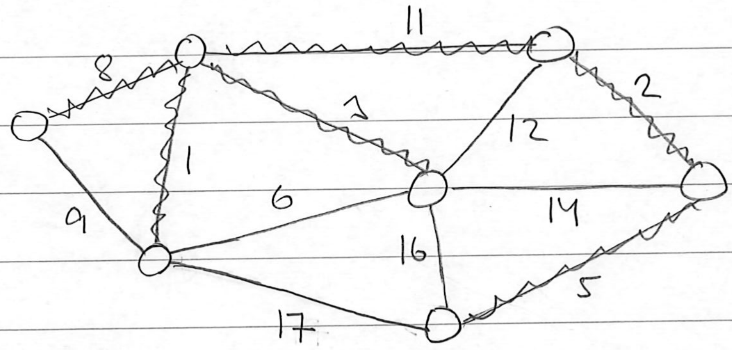
Minimaltråd

(eller: billigaste uppspännande träd)

<u>Givet</u>: Ett <u>oriktat</u> nätverk (N,B) med kostnader på bågarna.

<u>Definition</u>: Ett <u>uppspännande träd</u> är en bågmängd $T \subseteq B$ sådan att (N,T) är sammanhängande och cykelfri.

<u>Sökt</u>: Ett uppspännande träd av <u>minimal kostnad</u>.



ett minimaltråd

<u>Sats</u>: Följande utsagor är ekvivalenta:

i) T är ett uppspännande träd.

ii) $|T| = |N| - 1$ och (N,T) är sammanhängande.

iii) Mellan varje par av noder finns en <u>unik</u> väg i (N,T).

iv) (N,T) saknar cykler men om någon båge adderas så bildas en <u>unik</u> cykel.

Matematisk formulering

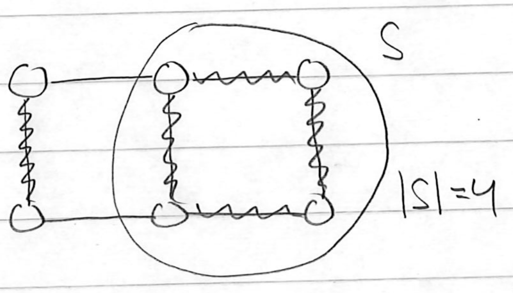
Variabler: $x_e = \begin{cases} 1 & \text{om båge } e \in B \text{ ingår i trädet} \\ 0 & \text{annars} \end{cases}$

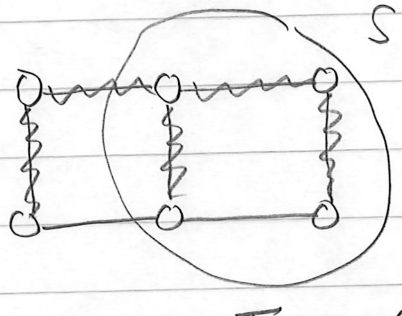
Välj $|N|-1$ bågar: $\sum_{e \in B} x_e = |N|-1$

(N,T) *ska sakna cykler!*

Låt $S \subset N$ med $|S| \geq 3$ och $E(S)$ = mängden av bågar *inom* S.

$|N|-1=5$





cykel: $\sum_{e \in E(S)} x_e = \underbrace{4}_{=|S|}$

inte cykel: $\sum_{e \in E(S)} x_e = \underbrace{3}_{=|S|-1}$

Alltså: villkoret $\sum_{e \in E(S)} x_e \leq |S|-1$ förbjuder cykeln som passerar alla noder i S.

Om $\sum_{e \in E(S)} x_e \leq |S|-1$ gäller för alla $S \subset N$ med $|S| \geq 3$ så förbjuds alla cykler.

($S=N$ behövs inte pga villkoret $\sum_{e \in B} x_e = |N|-1$.
$|S| \leq 2$ behövs inte pga att då gäller alltid att $\sum_{e \in E(S)} x_e \leq |S|-1$.)

Låt c_e = kostnad för båge $e \in B$.

Modell: $\min \sum_{e \in B} c_e x_e$

$$\text{då } \sum_{e \in A} x_e = |N| - 1 \qquad (1)$$

$$\sum_{e \in E(S)} x_e \leq |S| - 1, \quad S \subset N : |S| \geq 3 \qquad (2)$$

$$x_e \in \{0,1\}, \quad e \in B$$

(2) kallas subturförbjudande villkor.

Det finns $\underbrace{2^{|N|}}_{S \subseteq N} - \underbrace{2}_{\substack{S=\emptyset \\ S=N}} - \underbrace{|N|}_{|S|=1} - \underbrace{|N|(|N|-1)/2}_{|S|=2} \approx 2^{|N|}$!

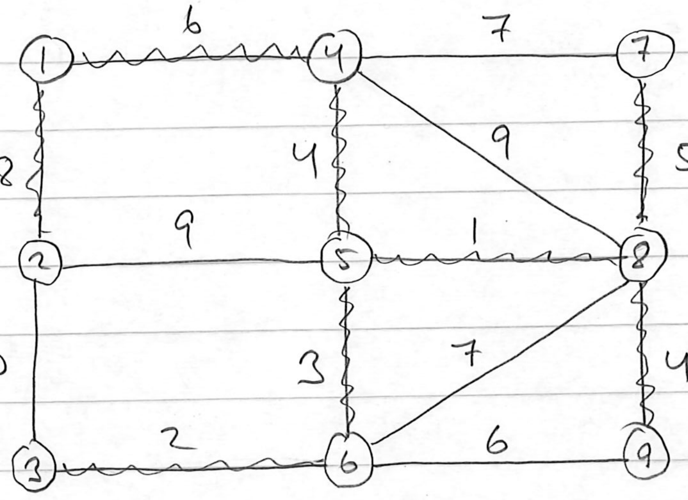
Konstruktion av minimalträd

Problemet löses mha en girig algoritm:

- konstruera en lösning
- gör i varje steg det lokalt bästa
- ångra aldrig ett beslut.

Exempelvis Kruskals algoritm:
Välj bågar i stigande kostnadsordning, givet att cykler inte får uppkomma.

Exempel:



ordning

(5,8) inkludera

(3,6) —"—

(5,6) —"—

(4,5) —"—

(8,9) —"—

(7,8) —"—

(6,9) cykel

(1,4) inkludera

(4,7) cykel

(6,8) —"—

(1,2) inkludera

9-1=8 bågar inkluderade ⇒ träd ☺