Exercise Session 5 21.05.

Announcements: • 28. + 30.5. → lectures, no ex. sess. next week
• exercise sheet no. 6 is the last for the first half of the semester

Topics: • Paths and Matchings
• min. cost perfect Matchings (bipartite)

Paths and Matchings:

• Relation between shortest s-t paths and weighted matchings

min. cost perf. matching: $G=(V,E)$, $c: E \to \mathbb{R}$, find perfect Matching M s.t. $c(M) := \sum_{e \in M} c(e)$ is minimal

shortest s-t path in directed network: $D = (V,A)$, $c: A \to \mathbb{R}$, find shortest s-t path in D, $s,t \in V$

Transformation: $D \rightsquigarrow \overline{G}$

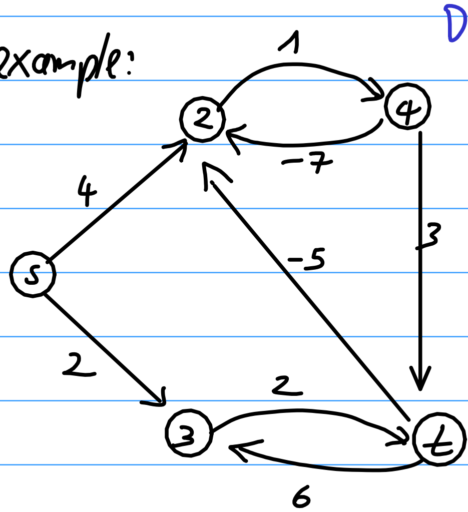
• split $v \in V$ into v', v'' $\longrightarrow$ $\overline{V} = \{v' \mid v \in V\} \cup \{v'' \mid v \in V\}$

• replace arcs (u,v) by $\{u', v''\}$ and $\{u', u''\}$
cost $c(u,v)$ cost 0

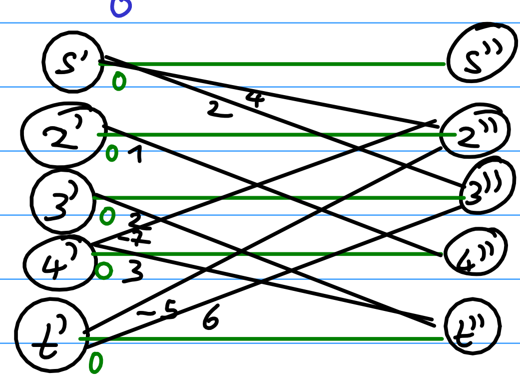
$\longrightarrow$ $\overline{E} = \underbrace{\{\{u', v''\} \mid \forall (u,v) \in A\}}_{c(\{u',v''\}) = c(u,v)} \cup \underbrace{\{\{u',u''\} \mid \forall u \in V\}}_{c(\{u',u''\}) = 0}$ artifical

example:

D



$\overline{G}$



Lemma: D contains neg. cycle $\iff \bar{G}$ has a perfect matching M with $c(M) < 0$

Proof: „$\Rightarrow$" cycle $c = v_1, \dots, v_k, v_1$ with length < 0

define $M = \underbrace{\{\{v_i', v_j''\} \mid \forall (v_i, v_j) \in c\}}_{\sum_e c(e) = \text{length of } c < 0} \cup \underbrace{\{\{v_i', v_i''\} \mid v_i \notin c\}}_{\sum_e c(e) = 0}$

$\rightarrow c(M) < 0$

- M is perfect matching, each v', v'' in matching

„$\Leftarrow$" perfect matching M, $c(M) < 0$,

consider edge (u', v'') in M as directed arcs $u \to v$ in D

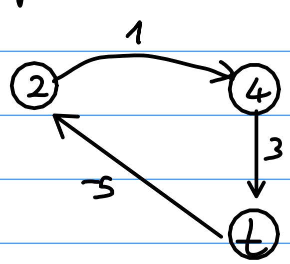
$\Rightarrow n^{*}$ arcs in D, one outgoing/incoming edge per vertex

$\Rightarrow$ set of cycles, each cycle with cost ≤ 0 (otherwise replace by $u' \to u''$ in matching)

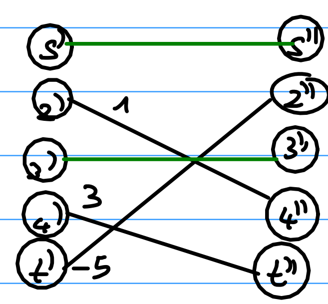
$\Rightarrow \exists$ cycle c with cost < 0 $\square$

* including arcs (u,u), u

example:



cycle of length -1



Lemma 2: D contains a shortest s-t-path of length ℓ $\iff$ min. cost perf. matching M in $\hat{G}$ with cost ℓ,

$V(\hat{G}) = V(\bar{G}) \setminus \{s'', t'\}$

$E(\hat{G}) = \{\{u,v\} \mid \{u,v\} \in E(\bar{G}) \text{ and } u, v \in V(\hat{G})\}$

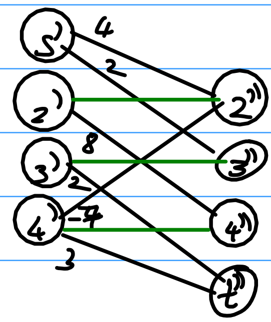
Proof (sketch) „$\Leftarrow$": matching M defines arcs (u,v) for all $u \in V \setminus \{t\}$ $\quad \delta^-(v) = 1$

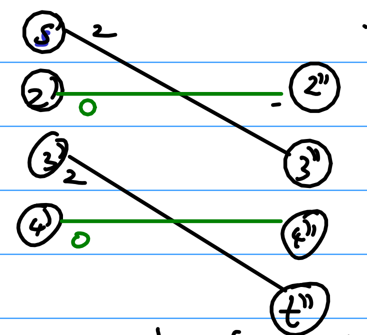
arcs (u,v) for all $v \in V \setminus \{s\}$ $\quad \delta^+(v) = 1$

- M defines s-t path + some cycles of cost 0
- M is min cost $\Rightarrow$ M defines min cost s-t path

„⟹" · use edges on path as matching edges
· add artifical edges for all $v \in V$ not in path
· $c(M)$ is length of path $\square$

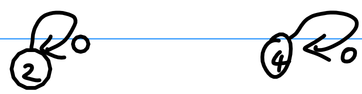
example!

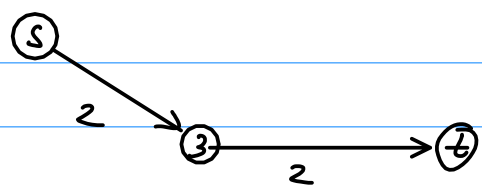




→ set $c(2,4) = 8$ ⇝ no neg. cycle

min cost perf. matching

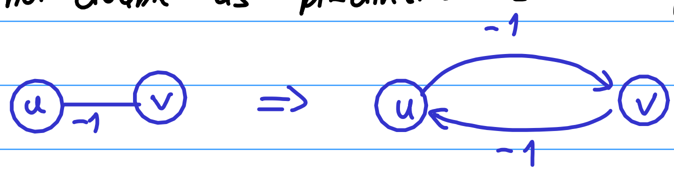




shortest path in D

Shortest Paths in Undirected Networks:

not doable as bi-directed shortest path



negative cycle in network for edge with neg. cost, no cycle in undir. graph

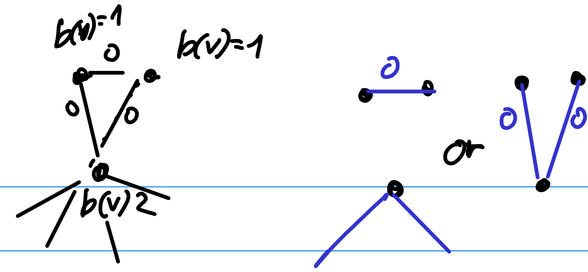
· using b-matchings: $b(v) \in \mathbb{N}$, M is b-matching

iff $|\{e \mid e \in M \text{ and } e = \{u,v\}\}| = b(v) \quad \forall v \in V$

our case: b-Matching $b(s) = b(t) = 1$, $b(v) = 2 \quad \forall v \in V \setminus \{s,t\}$

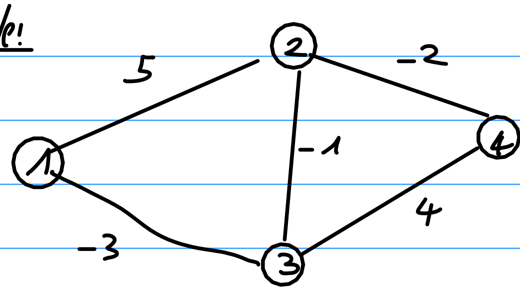
· adding self loops ⟹ those count twice

replaceable by

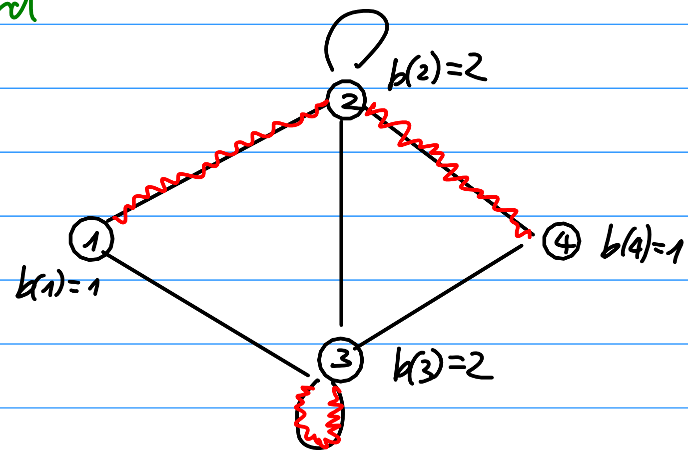


⇝ similar in graph at the end

example!

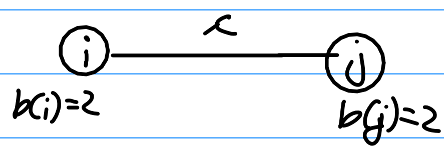


⇝

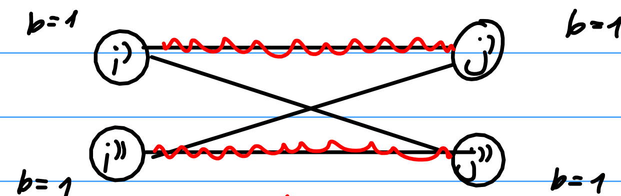


transform b-Matching to Matching:

first idea:

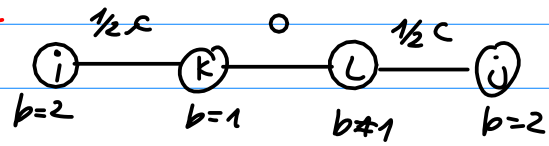


⇝

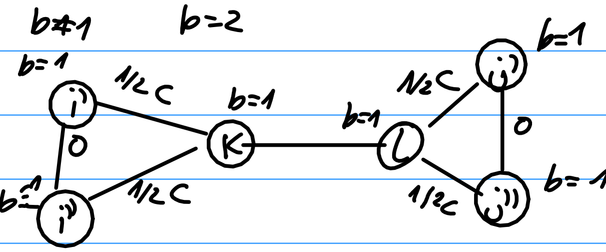


→ uses edge (i,j) twice

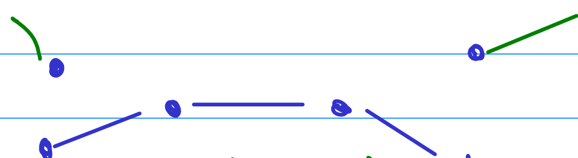
fix:



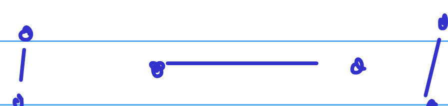
split this new edge:



possible matchings:

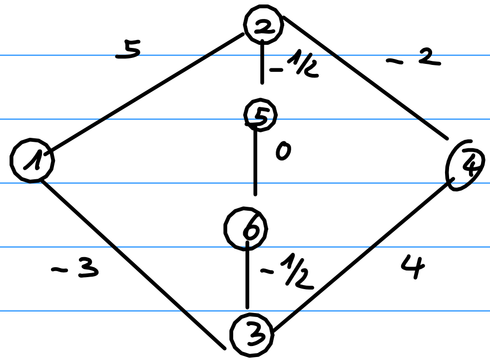


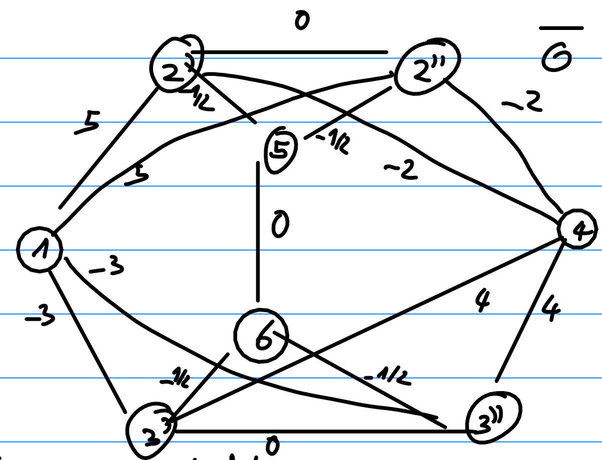
(u,v) + (u,w) + (x,v)



(u,v) not used

example:





⇝ · shortest s-t path problem solvable as Matching

· similar as in directed case

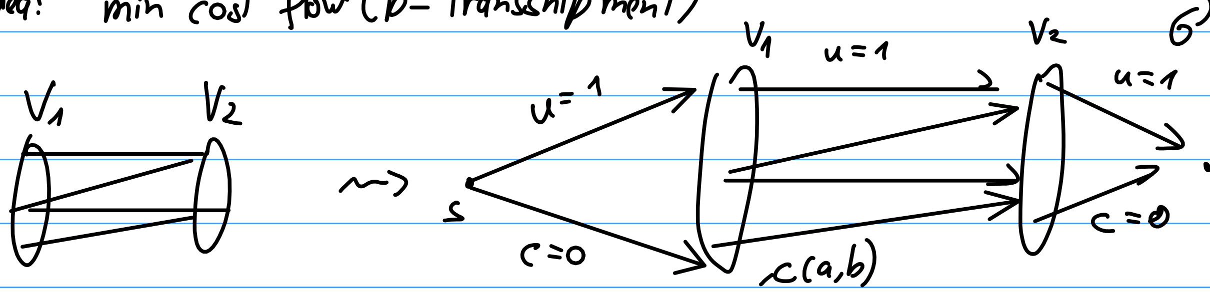
Lemma: Matching in $\overline{G}$ is equivalent to an s-t path and several cycles

Min cost perfect Matchings in bipartite graphs

· $G = (V,E)$, $V = V_1 \dot\cup V_2$, $E \subseteq V_1 \times V_2$, $c: E \to \mathbb{R}$

solving MCPM:

1. idea: min cost flow (b-Transshipment)



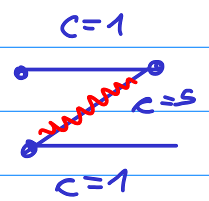
observation: min cost flow in G' corresponds to min. cost Matching in G

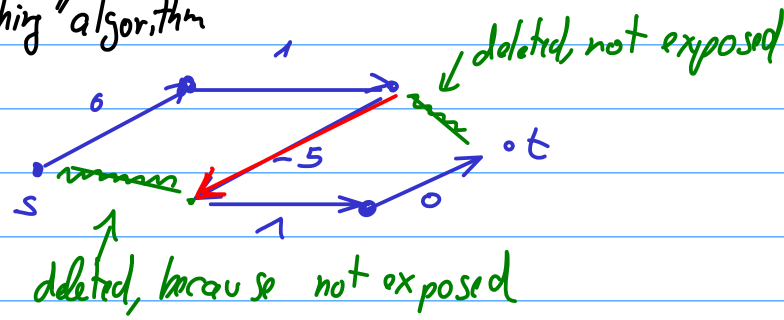
- optimal flow can be forced to be integral
- flow arcs correspond to matching arcs

Algorithm: · Edmonds Karp → find maximal flow (here: perfect Matching)

· Cycle Canceling Algo → find min cost flow (here: min cost perf. Matching)

2. approach: special „Matching" algorithm





→ s-t path is M augm path

Algo:
- $M = \emptyset$
- while $|M| < \frac{n}{2}$ do
  - construct directed graph G':

    $V(G') = V(G) \cup \{s,t\}$

    $E(G') = \{(s,v) : v \in V_1,\ v \text{ is exposed}\}$ $\quad c'(s,v) = 0$

    $\cup \{(v,t) : v \in V_2,\ v \text{ is exposed}\}$ $\quad c'(v,t) = 0$

    $\cup \{(w,v) : v \in V_1, w \in V_2, (v,w) \in M\}$ $\quad c'(w,v) = c(v,w)$

    $\cup \{(v,w) : v \in V_1, w \in V_2, (v,w) \in E(G) \setminus M\}$ $\quad c'(v,w) = c(v,w)$
  - find shortest s-t path in G' w.r.t. c'
  - augm. Matching M with s-t path
- return M

→ Algorithm finds min cost perfect Matching and terminates after a finite number of steps (→ $\nexists$ neg. cycle)

Proof idea: – define potential $\pi : V \to \mathbb{R}$; $c^\pi(v,w) = c(v,w) + \pi(v) - \pi(w)$

π feasible if: $c^\pi(v,w) = 0 \quad \forall (v,w) \in M$
$\geq 0 \quad \forall (v,w) \notin M$

↦ show: • for $M = \emptyset$, $\pi \equiv 0$ is a feas. potential
- neg cycle w.r.t. to c ⟺ neg. cycle w.r.t. to c^π
- each step defines new feas. potential

  $\pi'(v) = \pi(v) +$ shortest distance from s to w (matching Kanten)

  $\pi'(v) = \pi(v)$

- $\Rightarrow$ each step $\to$ no neg. cycle

$\Rightarrow$ terminate with perf. matching, no neg. cycle

$\Rightarrow$ min. cost perf. matching