Definiciones

miércoles, 23 de marzo de 2022 15:40

Grafo dirijido:

(V, E) es un grafo dirijido $\Leftrightarrow$

V es un conjunto finito

$E \subseteq V^2$

Notación:

$\overrightarrow{xy} = (x, y)$

Si $f: V^2 \to \mathbb{R}$ se define:

$f : \mathcal{P}(V)^2 \to \mathbb{R}$

$$f(A, B) = \sum_{\overrightarrow{xy} \in (A \times B) \cap E} f(\overrightarrow{xy})$$

Vecinos:

Sea:

G un grafo dirijido

$G = (V, E)$

$x \in V$

$\Gamma_G^+(x) = \{y \in V : \overrightarrow{xy} \in E\}$

$\Gamma_G^-(x) = \{y \in V : \overrightarrow{yx} \in E\}$

Network:

(V, E, c) es un network $\Leftrightarrow$

(V, E) es un grafo dirijido

$c : E \to \mathbb{R}_{>0}$

Sea:

(V, E, c) un network

(V, E, c) es entero $\Leftrightarrow \langle \forall e \in E : c(e) \in \mathbb{Z} \rangle$

in y out:

Sea:

(V, E, c) es un network

$g : E \to \mathbb{R}$

$\text{out}_g(x) = g(\{x\}, V)$

$\text{in}_g(x) = g(V, \{x\})$

Flujos:
Sea:
(V, E, c) un network
$s, t \in V$
$f : E \to \mathbb{R}_{\geq 0}$

f es un flujo de s a t $\Leftrightarrow$

$\forall \overrightarrow{xy} \in E : f(\overrightarrow{xy}) \leq c(\overrightarrow{xy})$ (feasibilidad)
$\forall x \in V - \{s, t\} : \text{in}_f(x) = \text{out}_f(x)$ (conservación)
$\text{out}_f(s) \geq \text{in}_f(s)$ (s es productor)
$\text{in}_f(t) \geq \text{out}_f(t)$ (t es consumidor)

Sea:
f es un flujo de s a t

f es un flujo entero $\Leftrightarrow \langle \forall e \in E : f(e) \in \mathbb{Z} \rangle$

Valor y flujo maximal:
Sea
(V, E, c) un network
f un flujo de s a t

$v(f) = \text{out}_f(s) - \text{in}_f(s)$
f es un flujo maximal $\Leftrightarrow \langle \forall g$ flujo de s a $t : v(f) \geq v(g) \rangle$

Corte:
Sea
(V, E, c) un network
$s, t \in V$
$S \subseteq V$

S es un corte $\Leftrightarrow s \in S \wedge t \notin S$

Si S es un corte, se define:

$\text{cap}(S) = c(S, V - S)$
S es minimal $\Leftrightarrow \langle \forall T$ corte : $\text{cap}(S) \leq \text{cap}(T) \rangle$

Caminos aumentantes:
Sea:
(V, E, c) un network
$s, t \in V$
f un flujo de s a t
$x_0, x_1, \ldots, x_r \in V$

$$x_0, x_1, \ldots, x_r \text{ es un } f\text{-camino aumentante} \Leftrightarrow$$
$$\left\langle \forall i \in \{0,1,\ldots,r-1\} : \left(\overrightarrow{x_i x_{i+1}} \in E \wedge f\left(\overrightarrow{x_i x_{i+1}}\right) < c\left(\overrightarrow{x_i x_{i+1}}\right)\right) \vee \left(\overrightarrow{x_{i+1} x_i} \in E \wedge f\left(\overrightarrow{x_{i+1} x_i}\right) > 0\right) \right\rangle$$

Integridad:
Sea:
(V, E, c) un network
$s, t \in V$
f un flujo de s a t

(V, E, c) es entero $\Leftrightarrow \langle \forall x \in E : c(e) \in \mathbb{Z} \rangle$
f es entero $\Leftrightarrow \langle \forall x \in E : f(e) \in \mathbb{Z} \rangle$

Teoremas

miércoles, 23 de marzo de 2022 14:42

Sea
$N = (V, E, c)$ un network
$s, t \in V$

$\forall f$ flujo, S corte : $v(f) = f(S, \bar{S}) - f(\bar{S}, S)$
$\forall f$ flujo, S corte : $v(f) \leq \text{cap}(S)$

Sea
$N = (V, E, c)$ un network
$s, t \in V$
f un flujo de s a t

f es maximal $\Leftrightarrow \nexists X$ camino : X es un camino f-aumentante
f es maximal $\Leftrightarrow \exists S$ corte de N : $\text{cap}(s) = v(f)$

Sea:
(V, E, c) un network entero

$\exists f$ flow : f es maximal $\wedge$ f es entero

Demostraciones (sin revisar)

miércoles, 23 de marzo de 2022 14:48

Flujos y cortes:

Sea

$N = (V, E, c)$ un network

$s, t \in V$

f un flujo de N

S un corte de N

Ⓐ $v(f) = f(S, \bar{S}) - f(\bar{S}, S)$

Ⓑ $v(f) \leq \text{cap}(S)$

Demostración de Ⓐ: $v(f) = f(S, \bar{S}) - f(\bar{S}, S)$

Pruebo algunas cosas antes:

① $f(A \cup B, C) = f(A, C) + f(B, C)$

(para $A, B, C \subseteq V$)

Demostración:

$$f(A \cup B, C)$$

=

$$\sum_{\overrightarrow{xy} \in E \cap (A \cup B) \times C} f(\overrightarrow{xy})$$

=

$$\sum_{\overrightarrow{xy} \in E \cap (A \times C \cup B \times C)} f(\overrightarrow{xy})$$

=

$$\sum_{\overrightarrow{xy} \in (E \cap A \times C) \cup (E \cap B \times C)} f(\overrightarrow{xy})$$

=

$$\sum_{\overrightarrow{xy} \in E \cap A \times C} f(\overrightarrow{xy}) + \sum_{\overrightarrow{xy} \in E \cap B \times C} f(\overrightarrow{xy})$$

=

$$f(A, C) + f(B, C)$$

② $f(A, B \cup C) = f(A, B) + f(A, C)$

(para $A, B, C \subseteq V$)

Demostración:

$$f(A, B \cup C)$$

=

$$\sum_{\overrightarrow{xy} \in E \cap A \times (B \cup C)} f(\overrightarrow{xy})$$

=

$$\sum_{\overrightarrow{xy} \in E \cap (A \times B \cup A \times C)} f(\overrightarrow{xy})$$

=

$$\sum_{\overrightarrow{xy} \in (E \cap A \times B) \cup (E \cap A \times C)} f(\overrightarrow{xy})$$

=

$$\sum_{\overrightarrow{xy} \in E \cap A \times B} f(\overrightarrow{xy}) + \sum_{\overrightarrow{xy} \in E \cap A \times C} f(\overrightarrow{xy})$$

=

$$f(A, B) + f(A, C)$$

Con esto pruebo lo que quiero probar:

$$v(f)$$

= {Definición v}

$$\mathrm{out}_f(s) - \mathrm{in}_f(s)$$

= {Sumo 0}

$$\mathrm{out}_f(s) - \mathrm{in}_f(s) + \sum_{x \in S - \{s\}} 0$$

= {S es corte

$\Rightarrow$ {Definición de corte}

$t \notin S$

$\Rightarrow$ {Todos los elementos que no son s ni t tienen in $=$ out}

$\forall x \in S - \{s\} : \mathrm{out}_f(x) - \mathrm{in}_f(x) = 0$

}

$$\mathrm{out}_f(s) - \mathrm{in}_f(s) + \sum_{x \in S - \{s\}} \left(\mathrm{out}_f(x) - \mathrm{in}_f(x)\right)$$

=

$$\sum_{x \in S} \left(\mathrm{out}_f(x) - \mathrm{in}_f(x)\right)$$

= {Definición de in y out}

$$\sum_{x \in S} f\left(\left(\{x\}, V\right) - f(V, \{x\})\right)$$

=

$$f(S, V) - f(V, S)$$

= {$V = S \cup \bar{S}$}

$$f(S, S \cup \bar{S}) - f(S \cup \bar{S}, S)$$

= {① y ②}

$$f(S, S) + f(S, \bar{S}) - f(S, S) - f(\bar{S}, S)$$

=

$$f(S, \bar{S}) - f(\bar{S}, S)$$

Demostración de Ⓑ: $v(f) \leq \mathrm{cap}(S)$

$$v(f)$$

= {Ⓐ}

$$f(S, \bar{S}) - f(\bar{S}, S)$$

$\leq$

$f(S, \bar{S})$

= {Por definición de cap}

$\mathrm{cap}(S)$

1)

viernes, 25 de marzo de 2022 15:49

I):Supongamos que cambiamos la definición de network permitiendo que cada lado tenga dos capacidades asociados: $c_1(\overrightarrow{xy})$ y $c_2(\overrightarrow{xy})$, con la condicion $c_1(\overrightarrow{xy}) \leq c_2(\overrightarrow{xy})$ para todo lado $\overrightarrow{xy}$. La definición de flujo ahora es modificada pidiendo que $c_1(\overrightarrow{xy}) \leq f(\overrightarrow{xy}) \leq c_2(\overrightarrow{xy})$ para todo lado $\overrightarrow{xy}$. (las demas condiciones para ser flujo quedan iguales).
Dar un ejemplo en donde con estas condiciones puede no existir ningún flujo de s a t.

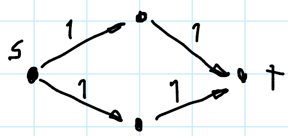
$$s \bullet \xrightarrow{1,2} \bullet \xrightarrow{3,4} \bullet\, t$$

2)

viernes, 25 de marzo de 2022 16:30

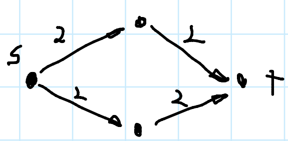
II):Si en un network dado las capacidades de todos los lados se incrementan en una constante k, (i.e, las nuevas capacidades son la viejas mas k en cada lado). ¿es cierto que el max flow value se incrementa en exactamente k unidades? ¿en a lo sumo k unidades? ¿en al menos k unidades?

¿Es cierto que el max flow se incrementa en exactamente k unidades?
Es falso, contra ejemplo:



max flow = 2

Agregando $k = 1$



maxflow = 4

¿en a lo sumo k unidades?
Es falso por el mismo contra-ejemplo

¿en al menos k unidades?
Es falso en el siguente Network:
$V = \{s, t\}$
$E = \emptyset$
$c : \emptyset \to \mathbb{R}_{>0}$

(V, E, c)

En el cual cualquier flujo tiene valor 0, tanto antes como después de agragarle k a las capacidades

Sin embargo si hay un camino entre s y t si es cierto, demostración:

Sea:
(V, E, c_1) un network conexo
$s, t \in V$

$c_2(x) = c_1(x) + k$

$$v(\text{maxflow}_{c_2}) \geq v(\text{maxflow}_{c_1}) + k$$

Demostración:

Sea:
f un flujo maximal con c_1
$x_1, x_2, \dots, x_r$ un camino de s a t

$g : E \to \mathbb{R}_{\geq 0}$
$g(\overrightarrow{x_i x_{i+1}}) = f(\overrightarrow{x_i x_{i+1}}) + k$
$g(e) = f(e)$ para el resto de aristas

Notar que g es un flujo ya que:

Para los lados del camino:

$$\begin{aligned} & g(\overrightarrow{x_i x_{i+1}}) \\ = & \{\text{Definición } g\} \\ & f(\overrightarrow{x_i x_{i+1}}) + k \\ \geq & \{f \text{ es maximal}\} \\ & c_1(\overrightarrow{x_i x_{i+1}}) + k \\ = & \{\text{Definición } c_2\} \\ & c_2(\overrightarrow{x_i x_{i+1}}) + k \end{aligned}$$

Para el resto de lados:

$$\begin{aligned} & v(\text{maxflow}_{c_2}) \\ \geq & \\ & v(g) \\ = & \\ & \text{out}_g(s) - \text{in}_g(s) \\ = & \end{aligned}$$

$$g(\{s\}, V - \{s\}) - g(V - \{s\}, \{s\})$$

$= \{$Solo un elemento de $\{s\} \times V - \{s\}$ y ninguno de $V - \{s\} \times \{s\}$ es $x_i x_{i+1}\}$

$$f(\{s\}, V - \{s\}) + k - f(V - \{s\}, \{s\})$$

$=$

$$\text{out}_f(s) - \text{in}_f(s) + k$$

$=$

$$v(f) + k$$

$= \{f$ es un flujo maximal$\}$

$$\text{maxflow}_{c_1} + k$$

3)

sábado, 26 de marzo de 2022 9:28

III):Supongamos que tiene una caja negra que que resuelve el max flow problem en networks sin arcos paralelos (en particular si existe $\overrightarrow{xy}$ no existe el $\overrightarrow{yx}$) ni loops (lados $\overrightarrow{xx}$). (i.e., la caja negra recibe un network en cierto formato y resuelve el max flow problem para ese network correctamente si el network no tiene lados paralelos ni loops.). Usted tiene sin embargo varios networks en los cuales hay lados paralelos entre vertices (tanto de ida como de vuelta, pero tambien lados multiples), y loops. ¿Como resuelve este problema? (lo que se pide es un algoritmo que transforme estos networks en networks permisibles para la caja negra y que luego transforme las respuestas de la caja negra en respuestas a su problema).

Input:
(V, E, c) que forman un seudo-network

Transformación en un network:

A cada arista la transformo en tres aristas, con dos vértices intermedios que solo están conectados ahí. Con las capacidades, le asigno a cada arista nueva la capacidad de la arista vieja.

O sea, para cada arista $e_k = \overrightarrow{x_i x_j}$ la transoformo en las aristas $\overrightarrow{x_i y_{k,0}}, \overrightarrow{y_{k,0} y_{k,1}}, \overrightarrow{y_{k,1}, x_j}$ (y agrego $y_{k,0}$ y y_{k1} al conjunto de vértices) y con las capacidades, si $c(e_k) = t_k$ hago que $c'(\overrightarrow{x_i y_{k,0}}) = t_k, c'(\overrightarrow{y_{k,0} y_{k,1}}) = t_k,\ c'(\overrightarrow{y_{k,1}, x_j}) = t_k$

Demostración de que obtengo un network:

Los nuevos vértices $y_{k,0}$ y $y_{k,1}$ solo están en ese segmento del network, lo cuál garantiza que no haya multiples aristas conectando los mismos vértices, y que no haya aristas que conectan un vértice con si mismo.

Las nuevas capacidades también son validas, porque si las viejas eran > 0, las nuevas también lo son

Transformación del flujo en el seudoflujo:

Si obtuve el flujo f', esta claro que $f'(\overrightarrow{x_i y_{k,0}}) = f'(\overrightarrow{y_{k,0} y_{k,1}}) = f'(\overrightarrow{y_{k,1}, x_j})$ porque $y_{k,0}$ y $y_{k,1}$ solo tienen una entrada y una salida, entonces hago que el seudoflujo $f(e_k) = f'(\overrightarrow{x_i y_{k,0}})$

Demostración de que el seudoflujo es valido:

$f'(\overrightarrow{x_i y_{k,0}}) \leq c'(\overrightarrow{x_i y_{k,0}})$

$\Rightarrow f(e_k) \leq c(k)$

Lo primero es cierto por que la caja negra es correcta, así que lo segundo también lo es

Demostración de que el seudo flujo es maximal:

Si f no fuera maximal, signifixa que existe g talque $v(g) > v(f)$. Sea:

$g'(\overrightarrow{x_i y_{k,0}}) = g(e_k)$

$g'(\overrightarrow{y_{k,0} y_{k,1}}) = g(e_k)$

$g'(\overrightarrow{y_{k,1}, x_j}) = g(e_k)$

Es claro que $v(g) = v(g')$ y $v(f) = v(f')$, por ende $v(g') > v(f')$, lo cuál significaria que f' no es maximal. Esta es una contradición con que la caja negra funciona bien, por ende, f es maximal

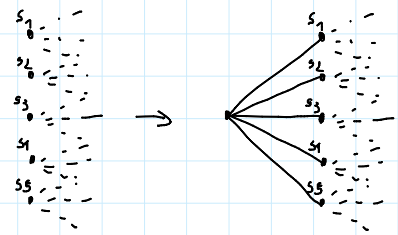
4)

miércoles, 30 de marzo de 2022 17:22

IV):Suponga que tiene un "network" con multiples fuentes $s_1, s_2, \ldots, s_k$ y multiples resumideros $t_1, t_2, \ldots, t_r$. Transforme este problema en un Max flow problem usual.

Sea:
(V, E, c) un seudonetwork

Transformación en un network:



$V' = V \cup \{s, t\}$
$E' = E \cup \{\overrightarrow{ss_1}, \overrightarrow{ss_2}, \ldots, \overrightarrow{ss_k}, \overrightarrow{t_1t}, \overrightarrow{t_2t}, \ldots, \overrightarrow{t_rt}\}$
$c' : V' \to \mathbb{R}_{>0}$

$$c'(ss_i) = \sum_{x \in V : \overrightarrow{s_ix} \in E} c(\overrightarrow{s_ix})$$

$$c'(t_it) = \sum_{x \in V : \overrightarrow{xt_i} \in E} c(\overrightarrow{xt_i})$$

$c'(e) = c(e)$ (para el resto de aristas)

Es un network porque V' es un conjunto, $E' \subseteq V'^2$ y $c' : V' \to \mathbb{R}_{>0}$

La caja negra devuelve un flow $f' : E' \to \mathbb{R}_{\geq 0}$
Transformación en el seudoflow $f : E \to \mathbb{R}_{\geq 0}$:
$f(e) = f'(e)$

Es un seudoflow valido, ya que por ser el flow valido:
$f'(e) \leq c'(e)$
$\Rightarrow f(e) \leq c(e)$

Es maximal, lo pruebo por contradición:
Sea $g : E \to \mathbb{R}_{\geq 0}$ un seudoflow tal que $v(g) > v(f)$
Y sea un $g' : E' \to \mathbb{R}_{\geq 0}$

$$g'^{(ss_i)} = \sum_{x \in V : \overrightarrow{s_ix} \in E} g(\overrightarrow{s_ix}) - \sum_{x \in V : \overrightarrow{xs_i} \in E} g(\overrightarrow{xs_i})$$

$$g'(t_it) = \sum_{x \in V : \overrightarrow{xt_i} \in E} g(\overrightarrow{xt_i}) - \sum_{x \in V : \overrightarrow{t_ix} \in E} g(\overrightarrow{t_ix})$$

$g'(e) = g(e)$

Notar que $v(f') = v(f)$ y $v(g') = v(g)$ ya que a s no entra nada, y lo que sale de s es exactamente lo mismo que sale de los s_i menos lo que entra a los s_i

Entonces:
$v(g) > v(f)$
$\Rightarrow v(g') > v(f')$

Sin embargo, esto es una contradicción porque la caja negra da un flow maximal, por ende, f es maximal

5)

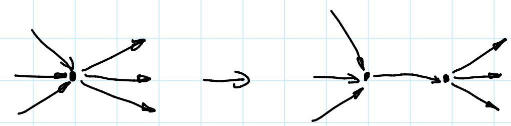
viernes, 1 de abril de 2022 8:34

V):Supongamos que tenemos un network donde ademas de las capacidades de los lados, los vertices tambien tienen capacidades. Un flujo, ademas de las restricciones en los lados tiene la restricción que el flujo que pasa por un vertice x no puede ser mayor a su capacidad. Se desea hallar un flujo maximal con estas condiciones. Transformar el problema en un problema de Max Flow común.

Input:
(V, E, c)
$c : E \cup V \to \mathbb{R}_{>0}$

Sea:
$V = \{x_0, x_1, \dots, x_{n-1}\}$
$s = x_\sigma$
$t = x_\tau$

Transformación:



$V' = \{x_{0,i}, x_{1,i} : x_i \in V\}$
$s' = x_{0,\sigma}$
$t' = x_{1,\tau}$
$E' = \{\overrightarrow{x_{1,i}x_{0,j}} : \overrightarrow{x_i x_j} \in E\} \cup \{\overrightarrow{x_{0,i}x_{1,i}} : x_i \in V\}$
$c' : E' \to \mathbb{R}_{>0}$
$c(\overrightarrow{x_{0,i}x_{1,i}}) = c(x_i)$
$c(\overrightarrow{x_{1,i}x_{0,j}}) = c(\overrightarrow{x_i x_j})$

Claramente (V', E', c') es un networck valido

El algoritmo devuelve un flow $f' : E' \to \mathbb{R}_{\geq 0}$, transformación en seudoflow:
$f : E \cup V \to \mathbb{R}_{\geq 0}$
$f(x_i) = f(\overrightarrow{x_{0,i}x_{1,i}})$
$f(\overrightarrow{x_i x_j}) = f(\overrightarrow{x_{1,i}x_{0,j}})$

Este es un seudoflow valido ya que le asigna un valor a todos los vértices y todas las aristas, y esta es la demostración de que es menor que las capacidades:

$f(x_i) \leq c(x_i) \wedge f(\overrightarrow{x_i x_j}) \leq c(\overrightarrow{x_i x_j})$
$\Leftarrow$
$f(\overrightarrow{x_{0,i}x_{1,i}}) \leq c(\overrightarrow{x_{0,i}x_{1,i}})\ f(\overrightarrow{x_{1,i}x_{0,j}}) \leq c(\overrightarrow{x_{1,i}x_{0,j}})$
$\Leftarrow \{f'$ es un flow valido$\}$
True

Demostración de que es maximal por contradición.

Si no lo fuera existiría $g : E \cup V \to \mathbb{R}_{\geq 0}$ tal que $v(g) > v(f)$
Sea:
$g' : V' \to \mathbb{R}_{\geq 0}$
$g(\overrightarrow{x_{0,i}x_{1,i}}) = g(x_i)$
$g(\overrightarrow{x_{1,i}x_{0,j}}) = g(\overrightarrow{x_i x_j})$

Notar que $v(g') = v(g)$ y $v(f') = v(f)$ ya que lo que sale de $x_{0,\sigma}$ es lo mismo que lo que pasa por x_σ, y a ambos entra todo lo mismo

$v(g) > v(f)$
$\Rightarrow$
$v(g') > v(f')$
$\Rightarrow \{f'$ es maximal$\}$
False

6)

miércoles, 30 de marzo de 2022 14:16

VI):Supongamos que queremos hallar el max flow value de un network y decidimos usar el siguiente algoritmo: 1) buscar todos los cortes del network. 2) calcular sus capacidades. 3) retornar la menor de esas capacidades. Este algoritmo es correcto por el Max Flow Min Cut theorem. Calcule la complejidad del mismo.

Sea:
$G = (V, E, c)$ un network de s a t

$v(\text{maxflow}) = \min\{\text{cap}(S) : S \text{ es corte de } V\}$

Sea:
$n = |V|$
$m = |E|$

$\quad O(\min\{\text{cap}(S) : S \text{ es corte de } V\})$

=

$\quad O\left(\text{cap}(S)\right) \cdot (2^n - 2)$

≈ {En el peor caso de cap}

$\quad m \cdot 2^n$

≈

$\quad m2^n$

7)

miércoles, 30 de marzo de 2022 14:35

VII):Un flujo f es par si $f(\overrightarrow{xy})$ es par $\forall$ lado $\overrightarrow{xy}$. (Definición similar para flujo impar).
a) Probar que en un network en donde las capacidades de todos los lados son pares siempre existe algún flujo maximal par.
b) Dar un ejemplo de un network en donde las capacidades de todos los lados son pares pero en el cual existe un flujo maximal no par.
c) Dar un ejemplo de un network en donde las capacidades de todos los lados son impares, pero en el cual ningún flujo maximal sea impar.

a)
Sea:
(V,E,c) un network par

$$\langle \exists f \text{ flow} : f \text{ es maximal} : f \text{ es par} \rangle$$

Demostración:
Sea:
$c' : E \to \mathbb{R}_{>0}$
$c'(e) = \dfrac{c(e)}{2}$

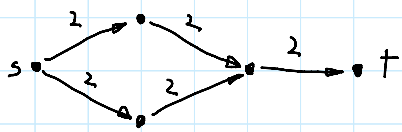
(V,E,c') es en network entero
Sea:
$f' : E \to \mathbb{R}_{\geq 0}$ un flujo maximal entero (con c') (existe por teorema de integridad)
$f : E \to \mathbb{R}_{\geq 0}$
$f(e) = 2f'(e)$

Claramente f es par, demuetro que es maximal por contradición:
Sea:
$g : E \to \mathbb{R}_{\geq 0}$ un flujo tal que $v(g) > v(f)$
$g' : E \to \mathbb{R}_{\geq 0}$
$g'(e) = \dfrac{g(e)}{2}$

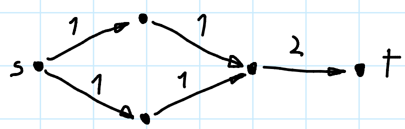
Notar que $v(g) = 2v(g')$ y $v(f) = 2v(f')$, ya que sale de s el doble en uno que en el otro
Pruebo que $v(g') > v(f')$:

$v(g) > v(f)$
$\Leftrightarrow$
$2v(g') > 2v(f')$
$\Leftrightarrow$
$v(g') > v(f')$
$\Leftrightarrow$ {f' es maximal}
False

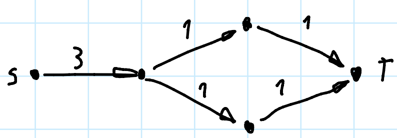
b)



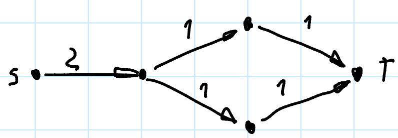
Es par, pero el siguiente flow es maximal no par



c)



El único flujo maximal es este:



El cual no es impar

8)

viernes, 1 de abril de 2022 17:20

VIII):El problema de los caminos disjuntos por lados es el siguiente: dado un grafo dirigido G y vertices s, t, hallar el mayor numero de caminos de s a t que no se toquen en ningun lado. Por ejemplo, si se desean mandar agentes secretos de s a t a traves de uno o mas vuelos intermedios sin que nunca dos agentes viajen en el mismo vuelo (pueden estar temporariariamente en la misma ciudad) y se desea saber cual es el numero maximo de agentes que se puede mandar. Reformular esto como un problema de flujo maximal entero, construyendo un network apropiado.

Input:
G un grafo dirijido
$G = (V, E)$
$s, t \in V$

Calcular la cantidad de caminos sin aristas en común es equivalente a encontrar una solución entera usando las capasidades:
$c : E \to \mathbb{R}_{>0}$
$c(e) = 1$

Esto es así, porque en ese caso, por que en el flow se manda 0 o 1, y por ende se pueden hacer los caminos sin que coincidan en aristas.
Con el algoritmo de Ford Fulkerson se puede obtener el flow entero

9)

viernes, 1 de abril de 2022 19:50

IX):Supongamos que tiene ya un flujo maximal en un network. Dar un algoritmo que en tiempo $O(m)$ encuentre un corte minimal a partir del flujo maximal dado.

Versión imperativa:

input:

(V, E, c) un network
$s, t \in V$
f un flujo maximal

Algoritmo:

$E' := \emptyset$

for $\overrightarrow{xy} \in E$:
  if $f(\overrightarrow{xy}) < c(\overrightarrow{xy})$:
    $E' := E' \cup \{\overrightarrow{xy}\}$
  elif $f(\overrightarrow{xy}) > 0$:
    $E' := E' \cup \{\overrightarrow{yx}\}$

$S = \mathrm{BFS}(V, E', s)$ (BFS devuelve el conjunto de vertices que visito)

return S

Es $\mathrm{O}(m)$ porque para construir E' se recorren los m lados y BFS es $\mathrm{O}(m)$

Demostración de que de devuelve un corte minimal

Lo pruebo probando que devuelve un corte con capacidad igual al valor del flujo:
Sean:
S el corte
E' el calculado por el algoritmo

$$v(f)$$
$$= \{\text{Definición } v\}$$
$$\sum_{\overrightarrow{xy} \in (S \times \bar{S}) \cap E} f(\overrightarrow{xy}) - \sum_{\overrightarrow{xy} \in (\bar{S} \times S) \cap E} f(\overrightarrow{xy})$$
$$= \{x \in S, y \notin S \Rightarrow \overrightarrow{xy} \notin E' \Rightarrow f(\overrightarrow{xy}) = c(\overrightarrow{xy})\}$$
$$\sum_{\overrightarrow{xy} \in (S \times \bar{S}) \cap E} c(\overrightarrow{xy}) - \sum_{\overrightarrow{xy} \in (\bar{S} \times S) \cap E} f(\overrightarrow{xy})$$
$$= \{x \notin S, y \in S \Rightarrow \overrightarrow{yx} \notin E' \Rightarrow f(\overrightarrow{xy}) = 0\}$$

$$= \sum_{\overrightarrow{xy} \in (S \times \bar{S}) \cap E} c(\overrightarrow{xy}) - \sum_{\overrightarrow{xy} \in (\bar{S} \times S) \cap E} 0$$

$$= \sum_{\overrightarrow{xy} \in (S \times \bar{S}) \cap E} c(\overrightarrow{xy})$$

$= \{\text{Definición cap}\}$

$$\mathrm{cap}(S)$$