

Control N°3 Miercoles 10 de junio de 2009

P1)

1-Sea un problema de programación entera y su correspondiente relajación lineal. Cuál de los dos tiene región factible mayor (o igual)? Cuál de los dos es más difícil de resolver (en términos de complejidad computacional)? Justifique.

2- Explique todos los criterios por los cuales una rama no debe ser ramificada en el algoritmo de Branch and Bound.

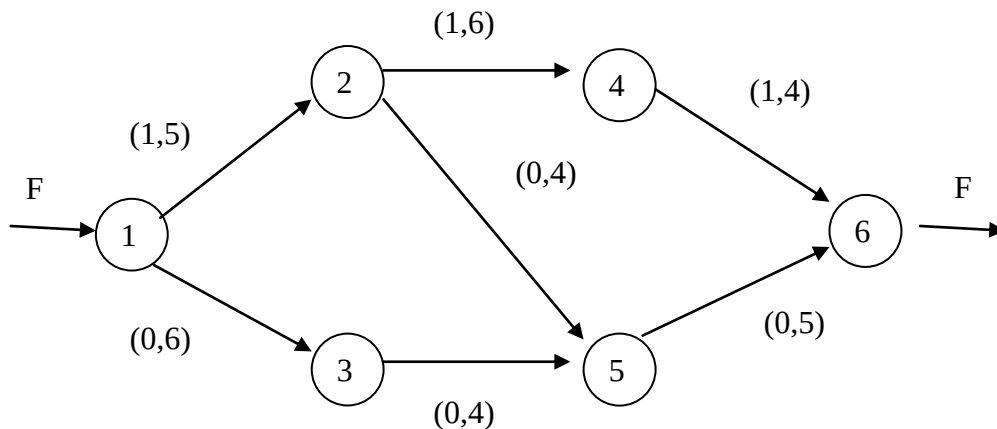
3- Suponga la siguiente red donde (i,j) significa la cota mínima y la capacidad máxima de flujo que soporta cada arco.

$(3,5)$ $(4,6)$
 $1 \dots > 2 \dots > 3$

Demuestre que dicha red admite un flujo inicial factible. Construya para eso la red auxiliar vista en clase de cátedra, aplique el algoritmo de Ford y Fulkerson sobre dicha red y explique qué resultado teórico está usando para poder afirmar la existencia de dicho flujo factible.

P2)

Determine el flujo máximo para el siguiente grafo aplicando el algoritmo de Ford Fulkerson partiendo con un flujo inicial factible.



Cada arco tiene asociado las cotas inferiores (l) y superiores (u) de su flujo factible indicado como (l, u) . Por ejemplo el arco $(1,2)$ tiene $(1,5)$ como cota inferior y superior para su flujo.

1- Indique el flujo máximo y los flujos en cada uno de los arcos (detalle todas las iteraciones del algoritmo de Ford y Fulkerson). (4 puntos)

2- Encuentre un corte mínimo y exprese qué propiedad se cumple en dicho corte. (2 puntos)

Observación: Comience a iterar con flujo factible (global) igual a 3 ($F=3$).

P3)

Usted está tratando de ganarse una “luquitas extras” y es por ello que esta con 2 trabajos. El primero es de repartidor y el segundo es de garzón. Sobre el primero, este tiene las siguientes características: Cada mañana usted llega a la bodega central a buscar los paquetes que debe repartir a lo largo de S sitios. Al final de la jornada usted debe volver a la bodega central. Para poder llevar a cabo de buena forma todas sus actividades, dispone de H unidades de tiempo para hacer este trabajo. El tiempo que demora en ir de un sitio a otro o desde la bodega a un sitio o de un sitio a la bodega es t_{ij} (considere la bodega como el sitio 0). Es posible que usted no alcance a llegar a todos los sitios dentro de las H horas, en tal caso usted posee 2 alternativas. La primera es no ir a ese (esos) lugar(es) lo que le significa una disminución en su sueldo, dicha disminución depende del sitio que no visito, si no fue al sitio s ($s \in S$) la merma de sueldo equivale a P_s . O usted puede visitarlos, pero cada unidad de tiempo que sigue trabajando como repartidor le significa una disminución de P unidades de su sueldo como garzón. Además usted sabe que si llega muy tarde a la “pega” de garzón lo pueden despedir y como usted no quiere que esto ocurra, como máximo seguirá trabajando como repartidos HH unidades de tiempo por sobre las H establecidas.

Su misión ahora, es realizar un PPL que le permita decidir su recorrido a través de los sitios, respetando las restricciones antes planteadas. No está demás decir, que usted desea que sus ingresos se vean penalizados de la menor forma posible. Recuerde que usted no va a visitar necesariamente todos los sitios.

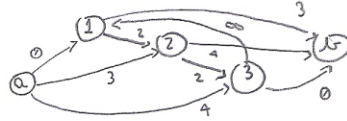
DURACIÓN: 2 horas y 30 minutos

Pauta Control 3

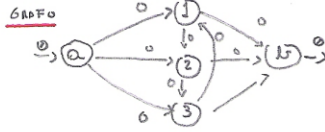
P1

- 1- La región factible del problema relajado es mayor o igual a la del problema entero, esto ocurre debido a que al relajar el problema se permite un mayor número de soluciones ya que estamos siendo menos restrictivos con las restricciones (pasamos de exigir solución entera a solución real). Por su parte el problema entero es más difícil de resolver. Los problemas del tipo entero son NP completos, en cambio el problema de programación lineal es un problema polinomial. (otra forma en que se podría argumentar es que cuando hacemos B&B para resolver un problema entero resolvemos al menos un problema lineal, y de hecho probablemente más de uno).
- 2- Son 3. Que llegamos a una solución entera. Que llegamos a una solución infactible. Que el valor de la función objetivo es menor que el del incumbente (caso maximización , es al revés en minimización)
- 3- La respuesta a continuación:

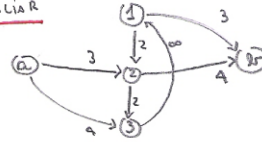
EL NUEVO GRUPO QUEBIA ASI:



LUEGO



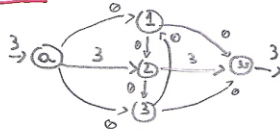
GRUPO AUXILIAR



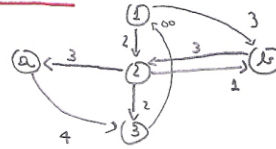
TOMAMOS CAMINO

1-2-4 (PODEMOS SER OTRO)
CON AUMENTO DE 3

GRUPO



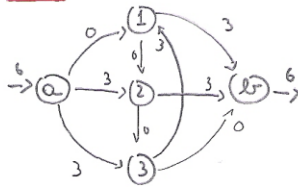
GRUPO AUXILIAR



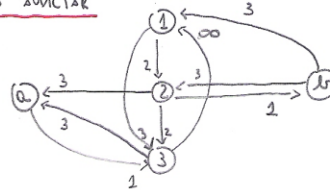
TOMAMOS

1-3-4 (PODEMOS SER OTRO)
CON AUMENTO DE 3

GRUPO



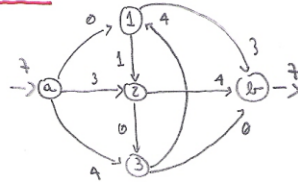
GRUPO AUXILIAR



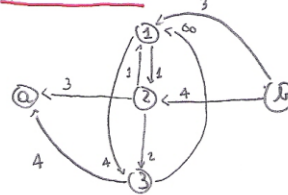
TOMAMOS

1-3-1-2-4
CON AUMENTO DE 1

GRUPO



GRUPO AUXILIAR



COMO NO HAY CAMINO
TERMINAMOS.

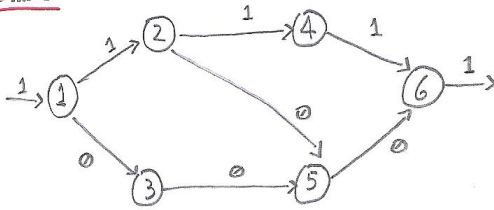
ESTA RED ADMITE UN FLUJO INICIAL FACTIBLE YA QUE SE CUMPLE:

$$F^* = \sum_{(i,j) \in A} l_{ij} \quad \text{ya que } 7 = 3+4.$$

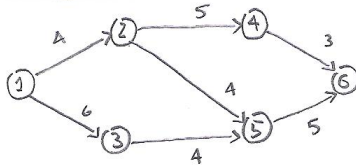
P2

1

GRAFO

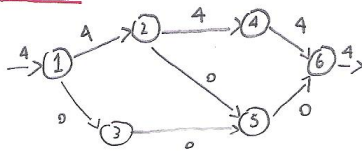


GRAFO AUXILIAR

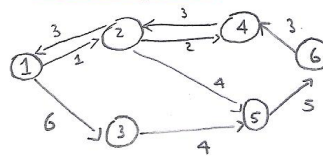


TOMAMOS CAMINO 1-2-4-6 (SE PODÍA TOMAR OTRO)
CON UN AUMENTO DE 3

GRAFO

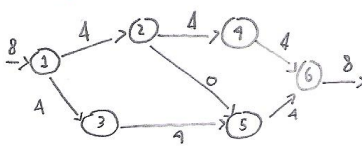


GRAFO AUXILIAR

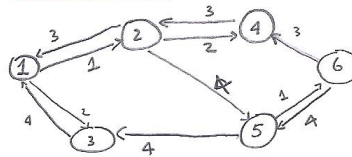


TOMAMOS CAMINO
1-3-5-6 (PODÍA SER
OTRO) Y AUMENTAMOS
4

GRAFO

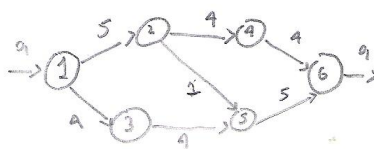


GRAFO AUXILIAR

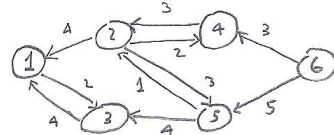


TOMAMOS CAMINO
1-2-5-6 Y
AUMENTAMOS EN 1

GRAFO



GRAFO AUXILIAR



NO HAY CAMINO
POSIBLE. POR LO
TANTO ESTAMOS LISTOS!

LUEGO EL FLUJO MÁXIMO ES 9. LOS FLUJOS ENTRE LOS ARCOS SON:

1-2:5, 2-4:4, 4-6:4, 1-3:4, 3-5:4, 5-6:5, 2-5:1

2

UN CORTE MÍNIMO SERÍA UNO EN EL QUE SEPARAMOS LOS NODOS EN $S = \{1, 3\}$ y $\bar{S} = \{2, 4, 5, 6\}$
ACÁ LA CAPACIDAD DEL CORTE ES 9. Y COMO SABEMOS, EL FLUJO MÁXIMO ES IGUAL
A LA CAPACIDAD DEL CORTE DE CAPACIDAD MÍNIMA.

P3

Variables de decisión

x_{ij} : 1 si va de sitio i a sitio j . 0 en caso contrario

T : unidades de tiempo extra que trabaja como repartidor

Restricciones

Naturaleza de las variables

$x_{ij} \in \{0,1\}$, $T \in$ reales positivos

Salida de la bodega

$$\sum_j x_{0j} = 1$$

Regresa a la bodega

$$\sum_j x_{j0} = 1$$

Tiempo de trabajo

$$\sum_{ij} x_{ij} t_{ij} \leq H + T$$

Si se entra a un lugar se sale de él. Si no se entra no se debe poder salir porque nunca se entro.

$$\sum_i x_{ij} = \sum_i x_{ji} \quad \forall j$$

Entro a un lugar máximo una vez

$$\sum_i x_{ij} \leq 1 \quad \forall j$$

Salgo a lo más una vez de un lugar

$$\sum_j x_{ij} \leq 1 \quad \forall i$$

(Esta última restricción no es necesaria porque se tiene con las otras 2 de más arriba. También se podría omitir la anterior si se escribe esta y la de si se entra a un lugar... En todo caso, no hay problema con ponerla)

No se puede entrar al mismo lugar que donde uno esta

$$x_{ii} = 0 \forall i$$

(Esta restricción se podría omitir trabajando los sumandos de las sumatorias)

Límite de tiempo extra que se sigue trabajando como repartidor

$$T \leq HH$$

Evitar Sub-tours

$$\sum_{ij \in U} x_{ij} \leq |U| - 1 \quad \forall U \text{ tal que } 2 \leq |U|$$

$\leq S - 2$. Con U subconjunto de sitios que no incluye la bodega

$$\sum_{ij \in U}^0 x_{ij} \leq |U| - 1 \quad \forall U \text{ tal que } 2 \leq |U|$$

$\leq S + 1 - 2$. Con U subconjunto de sitios más la bodega

Función Objetivo

$$\min \left[\sum_i P_i \cdot (1 - \sum_j x_{ij}) \right] + PT$$

Puntaje:

Variables de decisión: 0,8decimas

Restricciones: 4,5 decimas (todas valen lo mismo)

Función objetivo: 0,7 decimas