



Teórica 8: Planaridad y coloreo de grafos

1. Planaridad (*opcional*)

Volvamos al problema de Möbius y el rey de la tercera clase. Möbius planteó lo siguiente: Había una vez un rey con cinco hijos. En su testamento escribió que, después de su muerte, los hijos deberían dividir el reino en cinco regiones de forma tal que cada una limitara con las otros cuatro. Y preguntó si esto era posible.

Supongamos que esto fuera posible y construyamos un grafo G poniendo como vértices cada una de las regiones y haciendo dos vértices adyacentes si las regiones son limítrofes. En este caso, habremos construido K_5 . Si podemos dibujar este grafo en un papel de forma tal que no se crucen las aristas, el testamento del rey se podría cumplir. En caso contrario, no se podría.

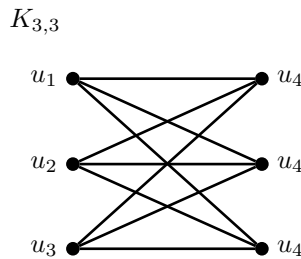
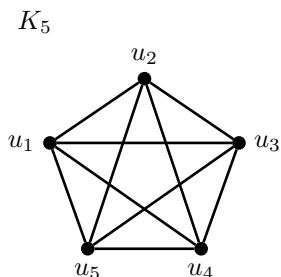
Definición 1.

- Una **representación planar** de un grafo G es un conjunto de puntos en el plano, que se corresponden con los vértices de G , unidos por curvas, que se corresponden con las aristas de G , sin que estas curvas se crucen entre sí.
- Un grafo es **planar** si admite una representación planar.
- Dada una representación planar de un grafo G , una **región** es el conjunto de todos los puntos alcanzables desde un punto (que no sea un vértice ni parte de una arista) sin atravesar vértices ni aristas.
- Todo representación planar de un grafo tiene exactamente una región de área infinita, la **región exterior**.
- La **frontera** de una región es el circuito que rodea a la región (puede tener vértices y aristas repetidos).
- El **grado o tamaño** de una región es el número de aristas que tiene su frontera.

El problema de planaridad consiste en, dado un grafo G , decidir si G es planar. Este problema está bien resuelto computacionalmente, es decir, existen algoritmos polinomiales para resolverlos.

El concepto de grafo planar es utilizado en ingeniería electrónica, mecánica y civil, por ejemplo en el diseño de circuitos impresos, en el diseño de rutas y autopistas, en sistemas de irrigación mediante canales que no pueden cruzarse, en localización de instalaciones en mapas.

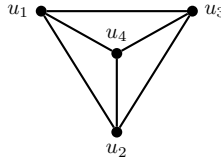
Proposición 1. K_5 y $K_{3,3}$ son grafos no planares. K_5 es el grafo no planar con el menor número de vértices y $K_{3,3}$ es el que tiene el menor número de aristas.





Demostración: Por contradicción. Supongamos que K_5 es planar. En la representación planar de K_5 el ciclo $C : v_1, v_2, v_3$ es una curva cerrada en el plano y v_4 debe estar adentro o afuera de ese ciclo.

Supongamos que está adentro:



Ahora, como $(v_3, v_5) \in X$, v_5 no puede estar dibujado en la región $C_1 : v_1, v_2, v_4$. Como $(v_2, v_5) \in X$, v_5 no puede estar dibujado en la región $C_2 : v_1, v_3, v_4$ y como $(v_1, v_5) \in X$, v_5 no puede estar dibujado en la región $C_3 : v_3, v_2, v_4$. Entonces v_5 debería estar en la región exterior, pero la arista (v_4, v_5) cruza otras aristas. Entonces v_4 no puede estar en el interior de C . Con la misma idea, podemos ver que v_4 no puede estar en el exterior de C , llegando a que K_5 no puede tener una representación planar.

La demostración de que $K_{3,3}$ es no planar la dejamos para más adelante. ■

Proposición 2. *Si un grafo contiene un subgrafo no-planar es no-planar. (Si G es planar, todo subgrafo de G también es planar.)*

Demostración: Por contradicción supongamos que $H \subset G$, G planar y H no planar.

Como G es planar tiene una representación planar. De esa representación podemos obtener una de H eliminando las aristas (curvas) y vértices (puntos) que no pertenecen a H . Como en la representación de G no hay cruzamientos de curvas, al eliminar cualquier conjunto de curvas y puntos tampoco habrá. Entonces H tiene una representación planar. ■

Teorema 1 (Ecuación poliedral de Euler, 1752). *Si $G = (V, X)$ es un grafo conexo planar entonces cualquier representación planar de G determina $r = m - n + 2$ regiones en el plano.*

Demostración: Haremos inducción en la cantidad de aristas, m .

Caso base: Si $m = 0$, G no tiene ciclos y como es conexo, G es árbol. Entonces $r = 1$ y $m - n + 2 = 0 - 1 + 2 = 1$. Vale la propiedad.

Paso inductivo: Para demostrar el paso inductivo, consideremos un grafo G conexo y planar con $m \geq 1$ aristas, n vértices y r regiones.

Nuestra hipótesis inductiva es: *si G' es conexo y planar con $m' \geq 1$, ($m' < m$) aristas, n' vértices y r' regiones, entonces $r' = m' - n' + 2$.*

Si G es árbol, entonces $r = 1$ y $m = n - 1$. Como $m - n + 2 = n - 1 - n + 2 = 1$ vale la propiedad.

Ahora supongamos que G no es árbol. Por lo tanto G tiene al menos un circuito simple. Sea $e \in X$ una arista de un ciclo simple y $G' = G - e = (V, X \setminus \{e\})$. Por Proposición de la clase de árboles, G' es conexo y por Proposición 2, G' es planar. Entonces G' cumple todas las hipótesis de la HI y podemos aplicársela. Aplicando la HI a G' , tenemos que $r' = m' - n' + 2$, donde r' es la cantidad de regiones de G' , m' la cantidad de aristas y n' la cantidad de vértices.



Al eliminar una arista de un circuito se unen dos regiones, pasando G' a tener $r' = r - 1$ regiones. Reemplazando,

$$r - 1 = r' = m' - n' + 2 = m - 1 - n + 2 \implies r = m - n + 2,$$

que es lo que queremos probar. ■

Corolario 1. Si G es conexo y planar con $n \geq 3$, entonces $m \leq 3n - 6$.

Demostración: Como G es conexo y planar con $n \geq 3$, el grado de toda región es al menos 3. Entonces, aplicando la ecuación de Euler,

$$2m = \sum_{f \in R} d(f) \geq 3r = 3m - 3n + 6 \implies 3n - 6 \geq m.$$

■
Corolario 2. K_5 es no planar.

Corolario 3. Si G es conexo, bipartito y planar con $n \geq 3$, entonces $m \leq 2n - 4$.

Demostración: Como G es conexo, planar y bipartito con $n \geq 3$, el grado de toda región es al menos 4. Entonces, aplicando la ecuación de Euler,

$$2m = \sum_{f \in R} d(f) \geq 4r = 4m - 4n + 8 \implies 2n - 4 \geq m.$$

■
Corolario 4. $K_{3,3}$ es no planar.

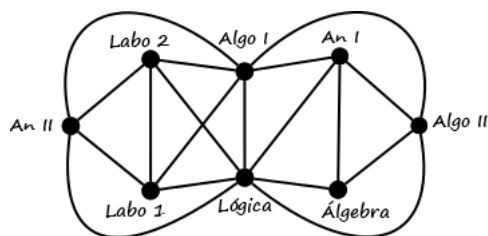
2. Coloreo de vértices

Analicemos el siguiente problema: Supongamos que se tienen cuatro aulas y las siguientes materias con sus respectivos horarios para un mismo día:

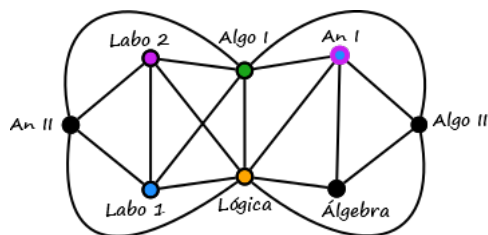
Algebra	8 a 12 hs.
Análisis I	10 a 14 hs.
Análisis II	14 a 18 hs.
Lógica	11 a 15 hs.
Algoritmos I	12 a 16 hs.
Algoritmos II	9 a 13 hs.
Laboratorio 1	14 a 18 hs.
Laboratorio 2	14 a 18 hs.

¿Existe una forma de asignar aulas de forma que se puedan dictar todas las materias respetando los horarios?

Podemos modelar el problema mediante un grafo $G = (V, X)$ donde el conjunto de vértices representa a las materias, y hacemos dos vértices adyacentes si las materias correspondientes se solapan en sus horarios. Éste es el grafo que obtenemos:

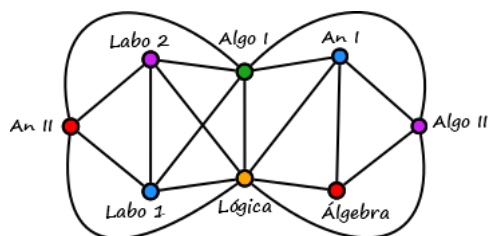


Supongamos que tenemos las aulas verde, naranja, violeta y celeste. Le asignamos una, por ejemplo la naranja, a Lógica. Algoritmos I no puede usar el aula naranja porque de 12 a 15 hs. se cursan las dos materias. Le asignamos la verde a Algo I. Laboratorio 1 no puede utilizar ninguna de esas dos aulas, porque se solapa con ambas materias, entonces le asignamos el aula celeste. A Análisis 1 no le podemos asignar ni el aula naranja ni la verde, pero sí podemos la celeste o la violeta. Para Laboratorio 2, la única alternativa es la violeta, porque se solapa con Labo 1, Algo 1 y Lógica. Por ahora vamos así:



Cuando queremos asignarle un aula a Análisis II, nos encontramos con que no le podemos asignar la celeste (porque se solapa con Labo1), ni la verde (porque se solapa con Algo I), ni la violeta (porque se solapa con Labo2), ni la naranja (porque se solapa con Lógica)... Ops! Nos encontramos en problemas.

Pero podemos ver que si agregamos un aula, la roja, si podemos dictar todas las materias:



Este tipo de problemas motiva las siguientes definiciones:

Definición 2.

- Un **coloreo de los vértices** de un grafo $G = (V, E)$ es una asignación $f : V \rightarrow C$, tal que $f(u) \neq f(v) \forall (u, v) \in E$.
- Los elementos de C son llamados **colores**. Muchas veces los colores son enteros positivos.
- Para todo entero positivo k , un **k -coloreo** de G es un coloreo de los vértices de G que usa exactamente k colores.
- Un grafo G se dice **k -coloreable** si existe un k -coloreo de G .
- El **número cromático** de G , $\chi(G)$, es el menor número de colores necesarios para colorear los vértices de G .



- Un grafo G se dice **k -cromático** si $\chi(G) = k$.

Ejemplo 1.

- $\chi(K_n) = n$.
- Si G es un grafo bipartito con $m > 0$, entonces $\chi(G) = 2$.
- Si H_{2k} es un circuito simple par, entonces $\chi(H_{2k}) = 2$.
- Si H_{2k+1} es un circuito simple impar, entonces $\chi(H_{2k+1}) = 3$.
- Si T es un árbol con $n > 1$, entonces $\chi(T) = 2$.

Hay numerosas aplicaciones que se modelan mediante coloreo de grafos o alguna de sus variantes, como asignación de frecuencias, asignación de registros, asignación de recursos, planificación horaria.

Para grafos en general, no se conocen algoritmos polinomiales para encontrar su número cromático.

Notación: Dado un grafo G y un coloreo de sus vértices, llamaremos H_{pq} al subgrafo inducido de G por los vértices que están pintados con color p o con color q .

2.1. Cotas para el número cromático de un grafo

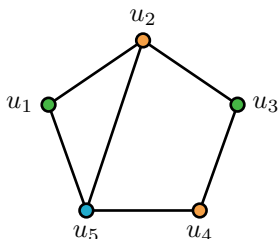
Comencemos estudiando algunas de cotas de χ .

Proposición 3. Si H es un subgrafo de G entonces $\chi(H) \leq \chi(G)$.

Demostración. Sea $G = (V, X)$, $H \subset G$, $H = (V_H, X_H)$. Dado un $\chi(G)$ -coloreo de G definido por $f : V \rightarrow C$, f restringido a V_H define un coloreo de H con a lo sumo $\chi(G)$ colores (ya que para todo $(u, v) \in X_H$, por ser $X_H \subset X$ se cumple que $(u, v) \in X$ y por lo tanto $f(u) \neq f(v)$). Entonces H es $\chi(G)$ -coloreable, lo que implica que $\chi(H) \leq \chi(G)$. ■

Definición 3. Una **clique** en un grafo es un subgrafo completo maximal. El **número clique**, $\omega(G)$, de un grafo es el número de vértices de una clique máxima de G .

Proposición 4. En todo grafo G , se cumple que $\chi(G) \geq \omega(G)$.



$$\chi(G) \leq 3 \text{ (porque exhibimos un 3-coloreo)}$$

$$\omega(G) \geq 3 \text{ (porque } \{u_1, u_2, u_5\} \text{ es una clique de tamaño 3)}$$

$$\chi(G) = \omega(G) = 3$$

Demostración. Sea $H \subseteq G$ una clique máxima de G . Como H es $K_{\omega(G)}$, $\chi(H) = \omega(G)$. Por la Proposición 2, $\chi(G) \geq \chi(H) = \omega(G)$. ■

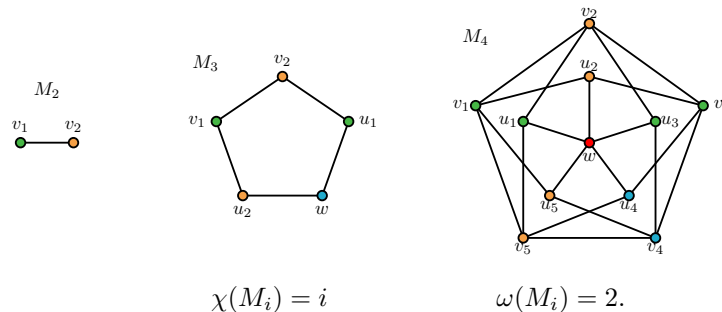


Tampoco se conocen algoritmos polinomiales para encontrar una clique máxima de un grafo (es más, existe para este problema sí, y sólo si, existe para el número cromático), entonces no podemos calcular esta cota de forma eficiente. Sin embargo, mediante una heurística podemos encontrar una clique *grande*, y si tenemos suerte y encontramos un coloreo con esa cantidad de colores, sabremos que ese es el número cromático y el número clique del grafo, como sucede en el ejemplo anterior.

Lamentablemente, esta no es una buena cota, en el sentido de que hay grafos donde estos dos valores difieren tanto como uno quiera, como lo muestran los grafos de Mycielski, definidos inductivamente de la siguiente manera:

Definición 4.

1. $M_1 = K_1$
2. $M_2 = K_2$
3. Para $i \geq 2$, M_{i+1} se construye a partir de M_i de la siguiente forma:
 - Si M_i tiene p vértices, v_1, \dots, v_p , M_{i+1} tendrá $2p + 1$ vértices, $v_1, \dots, v_p, u_1, \dots, u_p, w$, donde u_i es copia de v_i .
 - El conjunto de aristas de M_{i+1} tendrá todas las aristas de M_i , las aristas uniendo u_i con los vecinos de v_i en M_i y las aristas uniendo w con cada u_i .



Proposición 5. Sea M_i el grafo de Mycielski de orden $i \geq 2$. Entonces $\chi(M_i) = i$ y $\omega(M_i) = 2$.

Demostración. Ejercicio. ■

La proposición 5 muestra que la cota dada por la Proposición 4 puede ser tan mala como uno quiera. Esto es, existen grafos donde estos dos valores difieren tanto como uno quiera.

Proposición 6. Si $\Delta(G)$ es el grado máximo de G entonces

$$\chi(G) \leq \Delta(G) + 1.$$

La demostración de esta proposición la vamos a dejar para más adelante.

Si el grafo es conexo, esta cota sólo se cumple por igualdad para grafos completos o circuitos impares. El siguiente teorema la mejora en el resto de los casos.

Teorema 2 (Brooks). Sea G un grafo conexo que no es un circuito impar ni un grafo completo. Entonces

$$\chi(G) \leq \Delta(G).$$



Demostración: Esta demostración es opcional.

Sea $G = (V, X)$ un grafo conexo que no es un circuito impar ni un grafo completo. Vamos a considerar dos posibilidades:

- $\Delta(G) = 2$: G es un camino o un ciclo, distinto a K_1, K_2, C_{2k+1} . Entonces $\chi(G) = 2 = \Delta(G)$.
- $\Delta(G) \geq 3$: Por contradicción. Sea $G = (V, X)$ un contraejemplo minimal. Es decir, G conexo con $\Delta(G) \geq 3$, $G \neq K_n, C_{2k+1}$, no $\Delta(G)$ -coloreable y $\forall v \in V$, $G - v$ es $\Delta(G - v)$ -coloreable o no cumple las hipótesis. En todos los casos $G - v$ es $\Delta(G)$ -coloreable.

Lema 1. En todo $\Delta(G)$ -coloreo de $G - v$, los vecinos de v en G , usan todos los colores y $d_G(v) = \Delta(G) \forall v \in V$.

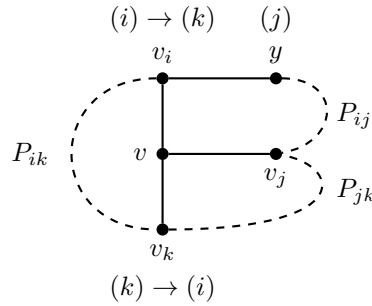
Sean $N(v) = \{v_1, \dots, v_{\Delta(G)}\}$ los vecinos de v y consideremos un $\Delta(G)$ -coloreo de $G - v$ donde el vértice v_i es pintado con color i . Para $i \neq j$, sea H_{ij} el subgrafo inducido por los vértices de $G - v$ pintados con colores i o j en ese $\Delta(G)$ -coloreo.

Lema 2. v_i y v_j pertenecen a la misma componente conexa de H_{ij} .

Lema 3. Si P_{ij} es la componente conexa de H_{ij} que contiene a v_i y a v_j , entonces P_{ij} es un camino en H_{ij} .

Lema 4. $P_{ij} \cap P_{ik} = \{v_i\}$, para colores $i \neq j \neq k$.

Como G no es completo, G tiene un vértice v tal que existen $v_i, v_j \in N(v)$ y $(v_i, v_j) \notin X$ y sean i y j los colores de estos vértices en un $\Delta(G)$ -coloreo de $G - v$ (seguro tienen color distinto por Lema 1). En el camino entre v_i y v_j en H_{ij} (es un camino por Lema 3) hay un vértice y adyacente a v_i , $y \neq v_j$ porque $(v_i, v_j) \notin X$. Obviamente, y tiene color j . Como $3 \leq \Delta(G) = d(v)$ por Lema 1, existe otro vértice $v_k \in N(v)$, $v_k \neq y$ porque v_k no tiene color j por Lema 1. Si intercambiamos los colores i y k en el camino P_{ik} de H_{ik} obtenemos un nuevo coloreo de $G - v$, donde v_i tiene color k y v_k color i . Podemos hacer esto porque $P_{ij} \cup P_{ik} = \{v_i\}$.



Llamamos P'_{jk} al camino entre v_j y el vértice que ahora tiene color k , v_i en H'_{jk} (el subgrafo inducido por los vértices que ahora tiene color j o k). Como y sigue teniendo color j y $(v_i, y) \in X$, $y \in P'_{jk}$. También $y \in P_{ij} - v_i \subset P'_{ij}$. Entonces $y \in P'_{jk} \cup P'_{ij}$, contradiciendo el Lema 4, porque $y \neq v_j$.

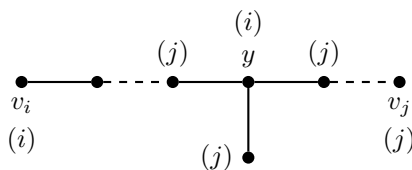
Entonces no puede existir G contraejemplo.

Demostración Lema 1: Sea $v \in V$. Si en un $\Delta(G)$ -coloreo de $G - v$ hay un color no usado en la vecindad de v en G , se le puede asignar ese color a v y obtener un $\Delta(G)$ -coloreo de G , contradiciendo la elección de G . ■

Demostración Lema 2: Si no fuera así, podríamos intercambiar los colores i y j en la componente conexa de v_j , obteniendo un $\Delta(G)$ -coloreo de $G - v$ donde el color j no es usado por los vecinos de v , contradiciendo el Lema 1. ■

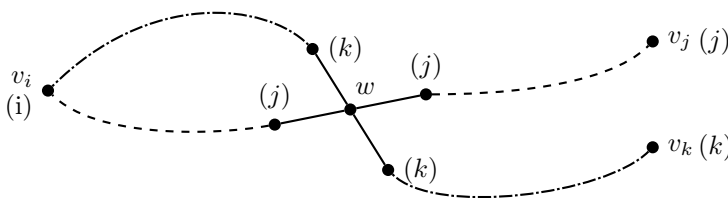


Demostración Lema 3: En el $\Delta(G)$ -coloreo de $G - v$, v_i contiene exactamente un vecino de color j , porque si tuviese más habría un color no usado en su vecindad, podríamos ponerle ese color a v_i y quedaría libre el color i para v y obtendríamos un $\Delta(G)$ -coloreo de G . Más aun, P_{ij} no puede contener un vértice de grado ≥ 3 en H_{ij} . Porque si fuera así, sea y el primer vértice de grado ≥ 3 en H_{ij} yendo desde v_i a v_j en P_{ij} . Supongamos que tiene color i (es lo mismo si tiene color j). Entonces al menos 3 vecinos de y tienen color j .



Hay un color no usado en la vecindad de y . Se lo podemos asignar a y así v_i y v_j quedarían en distintas componentes conexas de H_{ij} (como es el primero de grado ≥ 3 no puede haber otro camino), contradiciendo el Lema 2. ■

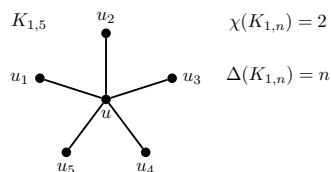
Demostración Lema 4: Suponemos que no, $\exists w \in P_{ij} \cap P_{ik}, w \neq v_i$.



Entonces w es adyacente a dos vértices de color j y a dos de color k . Como $d(w) = \Delta(G)$, hay un color $s \neq i$ no usado en la vecindad de w en el $\Delta(G)$ -coloreo de $G - v$. Le podemos poner el color s a w y así tendríamos un $\delta(G)$ -coloreo de $G - v$ con v_i y v_j en distintas componentes conexas de H_{ij} , contradiciendo el Lema 2. ■

Esta cota puede ser tan mala como uno quiera, como lo muestra el siguiente ejemplo.

Ejemplo 2.



3. Coloreo de grafos planares (*opcional*)

Ya mencionamos en las primeras clases el Teorema de los 4 colores. Ahora lo enunciaremos formalmente. No lo demostraremos, ya que, como dijimos, la demostración es muy complicada e involucra un programa informático. Nos conformaremos con demostrar que todo grafo planar es 5-coloreable que es mucho más fácil y está al alcance del curso.



Teorema 3 (Teorema de los 4 colores - Appel, Haken, 1976). *Si G es un grafo planar, entonces*

$$\chi(G) \leq 4.$$

Teorema 4 (Heawood, 1890). *Si $G = (V, X)$ es un grafo planar, entonces*

$$\chi(G) \leq 5.$$

Demostración: Haremos inducción en la cantidad de vértices, n .

Caso base: Si $n \leq 5$, obviamente $\chi(G) \leq 5$.

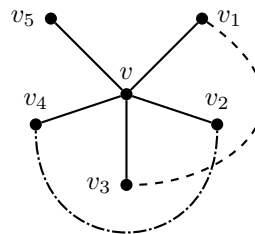
Paso inductivo: En este paso analizaremos un grafo planar G con n , $n \geq 6$, vértices.

La hipótesis inductiva es: **Todo grafo planar con n' ($n' < n$) vértices es 5-coloreable.**

Como G es planar, entonces $\exists v \in V$ con $d(v) \leq 5$ (ejercicio de la práctica). Veamos que siempre podemos obtener un 5-coloreo de $G - v$ que no use un color de esos 5 en la vecindad de v .

- Si $\exists v$ con $d(v) \leq 4$: Como $G - v$ es un grafo planar con $n - 1$ vértices cumple las hipótesis de la hipótesis inductiva. Entonces, por H.I., $G - v$ es 5-coloreable. En ese coloreo hay por lo menos un color no usado en la vecindad de v en G .
- Si para todo vértice u , $d(u) \geq 5$:
 - Si $\exists v$ tal que el 5-coloreo de $G - v$ (que existe por H.I.) no usa un color en la vecindad de v en G ya tenemos el coloreo de $G - v$ que buscamos.
 - Si $\forall v$ con $d(v) = 5$, el 5-coloreo de $G - v$ (que existe por H.I.) usa todos los colores en la vecindad de v en G , vamos a transformarlo en un coloreo con las características que buscamos.

Sean v_1, v_2, v_3, v_4, v_5 los vecinos de v ordenados en sentido horario según están dibujados en una representación planar de G . Llamos c_i al color del vértice v_i en un 5-coloreo de $G - v$. Llamemos H_{13} al subgrafo inducido por los vértices con color c_1 o c_3 .



- Si v_1 y v_3 pertenecen a distintas componentes conexas de H_{13} , podemos intercambiar el color c_1 con el c_3 en la componente conexa de v_1 . Como v_3 queda con color c_3 , en este nuevo 5-coloreo de $G - v$, c_1 no está usado en la vecindad de v , estando ahora en el caso anterior.
- Si v_1 y v_3 pertenecen a la misma componente conexa de H_{13} , tiene que haber un camino bicolor c_1, c_3 entre ellos (en H_{13}). Entonces analizamos H_{24} , el subgrafo inducido por los vértices con color c_2 o c_4 .

Veamos que los vértices v_2 y v_4 no pueden estar en la misma componente conexa de H_{24} . Supongamos por el absurdo que sí lo están. En ese caso, habría un camino bicolor c_2, c_4 entre ellos



(en H_{24}). Como H_{13} y H_{24} no tienen vértices en común, las aristas de estos dos caminos se tiene que cruzar en la representación planar de G , llegando a una contradicción.

Entonces, como si v_1 y v_3 pertenecen a la misma componente conexa de H_{13} implica que v_2 y v_4 están en distintas componentes conexas de H_{24} , intercambiamos los colores c_2 y c_4 en la componente de v_2 . Ahora el color c_2 no es usado en la vecindad de v , y conseguimos un 5-coloreo de $G - v$ del caso anterior.

En todos los casos conseguimos un 5-coloreo de $G - v$ que no usa un color en la vecindad de v . Podemos asignarle ese color a v para obtener un 5-coloreo de G y así demostrar que $\chi(G) \leq 5$.

■

4. Algoritmos para coloreo de grafos

Como ya mencionamos, coloreo de grafos es un problema *difícil*, computacionalmente no resuelto, ya que no se conocen algoritmos polinomiales para resolverlo.

Al ser un problema con muchas aplicaciones, existen muchos enfoques algorítmicos que lo abordan:

- Heurísticas y metaheurísticas.
- Algoritmos basados en backtracking (por ejemplo: DSATUR, Brelaz, 1979).
- Algoritmos exactos basados en programación lineal entera.

Veamos algunas alternativas.

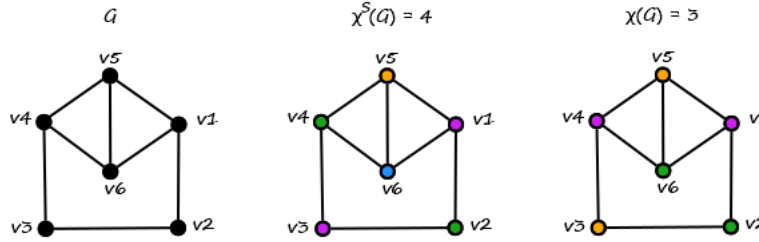
4.1. Algoritmo (heurística) secuencial (S)

Es la heurística más natural para colorear los vértices de un grafo. Su correctitud es trivial.

```
secuencialColoreo( $G$ )  
  entrada:  $G = (V, X)$  con un orden en los vertices  $v_1, \dots, v_n$   
  salida:  $f$  coloreo de los vertices de  $G$   
  
  para  $i = 1$  hasta  $n$  hacer  
     $f[v_i] \leftarrow \min\{h \in \mathbb{N} : f[v_j] \neq h \ \forall (v_j, v_i) \in X, 1 \leq j \leq i - 1\}$   
  fin para  
  retornar coloreo definido por  $f$ 
```

En el siguiente ejemplo vemos que mediante este algoritmo no obtenemos un coloreo óptimo.

Ejemplo 3. Llamamos $\chi^S(G)$ a la cantidad de colores utilizada por el coloreo retornado por la heurística secuencial:



Es posible acotar lo peor que nos puede pasar al utilizar esta heurística. Sin embargo, no es un algoritmo aproximado, ya que esta cota no es en función del valor óptimo, χ .

Definición 5.

$$u_S(G, v_1, v_2, \dots, v_n) = \max_{1 \leq i \leq n} \min\{i, d(v_i) + 1\}.$$

Proposición 7. Si $\chi^S(G)$ es el número de colores usado por el algoritmo secuencial para colorear G cuando los vértices son considerados en el orden v_1, \dots, v_n , entonces

$$\chi(G) \leq \chi^S(G) \leq u_S(G, v_1, v_2, \dots, v_n).$$

Demostración: En el algoritmo secuencial, el color asignado a un vértice v_i corresponde al menor color que no ha sido usado hasta el momento en su vecindad. Llegado el momento de colorear a v_i se han coloreado $i - 1$ vértices, por lo que seguro $f[v_i] \leq i$. Además, el peor caso es que ya hayan sido coloreados todos sus vecinos y con colores distintos, lo que muestra que $f[v_i] \leq d(v_i) + 1$.

Por lo tanto, $f[v_i] \leq \min\{i, d(v_i) + 1\}$. Entonces, el máximo color usado al finalizar el algoritmo es

$$\chi^S(G) = \max_{1 \leq i \leq n} f[v_i] \leq \max_{1 \leq i \leq n} \min\{i, d(v_i) + 1\} = u_S(G, v_1, v_2, \dots, v_n).$$

■

Ahora nos resulta directa la demostración de la Proposición 6.

Demostración Proposición 6:

$$\chi(G) \leq \chi^S(G) \leq \max_{1 \leq i \leq n} \min\{i, d(v_i) + 1\} \leq \max_{1 \leq i \leq n} d(v_i) + 1 = \Delta(G) + 1$$

■

(Desde acá hasta el final del apunte es opcional)

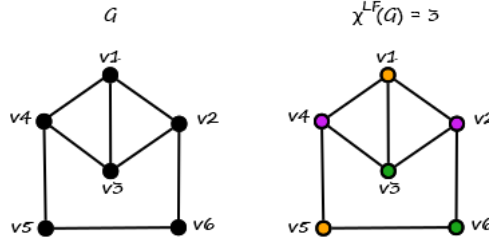
El orden en que consideramos los vértices para ser coloreados influye en el resultado obtenido. Esta cota induce a pensar que un orden donde los vértices de mayor grado se coloquen primero podría dar un mejor resultado de la heurística secuencial. Esto se basa en que, como $f[v_i] \leq \min\{i, d(v_i) + 1\}$, para los vértices que tienen el segundo de estos valores grande, se trataría que el primero sea chico.

Definición 6 (Orden *Largest First* - LF). En este orden, u_1, \dots, u_n , los vértices son ordenados de mayor a menor grado, $d(u_1) \geq d(u_2) \geq \dots \geq d(u_n)$.



En el siguiente ejemplo, el resultado retornado por la heurística secuencial ordenando sus vértices mediante LF mejoró con respecto al orden considerado anteriormente y retornó un coloreo óptimo. Ninguna de estas dos cosas se dan siempre.

Ejemplo 4. Llamamos $\chi^{LF}(G)$ al valor retornado por la heurística secuencial con orden LF:



La siguiente proposición muestra que la cota anterior dada por el orden LF es mejor que la cota en cualquier otro orden de los vértices (lo que no significa que el algoritmo de mejor, es sólo la cota mejor).

Proposición 8. Si $u_{LF}(G) = u_S(G, u_1, u_2, \dots, u_n)$ donde u_1, u_2, \dots, u_n están ordenados según LF, entonces

$$u_{LF}(G) \leq \min u_S(G, v_1, v_2, \dots, v_n)$$

donde el mínimo está tomado sobre todos los ordenes posibles, v_1, \dots, v_n .

Demostración: Ejercicio de la práctica. ■

Proposición 9. Otra cota (mejor) para el número de colores usados por el algoritmo secuencial aplicado con orden v_1, \dots, v_n es:

$$\chi^S(G) \leq 1 + \max_{1 \leq i \leq n} \{d_{G_i}(v_i)\}$$

donde $d_{G_i}(v_i)$ es el grado del vértice v_i en el subgrafo de G inducido por v_1, v_2, \dots, v_i .

Demostración. En el algoritmo secuencial, al momento de colorear un vértice se consideran prohibidos los colores utilizados por sus vecinos ya pintados. Esta cantidad prohibida es, a lo sumo, $d_{G_i}(v_i)$, ya que lo peor que podría pasar es que sus vértices vecinos hayan sido coloreados todos con colores distintos. Por lo tanto, en $\{1, \dots, d_{G_i}(v_i) + 1\}$ hay, al menos, un color permitido para colorear a v_i .

Entonces, el máximo color usado al finalizar el algoritmo usado es

$$\chi^S(G) = \max_{1 \leq i \leq n} f[v_i] \leq 1 + \max_{1 \leq i \leq n} d_{G_i}(v_i).$$

Esta cota sugiere utilizar un orden donde $d_{G_i}(v_i)$ sea lo más chico posible en cada paso. ■

Definición 7 (Orden *Smallest Last* - SL). Definimos el orden:

1. fijar como v_n al vértice de mínimo grado de G .
2. para $i = n - 1, \dots, 1$ definir como v_i el vértice de grado mínimo en el subgrafo de G inducido por $V \setminus \{v_n, v_{n-1}, \dots, v_{i+1}\}$.



Definición 8. Definimos

$$u_{SL}(G) = 1 + \max_{1 \leq i \leq n} \min_{1 \leq j \leq i} \{d_{G_i}(v_j)\}$$

donde $d_{G_i}(v_j)$ es el grado del vértice v_j en el grafo inducido por $V \setminus \{v_n, v_{n-1}, \dots, v_{i+1}\}$.

Se puede demostrar (ejercicio) que:

- $\chi_{SL}(G) \leq u_{SL}(G)$, donde $\chi_{SL}(G)$ es el máximo color usado por la heurística secuencial con orden SL.
- $u_{SL}(G) \leq u_{LF}(G)$.
- La heurística secuencial con orden SL colorea un grafo planar con 6 colores o menos.

4.2. Algoritmo secuencial con intercambio (SI)

Esta heurística es una mejora del algoritmo secuencial. Cuando es necesario utilizar un color nuevo, se trata de recolorear los vértices ya coloreados para no necesitar ese nuevo color.

Supongamos que estamos coloreando el vértice v y necesitamos un color nuevo, el $k+1$. Entonces v tiene vecinos coloreados con todos los colores $1, \dots, k$. El algoritmo intenta modificar el color de algunos vecinos de v_i para liberar un color de los $1, \dots, k$ y no necesitar el color nuevo.

Si existen p y q dos colores utilizados en el coloreo parcial, tal que en todas las componentes conexas de H_{pq} los vértices adyacentes a v tienen el mismo color, podemos realizar un p, q -intercambio en las componentes de H_{pq} con vértices adyacentes a v con color p (podría ser q). De esta manera, obtendremos un coloreo parcial de G con el color p no utilizado en la vecindad de v .

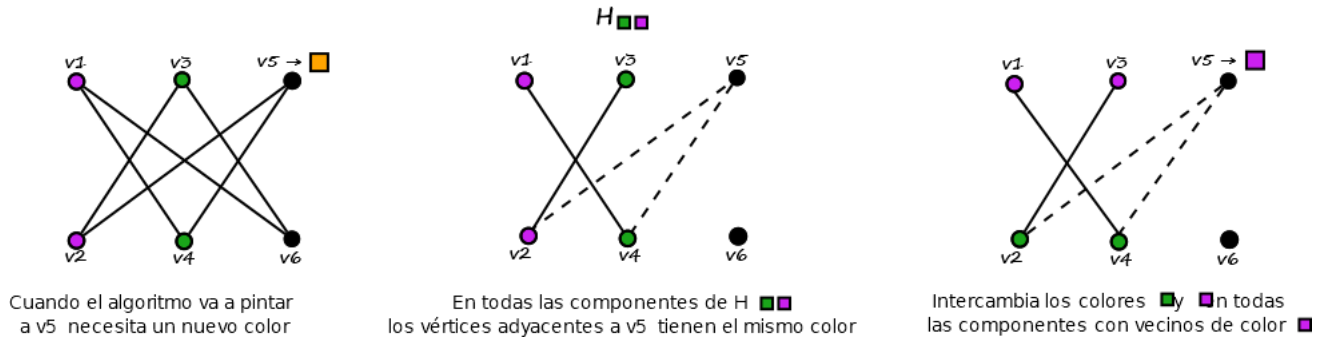
Este procedimiento se llama p, q -intercambio.

```
secuencialConIntercambio( $G$ )
  entrada:  $G = (V, X)$  con un orden en los vertices  $v_1, \dots, v_n$ 
  salida:  $f$  coloreo de los vertices de  $G$ 

   $k \leftarrow 0$ 
  para  $i = 1$  hasta  $n$  hacer
     $q \leftarrow \min\{h \in \mathbb{N} : f[v_j] \neq h \ \forall (v_j, v_i) \in X, 1 \leq j \leq i-1\}$ 
    si  $q \leq k$  hacer
       $f[v_i] \leftarrow q$ 
    sino
      si existen  $1 \leq p < q \leq k$ , tales que
        un  $p, q$ -intercambio libera  $p$  de  $N(v_i)$  entonces
          realizar el  $p, q$ -intercambio
           $f[v_i] \leftarrow p$ 
      sino
         $f[v_i] \leftarrow q$ 
         $k \leftarrow k + 1$ 
    fin si
  fin para
  retornar coloreo definido por  $f$ 
```



Ejemplo 5. Cuando el algoritmo quiere pintar el vértice v_5 evita utilizar un nuevo color realizando un intercambio.



El algoritmo con intercambio (SI) requiere un mayor esfuerzo computacional y no siempre logra una mejor solución que el algoritmo secuencial.

Se puede demostrar que:

- SI colorea un grafo bipartito con 2 colores (ejercicio).
- SI con el ordenamiento SL colorea un grafo planar con 5 colores como máximo.

4.3. Algoritmo secuencial con backtracking (exacto)

Por último veremos un algoritmo exacto que usa la técnica de backtracking para coloreo de vértices, es decir, que siempre colorea al grafo G con $\chi(G)$ colores. En cada paso se extiende una solución parcial coloreando un vértice aún no coloreado.

Una solución puede estar representada por (a_1, \dots, a_n) , con a_i indicando el color asignado al vértice v_i . Una solución parcial estará representada por (a_1, \dots, a_k) , $k \leq n$.

Dada una solución parcial (a_1, \dots, a_k) , $k \leq n$ que representa un coloreo parcial válido, construiremos el conjunto de sus vértices hijos, $\text{Sucesores}(a, k)$, asegurando que siga siendo un coloreo válido. Entonces el color asignado al vértice v_k no puede ser un color ya utilizado para pintar un vértice vecino de v_k . Además, para evitar generar soluciones que no serán óptimas (usan más colores que el mejor coloreo encontrado hasta el momento) y soluciones simétricas (se considera una solución que usará la misma cantidad de colores) se aplican procesos de poda.

$$\text{Sucesores}(a, k) = \{(a, a_k) : a_k \in U\}$$

con $j \in U$ si:

- j no es color asignado a un vecino de v_k ya coloreado (poda por factibilidad)
- $j \leq d(v_k) + 1$ (poda por simetría)
- $1 \leq j \leq l + 1$ (poda por simetría)
- si ya se encontró un coloreo del grafo con q colores entonces $j \leq q - 1$ (poda por optimalidad),

donde l es el máximo color usado en la solución parcial a , $l = \max_{1 \leq i \leq k-1} a_i$.

Con estas restricciones se hace una búsqueda completa. En el árbol de búsqueda se abre una rama a partir de cada vértice (correspondiente a un coloreo de v_1, \dots, v_{k-1}), para cada elemento de U . Se avanza por las ramas coloreando los siguientes vértices hasta que ocurre alguna de las siguientes situaciones:



- se llegó a un vértice con $U = \emptyset$: a partir de esta situación se hace *backtracking* a partir de v_{k-1} .
- se coloreó v_n : se encontró un nuevo coloreo del grafo, hay que actualizar q y hacer *backtracking*.

En base a estas ideas, podemos implementar un algoritmo iterativo de backtracking. Utilizaremos estas variables:

- q : cantidad de colores usados en la mejor solución encontrada hasta el momento.
- f^* : mejor solución encontrada hasta el momento.
- k : vértice siendo considerado.
- f : solución parcial.
- l : cantidad de colores utilizados en la solución parcial actual.
- l_k : l para el vértice v_k .
- $cotaInf$: cota inferior para el número cromático del grafo.

Como los colores son indistinguibles, sin pérdida de generalidad fijamos en 1 el color de v_1 , $f[v_1] = 1$.

Cuando se llega a colorear el vértice v_n se obtiene un coloreo completo. Se actualiza f^* y q al mayor color utilizado l . Además se borran los colores mayores o iguales a l de los conjuntos U 's y se poda la rama desde el vértice que utiliza el color l por primera vez ($k \leftarrow j - 1$), porque ya no nos interesa encontrar otra solución con l o más colores.



```
bktColoreo( $G$ )
  entrada:  $G = (V, X)$ 
  salida:  $f^*$  coloreo de los vertices de  $G$  con  $\chi(G)$  colores

   $q \leftarrow n + 1$ ,  $f[v_1] = 1$ ,  $k \leftarrow 1$ ,  $l \leftarrow 1$ 
  avanzar  $\leftarrow$  VERDADERO
  repetir
    si avanzar entonces
       $k \leftarrow k + 1$ ,  $l_k \leftarrow l$ , determinar  $U_k$ 
    fin si
    si  $U_k = \emptyset$  entonces
      avanzar  $\leftarrow$  FALSO,  $k \leftarrow k - 1$ ,  $l \leftarrow l_k$ 
    sino
       $j \leftarrow \min U_k$ ,  $U_k \leftarrow U_k \setminus \{j\}$ ,  $f(v_k) \leftarrow j$ 
      si  $j > l$  entonces  $l \leftarrow l + 1$ 
      si  $k < n$  entonces
        avanzar  $\leftarrow$  VERDADERO
      sino
         $f^* \leftarrow f$ 
         $j \leftarrow \min\{i \text{ tal que } f[v_i] = l\}$ 
        borrar  $l, l + 1, \dots, q - 1$  de  $U_1, \dots, U_{j-1}$ 
         $q \leftarrow l$ ,  $l \leftarrow q - 1$ ,  $k \leftarrow j - 1$ 
        avanzar  $\leftarrow$  FALSO
      fin sin
    fin sin
  hasta  $k = 1$  o  $q = cotaInf$ 
  retornar  $f^*$  y  $q$ 
```

5. Coloreo de aristas

Hay problemas donde su modelo requiere colorear las aristas del grafo en lugar de los v rtices.

Definici n 9.

- Un *coloreo v lido de las aristas* de un grafo G es un asignaci n de colores a las mismas en la cual dos aristas que tienen un v rtice en com n no tengan el mismo color.
- El * ndice crom tico* de un grafo G , $\chi'(G)$, es el menor n mero de colores con que se pueden colorear las aristas de un grafo.

Teorema 5 (Teorema de Vizing:). *Para todo grafo G se verifica que*

$$\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1.$$

6. Bibliograf a recomendada

- Cap tulos 9 y 10 de J. Gross and J. Yellen, *Graph theory and its applications*, CRC Press, 1999.
- Cap tulos 8 y 9 de J. Bondy and U. Murty, *Graph theory with applications*, Macmillan, 5ta edici n, 1982.