Sats: Varje tillåten baslösning till problemet

$$\begin{aligned} \min \quad & c^{\mathrm{T}}x \\ \text{då} \quad & \bar{A}x = \bar{b} \\ & x \geq 0 \end{aligned}$$

är 0/1-värd.

Bevis: Antag att basvariablerna ges av $x_B = (x_1, \ldots, x_{m-1})^{\mathrm{T}}$. (Det finns $m-1$ villkor och variablerna kan numreras så att basvariablerna får lägst index.)

Basvariablernas värden ges av lösningen till systemet $Bx_B = \bar{b}$, där $B = (\bar{A}_1, \bar{A}_2, \ldots, \bar{A}_{m-1})$. Eftersom B är en kvadratisk delmatris av anslutningsmatrisen A, vilken är fullständigt unimodulär, så gäller att $\det B = 0, \pm 1$. Av definitionen av baslösning följer att $\det B \neq 0$, varför alltså $\det B = \pm 1$ gäller.

Cramers regel ger att lösningen till systemet ges av

$$x_j = \frac{\det B^j}{\det B}, \quad j = 1, \ldots, m-1,$$

där $B^j = (\bar{A}_1, \ldots, \bar{A}_{j-1}, \bar{b}, \bar{A}_{j+1}, \ldots, \bar{A}_{m-1})$. Om $\det B^j$ beräknas genom utveckling längs kolumnen $\bar{b} = (0, 0, \ldots, -1)^{\mathrm{T}}$ så fås att

$$\det B^j = \pm \underbrace{\det(\text{kvadratisk delmatris av A})}_{0,\pm 1} = 0, \pm 1.$$

Alltså fås att $x_j = 0, \pm 1$, $j = 1, \ldots, m-1$.

Slutligen följer att $x_j = 0/1$, $j = 1, \ldots, m-1$, eftersom baslösningen är tillåten.

Följdsats: En optimal tillåten baslösning (hörnpunkt) löser 0/1-problemet.

Slutsats: bv-problemet är ett 0/1-problem men kan lösas som ett linjärt optimeringsproblem, och utom villkoren $x \leq 1$.

Speciellt: det duala problemet till bv-problemet kan utnyttjas!

Exemplet igen:

$$\min \; (1,3,1,4,1,2)\,x$$

då

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & -1 & 0 \\ 0 & -1 & -1 & 0 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -1 \end{pmatrix} x = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix} \quad \begin{array}{|l} y_1 \\ y_2 \\ y_3 \\ y_4 \end{array}$$

$$x \geq 0$$

svarar mot noder!

Dual: $\max \; v = y_1 - y_4$

då

$$\begin{cases} y_1 - y_2 \leq 1 \\ y_1 - y_3 \leq 3 \\ y_2 - y_3 \leq 1 \\ y_2 - y_4 \leq 4 \\ y_3 - y_2 \leq 1 \\ y_3 - y_4 \leq 2 \end{cases} \quad \begin{array}{|l} x_{12} \\ x_{13} \\ x_{23} \\ x_{24} \\ x_{32} \\ x_{34} \end{array}$$

Istället för att stryka första primala villkoret sätts senare $y_1 = 0$.

Gör ett teckenbyte på dualvariablerna. Ger en bättre tolkning av dem!

$$\max v = y_4 - y_1$$

$$\text{då } \begin{cases} y_2 - y_1 \le 1 \\ y_3 - y_1 \le 3 \\ y_3 - y_2 \le 1 \\ y_4 - y_2 \le 4 \\ y_2 - y_3 \le 1 \\ y_4 - y_3 \le 2 \end{cases}$$

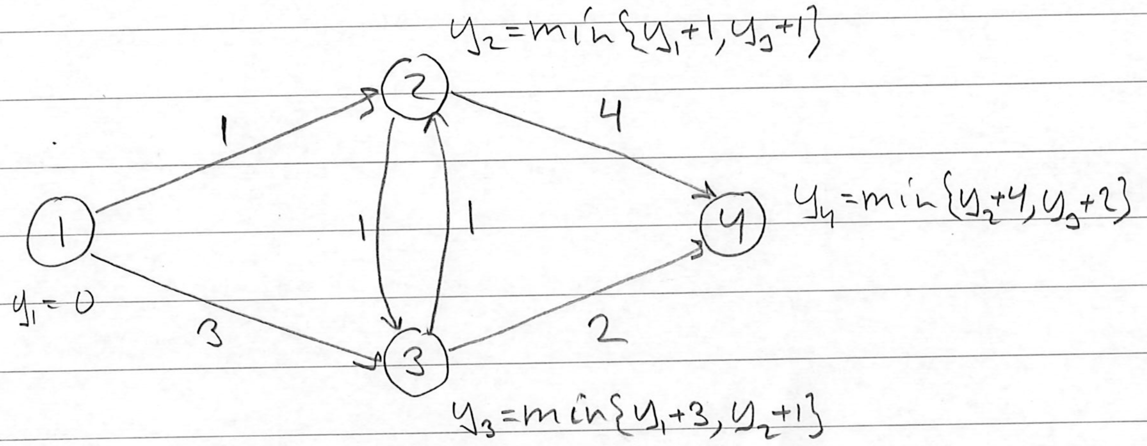
max y_3

$$y_3 = \min\{y_1 + 3, y_2 + 1\}$$

$$y_2 = \min\{y_1 + 1, y_3 + 1\}$$

$$y_4 = \min\{y_2 + 4, y_3 + 2\}$$

max y_4

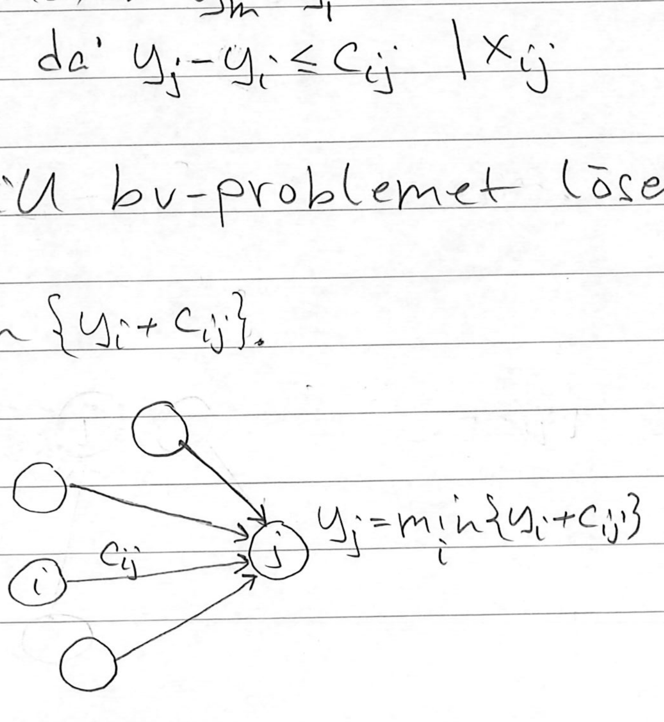


Generell dual: $\max v = y_m - y_1$

då $y_j - y_i \le c_{ij} \mid x_{ij}$

<u>Sats</u>: Dualen till bv-problemet löses av

$$\begin{cases} y_1 = 0 \\ y_j = \min\limits_i \{y_i + c_{ij}\}. \end{cases}$$



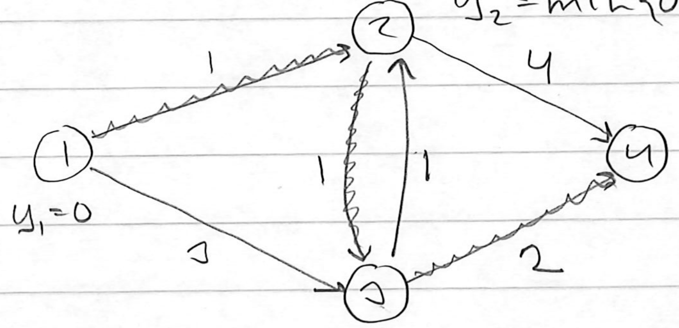
Tolkning: y_j = kostnad för en bv från nod 1 till nod j.

y_j kallas <u>nodpris</u>.

Villkoren i satsen kallas <u>Bellmans ekvationer</u>. Observera att de generellt utgör ett <u>ekvationssystem</u>.

Lösning i exemplet:

$y_2 = \min\{0+1, 2+1\} = 1$ för nod 1



$y_4 = \min\{1+4, 2+2\} = 4$ för nod 3

$y_3 = \min\{0+3, 1+1\} = 2$ för nod 2

Optimal väg? Fås mha komplementvillkoren: $x_{ij}(c_{ij} - y_j + y_i) = 0$, $\forall (i,j)$.

$$\begin{cases} y_j - y_i < c_{ij} \Rightarrow x_{ij} = 0 & \text{(båge } (i,j) \text{ inte i bv)} \\ x_{ij} = 1 \text{ (båge } (i,j) \text{ i bv)} \Rightarrow y_j - y_i = c_{ij} & \end{cases}$$

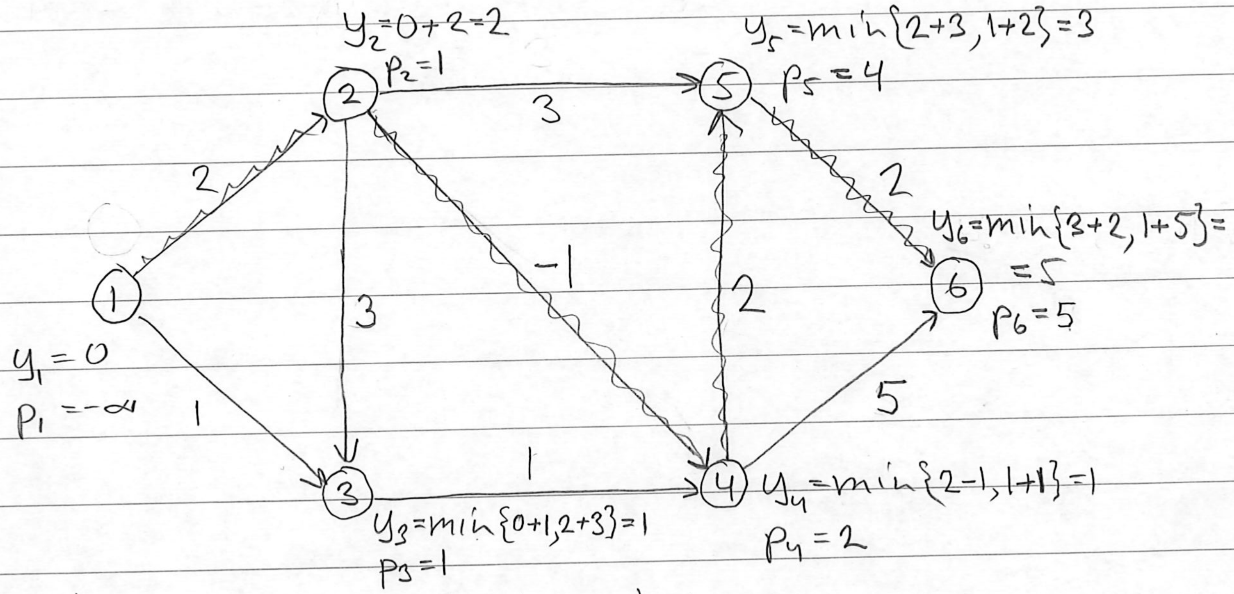
Svarar mot att studera för vilket i som $\min_i\{y_i + c_{ij}\}$ fås!

Uppnystning bakåt, från nod m till nod 1, ger en bv.

Hur lösa Bellmans ekvationer?
Beror på nätverkets egenskaper!

Acykliska nätverk

Nätverk utan cykler. Noderna kan då numreras så att för alla bågar (i,j) gäller att $j>i$. Detta kallas en topologisk numrering. Bellmans ekvationer kan därefter lösas i nodnummerordning mha "framåtsubstitution".



p_j = föregångare till nod j

bv-kostnad = 5

uppnystning ger bv: 1-2-4-5-6

även bv-träd fås!

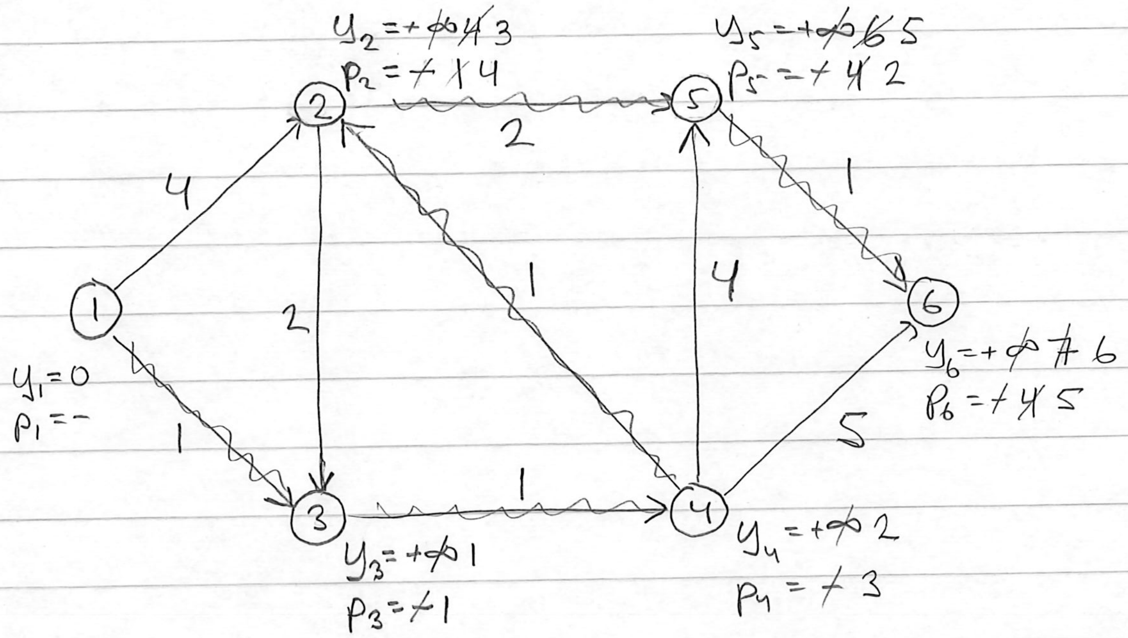
Obs: vid lösningen jämförs inkommande bågar till varje nod.

Icke-negativa bågkostnader

Avsök utgående bågar från noder i stigande nodprisordning, som kan konstrueras efterhand.

Initialt: $y_1 = 0$ och $y_j = +\infty$ för $j = 2, \dots, m$ samt $p_j = -$ för $j = 1, \dots, m$.

Uppdatera nodpris och föregångare när en kortare väg hittas.



Avsökningsordning: 1, 3, 4, 2, 5, 6

bv-kostnad = 6

uppnystning ger bv: 1-3-4-2-5-6

även bv-träd fås!

Kallas Dijkstras algoritm.

Det finns flera andra bv-algoritmer.
Bv-algoritmer ger typiskt bv-träd, även om endast en bv söks.

Det finns flera bv-liknande problem som kan lösas på liknande sätt:

- Dyraste väg i acykliskt nätverk

  Bellmans ekvationer:

  $$\begin{cases} y_1 = 0 \\ y_j = \max\limits_i \{y_i + c_{ij}\}, \quad j = 2, \ldots, m. \end{cases}$$

  Löses genom att avsöka noder i topologisk ordning och jämföra inkommande bågar.

- Väg med maximal tillförlitlighet

  $c_{ij} \in [0,1]$ är sannolikheten för att en båge inte går sönder. Finn en väg med maximal sannolikhet för att ingen båge längs vägen går sönder.

  Bellmans ekvationer

  $$\begin{cases} y_1 = 1 \\ y_j = \max\limits_i \{y_i \times c_{ij}\}, \quad j = 2, \ldots, m. \end{cases}$$

Löses med en släkting till Dijkstras algoritm.

Initialt: $y_1 = 1$ och $y_j = 0$, $j = 2, \ldots, m$.

Avsök noder i sjunkande nodprisordning.

- Väg med maximal kapacitet

$c_{ij} \geq 0$ är en bågkapacitet. En vägs kapacitet ges av den lägsta båg-kapaciteten längs vägen. Finn en väg där denna är maximal.

Bellmans ekvationer:

$$\begin{cases} y_1 = +\infty \\ y_j = \max\limits_i \min\{y_i, c_{ij}\}, \quad j = 2, \ldots, m. \end{cases}$$

Löses med en släkting till Dijkstras algoritm.

Initialt: $y_1 = +\infty$ och $y_j = 0$, $j = 2, \ldots, m$.

Avsök noder i sjunkande nodprisordning.