1 Flow Networks

1. Introduction
2. Maximum Flow
   a. The Ford-Fulkerson method
   b. The MAXFLOW-MINCUT theorem
   c. The Edmonds-Karp algorithm
3. Applications
   a. Connectivity of graphs
   b. Matchings in bipartite graphs

1. Introduction

Definition 1 *A <u>network</u> $N = (V, E, s, t)$ is an oriented graph (V, E) with a weight function $c : E \mapsto \mathbf{R}^{\geq 0}$ and two special nodes $s, t \in V$ (<u>source</u> and <u>sink</u>).*

If $(u, v) \notin E$ we extend $c(u, v)$ by setting $c(u, v) = 0$.

Definition 2 *Let $N = (V, E, s, t)$ be a network. A <u>flow</u> in N is a function $f : V \times V \mapsto \mathbf{R}$, such that:*

- $f(u, v) \leq c(u, v)$ *for any $u, v \in V$.* *(capacity constraint)*
- $f(u, v) = -f(v, u)$ *for any $u, v \in V$.* *(symmetry)*
- $\sum_{v \in V} f(u, v) = 0$ *for any $u \in V - \{s, t\}$.* *(flow conservation)*

The number $|f| = \sum_{v \in V} f(s, v)$ is called the <u>value</u> on f.

Therefore, $f(u, u) = 0$ and
if $(u, v) \notin E$ & $(v, u) \notin E \Rightarrow f(u, v) = f(v, u) = 0$.

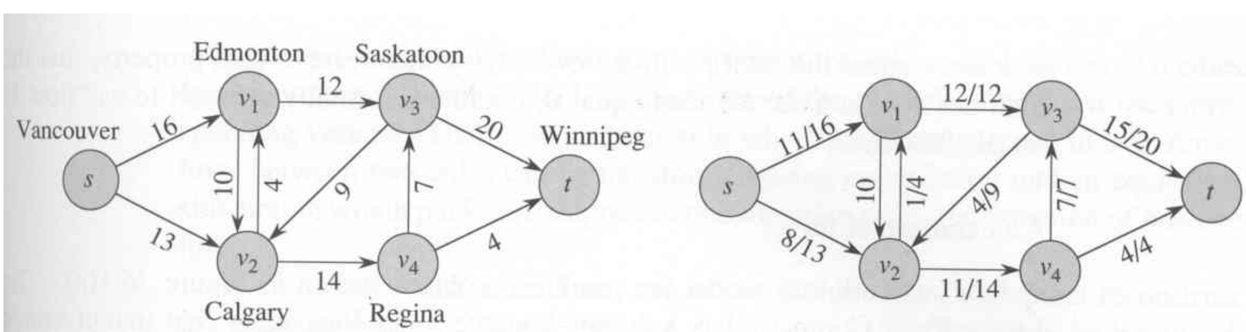

Figure 1: Example of a flow network

The Problem:

Given a network N construct a flow f for N with maximum value $|f|$ (maximal flow). Important ideas:

- The residual network
- The augmenting path
- The minimum cut

Definition 3 *Let $N = (V, E, s, t)$ be a network with a flow f. For each $u, v \in V$ put*

$$c_f(u, v) = c(u, v) - f(u, v)$$

and define the <u>residual network</u> $N_f = (V, E_f, s, t)$, where

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

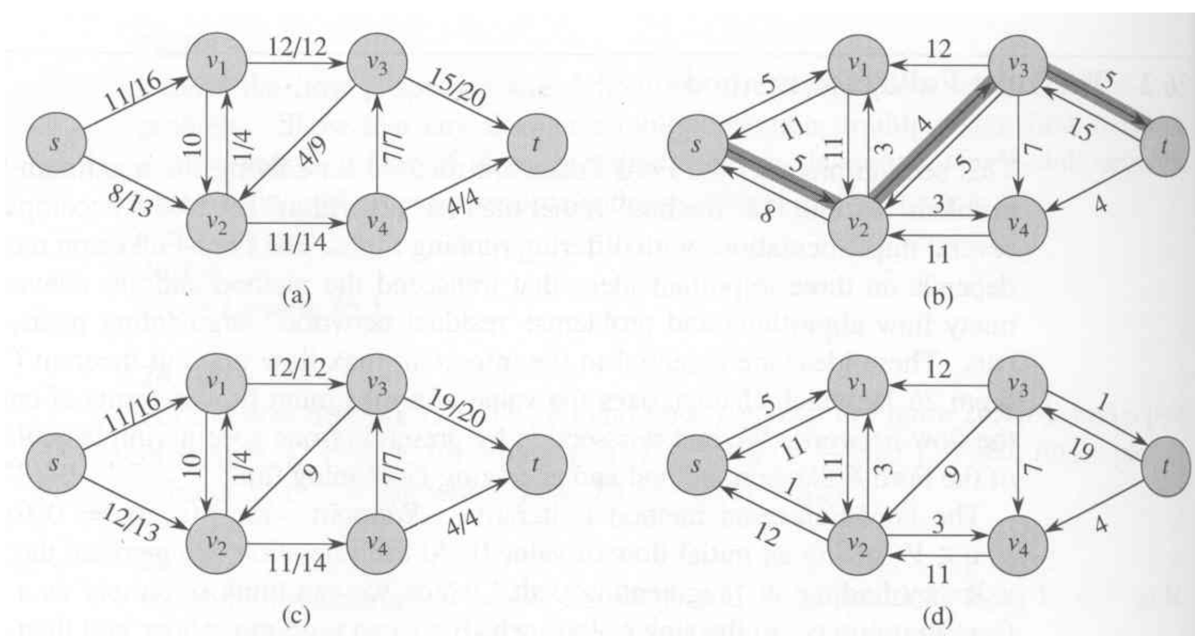

Figure 2: Residual network and augmenting path

Lemma 1 *Let $N = (V, E, s, t)$ a network with flow f and let f' be a flow on N_f. Then $f + f'$ is a flow on N and $|f + f'| = |f| + |f'|$.*

Proof:
Obviously, it holds: $(f + f')(u, v) = -(f + f')(v, u)$:

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

Since $f(u, v) \leq c(u, v)$ and $f'(u, v) \leq c_f(u, v)$, one has:

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

Therefore: $(u \in V - \{s, t\})$

$$\begin{aligned}\sum_{v \in V} (f + f')(u, v) &= \sum_{v \in V} (f(u, v) + f'(u, v)) \\ &= \sum_{v \in V} f(u, v) + \sum_{v \in V} f'(u, v) = 0.\end{aligned}$$

Finally:

$$\begin{aligned}|f + f'| &= \sum_{v \in V} (f(s, v) + f'(s, v)) \\ &= \sum_{v \in V} f(s, v) + \sum_{v \in V} f'(s, v) = |f| + |f'|. \qquad \square\end{aligned}$$

2a. The Ford-Fulkerson method

Definition 4 *Let* $N = (V, E, s, t)$ *be a network with a flow* f*. The* *augmenting path* p *is an oriented path* $s \rightsquigarrow t$ *in* N_f*.*
We put $c_f(p) := \min\{c_f(u, v) \mid (u, v) \in E(p)\}$.

Lemma 2 *Let* $N = (V, E, s, t)$ *be a network with a flow* f *and let* p *be an augmenting path in* N_f*. Define:*

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

Then f_p *is a flow on* N_f *and* $|f_p| = c_f(p) > 0$*.*

Corollary 1 *Let* $N = (V, E, s, t)$ *be a network with a flow* f *and let* p *be an augmenting path in* N_f*. Furthermore, let* $f' = f + f_p$*. Then* f' *is a flow with* $|f'| > |f|$*.*

A general method:

1. Set $f := 0$.
2. **while** $\exists$augmenting path p in N_f
   $f := f + f_p$.
3. **return** f

2b. The MAXFLOW-MINCUT theorem

Definition 5 *A* <u>*cut*</u> (S,T) *in a network* $N = (V, E, s, t)$ *is a partition of the vertex set* $V' = S \cup T$*, such that* $s \in S$*,* $t \in T$ *and* $S \cap T = \emptyset$*.*

We define:

$$\begin{aligned} f(S,T) &= \sum_{u \in S} \sum_{v \in T} f(u,v) \\ c(S,T) &= \sum_{u \in S} \sum_{v \in T} c(u,v). \end{aligned}$$

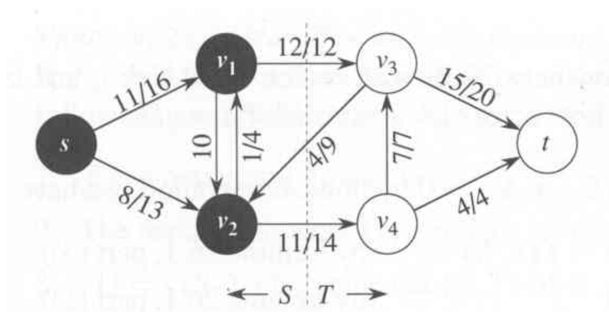

Figure 3: An (S,T)-cut in a network

Lemma 3 *Let* $N = (V, E, s, t)$ *be a network with a flow* f *and let* (S,T) *be a cut in* N*. Then* $f(S,T) = |f|$*.*

Proof:

$$\begin{aligned} f(S,T) &= \sum_{u \in S} \sum_{v \in T} f(u,v) = \sum_{u \in S} \sum_{v \in V} f(u,v) - \sum_{u \in S} \sum_{v \in S} f(u,v) \\ &= \sum_{u \in S} \sum_{v \in V} f(u,v) \\ &= \sum_{v \in V} f(s,v) + \sum_{u \in S - s} \sum_{v \in V} f(u,v) \\ &= \sum_{v \in V} f(s,v) = |f|. \qquad \square \end{aligned}$$

Theorem 1 (MAXFLOW-MINCUT THEOREM)
Let $N = (V, E, s, t)$ be a network with a flow f. The following statements are equivalent:

1. *f is a flow with maximum value $|f|$.*
2. *The network N_f has no augmenting path.*
3. *$|f| = c(S, T)$ for some cut (S, T) of N.*

Proof:

(1)⇒(2):
Assume N_f contains an augmenting path p.
(Corollary 1) $\Rightarrow |f + f_p| > |f|$, a contradiction

(2)⇒(3):
Denote:

$$S = \{v \in V \mid \exists \text{path } s \rightsquigarrow v \text{ in } N_f\}, \quad T = V - S.$$

Then (S, T) is a cut and $f(u, v) = c(u, v)$ for $u \in S$ and $v \in T$.
(Lemma 3) $\Rightarrow |f| = f(S, T) = c(S, T)$.

(3)⇒(1): Let (S, T) be a cut.

$$\begin{aligned} |f| = f(S,T) &= \sum_{u \in S} \sum_{v \in T} f(u, v) \\ &\leq \sum_{u \in S} \sum_{v \in T} c(u, v) = c(S, T). \end{aligned}$$

Since $[|f| = c(S, T)] \Rightarrow |f|$ is maximum. □

2c. The Ford-Fulkerson algorithm

Algorithm 1 FORD-FULKERSON(N, s, t);

for all $(u, v) \in E$ **do**
  $f[u, v] := 0$
  $f[v, u] := 0$

while ($\exists$augmenting path p in N_f) **do**
  $c_f(p) := \min\{c_f(u, v) \mid (u, v) \in p\}$
  for all $(u, v) \in p$ **do**
    $f[u, v] := f[u, v] + c_f(p)$
    $f[v, u] := -f[u, v]$

If all the costs $c(u, v)$ are integer numbers then the running time of FORD-FULKERSON is $O(|E| \cdot |f^*|)$, where $|f^*|$ is the value of the maximum flow constructed by the algorithm.

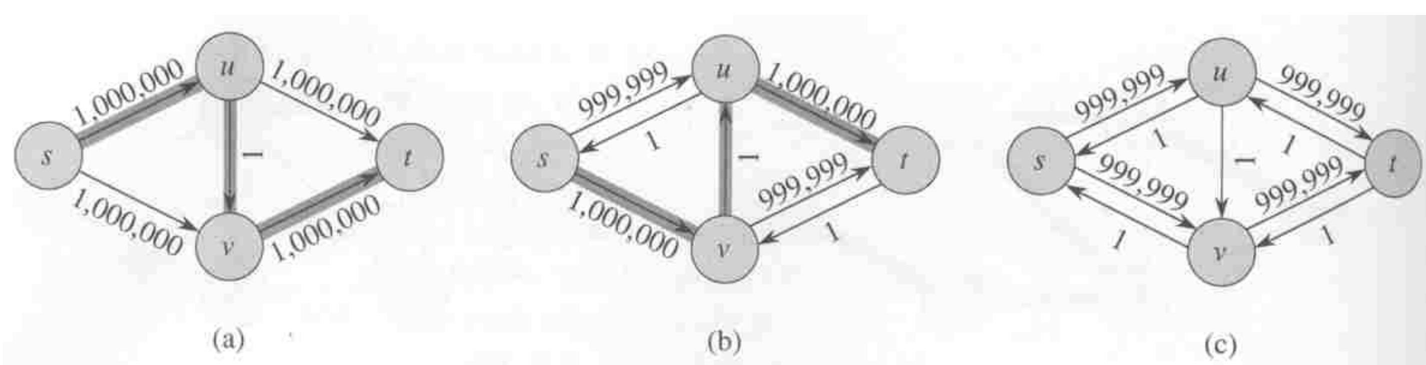

Figure 4: A very slow termination of the basic method

Example 1

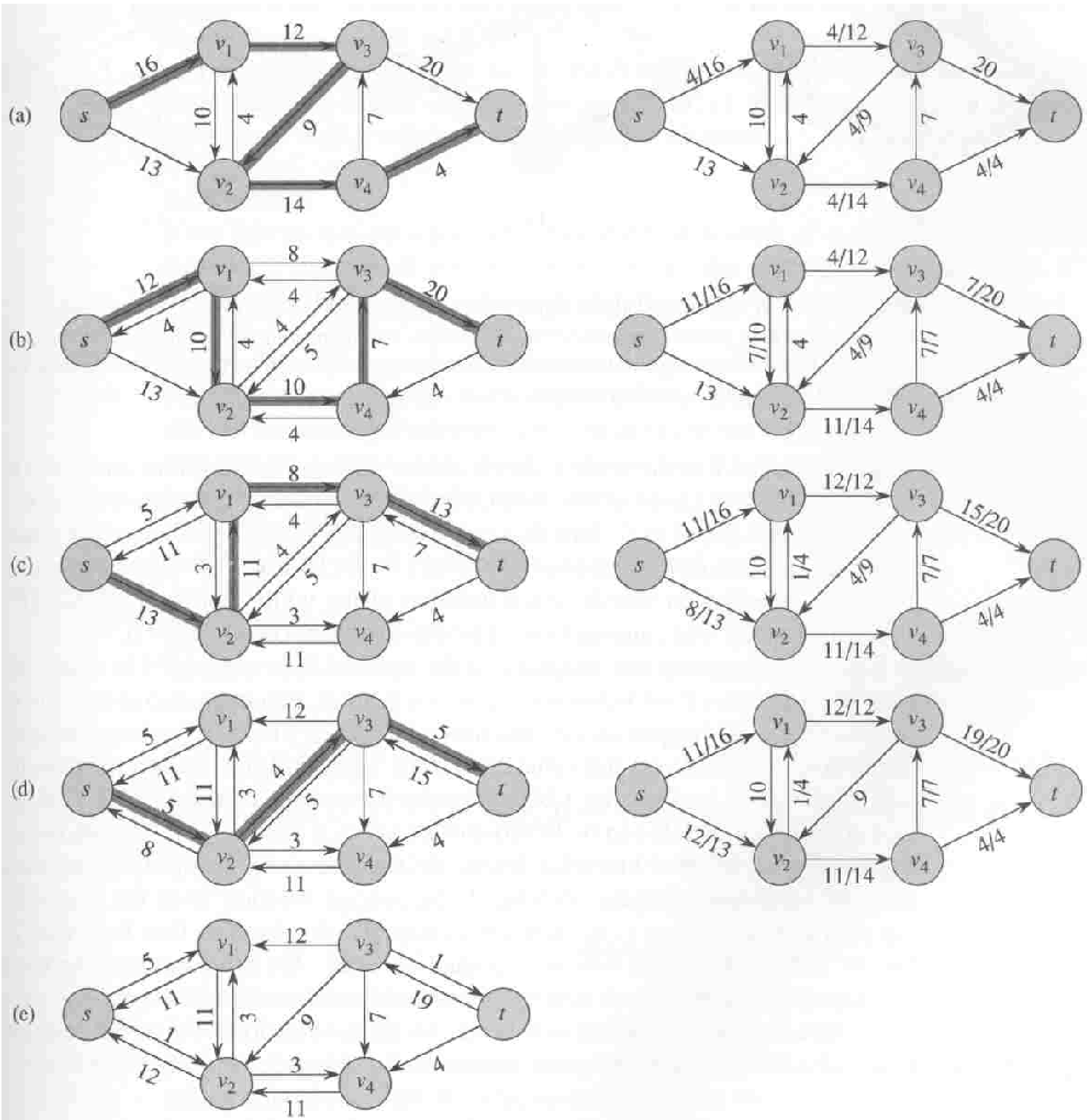

Figure 5: Execution of the basic Ford-Fulkerson algorithm

2c. The Edmonds-Karp algorithm

Let all the costs $c(u, v)$ be rational. Furthermore, assume that any augmenting path p constructed by $\textsc{Ford-Fulkerson}$ is a shortest path $s \leadsto t$ in N_f constructed by the DFS. So implemented Ford-Fulkerson method is called $\textsc{Edmonds-Karp}$ algorithm.

Let $\delta_f(u, v)$ denote the length of the shortest path $u \leadsto v$ in N_f (w.r.t. the weights $w(e) = 1$ for every $e \in E$).

Lemma 4 *After every execution of the* **while** *-loop in the* $\textsc{Edmonds-Karp}$ *algorithm the values* $\delta_f(s, v)$ *are increasing monotonically for all* $v \in V - \{s, t\}$.

Proof:
Assume the contrary and let v be the first vertex, s.t. $\delta_{f'}(s, v) < \delta_f(s, v)$, where the flow f' is obtained from f by a single augmentation.

Let $p = s \leadsto u \to v$ be a shortest path in $N_{f'}$, so $(u, v) \in E_{f'}$ and

$$\delta_{f'}(s, u) = \delta_{f'}(s, v) - 1 \qquad (1)$$

By the choice of u:

$$\delta_{f'}(s, u) \geq \delta_f(s, u). \qquad (2)$$

We claim $(u, v) \notin E_f$ because otherwise

$$\begin{aligned} \delta_f(s, v) &\leq \delta_f(s, u) + 1 \\ &\leq \delta_{f'}(s, u) + 1 \qquad \text{(by (2))} \\ &= \delta_{f'}(s, v) \qquad \text{(by (1))}. \end{aligned}$$

Now, $(u, v) \notin E_f$ and $(u, v) \in E_{f'}$.
$\Rightarrow$ $\exists$shortest path $s \rightsquigarrow v \rightarrow u \rightsquigarrow t$ in N_f. One has

$$\begin{aligned} \delta_f(s, v) &= \delta_f(s, u) - 1 \\ &\leq \delta_{f'}(s, u) - 1 \qquad \text{(by (2))} \\ &= \delta_{f'}(s, v) - 2 \qquad \text{(by (1))} \end{aligned}$$

which contradicts to $\delta_{f'}(s, v) < \delta_f(s, v)$. □

Theorem 2 *The number of times the* **while** *-loop in the* EDMONDS-KARP *algorithm is executed is* $O(|V| \cdot |E|)$.

Proof:
After each augmentation the distance from s to at least one vertex strictly increases.

For any $v \in V - \{s, t\}$ one has $\delta_f(s, v) \in \{1, \ldots, |E|, \infty\}$, so $\delta_f(s, v)$ takes on at most $|E|+1$ values in the coarse of the algorithm.

By L. 4 the $\delta_f(s, v)$ does not decrease. So the number of times the **while** -loop is executed does not exceed the number of steps to lift all $\delta_f(s, v)$ up to their final values. This number does not exceed $|E| + 1$ for each vertex v, so the total number of steps is not larger than $|V| \cdot (|E| + 1) = O(|V| \cdot |E|)$. □

Therefore, the running time of the EDMONDS-KARP algorithm is $O(|V| \cdot |E|^2)$.

3a. Connectivity of graphs

Definition 6 *An undirected graph $G = (V, E)$ is called κ-connected if for any two its vertices $u, v \in V$ ($u \neq v$) there exist κ vertex disjoint paths $u \leadsto v$.*

Definition 7 *Let $G = (V, E)$ be a non-oriented graph and $a, b \in V$. A set $S \subseteq V$ is called (a, b) vertex separator, if $\{a, b\} \subset V - S$ and any path $a \leadsto b$ in G contains a vertex of S.*

$N(a, b) :=$ minimum size of an (a, b) vertex separator.
$p(a, b) :=$ maximum size of a set of vertex-disjoint paths $a \leadsto b$.

Theorem 3 (Menger's Theorem)
If $(a, b) \notin E$ then $N(a, b) = p(a, b)$.

Proof: (using flows)
Given $G = (V, E)$, construct the following network $N_G = (V', E', s, t)$:

- $\forall v \in V$: put v', v'' in V' and the edge (v', v'') (internal edges). Set: $s = a''$ and $t = b'$.
- $\forall (u, v) \in E$: put (u'', v') and (v'', u') in N_G. (external edges)

Also set:

$$c(u, v) = \begin{cases} 1, & \text{for all internal edges} \\ \infty, & \text{for all external edges} \end{cases}$$

We show that for a maximal flow f from s to t one has:

$$|f| = p(a, b).$$

There exist $p(a, b)$ vertex-disjoint paths $a \rightsquigarrow b$ in G:

$$a \rightsquigarrow v_1 \rightsquigarrow \cdots \rightsquigarrow v_l = b.$$

$\Rightarrow$ there exist $p(a, b)$ vertex-disjoint paths $s \rightsquigarrow t$ in N_G:

$$s = a'' \rightsquigarrow v_1' \rightsquigarrow v_1'' \rightsquigarrow \cdots \rightsquigarrow v_l' \rightsquigarrow v_l'' = t$$

$\Rightarrow |f| \geq p(a, b)$.

Assume f is a maximal flow in N_G s.t. $f(e) \in \{0, 1\}$ $\forall e \in E'$.
$\Rightarrow$ each $s \rightsquigarrow t$-path contributes 1 to f
$\Rightarrow$ there exist at least $|f|$ paths, i.e. $p(a, b) \geq |f|$. □

Menger's Theorem leads to a method for computing $N(a, b)$:

1. Construct the network N_G as in Theorem 3.
2. Find a maximal flow f in N_G from s to t.
3. Find the minimum-size edge-cut C in N_G.
4. Construct corresponding (a, b) vertex separator from C in G.

κ vertex connectivity:
For all pairs (a, b) of vertices with $(a, b) \notin E$ compute the minimal vertex separator and find κ.

Edge-connectivity of Graphs

Definition 8 *A non-oriented graph $G = (V, E)$ is called κ edge-connected if for any two its vertices $u, v \in V$ ($u \neq v$) there exist κ edge-disjoint paths.*

Definition 9 *Let $G = (V, E)$ be an undirected graph and $a, b \in V$. A set $S \subseteq E$ is called (a, b) edge-separator, if any path $a \rightsquigarrow b$ contains an edge of S.*

$K(a, b) :=$ minimum size of (a, b)-separator.
$q(a, b) :=$ maximum size of a set of edge-disjoint paths $a \rightsquigarrow b$.

Theorem 4 *It holds: $K(a, b) = q(a, b)$.*

Proof:
Given $G = (V, E)$, construct the network $N_G = (V, E', a, b)$, where for each edge $(u, v) \in E$ there are two oriented edges (a, b) and (b, a). Set the capacity for each edge to be 1.

Let f be a maximal flow in N_G. Then $|f| = q(a, b)$. $\square$

Theorem 4 leads to computing $K(a, b)$ via computing $q(a, b)$. We apply this Method for all $a, b \in V$ and find the κ edge-connectivity.

3b. Matching in bipartite graphs

Definition 10 *Let $G = (V, E)$ be an undirected graph. A set $M \subseteq E$ is called* <u>*matching*</u> *if the edges in M have no common vertices. A matching is called* <u>*maximum*</u> *if it has a maximum number of edges.*

The Problem:
Given a bipartite graph $G = (V_1 \cup V_2, E)$, construct a maximum matching.

Let $N_G = (V', E', s, t)$ be the following network:

$$
\begin{aligned}
V' &= V_1 \cup V_2 \cup \{s, t\} \\
E' &= \{(s, u) \mid u \in V_1\} \\
&\quad \cup \{(u, v) \mid u \in V_1,\ v \in V_2,\ (u, v) \in E\} \\
&\quad \cup \{(v, t) \mid v \in V_2\} \\
c(u, v) &= 1 \quad \text{for every } (u, v) \in E'
\end{aligned}
$$

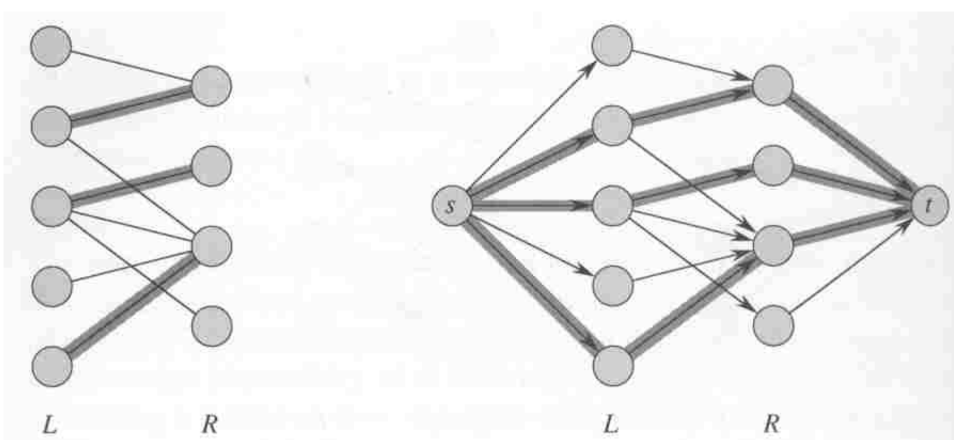

Figure 6: Maximum matching in bipartite graphs

Theorem 5 *Let $G = (V_1 \cup V_2, E)$ be a bipartite graph. For any matching M there is a flow f in N_G with $|f| = |M|$. For any integer-valued flow f in N_G there exists a matching M in G with $|M| = |f|$.*

Proof:
Let M be a matching in G. Consider the following flow:

$$\begin{aligned} f(s,u) &= f(u,v) = f(v,t) = 1 \\ f(u,s) &= f(v,u) = f(t,v) = -1 \end{aligned}$$

for $(u,v) \in M$ and $f(u,v) = 0$ otherwise.
For any $(u,v) \in M$ there exists a path $s \leadsto u \leadsto v \leadsto t$.
Let $S = \{s\} \cup V_1$ and $T = \{t\} \cup V_2$.
Then (S,T) is a cut in N_G and (Lemma 3) $|f| = |M|$.

Let f be an integer-valued flow in N_G. Consider

$$M = \{(u,v) \mid u \in V_1,\, v \in V_2,\ f(u,v) > 0\}.$$

Then M is a matching with $|M| = |f|$. $\square$

Corollary 2 *The size of a maximum matching in G equals to the value of a maximum flow in in N_G.*

Proof:
Let M be a maximum matching. Consider the flow f as in the proof of Theorem 5.
Assume f is not maximum. $\Rightarrow \exists$ an integer-valued flow f' in N_G with $|f'| > |f|$.
(Theorem 5) $\Rightarrow$ there is a matching M' in G with

$$|M'| = |f'| > |f| = |M|.$$

So, M is not a maximum matching, a contradiction. $\square$

Let $G = (V, E)$ be a graph and $A \subseteq V$. Define:

$$N(A) = \{v \in V - A \mid (v, w) \in E,\ w \in A\}.$$

Theorem 6 (P. Hall)
Let $G = (V_1 \cup V_2, E)$ be a bipartite graph. G has a matching M with $|M| = |V_1| \iff$ for any subset $A \subseteq V_1$ one has:

$$|N(A)| \geq |A|.$$

Proof:
$\Rightarrow$ Obvious.

$\Leftarrow$ Let $|N(A)| \geq |A|$ for any $A \subseteq V_1$. Let f be a maximal integer-valued flow in N_G and $S \subseteq V'$ be the vertices of the residual network which are reachable from s. Then $|f| = c(S, T)$, where $T = V(N_G) - S$.

Let $v \in S \cap V_1$ and $(v, w) \in E(N_G)$. We show $w \in S$.
Assume $w \notin S \Rightarrow f(v, w) = 1$ (otherwise $w \in S$).
$f(s, v) = 0 \Rightarrow \sum_{u \in V(N_G)} f(v, u) \neq 0$, a contradiction.

Hence: $N(S \cap V_1) \subseteq S$. Therefore:
$(v, w) \in E(V_G)$, $v \in S$, $w \in T \Rightarrow v = s$ or $w = t$.
Since $S \cap V_2 = N(S \cap V_1)$, then:

$$|f| = |V_1 - S| + |N(S \cap V_1)| \geq |V_1 - S| + |S \cap V_1| = |V_1| \geq |f|.$$

(Theorem 5) $\Rightarrow G$ has a matching with $|V_1|$ edges. $\square$

Corollary 3 *Any regular bipartite graph has a perfect matching.*