vértice inicial es igual a su vértice final. Un **circuito** es un paseo cerrado, y un **ciclo** es un circuito que no repite vértices, salvo el primero que es igual al último.

Dada la importancia de los ciclos, existen muchas condiciones suficientes para garantizar su existencia. Tenemos por ejemplo el siguiente resultado.

Proposición 1.6.1. *Sea G una gráfica. Si $\delta \geq 2$, entonces G tiene al menos un ciclo.*

Demostración. Sea $P = (x_0, \dots, x_k)$ una trayectoria de longitud máxima en G . Como $\delta \geq 2$, x_0 debe tener algún vecino x tal que $x \neq x_1$. Como P es máxima, debe existir $i \in \{2, \dots, k\}$ tal que $x = x_i$, pues de lo contrario, la trayectoria $(x, x_0)P$ sería de mayor longitud que P , lo cual no puede suceder por la elección de P . Así, $(Px_i)(x_i, x_0)$ es un ciclo en G . \square

Proposición 1.6.2. *Sea G una gráfica. Si W es un uv -camino en G , entonces existe una uv -trayectoria contenida en W .*

Demostración. Sea $W = (u = v_0, \dots, v_k = v)$ un uv -camino. Si $u = v$, entonces (u) es una uu -trayectoria de longitud cero. Supongamos que $u \neq v$, y procedamos por inducción sobre la longitud de W . Si W es de longitud 1, entonces es una trayectoria. Supongamos entonces que la longitud de W es estrictamente mayor que 1. Si W no repite vértices, entonces es la uv -trayectoria buscada. De lo contrario, sean i el primer índice tal que existe $j > i$ con $v_i = v_j$. Entonces, el uv -camino W' definido por $v_0Wv_iv_{j+1}Wv_k$ es un uv -camino contenido en W , pero estrictamente más corto que W . Aplicando la hipótesis inductiva a W' , obtenemos la uv -trayectoria deseada. \square

Es posible realizar una demostración muy semejante a la de la Proposición 1.6.2 para obtener el siguiente resultado, que será de gran utilidad más adelante.

Proposición 1.6.3. *Sea G una gráfica. Si W es un camino cerrado impar en G , entonces W contiene un ciclo impar.*

1.6.3. Conexidad

Recordemos que al final de la Sección 1.1 definimos una gráfica G como conexa si para cualquier partición de V en dos conjuntos X y Y , existe una arista con un extremo en X y el otro en Y . En ese momento, se mencionó que esta definición no corresponde con la noción intuitiva clásica de conexidad en otras áreas de las matemáticas, pues nos faltaba una noción de “alcanzabilidad” para poder decir que cualquier vértice “alcanza” a cualquier otro. Los caminos, definidos en la subsección anterior, son la herramienta faltante para poder plantear una caracterización de las gráficas conexas en terminos de alcanzabilidad. Diremos que un vértice u **alcanza** a un vértice v en G si existe un uv -camino en G .

Proposición 1.6.4. *Una gráfica G es conexa si y sólo si entre cualesquiera dos vértices existe un camino.*

Demostración. Supongamos primero que G es conexa, y sean u y v vértices en G . Sea X el subconjunto de V dado por

$$X = \{x \in V : \text{existe un } ux\text{-camino}\}.$$

Si $V - X = \emptyset$, entonces $v \in X$, y por lo tanto existe un uv -camino en G . De otra forma, $(X, V - X)$ es una partición de V , por lo que existe una $X(V - X)$ -arista,

digamos xy . Por definición, que $x \in X$ significa que existe un ux -camino W , lo que implica que $W(xy)$ es un uy -camino en G . Esto último resulta en una contradicción, pues tenemos que $y \in X \cap Y$. Como la contradicción surge de suponer que $V - X$ no es vacío, concluimos que $V - X = \emptyset$ y por lo tanto existe un uv -camino en G .

Recíprocamente, supongamos que entre cualesquiera dos vértices existe un camino. Sea (X, Y) una partición de V , y sean x, y tales que $x \in X$ y $y \in Y$. Por hipótesis existe una xy -trayectoria W en G con $W = (x = v_0, \dots, v_k = y)$. Como el primer vértice de W está en X , el último está en Y y $X \cap Y = \emptyset$, entonces existe un último índice $i \in \{0, \dots, k-1\}$ tal que $v_i \in X$. Así, $v_i v_{i+1}$ es una XY -arista. \square

*componentes
conexas*

A las subgráficas de una gráfica G , máximas por contención con la propiedad de ser conexas, se les llama **componentes conexas**. Denotamos por c_G al número de componentes conexas de una gráfica G . Alternativamente, podemos usar la relación de alcanzabilidad definida al principio de esta subsección. Considerada como una relación binaria sobre el conjunto de vértices de G , esta relación es claramente de equivalencia, y por lo tanto induce una partición de V . Las subgráficas inducidas por las clases de equivalencia de esta partición son las componentes conexas de G . De este modo, una gráfica es conexa si y sólo si tiene una única componente conexa.

distancia

geodésica

radio

diámetro

centro

corte por vértices

vértice de corte

corte por aristas

Gracias a la noción de longitud de un camino es posible definir una métrica sobre cualquier gráfica de la siguiente forma. Para una gráfica G , si u y v son vértices de G , la **distancia** entre u y v , $d(u, v)$, se define como la longitud de la trayectoria más corta entre u y v , si u y v están en la misma componente conexa, o como infinito, si no. Dados dos vértices u y v de G , una **uv -geodésica** es una uv -trayectoria de longitud $d(u, v)$. Una **geodésica** es una uv -geodésica para algún par de vértices u y v . Asociadas a la noción de distancia, existen dos parámetros que indican qué tan alejados están los vértices en una gráfica. El **radio** de una gráfica G se define como $\min_{u \in V_G} \{\max_{v \in V_G} \{d(u, v)\}\}$, y el **diámetro** de G se define como $\max_{u \in V_G} \{\max_{v \in V_G} \{d(u, v)\}\}$. Asociado a estos dos parámetros se tiene el concepto de **excentricidad** de un vértice; la excentricidad del vértice v en G se define como $\max_{u \in V_G} \{d(u, v)\}$, es decir, la máxima distancia que existe entre el vértice v y cualquier otro vértice de G . Claramente, el diámetro de G es el máximo de las excentricidades, y el radio es el mínimo de las excentricidades. Un **centro** de G es un vértice con excentricidad mínima en G . Aunque pudiera parecer contrario a la intuición, hay gráficas en las que cada vértice es un centro, por ejemplo los ciclos.

En una gráfica G un **corte por vértices** es un subconjunto V' de V tal que $G - V'$ es inconexa. Un **k -corte por vértices** (o simplemente **k -corte**) es un corte por vértices de cardinalidad k . Las gráficas completas no tienen cortes por vértices, y son las únicas con esta propiedad. Notemos que en una gráfica inconexa el conjunto vacío es un corte por vértices. Decimos que v es un **vértice de corte** si $G - v$ tiene más componentes conexas que G .

Para una gráfica G y subconjuntos X y Y de V , denotamos por $E[X, Y]$ al conjunto de aristas con un extremo en X y el otro en Y . Cuando $Y = V - X$, el conjunto $E[X, Y]$ es el **corte por aristas** de G asociado a X , y lo denotamos por

$\partial(X)$. El corte por aristas $\partial(v)$ asociado a un vértice v es un corte por aristas trivial. Un **k -corte por aristas** es un corte por aristas con k elementos. Una arista e tal que $G - e$ tiene más componentes conexas que G es una **arista de corte** o **punto de corte**. *punto de corte*

Lema 1.6.5. *Sea G una gráfica. Si $e \in E$, entonces e es un puente de G si y sólo si no pertenece a ciclo alguno en G .*

Demostración. Demostraremos ambas implicaciones por contrapositiva. Supongamos primero que e pertenece al ciclo C , donde $e = xy$ y $C = (x = x_0, y = x_1, x_2, \dots, x_k, x_0)$. Sean u y v vértices arbitrarios y distintos en V para los cuales existe un uv -camino W en G ; veamos que existe un uv -camino en $G - e$. Si W no usa a la arista e , entonces W es un uv -camino en $G - e$. De lo contrario, por la Proposición 1.6.2 podemos suponer que W es una trayectoria, y también podemos suponer sin pérdida de generalidad que e se recorre de y hacia x en G , por lo que proponemos al uv -camino W' en G de la siguiente forma

$$W' = WyCxW.$$

Claramente, W' no usa a la arista e , y es un uv -camino.

Recíprocamente, si e no es un puente, entonces, como x y y están en la misma componente conexas de G , existe un yx -camino en $G - e$. Se sigue de la Proposición 1.6.2 que existe una yx -trayectoria P en $G - e$. Claramente $xy \cup P$ es un ciclo en G que contiene a e . \square

1.7. Propiedades hereditarias

Si \mathcal{G} es el conjunto de todas las gráficas, entonces una **propiedad de las gráficas** es un subconjunto P de \mathcal{G} . Por ejemplo, la propiedad de “ser una gráfica bipartita” corresponde al conjunto de todas las gráficas bipartitas, que claramente es un subconjunto de \mathcal{G} . Una **propiedad hereditaria** es una propiedad P tal que si $G \in P$ entonces cualquier subgráfica inducida H de G también cumple $H \in P$. Por ejemplo, la propiedad “ser una gráfica completa” es una propiedad hereditaria, porque cualquier subgráfica inducida de una gráfica completa es, también, completa. La propiedad “ser una gráfica bipartita” también es una propiedad hereditaria, pues si G es bipartita con bipartición (X, Y) , entonces para cualquier subgráfica inducida H de G , se tiene que $(V_H \cap X, V_H \cap Y)$ es una bipartición de H . *propiedad hereditaria*

Si P es una propiedad hereditaria y $G \notin P$, entonces se cumple exactamente una de las siguientes afirmaciones, o bien existe un vértice v en G tal que $G - v \notin P$, o bien $G - v \in P$ para cada $v \in V_G$. Es decir, si G no cumple una propiedad hereditaria, o bien contiene una subgráfica inducida que tampoco la cumple, o todas sus subgráficas inducidas la cumplen. Una **obstrucción mínima** para una propiedad hereditaria P es una gráfica G que no tiene la propiedad, pero tal que cualquier subgráfica inducida sí la tiene. Resulta claro que toda gráfica que no cumple una propiedad hereditaria *obstrucción mínima*

P contiene una obstrucción mínima para P . Se sigue que si P es una propiedad hereditaria y \mathcal{O}_P es el conjunto de todas las obstrucciones mínimas para P , entonces una gráfica G tiene la propiedad P si y sólo si ninguna gráfica de \mathcal{O}_P es una subgráfica inducida de G .

La siguiente proposición es un ejemplo sencillo de caracterizaciones de propiedades hereditarias mediante un conjunto de obstrucciones mínimas. Claramente ser un cluster (definido en la Sección 1.2) es una propiedad hereditaria; recordemos que una gráfica es un cluster si cada componente conexa es una gráfica completa, por lo que cada subgráfica inducida también tiene esta propiedad. En otras palabras, en un cluster, dos vértices son adyacentes si y sólo si existe un camino entre ellos. Por lo tanto, en un cluster no puede existir una trayectoria inducida de longitud 2, pues sus extremos serían vértices que no cumplen la propiedad antes mencionada. Veamos que esta es la única obstrucción mínima para la propiedad de ser un cluster.

Proposición 1.7.1. *Una gráfica G es un cluster si y sólo si no contiene a P_3 como subgráfica inducida.*

Demostración. Procedamos por contrapositiva. Supongamos que $\{u, v, w\}$ es un subconjunto de V_G que induce una copia isomorfa de P_3 . Entonces, los vértices u, v y w están en la misma componente conexa, y en esa componente hay al menos un par de vértices no adyacentes, por lo que G no es un cluster.

Recíprocamente, si G no es un cluster, entonces alguna de sus componentes conexas, digamos C , no es una gráfica completa. Por lo tanto, existen vértices u y v en C tales que $uv \notin E_G$. Sea P una uv -trayectoria de longitud mínima en G , tal que $P = (u = x_0, \dots, x_k = v)$. Se sigue que $k \geq 2$, y por la elección de P , se tiene que $\{x_0, x_1, x_2\}$ induce a P_3 . \square

Posiblemente el resultado más conocido en esta dirección es la caracterización de las gráficas bipartitas con un conjunto de obstrucciones mínimas. La importancia de este resultado radica en que, aún cuando el conjunto de obstrucciones mínimas para caracterizar la propiedad de ser bipartita es infinito, estas obstrucciones mínimas conforman una familia suficientemente uniforme como para poder reconocer a sus elementos (y por lo tanto a las gráficas bipartitas) algorítmicamente. Se puede considerar un argumento de “propagación” para encontrar a una familia infinita de obstrucciones mínimas para la propiedad de ser bipartita. Supongamos que deseamos construir una bipartición (X, Y) de G . Podemos intentar construir la partición de G eligiendo un vértice arbitrario v , poniéndolo en el conjunto X , y propagando las decisiones que están forzadas por la bipartición, es decir, los vecinos de v tienen que estar en Y , los vecinos de los vecinos de v tienen que estar en X , etc. Si G contiene un ciclo impar C , con $C = (x_0, \dots, x_{2k}, x_0)$, entonces, al iniciar este proceso desde x_0 , obtendremos que $x_0 \in X$, lo que implica $x_1, x_{2k} \in Y$, de donde $x_2, x_{2k-1} \in X$, y este proceso continuará, dependiendo de k , hasta obtener $x_k, x_{k+1} \in X$ o $x_k, x_{k+1} \in Y$. En cualquiera de los dos casos, terminamos con dos vértices adyacentes en el mismo

conjunto, por lo que no es posible construir una bipartición de G . Por otro lado, si borramos cualquier vértice de un ciclo impar, obtenemos una trayectoria, que claramente es una gráfica bipartita. Por lo tanto, resulta claro que los ciclos impares son obstrucciones mínimas para la propiedad de ser bipartita. La pregunta natural es, ¿existen otras obstrucciones mínimas? El siguiente teorema confirma que los ciclos impares son las únicas obstrucciones mínimas para la propiedad de ser bipartita.

Teorema 1.7.2. *Sea G una gráfica. Son equivalentes:*

1. G es bipartita.
2. G no contiene ciclos impares.
3. G no tiene aristas múltiples y no contiene ciclos impares inducidos.

Demostración. Podemos suponer sin pérdida de generalidad que G es conexa. En caso de no serlo, es suficiente demostrar el resultado para cada componente conexa. En una dirección, esto es claro a partir de que un ciclo impar debe estar completamente contenido en una componente conexa. Para la otra dirección, basta observar que si C_1, \dots, C_k son las componentes conexas de G , y (X_i, Y_i) es una bipartición para C_i , con $i \in \{1, \dots, k\}$, entonces $(\bigcup_{i=1}^k X_i, \bigcup_{i=1}^k Y_i)$ es una bipartición de G . Demostremos primero que 1 es equivalente a 2.

Sea C un k -ciclo en una gráfica bipartita G , digamos $C = (x_1, \dots, x_k, x_1)$. Veamos que k es par. Sea (X, Y) la bipartición de G , y supongamos sin pérdida de generalidad que $x_1 \in X$. Utilicemos inducción matemática para demostrar que para cada $i \in \{1, \dots, k\}$, si i es par, entonces $x_i \in Y$ y si i es impar entonces $x_i \in X$. Por hipótesis, $x_1 \in X$, y como x_1 es adyacente a x_2 , tenemos que $x_2 \in Y$. Para $i > 1$, si i es par, entonces $i - 1$ es impar, y por hipótesis inductiva, $x_{i-1} \in X$, de donde $x_i \in Y$. Si i es impar, entonces $i - 1$ es par, y por hipótesis inductiva, $x_{i-1} \in Y$, por lo que $x_i \in X$, y concluye la inducción. Como x_k es adyacente a x_1 , y $x_1 \in X$, entonces necesariamente $x_k \in Y$, y al ser (X, Y) una partición, se sigue que $x_k \notin X$. De la contrapositiva de la afirmación que demostramos arriba por inducción concluimos que k no es impar, por lo que debe ser par.

Supongamos que G no contiene ciclos impares. Sea v un vértice en G , y sea X el subconjunto de V_G de todos los vértices que están a distancia par de v en G . Notemos que si Y es el conjunto de todos los vértices de G que están a distancia impar de v , entonces, ya que G es conexa, $X \cup Y = V_G$ y por definición de distancia, $X \cap Y = \emptyset$. Basta verificar que tanto X como Y son conjuntos independientes. Supongamos, procediendo por contradicción, que existen vértices $x, y \in X$ tales que $xy \in E_G$. Entonces, por la definición de X , existen una ux -trayectoria P , y una uy -trayectoria Q en G , ambas de longitud par. Claramente, R dado por $R = PxyQ^{-1}$, es un camino cerrado en G . Más aún, la longitud de R está dada por $\ell(R) = \ell(P) + 1 + \ell(Q)$, pero esta última expresión es un número impar, pues $\ell(P)$ y $\ell(Q)$ son pares por hipótesis. Así, G contiene un camino cerrado impar, lo que implica por la Proposición

1.6.3, que G contiene un ciclo impar, lo que resulta en una contradicción. Como la contradicción surgió de suponer que existe una arista con ambos extremos en X , concluimos que no existen aristas con ambos extremos en X , i.e., X es un conjunto estable. Análogamente puede demostrarse que Y es un conjunto independiente, lo que concluye la demostración.

Veamos ahora que 2 y 3 son equivalentes, verificando que sus contrapositivas lo son. Si G contiene un ciclo impar inducido, entonces contiene un ciclo impar. Recíprocamente, supongamos que G tiene un ciclo impar, y consideremos a C el ciclo impar más corto en G , supongamos que $C = (x_1, \dots, x_k, x_1)$. Queremos demostrar que C no tiene diagonales, por lo que procederemos por contradicción. Si C no es un ciclo inducido, y como G no tiene aristas múltiples, entonces existen $i, j \in \{1, \dots, k\}$ tales que $i + 1 < j$ y $v_i v_j \in E_G$. Dado que C es impar, necesariamente $v_i C v_j v_i$ o $v_j C v_i v_j$ es un ciclo impar, y por la elección de i y j , su longitud es estrictamente menor que la de C , lo que contradice la elección de C . \square

1.8. Operaciones en gráficas

1.8.1. Introducción

En muchas ramas de las matemáticas es común contar con operaciones que permitan la generación de nuevos objetos a partir de otros ya existentes. Es deseable además que el nuevo objeto creado comparta propiedades con aquellos objetos que fueron operados para crearlo. Esto resulta de particular interés en dos contextos diferentes.

El primero es cuando a partir de dos (o más) objetos dentro de nuestra teoría, podemos generar un nuevo objeto, que incluya como “factores” a cada uno de los operandos involucrados para obtenerlo. En este caso, se obtiene un objeto más complejo a partir de objetos más sencillos, y esto suele ser utilizado para obtener resultados de clasificación de aquellos objetos que pueden obtenerse a partir de factores primitivos. Un ejemplo claro de esto es la suma directa en grupos, y el Teorema Fundamental de Grupos Abelianos Finitamente Generados. Una ventaja adicional de este tipo de operaciones es que dan lugar al uso de inducción matemática para demostrar propiedades de los objetos que pueden crearse mediante la operación.

Por otro lado, tenemos operaciones 1-arias, que usualmente transforman un objeto con alguna regla bien definida, ya sea con la finalidad de obtener una estructura auxiliar que nos brinde información sobre alguna propiedad del objeto original, e.g., calcular el centralizador de un grupo, o con la finalidad de obtener un objeto que resulte “equivalente” al objeto original, y que nos permita considerar a ambos objetos como uno solo, con fines de clasificación.

En el primer tipo de operaciones, lo más común es considerar el producto cartesiano. En el caso de las gráficas, es natural definir al producto cartesiano de dos gráficas G y H como la gráfica con conjunto de vértices $V_G \times V_H$, el producto cartesiano

de los conjuntos de vértices de cada uno de los operandos. Pero, ¿cómo definimos al conjunto de aristas? No tiene mucho sentido definirlo como el producto cartesiano de los conjuntos de aristas, por lo que tenemos varias opciones para definirlo. Hay tres productos bien consolidados y conocidos en el ámbito de teoría de gráficas, el producto cartesiano, el producto débil o tensorial o categórico y el producto fuerte o normal.

1.8.2. Productos

Sean G y H gráficas. El **producto cartesiano** de G y H , denotado por $G \square H$, *producto cartesiano* es la gráfica cuyo conjunto de vértices es $V_G \times V_H$, el producto cartesiano de los conjuntos V_G y V_H , y tal que $(u_1, v_1)(u_2, v_2)$ está en las aristas de $G \square H$ si y sólo si $u_1 = u_2$ y $v_1v_2 \in E_H$, o $u_1u_2 \in E_G$ y $v_1 = v_2$. El **producto tensorial** de G y H , *producto tensorial* denotado por $G \times H$, es la gráfica cuyo conjunto de vértices es $V_G \times V_H$, el producto cartesiano de los conjuntos V_G y V_H , y tal que $(u_1, v_1)(u_2, v_2)$ está en las aristas de $G \times H$ si y sólo si $v_1v_2 \in E_H$ y $u_1u_2 \in E_G$. El **producto fuerte** de G y H , denotado *producto fuerte* por $G \boxtimes H$, es la gráfica cuyo conjunto de vértices es $V_G \times V_H$, el producto cartesiano de los conjuntos V_G y V_H , y tal que $(u_1, v_1)(u_2, v_2)$ está en las aristas de $G \boxtimes H$ si y sólo si $u_1 = u_2$ y $v_1v_2 \in E_H$, o $u_1u_2 \in E_G$ y $v_1 = v_2$, o $v_1v_2 \in E_H$ y $u_1u_2 \in E_G$. Una observación sencilla que se desprende de estas definiciones es que el conjunto de aristas del producto fuerte puede obtenerse al tomar la unión de los conjuntos de aristas del producto cartesiano y del producto tensorial.

En español existen nombres mnemotécnicos que ayudan a recordar cuál es cada uno de estos productos, basados en su comportamiento al operar K_2 consigo misma. Este mismo comportamiento se ve reflejado en la notación utilizada para cada uno de los productos. El producto cartesiano se conoce como el producto cuadro (\square), el producto tensorial se conoce como el producto tache (\times) y el producto fuerte como el producto tache-cuadro (\boxtimes). Vease la Figura 1.4 para una interpretación del por qué de estos nombres.

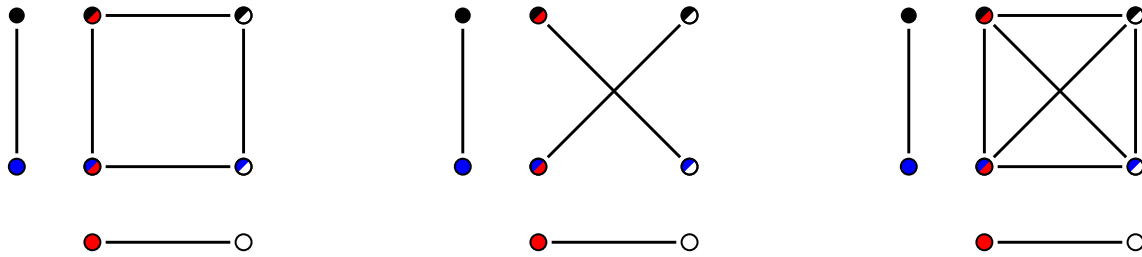


Figura 1.4: Las gráficas $K_2 \square K_2$, $K_2 \times K_2$ y $K_2 \boxtimes K_2$.

Nótese que el producto cartesiano es especialmente intuitivo, pues al considerar el producto $G \square H$ de dos gráficas G y H , para cada vértice $u \in V_G$ y cada vértice $v \in V_H$, se tiene que la subgráfica G_v de $G \square H$ inducida por todos los vértices que

tienen a v como segunda coordenada, es isomorfa a G , y la subgráfica H_u de $G \square H$ inducida por todos los vértices que tienen como primera coordenada a u , es isomorfa a H . De este modo, el producto cartesiano $G \square H$ tiene como subgráficas inducidas tantas copias de G como vértices tiene H y tantas copias de H como vértices tiene G .

rejilla de $m \times n$ Hay tres productos cartesianos que son de particular interés. La **rejilla de $m \times n$** es el producto cartesiano de las trayectorias P_m y P_n , es decir, $P_m \square P_n$. El **cilindro de $m \times n$** es el producto cartesiano de la trayectoria P_m y el ciclo C_n . La **rejilla toroidal de $m \times n$** es el producto cartesiano de los ciclos C_m y C_n .

1.8.3. Complemento

complemento Definimos el **complemento** de una gráfica G , denotado \overline{G} , como la gráfica con mismo conjunto de vértices que G , y tal que $E_{\overline{G}} = \binom{V_G}{2} - E_G$, o en otras palabras, para cualesquiera dos vértices distintos $u, v \in V_G$, se tiene que $uv \in E_{\overline{G}}$ si y sólo si $uv \notin E_G$.

autocomplementaria Una gráfica G es **autocomplementaria** si es isomorfa a su complemento, es decir, si $G \cong \overline{G}$. Por ejemplo, es fácil verificar que la gráfica P_4 es autocomplementaria, y es la gráfica autocomplementaria no trivial más pequeña que existe. En la Figura 1.5 se muestran dos gráficas y sus complementos.

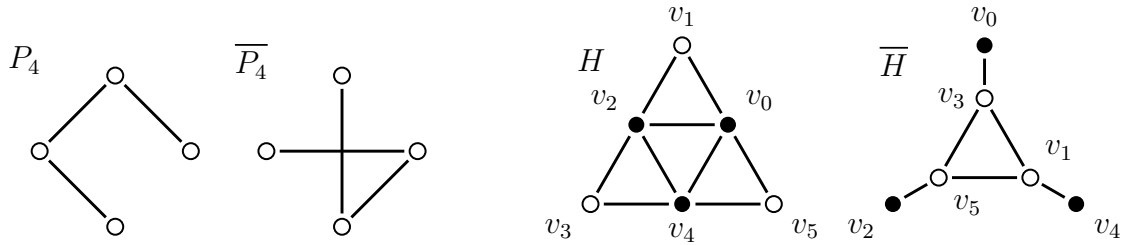


Figura 1.5: Dos gráficas y sus complementos.

Las siguientes dos propiedades, aunque son bastante sencillas de verificar, resultan sumamente útiles para trabajar con complementos.

Proposición 1.8.1. *Para cualesquiera gráficas G y H se cumplen las siguientes propiedades.*

1. $G \cong \overline{\overline{G}}$.
2. $G \cong H$ si y sólo si $\overline{G} \cong \overline{H}$.

Demostración. Para la primera propiedad, consideremos a la identidad de V_G , que es una biyección, y notemos que si u y v son vértices de G , entonces, por la definición de complemento de G , se tiene que $uv \in E_G$ si y sólo si $uv \notin E_{\overline{G}}$, y esto último sucede, por la definición de complemento de \overline{G} si y sólo si $uv \in E_{\overline{\overline{G}}}$.

Para la segunda propiedad, supongamos primero que $G \cong H$. Entonces, existe una biyección φ tal que $\varphi: V_G \rightarrow V_H$ y tal que, para cualesquiera dos vértices u y v de G , se tiene $uv \in V_G$ si y sólo si $\varphi(u)\varphi(v) \in V_H$. Proponemos a la misma biyección como un isomorfismo entre \overline{G} y \overline{H} . Notemos que, por definición de complemento y por la contrapositiva de la propiedad de φ mencionada arriba, $uv \in V_{\overline{G}}$ si y sólo si $uv \notin V_G$, lo que sucede si y sólo si $uv \notin V_H$, y finalmente esto pasa si y sólo si $uv \in V_{\overline{H}}$. Se sigue del último argumento que $\overline{G} \cong \overline{H}$.

La implicación restante se demuestra de forma análoga. \square

1.8.4. Otras operaciones binarias

Naturalmente, las gráficas heredan algunas operaciones de la teoría de conjuntos. Las más obvias son la unión y la intersección. Si G y H son gráficas, la **unión** de G y H , denotada $G \cup H$ es la gráfica cuyo conjunto de vértices es $V_G \cup V_H$, y cuyo conjunto de aristas es $E_G \cup E_H$. La **intersección** de G y H , denotada $G \cap H$, es la gráfica cuyo conjunto de vértices es $V_G \cap V_H$, y cuyo conjunto de aristas es $E_G \cap E_H$.

Un caso especial de la unión es la **unión ajena**, que es una unión de gráficas *unión ajena* en la que además $V_G \cap V_H = \emptyset$, y se denota por $G + H$. Hay autores que utilizan la notación conjuntista clásica para la unión ajena, sin embargo, la notación aditiva tiene la enorme ventaja de tener generalizaciones convencionales para más de dos uniendos. Por ejemplo, al tomar la unión de k copias de la misma gráfica G , podemos escribir $\sum_{i=1}^k G$, o bien, de manera mucho más natural, kG . Del mismo modo, la unión ajena de una familia de gráficas G_1, \dots, G_k , se denota por $\sum_{i=1}^k G_i$.

Otra operación que resulta de gran utilidad es el amalgamamiento o supremo de dos gráficas ajenas G y H . Si G y H son gráficas tales que $V_G \cap V_H = \emptyset$, el **amalgamamiento** de G y H , denotado $G \oplus H$, es la gráfica cuyo conjunto de *amalgamamiento* vértices es $V_G \cup V_H$, y cuyo conjunto de aristas es $E_G \cup E_H \cup \{uv: u \in V_G, v \in V_H\}$. En otras palabras, es la gráfica obtenida de $G + H$ al hacer adyacente a cada vértice de G con cada vértice de H .

1.9. Digráficas

Las gráficas no tienen suficiente poder expresivo para modelar algunos problemas, por ejemplo, si se desea modelar las calles de una ciudad con una gráfica, poniendo un vértice por cada intersección de dos o más calles, y una arista entre dos intersecciones si hay una calle que lleva de una a la otra, entonces todas las calles en el modelo serían de doble sentido. Es necesaria una estructura que permita distinguir, para cada arista, el inicio del final de la misma. En esta sección se introducirán las gráficas dirigidas o digráficas; muchas propiedades de las digráficas se pueden generalizar naturalmente a partir de las definiciones correspondientes para gráficas, por lo que se cubrirán con poco detalle.

Una **digráfica** D es una pareja ordenada $D = (V_D, A_D)$, donde V_D es un conjunto, y A_D es un subconjunto (posiblemente vacío) de $(V_D \times V_D) - L$, donde $L = \{(v, v) : v \in V_D\}$. Cuando no haya posibilidad de confusión, se omitirá el subíndice D , para escribir únicamente $D = (V, A)$. Análogamente al caso de las gráficas, los elementos de V serán llamados **vértices**, pero en este caso, los elementos de A serán llamados **flechas**. Una digráfica es **vacía** si su conjunto de flechas es vacío.

Para vértices u y v en la digráfica D , si $a = (u, v) \in A$, se escribirá $u \rightarrow v$, y se dirá que u **domina** a v , que v **absorbe** a u , y que u y v son **adyacentes**. También se dirá que u y v son la **cola** y la **cabeza** de a , respectivamente. Se dice que una flecha (u, v) es **simétrica** si $v \rightarrow u$, y **asimétrica** en otro caso. La **exvecindad** $N_D^+(v)$ del vértice v se define como $N_D^+(v) = \{x \in V : v \rightarrow x\}$; el **exgrado** $d_D^+(v)$ del vértice v es el número de flechas de las cuáles es cola v , es decir, $d_D^+(v) = |N_D^+(v)|$; el mínimo de los exgrados de D se denota por δ_D^+ , y el máximo de los exgrados por Δ_D^+ . De manera análoga se puede definir la **invecindad** $N_D^-(v)$ y el **ingrado** $d_D^-(v)$ del vértice v , así como δ_D^- y Δ_D^- . Una digráfica D es **k -regular** si $d^+(v) = d^-(v) = k$ para cada $v \in V$; una digráfica es **regular** si es k -regular para algún natural k .

Si D es una digráfica, es posible asociarle una gráfica, llamada la **gráfica subyacente** de D , denotada por U_D , y que se define por $V_{U_D} = V_D$, y $uv \in E_{U_D}$ si y sólo si $(u, v) \in A_D$ o $(v, u) \in A_D$.

Dos digráficas D y H son **iguales** o **idénticas** (denotado $D = H$) si $V_D = V_H$ y $A_D = A_H$. Se dice que D y H son **isomorfas**, denotado $D \cong H$, si existe una biyección $\varphi : V_D \rightarrow V_H$ tal que para cualesquiera vértices $u, v \in V_D$, se tiene que $u \rightarrow v$ si y sólo si $\varphi(u) \rightarrow \varphi(v)$; la biyección φ es un **isomorfismo** entre D y H . Un **automorfismo** es un isomorfismo $\varphi : D \rightarrow D$.

A cualquier digráfica D con n vértices le corresponde una matriz de $n \times n$, $A(D)$, llamada la **matriz de adyacencia** de D , con $A(D) = (a_{ij})$, donde

$$a_{ij} = \begin{cases} 1 & \text{si } v_i \rightarrow v_j, \\ 0 & \text{si no.} \end{cases}$$

Existen también matrices de incidencia para digráficas, que pueden definirse de manera análoga que para las gráficas.

Un **camino** en una digráfica D es una sucesión finita, no vacía $W = v_0 a_1 v_1 a_2 \cdots a_k v_k$ cuyos elementos alternan entre vértices y flechas, y tal que $a_i = (v_{i-1}, v_i)$ para cada $1 \leq i \leq k$. El vértice v_0 es llamado el **origen**, **inicio**, o **vértice inicial** de W , y el vértice v_k es el **final** o **vértice terminal** de W ; nos referiremos a v_0 y a v_k como los **extremos** de W , y a v_1, \dots, v_{k-1} como sus **vértices internos**. Decimos que W es un **$v_0 v_k$ -camino**. El entero no negativo k es la **longitud** de W , y se denota por $\ell(W)$.

Si $W = v_0 a_1 v_1 \cdots a_k v_k$ y $W' = v_k a_{k+1} v_{k+1} \cdots a_l v_l$ son caminos, entonces el camino $v_0 a_1 v_1 \cdots a_l v_l$ obtenido al concatenar W y W' se denota por WW' . Si todas las flechas de W son simétricas, el camino $v_k a_k v_{k-1} \cdots a_1 v_0$, obtenido al invertir el orden de W ,

se denota por W^{-1} . Una **sección** del camino W es un camino que es una subsucesión $v_i a_{i+1} v_{i+1} \cdots a_j v_j$ de elementos consecutivos de W , la denotaremos por $v_i W v_j$.

En una digráfica sin flechas paralelas, un camino $v_0 a_1 v_1 \cdots a_k v_k$ está determinado por la sucesión $v_0 v_1 \cdots v_k$ de sus vértices; de este modo, un camino en una digráfica sin flechas múltiples puede especificarse simplemente por su sucesión de vértices.

Si las flechas a_1, \dots, a_k de un camino W son distintas, W es llamado un **paseo**; en este caso, la longitud de W es igual a su número de flechas. Si además, los vértices v_0, \dots, v_k son distintos, W es una **trayectoria**. Un camino W es **cerrado** si su longitud es positiva y su vértice inicial coincide con su vértice final. Un paseo cerrado cuyo origen y vértices internos son distintos es un **ciclo**. El **cuello** de una digráfica es la longitud del ciclo más corto contenido en D ; si la digráfica no tiene ciclos definimos su cuello como infinito. La **circunferencia** de D es la longitud del ciclo más largo contenido en D ; si la digráfica no tiene ciclos definimos su circunferencia como 0.

Dos caminos W_1 y W_2 son **ajenos** o **ajenos por vértices** si $V_{W_1} \cap V_{W_2} = \emptyset$. Si W_1 y W_2 son xy -caminos para vértices x, y en alguna gráfica G , decimos que W_1 y W_2 son **internamente ajenos** si $V_{W_1} \cap V_{W_2} = \{x, y\}$. Análogamente, W_1 y W_2 son ajenos por flechas si $A_{W_1} \cap A_{W_2} = \emptyset$.

Un **paseo euleriano** en la digráfica D es un paseo que utiliza todas las flechas de D . Si un paseo euleriano además es cerrado, decimos que es un **circuito euleriano**. Una digráfica es **euleriana** si tiene un circuito euleriano. Un ciclo generador será llamado **ciclo hamiltoniano**, y una trayectoria generadora, **trayectoria hamiltoniana**.

La **distancia** del vértice u al vértice v en la digráfica D , denotada como $d_D(u, v)$, se define como $\min \{\ell(P) : P \text{ es una } uv\text{-trayectoria en } D\}$, si existe al menos una uv -trayectoria en D , y como infinito en otro caso. El **diámetro** de la digráfica D se define como $\max \{d(u, v) : u, v \in V_D\}$.

Una trayectoria dirigida es una digráfica para la que su conjunto de vértices admite un orden lineal de tal forma que cada vértice tiene una flecha únicamente a su sucesor inmediato en el orden. Un ciclo es una digráfica de al menos dos vértices en la que su conjunto de vértices admite un orden cíclico de tal forma que cada vértice tiene una flecha hacia su sucesor inmediato en el orden. Es fácil observar que una trayectoria dirigida con k vértices es única salvo isomorfismo, por lo tanto, podemos denotar a ésta por \vec{P}_k ; nótese que la longitud de \vec{P}_k es $k - 1$. Análogamente, el ciclo de k vértices se denotará por \vec{C}_k .

Una digráfica D es **fuertemente conexa**, o simplemente **fuerte** si para cualesquiera dos vértices u y v en V , se tiene la existencia de al menos un uv -camino. Las **componentes fuertemente conexas**, o **componentes fuertes** de D son las subgráficas maximas por contención con la propiedad de ser fuertemente conexas. Si D es una digráfica con conjunto de componentes fuertemente conexas S_1, \dots, S_k , la **condensación** de D , denotada por D^* , es la digráfica con conjunto de vértices v_1, \dots, v_k donde $v_i \rightarrow v_j$ si y sólo si existe una $S_i S_j$ -flecha en D .

Capítulo 2

Árboles

2.1. Introducción

Consideremos el problema de conectar una red. En muchas ocasiones es necesario conectar un conjunto de objetos mediante vías de comunicación, por ejemplo, computadoras mediante cable, poblaciones mediante carreteras, ferrocarriles, línea telefónica o fibra óptica, etc. Usualmente, el diseño de esta conexión implica que cualquier punto en la red pueda alcanzar a cualquier otro, pues lo que se busca que cualesquiera dos estén comunicados. Una solución obvia es conectar cualesquiera dos puntos de nuestra red, sin embargo, considerando que cada una de estas conexiones tiene un costo (e.g., el costo del cable o de la construcción de la carretera), esta solución puede ser inviable en la realidad. De hecho, en muchas ocasiones se trabaja con presupuesto limitado, por lo que se busca la solución que conecte a todos los puntos de la red con un costo mínimo. Otra restricción es que algunas conexiones pueden ni siquiera ser construidas en la realidad (por ejemplo, si algún accidente geográfico como un río o una cordillera separan a dos ciudades). Bajo estas condiciones, el problema se puede modelar fácilmente con una gráfica.

Es posible construir una gráfica donde cada vértice es uno de los puntos que se quieren comunicar en la red, y cada arista es una vía de comunicación que es factible construir. En la realidad, cada una de las aristas debería de tener un costo posiblemente distinto (el costo de construir la vía de comunicación correspondiente), sin embargo, en una primera aproximación al problema, supongamos que todas tienen el mismo costo; más adelante se atenderá el caso general. El problema es determinar cuáles de estas vías de comunicación debemos construir para conectar a toda la red, optimizando el costo de la construcción. Claro que, en una primera instancia, es necesario saber si es posible conectar todos los puntos de la red. Convenientemente, es posible determinar si una solución al problema existe, y encontrar dicha solución, al mismo tiempo.

Desde el punto de vista de la teoría de gráficas, se tienen entonces los siguientes problemas. Primero, dada una gráfica G , determinar si G es conexa. Sabiendo que

una gráfica G es conexa, encontrar una subgráfica generadora mínima por contención con la propiedad de ser conexa. Si G es una gráfica conexa donde cada arista tiene un costo positivo, encontrar una subgráfica generadora que sea conexa y tal que la suma de los costos de sus aristas sea el mínimo posible.

Usando la caracterización de conexidad dada por la Proposición 1.6.4, una posible solución al primer problema es encontrar un camino entre cualesquiera dos vértices de la gráfica. Nótese que, para este fin, basta con encontrar un camino desde un vértice de la gráfica hacia cualquier otro. ¿Cómo sería una solución al segundo problema? Podemos intentar encontrar una solución con el siguiente procedimiento. Se elige una arista de la gráfica conexa G , si al borrarla la gráfica se desconecta, entonces se marca esta arista como explorada, y se repite el proceso eligiendo otra arista, hasta que se verifique que ninguna arista puede borrarse sin desconectar a la gráfica, o hasta que se encuentre una que sí es posible borrar. En el primer caso, la gráfica G es conexa, y no contiene una subgráfica generadora propia que también lo sea, por lo que es una solución. En el segundo caso, se encontró una arista e tal que $G - e$ es conexa, por lo que es posible quitar la marca a todas las aristas, e iniciar nuevamente el proceso. Al final, se obtendrá una gráfica en la que cada arista es un puente, que, como se observó arriba, es una solución.

Entonces, las gráficas conexas en las que cada arista es un puente son gráficas que resultan de interés, pues, como se observó con anterioridad, no contienen subgráficas generadoras propias que sean conexas, o en otras palabras, son mínimas por contención con la propiedad de ser conexas. Ya que cada arista es un puente, el Lema 1.6.5 nos dice que estas gráficas no tienen ciclos (de lo contrario, las aristas del ciclo no serían puentes), por lo que en particular, no contienen lazos ni aristas múltiples. Recíprocamente, y nuevamente por el Lema 1.6.5, una gráfica conexa en la que no hay ciclos, tiene la propiedad de que cada arista es un puente. La propiedad de no tener ciclos es mucho más fácil de enunciar que la propiedad de que cada arista sea un puente; también es más fácil encontrar un ciclo que verificar si una arista es un puente. Por este motivo, esta es la propiedad con la que se suele definir a esta familia de gráficas.

árbol Un **árbol** es una gráfica conexa y acíclica. Como ya se observó, en un árbol cada arista es un puente. ¿Qué más se puede decir acerca de estas gráficas? En realidad es posible decir bastante acerca de su estructura. En la siguiente sección se explorarán las propiedades básicas de los árboles. Para volver al problema con el que se motivó el estudio de los árboles, el objetivo inicial será demostrar que toda gráfica conexa

árbol generador contiene una subgráfica generadora que sea un árbol, es decir, un **árbol generador**, por lo que el problema de conectar una red tendrá una solución siempre que la gráfica inicial sea conexa.

2.2. Caracterizaciones

El hecho de que los árboles sean gráficas conexas mínimas tiene como consecuencia que la forma de llegar de un vértice a otro en un árbol esté bastante restringida. De hecho, en un árbol, entre cualesquiera dos vértices hay una única trayectoria. Esta propiedad resulta deseable, pues, por ejemplo, permite calcular la distancia entre cualesquiera dos vértices de forma sumamente eficiente. El siguiente lema enuncia formalmente esta propiedad.

Lema 2.2.1. *Sea G una gráfica y sean u y v vértices de G . Si existen dos uv -trayectorias distintas en G , entonces su unión contiene un ciclo.*

Demostración. Sean P y Q uv -trayectorias en G , tales que $P = (x_0, \dots, x_k)$ y $Q = (y_0, \dots, y_l)$. El resultado se sigue trivialmente si en la unión de P y Q hay alguna arista múltiple. De otro modo, como P y Q son distintas, existe un primer vértice en el que no coinciden. Sea i el primer índice tal que $x_i \neq y_i$; como $x_0 = y_0 = u$, se tiene que $i \geq 1$. Ya que $x_{i-1} = y_{i-1}$, y ambas P y Q son trayectorias, tenemos que la arista $e = x_{i-1}x_i$ no es utilizada en Q . Notemos que $x_i P x_k Q^{-1} x_0 P x_{i-1}$ es un $x_i x_{i-1}$ -camino contenido en $(P \cup Q) - e$, y por la Proposición 1.6.2, hay una $x_i x_{i-1}$ -trayectoria R contenida en $(P \cup Q) - e$. Claramente $R + e$ es un ciclo contenido en $P \cup Q$. \square

Recíprocamente, es bastante fácil observar que si una gráfica sin lazos contiene un ciclo, entonces para al menos un par de vértices distintos, existen dos trayectorias distintas que los unen. De este modo, se tiene una caracterización adicional para los árboles. En el siguiente teorema se incluye esta caracterización, así como la que se discutió en la sección anterior.

Teorema 2.2.2. *Sea G una gráfica. Los siguientes enunciados son equivalentes.*

1. G es un árbol.
2. G es conexa y cada arista de G es un puente.
3. G no tiene lazos, y entre cualesquiera dos vértices de G existe una única trayectoria.

Demostración. Si G es un árbol, entonces es acíclica por definición. Por la Proposición 1.6.5, se tiene que cada arista de G es un puente.

Para 2 implica 3, consideremos la contrapositiva. Si G tiene un lazo e , entonces e es una arista que no es un puente. Si existe un par de vértices en G entre los que no hay una trayectoria, entonces G no es conexa. Finalmente, si existe un par de vértices en G entre los que existe más de una trayectoria, entonces, por el Lema 2.2.1, G contiene un ciclo, y cada arista del ciclo cumple con no ser un puente.

Para 3 implica 1, nuevamente demostraremos la contrapositiva. Si G no es un árbol, entonces no es conexa o contiene algún ciclo. Si no es conexa, existe un par

de vértices entre los que no existe una trayectoria. Si G contiene un ciclo, éste puede ser un lazo, o de lo contrario, tiene longitud al menos 2, si hay aristas múltiples, o 3, si G es simple. En cualquier caso, para cualesquiera dos vértices del ciclo existen al menos dos trayectorias distintas que los unen en G , a saber, las geodésicas que los unen en el ciclo. \square

El teorema anterior es la única herramienta que se necesita para demostrar que toda gráfica conexa contiene un árbol generador. Aunque se podría escribir una demostración siguiendo un camino muy parecido al descrito en la sección anterior (borrar una a una las aristas hasta encontrar los puentes de la gráfica), es posible considerar directamente el caso extremo, para lo cual, es útil responder a las siguientes preguntas ¿cómo debe ser la gráfica cuando el proceso termina?, ¿qué tienen en común las gráficas que se obtienen en los pasos intermedios del proceso? Todas las gráficas que aparecen en algún punto del proceso son subgráficas generadoras conexas de la gráfica original, y la gráfica con la que termina el proceso es mínima por contención con esta propiedad. Esta es la información que será utilizada para demostrar la existencia de un árbol generador.

Corolario 2.2.3. *Si G es una gráfica conexa, entonces G contiene un árbol generador.*

Demostración. Sea \mathcal{C} el conjunto de todas las subgráficas generadoras y conexas de G . Claramente \mathcal{C} es no vacío (pues G es uno de sus elementos), y además es un orden parcial con la relación “ser subgráfica de”. Sea H un elemento mínimo (local) con este orden en \mathcal{C} , que debe existir pues \mathcal{C} es un conjunto finito. Por la definición de \mathcal{C} , tenemos que H es generadora y conexa. Afirmamos que H es el árbol generador de G que buscamos. Para concluir la demostración, veamos que cada arista de H es un puente.

Sea e una arista de H . Como H es mínima en \mathcal{C} bajo la relación de subgráfica, y dado que $H - e$ es una subgráfica propia de H , concluimos que $H - e$ no es un elemento de \mathcal{C} . Como $H - e$ sí es una subgráfica generadora de G , entonces $H - e$ no es conexa, con lo que e es un puente de H . Como e fue elegida arbitrariamente, se sigue que toda arista de H es un puente, y por tanto H es un árbol generador de G . \square

Contando con la caracterización dada por el Teorema 2.2.2, resulta de interés tener información adicional sobre los puentes en una gráfica. En particular, al eliminar un puente de una gráfica, su número de componentes conexas no puede crecer mucho, como sí sucede, por ejemplo, al borrar el vértice central de una estrella.

Proposición 2.2.4. *Sea G una gráfica y sea e un puente de G . El número de componentes conexas de $G - e$ es igual al número de componentes conexas de G más uno.*

La Proposición 2.2.4 es de gran utilidad en el contexto de los árboles, pues al borrar cualquier arista de un árbol, se obtendrán dos componentes conexas. Como cualquier subgráfica de una gráfica acíclica es también acíclica, las dos componentes conexas obtenidas serán árboles, de orden y tamaño estrictamente menor que el árbol original. Este hecho nos permite utilizar fácilmente inducción sobre el número de vértices o de aristas para demostrar propiedades en árboles. La idea general es sencilla, para demostrar que todo árbol tiene la propiedad P , para el paso inductivo, se considera un árbol no trivial arbitrario T , se borra cualquier arista, y se obtienen dos árboles T_1 y T_2 , a los que se les puede aplicar la hipótesis inductiva. Sabiendo que T_1 y T_2 tienen la propiedad P , es posible unirlos nuevamente con la arista eliminada, para verificar que T también tiene la propiedad P . La demostración de la Proposición 2.2.5 ilustra esta técnica.

En la sección anterior, se planteó una posible solución al problema de conectar los nodos en una red. La idea era borrar todas las aristas que no desconectasen a la gráfica, hasta terminar con una gráfica en la que todas las aristas son puentes. Como queremos optimizar el número de aristas utilizadas en la gráfica resultante, la siguiente pregunta surge de forma natural, ¿El orden en el que se borran las aristas puede afectar el tamaño de la gráfica obtenida al terminar el proceso? En el caso de que la respuesta a esta pregunta fuese afirmativo, se tendría que para un número de vértices fijo hay árboles con distinto número de aristas, lo que volvería el problema de optimización más difícil, pues se tendría que garantizar que el árbol encontrado es óptimo entre todos los posibles. Por suerte, no es el caso.

Proposición 2.2.5. *Para todo árbol se cumple*

$$|E| = |V| - 1.$$

Demostración. Sea T un árbol. Procederemos por inducción sobre $|V|$. Si $|V| = 1$, entonces T es el árbol trivial, que no tiene aristas. Así, $|E| = 0 = |V| - 1$.

Supongamos que $|V| > 1$. Al ser conexo, debe existir al menos una arista e en T . Por el Teorema 2.2.2, e es un puente de T , y por la Proposición 2.2.4, $T - e$ tiene exactamente dos componentes conexas, T_1 y T_2 . Como las subgráficas de una gráfica acíclica también son acíclicas, T_i es una gráfica conexa y acíclica, para $i \in \{1, 2\}$, es decir, un árbol. Además, el orden de T_i es estrictamente menor que el de T , por lo que, por hipótesis inductiva, $|E_{T_i}| = |V_{T_i}| - 1$, para cada $i \in \{1, 2\}$. De las siguientes igualdades

$$|E_T| = |E_{T_1}| + |E_{T_2}| + 1,$$

$$|V_T| = |V_{T_1}| + |V_{T_2}|,$$

que se siguen directamente de la definición de T_1 y T_2 , se obtiene que

$$\begin{aligned} V_T &= |V_{T_1}| + |V_{T_2}| \\ &= (|E_{T_1}| + 1) + (|E_{T_2}| + 1) \\ &= (|E_{T_1}| + |E_{T_2}| + 1) + 1 \\ &= |E_T| + 1, \end{aligned}$$

con lo que concluye la demostración. \square

La Proposición 2.2.5 es importante por diversos motivos. En primer lugar, garantiza que para resolver el problema planteado en la introducción de este capítulo, donde se desea conectar una red minimizando el costo (cuando todas las aristas tienen el mismo costo), basta encontrar cualquier árbol generador, pues todos los árboles generadores de una gráfica conexa tendrán el mismo número de aristas y cualquiera de ellos será una solución al problema antes mencionado.

Por otro lado, saber que el número de aristas de una gráfica es una función lineal del número de vértices de la misma resulta útil al analizar el tiempo de ejecución de cualquier algoritmo que se ejecute sobre la misma, pues tenemos que $O(|E|) = O(|V|)$. Es común encontrar algoritmos que resuelven problemas en tiempo $O(|V| + |E|)$ para gráficas arbitrarias; en el caso antes mencionado, esta expresión se reduce a $O(|V|)$. Así, es común encontrar algoritmos que resuelven problemas interesantes en árboles, y cuyo tiempo de ejecución es $O(|V|)$, lo que suele ser óptimo, pues muchos algoritmos al menos tienen que procesar todos los vértices de una gráfica.

En tercer lugar, como los árboles son gráficas mínimas, respecto al número de aristas, con la propiedad de ser conexas, tenemos que, al menos de manera intuitiva (formalizaremos esta idea más adelante), cualquier gráfica con menos de $|V| - 1$ aristas es inconexa. Por otro lado, un ejercicio de la tarea afirma que cualquier gráfica con al menos $|V|$ aristas contiene un ciclo. Así, $|V| - 1$ es tanto el máximo número de aristas que puede tener una gráfica acíclica, como el mínimo número de aristas que puede tener una gráfica conexa. Si se consideran las tres condiciones, conexidad, aciclicidad y $|E| = |V| - 1$, la Proposición 2.2.5 dice que las dos primeras implican a la tercera. Resulta sorprendente que un resultado más fuerte se cumple, cualesquiera dos de estas condiciones implican a la tercera.

Proposición 2.2.6. *Sea G una gráfica. Si G satisface cualesquiera dos de las siguientes tres propiedades, entonces también se satisface la tercera, por lo que G es un árbol.*

1. G es conexa.
2. G es acíclica.
3. $|E| = |V| - 1$.

Demostración. La Proposición 2.2.5 implica que 3 se sigue de suponer 1 y 2. Demostraremos la contrapositiva de las dos implicaciones restantes.

Supongamos que G no es conexa, y sean C_1, \dots, C_k las componentes conexas de G ; notemos que $k \geq 2$. Si G es acíclica, entonces cada componente de G es un árbol, por lo que el número de aristas de G es

$$\begin{aligned} \sum_{i=1}^k |E_{C_i}| &= \sum_{i=1}^k (|V_{C_i}| - 1) \\ &= \sum_{i=1}^k |V_{C_i}| - \sum_{i=1}^k 1 \\ &= |V_G| - k \\ &< |V_G| - 1, \end{aligned}$$

con lo que 2 y 3 implica 1 queda demostrado.

Para la implicación restante, supongamos que G tiene un ciclo C . Si G es conexa, entonces se sigue del Corolario 2.2.3 que contiene un árbol generador T , y por la Proposición 2.2.5, T tiene $|V| - 1$ aristas. Como G tiene al menos un ciclo, pero T no, entonces G tiene más aristas que T , es decir $|E| \geq |V|$, por lo que $|E| \neq |V| - 1$. \square

2.3. Vértices de corte

En la sección anterior se analizaron algunas propiedades estructurales de los árboles. En particular, se demostró que cada arista de un árbol es un puente. ¿Es posible hacer una afirmación similar para los vértices de un árbol? Una **hoja** en un árbol es un vértice de grado 1, y un **vértice soporte**, o simplemente **soporte** es un vértice adyacente a una hoja. El resultado principal de esta sección afirma que los únicos vértices en un árbol que no son vértices de corte son sus hojas. Adicionalmente, se demostrará que todo árbol tiene al menos dos hojas, y como consecuencia del Corolario 2.2.3, se obtendrá que toda gráfica conexa tiene al menos dos vértices que no son de corte.

Proposición 2.3.1. *Sea G un árbol no trivial. Si v es un vértice de G , entonces v es de corte si y sólo si v no es una hoja de G .*

Demostración. Procediendo por contrapositiva, supongamos que v es una hoja de G . Como G es acíclica, también $G - v$ es acíclica. Además, resulta claro que $G - v$ tiene un vértice menos y una arista menos que G , por lo que se cumplen las condiciones 2 y 3 de la Proposición 2.2.6, de donde se sigue que $G - v$ es conexa. En otras palabras, v no es un vértice de corte.

Para la implicación restante, supongamos que v no es una hoja de G . Como G no es trivial, no tiene vértices aislados, por lo que v debe tener al menos dos vecinos

u y w . Según el Teorema 2.2.2, la única uw -trayectoria en G es (u, v, w) , por lo que no existen uw -trayectorias en $G - v$. Luego, v es un vértice de corte en G . \square

Como se discutió arriba, el resultado anterior puede combinarse con algún resultado que garantice la existencia de hojas en un árbol para garantizar la existencia de vértices que no son de corte. Es interesante comparar la demostración del siguiente resultado con la de la Proposición 1.6.1.

Proposición 2.3.2. *Sea G un árbol no trivial. Si P es una trayectoria en G que no puede ser extendida, entonces los extremos de P son hojas de G .*

En particular, todo árbol contiene al menos dos hojas.

Demostración. Supongamos que $P = (x_0, \dots, x_k)$. Como G es acíclica, resulta claro que x_0 no tiene vecinos en P además de x_1 , pues de lo contrario se formaría un ciclo. Por otro lado, x_0 tampoco puede tener vecinos fuera de P , pues en ese caso sería posible extender a P para formar una trayectoria más larga. Luego, el único vecino de x_0 en G es x_1 , por lo que x_0 es una hoja de G . Análogamente x_k también es una hoja de G .

Para la segunda afirmación, basta considerar una trayectoria de longitud máxima en G . Al ser de longitud máxima, ésta no puede ser extendida, por lo que sus extremos son hojas. \square

Resulta claro que una trayectoria no trivial es un árbol con exactamente dos hojas, por lo que la Proposición 2.3.2 plantea una cota justa respecto al número de hojas que un árbol debe tener. Sin embargo, es fácil relacionar la existencia de hojas adicionales en un árbol con el grado máximo del mismo; algunos ejercicios de esta sección exploran dicha relación.

Corolario 2.3.3. *Sea G una gráfica. Si G es conexa, entonces:*

1. *Si G no es trivial, entonces tiene al menos dos vértices que no son de corte,*
2. *G tiene al menos $|V| - 1$ aristas.*

Demostración. Si G es conexa, entonces contiene un árbol generador T . Como T tiene $|V| - 1$ aristas, se sigue el segundo punto del corolario. Por otro lado, si v es una hoja de T , entonces $T - v$ es conexa, y $T - v$ es una subgráfica generadora de $G - v$. Luego, $G - v$ también es conexa. Así, las hojas de T corresponden a vértices que no son de corte en G . El primer punto del corolario se sigue de la Proposición 2.3.2. \square

2.4. Búsqueda en gráficas

En la Sección 2.2 se demostró que toda gráfica conexa tiene un árbol generador, pero ¿cómo encontramos uno? Responder a esta pregunta de forma puramente teórica

es bastante sencillo, basta construir todos los subconjuntos de $|V| - 1$ aristas en una gráfica, y probar cada uno de ellos hasta dar con un árbol. Este procedimiento tiene algunos problemas, el primero de ellos es que resulta poco eficiente. El número de subconjuntos que tenemos que verificar es $\binom{|E|}{|V|-1}$, por lo que si el número de vértices es, por ejemplo, aproximadamente la mitad del número de aristas, se tendrían que verificar alrededor de $|E|^{\frac{|E|}{2}}$ subconjuntos. Además, este “procedimiento” no resuelve el problema de cómo verificar que una gráfica es conexa; para determinar si uno de los posibles subconjuntos de aristas antes mencionados corresponde o no a un árbol generador, es necesario determinar si induce una gráfica conexa, lo que genera un nuevo problema.

La forma de resolver el problema de encontrar un árbol generador, que se presentará en esta sección, apela a algoritmos clásicos de búsqueda en gráficas. La entrada de un algoritmo de búsqueda suele ser una gráfica G con un vértice distinguido r , donde se iniciará la búsqueda, y una función de verificación φ , con $\varphi: V \rightarrow \{0, 1\}$. Intuitivamente, la función φ tiene la información suficiente para determinar si un vértice dado de la gráfica es el vértice que estamos buscando (o uno de los vértices que estamos buscando). Así, la finalidad del algoritmo será recorrer los vértices de la gráfica, evaluando la función φ en cada vértice encontrado, y deteniéndose si en algún momento se encuentra algún vértice v con $\varphi(v) = 1$, o bien, si todos los vértices de la gráfica son recorridos sin encontrar alguno con dicha propiedad.

Aun cuando los algoritmos que se estudiarán en esta sección son referidos como “algoritmos de búsqueda”, se presentará una variante de los mismos cuya finalidad no es buscar un vértice específico de la gráfica, sino recorrer todos los vértices de la gráfica para construir un árbol generador de la misma. Sin embargo, es fácil modificar estos algoritmos para devolverlos a su función original de búsqueda, idea que se explora en algunos ejercicios de la sección.

El primer algoritmo considerado es la versión más general del algoritmo de búsqueda en gráficas, de la que se desprenden cuatro algoritmos que serán estudiados en este capítulo. El Algoritmo 1 construye un árbol generador para una gráfica G , empezando en un vértice distinguido r . En principio, el árbol obtenido no tiene propiedades especiales, y no tenemos estructuras de datos previamente determinadas. La idea central es empezar con un árbol de un solo vértice, que se hará crecer pegándole hojas de una en una. Como no es posible generar ciclos al agregar una hoja a una gráfica acíclica, es fácil garantizar que la gráfica obtenida al final del proceso será acíclica. La gráfica con la que se inicia es conexa, y en cada paso se le agregará una hoja para hacerla crecer, por lo que en cada paso este proceso generará un árbol que usa un vértice más que el anterior.

La forma de realizar el proceso antes descrito es utilizar una colección arbitraria C para guardar a los vértices explorados, mismos que se irán coloreando de negro para no explorarlos nuevamente. Mientras C sea no vacía, se elegirá un vértice arbitrario x en C , si desde x es posible alcanzar algún vértice “nuevo” y , i.e., que no ha sido explorado aún, dicho vértice es marcado como explorado (coloreado de negro) e

incluido en C . Adicionalmente, se guarda la información de que el vértice y fue explorado desde el vértice x . La idea es que los vértices negros conforman el árbol parcial que se hará crecer, y una vez que se encuentra un vértice no explorado, éste es añadido al árbol parcial mediante la arista que lo une al vértice desde donde fue encontrado. Cuando un vértice en C ya no tiene vecinos sin explorar, éste es eliminado de C .

Vale la pena resaltar que de forma natural, el Algoritmo 1, y todas sus variantes, devolverán un árbol generador arraigado en el vértice inicial. Este árbol estará dado únicamente por una función de parentesco que indica quién es el padre de cada vértice, a partir de la cual es fácil reconstruir el árbol. Adicionalmente, se asignará un tiempo de exploración a cada vértice.

Algoritmo 1: Búsqueda en gráficas

Input: Una gráfica G con un vértice distinguido r .

Output: Funciones de parentesco p y tiempo de exploración t .

```
1  $C \leftarrow \emptyset$ ;  $i \leftarrow 0$ ;  
2  $i \leftarrow i + 1$ ;  
3 colorear a  $r$  de negro;  
4  $C \leftarrow C \cup \{r\}$ ,  $t(r) \leftarrow i$ ,  $p(r) \leftarrow \emptyset$ ;  
5 while  $C \neq \emptyset$  do  
6   elegir algún vértice  $x$  en  $C$ ;  
7   if  $x$  tiene un vecino  $y$  sin colorear then  
8      $i \leftarrow i + 1$ ;  
9     colorear a  $y$  de negro;  
10     $C \leftarrow C \cup \{y\}$ ,  $t(y) \leftarrow i$ ,  $p(y) \leftarrow x$ ;  
11  end  
12  else  
13     $C \leftarrow C - \{x\}$ ;  
14  end  
15 end  
16 return  $(p, t)$ 
```

Proposición 2.4.1. *La función de parentesco devuelta por el Algoritmo 1 corresponde a un árbol generador de la componente conexa que contiene al vértice r de la gráfica que el algoritmo recibe como entrada.*

Demostración. La subgráfica que consta únicamente del vértice r es un árbol. Verifiquemos que, tras cada ejecución del ciclo **while** que inicia en la línea 5, la subgráfica de G inducida por las aristas dadas por la función de parentesco es un árbol. La afirmación anterior es claramente cierta si no se encuentra un vértice nuevo

en esta iteración del ciclo, es decir, si la condición del `if` en la línea 7 no se cumple. En otro caso, el vértice activo x tiene un vecino y sin colorear. El padre de y se define como x . Por hipótesis inductiva, la subgráfica de G inducida por el conjunto S de las aristas dadas por la función de parentesco, hasta antes de designar a x como el padre de y , es un árbol T . Al designar a x como padre de y , estamos agregando a la arista xy al conjunto S , por lo que esta acción corresponde con agregar una hoja al árbol T , a saber, la hoja y con soporte x . Claramente, al agregar una hoja a un árbol se obtiene nuevamente un árbol, por lo que la salida del Algoritmo 1 es realmente un árbol T .

Para ver que T es un árbol generador de la componente conexa que contiene a r , podemos suponer sin pérdida de generalidad que G es conexa, y basta demostrar que todo vértice de G es agregado en algún momento a C , i.e., todo vértice es coloreado de negro durante la ejecución del algoritmo.

Mientras haya algún vértice sin colorear, consideremos la partición (X, Y) de V donde X es el conjunto de los vértices negros, y Y el conjunto de los vértices sin colorear. Como G es conexa, debe de existir alguna XY -arista en G , digamos xy , con $x \in X$ y $y \in Y$. Así, un vértice en Y debe ser coloreado de negro a más tardar cuando x sea el vértice activo y se explore a su vecino y . \square

Se sigue de la Proposición 2.4.1 que el Algoritmo 1 puede ser utilizado para verificar si una gráfica es conexa; basta con ejecutarlo desde cualquier vértice, y al terminar, verificar si todos los vértices fueron coloreados de negro.

Es posible hacer un análisis de tiempo de ejecución del Algoritmo 1, aún cuando no contemos con información específica sobre C , que se podrá afinar una vez que se elija una estructura de datos específica.

Proposición 2.4.2. *Sea G una gráfica conexa y sea C una colección con funciones de tiempo a_C y e_C para agregar y eliminar elementos, respectivamente. El Algoritmo 1 se ejecuta en tiempo $O(|V|(a_C + e_C) + |E|)$.*

Demostración. Todas las operaciones utilizadas por el algoritmo se ejecutan en tiempo constante, excepto agregar un elemento o eliminar un elemento de C .

Notemos que cada vértice es agregado exactamente una vez y eliminado exactamente una vez de C , acciones que en total toman tiempo $O(|V|(a_C + e_C))$. Además, para cada vértice, cada uno de sus vecinos es explorado para verificar si está o no coloreado, lo que resulta en $\sum_{v \in V} d(v) = 2|E|$ verificaciones. Así, el tiempo total de ejecución es $O(|V|(a_C + e_C) + |E|)$. \square

2.5. BFS

Cuando utilizamos una cola Q como nuestra colección para guardar a los vértices explorados, el algoritmo que obtenemos se conoce como *Breadth-first Search*¹, o BFS

¹Búsqueda priorizando amplitud.

por sus siglas en inglés. Adicionalmente a la información con la que ya se contaba, se añade una función $\ell: V_G \rightarrow \mathbb{N}$, que le asocia a cada vértice el *nivel* en el que se encuentra. Un consecuencia de utilizar una cola, es que el nivel de un vértice v es la distancia de la raíz r a v , tanto en el árbol obtenido, como en la gráfica original. A un árbol generador obtenido por el algoritmo BFS le llamaremos un *árbol de BFS*. Notemos que el pseudocódigo de BFS (Algoritmo 2) es prácticamente el mismo que el del Algoritmo 1, salvo por la nueva función ℓ , y el uso de dos renglones adicionales para indicar cuando un vértice se encola en Q .

Usar una cola tiene como consecuencia que un vértice no saldrá de ésta hasta que todos sus vecinos hayan sido explorados. Es esta propiedad la que le da el nombre al algoritmo, pues la se prioriza la amplitud en la exploración alrededor de la raíz; primero se explorarán todos los vecinos de la raíz, después los vecinos de los vecinos de la raíz, y, en general, no se examinarán vértices más lejos de la raíz hasta no haber cubierto todos los que se encuentran en cierto radio de la misma.

Algoritmo 2: Breadth First Search

Input: Una gráfica conexa G con un vértice distinguido r .

Output: Funciones de parentesco p , nivel ℓ y tiempo de exploración t .

```
1  $Q \leftarrow []$ ;  $i \leftarrow 0$ ;  
2  $i \leftarrow i + 1$ ;  
3 colorear a  $r$  de negro;  
4 añadir  $r$  al final de  $Q$ ;  
5  $t(r) \leftarrow i$ ,  $p(r) \leftarrow \emptyset$ ,  $\ell(r) \leftarrow 0$ ;  
6 while  $Q \neq []$  do  
7   elegir a la cabeza  $x$  de  $Q$ ;  
8   if  $x$  tiene un vecino  $y$  sin colorear then  
9      $i \leftarrow i + 1$ ;  
10    colorear a  $y$  de negro;  
11    añadir  $y$  al final de  $Q$ ;  
12     $t(y) \leftarrow i$ ,  $p(y) \leftarrow x$ ,  $\ell(y) \leftarrow \ell(x) + 1$ ;  
13  end  
14  else  
15    eliminar  $x$  de  $Q$ ;  
16  end  
17 end  
18 return  $(p, \ell, t)$ 
```

Proposición 2.5.1. *El Algoritmo 2 devuelve un árbol generador de la gráfica conexa G que toma como argumento, y su tiempo de ejecución es $O(|V| + |E|)$.*

Demostración. La primera afirmación se sigue directamente de la Proposición 2.4.1. La segunda afirmación se sigue de la Proposición 2.4.2, notando que es posible agregar y eliminar elementos a una cola en tiempo constante. \square

La propiedad fundamental de los árboles de BFS es que la distancia de la raíz a cualquier vértice de la gráfica es igual en el árbol que en la gráfica, por lo que este algoritmo resulta útil para encontrar trayectorias de longitud mínima entre un vértice de la gráfica y cualquier otro. Para obtener esta propiedad, basta verificar que ambas distancias coinciden con el valor de la función ℓ devuelta por BFS.

Lema 2.5.2. *Si T es un árbol de BFS de una gráfica conexa G con raíz r , entonces se cumplen las siguientes afirmaciones:*

1. *para cada vértice de G , $\ell(v) = d_T(r, v)$,*
2. *cada arista de G une vértices en el mismo nivel, o en niveles consecutivos de T , es decir, para cada $uv \in E$ se cumple*

$$|\ell(u) - \ell(v)| \leq 1.$$

Demostración. Únicamente demostraremos la segunda afirmación. Basta demostrar que para cualquier arista uv , si $\ell(u) < \ell(v)$, entonces $\ell(u) = \ell(v) - 1$.

Procediendo por inducción sobre $\ell(u)$, veamos primero que si u y v son vértices tales que $\ell(u) < \ell(v)$, entonces u entró a Q antes que v . Si $\ell(u) = 0$, entonces u es la raíz de T , y fue el primer vértice en entrar a Q . Supongamos entonces que $\ell(u) = k$ con $k > 0$. Si $x = p(u)$ y $y = p(v)$, entonces tenemos por la línea 12 del Algoritmo 2 que $\ell(x) = \ell(u) - 1 < \ell(v) - 1 = \ell(y)$. Por hipótesis inductiva, x entró a Q antes que y . Como u es un vecino de x , entonces u entró a Q antes que v .

Supongamos que $uv \in E$ y $\ell(u) < \ell(v)$. Si $u = p(v)$, se sigue nuevamente de la línea 12 del algoritmo que $\ell(u) = \ell(v) - 1$. Si no, sea $y = p(v)$. Como v se agregó a T mediante la arista vy , y no mediante la arista uv , el vértice y entró a Q antes que u , y por la contrapositiva de la afirmación demostrada en el párrafo anterior, se tiene que $\ell(y) \leq \ell(u)$. Luego, $\ell(v) - 1 = \ell(y) \leq \ell(u) \leq \ell(v) - 1$, de donde se sigue que $\ell(u) = \ell(v) - 1$. \square

Para finalizar esta sección, se demuestra la propiedad principal antes mencionada de los árboles de BFS.

Teorema 2.5.3. *Sea G una gráfica conexa. Para cada vértice v de G , el valor $\ell(v)$, de la función ℓ devuelta por BFS, es la distancia de la raíz r a v , es decir $\ell(v) = d_G(r, v)$.*

Demostración. Sea v un vértice de G . Se sigue del Lema 2.5.2 que $\ell(v) = d_T(r, v)$, y como T es una subgráfica de G , se tiene que $\ell(v) \geq d_G(r, v)$. Demostremos la desigualdad restante por inducción sobre la longitud de una rv -trayectoria de longitud mínima.

Sea P una rv -trayectoria de longitud mínima en G , con $v \neq r$, y sea u el vértice anterior a v en P . Entonces, rPu es una ru -trayectoria de longitud mínima, y se sigue de la hipótesis inductiva que $\ell(u) \leq d_G(r, u)$. Por la elección de u , $d_G(r, u) = d_G(r, v) - 1$. Del Lema 2.5.2 se sigue que $\ell(v) - \ell(u) \leq 1$, por lo que

$$\ell(v) \leq \ell(u) + 1 \leq d_G(r, u) + 1 = d_G(r, v).$$

□

2.6. DFS

En el caso en que la colección para guardar a los vértices explorados sea una pila S , se obtiene el algoritmo conocido como *Depth-first Search*² o DFS por sus siglas en inglés. En este caso, se tendrán dos funciones de tiempo, una función f que indica el tiempo de entrada de un vértice a S (la misma que t en el Algoritmo 1), y una función l , que indica el tiempo de salida de S de cada vértice.

A diferencia de los árboles de BFS, en los árboles de DFS la distancia de la raíz a un vértice en el árbol no coincide con su distancia en la gráfica original, un ejemplo claro de esto son las gráficas completas; cualquier árbol de DFS de una gráfica completa es una trayectoria, por lo que hay vértices en el árbol a todas las posibles distancias entre 1 y $|V| - 1$ de la raíz, mientras que en la gráfica original cualesquiera dos vértices están a distancia 1. Entonces, ¿qué tienen de especial los árboles de DFS?

A diferencia de una cola, en la que los nuevos vértices se forman al final de la misma, y por lo tanto el vértice activo durante la búsqueda siempre será el mismo hasta que todos sus vecinos hayan sido explorados, en una pila, los vértices nuevos se añaden en la parte superior de la pila, lo que hace que el vértice activo cambie, potencialmente, en cada iteración del algoritmo. Cada que un vértice es agregado al árbol, éste quedará en la parte superior de la pila, por lo que el crecimiento del árbol se verá como una rama muy larga, que continuará hasta que se encuentre un vértice tal que todos sus vecinos hayan sido explorados. En ese momento, se empezarán a sacar vértices de la pila, hasta que se encuentre uno por el cuál es posible seguir avanzando; es hasta ese momento en el que habrá una ramificación en el árbol. Los árboles de DFS usualmente tendrán ramas largas con pocas ramificaciones, en contraposición con los árboles de BFS, que tendrán vértices de grado alto (muchas ramificaciones), y serán poco profundos (la profundidad puede ser a lo más el diámetro de la gráfica). Es por este motivo que en los árboles de DFS las ramificaciones se darán únicamente cuando se hayan terminado de explorar los vértices dentro de un mismo bloque de la gráfica. Esta condición se refleja en una condición un poco técnica de las funciones f y l , que se revisará tras presentar el pseudocódigo de DFS.

²Búsqueda priorizando profundidad.

A continuación se presenta el pseudocódigo de DFS. De forma similar a lo que sucedió con BFS, el pseudocódigo de DFS es prácticamente el mismo que el del Algoritmo 1. La función f en DFS tiene la misma finalidad que la función t en el Algoritmo 1, y adicionalmente se tiene una función l de tiempo de salida de la pila. La instrucción de agregar un vértice a la pila aparece en un renglón separado, lo que agrega dos líneas al pseudocódigo (líneas 4 y 11), y también se agrega la línea 16, donde se asigna tiempo de salida de la pila a un vértice cuando ya no tiene vecinos sin explorar.

Algoritmo 3: Depth First Search

Input: Una gráfica G con un vértice distinguido r .

Output: Funciones de parentesco p , tiempo de entrada f y tiempo de salida l .

```

1  $S \leftarrow []$ ;  $i \leftarrow 0$ ;
2  $i \leftarrow i + 1$ ;
3 colorear a  $r$  de negro;
4 añadir  $r$  en la parte superior de  $S$ ;
5  $f(r) \leftarrow i$ ,  $p(r) \leftarrow \emptyset$ ;
6 while  $S \neq []$  do
7    $i \leftarrow i + 1$ ;
8   elegir al tope  $x$  de  $S$ ;
9   if  $x$  tiene un vecino  $y$  sin colorear then
10    colorear a  $y$  de negro;
11    añadir  $y$  en la parte superior de  $S$ ;
12     $f(y) \leftarrow i$ ,  $p(y) \leftarrow x$ ;
13  end
14  else
15    eliminar  $x$  de  $S$ ;
16     $l(x) \leftarrow i$ ;
17  end
18 end
19 return  $(p, f, l)$ 

```

El siguiente lema explora la condición antes mencionada de las funciones f y l .

Lema 2.6.1. Sea G una gráfica y T un árbol de DFS de G . Además, sean u y v vértices de G con $f(u) < f(v)$.

1. Si u y v son adyacentes en G , entonces $l(v) < l(u)$.
2. u es un ancestro de v en T si y sólo si $l(v) < l(u)$.

Demostración. De acuerdo con el **if-else** de las líneas 9 y 14, un vértice u saldrá de S únicamente cuando sea el vértice activo y cada uno de sus vecinos haya sido explorado para verificar si puede ser agregado a T . Uno de estos vecinos es v , pues $f(u) < f(v)$. Luego, v se añade a la pila S mientras u todavía está en S , por lo que u no puede salir de S antes de que v salga. Por lo tanto, $l(v) < l(u)$.

Para la afirmación restante, supongamos que u es un ancestro de v en T . Por las líneas 7 y 12 de DFS, es fácil observar que los valores de f crecen a lo largo de la trayectoria uTv . Aplicando el primer inciso de esta proposición a cada arista de esta trayectoria se obtiene que $l(v) < l(u)$.

Supongamos ahora que u no es un ancestro de v en T . Como $f(u) < f(v)$, entonces v tampoco es un ancestro de u . Luego, u no está en la trayectoria rTv , y v no está en la trayectoria rTu . Sea s el último vértice común de estas dos trayectorias. Como $f(u) < f(v)$, los descendientes propios de s en la trayectoria rTv sólo pudieron haber sido agregados a S después de que todos los descendientes propios de s en la trayectoria rTu hubieran sido removidos de S , dejando a s en la parte superior de la pila. En particular, v sólo pudo haber sido agregado a S después de que u hubiera sido removido, por lo que $l(u) < f(v)$. Para terminar, es claro que $f(v) < l(v)$, de donde se concluye que $l(u) < l(v)$. \square

El resultado técnico del Lema 2.5.2 da lugar a una condición mucho más intuitiva, enunciada en el siguiente teorema.

Teorema 2.6.2. *Si T es un árbol de DFS de una gráfica G , entonces cada arista de G une vértices que están relacionados en T .*

Demostración. Sea uv una arista de G y supongamos sin pérdida de generalidad que $f(u) < f(v)$. Por el Lema 2.6.1, se tiene que $l(v) < l(u)$. El segundo inciso del mismo lema implica que u es un ancestro de v , por lo que u y v están relacionados en T . \square

2.7. Árboles de peso mínimo

Una *gráfica ponderada* es una gráfica G equipada con una función $w: E_G \rightarrow \mathbb{R}$, que asocia pesos a las aristas de G . Si H es una subgráfica de G , el peso de H , denotado por $w(H)$, se define como $w(H) = \sum_{e \in E_H} w(e)$. De esta manera, un árbol generador de la gráfica G es *óptimo* si su peso es mínimo respecto a la familia de todos los árboles generadores de G . En otras palabras, si $\mathcal{T} = \{T: T \text{ es un árbol generador de } G\}$, entonces T_0 es un árbol óptimo de G si $w(T_0) = \min_{T \in \mathcal{T}} w(T)$.

En el caso de las gráficas ponderadas, el Algoritmo 1 también puede ser modificado para encontrar un árbol óptimo. El algoritmo propuesto seguirá una estrategia glotona, o avariciosa, en la que en cada paso, se hará crecer el árbol parcial con la arista de peso mínimo que una a un vértice del árbol con un vértice fuera del árbol. De manera natural, se plantea el uso de una cola de prioridades como estructura de

datos para ir guardando los vértices explorados, pues esto permitirá obtener uno de peso mínimo de forma eficiente. Sin embargo, esto representa un problema, pues los pesos se encuentran en las aristas, no en los vértices. Una posible solución podría ser tener a las aristas en la frontera del árbol parcial en una cola de prioridades, para obtener la de peso mínimo de forma eficiente. Sin embargo, los vértices tienen que ser procesados para explorar sus vecindades, por lo que esto representaría trabajo doble.

La solución clásica dada por el Algoritmo de Jarník-Prim, es asignar una función de costo a cada vértice, que inicialmente será 0 para la raíz e infinito para todos los demás vértices, y que más adelante dependerá del peso de las aristas que lo unen al árbol parcial. Para cada árbol parcial T , su frontera será el corte por aristas $\partial(V_T)$. A diferencia del Algoritmo 1, donde se exploran los vecinos de un vértice sin eliminarlo de la colección, y éste sale de la colección cuando ya no tiene vecinos sin explorar, en el algoritmo que se presentará a continuación, el vértice cuyos vecinos serán explorados se elimina de la colección, y todos sus vecinos son procesados al mismo tiempo. Al procesar al vértice u , para cada vecino sin colorear v de u , se compara el costo de u con el peso de la arista que lo une a v , si el peso de la arista es menor que el costo de u , entonces se actualiza el costo de u al peso de uv (y se reordena la cola de prioridades), y se asigna a u como padre temporal de v ; si no, no se realizan cambios. Así, en cada momento, se tiene que el costo de cada vértice x representa cuánto aumentará el peso del árbol parcial al agregar a x mediante la arista que lo une a su padre temporal.

El siguiente resultado muestra que Algoritmo de Jarník-Prim devuelve un árbol óptimo en una gráfica ponderada.

Teorema 2.7.1. *Cada árbol de Jarník-Prim es un árbol óptimo.*

Demostración. Sea T un árbol de Jarník-Prim con raíz r . Demostraremos, por inducción sobre $|V|$, que T es un árbol óptimo. En primer lugar, demostraremos que la primera arista e agregada a T pertenece a algún árbol óptimo de G . La arista e fue la de menor peso en la frontera del árbol trivial cuyo único vértice es r , i.e., $w(e) \leq w(f)$ para cualquier arista f incidente en r . Sea T^* un árbol óptimo de G . Si e no es una arista de T^* , entonces $T + e$ contiene un único ciclo C . Si f es la otra arista de C que incide en r , entonces $T^{**} = (T^* + e) - f$ es un árbol generador de G . Además, como $w(e) \leq w(f)$, se tiene que

$$w(T^{**}) = w(T^*) + w(e) - w(f) \leq w(T^*).$$

Como T^* es un árbol óptimo, se debe cumplir la igualdad, por lo que T^{**} es un árbol óptimo que contiene a e .

Consideremos a la gráfica $G' = G/e$, y denotemos por r' al vértice obtenido al contraer a e . Más adelante veremos que hay una correspondencia biyectiva entre el conjunto de árboles generadores de G que contienen a e y el conjunto de todos los

Algoritmo 4: Jarník-Prim

Input: Una gráfica conexa ponderada (G, w) con un vértice distinguido r .
Output: Un árbol óptimo T con función de parentesco p y su peso $w(T)$.

```

1  $c(r) \leftarrow 0, w(T) \leftarrow 0;$ 
2 for  $v \in V - \{r\}$  do
3    $c(v) \leftarrow \infty;$ 
4    $p(v) \leftarrow \emptyset;$ 
5 end
6 while haya vértices sin colorear do
7   elegir un vértice sin colorear  $u$  de costo mínimo  $c(u)$ ;
8   colorear a  $u$  de negro;
9    $w(T) \leftarrow w(T) + c(u);$ 
10  for cada vértice sin colorear  $v$  tal que  $w(uv) < c(v)$  do
11     $p(v) \leftarrow u;$ 
12     $c(v) \leftarrow w(uv);$ 
13  end
14 end
15 return  $(p, w(T))$ 

```

árboles generadores de G' . Así, para demostrar que el árbol final T es un árbol óptimo de G , basta demostrar que $T' = T/e$ es un árbol óptimo de G' . Afirmamos que T' es un árbol de Jarník-Prim arraigado en r' .

Consideremos el árbol parcial T en algún punto de la ejecución del algoritmo. Supongamos que T no es únicamente el vértice r , y por lo tanto incluye a la arista e . Notemos que si $T' = T/e$, entonces $\partial(T) = \partial(T')$, por lo que una arista de peso mínimo en $\partial(T)$ también es una arista de peso mínimo en $\partial(T')$. Así, la siguiente elección en la cola de prioridades en la ejecución del algoritmo coincide para T y para T' , por lo que los árboles finales coincidirán salvo por el vértice r' . Como el árbol final T es un árbol de Jarník-Prim de G , deducimos que el árbol final T' es un árbol de Jarník-Prim de G' . Como G' tiene un vértice menos que G , la hipótesis inductiva nos dice que T' es un árbol óptimo de G' .

Si \mathcal{T}_e es el conjunto de todos los árboles generadores de G que contienen a la arista e , y \mathcal{T}' es el conjunto de todos los árboles generadores de G/e , no es difícil observar que la función $\varphi: \mathcal{T}_e \rightarrow \mathcal{T}'$ dada por $\varphi(T) = T/e$ es una biyección. Sea T^* un árbol óptimo de G . Entonces, como el árbol de Jarník-Prim T' de G' es óptimo, tenemos que $w(T) - w(e) = w(T/e) = w(T') \leq w(\varphi(T^*)) = w(T^*/e) = w(T^*) - w(e)$, por lo que $w(T) \leq w(T^*)$. Luego, el árbol T es un árbol óptimo de G . \square

El tiempo de ejecución del algoritmo de Jarník-Prim puede deducirse de manera casi directa de la Proposición 2.4.2. Sin embargo, hay que considerar que, potencialmente, cada vez que un vecino de un vértice sea explorado, se tendrá que reordenar la

cola de prioridades, lo que toma tiempo $O(\log |V|)$, y se hace $O(|E|)$ veces. Además, la complejidad de agregar o eliminar un vértice a la cola de prioridades es $O(\log |V|)$. Añadiendo esta información a la brindada por la Proposición 2.4.2, se obtiene que el Algoritmo de Jarník-Prim corre en tiempo $O((|V| + |E|) \log |V|)$.

Otro acercamiento para encontrar un árbol óptimo en una gráfica ponderada fue considerado por Borůvka y Kruskal, quienes propusieron independientemente el siguiente algoritmo. Este algoritmo también es de naturaleza avariciosa, pero a diferencia del algoritmo de Jarník-Prim, no intenta mantener una gráfica conexa en cada iteración, sino que simplemente elige la arista de menor peso disponible que no forme ciclos con las que se han elegido previamente.

Algoritmo 5: Borůvka-Kruskal

Input: Una gráfica ponderada conexa (G, w) .

Output: Un árbol óptimo $T = (V, F)$ de G y su peso $w(F)$.

```

1  $F \leftarrow \emptyset, w(F) \leftarrow 0;$ 
2 while haya una arista  $e \in E - F$  tal que  $F \cup \{e\}$  induce un bosque do
3   |   elegir una arista  $e$  con dicha propiedad de peso mínimo;
4   |    $F \leftarrow F \cup \{e\}, w(F) \leftarrow w(F) + w(e);$ 
5 end
6 return  $((V, F), w(F))$ 

```

El pseudocódigo del Algoritmo 5 realmente no corresponde a un algoritmo implementable, pues no indica cómo verificar si al agregar una arista al conjunto de aristas previamente seleccionadas, ésta forma un ciclo. Una modificación sencilla del algoritmo arregla este problema. Inicialmente se asigna una etiqueta distinta a cada vértice, utilizando los primeros $|V|$ enteros positivos como etiquetas. Cada vez que se agrega una arista, se verifican las etiquetas en los extremos de la misma, si éstas coinciden, quiere decir que la arista une dos vértices en la misma componente conexa, por lo que se formaría un ciclo al agregarla. Si no coinciden, la arista está uniendo dos componentes conexas distintas, por lo que no se forman ciclos al agregarla, pero es necesario cambiar las etiquetas para reflejar este cambio. Si la arista considerada tiene extremos con etiquetas i y j con $i < j$, entonces cada vértice etiquetado con j se reetiqueta con i . Esto puede hacerse eficientemente utilizando una estructura de datos conocida como *conjuntos ajenos*, por lo que puede demostrarse que el Algoritmo de Borůvka-Kruskal corre en tiempo $O(|E| \log |E|) = O(|E| \log |V|)$.

Para finalizar esta sección, se presenta el algoritmo de Dijkstra. Nótese que éste algoritmo resulta prácticamente igual que el Algoritmo de Jarník-Prim (Algoritmo 4), salvo por la línea 9 del algoritmo de Jarník-Prim, donde se actualiza el peso del árbol parcial, y la línea 12 del Algoritmo de Jarník-Prim, que corresponde a la línea 11 del Algoritmo de Dijkstra, donde se actualiza el costo de un vértice explorado.

Esta pequeña diferencia en la forma de recalcular el costo de un vértice hace que el algoritmo de Dijkstra, en lugar de calcular un árbol generador de peso mínimo, calcule la distancia ponderada mínima desde la raíz hasta cualquier otro vértice en la gráfica.

Algoritmo 6: Dijkstra

Input: Una digráfica ponderada (D, w) con un vértice distinguido r .

Output: Una ramificación en D con función de parentesco p y una función $\ell: V \rightarrow \mathbb{R}^+$ tal que $\ell(v) = d_D(r, v)$ para cada $v \in V_D$.

```
1  $\ell(r) \leftarrow 0$ ;  
2 for  $v \in V - \{r\}$  do  
3    $\ell(v) \leftarrow \infty$ ;  
4    $p(v) \leftarrow \emptyset$ ;  
5 end  
6 while haya un vértice sin colorear  $u$  con  $\ell(u) < \infty$  do  
7   elegir un vértice sin colorear  $u$  con  $\ell(u)$  mínimo;  
8   colorear a  $u$  de negro;  
9   for cada exvecino sin colorear  $v$  de  $u$  con  $\ell(v) > \ell(u) + w(u, v)$  do  
10     $p(v) \leftarrow u$ ;  
11     $\ell(v) \leftarrow \ell(u) + w(u, v)$ ;  
12  end  
13 end  
14 return  $(p, \ell)$ 
```

2.8. Isomorfismo de árboles

Teorema 2.8.1. *El problema de determinar si dos árboles de orden n son isomorfos puede resolverse en tiempo $O(n)$.*

Capítulo 3

Conexidad

3.1. Introducción

Como se estudió en el capítulo 2, los árboles son gráficas mínimas con la propiedad de ser conexas. Esto se ve reflejado en diversos resultados, por ejemplo, cada arista de un árbol es un puente, cada vértice distinto de una hoja es un vértice de corte, y quizá una de las características más reveladoras, es que entre cualesquiera dos vértices existe una única trayectoria. De hecho, los dos primeros resultados antes mencionados se siguen claramente del tercero, pues una arista es la única trayectoria entre sus extremos, y un vértice de grado al menos dos es el único vértice interno de la única trayectoria que une a cualesquiera dos de sus vecinos.

Así, resulta natural que para obtener una gráfica más robusta, en el sentido de continuar siendo conexa ante la pérdida de vértices o aristas, es necesario que existan distintas trayectorias entre pares de vértices. Si k es un entero positivo, y u y v son vértices en una gráfica G entre los que hay k trayectorias internamente ajenas, entonces siempre que S sea un conjunto con menos de k vértices, al que no pertenezca ni u ni v , en la gráfica $G - S$ existirá al menos una uv -trayectoria, pues a lo más se borrará un vértice de $|S|$ de ellas, pero $|S| < k$. Como consecuencia, si entre cualesquiera dos vértices de G existen al menos k trayectorias internamente ajenas, entonces siempre que se borren menos de k vértices de G , la gráfica resultante seguirá siendo conexa. Resulta natural preguntarse si el enunciado recíproco también es verdadero, ¿es cierto que si G es una gráfica en la que para cualquier subconjunto S de V con menos de k vértices se tiene que $G - S$ es conexa, entonces entre cualquier par de vértices existen al menos k trayectorias internamente ajenas?

Cuando $k = 1$, la respuesta a la pregunta enunciada en el párrafo anterior es trivialmente afirmativa. Cuando $k = 2$, la pregunta se reduce a, si una gráfica no tiene vértices de corte, ¿es cierto que entre cualesquiera dos vértices hay al menos dos trayectorias internamente ajenas? Cabe señalar que por la definición de vértice de corte, las gráficas completas K_1 y K_2 resultan ser casos patológicos, pues no contienen vértices de corte, pero no existen al menos dos trayectorias internamente

ajenas entre cualquier par de vértices. Sin embargo, si una gráfica tiene al menos tres vértices, entonces la respuesta a la pregunta anterior es afirmativa. En la siguiente sección se estudian las gráficas que no tienen vértices de corte.

En el resto del capítulo, consideramos el caso general de la pregunta, cuando k es cualquier entero positivo. Para este fin, se introducen los conceptos de conexidad, y conexidad por aristas. Notemos que tiene sentido plantear la misma pregunta considerando conjuntos de aristas y trayectorias ajenas por aristas, en lugar de conjuntos de vértices y trayectorias internamente ajenas.

3.2. Bloques

Aunque las nociones de puente y vértice de corte parecen coincidir naturalmente, cuando una gráfica tiene lazos, los vértices de corte no logran capturar completamente la misma esencia que tienen los puentes; los puentes separan a la gráfica en dos partes que no pueden alcanzarse si no es pasando por el puente.

descomposición Una **descomposición** de una gráfica G es una familia \mathcal{F} de subgráficas de G , ajenas por aristas dos a dos, y tales que

$$\bigcup_{F \in \mathcal{F}} E_F = E_G.$$

vértice separador Una **separación** de una gráfica conexa G es una descomposición de la gráfica en dos subgráficas conexas y no vacías que tienen exactamente un vértice en común. Este vértice común es un **vértice separador** de la gráfica. Si G es inconexa, sus vértices separadores son los vértices separadores de sus componentes conexas. Claramente, un vértice de corte es un vértice separador, sin embargo, no todo vértice separador es un vértice de corte; un vértice incidente en un lazo y cualquier otra arista es un vértice separador, pero no necesariamente es de corte. Sin embargo, en las gráficas sin lazos, los vértices de corte y los vértices separadores coinciden.

no separable Una gráfica es **no separable** si es conexa y no tiene vértices separadores; en otro caso es **separable**. Salvo isomorfismo, sólo hay dos gráficas no separables con un vértice, K_1 y K_1 con un lazo. Ninguna gráfica no separable con dos o más vértices tiene lazos. Las aristas múltiples son intrascendentes para la separabilidad de una gráfica, así, una gráfica sin lazos es no separable si y sólo si su gráfica simple subyacente es no separable.

bloque Un **bloque** de una gráfica G es una subgráfica de G no separable que es máxima con esta propiedad. Así, una gráfica no separable tiene un único bloque, que es la gráfica misma, por este motivo, usualmente las gráficas no separables serán llamadas bloques.

A continuación presentamos algunas propiedades de los bloques de una gráfica, que serán de utilidad para introducir una gráfica auxiliar que refleja la estructura de los bloques de una gráfica.

Proposición 3.2.1. *Si G es una gráfica, entonces:*

1. *cualesquiera dos bloques distintos de G tienen a lo más un vértice en común,*
2. *los bloques de G forman una descomposición de G ,*
3. *cada ciclo de G está contenido en un bloque de G .*

Demostración. Para el primer enunciado procederemos por contrapositiva, por lo que es buena idea escribirlo en forma de implicación. Demostraremos que para cualesquiera dos subgráficas distintas y no separables G_1 y G_2 de G , si G_1 es un bloque y G_2 es un bloque, entonces su intersección tiene a lo más un vértice. Así, para la contrapositiva, veamos que si G_1 y G_2 son subgráficas distintas de G , no separables y con al menos dos vértices en común, entonces alguna de las dos no es un bloque. Como cada una tiene al menos dos vértices, si alguna tiene un lazo, entonces no es separable, por lo que podemos suponer que ninguna tiene lazos. Sea $B = G_1 \cup G_2$, afirmamos que B es no separable. Sea v un vértice arbitrario de B . Entonces, $B - v = (G_1 - v) \cup (G_2 - v)$ es conexa, pues $G_1 - v$ y $G_2 - v$ son ambas conexas, y tienen al menos un vértice en común. Luego, B no tiene vértices de corte, y al no tener lazos, es no separable. Así, o bien, sin pérdida de generalidad G_1 contiene propiamente a G_2 , por lo que G_2 no es un bloque, o ni G_1 ni G_2 son gráficas máximas por contención con la propiedad de ser no separables, por lo que ninguna es un bloque.

Para el segundo enunciado, notemos que cada arista de G induce una subgráfica no separable de G (incluso si es un lazo), y por lo tanto está contenida en algún bloque de G . Por otro lado, se sigue del inciso anterior que ninguna arista está contenida en más de un bloque. Así, los bloques de G conforman una descomposición de G .

Finalmente, un ciclo de G es una subgráfica no separable, por lo que está contenido en un bloque de G . \square

Para cualquier gráfica conexa G existe una gráfica bipartita B_G que captura la estructura en bloques de la gráfica G . La bipartición de B_G es (\mathcal{B}, S) , donde \mathcal{B} es el conjunto de bloques de G y S es el conjunto de vértices separadores de G . Un bloque B es adyacente a un vértice separador v en B_G si y sólo si v es un vértice de B . Se deja como ejercicio al lector observar que B_G es de hecho un árbol; que B_G sea acíclica se sigue de la Proposición 3.2.1. Así, la gráfica B_G usualmente se conoce como el **árbol de bloques** de G . Si G es separable, un bloque de G que corresponde a una hoja de B_G se conoce como un **bloque terminal**. Un **vértice interno** de un bloque en una gráfica G es un vértice que no es separador en G .

árbol de bloques
bloque terminal

Muchas propiedades son fáciles de demostrar para gráficas separables procediendo por inducción sobre su número de bloques. Usualmente basta eliminar un bloque terminal, aplicar la hipótesis inductiva, y devolver el bloque terminal a la gráfica. Así, es posible demostrar algunas propiedades para todas las gráficas con la técnica antes descrita, si para la base de inducción se considera el caso en el que hay un

único bloque, i.e., cuando la gráfica es no separable. Para tal fin, en la Sección 3.3 se introduce una descripción recursiva de las gráficas no separables.

Se concluye esta sección con el Teorema de Caracterización de Bloques, que presenta diversas formas equivalentes de definir a un bloque.

Teorema 3.2.2. *Sea G una gráfica conexa, sin lazos y con al menos 3 vértices. Los siguientes enunciados son equivalentes:*

1. G es un bloque.
2. Entre cualesquiera dos vértices distintos existen dos trayectorias internamente ajenas.
3. Cualesquiera dos vértices de G están en un ciclo común.
4. Cualquier vértice y cualquier arista de G están en un ciclo común.
5. Cualesquiera dos aristas de G están en un ciclo común.
6. Dados dos vértices y una arista de G , hay una trayectoria que une a los vértices y pasa por la arista.
7. Para cualesquiera tres vértices distintos u, v y w de G , hay una uv -trayectoria que pasa por w .
8. Para cualesquiera tres vértices distintos u, v y w de G , hay una uv -trayectoria que no pasa por w .

Demostración. Iniciamos con 1 implica 2. Sean u y v dos vértices de G . Procederemos por inducción sobre la distancia $d(u, v)$. Si $d(u, v) = 1$, entonces u es adyacente a v . Como G es no separable, se sigue por un ejercicio de la sección que no tiene puentes. Así, la arista uv está en un ciclo por el Lema 1.6.5, por lo que hay dos trayectorias internamente ajenas entre u y v en G .

Si $d(u, v) = k$ con $k \geq 2$, entonces consideramos a w , el penúltimo vértice en una uv -trayectoria de longitud k . Por la elección de w , tenemos que $d(u, w) = k - 1$, y por hipótesis de inducción existen P y Q , uw -trayectorias internamente ajenas. Como G es no separable, $G - w$ es conexa, por lo que existe una uv -trayectoria P' en $G - w$; notemos que P' es una uv -trayectoria en G que no pasa por w . Sea x el último vértice de P' que interseca a $P \cup Q$, y supongamos sin pérdida de generalidad que $x \in V_P$. Notemos que x existe, pues u es un vértice en común para P' , P y Q ; además, es posible que $x = u$. Entonces PxP' y Qwv son dos uv -trayectorias internamente ajenas, con lo que concluimos la demostración de esta implicación.

Para 2 implica 3 no hay mucho que hacer, pues para cualesquiera dos vértices distintos u y v de G , la unión de dos uv -trayectorias internamente ajenas resulta en un ciclo que contiene a u y a v .

Para 3 implica 4, sean u un vértice y vw una arista de G . Por hipótesis, u y w están en un ciclo común C . Si v es un vértice de C , entonces podemos suponer sin pérdida de generalidad que C empieza en u y v aparece antes que w en C , por lo que $CvwC$ es un ciclo que contiene a u y a vw .

Si v no es un vértice de C , entonces, por hipótesis, u y v están en un ciclo común C' . Si w es un vértice de C' , entonces podemos repetir el argumento anterior, por lo que podemos suponer que C' no pasa por w . El ciclo C puede dividirse en dos uw -trayectorias P y Q internamente ajenas, y por las características de C' , existe una uv -trayectoria P' que no pasa por w . Así, tenemos la misma configuración que en 1 implica 2, por lo que podemos encontrar dos uv -trayectorias internamente ajenas, una de las cuales usa a la arista vw ; la unión de estas trayectorias es el ciclo buscado.

El resto de las implicaciones se dejan como ejercicio. \square

3.3. Descomposiciones por orejas

Las únicas gráficas no separables y acíclicas son K_1 y K_2 . Una descomposición por orejas de una gráfica no separable G es una descripción recursiva de G que empieza en un ciclo arbitrario y construye a G agregando trayectorias entre vértices previamente agregados.

Si G es una gráfica y F es una subgráfica de G , entonces una oreja de F en G es una trayectoria no trivial de G , cuyos vértices no tienen lazos, y cuyos extremos están en F , pero tal que sus vértices internos no.

Proposición 3.3.1. *Sea G una gráfica no separable. Si F es una subgráfica propia y no trivial de G , entonces F tiene una oreja en G .*

Demostración. Si F es una subgráfica generadora de G , entonces el conjunto $E_G - E_F$ es no vacío, pues F es una subgráfica propia de G . Cualquier arista en $E_G - E_F$ es una oreja de F en G . Por lo tanto, podemos suponer que F no es generadora.

Como G es conexa, existe una arista xy en G con $x \in V_F$ y $y \in V_G - V_F$. Además, G es no separable, por lo que $G - x$ es conexa, de donde se tiene la existencia de una $y(F - x)$ -trayectoria Q en $G - x$. La trayectoria xyQ es una oreja de F . \square

Proposición 3.3.2. *Sea G una gráfica, y sea F una subgráfica propia y no separable de G . Si P es una oreja de F , entonces $F \cup P$ es no separable.*

Demostración. Sea $G' = F \cup P$, y supongamos que $P = (x_0, \dots, x_k)$. Como F es no separable y tiene al menos dos vértices, sabemos que no tiene lazos. Así, basta demostrar que G' no tiene vértices de corte. Sea v un vértice de G' ; notemos que $G' - v = (F - v) \cup (P - v)$ y que $F - v$ y $P - v$ tienen al menos un vértice en común. Consideremos tres casos.

Si $v \in \{x_0, x_k\}$, entonces, al ser F no separable, $F - v$ es conexa. Como v es un vértice terminal de P , también $P - v$ es conexa. Luego, $G' - v$ es conexa.

Si $v \in V_F - \{x_0, x_k\}$, entonces $P - v = P$, que es conexa, y $F - v$ es conexa, por lo que nuevamente $G' - v$ resulta ser conexa.

Finalmente, si $v = x_i$ con $i \in \{1, \dots, k-1\}$, entonces $P - v$ es inconexa, y tiene exactamente dos componentes conexas G_1 y G_2 , con $G_1 = x_0 P x_{i-1}$ y $G_2 = x_{i+1} P x_k$. Así, $G' - v = F \cup G_1 \cup G_2$. Como F es conexa y comparte un vértice con G_1 y un vértice con G_2 , se sigue que $G' - v$ es conexa. \square

descomposición por orejas Una sucesión anidada de gráficas es una sucesión (G_0, \dots, G_k) de gráficas tales que $G_i \subseteq G_{i+1}$ para $i \in \{0, \dots, k-1\}$. Una **descomposición por orejas** de una gráfica no separable G es una sucesión anidada (G_0, \dots, G_k) de gráficas no separables tales que:

1. G_0 es un ciclo,
2. $G_{i+1} = G_i \cup P_i$, donde P_i es una oreja de G_i en G para $i \in \{0, \dots, k-1\}$,
3. $G_k = G$.

Teorema 3.3.3. *Toda gráfica no separable G , distinta de K_1 y K_2 , tiene una descomposición por orejas.*

Demostración. Recordemos que toda gráfica no separable distinta de K_1 y K_2 tiene al menos un ciclo C , que usamos como base para la siguiente sucesión recursiva.

- Sea $G_0 = C$.
- Si $G_i = G$, la recursión termina.
- Sea P_i una oreja de G_i en G , y sea $G_{i+1} = G_i \cup P_i$.

La sucesión está bien definida, pues por la Proposición 3.3.1, siempre que $G_i \neq G$, tenemos que P_i existe. Por otro lado, la Proposición 3.3.2 garantiza que la clausula de escape se cumple en algún momento, y por lo tanto la recursión termina.

Por definición, la sucesión obtenida es una descomposición por orejas de G . \square

3.4. Teorema de Menger

conexidad local La conexidad de una gráfica será definida en términos de su conexidad local, es decir, qué tan difícil es desconectar a un par dado de vértices. La **conexidad local** entre vértices distintos x y y de una gráfica G , $p(x, y)$, se define como el máximo número de xy -trayectorias internamente ajenas; la conexidad local no está definida cuando $x = y$.

conexidad Dado un número natural k , una gráfica no trivial es k -conexa si $p(u, v) \geq k$ para cualesquiera dos vértices u y v . Por convención, una gráfica trivial es 0-conexa, pero no es k conexa para toda $k > 0$. La **conexidad** κ_G de la gráfica G es el máximo

valor de k para el que G es k -conexa. Así, para una gráfica no trivial G , se tiene que

$$\kappa_G = \min \{p(u, v) : u, v \in V, u \neq v\}.$$

Una gráfica es 1-conexa si y sólo si es no trivial y conexa, y tiene conexidad cero si y sólo si es trivial o inconexa. Las gráficas no separables con al menos tres vértices son 2-conexas, y recíprocamente, las gráficas 2-conexas y sin lazos son no separables.

Como se discutió al principio del capítulo, se espera obtener una dualidad entre la conexidad de una gráfica, definida en términos del número de trayectorias internamente ajenas entre pares de vértices, y el número de vértices que podemos borrar de la gráfica sin que ésta se desconecte. Resulta natural considerar una noción de conexidad que dependa de cortes por vértices en lugar de trayectorias internamente ajenas. Sin embargo, nos enfrentamos a una pequeña dificultad, las gráficas completas no tienen cortes por vértices, por lo que una definición de este tipo para gráficas completas necesariamente se dará por convención.

Cualesquiera dos vértices distintos x y y en K_n están unidos por una arista y $n - 2$ trayectorias internamente ajenas de longitud 2. Por lo tanto $p(x, y) = n - 1$, de donde $\kappa_{K_n} = n - 1$ para cualquier $n \geq 2$; por convención, $\kappa_{K_1} = 0$. Para una gráfica de orden n en la que existe un par de vértices no adyacentes, digamos x y y , hay a lo más $n - 2$ xy -trayectorias internamente ajenas, por lo que su conexidad, κ , no puede exceder $n - 2$.

Sean G una gráfica y x y y vértices no adyacentes en G . Un **xy -corte por vértices** es un subconjunto S de $V - \{x, y\}$ tal que x y y pertenecen a componentes distintas de $G - S$; se dice que el conjunto S **separa** a x de y . La mínima cardinalidad de un xy -corte por vértices se denota por $c(x, y)$. La función c , llamada la función de corte local de G , no está definida si $x = y$ o si x y y son adyacentes. Recordemos que un corte por vértices de G es un corte por vértices para alguna pareja de vértices no adyacentes de G , y un corte por vértices de cardinalidad k es un k -corte por vértices. Una gráfica completa no tiene cortes por vértices, y de hecho, las únicas gráficas que no tienen cortes por vértices son aquellas que contienen subgráficas completas generadoras. El resultado principal de la esta sección demuestra que si una gráfica G tiene al menos un par de vértices no adyacentes, entonces la mínima cardinalidad de un corte por vértices en G , es igual a la conexidad de G . La versión del Teorema de Menger que se demostrará en este capítulo afirma que esto último es cierto localmente, es decir, para cualquier par de vértices no adyacentes x y y , se tiene que $p(x, y) = c(x, y)$.

En la demostración del siguiente teorema se utilizará la operación de contracción de un conjunto de vértices en una gráfica. Hay distintas formas de definir esta operación, dependiendo de si se permiten lazos o aristas múltiples; en este texto se presentará una de las versiones más sencillas que devuelve una gráfica simple al aplicársela a una gráfica simple. Si G es una gráfica y X un subconjunto de V , la gráfica G/X obtenida al **contraer** al conjunto X es la gráfica obtenida a partir de G al borrar todas las aristas con ambos extremos dentro de X , identificar a todos los

vértices de X en un solo vértice, y borrar las aristas múltiples que pudieran haberse creado.

Teorema 3.4.1 (Teorema de Menger). *Sea G una gráfica, y sean x y y vértices no adyacentes de G . El máximo número de xy -trayectorias internamente ajenas es igual al mínimo número de vértices en un xy -corte por vértices, es decir,*

$$p(x, y) = c(x, y).$$

Demostración. Por inducción sobre $|E|$. Por conveniencia, sea $k = c_G(x, y)$. Para cualquier familia \mathcal{P} de xy -trayectorias internamente ajenas, cualquier xy -corte por vértices debe incluir al menos un vértice en cada uno de los elementos de \mathcal{P} , por lo que todo xy -corte por vértices tiene al menos $|\mathcal{P}|$ vértices, es decir, $p_G(x, y) \leq k$. Por lo tanto, basta demostrar que $p_G(x, y) \geq k$. Podemos suponer que existe al menos una arista e con extremos u y v , que no es incidente en x ni en y ; de lo contrario, toda xy -trayectoria tiene longitud 2, y el resultado se sigue fácilmente.

Sea H la subgráfica $H = G - e$ de G . Como H es una subgráfica de G , se tiene que $p_G(x, y) \geq p_H(x, y)$. Por hipótesis inductiva tenemos que $p_H(x, y) = c_H(x, y)$. Además, como cualquier xy -corte por vértices de H puede extenderse a un xy -corte por vértices de G al añadirle un extremo de e , tenemos que $c_G(x, y) \leq c_H(x, y) + 1$. Por lo tanto, tenemos que

$$p_G(x, y) \geq p_H(x, y) = c_H(x, y) \geq c_G(x, y) - 1 = k - 1.$$

Si alguna de las desigualdades es estricta, tenemos que $p_G(x, y) \geq k$, como queríamos demostrar. Por lo tanto, podemos suponer que todas las desigualdades son igualdades y, en particular, $c_H(x, y) = k - 1$. Sea S un xy -corte por vértices mínimo en H , con $S = \{v_1, \dots, v_{k-1}\}$, sea X el conjunto de vértices alcanzables desde x en $H - S$ y sea Y el conjunto de vértices alcanzables desde y en $H - S$. Como $|S| = k - 1$, el conjunto S no es un xy -corte por vértices de G , por lo que hay una xy -trayectoria en $G - S$, que necesariamente usa a la arista e (pues dicha trayectoria no existe en H). Por lo tanto, podemos suponer sin pérdida de generalidad que $u \in X$ y $v \in Y$.

Consideremos la gráfica G/Y obtenida al contraer al conjunto Y a un vértice y . Notemos que todo xy -corte por vértices T en G/Y es un xy -corte por vértices en G , pues si P fuera una xy -trayectoria en G que no interseca al conjunto T , entonces P/Y sería un xy -camino en G/Y que no interseca al conjunto T . Por lo tanto, $c_{G/Y}(x, y) \geq k$. Por otro lado, $c_{G/Y}(x, y) \leq k$, pues $S \cup \{u\}$ es un xy -corte por vértices en G/Y . Por lo tanto $S \cup \{u\}$ es un xy -corte por vértices mínimo en G/Y . Por hipótesis de inducción, existen k xy -trayectorias internamente ajenas P_1, \dots, P_k en G/Y , y cada vértice de $S \cup \{u\}$ interseca exactamente a una de ellas. Podemos suponer sin pérdida de generalidad que $v_i \in V_{P_i}$ para $i \in \{1, \dots, k-1\}$, y que $u \in V_{P_k}$. Análogamente, existen k xy -trayectorias internamente ajenas Q_1, \dots, Q_k en G/X , obtenidas al contraer al conjunto X a x , y tales que $v_i \in V_{Q_i}$ para $i \in \{1, \dots, k-1\}$ y $v \in V_{Q_k}$.

Por lo tanto, existen k xy -trayectorias internamente ajenas en G , a saber, $xP_i v_i Q_i y$, para $i \in \{1, \dots, k-1\}$ y $xP_k uv Q_k y$. \square

Como consecuencia directa del Teorema 3.4.1 tenemos que

$$\min \{p(u, v) : u, v \in V, u \neq v, uv \notin E\} = \min \{c(u, v) : u, v \in V, u \neq v, uv \notin E\}.$$

Si G es una gráfica con al menos un par de vértices no adyacentes, entonces el lado derecho de la ecuación anterior es la cardinalidad de un corte por vértices mínimo en G . Sin embargo, esta ecuación no nos permite concluir que la conexidad de G es igual a la cardinalidad de un corte por vértices mínimo en G , pues, de acuerdo a la definición de conexidad, κ es el mínimo valor de $p(u, v)$ tomado sobre todas las parejas de vértices distintos u, v , no necesariamente no adyacentes (como en la igualdad de arriba). Sin embargo, el siguiente resultado, demostrado originalmente por Whitney, afirma que el mínimo valor de $p(u, v)$ es igual al considerarlo sobre parejas de vértices distintos y no adyacentes, y al considerarlo sobre parejas de vértices distintos (adyacentes o no).

Teorema 3.4.2. *Si G tiene al menos un par de vértices no adyacentes, entonces*

$$\kappa_G = \min \{p(u, v) : u, v \in V, u \neq v, uv \notin E\}.$$

Demostración. Ni los lazos ni las aristas múltiples tienen efecto sobre la conexidad local, por lo que podemos suponer que la gráfica G es simple.

Por definición, tenemos que $\kappa_G = \min \{p(u, v) : u, v \in V, u \neq v\}$, por lo que podemos suponer que este mínimo se alcanza en la pareja x, y , i.e., $\kappa_G = p(x, y)$. Si x y y no son adyacentes, no hay nada que demostrar. Supongamos entonces que x y y son adyacentes.

La idea principal de la demostración es encontrar un vértice z que no sea adyacente a x y tal que $p(x, z) = p(x, y)$.

Sea H la subgráfica de G , obtenida al borrar la arista xy de G . Claramente $p_G(x, y) = p_H(x, y) + 1$. Además, por el Teorema de Menger, $p_H(x, y) = c_H(x, y)$. Sea X un xy -corte por vértices mínimo en H , de tal suerte que $|X| = p_H(x, y) = c_H(x, y)$, y $p_G(x, y) = |X| + 1$. Si $V - X = \{x, y\}$, entonces

$$\kappa_G = p_G(x, y) = |X| + 1 = (n - 2) + 1 = n - 1,$$

lo cual no puede suceder, pues G no es completa.

Por lo tanto, podemos suponer que $V - X$ tiene al menos tres vértices, x, y, z . Podemos suponer también, sin pérdida de generalidad, que x y z están en componentes distintas de $H - X$. Por lo tanto, x y z son no adyacentes en G y $X \cup \{y\}$ es un corte por vértices de G que separa a x de z . Por lo tanto

$$c(x, z) \leq |X \cup \{y\}| = p(x, y).$$

Por el Teorema de Menger, se tiene que $p(x, z) = c(x, z)$. Luego, $p(x, z) \leq p(x, y)$, y por la elección de x y y , obtenemos $p(x, z) = p(x, y) = \kappa_G$. Como x y z son no adyacentes, entonces

$$\kappa_G = p(x, z) = \min \{p(u, v) : u, v \in V, u \neq v, uv \notin E\}.$$

□

Por lo tanto, es posible concluir de los Teoremas 3.4.1 y 3.4.2 que si G es una gráfica con al menos un par de vértices no adyacentes, entonces

$$\kappa_G = \min \{c(u, v) : u, v \in V, u \neq v, uv \notin E\}.$$

Recordando que las gráficas no separables con al menos tres vértices son 2-conexas, se puede interpretar que al Teorema de Menger como una generalización de la propiedad 2 del Teorema de Caracterización de Bloques (Teorema 3.2.2). La propiedad 3 también admite una generalización al caso de las gráficas k -conexas. Es posible demostrar que si G es una gráfica k -conexa, entonces para cualquier conjunto de a lo más k vértices, existe un ciclo que los contiene a todos. Para obtener la demostración de este resultado, vale la pena desarrollar algunas herramientas derivadas del Teorema de Menger, que tienen una gran utilidad por sí solas.

Lema 3.4.3. *Sea G una gráfica k -conexa. Si H es una gráfica obtenida a partir de G al agregar un vértice nuevo y y hacerlo adyacente a al menos k vértices de G , entonces H es k -conexa.*

Demostración. Si H contiene una subgráfica generadora completa, el resultado se sigue trivialmente, pues $|V_H| \geq k + 1$.

Basta demostrar que si S es un subconjunto de V_H de cardinalidad $k - 1$, entonces $H - S$ es conexa. Supongamos primero que $y \in S$. Entonces $H - S = G - (S - \{y\})$. Por hipótesis, G es k -conexa, y como $|S - \{y\}| = k - 2$, tenemos que $H - S$ es conexa.

Si por otro lado $y \notin S$, como por hipótesis y tiene al menos k vecinos en V_G y $|S| = k - 1$, entonces y debe tener al menos un vecino z en $V_G - S$. Como G es k -conexa, entonces $G - S$ es conexa. Además, z es un vértice de $G - S$, por lo que zy es una arista de $H - S$, de donde se obtiene que $(G - S) + yz$ es una subgráfica generadora conexa de $H - S$. Luego, $H - S$ es conexa. □

La siguiente proposición es una consecuencia sencilla del lema anterior. En algunos textos este resultado se enuncia como el Teorema de Menger.

Proposición 3.4.4. *Sea G una gráfica k -conexa. Si X y Y son subconjuntos de V de cardinalidad al menos k , entonces existe una familia de k XY -trayectorias internamente ajenas en G .*

Demostración. Sea H la gráfica obtenida de G al agregar dos nuevos vértices, x y y , y hacer adyacente a x a cada vértice en X , y a y a cada vértice en Y . Por el Lema 3.4.3, H es k -conexa, por lo que el Teorema de Menger implica la existencia de k xy -trayectorias internamente ajenas en H . Al eliminar a x y y de cada una de estas trayectorias, obtenemos k trayectorias en G , digamos Q_1, \dots, Q_k , cada una con vértice inicial en X y vértice terminal en Y .

Para cualquier $i \in \{1, \dots, k\}$, si u es el último vértice de Q_i en X y v es el primer vértice de Q_i en Y , entonces uQ_iv es una XY -trayectoria contenida en Q_i . Así, tenemos que existen k XY -trayectorias ajenas dos a dos en G . \square

Dados un vértice x y un subconjunto Y de $V - \{x\}$, un k -**abanico** de x a Y es k -*abanico* una familia de k xY -trayectorias internamente ajenas cuyos vértices terminales son distintos.

Lema 3.4.5 (Lema del Abanico). *Sea G una gráfica k -conexa. Si x es un vértice de G , y Y es un subconjunto de $V - \{x\}$ de cardinalidad al menos k , entonces existe un k -abanico de x a Y .*

A continuación se presenta la generalización arriba mencionada del punto 3 del Teorema 3.2.2.

Teorema 3.4.6. *Sean k un entero con $k \geq 2$, y G una gráfica k -conexa. Si S es un subconjunto de V de cardinalidad k , entonces existe un ciclo en G que pasa por todos los vértices en S .*

Demostración. Por inducción sobre k . Por el Teorema 3.2.2, el enunciado es cierto para $k = 2$, por lo que podemos suponer que $k \geq 3$. Sea x un vértice en S , y definamos a T como $T = S - x$. Como G es k -conexa, también es $(k - 1)$ -conexa, por lo que se sigue de la hipótesis inductiva que hay un ciclo C en G que pasa por todos los vértices de T . Si $x \in V_C$, entonces C es el ciclo que buscamos. Luego, podemos suponer que $x \notin V_C$. Si $|V_C| \geq k$, el Lema del Abanico (Lema 3.4.5) garantiza la existencia de un k -abanico de x a V_C en G . Como $|T| = k - 1$, el conjunto T divide a C en $k - 1$ segmentos ajenos por aristas. Por el Principio de las Casillas de Dirichlet, hay dos trayectorias del abanico, digamos P y Q , que terminan en el mismo segmento de C . La subgráfica $C \cup P \cup Q$ contiene tres ciclos, uno de los cuales incluye a S . Si $|V_C| = k - 1$, el Lema del Abanico implica la existencia de un $(k - 1)$ -abanico de x a V_C en G , donde cada vértice de C es el extremo de una trayectoria, por lo que podemos usar el mismo argumento que en el caso anterior. \square

3.5. Conexidad por aristas

La noción de corte por vertices generaliza a la de vértice de corte. De manera análoga, existe la noción de corte por aristas, que generaliza a la de puente. Pero tal como se hizo en el caso de conexidad por vértices, es posible introducir nociones

de conexidad local y separadores por aristas, para después obtener una versión para aristas del Teorema de Menger que las relacione.

conexidad local por aristas Para una gráfica G y sean x y y vértices de G . La **conexidad local por aristas** entre x y y , $p'(x, y)$, es el máximo número de xy -trayectorias ajenas por aristas en G , si $x \neq y$, y está indefinido cuando $x = y$. Si k es un número natural y G es no trivial, entonces G es k -conexa por aristas si $p'(u, v) \geq k$ para cualesquiera vértices distintos u y v de G . Por convención, una gráfica trivial es 0-conexa y 1-conexa por aristas, pero no es k -conexa por aristas para cualquier $k > 1$. La **conexidad por aristas** κ'_G de G es el máximo valor de k para el que G es k -conexa por aristas.

conexidad por aristas Para vértices distintos x y y en una gráfica G , el corte por aristas $\partial(X)$ **separa** a x de y si $x \in X$ y $y \in V - X$. Un **xy -corte por aristas** es un corte por aristas que separa a x de y . La mínima cardinalidad de un xy -corte por aristas se denotará por $c'(x, y)$. Recordemos que un k -corte por aristas es un corte por aristas $\partial(X)$, donde X no es ni vacío ni igual a V , y $|\partial(X)| = k$, i.e., un corte por aristas con k elementos que separa a algún par de vértices. Toda gráfica no trivial tiene al menos un corte por aristas, por lo que el siguiente resultado, que es la versión por aristas del Teorema de Menger, afirma que la conexidad por aristas κ'_G de una gráfica no trivial G es igual al menor entero no negativo k para el que G tiene un k -corte por aristas.

Teorema 3.5.1. *Si G es una gráfica, entonces para cualesquiera vértices x y y de G , se tiene que*

$$p'(x, y) = c'(x, y).$$

Demostración. Considere la gráfica $G' = (V_G \cup \{x', y'\}, E_G \cup \{xx', yy'\})$. Por el Teorema de Menger $p'_G(x, y) = p_{L_{G'}}(xx', yy') = c_{L_{G'}}(xx', yy') = c'_G(x, y)$. \square

Gracias a los Teoremas 3.4.1 y 3.5.1, los parámetros κ y κ' pueden ser considerados en función de cardinalidades mínimas de cortes por vértices y por aristas, respectivamente. Desde este punto de vista, sabemos que, para una gráfica conexa con al menos tres vértices, la existencia de un puente ($\kappa' = 1$) implica la existencia de un vértice de corte ($\kappa = 1$). Aunque estos parámetros no tienen por qué coincidir en general, el siguiente resultado muestra que sí existe una relación entre los valores de κ y κ' .

Proposición 3.5.2. *Para cualquier gráfica no trivial G*

$$\kappa \leq \kappa' \leq \delta.$$

Demostración. Si G es inconexa, $\kappa' = 0 \leq \delta$. De otro modo, el conjunto de aristas incidentes en cualquier vértice de grado δ es un δ -corte por aristas de G , por lo que $\kappa' \leq \delta$.

Para la desigualdad restante, procedemos por inducción sobre κ' . Si $\kappa' = 0$, entonces G es inconexa, en cuyo caso $\kappa = 0$. Supongamos entonces que $\kappa'_G = k$ con $k > 0$,

y sea e una arista en un k -corte por aristas de G . Si $H = G - e$, entonces $\kappa'_H = k - 1$, por lo que, a partir de la hipótesis de inducción concluimos que $\kappa_H \leq k - 1$.

Cuando G no es simple, existe la posibilidad de que H contenga una gráfica completa como subgráfica generadora, en cuyo caso G también la contiene, y tenemos que

$$\kappa_G = \kappa_H \leq k - 1.$$

En otro caso, H tiene un corte por vértices S con κ_H elementos. Como $H - S$ es inconexa, o bien $G - S$ es inconexa, en cuyo caso

$$\kappa_G \leq \kappa_H \leq k - 1,$$

o bien $G - S$ es conexa y e es un puente de $G - S$. En este último caso tenemos que, o bien $G - S$ tiene únicamente dos vértices, y por lo tanto

$$\kappa_G \leq |V| - 1 = \kappa_H + 1 \leq k,$$

o $G - S$ tiene un vértice de corte v , por lo que $S \cup \{v\}$ es un corte por vértices de G , y

$$\kappa_G \leq \kappa_H + 1 \leq k.$$

En cualquier caso obtenemos que $\kappa_G \leq k = \kappa'_G$, lo que concluye la demostración por inducción. \square

Capítulo 4

Gráficas eulerianas y hamiltonianas

4.1. Gráficas eulerianas

Un paseo T en una gráfica G es un **paseo euleriano** si $E_T = E_G$. Un **circuito euleriano** es un paseo euleriano cerrado, es decir, un paseo cerrado que usa todas las aristas de G . Una gráfica es **euleriana** si tiene un circuito euleriano y no tiene vértices aislados.

No es difícil encontrar una condición necesaria para que una gráfica sea euleriana. Si G tiene un circuito euleriano C , y al recorrerlo se van eliminando las aristas que se utilizan, entonces al terminar de recorrerlo se deben de haber eliminado todas las aristas de la gráfica. Para que esto sea posible, es necesario que siempre que se llega a un vértice, como se eliminó la arista por la que se llegó, debe haber una arista distinta por la que se avanza hacia el próximo vértice. Así, cada vez que C pasa por un vértice intermedio, se utilizan dos aristas distintas, una para “entrar” y una para “salir”, por lo que el número total de aristas incidentes en cada vértice debe ser par. Como C es cerrado, cada vértice puede considerarse un vértice intermedio al iniciar el recorrido de C en un vértice distinto. Por lo tanto, si G es euleriana, entonces es par¹.

Al tener una condición necesaria tan simple, siempre vale la pena preguntarse si también es suficiente, i.e., si una gráfica es par, ¿entonces es euleriana? Afortunadamente, al agregar la hipótesis de conexidad, en este caso la respuesta es sí. Más aún, el que todos los vértices de una gráfica tengan grado par implica otras propiedades interesantes sobre su estructura, por lo que es posible proponer fácilmente al menos dos caracterizaciones de las gráficas eulerianas.

Teorema 4.1.1. *Sea G una gráfica conexa. Son equivalentes:*

1. G es euleriana.

¹Una gráfica es par si cada vértice tiene grado par.

2. G es par.

3. G tiene una descomposición en ciclos.

Demostración. Si G tiene orden 1, entonces todas sus aristas son lazos, y el resultado se sigue trivialmente. Luego, podemos suponer que G es no trivial.

Sea C un circuito euleriano de G , con $C = (x_0, \dots, x_k, x_0)$. Sea v un vértice de G , y supongamos que $v \neq x_0$. Sea S el subconjunto de $\{0, \dots, k\}$ de índices tales que $v = x_i$. Como G es conexa, $S \neq \emptyset$. Por ser C un paseo, para cada $i \in S$, las aristas $x_{i-1}x_i$ y $x_i x_{i+1}$ aparecen una única vez en C , y como C es euleriano, el conjunto $\partial(v)$ de todas las aristas incidentes en v cumple $\partial(v) = \bigcup_{i \in S} \{x_{i-1}x_i, x_i x_{i+1}\}$, por lo que $d(v) = 2|S|$. Si $v = x_0$, como C es cerrado, podemos suponer que empieza en un vértice distinto, y aplicar el mismo argumento que antes.

Para 2 implica 3, procederemos por inducción sobre el número de aristas. Como G tiene al menos dos vértices, al ser conexa y par, tenemos que $\delta \geq 2$, por lo que tiene un ciclo C . Sean G_1, \dots, G_k las componentes conexas de $G - C$. Para cada vértice v de G se tiene que $d_{G-C}(v) = d_G(v)$ o $d_{G-C}(v) = d_G(v) - 2$, por lo que $G - C$ es par. Por hipótesis inductiva, G_i tiene una descomposición en ciclos, \mathcal{C}_i , para cada $i \in \{1, \dots, k\}$. Al ser componentes conexas distintas, y por la definición de descomposición, tenemos que $E_C \cup \bigcup_{i=1}^k \mathcal{C}_i$ es una descomposición en ciclos de G .

Sea C_1, \dots, C_k una descomposición en ciclos de G . Demostraremos que G es euleriana por inducción sobre k . Si $k = 1$, entonces, al ser G conexa, G es un ciclo, y trivialmente euleriana. Supongamos que $k > 1$, y sean G_1, \dots, G_ℓ las componentes conexas de $G - C_k$. Por hipótesis inductiva, G_i es euleriana para cada $i \in \{1, \dots, \ell\}$; sea T_i un circuito euleriano de G_i . Como G es conexa, C_k intersecta a cada componente de $G - C_k$. Sea $C_k = (v_1, \dots, v_{n_k}, v_1)$, y sea v_{i_j} el primer vértice de C_k que intersecta a G_j , para cada $j \in \{1, \dots, \ell\}$. El paseo

$$v_1 C_k v_{i_1} T_1 v_{i_1} C_k v_{i_2} T_2 v_{i_2} \cdots v_{i_\ell} T_\ell v_{i_\ell} C_k v_1$$

es un circuito euleriano de G . □

No es difícil observar que al eliminar la hipótesis de conexidad del teorema anterior todavía es posible demostrar que una gráfica es par si y sólo si tiene una descomposición en ciclos. La demostración de 2 implica 3 en el Teorema 4.1.1 es válida aún cuando la gráfica no sea conexa, y la recíproca es directa, al notar que el grado de cada vértice es $2k$, donde k es el número de ciclos de la composición a los que pertenece.

Por otro lado, el Teorema 4.1.1 sugiere un algoritmo de tiempo lineal para verificar si una gráfica es euleriana. Basta recorrer la sucesión de grados de la gráfica, si todos los grados son pares, entonces la gráfica es euleriana, y en otro caso, no lo es. Este procedimiento claramente puede llevarse a cabo en tiempo $O(|V|)$. Si se quisiera encontrar un circuito euleriano en una gráfica que ya se sabe euleriana, sería necesario, al menos, considerar a cada arista de la gráfica para agregarla al

circuito. Así, el tiempo mínimo de ejecución de un algoritmo para encontrar un circuito euleriano en una gráfica euleriana es $O(E)$. Existen diversos algoritmos que logran alcanzar esta cota inferior, por ejemplo, el algoritmo de Hierholzer.

4.2. El algoritmo de Hierholzer

A continuación se presenta un algoritmo eficiente para encontrar un circuito euleriano en una gráfica euleriana. Existen algoritmos que toman como entrada una gráfica arbitraria, y devuelven un circuito euleriano, en caso de que exista uno, o responden que la gráfica no es euleriana. Debido a la caracterización de las gráficas eulerianas dada en la sección anterior, es posible verificar muy eficientemente si una gráfica es euleriana o no, por lo que no tiene un costo significativo realizar dicha verificación antes de ejecutar el algoritmo para buscar un circuito euleriano.

Para poder realizar un paso fundamental del algoritmo de forma eficiente, es necesario utilizar las estructuras de datos adecuadas, por lo que es lo primero que se introduce.

Algoritmo 4.2.1 (Hierholzer). *Sea $G = (V, E)$ un gráfica conexa euleriana, dirigida o no, con $V = \{1, \dots, n\}$. Además, sea s un vértice de G . Construimos un circuito euleriano K (que será dirigido si G lo es) con vértice inicial s .*

Estructuras de datos necesarias.

1. Listas de incidencia A_1, \dots, A_n ; para cada arista e , denotamos el vértice final de e mediante $\text{end}(e)$.
2. Listas K y C para guardar las sucesiones de aristas que forman un paseo cerrado. Utilizaremos listas doblemente ligadas; es decir, cada elemento de la lista está ligado a su predecesor y su sucesor, de manera que éstos pueden ser encontrados fácilmente.
3. Una función booleana *used* sobre el conjunto de vértices, donde $\text{used}(v)$ tiene valor **true** si v ocurre en K , y **false** de otro modo, y una lista L que contiene todos los vértices v tales que $\text{used}(v) = \text{true}$.
4. Para cada vértice v , un apuntador $e(v)$, que inicialmente está indefinido, y después apunta a una arista en K que empieza en v .
5. Una función booleana *new* sobre el conjunto de aristas, donde $\text{new}(e)$ vale **true** si e no está contenida en el paseo cerrado.
6. Variables u, v para vértices y e para aristas.

Algoritmo 7: TRACE($v, \text{new}; C$)

Input: Una gráfica euleriana G , un vértice v de G y un paseo cerrado C , inicialmente vacío.

Output: Un paseo cerrado C de aristas no utilizadas, que empieza en v .

```
1 while  $A_v \neq \emptyset$  do
2   eliminar la primera arista  $e$  de  $A_v$ ;
3   if  $\text{new}(e) = \text{true}$  then
4     añadir  $e$  al final de  $C$ ;
5     if  $e(v)$  no está definida then
6       | asignar la posición en la que  $e$  ocurre en  $C$  a  $e(v)$ ;
7     end
8      $\text{new}(e) \leftarrow \text{false}$ ,  $v \leftarrow \text{end}(e)$ ;
9     if  $\text{used}(v) = \text{false}$  then
10      | añadir  $v$  a  $L$ ;
11      |  $\text{used}(v) \leftarrow \text{true}$ ;
12    end
13  end
14 end
15 return  $C$ 
```

El Algoritmo TRACE($v, \text{new}; C$) (Algoritmo 7) construye un paseo cerrado C que consiste de aristas que no hayan sido utilizadas todavía, empezando en el vértice v .

A continuación demostraremos que el Algoritmo 7 es correcto. Este hecho será de utilidad para demostrar que el Algoritmo 8 también es correcto.

Lema 4.2.2. *El Algoritmo TRACE es correcto, es decir, la sucesión que devuelve el Algoritmo 7 es un paseo cerrado de aristas nuevas, que empieza en el vértice de entrada v . Más aún, todas las aristas nuevas que inciden en v se utilizan en dicho paseo.*

Demostración. Veamos primero que el algoritmo siempre termina su ejecución. Para este fin basta verificar que cada vez que se ejecuta el ciclo principal se elimina una arista de la lista de incidencias del vértice activo. Como el ciclo se rompe cuando la lista de adyacencias del vértice activo es vacía, y la cardinalidad de dicha lista decrece con cada ejecución, en un número finito de ejecuciones el vértice activo tendrá una lista de incidencias vacía. Por lo tanto, la ejecución de TRACE siempre termina.

El paseo construido por C no puede repetir aristas, ni utilizar aristas que no sean nuevas, ya que en la línea 3 se hace una verificación para ver que la arista que

está siendo procesada cumpla que $\text{new}(e)$ sea verdadero, y en la línea 8 se cambia el estado de este valor a falso. De aquí podemos concluir que C realmente es un paseo.

Falta verificar que es cerrado y que se utilizan todas las aristas nuevas que inciden en v . Utilizaremos la siguiente invariante de ciclo para demostrar las afirmaciones restantes. La sucesión C es un paseo para el que se cumple alguna de las siguientes condiciones:

1. El vértice inicial es igual al vértice activo, y el número de aristas nuevas incidentes en cada vértice es par.
2. El vértice inicial es distinto al vértice activo, y el número de aristas nuevas incidentes en cada vértice es par, excepto para el vértice inicial y el vértice activo que es impar.

Veamos primero que en caso de que la invariante de ciclo sea cierta, el paseo es cerrado, y todas las aristas nuevas incidentes en v son utilizadas en el mismo. Sea u el vértice activo en la iteración actual del algoritmo. Según la invariante, si $u \neq v$, entonces hay un número impar de aristas nuevas incidentes en u , por lo que hay al menos una, y la condición $A_u \neq \emptyset$ en la línea 1 se cumple. Al entrar al ciclo, el vértice activo cambia en la línea 8, por lo que la ejecución del algoritmo no puede terminar en un vértice distinto a v . Luego, al terminar la ejecución del algoritmo el paseo C es cerrado. Si en la ejecución actual hay alguna arista incidente en v nueva y sin utilizar, nuevamente la condición en la línea 1 se cumple, por lo que el algoritmo no puede terminar mientras haya aristas nuevas incidentes en v . Concluimos que todas las aristas nuevas incidentes en v deben ser utilizadas en C .

Para terminar, demostremos la invariante de ciclo propuesta. Cuando TRACE se ejecuta por primera vez, todas las aristas son nuevas, y al ser G euleriana, todos los vértices tienen grado par. Además, el vértice inicial es el vértice activo, por lo que se cumple la condición 1. Supongamos que el ciclo principal de TRACE ya se ha ejecutado con anterioridad, está por iniciar la siguiente ejecución, y se cumple la invariante de ciclo. En caso de que la condición en la línea 3 falle, el paseo no se modifica, y la invariante se sigue cumpliendo. Por tanto, podemos suponer que la arista que se eligió es nueva. Supongamos primero que se cumple el caso 1. Si ya no hay más aristas incidentes en v , entonces el algoritmo termina, no se modifica el paseo, y la invariante se sigue cumpliendo. En otro caso, se agrega la arista vu al paseo, misma que se marca como usada. Entonces, el número de aristas nuevas incidentes en u y en v disminuye en 1, y el vértice activo cambia a u , por lo que se cumple la condición 2 de la invariante de ciclo.

Cuando se cumple el caso 2, nuevamente tenemos dos subcasos. Primero, si el siguiente vértice activo es v , entonces, al marcar la arista uv como usada, el número de aristas nuevas incidentes en u y v se vuelve par, y el nuevo vértice activo es v , por lo que caemos en el caso 1. En el segundo caso, si el siguiente vértice activo es distinto de v , digamos w , entonces al marcar la arista uw como usada, el número de

aristas nuevas incidentes en u y w disminuye en uno, volviéndose par para el primero e impar para el segundo. Nuevamente, al terminar la ejecución del ciclo, los únicos vértices con un número impar de aristas nuevas incidentes en ellos son v y w (el vértice activo), por lo que estamos en el caso 1 de la invariante. \square

El Algoritmo $\text{EULER}(G, s; K)$ (Algoritmo 8) utiliza el Algoritmo TRACE para construir un circuito euleriano K , con vértice inicial s en una gráfica G .

Algoritmo 8: $\text{EULER}(G, s; K)$

Input: Una gráfica euleriana conexa G y un vértice s .

Output: Un circuito euleriano K con vértice inicial s .

```

1  $K \leftarrow \emptyset, L \leftarrow \emptyset;$ 
2 for  $v \in V$  do
3    $\text{used}(v) \leftarrow \text{false};$ 
4 end
5 for  $e \in E$  do
6    $\text{new}(e) \leftarrow \text{true};$ 
7 end
8  $\text{used}(s) \leftarrow \text{true};$ 
9 añadir  $s$  a  $L$ ;
10  $K \leftarrow \text{TRACE}(s, \text{new}; K);$ 
11 while  $L \neq \emptyset$  do
12   sea  $u$  el último elemento de  $L$ ;
13   eliminar  $u$  de  $L$ ;
14    $C \leftarrow \emptyset;$ 
15    $C \leftarrow \text{TRACE}(u, \text{new}; C);$ 
16   insertar  $C$  en la posición  $e(u)$  en  $K$ ;
17 end
18 return  $K$ 

```

Teorema 4.2.3. *El algoritmo de Hierholzer es correcto, es decir, la sucesión K que devuelve la Algoritmo 8 es un circuito euleriano de la gráfica que recibe como entrada.*

Demostración. En primer lugar, es necesario verificar que el algoritmo termina. Los únicos lugares donde el algoritmo podría ciclarse es en las llamadas a TRACE , o en el **while** de la línea 11. Por el Lema 4.2.2, TRACE siempre termina. Veamos entonces que el algoritmo no puede ciclarse en el **while**, ya que cada vértice de G entra a lo más una vez a L , pues sólo puede entrar cuando no ha sido usado, y al entrar a L se marca como usado. Además, al procesar un vértice en L , éste se elimina (línea 13). Por este motivo, el ciclo se realiza a lo más $|V|$ veces, a saber, cuando todos los vértices de la gráfica hayan sido procesados. En otras palabras, el algoritmo termina.

Se sigue del Lema 4.2.2 que, al invocar a TRACE desde un vértice, todas las aristas incidentes en el mismo serán utilizadas en un circuito que será agregado al circuito K . Por lo tanto, garantizar que TRACE se invoca usando a cada vértice de la gráfica como entrada es suficiente para garantizar que K utiliza a todas las aristas de G . Notemos que TRACE se invoca usando como entrada a cada vértice en L . Se sigue que podemos concluir si demostramos que todo vértice entra al menos una vez a L .

Inicialmente s entra a L en la línea 9, y el algoritmo termina cuando L es vacía. Supongamos entonces que L es no vacía, y hay al menos un vértice u para el que $\text{used}(u)$ es falso. Si X es el conjunto de los vértices x para los que $\text{used}(x) = \text{true}$, y definimos a Y como $V - X$, entonces $u \in Y$, y (X, Y) es una partición de V . Como G es conexa, existe una arista xy con $x \in X$ y $y \in Y$. Afirmamos que $x \in L$, pues todos los vértices que ya han sido utilizados entraron a L , y la única forma de que un vértice z salga de L es a través de la línea 13 del algoritmo, lo que implica que TRACE se invocó en la línea 15 usando a z como entrada. Pero en este caso, el Lema 4.2.2 implica que todas las aristas incidentes en z han sido utilizadas en el circuito, y por lo tanto, todos los vecinos de z han sido marcados como usados y agregados a L . Como x tiene un vecino que no ha sido usado, concluimos que x está en L . Por lo tanto, en un número finito de pasos, x será el último elemento de L , por lo que se hará una llamada a TRACE usando a x como entrada, y a más tardar en ese momento el vértice y entrará a L (podría ser antes, si y es alcanzado a través de otro vecino, o si la arista xy se utiliza en una invocación anterior de TRACE). Por lo tanto, mientras haya vértices sin usar, siempre podemos garantizar que se agregarán vértices nuevos a L . Así, todo vértice debe usarse, y por lo tanto todo vértice entra a L exactamente una vez. \square

Teorema 4.2.4. *El algoritmo de Hierholzer se ejecuta en tiempo $O(|E|)$.*

Demostración. Empecemos notando que los ciclos `for` en las líneas 2 y 5 utilizan tiempo $O(|V|)$ y $O(|E|)$, respectivamente. Además, las instrucciones en todas las demás líneas del Algoritmo 8 se ejecutan en tiempo constante, excepto las llamadas al algoritmo TRACE. Es importante notar que la acción en la línea 16, i.e., insertar C en la posición $e(u)$ en K , puede realizarse en tiempo constante gracias a que contamos con el apuntador $e(u)$. También, podemos notar que las operaciones dentro del `while` de la línea 11 se llevarán a cabo $|V|$ veces, pues cada vértice de la gráfica entra una única vez a L , ya que para entrar a L es necesario que un vértice no esté marcado como usado, y cuando entra se marca. Por lo tanto, salvo las llamadas a TRACE, todas las instrucciones en este bloque se ejecutan en tiempo $O(|V|)$. Entonces, basta analizar cuál es el tiempo de ejecución del algoritmo TRACE.

Prácticamente todo el cuerpo del Algoritmo 7 está envuelto en el `while` de la línea 1. Más aún, todas las operaciones que se llevan a cabo dentro de este algoritmo se ejecutan en tiempo constante. Además, cada vez que pasamos por la línea 1, en la siguiente línea se elimina una arista de la lista de adyacencias del vértice activo v .

Por lo tanto, para cada vértice v de la gráfica, el cuerpo del Algoritmo 7 se ejecutará $d(v)$ veces. En total, el **while** principal se ejecuta $\sum_{v \in V} d(v) = 2|E|$ veces y en cada una de ellas se hace un número constante de operaciones, cada una de tiempo constante. Por lo tanto, el tiempo total de todas las llamadas que hace el Algoritmo 8 al Algoritmo 7 es $O(|E|)$.

De lo anterior, concluimos que el tiempo total de ejecución del algoritmo es $O(|V| + |E|)$. Sin embargo, como la gráfica de entrada es conexa, sabemos que $|E| \geq |V| - 1$, por lo que podemos expresar la complejidad como $O(|E|)$. \square

Como la construcción de un circuito euleriano requiere procesar todas las aristas de la gráfica, una cota inferior par el tiempo de ejecución de cualquier algoritmo que constuya un circuito euleriano es $O(|E|)$. Por lo tanto, se sigue del Teorema 4.2.4 que el Algoritmo 8 es óptimo.

4.3. Gráficas hamiltonianas

Conceptualmente, las gráficas hamiltonianas son semejantes a las gráficas eulerianas. Mientras que las gráficas eulerianas son aquellas en las que todas las aristas pueden recorrerse sin pasar más de una vez por cada arista, y terminando en el vértice en el que se empezó, las gráficas hamiltonianas son aquellas en las que todos los vértices pueden recorrerse sin pasar más de una vez por cada uno, y terminando en el vértice en el que se empezó. Sin embargo, como se estudiará en esta sección, los problemas correspondientes son sumamente diferentes; es fácil determinar computacionalmente si una gráfica arbitraria es euleriana, pero sigue siendo un problema abierto encontrar un algoritmo eficiente para determinar si una gráfica es hamiltoniana.

trayectoria ha-
miltoniana
ciclo hamiltoniano

Una **trayectoria hamiltoniana** en una gráfica G es simplemente una trayectoria generadora, es decir, una cuyo conjunto de vértices coincide con el conjunto de vértices de la gráfica. Análogamente, un **ciclo hamiltoniano** es un ciclo generador. Una gráfica que contiene una trayectoria hamiltoniana es llamada **trazable** y una que contiene un ciclo hamiltoniano es llamada **hamiltoniana**. A diferencia de las gráficas eulerianas, no se conocen condiciones suficientes y necesarias sencillas (ni útiles) para determinar cuando una gráfica es hamiltoniana. Sin embargo, es posible presentar diversas condiciones, necesarias o suficientes para la existencia de ciclos hamiltonianos.

Proposición 4.3.1. *Sea G una gráfica. Si G es hamiltoniana, entonces para cada subconjunto propio S de V , $c_{G-S} \leq |S|$.*

Demostración. Sea C un ciclo hamiltoniano de G , y sea S un subconjunto propio de V . No es difícil demostrar por inducción sobre $|S|$ que $c_{C-S} \leq |S|$. Además, como $C - S$ es una subgráfica generadora de $G - S$, tenemos que $c_{G-S} \leq c_{C-S}$. \square

Como cualquier condición necesaria, la contrapositiva de la Proposición 4.3.1 puede ser utilizada para verificar cuando una gráfica no es hamiltoniana. Como un ejemplo bastante sencillo, una gráfica G con un vértice de corte tiene un subconjunto de cardinalidad uno (el unitario del vértice de corte) tal que al eliminarlo, la gráfica resultante tiene al menos dos componentes conexas, la contrapositiva de la Proposición 4.3.1 afirma que G no tiene un ciclo hamiltoniano.

Por otro lado, como cualquier condición necesaria que no es también suficiente, existen gráficas que no son hamiltonianas, pero es imposible deducir esta información a partir de la Proposición 4.3.1. Un ejemplo clásico de este fenómeno es la gráfica de Petersen.

A continuación se presentan algunas condiciones suficientes para la existencia de ciclos hamiltonianos. Como una gráfica es hamiltoniana si y sólo si su gráfica simple subyacente lo es, es posible restringir el estudio de gráficas hamiltonianas a gráficas simples. El siguiente resultado fue demostrado por Dirac en 1951.

Teorema 4.3.2. *Sea G una gráfica simple. Si $|V| \geq 3$ y $\delta \geq \frac{|V|}{2}$, entonces G es hamiltoniana.*

Demostración. Procediendo por contradicción, supongamos que G es una gráfica con $|V| \geq 3$, $\delta \geq \frac{|V|}{2}$, no hamiltoniana y con el máximo número de aristas entre todas las posibles gráficas con estas tres propiedades, i.e., para cualesquiera vértices u y v de G tales que $uv \notin E$, la gráfica $G + uv$ es hamiltoniana. Como las gráficas completas de orden al menos tres son hamiltonianas, sabemos que G no es completa.

Sean u y v vértices no adyacentes en G . Por la elección de G , $G + uv$ es hamiltoniana. Además, como G no es hamiltoniana, cualquier ciclo hamiltoniano de $G + uv$ usa a la arista uv . Por lo tanto, existe una trayectoria hamiltoniana en G que empieza en u y termina en v , digamos $(v_1, \dots, v_{|V|})$. Sean

$$S = \{v_i : uv_{i+1} \in E\}$$

y

$$T = \{v_i : v_i v \in E\}.$$

Como $v_{|V|} \notin S \cup T$, tenemos que

$$|S \cup T| < |V|.$$

Además, si v_i fuera un vértice en $S \cap T$, entonces $v_1 C v_i v_{|V|} C^{-1} v_{i+1} v_1$ sería un ciclo hamiltoniano en G , contradiciendo la elección de G . Por lo tanto, obtenemos

$$|S \cap T| = 0.$$

Como $|S \cup T| = |S| + |T| - |S \cap T|$, de la información anterior podemos concluir que

$$d(u) + d(v) = |S| + |T| = |S \cup T| + |S \cap T| < |V|,$$

lo que resulta en una contradicción a que $\delta \geq \frac{|V|}{2}$. □

La demostración del resultado anterior puede utilizarse para demostrar, por un lado, que si la suma de los grados de cualesquiera dos vértices no adyacentes es al menos $|V|$ (condición de Ore), entonces G es hamiltoniana. Por otro lado, Bondy y Chvátal (1974) utilizaron la misma idea para definir una operación que preserva la hamiltonicidad de una gráfica.

Lema 4.3.3. *Sea G una gráfica simple de orden al menos 3. Si u y v son vértices no adyacentes en G tales que $d(u) + d(v) \geq |V|$, entonces G es hamiltoniana si y sólo si $G + uv$ es hamiltoniana.*

Demostración. Si G es hamiltoniana, entonces $G + uv$ también lo es. Recíprocamente, si $G + uv$ es hamiltoniana pero G no lo es, entonces podemos obtener la misma contradicción que en la demostración del Teorema 4.3.2. \square

clausura

A partir del Lema 4.3.3 resulta natural la siguiente definición. La **clausura** de G es la gráfica obtenida de G al unir recursivamente parejas de vértices no adyacentes tales que la suma de sus grados es al menos $|V|$, hasta que no queden parejas con esta propiedad.

Lema 4.3.4. *Sea G una gráfica. La clausura de G está bien definida.*

Demostración. Sean G_1 y G_2 gráficas obtenidas al unir recursivamente pares de vértices no adyacentes cuya suma de grados es al menos $|V|$, hasta que no haya pares con esta propiedad. Sean e_1, \dots, e_m y f_1, \dots, f_n las sucesiones de aristas agregadas a G para obtener G_1 y G_2 , respectivamente. Veamos que e_i es una arista de G_2 para cada $i \in \{1, \dots, m\}$ y que f_j es una arista de G_1 para cada $j \in \{1, \dots, n\}$.

Supongamos que alguna arista de G_1 no es una arista de G_2 , y sea e_{k+1} la primera en la sucesión e_1, \dots, e_n con esta propiedad, $e_{k+1} = uv$. Sea $H = G + \{e_1, \dots, e_k\}$. De la definición de G_1 se tiene que $d_H(u) + d_H(v) \geq |V|$.

Por la elección de e_{k+1} , se tiene que H es una subgráfica de G_2 , por lo que $d_{G_2}(u) + d_{G_2}(v) \geq |V|$. Pero esto resulta en una contradicción, pues, por construcción, u y v deben ser adyacentes en G_2 . Por lo tanto, e_i es una arista de G_2 para cada $i \in \{1, \dots, m\}$, y análogamente se puede demostrar que f_j es una arista de G_1 para cada $j \in \{1, \dots, n\}$. Por lo tanto, $G_1 = G_2$, y la clausura de una gráfica está bien definida. \square

Finalmente, se obtiene una caracterización de las gráficas hamiltonianas en términos de su clausura.

Teorema 4.3.5. *Una gráfica simple es hamiltoniana si y sólo si su clausura es hamiltoniana.*

Demostración. Basta con proceder inductivamente, aplicando el Lema 4.3.3 cada vez que una arista es agregada en la construcción de la clausura. \square

Corolario 4.3.6. *Sea G una gráfica simple de orden al menos tres. Si la clausura de G es completa, entonces G es hamiltoniana.*

El corolario 4.3.6 suele ser útil para verificar si una gráfica es hamiltoniana. Por desgracia, hay muchas gráficas hamiltonianas para las que no es posible aplicar este corolario. Como un ejemplo extremo, consideremos un ciclo de longitud al menos 5. El grado mínimo en cualquier ciclo es 2, y si el orden es al menos 5, entonces la suma de los grados de cualesquiera vértices no adyacentes es menor que el orden del ciclo, por lo que el ciclo coincide con su clausura.

Para terminar esta sección, cabe señalar que la condición dada por el Teorema de Dirac (Teorema 4.3.2) tiene consecuencias más fuertes que la hamiltonicidad de la gráfica. Una gráfica simple es **pancíclica** si tiene ciclos de todas las longitudes entre 3 y $|V|$. Es posible demostrar que cualquier gráfica simple que cumple las condiciones del Teorema de Dirac es pancíclica. Esto es cierto para una gran cantidad de condiciones suficientes para que una gráfica sea hamiltoniana, por lo que Bondy conjeturó que toda condición no trivial que implique que una gráfica es hamiltoniana, también implica que la gráfica es pancíclica, salvo quizá un conjunto de excepciones esporádicas. *pancíclica*

Capítulo 5

Apareamientos

5.1. Apareamientos

Un **apareamiento** en una gráfica G es un subconjunto M de aristas de G que *apareamiento* no son lazos y no comparten extremos dos a dos. Dado un apareamiento M y un vértice v en la gráfica G , se dice que v está **M -saturado**, o que M **satura a v** , si alguna arista de M incide en v , en otro caso se dice que v está **M -insaturado**. Un apareamiento M es **perfecto** si cada vértice de G está M -saturado. Si M es un *perfecto* apareamiento, se denotará por \overline{M} a $E - M$, el complemento de M relativo a E .

Un **apareamiento máximo** en G es un apareamiento M tal que $|M| \geq |M'|$ pa- *apareamiento*
ra cualquier otro apareamiento M' de G . Los apareamientos son la noción, en aristas, *máximo*
correspondiente a la de conjunto independiente, por este motivo, se denota por α'_G a la cardinalidad de un apareamiento máximo en la gráfica G ; el subíndice se omitirá cuando el contexto así lo permita. En ocasiones resultarán de interés apareamientos máximos por contención, i.e., apareamientos que no están contenidos propiamente en apareamiento alguno, aunque éstos no necesariamente sean máximos. Debido a sus múltiples aplicaciones, es importante determinar cuando un apareamiento es máximo. Por suerte, Berge encontró una condición simple, y fácil de verificar que caracteriza a los apareamiento máximos.

Dado un apareamiento M , una trayectoria es **M -alternante** si utiliza, alternadamente, aristas en M y en \overline{M} . Una trayectoria M -alternante P es **M -aumentable** si el primer y el último vértice de P están M -insaturados.

Lema 5.1.1. *Sea G una gráfica. Si M es un apareamiento de G , entonces M es máximo si y sólo si G no contiene trayectorias M -aumentables.*

Demostración. Ambas implicaciones serán demostradas por contrapositiva. Primero, demostraremos que si G contiene una trayectoria M -aumentable, entonces existe un apareamiento M' tal que $|M'| = |M| + 1$.

Sea P una trayectoria M -aumentable de G . Como el primer y el último vértice de P deben estar M -insaturados, la longitud de P debe ser impar, por lo que podemos

suponer que $P = (x_0, \dots, x_{2k+1})$. Así, $x_{2i}x_{2i+1} \in \overline{M}$ para cada $i \in \{0, \dots, k\}$, y $x_{2i-1}x_{2i} \in M$ para cada $i \in \{1, \dots, k\}$. Por lo tanto, si S y T son subconjuntos de E dados por

$$S = \{x_{2i-1}x_{2i} : i \in \{1, \dots, k\}\},$$

y

$$T = \{x_{2i}x_{2i+1} : i \in \{0, \dots, k\}\}$$

entonces $M' = (M - S) \cup T$ es un apareamiento en G tal que $|M'| = |M| + 1$, por lo que M no es máximo.

Supongamos ahora que M es un apareamiento que no es máximo. Sea M' un apareamiento máximo de G , entonces $|M| < |M'|$, y por lo tanto, la diferencia simétrica $M \Delta M'$ es no vacía. Sea H la gráfica $H = G[M \Delta M']$. Como un vértice v de G es un vértice de H si y sólo si v es M -saturado o M' -saturado, entonces cada vértice en H tiene grado 1 o 2. Luego, cada componente de G es una trayectoria M -alternante o un ciclo M -alternante.

Como $|M| < |M'|$, y cada ciclo en H tiene el mismo número de aristas de M y de M' , entonces alguna componente de H debe ser una trayectoria M -alternante que tenga más aristas de M' que de M . La única forma en la que esto puede suceder es que dicha trayectoria empiece y termine una arista en $M' - M$, por lo que sus extremos están M -insaturados, y resulta ser M -aumentable. \square

5.2. Apareamientos en gráficas bipartitas

Dados una gráfica G y un conjunto S de vértices de G , definimos a la **vecindad de S** , $N_G(S)$, como el conjunto de todos los vértices adyacentes a vértices en S . Cuando G es una gráfica bipartita, con bipartición (X, Y) , suele ser de interés para diversas aplicaciones encontrar un apareamiento que sature a todos los vértices de X . El siguiente teorema, demostrado originalmente por Hall en 1935, en el contexto de la teoría de grupos, caracteriza a las gráficas bipartitas con esta propiedad.

Teorema 5.2.1. *Si G es una gráfica bipartita con bipartición (X, Y) , entonces G contiene un apareamiento que satura a cada vértice en X si y sólo si $|S| \leq |N(S)|$ para cada $S \subseteq X$.*

Demostración. Supongamos primero que G tiene un apareamiento M que satura a cada vértice de X , y sea S un subconjunto de X . Como M induce una función inyectiva de S en $N(S)$ (la imagen de un vértice en S es el otro extremo de la arista de M que incide en él), es claro que $|S| \leq |N(S)|$.

Recíprocamente, supongamos que para cualquier subconjunto S de X , se cumple $|S| \leq |N(S)|$. Sea M un apareamiento que no satura a todos los vértices de X . Demostraremos, exhibiendo una trayectoria M -aumentable, que M no es máximo, por lo que un apareamiento máximo debe saturar a cada vértice en X .

Por la elección de M , existe un vértice v en X que está M -insaturado. Sea Z el conjunto de todos los vértices conectados a v mediante trayectorias M -alternantes. Sean S y T definidos por $S = X \cap Z$ y $T = Y \cap Z$. Como G es bipartita, y v está M -insaturado, la última arista de la trayectoria M -alternante con la que se alcanza a un vértice en S debe estar en M . Análogamente, la última arista de la trayectoria M -alternante con la que se alcanza a un vértice en T debe estar en \overline{M} .

Sea u es un vértice en $N(S)$, y sea $w \in S$ tal que $uw \in E$. Además, sea P la vw -trayectoria M -alternante que existe por la definición de S . Si u es un vértice de P , entonces u está en T . Si no, como la última arista de P está en M , tenemos que $uw \notin M$, por lo que Pwu es una vu -trayectoria M -alternante, y por lo tanto $u \in T$. Luego, $N(S) \subseteq T$, y por hipótesis, obtenemos $|S| \leq |T|$. Como S tiene un vértice M -insaturado, entonces también T debe tener un vértice M -insaturado, digamos z . Entonces, la uz -trayectoria M -alternante que existe en G es una trayectoria M -aumentable. \square

Es posible extraer una idea para un algoritmo que encuentre un apareamiento máximo en una gráfica bipartita a partir de la demostración del Teorema 5.2.1. Intuitivamente, se puede iniciar con cualquier apareamiento (por ejemplo el vacío), y si se encuentra un vértice insaturado, se construye el conjunto de todos los vértices que se pueden alcanzar desde éste mediante trayectorias alternantes. Si esta construcción resulta en una trayectoria aumentable, entonces se construye un apareamiento con una arista más que el actual, como en la demostración del Lema 5.1.1, y se repite el procedimiento.

Corolario 5.2.2. *Sea k un entero positivo. Si G es una gráfica bipartita k -regular, entonces G tiene un apareamiento perfecto.*

Demostración. Sea G una gráfica bipartita k -regular con bipartición (X, Y) . Como G es k -regular, se tiene que $k|X| = |E| = k|Y|$, y como k es positiva, se concluye $|X| = |Y|$. Para un subconjunto S de X , denotemos por E_1 y E_2 a los conjuntos de aristas incidentes en vértices de S y $N(S)$, respectivamente. Por la definición de $N(S)$, $E_1 \subseteq E_2$, y por lo tanto $k|N(S)| = |E_2| \geq |E_1| = k|S|$.

Luego, $|N(S)| \geq |S|$, y como S fue elegido arbitrariamente, se sigue del Teorema 5.2.1 que G tiene un apareamiento que satura a cada vértice de X . Como $|X| = |Y|$, tal apareamiento es un apareamiento perfecto. \square

Una **cubierta** en una gráfica G es un subconjunto K de V tal que cada arista *cubierta* de G tiene al menos un extremo en K . Una **cubierta mínima** es una cubierta K tal que para cualquier otra cubierta K' se tiene $|K| \leq |K'|$. La cardinalidad de una cubierta mínima en G se denota por β_G , donde el subíndice será usualmente omitido cuando pueda inferirse a partir del contexto.

Las cubiertas tienen una relación bastante natural con los apareamientos. Si K es una cubierta de G y M es un apareamiento de G , como no hay dos aristas de M que compartan extremos, entonces K necesita tener al menos un extremo de cada arista

en M , por lo que $|M| \leq |K|$. Como la elección de K y M fue arbitraria, esta relación se vale para cualesquiera cubierta y apareamiento, por lo que, sin en particular M^* es un apareamiento máximo y \tilde{K} es una cubierta mínima, entonces $|M^*| \leq |\tilde{K}|$, o en otras palabras

$$\alpha' \leq \beta.$$

Desafortunadamente, la igualdad anterior no siempre se cumple. Considérese por ejemplo a K_3 , donde claramente $\alpha' = 1$, y $\beta = 2$. Sin embargo, König demostró en 1931 que $\alpha' = \beta$ siempre es cierto para gráficas bipartitas. Antes de enunciar y demostrar el teorema antes referido, es necesaria una observación sencilla.

Lema 5.2.3. *Sean G una gráfica, M un apareamiento y K una cubierta de G . Si $|M| = |K|$, entonces M es máximo y K es mínimo, por lo que $\alpha' = \beta$.*

Demostración. Si M^* es un apareamiento máximo y \tilde{K} es una cubierta mínima, entonces, como $\alpha' \leq \beta$, tenemos que

$$|M| \leq |M^*| = \alpha' \leq \beta = |\tilde{K}| \leq |K|.$$

Como $|M| = |K|$, todas las desigualdades son en realidad igualdades, y por lo tanto $|M| = \alpha'$ y $|K| = \beta$. \square

De forma un tanto sorprendente, el Teorema de Hall (Teorema 5.2.1) y el siguiente resultado, que afirma $\alpha' = \beta$ en gráficas bipartitas, son equivalentes, es decir, es posible demostrar uno a partir del otro y viceversa. De manera aún más sorprendente, ambos resultados son equivalentes al Teorema de Menger, en cualquiera de sus versiones (Teoremas 3.4.1 y 3.5.1). De hecho, hay muchas otras equivalencias para estos teoremas, varias de ellas enunciadas a través de relaciones min-max (e.g., la cardinalidad de una cubierta mínima coincide con la de un apareamiento máximo en gráficas bipartitas, o el máximo número de xy -trayectorias internamente ajenas en una gráfica coincide con la cardinalidad de un xy -corte mínimo, cuando x y y no son adyacentes). Al ser relativamente fácil encontrar en la literatura demostraciones del Teorema de König que no utilizan alguna de estas equivalencias, a continuación se presenta una demostración que utiliza el Teorema de Hall.

Teorema 5.2.4. *Sea G una gráfica. Si G es bipartita, entonces*

$$\alpha' = \beta.$$

Demostración. Como los vértices aislados son intrascendentes para calcular las cardinalidades de apareamientos máximos y cubiertas mínimas, podemos suponer sin pérdida de generalidad que G no tiene vértices aislados.

Sea (X, Y) la bipartición de G y sea K una cubierta mínima en G . Construiremos un apareamiento M tal que $|M| = |K|$; el resultado deseado se sigue entonces directamente del Lema 5.2.3.

Sean A y B dados por $A = K \cap X$ y $B = K \cap Y$. Para construir al apareamiento M construiremos dos apareamientos, M_1 de cardinalidad $|A|$, y M_2 de cardinalidad $|B|$, ajenos y tales que $M_1 \cup M_2 = M$. Empecemos entonces por M_1 .

Sea H la subgráfica de G inducida por $A \cup (Y - K)$. Como K es una cubierta de G , y ninguna arista con un extremo en $Y - K$ está cubierta por B , tenemos que A es una cubierta de H ; análogamente B es una cubierta de $G[V - V_H]$. Además, si hubiera una cubierta de H con menos vértices que A , digamos K_H , entonces, como todas las aristas con un extremo en $V - V_H$ están cubiertas por B , tenemos que $(K - A) \cup K_H$ sería una cubierta de G con menos vértices que K , lo cuál no puede suceder. Así, A es una cubierta mínima de H . Por lo tanto, para cualquier $S \subseteq A$ se tiene que $|S| \leq |N(S)|$, pues de lo contrario, $(A - S) \cup N(S)$ sería una cubierta de H con menos vértices que A . Así, H cumple las hipótesis del Teorema de Hall, y por lo tanto existe un apareamiento M_1 en H que satura a cada vértice en A , y tal que $|M_1| = |A|$.

Análogamente, existe un apareamiento M_2 de $G[V - V_H]$ de cardinalidad $|B|$. Como H y $G[V - V_H]$ son ajenas, es claro que $M_1 \cup M_2$ es un apareamiento en G de cardinalidad $|A| + |B| = |K|$. \square

5.3. Apareamientos perfectos

La caracterización más conocida de las gráficas con un apareamiento perfecto se debe a Tutte (1947). Se presenta la ya clásica demostración de Lovász (1973). Una componente conexa de una gráfica será llamada **par** o **impar** de acuerdo a su número de vértices. Denotamos por o_G al número de componentes impares de G .

Teorema 5.3.1. *Si G es una gráfica, entonces G tiene un apareamiento perfecto si y sólo si $o_{G-S} \leq |S|$ para cada $S \subset V$.*

Demostración. Podemos suponer sin pérdida de generalidad que G es simple.

Supongamos primero que G tiene un apareamiento perfecto M . Sea S un subconjunto propio de V , y sean G_1, \dots, G_k las componentes impares de $G - S$. Como G_i es impar, entonces algún vértice u_i de G_i debe estar apareado bajo M con algún vértice v_i de S , para cada $i \in \{1, \dots, k\}$. Por lo tanto, como $\{v_1, \dots, v_k\} \subseteq S$, se tiene

$$o_{G-S} = k = |\{v_1, \dots, v_k\}| \leq |S|.$$

Recíprocamente, procediendo por contradicción, supongamos que G satisface la condición suficiente del teorema, pero no tiene un apareamiento perfecto. En particular, $o_{G-\emptyset} \leq |\emptyset| = 0$, por lo que el G tiene orden par. Sea G^* una supergráfica generadora de G que sea máxima por contención con la propiedad de no tener un apareamiento perfecto. Como el orden de G es par, G^* no es completa. Además, para cualquier subconjunto S de V , $G - S$ es una subgráfica generadora de $G^* - S$,

por lo que $o_{G^*-S} \leq o_{G-S}$, y por hipótesis se sigue que $o_{G^*-S} \leq |S|$. Así, G^* es un contraejemplo máximo al teorema.

Sea U el conjunto de vértices de grado $|V| - 1$ en G^* . Como G^* no es completa, tenemos que $U \neq V$. Procederemos por contradicción para demostrar que $G^* - U$ es un cluster. Supongamos que alguna componente de $G^* - U$ no es completa. Entonces, esta componente contiene un P_3 como subgráfica inducida, digamos, (x, y, z) . Además, como $y \notin U$, existe un vértice w en $G^* - U$ tal que $yw \notin E_{G^*}$.

Por ser G^* máxima con la propiedad de no tener un apareamiento perfecto, entonces $G^* + xz$ y $G^* + yw$ tienen apareamientos perfectos M_1 y M_2 , respectivamente. Sea H la subgráfica de $G^* \cup \{xz, yw\}$ inducida por $M_1 \Delta M_2$. Al ser M_1 y M_2 apareamientos perfectos, todo vértice de H tiene grado 2 (pues está M_1 - y M_2 -saturado), por lo que cada componente de H es un ciclo par que alterna aristas de M_1 y M_2 . Consideramos dos casos.

Caso 1. Si xz y yw están en componentes distintas de H . Si C es la componente de H que contiene a yw , entonces las aristas de M_1 en C , junto con las aristas de M_2 que no están en C , conforman un apareamiento perfecto de G^* , lo que resulta en una contradicción.

Caso 2. Si xz y yw están en la misma componente C de H , entonces, podemos suponer sin pérdida de generalidad, que los vértices x, y, w y z ocurren en ese orden en C . Entonces, las aristas de M_1 en la trayectoria $ywCz$, junto con la arista yz y las aristas de M_2 fuera de $ywCz$, conforman un apareamiento perfecto de G^* , resultando nuevamente en una contradicción.

Se sigue de ambos casos que $G^* - U$ es un cluster.

Como $o_{G^*-U} \leq |U|$, a lo más $|U|$ de las componentes de $G^* - U$ son impares. Entonces, es fácil construir un apareamiento perfecto para G^* : un vértice en cada componente impar de $G^* - U$ está apareado con un vértice en U ; los vértices restantes se encuentran en gráficas completas pares, por lo que admiten un apareamiento perfecto. Pero esto resulta en una contradicción. \square

El Teorema 5.3.1 también es equivalente al Teorema de Hall.

Una consecuencia sencilla del del Teorema de Tutte es el siguiente resultado sobre apareamientos perfectos en gráficas cúbicas, demostrado originalmente por Petersen en 1891.

Corolario 5.3.2. *Toda gráfica cúbica sin puentes tiene un apareamiento perfecto.*

Demostración. Sea G una gráfica cúbica sin puentes, y sea S un subconjunto propio de V . Además, sean G_1, \dots, G_k las componentes impares de $G - S$, y sea m_i el número de aristas con un extremo en G_i y el otro en S , para $i \in \{1, \dots, k\}$. Como G es 3-regular, tenemos que, para cada $i \in \{1, \dots, k\}$,

$$\sum_{v \in V_{G_i}} d(v) = 3|V_{G_i}|,$$

y

$$\sum_{v \in S} d(v) = 3|S|.$$

Por lo tanto, $m_i = \sum_{v \in V_{G_i}} d(v) - 2|E_{G_i}|$ es impar. Como G no tiene puentes, tenemos que $m_i \neq 1$. Por lo tanto, para cada $i \in \{1, \dots, k\}$, tenemos que $m_i \geq 3$.

Por lo tanto,

$$o_{G-S} = k \leq \frac{1}{3} \sum_{i=1}^k m_i \leq \frac{1}{3} \sum_{v \in S} d(v) = |S|.$$

Se sigue del Teorema 5.3.1 que G tiene un apareamiento perfecto. □

5.4. El algoritmo de Egerváry

Capítulo 6

Coloraciones

6.1. Coloración por aristas

Sea G una gráfica. Para un entero positivo k , una **k -coloración por aristas** de G es una función $c: E \rightarrow S$, donde S es un conjunto de cardinalidad k . En este contexto, a los elementos de S se les llama colores, por lo que se suele decir que una k -coloración por aristas es una asignación de k colores a las aristas de G . Usualmente, se utiliza a $\{1, \dots, k\}$ como el conjunto S de colores. Para cada color $i \in S$, la imagen inversa $c^{-1}[i]$ es una **clase cromática** de la coloración c ; las clases cromáticas pueden ser vacías. Resulta entonces natural considerar, de forma equivalente, a la coloración c como una partición¹ (E_1, \dots, E_k) , donde $E_i = c^{-1}[i]$, i.e., E_i es el conjunto de todas las aristas a las que se les asignó el color i .

Una coloración por aristas es **propia** si aristas adyacentes reciben colores distintos. Así, una k -coloración por aristas propia es una coloración por aristas (M_1, \dots, M_k) , en la que cada clase cromática M_i es un apareamiento. Como los lazos son aristas adyacentes a sí mismas, sólo las gráficas sin lazos admiten coloraciones por aristas propias. Como en este capítulo sólo se considerarán coloraciones propias, las gráficas en esta sección no tendrán lazos, y se utilizará “coloración por aristas” para referirse a una coloración por aristas propia.

Una gráfica es k -coloreable por aristas si admite una k -coloración. Claramente, si una gráfica es k -coloreable por aristas, entonces también es ℓ -coloreable por aristas para cualquier $\ell < k$ (algunas clases cromáticas pueden ser vacías); cualquier gráfica es $|E|$ -coloreable por aristas. El **número cromático por aristas** o **número cromático lineal**, χ'_G es el menor entero k talque G es k -coloreable por aristas, y G es **k -cromática por aristas** si $\chi'_G = k$.

En cualquier coloración por aristas, todas las aristas incidentes con cualquier vértice deben recibir colores distintos, por lo que trivialmente se tiene la siguiente

¹Como es usual en combinatoria, no se requiere que cada parte sea no vacía.

cota inferior

$$\chi' \geq \Delta.$$

No es difícil construir ejemplos donde la desigualdad anterior sea estricta. Sin embargo, en muchas familias de gráficas es posible verificar que la igualdad siempre se cumple. Un ejemplo sencillo se presenta para las gráficas bipartitas en el siguiente resultado.

Dada una gráfica G , una subgráfica generadora H de G y una k -coloración c de H , se dice que un color i está **representado** en un vértice v si está asignado a alguna arista incidente en v ; en otro caso esta **disponible** en v . Un color está **disponible** para una arista $E_G - E_H$ si está disponible en ambos extremos de la arista. Así, si una arista e de G no está coloreada, es posible asignarle cualquier color que tenga disponible para extender c a una k -coloración por aristas de $H + e$.

Sean i y j dos colores distintos, y sea H_{ij} la gráfica inducida en H por $M_i \cup M_j$, i.e., $H_{ij} = H[M_i \cup M_j]$. Como cada clase cromática es un apareamiento, las componentes de H_{ij} son ciclos pares o trayectorias; estas últimas serán llamadas *ij-trayectorias*.

Proposición 6.1.1. *Sea G una gráfica. Si G es bipartita, entonces*

$$\chi' = \Delta.$$

Demostración. Por inducción sobre $|E|$. Sea $e = uv$ una arista de G y sea H la subgráfica $H = G - e$ de G . Por hipótesis inductiva H tiene una Δ -coloración por aristas (M_1, \dots, M_Δ) . Si hay algún color disponible para e , basta con asignarle ese color a e para obtener una Δ -coloración de G . Luego, podemos suponer que la unión de los colores representados en u y v es el conjunto $\{1, \dots, \Delta\}$. Como el grado de u en H es a lo más $\Delta - 1$, al menos un color i está disponible en u , y por lo tanto está representado en v . Análogamente, hay un color j disponible en v y representado en u . En la gráfica H_{ij} , el vértice u tiene grado 1, por lo que la componente en la que está u es una *ij-trayectoria*. Esta trayectoria no termina en v , pues de ser así, y al empezar con una arista de color i y terminar en una arista de color j , $P + e$ sería un ciclo de longitud par en G , contradiciendo que G es bipartita. Por lo tanto, al intercambiar los colores en P , obtenemos una nueva Δ -coloración por aristas de H , respecto a la cuál el color i está disponible en u y en v . Asignando el color i a e obtenemos una Δ -coloración por aristas de G . \square

Es posible extraer un algoritmo de tiempo polinomial para obtener una Δ -coloración por aristas de la demostración de la Proposición 6.1.1.

Por otro lado, basta notar que un ciclo impar tiene número cromático por aristas igual a 3 para concluir que si G no es bipartita, entonces no necesariamente $\chi' = \Delta$. Sorprendentemente, Vizing, en 1964, e independientemente Gupta, en 1966, demostraron que para cualquier gráfica simple, se cumple que $\chi' = \Delta$ o $\chi' = \Delta + 1$.

La demostración se hará por inducción sobre el número de aristas. De manera semejante a la demostración de la Proposición 6.1.1, supondremos que, para alguna

arista e de G , hay una $(\Delta + 1)$ -coloración por aristas de $G - e$, que puede ser utilizada para construir una $(\Delta + 1)$ -coloración por aristas de G . Se demostrará un resultado más general; si k es un entero con $k \geq \Delta$, y e es alguna arista de G , entonces es posible obtener una k -coloración por aristas de G a partir de una k -coloración por aristas de $G - e$.

Lema 6.1.2. *Sean G una gráfica simple, v un vértice de G , e una arista de G incidente a v y k un entero con $k \geq \Delta$. Si $G - e$ tiene una k -coloración por aristas c , con respecto a la cuál cada vecino de v tiene al menos un color disponible, entonces G es k -coloreable por aristas.*

Demostración. Sea c una k -coloración por aristas de $G - e$. Definamos a la gráfica bipartita $H[X, Y]$, donde $X = N_G(v)$, $Y = \{1, \dots, k\}$, y tal que los vértices $x \in X$ e $i \in Y$ son adyacentes si y sólo si el color i está disponible en el vértice x en la restricción \tilde{c} de c a $G - v$. En particular, si u es el otro extremo de e , entonces para cada $x \in X - \{u\}$, el color de la arista xv está disponible en x en $G - v$, por lo que H contiene al apareamiento M dado por

$$M = \{xc(xv): x \in X - \{u\}\}.$$

Recíprocamente, cada apareamiento en H corresponde a una coloración parcial de $\partial(v)$ que es compatible con \tilde{c} . En particular, cualquier apareamiento en H que sature a X corresponde a una coloración de todas las aristas en $\partial(v)$, y por lo tanto, al tomar la unión con \tilde{c} , obtenemos una k -coloración de G . Así, podemos suponer que H no tiene una coloración de este tipo. Nuestro objetivo es modificar la coloración c a una coloración c' de tal forma que la gráfica bipartita H' correspondiente contenga un apareamiento que sature a X .

Por hipótesis, cada vértice de $X - \{u\}$ es incidente con al menos una arista de $H - M$, y dado que

$$d_{G-e}(u) = d_G(u) - 1 \leq \Delta_G - 1 \leq k - 1,$$

el vértice u también es incidente con al menos una arista de $H - M$. Por lo tanto, cada vértice de X es incidente con al menos una arista de $H - M$.

Sea Z el conjunto de todos los vértices en H alcanzables desde u mediante trayectorias M -alternantes, y sean R y B dados por $R = X \cap Z$ y $B = Y \cap Z$. De forma semejante a la demostración del Teorema de Hall (Teorema 5.2.1), $N_H(R) = B$ y B está apareado bajo M con $R - \{u\}$, por lo que $|B| = |R| - 1$. Como cada vértice de R es incidente con al menos una arista de $H - M$, por el principio de las casillas existen dos vértices x y y en R que son adyacentes en $H - M$ al mismo color $i \in B$. Por lo tanto, el color i está disponible en x y en y . Observemos que cada color de B está representado en v , porque B está apareado bajo M con $R - \{u\}$; en particular, el color i está representado en v . Por otro lado, como el grado de v en $G - e$ es a lo más $k - 1$, algún color j está disponible en v . Además, como cada color de B está

representado en v , tenemos que $j \notin B$. Por lo tanto, j está representado en cada vértice de R , en particular en x y en y .

Volviendo a $G - e$, por las observaciones anteriores, cada uno de los vértices v, x y y es el extremo de una ij -trayectoria en $G - e$. Consideremos a la ij -trayectoria que empieza en v . Como esta trayectoria no puede terminar en x y en y , podemos considerar al vértice z , el vértice terminal de la ij -trayectoria P que empieza en y . Intercambiando los colores i y j en P , obtenemos una nueva coloración c' de $G - e$.

Sea $H'[X, Y]$ la gráfica bipartita que corresponde a c' . Dado que los vértices internos de P tienen los mismos colores representados en c y en c' , las únicas diferencias en los conjuntos de aristas de H y H' ocurren en y , y quizá en z (si $z \in X$). Además, como v no está en P , el apareamiento M sigue siendo un apareamiento en H' . Consideremos a la uy -trayectoria alternante Q en H . Si Q pasa por z , entonces $z \in R$, y, dado que termina con una arista de M , la trayectoria alternante uQz es una trayectoria M -alternante en H' . Además, como $j \notin B$, originalmente, la trayectoria P debe haber terminado en z usando una arista de color j , y ahora termina en usando una arista de color i . Entonces, el color j está disponible respecto a la coloración c' , y $Q' = uQzj$ es una trayectoria M -aumentable en H' . Por otro lado, si z no está en Q , entonces $Q' = uQyj$ es una trayectoria M -aumentable en H' .

Sea $M' = M \Delta E_{Q'}$. Entonces M' es un apareamiento en H' que cubre a cada vértice de X , por lo que corresponde a una coloración de cada arista en $\partial(v)$. Tomando la unión de esta coloración con la restricción de c' a $G - v$, obtenemos una k -coloración por aristas de G . \square

La demostración anterior da lugar a un algoritmo de tiempo polinomial para encontrar una k -coloración por aristas en una gráfica simple G , a partir de una coloración de $G - e$ que satisfaga las hipótesis del Lema 6.1.2. Como las hipótesis del Lema 6.1.2 se satisfacen cuando $k = \Delta + 1$, el Teorema de Vizing se sigue directamente al aplicar inducción sobre $|E|$. De hecho, es posible encontrar una $(\Delta + 1)$ -coloración de cualquier gráfica simple G , en tiempo polinomial, agregando una arista a la vez.

Teorema 6.1.3. *Para cualquier gráfica simple G ,*

$$\chi' \leq \Delta + 1.$$

En vista del Teorema de Vizing, para determinar el número cromático por aristas de una gráfica basta saber si ésta es Δ -coloreable; si lo es, su número cromático por aristas es Δ , y si no, es $\Delta + 1$. Desafortunadamente, el problema de determinar si una gráfica admite una Δ -coloración es NP -completo, por lo que hay pocas esperanzas de encontrar una caracterización útil de las gráficas para las que $\chi' = \Delta$.

6.2. Coloración por vértices

Recordando que una coloración por aristas propia se puede interpretar como una partición del conjunto de aristas de una gráfica en apareamientos, y que los apa-

reamientos son el análogo en aristas a los conjuntos independientes, resulta natural definir una coloración por vértices propia como una partición del conjunto de vértices de una gráfica en conjuntos independientes. Sin embargo, al igual que en el caso de las coloraciones por aristas, se definirá primero el concepto general de coloración por vértices, no necesariamente propia.

Dada una gráfica G y un entero positivo k , una **k -coloración por vértices**, o simplemente **k -coloración**, de G es una función $c: V \rightarrow S$, donde S es un conjunto de cardinalidad k , cuyos elementos serán llamados *colores*. Así, una k -coloración de G es una asignación de colores a los vértices de G , usando a lo más k colores. Aunque el conjunto S no tiene restricciones, usualmente se utilizará a $\{1, \dots, k\}$ como conjunto de colores. Una coloración es **propia** si vértices adyacentes reciben colores distintos. Cada lazo en una gráfica hace adyacente a un vértice a sí mismo, por lo que las gráficas con lazos no admiten coloraciones propias. En el presente texto sólo se considerarán coloraciones propias, por lo que el adjetivo “propia” usualmente será omitido. Dado un entero positivo k , una gráfica es **k -coloreable** si admite una k -coloración. Una gráfica es k -coloreable si y sólo si su gráfica simple subyacente es k -colorable, por lo que el resto del capítulo sólo se considerarán gráficas simples. *k-coloración*
k-coloreable

Análogamente al caso de la coloración por aristas, para un color $i \in S$, se define a la **clase cromática** V_i de la coloración c como $V_i = c^{-1}[i]$, i.e., es el conjunto de todos los vértices de G a los que se les asignó el color i . Por lo tanto, una k -coloración puede definirse, alternativamente, como una partición (V_1, \dots, V_k) de V , donde V_i es el conjunto (posiblemente) vacío de vértices que reciben el color i . En una coloración propia, cada clase cromática es un conjunto independiente. Esta última observación da lugar a ajustar la definición de coloración propia usando homomorfismos de gráficas.

Dadas dos gráficas G y H , un **homomorfismo** de G en H , o un H -homomorfismo de G , es una función $\varphi: V_G \rightarrow V_H$ tal que, para cualesquiera dos vértices u y v en V_G , se tiene que $uv \in E_G$ implica $\varphi(u)\varphi(v) \in E_H$; es decir, un homomorfismo es una función que preserva adyacencia. Vale la pena notar que en la definición de homomorfismo de gráficas sólo se tiene una implicación, en contraste con el bicondicional en la definición de isomorfismo. Mientras que en un isomorfismo se preserva la adyacencia y la no adyacencia (ambas implicaciones), en un homomorfismo sólo se preserva la adyacencia, lo que tiene como resultado que vértices distintos pueden tener la misma imagen, o que vértices no adyacentes pueden tener imágenes adyacentes. *homomorfismo*

De acuerdo a la definición de coloración, un H -homomorfismo de la gráfica G es una $|V_H|$ -coloración de G , por lo que usualmente se utiliza el término H -coloración en lugar de H -homomorfismo. Cuando G y H son gráficas sin lazos, se tiene que vértices adyacentes en G no pueden tener la misma imagen bajo un H -homomorfismo, pues esto implicaría que su imagen tiene un lazo. Así, dado un H -homomorfismo φ de G , cada clase cromática es un conjunto independiente. Podría parecer que si $|V_H| = k$, la existencia de un H -homomorfismo es equivalente a la existencia de una k -coloración, sin embargo, a partir de la definición de homomorfismo se debe considerar que al

existir una H -coloración de G , la estructura de H impone restricciones sobre la estructura de G . Por ejemplo, aunque $|V_{P_3}| = 3 = |V_{K_3}|$, y aún cuando K_3 es 3-coloreable, no existe un P_3 -homomorfismo de K_3 , pues necesariamente vértices adyacentes en K_3 deben tener imágenes distintas y adyacentes en P_3 . Por otro lado, sí existe un K_3 -homomorfismo de P_3 . De hecho, en geneneral, si k es un entero positivo, como cualesquiera dos vértices son adyacentes en K_k , cualquier función de V_G en V_{K_k} en la que vértices adyacentes reciban colores distintos es una K_k -coloración de G . Luego es equivalente que una gráfica admita una k -coloración propia y que admita un K_k -homomorfismo.

número
cromático

El mínimo entero k para el que una gráfica G es k -coloreable es su **número cromático**, y se denota χ_G . Si $\chi_G = k$, se dice que G es k -**cromática**. Si H es una subgráfica de G , y G es k -coloreable, entonces la restricción a H de la k -coloración de G es una k -coloración de H . Luego, $\chi_H \leq \chi_G$. En particular, si G contiene una copia de la gráfica completa K_r , entonces $\chi_G \geq r$. Así, para cualquier gráfica G se tiene que $\chi \geq \omega$, donde ω es la cardinalidad de un clan máximo en G .

Claramente, una gráfica es 1-cromática si y sólo si es vacía, y es 2-cromática si y sólo si es bipartita y no vacía. Por lo tanto, es fácil determinar de forma eficiente cuando una gráfica tiene número cromático 1 o 2. Sin embargo, el problema de la k -coloración es NP -completo para cualquier $k \geq 3$. Por este motivo, es común encontrar resultados que acotan superiormente al número cromático de una gráfica, o una familia de gráficas. Es común encontrar cotas superiores exhibiendo coloraciones con un número dado de colores.

Proposición 6.2.1. *Para cualquier gráfica G ,*

$$\chi \leq \Delta + 1.$$

Demostración. Veamos que siempre es posible encontrar una $(\Delta + 1)$ -coloración de G en tiempo $O(|V| + |E|)$. Utilizaremos la heurística avariciosa o glotona, que consiste simplemente en colorear los vértices uno a uno, y utilizar siempre el primer color disponible.

Sea $\{v_1, \dots, v_{|V|}\}$ un orden total de los vértices de G . Para cada $i \in \{1, \dots, |V|\}$, coloreamos a v_i con el primer entero positivo que no ocurra en la vecindad de v_i . Notemos que a lo más se utilizan $\Delta + 1$ colores, pues para cualquier vértice pueden ocurrir a lo más Δ colores en su vecindad.

A cada vértice se le tiene que asignar un color, lo que en total toma tiempo $O(|V|)$; y para hacer esta asignación, se tienen que revisar los colores de todos sus vecinos, por lo que todas las aristas de la gráfica se tienen que explorar dos veces, lo que toma tiempo $O(|E|)$. \square

heurística avariciosa

El desempeño (respecto al número de colores utilizados) de la **heurística avariciosa**, utilizada en la demostración del Teorema 6.2.1, depende fuertemente del orden total elegido para colorear a los vértices. En uno de los ejercicios de la sección

se observa que siempre existe un orden total de los vértices para el que esta heurística produce una coloración óptima. Por desgracia, en general es difícil encontrar este orden, aunque, como se explora en algunos otros ejercicios de la sección, es posible hacerlo al suponer cierta estructura para la gráfica.

Como es posible constatar con los ciclos impares y las gráficas completas, la cota superior para el número cromático dada por la Proposición 6.2.1 es justa. Sin embargo, Brooks demostró que únicamente las gráficas conexas en estas dos familias necesitan $\Delta + 1$ colores para ser coloreadas óptimamente. Se presentará una demostración de Lovasz, en el que se elige un orden particular para aplicar la heurística avariciosa que resulta en una Δ -coloración. Antes, es necesario desarrollar algunos resultados técnicos.

Dado un entero positivo k , una gráfica G es **k -crítica** si es k -cromática y para cualquier subgráfica propia H de G , se tiene que $\chi_H < \chi_G$. Una gráfica es **crítica** si es k -crítica para algún entero positivo k . Como el número cromático de una gráfica es el máximo de los números cromáticos de sus componentes conexas, es fácil observar que una gráfica crítica es conexa. Los siguientes resultados establecen algunas otras propiedades de las gráficas críticas.

Proposición 6.2.2. *Sea k un entero positivo. Si G es k -crítica, entonces $\delta \geq k - 1$.*

Demostración. Procendiendo por contradicción, supongamos que G es una gráfica k -crítica con $\delta < k - 1$, y sea v un vértice de grado δ en G . Entonces $G - v$ tiene una $(k - 1)$ -coloración (V_1, \dots, V_{k-1}) , en la que a lo más $\delta < k - 1$ colores ocurren en la vecindad de v . Podemos suponer sin pérdida de generalidad que el color 1 no ocurre en la vecindad de v , por lo que $(V_1 \cup \{v\})$ es una $(k - 1)$ -coloración de G , contradiciendo que G es k -cromática.

Como la contradicción surge de suponer que $\delta < k - 1$, concluimos que $\delta \geq k - 1$. \square

Corolario 6.2.3. *Toda gráfica k -cromática tiene al menos k vértices de grado al menos $k - 1$.*

Demostración. Sea G una gráfica k -cromática y sea H una subgráfica k -crítica de G . Por la Proposición 6.2.2, cada vértice de H tiene grado al menos $k - 1$ en H , y por lo tanto también en G . Así, como cualquier vértice en H tiene al menos $k - 1$ vecinos, se tiene que H tiene al menos k vértices. \square

Sea S un corte por vértices de una gráfica conexa G , y sean V_1, \dots, V_k los conjuntos de vértices de las componentes de $G - S$. Las **S -componentes** de G son las subgráficas G_1, \dots, G_k donde $G_i = G[V_i \cup S]$. Coloraciones para distintas S -componentes de G son **compatibles** si para cualquier $v \in S$, las coloraciones le asignan el mismo color a v .

Proposición 6.2.4. *En una gráfica crítica, ningún corte por vértices es un clan.*

Demostración. Sea k un entero positivo, y sea G una gráfica k -crítica. Procediendo por contradicción, supongamos que S es un corte por vértices de G que es un clan. Sean G_1, \dots, G_ℓ las S -componentes de G . Como G es k -crítica, cada G_i tiene una $(k-1)$ -coloración c_i , y como S es un clan, cualesquiera dos vértices en S reciben colores distintos bajo c_i , para cada $i \in \{1, \dots, \ell\}$.

Para cualesquiera $i, j \in \{1, \dots, \ell\}$, si $i \neq j$, existe una permutación $p \in S_\ell$ tal que $p \circ c_i(v) = c_j(v)$ para cada $v \in S$. Por lo tanto, al aplicar una permutación tras cada coloración, podemos suponer que todas las coloraciones de las S -componentes son compatibles. Luego, $\bigcup_{i=1}^\ell c_i$ es una $(k-1)$ -coloración de G , contradiciendo que G es k -cromática. \square

Corolario 6.2.5. *Toda gráfica crítica es un bloque.*

Demostración. Procediendo por contrapositiva, supongamos que G tiene un vértice de corte v , entonces $\{v\}$ es un corte por vértices que es un clan. Por la contrapositiva de la Proposición 6.2.4, G no es crítica. \square

Siguiendo el espíritu del Corolario 6.2.5, si G es una gráfica k -crítica con un 2-corte $\{u, v\}$, entonces $uv \notin E$. En este caso una $\{u, v\}$ -componente H de G es del **tipo 1** si cada $(k-1)$ -coloración de H le asigna el mismo color a u y a v , y es del **tipo 2** si cada $(k-1)$ -coloración de H le asigna distintos colores a u y a v .

Proposición 6.2.6. *Sea G una gráfica k -crítica con un 2-corte por vértices $\{u, v\}$. Entonces*

1. $G = G_1 \cup G_2$, donde G_i es una $\{u, v\}$ -componente del tipo i , para $i \in \{1, 2\}$, y
2. $G_1 + uv$ y $G_2 / \{u, v\}$ son k -críticas.

Demostración. Para el primer enunciado, como G es k -crítica, cada $\{u, v\}$ -componente de G es $(k-1)$ -coloreable. Si todas las $\{u, v\}$ -componentes de G tuvieran una coloración que asigna el mismo color a u y a v , entonces, como en la demostración de la Proposición, 6.2.4, podríamos encontrar una $(k-1)$ -coloración de G . Por lo tanto, todas las $(k-1)$ -coloraciones de al menos una $\{u, v\}$ -componente de G asignan distintos colores a u y a v , i.e., al menos una $\{u, v\}$ -componente de G es del tipo 2. Análogamente al menos una $\{u, v\}$ -componente de G es del tipo 1.

Sea G_i una $\{u, v\}$ -componente del tipo i de G . Si $G_1 \cup G_2$ fuera una subgráfica propia de G , entonces $G_1 \cup G_2$ tendría una $(k-1)$ -coloración, cuya restricción a G_1 y a G_2 resultaría en $(k-1)$ -coloraciones compatibles de G_1 y G_2 , lo que resulta en una contradicción. Por lo tanto $G = G_1 \cup G_2$.

Para el segundo enunciado, sólo demostraremos que $G_2 / \{u, v\}$ es k -crítica, la otra afirmación puede demostrarse análogamente.

Llamemos G_2' a $G_2 / \{u, v\}$, y sea x_{uv} el vértice de G_2' que resulta al contraer al conjunto $\{u, v\}$. Es fácil encontrar una k -coloración de G_2' , pues basta usar la restricción de la $(k-1)$ -coloración de G_2 a $G_2 - \{u, v\}$, junto con un nuevo color para x_{uv} . Por otro

lado, si G'_2 tuviera una $(k-1)$ -coloración (V_1, \dots, V_{k-1}) , y suponiendo sin pérdida de generalidad que x_{uv} recibe el color 1, entonces $((V_1 \setminus \{x_{uv}\}) \cup \{u, v\}, V_2, \dots, V_{k-1})$ sería una $k-1$ -coloración de G_2 donde u y v reciben el mismo color, contradiciendo que G_2 es del tipo 2. Por lo tanto G'_2 es k -cromática.

Para ver que G'_2 es k -crítica, basta verificar que para cualquier arista e de G'_2 , la subgráfica $G'_2 - e$ es $(k-1)$ -coloreable. Sea e una arista de G'_2 . Si e no incide en x_{uv} , entonces es una arista de G , por lo que $G - e$ tiene una $(k-1)$ -coloración c . Como $G_1 \subseteq G - e$, entonces c asigna el mismo color a u y a v , por lo que es posible obtener una $(k-1)$ -coloración de $G'_2 - e$ a partir de la restricción de c a $G_2 - e$, y asignando a x_{uv} el mismo color que c asigna a u y a v . Si e incide en x_{uv} , y w es su otro extremo, entonces podemos suponer sin pérdida de generalidad que vw es una arista de G . Así, procediendo análogamente al caso anterior, pero considerando la $(k-1)$ -coloración de $G - vw$ en lugar de c , podemos encontrar nuevamente una $(k-1)$ -coloración de $G'_2 - e$. \square

Corolario 6.2.7. *Si G es una gráfica k -crítica con un 2-corte por vértices $\{u, v\}$, entonces*

$$d(u) + d(v) \geq 3k - 5.$$

Demostración. Sea G_i la $\{u, v\}$ -componente del tipo i de G para $i \in \{1, 2\}$. Sean $H_1 = G_1 + uv$ y $H_2 = G_2 / \{u, v\}$. Se sigue de las Proposiciones 6.2.2 y 6.2.6 que

$$d_{H_1}(u) + d_{H_1}(v) \geq 2k - 2$$

y

$$d_{H_2}(x_{uv}) \geq k - 1,$$

donde x_{uv} es el vértice obtenido al identificar a u y a v . Por la definición de H_1 y H_2 , se tiene que

$$d_{G_1}(u) + d_{G_1}(v) \geq 2k - 4$$

y

$$d_{G_2}(u) + d_{G_2}(v) \geq k - 1.$$

Como $d_G(x) = d_{G_1}(x) + d_{G_2}(x)$ para $x \in \{u, v\}$, sumando ambas desigualdades se obtiene el resultado deseado. \square

Ahora es posible demostrar el resultado principal de esta sección.

Teorema 6.2.8. *Si G es una gráfica simple y conexa que no es completa ni un ciclo impar, entonces $\chi \leq \Delta$.*

Demostración. Sea G una gráfica k -cromática que satisface las hipótesis del teorema. Al ser k -cromática, G contiene una subgráfica k -crítica H . Por ser una subgráfica de G , tenemos que $\Delta_H \leq \Delta_G$. Además, $\chi_H = \chi_G = k$, por lo que para demostrar que $\chi \leq \Delta$, basta demostrar que $k \leq \Delta_H$. Luego, podemos suponer sin pérdida de generalidad que G es k -crítica.

Por el Corolario 6.2.5, G no tiene vértices de corte. Además, como las gráficas 1-críticas y 2-críticas son completas, y las 3-críticas son ciclos impares, podemos suponer que $k \geq 4$. Si G tiene un 2-corte por vértices $\{u, v\}$, se sigue del Corolario 6.2.7 que

$$2\Delta \geq d(u) + d(v) \geq 3k - 5 \geq 2k - 1.$$

Como 2Δ es par, se obtiene $\chi = k \leq \Delta$.

Por lo tanto, podemos suponer que G es 3-conexa. Como G no es completa, existen vertices u, v y w en G que inducen un P_3 , digamos (u, v, w) . Sean $u = v_1$, $w = v_2$ y sea $v_3, \dots, v_{|V|} = v$ el orden contrario al que se exploran los vértices al ejecutar BFS en $G - \{u, v\}$ desde v . Entonces, para cada $i < |V|$, el vértice v_i es adyacente a algún v_j con $j > i$. Al colorear u y w de color 1, y el resto de los vértices de G con este orden utilizando la heurística avariciosa, al momento de elegir el color para cualquier vértice distinto de v , su padre en el árbol de BFS no está coloreado, por lo que ocurren a lo más $\Delta - 1$ colores en su vecindad. Además, al momento de colorear a v , como u y w tienen el mismo color, ocurren a lo más $\Delta - 1$ colores distintos en su vecindad, por lo que hay un color disponible para él. Así, la heurística avariciosa genera una Δ -coloración de G . \square

Teorema 6.2.9. *Para cualquier entero positivo k existe una gráfica k -cromática y libre de triángulos.*

Demostración. Proponemos una sucesión recursiva, donde cada una de las gráficas tiene las propiedades deseadas. Sean $G_1 = K_1$ y $G_2 = K_2$. Para $k \geq 2$, si G_k está definida, y su conjunto de vértices es v_1, \dots, v_n , entonces construimos a G_{k+1} a partir de G_k de la siguiente forma. Agregamos $n + 1$ vértices nuevos u_1, \dots, u_n, v , y, para cada $i \in \{1, \dots, n\}$, hacemos adyacente a u_i a cada vecino de v_i y a v .

Ahora, es fácil usar inducción para demostrar que G_k es una gráfica k -cromática y libre de triángulos para cada entero positivo k . La base se cumple trivialmente para G_1 y G_2 . Para el paso inductivo, supongamos que G_k es una gráfica k -cromática y libre de triángulos, y consideremos a G_{k+1} .

Notemos que $\{u_1, \dots, u_n\}$ es un conjunto independiente en G_{k+1} , por lo tanto, ningún triángulo puede tener a más de uno de estos vértices, y en particular, ningún triángulo puede usar al vértice v . Si (u_r, v_s, v_t, u_r) fuera un triángulo en G_{k+1} , entonces (v_r, v_s, v_t, v_r) sería un triángulo en G_k , lo que resulta en una contradicción. Luego, G_{k+1} es libre de triángulos.

Para ver que G_{k+1} es $(k + 1)$ -cromática, empecemos observando que cualquier k -coloración de G_k puede extenderse a una $(k + 1)$ -coloración de G_{k+1} asignando a u_i el mismo color que v_i , para cada $i \in \{1, \dots, k\}$, y usando un color nuevo para v . Por lo tanto, tenemos que G_{k+1} es $(k + 1)$ -coloreable; falta verificar que G_{k+1} no es k -coloreable.

Supongamos que G_{k+1} tiene una k -coloración. Cuando se restringe esta k -coloración a $\{v_1, \dots, v_n\}$, obtenemos una k -coloración de la gráfica G_k . Como G_k es k -cromática, para cada color j existe un vértice v_i , de color j , y que es adyacente a al menos

un vértice de cada color en G_k . Como u_i tiene exactamente a los mismos vecinos que v_i en G_k , se sigue que u_i también recibe el color j . De esta forma, cada uno de los k colores aparece en al menos uno de los vértices u_i . Sin embargo, ahora no hay color alguno disponible para v , contradiciendo que la coloración original era una k -coloración de G_{k+1} . Por lo tanto, G_{k+1} es una gráfica k -cromática y libre de triángulos. El resultado deseado se sigue del Principio de Inducción Matemática. \square

Capítulo 7

Gráficas Planas

7.1. Gráficas planas y aplanables

Un **arco poligonal**, o simplemente **arco**, es un subconjunto de \mathbb{R}^2 que es la unión de un número finito de segmentos de recta y que es homeomorfo al intervalo cerrado unitario $[0, 1]$. Las imágenes de 0 y 1 bajo tal homeomorfismo son los **extremos** del arco. Si P es un arco entre x y y , el conjunto $P - \{x, y\}$ es el **interior** de P . Un **polígono** es un subconjunto de \mathbb{R}^2 que es la unión de un número finito de segmentos de recta, y que es homeomorfo al círculo unitario S^1 . Al ser imágenes continuas de $[0, 1]$, los arcos, y sus uniones finitas (por ejemplo, los polígonos) son compactos, y por lo tanto, cerrados en \mathbb{R}^2 . Por lo tanto, sus complementos son abiertos.

Si P es un polígono, entonces $\mathbb{R}^2 - P$ es un abierto que tiene exactamente dos regiones ajenas, el **interior** y el **exterior** de P , denotadas $\text{int}(P)$ y $\text{ext}(P)$, respectivamente. Cada una de estas regiones tiene a todo el polígono P como frontera. Esta afirmación es el enunciado del **Teorema de la Curva de Jordan** (en su versión para polígonos). A partir de éste, es posible demostrar que cualquier arco con un extremo en $\text{int}(P)$ y el otro extremo en $\text{ext}(P)$, debe intersectar a P en al menos un punto.

Una **gráfica plana** es una pareja ordenada (V, E) de conjuntos finitos, donde los elementos de V son llamados vértices y los de E son llamados aristas, con las siguientes propiedades:

1. $V \subseteq \mathbb{R}^2$:
2. cada arista es un arco entre dos vértices.
3. el interior de una arista no contiene vértices ni puntos de otras aristas.

Una gráfica es **aplanable** si puede ser dibujada en el plano de tal forma que sus aristas se intersecten únicamente en sus extremos, es decir, en vértices de la gráfica. Un dibujo de la gráfica aplanable G con tales características es una **inmersión plana** de G . Claramente, una inmersión plana de una gráfica G corresponde a una gráfica

plana, y recíprocamente, toda gráfica plana es la inmersión plana de alguna gráfica aplanable. Por este motivo, mientras no haya ambigüedad, se utilizará el símbolo G para referirse a una gráfica plana (V, E) , a su gráfica aplanable asociada y al subconjunto $V \cup \bigcup E$ de \mathbb{R}^2 (el dibujo de la gráfica).

Si G es una gráfica plana, las regiones de $\mathbb{R}^2 - G$ son las **caras** de G . Cada cara de G es un abierto en \mathbb{R}^2 y tiene su frontera en G . Como G es acotada, exactamente una de sus caras es no acotada; ésta es la **cara exterior** de G . Las demás caras de G son **caras interiores**. El conjunto de caras de G se denota por F_G . Es posible demostrar algunas propiedades básicas de las gráficas planas utilizando topología elemental del plano, sin embargo, al no ser la finalidad de este texto, únicamente serán enunciadas. No es difícil observar que cada ciclo en una gráfica plana es un polígono. Dada una arista e de la gráfica plana G , si e está en algún ciclo de G , entonces está en la frontera de exactamente dos caras de G ; si e es un puente de G , entonces está en la frontera de exactamente una cara de G . Se sigue de la última afirmación del enunciado anterior que un bosque plano tiene exactamente una cara. Por otro lado, es intuitivamente fácil observar que en una gráfica 2-conexa con al menos 3 vértices, cada cara está acotada por un ciclo. De esta última afirmación se sigue fácilmente que en una gráfica sin lazos, plana y 3-conexa, los vecinos de cualquier vértice están en un ciclo común.

Los conceptos introducidos hasta el momento son suficientes para demostrar la existencia de gráficas que no son aplanables.

Proposición 7.1.1. K_5 no es aplanable.

Demostración. Procediendo por contradicción, supongamos que G es una inmersión plana de K_5 , con vértices v_1, \dots, v_5 . Al ser completa, cualesquiera dos vértices de G están unidos por una arista, por lo que (v_1, v_2, v_3, v_1) es un ciclo C en G . Como v_4 no puede estar en la frontera de C , debe estar en $\text{int}(C)$ o en $\text{ext}(C)$. Supongamos, sin pérdida de generalidad, que $v_4 \in \text{int}(C)$. Las aristas v_1v_4, v_2v_4 y v_3v_4 están completamente contenidas en $\text{int}(C)$ (salvo por sus extremos v_1, v_2, v_3), por lo que es posible formar tres ciclos nuevos C_1, C_2 y C_3 dados por $C_1 = (v_2, v_3, v_4, v_2)$, $C_2 = (v_3, v_1, v_4, v_3)$ y $C_3 = (v_1, v_2, v_4, v_1)$.

Claramente, $\text{int}(C_i) \subseteq \text{int}(C)$ para cada $i \in \{1, 2, 3\}$. Como cada v_i está en la frontera de C , se tiene que $v_i \in \text{ext}(C_i)$ para cada $i \in \{1, 2, 3\}$. Como v_5 es adyacente a v_i , entonces debe estar en $\text{ext}(C_i)$ para cada $i \in \{1, 2, 3\}$, por lo que también debe estar en $\text{ext}(C)$. Entonces, la arista v_4v_5 intersecta al ciclo C , lo que contradice la planaridad de G . \square

Con un argumento semejante al utilizado para demostrar la Proposición 7.1.1 es posible obtener que $K_{3,3}$ tampoco es aplanable. Es fácil verificar que estas dos gráficas son mínimas con la propiedad de ser no aplanables (basta exhibir una inmersión plana de cada gráfica al eliminar una arista). Por otro lado, es posible utilizar a K_5 y a $K_{3,3}$ para encontrar familias infinitas de gráficas no aplanables.

Cualquier gráfica obtenida a partir de una gráfica G mediante una sucesión de subdivisiones es una **subdivisión de G** o una **G -subdivisión**. Aunque algo técnica, la siguiente proposición es fácil de verificar. *subdivisión de G*

Proposición 7.1.2. *Una gráfica G es aplanable si y sólo si cualquier subdivisión de G es aplanable.*

Se sigue de la Proposición 7.1.2 que una gráfica aplanable no contiene subdivisiones de K_5 ni de $K_{3,3}$. Kuratowski demostró que el recíproco de esta afirmación también es cierto, es decir, si una gráfica no contiene subdivisiones de K_5 ni de $K_{3,3}$, entonces es aplanable. Por salir del alcance de este texto, este resultado será enunciado sin demostración.

Teorema 7.1.3 (Kuratowski). *Una gráfica G es aplanable si y sólo si G no contiene subdivisiones de K_5 ni de $K_{3,3}$.*

Es posible obtener un resultado semejante al de Kuratowski usando menores en lugar de subdivisiones. Si G es una gráfica, un **menor de G** es una gráfica obtenida a partir de G mediante una sucesión de eliminación de vértices, eliminación de aristas, o contracción de aristas. Dada una gráfica H , se dice que G contiene a H como menor, si H es un menor de G . Nuevamente, el siguiente resultado, aunque técnico, es fácil de verificar. *menor de G*

Proposición 7.1.4. *Una gráfica G es aplanable si y sólo si cualquier menor de G es aplanable.*

De nuevo, se sigue de la Proposición 7.1.4 que una gráfica aplanable no contiene a K_5 ni a $K_{3,3}$ como menor. Wagner demostró que es posible obtener un resultado semejante al Teorema de Kuratowski utilizando menores en lugar de subdivisiones.

Teorema 7.1.5. *Una gráfica es aplanable si y sólo si no contiene a K_5 ni a $K_{3,3}$ como menor.*

Es posible considerar inmersiones de gráficas en superficies distintas al plano. Es común estudiar, e.g., coloraciones de gráficas que admiten una inmersión en el toro, o en la banda de Möbius. Por otro lado, resulta bastante natural preguntarse por las gráficas que admiten una inmersión en la esfera. Recordando que la esfera sin un punto z es homeomorfa al plano, mediante la proyección estereográfica¹, no es de extrañarse que las gráficas aplanables coincidan con aquellas que admiten una inmersión en la esfera.

¹Dada una esfera S tangente al plano P en el punto y , si z es el punto antipodal a y en S , la proyección estereográfica π es la función $\pi: S - \{z\} \rightarrow P$, dada por $\pi(s) = p$ si y sólo si los puntos z, s y p son colineales.

Proposición 7.1.6. *Una gráfica es aplanable si y sólo si admite una inmersión en la esfera. Más aún, si G es una gráfica aplanable y f es una cara en alguna inmersión plana de G , entonces G admite una inmersión plana cuya cara exterior tiene la misma frontera que f .*

Demostración. Si G tiene una inmersión \tilde{G} en la esfera, entonces eligiendo cualquier punto z de la esfera que no sea un punto de \tilde{G} , y aplicando la proyección estereográfica desde z , obtenemos una inmersión plana de G . El recíproco se obtiene análogamente.

Para la segunda afirmación, consideremos además a la cara \tilde{f} de \tilde{G} que corresponde a f . Al elegir a z como un punto dentro de \tilde{f} , la imagen de \tilde{G} bajo la proyección estereográfica desde z es una inmersión plana de G con la propiedad buscada. \square

7.2. Gráficas duales

Sean G una gráfica plana y f una cara de G . Mientras no haya lugar a ambigüedad, se utilizará $\partial(f)$ para denotar a la frontera de f . Si G es conexa, entonces $\partial(f)$ corresponde con un camino cerrado en el que cada puente se recorre dos veces (de *ida* y de *regreso*); si $\partial(f)$ no contiene puentes, entonces es un ciclo de G .

Se dice que una cara f es **incidente** con los vértices y las aristas en su frontera. Se dice que una arista **separa** a las caras que inciden en ella (incluso si es la misma cara, en el caso de los puentes). El grado, $d(f)$ de la cara f , es el número de aristas que inciden en ella, i.e., el número de aristas en $\partial(f)$, donde los puentes se cuentan doble.

Sean G una gráfica plana y conexa y f una cara de G , por **subdividir** a f , se entenderá agregar una nueva arista e que une a dos vértices en la frontera de f , y tal que todos sus puntos interiores se encuentran en $\text{int}(f)$. El resultado de esta operación es una gráfica plana, $G+e$, con exactamente una cara más que G ; todas las caras de G son también caras de $G+e$, excepto f , que es reemplazada por dos nuevas caras f_1 y f_2 , que son separadas por la arista e . Recíprocamente, si e es una arista que no es de corte en una gráfica plana G , entonces separa a dos caras distintas, f_1 y f_2 , por lo que $G-e$ es una gráfica plana que tiene exactamente una cara menos que G ; todas las caras de G son también caras de $G-e$, excepto f_1 y f_2 , que son reemplazadas por una nueva cara f , tal que $\text{int}(f) = \text{int}(e) \cup \text{int}(f_1) \cup \text{int}(f_2)$.

Es posible definir un operador que a cada gráfica plana le asocia otra gráfica plana, que cuenta con algunas propiedades semejantes a las de la gráfica de la cuál proviene. Dada una gráfica plana G , la **gráfica dual**, o simplemente **dual** de G , denotada G^* , es una gráfica que tiene un vértice f^* por cada cara f de G , una arista e^* por cada arista e de G , y tal que dos vértices f^* y g^* son extremos de la arista e^* en G^* si y sólo si las caras correspondientes f y g son separadas por la arista e en G . De la definición de dual se sigue que si e es un puente de G , entonces separa a una cara f de sí misma, por lo que e^* es un lazo en f^* ; recíprocamente, si e es un lazo de G , que separa a la cara f , al interior de e , de la cara g , entonces f^* tiene

como único vecino a g^* , al que es adyacente mediante la arista e^* , que resulta ser un puente.

Como se mencionó anteriormente, la dual G^* de la gráfica plana G es también plana. Para verificar esto, podemos considerar la siguiente inmersión de G^* en el plano. Partiendo de la inmersión plana de G , elegimos un punto en el interior de f para representar a f^* y, para cada arista e incidente en f , dibujamos una arista e^* de tal forma que cruce a e una única vez, y no cruce otras aristas de G . Este procedimiento está ilustrado en la Figura 7.1, donde la dual se dibuja en azul.

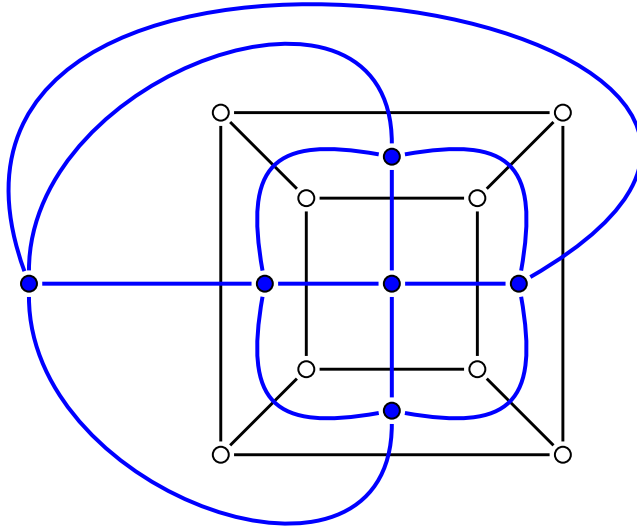


Figura 7.1: El 3-cubo, Q_3 , y su dual (en azul).

Aún cuando la definición de la dual G^* de una gráfica plana G es la de una gráfica (y no la de una gráfica plana), es conveniente considerar a G^* como una gráfica plana con la inmersión plana antes descrita. En tal caso, G^* será referida usualmente como la **dual plana** de G . Naturalmente, al ser una gráfica plana, es posible considerar a la dual plana G^{**} de G^* . Cuando G es conexa, G^{**} resultará isomorfa a G (ver los ejercicios de la sección). Es importante recalcar que gráficas planas isomorfas pueden tener duales no isomorfos. Por lo tanto, el concepto de dual sólo tiene sentido para gráficas planas, no puede ser extendido a gráficas aplanables.

A partir de la definición de G^* , es fácil verificar que $|V_{G^*}| = |F_G|$, $|E_{G^*}| = |E_G|$ y que para cada $f \in F_G$, $d_{G^*}(f^*) = d_G(f)$. De las relaciones anteriores, es fácil obtener la siguiente proposición.

Proposición 7.2.1. *Si G es una gráfica plana, entonces*

$$\sum_{f \in F} d(f) = 2|E|.$$

Demostración. Sea G^* la dual de G . A partir de las relaciones previas a esta proposición, y de la Proposición 1.3.1 se obtienen las siguientes igualdades

$$\sum_{f \in F_G} d(f) = \sum_{f^* \in V_{G^*}} d(f^*) = 2|E_{G^*}| = 2|E_G|.$$

□

triangulación
plana

Una gráfica plana y conexa donde todas las caras tienen grado tres es una **triangulación plana** o simplemente **triangulación**. A partir de las igualdades previas a la Proposición 7.2.1, es fácil observar que una gráfica plana y conexa es una triangulación plana si y sólo si su dual es cúbica. Es posible demostrar que toda gráfica simple y plana es la subgráfica generadora de una triangulación. Por otro lado, ninguna supergráfica simple de una triangulación plana es aplanable. Por este motivo, a las triangulaciones planas también se les conoce como **gráficas aplanables máximas**.

7.3. La fórmula de Euler

El siguiente resultado es una fórmula que relaciona los números de vértices, aristas y caras en una gráfica plana. Fue descubierta y demostrada originalmente por Euler, en el estudio de los sólidos platónicos (1752), por lo que se conoce como la **Fórmula de Euler**.

Fórmula
de Euler

Teorema 7.3.1. Sea G una gráfica. Si G es conexa y plana, entonces

$$|V| - |E| + |F| = 2.$$

Demostración. Por inducción sobre el número de caras de G . Si $|F| = 1$, entonces la única cara de G es la cara exterior, por lo que G es acíclica y al ser conexa, es un árbol. Por la Proposición 2.2.5, tenemos que $|E| = |V| - 1$, con lo que claramente se cumple la igualdad buscada.

Sea G una gráfica plana conexa con al menos dos caras. Como G tiene más de una cara, tiene algún ciclo, y por lo tanto alguna arista e que no es un puente. Luego, $G - e$ es una gráfica plana y conexa, y dado que las dos caras separadas por e en G se combinaron para formar una nueva cara en $G - e$, ésta tiene una cara menos que G . Así, aplicando la hipótesis inductiva,

$$|V_{G-e}| - |E_{G-e}| + |F_{G-e}| = 2,$$

y dado que $|V_{G-e}| = |V_G|$, $|E_{G-e}| = |E_G| - 1$ y $|F_{G-e}| = |F_G| - 1$, concluimos que

$$|V_G| - |E_G| + |F_G| = 2.$$

El resultado deseado se sigue del Principio de Inducción Matemática. □

Corolario 7.3.2. *Todas las inmersiones planas de una gráfica aplanable fija tienen el mismo número de caras.*

Demostración. Sean G y H inmersiones planas de la misma gráfica aplanable. Como G y H son isomorfas, tienen el mismo número de vértices y el mismo número de aristas. Aplicando el Teorema 7.3.1, obtenemos

$$|F_G| = |E_G| - |V_G| + 2 = |E_H| - |V_H| + 2 = |F_H|.$$

□

El siguiente corolario acota superiormente el número de aristas de una gráfica aplanable con una función lineal de su número de vértices. Como consecuencia, se tiene que para las gráficas aplanables $O(|E|) = O(|V|)$, por lo que muchos algoritmos tienen un desempeño mucho mejor en gráficas aplanables que en gráficas donde $O(|E|) = O(|V|^2)$. Por ejemplo, BFS y DFS tienen un tiempo de ejecución $O(|V|)$ sobre gráficas aplanables.

Corolario 7.3.3. *Sea G una gráfica simple con al menos tres vértices. Si G es aplanable, entonces*

$$|E| \leq 3|V| - 6.$$

Demostración. Podemos suponer sin pérdida de generalidad que G es conexa. Como $|V| \geq 3$ y G es simple, cada cara de G incide en al menos tres aristas, es decir, para cada $f \in F$, tenemos $d(f) \geq 3$, por lo que

$$\sum_{f \in F} d(f) \geq 3|F|.$$

Por la Proposición 7.2.1 tenemos que

$$2|E| \geq 3|F|.$$

A partir de esta desigualdad y aplicando el Teorema 7.3.1, tenemos que

$$\begin{aligned} 2 &= |V| - |E| + |F| \\ &\leq |V| - |E| + \frac{2}{3}|E| \\ &= |V| - \frac{1}{3}|E|. \end{aligned}$$

De donde claramente se obtiene

$$|E| \leq 3|V| - 6.$$

□

Corolario 7.3.4. *Si G es una gráfica simple y aplanable, entonces*

$$\delta \leq 5.$$

Demostración. El resultado es trivial si G tiene menos de 3 vértices. Si $|V| \geq 3$, entonces por el Corolario 7.3.3, se tiene

$$\delta|V| \leq \sum_{v \in V} d(v) = 2|E| \leq 6|V| - 12.$$

De la desigualdad anterior obtenemos

$$12 \leq |V|(6 - \delta),$$

que se cumple sólo si $\delta \leq 5$. □

Otra consecuencia interesante de la fórmula de Euler es una demostración, sin utilizar el Teorema de la Curva de Jordan, de que K_5 y $K_{3,3}$ son no aplanables.

Corolario 7.3.5. *K_5 no es aplanable.*

Demostración. Si K_5 fuera aplanable, entonces el Corolario 7.3.3 implicaría que

$$10 = |E_{K_5}| \leq 3|V_{K_5}| - 6 = 9.$$

Por lo tanto, K_5 no es aplanable. □

Corolario 7.3.6. *$K_{3,3}$ no es aplanable.*

Demostración. Procediendo por contradicción, supongamos que $K_{3,3}$ es plana, y sea G una inmersión plana de $K_{3,3}$. Como $K_{3,3}$ tiene cuello 4, cada cara de G debe tener grado al menos 4, por lo que la Proposición 7.2.1 implica que

$$4|F| \leq \sum_{f \in F} d(f) = 2|E| = 18,$$

y como $|F|$ es un entero, obtenemos que

$$|F| \leq 4.$$

Por lo tanto, se sigue del Teorema 7.3.1 que

$$2 = |V| - |E| + |F| \leq 6 - 9 + 4 = 1,$$

lo que resulta en una contradicción. □

7.4. El Teorema de los Cinco Colores

En 1852 el matemático Francis Guthrie notó que todos los condados de Inglaterra podían ser coloreados utilizando únicamente 4 colores, de tal suerte que, condados que tenían frontera común recibieran colores distintos, preguntándose entonces si sería posible colorear cualquier mapa de esta forma. Francis Guthrie compartió esta pregunta con su hermano Frederick, quien se la propuso a su entonces profesor, Augustus De Morgan. De Morgan quedó cautivado con esta pregunta, y en 1852 escribió a Sir Rowan Hamilton al respecto:

A student of mine asked me today to give him a reason for a fact which I did not know was a fact—and do not yet. He says that if a figure be anyhow divided, and the compartments differently colored so that figures with any portion of common boundary line are differently colored—four colors may be wanted, but not more ...Query cannot a necessity for five or more be invented? ...What do you say? And has it, if true, been noticed? ...The more I think of it the more evident it seems. If you retort with some very simple case which makes me out a stupid animal, I think I must do as the Sphynx did.

Sin embargo, Hamilton no se interesó por el problema, respondiendo “I am not likely to attempt your ‘quaternion of colors’ very soon”. De Morgan intentó interesar a algunos otros matemáticos con este problema, sin embargo, la comunidad permaneció relictante a trabajar en el problema hasta 1878, cuando Arthur Cayley lo presentó en una reunión de la Sociedad Matemática de Londres. A partir de ese momento, este problema ha sido uno de los favoritos de la comunidad matemática y los divulgadores de la ciencia.

En particular, el Problema de los Cuatro Colores, resuelto afirmativamente en 1976 por Appel y Haken quienes dieron una demostración asistida por computadora, impulsó el desarrollo de la teoría de gráficas durante muchos años, en los que se desarrollaron diversas técnicas y conceptos nuevos en el intento por resolverlo. Este problema, que permaneció abierto por más de 100 años, tuvo algunos momentos históricos notables, como la demostración presentada por Kempe en 1879, que fue aceptada por la comunidad matemática por una década, hasta que Heawood encontró un error en la misma. Sin embargo, las técnicas introducidas por Kempe eran bastante ingeniosas, y Heawood las utilizó para demostrar el Teorema de los Cinco Colores. A continuación se presenta dicho teorema, con la demostración clásica que hace uso de cadenas de Kempe.

Teorema 7.4.1. *Toda gráfica aplanable es 5-coloreable.*

Demostración. Procediendo por contradicción, supongamos que existen gráficas aplanables que no son 5-coloreables. Como toda gráfica aplanable es 6-coloreable, entonces debe de existir una gráfica 6-crítica G . Como las gráficas críticas son simples,

se sigue del Corolario 7.3.4 que $\delta \leq 5$. Por otro lado, al ser crítica, la Proposición 6.2.2 implica que $\delta \geq 5$. Por lo tanto, $\delta = 5$ y podemos elegir a un vértice v de grado 5 en G . Al ser G 6-crítica, $G - v$ admite una 5-coloración (V_1, \dots, V_5) , y como G no es 5-coloreable, los cinco colores deben de ocurrir en la vecindad de v . Por lo tanto, podemos suponer que los vecinos de v aparecen en orden natural alrededor de v , y son v_1, \dots, v_5 , donde $v_i \in V_i$ para cada $i \in \{1, \dots, 5\}$.

Para cualesquiera $i, j \in \{1, \dots, 5\}$, sea G_{ij} la subgráfica de G inducida por $V_i \cup V_j$, es decir $G_{ij} = G[V_i \cup V_j]$. Notemos que si v_i y v_j no pertenecen a la misma componente de G_{ij} , entonces podemos intercambiar los colores i y j en la componente que contiene a v_i , con lo que obtenemos una nueva 5-coloración de $G - v$ en la que v_i recibe el color j , y por lo tanto i no está representado en la vecindad de v . Pero esto último no puede suceder, pues podríamos asignarle el color i a v , obteniendo así una 5-coloración para G . Por lo tanto, v_i y v_j se encuentran en la misma componente de G_{ij} . Sea P_{ij} una $v_i v_j$ -trayectoria en G_{ij} ², y sea C el ciclo $vv_1 P_{13} v_3 v$.

Necesariamente, C separa a v_2 de v_4 , por lo que se sigue del Teorema de la Curva de Jordan que la trayectoria P_{24} debe intersectar a C en algún punto. Como G es una gráfica plana, este punto de intersección debe ser un vértice de G (distinto de v , pues v no es un vértice de P_{24}), lo cual resulta en una contradicción, pues los vértices de C tienen colores 1 y 3, mientras que los vértices de P_{24} tienen colores 2 y 4. \square

Teorema 7.4.2. *Los siguientes tres enunciados son equivalentes.*

1. *Toda gráfica aplanable es 4-coloreable por vértices.*
2. *Toda gráfica plana es 4-coloreable por caras.*
3. *Toda gráfica simple, cúbica, aplanable y 2-conexa por aristas es 3-coloreable por aristas.*

Demostración. Supongamos que cada gráfica aplanable es 4-coloreable por vértices, y sea G una gráfica plana G . Entonces, la dual G^* de G es una gráfica aplanable, que por hipótesis cumple $\chi_{G^*} = 4$. El resultado deseado se sigue de recordar que $\chi_G^* = \chi_{G^*}$.

Supongamos ahora que toda gráfica plana es 4-coloreable por caras, y sea G una gráfica simple, cúbica, aplanable y 2-conexa por aristas. Sea además \tilde{G} una inmersión plana de G . Por hipótesis, \tilde{G} tiene una 4-coloración por caras, c . En lugar de la elección usual de $\{1, 2, 3, 4\}$ como conjunto de colores, utilizaremos a los elementos del grupo $\mathbb{Z}_2 \times \mathbb{Z}_2$ (el grupo V de Klein) como colores. Entonces, es posible obtener una 3-coloración por aristas de \tilde{G} asignándole a cada arista la suma de los colores de las caras que separa.

Si c_i, c_j y c_k son los colores asignados a las tres caras incidentes en un vértice v , entonces los colores asignados a las aristas son $c_i + c_j$, $c_j + c_k$ y $c_k + c_i$. Como G es

²Estas trayectorias son conocidas como cadenas de Kempe.

2-conexa por aristas, toda arista separa a dos caras distintas, por lo que ninguno de los colores asignados a las aristas es $(0, 0)$, ya que cada elemento en $\mathbb{Z}_2 \times \mathbb{Z}_2$ es su propio inverso, y los colores c_i, c_j y c_k son todos distintos. Por lo tanto, la coloración por aristas obtenida es una 3-coloración. Explorando la tabla de sumar de $\mathbb{Z}_2 \times \mathbb{Z}_2$, es fácil observar que los tres colores asignados a las aristas son distintos, por lo que la coloración es propia. Así, encontramos una 3-coloración por aristas propia de \widetilde{G} , y por lo tanto también de G .

Para la última implicación procederemos por contradicción. Supongamos que toda gráfica simple, cúbica, aplanable y 2-conexa por aristas es 3-coloreable por aristas, y que existe una gráfica aplanable que no es 4-coloreable por vértices. Entonces, existe una gráfica G , aplanable y 5-crítica. Sea \widetilde{G} una inmersión plana de G , y sea H una supergráfica generadora de \widetilde{G} que sea una triangulación plana. La dual H^* de H es una gráfica simple, 2-conexa por aristas, cúbica y aplanable, por lo que tiene una 3-coloración por aristas propia, digamos, (E_1, E_2, E_3) . Para cualesquiera $i, j \in \{1, 2, 3\}$, con $i \neq j$, sea H_{ij}^* la subgráfica de H^* inducida por $E_i \cup E_j$. Como en cada vértice de H^* incide una arista de E_i y una arista de E_j , se tiene que H_{ij}^* es una unión ajena de ciclos, y es, por lo tanto, 2-coloreable por caras. Cada cara de H^* es la intersección de una cara de H_{12}^* y H_{23}^* . Luego, dadas 2-coloraciones por caras de H_{12}^* y H_{23}^* , es posible obtener una 4-coloración por caras de H^* , asignando a cada cara f la pareja ordenada de los colores asignados a las caras de H_{12}^* y H_{23}^* cuya intersección es f . Como $H^* = H_{12}^* \cup H_{23}^*$, y las coloraciones para H_{12}^* y H_{23}^* son propias, es fácil verificar que esta coloración es una 4-coloración por caras propia de H^* . Como H es una supergráfica de G , se tiene que

$$5 = \chi_G \leq \chi_H = \chi_{H^*}^* \leq 4,$$

lo que resulta en una contradicción. □

Índice de conceptos

- H -libre, 15
- S -componentes, 89
- k -abanico, 59
- k -coloración, 87
- k -coloración por aristas, 83
- k -coloreable, 87
- k -crítica, 89
- k -cromática por aristas, 83
- k -factor, 15
- k -factorizable, 15
- xy -corte por vértices, 55
- árbol, 30
- árbol de bloques, 51
- árbol generador, 30
- amalgamamiento, 25
- apareamiento, 75
- apareamiento máximo, 75
- aplanable, 95
- autocomplementaria, 24
- bipartita, 8
- bipartita completa, 9
- bloque, 50
- bloque terminal, 51
- camino, 16
- centro, 18
- ciclo, 9
- ciclo hamiltoniano, 70
- circuito euleriano, 63
- clan, 7
- clausura, 72
- cluster, 9
- complemento, 24
- componentes conexas, 18
- condensación, 27
- conexa, 8
- conexidad, 54
- conexidad local, 54
- conexidad local por aristas, 60
- conexidad por aristas, 60
- corte por aristas, 18
- corte por vértices, 18
- cubierta, 77
- cuello, 27
- descomposición, 50
- descomposición por orejas, 54
- diámetro, 18, 27
- digráfica, 26
- distancia, 18, 27
- dual, 98
- escindible, 9
- exgrado, 26
- Fórmula de Euler, 100
- fuertemente conexa, 27
- geodésica, 18
- gráfica, 5
- gráfica de intersección, 9
- gráfica de intervalos, 10
- gráfica de líneas, 10

- gráfica plana, 95
- gráfica simple, 6
- gráfica subyacente, 26
- grado, 7

- heurística avariciosa, 88
- hoja, 35
- homomorfismo, 87

- inconexa, 8
- independiente, 7
- ingrado, 26
- inmersión plana, 95
- internamente ajenos, 27

- longitud, 26

- menor de G , 97
- multipartita, 9
- multipartita completa, 9

- número cromático, 88
- número cromático por aristas, 83
- número de clan, 8
- número de independencia, 7
- no separable, 50

- obstrucción mínima, 19

- pancíclica, 73
- paseo, 16, 27
- paseo euleriano, 27
- perfecto, 75
- producto cartesiano, 23
- producto fuerte, 23
- producto tensorial, 23
- propiedad hereditaria, 19
- punto, 19

- radio, 18
- regular, 7
- rejilla de $m \times n$, 24

- subdivisión de G , 97
- subgráfica generadora, 15
- subgráfica inducida, 15

- trayectoria, 9, 16, 27
- trayectoria hamiltoniana, 70
- triangulación plana, 100

- unión ajena, 25

- vértice de corte, 18
- vértice separador, 50
- vecindad, 6