Contents

Flujos: Greedy.

notación g(A, B)

g es una función sobre los lados y $A, B \subseteq V$ $g(A, B) = \quad x,y[x \in A][y \in B][\overrightarrow{xy} \in E]g(\overrightarrow{xy})$

Propiedad:

Sean f, g funciones sobre los lados tales que

$g(\overrightarrow{xy}) \leq f(\overrightarrow{xy}) \ \forall \overrightarrow{xy} \in E$

Entonces

$g(A, B) \leq f(A, B) \ \forall A, B \subseteq V$

Criterio simple para maximalidad

Propiedad:

Sea f flujo en un network N tal que $v(f) = c(\{s\}, V)$. Entonces f es maximal.

Existencia

de la definición no es claro que EXISTA un flujo maximal.

flujo sea “entero”,

es decir que las capacidades y el flujo en cada lado deben ser números enteros,

como hay una cantidad finita de flujos enteros, es claro que existe un flujo entero maximal.

Encontrando flujos maximales

Comenzando desde algún flujo (pej el nulo) ir encontrando caminos dirigidos desde s a t

aumentando el flujo a lo largo de ese camino teniendo en cuenta de no mandar mas flujo por el mismo que lo que pueden soportar los lados.

una vez detectado un camino y obtenido una cota superior para cuanto podemos mandar por ese camino: ¿cuanto mandamos?

Lo mas obvio sería ser greedy y mandar todo lo que se pueda, y eso hace el algoritmo Greedy

Greedy

Algoritmo

Comenzar con $f = 0$ (es decir, $f(\overrightarrow{xy}) = 0 \ \forall \overrightarrow{xy} \in E$).

Buscar un camino dirigido $s = x0, x1, ..., xr = t$, con $\overrightarrow{xixi+1} \in E$ tal que $f(\overrightarrow{xixi+1}) < c(\overrightarrow{xixi+1})$ para todo $\in i = 0, ..., r - 1$.

(llamaremos a un tal camino un camino dirigido “no saturado” .)

Calcular $= \min\{c(\overrightarrow{xixi+1}) - f(xixi+1)\}$.

Aumentar f a lo largo del camino de 2. en , como se explicó antes.

Repetir 2 hasta que no se puedan hallar mas caminos con esas condiciones.

Conclusiones sobre Greedy

este Greedy no necesariamente va a encontrar un flujo maximal.

eligiendo inteligentemente los caminos encontramos un flujo maximal.

el Greedy de caminos puede ser modificado para encontrar un flujo maximal en tiempo polinomial

Not Greedy

En el caso de flujos, se puede construir un algoritmo que corre Greedy y cuando llega a un cierto punto, "SE DA CUENTA" que se equivocó en la elección de los caminos

y CORREGIR los errores.

Definición de Corte

Un Corte es un subconjunto de los vertices que tiene a s pero no tiene a t.

Capacidad de un Corte

La capacidad de un corte es $\text{cap}(S) = c(S, S)$, donde $S = V - S$

Definición:

Un corte es MINIMAL si su capacidad es la menor de las capacidades de todos los cortes.