MA4701-1 - Optimización Combinatorial
Profesor: José Soto
Auxiliares: Nicolás Sanhueza - Christian von Borries

Auxiliar N°11

7 de noviembre de 2013

P1. Considere una red (G, u, s, t) donde las capacidades son enteras. Edmonds-Karp, en cada iteración, busca encontrar un s-t-flujo máximo, lo que se puede hacer usando la estrategia del *camino más gordo*. Generalmente resolvemos este subproblema usando Dijkstra, probaremos ahora una nueva estrategia. La idea es hacer «búsqueda binaria». Dado un grafo residual (G^f, u^f, s, t), el flujo adicional máximo que podremos agregar toma un valor entre 0 y $u^* = \max_e u_e$. Considere el siguiente algoritmo.

Algoritmo 1: $\textsc{CaminoMáximo}(G^f, u^f, s, t)$

$k \leftarrow 0$;
$a_0 \leftarrow 0 \qquad b_0 \leftarrow u^* + 1 \qquad m_0 = \lfloor (a_0 + b_0)/2 \rfloor$;
while $b_k - a_k > 1$ **do**
  Busca un (s, t)-camino P_k en G^f que sólo use aristas de capacidad mayor igual que m_k;
  if NO EXISTE **then**
    $a_{k+1} \leftarrow a_k \qquad b_{k+1} \leftarrow m_k \qquad m_{k+1} = \lfloor (a_{k+1} + b_{k+1})/2 \rfloor$;
  else
    $a_{k+1} \leftarrow m_k \qquad b_{k+1} \leftarrow a_k \qquad m_{k+1} = \lfloor (a_{k+1} + b_{k+1})/2 \rfloor$;
    $P \leftarrow P_k$;
  $k \leftarrow k + 1$;
if $a_k > 0$ **then**
  return (P)

Muestre que este algoritmo es correcto y analice su complejidad. Modificando Edmonds-Karp para que use este algoritmo como subrutina en lugar de Dijsktra, ¿cómo se comparan ambas complejidades? ¿Qué ventajas presenta este algoritmo?

P2. Sea (G, u, s, t) una red donde todos los arcos tienen capacidad cero o uno. Encuentre un algoritmo que encuentre un s-t-flujo máximo en orden $O(|V||E|)$. Pruebe su correctitud.

P3. Considere dos conjuntos de alumnos beauchefianos, *computines* y *matemáticos*, representados por C y D, respectivamente. Cada una de estas carreras es muy unida, así que puede suponer que todos los alumnos computines son amigos entre sí y todos los alumnos matemáticos son amigos entre sí. Naturalmente, hay además algunos pares de amigos mutuos formados entre computines y matemáticos. Definimos el conjunto total de *alumnos* a estudiar como $U := C \cup D$. Cuando se matricularon por primera vez, a cada alumno se le tomó un test de diagnóstico, así que cada alumno $u \in U$ tiene asociado un *índice de coeficiente intelectual* dado por una función $q : U \to \mathbf{R}_0^+$.

El decano quiere seleccionar un *equipo* $S \subseteq U$ para que represente a la Facultad en el Mathemathical Olympiad in Combinatorial Optimization (MOCO). Como se necesita un equipo comprometido y con compareñismo, el decano exije que este equipo S cumpla que todo par de personas en S sean amigos entre sí. Llamamos $\mathcal{S}$ a los equipos de alumnos que cumplen tal condición. Pero queremos un equipo inteligente así que además, entre todos los $S \in \mathcal{S}$, queremos escoger el que maximice la suma de los coeficientes intelectuales de cada uno. En otras palabras queremos encontrar $S \in \mathcal{S}$ tal que $q(S) = \sum_{s \in S} q(s)$ sea máximo.

El objetivo de esta pregunta es encontrar un algoritmo eficiente que resuelva el problema planteado.

(a) Dado un grafo $G = (V, E)$ conjunto $S \subseteq V$ se dice *independiente* si ningún par de elementos en S están conectados entre sí. Muestre el siguiente lema: S es independiente si y sólo si $V \setminus S$ es cubrimiento de vértices.

(b) Usando lo anterior, muestre que el problema del decano es equivalente a uno de encontrar un cubrimiento de vértices de peso mínimo en un grafo bipartito.

(c) Modele este nuevo problema como un problema de corte mínimo sobre una red.

(d) Usando todo lo anterior, encuentre un algoritmo polinomial en $|U|$ que resuelva el problema. Muestre su correctitud y analice su complejidad.