

Uantwerk

~~Wed, 5.12. 18:00~~
Thu, 6.12. 18:00

Café Campus

Duality

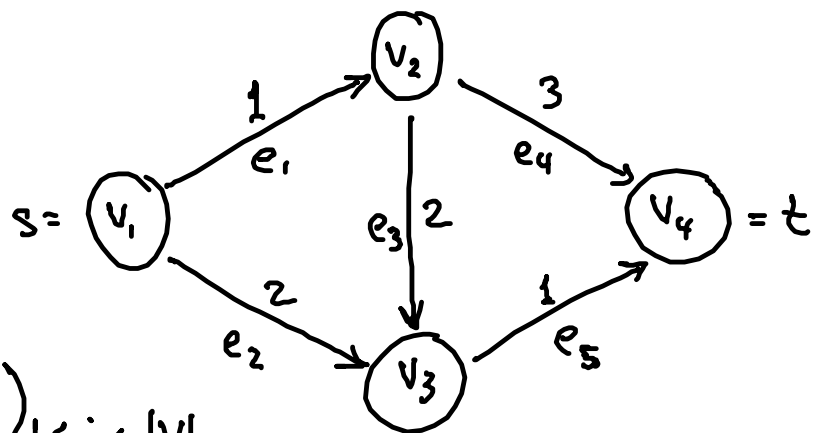
An Example from combinatorial optimization.

► Graph $G=(V, E)$ connected, directed edges, weights $c(e) \forall e \in E$ (cost)

pick two vertices $s, t \in V$

problem: find a path W from s to t with

minimal cost $\sum_{e \in W} c(e)$



► Incidence Matrix $A = (a_{ij})_{\substack{1 \leq i \leq |V| \\ 1 \leq j \leq |E|}}$

$$a_{ij} := \begin{cases} -1 & \text{if } \xrightarrow{e_j} (v_i) \\ +1 & \text{if } (v_i) \xrightarrow{e_j} \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{pmatrix}$$

$$\rightarrow \sum_{v_j \in V} a_{ij} = 0 \quad \forall i$$

in fact: $\text{rank } A = |V| - 1$

▷ $f \in \mathbb{R}^{|E|}$

$f \geq 0$

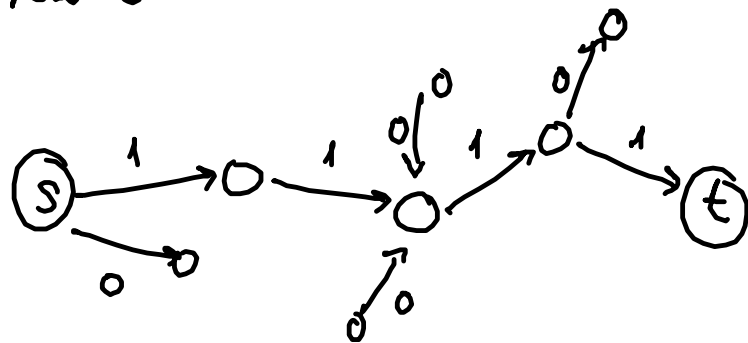
→ interpretation: f_j flow through edge e_j

▷ flow conservation: for all vertices v_i (except s, t)

$$\sum_{\substack{e_j \in E \\ e_j \rightarrow v_i}} f_j = \sum_{\substack{e_j \in E \\ v_i \rightarrow e_j}} f_j \iff \sum_{j=1}^{|E|} a_{ij} \cdot f_j = 0$$

$$\rightsquigarrow A \cdot f = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \alpha \\ \vdots \\ 0 \\ -\beta \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \leftarrow \text{row } s \\ \\ \\ \leftarrow \text{row } t \end{matrix}$$

▷ path from s to t
is a flow $f \in \{0, 1\}^{|E|}$
with



$$A \cdot f = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 0 \\ -1 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \leftarrow \text{row } s \\ \\ \\ \leftarrow \text{row } t \end{matrix}$$

► Consider the linear program

$$\min c^T f$$

$$c = (c(e_i))_{1 \leq i \leq |E|}$$

$$\text{s.t. } A \cdot f = b$$

$$b =$$

$$f \geq 0.$$

If it has an optimal solution, then it also has an optimal solution with $f \in \{0, 1\}^{|E|}$, and the simplex algorithm finds such a solution.

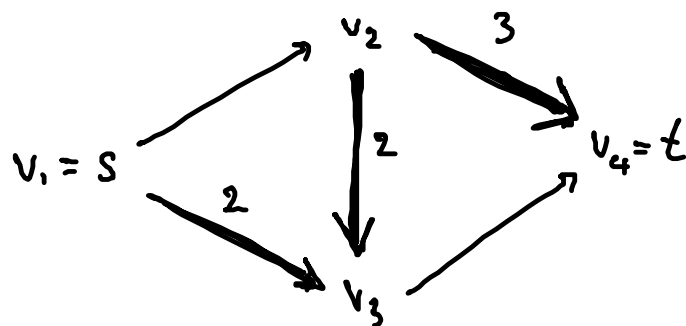
► Example: $\min (1, 2, 2, 3, 1) \cdot f$

$$\text{s.t. } \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \end{pmatrix} \cdot f = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$f_1, \dots, f_5 \geq 0$$

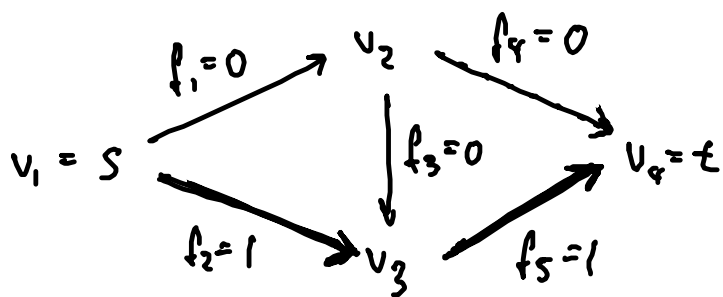
basis: 2, 3, 4

		f_2	f_3	f_4	
3		1	0	0	0
$f_2 =$		1	1	0	0
$f_3 =$		-1	0	1	(-1)
$f_4 =$		0	0	0	1



new basis: 2, 5, 4

	3	1	0	0	0	0
$p_2 =$	1	1	1	0	0	0
$p_5 =$	1	1	0	-1	0	1
$p_4 =$	0	-1	0	1	1	0



⇒ shortest path over edges e_2 and e_5 with cost 3

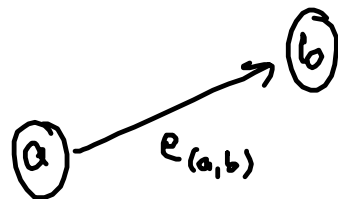
► dual problem:

$$\begin{aligned} \max \quad & b^T p \\ \text{s.t.} \quad & A^T \cdot p \leq c \\ & p \text{ free} \end{aligned}$$

interpretation:

$$\max \quad p_s - p_t$$

$$\text{s.t.} \quad p_a - p_b \leq c(e_{(a,b)})$$



objective function: $p_s - p_t$

$$= \underbrace{(p_s - p_{i_1})}_{\leq c(e_{j_1})} + \underbrace{(p_{i_1} - p_{i_2})}_{\leq c(e_{j_2})} + p_{i_2} \dots - p_{i_k} + \underbrace{(p_{i_k} - t)}$$

$$\leq \sum_{e \in W} c(e)$$



→ dual program maximises the lower bound on the costs of paths between s and t

▶ complementary slackness: for p and f optimal dual/primal solution

$$\bullet p_i \cdot \left(\sum_j a_{ij} f_j - b_i \right) = 0 \quad \rightarrow \checkmark \text{ (std. form!)}$$

$$\bullet f_j \cdot \left(c_j - \sum_i p_i a_{ij} \right) = 0$$

→ if an edge e_j lies on a shortest path ($f_j > 0$), then it is pulled tight ($\sum_i p_i a_{ij} = c_j$); if it is not tight, then it cannot lie on the shortest path

▶ shadow prices ...

