CI2613: Algoritmos y Estructuras III

Blai Bonet

Universidad Simón Bolívar, Caracas, Venezuela

Enero-Marzo 2015

Flujo máximo en redes

Problema de flujo máximo

Dada un red de distribución y dos vértices distinguidos s y t, uno quiere enviar la mayor **cantidad de producto** de s a t sujeto a ciertas **restricciones de flujo**

Es un problema muy importante en las áreas de optimización combinatoria, investigación de operaciones, y computación

Es un caso especial del **problema de circulación**

Redes de flujo

Una red de flujo es un grafo dirigido $G = (V, E)$ con dos vértices distinguidos s y t, y con **capacidades de flujo** $c : V \times V \to \mathbb{R}^{\geq 0}$

Por lo general, el grafo G es **conectado**, y los vértices s y t son el único **vértice fuente** y único **vértice sumidero en G**

Suposiciones (sin perdida de generalidad):

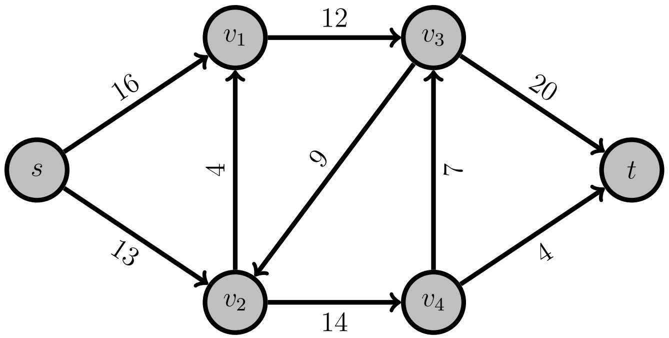
– Si E contiene la arista (u, v), entonces $(v, u) \notin E$

– Si $(u, v) \notin E$, entonces $c(u, v) = 0$

– No existen lazos; i.e. aristas (u, u)

– Todo vértice u pertenece a algún camino $s \rightsquigarrow t$
(por lo tanto, $|E| \geq |V| - 1$)

Redes de flujo: Ejemplo



Flujo

Una función de flujo $f : V \times V \to \mathbb{R}^{\geq 0}$ indica la **cantidad de producto** que debe enviarse entre cada par de nodos

El flujo se define sobre todos los pares $V \times V$, aunque a nosotros sólo nos interesan flujos f tales que si $f(u, v) > 0$, entonces $(u, v) \in E$

Para que una función $f : V \times V \to \mathbb{R}^{\geq 0}$ sea flujo, debe satisfacer:

– **Restricciones de capacidad:** para todo $u, v \in V$,

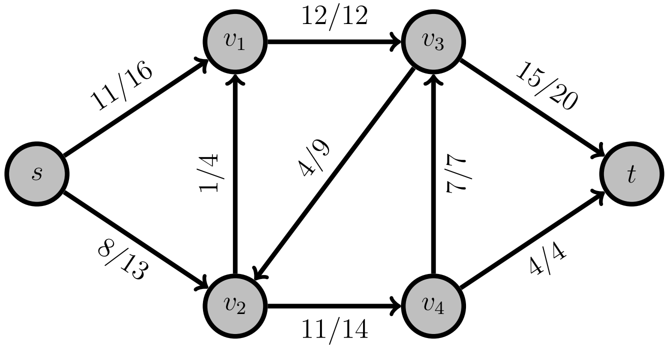
$$0 \leq f(u, v) \leq c(u, v)$$

– **Conservación de flujo:** para todo vértice $u \in V \setminus \{s, t\}$,

$$\sum_v f(u, v) \; - \; \sum_v f(v, u) \; = \; 0$$

Valor del flujo: $|f| \; = \; \sum_v f(s, v) - \sum_v f(v, s)$

Flujo: Ejemplo



Funciones de flujo básicas

- **Flujo nulo**: $f_\circ(u, v) = 0$ para todo par $u, v \in V$

- **Flujo a lo largo de un camino**: sea $p = (v_0, \ldots, v_k)$ un camino simple de s a t

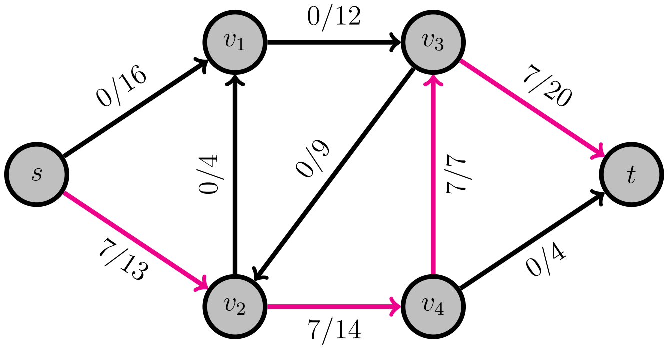
  Sea $c(p) = \min\{c(v_{i-1}, v_i) : i = 1, \ldots, k\}$ la menor capacidad a lo largo del camino p. Definimos

$$f_p(u, v) \;=\; \begin{cases} c(p) & \text{si } (u, v) \in p \\ 0 & \text{en otro caso} \end{cases}$$

Lema

Sea $G = (V, E)$ una red de flujo con capacidades $c : V \times V \to \mathbb{R}^{\geq 0}$. Sea p un camino simple de s a t en G. Las funciones $f_\circ$ y f_p son funciones de flujo sobre G

Flujo a lo largo de un camino: Ejemplo

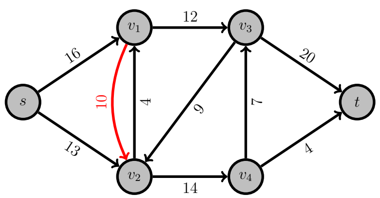


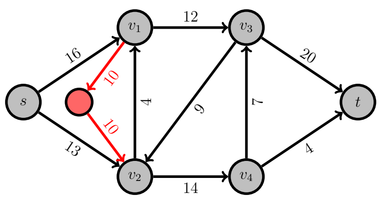
Problema de Flujo Máximo

Dada una red de flujo $G = (V, E)$ y capacidades $c : V \times V \to \mathbb{R}^{\geq 0}$, se quiere un flujo $f : V \times V \to \mathbb{R}^{\geq 0}$ de valor $|f|$ máximo

Consideraciones de modelaje

- Existencia de **aristas antiparalelas:**





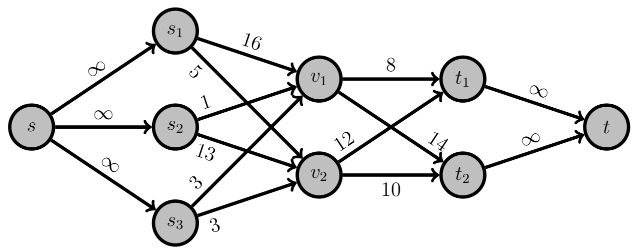
Consideraciones de modelaje

- Redes con **múltiples fuentes y mútiples sumideros:**

  Si existen múltiples fuentes $s_1, s_2, \ldots, s_m$ y sumideros $t_1, t_2, \ldots, t_n$, basta agregar una **superfuente** s y un **supersumidero** t, y las siguientes aristas y capacidades:

  – Aristas (s, s_i) con capacidad ∞ para $i = 1, \ldots, m$

  – Aristas (t_j, t) con capacidad ∞ para $i = 1, \ldots, n$



Método de Ford-Fulkerson

Método general para solucionar el problema de flujo máximo

Tres ideas fundamentales que también aparecen en otros algoritmos:

- Redes residuales
- Caminos de aumento de flujo
- Cortes

Provee una prueba al **Teorema de flujo máximo – corte mínimo**

Método de Ford-Fulkerson

Comenzamos desde el **flujo nulo**: $f_\circ(u,v) = 0$ para todo $u, v \in V$

En cada iteración, aumentamos el flujo utilizando un **camino de aumento** en la **red residual**

Termina cuando no existen caminos de aumento en la red residual

- Las implementaciones varían en torno al camino de aumento utilizado para aumentar el flujo
- El algoritmo de Edmonds-Karp es una instancia del método de Ford-Fulkerson

Método de Ford-Fulkerson: Pseudocódigo

```
Flujo Metodo-Ford-Fulkerson(G, s, t):
    % inicializar flujo inicial nulo
    Flujo f = 0

    % incrementar flujo de forma iterativa
    while exista un camino de aumento p en la red residual G_f
        aumentar el flujo f a lo largo del camino p

    % retornar flujo encontrado
    return f
```


Capacidades residuales

Para aumentar un flujo f cambiamos el flujo de algunas aristas

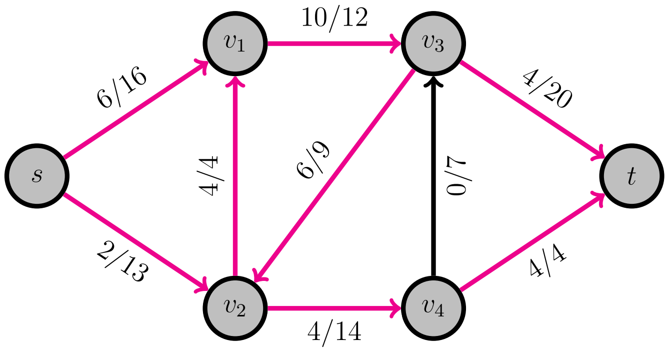
Aún cuando el flujo aumentado f' es de mayor valor (i.e. $|f'| > |f|$), es posible que el flujo a través de una o más aristas decrezca

La **posibilidad de cuanto aumentar/decrecer** el flujo sobre las aristas del grafo se hace explícita en la **red residual** G_f, la cual es definida con las capacidades residuales c_f:

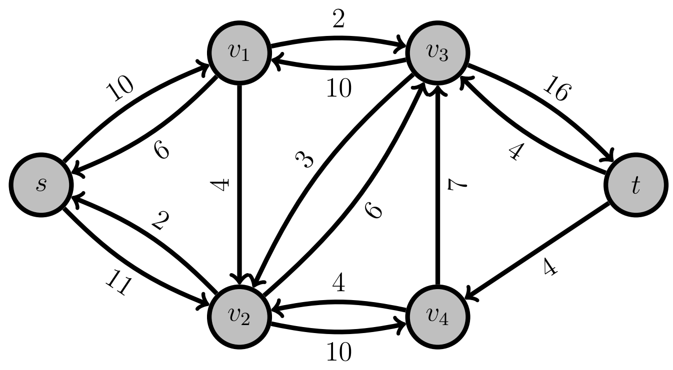
Las capacidades residuales $c_f : V \times V \to \mathbb{R}^{\geq 0}$ son:

$$c_f(u,v) = \begin{cases} c(u,v) - f(u,v) & \text{si } (u,v) \in E \\ f(v,u) & \text{si } (v,u) \in E \\ 0 & \text{en otro caso} \end{cases}$$

Redes residuales: Ejemplo



Redes residuales: Ejemplo



Redes residuales

Las capacidades residuales c_f definen la **red residual** $G_f = (V, E_f)$:

$$E_f \;=\; \{(u,v) \in V \times V : c_f(u,v) > 0\}$$

Cada arista $(u,v) \in E$ puede **generar hasta dos aristas en G_f**: una en el sentido $u \to v$ y otra en el sentido $v \to u$:

– $(u,v) \in G_f \iff c_f(u,v) > 0 \iff f(u,v) < c(u,v)$

– $(v,u) \in G_f \iff f(u,v) > 0$

Claramente $|E_f| \leq 2|E|$

Flujos en la red residual

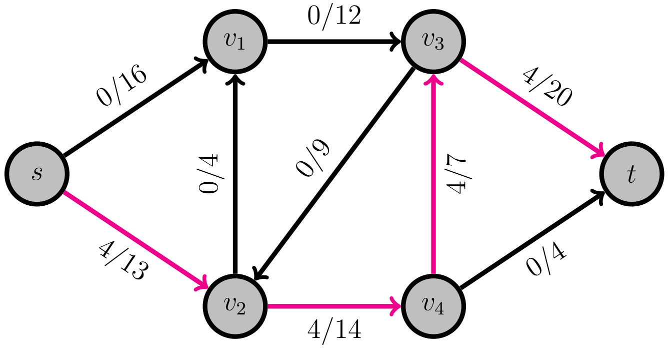
La red residual junto a las capacidades residuales forman un problema de flujo en redes (excepto que G_f puede tener **aristas antiparalelas**)

Sea f un flujo sobre G y f' un flujo sobre la red residual G_f

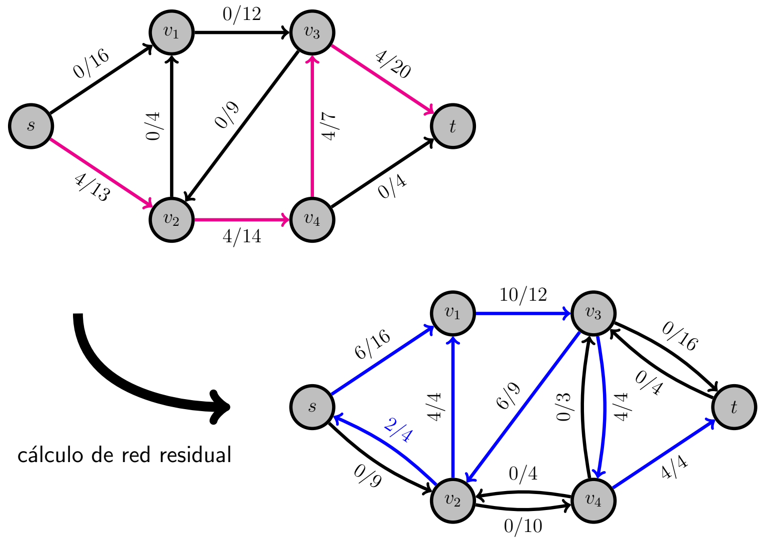
El aumento de f por f', escrito $f \uparrow f'$, es:

$$(f \uparrow f')(u,v) \;=\; \begin{cases} f(u,v) + f'(u,v) - f'(v,u) & \text{si } (u,v) \in E \\ 0 & \text{en otro caso} \end{cases}$$

Aumento de flujo: Ejemplo

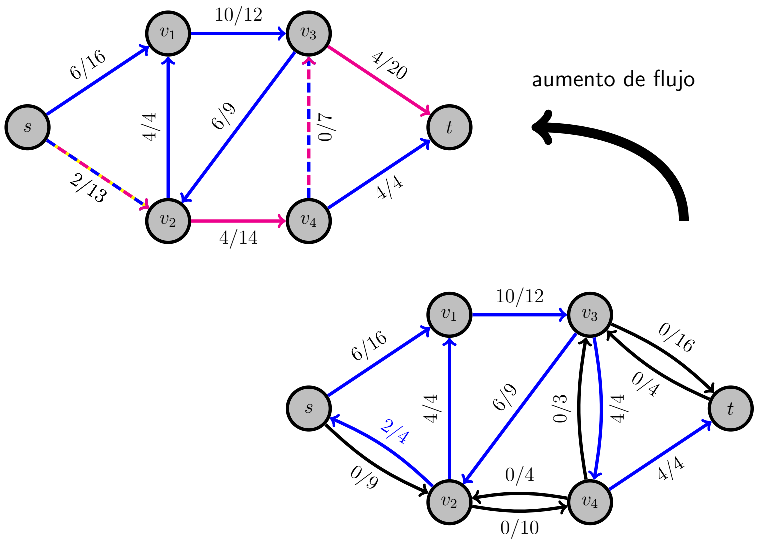


Aumento de flujo: Ejemplo



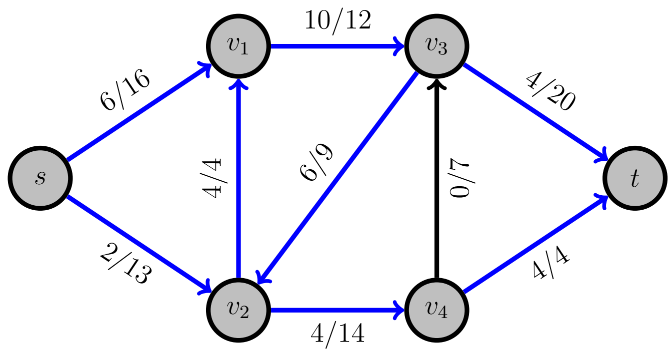
flujo en red residual

Aumento de flujo: Ejemplo



flujo en red residual

Aumento de flujo: Ejemplo



Lema de aumento de flujo

Lema

Sea $G = (V, E)$ una red de flujo y f un flujo sobre G. Sea G_f la red residual para f y f' un flujo sobre G_f. Entonces, $f \uparrow f'$ es un flujo sobre G con valor $|f \uparrow f'| = |f| + |f'|$

Prueba:

Debemos:

1. Verificar que $f \uparrow f'$ satisface las restricciones de capacidad
2. Verificar que $f \uparrow f'$ satisface las restricciones de flujo
3. Calcular el valor de $f \uparrow f'$

Lema de aumento de flujo: Prueba

1. **Restricciones de capacidad:**

- Si $(u, v) \notin E$, $(f \uparrow f')(u, v) = 0$
- Si $(u, v) \in E$,

$$\begin{aligned} (f \uparrow f')(u,v) &= f(u,v) + f'(u,v) - f'(v,u) \\ &\geq f(u,v) + f'(u,v) - f(u,v) \\ &= f'(u,v) \\ &\geq 0 \end{aligned}$$

$$\begin{aligned} (f \uparrow f')(u,v) &= f(u,v) + f'(u,v) - f'(v,u) \\ &\leq f(u,v) + f'(u,v) \\ &\leq f(u,v) + c_f(u,v) \\ &= f(u,v) + c(u,v) - f(u,v) \\ &= c(u,v) \end{aligned}$$

Lema de aumento de flujo: Prueba

2. **Conservación de flujo:**

$$\begin{aligned} \sum_v (f \uparrow f')(u,v) &= \sum_v f(u,v) + f'(u,v) - f'(v,u) \\ &= \sum_v f(u,v) + \sum_v f'(u,v) - \sum_v f'(v,u) \\ &= \sum_v f(v,u) + \sum_v f'(v,u) - \sum_v f'(u,v) \\ &= \sum_v f(v,u) + f'(v,u) - f'(u,v) \\ &= \sum_v (f \uparrow f')(v,u) \end{aligned}$$

Lema de aumento de flujo: Prueba

3. **Cálculo del valor del flujo $|f|$:**

Defina $V_1 = \{v \in V : (s, v) \in E\}$ y $V_2 = \{v \in V : (v, s) \in E\}$

Por la inexistencia de aristas antiparalelas, $V_1 \cap V_2 = \emptyset$

$$\begin{aligned} |f \uparrow f'| &= \sum_v (f \uparrow f')(s,v) - \sum_v (f \uparrow f')(v,s) \\ &= \sum_{v \in V_1} (f \uparrow f')(s,v) - \sum_{v \in V_2} (f \uparrow f')(v,s) \\ &= (f \uparrow f')(s, V_1) - (f \uparrow f')(V_2, s) \end{aligned}$$

donde $f(X, Y) = \sum_{x \in X} \sum_{y \in Y} f(x, y)$

Lema de aumento de flujo: Prueba

❸ Cálculo del valor del flujo $|f|$:

$$
\begin{aligned}
|f \uparrow f'| &= (f \uparrow f')(s, V_1) \; - \; (f \uparrow f')(V_2, s) \\
&= [f(s, V_1) + f'(s, V_1) - f'(V_1, s)] - [f(V_2, s) + f'(V_2, s) - f'(s, V_2)] \\
&= f(s, V_1) - f(V_2, s) + f'(s, V_1) + f'(s, V_2) - f'(V_1, s) - f'(V_2, s) \\
&= f(s, V_1) - f(V_2, s) + \sum_v f'(s, v) - \sum_v f'(v, s)
\end{aligned}
$$

Observar $f(s, V_1) = \sum_v f(s, v)$ y $f(V_2, s) = \sum_v f(v, s)$ (ejercicio)

$$
\begin{aligned}
&= \sum_v f(s, v) - \sum_v f(v, s) + \sum_v f'(s, v) - \sum_v f'(v, s) \\
&= |f| + |f'|
\end{aligned}
$$

□

Caminos de aumento

Dada una red de flujo $G = (V, E)$ y un flujo p, un camino de aumento es un **camino simple p de s a t en la red residual G_f**

El camino p define una función de flujo f_p sobre la red residual

Por el Lema de aumento de flujo, $f \uparrow f_p$ es un flujo sobre G con valor $|f| + |f_p| = |f| + c(p)$

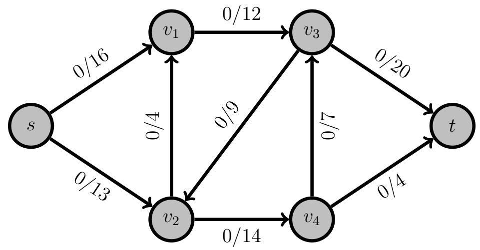
Por definicion de G_f: $(u, v) \in E_f \iff c_f(u, v) > 0$

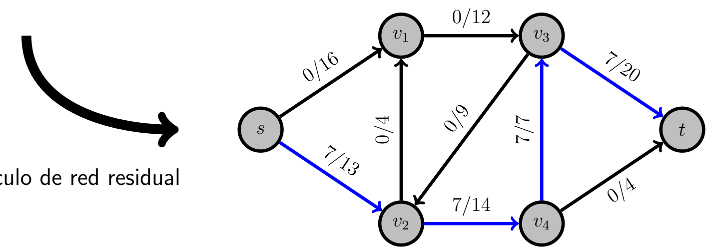
Por lo tanto, $c(p) > 0$ y $|f \uparrow f_p| = |f| + |f_p| = |f| + c(p) > |f|$

Algoritmo básico de Ford-Fulkerson: Pseudocódigo

```
Flujo Ford-Fulkerson(G, s, t):
    % inicializar flujo inicial nulo
    foreach (u,v) ∈ E
        f(u,v) = 0

    % incrementar flujo de forma iterativa
    while exista un camino p de s a t en la red residual G_f
        c(p) = min { c_f(u,v) : (u,v) ∈ p }
        foreach (u,v) ∈ p
            if (u,v) ∈ E
                f(u,v) = f(u,v) + c(p)
            else
                f(u,v) = f(u,v) - c(p)

    % retornar flujo encontrado
    return f
```


Algoritmo de Ford-Fulkerson: Ejemplo

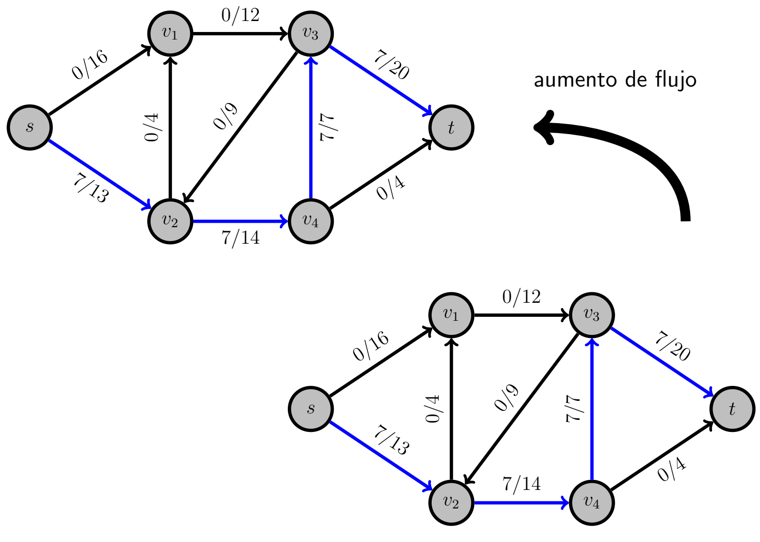




cálculo de red residual

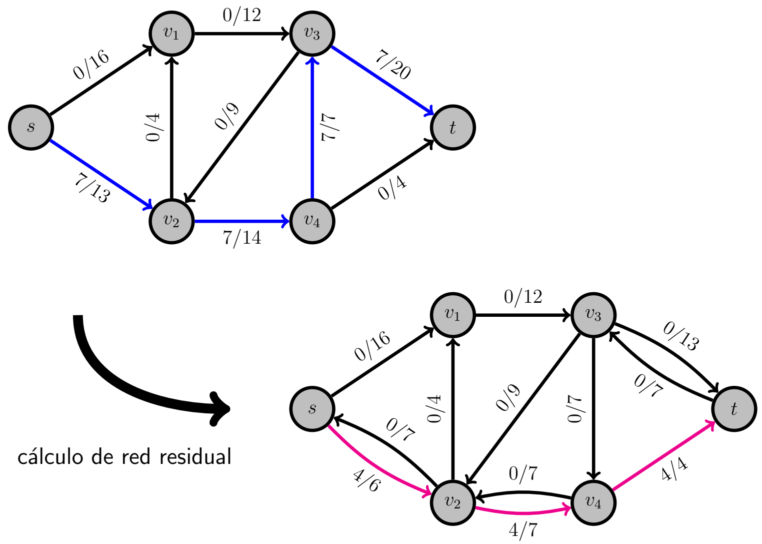
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



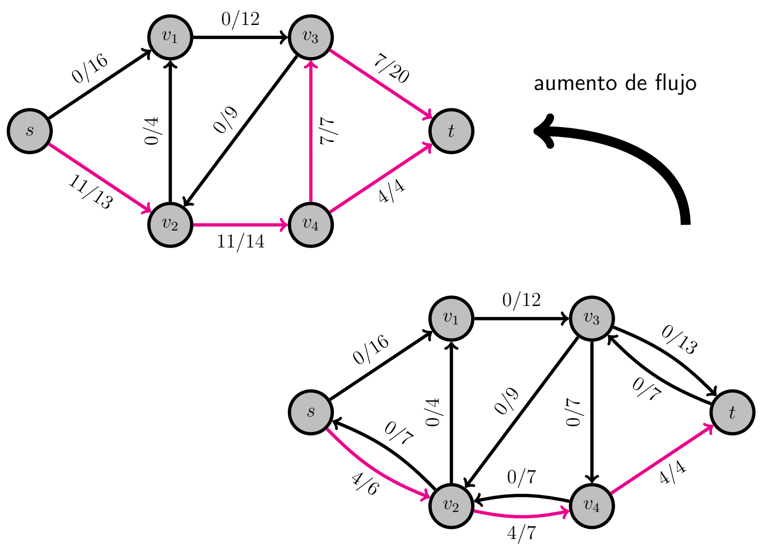
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



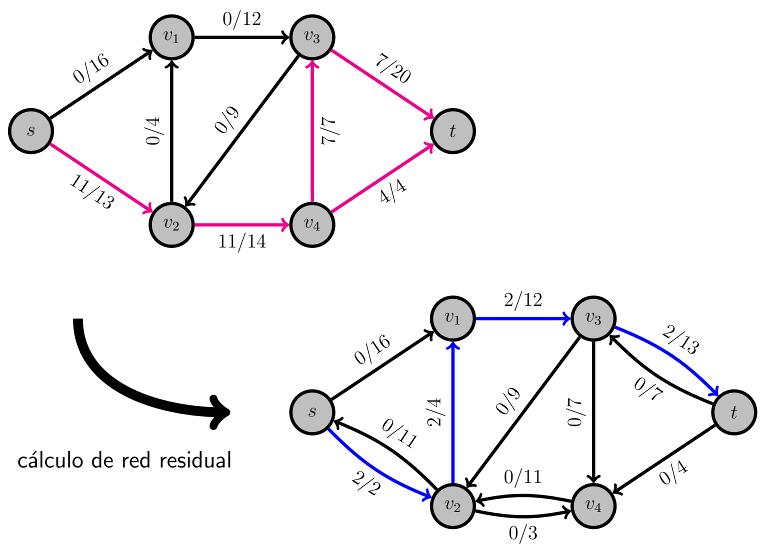
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



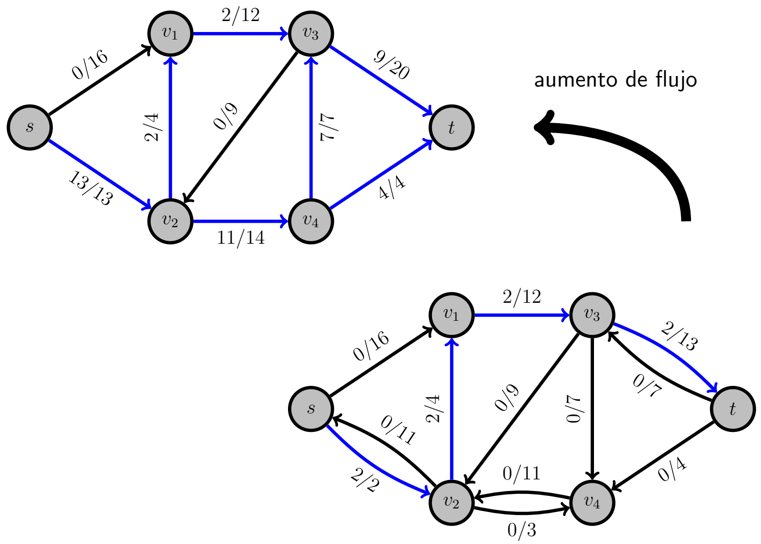
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



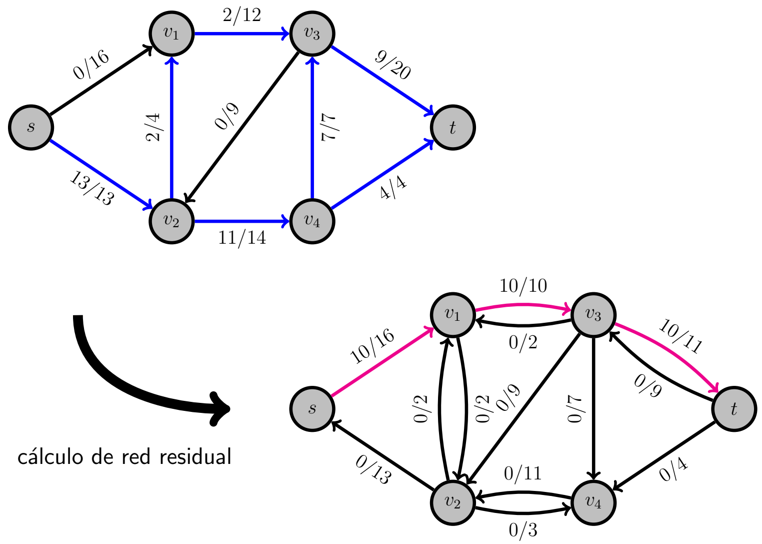
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



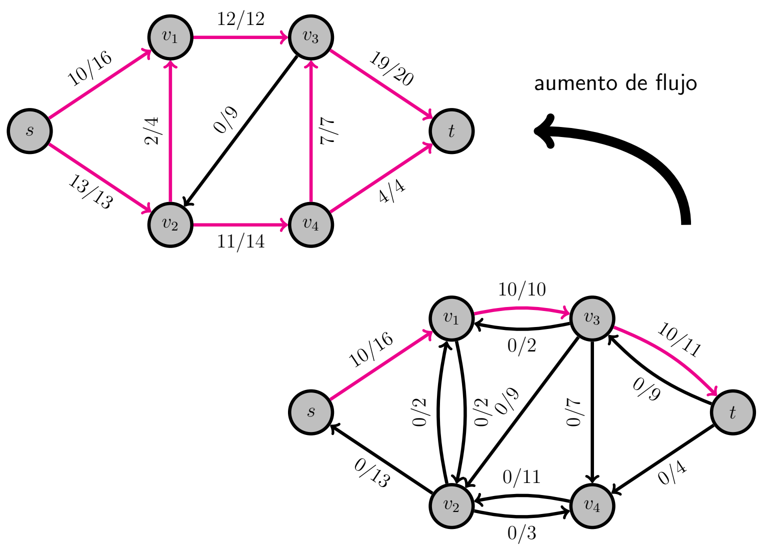
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



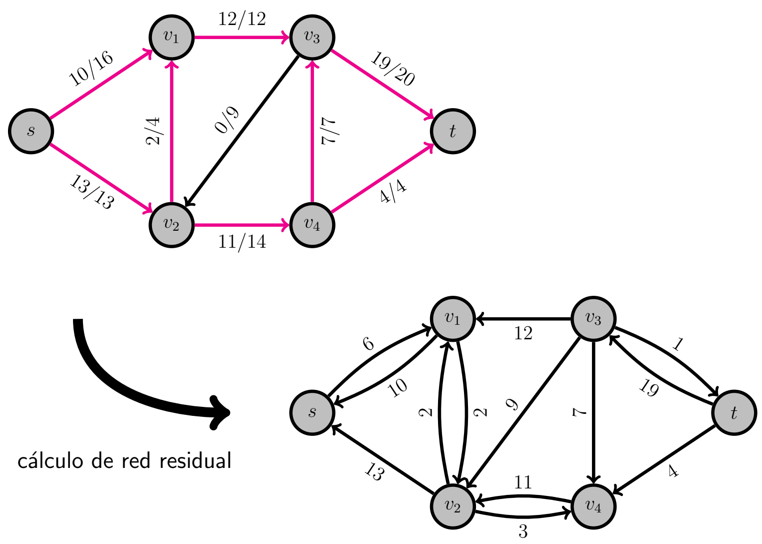
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



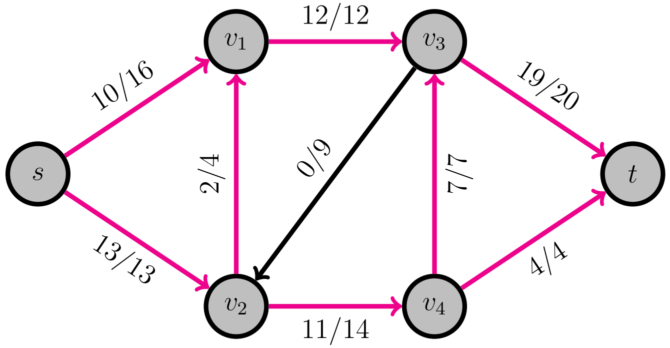
flujo de camino en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



no existe camino de s a t en red residual

Algoritmo de Ford-Fulkerson: Ejemplo



Cortes en redes de flujo

Un corte (S,T) de una red $G=(V,E)$ es una partición de V tal que $s \in S$ y $t \in T$

Si f es un flujo sobre G, el **flujo neto que cruza el corte** es

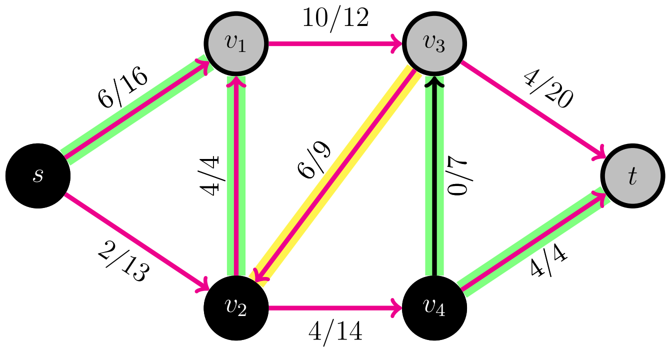
$$f(S,T)^* \;=\; f(S,T) - f(T,S) \;=\; \sum_{u \in S} \sum_{v \in T} f(u,v) - \sum_{v \in T} \sum_{u \in S} f(v,u)$$

La **capacidad del corte** es $c(S,T) = \sum_{u \in S} \sum_{v \in T} c(u,v)$

Un corte (S,T) es **mínimo** si es de mínima capacidad; i.e. $c(S,T) \leq c(S',T')$ para cualquier otro corte (S',T')

(Observar asimetría en las definiciones de flujo neto y capacidad de un corte)

Cortes en redes de flujo: Ejemplo



$(S_1 = \{s\}, T_1 = \{v_1,v_2,v_3,v_4,t\})$, $f(S_1,T_1) = 8$, $c(S_1,T_1) = 29$

$(S_2 = \{s,v_1,v_2\}, T_2 = \{v_3,v_4,t\})$, $f(S_2,T_2) = 8$, $c(S_2,T_2) = 26$

$(S_3 = \{s,v_2,v_4\}, T_3 = \{v_1,v_3,t\})$, $f(S_3,T_3) = 8$, $c(S_3,T_3) = 31$

Relación entre flujos y cortes

Lema

Sea f un flujo sobre una red $G=(V,E)$ con vértices s y t. Sea (S,T) un corte de G. Entonces, el flujo neto a través del corte es igual a $|f|$; i.e. $f(S,T)^* = |f|$

Prueba: para cualquier $u \in V \setminus \{s,t\}$, por conservación de flujo:

$$\sum_v f(u,v) - \sum_v f(v,u) = 0$$

Sumando las igualdades para todos los $u \in S \setminus \{s\}$:

$$|f| = \sum_v f(s,v) - \sum_v f(v,s) + \underbrace{\sum_{u \in S \setminus \{s\}} \left(\sum_v f(u,v) - \sum_v f(v,u) \right)}_{=0}$$

Relación entre flujos y cortes

$$
\begin{aligned}
|f| &= \sum_v f(s,v) - \sum_v f(v,s) + \sum_{u\in S\setminus\{s\}} \Big(\sum_v f(u,v) - \sum_v f(v,u) \Big) \\
&= \sum_v f(s,v) - \sum_v f(v,s) + \sum_v \sum_{u\in S\setminus\{s\}} f(u,v) - \sum_v \sum_{u\in S\setminus\{s\}} f(v,u) \\
&= \sum_v f(s,v) - \sum_v f(v,s) + \sum_v f(S\setminus\{s\},v) - \sum_v f(v,S\setminus\{s\}) \\
&= \sum_v \Big(f(s,v) + f(S\setminus\{s\},v) \Big) - \sum_v \Big(f(v,s) + f(v,S\setminus\{s\}) \Big) \\
&= \sum_v f(S,v) - \sum_v f(v,S) \\
&= f(S,S) + f(S,T) - f(S,S) - f(T,S) \\
&= f(S,T) - f(T,S) \ = \ f(S,T)^*
\end{aligned}
$$

□

El flujo es acotado por la capacidad de los cortes

Corolario

Sea f un flujo sobre una red $G = (V,E)$ con vértices s y t. Sea (S,T) un corte de G. Entonces, $|f| \leq c(S,T)$

Prueba:

$$
\begin{aligned}
|f| &= f(S,T)^* \\
&= f(S,T) - f(T,S) \\
&\leq f(S,T) \\
&\leq c(S,T)
\end{aligned}
$$

□

Teorema de flujo máximo – corte mínimo

Teorema

Sea f un flujo sobre una red $G = (V,E)$ con vértices s y t. Las siguientes condiciones son equivalentes:

1. *f es un flujo máximo sobre G*
2. *No existe camino de aumento en la red residual G_f*
3. *$|f| = c(S,T)$ para algún corte (S,T) de G*

Prueba: Mostraremos la cadena de implicaciones 1 ⇒ 2 ⇒ 3 ⇒ 1

1 ⇒ 2: Si existe un camino de aumento p en G_f, $f \uparrow f_p$ es un flujo de valor mayor a $|f|$. Por lo tanto, f no puede ser de flujo máximo

3 ⇒ 1: Por Corolario, para todo flujo f' y corte (S',T'), $|f'| \leq c(S',T')$. Si $|f| = c(S,T)$ para algún corte (S,T), entonces $|f|$ es máximo

Teorema de flujo máximo – corte mínimo

Teorema

Sea f un flujo sobre una red $G = (V,E)$ con vértices s y t. Las siguientes condiciones son equivalentes:

1. *f es un flujo máximo sobre G*
2. *No existe camino de aumento en la red residual G_f*
3. *$|f| = c(S,T)$ para algún corte (S,T) de G*

Prueba: Mostraremos la cadena de implicaciones 1 ⇒ 2 ⇒ 3 ⇒ 1

2 ⇒ 3: Suponga que no existe camino de s a t en G_f

Defina $S = \{v : \text{existe un camino de } s \text{ a } v \text{ en } G_f\}$ y $T = V \setminus S$.

Claramente $s \in S$, $t \in T$, y (S,T) es un corte de G

Teorema de flujo máximo – corte mínimo

Teorema

Sea f un flujo sobre una red $G = (V, E)$ con vértices s y t. Las siguientes condiciones son equivalentes:

1. *f es un flujo máximo sobre G*
2. *No existe camino de aumento en la red residual G_f*
3. *$|f| = c(S, T)$ para algún corte (S, T) de G*

Prueba: Mostraremos la cadena de implicaciones ❶ ⇒ ❷ ⇒ ❸ ⇒ ❶

❷ ⇒ ❸: Corte $(S = \{v : \text{existe un camino de } s \text{ a } v \text{ en } G_f\}, T = V \setminus S)$

Considere vértices $u \in S$ y $v \in T$:

– Si $(u, v) \in E$, $f(u, v) = c(u, v)$ (sino $(u, v) \in E_f$ y $v \in S$)

– Si $(v, u) \in E$, $f(v, u) = 0$ (sino $c_f(u, v) > 0$, $(u, v) \in E_f$ y $v \in S$)

– Si $(u, v) \notin E$ y $(v, u) \notin E$, $f(u, v) = f(v, u) = 0$

Teorema de flujo máximo – corte mínimo

Teorema

Sea f un flujo sobre una red $G = (V, E)$ con vértices s y t. Las siguientes condiciones son equivalentes:

1. *f es un flujo máximo sobre G*
2. *No existe camino de aumento en la red residual G_f*
3. *$|f| = c(S, T)$ para algún corte (S, T) de G*

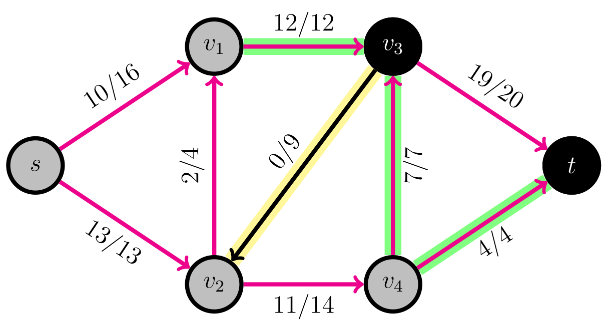
Prueba: Mostraremos la cadena de implicaciones ❶ ⇒ ❷ ⇒ ❸ ⇒ ❶

❷ ⇒ ❸:

$$\begin{aligned} |f| = f(S,T)^* &= f(S,T) - f(T,S) \\ &= \textstyle\sum_{u \in S} \sum_{v \in T} f(u, v) - \sum_{v \in T} \sum_{u \in S} f(v, u) \\ &= \textstyle\sum_{u \in S} \sum_{v \in T} c(u, v) - \sum_{v \in T} \sum_{u \in S} 0 \\ &= c(S, T) \end{aligned}$$

□

Teorema de flujo máximo – corte mínimo: Ejemplo



$(S = \{s, v_1, v_2, v_4\}, T = \{v_3, t\})$, $f(S, T) = 23$, $c(S, T) = 23$

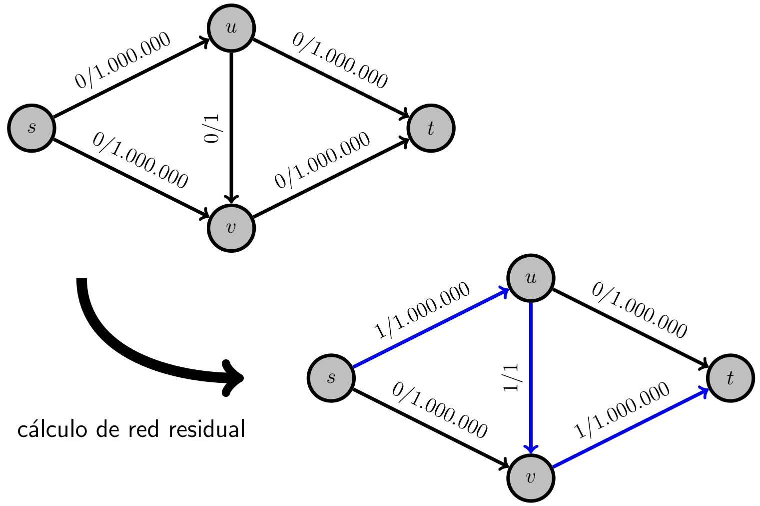
Teorema de flujo máximo – corte mínimo

Corolario

Si el algoritmo de Ford-Fulkerson **termina***, el flujo retornado es de valor máximo*

Prueba: El algoritmo de Ford-Fulkerson termina al no haber camino de aumento en la red residual G_f. Por el Teorema, f es de valor máximo □

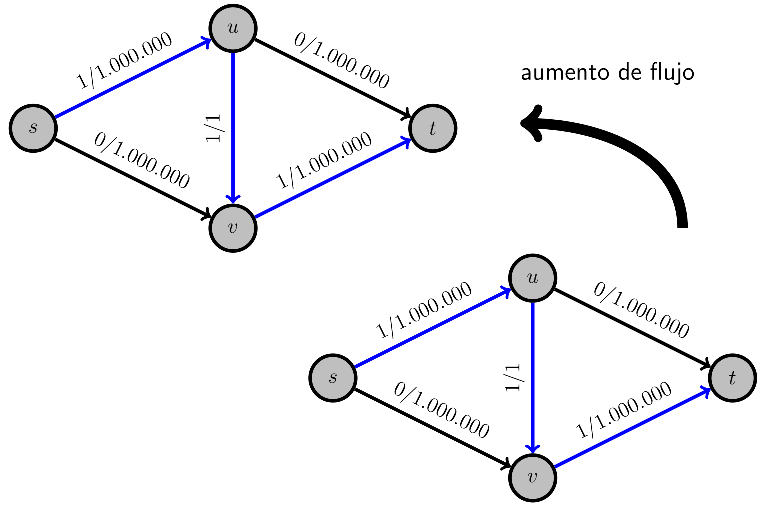
Algoritmo de Ford-Fulkerson: Patología I



cálculo de red residual

flujo de camino en red residual

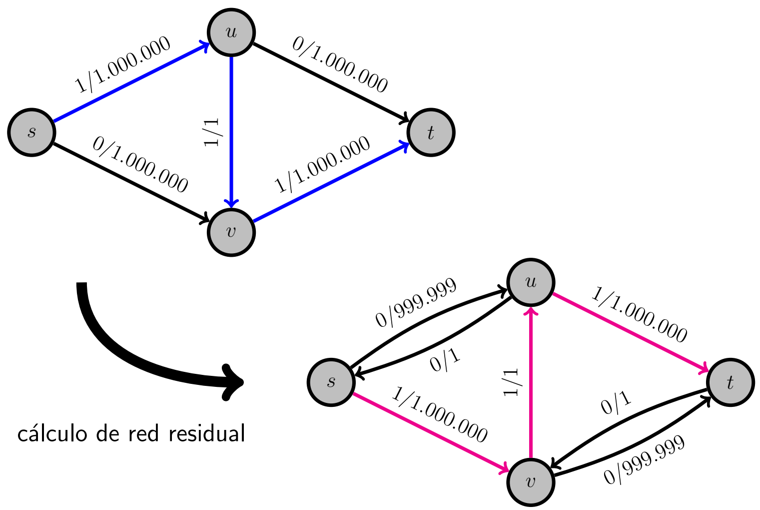
Algoritmo de Ford-Fulkerson: Patología I



aumento de flujo

flujo de camino en red residual

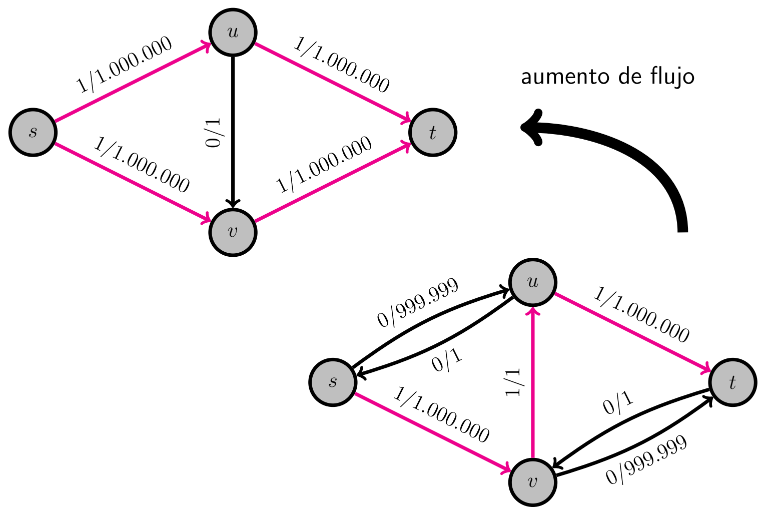
Algoritmo de Ford-Fulkerson: Patología I



cálculo de red residual

flujo de camino en red residual

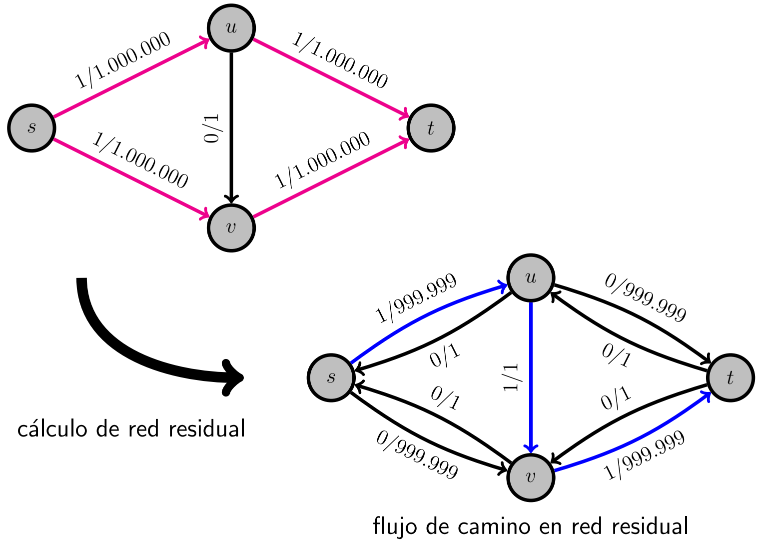
Algoritmo de Ford-Fulkerson: Patología I



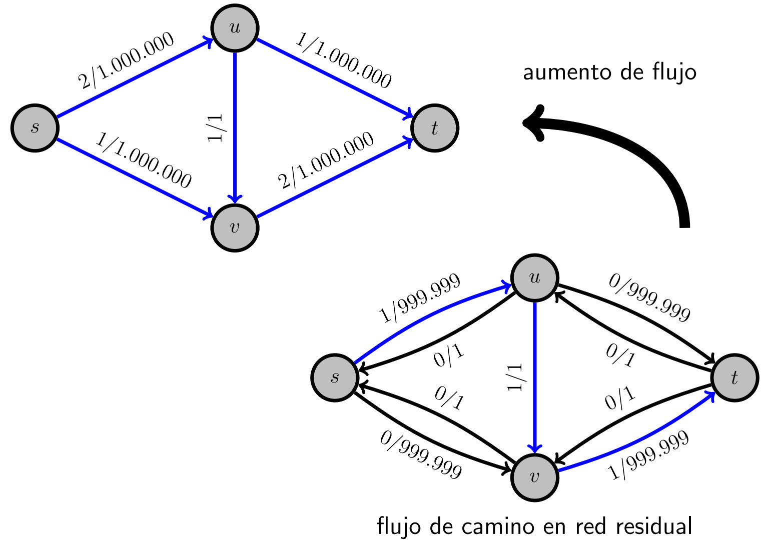
aumento de flujo

flujo de camino en red residual

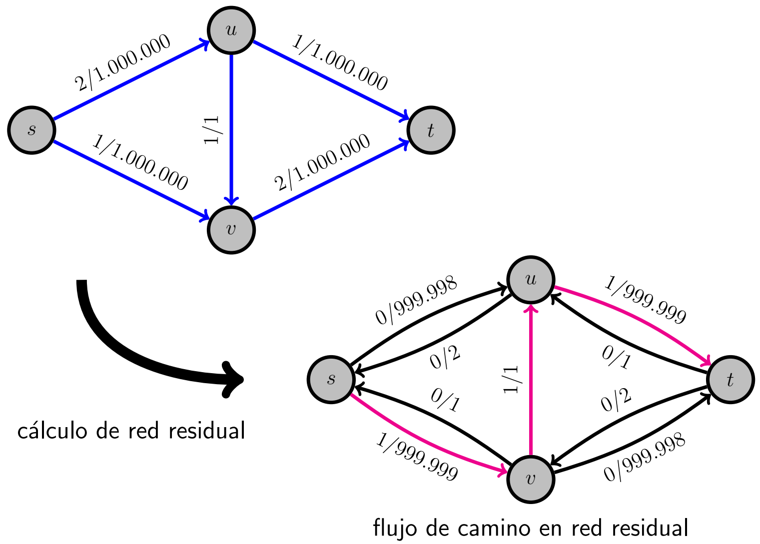
Algoritmo de Ford-Fulkerson: Patología I



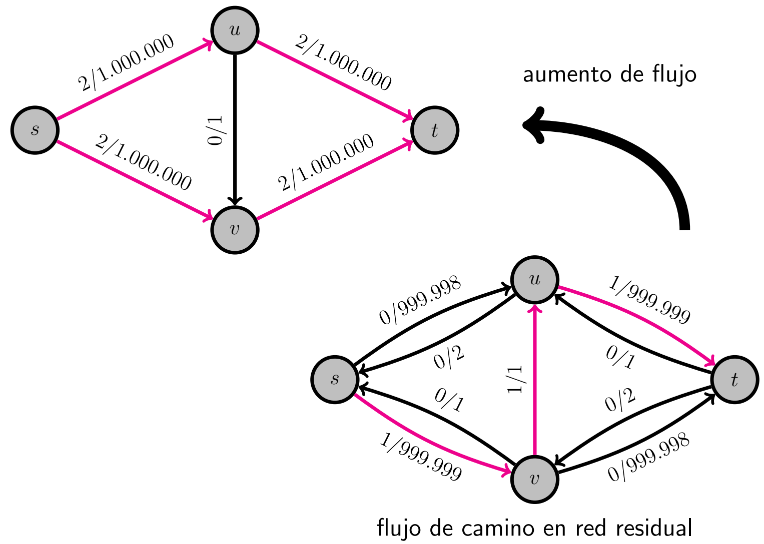
Algoritmo de Ford-Fulkerson: Patología I



Algoritmo de Ford-Fulkerson: Patología I



Algoritmo de Ford-Fulkerson: Patología I



Algoritmo de Ford-Fulkerson: Análisis

Suposición: todas las capacidades son integrales

1. El cálculo de un camino de s a t en la red residual $G_f = (V, E_f)$ toma tiempo $O(V + E_f) = O(E)$ ya que $|E_f| \leq 2|E|$
2. En cada iteración, las capacidades residuales y flujos son integrales
3. En el peor caso, el flujo aumenta 1 en cada iteración
4. Por lo tanto, Ford-Fulkerson toma tiempo $O(E \times |f^*|)$ donde f^* es flujo óptimo (i.e. **tiempo exponencial** en el tamaño de la entrada)

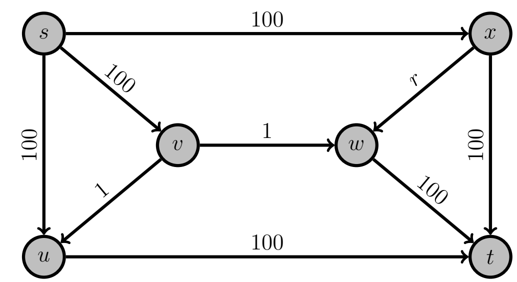
Si las capacidades son racionales, se pueden multiplicar por una constante para convertirlas en integrales

Algoritmo de Ford-Fulkerson: Patología II

¿Qué sucede si las capacidades son irracionales?

Ford-Fulkerson puede **no terminar** al quedar atrapado en un lazo

Algoritmo de Ford-Fulkerson: Patología II



donde r es tal que $r^2 = 1 - r \implies r = \frac{\sqrt{5}-1}{2}$

Es fácil ver que $|f^*| = 201$

Sin embargo, existe una secuencia de caminos de aumento tal que el flujo se incrementa de forma:

$$1 + 2r + 2r^2 + 2r^3 + \cdots \;\to\; 1 + \textstyle\sum_{k \geq 1} 2r^k \;=\; 3 + 2r \;<\; 5$$

Algoritmo de Edmonds-Karp

El algoritmo de Edmonds-Karp es una instancia del método de Ford-Fulkerson en donde el camino de aumento es un **camino más corto** entre s y t

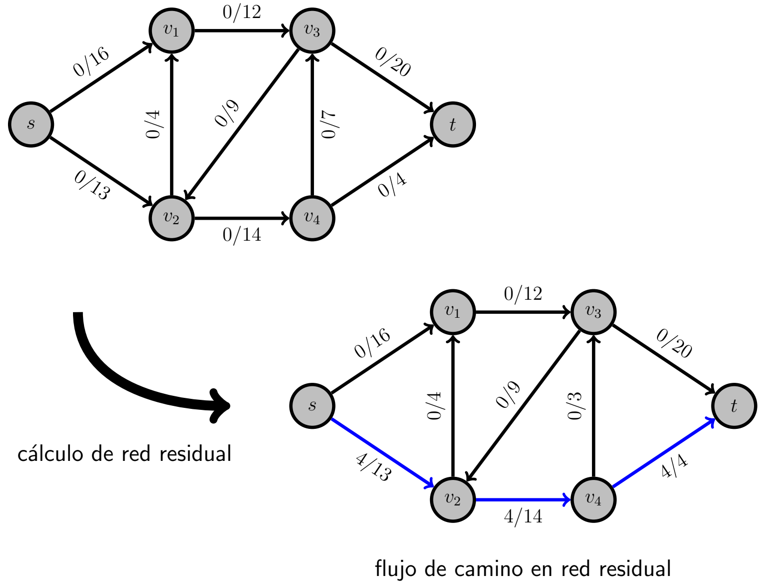
Edmonds-Karp encuente un camino de aumento en la red residual G_f utilizando **búsqueda en amplitud** (BFS)

Esta simple modificación garantiza encontrar un flujo máximo en tiempo $O(VE^2)$

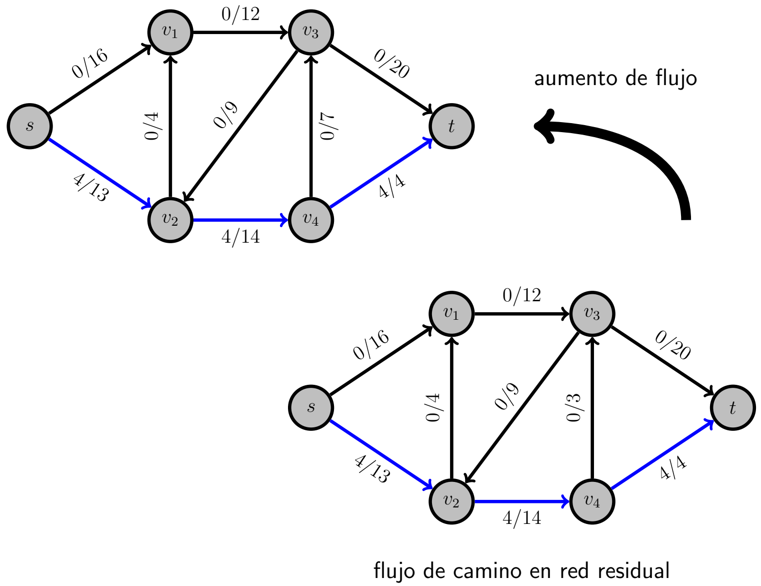
Algoritmo de Edmonds-Karp: Pseudocódigo

```
Flujo Edmonds-Karp(G, s, t):
    % inicializar flujo inicial nulo
    foreach (u,v) ∈ E
        f(u,v) = 0

    % incrementar flujo de forma iterativa
    while exista un camino p de s a t en la red residual G_f
        Calcular con BFS un camino más corto p de s a t en G_f
        c(p) = min { c_f(u,v) : (u,v) ∈ p }
        foreach (u,v) ∈ p
            if (u,v) ∈ E
                f(u,v) = f(u,v) + c(p)
            else
                f(u,v) = f(u,v) - c(p)

    % retornar flujo encontrado
    return f
```

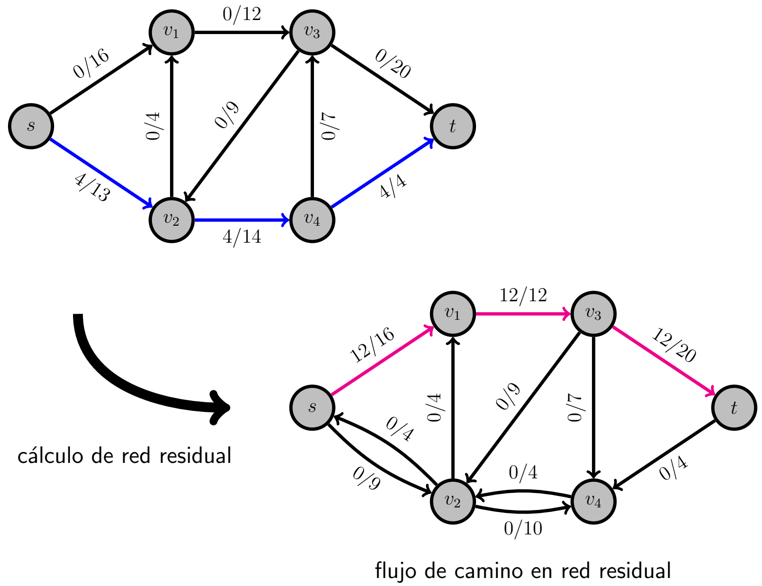

Algoritmo de Edmonds-Karp: Ejemplo



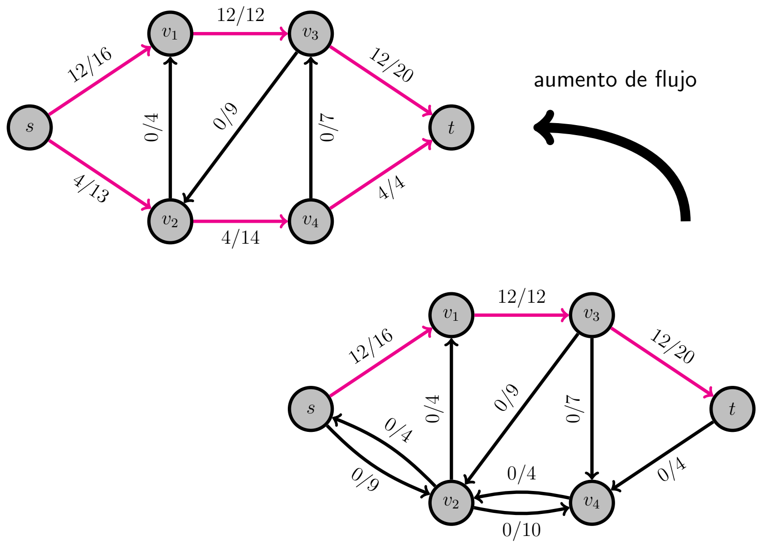
Algoritmo de Edmonds-Karp: Ejemplo



Algoritmo de Edmonds-Karp: Ejemplo

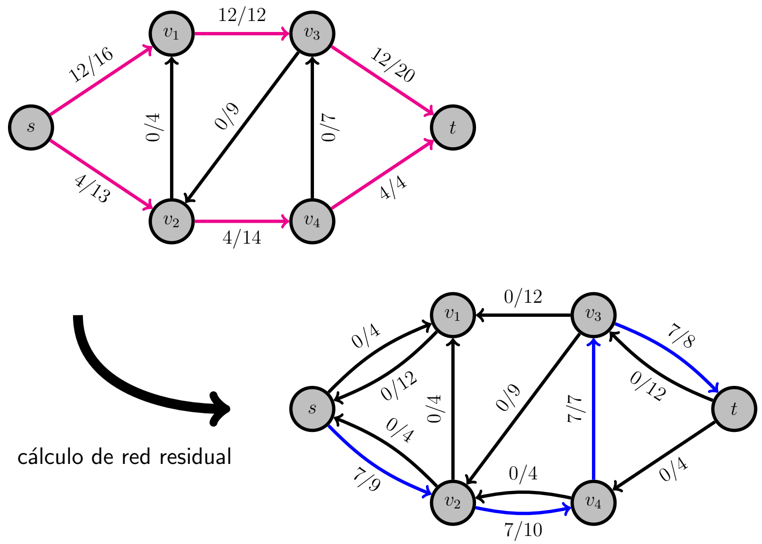


Algoritmo de Edmonds-Karp: Ejemplo



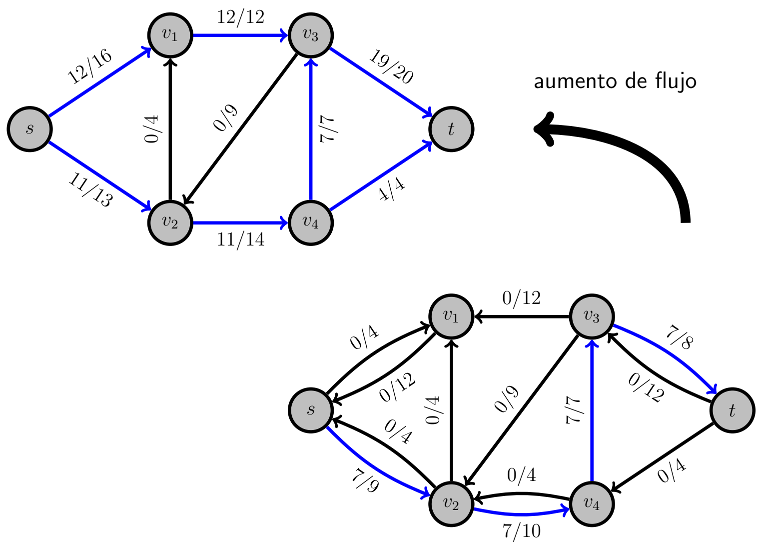
flujo de camino en red residual

Algoritmo de Edmonds-Karp: Ejemplo



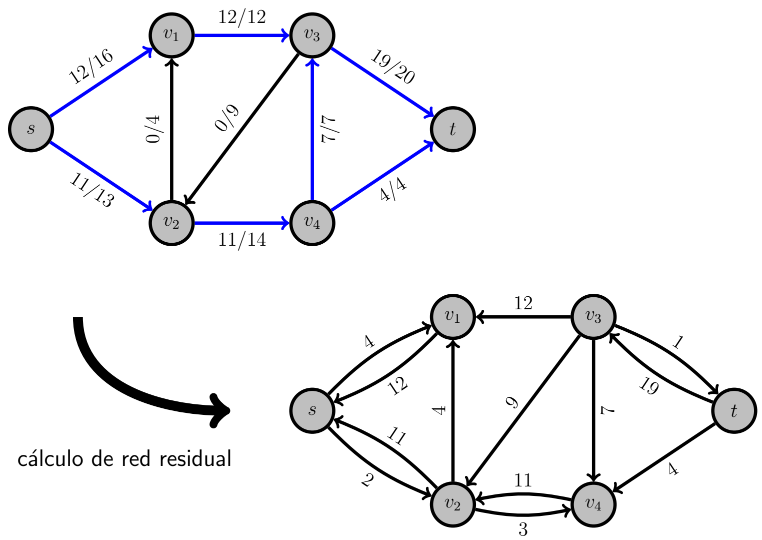
flujo de camino en red residual

Algoritmo de Edmonds-Karp: Ejemplo



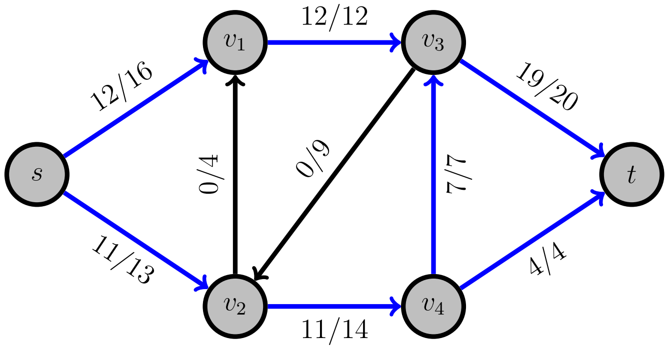
flujo de camino en red residual

Algoritmo de Edmonds-Karp: Ejemplo



no existe camino de s a t en red residual

Algoritmo de Edmonds-Karp: Ejemplo



Algoritmo de Edmonds-Karp: Análisis

Notación: $\delta_f(u,v)$ es la distancia de u a v en la red residual G_f

Lema

Considere la ejecución del algoritmo de Edmonds-Karp sobre la red $G = (V, E, s, t)$. Entonces, para todo vértice $v \in V \setminus \{s,t\}$, $\delta_f(s,v)$ se incrementa monotónicamente con cada aumento de flujo

Prueba: supondremos que para algún aumento de flujo la distancia $\delta_f(s,v)$ decrece para algún vértice v, y llegaremos a una contradicción

Algoritmo de Edmonds-Karp: Análisis

Prueba del Lema:

– Sea f el flujo justo antes del **primer aumento que decrece la distancia de algún vértice** y sea f' el flujo resultante de dicho aumento

– Sea $v \in V$ tal que la distancia decrece y $\delta_{f'}(s,v)$ es mínimo

– Sea $p = s \rightsquigarrow u \to v$ un camino más corto en $G_{f'}$: $\delta_{f'}(s,v) = \delta_{f'}(s,u) + 1$

Por definición, $\delta_f(s,v) > \delta_{f'}(s,v)$ y $\delta_f(s,u) \leq \delta_{f'}(s,u)$ por elección de v

• Veamos que $(u,v) \notin E_f$. Suponga que $(u,v) \in E_f$:

$$\begin{aligned} \delta_f(s,v) &\leq \delta_f(s,u) + 1 && \text{(por desigualdad triangular)} \\ &\leq \delta_{f'}(s,u) + 1 && \text{(por desigualdad de arriba)} \\ &= \delta_{f'}(s,v) && \text{(por optimalidad de } s \rightsquigarrow u \to v \text{ en } G_{f'}\text{)} \end{aligned}$$

Esto contradice $\delta_f(s,v) > \delta_{f'}(s,v)$. Por lo tanto, $(u,v) \notin E_f$

Algoritmo de Edmonds-Karp: Análisis

Prueba del Lema:

– Sea f el flujo justo antes del **primer aumento que decrece la distancia de algún vértice** y sea f' el flujo resultante de dicho aumento

– Sea $v \in V$ tal que la distancia decrece y $\delta_{f'}(s,v)$ es mínimo

– Sea $p = s \rightsquigarrow u \to v$ un camino más corto en $G_{f'}$: $\delta_{f'}(s,v) = \delta_{f'}(s,u) + 1$

Por definición, $\delta_f(s,v) > \delta_{f'}(s,v)$ y $\delta_f(s,u) \leq \delta_{f'}(s,u)$ por elección de v

• Tenemos que el flujo cambia de f a f', $(u,v) \notin E_f$ y $(u,v) \in E_{f'}$

Razonamos por casos: $(u,v) \in E$ y $(u,v) \notin E$

1 Caso $(u,v) \in E$. Tenemos $0 = c_f(u,v) = c(u,v) - f(u,v)$ y $0 < c_{f'}(u,v) = c(u,v) - f'(u,v)$. Entonces, $f'(u,v) < f(u,v)$.

Como $f'(u,v) = f(u,v) + f_q(u,v) - f_q(v,u)$ donde q es camino de aumento en G_f, la arista (v,u) debe pertenecer a q. Es decir, **(v,u) pertenece a un camino más corto de s a t en G_f**

Algoritmo de Edmonds-Karp: Análisis

Prueba del Lema:

– Sea f el flujo justo antes del **primer aumento que decrece la distancia de algún vértice** y sea f' el flujo resultante de dicho aumento

– Sea $v \in V$ tal que la distancia decrece y $\delta_{f'}(s,v)$ es mínimo

– Sea $p = s \rightsquigarrow u \rightarrow v$ un camino más corto en $G_{f'}$: $\delta_{f'}(s,v) = \delta_{f'}(s,u) + 1$

Por definición, $\delta_f(s,v) > \delta_{f'}(s,v)$ y $\delta_f(s,u) \leq \delta_{f'}(s,u)$ por elección de v

• Tenemos que el flujo cambia de f a f', $(u,v) \notin E_f$ y $(u,v) \in E_{f'}$

Razonamos por casos: $(u,v) \in E$ y $(u,v) \notin E$

2 Caso $(u,v) \notin E$. Tenemos $0 = c_f(u,v) = f(v,u)$ y $0 < c_{f'}(u,v) = f'(v,u)$. Entonces, $f'(v,u) > f(v,u)$.

Como $f'(v,u) = f(v,u) + f_q(v,u) - f_q(u,v)$ donde q es camino de aumento en G_f, la arista (v,u) debe pertenecer a q. Es decir, **(v,u) pertenece a un camino más corto de s a t en G_f**

Algoritmo de Edmonds-Karp: Análisis

Prueba del Lema:

– Sea f el flujo justo antes del **primer aumento que decrece la distancia de algún vértice** y sea f' el flujo resultante de dicho aumento

– Sea $v \in V$ tal que la distancia decrece y $\delta_{f'}(s,v)$ es mínimo

– Sea $p = s \rightsquigarrow u \rightarrow v$ un camino más corto en $G_{f'}$: $\delta_{f'}(s,v) = \delta_{f'}(s,u) + 1$

Por definición, $\delta_f(s,v) > \delta_{f'}(s,v)$ y $\delta_f(s,u) \leq \delta_{f'}(s,u)$ por elección de v

• Tenemos que el flujo cambia de f a f', $(u,v) \notin E_f$ y $(u,v) \in E_{f'}$

• (v,u) pertenece a un camino más corto de s a t en G_f

$$\begin{aligned}
\delta_f(s,v) &= \delta_f(s,u) - 1 && \text{(porque } (v,u) \text{ pertenece a camino óptimo)} \\
&\leq \delta_{f'}(s,u) - 1 && \text{(por desigualdad de arriba)} \\
&= \delta_{f'}(s,v) - 2 && \text{(por optimalidad de } s \rightsquigarrow u \rightarrow v\text{)}
\end{aligned}$$

Contradicción con $\delta_f(s,v) > \delta_{f'}(s,v)$. Entonces, no existe tal vértice v □

Algoritmo de Edmonds-Karp: Análisis

Teorema

El número de aumentos de flujo que realiza el algoritmo de Edmonds-Karp en una red $G = (V,E)$ es $O(VE)$. Por lo tanto, Edmonds-Karp finaliza en tiempo $O(VE^2)$

Algoritmo de Edmonds-Karp: Análisis

Prueba del Teorema:

– Una arista (u,v) en G_f es **crítica** para el camino p si $c_f(p) = c_f(u,v)$

– Todo camino de aumento tiene al menos un arista crítica

– Si el flujo f se aumenta con p y (u,v) es crítica para p, entonces (u,v) desaparece de la nueva red residual

Mostraremos que cada arista puede ser crítica a lo sumo $|V|/2$ veces

• Considere arista $(u,v) \in E_f$. Como los caminos de aumento son caminos más cortos, si (u,v) es crítica en G_f: $\delta_f(s,v) = \delta_f(s,u) + 1$

Una vez que se hace crítica, (u,v) desaparece de la nueva red residual. Para que vuelva a ser crítica, el flujo sobre (u,v) debe decrecer. Esto sólo pasa si (v,u) aparece en un camino de aumento p

Algoritmo de Edmonds-Karp: Análisis

Prueba del Teorema:

– Una arista (u, v) en G_f es **crítica** para el camino p si $c_f(p) = c_f(u, v)$

– Todo camino de aumento tiene al menos un arista crítica

– Si el flujo f se aumenta con p y (u, v) es crítica para p, entonces (u, v) desaparece de la nueva red residual

Mostraremos que cada arista puede ser crítica a lo sumo $|V|/2$ veces

• Sea f' el flujo con p camino de aumento en $G_{f'}$ y (v, u) crítica para p. Tenemos $\delta_{f'}(s, u) = \delta_{f'}(s, v) + 1$

$$\begin{aligned} \delta_{f'}(s,u) &= \delta_{f'}(s,v)+1 && ((v,u) \in \text{camino más corto en } G_{f'}) \\ &\geq \delta_f(s,v)+1 && (\text{por el Lema}) \\ &= \delta_f(s,u)+2 && ((u,v) \in \text{camino más corto en } G_f) \end{aligned}$$

Cada vez que (u, v) se "rehace" crítica, la distancia de s a u aumenta en 2. Como la distancia $\leq |V|$, máximo #veces que (u, v) es crítica es $\leq |V|/2$

Algoritmo de Edmonds-Karp: Análisis

Prueba del Teorema:

– Una arista (u, v) en G_f es **crítica** para el camino p si $c_f(p) = c_f(u, v)$

– Todo camino de aumento tiene al menos un arista crítica

– Si el flujo f se aumenta con p y (u, v) es crítica para p, entonces (u, v) desaparece de la nueva red residual

Mostraremos que cada arista puede ser crítica a lo sumo $|V|/2$ veces

• Existen $O(E)$ aristas que pueden aparecer en una red residual

• Cada una de dichas aristas puede ser crítica $O(V)$ veces

• Cada camino de aumento tiene al menos una arista crítica

• Entonces, pueden existir a lo sumo $O(VE)$ caminos de aumento

Algoritmo de Edmonds-Karp: Análisis

Prueba del Teorema:

– Una arista (u, v) en G_f es **crítica** para el camino p si $c_f(p) = c_f(u, v)$

– Todo camino de aumento tiene al menos un arista crítica

– Si el flujo f se aumenta con p y (u, v) es crítica para p, entonces (u, v) desaparece de la nueva red residual

Mostraremos que cada arista puede ser crítica a lo sumo $|V|/2$ veces

Análisis final de Edmonds-Karp:

- Existen $O(VE)$ iteraciones (número máximo de caminos de aumento)
- En cada iteración se debe encontrar un camino más corto en G_f
- Como $|V| = O(E)$ y $|E_f| = O(E)$, BFS tarda tiempo $O(V + E_f) = O(E)$
- El tiempo total para Edmonds-Karp es $O(VE^2)$

$\square$