FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS UNIVERSIDAD DE CHILE

IN770: Modelos y Algoritmos de Optimización
Profesores : Cristián Cortés, Daniel Espinoza
Auxiliares : José Muñoz, Rodrigo López, Diego Morán

Ejercicios resueltos 1

Martes 05 de Abril de 2009

P1.- Sea $\mathcal{G} = (\mathcal{A}, \mathcal{N})$ un grafo dirigido. Suponga que los costos de los arcos satisfacen $a_{ij} \geq 0,\ \forall (i,j) \in \mathcal{A}$.

Considere el problema de rutas mínimas *uno-a-todos* resuelto con algún método de *Label Setting*, pero en el que sólo se está interesado en conocer la ruta más corta del nodo 1 a un subconjunto $\mathcal{T}$ de nodos. Sea

$$\overline{a} = \text{mín}\{a_{it} \ : \ (i,t) \in \mathcal{A}, \quad t \in \mathcal{T}\}$$

Si $\overline{a} > 0$, demuestre que se puede detener el método cuando uno de los nodos de mínima etiqueta en la lista V, que denotaremos J, tiene una etiqueta d_J que cumple:

$$d_J + \overline{a} > \text{máx}\{d_s \ : \ s \in \mathcal{T}\} \quad (*)$$

Sol: Es bueno tener en cuenta de que estamos aplicando el *Algoritmo de Dijkstra* con **costos positivos**, es decir, el caso interesante, pues se cumplen las buenas propiedades :D.

Basados en la observación anterior, de $(*)$ podemos concluir que, dado $t \in \mathcal{T}$, entonces

$$d_t \leq \text{máx}\{d_s \ : \ s \in \mathcal{T}\} < d_J + \overline{a} < \infty$$

es decir, todas las etiquetas de elementos en $\mathcal{T}$ se han modificado (pues son finitas).

Luego, para un nodo $t \in \mathcal{T}$ se tienen sólo 2 posibles casos: $t \in V$ ó t salió de V (y por lo tanto fue nodo de etiqueta mínima en algún momento, por lo que no vuelve a entrar a V).

Para probar que podemos detener el algoritmo, basta probar que, en ninguno de los 2 casos anteriores, las etiquetas de nodos en $\mathcal{T}$ serán modificadas. Veamos:

- **Si** t **salió** V: por propiedad de *Dijkstra* con costos positivos, d_t no se vuelve a modificar (ya que en alguna etapa anterior t fue el nodo de etiqueta mínima que salió de V).
- **Si** $t \in V$: La única forma de que la etiqueta de un nodo cualquiera $i \in V$ se modifique, en la iteración en que el nodo que sale de V es J, es que se cumpla:

$$d_i > d_J + a_{Ji}$$

Veamos que esto no se cumple para $t \in \mathcal{T}$. Si se tuviera que:

$$d_t > d_J + a_{Jt} \quad (**)$$

podemos usar $(**)$, que $d_t \leq \text{máx}\{d_s \ : \ s \in \mathcal{T}\}$ y que $\overline{a} \leq a_{Jt}$ para concluir que:

$$\text{máx}\{d_s \ : \ s \in \mathcal{T}\} > d_J + \overline{a}$$

lo que contradice $(*)$. Por lo tanto ningún nodo en $\mathcal{T}$ puede cumplir $(**)$.

Con esto, si $t \in \mathcal{T}$, entonces d_t no se modifica en esta iteración.

¿Se modifican las etiquetas de los nodos en $\mathcal{T}$ en las iteraciones siguientes?

La respuesta es no, pues si $I \in V$ es el nodo que sale de V en la siguiente iteración, este nodo cumple:

$$d_J \leq d_I$$

pues J tenía etiqueta mínima en la iteración anterior, cuando salió de V, y por propiedad de *Dijkstra* con costos positivos, las "etiquetas mínimas" son de valor creciente a medida que se avanza en las iteraciones.

Con esto, el nodo I también cumple la propiedad $(*)$. Luego, los nodos en $\mathcal{T}$ tampoco cambian el valor de sus etiquetas en esta iteración.

Conclusión: Al partir de la iteración en que se cumple $(*)$ el valor las etiquetas d_t, con $t \in \mathcal{T}$ no se modifican en ninguna de las iteraciones siguientes, y como estas etiquetas representan el valor del camino más corto desde el nodo 1 al nodo t (pues ya no serán modificadas), podemos dar por terminado el objetivo de encontrar todos los caminos más cortos desde 1 al subconjunto $\mathcal{T}$ de los nodos.

`P2.-` Suponga que todos los nodos de un grafo G tiene grado estrictamente mayor que uno. Pruebe que G contiene al menos un ciclo.

OBS: Notemos que esta fue la propiedad que usamos en la clase auxiliar 1 2007 para garantizar la existencia de un ciclo en uno de los pasos del algoritmo de la parte a) del problema.

`Sol:` A continuación describiremos un algoritmo diseñado para encontrar un ciclo en el grafo G, lo que nos permitirá probar la propiedad:

`ALGORITMO`

Inicialización: $v_0 \in \mathcal{N}$ cualquiera, $V = \{v_0\}$ conjunto de nodos visitados, $P = v_0$ camino auxiliar, $C = \emptyset$ para guardar el ciclo, $u = v_0$ variable auxiliar.

(1) Elegir un nodo w tal que (u, w) o $(w, u) \in \mathcal{A}$.

(2) Si $w \notin V$

- actualizar $P = Pw$, $u = w$, $V = V \cup \{w\}$. Volver a (1).

En caso contrario

- hacer $C = wv_{k+l} \dots v_{k+l}w$ (estamos suponiendo que P tiene la forma $P = v_0v_1 \dots v_{k-1}wv_{k+l} \dots v_{k+l}$).
  Detener el algoritmo.

Salida: entrega el ciclo C y/o avisa de que efectivamente G posee un ciclo.

Lo que hace el algoritmo es, partiendo de un nodo arbitrario, recorre el grafo, construyendo un camino, hasta dar con nodo que ya haya aparecido antes, cuando esto ocurre podemos identificar un ciclo dentro del camino construído, más aún identificamos un ciclo en G.

Hasta el momento nos hemos convencido de que la idea del algoritmo es buena, pero debemos argumentar cada paso del algoritmo. El paso (1) es siempre posible de hacer, es decir sin importar el nodo u siempre podemos encontrar el w que se pide, pues por hipótesis todos los nodos de G tienen grado estrictamente mayor que 1, por lo tanto cada nodo tiene arcos asociados. En el paso (2) no hay nada que argumentar, dado que son simplemente actualizaciones de las variables.

P3.- *(a)* Muestre que todo árbol (ie grafo *conexo* y *acíclico*) con al menos 2 nodos posee al menos 2 nodos con grado 1.

(b) Pruebe que un grafo es un árbol sí y sólo sí es conexo y el número de arcos es uno menos que el número de nodos.

Sol: *(a)* Hay varias formas de hacer este problema, en particular se puede proponer una idea similar a la respuesta de la pregunta 1. Nosotros la haremos truculentamente, con una idea sugerida por un compañero del DIM que había tomado el curso MA606 TEORIA DE GRAFOS.

Usaremos la noción de *Ruta Maximal* en un grafo. Una *Ruta Maximal* es una ruta en el grafo que **no** está incluida en ninguna otra ruta del grafo, o en otras palabras, es una ruta que no se puede extender. En un grafo pueden haber muchas rutas maximales. OBS: una ruta maximal **no** necesariamante es la ruta más larga del grafo, son nociones distintas. Uno siempre puede decir que una ruta de largo máximo es una ruta maximal, pero no al revés (¿por qué?).

Volvamos al problema. Lo primero que podemos decir es que efectivamente el árbol que estamos considerando tiene al menos una ruta maximal ya que, por ser conexo, el grafo posee rutas (podría no tenerlas) y además, como los grafos poseen una cantidad finita de nodos y arcos, no puede ser que encontremos una ruta que se pueda extender "infinitamente". Podemos decir entonces que tenemos P_M una ruta maximal con

$$P_M = v_1 v_2 v_3 \dots v_p$$

Puesto que el árbol posee al menos 2 nodos y es un grafo acíclico podemos concluir que $v_1 \neq v_p$. Como los nodos v_1, v_p están en la ruta P_M entonces $grado(v_1),\ grado(v_p) \geq 1$. Por otro lado, si $grado(v_1)$ *o* $grado(v_p) > 1$ entonces la ruta P_M se podría extender, lo que es una contradicción con el hecho de que P_M es maximal. Así, podemos concluir que $grado(v_1),\ grado(v_p) = 1$ y como $v_1 \neq v_p$ el árbol posee al menos 2 nodos con grado 1.

(b) PROPUESTO. Aunque daremos algunas indicaciones:

- Primero, para facilitarle el asunto, hágalo como una *doble implicancia* ($\Rightarrow$ y $\Leftarrow$).
- Dem. que *conexo* $\Rightarrow$ $|\mathcal{A}| \geq |\mathcal{N}| - 1$. Para esto, desde un nodo fijo v_0 encuentre los caminos a los demás $|\mathcal{N}| - 1$ nodos y a cada camino asociele un arco distinto. Concluya.
- Dem. que *acíclico* $\Rightarrow$ $|\mathcal{A}| \leq |\mathcal{N}| - 1$. Hágalo por inducción en $n = |\mathcal{N}|$. Para el paso inductivo, elija un arco arbitrario uv y con esto separe el grafo completo en 2 subgrafos. Cuente los arcos basado en lo anterior y en la hipótesis inductiva.

P4.- Considere la siguiente variación del problema de flujo de costo mínimo:

$$min \quad \sum_{(i,j)\in\mathcal{A}} a_{ij}x_{ij}$$

$s.a.$

$$\forall i \in \mathcal{N} \qquad \underline{s}_i \leq \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij} - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji} \leq \overline{s}_i$$

$$\forall (i,j) \in \mathcal{A} \qquad b_{ij} \leq x_{ij} \leq c_{ij}$$

Muestre que el problema puede ser convertido a la forma standard del problema de flujo de costo mínimo mediante la adición de un nodo extra K y arcos (K,i) desde este nodo a todos los nodos $i \in \mathcal{N}$, con cotas para el flujo dadas por 0 y $\overline{s}_i - \underline{s}_i$.

Sol: Pensemos que la restricción

$$\forall i \in \mathcal{N} \qquad \underline{s}_i \leq \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij} - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji} \leq \overline{s}_i$$

la podemos escribir de la siguiente manera

$$\forall i \in \mathcal{N} \qquad \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij} - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji} = \delta_i$$

con $\delta_i \in [\underline{s}_i, \overline{s}_i]$

Es decir, hemos cambiado las restricciones con desigualdades a una sólo de igualdad, pero pagando el precio de crear una variable adicional δ_i, con sus respectivas restricciones.

Haciendo un cambio de variable $\varepsilon_i = \delta_i - \underline{s}_i$, con $\varepsilon_i \in [0, \overline{s}_i - \underline{s}_i]$ y restando, nos queda

$$\forall i \in \mathcal{N} \qquad \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij} - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji} - \varepsilon_i = \underline{s}_i$$

Mirando el signo de ε_i (negativo) vemos que nos gustaría incluirlo en una de las sumatorias (la con signo negativo), pero para incluirlo debemos asociarle un nodo, pues la suma contiene sólo términos de la forma x_{ji} con $j : (j,i) \in \mathcal{A}$.

Así, debemos agregar al grafo un nodo ficticio K y arcos (K,i), $i \in \mathcal{N}$ y renombrar la variable $\varepsilon_i = x_{Ki}$. Entonces podemos crear un nuevo grafo G' definiendo $\mathcal{A}' = \mathcal{A} \cup \{(K,i)\}$, $\mathcal{N}' = \mathcal{N} \cup K$ con esto la restricción se puede escribir "legalmente" como:

$$\forall i \in \mathcal{N} \qquad \sum_{\{j:(i,j)\in\mathcal{A}'\}} x_{ij} - \sum_{\{j:(j,i)\in\mathcal{A}'\}} x_{ji} = \underline{s}_i$$

Recordando la restricción sobre ε_i obtenemos:

$$0 \leq x_{Ki} \leq \overline{s}_i - \underline{s}_i$$

Por otro lado

$$\sum_{i \in \mathcal{N}} \sum_{\{j:(i,j) \in \mathcal{A}\}} x_{ij} - \sum_{\{j:(j,i) \in \mathcal{A}\}} x_{ji} = 0 \ (porqué?)$$

luego,

$$\sum_{i \in \mathcal{N}} x_{Ki} = -\sum_{i \in \mathcal{N}} \underline{s}_i$$

Y en el nuevo grafo $G' = (\mathcal{A}', \mathcal{N}')$ se puede escribir el problema de flujo a costo mínimo:

$$min \quad \sum_{(i,j) \in \mathcal{A}} a_{ij} x_{ij}$$

$s.a.$

$$\forall i \in \mathcal{N} \qquad \sum_{\{j:(i,j) \in \mathcal{A}'\}} x_{ij} - \sum_{\{j:(j,i) \in \mathcal{A}'\}} x_{ji} = \underline{s}_i$$

$$\sum_{i \in \mathcal{N}} x_{Ki} = -\sum_{i \in \mathcal{N}} \underline{s}_i$$

$$\forall (i,j) \in \mathcal{A} \qquad b_{ij} \leq x_{ij} \leq c_{ij}$$

$$\forall i \in \mathcal{N} \qquad 0 \leq x_{Ki} \leq \overline{s}_i - \underline{s}_i$$

Luego, hemos logrado, paso por paso, formular el problema anterior como uno de flujo a costo mínimo standard. Una de las lecciones que uno puede aprender con esta solución es que agregar nodos ficticios nos permite crear variables que afecten convenientemente las ecuaciones de divergencia. Esto nos puede ser de utilidad para futuros problemas de modelamiento.

Para terminar el problema sólo nos falta demostrar que ambas formulaciones son *equivalentes*, en el sentido, de que desde la **solución óptima** de una de las formulaciones podemos recuperar la **solución óptima** de la otra. Esta parte del problema, no es difícil, pero queda PROPUESTA, a pesar que no es menos importante que lo ya resuelto.