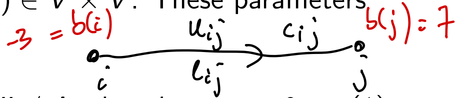
Network flows and their applications, Shandong summer school, July 2025

Jørgen Bang-Jensen

University of Southern Denmark

What is a Network?

A **network** is a directed graph $D = (V, A)$ associated with the following functions on $V \times V$: a **lower bound** $l_{ij} \geq 0$, a **capacity** $u_{ij} \geq l_{ij}$ and a **cost** c_{ij} for each $(i, j) \in V \times V$. These parameters satisfy the following requirement:



$$\text{For every } (i, j) \in V \times V, \text{ if } ij \notin A, \text{ then } l_{ij} = u_{ij} = 0. \quad (1)$$

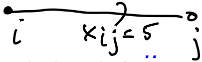
In some cases we also have a function $b : V \to \mathcal{R}$ called a **balance vector** which associates a real number with each vertex of D and satisfies that

$$\sum_{v \in V} b(v) = 0. \quad (2)$$

We use the shorthand notation $\mathcal{N} = (V, A, l, u, b, c)$ to denote a network with corresponding digraph $D = (V, A)$ and parameters l, u, b, c. If there are no costs specified, or there is no prescribed balance vector, then we omit the relevant letters from the notation.

What is a flow?

A **flow** in a network $\mathcal{N}$ is a function $x : A \to \mathcal{R}_0$ on the arc set of $\mathcal{N}$. We denote the value of x on the arc ij by x_{ij}.

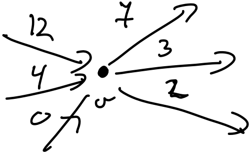


An **integer flow** in $\mathcal{N}$ is a flow x such that $x_{ij} \in \mathcal{Z}_0$ for every arc ij

For a given flow x in $\mathcal{N}$ the **balance vector of** x is the following function b_x on the vertices:

$$b_x(v) = \sum_{vw \in A} x_{vw} - \sum_{uv \in A} x_{uv} \quad \forall v \in V. \tag{3}$$

That is, $b_x(v)$ is the difference between the flow on arcs with tail v and the flow on arcs with head v.



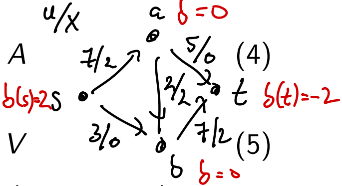
$b_x(v) = -4$

We say that the flow x is a **feasible flow** in $\mathcal{N} = (V, A, \ell, u, b, c)$ if

$$\ell_{ij} \le x_{ij} \le u_{ij} \text{ for all } ij \in A \tag{4}$$

and

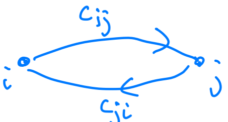
$$b_x(v) = b(v) \text{ for all } v \in V \tag{5}$$



If no balance vector is specified, that is $\mathcal{N} = (V, A, \ell, u, c)$ then a flow x is feasible if (4) holds.

Theorem 1 (Integrality Theorem for flows)

Let $\mathcal{N} = (V, A, l, u, b, c)$ be a network in which all lower bounds and capacities are non-negative integers. There exists a feasible flow in $\mathcal{N}$ if and only if there exists a feasible integer flow in $\mathcal{N}$.



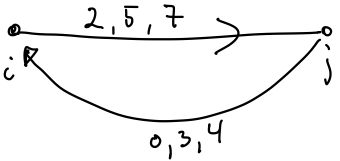
$c_{ij} = -c_{ji}$

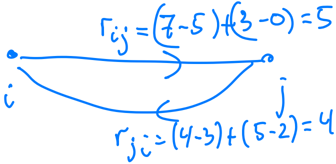
For a given flow x in a network $\mathcal{N} = (V, A, l, u, c)$, define the **residual capacity** r_{ij} from i to j as follows:

$$r_{ij} = (u_{ij} - x_{ij}) + (x_{ji} - l_{ji}). \tag{6}$$

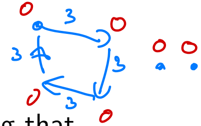
The **residual network** $\mathcal{N}(x)$ with respect to x is defined as $\mathcal{N}(x) = (V, A(x), \tilde{l} \equiv 0, r, c)$, where $A(x) = \{ij : r_{ij} > 0\}$. That is, the cost function is the same as for $\mathcal{N}$ and all lower bounds are zero.

ℓ, x, u





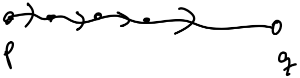
Special flows

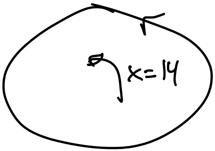


Let $\mathcal{N} = (V, A, \ell, u, c)$ be a network.

- A **circulation** in $\mathcal{N}$ a network is a flow x satisfying that $b_x(v) = 0$ for every $v \in V$
- An $(\mathbf{s}, \mathbf{t})$**-flow** in $\mathcal{N}$ is a flow x satisfying that $b_x(v) = 0$ for every $v \in V - \{s, t\}$ and $b_x(t) = -b_x(s)$ where $b_x(s) \geq 0$.
- A **path flow** $f(P)$ along a path P in $\mathcal{N}$ is a flow with the property that there is some number $k \in \mathcal{R}_0$ such that $f(P)_{ij} = k$ if ij is an arc of P and otherwise $f(P)_{ij} = 0$.
- Analogously, we can define a **cycle** flow $f(W)$ for any cycle W in D.

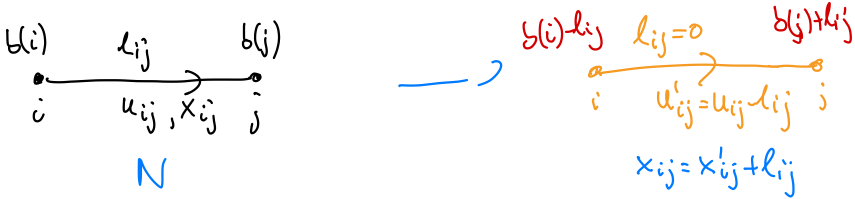
x = c e.g x = 15





Lemma 2

Let $\mathcal{N} = (V, A, l, u, b, c)$ be a network. There exists a network $\mathcal{N}_{l \equiv 0}$ in which all lower bounds are zero such that every feasible flow x in $\mathcal{N}$ corresponds to a feasible flow x' in $\mathcal{N}_{l \equiv 0}$ and vice versa. Furthermore, the costs of these two flows are related by $c^T x = c^T x' + \sum_{ij \in A} l_{ij} c_{ij}$.



Reducing general balances to an (s,t)-flow problem

Lemma 3

Let $\mathcal{N} = (V, A, l \equiv 0, u, b, c)$ be a network. Let $M = \sum_{\{v:b(v)>0\}} b(v)$ and let $\mathcal{N}_{st} = (V \cup \{s,t\}, A', l' \equiv 0, u', b', c')$, where

(a) $A' = A \cup \{sr : b(r) > 0\} \cup \{rt : b(r) < 0\}$,

(b) $u'_{ij} = u_{ij}$ *for all* $ij \in A$, $u_{sr} = b(r)$ *for all* r *such that* $b(r) > 0$ *and* $u_{qt} = -b(q)$ *for all* q *such that* $b(q) < 0$,

(c) $c'_{ij} = c_{ij}$ *for all* $ij \in A$ *and* $c' = 0$ *for all arcs leaving* s *or entering* t,

(d) $b'(v) = 0$ *for all* $v \in V$, $b'(s) = M$, $b'(t) = -M$.

Then every feasible flow x *in* $\mathcal{N}$ *corresponds to a feasible flow* x' *in* $\mathcal{N}_{st}$ *and vice versa. Furthermore, the costs of* x *and* x' *are related by* $c^T x = c'^T x'$.

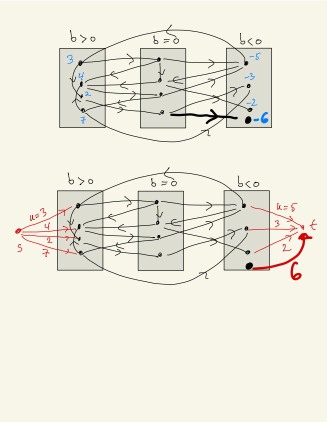


Figure: Reducing networks with general balances to an (s, t)-flow problem.

Reducing (s,t)-flow to a circulation problem

Lemma 4

Let $\mathcal{N} = (V, A, l \equiv 0, u, b, c)$ be a network with distinct vertices s, t and let the balance vector of $\mathcal{N}$ satisfy $b(v) = 0$ for all $v \in V - \{s, t\}$, $b(s) = M$, $b(t) = -M$, for some $M \in \mathcal{R}_0$. Let $\mathcal{N}^* = (V, A \cup \{ts\}, l'', u'', c'')$ be the network obtained from $\mathcal{N}$ by adding a new arc ts with lower bound $l_{ts} = M$, capacity $u_{ts} = M$ and cost $c''_{ts} = 0$, keeping the lower bound, capacity and cost of each original arc and posing no restriction on the balance vector of $\mathcal{N}^*$. Then every feasible (s, t)-flow x in $\mathcal{N}$ corresponds to a feasible circulation x'' in $\mathcal{N}^*$ and vice versa. Furthermore, the costs of x and x' are related by $c^T x = c''^T x''$.

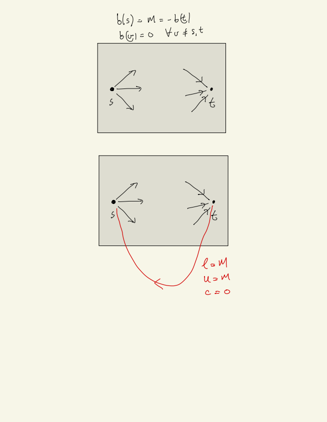


Figure: Reducing an (s, t)-flow problem to a circulation problem

Bounds on vertices and the vertex splitting technique

Sometimes it is useful to add lower and/or upper bound the the amount of flow that may pass through each vertex v in a network. This is useful

- when we are looking for vertex disjoint objects. Here we can use upper bounds to ensure this.
- when we want to ensure that the vertex v is included in the structure we find via flows. Here we can use a lower bounds on v

Lemma 5

For every network $\mathcal{N}^*$ with bounds on both vertices and arcs there exists an equivalent network $\mathcal{N}$ such that every feasible flow in $\mathcal{N}^*$ corresponds to a feasible flow in $\mathcal{N}$ of the same cost.

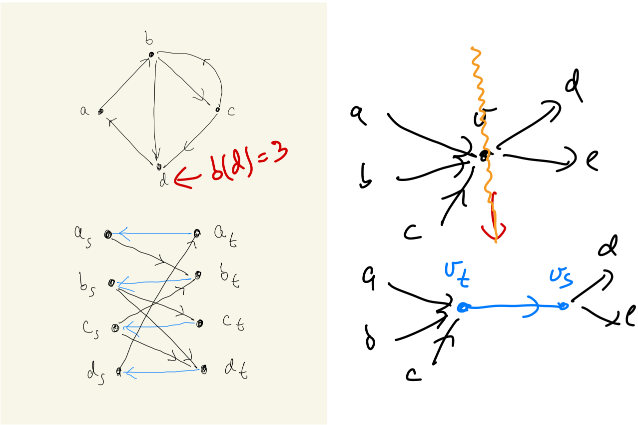
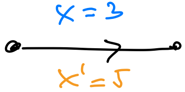


Figure: The vertex splitting procedure.

Decomposing a flow

The **arc sum** of two flows x, x', denoted $x + x'$, is simply the flow obtained by adding the two flows arc-wise, that is, $(x + x')_{ij} = x_{ij} + x'_{ij}$.



Theorem 6

Every flow x in $\mathcal{N}$ can be represented as the arc sum of some path and cycle flows $f(P_1), f(P_2), \ldots, f(P_\alpha), f(C_1), \ldots, f(C_\beta)$ with the following two properties:

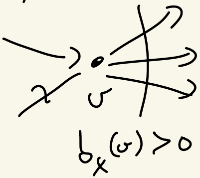
(a) Every directed path P_i, $1 \leq i \leq \alpha$ with positive flow connects a vertex with $b_x > 0$ to a vertex with $b_x < 0$.

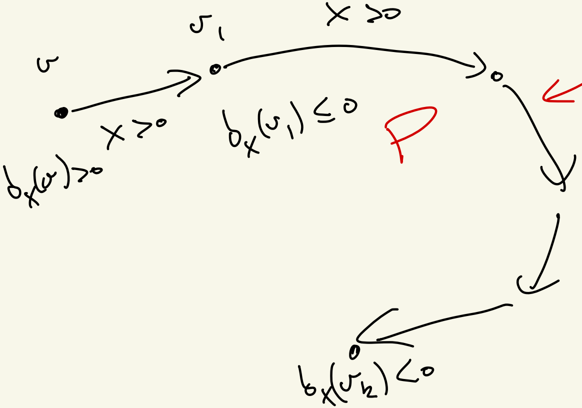
(b) $\alpha + \beta \leq n + m$ and $\beta \leq m$.

Furthermore, given an arbitrary flow x in $\mathcal{N}$ one can find a decomposition of x into at most $n + m$ path and cycle flows, at most m of which are cycle flows in time $O(nm)$.

start x feasible in $N=(V,A \dots)$

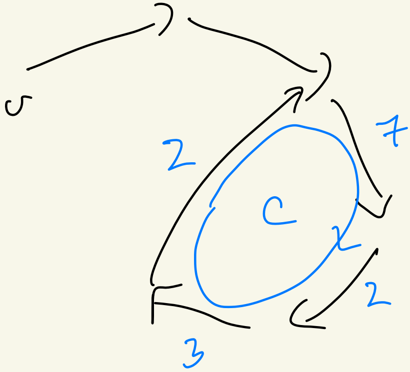
can 1 $\exists\, v \in V$ s.t $b_x(v) > 0$





min x value $= 3$

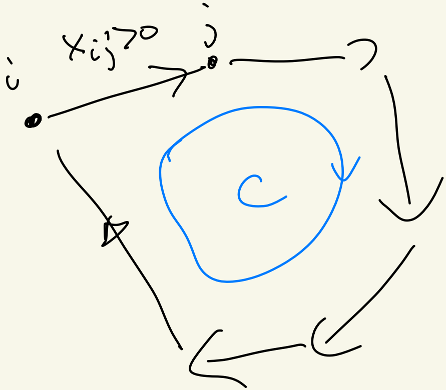
reduce x by 3 along P



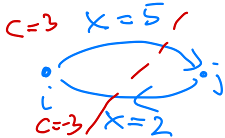
min x-value on C

= 2

Can 2 current x is a circulation

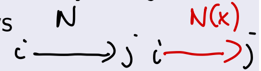


We always assume that if ij and ji are both arcs, then $\min\{x_{ij}, x_{ji}\} = 0$. Such a flow is called a **netto flow**



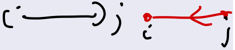
Definition 7

Let x be a feasible flow in $\mathcal{N} = (V, A, l \equiv 0, u, c)$ and let $\tilde{x}$ be a feasible flow in $\mathcal{N}(x)$. Define the flow $x^* = x \oplus \tilde{x}$ as follows: Start by letting $x^*_{ij} := x_{ij}$ for every $ij \in A$ and then for every arc ij in $\mathcal{N}(x)$ such that $\tilde{x}_{ij} > 0$ we modify x^* as follows



(a) If $x_{ji} = 0$ then $x^*_{ij} := x_{ij} + \tilde{x}_{ij}$.

(b) If $x_{ij} = 0$ and $x_{ji} < \tilde{x}_{ij}$ then $x^*_{ij} := \tilde{x}_{ij} - x_{ji}$ and $x^*_{ji} := 0$.



(c) If $x_{ji} \geq \tilde{x}_{ij}$ then $x^*_{ji} := x_{ji} - \tilde{x}_{ij}$.

Theorem 8

Let x be a feasible flow in $\mathcal{N} = (V, A, l \equiv 0, u, c)$ with balance vector b_x and $\tilde{x}$ is a feasible flow in $\mathcal{N}(x) = (V, A(x), r, c)$ with balance vector $b_{\tilde{x}}$. Then $x^* = x \oplus \tilde{x}$ is a feasible flow in $\mathcal{N}$ with balance vector $b_x + b_{\tilde{x}}$ and the cost of x^* is given by $c^T x^* = c^T x + c^T \tilde{x}$.

$b_x = 3$ s

$b_{\tilde{x}} = -2$ s

The next theorem shows that the difference between any two feasible flows in a network can be expressed as a feasible flow in the residual network with respect to any of those flows.

Theorem 9

Let $\mathcal{N} = (V, A, l \equiv 0, u, c)$ be a network and let x and x' be feasible netto flows in $\mathcal{N}$ with balance vectors b_x and $b_{x'}$. There exists a feasible flow $\bar{x}$ in $\mathcal{N}(x)$ with balance vector $b_{\bar{x}} = b_{x'} - b_x$ such that $x' = x \oplus \bar{x}$. Furthermore, the costs of these flows satisfy $c^T \bar{x} = c^T x' - c^T x$.

The following immediate corollary of Theorem 9 and Theorem 6 will be useful for instance when we study minimum cost flows.

Corollary 10

If x and x' are feasible flows in the network $\mathcal{N} = (V, A, l \equiv 0, u, c)$ such that $b_x = b_{x'}$, then there exist a collection of at most m cycles $W_1, W_2, \dots, W_k$ in $\mathcal{N}(x)$ and cycle flows $f(W_1), \dots, f(W_k)$ in $\mathcal{N}(x)$ such that the following holds:

(a) $x' = x \oplus (f(W_1) + \ldots + f(W_k)) = (\ldots((x \oplus f(W_1)) \oplus f(W_2)) \oplus \ldots) \oplus f(W_k);$

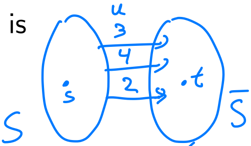
(b) $c^T x' = c^T x + \sum_{i=1}^{k} c^T f(W_i).$

Maximum flows

The **maximum flow problem** is the following

> Max flow
> **Input:** A network $\mathcal{N} = (V, A, \ell \equiv 0, u)$ and distinct vertices $s, t \in V$
> **Question:** Find the maximum k such that there exists a feasible (s,t)-flow in $\mathcal{N}$ with $b_x(s) = k$.

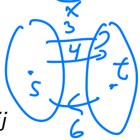
An $(\mathbf{s},\mathbf{t})$**-cut** is a set of arcs of the form $(S, \bar{S})$ such that $s \in S$ and $t \in \bar{S}$. The **capacity** of an the (s,t)-cut $(S, \bar{S})$ is



$$u(S, \bar{S}) = \sum_{\{ij \in A | i \in S, j \in \bar{S}\}} u_{ij}$$

.

Similarly, we let

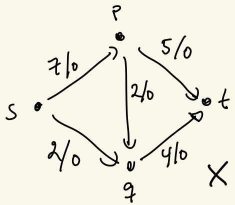


$$x(S, \bar{S}) = \sum_{\{ij \in A | i \in S, j \in \bar{S}\}} x_{ij} \text{ and } x(\bar{S}, S) = \sum_{\{ij \in A | i \in \bar{S}, j \in S\}} x_{ij}$$

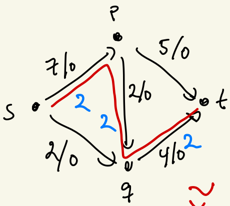
N

N(x)

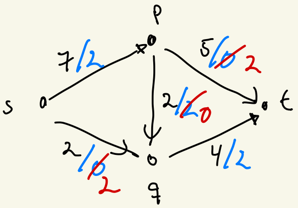
$r_{ij} = (u_{ij} - x_{ij}) + x_{ji}$



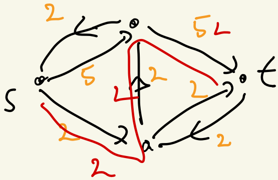
X



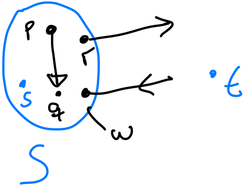
$\tilde{X}$



$X \oplus \tilde{X}$

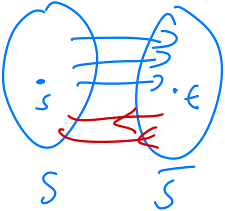


For every (s,t)-cut $(S,\bar{S})$:



$\sum_{v \in S} b_x(v)$

$$
\begin{aligned}
b_x(s) &= x(S,\bar{S}) - x(\bar{S},S) \\
&\leq u(S,\bar{S}) - \ell(\bar{S},S) \\
&\leq u(S,\bar{S})
\end{aligned}
$$



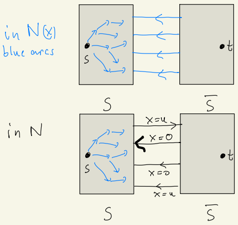
Theorem 11 (Max-flow Min-cut theorem)

Let $\mathcal{N} = (V, A, l \equiv 0, u)$ be a network with source s and sink t. For every feasible (s,t)-flow x in $\mathcal{N}$ the following are equivalent:

- *(a) The flow x is a maximum (s,t)-flow.*
- *(b) There is no (s,t)-path in $\mathcal{N}(x)$.*
- *(c) There exists an (s,t)-cut $(S,\overline{S})$ such that $b_x(s) = u(S,\overline{S})$.*

(b) $\Rightarrow$ (c) assume no (s,t)-path in $N(x)$

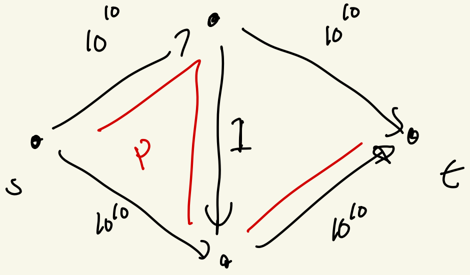
$S = \{v \mid \exists (s,v)\text{-path in } N(x)\}$



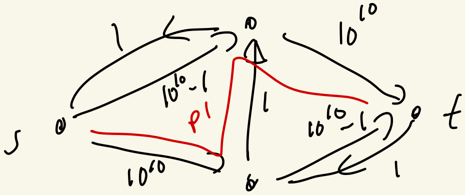
$$b_x(s) = x(S,\bar{S}) - x(\bar{S},S)$$
$$= u(S,\bar{S}) - 0 = u(S,\bar{S})$$

Figure: Proof of (b) $\Rightarrow$ (c) in the Max-flow Min-cut theorem

$2 \cdot 10^{10}$



$\delta(p) = 1$



$\delta(p') = 1$

Edmonds-Karp: shortest augmenting paths $\Rightarrow O(VE^2)$

Theorem 12 (Integrality theorem for maximum (s,t)-flows)

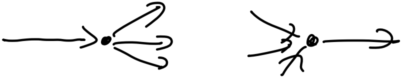
Let $\mathcal{N} = (V, A, l \equiv 0, u)$ be a network with source s and sink t. If all capacities are integers, then there exists an integer maximum (s,t)-flow in $\mathcal{N}$.

Theorem 13 (Orlin)

There exists an $O(nm)$ algorihm for finding a maximum (s,t)-flow in a given network $\mathcal{N} = (V, A, \ell \equiv 0, u)$ (here $n = |V|$, $m = |A|$).

Unit capacity (simple) networks

$u \equiv 1$



A **simple network** is a network $\mathcal{N} = (V, A, l \equiv 0, u)$ with special vertices s, t in which every vertex in $V - \{s, t\}$ has precisely one arc entering or precisely one arc leaving.

Theorem 14 (Max-Flow in unit capacity (simple) networks)

Let $\mathcal{N} = (V, A, \ell \equiv 0, u \equiv 1)$ and let $s, t \in V$ be distinct. One can find a maximum (s, t)-flow in $\mathcal{N}$ in time $O(n^{\frac{2}{3}} m)$ and in time $O(\sqrt{n} m)$ if $\mathcal{N}$ is also simple.

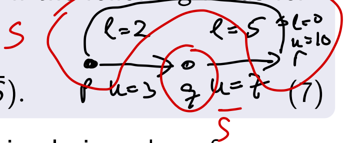
Corollary 15

One can find a maximum matching in a bipartite graph $G = (V, E)$ in time $O(\sqrt{|V|}|E|)$.

Theorem 16 (Hoffman's circulation theorem)

A network $\mathcal{N} = (V, A, l, u)$ with non-negative lower bounds on the arcs has a feasible circulation if and only if the following holds for every proper subset S of V:

$$l(\bar{S}, S) \leq u(S, \bar{S}). \tag{7}$$



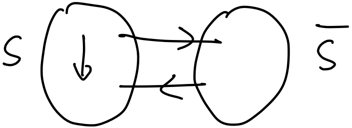
Proof: First note that if x is a feasible circulation, then, for every proper subset S of V we have

$\forall S \subset V$, $\neq \emptyset$

$$u(S, \bar{S}) \geq x(S, \bar{S}) = x(\bar{S}, S) \geq l(\bar{S}, S),$$

$x(S, \bar{S}) = x(\bar{S}, S)$

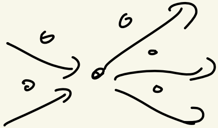
showing that (7) must hold.



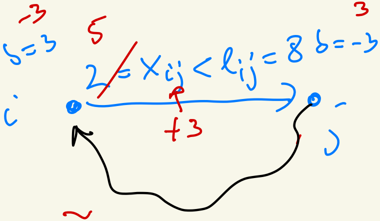
To prove the converse we assume that (7) holds for all $S \subset V$ and give an algorithmic proof showing how to construct a feasible circulation starting from the all-zero circulation.

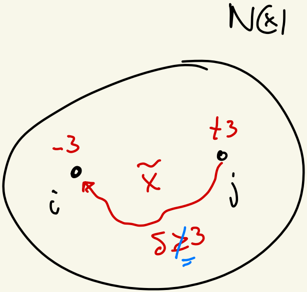
- Clearly $x \equiv 0$ is a circulation in $\mathcal{N}$ and if $l \equiv 0$, then we are done. So we may assume that $l_{ij} > x_{ij}$ for some $ij \in A$.
- We try to find a (j, i)-path in $\mathcal{N}(x)$. If such a path P exists, then we let $\delta(P) > 0$ be the minimum residual capacity of an arc on P. Let $\epsilon = \min\{\delta(P), l_{ij} - x_{ij}\}$.
- By Theorem 8 (which also holds when some lower bounds are non-zero), we can increase the current flow x by ϵ units along the cycle iP and obtain a new circulation.
- We claim that we can continue this process until the current circulation x has $l_{ij} \leq x_{ij} \leq u_{ij}$ for all arcs $ij \in A$, that is, we can obtain a feasible circulation in $\mathcal{N}$ (observe that the procedure above preserves the inequality $x \leq u$).

start with $x \equiv 0$



suppose $l_{ij} > x_{ij} \geq 0$





$x \oplus \tilde{x}$

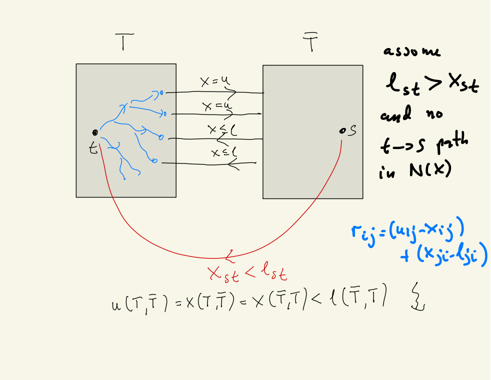
$b_{x'} = b_{\tilde{x}} + b_x$

$\tilde{x}$ path flow of value 3 from $j \to i$

- Suppose this is not the case and that at some point we have $x_{st} < l_{st}$ for some arc st and there is no (t, s)-path in $\mathcal{N}(x)$.
- Define T as follows:

$$T = \{r : \text{ there exists a } (t, r)\text{-path in } \mathcal{N}(x)\}.$$

- It follows from the definition of the residual network $\mathcal{N}(x)$ (in particular (6)) that in $\mathcal{N}$ we have $x_{ij} = u_{ij}$ for all arcs ij with $i \in T$ and $j \in \overline{T}$ and $x_{qr} \leq l_{qr}$ for all arcs qr with $q \in \overline{T}$ and $r \in T$.



- Using that $s \in \overline{T}$ and $x_{st} < l_{st}$ we obtain that

$$u(T, \overline{T}) = x(T, \overline{T}) = x(\overline{T}, T) < l(\overline{T}, T),$$

  contradicting the assumption that (7) holds.
- This and the fact that all data are integers shows that the algorithm we described above will indeed find a feasible circulation in $\mathcal{N}$. □

It is not difficult to turn the proof above into a polynomial algorithm which, given a network $\mathcal{N} = (V, A, l, u)$, either finds a feasible circulation x in $\mathcal{N}$, or a subset S violating (7)

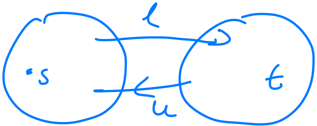
Minimum value flows

Let $\mathcal{N} = (V, A, l, u)$ be a network with source s, sink t and non-negative lower bounds on the arcs.

A **minimum value** feasible (s, t)-flow in $\mathcal{N}$ is a feasible (s, t)-flow x whose value $b_x(s)$ is minimum possible among all feasible (s, t)-flows.

Although at first glance this problem may seem somewhat artificial, it turns out that for many applications it is actually a minimum feasible flow that is sought.
To estimate the value of a minimum (s, t)-flow, let us define the **demand**, $\gamma(S, \overline{S})$ of an (s, t)-cut $(S, \overline{S})$ as the number

$$\gamma(S, \overline{S}) = l(S, \overline{S}) - u(\overline{S}, S). \tag{8}$$



Let x be a feasible flow. Then for every (s,t)-cut $(S,\overline{S})$ we have

$$\begin{aligned} b_x(s) &= x(S,\overline{S}) - x(\overline{S},S) \\ &\geq l(S,\overline{S}) - u(\overline{S},S) \\ &= \gamma(S,\overline{S}). \end{aligned} \tag{9}$$

Theorem 17 (Min-flow Max-demand theorem)

Let $\mathcal{N} = (V, A, l, u)$ be a network with non-negative lower bounds on the arcs. Suppose x is a minimum feasible (s,t)-flow in $\mathcal{N}$. Then

$$b_x(s) = \max\{\gamma(S,\overline{S}) : s \in S, t \in \overline{S}\}. \tag{10}$$

Furthermore we can find a minimum feasible (s,t)-flow by two applications of any algorithm for finding a maximum (s,t)-flow.

The **cost** of a feasible flow x in a network $\mathcal{N} = (V, A, \ell, u, b, c)$ is

$$c(x) = \sum_{ij \in A} c_{ij} x_{ij}$$

. The minimum cost flow problem is as follows

Min cost flow **Input:** A network $\mathcal{N} = (V, A, \ell \equiv 0, u, b, c)$ **Question:** Find a feasible flow x in $\mathcal{N}$ which minimizes the cost $c(x)$ over all feasible flows in $\mathcal{N}$.

x flow

$x' = x \oplus \tilde{x}$ $\quad b_{x'} = b_x$

$N(x)$

$\delta(W) = 3$

$\tilde{x}$

W

$c(W) < 0$

$\overset{\shortparallel}{-5}$

Theorem 18

Let x be a feasible flow in the network $\mathcal{N} = (V, A, l, u, b, c)$. Then x is a minimum cost feasible flow in $\mathcal{N}$ if and only if $\mathcal{N}(x)$ contains no directed cycle of negative cost.

The cycle canceling algorithm
Input: A network $\mathcal{N} = (V, A, l, u, b, c)$.
Output: A minimum cost feasible flow in $\mathcal{N}$.

1. Find a feasible flow x in $\mathcal{N}$.
2. Search for a negative cycle in $\mathcal{N}(x)$.
3. If such a cycle W is found then augment x by $\delta(W)$ units along W and go to Step 2.
4. Return x.

Theorem 19 (Integrality theorem for minimum cost flows)

If all lower bounds, capacities and balance vectors of the network $\mathcal{N}$ are integers, then there exists an integer valued minimum cost flow.

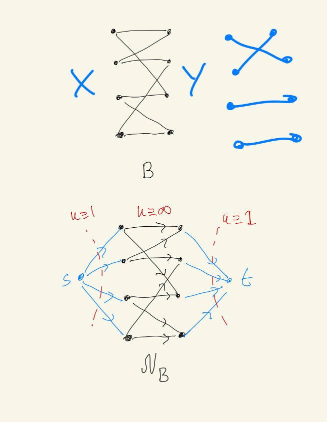
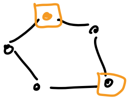


Figure: A bipartite graph and the corresponding network. Capacities are one on all arcs of the form sv, ut and ∞ on all arcs corresponding to edges of B.

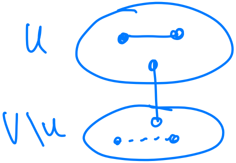


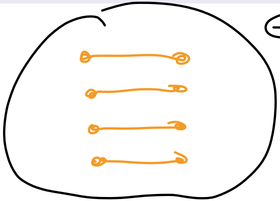
Theorem 20

Let $B = (X, Y, E)$ be a bipartite graph and let $\mathcal{N}_B$ be the associated network and let x be a maximum flow in $\mathcal{N}_B$ The size of a maximum matching M in B satisfies $|M| = b_x(s)$.

Theorem 21 (König's theorem)

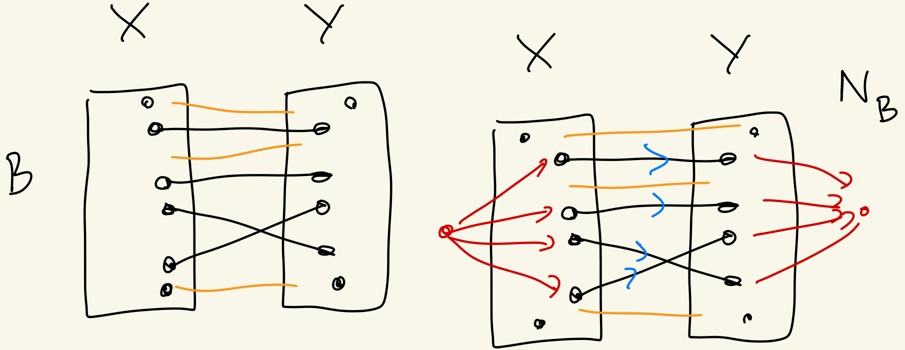
Let $B = (X, Y; E)$ be an undirected bipartite graph with bipartition (X, Y). The size of a maximum matching in B equals the size of a minimum vertex cover in B.





G in general

|max matching| $\leq$ min vertex cover



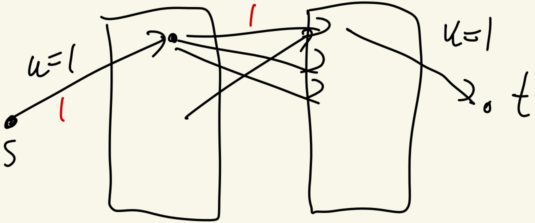
$|M^*|$ max size of a matching in B

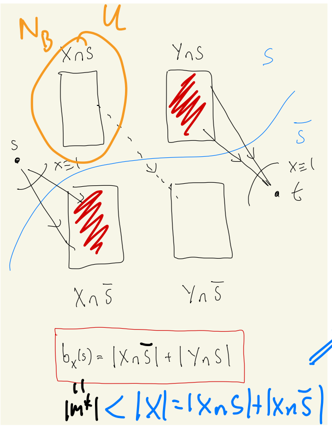
$\leq \max\{ b_x(s) \mid x \text{ is an } (s,t)\text{-flow in } N_B \}$

let x be a flow of value k in N_B

integer

return edges with possitive flow $(=1)$ between X and Y





$|X \cap S| > |Y \cap S|$

$U \qquad N(U)$

$\cap\!|$

$Y \cap S$

Figure: The situation when a maximum flow has been found. The dotted arc indicates that there is no arc between the two sets $X \cap S$ and $Y \cap \overline{S}$.

Theorem 22 (Hall's theorem)

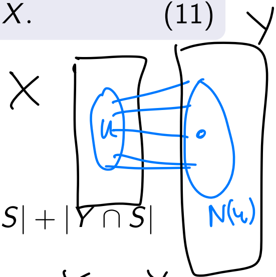
A bipartite graph $B = (X, Y; E)$ has a perfect matching if and only if $|X| = |Y|$ and the following holds:

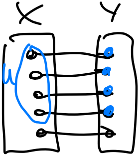
$$|N(U)| \geq |U| \qquad \text{for every } U \subset X. \tag{11}$$

Proof: Suppose (11) holds but $|M| < |X|$.
Then

$$\begin{aligned} |X \cap S| + |X \cap \bar{S}| &= |X| \\ &> |M| = b_x(s) = |X \cap S| + |Y \cap S| \end{aligned}$$

So $|N(X \cap S)| \leq |Y \cap S| < |X \cap S|$, contradiction.





Finding subdigraphs with prescribed degrees

Theorem 23

There exists a polynomial algorithm for the following problem. Given a digraph $D = (V, A)$ with $V = \{v_1, v_2, \ldots, v_n\}$ and integers $a_1, a_2, \ldots, a_n$, $b_1, b_2, \ldots, b_n$, find a subdigraph $D' = (V, A^*)$ of D which satisfies $d^+_{D'}(v_i) = a_i$ and $d^-_{D'}(v_i) = b_i$ for each $i = 1, 2, \ldots, n$, or show that no such subdigraph exists. Furthermore, if there are costs specified for each arc, then we can also find in polynomial time the cheapest (minimum cost) subdigraph which satisfies the degree conditions.

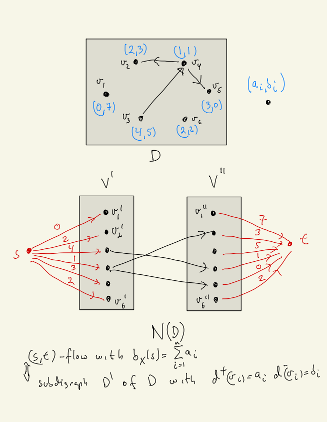


Figure: Looking for subdigraphs with prescribed in- and out-degrees.

A **q-path-cycle subdigraph** $\mathcal{F}$ of a digraph D is a collection $\mathcal{F} = P_1 \cup \ldots \cup P_q \cup C_1 \cup \ldots \cup C_t$ of q paths and t such that all of $P_1, \ldots, P_q, C_1, \ldots, C_t$ are pairwise disjoint (possibly, $q = 0$ or $t = 0$).

q

- A **q-path-cycle factor** is a spanning q-path-cycle subdigraph.
- If $q = 0$ we say that $\mathcal{F}$ is a **t-cycle subdigraph** (or just a **cycle subdigraph**) and it is a **t-cycle factor** (or just a **cycle factor**) if it is spanning.
- If $t = 0$, $\mathcal{F}$ is a **q-path subdigraph** and it is a **q-path factor** (or just a **path-factor**) if it is spanning.
- The **path covering number** $pc(D)$ of D is the minimum positive integer k such that D contains a k-path factor. In particular, $pc(D) = 1$ if and only if D is traceable.
- The **path-cycle covering number** $pcc(D)$ of D is the minimum positive integer k such that D contains a k-path-cycle factor.

Theorem 24

There is an $O(n^3)$ algorithm which finds, for any given digraph D, a cycle subdigraph covering the maximum number of vertices in D.

Proposition 25 (Jackson and Ordaz)

If D is a k-strong digraph such that the maximum size of an independent set in D is at most k, then D has a spanning cycle subdigraph.

Theorem 26

Let $D = (V, A)$ be a k-strong digraph and let $X \subset V(D)$ be such that $\alpha(D\langle X\rangle) \leq k$, then D has a cycle subdigraph (not necessarily spanning) covering X.

Proposition 27

Let D be a directed pseudograph and let k be a fixed non-negative integer. Then

(a) *In time $O(\sqrt{n}m)$ we can check whether D has a k-path-cycle-factor and construct one (if it exists).*

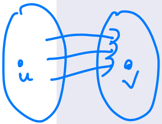
(b) *Given a k-path-cycle factor in D, in time $O(m)$, we can check whether D has a $(k-1)$-path-cycle factor and construct one (if it exists).*

Corollary 28

We can determine the minimum k such that a digraph D has a path-cycle factor with k paths in polynomial time. In particular the path covering number of an acyclic digraph can be found in polynomial time

Theorem 29 (Menger's theorem)

Let D be a directed multigraph and let $u, v \in V(D)$ be a pair of distinct vertices. Then the following holds:



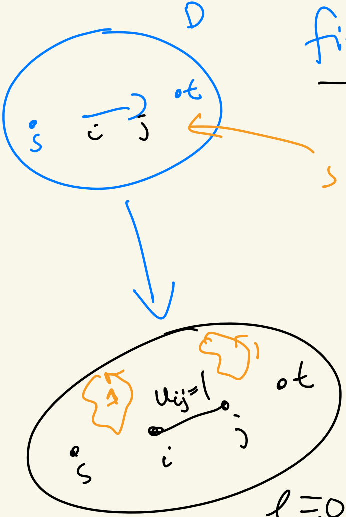
(a) *The maximum number of arc-disjoint (u, v)-paths equals the minimum number of arcs covering all (u, v)-paths and this minimum is attained for some (u, v)-cut $(X, \bar{X})$.*

(b) *If the arc uv is not in $A(D)$, then the maximum number of internally disjoint (u, v)-paths equals the minimum number of vertices in a (u, v)-separator.*

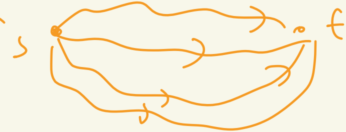
Theorem 30

Let D be a digraph with is not complete.

- *We can determine the arc-connectivity $\lambda(D)$ of D and produce a cut $(X, \bar{X})$ with $\lambda(D) = |(X, \bar{X})|$ in polynomial time.*
- *We can determine the vertex connectivity $\kappa(D)$ of D and produce a separator X with $|X| = \kappa$ in polynomial time.*



find $\lambda(s,t) = \max \#$ of arc-disjoint (u,v)-paths



$\ell \equiv 0$

N_D

$B = \max\{ b_x(s) \mid x \text{ is an } (s,t)\text{-flow in } N_D \}$

$\|$

$\lambda(s,t)$

$B \geq \lambda(s,t)$

$B \leq \lambda(s,t)$

flow decomp.!!

let x have value B in N_D

decompose x into path and cycle flows

So $\lambda(s,t) = b_{x^*}(s)$ $\quad x^*$ max flow

$\|$ max flow min cut

$\|$ $\quad \min u(S,\bar{S})$

$d^+(S)$

$\|$

$u(S,S)$