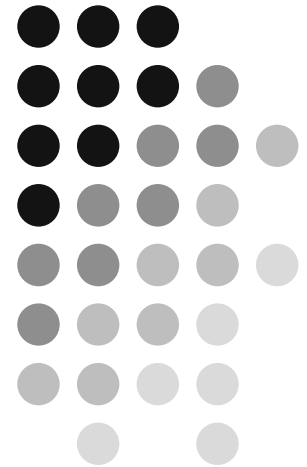


Grafos: flujo máximo

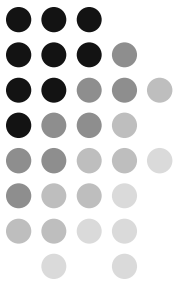


UNIVERSIDAD
DE LOS ANDES

Diseño y Análisis de Algoritmos
Cátedra de Programación
Carrera de Ingeniería de Sistemas
Prof. Isabel Besembel Carrera

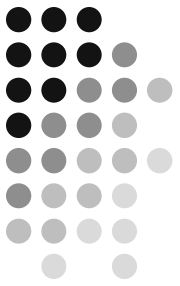


Definición del problema



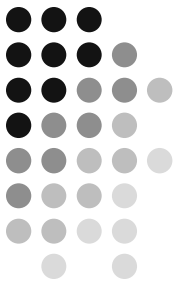
- El flujo de algún material a través de una red es la tasa de movimiento del material de un punto de la red a otro.
- **Aplicaciones:**
 - ❖ Modelado de líquidos fluyendo a través de tuberías,
 - ❖ piezas a través de líneas de ensamblaje,
 - ❖ corriente a través de redes eléctricas,
 - ❖ información a través de redes de comunicación, etc.
- **Problema del flujo máximo:** encontrar la rata máxima a la cual el material puede ser transportado de una fuente (s) a un destino (t), sin violar las restricciones de capacidad de la red.

Red de flujo



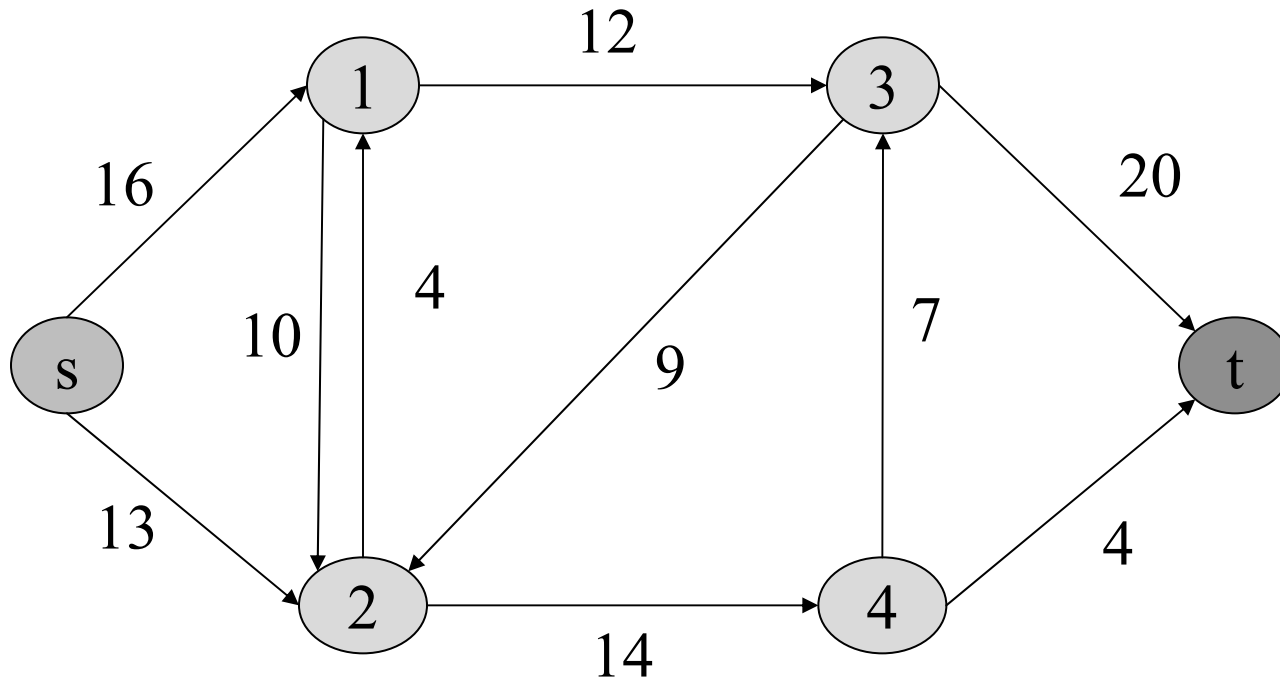
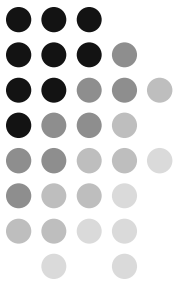
- Una red de flujo $G = (\mathbf{N}, A)$ es un digrafo etiquetado cuyos pesos se denominan capacidad (c) y son siempre positivos, $c(u, v) \geq 0$.
- Si $(u, v) \notin A \Rightarrow c(u, v) = 0$
- Por cada nodo $v \in \mathbf{N}$ hay un camino de s a t que pasa por v
 $\Rightarrow G$ es conexo ($|A| \geq |\mathbf{N}| - 1$)
- Un flujo en G es una función real $f : \mathbf{N} \times \mathbf{N} \rightarrow \mathfrak{R}$ que satisface las propiedades siguientes:
 - ❖ **Restricción de capacidad:** $\forall u, v \in \mathbf{N}$ se requiere que $f(u, v) \leq c(u, v)$
 - ❖ **Simetría oblicua:** $\forall u, v \in \mathbf{N}$ se requiere que $f(u, v) = -f(v, u) \Rightarrow f(u, u) = 0$

Red de flujo

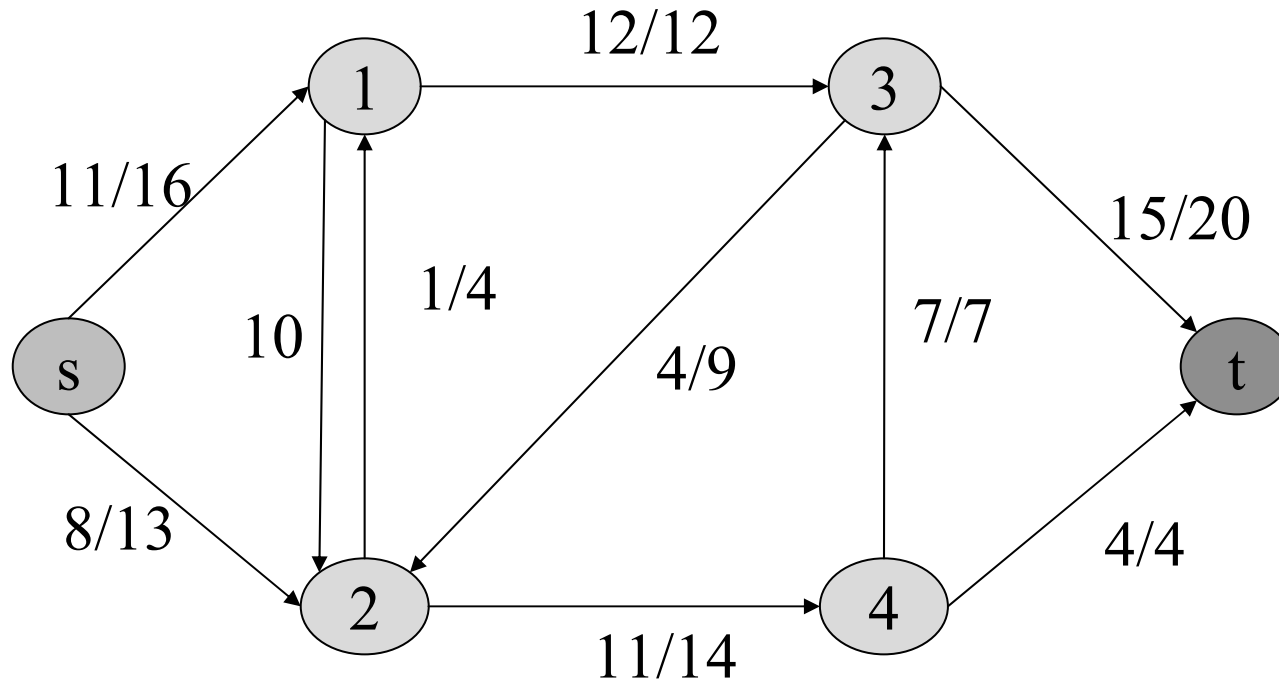
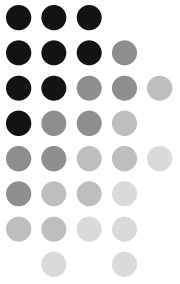


- ❖ **Conservación de flujo:** $\forall u \in N - \{s, t\}$ se requiere que $\sum_{v \in N} f(u, v) = 0$ denominado el flujo neto desde u hasta v .
- $f(u, v)$ puede ser positivo o negativo, pero el flujo neto total dentro de un nodo debe ser cero.
- El flujo neto total desde s es $FNT = \sum_{v \in N} f(s, v)$
- **El problema del flujo máximo se define como:** dado G , s (fuente) y t (destino) encontrar el flujo neto total máximo desde s hasta t
- Flujo neto positivo entrando a un nodo v es $\sum_{u \in N, f(u, v) > 0} f(u, v)$ y el que sale se define simétricamente.

Ejemplo red de flujo

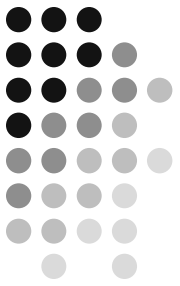


Ejemplo red de flujo



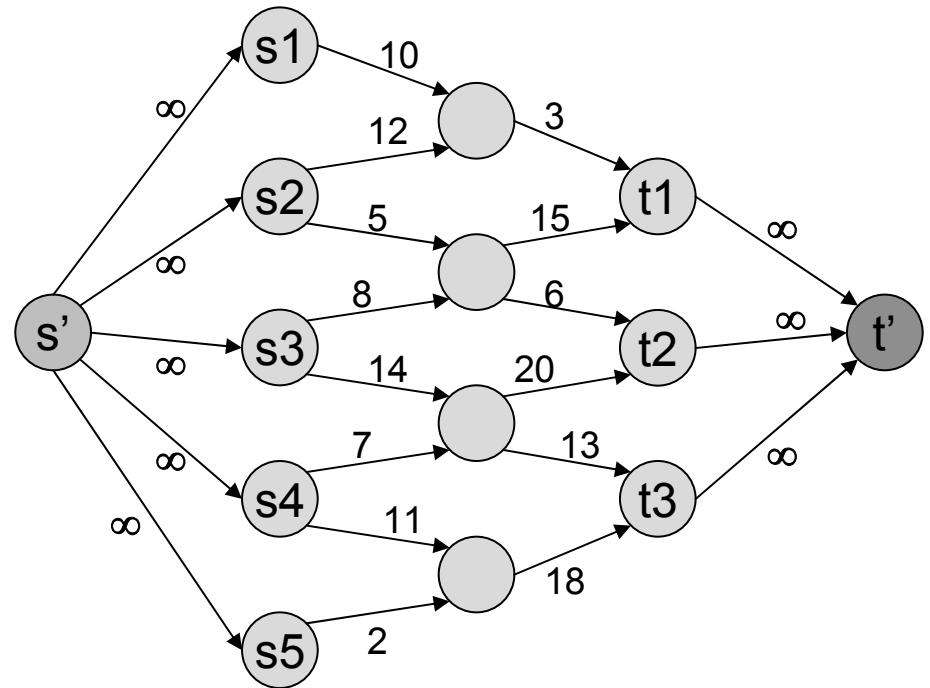
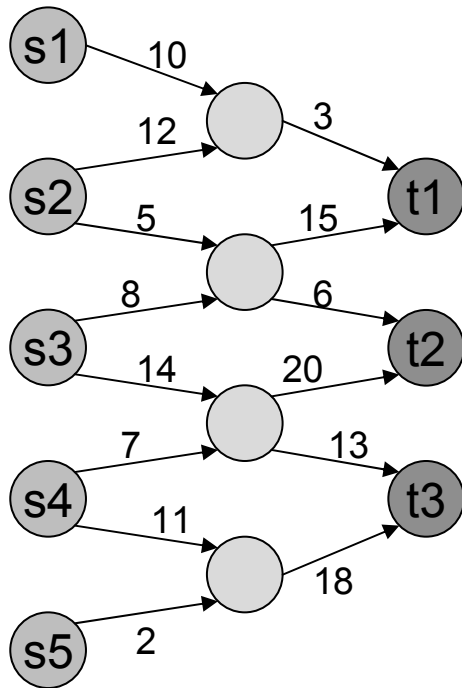
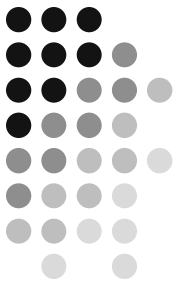
$f(u, v)/c(u, v)$

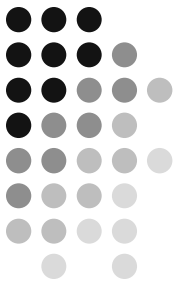
Red de flujo



- Por la propiedad de conservación de flujo se tiene que el flujo neto positivo entrando a un nodo diferente de s o t debe ser igual al flujo neto positivo dejando el nodo
- Redes con varios nodos fuente y varios nodos destino:
 - ❖ se reduce al anterior colocando dos nodos ficticios denominados:
 - el nodo superfuente s_y
 - el nodo superdestino t_j
 - ❖ ambos nodos se conectan a los otros nodos con arcos de capacidad ∞ .
 - ❖ Anexar
 - un arco (s, s_i) por cada nodo fuente de G con $i = 1, 2, \dots, m$ y
 - un arco (t_j, t) por cada nodo destino de G con $j = 1, 2, \dots, n$

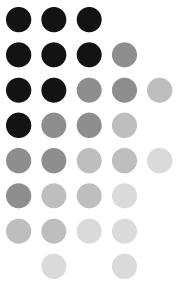
Redes con varias fuentes y varios destinos





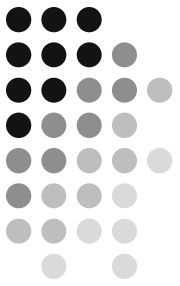
- **Lema 19.1:** Sea $G = (N, A)$ una red de flujo y f el flujo de G , entonces:
- ❖ Para $X \subseteq N$, $f(X, X) = 0$.
 - ❖ Para $X, Y \subseteq N$, $f(X, Y) = -f(Y, X)$.
 - ❖ Para $X, Y, Z \subseteq N$ con $X \cap Y = \emptyset$,
 $f(X \cup Y, Z) = f(X, Z) + f(Y, Z)$ y
 $f(Z, X \cup Y) = f(Z, X) + f(Z, Y)$

Método de Ford-Fulkerson



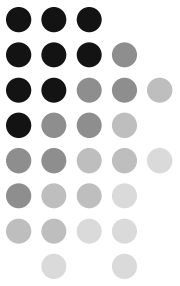
- Es un método iterativo que inicia $f(u, v) = 0$ para todos los nodos de G , en cada iteración el flujo se aumenta por medio del alargamiento del camino desde s hasta t , se repite el proceso hasta que no hay alargamiento posible.
- **Algoritmo general:** métodoFordFulkerson(G, s, t)
 - 1) iniciar f en cero
 - 2) mientras exista un alargamiento del camino p
aumente el flujo f a lo largo de p
 - 3) regrese f

Red residual



- Dados una red de flujo y un flujo, la red residual consiste de los arcos que pueden admitir mas flujo neto.
- La cantidad adicional de flujo neto que se puede enviar desde u hasta v antes de exceder $c(u, v)$ es la capacidad residual de (u, v) , $cf(u, v) = c(u, v) - f(u, v)$
 - ❖ Si $c(u, v) = 16$ y $f(u, v) = 11$, entonces se pueden enviar $cf(u, v) = 5$ unidades adicionales antes de exceder la capacidad del arco
- Dada una red de flujo $G = (\mathbb{N}, A)$ y un flujo f , la red residual de G inducida por f es $G_f = (\mathbb{N}, A_f)$, donde $A_f = \{(u, v) \in \mathbb{N} \times \mathbb{N} : cf(u, v) > 0\}$

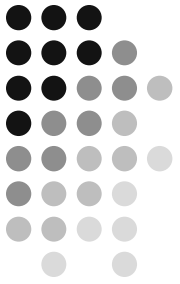
Red residual



- Un arco (u, v) aparece en G_f solo si $(u, v) \in A$ y hay un flujo neto positivo de u a v

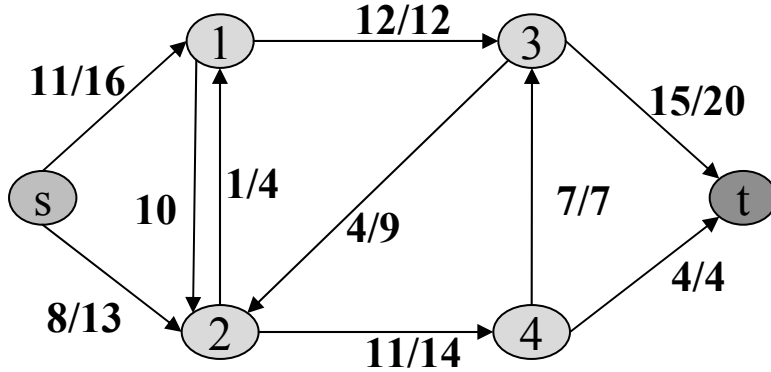
$$|A_f| \leq 2|A|$$

- **Lema 19.2:** Sea $G=(N, A)$ una red de flujo con s, t y f . Sea G_f la red residual de G inducida por f y sea f' el flujo de G_f . Entonces, el flujo $f + f'$ definido por $(f + f')(u, v) = f(u, v) + f'(u, v)$ es un flujo en G cuyo valor es $|f + f'| = |f| + |f'|$

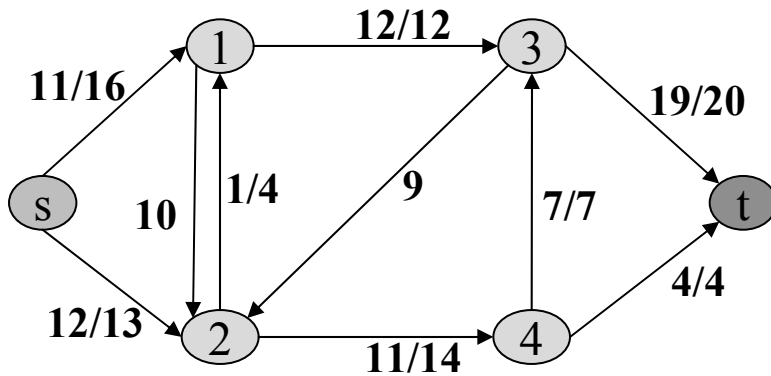
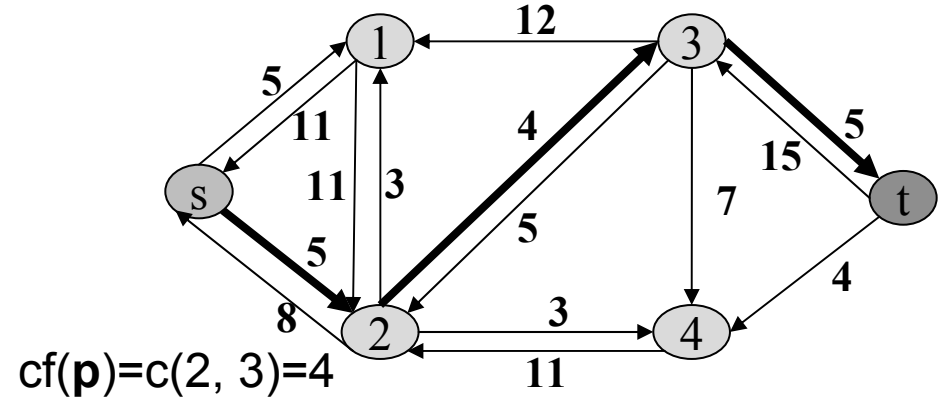


Red residual

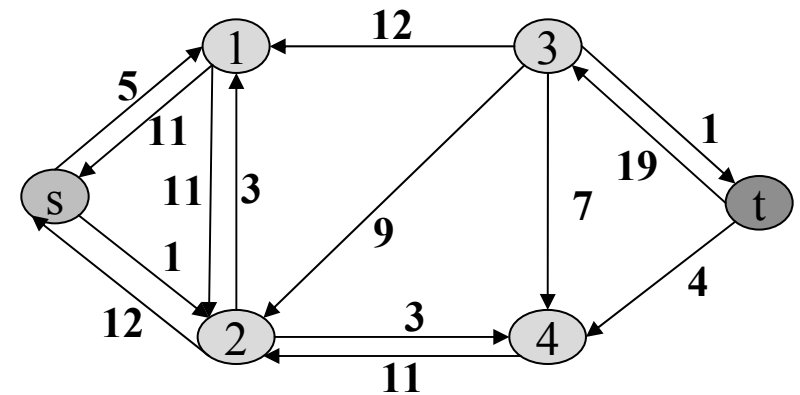
G con flujo f



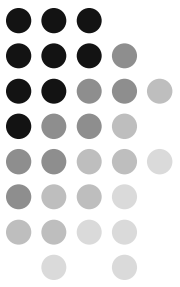
Gf con camino aumentado p



El flujo en G que resulta de aumentar en p su capacidad residual de 4



Red residual inducida por el flujo



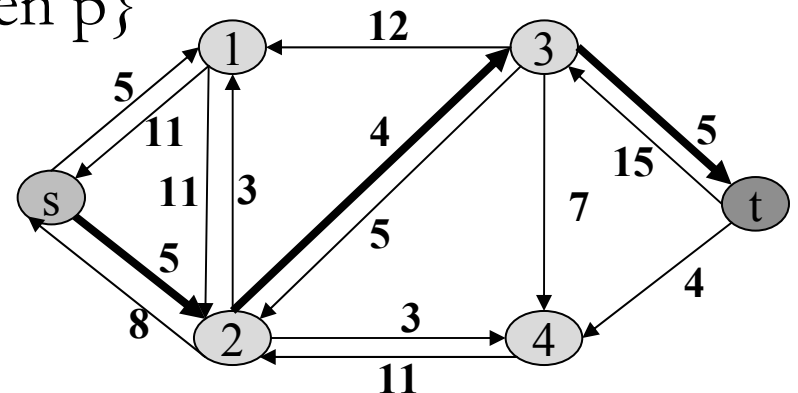
Camino aumentado

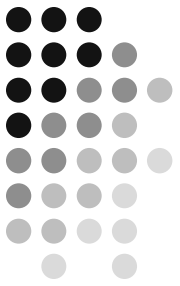
- Un camino aumentado p es un camino simple desde s a t en la red residual G_f .
- La capacidad residual de p es la máxima cantidad de flujo neto que se puede transportar a lo largo de los arcos de p y se define como:

$$cf(p) = \min \{ cf(u, v) : (u, v) \text{ está en } p \}$$

G_f con camino aumentado p

$$cf(p) = c(2, 3) = 4$$





Camino aumentado

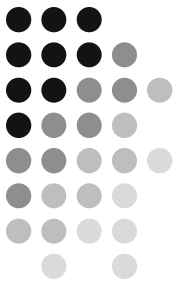
- **Lema 19.3:** Sea $G = (\mathbb{N}, A)$ una red de flujo con flujo f y sea p el camino aumentado de Gf , la función $f_p : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{R}$ se define como:

$$f_p(u, v) = \begin{cases} c_f(p) & \text{si } (u, v) \text{ está en } p, \\ -c_f(p) & \text{si } (v, u) \text{ está en } p, \\ 0 & \text{sino} \end{cases}$$

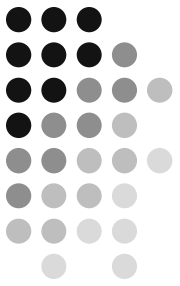
Donde f_p es el flujo en Gf con $|f_p| = c_f(p) > 0$

- Un flujo en G es $f' = f + f_p$ y cuyo valor en G es $|f'| = |f| + |f_p| > f$

Cortes en la red de flujo

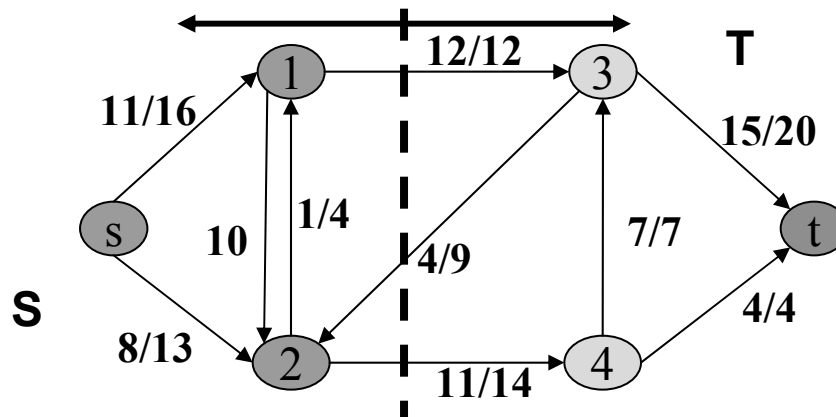


- Un corte (S, T) en una red de flujo $G = (N, A)$ es una partición de N en S y $T = N - S$ tal que $s \in S$ y $t \in T$
- Si f es un flujo, entonces el flujo neto a través del corte (S, T) está definido por $f(S, T)$ y su capacidad es $c(S, T)$.
- El flujo neto a través del corte (S, T) es $f(S, T) = |f|$
- El valor de cualquier flujo f en una red de flujos G está limitado superiormente por la capacidad de cualquier corte de G .



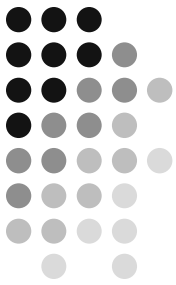
Cortes en la red de flujo

- **Teorema 1:** Si f es un flujo en la red de flujo $G = (N, A)$ con fuente s y destino t , entonces las siguientes condiciones son equivalentes:
 - ❖ f es un flujo máximo de G
 - ❖ La red residual G_f no contiene caminos aumentados
 - ❖ $|f| = c(S, T)$ para algún corte (S, T) de G .



Corte (S, T)
 $S = \{s, 1, 2\}$
 $T = \{3, 4, t\}$
 $f(S, T) = 19$
 $c(S, T) = 26$

Algoritmo de Ford-Fulkerson



26/11/98

fordFulkerson(Nodo: s, t)

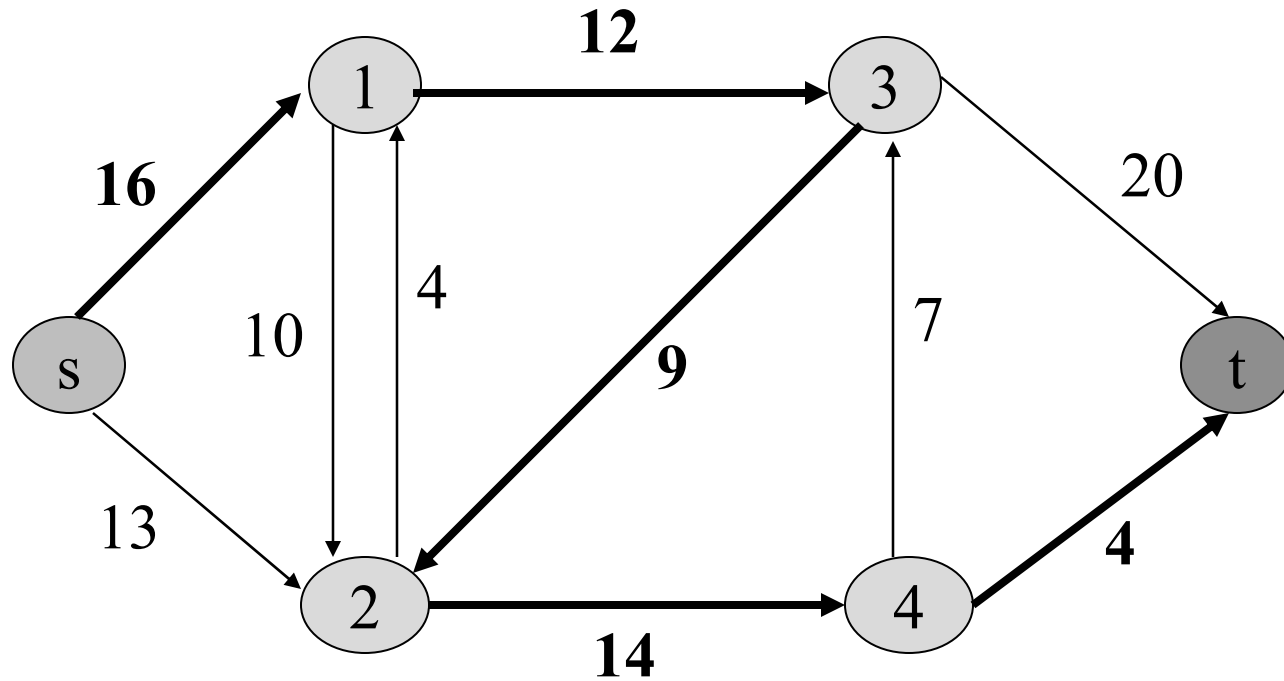
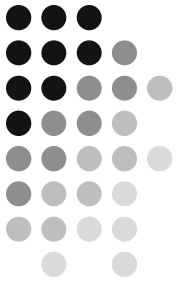
{pre: $n > 0$ }

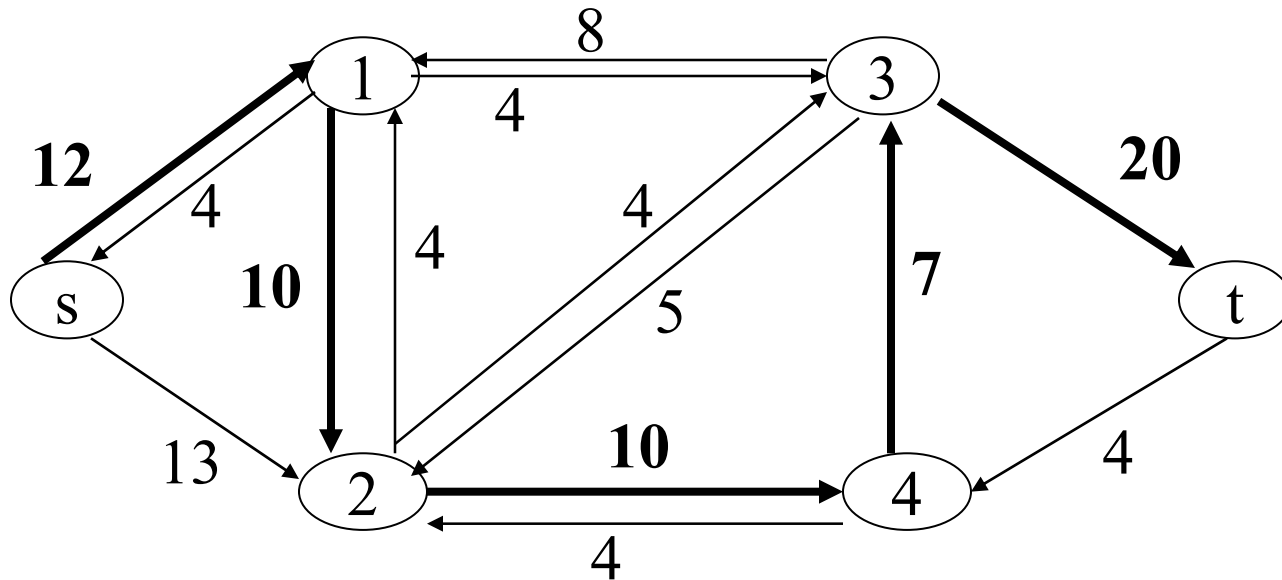
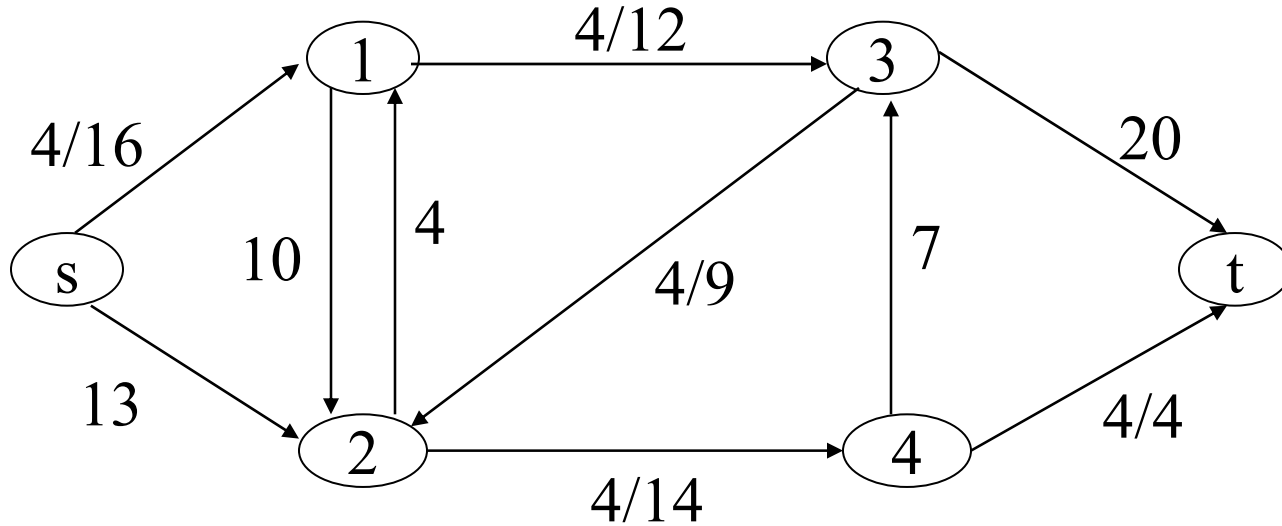
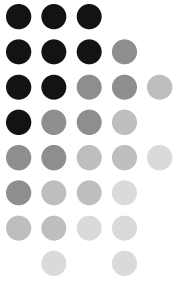
{pos: $n > 0$ }

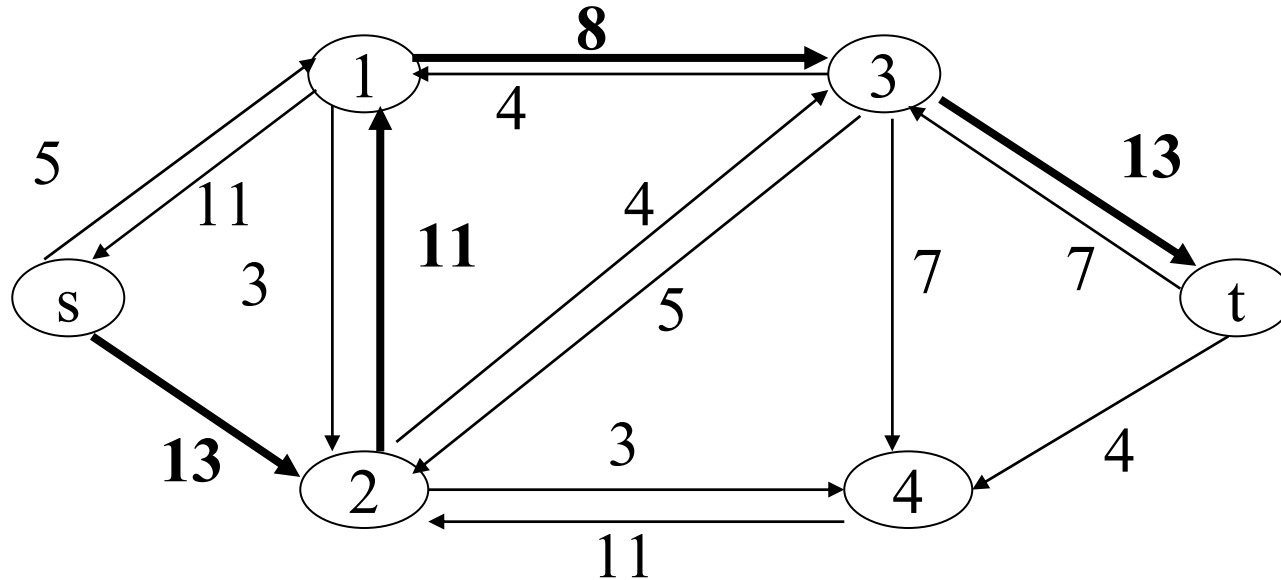
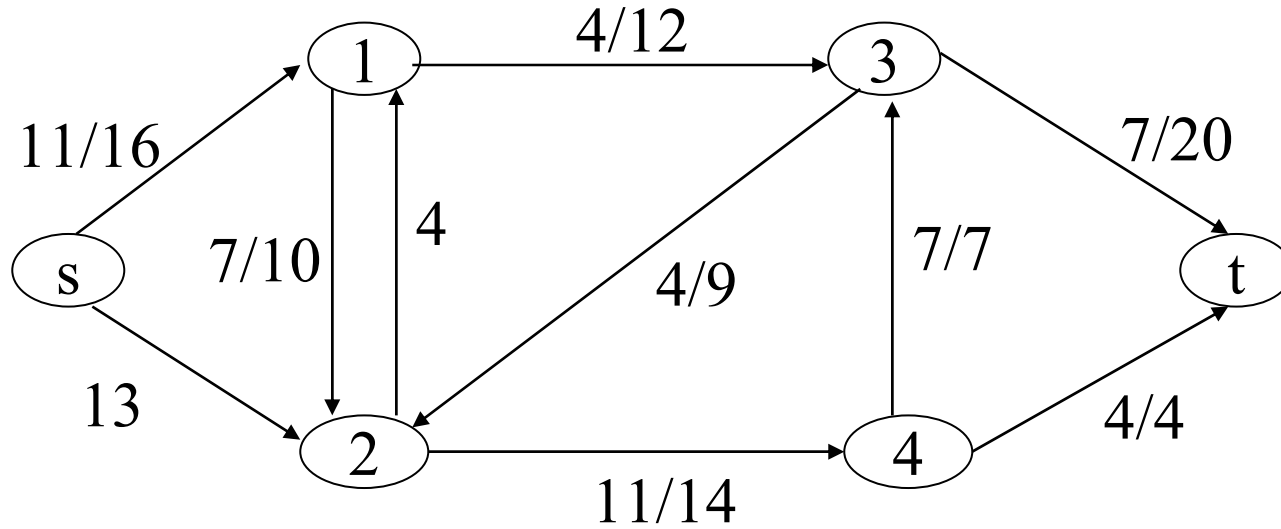
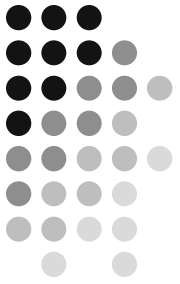
```

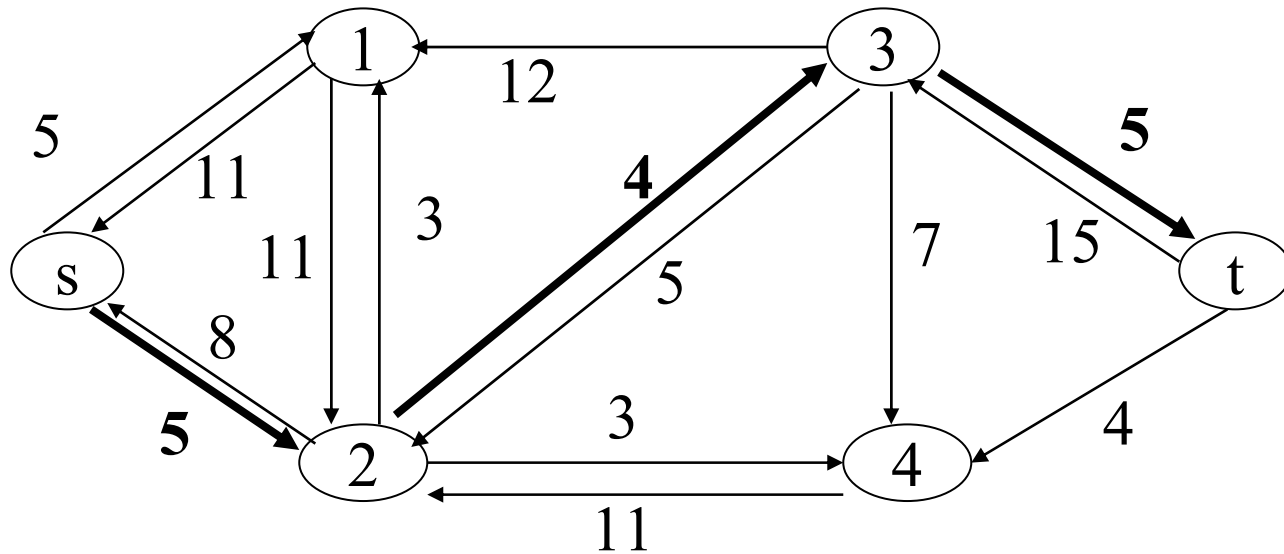
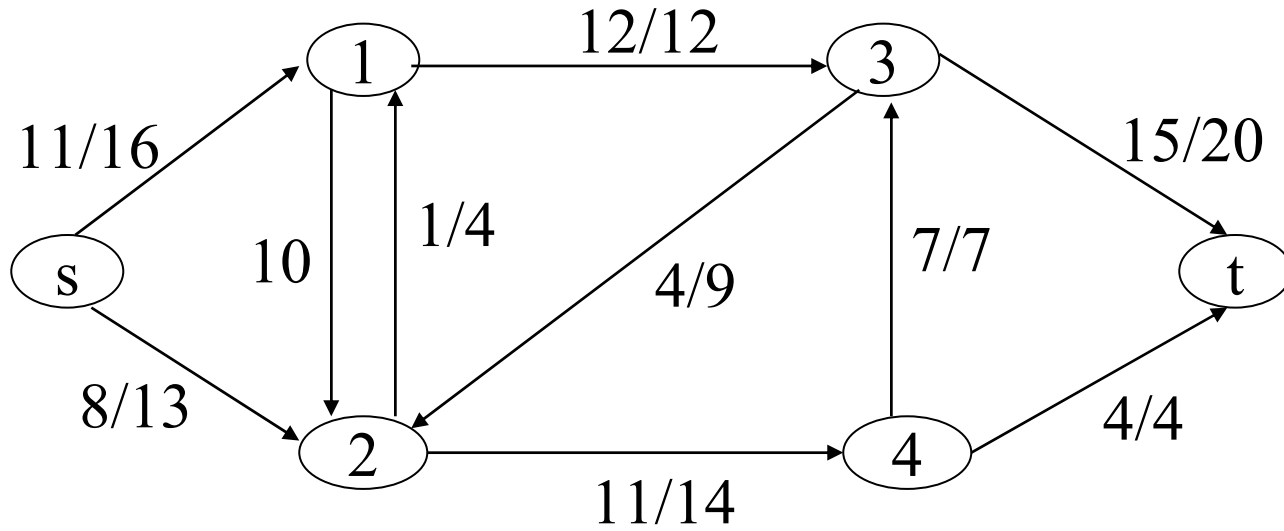
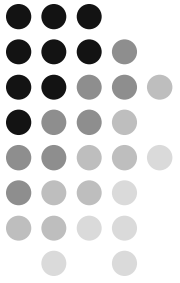
1 [ f(u, v), f(v, u) = 0, 0 ] (u, v) ∈ A // Inicia f
2 ( ∃ un p de s a t en Gf ) [ cf(p) = min(cf(u, v) : (u, v) está en p
    [ f(u, v) = f(u, v) + cf(p)
      f(v, u) = - f(u, v) ] (u, v) en p
    ] // Aumenta el flujo a lo largo de p
    // por cf (p) hasta que no hayan mas p
  
```

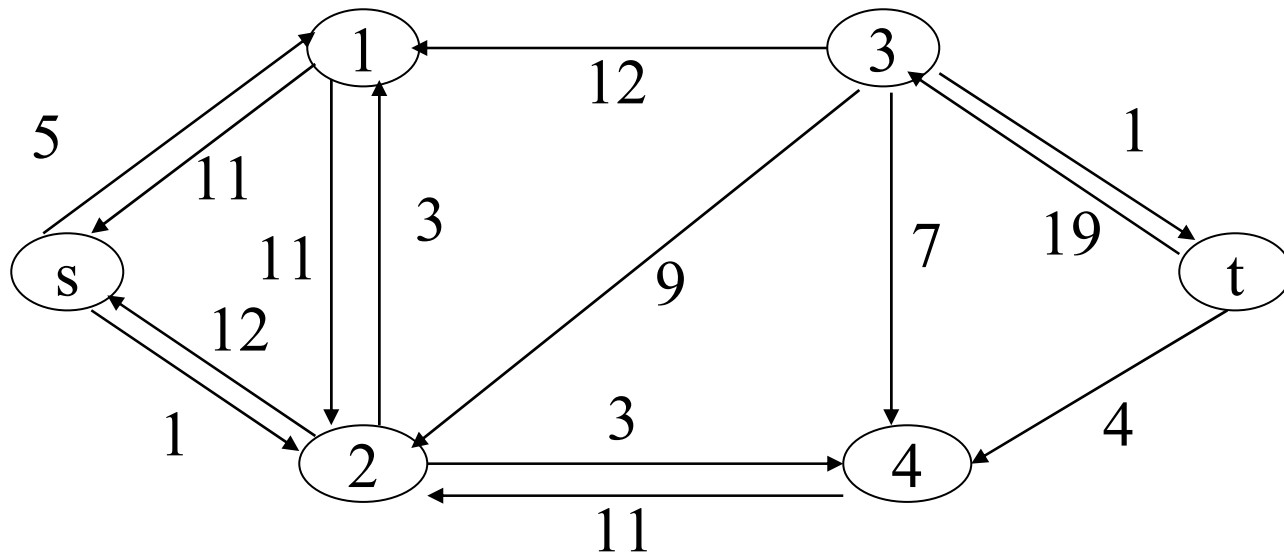
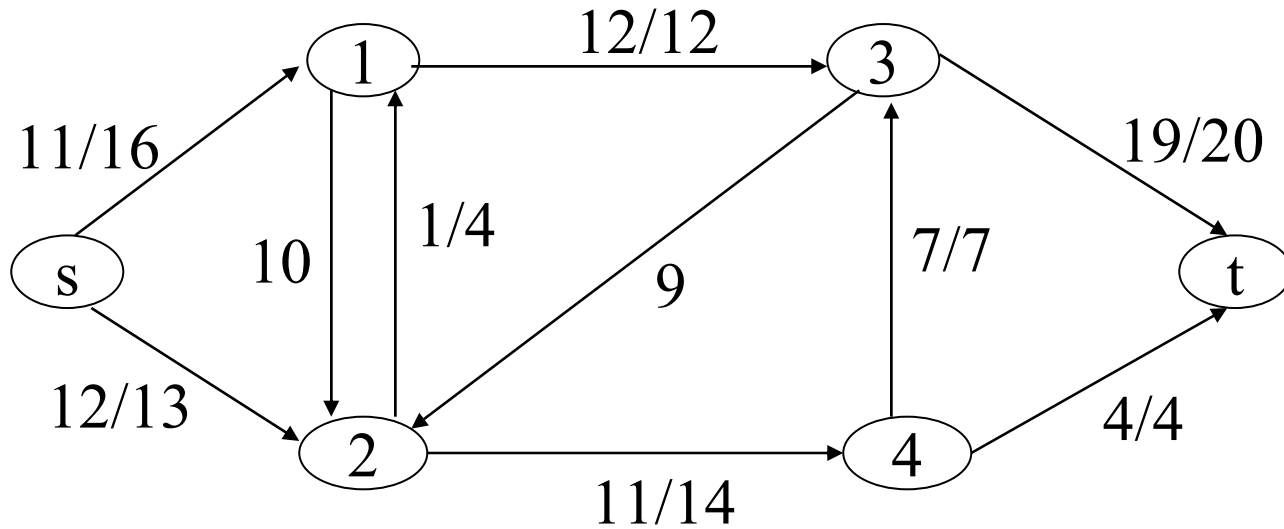
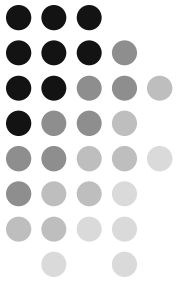
-**u, v**: Nodo. Nodos del grafo.
 -**Gf**: Grafo. Grafo residual.
 -**cf**: Entero. Capacidad residual
 -**p**: Camino aumentado de Gf.
 -**f**: Arreglo($n \times n$) De [Entero].
 Flujo del arco.



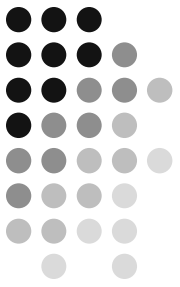






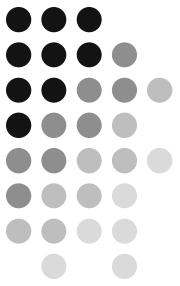


Análisis del algoritmo de Ford-Fulkerson



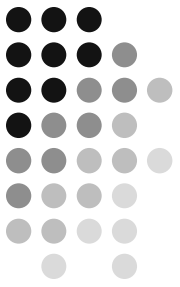
- El análisis del algoritmo depende de cómo se encuentren los caminos aumentados, pues es posible que no converja
- Con capacidades enteras, su $T(n) = O(A \mid f^* \mid)$, donde f^* es el máximo flujo encontrado

Algoritmo de Edmonds-Karp



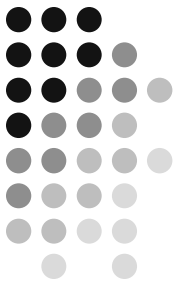
- Si se escogen los caminos aumentados con el método usado por la búsqueda en amplitud en el lazo del paso 2, entonces se obtiene el algoritmo Edmonds-Karp
- con tiempo polinomial $T(n) = O(N A^2)$, para cada iteración de Ford-Fulkerson $O(A)$ y para la búsqueda en amplitud $O(N A)$

Análisis del algoritmo Edmonds-Karp

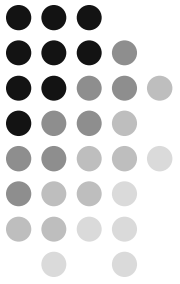


- **Lema 19.4:** Si el algoritmo Edmonds-Karp se ejecuta con una red de flujo G con s y t , entonces para todos los nodos $v \in N - \{s, t\}$, la distancia del camino más corto $\delta_f(s, t)$ en la red residual G_f se incrementa monotonamente con cada aumento del flujo
- **Teorema 2:** Si el algoritmo Edmonds-Karp se ejecuta con una red de flujo G con s y t , entonces el número total de aumentos del flujo desarrollados por el algoritmo es a lo sumo $O(N A)$.

Máximo "matching" en grafos bipartitos



- Dado un grafo no dirigido $G = (N, A)$, un "matching" es un subconjunto de aristas $M \subseteq A$, tal que para todos los nodos $v \in N$, a lo sumo una arista de M incide en v .
- Un "matching" M es máximo si tiene una cardinalidad máxima tal que para cualquier M' se tiene $|M'| \geq |M|$.
- **Aplicación:** Asignar un conjunto de R tareas que se efectuarán simultáneamente sobre un conjunto de L máquinas.
 - ❖ Se coloca una arista de la máquina i a la tarea j si ésta última se puede realizar en la máquina i .
 - ❖ La asignación es máxima cuando se asigna la mayor cantidad posible de tareas

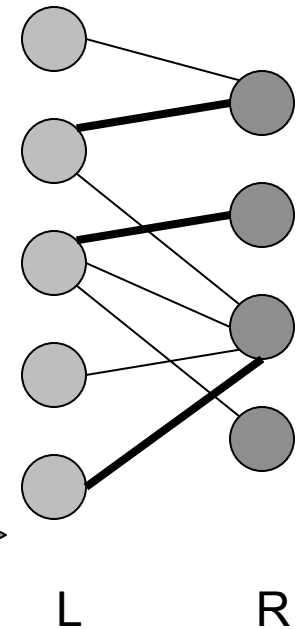
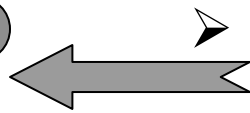
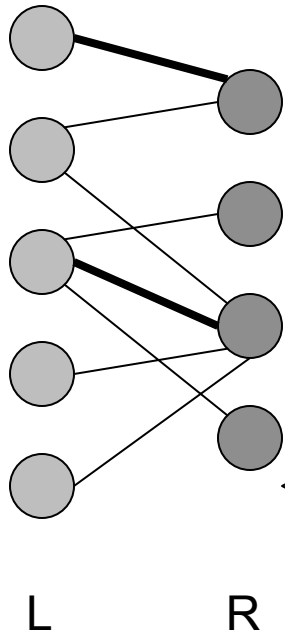


➤ $N = L \cup R$

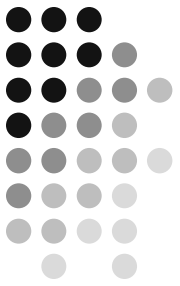
➤ Asignación de R tareas que serán desarrolladas simultáneamente en L máquinas

➤ Asignación de cardinalidad 2

➤ De cardinalidad 3



Grafo bipartito

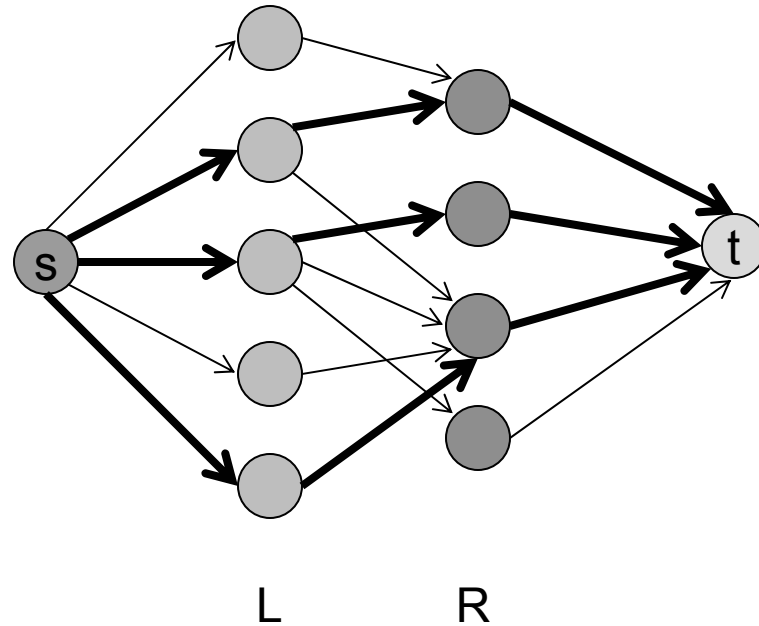
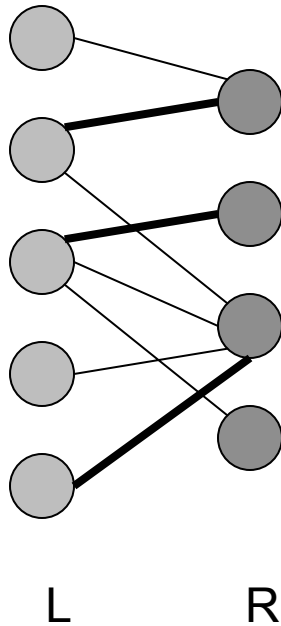
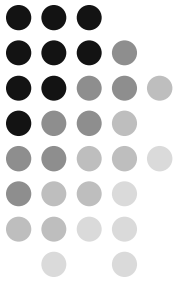


- **Grafo bipartito:** Aquel donde N puede ser dividido en $N = L \cup R$, $L \cap R = \emptyset$ y todas las aristas en A van de los nodos de L a los de R .
- El problema se resuelve con el algoritmo de Ford-Fulkerson construyendo una red de flujo $G' = (N', A')$ con dos nodos ficticios s y t , $N' = N \cup \{s, t\}$ y los arcos de G' son:

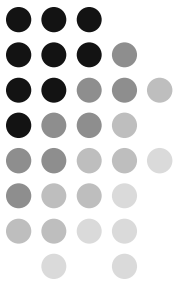
$$A' = \{(s, u) : u \in L\} \cup \{(u, v) : u \in L, v \in R, (u, v) \in A\} \cup \{(v, t) : v \in R\}$$

y se le asigna capacidad de 1 a los arcos de A' .

Aplicación del algoritmo



Asignación máxima en un grafo bipartito



- Si M es una asignación ("matching") en G , entonces hay un flujo de valor entero en G' cuyo valor $|f| = |M|$ y viceversa.
- **Teorema 3:** Si la función de capacidad c toma solamente valores enteros, entonces el flujo máximo f producido por el método de Ford-Fulkerson tiene la propiedad de ser entero ($|f|$) y $\forall u, v$ el valor de $f(u, v)$ es un entero.
- La cardinalidad de la asignación máxima en un grafo bipartito G es el valor del flujo máximo en su correspondiente red de flujo G' .
- Encontrar la asignación máxima en un grafo bipartito toma $O(N A)$.