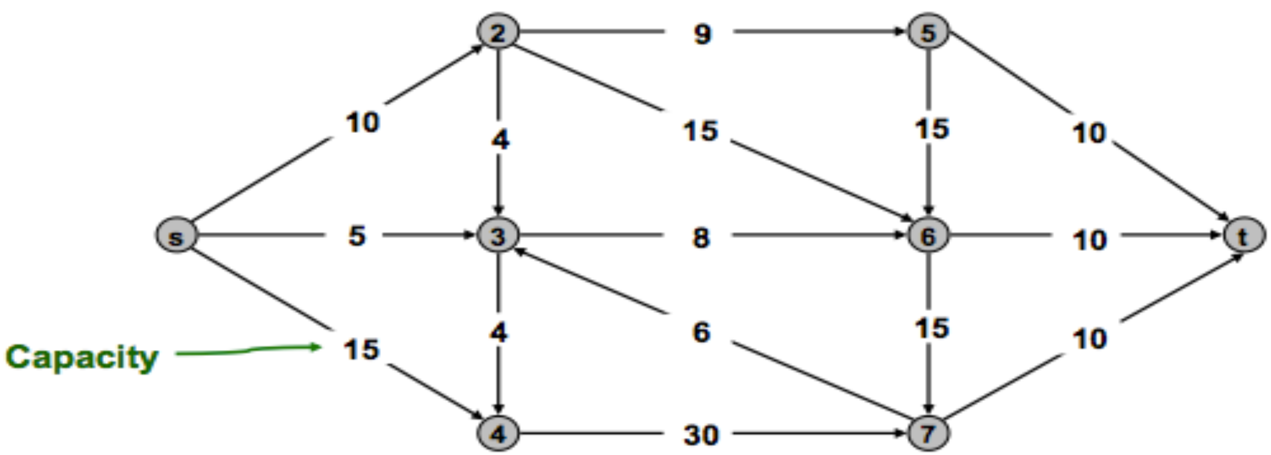
MAX FLOW PROBLEM

Max Flow Network

- Abstraction for material flowing through the edges.
- Max flow network: G = (V, E, s, t, u) .

  (V, E) = directed graph with source $s \in V$ and sink $t \in V$.

  no parallel edges, no edge enters s, no edge leaves t
- Two distinguished nodes: s = source, t = sink.
- u(e) = capacity of arc e.



Max Flow Problem & some definitions

- Max Flow Problem

  Given a flow network G with source s and sink t

  Find a flow of maximum value from s to t.

- Capacity – It is the sum of the capacities of the edges from *A* to *B*.

  cap(A,B) = ∑ c(e) e out of A

- **Capacity constraint:** For all u; v € V, we require 0 ≤ f(u,v) ≤ c(u,v)

- **Flow conservation:** For all u € V - { s; t } , we require

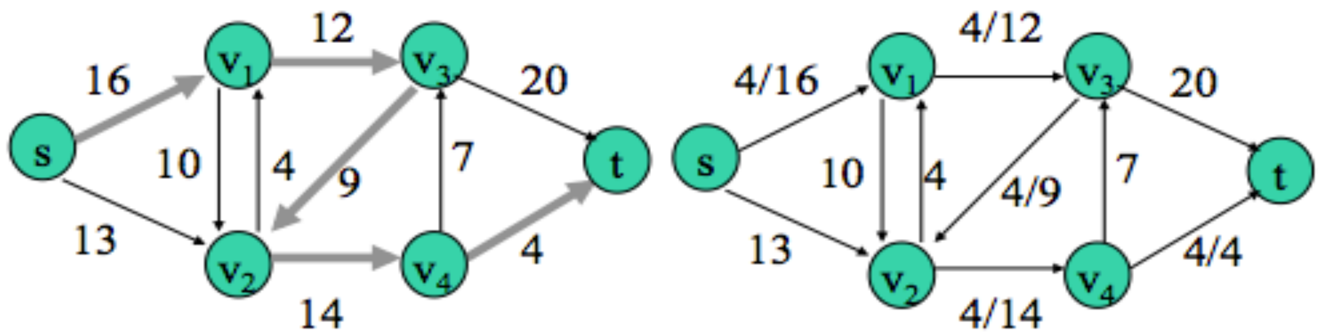
  Σ f(v,u) = Σ f(u,v) for all v € V

  When(u,v) is not in V, there can be no flow from u to v , and f(u,v) = 0.

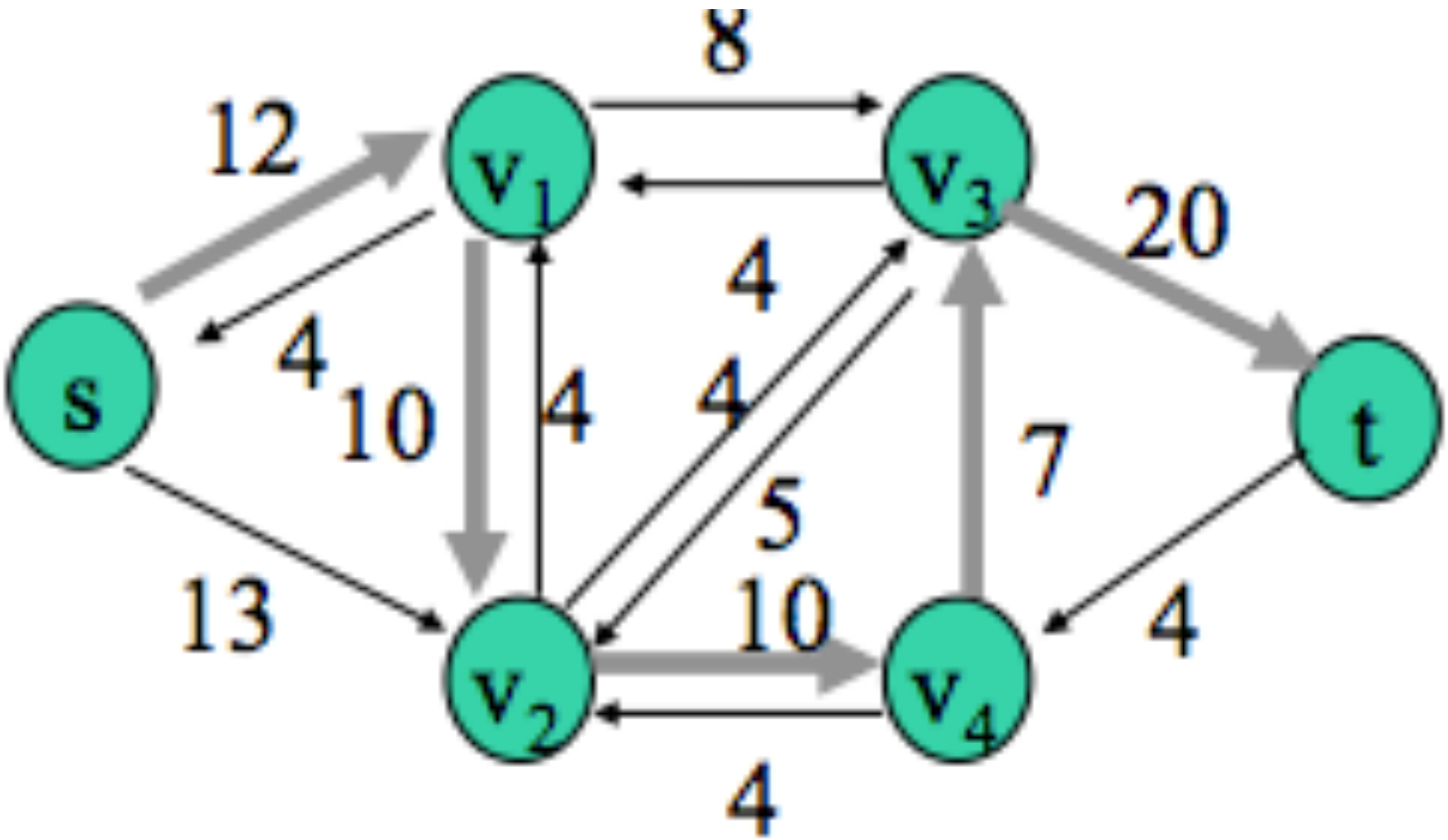
Residual Networks

- Given a flow network and a flow, the **residual network** consists of edges that can admit more net flow.
- G=(V,E) --a flow network with source s and sink t
- f: a flow in G.
- The amount of additional net flow from u to v before exceeding the capacity c(u,v) is the residual capacity of (u,v), given by: $c_f(u,v)=c(u,v)-f(u,v)$

  in the other direction: $c_f(v, u)=c(v, u)+f(u, v)$.



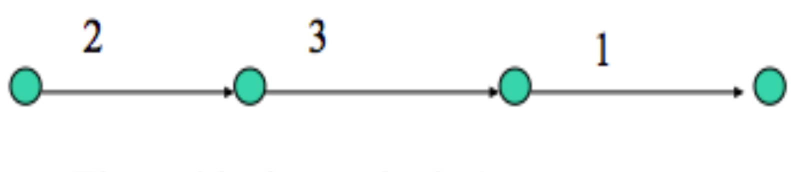
Residual Continued

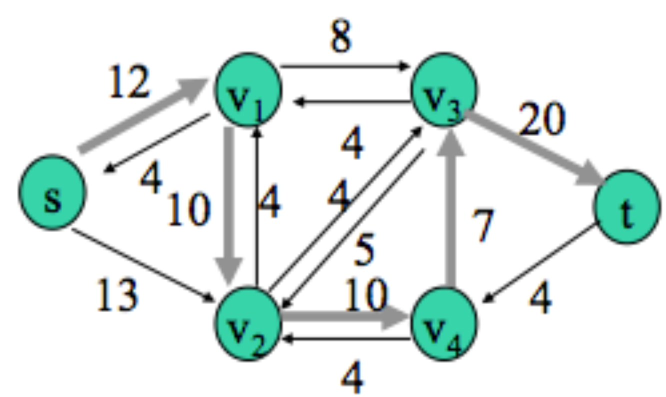


Augmenting Path

- Given a flow network G=(V,E) and a flow f, an augmenting path is a simple path from s to t in the residual network G_f.

- Residual capacity of p : the maximum amount of net flow that we can ship along the edges of an augmenting path p, i.e., $c_f(p)=\min\{c_f(u,v):(u,v) \text{ is on } p\}$.





- The residual capacity is 1.

Finding the minimum cut

- Let d*( ) be the distance labels at the end of the algorithm.
- Let k* be the minimum positive value such that there is a gap at level k*.
- Let S* = {j : d*(j) < k*}.   Let T* = {j : d(j*) > k*}.

Theorem.   (S*, T*) is a minimum capacity cut, and the capacity of the cut is the amount of flow into t.

Preflows

At each intermediate stages we permit more flow arriving at nodes than leaving (except for s)

A *preflow* is a function $x: A \rightarrow R$ s.t. $0 \leq x \leq u$ and such that

$$e(i) = \sum_{j \in N} x_{ji} - \sum_{j \in N} x_{ij} \geq 0,$$

for all $i \in N - \{s, t\}$.

i.e., e(i) = *excess* at i = net excess flow into node i.

The excess is required to be nonnegative.

Push Operation

PUSH(u, v)

// **Applies when**: u is overflowing, $c_f(u, v) > 0$, and $u.h = v.h + 1$.
// **Action:** Push $\Delta_f(u, v) = \min(u.e, c_f(u, v))$ units of flow from u to v.
$\Delta_f(u, v) = \min(u.e, c_f(u, v))$
if $(u, v) \in E$
    $(u, v).f = (u, v).f + \Delta_f(u, v)$
else $(v, u).f = (v, u).f - \Delta_f(u, v)$
$u.e = u.e - \Delta_f(u, v)$
$v.e = v.e + \Delta_f(u, v)$

Relabel Operation

RELABEL(u)

// **Applies when:** u is overflowing and for all $v \in V$ such that $(u, v) \in E_f$, we have $u.h \leq v.h$.

// **Action:** Increase the height of u.

$u.h = 1 + \min\{v.h : (u, v) \in E_f\}$

Preflow vs Augmenting Path

Augmenting Path Algorithm

Flow into i = Flow out of i

Push flow along a path from s to t

d(j) = distance from j to t in the residual network.

Preflow Algorithm

Flow into i ≥ Flow out of i for i ≠ s.

Push flow in one arc at a time

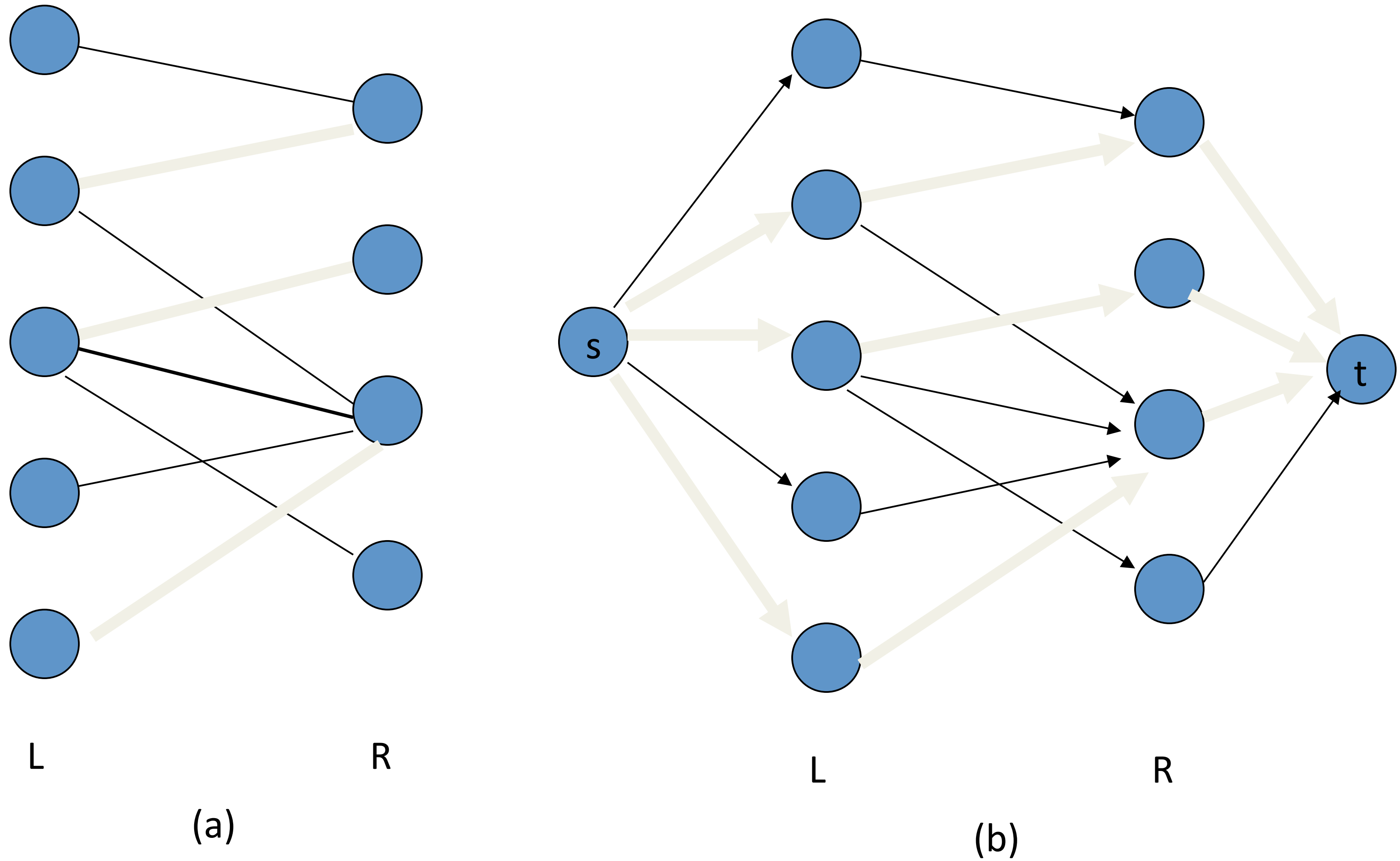
d(j) ≤ distance from j to t in the residual network

- d(t) = 0
- d(i) ≤ d(j) + 1 for each arc (i, j) ∈ G(x),

Finding a maximum bipartite matching:

- We define the corresponding flow network G’ =(V’ ,E’) for the bipartite graph G as follows. Let the source s and sink t be new vertices not in V, and let V’ =V∪{s,t}.If the vertex partition of G is V=L ∪R, the directed edges of G’ are given by E’ ={(s,u):u∈L} ∪{(u,v):u ∈L,v ∈R,and (u,v) ∈E} ∪{(v,t):v ∈R}.Finally, we assign unit capacity to each edge in E’ .

- We will show that a matching in G corresponds directly to a flow in G’ s corresponding flow network G’ . We say that a flow f on a flow network G=(V,E) is integer-valued if f(u,v) is an integer for all (u,v) ∈V*V.



(a)The bipartite graph G=(V,E) with vertex partition V=L∪R. A maximum matching is shown by shaded edges.(b) The corresponding flow network.Each edge has unit capacity.Shaded edges have a flow of 1,and all other edges carry no flow.

THE END