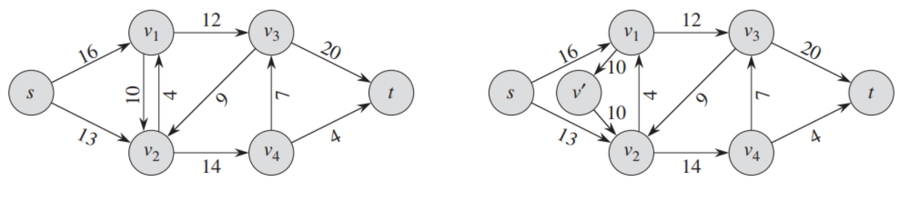
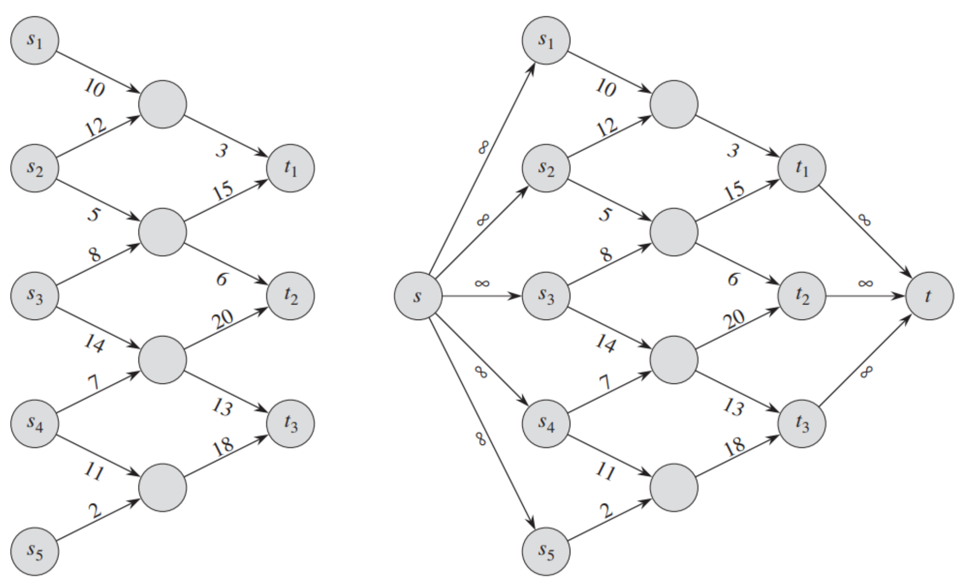
Flujo Máximo

- El problema de flujo máximo consiste en obtener la cantidad de flujo máximo que podemos transportar desde el source al sink conservando el flujo de todos los vertices intermedios.
- **Conservación de flujo**: El flujo total que entra a un vertice tiene que ser igual al flujo total que sale del mismo. (Excepto soruce y sink).
- **flow network**: Un grafo dirigido sin ciclos $G = (V, E)$ donde todos los $(u, v) \in G.E$ tienen un capacidad $c(u, v) \geq 0$. Y se distinguen 2 vertices: s (source) y t (sink). Las aristas que no estan en el grafo tienen costo 0. Y para cada vertice v del grafo, hay un camino de s a v y de v a t (grafo conectado, $|E| \geq |V|$). (Tampoco se permiten self-loops).
- *Una restricción adicional*: si $(u, v) \in E, entonces (v, u) \notin E$. Si tenemos la arista de ida y quicieramos agregar la de la vuelta, podríamos agregar un vertice intermedio v' talque $v \to v'$ y $v' \to u$ ambas con el costo deseado.



(a) (b)

- **flujo**: Dado un grafo de flujo con función de capacidad c, soruce s y sink t, se define el *flujo* como $f : V \times V \to \mathbb{R}$ (recibe una arista y devuelve un real), que satisface las restricciones de:
  - Capacidad: $0 \leq f(u, v) \leq c(u, v)$
  - Conservación: $\sum_{v \in V} f(v, u) = \sum_{v \in V} f(u, v)$. Para $(u, v) \in E, sino f(u, v) = 0$
- **Cantidad de flujo**: $|f| = \sum_{v \in V} f(s, v) - \sum_{v \in V} f(v, s)$, (el segundo término es 0). El problema de máximo flujo desea maximizar ésta cantidad.
- **Multiples sources y sinks**: Se quiere obtener el máximo flujo pero ahora se tiene un conjunto $\{s_1, ..., s_m\}$ de sources y un conjunto $\{t_1, ..., t_n\}$ de sinks. Se puede modelar el problema agregando un "supersource" s conectado a todos los sources tq $c(s, s_i) = \infty$, y un "supersink" t conectado a todos los sinks tq $c(t_i, t) = \infty$. Y el problema es equivalente a el de un source y un sink.



(a) (b)

El Método de Ford-Fulkerson

- Se lo llama método ya que puede tener distintas implementaciones con distintas complejidades.

- Iterativamente incrementa el valor del flow. Se inicia con $f(u, v) = 0$ para todas las aristas, dando un valor de flujo inicial de 0. En cada iteración se busca un camino de aumento en el "grafo resudial" que nos permitirá saber que aristas aumentar y disminuir el flujo para aumentar el flujo total. Se repite hasta que en el grafo residual no haya más caminos de aumento.

```
Ford-Fulkerson-Method(G, s, t):
    initialize flow f to 0
    while there exists an augmenting path p in the residual network G_f
        augment flow f along p
    return f
```

Grafos reciduales

- Dado un grafo G de flujo y f un flujo, el grafo residual G_f consiste en los mismos vertices de G pero con aristas que representan los aumentos y decrementos posibles al flujo de las aristas de G. La diferencia es que si en G está la arista (u, v), en G_f también está la arista (v, u).
- **Capacidad residual**: definimos las capacidades de las aristas del grafo residual cómo:

$$c_f(u, v) = \begin{cases} c(u, v) - f(u, v) & \text{if } (u, v) \in E \ , \\ f(v, u) & \text{if } (v, u) \in E \ , \\ 0 & \text{otherwise} \ . \end{cases}$$

```
- Si  $(u, v) ∈ G.E$, la capacidad residual indica cuanto flujo más podemos enviar desde u hacia v.
- Si $(u, v) \notin  G.E$, entonces si $(v, u) ∈ G.E$ indica cuanto flujo podemos "devolver" de v hacia u. (que es justamente e
- Caso contrario en G se está enviando el maximo flujo posible o no hay arista de u a v ni v a u.
- $|E_{f}| \leq 2*|E|$
```

- El flujo en el grafo recidual indica cómo cambiar el flujo en el grafo original para aumentar el flujo total.

A flow in a residual network provides a roadmap for adding flow to the original flow network. If f is a flow in G and f' is a flow in the corresponding residual network G_f, we define $f \uparrow f'$, the ***augmentation*** of flow f by f', to be a function from $V \times V$ to $\mathbb{R}$, defined by

$$(f \uparrow f')(u, v) = \begin{cases} f(u, v) + f'(u, v) - f'(v, u) & \text{if } (u, v) \in E \ , \\ 0 & \text{otherwise} \ . \end{cases} \tag{26.4}$$

Aumentando caminos