1 Motivacija

m sirovina, n proizvoda,

$\rho_i =$ cena jedinice i-te sirovine, $i = 1,2,\ldots,m$

$\sigma_j =$ cena jedinice j-tog proizvoda, $j = 1,2,\ldots,n$

$a_{i,j} =$ količina jedinica i-te sirovine u j-tom proizvodu
$i = 1,\ldots,m$, $j = 1,\ldots,n$

$c_j = \sigma_j - \sum_{i=1}^{m} \rho_i a_{i,j} =$ dobit od jedinice j-tog proizvoda,
$j = 1,2,\ldots,n$

$b_i =$ zaliha jedinica i-te sirovine, $i = 1,2,\ldots,m$

$x_j =$ količina jedinica j-tog proizvoda, $j = 1,2,\ldots,n$

1.1 Pogled proizvođača

Odrediti koliko jedinica x_j proizvoda $j = 1,2,\ldots,n$ treba proizvesti da bi se ostvarila maksimalna dobit ζ.

$$\zeta = \sum_{j=1}^{n} c_j\, x_j \to \max$$

$$\sum_{j=1}^{n} a_{i,j}\, x_j \leq b_i,\; i = 1,2,\ldots,m \tag{1}$$

$$x_j \geq 0,\; j = 1,2,\ldots,n$$

Problem (1) zovemo **problem linearnog programiranja**, za ovaj način zadavanja kažemo da je **standardni oblik**.

1.2 Pogled magacionera

Odrediti inventarsku vrednost w_i sirovine $i = 1,2,\ldots,m$ tako da ukupni lager bude minimalan, tako da vrednosti w_i ne budu manje od tržišnih vrednosti ρ_i, $i = 1,2,\ldots,m$ i da vrednost jedinice proizvoda j sa ugrađenim sirovinama vrednosti w_i ne bude manja od tržišne vrednosti jedinice proizvoda σ_j, $j = 1,2,\ldots,n$.

$$
\begin{aligned}
&\sum_{i=1}^{m} w_i b_i \to \min \\
&\sum_{i=1}^{m} w_i\, a_{i,j} \geq \sigma_j,\ j = 1,2,\ldots,n \\
&w_i \geq \rho_i,\ i = 1,2,\ldots,m
\end{aligned}
\tag{2}
$$

Ako se za problem (2) uvedu veličine "dodate vrednosti" jedinice sirovine i:

$$y_i = w_i - \rho_i,\ i = 1,2,\ldots,m,$$

dobija se problem

$$\xi = \sum_{i=1}^{m} y_i b_i \to \min$$

(3)
$$\sum_{i=1}^{m} y_i\, a_{i,j} \geq c_j,\ j = 1,2,\ldots,n$$
$$y_i \geq 0,\ i = 1,2,\ldots,m.$$

2 Primer

Fabrika proizvodi artikle A, B i C.

Za Proizvodnju artikla A treba 2 jedinice sirovine S1, 3 jedinice sirovine S2 i 4 jedinice sirovine S3. Za proizvod B treba redom 1, 3, 2 jedinice sirovina S1, S2 i S3. Za artikal C treba redom 3, 1, 4 jedinice sirovina S1, S2 i S3. Na raspolaganju nam je 10 jedinica sirovine S1, 25 jedinica sirovine S2 i 30 jedinica sirovine S3 dnevno.

Artikal A se na mašini M1 obrađuje 3 sata, a na mašini M2 4 sata. Artikal B se obrađuje na mašini M1 4 sata i na mašini M2 3 sata. Artikal C se obrađuje na mašini M1 2 sata i na mašini M2 3 sata. Mašine M1 i M2 mogu biti istovremeno angažovane na jednom artiklu. Na jednoj mašini se može obrađivati samo jedan artikal u jednom momentu.

Cena jednog komada artikla A je 8 novčanih jedinica, artikla B je 5 novčanih jedinica, artikla C je 7 novčanih jedinica.

Koliko dnevno treba proizvoditi kojeg proizvoda, da bi zarada bila maksimalna?

Uvešćemo veličine:

$x_1 =$ broj proizvedenih artikala A

$x_2 =$ broj proizvedenih artikala B

$x_3 =$ broj proizvedenih artikala C

$\zeta =$ zarada

$$\zeta = 8x_1 + 5x_2 + 7x_3 \to \max$$

$$\begin{array}{rcrcrcr} 2x_1 & + & x_2 & + & 3x_3 & \leq & 10 \\ 3x_1 & + & 3x_2 & + & 1x_3 & \leq & 25 \\ 4x_1 & + & 2x_2 & + & 4x_3 & \leq & 30 \\ 3x_1 & + & 4x_2 & + & 2x_3 & \leq & 24 \\ 4x_1 & + & 3x_2 & + & 3x_3 & \leq & 24 \end{array}$$

$$x_1 \geq 0, \;\; x_2 \geq 0, \;\; x_3 \geq 0.$$

3 Rešenje problema linearnog programiranja

Za problem (1) **skup dopustivih vrednosti** $\mathcal{S}$ je skup uređenih n-torki $(x_1, x_2, \ldots, x_n)$ koje zadovoljavaju

$$\sum_{j=1}^{n} a_{i,j}\, x_j \le b_i,\ i = 1,2,\ldots,m$$

$$x_j \ge 0,\ j = 1,2,\ldots,n.$$

Uređena n-torka $(x_1, x_2, \ldots, x_n)$ iz $\mathcal{S}$ za koju **funkcija dobiti**

$$\zeta = \sum_{j=1}^{n} c_j\, x_j$$

ostvaruje maksimum je **rešenje problema** (1).

Ako je skup dopustivih vrednosti prazan, $\mathcal{S} = \emptyset$, problem (1) nema rešenje, kažemo da je **nemoguć** (*infeasible*, EN).

Ako za proizvoljan broj $M > 0$ postoji n-torka iz $\mathcal{S}$ za koju je $\zeta > M$, problem (1) nema rešenje, kažemo da je **neograničen** (*unbounded*, EN), pišemo $\zeta \to \infty$.

Ako postoji broj $R > 0$ takav da za sve elemente iz $\mathcal{S}$ važi

$$x_1^2 + x_2^2 + \cdots + x_n^2 < R^2,$$

kažemo da je $\mathcal{S}$ **ograničen**. U protivnom $\mathcal{S}$ je **neograničen**.

Funkcija dobiti je neprekidna i nad ograničenim, zatvorenim, skupom $\mathcal{S}$ dostiže maksimum.

Ako je problem (1) neograničen, onda je $\mathcal{S}$ neograničen.

Obrnuto nije tačno.

4 Simplex algoritam

Posmatramo problem (1) i dovodimo ga na **standardni oblik sa jednakostima** tako što levoj strani nejednakosti i dodajemo **dodatnu promenljivu** w_i, $w_i \geq 0$, $i = 1,2,\ldots,m$ i nejednakost pretvaramo u jednakost.

Ograničenja nenegativnosti promenljivih

$$x_j \geq 0,\ j = 1,2,\ldots,n,\ w_i \geq 0,\ i = 1,2,\ldots,m$$

odvajamo i ostaje nam sistema m linearnih jednačina:

$$\sum_{j=1}^{n} a_{i,j}\, x_j + w_i = b_i,\ i = 1,2,\ldots,m. \tag{4}$$

4.1 Rečnik

Kad iz (4) izrazimo m promenljivih (**bazične**) preko preostalih n promenljivih (**nebazične**) i funkciju dobiti izrazimo preko nebazičnih promenljivih, to je **rečnik**.

Rečničko rešenje nekog rečnika dobijamo kad nebazičnim promenljivama dodelimo nule, a bazične promenljive i funkciju dobiti izračunamo iz (4).

Ako su u rečničkom rešenju izračunate vrednosti bazičnih promenljivih nenegativne, to rečničko rešenje pripada $\mathcal{S}$, ono je dopustivo, odnosno, za taj rečnik kažemo da je **dopustiv**.

Ako su u (4) sve $b_i \geq 0$, $i = 1, 2, \ldots, m$, izbor dodatnih promenljivih za bazične daje dopustiv rečnik.

Ako imamo dopustiv rečnik, skup dopustivih tačaka je neprazan: $\mathcal{S} \neq \emptyset$ i sledeći algoritam nalazi rešenje problema (1) ako ono postoji.

4.2 Algoritam

1. Odabrati bazične promenljive i naći dopustiv rečnik. Ako dopustiv rečnik ne postoji, onda $\mathcal{S} = \emptyset$, kraj.

2. Odabrati nebazičnu promenljivu čijim povećavanjem bi se povećala vrednost funkcije dobiti. Ako takva promenljiva ne postoji, onda je trenutni rečnik optimalan, kraj.

3. Iz rečnika odrediti najmanju gornju granicu odabrane nebazične promenljive. Ako takva granica ne postoji, onda $\zeta \to \infty$, kraj.

4. Ekvivalentnim transformacijama sistema napraviti rečnik u kome odabrana nebazična promenljiva menja mesto sa bazičnom promenljivom kod koje je pronađena najmanja gornja granica (pivotizacija). Pređi na korak 2.

4.3 Zapis rečnika

Dodatne promenljive w_i, $i = 1,2,\ldots,m$ su u Simplex algoritmu ravnopravne sa promenljivama x_j, možemo ih označavati x_{n+i}, $i = 1,2,\ldots,m$:

$$(x_1,\ldots,x_n,w_1,\ldots,w_m) = (x_1,\ldots,x_n,x_{n+1},\ldots,x_{n+m}).$$

Sa takvim oznakama početni rečnik je:

$$\begin{array}{rcl}
\zeta & = & 0 + \sum_{j=1}^{n} c_j x_j \\
\hline
x_{n+1} & = & b_1 - \sum_{j=1}^{n} a_{1,j} x_j \\
x_{n+2} & = & b_2 - \sum_{j=1}^{n} a_{2,j} x_j \\
\vdots & & \\
x_{n+m} & = & b_m - \sum_{j=1}^{n} a_{m,j} x_j
\end{array}$$

Rečničko rešenje koje odgovara početnom rečniku je:

$$(x_1,\ldots,x_n,w_1,\ldots,w_m) = (0,\ldots,0,b_1,\ldots,b_m).$$

U rečniku, skup baznih promenljivih označavamo sa $\mathcal{B}$, sa $\mathcal{N}$ označavamo skup nebaznih promenljivih.

Proizvoljni **rečnik je oblika**:

$$
\begin{array}{rcl}
\zeta & = & \bar{\zeta} + \sum_{j \in \mathcal{N}} \bar{c}_j x_j \\
\hline
x_i & = & \bar{b}_i - \sum_{j \in \mathcal{N}} \bar{a}_{i,j} x_j \,,\; i \in \mathcal{B}
\end{array} \tag{5}
$$

Svi rečnici su ekvivalentni između sebe i sa (4).

Proizvoljan rečnik je određen izborom skupa baznih promenljivih.

Postoji $\binom{n+m}{m}$ mogućih izbora m promenljivih od $n+m$.

U optimalnom rečniku $\bar{c}_j < 0$ za sve $j \in \mathcal{N}$.

Rečničko rešenje optimalnog rečnika je rešenje problema (1).

4.4 Pivotizacija

Promenljiva koja ulazi u bazu bira se iz skupa nebazičnih promenljivih za koje je $\bar{c}_k$ pozitivno. Neka je to x_k.

U novom rečničkom rešenju

$$x_k := \min_{i \in \mathcal{B}: \bar{a}_{i,k} > 0} \frac{\bar{b}_i}{\bar{a}_{i,k}}. \tag{6}$$

Promenljiva koja izlazi iz baze je jedna od promenljivih za koju je minimum u (6) ostvaren, recimo x_l.

U novom rečničkom rešenju $x_l := 0$. Ostale vrednosti u novom rečniku se dobijaju ekvivalentnim transformacijama.

Par promenljivih koja ulazi u bazu i koja izlazi iz baze zovemo **pivot**.

Postoji više mogućnosti za izbor pivota u jednoj iteraciji.

Pravilo kojim se bira pivot određuje simplex algoritam.

4.5 Inicijalizacija

Ako u (4) za neko i važi $b_i < 0$, onda izbor dodatnih promenljivih za bazične ne daje dopustiv rečnik.

Polazni dopustiv rečnik, ako postoji, dobijamo rešavanjem **pomoćnog problema**:

$$
\begin{aligned}
\zeta_1 = -x_0 &\to \max \\
\sum_{j=1}^{n} a_{i,j}\, x_j - x_0 &\le b_i,\ i = 1,2,\ldots,m \qquad (7)\\
x_j &\ge 0,\ j = 0,1,2,\ldots,n.
\end{aligned}
$$

Uzimanje dovoljno velikog x_0 daje dopustivo rešenje za (7). Polazni dopustiv rečnik za (7) se dobija pivotizacijom u kojoj x_0 ulazi u bazu, a promenljiva w_i kod koje je najnegativnije b_i izlazi iz baze.

Pomoćni problem ima rešenje. Ako u optimalnom rečniku za (7) imamo $x_0 = 0$, onda smo dobili dopustiv rečnik za (1), inače $\mathcal{S} = \emptyset$.

4.6 Simplex tabela

Simplex algoritam ćemo, zbog jednostavnosti, izvršavati na **Simplex tabelama** koje su ekvivalentne rečnicima. Na primer: problem

$$
\begin{array}{rcrcrcl}
\zeta & = & 2x_1 & - & x_2 & \to & \max \\
 & & x_1 & - & x_2 & \leq & -1 \\
 & & 2\,x_1 & + & x_2 & \leq & 3 \\
 & & x_1 & + & 2\,x_2 & \leq & 4 \\
 & & & x_1 \geq 0 & & x_2 \geq 0 &
\end{array} \tag{8}
$$

dovodimo na standardni oblik sa $=$, postavljamo pomoćni problem:

$$
\begin{array}{rcrcrcrcrcrcrcl}
\zeta_1 & = & & & & & & & & & & - & x_0 & \to & \max \\
\zeta & = & 2x_1 & - & x_2 & & & & & & & & & \to & \max \\
 & & x_1 & - & x_2 & + & w_1 & & & & & - & x_0 & = & -1 \\
 & & 2\,x_1 & + & x_2 & & & + & w_2 & & & - & x_0 & = & 3 \\
 & & x_1 & + & 2\,x_2 & & & & & + & w_3 & - & x_0 & = & 4
\end{array}
$$

uz zahteve nenegativnosti: $x_1 \geq 0, x_2 \geq 0, w_1 \geq 0, w_2 \geq 0, w_3 \geq 0, x_0 \geq 0$.
Početni rečnik i tabela koja mu odgovara su:

$$
\begin{array}{rcrcrcrc}
\zeta_1 = & & & & & & & - x_0 \\
\hline
\zeta = & & & 2x_1 & - & x_2 & & \\
\hline
w_1 = & -1 & - & x_1 & + & x_2 & & + x_0 \\
w_2 = & 3 & - & 2x_1 & - & x_2 & & + x_0 \\
w_3 = & 4 & - & x_1 & - & 2x_2 & & + x_0
\end{array}
$$

$$
\begin{array}{r|cccccc|r}
-1 & x_1 & x_2 & w_1 & w_2 & w_3 & x_0 & \\
\hline
w_1 & 1 & -1 & 1 & 0 & 0 & \mathbf{-1} & -1 \\
w_2 & 2 & 1 & 0 & 1 & 0 & -1 & 3 \\
w_3 & 1 & 2 & 0 & 0 & 1 & -1 & 4 \\
\hline
\zeta & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\
\zeta_1 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{array}
$$

Dalje idu pivotizacije:

$$
\begin{array}{r|cccccc|r}
0 & x_1 & x_2 & w_1 & w_2 & w_3 & x_0 & \\
\hline
x_0 & -1 & \mathbf{1} & -1 & 0 & 0 & 1 & 1 \\
w_2 & 1 & 2 & -1 & 1 & 0 & 0 & 4 \\
w_3 & 0 & 3 & -1 & 0 & 1 & 0 & 5 \\
\hline
 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\
 & 1 & -1 & 1 & 0 & 0 & 0 & -1
\end{array}
$$

$$
\begin{array}{r|ccccc|r}
1 & x_1 & x_2 & w_1 & w_2 & w_3 & \\
\hline
x_2 & -1 & 1 & -1 & 0 & 0 & 1 \\
w_2 & \mathbf{3} & 0 & 1 & 1 & 0 & 2 \\
w_3 & 3 & 0 & 2 & 0 & 1 & 2 \\
\hline
 & -1 & 0 & 1 & 0 & 0 & -1
\end{array}
$$

U poslednjoj tabeli je x_0 izašlo iz baze i time postalo 0 u rečničkom rešenju. Pošto $x_0 \geq 0$, time smo dobili rešenje pomoćnog problema i tu je kraj prve faze: $\mathcal{S} \neq \emptyset$. Imamo dopustivu tabelu početnog problema. Stoga smo tabelu 1 pisali bez kolone x_0 i vrste ζ_1, da bismo nastavili drugu fazu. Posle pivotizacije dobijamo optimalnu tabelu:

2	x_1	x_2	w_1	w_2	w_3	
x_2	0	1	$-2/3$	$1/3$	0	$5/3$
x_1	1	0	$1/3$	$1/3$	0	$2/3$
w_3	0	0	1	-1	1	0
	0	0	$4/3$	$1/3$	0	$-1/3$

Rešenje je:

$x_1^* = \frac{2}{3}$, $x_2^* = \frac{5}{3}$, $\zeta^* = -\frac{1}{3}$.

Rečnik koji odgovara je:

$$\begin{array}{rcrcrcr}
\zeta & = & -\frac{1}{3} & - & \frac{4}{3}w_1 & - & \frac{1}{3}w_2 \\
\hline
x_2 & = & \frac{5}{3} & + & \frac{2}{3}w_1 & - & \frac{1}{3}w_2 \\
x_1 & = & \frac{2}{3} & - & \frac{1}{3}w_1 & - & \frac{1}{3}w_2 \\
w_3 & = & 0 & - & w_1 & + & w_2
\end{array}$$

5 Grafičko rešavanje

Ako u problemu (1) $n = 2$ ili $n = 3$, skup dopustivih vrednosti $\mathcal{S}$ se može nacrtati i lako se može videti šta je rešenje problema.

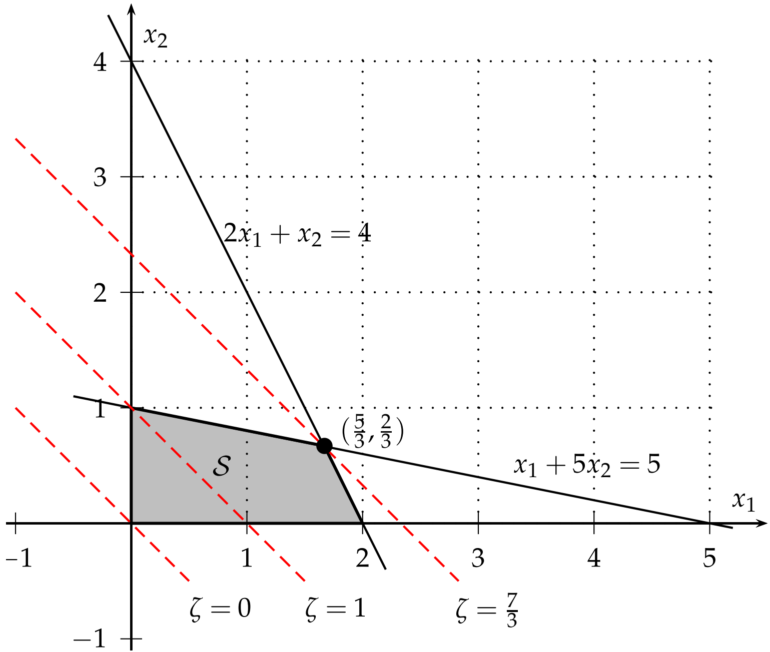
Na primer, za problem

$$\begin{array}{rcrcrcl} \zeta & = & x_1 & + & x_2 & \to & \max \\ & & x_1 & + & 5\,x_2 & \leq & 5 \\ & & 2\,x_1 & + & x_2 & \leq & 4 \\ & & \multicolumn{5}{c}{x_1 \geq 0 \quad x_2 \geq 0,} \end{array}$$

optimalno rešenje očitavamo sa crteža: $x_1^* = \dfrac{5}{3}$, $x_2^* = \dfrac{2}{3}$, $\zeta^* = \dfrac{7}{3}$.

Koordinate $(\frac{5}{3}, \frac{2}{3})$ dobijamo rešavanjem sistema

$$x_1 + 5x_2 = 5,\ 2x_1 + x_2 = 4.$$



Isprekidana linija predstavlja vrednosti funkcije $\zeta = x_1 + x_2$.

6 Degenerisani rečnici, degenerisani pivoti

Rečnik (5) je **degenerisan** ako za neko $i \in \mathcal{B}$ važi $\bar{b}_i = 0$.

U degenerisanom rečniku pivot (x_k, x_l) je **degenerisan** ako je vrednost (6) promenljive koja ulazi u bazu u novom rečniku jednaka 0.

Na primer, ovo je degenerisani rečnik:

$$
\begin{array}{rcrcrcrcr}
\zeta & = & 3 & - & \frac{1}{2}\,x_1 & + & 2\,x_2 & - & \frac{3}{2}\,w_1 \\
\hline
x_3 & = & 1 & - & \frac{1}{2}\,x_1 & & & - & \frac{1}{2}\,w_1 \\
w_2 & = & & & x_1 & - & x_2 & + & w_1
\end{array}
$$

Rečničko rešenje je: $(x_1, x_2, x_3, w_1, w_2) = (0,0,1,0,0)$.

U bazu ulazi x_2, izlazi w_2, to je degenerisani pivot.

Pivotizacija daje novi degenerisani rečnik sa istim rečničkim rešenjem $(x_1, x_2, x_3, w_1, w_2) = (0,0,1,0,0)$.

$$
\begin{array}{rcrcrcrcr}
\zeta & = & 3 & + & \frac{3}{2}\,x_1 & - & 2\,w_2 & + & \frac{1}{2}\,w_1 \\
\hline
x_3 & = & 1 & - & \frac{1}{2}\,x_1 & & & - & \frac{1}{2}\,w_1 \\
x_2 & = & & & x_1 & - & w_2 & + & w_1
\end{array}
$$

Vrednost funkcije dobiti se nije promenila. Iako je i ovo degenerisani rečnik, sledeći pivot (x_1, x_3) nije degenerisan:

$$
\begin{array}{rcrcrcrcr}
\zeta & = & 6 & - & 3\,x_3 & - & 2\,w_2 & - & w_1 \\
\hline
x_1 & = & 2 & - & 2\,x_3 & & & - & w_1 \\
x_2 & = & 2 & - & 2\,x_3 & - & w_2 & &
\end{array}
$$

Optimalan rečnik, $(x_1^*, x_2^*, x_3^*, w_1^*, w_2^*) = (2,2,0,0,0)$, $\zeta^* = 6$.

Pravilo za izbor pivota koje za promenljivu koja ulazi u bazu bira promenljivu sa najvećim koeficijentom $\bar{c}_j$, a od onih koje daju minimum u (6) promenljivu sa najmanjim rednim brojem, može dovesti do kruženja u rečnicima, odnosno, da se vratimo u rečnik u kome smo bili.

6.1 Anticiklin

U novom rečničkom rešenju posle degenerisanog pivota vrednost funkcije dobiti ostaje ista. Za problem (1) broj mogućih rečnika je konačan: $\binom{n+m}{m}$. *Simplex algoritam neće jedino završiti ako se stalno vraća u rečnik u kome je bio, a to je moguće samo u nizu degenerisanih pivota, jer se pri pivotizaciji nedegenerisanim pivotom vrednost funkcije dobiti povećava*. Inače Simplex algoritam mora završiti.

Blandovo pravilo za izbor pivota:

Pri izboru promenljive koja će ući u bazu, birati promenljivu sa pozitivnim koeficijentom $\bar{c}_j$ koja ima najmanji redni broj.

Za izbor promenljive koja će izaći iz baze među bazičnim promenljivama koje daju minimalni količnik u (6)*, izabrati promenljivu sa najmanjim rednim brojem.*

Blandovo pravilo za izbor pivota sprečava kruženje u rečnicima.

7 Dualnost

7.1 Motivacija na primeru

Dopustivo rešenje za problem linearnog programiranja

$$
(9)\quad
\begin{array}{rrcrcrcrcll}
 & \zeta & = & 4\,x_1 & + & x_2 & + & 3\,x_3 & \to & \max & \\
(9.1) & & & x_1 & + & 4\,x_2 & & & \leq & 1 & /\cdot 2\\
(9.2) & & & 3\,x_1 & - & x_2 & + & x_3 & \leq & 3 & /\cdot 3\\
 & & & \multicolumn{6}{c}{x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0} & &
\end{array}
$$

recimo $(x_1, x_2, x_3) = (1, 0, 0)$, daje **donju granicu**: $4 \leq \zeta^*$.
Dopustivo rešenje $(x_1, x_2, x_3) = (0, 0, 3)$ daje bolju granicu: $9 \leq \zeta^*$.
Ako pomnožimo prvu nejednakost sa 2 i saberemo sa drugom pomnoženom sa 3 dobijamo **gornju granicu**: $11x_1 + 5x_2 + 3x_3 \leq 11$. Naime, $4x_1 + x_2 + 3x_3 \leq 11x_1 + 5x_2 + 3x_3$, odakle $\zeta^* \leq 11$.

Ako pomnožimo nejednakosti (9.1) i (9.2) nenegativnim brojevima y_1 i y_2 i saberemo, dobijamo

$$
\begin{array}{rcrcrcl}
x_1 & + & 4\,x_2 & & & \leq & 1 \qquad / \cdot y_1 \\
3\,x_1 & - & x_2 & + & x_3 & \leq & 3 \qquad / \cdot y_2 \\
\hline
(y_1 + 3y_2)\,x_1 & + & (4y_1 - y_2)\,x_2 & + & (y_2)\,x_3 & \leq & y_1 + 3y_2
\end{array}
$$

Možemo tražiti da koeficijenti uz x_j budu barem koliki su u funkciji dobiti i da je granica, data desnom stranom nejednakosti, minimalna. Tako dobijamo problem linearnog programiranja:

$$
\begin{array}{rcrcrcl}
\zeta & = & y_1 & + & 3\,y_2 & \to & \min \\
 & & y_1 & + & 3\,y_2 & \geq & 4 \\
 & & 4\,y_1 & - & y_2 & \geq & 1 \\
 & & & & y_2 & \geq & 3 \\
 & & \multicolumn{5}{c}{y_1 \geq 0, \quad y_2 \geq 0}
\end{array}
\tag{10}
$$

Dobili smo **dualni problem**. Slično je u opštem slučaju.

Za problem (1) **dual** je problem (3)

$$\zeta = \sum_{j=1}^{n} c_j\, x_j \to \max$$

$$\sum_{j=1}^{n} a_{i,j}\, x_j \le b_i,\ i = 1,2,\dots,m$$

$$x_j \ge 0,\ j = 1,2,\dots,n$$

$$\xi = \sum_{i=1}^{m} y_i\, b_i \to \min$$

$$\sum_{i=1}^{m} y_i\, a_{i,j} \ge c_j,\ j = 1,2,\dots,n$$

$$y_i \ge 0,\ i = 1,2,\dots,m.$$

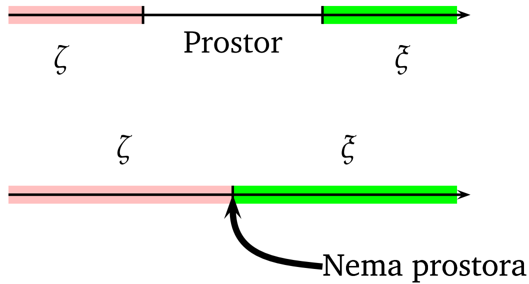
Levi problem zovemo **primar**, desni je njegov **dual**.

Dual od duala je primar.

Slaba teorema dualnosti:

Za dopustive vrednosti primara $(x_1,\dots,x_n)$ i duala $(y_1,\dots,y_m)$ važi

$$\zeta = \sum_{j=1}^{n} c_j\, x_j \le \sum_{i=1}^{m} y_i\, b_i = \xi\,. \tag{11}$$



Da li postoji prostor između ζ i ξ
za dopustive vrednosti primara i duala?

Jaka teorema dualnosti:
Ako primar ima optimalno rešenje $(x_1^*, \ldots, x_n^*)$, onda i dual ima optimalno rešenje $(y_1^*, \ldots, y_m^*)$ i važi

$$\zeta^* = \sum_{j=1}^{n} c_j \, x_j^* = \sum_{i=1}^{m} y_i^* \, b_i = \xi^* \, . \tag{12}$$

Dokaz jake teoreme dualnosti ćemo objasniti na primeru.

Dovedimo probleme (9) i (10) na standardni oblik sa jednakostima i napišimo početne rečnike. Dodatne promenljive duala su $z_j, j = 1,2,3$.

$$
\begin{array}{rcrcrcrcr}
\zeta & = & & & 4x_1 & + & x_2 & + & 3x_3 \\
\hline
w_1 & = & 1 & - & x_1 & - & 4x_2 & & \\
w_2 & = & 3 & - & 3x_1 & + & x_2 & - & x_3
\end{array}
\qquad
\begin{array}{rcrcrcr}
-\xi & = & & & -y_1 & - & 3y_2 \\
\hline
z_1 & = & -4 & + & y_1 & + & 3y_2 \\
z_2 & = & -1 & + & 4y_1 & - & y_2 \\
z_3 & = & -3 & & & + & y_2
\end{array}
$$

Ako posmatramo matrice koeficijenata sa desne strane jednakosti uočavamo **negativno transponovanje**.

$$
\begin{bmatrix} 0 & 4 & 1 & 3 \\ 1 & -1 & -4 & 0 \\ 3 & -3 & 1 & -1 \end{bmatrix}
\overset{\text{neg. tr.}}{\longleftarrow - \longrightarrow}
\begin{bmatrix} 0 & -1 & -3 \\ -4 & 1 & 3 \\ -1 & 4 & -1 \\ -3 & 0 & 1 \end{bmatrix}
$$

Posle pivotizacije primara (x_3, w_2) i duala (y_2, z_3) dobijamo

$$\begin{array}{rcrcrcrcr}
\zeta & = & 9 & - & 5x_1 & + & 4x_2 & - & 3w_2 \\
\hline
w_1 & = & 1 & - & x_1 & - & 4x_2 & & \\
x_3 & = & 3 & - & 3x_1 & + & x_2 & - & w_2
\end{array}
\qquad
\begin{array}{rcrcrcr}
-\xi & = & -9 & - & y_1 & - & 3z_3 \\
\hline
z_1 & = & 5 & + & y_1 & + & 3z_3 \\
z_2 & = & -4 & + & 4y_1 & - & z_3 \\
y_2 & = & 3 & & & + & z_3
\end{array}$$

Opet imamo negativno transponovanje. Pivotizacija primara (x_2, w_1) i duala (y_1, z_2) daje optimalne rečnike primara i duala.

$$\begin{array}{rcrcrcrcr}
\zeta & = & 10 & - & 6x_1 & - & w_1 & - & 3\,w_2 \\
\hline
x_2 & = & \frac{1}{4} & - & \frac{1}{4}x_1 & - & \frac{1}{4}w_1 & & \\
x_3 & = & \frac{13}{4} & - & \frac{13}{4}x_1 & - & \frac{1}{4}w_1 & - & w_2
\end{array}
\qquad
\begin{array}{rcrcrcr}
-\xi & = & -10 & - & \frac{1}{4}z_2 & - & \frac{13}{4}z_3 \\
\hline
z_1 & = & 6 & + & \frac{1}{4}z_2 & + & \frac{13}{4}z_3 \\
y_1 & = & 1 & + & \frac{1}{4}z_2 & + & \frac{1}{4}z_3 \\
y_2 & = & 3 & & & + & z_3
\end{array}$$

Pivotizaciji primara (x_j, w_i) odgovara pivotizacija duala (y_i, z_j).

7.2 Negativno transponovanje

Svakom rečniku primara odgovara rečnik duala u kome su koeficijenti u rečniku dobijeni transponovanjem i množenjem sa -1 od koeficijenata rečnika primara (**negativno transponovanje**).

Bazične promenljive primara su nebazične promenljive duala sa slovima z i x zamenjenim, odnosno y i w zamenjenim i obrnuto.

Svakoj pivotizaciji primara odgovara pivotizacija duala koja očuvava osobinu negativnog transponovanja.

Ako rečniku primara negativnim transponovanjem odgovara dualni dopustiv rečnik, kažemo da je primarni rečnik **dualno dopustiv**. Očigledno, rečnik koji je i primarno i dualno dopustiv je optimalan i za primar i za dual. Zbog negativnog transponovanja sledi $\zeta = \xi$ i to je kraj dokaza jake teoreme dualnosti.

7.3 Komplementarnost dodatnih promenljivih

Neka je $(x_1, \ldots, x_n)$ dopustivo rešenje primara (1) i neka je $(y_1, \ldots, y_m)$ dopustivo rešenje duala (3).

Neka su $(w_1, \ldots, w_m)$ odgovarajuće dodatne promenljive primara i $(z_1, \ldots, z_n)$ odgovarajuće dodatne promenljive duala.

$(x_1, \ldots, x_n)$ je optimalno rešenje primara i $(y_1, \ldots, y_m)$ je optimalno rešenje duala ako i samo ako

$$\begin{aligned} x_j z_j &= 0 \text{ za } j = 1, 2, \ldots, n \\ y_i w_i &= 0 \text{ za } i = 1, 2, \ldots, m. \end{aligned} \tag{13}$$

Dokaz.

Iskaz $x_j z_j = 0$ zapravo tvrdi da se barem jedna od x_j i z_j anulira za svako j. Pođimo od dokaza teoreme slabe dualnosti. Teorema jake dualnosti kaže da za optimalno rešenje važi jednakost $\zeta = \xi$.

$$\zeta = \sum_{j=1}^{n} c_j x_j \leq \sum_{j=1}^{n} \left(\sum_{i=1}^{m} y_i a_{i,j} \right) x_j = \tag{14}$$

$$= \sum_{i=1}^{m} \left(\sum_{j=1}^{n} a_{i,j} x_j \right) y_i \leq$$

$$\leq \sum_{i=1}^{m} b_i y_i = \xi \tag{15}$$

Zbog nenegativnosti x_j sledi da u (14) imamo jednakost akko

$$x_j = 0 \;\text{ ili }\; c_j = \sum_{i=1}^{m} y_i a_{i,j} \Leftrightarrow z_j = 0, \text{ za } j = 1,2,\ldots,n.$$

Zbog nenegativnosti y_i sledi da u (15) imamo jednakost akko

$$y_i = 0 \;\text{ ili }\; b_i = \sum_{j=1}^{n} a_{i,j} x_j \Leftrightarrow w_i = 0, \text{ za } i = 1,2,\ldots,m.$$

7.4 Dualni simplex algoritam

Mnogi problemi se lakše rešavaju tako što se pređe na dual, reši se dual, pa se iz optimalnog rečnika duala, koristeći osobinu negativnog transponovanja, pročita rešenje primara.

Umesto da se pređe na dual, može se raditi sa rečnicima primara tako što se simulira Simplex algoritam na dualu. To je **Dualni simplex algoritam**, primenjuje se kad je rečnik dualno dopustiv.

Dualne pivotizacije se izvode isto kao primarne, a pivot se određuje tako što se prvo bira promenljiva x_l koja će izaći iz baze, treba da ima negativnu vrednost u rečničkom rešenju.

Potom se bira promenljiva x_k koja će ući u bazu, kao promenljiva za koju se ostvaruje minimum

$$x_k := \min_{j \in \mathcal{N} : \bar{a}_{l,j} < 0} -\frac{\bar{c}_j}{\bar{a}_{l,j}}. \tag{16}$$

Pri izboru pivota možemo se držati ekvivalenta Blandovog pravila ako želimo da sprečimo kruženje u rečnicima.

7.5 Inicijalizacija za Dualni simplex algoritam

Ako se dobije problem kod kojeg izbor dodatnih promenljivih za bazične ne daje dualno dopustiv rečnik, može se prvo rešiti Dualnim simplex algoritmom **novi problem** sa istim $\mathcal{S}$ i funkcijom dobiti koja daje dualno dopustiv rečnik, recimo: $-\eta = x_1 + \cdots + x_n \to \min$.

Dual za novi problem sigurno nije nemoguć. Jedini slučaj da za novi problem Simplex algoritam ne nađe rešenje je da je dual neograničen.

Ali, tada je na osnovu slabe teoreme dualnosti primar nemoguć.

Kad se dobije rešenje novog problema, dobijeno je dopustivo rešenje početnog primara, prelazi se na njegovo rešavanje Simplex algoritmom sa originalnom funkcijom dobiti ζ.

7.6 Duali problema u opštem obliku

Za problem sa jednakostima razdvajamo "=" na "$\leq$" i "$\geq$".

$$\zeta = \sum_{j=1}^{n} c_j\, x_j \to \max$$

$$\sum_{j=1}^{n} a_{i,j}\, x_j = b_i,\ i = 1,2,\ldots,m$$

$$x_j \geq 0,\ j = 1,2,\ldots,n$$

$$\zeta = \sum_{j=1}^{n} c_j\, x_j \to \max$$

$$\sum_{j=1}^{n} a_{i,j}\, x_j \leq b_i,\ i = 1,2,\ldots,m$$

$$\sum_{j=1}^{n} a_{i,j}\, x_j \geq b_i,\ i = 1,2,\ldots,m$$

$$x_j \geq 0,\ j = 1,2,\ldots,n$$

Nejednakosti $\sum_{j=1}^{n} a_{i,j}\, x_j \geq b_i$ množimo sa (-1). Prelazimo na dual.

Dualne promenljive delimo na dve grupe: y_i^+ koje odgovaraju nejednakostima sa $\leq$ i y_i^- koje odgovaraju nejednakostima sa $\geq$.

$$\xi = \sum_{i=1}^{m} y_i^+ b_i - \sum_{i=1}^{m} y_i^- b_i \to \min$$

$$\sum_{i=1}^{m} y_i^+ a_{i,j} - \sum_{i=1}^{m} y_i^- a_{i,j} \geq c_j,\ j = 1,2,\dots,n$$

$$y_i^+ \geq 0,\ y_i^- \geq 0,\ i = 1,2,\dots,m.$$

Dobili smo dual koji smenom $y_i = y_i^+ - y_i^-, i = 1,\dots,m$, uprostimo:

$$\xi = \sum_{i=1}^{m} y_i b_i \to \min$$

$$\sum_{i=1}^{m} y_i a_{i,j} \geq c_j,\ j = 1,2,\dots,n$$

U problemu koji smo dobili za promenljive ne tražimo nenegativnost. Kažemo da su te promenljive **slobodne**.

Dual od poslednjeg problema je početni primar sa jednakostima.

Na osnovu toga možemo izvući pravila za postavljanje duala:

Primar	Dual
jednakosti u uslovima	slobodne promenljive
nejednakosti u uslovima	nenegativne promenljive
slobodne promenljive	jednakosti u uslovima
nenegativne promenljive	nejednakosti u uslovima

Na primer, evo jedan par primara i duala u opštem obliku:

$$(P) \quad \begin{aligned} \zeta &= -2x_1 + x_2 \to \max \\ & -\ x_1 + x_2 = 1 \\ & -3x_1 + x_2 \leq -1 \\ & \quad x_1 \geq 0 \end{aligned} \qquad\qquad (D) \quad \begin{aligned} \xi &= y_1 - y_2 \to \min \\ & -y_1 - 3y_2 \geq -2 \\ & \quad y_1 + y_2 = 1 \\ & \quad y_2 \geq 0 \end{aligned}$$

8 Matrični zapis

Polazimo od problema (1),

$$\zeta = \sum_{j=1}^{n} c_j \, x_j \to \max$$

$$\sum_{j=1}^{n} a_{i,j} \, x_j \leq b_i, \; i = 1,2,\ldots,m$$

$$x_j \geq 0, \; j = 1,2,\ldots,n.$$

Dovodimo ga na standardni oblik sa jednakostima.

Dodatne promenljive označavamo:

$$w_i = x_{n+i} = b_i - \sum_{j=1}^{n} a_{i,j} \, x_j \,, i = 1,\ldots,m.$$

Problem (1) je ekvivalentan problemu:

$$\begin{aligned} \zeta = c^T x &\to \max \\ A\,x &= b \\ x &\geq 0, \end{aligned} \tag{17}$$

gde je

$$A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & \cdots & a_{m,n} & 0 & \cdots & 1 \end{bmatrix}, b = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}, c = \begin{bmatrix} c_1 \\ \vdots \\ c_n \\ 0 \\ \vdots \\ 0 \end{bmatrix}, x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ x_{n+1} \\ \vdots \\ x_{n+m} \end{bmatrix}$$

Koristimo osobinu množenja matrica $A\,x = b$, da se proizvod ne menja ako se kolone matrice A i vrste matrice x permutuju na isti način. Koristimo i množenje matrica po blokovima.

Permutujemo matricu A tako da se u levi deo postave kolone baz̆ičnih promenljivih $\mathcal{B}$, one čine podmatricu B, a u desni deo kolone nebazičnih promenljivih $\mathcal{N}$, one čine podmatricu N. $A = [B\ N]$

Istim redosledom permutujemo vrste matrica $x = \begin{bmatrix} x_B \\ x_N \end{bmatrix}$ i $c = \begin{bmatrix} c_B \\ c_N \end{bmatrix}$.

Onda su jednakosti iz (17) ekvivalentne sa

$$\zeta = c^T x = [c_B^T\ c_N^T] \begin{bmatrix} x_B \\ x_N \end{bmatrix} = c_B^T x_B + c_N^T x_N \tag{18}$$

$$b = Ax = [B\ N] \begin{bmatrix} x_B \\ x_N \end{bmatrix} = Bx_B + Nx_N \tag{19}$$

Pri tome je matrica B invertibilna, jer je sistem (17) rešiv po x_B.

Rečnik se dobija rešavanjem x_B i ζ preko x_N iz (19) i (18).

$$
\begin{aligned}
\zeta &= c_B^T B^{-1} b - \left((B^{-1}N)^T c_B - c_N \right)^T x_N \\
x_B &= B^{-1} b - B^{-1} N x_N
\end{aligned}
\tag{20}
$$

Ako uporedimo matrični zapis (20) sa zapisom rečnika iz (5), imamo:

$$
\begin{array}{rcl}
\zeta &=& \bar{\zeta} + \sum_{j \in \mathcal{N}} \bar{c}_j x_j \\
\hline
x_i &=& \bar{b}_i - \sum_{j \in \mathcal{N}} \bar{a}_{i,j} x_j \, , \; i \in \mathcal{B}
\end{array}
\tag{5}
$$

$$
\begin{aligned}
\bar{\zeta} &= c_B^T B^{-1} b \\
[\bar{c}_j] &= c_N - (B^{-1}N)^T c_B \\
[\bar{b}_i] &= B^{-1} b \\
[\bar{a}_{i,j}] &= B^{-1} N
\end{aligned}
$$

U zapisu desno uglaste zagrade označavaju matricu elemenata dok indeksi idu: indeks i od 1 do m, j od 1 do n. Rečničko rešenje je

$$
x_N^* = 0, \;\; x_B^* = B^{-1} b, \;\; \zeta^* = c_B^T B^{-1} b.
\tag{21}
$$

Ako želimo da dobijemo rečnik duala koristeći negativno transponovanje i da zadržimo jednostavno indeksiranje za primenu komplementarnosti dualnih promenljivih, slično preimenovanju u primaru:

$$(x_1, \ldots, x_n, w_1, \ldots, w_m) \to (x_1, \ldots, x_n, x_{n+1}, \ldots, x_{n+m}),$$

dodatne dualne promenljive ćemo staviti na početak, a dualne promenljive ćemo preimenovati u nastavku:

$$(z_1, \ldots, z_n, y_1, \ldots, y_m) \to (z_1, \ldots, z_n, z_{n+1}, \ldots, z_{n+m}).$$

Dualni rečnik koji negativnim transponovanjem odgovara (20) je:

$$\begin{aligned} -\xi &= -c_B^T B^{-1} b - \left(B^{-1} b\right)^T z_B \\ z_N &= \left(B^{-1} N\right)^T c_B - c_N + \left(B^{-1} N\right)^T z_B. \end{aligned} \tag{22}$$

Rečničko rešenje dualnog rečnika:

$$z_B^* = 0,\ z_N^* = \left(B^{-1} N\right)^T c_B - c_N,\ -\xi^* = -\zeta^*. \tag{23}$$

Koristeći oznake sa zvezdicom: (21) i (23), zapis (20) i (22) je:

$$\begin{array}{lrcllrcl} & & \text{Primar} & & & & \text{Dual} & \\ (24) \quad & \zeta & = & \zeta^* - (z_N^*)^T x_N & \quad & -\xi & = & -\zeta^* - (x_B^*)^T z_B \\ & x_B & = & x_B^* - B^{-1} N x_N & & z_N & = & z_N^* + (B^{-1}N)^T z_B, \end{array}$$

gde je

$$(25) \qquad x_B^* = B^{-1}b, \ \zeta^* = c_B^T B^{-1} b, \ z_N^* = \left(B^{-1}N\right)^T c_B - c_N.$$

U zapisu (24) se vidi negativno transponovanje i simetričnost primarnog i odgovarajućeg dualnog rečnika.

Takođe je očigledno da je rečnik optimalan ako i samo ako je primarno dopustiv ($x_B^* \geq 0$) i dualno dopustiv ($z_N^* \geq 0$), zato što $x \geq 0$ i $z \geq 0$.

Iz (24) se vidi dokaz jake teoreme dualnosti.

9 Analiza osetljivosti

Za problem (17), iz izraza (25) vidimo da promena koeficijenata iz c u funkciji dobiti ne utiče na promenu vrednosti baznih promenljivih x_B^* u rečniku. Promena koeficijenata iz c utiče na optimalnost.
Ako je promena koeficijenata funkcije dobiti oblika

$$c := c + t\Delta c,$$

onda se menja z_N^*:

$$z_N^* := z_N^* + t\Delta z_N,$$

gde se, uobičajeno, Δc razdvaja na Δc_B i Δc_N i gde je

$$\Delta z_N = \left(B^{-1}N\right)^T \Delta c_B - \Delta c_N.$$

Interval vrednosti t za koje će rečničko rešenje (24) ostati optimalno dobijamo iz uslova $z_N^* \geq 0$.

Dobija se da t mora biti u intervalu:

$$\left(\min_{j\in\mathcal{N}} -\frac{\Delta z_j}{z_j^*}\right)^{-1} \leq t \leq \left(\max_{j\in\mathcal{N}} -\frac{\Delta z_j}{z_j^*}\right)^{-1}.$$

Slično, ako je promena b oblika

$$b := b + t\Delta b,$$

da bi rešenje (24) ostalo optimalno, treba da bude

$$x_B^* := x_B^* + t\Delta x_B \geq 0, \text{ gde je } \Delta x_B = B^{-1}\Delta b.$$

Odatle dobijamo da t mora biti u intervalu:

$$\left(\min_{i\in\mathcal{B}} -\frac{\Delta x_i}{x_i^*}\right)^{-1} \leq t \leq \left(\max_{i\in\mathcal{B}} -\frac{\Delta x_i}{x_i^*}\right)^{-1}.$$

10 Parametarski self dual simplex algoritam

Pođimo od matričnog zapisa rečnika:

$$\begin{array}{rcl} \zeta & = & c_B^T x_B^* - (z_N^*)^T x_N \\ \hline x_B & = & x_B^* - B^{-1} N x_N \end{array} \qquad \begin{array}{rcl} x_B^* & = & B^{-1} b, \\ z_N^* & = & \left(B^{-1} N\right)^T c_B - c_N. \end{array}$$

Ako ovaj rečnik nije primarno i dualno dopustiv, perturbujemo ga dodajući perturbacije $\bar{x}_B > 0$ i $\bar{z}_N > 0$ pomnožene sa μ na x_B^* i z_N^* redom. Dobijamo familiju rečnika na kojima vršimo pivotizacije.

$$\begin{array}{rcl} \zeta & = & c_B^T x_B^* - (z_N^* + \mu \bar{z}_N)^T x_N \\ x_B & = & (x_B^* + \mu \bar{x}_B) - B^{-1} N x_N. \end{array} \tag{26}$$

Parametarski self dual simplex algoritam generiše niz rečnika koji nastaju primarnom ili dualnom pivotizacijom kod vrednosti

$$\mu^* = \min\{\mu : z_N^* + \mu \bar{z}_N \geq 0 \;\text{ i }\; x_B^* + \mu \bar{x}_B \geq 0\}.$$

Na primer, za problem (8), perturbovani početni rečnik je:

$$\begin{array}{rcccc}
\zeta = & & -(-\mathbf{2}+\mu)x_1 & - & (1+\mu)x_2 \\
\hline
w_1 = & (-1+\mu) & - \quad x_1 & + & x_2 \\
w_2 = & (3+\mu) & - \quad 2x_1 & - & x_2 \\
w_3 = & (4+\mu) & - \quad x_1 & - & 2x_2
\end{array}$$

Pri tome smo uzeli $\bar{x}_B = [1,1,1]^T$, $\bar{z}_N = [1,1]^T$.

Početni rečnik je primarno i dualno dopustiv za $\mu \geq 2$. To smo dobili kao presek uslova nenegativnosti za sve koeficijente u zagradama.

Pivotizacija se vrši kod vrednosti μ^* koja iznosi 2. Biće to primarna pivotizacija, u bazu ulazi x_1. Iz baze izlazi w_1, jer se kod nje ostvaruje minimalni količnik (6) za $\mu = 2$:

$$\min_{i \in \mathcal{B}:\bar{a}_{i,k}>0} \frac{\bar{b}_i}{\bar{a}_{i,k}} = \min\left\{\frac{-1+\mu}{1}, \frac{3+\mu}{2}, \frac{4+\mu}{1}\right\} = \frac{-1+\mu}{1} = 1.$$

Dalje ćemo raditi sa tabelama, koeficijente uz μ ćemo dopisati.

$$\begin{array}{c|ccccc|cc}
 & x_1 & x_2 & w_1 & w_2 & w_3 & \multicolumn{2}{c}{2 \le \mu} \\
\hline
w_1 & \mathbf{1} & -1 & 1 & 0 & 0 & -1 & 1 \\
w_2 & 2 & 1 & 0 & 1 & 0 & 3 & 1 \\
w_3 & 1 & 2 & 0 & 0 & 1 & 4 & 1 \\
\hline
 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\
 & 1 & 1 & 0 & 0 & 0 & 0 & 0
\end{array}
\quad
\begin{array}{l}
\\
(-1 + 1\mu \ge 0 \Rightarrow \mu \ge 1) \\
(3 + 1\mu \ge 0 \Rightarrow \mu \ge -3) \\
\cdots \\
\\
\\
\end{array}$$

Sledeća pivotizacija je dualna, pivot je (x_2, x_1), vidi (16):

$$\begin{array}{c|ccccc|cc}
 & x_1 & x_2 & w_1 & w_2 & w_3 & \multicolumn{2}{c}{1 \le \mu \le 2} \\
\hline
x_1 & 1 & \mathbf{-1} & 1 & 0 & 0 & -1 & 1 \;\leftarrow \\
w_2 & 0 & 3 & -2 & 1 & 0 & 5 & -1 \\
w_3 & 0 & 3 & -1 & 0 & 1 & 5 & 0 \\
\hline
 & 0 & -1 & 2 & 0 & 0 & -2 & 2 \\
 & 0 & 2 & -1 & 0 & 0 & 1 & -1
\end{array}$$

U tabeli koja sledi pivot je (x_1, w_2)

	x_1	x_2	w_1	w_2	w_3	$\frac{1}{2} \leq \mu \leq 1$	
x_2	-1	1	-1	0	0	1	-1
w_2	$\mathbf{3}$	0	1	1	0	2	2
w_3	3	0	2	0	1	2	3
	-1	0	1	0	0	-1	1
	2	0	1	0	0	-1	1

Dobijamo optimalnu tabelu, jer za nju $\mu = 0$ daje optimalan rečnik:

	x_1	x_2	w_1	w_2	w_3	$-1 \leq \mu \leq \frac{1}{2}$	
x_2	0	1	$-2/3$	$1/3$	0	$5/3$	$-1/3$
x_1	1	0	$1/3$	$1/3$	0	$2/3$	$2/3$
w_3	0	0	1	-1	1	0	1
	0	0	$4/3$	$1/3$	0	$-1/3$	$5/3$
	0	0	$1/3$	$-2/3$	0	$-7/3$	$-1/3$

Očitavamo rešenje $x_1^* = \frac{2}{3}$, $x_2^* = \frac{5}{3}$, $\zeta^* = -\frac{1}{3}$.

11 Mrežni protok - minimizacija cene transporta

Posmatramo mrežu protoka neke robe datu povezanim usmerenim grafom sa skupom čvorova $\mathcal{N}$, skupom grana $\mathcal{A}$ i relacijom susedstva. Podrazumevaćemo da je susedstvo definisano imenima grana. Postojanje grane od a do b znači da je moguć protok robe od a do b.

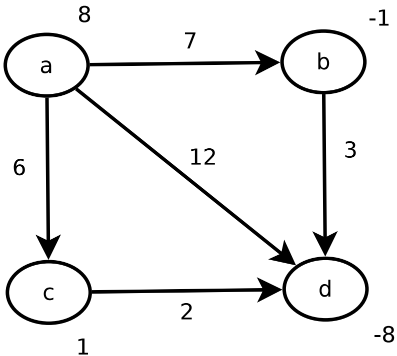
Posmatraćemo funkciju koja čvorovima pridružuje težine (na crtežu pored čvorova) koje predstavljaju zalihu robe u čvoru ako je težina pozitivna, odnosno narudžbinu robe ako je negativna.

Takođe ćemo granama pridružiti težine (pored grana) koje predstavljaju cene transporta jedinice robe (nenegativni brojevi).

Zadatak je nalaženje transporta koji će zadovoljiti narudžbine koristeći zalihe svih čvorova i ima najmanju cenu.

Rešavamo samo ako je graf povezan i zbir težina čvorova 0.

Primer problema mrežnog protoka



$$\mathcal{N} = \{\text{a,b,c,d}\}, \#\mathcal{N} = m = 4$$

$$\mathcal{A} = \{\text{ab,ac,ad,bd,cd}\}, \#\mathcal{A} = n = 5$$

$$\begin{array}{ccccccc} & & \text{a} & \text{b} & \text{c} & \text{d} & \\ b & = & [\; 8 & -1 & 1 & -8 \;]^T \end{array}$$

$$\begin{array}{cccccccc} & & \text{ab} & \text{ac} & \text{ad} & \text{bd} & \text{cd} & \\ c & = & [\; 7 & 6 & 12 & 3 & 2 \;]^T \end{array}$$

x_{ij} = količina robe koju ćemo prevesti od čvora i do j, $(i,j) \in \mathcal{A}$

c_{ij} = cena prevoza jedinice robe od čvora i do j, $(i,j) \in \mathcal{A}$

ζ = ukupna cena transporta

Za svaki čvor: ulaz $-$ izlaz = narudžbina = $-$zaliha

x_{ij} su nenegativni celi brojevi za $(i,j) \in \mathcal{A}$

$$
\begin{array}{rrcrcrcrcrcrcr}
& \zeta & = & 7\,x_{\mathsf{ab}} & + & 6\,x_{\mathsf{ac}} & + & 12\,x_{\mathsf{ad}} & + & 3\,x_{\mathsf{bd}} & + & 2\,x_{\mathsf{cd}} & \to & \min \\
\mathsf{a}: & & & -\,x_{\mathsf{ab}} & - & x_{\mathsf{ac}} & - & x_{\mathsf{ad}} & & & & & = & -8 \\
\mathsf{b}: & & & x_{\mathsf{ab}} & & & & & - & x_{\mathsf{bd}} & & & = & 1 \\
\mathsf{c}: & & & & & x_{\mathsf{ac}} & & & & & - & x_{\mathsf{cd}} & = & -1 \\
\mathsf{d}: & & & & & & & x_{\mathsf{ad}} & + & x_{\mathsf{bd}} & + & x_{\mathsf{cd}} & = & 8
\end{array}
$$

$$
x_{\mathsf{ab}} \geq 0, \quad x_{\mathsf{ac}} \geq 0, \quad x_{\mathsf{ad}} \geq 0, \quad x_{\mathsf{bd}} \geq 0, \quad x_{\mathsf{cd}} \geq 0
$$

$$
x_{\mathsf{ab}} \in \mathbb{Z}, \quad x_{\mathsf{ac}} \in \mathbb{Z}, \quad x_{\mathsf{ad}} \in \mathbb{Z}, \quad x_{\mathsf{bd}} \in \mathbb{Z}, \quad x_{\mathsf{cd}} \in \mathbb{Z}
$$

Problem mrežnog protoka je problem celobrojnog linearnog programiranja.

Ako su zalihe (narudžbine) celi brojevi i ako postoji rešenje, onda postoji celobrojno rešenje.

Ako rešenje ne postoji, problem je nedopustiv (nije neograničen).

U opštem slučaju

$$\zeta = \sum_{(i,j)\in\mathcal{A}} c_{ij}\, x_{ij} \to \min$$

$$\sum_{\substack{i \\ (i,k)\in\mathcal{A}}} x_{ik} - \sum_{\substack{j \\ (k,j)\in\mathcal{A}}} x_{kj} = -b_k\,, \;\text{ za }\; k \in \mathcal{N}$$

$$x_{ij} \geq 0\,,\; x_{ij} \in \mathbb{Z}\,, \;\text{ za }\; (i,j) \in \mathcal{A}$$

Koeficijente c_{ij}, x_{ij}, b_k smeštamo redom u vektor kolone c, x, b.

Primar	Dual
$\zeta = c^T x \to \min$	$-\xi = -b^T y \to \max$
$Ax = -b$	$A^T y + z = c$
$x \geq 0$	$z \geq 0$

Jednačine duala glase: $\forall (i,j) \in \mathcal{A}$, $-y_i + y_j + z_{ij} = c_{ij}$, $z_{ij} \geq 0$.

Formati matrica: $A \to m \times n$; $c,x,z \to n \times 1$; $b,y \to m \times 1$.

A je matrica sistema, $\text{rang}(A) = m - 1$.

$$A = \begin{bmatrix} -1 & -1 & -1 & & & \\ 1 & & & -1 & & \\ & 1 & & & & -1 \\ & & 1 & 1 & 1 & \end{bmatrix} \text{ je matrica incidencija grafa.}$$

Struktura matrice sistema je takva da proizvoljnu jednačinu oblika ulaz $-$ izlaz $= -$zaliha možemo izbaciti. Zato rečnik dobijamo kad $m-1$ promenljivu primara (bazične) izrazimo preko nebazičnih. Nebazične promenljive u rečničkom rešenju su 0.

U odgovarajućem dualnom rečniku z promenljive koje odgovaraju bazičnim promenljivama primara su 0.

Za povezan graf postoji pokrivajuće drvo sa $m-1$ grana. Grane pokrivajućeg drveta su bazične promenljive primara. Njihove vrednosti u rečničkom rešenju se dobijaju iz sistema jednačina zaliha za čvorove. Dobijeni protok nazivamo **balansirani** protok.

Ako su vrednosti bazičnih promenljivih u balansiranom protoku nenegativne, kažemo da je balansirani protok **dopustiv**.

Koristeći teoriju Simplex algoritma zaključujemo da ako postoji optimalno rešenje onda postoji i optimalno rešenje sa $m-1$ bazičnih promenljivih.

Osim, ako problem nije dopustiv, odnosno, ako ne postoji dopustiv protok, tada je $\mathcal{S}=\emptyset$. Zbog nenegativnosti cena nije moguć slučaj da je problem neograničen ($\zeta \to -\infty$ nije moguće).

U praktičnoj primeni balansirani protok dobijamo polazeći od "lista" pokrivajućeg drveta. Ako je zbir svih zaliha jednak 0 onda za dato pokrivajuće drvo postoji balansirani protok.

Zato što je jedna jednačina primara zavisna (i može da se izbaci), rešenje jednačina duala nije jedinstveno. Može se pokazati da je jedna od y_k promenljivih proizvoljna. Biramo da to bude y_k koja odgovara izbačenoj jednačini primara, čvor k nazivamo "koren".

Ostale y promenljive računamo iz jednačina duala koje odgovaraju granama pokrivajućeg drveta (bazičnim), vrednosti pišemo u čvorove. Potom koristeći ostale (nebazične) grane, koristeći jednačine duala koje njima odgovaraju, računamo z promenljive, njih pišemo pored grana.

Umesto u tabelama, vrednosti promenljivih pišemo na crtežu grafa. Na granu pišemo x vrednost ako je bazična, odnosno, z vrednost ako je nebazična a y vrednosti pišemo unutar čvorova.

Simplex algoritam ćemo izvoditi na crtežu grafa, bazične promenljive ćemo obeležavati crvenom bojom. Vodićemo samo rečnička rešenja koja odgovaraju izabranoj bazi.

Zamena nebazične i bazične promenljive je pivotizacija. Novo rečničko rešenje dobijamo primenom pravila kontura i pravila mostova.

Ako je balansirani protok dopustiv do optimuma dolazimo koristeći:

12 Primarni simplex algoritam

Primarna pivotizacija: U bazu ulazi grana koja ima negativnu z vrednost. Ta grana zatvara konturu sa baznim granama. Po toj konturi ažuriramo x vrednosti sa nekim t primenom pravila kontura.

Pravilo kontura: *Uslovi zaliha ostaju zadovoljeni ako se idući po konturi u zavisnosti od smera dodaje ili oduzima ista vrednost t.*

Biramo vrednost $t > 0$ tako da ni jedna od baznih promenljivih ne postane negativna, a bar jedna postane 0. Jedna od promenljivih koja postane 0 se bira da izađe iz baze. Izbor grane sa konture za ulazak u bazu očuvava pokrivajuće drvo. Novi rečnik će biti primarno dopustiv.

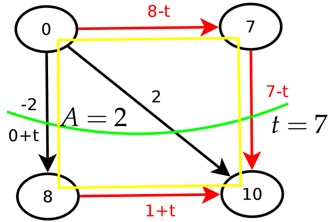
Vraćamo se na korak izbora negativne z vrednosti dok sve z vrednosti ne postanu nenegativne. Tada smo dobili dualno dopustiv rečnik, odnosno optimalan rečnik. Ako se ne vratimo u rečničko rešenje u kome smo bili, po teoriji Simplex algoritma, posle konačnog broja pivotizacija ćemo doći u optimalno rešenje.

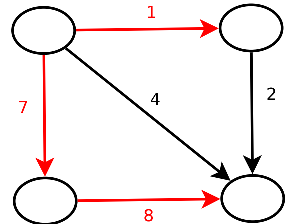
Dualne promenljive ažuriramo koristeći pravilo mostova. Ako granu odabranu da izađe iz baze izbacimo pre nego što ubacimo novu granu u bazu, čvorovi se, prema povezanosti, dele u dve grupe, između te dve grupe povlačimo granicu.

Pravilo mostova: *Pri odabranoj pivotizaciji, menjaju se z promenljive dualnog rečničkog rešenja samo za grane koje idu preko granice dodavanjem ili oduzimanjem iste vrednosti A u zavisnosti od smera prelaska preko granice.*

Vrednost $A > 0$ u pravilu mostova uzimamo kao suprotnu od z vrednosti grane koja ulazi u bazu, jer ona postaje 0.

Rešiti problem dat kao primer uzimajući početno pokrivajuće drvo: ab, bd, cd.





$\zeta^* = 1 \cdot 7 + 7 \cdot 6 + 8 \cdot 2 = 65$

Pivotizaciju vršimo na grani ac.

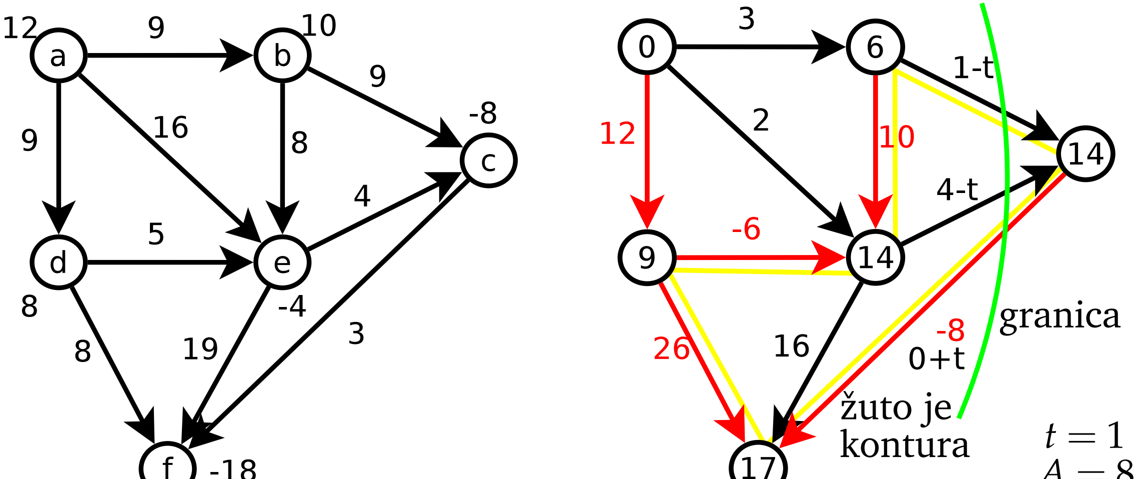
Pravilo kontura (žuta na slici) nam određuje t vrednost ($t = 7$). Pomoću pravila mostova preko granice (zelena) ažuriramo z vrednosti.

Kad se za dati problem minimizacije mrežnog protoka izračunaju x, y i z vrednosti rečničkog rešenja, ako su z vrednosti nenegativne ($z \geq 0$), a nisu sve x vrednosti nenegativne, kažemo da je dati plan protoka **dualno dopustiv** (a nije primarno dopustiv).

Za protok koji je dualno dopustiv, ako postoji rešenje, ono se može naći pomoću Dualnog simplex algoritma. Ili da nema rešenja ($\mathcal{S} = \emptyset$).

13 Dualni simplex algoritam

Rešiti problem mrežnog protoka (levo) polazeći od pokrivajućeg drveta ad, be, cf, de, df. Izračunali smo (desno) x, y, z vrednosti.

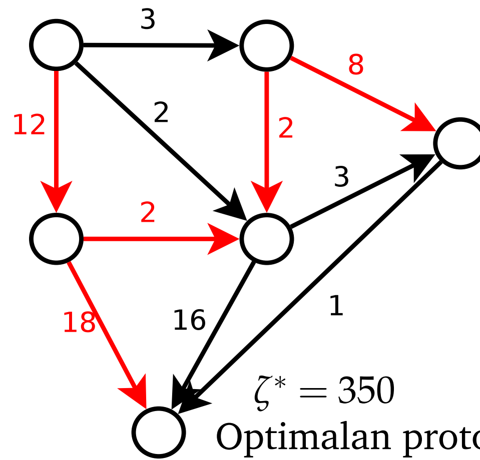


Vidimo da je balansirani protok dualno dopustiv ($z \geq 0$) i nije primarno dopustiv ($x_{\mathsf{cf}} = -8 < 0$).

Vršimo dualnu pivotizaciju na grani cf. Izbacujemo granu cf iz baze, skup čvorova se deli na dve grupe, na slici su odvojene granicom.

Vrednost $z_{\mathsf{cf}} = 0$ uvećavamo za t. Po pravilu mostova ažuriranje z vrednosti se vrši tako što ćemo vrednosti $z_{\mathsf{bc}} = 1$ i $z_{\mathsf{ec}} = 4$ umanjiti za t. Biramo $t = 1$, grana bc ulazi u bazu. Ako bi grana bc ušla u bazu pre nego što cf izađe iz baze, nastala bi kontura cfebc. Po pravilu kontura vrednosti transporta (x) po konturi se u jednom smeru povećavaju, u suprotnom smanjuju za vrednost A.

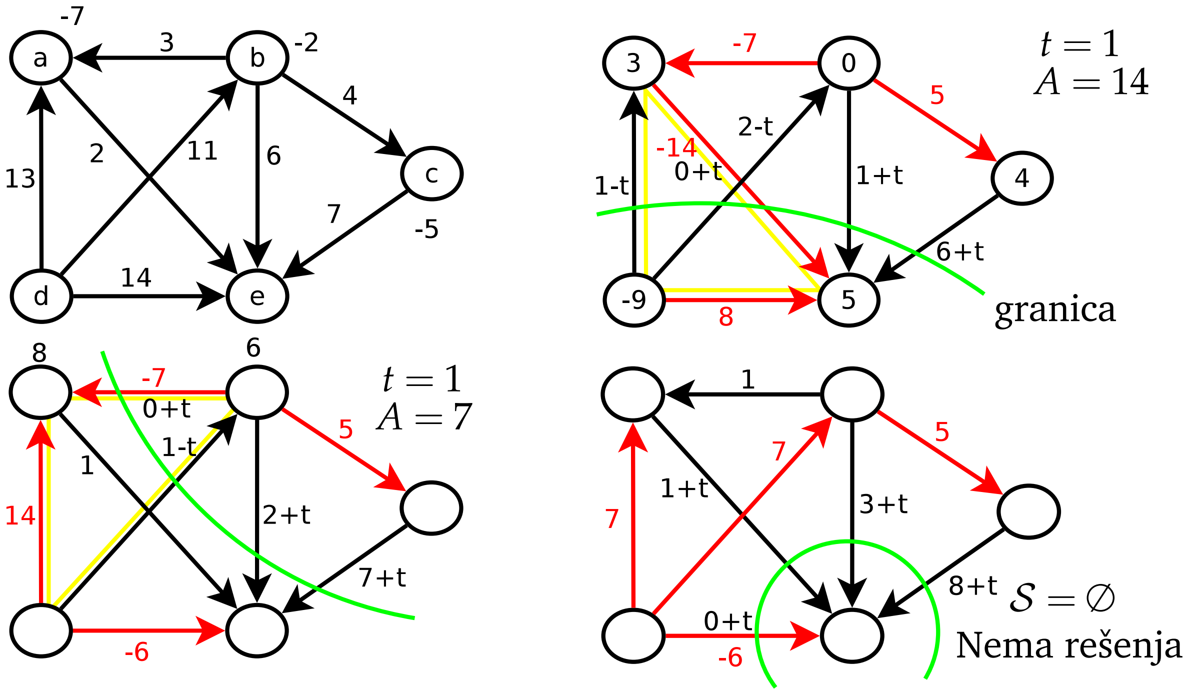
cf izlazi iz baze $\Rightarrow A = 8$.



$\zeta^* = 350$

Optimalan protok

Rešiti problem dat levo polazeći od ae, ba, bc, de.



Prethodni problem nema rešenja, odnosno: ne postoji dopustivi balansirani protok, odnosno: skup dopustivih tačaka problema linearnog programiranja je prazan ($\mathcal{S} = \emptyset$).

To se vidi zato što dualna pivotizacija koja bi izbacila granu de iz baze daje neograničen dual, pa je zbog slabe teoreme dualnosti primar nemoguć ($\mathcal{S} = \emptyset$).

Iako se to moglo primetiti na početku nalaženjem oblasti iz koje ne izlaze grane a ima viška zaliha, ne vrši se provera svih mogućih oblasti na početku, jer ih može biti veliki broj.

14 Parametarski self-dual simplex algoritam

Ako polazno drvo nije primarno dopustivo ni dualno dopustivo, ne znamo da postoji rešenje. Do rešenja ne možemo doći primenom primarnog ili dualnog simplex metoda.

Zbog toga se izračunate vrednosti x i z perturbuju parametrom μ.

$$x_B^* + \mu\bar{x}_B,\ z_N^* + \mu\bar{z}_N, \text{ gde je } \bar{x}_B > 0,\ \bar{z}_N > 0.$$

Za svaki protok nalazimo oblast vrednosti μ za koje je primarno i dualno dopustiv i pišemo pored grafa. Pivotizaciju vršimo na grani gde je realizovana najmanja vrednost oblasti dopustivih μ.

$$\mu^* = \min\{\mu : z_N^* + \mu\bar{z}_N \geq 0 \ \text{ i } \ x_B^* + \mu\bar{x}_B \geq 0\}.$$

Pivotizacija može biti primarna ili dualna, zavisi od odabrane grane.

Za izbor vrednosti t na grani koja menja mesto sa izabranom granom, poređenje po veličini se vrši sa $\mu := \mu^*$.

U pivotizacijama vodimo μ u nomenklaturi sa zarezom: $a + b\mu = a,b$.

Kad dobijemo da oblast dopustivih vrednosti za μ sadrži 0, dobili smo optimalni protok (uvrštavanjem $\mu = 0$).

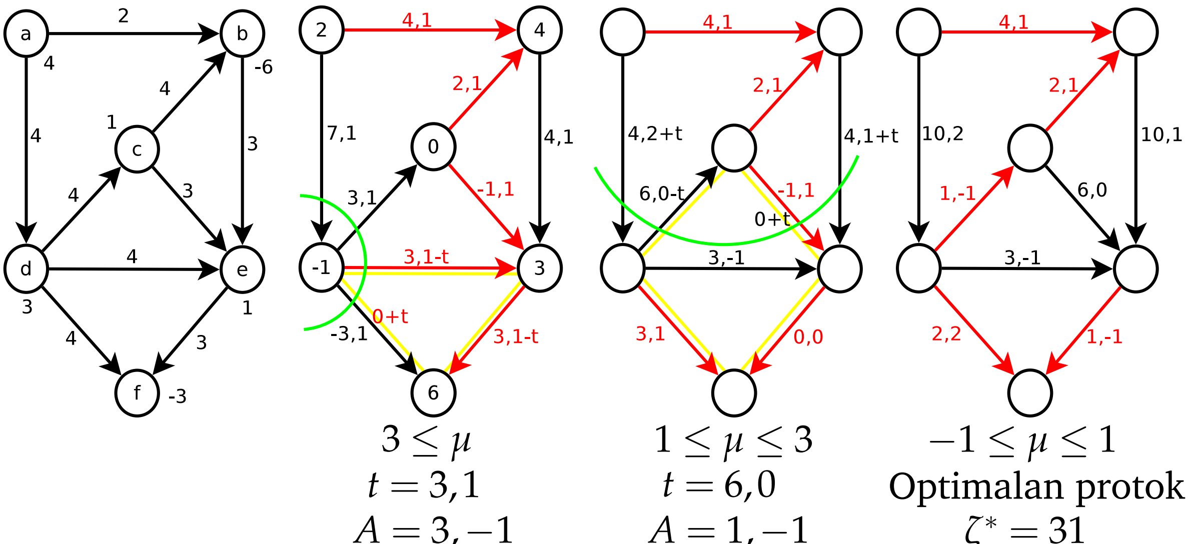
Ako uvrštavanje $\mu = 0$ daje primarno ili dualno dopustiv protok, možemo nastaviti sa Primarnim, odnosno Dualnim simplex algoritmom.

Moguće je da prelazak na dualni simplex algoritam daje protok za koji se vidi da ne postoji rešenje jer je $\mathcal{S} = \emptyset$.

Najčešće uzimamo $\bar{x}_B = [1, \ldots, 1]^T$, $\bar{z}_N = [1, \ldots, 1]^T$, odnosno, pored x i z vrednosti dopišemo ",1".

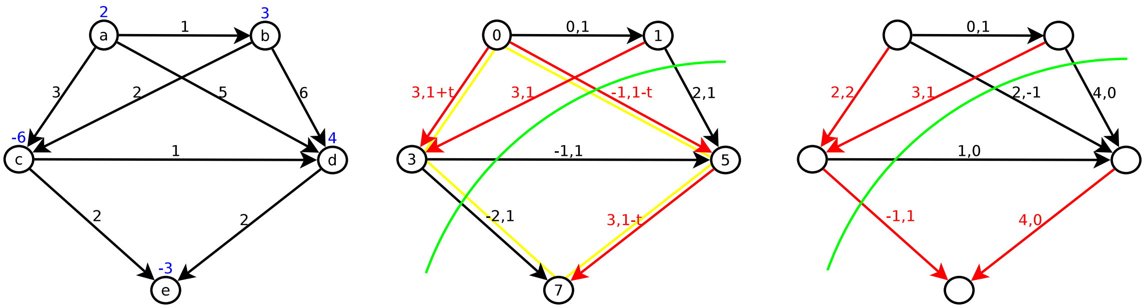
Može se dokazati da će se, bez degenerisanih pivotizacija, ako rešenje postoji, u konačnom broju koraka, doći do optimuma.

Rešiti problem dat levo polazeći od ab, cb, ce, de, ef.



Prvo je primenjena primarna pivotizacija na grani df, potom dualna pivotizacija na grani ce i dobijen je optimalan protok, $\mu = 0 \in [-1,1]$.

Rešiti problem dat levo polazeći od ac, ad, bc, de.



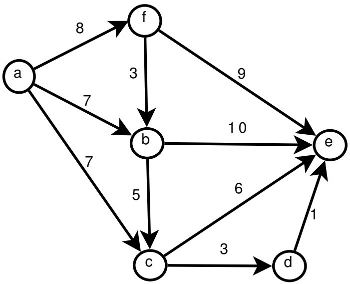
$$2 \leq \mu$$

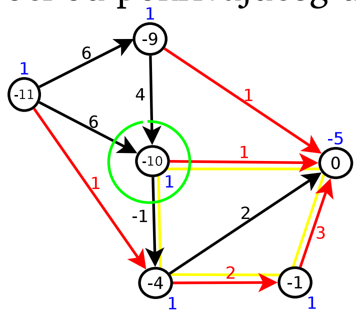
$$t = -1,1;\ A = 2,-1$$

Posmatramo poslednji protok. Uvrštavanje $\mu = 0$ daje dualno dopustivi protok. Nema grane koja ide preko granice u suprotnom smeru od grane c e, zaključujemo da problem nema rešenja, $\mathcal{S} = \emptyset$.

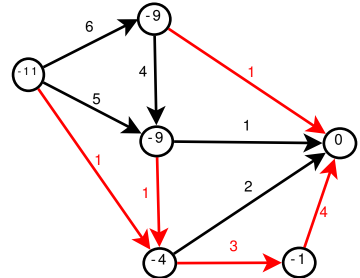
15 Najkraći put u mreži

Naći najkraće puteve od čvorova a, b, c, d, f do čvora e ako su na slici levo date udaljenosti. Poći od pokrivajućeg drveta ac, cd, de, be, fe.





$t = 1$
$A = 1$



a	b	c	d	e	f
11	9	4	1	0	9

Problem nalaženja najkraćeg puta do jednog čvora (e) svodimo na pronalaženje cene minimalnog transporta kroz mrežu, gde su cene transporta jednake udaljenostima i zaliha robe u svakom čvoru 1, osim u odabranom čvoru narudžbina je broj čvorova minus jedan.

16 Gornje ograničenje protoka

Pretpostavimo da je dat problem minimizacije cene mrežnog protoka i da postoji ograničenje koliko maksimalno komada robe nekom granom (i,j) može da se preveze, $x_{i,j} \le u_{i,j}$.

To ograničenje možemo dopuniti do jednakosti $x_{i,j} + t_{i,j} = u_{i,j}$, $t_{i,j} \ge 0$ i dodati na ostala ograničenje problema linearnog programiranja koji definiše problem minimizacije cene mrežnog protoka.

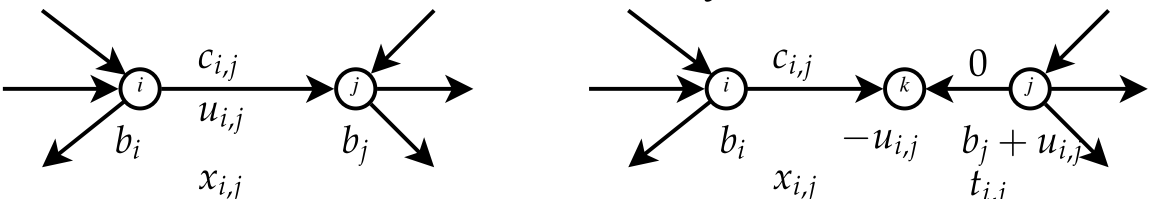
$$\begin{array}{ccccccc} \cdots & -x_{i,j} & \cdots & & = & -b_i \\ \cdots & x_{i,j} & \cdots & & = & -b_j \\ & x_{i,j} & \cdots & +t_{i,j} & = & u_{i,j}. \end{array}$$

Da bismo rekonstruisali oblik problema linearnog programiranja koji odgovara problemu minimizacije cene protoka kroz mrežu, pomnožićemo poslednju jednakost sa (-1) i dodati pretposlednjoj:

$$\begin{array}{cccccc} \cdots & -x_{i,j} & \cdots & & = & -b_i \\ \cdots & & \cdots & -t_{i,j} & = & -b_j - u_{i,j} \\ & x_{i,j} & \cdots & +t_{i,j} & = & u_{i,j}. \end{array}$$

Dobijeni sistem jednačina odgovara uslovima protoka koji ima jedan čvor više i jednu granu više. Neka balans ulaza i izlaza za novi čvor k bude srednja od tri prikazane jednakosti. Novoj grani odgovara količina transporta $t_{i,j}$, cena na njoj je 0.

Prikažimo staro (levo) i novo (desno) stanje slikama:

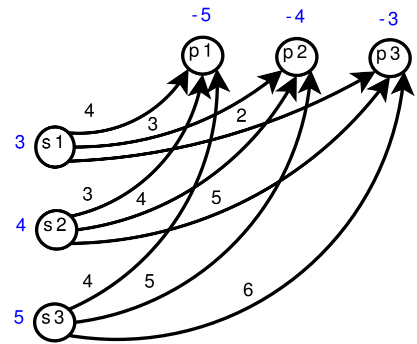


Sa desne slike se vidi gde čitamo vrednosti $x_{i,j}$ i $t_{i,j}$ u optimalnom rešenju problema sa ograničenjem protoka.

17 Hičkokov problem

Cene prevoza, potrebe tri prodavnice i zalihe u tri skladišta date su u tabeli. Naći optimalni plan transporta.

	p1	p2	p3	zal
s1	4	3	2	3
s2	3	4	5	4
s3	4	5	6	5
pot	5	4	3	



Graf u zadatku je kompletni bipartitni: čvorovi se mogu podeliti na one koji imaju zalihe, iz kojih polaze transporti i čvorove koji imaju potrebe, u njih ulaze transporti. Postoje grane od svakog skladišta do svake prodavnice. Ovo je **Hičkokov (Hitchcock) problem**.

Potreban uslov da problem minimizacije cene protoka ima rešenje je da su suma zaliha i suma potreba jednake.

Ako ima viška zaliha, možemo uvesti fiktivnog potrošača; ako ima viška potreba, možemo uvesti fiktivnog snabdevača. Cene ka njima i od njih se stavljaju da su 0.

Jedno primarno dopustivo rešenje Hičkokovog problema se može dobiti metodom severozapadnog ugla.

Posledica: **Dovoljan uslov da Hičkokov problem ima optimalno rešenje je da je zbir zaliha jednak zbiru potreba.**

17.1 Matematička formulacija Hičkokovog problema

Neka su zalihe u skladištima $a_1, a_2, \ldots, a_m$, neka su potrebe u prodavnicama $b_1, b_2, \ldots, b_n$ i neka su $c_{i,j}$ cene prevoza od skladišta i do prodavnice j, za $i = 1, 2, \ldots m$, $j = 1, 2, \ldots n$.

Neka je $x_{i,j}$ količina robe koja se prevozi od skladišta i do prodavnice j. Ukupna cena prevoza je ζ.

$$\zeta = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{i,j} x_{i,j} \to \min$$

$$\sum_{j=1}^{n} x_{i,j} = a_i,\ i \in \{1,2,\dots,m\},$$

$$\sum_{i=1}^{m} x_{i,j} = b_j,\ j \in \{1,2,\dots,n\},$$

$$0 \le x_{i,j} \in \mathbb{Z},\ i \in \{1,2,\dots,m\},\ j \in \{1,2,\dots,n\}.$$

Ovo je problem celobrojnog linearnog programiranja.

Rešavaćemo ga pomoću tabela transportnog problema.

17.2 Tabele transportnog problema

U polja tabele unosimo $x_{i,j}$ ili $z_{i,j}$, zavisno da li je grana od i-tog snabdevača do j-tog potrošača primarno bazična ili nebazična. Vrednosti $z_{i,j}$ upisujemo u zagrade.

U gornji levi ugao svakog polja upisujemo cenu $c_{i,j}$ transporta, a na marginama y vrednosti.

17.3 Transportni algoritam

1. $x :=$ `primarno dopustivo rešenje;`
2. `WHILE NOT test_optimalnosti(`x`)`
3. `    pivotizacija(`x`);`

17.4 Algoritam severozapadnog ugla

1. Poći od polja $(i, j) := (1, 1)$
2. $x_{i,j} := \min(b_i, -b_j)$; $b_i := b_i - x_{i,j}$; $b_j := b_j + x_{i,j}$;
3. IF $b_i = 0$ THEN pređi u polje $(i+1, j)$
   ELSE pređi u polje $(i, j+1)$;
4. Ponavljaj korake 2. do 3. i dođi do poslednjeg polja gde se moraju složiti preostala potreba i zaliha.

4 3	3	2	3
3 2	4 2	5	4
4	5 2	6 3	5
5	4	3	

17.5 Test optimalnosti

1. Biramo jedno y da je nula,
2. Računamo ostale y vrednosti tako da je za bazične promenljive $y_j - y_i = c_{i,j}$
3. Računamo z vrednosti za polja koja nisu bazična tako da je $y_j - y_i + z_{i,j} = c_{i,j}$
4. `IF` $\forall(i,j) z_{i,j} \geq 0$ `THEN` optimum `ELSE` negativan $z_{i,j}$ je pivot

^{4} 3	^{3} (-2)	^{2} $(-\mathbf{4})$	0
^{3} 2	^{4} 2	^{5} (0)	1
^{4} (0)	^{5} 2	^{6} 3	0
4	5	6	

17.6 Pivotizacija

1. Sastaviti cikl od pivota i polja baznih promenljivih.
2. Polazeći od pivota, idući po ciklu, naizmenično dodeliti znak $+$ i $-$ poljima cikla.
3. Od polja koja su dobila $-$ naći polje sa najmanjom vrednošću koja iznosi t.
4. Polazeći od pivota, idući po ciklu, naizmenično dodati i oduzeti vrednost t promenljivama $x_{i,j}$ u ciklu.
5. Jedno od polja koje je u prethodnom koraku dobilo vrednost 0 izbaciti iz baze, a pivota ubaciti u bazu.

Na našem primeru, pivotizacija daje

^{4} 1	^{3} (2)	^{2} 2	4
^{3} 4	^{4} (4)	^{5} (4)	5
$^{4}(-\mathbf{4})$	^{5} 4	^{6} 1	0
8	5	6	

Uzimamo za pivota $z_{3,1} = -4$, nova pivotizacija omogućava da iz baze izbacimo $x_{1,1}$ ili $x_{3,3}$. Biramo $x_{1,1}$.

^{4} (4)	^{3} (2)	^{2} 3	4
^{3} 4	^{4} (0)	^{5} (0)	1
^{4} 1	^{5} 4	^{6} 0	0
4	5	6	

Više nema negativnih z vrednosti, dobili smo dualno dopustivu, odnosno optimalnu, tabelu. Rešenje nije jedinstveno, ima nula z vrednosti. Najjeftiniji transport ima cenu
$\zeta = 2 \times 3 + 3 \times 4 + 4 \times 1 + 5 \times 4 + 6 \times 0 = 42.$

Zadatak

Cene prevoza, potrebe četiri prodavnice i zalihe u četiri skladišta date su u tabeli. Naći optimalni plan transporta.

	P_1	P_2	P_3	P_4	zal
S_1	4	5	6	3	7
S_2	5	4	6	6	8
S_3	7	8	9	6	10
S_4	10	12	15	8	15
pot	10	10	12	8	

17.7 Vogelova metoda

1. Za sve vrste i kolone izračunati apsolutnu vrednost razlike dve najmanje nedodeljene cene transporta.

2. U vrsti ili koloni koja daje najveću razliku otići u polje sa najmanjom nedodeljenom cenom transporta i dodeliti $x_{i,j} := \min(b_i, -b_j)$; $b_i := b_i - x_{i,j}; b_j := b_j + x_{i,j}$.

3. Ponavljaj korake 1. do 3. sve dok ne ostane samo jedna vrsta ili kolona.

4. Dodeliti nedodeljene zalihe odnosno potrebe preostalim (bazičnim) promenljivama.

5. Ako u bazi nema $n + m - 1$ elemenata, dodati potreban broj elemenata, ali tako da ne zatvaraju cikl sa postojećim bazičnim promenljivama.

Dobijamo

$^{4}\,(-1)$	$^{5}\,(-2)$	$^{6}\,(-\mathbf{4})$	$^{3}\,7$	5
$^{5}\,(3)$	$^{4}\,8$	$^{6}\,(-1)$	$^{6}\,(6)$	8
$^{7}\,(3)$	$^{8}\,(2)$	$^{9}\,10$	$^{6}\,(4)$	6
$^{10}\,10$	$^{12}\,2$	$^{15}\,2$	$^{8}\,1$	0
10	12	15	8	

$^{4}\,(-1)$	$^{5}\,(-\mathbf{2})$	$^{6}\,2$	$^{3}\,5$	5
$^{5}\,(3)$	$^{4}\,8$	$^{6}\,(3)$	$^{6}\,(6)$	8
$^{7}\,(-1)$	$^{8}\,(-2)$	$^{9}\,10$	$^{6}\,(0)$	2
$^{10}\,10$	$^{12}\,2$	$^{15}\,(4)$	$^{8}\,3$	0
10	12	11	8	

$^{4}\,(-1)$	$^{5}\,2$	$^{6}\,2$	$^{3}\,3$	0
$^{5}\,(1)$	$^{4}\,8$	$^{6}\,(1)$	$^{6}\,(4)$	1
$^{7}\,(-\mathbf{1})$	$^{8}\,(0)$	$^{9}\,10$	$^{6}\,(0)$	-3
$^{10}\,10$	$^{12}\,(2)$	$^{15}\,(4)$	$^{8}\,5$	-5
5	5	6	3	

$^{4}\,(0)$	$^{5}\,2$	$^{6}\,5$	$^{3}\,(1)$	0
$^{5}\,(2)$	$^{4}\,8$	$^{6}\,(1)$	$^{6}\,(5)$	1
$^{7}\,3$	$^{8}\,(0)$	$^{9}\,7$	$^{6}\,(1)$	-3
$^{10}\,7$	$^{12}\,(1)$	$^{15}\,(3)$	$^{8}\,8$	-6
4	5	6	2	

Optimalna tabela, $\zeta = 290$.

17.8 Problem angažovanja

Zadatak

Trener plivačke reprezentacije ima za štafetu 4X100m na raspolaganju četiri plivača čija su vremena na 100m po stilovima: slobodno, leđno, prsno, baterflaj:

	S	L	P	B
A	57	61	64	62
B	55	63	65	64
C	59	64	66	63
D	56	62	67	64

Kako da sastavi najbolju štafetu?

Rešenje

Ovo je transportni problem: plivači imaju na raspolaganju **jedno** plivanje koje treba da "prebace" na **jednu** stazu, cena transporta je jednaka vremenu koje plivaču treba da ispliva.

Svaku stazu može plivati jedan plivač, jedan plivač treba da pliva jednu stazu.

Ukupna cena transporta je jednaka ukupnom vremenu štafete.

Problem angažovanja je ekvivalentan Hičkokovom problemu sa $a_i = 1, i = 1, \ldots, m$, $b_j = 1, j = 1, \ldots, n$.

Vrednosti $x_{i,j}$ su 1 ako plivač i pliva stazu j, inače su 0.

Polazno rešenje možemo naći Vogelovom metodom:

57 (2)	61 0	64 1	62 0	0
55 1	63 (2)	65 (1)	64 (2)	0
59 (3)	64 (2)	66 (1)	63 1	-1
56 0	62 1	67 (2)	64 (1)	-1
55	61	64	62	

Vidimo da je transport optimalan, tako da plivači plivaju:

$A \to$ prsno, $B \to$ slobodno, $C \to$ baterflaj, $D \to$ leđno.

Ukupno vreme štafete je 244.

18 Matrične igre za dva igrača sa sumom nula

Zero-Sum Two-Person Games

Matrična igra definisana matricom $A = [a_{i,j}]_{m\times n}$ je:
(1) Igrač R bira vrstu i.
(2) Igrač C bira kolonu j.
(3) Igrač R dobija od igrača C (daje igraču C) $a_{i,j}$ dinara.

Igrači znaju sadržaj matrice A, treba istovremeno da se odluče za izbor vrste (igrač R) i kolone (igrač C).

Izbor određene vrste i kolone igrači će izvršiti nezavisno jedan od drugog u skladu sa unapred smišljenom strategijom.

Pošto je izbor mogućih vrsta i kolona konačan, za ovakvu igru kažemo da je **konačna**. Skup mogućih izbora igrača R označićemo X, skup mogućih izbora igrača C označićemo Y.

Igraču R cilj je da ostvari maksimalni dobitak, koji je gubitak ako je negativan broj, igraču C obrnuto.

18.1 Striktno određene matrične igre

Element $a_{i,j}$ matrice A je **sedlasta tačka** matrične igre A ako je najmanji u svojoj vrsti i najveći u svojoj koloni. Za matricu koja ima sedlastu tačku kažemo da je **striktno određena**.

Optimalne strategija matrične igre A sa sedlastom tačkom $a_{i,j}$ su: R bira i-tu vrstu, C bira j-tu kolonu, vrednost igre je $v = a_{i,j}$.

Nisu sve matrične igre striktno određene, stoga uvodimo mešovite strategije kao raspodelu verovatnoće izbora vrste (za igrača R) i kolone (za igrača C).

18.2 Mešovite strategije

Strategije igrača R i C su redom raspodele verovatnoće

$$\begin{pmatrix} 1 & \dots & m \\ p_1 & \dots & p_m \end{pmatrix}, \quad \begin{pmatrix} 1 & \dots & n \\ q_1 & \dots & q_n \end{pmatrix}.$$

Izbori su nezavisni i zato je verovatnoća izbora i-te vrste i j-te kolone (elementa $a_{i,j}$) jednaka $p_i q_j$.

Označimo skupove **svih strategija** igrača R i C redom:
$X^* = \{p = [p_1, p_2, \dots, p_m]^T | \sum_{i=1,\dots,m} p_i = 1, p_i \geq 0, i = 1, \dots, m\}$ i
$Y^* = \{q = [q_1, q_2, \dots, q_n]^T | \sum_{j=1,\dots,n} q_j = 1, q_j \geq 0, j = 1, \dots, n\}$.
Označimo skupove **čistih** strategija, onih koje imaju jednu jedinicu i sve ostalo nule, kao i do sad, sa X i Y redom za igrača R i C.

Očigledno $X \subset X^*$ i $Y \subset Y^*$.

Za igru definisanu matricom $A = [a_{i,j}]_{m \times n}$, za strategije $p \in X^*$ i $q \in Y^*$

igrača R i C, **očekivani dobitak** igrača R (gubitak igrača C) je

$$E(p,q,A)=\sum_{i=1}^{m}p_i\left(\sum_{j=1}^{n}a_{i,j}q_j\right)=\sum_{j=1}^{n}q_j\left(\sum_{i=1}^{m}p_i a_{i,j}\right)=p^TAq=q^TA^Tp.$$

Pretpostavimo da je igrač C odabrao strategiju $q\in Y^*$

Igrač R će onda odabrati jednu od strategija $\arg\max\limits_{p\in X^*} p^TAq$.

Važi da je $\max\limits_{p\in X^*}p^TAq=\max\limits_{p\in X}p^TAq=\max\limits_{1\le i\le m}\sum\limits_{j=1}^{n}a_{i,j}q_j=\max\limits_{1\le i\le m}(Aq)_i$,
gde je i oznaka za element u i-toj vrsti vektora Aq. Naime:

Poslednje dve jednakosti su očigledne. Pošto je $X\subset X^*$, između prva dva izraza važi $\geq$. Izraz p^TAq je neprekidna funkcija po p koja nad kompaktnim skupom X^* dostiže maksimum. Taj maksimum je prosek vrednosti vektora Aq, zato je manji ili jednak najvećoj vrednosti $(Aq)_i$, pa za prva dva izraza važi znak $\leq$, odatle prva jednakost.

Saznanjem koju strategiju je odabrao igrač C prednost dobija igrač R, zato će igrač C odabrati jednu od strategija q koja postiže

$$\min_{q \in Y^*} \max_{p \in X^*} p^T A\, q = \min_{q \in Y^*} \max_{1 \le i \le m} \sum_{j=1}^{n} a_{i,j}\, q_j =: v_2$$

Rečima: igrač C želi da minimizuje maksimalni mogući gubitak. Strategija q_0 koja to postiže postoji jer je u pitanju minimizacija neprekidne funkcije nad kompaktnim skupom. Nazivamo je **minimax** strategija igrača C.

Pretpostavimo da igrač R bira strategiju $p \in X^*$

Slično kao kad strategiju bira igrač C, igrač R će odabrati jednu od strategija p koja postiže

$$v_1 := \max_{p \in X^*} \min_{q \in Y^*} p^T A\, q = \max_{p \in X^*} \min_{1 \le j \le n} \sum_{i=1}^{m} p_i\, a_{i,j}.$$

Strategiju p koja to postiže nazivamo **minimax** strategija igrača R.

Rečima: igrač R želi da maksimizuje minimalni mogući dobitak. Strategija p koja to postiže postoji jer je u pitanju maksimizacija neprekidne funkcije nad kompaktnim skupom.

Za svaku realnu funkciju $f(x,y)$ i skupove X^* i Y^* važi

$$\max_{x\in X^*}\min_{y\in Y^*} f(x,y) \leq \min_{y\in Y^*}\max_{x\in X^*} f(x,y),$$

stoga važi $v_1 \leq v_2$.

Ako je $v_1 = v_2 =: v$, onda odgovarajuće optimalne strategije p_0 i q_0 igrača R i C i **vrednost igre** v nazivamo **rešenje igre**.

Minimax teorema (Von Neumann)

Za svaku matričnu igru postoji rešenje: optimalne strategije p_0 i q_0 i vrednost igre $v := v_1 = v_2 = E(p_0, q_0, A) = p_0^T A\, q_0$.

18.3 Rešavanje linearnim programiranjem (jedan način)

Neka je matrična igra data matricom $A_{m\times n} = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix}$.

Za odabranu strategiju $p = [p_1, \ldots, p_m]^T$ igrača R i odabranu strategiju $q = [q_1, \ldots, q_n]^T$ igrača C očekivani dobitak igrača R (gubitak igrača C) je $E(p,q,A) = p^T A q$.

Pretpostavimo da je $v_1 > 0$ i $v_2 > 0$. Naime: možemo rešavati igru $A^* = A + k$ jer za neko k:

$$E(p,q,A^*) = p^T(A+k)q = p^T A q + k\, p^T \mathbf{1}_{m\times n} q = E(p,q,A) + k,$$

gde je $\mathbf{1}_{m\times n}$ matrica jedinica. Ako izaberemo dovoljno veliko k da svi elementi igre A^* budu pozitivni, dobijamo da su optimalne vrednosti $v_1 > 0$ i $v_2 > 0$. Broj k biramo tako da ne dobijemo velike brojeve.

Posmatrano sa strane igrača R

$$\min_{1\le j\le n}\sum_{i=1}^{m} p_i a_{i,j} \to \max$$
$$p_1+\cdots+p_m=1$$
$$p_1\ge 0,\ldots,p_m\ge 0$$

Ovaj problem je ekvivalentan sa problemom linearnog programiranja

$$v_1\to\max$$
$$a_{1,1}p_1+\cdots+a_{m,1}p_m\ge v_1$$
$$\ddots$$
$$a_{1,n}p_1+\cdots+a_{m,n}p_m\ge v_1$$
$$p_1+\cdots+p_m=1$$
$$p_1\ge 0,\ \cdots,\ p_m\ge 0.$$

Posmatrano sa strane igrača C

$$\max_{1\le i\le m}\sum_{j=1}^{n} a_{i,j} q_j \to \min$$
$$q_1+\cdots+q_n=1$$
$$q_1\ge 0,\ldots,q_n\ge 0$$

Ovaj problem je ekvivalentan sa problemom linearnog programiranja

$$v_2\to\min$$
$$a_{1,1}q_1+\cdots+a_{1,n}q_n\le v_2$$
$$\ddots$$
$$a_{m,1}q_1+\cdots+a_{m,n}q_n\le v_2$$
$$q_1+\cdots+q_n=1$$
$$q_1\ge 0,\ \cdots,\ q_n\ge 0.$$

Smenimo $y_i = p_i/v_1$, $i = 1,\ldots,m$, $\quad x_j = q_j/v_2$, $j = 1,\ldots,n$.

Uslov $p_1 + \cdots + p_m = 1$ postaje $y_1 + \cdots + y_m = 1/v_1$,
uslov $q_1 + \cdots + q_n = 1$ postaje $x_1 + \cdots + x_n = 1/v_2$.

Onda je uslov $v_1 \to \max$ ekvivalentan uslovu $y_1 + \cdots + y_m \to \min$ i uslov $v_2 \to \min$ ekvivalentan uslovu $x_1 + \cdots + x_n \to \max$ i gornji problemi su ekvivalentni sa dva dualna problema lin. programiranja:

$$
\begin{array}{cc}
\xi = \frac{1}{v_1} = y_1 + \cdots + y_m \to \min & \zeta = \frac{1}{v_2} = x_1 + \cdots + x_n \to \max \\
a_{1,1}y_1 + \cdots + a_{m,1}y_m \geq 1 & a_{1,1}x_1 + \cdots + a_{1,n}x_n \leq 1 \\
\ddots & \ddots \\
a_{1,n}y_1 + \cdots + a_{m,n}y_m \geq 1 & a_{m,1}x_1 + \cdots + a_{m,n}x_n \leq 1 \\
y_1 \geq 0,\ldots,y_m \geq 0 & x_1 \geq 0,\ldots,x_n \geq 0.
\end{array}
$$

Dovoljno je da Simplex metodom rešimo desni problem (primar) i iz optimalne tabele (rečnika) očitamo rešenje levog (duala).

18.4 Algoritam

1. Formirati matricu $A^* := A + k$ koja ima sve pozitivne elemente.
2. Rešiti probleme linearnog programiranja $\zeta \to \max$ i $\xi \to \min$.
3. Označimo sa $P = [x_1, \ldots, x_n]^T$ rešenje primarnog problema, sa $Q = [y_1, \ldots, y_m]^T$ rešenje dualnog problema, a $z^* := \zeta^* = \xi^*$.
4. Rešenje igre je $p_0 = \frac{1}{z^*}Q$, $q_0 = \frac{1}{z^*}P$, $v = \frac{1}{z^*} - k$.

18.5 Dominantnost

Ako za vrste r_{i_1} i r_{i_2} matrice A važi $r_{i_1} \leq r_{i_2}$, kažemo da je r_{i_1} **recesivna vrsta**. Postoji optimalna strategija u kojoj je verovatnoća izbora te vrste jednaka 0, pa je možemo izbaciti iz daljeg razmatranja. Analogno, ako za dve kolone važi $c_{j_1} \leq c_{j_2}$, onda je c_{j_2} **recesivna kolona**.

18.6 Igre 2×2

Matrična igra $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ nije striktno određena ako

$$(a > b \text{ i } d > b \text{ i } a > c \text{ i } d > c) \text{ ili } (a < b \text{ i } d < b \text{ i } a < c \text{ i } d < c)\,.$$

Tada je njeno rešenje

$$p_0 = \frac{1}{I}\left[d - c, a - b\right]^T,$$

$$q_0 = \frac{1}{I}\left[d - b, a - c\right]^T,$$

$$v = \frac{\det A}{I},$$

gde je $I = a + d - b - c$.

Primeri zadataka

1. Dva igrača istovremeno pokazuju jedan, dva ili tri prsta. U slučaju da je zbir pokazanih brojeva paran, prvi igrač dobija zbir pokazanih brojeva dinara od drugog igrača, u protivnom drugi igrač dobija zbir pokazanih brojeva dinara od prvog igrača. Sastaviti i rešiti matricu koja odgovara ovoj matričnoj igri (naći optimalne strategije prvog i drugog igrača, kao i vrednost igre).

2. Dva deteta istovremeno pokazuju simbol za jedan od tri predmeta: **k**amen, **m**akaze, **h**artija. U slučaju da su pokazani isti simboli, nikom ništa. Kamen pobeđuje makaze, makaze pobeđuju hartiju, hartija pobeđuje kamen. Pobeđeni pobedniku daje jedan dinar.

   Rešiti datu matričnu igru.

3. Osoba A i osoba B ulažu u igru stavljajući u desnu šaku 4 ili 5 dinara. Istovremeno otvaraju šake i ako je ukupan broj novčića paran A uzima sve novčiće. Ako je neparan, B uzima novčiće.

   Rešiti matričnu igru.