MA3705. Algoritmos Combinatoriales. 2014.
Profesor: José Soto
Escriba(s): Ian Letter, Obed Ulloa e Ian Vidal.
Fecha: 26 de Septiembre 2014 .

Cátedra 13

1. Recuerdo Cátedra Anterior

Teorema 1. *Un matching M es de tamaño máximo en $G \iff G$ no posee caminos M-aumentantes.*

Observación 1. No es trivial verificar si un matching posee o no caminos M-aumentantes.

Ejemplo 1. En este caso dada la cantidad de aristas es dificil identificar si un matching es de tamaño máximo. Para facilitar la tarea definiremos lo que es un cubrimiento.

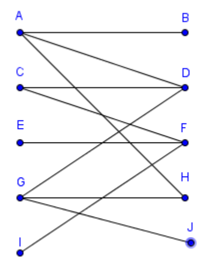


Figura 1. El conjunto $\{A, F, G, H\}$ es un cubrimiento.

Definición 1. Sea $G = (V, E)$ grafo. Sea $C \subseteq V$. Diremos que C es cubrimiento (por vértices) si toda arista $e \in E$ incide en C (es decir, es incidente a algun vértice de C).

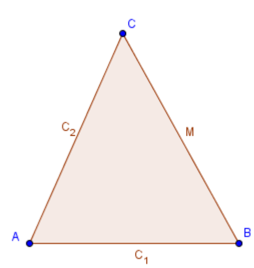
Teorema 2. *Sea M matching de G, C cubrimiento de G. Luego $|M| \leq |C|$.*

Demostración. Cada arista de M requiere un vértice distinto de C para ser cubierto (pues las aristas de M son disjuntas). Cada vértice de C cubre a lo más una arista de M $\Rightarrow |M| \leq |C|$. $\square$

Observación 2. Esto es útil para demostrar que un matching no puede ser más que grande que cierta cota. Se puede observar que en el ejemplo 1 $|C| = 4 \Rightarrow |M| \leq 4$.

Corolario 1. *:* $\max\limits_{M, matching} |M| \leq \min\limits_{C, cubrimiento} |C|$.

Observación 3. No es siempre posible obtener la igualdad.



Ejemplo 2.

Figura 2. El cubrimiento mínimo tiene tamaño 2, mientras que el matching máximo tiene tamaño 1.

En este caso se observa que trivialmente $|M| = 1$, pero se tiene que $|C| = 2$.

Observación 4. : Si hay igualdad es posible crear un matching de tamaño máximo. Ahora bien, no es fácil verificar si se puede obtener la igualdad o no. El caso particular en el que nos enfocaremos en lo que sigue es cuando G es un grafo bipartito.

Definición 2. G es bipartito ssi $\exists L, R \subseteq V; L \cap R = \emptyset \wedge L \cup R = V$ t.q: $\forall e \in E, e = uw, u \in L, w \in R$, es decir, toda arista tiene un extremo en L y otro en R. El grafo $G = (V, E)$ también se denotará como $G = (L, R; E)$.Además las aristas $uw \in E$ se identificarán como el par ordenado $(u, w) \in L \times R$.

Observación 5. Lo anterior es equivalente a decir que G no tiene ciclos de largo impar.

Teorema 3. *König: Sea G grafo bipartito. Luego:*

$$\max_{M, matching} |M| = \min_{C, cubrimiento} |C|$$

Demostración 1. Esta demostración se hará algoritmicamente. Para ello crearemos una rutina que dado M matching, encuentre un camino M-aumentante (y así podemos encontrar un matching mayor) o reporte un cubrimiento de C con $|C| = |M|$. Notemos que si este procedimiento siempre se puede llevar a cabo, entonces se obtiene el resultado.

Con lo anterior podemos partir de $M = \emptyset$ y en cada iteración aumentar el matching con el camino M-aumentante, luego cuando no se pueda aumentar comprobamos que $|M| = |C|$.

Luego sea $G = (L, R; E)$. Llamaremos A a los vértices M-expuestos en L y B a los M-expuestos en R. Por simplicidad del algoritmo, creamos el siguiente digrafo auxiliar $D(G, M)$.

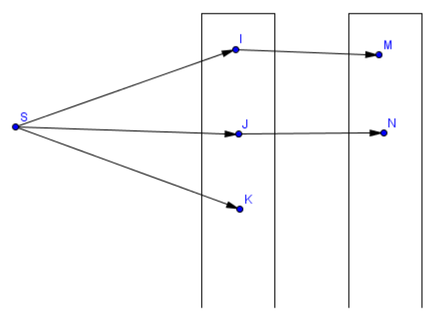


Figura 3.

O sea, dirigimos la arista e de L a R si e no está en el matching, y de R a L si lo está. Además, agregamos s un nodo auxiliar que esta conectado con un arista con todos los nodos en A. Este grafo tiene por nodos $\{s\} \cup V$ y los arcos de $D(G, M)$ son de 3 tipos:

- $\{sv;\ v \in A\}$ (Aristas dirigidas de s a los nodos de A)
- $\{vw;\ v \in L, w \in R, vw \in E/M\}$ (Aristas que no están en el matching dirigidas de L a R)
- $\{wv; v \in L, w \in R, vw \in M\}$ (Aristas que si están en el matching dirigidas de R a L)

Necesitaremos el siguiente resultado:

Lema 1. *Todo s-w dicamino en $D(G, M)$ con $w \in B$ está en correspondencia con un camino M-aumentante.*

Demostración 2. Esto se hará por construcción. En primer lugar; ¿Cómo encontramos un s-w dicamino? Para esto usamos BFS desde s, cuando encuentre w paramos; y así obtenemos el dicamino.

Luego, para encontrar el camino M-aumentante en G utilizaremos el siguente algoritmo:

Algoritmo 1: Algoritmo para encontrar camino M-aumentante en G

Construir $D(G, M)$
BFS desde s
$T \leftarrow \{$nodos alvanzables desde s por BFS$\}$
if $\exists w \in T \cap B$ **then**
| **Return** Camino $M - aumentante$ asociado al $s - w$ dicamino.
end
else
| **Return** Cubrimiento $C = (L \setminus T) \cup (R \cap T)$
end

Lo anterior será suficiente para concluir el resultado, ya que el C retornado en la parte anterior, será el cubrimiento que buscamos. Para ello demostraremos lo siguiente:

Teorema 4. *Si M es máximo $\Rightarrow$ el conjunto $C = (L \setminus T) \cup (R \cap T)$ es un cubrimiento que cumple $|C| = |M|$.*

Demostración. Hay que ver primero que C es cubrimiento y luego que $|M| = |C|$.

- C es cubrimiento de G: supongamos $\exists e = vw \in E$ no cubierto por C. *s.p.g.* $v \in L, w \in R$. Luego $v \in L \cap T \wedge w \in R \setminus T.$, ya que si no e estaría cubierto por C.

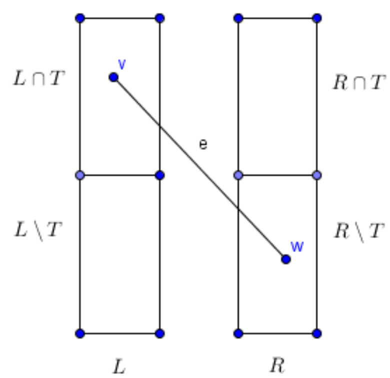


Figura 4.

  - Si e no está en el matching, entonces por construcción, se dirige de L a R. En ese caso conecta a v, que está en T y por tanto es alcanzable, con w, que no está en T. Así se obtiene una contradicción ya que conecta a un alcanzable con un no alcanzable.
  - Si e está en el matching, entonces se dirige de R a L. Con esto sabemos que v no está en A, pues $e \in M$ incide en v (y A corresponde a los M-expuestos de L).

    Con esto, además se deduce que $\deg^-(v) = 1$, pues está cubierto por el matching (Ya que la única forma de entrar a v es mediante M).

    Lo último implica que $C = (L \setminus T) \cup (R \cap T)$ como $v \in T$ entonces es alcanzable solo por e. Osea, si v es alcanzable en algún momento se ha pasado por w uno no alcanzable. Contradicción.

  Con eso ya tenemos que C es cubrimiento.

- Ahora veamos que $|M| = |C|$.

  Sabemos por König que $|M| \leq |C|$. Luego solo falta ver que $|C| \leq |M|$. Para esto contruiremos una inyección de C a M, sea:

$$\begin{aligned} \phi : C &\to M \\ v &\mapsto e_v \end{aligned}$$

donde e_v es la ínica arista de M incidente a v. Falta ver que esto siempre existe, es decir, que está bien definido, y que es inyección.

Para lo primero, (recordar que $C = (L \setminus T) \cup (R \cap T)$) sea $v \in L \setminus T$. Si v fuese M-expuesto entonces $v \in A \subseteq T$, lo que es una contradicción. Luego v es M-cubierto.

Luego sea $w \in R \cap T$. Si w fuese M-expuesto entonces $w \in B$. Pero C fue construido por el algoritmo anterior, y si w hubiese estado en B este hubiese reportado un camino M-aumentante. Esto es una contradicción ya que M es máximo. Por lo tanto w es M-cubierto. Luego todo nodo en C es $M-$cubierto, por lo tanto existe una única arista incidente a él (por ser M-matching). Por lo tanto, la función anterior está bien definida.

Falta ver que es inyección. Supongamos que no; luego $\exists v \in L \setminus T$; $w \in R \cap T$ t.q: $vw \in M$ (Notemos que definimos que la funcion que entregaba la arista incidente a v. Luego si no existe arista del tipo anterior y por ser M matching la funcion será inyectiva). Por construcción del digrafo entonces e va de w a v. Lo que es una contradicción, pues $w \in T$ y $v \in T^c$. Es decir, v no es alcanzable y el único arco que entra a v viene desde un vértice alcanzable. Con esto, se tiene que la función es una inyección.

□

Sintetizando, el algoritmo para encontrar M de tamaño máximo es:

Algoritmo 2: Algoritmo para encontrar M de tamaño máximo

```
M ← ∅
while True do
    Usar la rutina anterior
    if ∃ camino M-aumentante then
        Aumentar M con el camino
    end
    else
        Return M y C
    end
end
```

Teorema 5. *El algoritmo anterior es correcto y resuelve el problema de atching máximo M^* y cubrimiento mínimo C^* en $\mathcal{O}(n(m+n))$.*

Demostración. La demostración de por que es correcta se deduce del teorema 4. Para el orden, notemos lo siguiente; el **While** se hace a lo mas $|M^*|$ veces. Donde M^* es el matching de tamaño máximo. por otro lado,dentro de cada iteración la rutina demora $\mathcal{O}(n+m)$ (Para ello notar que es hecha en base a BFS). Notando que $|M^*| \leq n$ se concluye el resultado. □

Observación 6. El mejor algoritmo para el problema anterior es el de Hopcroft-Kar, que es $\mathcal{O}(m\sqrt{n})$. A grandes rasgos, este funciona encontrado en cada paso una colección de caminos M-aumentante disjuntos.

Ejemplo 3. Veremos una aplicación de problema de cubrimiento mínimo. En un museo, ¿cuál es la menor cantidad de guardias,que debemos ubicar en las esquinas, para cuidar cada pasillo?.

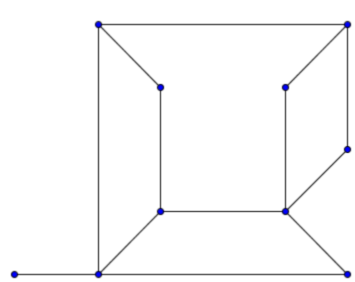


Figura 5.

Es claro que la solución para esto es la de cubrimiento mínimo.

Corolario 2. *Teorema de König es cierto.*

Esto se deduce de la construcción ya hecha. El siguiente resultado es interesante.

Corolario 3. *Teorema del matrimonio de Hall.*
Sea G un grafo bipartito, con bipartición L y R. Entonces $\exists M$ matching que cubre a todo L (es decir, cubre todos los vértices de L)$\iff \forall A \subseteq L$; $|N(A)| \geq |A|$, y al ser bipartito el Matching es máximo.

Ejemplo 4. Una aplicación curiosa es la siguiente:
Sea M una Matriz con coeficientes en $\{0,1\}$. Definimos transversal como un conjunto de 1 t.q. no hay dos en la misma fila o columna. Luego existe una transversal que toca a todas las filas de $M \iff \forall A \subseteq filas(M)$ "la cantidad de 1 de A"$\geq |A|$.

Demostración. (De Hall) Primero, la implicancia hacia la derecha. A cada $v \in A$ asociamos M_v el único vértice emperajado por M en $N(A)$. Luego $|N(A)| \geq |A|$, ya que construimos una inyección de A a $N(A)$.

Ahora la implicancia hacia la izquierda. Sea G un grafo t.q $|N(A)| \geq |A| \forall A \subseteq L$. Sea C un cubrimiento mínimo; luego C toca algunos nodos de la bipartición izquierda y otros de la derecha. Por König

$$|M^*| = |C| = |C \cap L| + |C \cap R| \geq |C \cap L| + |N(L \setminus C)|$$

.

En lo anterior hemos usado además que toda arista de L/C llega $C \cap R$, pues C es cubrimiento. Finalmente, por la hipótesis

$$|C \cap L| + |N(L/C)| \geq |C \cap L| + |L/C| = |L|$$

.

Así concluimos que el matching cubre todo L y tenemos asi lo deseado. □