

4.5. Una gran part del potencial d'empreses com Yahoo!, Google, Amazon, etc. es basa en el fet que milions de persones visiten cada dia les seves pàgines, (en anglès aquest fet s'anomena "eyeballs"). Un cop una persona visita una pàgina web d'aquestes companyes, de manera subliminal (o no tant) se intenta convèncer per a que deixen informació personal, la qual cosa permet que en el futur, si la mateixa persona torna a visitar la mateixa pàgina web, rebi informació i anuncis extremadament personalitzats i adreçats a l'usuari. Per exemple, si l'usuari ha transmès informació a Yahoo!, que té 20 anys i estudia a la UPC, a Barcelona, rebrà tota mena d'informació sobre lloguers d'habitacions a Barcelona, acadèmies etc. D'altra banda, si l'usuari és un alt executiu rebrà anuncis sobre cotxes de luxe, creuers per illes exòtiques o vacances a la lluna. Una tasca algorísmica interessant és personalitzar els anuncis seleccionant els més adients a cada persona.

Suposem que els administradors d'una web popular han identificat k grups de característiques diferents (perfils) G_1, G_2, \dots, G_k , que no són necessàriament disjunts, per exemple G_i pot ser viure a l'Hospitalet, G_j ser dona i G_k que estudia a la UPC. La companyia propietària de la web té contractes amb m empreses anunciadores per mostrar un nombre determinat de còpies dels seus anuncis als usuaris de les seves pàgines web. El contracte que la companyia i -èsima fa amb Yahoo! és del tipus:

- Per a un subconjunt $X_i \subseteq \{G_1, \dots, G_k\}$ dels grups demogràfics, i vol que els seus anuncis apareguin únicament a grups que són a X_i .
- Donat un enter r_i l'anunciant vol que els seus anuncis es mostren al menys a r_i usuaris.

Suposem que en un moment donat, hi ha n usuaris visitant la web. Com que tenim la informació de cadascun d'aquests usuaris, és fàcil conèixer el perfil de l'usuari j (per $j = 1, 2, \dots, n$), es dir a quin subconjunt $U_j \subseteq \{G_1, \dots, G_k\}$, pertanyé. Voldríem dissenyar un mecanisme tal que cada usuari vegi un únic anunci (d'uns pocs segons) de manera que per se satisfan restriccions imposades per els m anunciantes? És dir, per a cada $i = 1, 2, \dots, m$ com a mínim r_i usuaris, on cadascun pertanyé a almenys a un grup a X_i , veu un anunci proporcionat per l'anunciant i . Donar un algoritme eficient (polinòmic) per decidir si això és possible, i si és així, per a triar un anunci que aparegui per a cada usuari.

Este problema es un problema de asignación típico de los problemas de flujos.

Construiremos un grafo que represente el problema, encontrando el flujo máximo, encontraremos la asignación (si existe) y la podremos dar.

Construimos un grafo de la forma.

Sabiendo que tenemos k grupos diferentes, m empresas anunciadoras, el número de usuario que cada empresa quiere dar como mínimo, n usuarios y por cada usuario y empresa el subconjunto de características que tiene o en las que esta interesada respectivamente.

Tenemos un vértice s , del que salen n aristas de capacidad 1 a un vértice. Cada uno de los n vértices representa un usuario.

Al otro lado del grafo, tenemos un vértice t , al que llegan m aristas de capacidad r_i (siendo i cada una de las empresas).

En medio del grafo, conectamos cada uno de los vértices que representa un usuario con aquellas empresas con las que hace *matching*. Estas aristas tienen capacidad infinita, o suficientemente grande (incluso capacidad 1 ya sirve).

Con esta configuración, si al usar Ford Fulkerson tenemos que el flujo máximo es n (que es el flujo más alto posible dado que desde s sale como mucho n unidades de flujo), sabemos que sí es posible asignar a cada usuario un anuncio cumpliendo las restricciones del enunciado.

Finalmente, para decidir que anuncio corresponde a cada usuario, hemos de ver el grafo residual. Encontramos los vértices a los que se puede llegar desde s en el grafo residual (por ejemplo con un DFS). Una vez los encontremos, tenemos el (s, t) corte de capacidad máxima, y en particular las

aristas (que son las que hay desde los vértices accesibles desde s a los que no lo son) que participan en el corte. Esas aristas son las que nos están dando la asignación, pues van de un vértice que representa un usuario a un vértice que representa una compañía anunciante.

El coste del algoritmo se divide en varias fases

La construcción del grafo es $O(n+m+n*m)$ (m aquí es el número de empresas no de aristas)

Ford fulkerson tiene coste $O(C*(n+m))$ (otra vez, m es el número de empresas). Ya que C es el máximo flujo posible y este es como mucho n , el coste de este paso es $O(n*(n+m)) = O(n^2+n*m)$

Finalmente, el paso de encontrar la asignación en sí, tiene coste $O(n+m)$

Por lo que el coste final del algoritmo es $O(n^2+n*m)$