Asignación de vuelos a pilotos

Algoritmia - FIB - UPC

Q1 OTOÑO

Henry Qiu, Jaime Arroyo, Josué Alcántara

January 1, 2018

Contents

0.1 El problema

Se nos plantea el problema de asignación óptima de pilotos a vuelos de en una compañía aérea en un día. En otras palabras, tenemos que encontrar el número de pilotos mínimo para servir todos los viajes de la compañía.

Para cada viaje tenemos su destino, origen, tiempo de inicio y tiempo de llegada. Por facilidad, los tiempos serán expresados en minutos. Como nos restringimos a un día, los tiempos serán enteros en el rango $0..(24 * 60)$.

Cada viaje será representado por 4 enteros de la siguiente forma: $(o_i, d_i, h_{1_i}, h_{2_i})$ donde o_i es el origen, d_i es el destino, h_{1_i} es la hora de inicio y h_{2_i} es la hora de llegada.

0.2 Algunas consideraciones

- Cada vuelo tiene exactamente un piloto asignado, que es el que lo pilota.
- Un piloto solo puede pilotar aquellos viajes que le han sido asignados.
- Por simplicidad, solo tendremos en cuenta que los viajes asignados a un piloto no se pueden solapar en el tiempo.
- Dos viajes son alcanzables si des del destino del primero puedo llegar al origen del segundo cumpliendo unas condiciones previas. En este proyecto tendremos dos versiones que nos definen si dos vuelos son alcanzables o no. Propondremos implementaciones para ambas versiones.
- Decimos que k pilotos son suficientes para realizar todos los vuelos si se puede asignar un piloto a cada vuelo de manera factible, cumpliendo las consideraciones anteriores.

0.3 Formalización

Formalicemos un poco mas el problema. Tenemos n trayectos $T = T1, T2, ..., Tn$. Cada trayecto es una tupla $Ti = (o_i, d_i, h_{1_i}, h_{2_i})$ donde o_i es el origen, d_i es el destino, h_{1_i} es la hora de inicio y h_{2_i} es la hora de llegada.

Por ejemplo, con 3 ciudades (Barcelona (BCN), París (CDG), Londres (LGW)) podríamos tener el siguiente conjunto de 6 trayectos,
$T1 = (BCN, CDG, 0, 100), T2 = (CDG, BCN, 450, 550), T3 = (CDG, LGW, 150, 250),$
$T4 = (LGW, CDG, 600, 700), T5 = (LGX, CDG, 300, 400), T6 = (CDG, BCN, 750, 850).$

La entrada del ejemplo anterior sería la siguiente:

0 1 0 100
1 0 450 550
1 2 150 250
2 1 600 700
2 1 300 400
1 0 750 850

0.4 Solución

Se nos propone solucionar el problema siguiendo el diseño planteado en el libro *Algorithm Design - Kleinberg and Tardos*, explicado a continuación.

Un algoritmo eficiente para resolver el problema de asignación óptima de pilotos a vuelos está basada en redes de flujo. Se trata de diseñar la siguiente red:

Conjunto de nodos

- Para cada vuelo i (u_i, v_i) añadimos dos nodos enlazados con una arista de capacidad máxima y mínima de 1. De esta forma obligamos a que el vuelo i sea atendido.
- Tendremos un nodo fuente s y el sumidero t.

Conjunto de aristas

- Para cada vuelo i, hay una arista (u_i, v_i) de capacidad máxima y mínima 1. (Obligamos a que el vuelo i sea asignado).
- Para cada vuelo i y j, si j es alcanzable desde i entonces añadimos una arista (v_i, u_j) de capacidad máxima 1 y mínima 0. (El mismo piloto puede realizar los viajes i y j).
- Para cada vuelo i hay una arista (s, u_i) de capacidad máxima 1 y mínima 0.
- Para cada vuelo i hay una arista (v_i, t) de capacidad máxima 1 y mínima 0.
- Tenemos una arista (s, t) de capacidad máxima k y mínima 0. (Si tenemos pilotos extra no los necesitamos).

Finalmente, el nodo s tendrá una demanda de -k, mientras que el nodo t la tendrá de k. El resto de nodos tendrá una demanda de 0.a

Se puede notar que tendremos un grafo resultante con demandas y capacidades mínimas. Por tanto, hemos reducido nuestro problema a un problema de flujo bajo circulación de demandas y capacidades mínimas. En las siguiente sección vamos a ver como podemos reducir este problema a un problema de flujo, que prodemos usar como entrada a *MaxFlow*.

0.5 Circulación bajo demanda y capacidades mínima

Dado un grafo $G = (V, E)$ con $c(e)$, $c(e) >= l(e) >= 0$ para cada arista $e \in E$ y $d(v) v \in V$, definimos la circulación como una función $f : E \to R^+$ tal que:

1. *Criterio de capacidad*: Para toda $e \in E$,

$$l(e) \leq f(e) \leq c(e)$$

2. *Criterio de conservación* Para todo $v \in V$

En este tipo de problemas lo que queremos encontrar es si existe una circulación factible en G que satishaga las dos condiciones anteriores. Para encontrar una circulación factible en G tenemos que hacer las siguientes transformaciones para eliminar las demandas y las capacidades mínimas de G, obteniendo un nuevo grafo $G^{'}$:

1. *Eliminar capacidades mínimas:* para cada $e = (u, v) \in E$,

   a) Actualizar las capacidades

$$c' = c(e) - l(e)$$

   b) Actualizar demandas

$$d'(u) = d(e) + l(e)$$

$$d'(v) = d(e) - l(e)$$

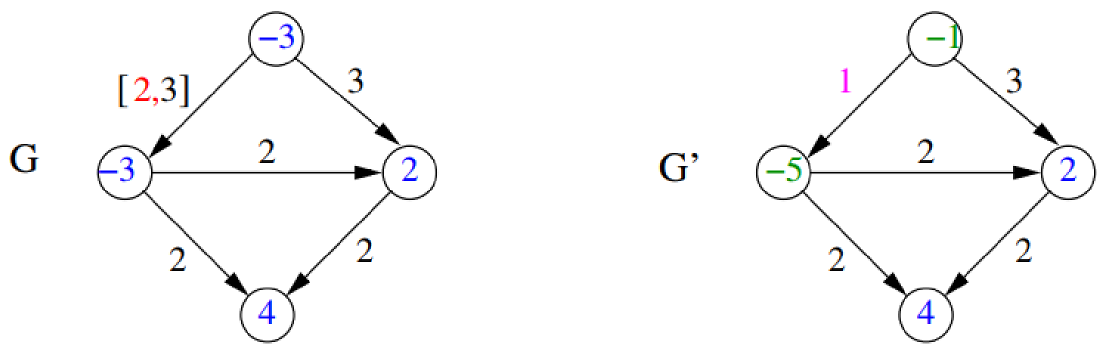

Figure 1: Ejemplo eliminación de capacidades mínimas

2. *Eliminar las demandas:*

   a) Añadir una nueva fuente s y un nuevo drenador t.

   b) Para cada $v \in V$ tal que $(d(v) < 0)$ añadir la arista (s, v) con capacidad $-d(v)$.

   c) Para cada $v \in V$ tal que $(d(v) > 0)$ añadir la arista (v, t) con capacidad $d(v)$.

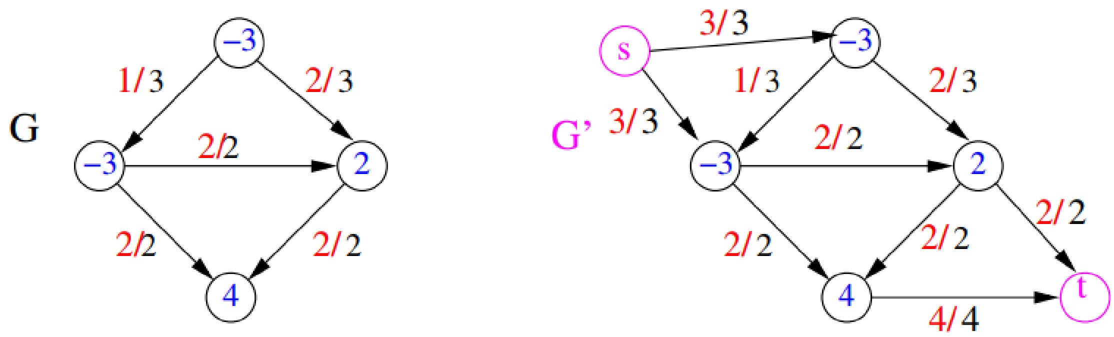

Figure 2: Ejemplo eliminación de demandas