MA-47A Optimización Combinatorial	**7 de Julio, 2000**

Pauta Examen

Prof. Cátedra: M. Kiwi — *Prof. Auxiliar: J. R. Correa*

Problema 1:

(i).- El lenguaje de decisión asociado al problema del conjunto independiente es

$$INDEP \;=\; \{\langle G,k\rangle \,:\, G \text{ es un grafo no–dirigido, } k\in\mathbb{N},\ \alpha(G)\geq k\}\,.$$

Primero veamos que $INDEP \in NP$. En efecto, sea $\langle G,k\rangle$ donde G es un grafo no–dirigido y $k\in\mathbb{N}$. El certificado $v_1,\ldots,v_k \in V(G)$ y el proceso de verificación que consiste en comprobar que para todo $i,j\in\{1,\ldots,k\}$ se tiene que $(v_i,v_j)\notin E(G)$, permiten concluir en tiempo polinomial si es que $\langle G,k\rangle$ pertenece o no a $INDEP$.

Para ver que $INDEP$ es NP-duro, demostraremos que $CLIQUE \leq^P_m INDEP$.[1] Sea $G=(V,E)$ un grafo no–dirigido y sea $\bar{G}=(V,\bar{E})$ tal que $e\in\bar{E}$ sí y sólo si $e\notin E$. Si G es un grafo no–dirigido, notar que $S\subseteq V(G)$ induce un subgrafo completo en G de tamaño k sí y sólo si S es un conjunto independiente en $\bar{G}$ de tamaño k. Luego,

$$\langle G,k\rangle \in CLIQUE \iff \langle \bar{G},k\rangle \in INDEP\,.$$

Como la reducción que a $\langle G,k\rangle$ le asocia $\langle \bar{G},k\rangle$ es claramente a tiempo polinomial se concluye que $CLIQUE \leq^P_m INDEP$.

(ii).- Sea $\mathcal{A}$ el algoritmo de la Fig. 1.

Entrada:		$G=(V,E)$ grafo no–dirigido de grado máximo d+1.
Código:	1.	$I \leftarrow \emptyset$
	2.	**while** $V\neq\emptyset$ **do**
	3.	Elegir $v\in V$ cualquiera
	4.	$I\leftarrow I\cup\{v\}$
	5.	$V\leftarrow V\setminus(\{v' \,:\, (v,v')\in E\}\cup\{v\})$
	6.	**return** I

Figure 1: Algoritmo $\mathcal{A}$ de aproximación para $\alpha(G)$.

Observar que cada vez que se ejecutan los pasos 4 y 5 de $\mathcal{A}$ el conjunto I crece en 1 y V decrece en a lo más d+1. Sigue que al finalizar el algoritmo $|I|\geq |V|/(d{+}1)$. Como $|V|\geq\alpha(G)$, se tiene que $(d{+}1)|I|\geq\alpha(G)$, i.e., $\mathcal{A}$ es un algoritmo de $(d{+}1)$–aproximación.

(iii).- Sea A un algoritmo de K-aproximación aditiva para el problema del conjunto independiente, i.e., A es una función a valores en $\mathbb{N}$ definida sobre $\{\langle G\rangle \,:\, G \text{ es un grafo no–dirigido}\}$, calculable en

[1] Recordar que $\langle G,k\rangle\in CLIQUE$ sí y sólo si G es un grafo no–dirigido donde existe $S\subseteq V(G)$ tal que $G|_S$ es un subgrafo completo de tamaño k.

tiempo polinomial, tal que para todo G,

$$0 \leq \alpha(G) - A(\langle G\rangle) \quad \leq \quad K\,.$$

Sea G' un grafo que consiste en $\lceil K\rceil + 1$ copias de G. Sigue que $\alpha(G') = (\lceil K\rceil + 1)\alpha(G)$.

Sea B la función cuyo dominio y recorrido son iguales a los de A y tal que

$$B(\langle G\rangle) \quad = \quad \left\lceil \frac{A(\langle G'\rangle)}{\lceil K\rceil + 1} \right\rceil\,.$$

Como K es una constante se tiene que G' puede ser construido a partir de una descripción de G en tiempo polinomial. Luego, como A es calculable en tiempo polinomial y K es una constante, B es calculable en tiempo polinomial, y para todo grafo G se tiene que

$$B(\langle G\rangle) - \alpha(G) \quad \geq \quad \frac{1}{\lceil K\rceil + 1}\,(A(\langle G'\rangle) - \alpha(G')) \;\geq\; -\frac{K}{\lceil K\rceil + 1} \;>\; -1\,.$$

Luego, como $B(\langle G\rangle)$ y $\alpha(G)$ son enteros, sigue que $B(\langle G\rangle) \geq \alpha(G)$. Además, si G' tiene un conjunto independiente de tamaño al menos $A(\langle G'\rangle)$ entonces, $\alpha(G) \geq A(\langle G'\rangle)/(\lceil K\rceil + 1)$. Como $\alpha(G)$ es un entero, se concluye que $\alpha(G) \geq B(\langle G\rangle)$. En resumen $B(\langle G\rangle) = \alpha(G)$ es calculable en tiempo polinomial. Puesto que calcular $\alpha(\cdot)$ es un problema NP-duro (dado que $INDEP$ es NP-completo) lo anterior implica que P = NP,

PROBLEMA 2:

(i.1).- Sea $p = (u_0, \ldots, u_m)$ un camino en G_k. Observar que el peso de p con respecto a $\bar{w}_k$ es

$$\begin{aligned}
\bar{w}_k(p) \quad &= \quad \sum_{i=0}^{m-1} \bar{w}_k(u_i, u_{i+1}) \\
&= \quad \sum_{i=0}^{m-1} w_k(u_i, u_{i+1}) + \delta_{G_{k-1},w_{k-1}}(s, u_0) - \delta_{G_{k-1},w_{k-1}}(s, u_m) \\
&= \quad w_k(p) + \delta_{G_{k-1},w_{k-1}}(s, u_0) - \delta_{G_{k-1},w_{k-1}}(s, u_m)\,.
\end{aligned}$$

Luego, el peso con respecto a $\bar{w}_k$ de todo camino de u_0 a u_m difiere del peso con respecto a w_k en una constante que depende sólo de u_0 y u_m. Sigue que el peso mínimo con respecto a $\bar{w}_k$ de un camino de u_0 a u_m difiere del peso mínimo con respecto a w_k en $\delta_{G_{k-1},w_{k-1}}(s, u_0) - \delta_{G_{k-1},w_{k-1}}(s, u_m)$. En particular, si $u_0 = s$ y $u_m = v$, como $\delta_{G_{k-1},w_{k-1}}(s, s) = 0$, entonces

$$\delta_{G_k,w_k}(s, v) \quad = \quad \delta_{G_k,\bar{w}_k}(s, v) + \delta_{G_{k-1},w_{k-1}}(s, v)\,.$$

(i.2).- Sea $(u, v) \in E(G_k)$. Necesariamente se tiene que $(u, v) \in E(G_{k-1})$ o $(v, u) \in E(G_{k-1})$. Observar además, que si $(u, v) \in E(G_{k-1})$, entonces P_{k-1} no ocupa el arco (u, v). si $(u, v) \notin E(G_{k-1})$, entonces P_{k-1} ocupa el arco (v, u). Luego, si $(u, v) \in E(G_{k-1})$, entonces $w_k(u, v) = w_{k-1}(u, v)$, y si $(u, v) \notin E(G_{k-1})$, entonces $w_k(u, v) = -w_{k-1}(v, u)$.

Consideremos primero el caso en que $(u, v) \in E(G_{k-1})$. Como

$$\delta_{G_{k-1},w_{k-1}}(s, v) \quad \leq \quad \delta_{G_{k-1},w_{k-1}}(s, u) + w_{k-1}(u, v)\,,$$

y además $w_{k-1}(u,v) = w_k(u,v)$, se tiene el resultado deseado, i.e., que $\bar{w}_k(u,v) \geq 0$.

Si consideramos el caso en que $(u,v) \notin E(G_{k-1})$, entonces tenemos que P_{k-1} ocupa el arco (v,u) y que $w_k(u,v) = -w_{k-1}(v,u)$. Por sub–optimalidad del camino de peso mínimo, sigue que el camino de peso mínimo con respecto a w_{k-1} de s a u pasa primero por v y luego ocupa el arco (v,u). Por lo tanto,

$$\begin{aligned} \delta_{G_{k-1},w_{k-1}}(s,u) &= \delta_{G_{k-1},w_{k-1}}(s,v) + w_{k-1}(v,u) \\ &= \delta_{G_{k-1},w_{k-1}}(s,v) - w_k(u,v)\,. \end{aligned}$$

Sigue que $\bar{w}_k(u,v) = 0$.

(i.3).- Se utiliza el mismo algoritmo que se vio en clases (descrito más arriba), pero en vez de utilizar BELLMAN-FORD para determinar el peso mínimo con respecto a w_k de los caminos en G_k, se usa el algoritmo de Dijkstra y se determina el peso mínimo con respecto a $\bar{w}_k$ de los caminos en G_k. Por (i.2) tenemos que el algoritmo de Dijkstra encontrará los caminos de peso mínimo con respecto a $\bar{w}_k$. Por (i.1) tenemos que los caminos que encuentre el algoritmo de Dijkstra serán caminos de largo mínimo en G_k también con respecto a w_k.

La complejidad del nuevo algoritmo basado en Dijkstra es $O((|V(G)| + |E(G)|)\log(|V(G)|))$ por iteración. Luego, como encontrar un matching de tamaño k con el procedimiento visto requiere de k iteraciones, el nuevo algoritmo tendrá una complejidad de tiempo de $O(k(|E(G)| + |V(G)|)\log(|V(G)|))$.

(ii).- Supongamos primero que M es matching perfecto en G. Sigue que cualquiera sea $S \subseteq L$ se tiene que $\mathcal{N}(S) \supseteq \{r \in R : (l,r) \in M, l \in S\}$. Luego, $|\mathcal{N}(S)| \geq |\{r \in R : (l,r) \in M, l \in S\}|$. Pero, M es un matching perfecto, luego para todo $l \in S$ existe un $r_l \in R$ tal que $(l,r_l) \in M$ y si $l \neq l'$, entonces $r_l \neq r_{l'}$. Sigue que $|\mathcal{N}(S)| \geq |S|$.

El converso lo obtendremos como un corolario del Teorema de flujo–máximo corte–mínimo. Para ello le asociamos a $G = (L,R,E)$ grafo no–dirigido bipartito la red $G' = (V',E')$ con capacidades máximas en los arcos dadas por $u{:}E' \to \mathbb{R}$ (asumimos que los arcos sólo acarrean flujo no–negativo) donde

$$\begin{aligned} V' &= L \cup R \cup \{s,t\}\,, \\ E' &= \bigcup_{l\in L}\{(s,l)\} \cup \bigcup_{r\in R}\{(r,t)\} \cup \{(l,r) \in L\times R : (l,r) \in E\}\,, \end{aligned}$$

y $u(e) = 1$ cualquiera sea $e \in E'$.

Como u es integral existe un (s,t)–flujo máximo $x = (x_e)_{e\in E'}$ integral en G'. Afirmamos que dicho flujo debe tener valor $|L| = |R|$. En efecto, $\delta(\{s\})$ es un (s,t)–corte de tamaño $|L| = |R|$. Además, si $\delta(S)$ es un (s,t)–corte cualquiera se tiene que $s \in S$, $t \notin S$, y por lo tanto

$$\delta(S) = \bigcup_{l\in L\setminus S}\{(s,l)\} \cup \bigcup_{r\in S\cap R}\{(r,t)\} \cup (\mathcal{N}(S\cap L) \cap (R\setminus S))\,.$$

Luego,

$$|\delta(S)| = |L\setminus S| + |S\cap R| + |\mathcal{N}(S\cap L)| - |\mathcal{N}(S\cap L) \cap (S\cap R)| \geq |L\setminus S| + |\mathcal{N}(S\cap L)|$$

Pero $|\mathcal{N}(S \cap L)| \geq |S \cap L|$ por hipótesis, luego $|\delta(S)| \geq |L \setminus S| + |S \cap L| = |L| = |R|$. Por lo tanto, todo (s,t)–corte en G' tiene un valor $|L| = |R|$. Sigue, por el Teorema de flujo–máximo corte–mínimo, que x tiene valor $|L| = |R|$.

Sea entonces $M = \{(l,r) \in E : l \in L, r \in R, x_{l,r} = 1\}$. Como x es integral y tiene valor $|L| = |R|$ se tiene que $|M| = |L| = |R|$. Luego, si M es un matching, necesariamente será un matching perfecto. Veamos que M es matching, i.e., que en cada nodo de G incide a lo más un arco de M. En efecto, supongamos que en un nodo $l \in L$ inciden dos arcos de M. Esto significa, que de l salen al menos 2 unidades de flujo. Por conservación de flujo, deben llegar al menos 2 unidades de flujo a l. Como dicho flujo sólo puede venir de s y el arco (s,l) tiene capacidad máxima 1 obtenemos una contradicción. Supongamos ahora que en un nodo $r \in R$ inciden dos arcos de M. Luego, a r deben llegar al menos 2 unidades de flujo. Por conservación de flujo, deben salir al menos 2 unidades de flujo de r. Dicho flujo sólo puede ir a parar a t pero el arco (r,t) tiene capacidad máxima 1 y nuevamente obtenemos una contradicción.