Ma47a Optimización Combinatorial	8 de Abril, 2006

Pauta Control No. 1

Prof. Cátedra: M. Kiwi — *Prof. Auxiliar: G. Sanchez*

PROBLEMA 1:

(i.1).- Por inducción en el número de nodos. La base de la inducción es obvia. Supongamos entonces que se cumple la afirmación para $|V| = n \geq 1$. Sea entonces $v \in V$ cualquiera, $|V| = n+1$, y sea d el grado de v. Observar que $d \geq 1$ porque G es conexo. Sean $C_1, \ldots, C_k$ las componentes conexas de $G \setminus v$. Observar que $k \leq d$ y que por hipótesis de inducción, para todo $j = 1, \ldots, k$ se tiene que

$$|E(C_j)| \geq |V(C_j)| - 1\,.$$

Luego,

$$|E| = d + \sum_j |E(C_j)| \geq d - k + \sum_j |V(C_j)| \geq \sum_j |V(C_j)| = |V| - 1\,.$$

(i.2).- Si G no posee circuitos, entonces es un bosque, i.e., cada una de sus componentes conexas es un árbol. Como todo árbol tiene un arco menos que el número de sus nodos, el resultado deseado sigue inmediatamente.

(ii).- Para orientar un grafo cuyos nodos tienen todos grado par basta repetir el siguiente procedimiento mientras quede algún arco $uv \in E$ no orientado, inicialmente $u' = u$ y $v' = v$:

> Orientar el arco $u'v'$ de u' a v', elegir un arco no orientado $v'w' \in E$ y repetir reemplazando u' y v' por v' y w' respectivamente. El procedimiento continua hasta que no hayan arcos no orientados incidentes en v'.

No es difícil ver que como todo nodo en el grafo original es de grado par, entonces el procedimiento descrito se detiene necesariamente cuando $v' = u$, donde u es el nodo que queda en la cola del arco que primero se orientó al comienzo del procedimiento.

Trabajando sobre una representación en términos de matriz de adyacencia del grafo de entrada, el procedimiento descrito puede implementarse como se ilustra en el Algoritmo 1. Para determinar la eficiencia del Algoritmo 1 notar que las instrucciones 2–6 requieren $O(|Adj(v)|)$, de donde se concluye que todo el procedimiento requiere $O(|V|) + O(\sum_{v\in V} |Adj(v)|) = O(|V| + |E|)$.

PROBLEMA 2:

(i).- Sea $e^\star = u^\star v^\star$ tal que $\omega(e^\star) = \min_{e\in\delta(S)} \omega(e)$. Como $\omega(\cdot)$ toma valores diferentes sobre cada arco de G, debe tenerse que $e^\star \in \delta(S)$. Para efectos de obtener una contradicción supongamos que T es un árbol generador de peso mínimo que no contiene a $e^\star$. Sea P el $(u^\star, v^\star)$–camino en P (dicho camino es único y siempre existe). Como $u^\star \in S$ o (exclusivo) $v^\star \in S$, entonces debe haber un arco

Algorithm 1 Determinación de una orientación Euleriana

Require: $G = (V, E)$ grafo tal que todo $v \in V$ es de grado par
Ensure: $|\{uv \in E \,:\, \Pi(uv) = v\}| = |\{vw \in E \,:\, \Pi(vw) = w\}|$ para todo $v \in V$
for $v \in V$ **do**
  for $Adj(v) \neq \emptyset$ **do**
    $w \leftarrow \texttt{EXTRACT}(Adj(v))$;
    /$\star$ $\texttt{EXTRACT}(Adj(v))$ extrae de la lista de adyacencia de v uno de sus vecinos. $\star$/
    $\Pi(vw) \leftarrow w$;
    /$\star$ $\Pi(v_0v_1) = v'$ indica que v_0v_1 se orienta de forma que su cabeza es $v' \in \{v_0, v_1\}$. $\star$/
    $v \leftarrow w$;
  end for
end for
return Π;

$e = uv \in P$ tal que $e \in \delta(S)$. Sin pérdida de generalidad supongamos que $u^\star$ y u están en S, mientras que $v^\star$ y v están en $V \setminus S$. Observar que $T^\star = (T \setminus e) \cup \{e^\star\}$ también es árbol generador y que

$$\omega(T^\star) = \omega(T) - \omega(e) + \omega(e^\star) < \omega(T)\,,$$

donde la última igualdad se tiene porque $\omega(e^\star) \leq \omega(e)$ para todo $e \in \delta(S)$, $e^\star \neq e$, y porque $\omega(\cdot)$ toma valores diferentes sobre cada arco de G. Pero $\omega(T^\star) < \omega(T)$ contradice la optimalidad de T.

(ii).- Para efectos de obtener una contradicción, supongamos que existe T árbol generador de peso mínimo de G que contiene a $e^\star$. Sean S y $V \setminus S$ los nodos de las dos componentes conexas de $T \setminus e^\star$. Como C es un circuito que contiene a $e^\star$, sigue que existe un arco en $e \in C \setminus e^\star$ tal que $e \in \delta(S)$. Luego, $\omega(e) \leq \omega(e^*)$, pero como también $\omega(e) \neq \omega(e^*)$, se debe tener que $\omega(e) < \omega(e^\star)$. Sea $T^\star = (T \setminus e^\star) \cup \{e\}$. Notar que $T^\star$ es un árbol generador de G y que

$$\omega(T^\star) = \omega(T) - \omega(e^\star) + \omega(e) < \omega(T),$$

lo que contradice la optimalidad de T.

(iii).- Si existe un (v, u)–camino $\mathcal{P}$ en G tal que $\omega(e^\star) > \omega(e)$ para todo $e \in \mathcal{P}$, entonces existe un circuito $C = e^\star P$ en G tal que

$$\omega(e^\star) = \max_{e \in C} \omega(e)\,.$$

De la parte (ii) se concluye que $e^\star$ no puede pertenecer a ningún árbol generador de peso mínimo de G.

Supongamos ahora que no existe un (v, u)–camino $\mathcal{P}$ en G tal que $\omega(e^\star) > \omega(e)$ para todo $e \in \mathcal{P}$. Sea $G' = (V, E')$ el grafo que se obtiene a partir de G al remover todos los arcos $e \in E$ tales que $\omega(e) \geq \omega(e^*)$. Sea además,

$$S = \{v' \in V \,:\, \text{hay un } (u, v')\text{-camino en } G'\}\ .$$

Por hipótesis, se tiene que $v \notin S$.

Luego, si $\delta(S)$ denota el corte inducido por S en G, sigue que $e^\star \in \delta(S)$. Además, como $e \in \delta(S)$ implica que $\omega(e) \geq \omega(e^*)$, sigue que

$$\omega(e^\star) = \min_{e \in \delta(S)} \omega(e)\,.$$

De la parte (i) se concluye que $e^\star$ debe pertenecer a todo árbol generador de peso mínimo de G.

(iv).- En virtud de la parte anterior, basta remover del grafo todos los arcos e tal que $\omega(e) \geq \omega(e^*)$ y determinar si uno de los extremos de $e^\star$ se puede alcanzar desde el otro (algorítmicamente lo anterior se puede realizar vía una búsqueda en profundidad tomando como raíz uno de los extremos de e y verificando si el otro extremo es descubierto o no en la búsqueda).

Trabajando sobre una representación en términos de matriz de adyacencia del grafo de entrada, el procedimiento descrito en el párrafo anterior puede implementarse como se ilustra en el Algoritmo 2. Para determinar la eficiencia del procedimiento, observar que las instrucciones 1, 2–6, 7, 8–12 toman $O(1)$, $O(|E|)$, $O(|V|+|E|)$, y $O(1)$ respectivamente — es decir, $O(|V|+|E|)$ en total.

Algorithm 2 Determinación de pertenencia de un arco en algún árbol generador de peso mínimo

Require: $G=(V,E)$ grafo, $\omega : E \to \mathbb{Z}$ función inyectiva, y $e^\star = u^\star v^\star \in E$.
Ensure: $e^\star$ pertenece a algún árbol generador de peso mínimo de G
1: $F \leftarrow \emptyset$;
2: **for** $e \in E$ **do**
3:    **if** $\omega(e) < \omega(e^\star)$ **then**
4:       $F \leftarrow F \cup \{e\}$;
5:    **end if**
6: **end for**
7: $\Pi \leftarrow \texttt{BFS}(G,u^\star)$
$/\star$ $\texttt{BFS}(G,u^\star)$ retorna el árbol Π resultante de una búsqueda horizontal en G con raíz u. $\star/$
8: **if** $\Pi(v^\star) \neq \texttt{NULL}$ **then**
9:    **return** `TRUE`;
10: **else**
11:    **return** `FALSE`;
12: **end if**

Problema 3: Ver [CLR90, Chap. 26, § 3]

Referencias

[CLR90] T.H. Cormen, C.E. Leiserson, y R.L. Rivest, “Introduction to Algorithms”, MIT Press, primera edición, 1990.