Grafos y Flujo

Marcelo Lemus

marcelo.lemus@progcomp.cl

Importancia de flujo en ICPC

- En casi todas las regionales ICPC hay un problema
- La solución siempre suele ser armar el grafo y correr un algoritmo de flujo (que veremos ahora que posibilidades hay)
- Muchos no saben/manejan flujo
- El problema de flujo no parece que sea de flujo
- Junto con geometria computacional, el equipo que tenga a alguien que se maneje especialmente bien en alguno de estos temas, puede llegar a garantizar la victoria sobre los demas equipos.

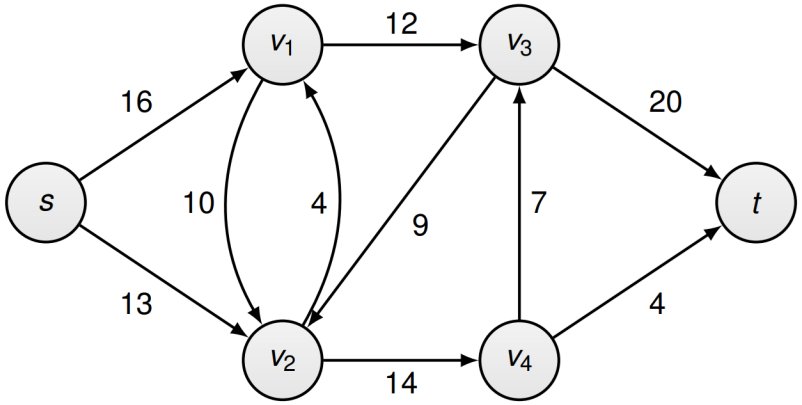
Redes de flujo

Una red flujo consiste en:

- Un grafo dirigido G con pesos en las aristas
- Dos nodos especiales: s (fuente) y t (sumidero)

Nuestro objetivo es mandar la mayor cantidad de unidades de flujo de s a t

- Los pesos en las aristas representan la cantidad maxima de flujo que soportan



Flujo sobre una Red

Es una asignacion de un real no negativo a cada arista tal que:

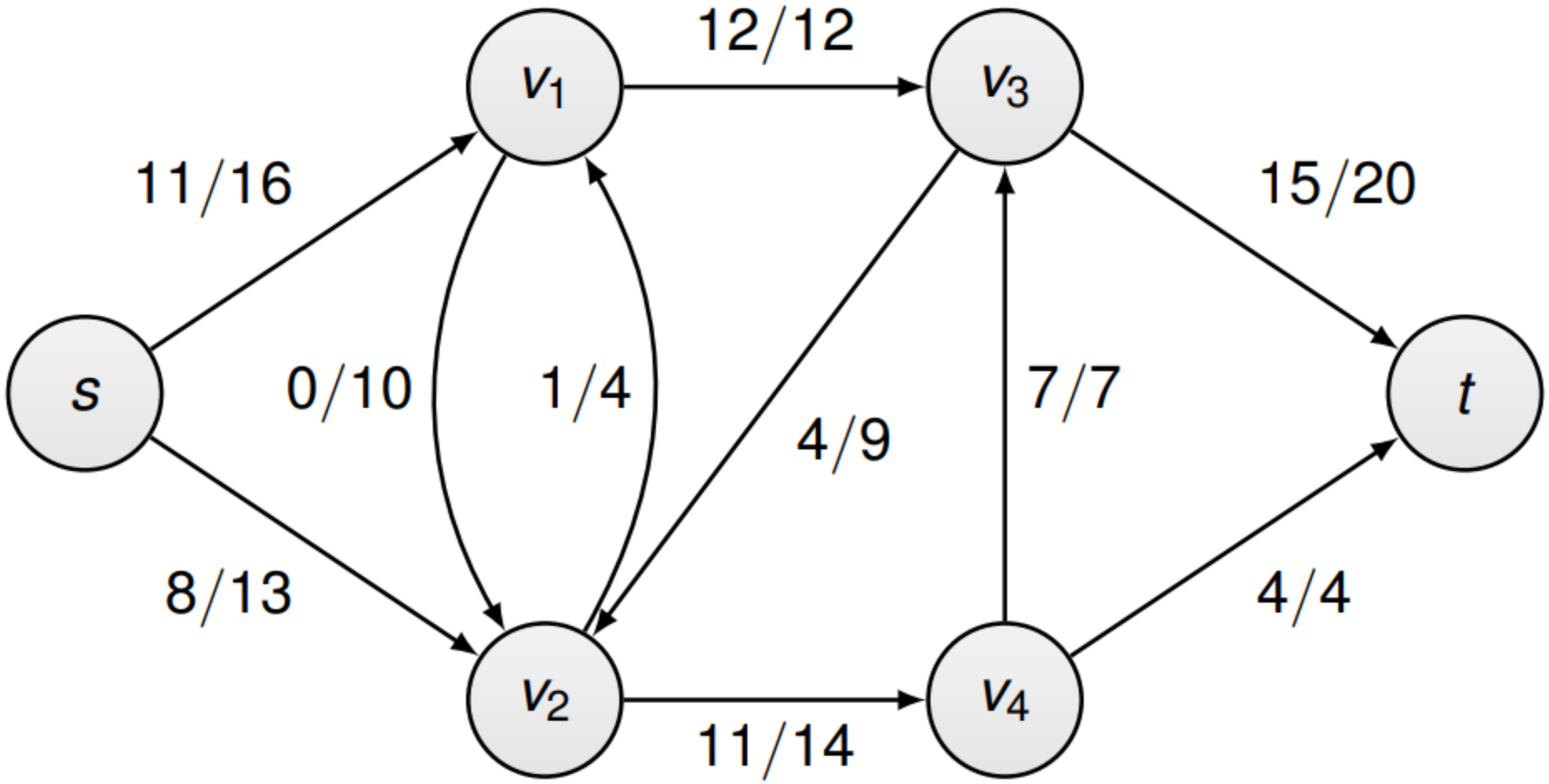
- Para cada arista digo cuantas unidades de flujo pasan por ella.
- No asigno a una arista mas de su capacidad maxima.
- En cada nodo excepto s y t el flujo que entra tiene que ser igual al que sale.

El valor del flujo es:

- La cantidad de flujo que sale de s.
- La cantidad de flujo que entra a t.

Ambos numeros siempre coinciden, queremos encontrar cual es el mayor valor posible de un flujo sobre la red.

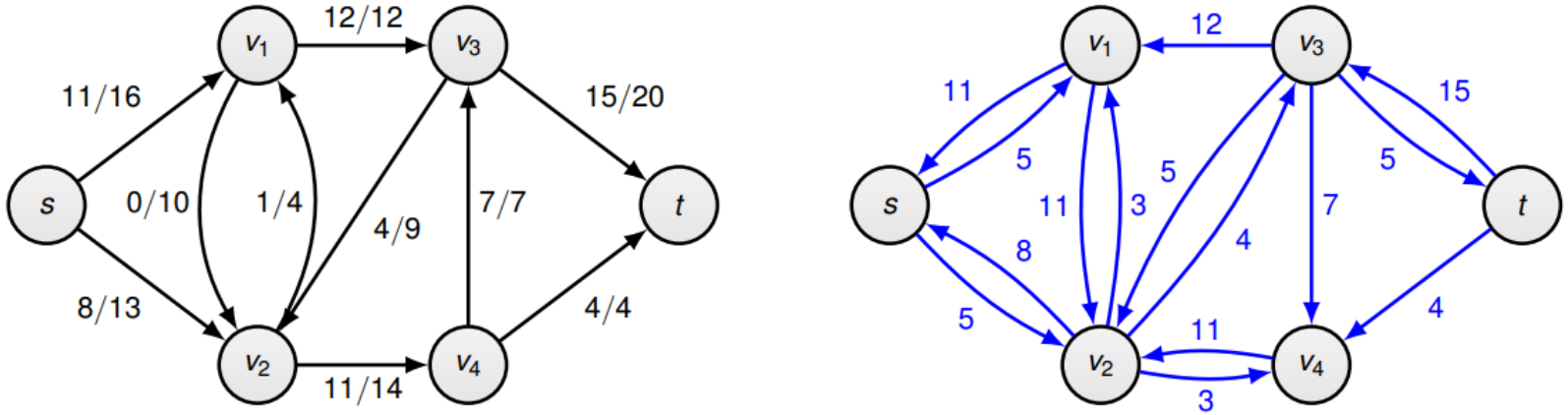
Ejemplo



Red Residual

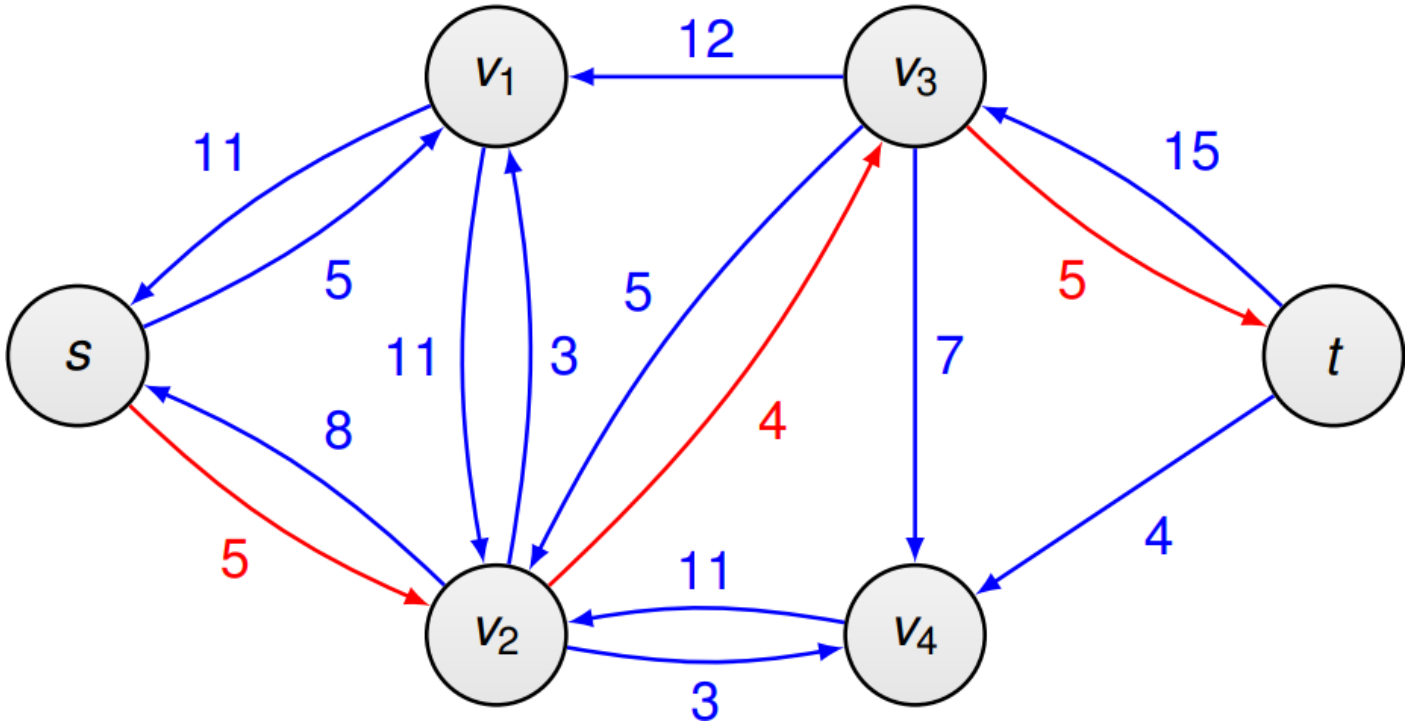
La red residual de un flujo es un grafo dirigido tal que:

- Los nodos son los mismos que en la red original.
- Hay una arista de u a v con capacidad la suma de:
  - Lo que me falta para llenar la capacidad de u a v en el grafo original.
  - El flujo que estoy mandando de v a u.



Augmenting Path (Camino de aumentanto)

Un camino de aumento es un camino de s a t en la red residual.
Dado un camino de aumento podemos conseguir un flujo mayor mandando todo lo posible por este camino.



Ford Fulkerson

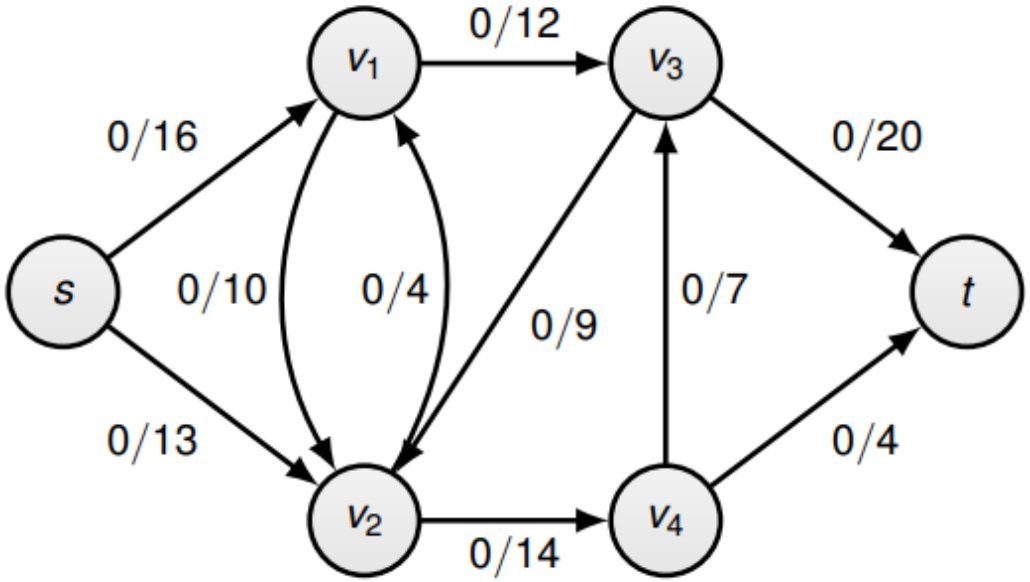
Este seria un posible algoritmo para calcular el flujo maximo.

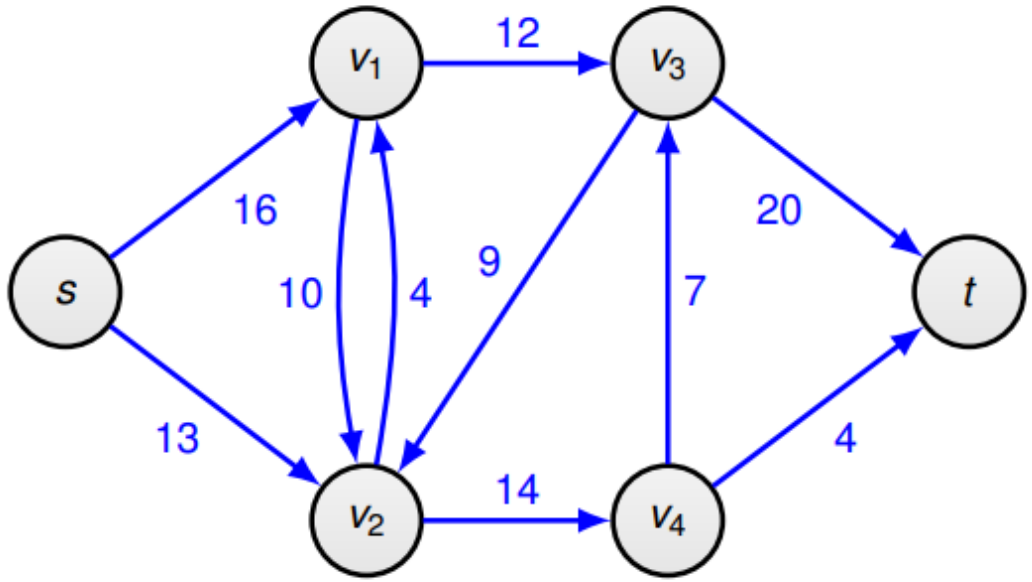
- Empiezo con el flujo vacio (el flujo de cada arista es 0)
- Construyo la red residual
- Mientras haya camino de s a t en la red residual:
  - Busco el la arista con menor peso (x) del camino.
  - Actualizo el flujo para mandar x unidades de flujo por ese camino.
  - Actualizo la red residual.

Devuelvo el flujo obtenido.

Ejemplo de flujo con Ford Fulkerson

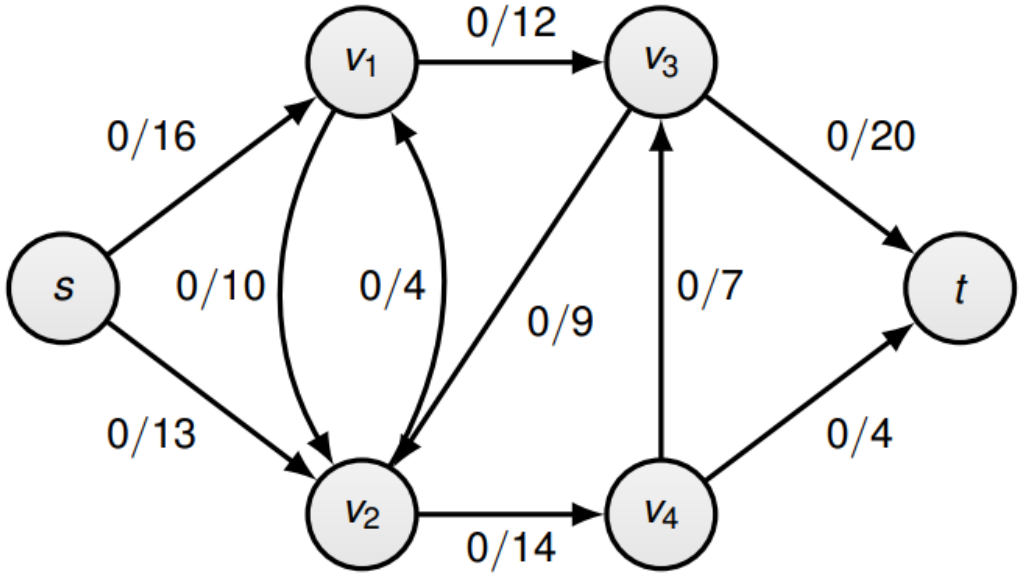
Empezamos con 0 en todas las aristas (flujo enviado)

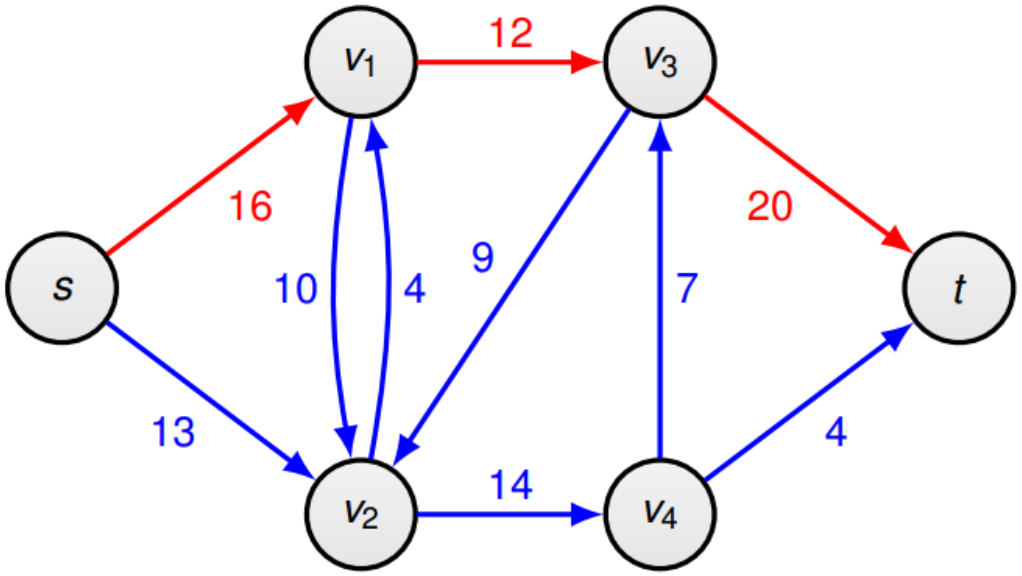




Ejemplo de flujo con Ford Fulkerson

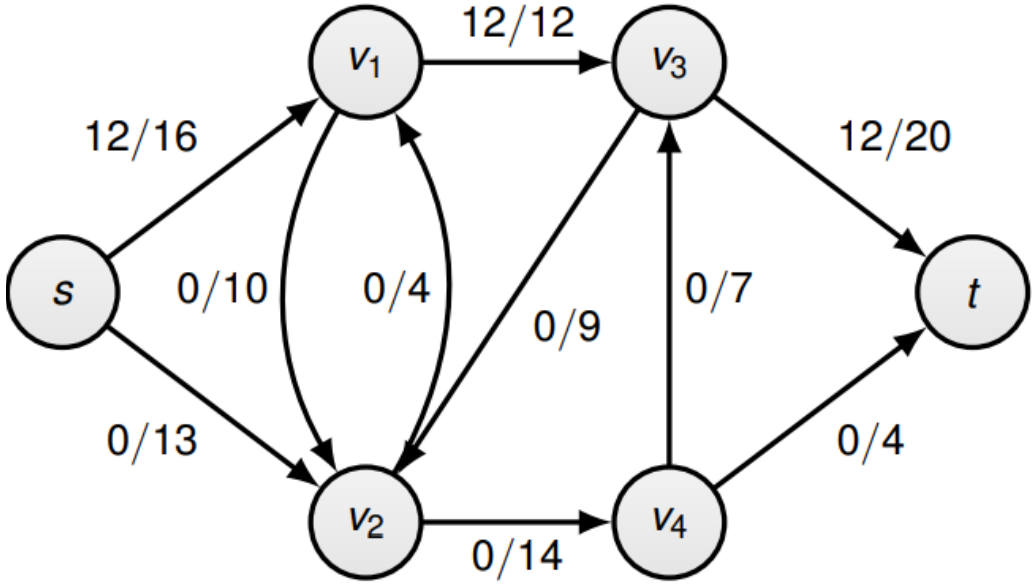
Encontramos un camino aumentante en la red residual.

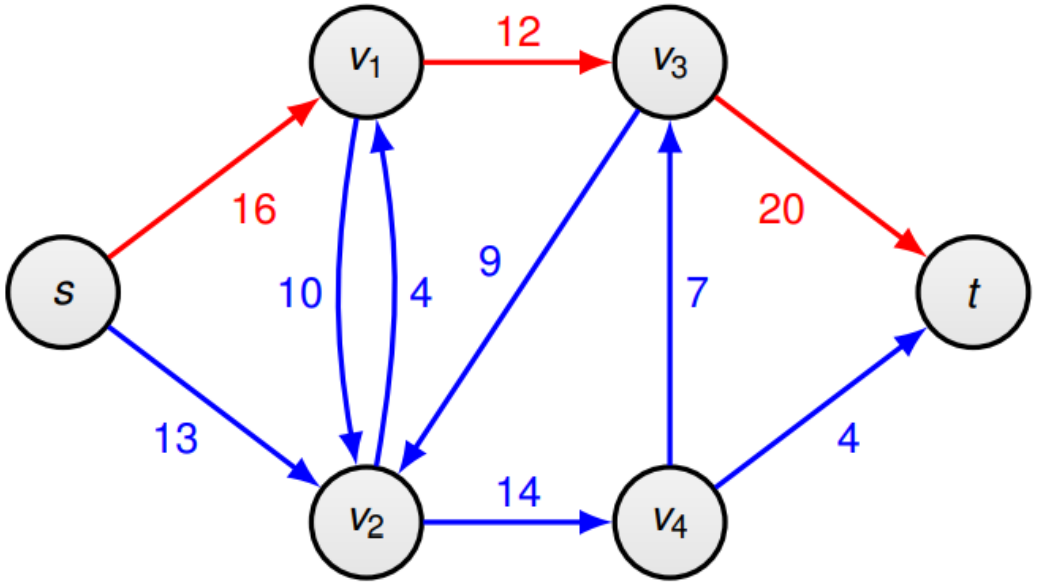




Ejemplo de flujo con Ford Fulkerson

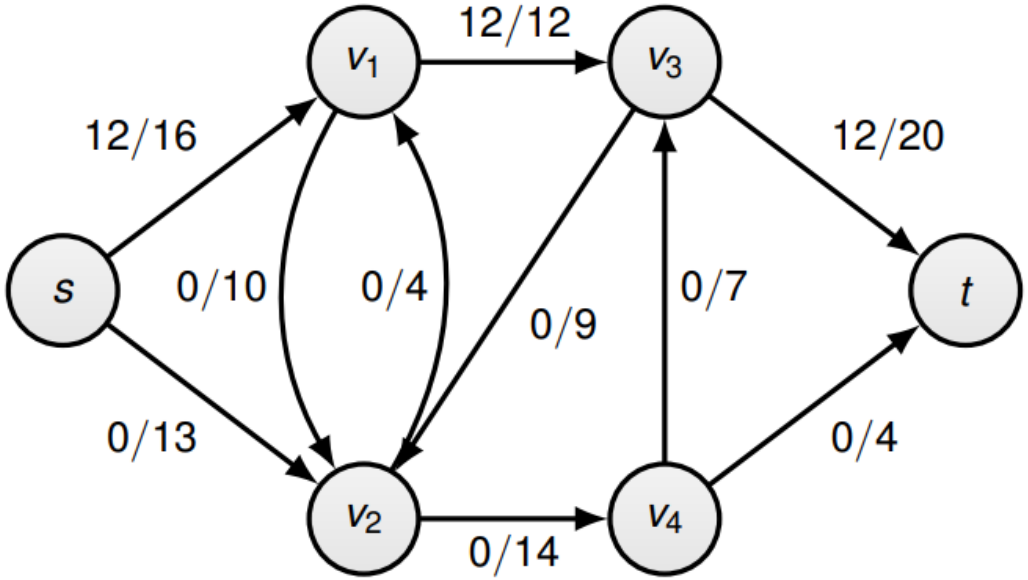
Mandamos todo lo que podemos por ese camino

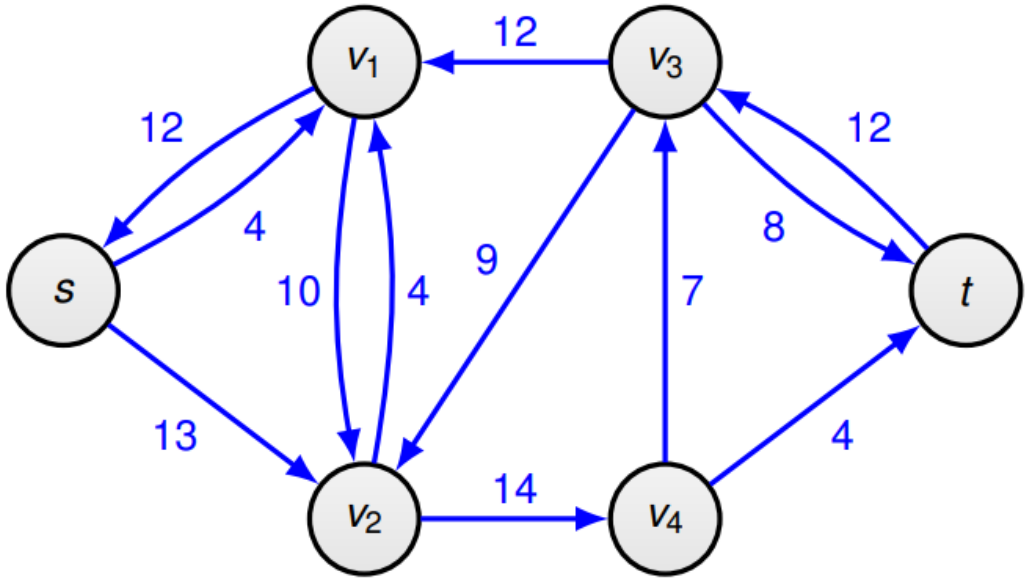




Ejemplo de flujo con Ford Fulkerson

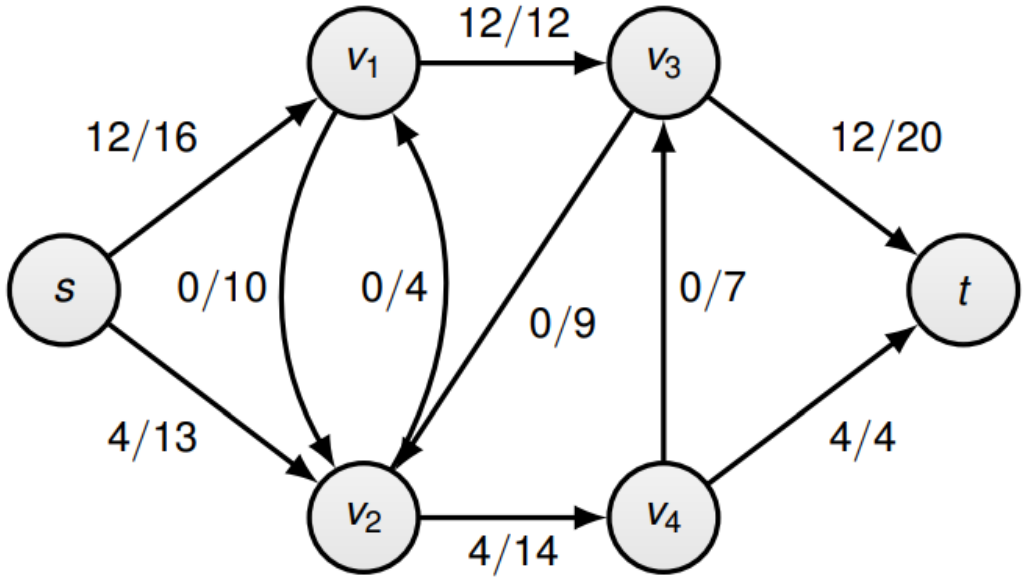
Actualizamos la red residual.

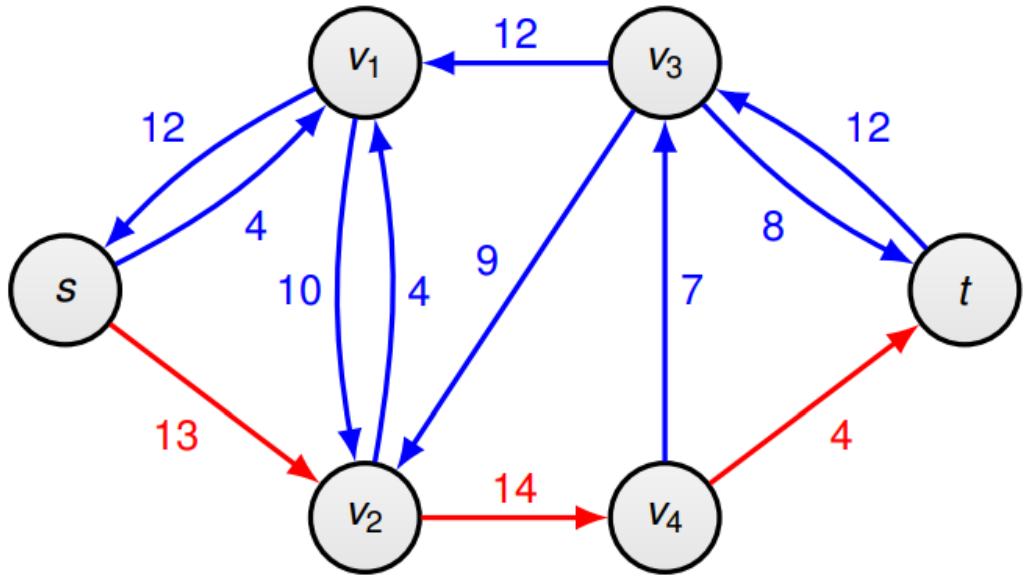




Ejemplo de flujo con Ford Fulkerson

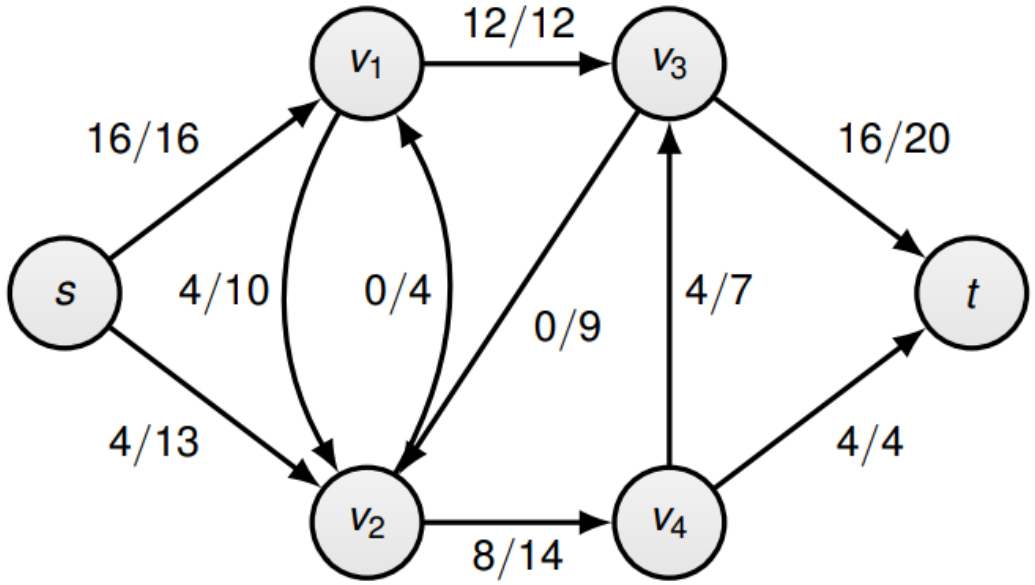
Volvemos a buscar un camino aumentante en la red residual y mandamos flujo por el.

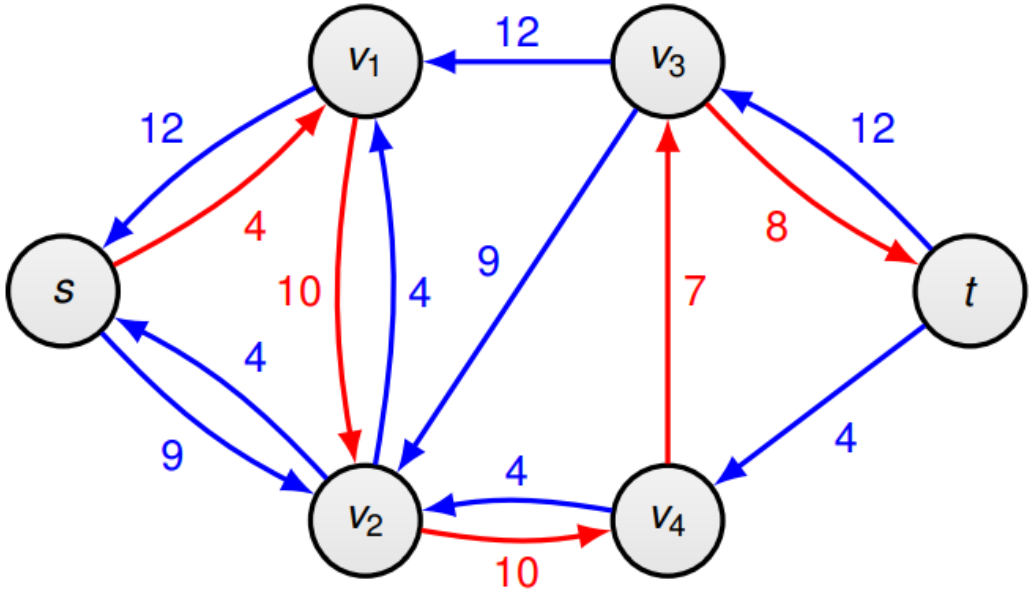




Ejemplo de flujo con Ford Fulkerson

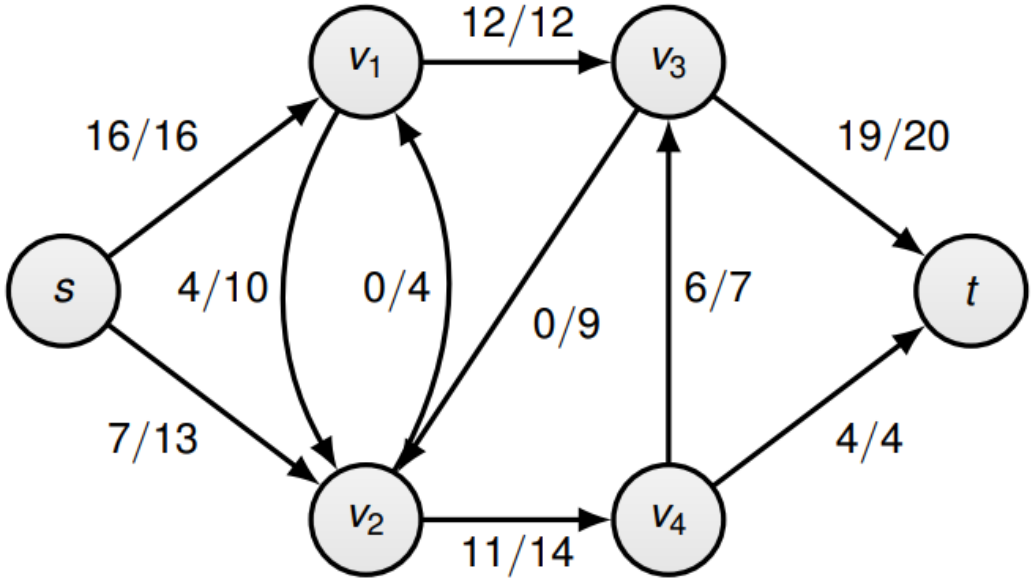
Actualizamos la red residual y repetimos.

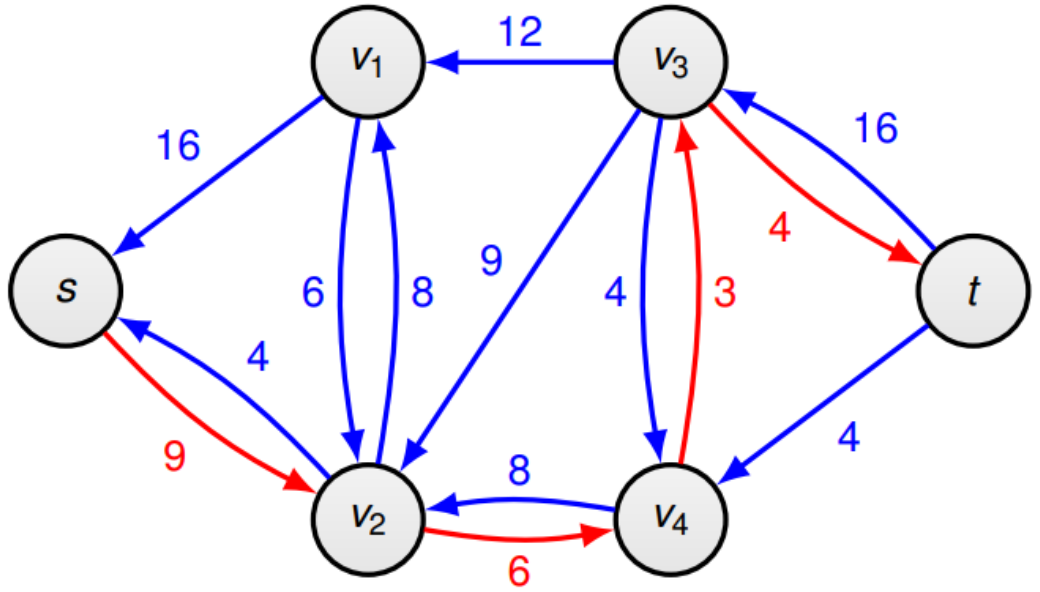




Ejemplo de flujo con Ford Fulkerson

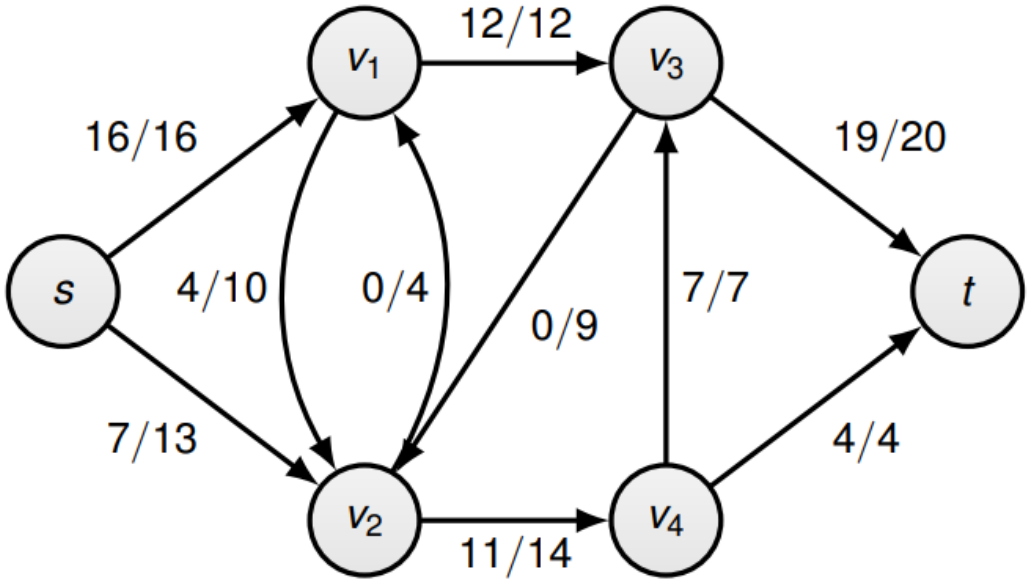
Actualizamos la red residual y repetimos.

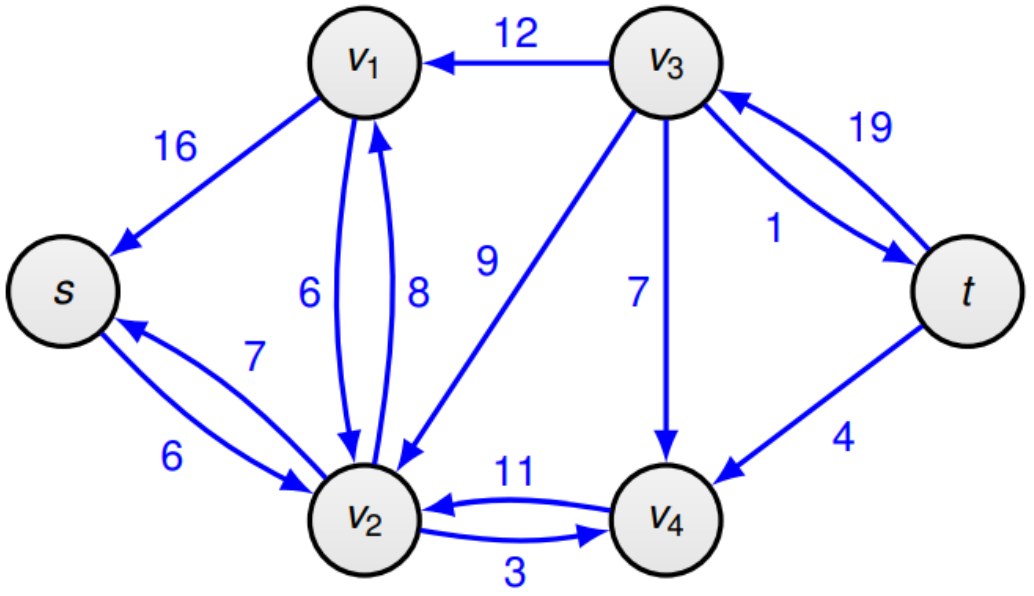




Ejemplo de flujo con Ford Fulkerson

No hay mas caminos aumentantes en la red residual, paramos.





Edmond Karp y Dinitz (Dinic)

Si ademas buscamos el camino de s a t con BFS:

- El algoritmo queda $O(nm^2)$ con n la cantidad de nodos y m la de aristas.
- Este algoritmo es conocido como Edmond Karp.

Tambien existe el algoritmo de Dinic

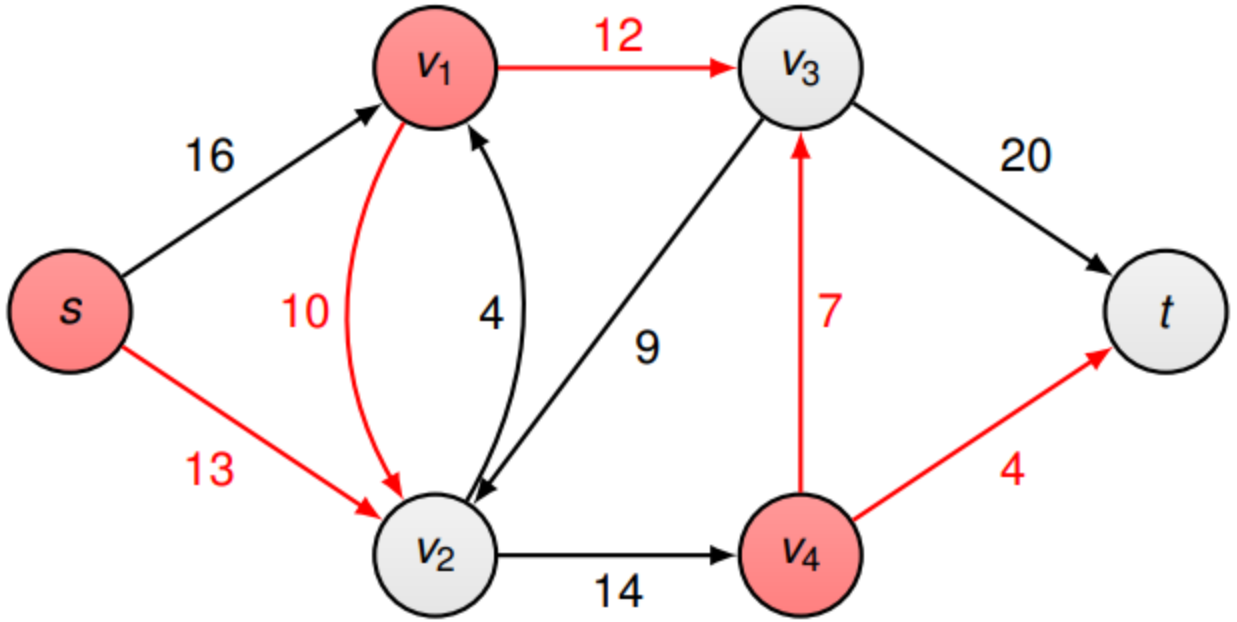
- construye los niveles usando bfs y luego encuentra todas las rutas disjuntas posibles de esa capa utilizando DFS antes de actualizar el grafo.
- El algoritmo queda $O(n^2m)$ por lo que es mas rapido que Edmond Karp.
- Es bastante mas largo de implementar.

Siempre debemos considerar la cota $O(mF)$ para todo algoritmo de maximo flujo, donde F es el valor del flujo máximo.

Hay otros algoritmos situacionales como push relabel y MPM, pero en la practica y en las competencias, entonces no son necesarios.

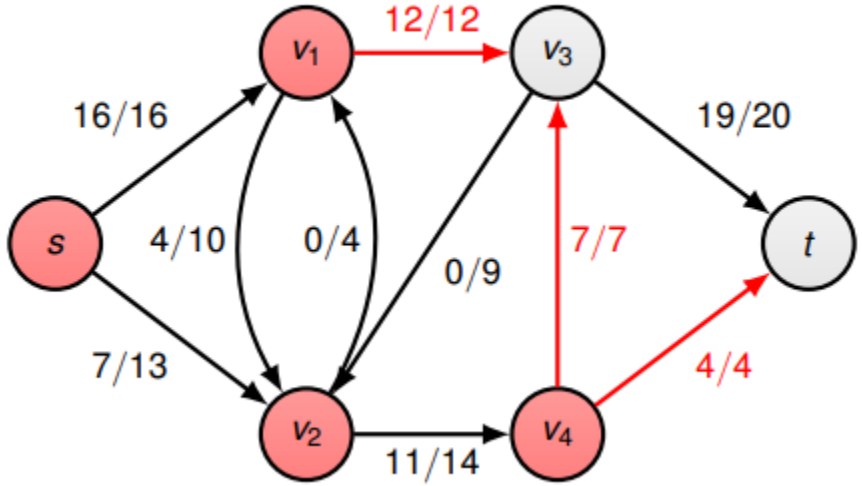
Definicion de corte

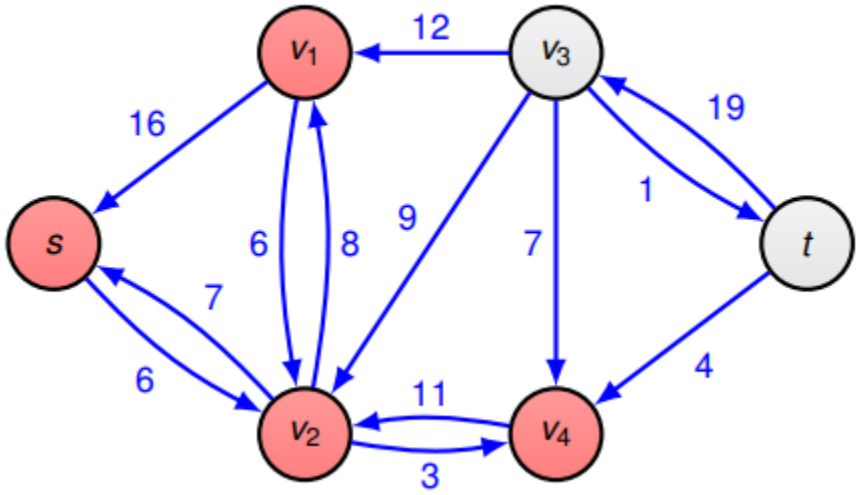
Un corte en una red es una partición de los nodos en dos conjuntos U y V tal que $s \in U$ y $t \in V$
.
Dado un corte en una red su valor es la suma de las capacidades de todas las aristas que van de un nodo de U a uno de V.



Teorema de Max-Flow Min-Cut

- El valor del corte mínimo coincide con el del flujo máximo.
- Podemos encontrar un corte mínimo definiendo:
  - U como los nodos alcanzables desde s en la red residual del flujo máximo.
  - V como el resto de los nodos.



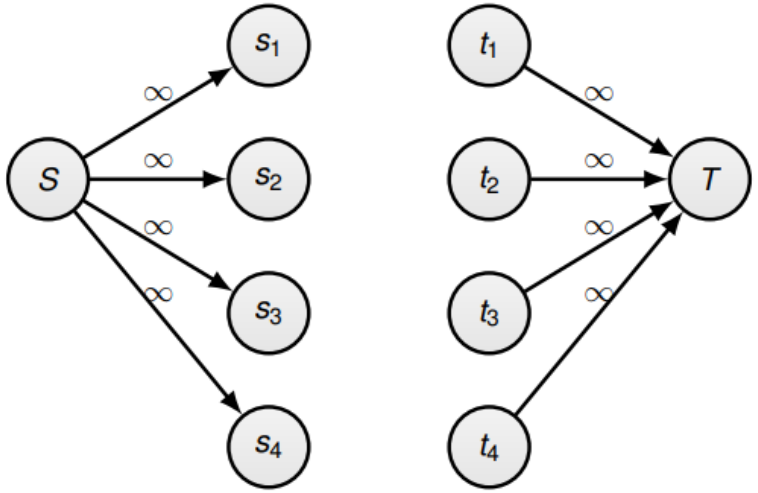


Trucos de Modelacion

Multiples fuentes/sumideros

Supongamos que tenemos un problema donde hay varios nodos de donde ”sale” flujo y/o varios nodos a los que tiene que llegar flujo, ¿Qué hacemos?

- Creamos dos nodos s y t nuevos.
- Conectamos s a todos los nodos de los que ”sale” flujo con capacidad ∞
- Conectamos todos los nodos a los que llega flujo con t con capacidad ∞.



Restricciones sobre los nodos

¿Qué pasa si tenemos restricciones de flujo que pasa por un nodo v en lugar de por un eje?

- Duplicamos el nodo en dos nodos v_{in} y v_{out}.
- Conectamos todas las aristas que entraban a v a v_{in}.
- Conectamos todas las aristas que salían de v a v_{out}.
- Conectamos v_{in} a v_{out} con un eje de la capacidad máxima del nodo