Vaja 4 – Linearno programiranje, metoda Simplex

1. Splošna predstavitev problema

Linearno programiranje je matematična metoda, s pomočjo katere je mogoče poiskati način, kako doseči najboljši možni rezultat. V ekonomiji je to tipičen problem, kako doseči maksimalni dobiček ali minimalne stroške. Linearni program je tehnika za optimizacijo linearne ciljne oziroma kriterijske funkcije, glede na linearne enakosti in neenakosti. Linearni program lahko v kanonični obliki zapišemo kot:

Poišči maksimum za: $c^T x$
glede na omejitve: $Ax \leq b$.

Linearno programiranje služi predvsem za reševanje problemov v ekonomiji, ni pa zgolj omejeno na to področje in ga lahko najdemo tudi v tehniki. Sam problem reševanja sistemov linearnih neenačb datira v čas francoskega matematika Fouriera, medtem ko je sama tehnika linearnega programa veliko mlajša, saj je bila razvita med 2. svetovno vojno, uporabljena pa je bila za upravljanje izdatkov za vojsko, z namenom zmanjšanja stroškov in povečanja izgub pri sovražniku. Sama metoda Simplex pa je bila objavljena po koncu vojne, leta 1947.

Kot primer problema, ki ga lahko rešimo s pomočjo linearnega programiranja vzemimo problem kmeta, ki ima 70 hektarjev zemlje, ki jo mora posejati s koruzo, pšenico in sojo. Pri tem ima s setvijo koruze 60 evrov stroškov, s setvijo pšenice 80 evrov, s setvijo soje pa 120 evrov po hektaru. Za setev ima na voljo 6000 evrov. Pri spravilu potrebuje za koruzo za skladiščenje po hektaru 6 m^3, za pšenico 4 m^3 in za sojo 5 m^3, na voljo pa ima 330 m^3 veliko skladišče. Pri koruzi znaša dobiček po hektarju 80 evrov, pri pšenici 95 evrov in pri soji 110 evrov. Koliko hektarov mora kmet zasejati s katero od poljščin, da bo njegov dobiček največji? Iz danega primera lahko zapišemo kriterijsko funkcijo, za katero želimo poiskati maksimum. Ta je naslednja:

$$80x_1 + 95x_2 + 110x_3 \ .$$

Ob tem imamo naslednje omejitve:

$$x_1 + x_2 + x_3 \leq 70 \ ,$$
$$60x_1 + 80x_2 + 120x_3 \leq 6000 \ ,$$
$$6x_1 + 4x_2 + 5x_3 \leq 330 \ \text{ in}$$
$$x_1, x_2, x_3 \geq 0 \ .$$

Pri tem spremenljivke x_1, x_2, x_3 predstavljajo število hektarov posejanih s koruzo, s pšenico in s sojo. Prva omejitev izhaja iz velikosti kmetijskega zemljišča, ki ga ima kmet na razpolago, druga omejitev izhaja iz stroškov za setev in razpoložljivih sredstev s katerimi razpolaga, tretja omejitev izhaja iz skladiščnega prostora, s katerim razpolaga, zadnja omejitev pa izhaja iz same narave problema.
Da lahko poiščemo rešitev problema s pomočjo metode Simplex, je potrebno enačbe zapisati v posebni obliki, ki ji pravimo tudi ohlapna oblika. Do te oblike pridemo tako, da najprej v

neenačbah uvedemo nove spremenljivke, ki jih rečemo tudi ohlapne spremenljivke x_4, x_5, x_6. Tako dobimo:

$$80\,x_1 + 95\,x_2 + 110\,x_3\ .$$

Ob tem imamo naslednje omejitve:

$$x_4 = 70 - x_1 - x_2 - x_3\ ,$$
$$x_5 = 6000 - 60\,x_1 - 80\,x_2 - 120\,x_3\ ,$$
$$x_6 = 330 - 6\,x_1 - 4\,x_2 - 5\,x_3 \text{ in}$$
$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0\ .$$

Sedaj je potrebno označiti še vrednost kriterijske funkcije, kar storimo s črko z. Na ta način pridemo končno do ohlapnega formata, ki ima naslednjo obliko:

$$z = 80\,x_1 + 95\,x_2 + 110\,x_3\ ,$$
$$x_4 = 70 - x_1 - x_2 - x_3\ ,$$
$$x_5 = 6000 - 60\,x_1 - 80\,x_2 - 120\,x_3\ ,$$
$$x_6 = 330 - 6\,x_1 - 4\,x_2 - 5\,x_3\ .$$

Spremenljivke v sistemu enačb razdelimo v dve skupini, in sicer neosnovne spremenljivke, to so tiste spremenljivke, s katerimi smo sestavili osnovni sitem neenačb. Indekse teh spremenljivk označujemo z množico N. Za razliko od tega spremenljivke, ki smo jih dodali ob pretvorbi sistema neenačb v ohlapno obliko, imenujemo osnovne spremenljivke, njihove indekse pa shranjujemo v množico B. Moč množice N označujemo z n, moč množice B pa z m. V splošnem vrednosti m in n nista enaki. Sistem v ohlapni obliki lahko zapišemo v matriko **A** velikosti $(m+n)\times(m+n)$ ter dva vektorja **b** in **c**, oba velikosti $(m+n)\times 1$. Pri tem indeksi osnovnih spremenljivk določajo vrstico, indeksi neosnovnih spremenljivk pa stolpec v matriki **A**. Indeks osnovnih spremenljivk določa tudi indeks vrstice v vektorju **b**, indeksi neosnovnih spremenljivk pa indekse vrstic v vektorju **c**. Opisana matrika **A** ter vektorja **b** in **c** so tako naslednji:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 60 & 80 & 120 & 0 & 0 & 0 \\ 6 & 4 & 5 & 0 & 0 & 0 \end{bmatrix},\ b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 70 \\ 6000 \\ 330 \end{bmatrix},\ c = \begin{bmatrix} 80 \\ 95 \\ 110 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

ob tem sta množici N in B naslednji: $N = \{1,2,3\}$, $B = \{4,5,6\}$. Izhodiščna vrednost kriterijske funkcije $v=z$ je enaka 0. To vrednost izračunamo tako, da v kriterijsko funkcijo vstavimo izhodiščne vrednosti. Ker kmet v našem primeru ni posejal še ničesar, so vrednosti spremenljivk x_1, x_2 in x_3 enake 0. Če to vstavimo v kriterijsko funkcijo, dobimo:

$$z = 80\cdot 0 + 95\cdot 0 + 110\cdot 0\ .$$

Naša naloga je povečati vrednost kriterijske funkcije in sicer tako, da vzamemo eno izmed neosnovnih spremenljivk, x_e, za katero velja, da je vrednost $c_e > 0$. Ker so v našem primeru vse neosnovne spremenljivke takšne, se odločimo za x_1. To spremenljivko moramo sedaj izraziti z ustrezno osnovno spremenljivko in sicer takšno, da je pogoj neničelnosti zagotovo izpolnjen. To spremenljivko poiščemo tako, da v enačbe ohlapne oblike vse neosnovne spremenljivke, razen spremenljivke x_e, (v našem primeru je to x_1) zamenjamo z vrednostmi 0 in izračunamo rešitve enačb, ki jih tako dobimo. V našem primeru so to enačbe:

$$0=70-x_1 \qquad \text{;pri spremenljivki } x_4$$
$$0=6000-60\,x_1 \qquad \text{;pri spremenljivki } x_5$$
$$0=330-6\,x_1 \qquad \text{;pri spremenljivki } x_6$$

Iz treh enačb dobimo naslednje tri rešitve $x_1 = 70$, $x_1 = 100$ in $x_1 = 55$. Izberemo najmanjšo, to pa dobimo pri enačbi, ki določa spremenljivko x_6. To pa pomeni, da bomo spremenljivko x_1 izrazili s spremenljivko x_6 in to spremembo vstavili v vse enačbe ohlapnega formata. Tako dobimo naslednji sistem enačb.

$$z=41.67\,x_2+43.33\,x_3-13.33\,x_6\ ,$$

$$x_1=55-\frac{2}{3}x_2-\frac{5}{6}x_3-\frac{1}{6}x_6\ ,$$

$$x_4=15-\frac{1}{3}x_2-\frac{1}{6}x_3+\frac{1}{6}x_6\ ,$$

$$x_5=2700-40\,x_2-70\,x_3+10\,x_6\ .$$

S spremembo enačb v sistemu, se spremenijo tudi vrednosti v matriki **A** in vektorjih **b** in **c**. Te so naslednje:

$$A=\begin{bmatrix} 0 & \frac{2}{3} & \frac{5}{6} & 0 & 0 & \frac{1}{6} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & \frac{1}{6} & 0 & 0 & -\frac{1}{6} \\ 0 & 40 & 70 & 0 & 0 & -10 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},\ b=\begin{bmatrix} 55 \\ 0 \\ 0 \\ 15 \\ 2700 \\ 0 \end{bmatrix},\ c=\begin{bmatrix} 0 \\ 41.67 \\ 43.33 \\ 0 \\ 0 \\ -13.33 \end{bmatrix}.$$

Ker sta spremenljivki x_1 in x_6 zamenjali strani enačb, sta zamenjali tudi množici osnovnih in neosnovnih spremenljivk. Tako sta množici N in B sedaj naslednji: $N=\{2,3,6\}$, $B=\{1,4,5\}$. Preden nadaljujemo je potrebno izračunati še novo vrednost kriterijske funkcije. To lahko izračunamo preprosto tako, da k predhodni vrednosti kriterijske funkcije prištejemo doprinos spremenljivke x_1, to pa je produkt stare vrednosti c_1 in na novo izračunane vrednosti b_1. V našem primeru je to $80{\cdot}55$. Najmanjši dobiček, ki ga lahko kmet doseže tako znaša 4400 evrov.

Ker imamo v vektorju **c** še vedno nenegativne vrednosti je izračunani dobiček mogoče izboljšati in sicer vzamemo naslednjo neosnovno spremenljivko, za katero velja, da je ustrezna komponenta vektorja **c** večja od 0. Naslednja neosnovna spremenljivka, ki jo bomo zamenjali z eno od osnovnih bo tako vrednost x_2. Spet gremo iskat, katera od osnovnih spremenljivk bo za to ustrezna, kar lahko storimo na že zgoraj opisan način, ali pa smo nekoliko bolj direktni in preprosto izračunamo kvocient Δ_i, ki je definiran z naslednjo enačbo:

$$\Delta_i = \frac{b_i}{a_{ie}},$$

pri čemer velja, da $i \in B$, e pa je indeks izbrane neosnovne spremenljivke. V kolikor je a_{ei} negativen, dobi kvocient Δ_i vrednost ∞. Indeks i, pri katerem je kvocient Δ_i najmanjši, je indeks iskane osnovne spremenljivke.
V našem primeru je e enak 2, B pa vsebuje elemente {1, 4, 5}. Tako izračunamo tri kvociente in sicer:

$$\Delta_1 = \frac{b_1}{a_{12}} = \frac{55}{\frac{2}{3}} = 82.5$$

$$\Delta_4 = \frac{b_4}{a_{42}} = \frac{15}{\frac{1}{3}} = 45$$

$$\Delta_5 = \frac{b_5}{a_{52}} = \frac{2700}{40} = 67.5$$

Vidimo lahko, da je Δ_i najmanjši pri vrednosti indeksa i = 4. To pomeni, da bomo spremenljivko x_2 zamenjali s spremenljivko x_4. Pri tem dobimo:

$$A = \begin{bmatrix} 0 & 0 & \frac{1}{2} & -2 & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 3 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 50 & -120 & 0 & 10 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \; b = \begin{bmatrix} 25 \\ 45 \\ 0 \\ 0 \\ 900 \\ 0 \end{bmatrix}, \; c = \begin{bmatrix} 0 \\ 0 \\ 22.5 \\ -125 \\ 0 \\ 7.5 \end{bmatrix}.$$

Množici N in B sta naslednji: $N = \{3,4,6\}$, $B = \{1,2,5\}$. Trenutna vrednost kriterijske funkcije pa je naslednja: $v = 4400 + 41.67 \cdot 45 = 6275.15$.
Ker imamo v vektorju **c** še vedno nenegativne vrednosti, lahko kriterijsko funkcijo še izboljšamo. Tako je treba iz množice N izbrati tak indeks e, da bo veljalo, da je $c_e > 0$. Če izberemo za e vrednost 3, je pogoj izpolnjen, poiskati pa moramo ustrezno osnovno spremenljivko, s katero bomo nadomestili spremenljivko x_3. Tudi v tem primeru bomo izračunali ustrezne kvociente Δ_i, $i \in$ {1, 2, 5}, ki so v tem primeru naslednji:

$$\Delta_1 = \frac{b_1}{a_{13}} = \frac{25}{\frac{1}{2}} = 50$$

$$\Delta_2 = \frac{b_2}{a_{23}} = \frac{45}{\frac{1}{2}} = 90$$

$$\Delta_5 = \frac{b_5}{a_{53}} = \frac{900}{50} = 18$$

Vidimo, da je iskani indeks i enak 5, kar pomeni, da spremenljivko x_3 izrazimo s spremenljivko x_5. Pri tem se znova spremeni matrika **A** in vektorja **b** in **c**, ki so sedaj naslednji:

$$A = \begin{bmatrix} 0 & 0 & 0 & -0.8 & -0.01 & 0.40 \\ 0 & 0 & 0 & 4.2 & -0.01 & -0.6 \\ 0 & 0 & 0 & -2.4 & 0.02 & 0.2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \ b = \begin{bmatrix} 16 \\ 36 \\ 18 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \ c = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -71 \\ -0.45 \\ 3 \end{bmatrix}.$$

Pri tem sta množici N in B naslednji: $N = \{4,5,6\}$, $B = \{1,2,3\}$. Poveča se pa tudi vrednost kriterijske funkcije in sicer za vrednost 22.5·18 in sedaj znaša 6680.15, ker pa v vektorju **c** še vedno nimamo samih negativnih vrednosti, ali pa vrednosti 0, lahko kriterijsko funkcijo še izboljšamo. Edina možnost, ki nam v tem primeru ostane, je spremenljivka x_6, ki jo moramo nadomestiti z eno od osnovnih spremenljivk. V ta namen zopet izračunamo kvocient Δ_i, kjer $i \in \{1, 2, 3\}$. Tako dobimo:

$$\Delta_1 = \frac{b_1}{a_{16}} = \frac{16}{0.4} = 40$$

$\Delta_2 = \infty$, saj je $a_{26} < 0$ in

$$\Delta_3 = \frac{b_3}{a_{36}} = \frac{18}{0.2} = 90 .$$

Kvocient je minimalen pri vrednosti $i = 1$, kar pomeni, da je treba spremenljivko x_1 izraziti s spremenljivko x_6. Matrika **A** in vektorja **b** in **c** dobijo naslednjo obliko:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1.5 & 0 & 0 & 3 & -0.025 & 0 \\ -0.5 & 0 & 0 & -2 & 0.025 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 2.5 & 0 & 0 & -2 & -0.025 & 0 \end{bmatrix}, \ b = \begin{bmatrix} 0 \\ 60 \\ 10 \\ 0 \\ 0 \\ 40 \end{bmatrix}, \ c = \begin{bmatrix} -7.5 \\ 0 \\ 0 \\ -65 \\ -0.375 \\ 0 \end{bmatrix}.$$

.

Množici N in B sta sedaj enaki $N = \{1,4,5\}$ in $B = \{2,3,6\}$. Vrednost kriterijske funkcije se poveča za vrednost 3·40 na 6800.15.

Ker so sedaj vse vrednosti v vektorju **c** manjše ali enake 0, vrednosti kriterijske funkcije ne moremo več izboljšati in vektor **b** postane naš vektor rešitve. Odtod lahko razberemo, da bo kmet najuspešnejši tedaj, če bo svoje njive zasejal s 60 hektarji pšenice in 10 hektarji soje. Njegov dobiček bo znašal 6800 evra. Razlika, ki je nastala med našo kriterijsko funkcijo in dejanskim izračunom je posledica zaokroževanja števil na dve decimalki natančno, kar pa v primeru reševanja s pomočjo računalnika ne bo več problem.

2. Pomoč pri implementaciji

Opisani postopek lahko zapišemo v obliki psevdokoda, ki je prikazan v izpisu 1.

```
function SIMPLEX(A, b, c, x, z)
 begin
   if not INITIALIZE_SIMPLEX(A,b,c,N,B,v) then
     return ERROR(Negativna vrednost v vektorju b);

   while obstaja c_k > 0, k ∈ N do
     begin
       Izberi e ∈ N, tako da velja c_e > 0;
       for vsak j ∈ B do
         if a_je > 0 then Δ_j := b_j/a_je;
         else Δ_j := ∞;
       Izberi indeks l ∈ B od minimalne vrednosti v Δ

       if Δ_l = ∞ then
         return ERROR(Neomejen program);
       else
         PIVOT(A,b,c,N,B,v,l,e);
     end

   for i:=1 to |N|+|B| do
     if i ∈ B then
       x_i := b_i
     else
       x_i := 0;

   z := v;
end
```

Izpis 1: Reševanje linearnega programa z metodo Simplex

Kot vhod v funkcijo Simplex dobimo linearni program v ohlapni obliki, zapisan v matriki **A** ter vektorjih **b** in **c**. Funkcija kot izhod vrne vektor rešitev **x** in maksimalno vrednost kriterijske funkcije. Pri tem je treba poudariti, da so za uporabnika v rešitvi zanimive le tiste komponente, ki so nastopale v izhodiščnem sistemu pred njegovim zapisom v ohlapno obliko. Funkcija v izpisu 1 izbira med ustreznimi neosnovnimi spremenljivkami in nato poišče ustrezno osnovno spremenljivko, s katero bo izbrano neosnovno spremenljivko nato izrazila. Iz prej opisanega vemo, da se kot posledica spremenijo matrika **A**, vektorja **b** in c, množici N in B ter vrednost kriterijske funkcije. Vse to opravi funkcija `PIVOT`, katere psevdokod je prikazan v izpisu 2.

```
function PIVOT(A,b,c,N,B,v,l,e)
  begin
   n_be := bl/ale;

   for vsak j ∈ N - {e} do
     n_aej := alj/ale;
   n_ael := 1/ale;

   for vsak i ∈ B - {l} do
     begin
       n_bi := bi - aie*n_be;
       for vsak j ∈ N - {e} do
         n_aij := aij - aie*n_aej;
       n_ail := -aie*n_ael;
     end

   v := v + ce*n_be;
   for vsak j ∈ N - {e} do
     n_cj := cj - ce*n_aej;
   n_cl := -ce*n_ael

   N := N-{e} ∪ {l};
   B := B-{l} ∪ {e};
   A := n_A;
   b := n_b;
   c := n_c;
end
```

Izpis 2: Psevdokod funkcije PIVOT

Funkcija `PIVOT` ima kot vhodne parametre matriko **A**, vektorja **b** in **c**, množici N in B, indeksa neosnovne in osnovne spremenljivke ter trenutno vrednost kriterijske funkcije. V proceduri se vsi ti parametri, razen obeh indeksov, določijo na novo. Novo izračunane vrednosti matrike **A** shranjujemo v matriki **n_A**, enako pa je tudi z vektorjema **b** in **c**, kjer nove vrednosti shranjujemo v vektorja **n_b** in **n_c**. V izpisu 2 je izračun obeh vektorjev in matrike podan po komponentah. Pred izstopom iz procedure nato matriko **A** in oba vektorja prepišemo z na novo izračunanimi vrednostmi. Enako pa na novo določimo tudi indekse v množicah N in B, kjer se izmenjata indeksa *e* in *l*. Indeks *e* tako postane član množice B, brez indeksa *l* seveda, indeks *l* pa postane član množice N, ki je manjša za indeks *e*.

V funkciji `SIMPLEX` pa lahko zasledimo klic še ene funkcije, to je funkcije `INITIALIZE_SIMPLEX`. Ta klic funkcije v primeru našega linearnega programa ne bi imel nobenega vpliva, saj so vse vrednosti vektorja **b** pozitivne. Če temu ne bi bilo tako, bi bil lahko linearni program nerešljiv, obstajajo pa tudi primeri, kjer lahko program preoblikujemo iz nedopustne v dopustno obliko in rešitev tako tudi najdemo, vendar se pri naših vajah s tovrstnimi linearnimi programi ne bomo ukvarjali, saj so precej bolj redki od teh, ki smo jih uporabili kot naše učne primere. Psevdokod funkcije `INITIALIZE_SIMPLEX` bo tako nekoliko poenostavljen in ga lahko vidimo v izpisu 3.

```
function INITIALIZE_SIMPLEX(A,b,c,N,B,v)
 begin

   if min(b) ≥ 0 then
     begin
       N := {1,2,..., n};
       B := {n+1,n+2,... , n+m};
       v := 0;
       return true;
     end

   return false;
end
```

Izpis 3: Psevdokod funkcije `INITIALIZE_SIMPLEX`

Funkcija `INITIALIZE_SIMPLEX` nam poleg inicializacije preveri, ali je začetna rešitev dopustna ali pa ne. Čeprav obstajajo tudi rešljivi linearni programi z nedopustno začetno rešitvijo, pa bomo to možnost v naši vaji zanemarili.

3. Zahteve naloge

Izdelati je potrebno aplikacijo, ki bo reševala linearne programe s pomočjo metode Simplex. Linearni program v ohlapni obliki bo podan v tekstovni datoteki, v naslednji obliki:

n m

$\mathbf{A}$

b^T

c^T

Primer datoteke z opisom našega učnega linearnega programa je prikazan v izpisu 4.

```
3 3

0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
1 1 1 0 0 0
60 80 120 0 0 0
6 4 5 0 0 0

0 0 0 70 6000 330

80 95 110 0 0 0
```

Izpis 4: Primer datoteke z linearnim programom

Privzeto ime datoteke z linearnim programom naj bo *lprogram.txt*, aplikacija pa naj preko dialoga za nalaganje datotek omogoča nalaganje tudi ostalih datotek z linearnim programom. Program naj kot rezultat izpiše vektor rešitev v naslednji obliki:

x1: 0
x2: 60
x3: 10
x4: 0
x5: 0
x6: 40

z: 6800

Tudi v tem primeru je rezultat vezan na naš učni primer. V kolikor rešitev ne obstaja, mora program izpisati, ali gre v tem primeru za neomejen sistem, ali pa je zgolj začetna rešitev nedopustna.