SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO–MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK

Antonio Lukačević

LINEARNO PROGRAMIRANJE

Diplomski rad

Voditelj rada:
doc. dr. sc. Marko Erceg

Zagreb, studeni, 2019.

Ovaj diplomski rad obranjen je dana ____________________ pred ispitnim povjerenstvom u sastavu:

1. ____________________________, predsjednik
2. ____________________________, član
3. ____________________________, član

Povjerenstvo je rad ocijenilo ocjenom _______________.

Potpisi članova povjerenstva:

1. ____________________________
2. ____________________________
3. ____________________________

Ovaj diplomski rad posvećujem svojoj obitelji i zahvaljujem joj se na pruženoj potpori tijekom školovanja i života. Zahvaljujem se svom mentoru, doc. dr. sc. Marku Ercegu na razumijevanju i pomoći.

Sadržaj

Uvod

Linearno programiranje je relativno nova grana matematike koja se bavi optimizacijom problema uz zadana afina ograničenja. Njena primjena dolazi do izražaja kod rješavanja problema iz svakodnevnog života. Poslovne odluke, planiranja proizvodnje, problemi transporta i rasporedi zaposlenika, samo su neka od širokog područja primjene linearnog programiranja. Osnovna zadaća linearnog programiranja je određivanje optimalne vrijednosti (minimuma ili maksimuma) linearne funkcije cilja s obzirom na ograničenja zadana linearnim nejednadžbama. Povijesno gledano, do razvoja linearnog programiranja dolazi tijekom 30-tih godina 20. stoljeća. Ruski matematičar L. V. Kantorovič već je 1939. godine proučavao probleme proizvodnje i uočio da se ti problemi mogu matematički modelirati i riješiti numeričkim metodama. Međutim, njegov rad je ostao nezapažen. Frank Hitchcock je 1941. godine formulirao problem transporta, a George Stigler 1945. godine problem dijete. Kroz te i ostale probleme, a pogotovo probleme povezane s Drugim svjetskim ratom, razvila se potreba za stvaranjem metode rješavanja problema linearnog programiranja. George Dantzig je 1947. godine proučavao različite probleme programiranja i planiranja u američkoj vojsci, te je otkrio sustavnu metodu rješavanja problema linearnog programiranja, poznatu kao simpleks metoda. John von Neumann otkrio je važnost koncepta dualnosti, a Gale, Kuhn i Tucker su prvi objavili dokaz teorema dualnosti (više o povijesnom pregledu u [6]). Cilj ovog rada je prikazati osnovne koncepte i metode rješavanja zadaće linearnog programiranja. Pri pisanju rada uglavnom smo slijedili [2, 8. poglavlje].

U prvog poglavlju dana je geometrijska interpretacija linearnih nejednadžbi, te su iskazani temeljni pojmovi. Prikazan je jednostavan problem dijete u kojem je promatrana primarna i dualna zadaća problema linearnog programiranja. Na kraju poglavlja, na dva načina, riješen je primjer transporta. U drugom poglavlju iskazana je simpleks metoda, dan njezin algoritam i geometrijska intepretacija. Uvedena je simpleks tablica te je ilustrirano njeno korištenje na primjeru. U trećem poglavlju, ulazimo *dublje* u pojam dualne zadaće. Iskazan je i dokazan teorem dualnosti te ukratko teorija nejednakosti. Na kraju, u posljednjem poglavlju, riješena su tri primjera, od kojih jedan pomoću programa MS Excel.

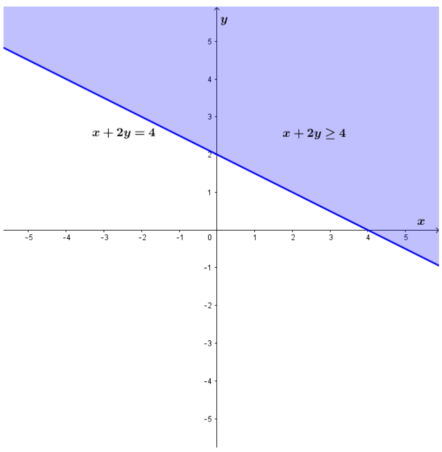
Poglavlje 1

Linearne nejednadžbe

1.1 Geometrijska intepretacija

Promatranje zadaće linearnog programiranja započnimo geometrijskom intepretacijom linearnih nejednadžbi. Linearna nejednadžba n dimenzionalnog prostora dijeli prostor na dva poluprostora. Jedan poluprostor zadovoljava danu linearnu nejednadžbu, a drugi ne zadovoljava. Promotrimo sljedeći primjer linearne nejednadžbe s dvije nepoznanice:

Primjer 1.1.1. $x + 2y \geq 4$

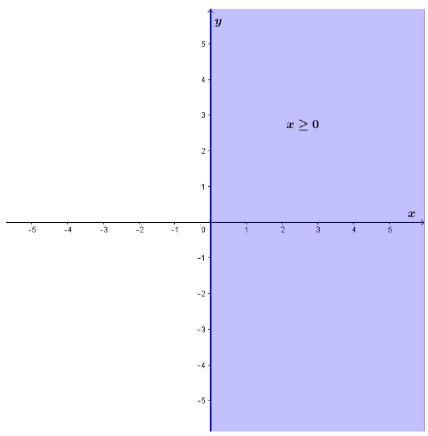


Slika 1.1: Skup točaka ravnine koje zadovoljavaju nejednadžbu $x + 2y \geq 4$

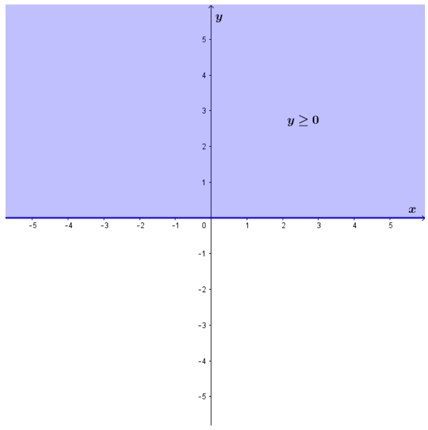
Kao što je prikazano na Slici 1.1, pravac $x + 2y = 4$ predstavlja **graničnu liniju** između dviju poluravnina. Ovisno o znaku nejednakosti određujemo pripadnost pravca poluravnini. Iscrtkanom linijom označujemo graničnu liniju koja ne pripada poluravnini, a punom označujemo graničnu liniju koja pripada poluravnini. U slučaju uspravnog pravca, granična linija dijeli ravninu na **lijevu poluravninu** i **desnu poluravninu**, a u slučaju kosog pravca, granična linija dijeli ravninu na **gornju poluravninu** i **donju poluravninu**.

Na sličan način možemo promatrati i linearne nejednadžbe zadane u tri dimenzije, odnosno u prostoru. U tom slučaju, umjesto graničnih linija promatramo granične ravnine. Primijetimo da je u n dimenzionalnom prostoru, dimenzija graničnog objekta $n - 1$.

Osnovno ograničenje zadaće linearnog programiranja je nenegativnost nepoznanica (u našem slučaju x i y). Nejednakosti $x \geq 0$ i $y \geq 0$ stvaraju dvije nove poluravnine, odnosno dva nova poluprostora. Ograničena su koordinatnim osima: $x \geq 0$ uključuje sve točke ravnine desno od pravca $x = 0$, dok $y \geq 0$ uključuje sve točke ravnine iznad pravca $y = 0$. Na Slici 1.2, u koordinatnom sustavu u ravnini, prikazana je nejednakost $x \geq 0$, a na Slici 1.3, u koordinatnom sustavu u ravnini, prikazana je nejednakost $y \geq 0$.



Slika 1.2: Skup točaka ravnine koje zadovoljavaju nejednadžbu $x \geq 0$.



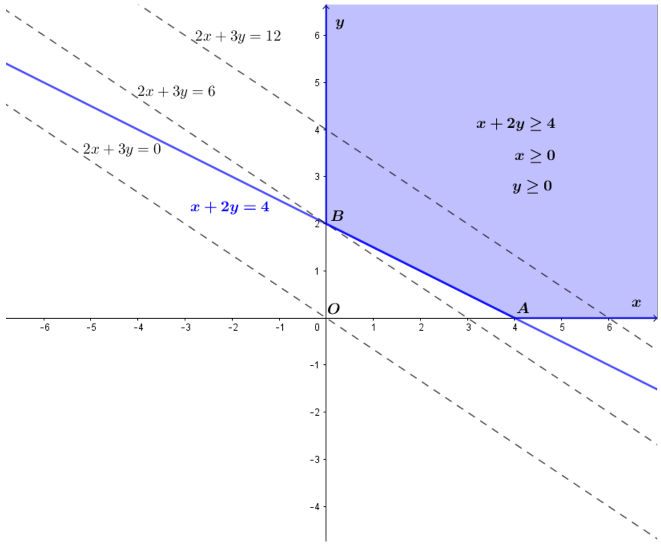
Slika 1.3: Skup točaka ravnine koje zadovoljavaju nejednadžbu $y \geq 0$.

1.2 Područje rješenja i funkcija cilja

Važan korak u rješavanju zadaće linearnog programiranja je određivanje skupa svih točaka ravnine (prostora) koje zadovoljavaju dana ograničenja (linearne nejednadžbe). Kao što je i prikazano na Slici 1.4, presjekom triju poluravnina $x + 2y \geq 4$, $x \geq 0$ i $y \geq 0$ dobivamo osjenčano područje koje nazivamo **područje rješenja** ili dopušteno područje. Područje rješenja sastavljeno je od skupa rješanja linearnih nejednadžbi oblika $Ax \geq b$ (presjek m poluravnina (poluprostora)). Budući da vrijednosti od x moraju biti nenegativna, dobivamo još n poluravnina (poluprostora). Povećavanjem broja ograničenja, područje rješenja se smanjuje.

Područje rješenja može biti ograničeno, neograničeno ili prazno. Na Slici 1.4 prikazano je neograničeno područje rješenja. Ukoliko bismo umjesto nejednakosti $x + 2y \geq 4$ promatrali nejednakost $x+2y \leq 4$ tada bismo dobili ograničeno područje rješenja određeno trokutom OAB. Kombinacijom nejednakosti $x+2y \geq 4$ i $x+2y \leq 4$ za područje rješenja dobili bismo pravac $x+2y = 4$. Postavljanjem kontradiktornog ograničenja popout $x+2y \leq -4$ ($x \geq 0$, $y \geq 0$) područje rješenja bi bilo prazno.

Linearne nejednadžbe i područja rješenja samo su dio u rješavanju zadaće linearnog programiranja. Još jedan od bitnih dijelova linearnog programiranja je i **funkcija cilja**. Odabirom različitih točaka iz područja rješenja maksimiziramo ili minimiziramo određenu funkciju cilja. Zadaća linearnog programiranja je pronaći točku područja rješenja koja maksimizira, odnosno mimimizira vrijednost funkcije cilja.



Slika 1.4: Područje rješenja i funkcija cilja

Na Slici 1.4 nalazi se područje rješenja zadano s $x + 2y \geq 4$ te dva nenegativna ograničenja $x \geq 0$ i $y \geq 0$. Neka je s $2x + 3y$ zadana funkcija cilja. Cilj je za danu funkciju cilja $2x + 3y$ pronaći točku područja rješenja koja će minimizirati njezinu vrijednost. Postavlja se sada pitanje kako među svima mogućim točkama područja rješenja pronaći onu koja će dati minimalnu vrijednost? Budući da uzastopna provjera svih točaka područja rješenja nije moguća (beskonačno mnogo točaka), nužno je pronaći drugi način određivanja tražene točke. Dodijelimo li funkciji cilja $2x + 3y$ određenu vrijednost i dobivenu jednadžbu nacrtamo u koordinatni sustav dobivamo **liniju konstantne vrijednosti**. Za liniju konstantne vrijednosti je svojstveno to da svaka točka te linije, koja se nalazi unutar područja rješenja, daje istu vrijednost. Za različite vrijednosti funkcije cilja dobivamo skup međusobno paralelnih linija konstantne vrijednosti (isti nagib). Minimalna vrijednost funkcije cilja nastaje u točki područja rješenja u kojoj je linija konstantne vrijednosti najbliža ishodištu. U našem slučaju to se događa u točki B gdje su $x^* = 0$ i $y^* = 2$. Dakle, minimalna vrijednost iznosi:

$$2x^* + 3y^* = 2 \cdot 0 + 3 \cdot 2 = 6 \, .$$

Točku (0,2) zovemo **optimalnim rješenjem** zadaće linearnog programiranja, jer ona minimizira funkciju cilja i nalazi se u području rješenja. Minimalna vrijednost predstavlja vrijednost programa. Optimalno rješenje označujemo zvijezdicom.

Općenito, prethodnu zadaću linearnog programiranja, zadanu funkcijom cilja i ograničenjima, nazivamo **primarnom zadaćom** te zapisujemo:

$$\text{Minimiziraj } cz \text{ s obzirom na } Az \geq b \text{ i } z \geq 0.$$

Varijable x i y su uključene u varijablu z primarne zadaće. Zadaću linearnog programiranja promatramo na nenegativnom dijelu $\mathbb{R}^n$. Uvjet $z \geq 0$ osigurava netrivijalnost zadaće linearnog programiranja. Vezano uz prethodni primjer, vrijedi: $A = \begin{bmatrix} 1 & 2 \end{bmatrix}$, $c = \begin{bmatrix} 2 & 3 \end{bmatrix}$, $b = 4$.

Definicija 1.2.1. *Uređaj na* $\mathbb{R}^n$*:*

$$x \geq y \quad \Longleftrightarrow \quad (\forall i \in \{1, ..., n\}, x_i \geq y_i) \ .$$

Optimalno rješenje, ako postoji, postiže su u vrhu područja rješenja. Geometrijski gledano, linije ili ravnine zadane funkcijom cilja postepeno rastu dok ne presijeku područje rješenja. Prvi dodir s područjem rješenja bit će upravo u jednom od njegovih vrhova i na taj naći postignut ćemo optimalno rješenje. Isti zaključak jednostavno možemo dobiti i analitičkim putem, na primjer, korištenjem diferencijalnog računa.

Simpleks metoda prolazi od jednog vrha područja rješenja do drugog vrha i pronalazi vrh koji ostvaruje najmanju vrijednost. S druge strane, metoda unutarnjih točaka, koju smo izostavili iz ovog rada, aproksimira optimalno rješenje biranjem točaka unutar područja rješenja.

Napomena 1.2.2. *Presjek funkcije cilja i područja rješenja ne mora biti samo jedna točka, odnosno samo jedan vrh. Na primjer, da smo u prethodnom primjeru za funkciju cilja uzeli* $x + 2y$*, onda bi svaka točka s ruba područja rješenja između vrhova A i B (vidi Sliku 1.4) bila optimalno rješenje. Minimalna vrijednost za sve točke tog ruba iznosila bi 4. Ako bi u ovom slučaju promatrali problem maksimizacije, onda naš problem ne bi imao rješenja budući da je zadano područje rješenja neograničeno.*

Svaka zadaća linearnog programiranja pripada jednoj od tri mogućnosti:

- Područje rješenja je prazno.
- Funkcija cilja je neograničena na području rješenja.
- Funkcija cilja poprima minimum (maksimum) na području rješenja.

U pravim zadaćama linearnog programiranja, iz područja ekonomije i poslovanja, rijetki su slučajevi u kojima su područja rješenja prazni ili neograničeni skupovi. Ukoliko se takvi skupovi i dogode, zadaće traba dodatno razmotriti i ispitati jesu li dobro postavljene.

Napomena 1.2.3. *Kod problema maksimizacije, zamjenom $c \leftrightarrow -c$ se svodimo na problem minimizacije. Zato bez smanjenja općenitosti promatramo samo probleme minimizacije primarne zadaće.*

1.3 Poravnavajuće varijable

Postoji jednostavan način pretvorbe zadane linearne nejednadžbe u linearnu jednadžbu. Uvođenjem nepoznanice kao **poravnavajuće varijable** poravnava se razlika između lijeve i desne strane linearne nejednadžbe i dobiva se linearna jednadžba. Na primjer, za linearnu nejednadžbu $x + 2y \geq 4$ linearna jednadžba glasi $w = x + 2y - 4$, pri čemu je w poravnavajuća varijabla. Ograničenje zadano nejednadžbom $x + 2y \geq 4$ pretvorili smo u ograničenje $w \geq 0$, u skladu sa zahtjevima linearnog programiranja. Na taj način dobili smo linearnu jednadžbu i nenegativna ograničenja na x, y i w. Poravnavajuća varijabla w sada je uključena u varijablu z primarne zadaće:

$$\text{Minimiziraj } \tilde{c}z \text{ s obzirom na } \tilde{A}z = b \text{ i } z \geq 0.$$

Vektor redak $\tilde{c} = \begin{bmatrix} 2 & 3 & 0 \end{bmatrix}$ zadan je funkcijom cilja $2x + 3y$. Matrica $\tilde{A} = \begin{bmatrix} 1 & 2 & -1 \end{bmatrix}$ zadana je ograničenjem $x + 2y - w = 4$.

1.4 Problem dijete

Problem dijete jedan je od klasičnih primjera zadaće linearnog programiranja. Problem je usmjeren na ostvarivanje minimalnog troška dijete uz uvjet zadovoljavanja nutritivnih potreba osobe. U ovom slučaju promatrat ćemo dijetu u kojoj želimo, uz minimalni trošak, zadovoljiti dnevne potrebe proteina. Na raspolaganju imamo dva izvora proteina: maslac od kikirikija i odrezak. Maslac od kikirikija daje nam jedinicu proteina, a odrezak dvije jedinice proteina. Kako bi zadovoljile potrebe dijete, osobe moraju unijeti najmanje četiri jedinice proteina. Uvedimo sljedeće **varijable odluke**:

- x = broj maslaca od kikirikija
- y = broj odrezaka

Iz dnevne potrebe za proteinima, dobivamo ograničenja zadaće linearnog programiranja: $x + 2y \geq 4$, $x \geq 0$ i $y \geq 0$. Ograničenje zadano linearnom nejednadžbom $x + 2y \geq 4$ nazivamo **problemskim ograničenjem**. Posljednja dva ograničenja nazivamo **nenegativnim ograničenjima** jer nije moguće imati negativan broj maslaca od kikirikija i odrezaka. Kako bismo izgradili matematički model zadaće linearnog programiranja, nedostaje nam oblikovati funkciju cilja. Pretpostavimo da je vrijednost maslaca od kikirikija \$2, a vrijednost odreska \$3, tada je cijena dijete zadana sljedećom funkcijom cilja $2x + 3y$. Kao što smo pokazali i ranije, optimalno rješenje dane zadaće linearnog programiranja nalazi se u točki $(0, 2)$ područja rješenja. Dakle, optimalna dijeta sadrži dva odrezaka i niti jedan maslac od kikirikija ($x^* = 0$ i $y^* = 2$), te trošak iznosi \$6. Ovaj problem dijete riješili smo geometrijskim pristupom zadaće linearnog programiranja, kako je i prikazano na Slici 1.4.

Svaka zadaća minimizacije povezana je sa zadaćom maksimizacije, odnosno sa svojom **dualnom zadaćom**. Ako je izvorna zadaća zadaća minimizacije, onda je njoj dualna zadaća zadaća maksimizacije. Dakle, minimun u *primarnoj zadaći* jednak je maksimumu u njoj *dualnoj zadaći*. To predstavlja ključ linearnog programiranja. Kako bismo intepretirali problem dijete promatrajući njegovu dualnu zadaću?

Budući da smo primarnu zadaću problema dijete promatrali sa strane kupca, njegovu dualnu zadaću promatrat ćemo sa strane prodavača, na primjer, ljekarnika. Pretpostavimo da ljekarnik prodaje proteinske tablete po cijeni p. Ljekarnik hoće maksimizirati dobit. Cijena sintetičkog proteina ne smije biti veća od cijene proteina koja se nalazi u maslacu od kikirikija (\$2 jedinica proteina) i odresku (\$3 dvije jedinice proteina). Također, cijena proteinskih tableta ne smije biti negativna. Iz uvjeta da osoba mora unijeti 4 jedinice proteina, dobivamo da pripadna dualna zadaća glasi:

$$\text{Maksimiziraj } 4p \text{ s obzirom na } p \leq 2,\ 2p \leq 3 \text{ i } p \geq 0.$$

U ovom primjeru lakše je riješiti dualnu zadaću u odnosu na njenu primarnu. U dualnoj zadaći imamo samo jednu varijablu odluke (cijenu p proteinskih tableta). Iz problemskog ograničenja $2p \leq 3$ i uvjeta da želimo maksimizirati dobit, slijedi da je maksimalna cijena proteinske tablete $p = \$1.50$, pa maksimalna dobit iznosi $4p = 4 \cdot \$1.50 = \$6$. Dakle, zaključujemo da su zadaće minimizacije i maksimizacije dale jednake novčane vrijednosti, odnosno da ćemo platiti isti iznos za proteine iz namirnica te sintetičke proteine. U trećem poglavlju, sistematično će se objasniti kako iz primarne zadaće dobiti njenu dualnu zadaću.

1.5 Primjena

U problemu transporta koristimo geometrijski pristup rješavanja zadaće linearnog programiranja. Tekst primjera preuzet iz [3].

Primjer 1.5.1. ***Problem transporta****. Profesori završnog razreda srednje škole za potrebe maturalnog putovanja planiraju unajmiti velike i male autobuse. Svaki veliki autobus može prevoziti 40 učenika, mora imati 3 pratitelja i njegov najam košta* \$1200*. Svaki mali autobus može prevoziti 8 učenika, mora imati jednog pratitelja i njegov najam košta* \$100*. Za* *potrebe putovanja nužno je osigurati mjesta za najmanje 400 učenika. Kako se za pratnju učenika prijavilo samo 36 roditelja, profesori moraju planirati najviše 36 pratitelja. Koliko vozila svakog tipa škola mora unajmiti kako bi minimizirala troškove transporta? Koliki su minimalni troškovi transporta?*

Rješenje. Rješavanje zadaće linearnog programiranja započinjemo analiziranjem pitanja postavljenog u iskazu problema. Iz njega saznajemo da je cilj profesora minimizirati troškove transporta. Budući da su troškovi najma većeg i manjeg autobusa različiti, profesori moraju odlučiti koliko će većih, odnosno manjih, autobusa unajmiti. Stoga uvodimo sljedeće varijable odluke:

- x = broj unajmljenih većih autobusa
- y = broj unajmljenih manjih autobusa

U Tablici 1.1 prikazani su zahtjevi, ciljevi i ograničenja zadaće linearnog programiranja. Varijable odluke pridružene su stupcima tablice. Koristeći varijable odluke i informacije iz Tablice 1.1 možemo oblikovati funkciju cilja:

$$T = 1200x + 100y$$

Cilj je pronaći vrijednosti varijabli odluke koje će dati optimalnu (minimalnu) vrijednost funkcije cilja. Nadalje, utvrdimo problemska ograničenja. Iz teksta problema i Tablice 1.1 možemo iščitati dva važna ograničenja. Ograničenje vezano uz broj učenika i ograničenje vezano uz broj pratitelja:

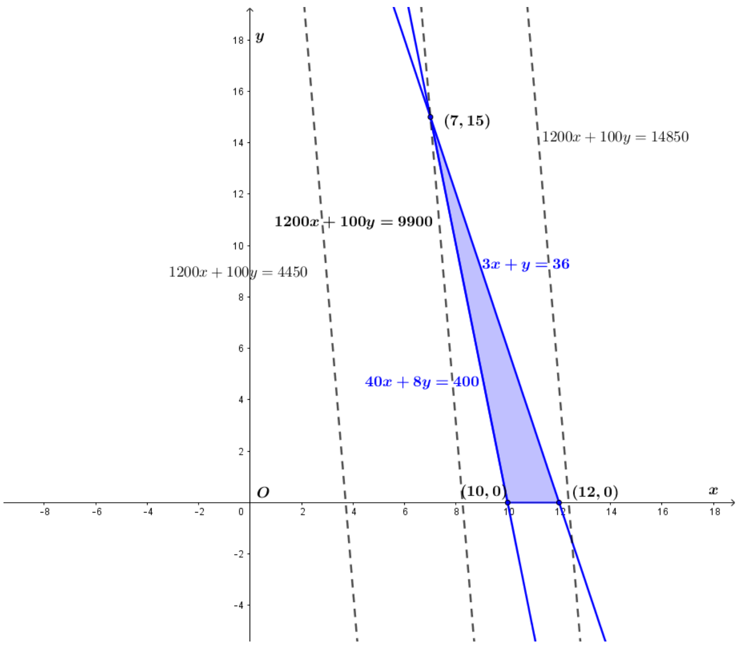
$$\begin{aligned} 40x + 8y &\geq 400 \\ 3x + y &\leq 36 \end{aligned}$$

	Veliki autobus	Mali autobus	Ograničenje
Broj učenika	40	8	≥ 400
Broj pratitelja	3	1	≤ 36
Trošak unajmljivanja	\$1200	\$100	

Tablica 1.1: Tablica zadaće linearnog programiranja.

Budući da ne možemo unajmiti negativan broj malih i velikih autobusa, uvodimo nenegativna ograničenja: $x \geq 0$ i $y \geq 0$ te za promatrani problem transporta, dobivamo sljedeći matematički model:

$$\begin{gathered}\text{Minimizirati } T = 1200x + 100y \text{ s obzirom na} \\ 40x + 8y \geq 400\ , \\ 3x + y \leq 36\ , \\ x,\ y \geq 0\ .\end{gathered}$$



Slika 1.5: Područje rješenja i funkcija cilja

Grafičkim rješavanjem sustava linearnih nejednadžbi, zadanih ograničenjima, dobivamo područje rješenja prikazano na Slici 1.5. Uočimo da je ovdje područje rješenja omeđeno i zatvoreno, dakle kompaktno, pa optimalno rješenje sigurno postoji (neprekidna funkcija postiže minimum na kompaktnom skupu). Ranije smo spomenuli da se optimalno rješenje zadaće linearnog programiranja može odrediti na dva načina: korištenjem linije konstantne vrijednosti (u terminima primjera: linije konstantnog troška) te provjeravanjem vrijednosti funkcije cilja u vrhovima područja rješenja.

I. način

Linija konstantnog troška, za bilo koji iznos troška T, ima nagib $-\frac{1}{12}$. S porastom troška T linija se odmiče od ishodišta. Najmanji trošak nastupit će u vrhu područja rješenja u kojem je linija konstantnog troška najbliža ishodištu koordinatnog sustava. U našem slučaju se to događa u točki (7, 15), kao što je i vidljivo sa Slike 1.5.

II. način
Odredimo vrijednosti funkcije cilja u svakom vrhu područja rješenja. Koordinate vrhova područja rješenja dobit ćemo rješavajući sustave dviju linearnih jednadžbi s dvije nepoznanice, zadanih s:

$$40x + 8y = 400, \; 3x + y = 36 \text{ i } y = 0.$$

U Tablici 1.2 nabrojani su vrhovi područja rješenja i vrijednosti funkcije cilja. Ispitujući vrijednosti u tablici, vidimo da je minimalni trošak unajmljivanja \$9900 i da se postiže u točki $(7, 15)$.

Vrh (x, y)	Vrijednost funkcije cilja (T)
$(10, 0)$	\$12000
$(12, 0)$	\$14400
$(7, 15)$	\$9900

Tablica 1.2: Tablica vrijednosti funkcije cilja u vrhovima područja rješenja.

Vidimo da oba načina rješavanja daju isto rješenje zadaće linearnog programiranja. Preostaje nam intepretirati rješenje u okviru zadanog problema. Dakle, minimalni trošak unajmljivanja profesori će postići ako unajme 7 velikih i 15 malih autobusa. Vrijednost minimalnog troška, gdje su $x^* = 7$ i $y^* = 15$ iznosit će:

$$1200 \cdot x^* + 100 \cdot y^* = 1200 \cdot 7 + 100 \cdot 15 = \$9900 ,$$

te bi se svi roditelji iskoristili za pratnju $(3x^* + y^* = 36)$.

Poglavlje 2

Simpleks metoda

Geometrijska metoda rješavanja zadaće linearnog programiranja omogućava nam dobar uvid u područje linearnog programiranja. Međutim, korisna je samo za rješavanje zadaća u kojima imamo dvije varijable odluke i mali broj problemskih ograničenja. Time se prirodno postavlja sljedeće pitanje: kako riješiti zadaću linearnog programiranja s više varijabla odluke i mnogo problemskih ograničenja?

U ovom poglavlju promatrat ćemo zadaće linearnog programiranja zadanih s n varijabli odluke i m problemskih ograničenja. Najbolji pristup rješavanja je zapisati zadaću u matričnom obliku. Neka su dani:

- realna matrica A dimenzija $m \times n$
- realan vektor stupac b dimenzije m
- realan vektor redak c (vektor cilja) dimenzije n

Sada zadaću minimizacije zapisujemo u sljedećem obliku:

$$\text{Minimiziraj } cx \text{ s obzirom na } Ax \geq b \text{ i } x \geq 0.$$

Točke područja rješenja moraju zadovoljavati nejednakost $x \geq 0$. Uvrstimo li vektor $x = (x_1, x_2, ..., x_n)$ u danu funkciju cilja, dobit ćemo njezinu vrijednost, odnosno $cx = c_1x_1 + c_2x_2 + ... + c_nx_n$. Područje rješenja obuhvaća $m + n$ nejednakosti, odnosno točka x nalazi se u presjeku $m + n$ poluprostora. Ranije smo spomenuli da područje rješenja ima ravne rubove te da može biti ograničeno, neograničeno ili prazno.

Cilj zadaće linearnog programiranja je odrediti točku x područja rješenja koja daje optimalno rješenje, odnosno točku minimuma (maksimuma) funkcije cilja na području rješenja. U prethodnom poglavlju, točku x^* određivali smo na dva načina: linijom konstantne vrijednosti i određivanjem vrijednosti funkcije cilja u vrhovima područja rješenja. S porastom broja varijabli i ograničenja, broj vrhova postaje jako velik, pa nije praktično određivati sve vrhove područja rješenja kako bismo pronašli optimalno rješenje. Iz tog razloga koristimo **simpleks metodu**.

2.1 Geometrijska intepretacija simpleks metode

Promatranje geometrijske intepretacije simpleks metode možemo podijeliti u dvije faze. U prvoj fazi, simpleks metoda odabire jedan vrh područja rješenja. Temeljna ideja metode je prolazak od jednog vrha do drugog rubovima zadanog područja rješenja. Iz odabranog vrha moguće je izabrati n rubova područja rješenja. Neki rubovi mogu nas udaljiti od vrha s optimalnim rješenjem, a neki nas postepeno mogu voditi prema njemu. Simpleks metoda uvijek vodi prema vrhu s optimalnim rješenjem, odnosno prema vrhu u kojem zadana funkcija cilja poprima minimalnu vrijednost. Metoda završava kada odredimo vrh iz kojeg rubovi područja rješenja vode prema vrhovima u kojima funkcija cilja poprima veće vrijednosti od trenutne. Taj vrh predstavlja optimalno rješenje zadaće linearnog programiranja x^*.

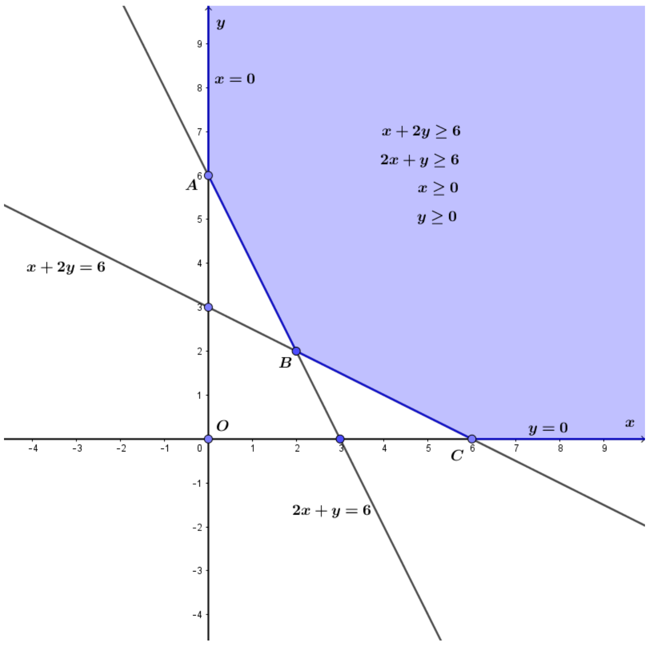
Geometrijski intepretiranu simpleks metodu promotrimo sa strane linearne algebre. U vrhu područja rješenja sijeku se n različitih ravnina, zadanih s n jednadžbi. Na primjer, u dvodimenzionalnom prostoru presjek dva pravca jedinstveno određuju točku, dok su u trodimenzionalnom prostoru potrebne tri ravnine. Za određivanje vrha područja rješenja, moramo odabrati n od $n + m$ nejednakosti $Ax \geq b$ i $x \geq 0$, pretvorit ih u jednakosti te pronaći njihovo sjecište. Naravno, točka sjecišta bit će u točki koja zadovoljava preostalih m ograničenja. U suprotnom, točka se neće nalaziti u području rješenja.

Na Slici 2.1 prikazano je područje rješenja zadano s dvije varijable odluke ($n = 2$) i dva problemska ograničenja ($m = 2$). Presjekom poluravnina dobivamo šest točaka presjeka. Točke A, B i C su vrhovi područja rješenja. Koordinate točaka su $(0, 6)$, $(2, 2)$ i $(6, 0)$. Jedna točaka vrha područja rješenja dat će optimalno rješenje zadaće linearnog programiranja. Preostale tri točke, uključujući i ishodište koordinatnog sustava, su *lažne* točke, jer njihove koordinate ne zadovoljavaju sva ograničenja. Općenito, postoji najviše

$$\binom{m+n}{n} = \frac{(m+n)!}{n! \cdot (m+n-n)!} = \frac{(m+n)!}{n! \cdot m!}$$

mogućih točaka presjeka. Iskazani binomni koeficijent predstavlja broj načina odabira n jednadžbi ravnina od ukupnih $m+n$. Za velike vrijednosti m i n, određivanje ukupnog broja vrhova je nepraktično i gotovo nemoguće.

Zaključimo, zadatak prve faze je pronaći odgovarajući vrh područja rješenja ili utvrditi da je područje rješenja prazan skup. Nastavimo s pretpostavkom da je pronađen odgovarajući vrh područja rješenja.



Slika 2.1: Vrhovi i rubovi područja rješenja

Pretpostavimo da je uklonjena jedna od n presječnih ravnina. Točke koje zadovoljavaju preostalih $n-1$ jednadžbi čine rub koji izlazi iz odgovarajućeg vrha područja rješenja. Taj rub je presjek $n-1$ ravnina. Kako bismo ostali unutar područja rješenja, dopušten je samo jedan smjer duž svakog ruba. Dolazimo do druge faze. U drugoj fazi moramo odlučiti kojim od n različitih rubova područja rješenja krenuti.

Kako bismo opisali drugu fazu, preoblikujmo ograničenja $Ax \geq b$ koristeći poravnavajuće varijable kao što je opisano u potpoglavlju 1.3. Naime, za $w = Ax - b$, ograničenje zadano s $Ax \geq b$ preoblikujemo u ograničenja $w_1 \geq 0$, $w_2 \geq 0$, ..., $w_m \geq 0$, s jednom poravnavajućom varijablom za svaki redak od A. Jednadžbu $w = Ax - b$ ili $Ax - w = b$, zapišimo u matričnom obliku:

$$\begin{bmatrix} A & -I \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} = b.$$

Poravnavajuće varijable daju m jednadžbi. Područje rješenja upravlja s tim jednadžbama, kao i s $n + m$ nejednakosti $x \geq 0$, $w \geq 0$, odnosno nenegativnim ograničenjima.

Budući da simpleks metoda ne razlikuje varijable x i w, možemo pojednostaviti i uvesti sljedeće supstitucije:

$$\tilde{A} = \begin{bmatrix} A & -I \end{bmatrix}, \quad z = \begin{bmatrix} x \\ w \end{bmatrix}, \quad \tilde{c} = \begin{bmatrix} c & 0 \end{bmatrix}.$$

Nakon pojednostavljivanja, matematički model zadaće linearnog programiranja glasi:

$$\text{Minimiziraj } \tilde{c}z \text{ s obzirom na } \tilde{A}z = b \text{ i } z \geq 0.$$

Jedini trag poravnavajuće varijable w sadržan je u dimenzijama nove matrice $\tilde{A}$ i vektora stupca $\tilde{z}$. Nova matrica $\tilde{A}$ je dimenzija $m \times (n + m)$, a vektor stupac $\tilde{z}$ je dimenzije $n + m$.

Primjer 2.1.1. *Zadaća linearnog programiranja sa Slike 2.1 zadana je problemskim ograničenjima:* $x + 2y \geq 6$ *i* $2x + y \geq 6$, $x \geq 0$, $y \geq 0$. *Neka je* $x + y$ *funkcija cilja. Tada je:*

$$\tilde{A} = \begin{bmatrix} 1 & 2 & -1 & 0 \\ 2 & 1 & 0 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 6 \\ 6 \end{bmatrix},$$
$$\tilde{c} = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}.$$

Primijetimo da u Primjeru 2.1.1 imamo četiri nepoznanice: x, y i dvije poravnavajuće varijable.

Napomena 2.1.2. *Zbog jednostavnosti, u nastavku rada, promatramo zadaće linearnog programiranja u kojima su funkcije cilja zadane s* cx*, a ograničenja dana s* $Ax = b$ *i* $x \geq 0$*, pri čemu je* $x \in \mathbb{R}^{n+m}$ *i* $A \in M_{m,m+n}(\mathbb{R})$ *(realna matrica dimenzije* $m \times (n + m)$*).*

2.2 Algoritam simpleks metode

Nakon što smo problemska ograničenja izrazili jednakostima, možemo započeti sa simpleks metodom. Vrh područja rješenja sada predstavlja točku u kojoj n varijabli vektora stupca x, sustava $Ax = b$, imaju vrijednost nula. Tih n varijabli vektora stupca x nazivamo **nebazičnim varijablama**. Preostalih m varijabli vektora stupca x nazivamo **bazičnim varijablama**. **Bazično rješenje** x je rješenje sustava $Ax = b$ koje se dobije kada se za n zadanih nebazičnih varijabli uzme vrijednost nula i sustav se riješi po bazičnim varijablama.

Napomena 2.2.1. *Bazično rješenje x ne pripada području rješenja zadaće linearnog programiranja ako uključuje barem jednu negativnu vrijednost, a pripada ako ne uključuje niti jednu takvu vrijednost. Dakle, pripada li bazično rješenje području rješenja ili ne, možemo jednostavno utvrditi iz predznaka svih varijabli rješenja. Bazično rješenje x koje pripada području rješenja nazivamo* ***dopušteno bazično rješenje****.*

Napomena 2.2.2. *Prva faza simpleks metode pronalazi dopuštena bazična rješenja. Druga faza vodi prema optimalnom rješenju x^*.*

Na Slici 2.1, točka $(0, 6)$ je sjecište pravaca $x = 0$ i $2x + y - 6 = 0$. Točka $(0, 6)$ ujedno je i vrh područja rješenja. Naime, za $x = \begin{bmatrix} 0 & 6 & 6 & 0 \end{bmatrix}^T$ imamo:

$$Ax = \begin{bmatrix} 1 & 2 & -1 & 0 \\ 2 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 6 \\ 6 \\ 0 \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \end{bmatrix} = b.$$

Dakle, za vrh $(0, 6)$ bazično rješenje je $(0, 6, 6, 0)$. Budući da su sve varijable od x nenegativne, zaključujemo da je $(0, 6, 6, 0)$ dopušteno bazično rješenje.

U koji vrh područja rješenja dalje krenuti? Želimo se rubom područja rješenja pomaknuti do susjednog vrha. Budući da prelazimo u susjedni vrh, $m - 1$ baziĉnih varijabli će ostati bazičnima. Jednoj od m bazičnih varijabli dodijelit će se vrijednost nula. Preostalih $m - 1$ varijabli će promijeniti vrijednost, ali će ostati pozitivne. Simpleks metoda će jednu od nebazičnih varijabli proglasiti bazičnom. Izborom ruba područja rješenja odlučujemo koju bazičnu varijablu će zamijeniti nebazična. Bazične varijable računaju se rješavanjem sustava $Ax = b$. Nebazične varijable, vektora stupca x, imaju vrijednost nula. Koju nebazičnu varijablu treba izabrati i proglasiti je bazičnom? Pogledajmo sljedeći primjer:

Primjer 2.2.3. *Minimiziraj $7x_3 - x_4 - 3x_5$ s obzirom na:*

$$\begin{array}{ccccccccccc} x_1 & & & + & x_3 & + & 6x_4 & + & 2x_5 & = & 8\,, \\ & & x_2 & + & x_3 & & & + & 3x_5 & = & 9\,. \end{array} \tag{2.1}$$

Zadaća linearnog programiranja zadana je s tri ($n = 3$) varijable odluke: x_3, x_4 i x_5 i dva ($m = 2$) problemska ograničenja. Funkcija cilja dana je s $7x_3 - x_4 - 3x_5$. Kako je sustav zadan s dvije jednadžbe i pet varijabli, zaključujemo da ima dvije bazične i tri nebazične varijable. Krenimo od vrha u kojem su $x_1 = 8$ i $x_2 = 9$ bazične varijable. Preostale varijable su nebazične i vrijedi: $x_3 = x_4 = x_5 = 0$. Dobili smo dopušteno bazično rješenje. Uvrštavanjem dobivenih vrijednosti u funkciju cilja, dobivamo da je njezina vrijednost 0, ali ne znamo je li to i minimalna vrijednost.

Promotrimo funkciju cilja: $7x_3 - x_4 - 3x_5$. Bilo bi krivo promatrati varijablu x_3 kao bazičnu varijablu, jer je njezin koeficijent u funkciji cilja $+7$, a želimo minimizirati vrijednost. Za bazičnu varijablu odaberimo varijablu x_5. Koeficijent varijable x_5 u funkciji cilja iznosi -3 i najmanji je od ostalih. Stoga za novu bazičnu varijablu (do sada nebazičnu) biramo varijablu x_5 i zovemo je **ulazna varijabla**.

Sada kada smo nebazičnu varijablu x_5 izabrali kao ulaznu varijablu, moramo između dosadašnji bazičnih varijabli x_1 i x_2 odabrati jednu koja će postati nebazičnom. Kako i dalje imamo $x_3 = x_4 = 0$, iz (2.1) slijedi:

$$x_1 + 2x_5 = 8, \quad x_2 + 3x_5 = 9 \ .$$

Ako je x_1 nova nebazična varijabla, tada je nužno $x_5 = 4$, ali onda $x_2 = -3 < 0$, pa to nije dopušteno bazično rješenje. Međutim, ako je x_2 nova nebazična varijabla, onda je $x_5 = 3$, te $x_1 = 2$. Dakle, uzimamo varijablu x_2 kao nebaziČnu varijablu. Varijablu x_2 zovemo **izlaznom varijablom** jer napušta skup baziČnih varijabli, kako bi postala nebaziČnom varijablom. Rješavanjem sustava jednadžbi (2.1) po baziČnim varijablama x_1 i x_5, dobivamo da je $x = (2, 0, 0, 0, 3)$ novi vrh područja rješenja. Vrijednost funkcije cilja u vrhu područja rješenja iznosi:

$$7x_3 - x_4 - 3x_5 = 7 \cdot 0 - 0 - 3 \cdot 3 = -9 \ .$$

Zaključimo, ulazne i izlazne varijable vode nas u novi vrh područja rješenja.

Brži način

Ako vrijednosti desne strane sustava (2.1) podijelimo s koeficijentima uz ulaznu varijablu x_5, dobivamo vrijednosti: $\frac{8}{2}$ i $\frac{9}{3}$. Najmanji količnik govori koja će varijabla prva pogoditi nulu i postati izlazna varijabla. U našem slučaju, varijabla x_2 je izlazna varijabla. U obzir uzimamo samo pozitivne količnike. Naime, ako bi koeficijent uz varijablu x_5 bio -3 tada bi povećanje varijable x_5 rezultiralo i povećanjem varijable x_2. Na primjer, uvrstimo li $x_5 = 10$ u drugu jednadžbu (uz koeficijent -3), dobivamo $x_2 = 39$. U slučaju negativnosti svih koeficijenata varijable x_5, zaključujemo da je funkcija neograničena na području rješenja.

Prethodni korak završava u novom vrhu $x = (2, 0, 0, 0, 3)$. Ako su uz bazične varijable x_1 i x_5 koeficijenti u sustavu (2.1) jednaki jedan, kao što je bio slučaj za x_1 i x_2, onda je sljedeći korak jednostavan. U suprotnome, *pivotiramo* zamijenom varijable x_5 u funkciji cilja i prvoj jednadžbi. Iskažimo drugu jednadžbu sustava (2.1) preko varijable x_5:

$$x_5 = 3 - \tfrac{1}{3}x_2 - \tfrac{1}{3}x_3 \ ,$$

te uvrstimo u prvu jednadžbu i funkciju cilja:

$$7x_3 - x_4 - 3x_5 = 7x_3 - x_4 - 3(3 - \tfrac{1}{3}x_2 - \tfrac{1}{3}x_3) = x_2 + 8x_3 - x_4 - 9\ ,$$

i

$$\begin{array}{rcl} x_1 + x_3 + 6x_4 + 2x_5 = 8 & \Longleftrightarrow & x_1 + x_3 + 6x_4 + 2(3 - \tfrac{1}{3}x_2 - \tfrac{1}{3}x_3) = 8 \\ & \Longleftrightarrow & x_1 - \tfrac{2}{3}x_2 + \tfrac{1}{3}x_3 + 6x_4 = 2 \end{array} .$$

Time za novi vrh područja rješenja dobivamo sljedeću zadaću linearnog programiranja:

Minimiziraj $x_2 + 8x_3 - x_4 - 9$ *s obzirom na*

$$\begin{array}{ccccccccc} x_1 & - & \frac{2}{3}x_2 & + & \frac{1}{3}x_3 & + & 6x_4 & & & = & 2\ , \\ & & \frac{1}{3}x_2 & + & \frac{1}{3}x_3 & & & + & x_5 & = & 3\ . \end{array} \tag{2.2}$$

Sljedeći korak je jednostavan. U funkciji cilja, jedino varijabla x_4 ima negativan koeficijent -1, stoga nju uzimamo kao ulaznu varijablu. Podijelimo li desnu stranu sustava (2.2) s koeficijentima uz varijablu x_4 dobivamo količnike: $\frac{2}{6}$ i $+\infty$ (koristili smo konvergenciju $\frac{2}{0} = +\infty$). Iz vrijednosti količnika, zaključujemo da je varijabla x_1 izlazna varijabla. Rješavajući sustav (2.2) po baznim varijablama x_4 i x_5, dobivamo novi vrh područja rješenja: $x = (0, 0, 0, \frac{1}{3}, 3)$. Vrijednost funkcije cilja u dobivenom vrhu je:

$$x_2 + 8x_3 - x_4 - 9 = 0 + 8 \cdot 0 - \tfrac{1}{3} - 9 = -\tfrac{28}{3}\ .$$

Prethodno iskazana vrijednost predstavlja minimalnu vrijednost zadaće linearnog programiranja, zato se funkcija cilja u idućem koraku ne mijenja (kod pivotiranja je dovoljno samo prvu jednadnžbu u (2.2) podijeliti sa 6), a osim x_4 (koja je već bazična) nemamo niti jednu drugu varijablu s negativnim koeficijentima, pa bi svaka promjena baznih varijabli povećala vrijednost funkcije cilja.

U velikim zadaćama linearnog programiranja, može se dogoditi da se izlazna varijabla vrati među ulazne, odnosno bazične varijable. Bez obzira na to, vrijednost funckije cilja će opadati, izuzev degenerirajućih slučajeva (pogledaj Napomenu 2.2.4), stoga se m baznih varijabli stalno mijenja. Nikad se ne vraćamo u vrhove u kojima smo već bili. Simpleks metoda mora završiti u optimalnom vrhu. Izvanredna je brzina kojom simpleks metoda pronalazi optimalno rješenje x^* zadaće linearnog programiranja. Područje rješenja može imati stotine ili tisuće vrhova, a da simpleks metoda pronađe optimalno rješenje u desetak koraka.

Rezimirajmo, koeficijenti funkcije cilja u prvom vrhu $7, -1, -3$ i $1, 8, -1$ u drugom vrhu odlučuju o ulaznoj varijabli. Ti brojevi ulaze u vektor r, važan vektor definiran u sljedećem odjeljku. Količnici, dobiveni dijeljenjem desne strane sustava s koeficijentima ulazne varijable, odlučuju o izlaznoj varijabli.

Napomena 2.2.4. *Vrh područja rješenja je degeneriran ako je više od n varijabli vektora stupca x vrijednosti nula. Dakle, više od n ravnina prolazi jednim vrhom područja rješenja, stoga se dogodi da bazična varijabla nestane. Količnici, koji odlučuju o izlaznoj varijabli, uključivat će nule i moguće je da se bazične varijable promijene bez promjene vrha područja rješenja. U teoriji, moguće je da ostanemo uvijek u jednom vrhu, a da samo ciklički mijenjamo bazične varijable. Iako je degeneriranost česta u primjenama, cikliranje se rijetko događa. Već nakon nekoliko ponavljanja iste vrijednosti funkcije cilja, algoritam prelazi na vrh s optimalnijom vrijednosti.*

2.3 Implementacija pomoću simpleks tablice

Svaki korak simpleks metode sadrži odluke koje se sastoje u odabiru ulaznih i izlaznih varijabli. Korake simpleks metode možemo organizirati postavimo li A, b i c u veliku matricu ili **simpleks tablicu**. Simpleks tablica T je matrica dimnezija $(m+1)\times(m+n+1)$, oblika

$$T = \begin{bmatrix} A & b \\ c & 0 \end{bmatrix}.$$

Na početku, bazične varijable mogu biti pomiješane s nebazičnim varijablama. Ponovno numeriranje je nužno. Pretpostavimo da su $x_1, ..., x_m$ bazične varijable u nekom vrhu područja rješenja. Prvih m stupaca matrice A tvore kvadratnu matricu B, odnosno *bazičnu matricu*. Matrica B mora biti regularna ako smo dobro napravili prvu fazu traženja vrha područja rješenja, jer bi u protivnom presjek odabranih n ravnina bio beskonačan skup. Preostalih n stupaca matrice A tvore matricu N, dimenzije $m \times n$, odnosno *nebazičnu matricu*. Vektor cilja c zapisujemo kao vektor redak oblika $\begin{bmatrix} c_B & c_N \end{bmatrix}$. Nepoznanicu x zapisujemo kao (x_B, x_N).

U vrhu područja rješenja, nebazične varijable x_N imaju vrijednost nula. Stoga, sustav $Ax = b$ pretvaramo u ekvivalentni sustav $Bx_B = b$, odnosno $x_B = B^{-1}b$. Time posebno imamo da su sve komponente vektora $B^{-1}b \geq 0$. Nadalje, vrijednost funkcije cilja je $cx = c_B x_B = c_B B^{-1} b$. Dakle, trenutno imamo sljedeće:

$$\textbf{Simpleks tablica vrha: } T = \left[\begin{array}{c:c:c} B & N & b \\ \hdashline c_B & c_N & 0 \end{array}\right], x_N = 0, x_B = B^{-1}b, \text{cilj} = c_B B^{-1} b.$$

Kada sustav $Ax = b$, prikazanog u simpleks tablici vrha T, pomnožimo s B^{-1}, bazične varijable će ostati same i dobivamo:

$$\textbf{Smanjena simpleks tablica: } T' = \left[\begin{array}{c:c:c} I & B^{-1}N & B^{-1}b \\ \hdashline c_B & c_N & 0 \end{array}\right].$$

Potpuno smanjenu simpleks tablicu dobit ćemo ako gornji blok redak matrice T' pomnožimo s c_B i oduzmemo od donjeg blok retka tablice T':

$$\textbf{Potpuno smanjena simpleks tablica: } R = \left[\begin{array}{c:c:c} I & B^{-1}N & B^{-1}b \\ \hdashline 0 & c_N - c_B B^{-1}N & -c_B B^{-1}b \end{array}\right].$$

Uočimo da smo matricu R dobili iz T primjenom konačno mnogo Gaussovih eliminacija. Promotrimo, sa strane algebre, značenje varijabli danih u simpleks tablici:

$$\text{Ograničenja: } Bx_B + Nx_N = b \Leftrightarrow x_B + B^{-1}Nx_N = B^{-1}b \ ,$$
$$\text{Vrh: } x_B = B^{-1}b, x_N = 0 \ ,$$

što možemo pročitati iz gornjih blokova matrice R. Funkcija cilja zadana sa $c_Bx_B + c_Nx_N$ pretvorena je u:

$$\text{Funkcija cilja: } cx = (c_N - c_B B^{-1}N)x_N + c_B B^{-1}b \ ,$$
$$\text{Funkcija cilja u vrhu: } = c_B B^{-1}b \ ,$$

što možemo pročitati iz posljednjeg retka matrice R. U potpuno smanjenoj simpleks tablici R pojavljuju se sve varijable važne za provedbu simpleks metode. U srednjem stupcu donjeg retka blok tablice R nalazi se r, zadan s $r = c_N - c_B B^{-1}N$. Njime utvrđujemo optimalnost vrha područja rješenja. Ako je bilo koji element od r negativan, onda vrijednost funkcije cilja možemo dodatno minimizirati. Ako je vrijednost od r nenegativna, onda je pronađen najbolji vrh, odnosno optimalno rješenje. Dakle, s r je određen kraj simpleks metode i dan **uvjet optimalnosti**: vrh područja rješenja je optimalno rješenje zadaće linearnog programiranja ako vrijedi: $r = c_N - c_B B^{-1}N \geq 0$. Vrijednost funkcije cilja u optimalnom vrhu je $c_B B^{-1}b$. Negativni elementi varijable r odgovaraju rubovima područja rješenja u kojima vrijednost funkcije cilja opada. Ulazna varijabla x_i odgovara elementu varijable r s najmanjom negativnom vrijednosti.

Pretpostavimo da je r_i najmanji negativni element varijable r. Tada je i-ti element od x_N ulazna varijabla. Budući da je varijabla x_i bila nebazična, njezina vrijednost raste od nule do pozitivne vrijednosti α, koju će imati u novom vrhu područja rješenja. Kako se vrijednost x_i povećava, ostali elementi varijable x mogu se smanjiti da održe jednakost sustava $Ax = b$. Izlaznu varijablu dobivamo tako da tražimo minimum količnika

$$\frac{(B^{-1}b)_j}{(B^{-1}N)_{ji}} \ , j = 1, \ldots, n \ ,$$

pri čemu uzimamo u obzir samo one j za koje $(B^{-1}b)_{ji} > 0$. Pretpostavimo da se najmanja vrijednost postiže za $j = k$. Tada je k-ti element vektora x_B izlazna varijabla, a vrijednost ulazne varijable je

$$\alpha = \frac{(B^{-1}b)_k}{(B^{-1}N)_{ki}} \ .$$

Sada zamijenimo k-ti i $(n+1)$-ti stupac matrice R, te ponovimo postupak Gaussovih eliminacija kao kod početne matrice T. Primijetimo da je gornji lijevi $m \times n$ blok transformiranje matrice ponovno regularna matrica jer $(B^{-1}N)_{ki} \neq 0$.

Ako bi sve vrijednosti $(B^{-1}N)_{ji}, j = 1, \ldots, n,$ bile manje ili jednake nula, odnosno ako bi svi elementi stupca matrice $B^{-1}N$ iznad najmanje negativne vrijednosti vektora r bili negativni ili jednaki nula, tada bi se novi vrh nalazio beskonačno daleko i minimalna vrijednost funkcije cilja bi iznosila $-\infty$. U sljedećem primjeru koristimo simpleks tablicu za rješavanje zadaće minimizacije, čije područje rješenja je prikazano na Slici 2.1.

Primjer 2.3.1. *Minimizirajmo $x + y$ s obzirom na $x + 2y \geq 6$ i $2x + y \geq 6$.*

Rješenje. Zadaća linearnog programiranja zadana je funkcijom cilja $x + y$ i dva problemska ograničenja: $x + 2y \geq 6$ i $2x + y \geq 6$. Budući da su: $n = 2$ i $m = 2$, simpleks tablica T, određena s A, b i c, poprimit će dimenziju 3×5. Problemska ograničenja zadana nejednadžbama, pomoću poravnavajućih varijabli, pretvorimo u jednadžbe. Slijedi:

$$A = \begin{bmatrix} 1 & 2 & -1 & 0 \\ 2 & 1 & 0 & -1 \end{bmatrix}, \; b = \begin{bmatrix} 6 \\ 6 \end{bmatrix} \text{ i } c = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}.$$

Uvrštavanjem u simpleks tablicu T, dobivamo:

$$T = \begin{bmatrix} A & b \\ c & 0 \end{bmatrix} = \left[\begin{array}{cccc:c} 1 & 2 & -1 & 0 & 6 \\ 2 & 1 & 0 & -1 & 6 \\ \hdashline 1 & 1 & 0 & 0 & 0 \end{array}\right].$$

Pravac $x = 0$ siječe pravac $2x + y = 6$ u točki $A = (0, 6)$, odnosno u vrhu područja rješenja. Organizirajmo simpleks tablicu T tako da bazične varijable stavimo ispred nebazičnih. U našem slučaju, zamijenimo prvi i treći stupac:

$$\textbf{Simpleks tablica vrha } A\textbf{:}\; T_A = \left[\begin{array}{cc:cc:c} -1 & 2 & 1 & 0 & 6 \\ 0 & 1 & 2 & -1 & 6 \\ \hdashline 0 & 1 & 1 & 0 & 0 \end{array}\right].$$

Kako bismo dobili potpuno smanjenu simpleks tablicu vrha $(0, 6)$, za početak moramo prvi redak od T_A pomnožiti s -1. Budući da se lijevo, u gornjem retku blok tablice T_A, mora nalaziti jedinična matrica, drugi redak tablice moramo pomnožiti s 2 i pridodati prvom retku tablice. Isto tako, lijevo, u donjem retku blok tablice T_A, mora se nalaziti nul vektor redak. Dobivamo ga tako da drugi redak simpleks tablice T_A pomnožimo s -1 i pridodamo zadnjem retku. Na taj način dobivamo potpuno smanjenu simpleks tablicu vrha A, odnosno R_A.

Potpuno smanjena simpleks tablica vrha A:

$$R_A = \left[\begin{array}{cc:cc:c} 1 & -2 & -1 & 0 & -6 \\ 0 & 1 & 2 & -1 & 6 \\ \hdashline 0 & 1 & 1 & 0 & 0 \end{array}\right] \sim \left[\begin{array}{cc:cc:c} 1 & 0 & 3 & -2 & 6 \\ 0 & 1 & 2 & -1 & 6 \\ \hdashline 0 & 1 & 1 & 0 & 0 \end{array}\right] \sim \left[\begin{array}{cc:cc:c} 1 & 0 & 3 & -2 & 6 \\ 0 & 1 & 2 & -1 & 6 \\ \hdashline 0 & 0 & -1 & 1 & -6 \end{array}\right].$$

Promotrimo $r = \begin{bmatrix} -1 & 1 \end{bmatrix}$ u donjem srednjem retku potpuno smanjene simpleks tablice R_A. Zbog negativne vrijednosti (-1) koja se pojavljuje u trećem stupcu od R_A, treća varijabla će biti ulazna varijabla. U trenutnom vrhu A, vrijednost funkcije cilja iznosi +6 i nije optimalna. Stupac iznad negativne vrijednosti (-1) je $(3, 2)$. Izlaznu varijablu određujemo odabirom najmanjeg pozitivnog količnika dobivenog dijeljenjem koeficijenata posljednjeg s koeficijentima trećeg stupca tablice R_A. Dobivamo sljedeće količnike: $\frac{6}{3} = 2$ i $\frac{6}{2} = 3$. Budući da je prvi količnik manji, varijabla dana prvim stupcem tablice R_A, postaje nebazičnom (izlaznom) varijablom. Na taj način pomaknuli smo se iz vrha $A = (0, 6)$ u vrh $B = (2, 2)$ područja rješenja prikazanog na Slici 2.1. Provodimo isti, ranije opisani, postupak. Organizirajmo simpleks tablicu tako da bazične varijable stavimo ispred nebazičnih, odnosno zamijenimo prvi i treći stupac:

$$\textbf{Simpleks tablica vrha } B\textbf{:}\ T_B = \left[\begin{array}{cc:cc:c} 3 & 0 & 1 & -2 & 6 \\ 2 & 1 & 0 & -1 & 6 \\ \hdashline -1 & 0 & 0 & 1 & -6 \end{array}\right].$$

Potpuno smanjenu simpleks tablicu vrha B dobivamo u nekoliko koraka. Za početak podijelimo s 3 prvi redak od T_B. Budući da se u gornjem lijevom retku blok tablice T_B mora nalaziti jedinična matrica, prvi redak tablice trebamo pomnožiti s -2 i pridodati drugom retku. U donjem lijevom retku blok tablice T_B mora se nalaziti nul redak. Nul redak dobivamo tako da prvi redak tablice T_B pridodamo zadnjem retku. Na taj način dobili smo potpuno smanjenu simpleks tablicu vrha B, odnosno R_B.

Potpuno smanjena simpleks tablica vrha B:

$$R_B = \left[\begin{array}{cc:cc:c} 1 & 0 & \frac{1}{3} & -\frac{2}{3} & 2 \\ 2 & 1 & 0 & -1 & 6 \\ \hdashline -1 & 0 & 0 & 1 & -6 \end{array}\right] \sim \left[\begin{array}{cc:cc:c} 1 & 0 & \frac{1}{3} & -\frac{2}{3} & 2 \\ 0 & 1 & -\frac{2}{3} & \frac{1}{3} & 2 \\ \hdashline 0 & 1 & 1 & 0 & 0 \end{array}\right] \sim \left[\begin{array}{cc:cc:c} 1 & 0 & \frac{1}{3} & -\frac{2}{3} & 2 \\ 0 & 1 & -\frac{2}{3} & \frac{1}{3} & 2 \\ \hdashline 0 & 0 & \frac{1}{3} & \frac{1}{3} & -4 \end{array}\right].$$

U potpuno smanjenoj simpleks tablici R_B, $r = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \end{bmatrix}$. Budući da su svi elementi od r pozitivni, simpleks metoda staje. Zaključujemo da se u vrhu $B = (2, 2)$ postiže optimalno rješenje zadaće linearnog programiranja i da miniminalna vrijednost funkcije cilja iznosi 4.

2.4 Ogranizacija simpleks metode

Geometrijska intepretacija simpleks metode iskazana je algebarski. Vrhovi područja rješenja dani su dopuštenim baz̄ičnim rješenjima. U simpleks metodi, odlučujuću ulogu igraju vektor redak r i količnik α. Oni predstavljaju srž simpleks metode, a mogu biti organizirani na tri različita načina:

1. U simpleks tablici.
2. Ažuriranjem B^{-1} prilikom zamjene stupca u iz N i stupca k iz B.
3. Računanjem $B = LU$, odnosno LU faktorizacijom kvadratne matrice B, te ažuriranjem matrica L i U prilikom zamjene stupca u iz N i stupca k iz B.

Prethodnim popisom predstavljena je kratka povijest simpleks metode. Simpleks tablica je dominirala mnogo godina. Za mnoge je unijela dašak zagonetnosti u linearno programiranje, prvenstveno zbog toga što je gotovo u potpunosti uspjela izbjeći matričnu notaciju. Dani simpleks tablice, za velike zadaće, su završeni. Promotrimo zašto. Najmanji negativni element od r pokazuje koji će stupac u postati ulazna varijabla. Preostali stupci iznad r se neće koristiti. Njihovo računanje predstavlja gubitak vremena. U većim zadaćama linearnog programiranja, vrijednosti elemenata stotine stupaca određivat će se u svakom koraku, čekajući da postanu ulazne varijable. U teoriji, to omogućava shvaćanje simpleks metode u potpunosti, ali u praksi, to može biti dugotrajan proces.

Brži i jednostavniji način je pogledati koji su izračuni stvarno potrebni. U svakom koraku simpleks metode, stupac matrice N i stupac matrice B zamijene mjesta. Koji stupac se mijenja kojim, određeno je vrijednostima od r i α. Metoda započinje s danom baz̄ičnom kvadratnom matricom B i trenutnim rješenjem $x_B = B^{-1}b$.

Koraci simpleks metode

1. Izračunati vektor redak $\lambda = c_B B^{-1}$ i smanjiti vrijednost $r = c_N - \lambda N$.
2. Ako je $r \geq 0$, simpleks metoda staje i trenutni vrh je optimalno rješenje zadaće linearnog programiranja. U suprotnom, ako je r_i najmanji negativni element od r, onda uzimamo stupac u, odnosno i-ti stupac od N, kao ulaznu varijablu.
3. Odrediti količnike dobivene dijeljenjem elemenata $B^{-1}b$ s $B^{-1}u$, promatrajući samo pozitivne elemente od $B^{-1}u$. Ako je $B^{-1}u < 0$, onda je minimalna vrijednost $-\infty$. Ako se najmanji količnik postiže u k-tom elementu od $B^{-1}u$, onda k-ti stupac trenutne bazične matrice postaje izlazna varijabla.
4. Ažurirati B^{-1} ili matrice L i U i rješenje $x_B = B^{-1}b$. Vratiti se na prvi korak.

Ponekad simpleks metodu, danu u prethodnim koracima, nazivamo izmijenjenom simpleks metodom kako bi se razlikovala od operacija u simpleks tablici.

Poglavlje 3

Dualna zadaća

3.1 Uvod u dualnu zadaću

Iako nam algoritam simpleks metode omogućava rješavanje zadaće linearnog programiranja, u središtu njezinog razmatranja nalazi se teorija dualnosti. Teoriju započinjemo danom *primarnom zadaćom*:

$$\text{Minimiziraj } cx \text{ s obzirom na } Ax \geq b \text{ i } x \geq 0\ .$$

Kao i primarna, dualna zadaća zadana je s A, b i c, ali preokrenuto. U primarnoj zadaći c je bio zadan funkcijom cilja, a b problemskim ograničenjima. U dualnoj zadaći je obratno, odnosno c je zadan problemskim ograničenjima, a b funkcijom cilja. Nepoznanica y dualne zadaće je vektor redak dimenzije m. Za razliku od primarne zadaće u kojoj su problemska ograničenja dana s $Ax \geq b$, u dualnoj su dana s $yA \leq c$. Svaka zadaća minimizacije povezana je sa zadaćom maksimizacije, stoga **dualna zadaća** problema linearnog programiranja glasi:

$$\text{Maksimiziraj } yb \text{ s obzirom na } yA \leq c \text{ i } y \geq 0\ .$$

Dualna zadaća dualne zadaće je početna primarna zadaća. Naime, zapišimo dualnu zadaću kao zadaću minimizacije:

$$\max\, yb = -\min\,(-yb) = -\min\,((-b)^T y^T)\ ,$$

uz

$$(-A^T)y^T = -(yA)^T \geq -c^T \quad \text{i} \quad y^T \geq 0\ .$$

Dakle, y^T zadovoljava standardan oblik minimizacijskog problema uz matricu $-A^T$ i desnu stranu nejednakosti $-c^T$, te vektor $-b^T$ u funkciji cilja.

Ako ovu zadaću shvatimo kao primarnu, pripadna dualna zadaća sada glasi

$$-\max\,(z(-c^T))$$

uz uvjete

$$\begin{aligned} z(-A^T) &\leq -b^T \ , \\ z &\geq 0 \ , \end{aligned}$$

gdje je z vektor redak dimenzije n. Uz $x = z^T$ konačno dobivamo

$$-\max\,(z(-c^T)) = \min\,(zc^T) = \min\, cz^T = \min\, cx \ ,$$

te

$$\begin{aligned} z(-A^T) \leq -b^T &\iff zA^T \geq b^T \\ &\iff Az^T \geq b \\ &\iff Ax \geq b \ , \end{aligned}$$

čime smo zaista dobili primarnu zadaću. Kao i svaku drugu zadaću linearnog programiranja, dualnu zadaću možemo rješavati primjenom simpleks metode. Čak štoviše, obje zadaće možemo riješiti odjednom.

Intepretirajmo primarnu i dualnu zadaću problema dijete promatranog u prvom poglavlju, ali općenito. U problemu moramo minimizirati trošak dijete. U dijetu ulaze n vrsta namirnica. Od svake vrste $x_1, ..., x_n$ moramo odabrati nenegativan broj komada. Ograničenja su dana s m vrsta potrebnih vitamina. Sukladno primarnoj zadaći u kojoj vrijedi: $Ax \geq b$, $x \geq 0$, ograničenja možemo zapisati tako da je s a_{ij} označena količina i-tog vitamina u j-toj namirnici. i-ti redak sustava $Ax \geq b$ mora sadržavati najmanje b_i određenog vitamina kako bi se zadovoljile potrebe dijete. Ako je s c_j označen trošak j-te namirnice, tada je trošak dijete: $cx = c_1x_1 + ... + c_nx_n$. Promatrano u matričnom obliku imamo:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}, \quad c = \begin{bmatrix} c_1 & c_2 & \cdots & c_n \end{bmatrix}.$$

Promotrimo dualnu zadaću, zadaću maksimizacije. Pretpostavimo da ljekarnik prodaje vitaminske tablete po cijeni $y_i \geq 0$. Budući da namirnica j sadrži vitamine u količini a_{ij}, ljekarnikova cijena vitaminskih tableta ne može prekoračiti cijenu namirnice c_j. To predstavlja j-to ograničenje u $yA \leq c$. Radeći unutar ograničenja cijena vitamina, ljekarnik može prodati potrebnu količinu b_i svakog vitamina. Ukupan prihod prodaje iznosi:

$y_1 b_1 + ... + y_m b_m = yb$. Prihod od prodaje ljekarnik želi maksimizirati.

Primarna zadaća je podskup $\mathbb{R}^n$ (odnosno vektor stupac dimenzije n), dana ograničenjima: $Ax \geq b$ i $x \geq 0$. Dualna zadaća je podskup $\mathbb{R}^m$ (odnosno vektor redak dimenzije m) određena sa: $y \geq 0$, $yA \leq c$. O vezi između primarne i dualne zadaće ovisi cijela teorija linearnog programiranja. U sljedećem teoremu dan je osnovni rezultat:

Teorem 3.1.1. ***Teorem dualnosti.*** *Ako su područja rješenja primarne i dualne zadaće neprazna, onda imaju optimalna rješenja x^* i y^*. Minimalna vrijednost (trošak) cx^* primarne zadaće jednaka je maksimalnoj vrijednosti y^*b (dobit) dualne zadaće.*

Dokaz teorema dan je u sljedećem odjeljku.

Ako optimalna rješenja x^* i y^* ne postoje, onda imamo dvije mogućnosti (slijedi iz slabe dualnosti, Teorem 3.1.2.):

- Područja rješenja primarne i dualne zadaće su prazni skupovi.
- Jedno područje rješenja je prazan, a drugo područje rješenja neograničeni skup (maksimum je $+\infty$ ili minimum je $-\infty$).

Teorem dualnosti utvrđuje konkurentnost između trgovca namirnicama i ljekarnika. Ne postoji razlog zbog kojeg bi kupac preferirao kupnju vitaminskih tableta umjesto namirnica, iako ljekarnik jamči zadovoljavanje svih vitamina koji se mogu naći i u namirnicama. Pokazat ćemo da skupe namirnice (kao maslac od kikirikija) neće ući u dijetu, kako bi ishod zadaća bio ujednačen.

Optimalna rješenja sadrže važne informacije. U primarnoj zadaći, x^* govori kupcu koje namirnice kupiti. U dualnoj zadaći, y^* upravlja cijenama proizvoda (praćenjem vrijednosti) na kojima se treba temeljiti gospodarstvo. Ako naša zadaća linearnog programiranja odražava pravo gospodarstvo, tada x^* i y^* predstavljaju važne odluke.

Želimo dokazati da vrijedi: $cx^* = y^*b$. Ljekarnik može povećati cijenu vitaminskih tableta y^* do te mjere da dostigne cijene namirnica x^*. Jedna činjenica je potpuno jasna: Sve dok, bez povećavanja troška, namirnice možemo zamijeniti s njihovim vitaminskim ekvivalentima, cijene dijeta moraju iznositi barem koliko i cijene vitaminskih tableta. To predstavlja jednostranu nejednakost: *cijena vitamina ≤ cijena namirnica*. Prethodnu nejednakost nazivamo slaba dualnost i lako se dokazuje za bilo koju zadaću linearnog programiranja i njenu dualnu zadaću.

Teorem 3.1.2. ***Slaba dualnost.*** *Ako su x i y točke područja rješenja primarne i dualne zadaće, onda je $yb \leq cx$.*

Dokaz. Budući da su x i y točke područja rješenja, moraju biti zadovoljena sljedeća ograničenja: $Ax \geq b$ i $yA \leq c$. Iz definicije primarne i dualne zadaće slijedi: $x \geq 0$ i $y \geq 0$. Iz nenegativnosti komponenti vektora x i y, množenjem s vektorom x, odnosno y, smjer nejednakosti ostaje sačuvan:

$$yAx \geq yb \quad \text{i} \quad yAx \leq cx \,.$$

Budući da vrijedi: $yb \leq yAx \leq cx$, slijedi slaba dualnost $yb \leq cx$. □

Jednostrana nejednadžba ne dozvoljava da obje zadaće (primarna i dualna) linearnog programiranja budu neograničene. Ako je vrijednost yb proizvoljno velika, točka x mogla bi proturječiti nejednakosti $yb \leq cx$. Slično, ako vrijednost od cx ode u $-\infty$, onda dualna zadaća ne može odabrati proizvoljnu točku y područja rješenja. Dakle, ako je dualna zadaća neomeđena (odozgo), onda je područje rješenja primarne zadaće nužno prazan skup, i obratno. Također, ako želimo postići jednakost $yb = cx$, x i y moraju biti optimalne točke. Možemo prepoznati optimalni trošak dijete i optimalnu cijenu vitaminskih tableta ako kupac nema ništa drugo za izabrati:

Propozicija 3.1.3. *Ako su x i y točke područja rješenja i ako je $cx = yb$, onda su x i y optimalna rješenja.*

Dokaz. Budući da točka y područja rješenja ne može ostvariti vrijednost yb veću od cx, znači da y doseže svoje optimalno rješenje. Slično, za bilo koji x, koji dostiže vrijednost $cx = yb$, mora vrijediti da je optimalan (x^*). □

U sljedećem primjeru, dan je problem dijete određen s dvije namirnice i dva vitamina. Primijetite da se transponirana matrica A pojavljuje u zapisu dualne zadaće, budući da $yA \leq c$ za retke, podrazumijeva $A^T y^T \leq c^T$ za stupce.

Primjer 3.1.4. ***Primarna zadaća.*** *Minimiziraj $x_1 + 4x_2$ s obzirom na*

$$\begin{gathered} 2x_1 + x_2 \geq 6 \,, \\ 5x_1 + 3x_2 \geq 7 \,, \\ x_1 \geq 0 \,,\ x_2 \geq 0 \,. \end{gathered}$$

Dualna zadaća. *Maksimiziraj $6y_1 + 7y_2$ s obzirom na*

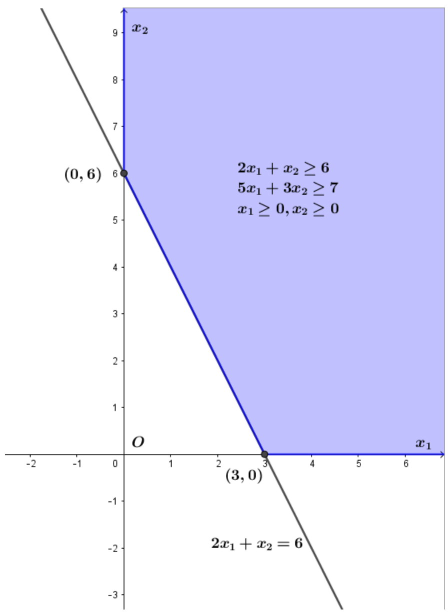
$$\begin{gathered} 2y_1 + 5y_2 \leq 1 \,, \\ y_1 + 3y_2 \leq 4 \,, \\ y_1 \geq 0, \ y_2 \geq 0 \,. \end{gathered}$$

Rješenje. Zadaće linearnog programiranja možemo riješiti jednom od ranije prikazanih metoda rješavanja. U ovom slučaju, promotrimo na Slici 3.1 područje rješenja primarne i na Slici 3.2 područje rješenja njene dualne zadaće linearnog programiranja. Vektor $x = (x_1, x_2) = (3, 0)$ pripada području rješenja primarne zadaće, a vrijednost funkcije cilja u danom vektoru iznosi: $x_1 + 4x_2 = 3 + 4 \cdot 0 = 3$. U dualnoj zadaći, vektor $y = (y_1, y_2) = (\frac{1}{2}, 0)$ pripada području rješenja, a vrijednost funckije cilja u danom vektoru iznosi: $6y_1 + 7y_2 = 6 \cdot \frac{1}{2} + 7 \cdot 0 = 3$. Primijetimo da primarna i dualna zadaća imaju jednake vrijednosti, pa prema Propoziciji 3.1.3. slijedi da su x i y optimalna rješenja primarne, odnosno dualne zadaće linearnog programiranja.

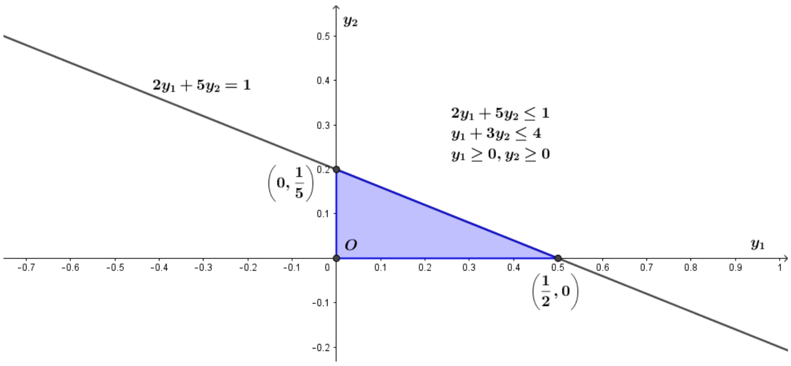
Pogledajmo što se zapravo događa u trenutku kada je $yb = cx$. Neka ograničenja, zadana nejednadžbama, su *jaka* ograničenja, koja omogućuju da prethodna jednakost vrijedi. Preostala ograničenja su *slaba* ograničenja, čije ključno pravilo daje gospodarski smisao:

1. Kada je cijena namirnice j veća od cijene vitaminskog ekvivalenta, onda je količina $x_j^* = 0$.

2. Kada je količina vitamina i u dijeti x^* preobilna, onda je cijena $y_i^* = 0$.



Slika 3.1: Područje rješenja primarne zadaće



Slika 3.2: Područje rješenja dualne zadaće

U prethodnom primjeru, varijabla x_2 je imala vrijednost nula, zato što je druga namirnica bila preskupa. Njena cijena prelazi cijenu vitamina. S ograničenjem $y_1 + 3y_2 \leq 4$ dana je jaka nejednakost, jer vrijedi $\frac{1}{2} + 0 < 4$. Slično, dijeta je zahtjevala zadovoljavanje barem sedam jedinica drugog vitamina, ali zapravo isporučuje $5x_1 + 3x_2 = 5 \cdot 3 + 3 \cdot 0 = 15$ jedinica drugog vitamina, dakle 8 jedinica više nego što je potrebno. U dualnoj zadaći, $y_2 = 0$ što znači da su ti vitamini *besplatno dobro*. Vidimo da je dualnost potpuna.

U terminima matrica, jednostavno možemo razumijeti ove uvjete optimalnosti. Iz nejednakosti $yAx \geq yb$ želimo dobiti jednakost $y^*Ax^* = y^*b$, koja se postiže za optimalno rješenje zadaće linearnog programiranja, po Teoremu 3.1.1. (kojeg dokazujemo u sljedećem odjeljku). Područje rješenja zahtjeva $Ax^* \geq b$, stoga tražimo bilo koji element $(Ax^*)_i$ za koji vrijedi $(Ax^*)_i = b_i$. To odgovara prevelikoj količini vitamina iz našeg primjera, stoga je njegova cijena $y_i^* = 0$. U isto vrijeme, imamo $y^*A \leq c$. Sve jake nejednakosti (skupe namirnice) odgovaraju $x_j^* = 0$ (izostavljanje dijete). To nam je ključ potreban za $y^*Ax^* = cx^*$. Dani uvjeti optimalnosti su **komplementarni poravnavajući uvjeti** linearnog programiranja i **Kuhn-Tuckerov uvjet** nelinearnog programiranja:

Propozicija 3.1.5. *Neka su x i y iz područja rješenja primarne, odnosno dualne zadaće. x i y su optimalna rješenja primarne, odnosno dualne zadaće ako i samo ako vrijedi*

$$y(Ax - b) = 0 \quad i \quad (c - yA)x = 0 \,.$$

Preciznije, x i y su optimalna rješenja ako i samo ako vrijedi:

i. $y_i > 0 \Longrightarrow (Ax)_i = b_i$;

ii. $(Ax)_i > b_i \Longrightarrow y_i = 0$;

iii. $x_i > 0 \Longrightarrow (yA)_i = c_i$;

iv. $(yA)_i < c_i \Longrightarrow x_i = 0$.

Dokaz. Kako su x i y optimalna rješenja, po Teoremu 3.1.1 vrijedi

$$cx = yb \ .$$

S druge strane, x i y su iz područja rješenja pa vrijedi

$$yb \leq yAx \leq cx \ ,$$

odnosno, nužno vrijedi $yb = yAx = cx$. Sada očito vrijedi $y(Ax - b) = yAx - yb = 0$ i $(c - yA)x = cx - yAx = 0$.

Obratno, neka su x i y točke područja rješenja takve da vrijedi

$$y(Ax - b) = 0 \quad \text{i} \quad (c - yA)x = 0 \ .$$

Dakle, vrijedi

$$yAx = yb \quad \text{i} \quad yAx = cx$$

odnosno

$$yb = yAx = cx \ .$$

Time smo dokazali da su x i y optimalna rješenja zadaće linearnog programiranja. □

Promotrimo teoriju dualnosti kao proširenje Lagrangeove metode određivanja uvjetnih ekstrema. Tu metodu nazivamo i metodom Lagrangeovih multiplikatora. Promatramo minimizaciju funkcije $f(x)$, pri čemu $x \in \mathbb{R}^n$ zadovoljava $g_i(x) \leq 0, i = 1, 2, ..., m$. Ako minimum x^* postoji, onda postoje $\lambda_1, \lambda_2, ..., \lambda_m \geq 0$ tako da

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^*) = 0 \ , \tag{3.1}$$

te $\lambda_i g_i(x^*) = 0, i = 1, 2, \ldots, m$. Promotrimo funkciju $f = cx$ koju treba minimizirati uz dodatne uvjete $Ax \geq b$ i $x \geq 0$. Iz dodatnih uvjeta slijedi:

$$\begin{gathered}
g_1(x) := b_1 - a_{11}x_1 - a_{12}x_2 - \ldots - a_{1n}x_n \leq 0 \ , \\
\vdots \\
g_m(x) := b_m - a_{m1}x_1 - a_{m2}x_2 - \ldots - a_{mn}x_n \leq 0 \ , \\
h_1(x) := -x_1 \leq 0 \ , \\
\vdots \\
h_n(x) := -x_n \leq 0 \ .
\end{gathered}$$

Promatranjem (3.1) u gornjim terminima linearnog programiranja, uvrštavanjem poznatih veličina dobivamo postojanje $\lambda_1, \lambda_2, ..., \lambda_m, \mu_1, \mu_2, ..., \mu_n \geq 0$ tako da

$$c^T + \lambda_1 \begin{bmatrix} -a_{11} \\ \vdots \\ -a_{1n} \end{bmatrix} + \cdots + \lambda_m \begin{bmatrix} -a_{m1} \\ \vdots \\ -a_{mn} \end{bmatrix} + \mu_1 \begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \cdots + \mu_n \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix},$$

te $\lambda_i g_i(x) = 0, \ i = 1, ..., n$.
Označimo

$$\lambda := \begin{bmatrix} \lambda_1 & \lambda_2 & \cdots & \lambda_m \end{bmatrix} \quad \text{i} \quad \mu := \begin{bmatrix} \mu_1 & \mu_2 & \cdots & \mu_n \end{bmatrix},$$

pa dobivamo

$$\lambda A = c - \mu \leq c, \text{jer je } \mu \geq 0 \ .$$

Dodatno, iz $\lambda_i g_i(x) = 0$ slijedi

$$\lambda(b - Ax^*) = 0 \ ,$$

a iz $\mu_i x_i^* = 0$

$$0 = \mu x = (c - \lambda A)x \ ,$$

pa po Propoziciji 3.1.5 slijedi da je λ zapravo optimalno rješenje dualne zadaće. Dakle, Lagrangeovi multiplikatori predstavljaju optimalno rješenje dualne zadaće.

3.2 Dokaz dualnosti

Dokazati jednostranu nejednakost $yb \leq cx$ bilo je poprilično jednostavno. Njome možemo brzo i lako provjeriti jesu li vektori x i y optimalna rješenja zadaće linearnog programiranja. U slučaju da jesu, dobivamo jednakost $yb = cx$. Preostalo je dokazati da je jednakost $y^*b = cx^*$ moguća. Dok ne odredimo optimalna rješenja, teorem dualnosti nije gotov.

Kako bismo odredili y^* moramo se vratiti u simpleks metodu, kojom smo ranije odredili x^*. Moramo pokazati da se metoda zaustavila na pravom mjestu dualne zadaće, iako je konstruirana za rješavanje primarne zadaće. Prisjetimo se, uvođenjem poravnavajućih varijabli $w = Ax - b$, m nejednadžbi sustava $Ax \geq b$ pretvorili smo u jednadžbe:

$$\textbf{Primarna zadaća:} \begin{bmatrix} A & -I \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} = b \text{ i } \begin{bmatrix} x \\ w \end{bmatrix} \geq 0.$$

Svaki korak simpleks metode odabire m stupaca matrice $\begin{bmatrix} A & -I \end{bmatrix}$ za bazične varijable i pomiče ih na početak matrice. Time nastaje matrica oblika $\begin{bmatrix} B & N \end{bmatrix}$. Istim pomicanjem, vektor redak funckije cilja $\begin{bmatrix} c & 0 \end{bmatrix}$ pretvara se u vektor redak oblika $\begin{bmatrix} c_B & c_N \end{bmatrix}$. S $r = c_N - c_B B^{-1} N \geq 0$ dan je uvjet zaustavljanja, čime se simpleks metoda privodi kraju. Budući da je broj vrhova dopuštenog područja konačan, uvjet $r \geq 0$ će se u konačno koraka ispuniti. U tom trenutku vrijednost funkcije cilja je minimalna:

$$\textbf{Minimalna vrijednost funkcije cilja: } cx^* = \begin{bmatrix} c_B & c_N \end{bmatrix} \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix} = c_B B^{-1} b \,.$$

Ako u dualnoj zadaći možemo odabrati $y^* = c_B B^{-1}$, onda sigurno vrijedi $y^* b = cx^*$, pa je po Propoziciji 3.1.3 y^* rješenje dualne zadaće. Dakle, potrebno je pokazati da y^* zadovoljava ograničenja dualne zadaće $yA \leq c$ i $y \geq 0$:

$$\textbf{Dualna zadaća: } y \begin{bmatrix} A & -I \end{bmatrix} \leq \begin{bmatrix} c & 0 \end{bmatrix} .$$

Kada simpleks metoda *promiješa* stupce matrice $\begin{bmatrix} A & -I \end{bmatrix}$, kako bi bazične varijable postavila na početak matrice za dualnu zadaću, dobivamo:

$$y \begin{bmatrix} B & N \end{bmatrix} \leq \begin{bmatrix} c_B & c_N \end{bmatrix} .$$

1) Permutaciju prvih $n + m$ stupaca simpleks tablice možemo realizirati množenjem s matricom permutacije s desna, odnosno:

$$\begin{bmatrix} A & -I \\ c & 0 \end{bmatrix} \cdot P_{n+m} = \begin{bmatrix} B & N \\ c_B & c_N \end{bmatrix} .$$

   Iz prethodne jednakosti slijedi:

$$\begin{bmatrix} A & -I \end{bmatrix} \cdot P_{n+m} = \begin{bmatrix} B & N \end{bmatrix} , \begin{bmatrix} c & 0 \end{bmatrix} \cdot P_{n+m} = \begin{bmatrix} c_B & c_N \end{bmatrix} ,$$

   odnosno

$$\begin{gathered} y \begin{bmatrix} A & -I \end{bmatrix} \leq \begin{bmatrix} c & 0 \end{bmatrix} \ / \cdot P_{n+m} \\ y \begin{bmatrix} B & N \end{bmatrix} \leq \begin{bmatrix} c_B & c_N \end{bmatrix} \\ \max yb \end{gathered} \tag{3.2}$$

   što predstavlja ekvivalentnu zadaću početne dualne zadaće.

2) Kada sustav $Ax = b$ prikazanog u simpleks tablici pomnožimo s B^{-1}, bazične varijable će ostati same i dobivamo smanjenu simpleks tablicu:

$$\begin{bmatrix} I & B^{-1}N & B^{-1}b \\ c_B & c_N & 0 \end{bmatrix}.$$

Uvedimo supstituciju $z := yB$. Tada iz (3.2) slijedi da z zadovoljava:

- $z\begin{bmatrix} I & B^{-1}N \end{bmatrix} = yB\begin{bmatrix} I & B^{-1}N \end{bmatrix} = y\begin{bmatrix} B & N \end{bmatrix} \leq \begin{bmatrix} c_B & c_N \end{bmatrix}$,
- $\max yb = \max z(B^{-1}b)$,

gdje je z vektor redak dimenzije m, c_B vektor redak dimenzije m i c_N vektor redak dimenzije n. Posljednja jednakost vrijedi zato što je B regularna matrica, pa za svaki z imamo jedinstveni y i obratno.

3) Ako gornji blok redak smanjene simpleks tablice pomnožimo s c_B i oduzmemo od donjeg blok retka dobivamo potpuno smanjenu simpleks tablicu oblika:

$$\begin{bmatrix} I & B^{-1}N & B^{-1}b \\ 0 & c_N - c_B B^{-1}N & -c_B B^{-1}b \end{bmatrix}$$

Uvedimo supstituciju $\tilde{z} := z - c_B$. Time dobivamo ekvivalentnu zadaću početnoj dualnoj

$$\max \tilde{z}(B^{-1}b) + c_B(B^{-1}b) ,$$

gdje $\tilde{z}$ zadovoljava

$$\tilde{z}\begin{bmatrix} I & B^{-1}N \end{bmatrix} \leq \begin{bmatrix} 0 & c_N - c_B B^{-1}N \end{bmatrix}.$$

Konačno smo dobili da umjesto dualne zadaće

$$\begin{matrix} yA \leq c \\ y \geq 0 \\ \max yb \end{matrix} \iff \begin{matrix} y\begin{bmatrix} A & -I \end{bmatrix} \leq \begin{bmatrix} c & 0 \end{bmatrix} \\ \max yb \end{matrix},$$

koja odgovara početnom koraku simpleks metode za primarnu zadaću, promatramo zadaću

$$\begin{matrix} \max \tilde{z}(B^{-1}b) + c_B(B^{-1}b) , \\ \tilde{z}\begin{bmatrix} I & B^{-1}N \end{bmatrix} \leq \begin{bmatrix} 0 & c_N - c_B B^{-1}N \end{bmatrix}, \end{matrix}$$

koja odgovara simpleks tablici

$$\begin{bmatrix} I & B^{-1}N & B^{-1}b \\ 0 & c_N - c_B B^{-1}N & -c_B B^{-1}b \end{bmatrix}.$$

Dakle, ako je simpleks tablica danas s

$$\begin{bmatrix} A_1 & A_2 & a_3 \\ a_4 & a_5 & -a_6 \end{bmatrix},$$

tada pripadna dualna zadaća glasi

$$\begin{gathered} \tilde{z}\begin{bmatrix} A_1 & A_2 \end{bmatrix} \leq \begin{bmatrix} a_4 & a_5 \end{bmatrix} \\ \max \tilde{z}a_3 + a_6 \ . \end{gathered}$$

Pretpostavimo da je simpleks algoritam stao, odnosno za

$$\begin{bmatrix} I & B^{-1}N & B^{-1}b \\ 0 & c_N - c_B B^{-1}N & -c_B B^{-1}b \end{bmatrix},$$

vrijedi $c_N - c_B B^{-1}N \geq 0$, pa je rješenje primarne zadaća dano s $c_B B^{-1}b$. Uvjerimo se da tu vrijednost zaista postižemo za dualnu zadaću:

1) $\tilde{z} \equiv 0$ je dopustiv, naime $0\begin{bmatrix} I & B^{-1}N \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} \leq \begin{bmatrix} 0 & c_N - c_B B^{-1}N \end{bmatrix}$,

2) $0(B^{-1}b) + c_B(B^{-1}b) = c_B(B^{-1}b)$.

Dakle, maksimum dualne zadaće je zaista $c_B(B^{-1}b)$. Štoviše, može se pokazati da vektor redak $\begin{bmatrix} 0 & c_N - c_B B^{-1}N \end{bmatrix}$ predstavlja vrijednosti optimalnog rješenja dualne zadaće i pripadnih poravnavajućih varijabli. Naime, to slijedi iz činjenice da je simpleks tablica dualne zadaće, nakon zapisivanja u obliku minimizacijske zadaće kao što je prikazano ranije, zapravo jednaka transponiranoj simpleks tablici primarne zadaće sa suprotnim predznakom.

Za $y^* = c_B B^{-1}$ je prva polovica jednakost, a druga polovica je $c_B B^{-1}N \leq c_N$, što vrijedi jer je na kraju simpleks metode ispunjen uvjet zaustavljanja $r = c_N - c_B B^{-1}N \geq 0$. Dakle, vektor y^* je optimalno rješenje dualne zadaće. Prema tome, dualni teorem je dokazan. Pri određivanju bazične matrice B, moramo pripaziti na degenerativnost (vidi Napomenu 2.2.4), kako bi simpleks metoda uspjela odrediti optimalna rješenja x^* i y^*.

3.3 Teorija nejednakosti

Postoji više načina proučavanja dualnosti. Dokazali da vrijedi slaba dualnost $yb \leq cx$, a potom iskoristili simpleks metodu da bismo dobili jednakost. Jednakost smo dokazali konstruktivnim dokazom u kojem smo zapravo izračunali vrijednosti x^* i y^*. Ukratko, promotrimo drugačiji pristup, koji izostavlja algoritam simpleks metode i sagleda s geometrijskog aspekta.

Neka je A realna matrica dimenzija $m \times n$. Ako matricu gledamo kao linearni operator, onda joj je domena $\mathbb{R}^n$, a kodomena $\mathbb{R}^m$. Nadalje, iz linearne algebre znamo da je dimenzija slike ($\dim(ImA)$) jednaka rangu matrice A, dok dimenziju jezgre zovemo defekt. Po definiciji znamo da je rang matrice A jednak rangu matrice A^T (transponirana matrica).

Promotrimo sada dva potprostora: $Ker(A^T)$ i $Im(A)$. Oba su potprostori od $\mathbb{R}^m$. Lako se pokaže da su ortogonalni (koristi se $\langle Ax, y\rangle = \langle x, A^T y\rangle$) . Kako su ortogonalni, imamo da je dimenzija sume ta dva prostora jednaka sumi dimenzija. Sada koristimo Teorem o rangu i defektu da bismo izrazili dimenzije:

$$\dim Ker(A^T) = d(A^T) = \dim(\mathbb{R}^m) - r(A^T) = m - r(A) .$$

Dakle,

$$\dim(Ker(A^T) + Im(A)) = m - r(A) + r(A) = m .$$

Time smo dobili da je $Ker(A^T) + Im(A)$ potprostor prostora $\mathbb{R}^m$ i da je iste dimenzije m, pa su nužno jednaki, odnosno $Ker(A^T) + Im(A) = \mathbb{R}^m$. Dakle, $Ker(A^T)$ je ortogonalni komplement prostora $Im(A)$.

Propozicija 3.3.1. *Sustav $Ax = b$ ima rješenje ili postoji y za koji vrijedi $yA = 0$ i $yb \neq 0$.*

Dokaz. Ako postoji $x \in \mathbb{R}^n$ takav da je $Ax = b$, to znači da se b nalazi u $Im(A)$. Prema gore iskazanom, to znači da je okomit na sve vektore iz prostora $Ker(A^T)$, odnosno za svaki $y \in R^m$, takav da je $A^T y = 0$, vrijedi $\langle y, b\rangle = 0$. (Ako razmišljamo da je y vektor redak, onda gore iskazano možemo zapisati kao $yA = 0$ i $yb = 0$.) Druga mogućnost je da ne postoji $x \in \mathbb{R}^n$ tako da je $Ax = b$, a to onda znači da vrijede negacije gornje tvrdnje, odnosno da postoji $y \in \mathbb{R}^n$ takav da je $A^T y = 0$ i vrijedi $\langle y, b\rangle \neq 0$. (Također, ako razmišljamo da je y vektor redak, onda gore iskazano možemo zapisati kao $yA = 0$ i $yb \neq 0$.) □

Lema 3.3.2. *Sustav $Ax = b$ ima rješenje $x \geq 0$ ili postoji $y \leq 0$ za koji vrijedi $yA \geq 0$ i $yb < 0$.*

Prethodno iskazana lema je varijanta Farkasove leme koja je poopćenje Propozicije 3.3.1 za nenegativna rješenja i za slučaj nejednakosti.

U nastavku ćemo dati alternativni dokaz Teorema dulanosti (Teorem 3.1.1), odnosno dokazati da vrijedi jednakost u $yb \leq cx$ bez korištenja simpleks metode.

Dokaz. Pretpostavimo suprotno. Neka je $\max yb < \min cx$. Definirajmo

$$\alpha := \max yb ,$$

pa imamo $\min cx > \alpha$. Pokazat ćemo da onda slijedi $\max yb > \alpha$, pa dobivamo kontradikciju. Ako je $\min cx > \alpha$, to znači da ne postoji $x \geq 0$ takav da $Ax \geq b$ i $cx \leq \alpha$, odnosno ne postoji rješenje problema:

$$\begin{bmatrix} A \\ -c \end{bmatrix} x \geq \begin{bmatrix} b \\ -\alpha \end{bmatrix},$$
$$x \geq 0\,.$$

Lema 3.3.2 $\Longrightarrow \exists\, y, z$ ($y \in \mathbb{R}^m$ vektor redak, $z \in \mathbb{R}$) takvi da

- $\begin{bmatrix} y & z \end{bmatrix} \leq 0$ ($\Longleftrightarrow y \leq 0$ & $z \leq 0$) ;
- $\begin{bmatrix} y & z \end{bmatrix} \begin{bmatrix} A \\ -c \end{bmatrix} = yA - zc \geq 0$;
- $\begin{bmatrix} y & z \end{bmatrix} \begin{bmatrix} b \\ -\alpha \end{bmatrix} = yb - \alpha z < 0$.

Promotrimo slučaj kada je $z = 0$. Tada postoji y takav da je

$$y \leq 0\,,$$
$$yA \geq 0\,,$$
$$yb < 0\,.$$

Prema Lemi 3.3.2, slijedi da ne postoji $x \geq 0$, rješenje sustava $Ax \geq b$. To je u kontradikciji s početnom pretpostavkom.
Promotrimo slučaj kada je $z < 0$. Tada je

$$yA - zc \geq 0 \ / : (z < 0)$$
$$(\tfrac{1}{z}y)A \leq c$$
$$(\tfrac{1}{z}y)b > \alpha\,.$$

Budući da je $y \leq 0$, slijedi da je $\frac{1}{z}y \geq 0$, odnosno $\frac{1}{z}y$ se nalazi u području rješenja dualne zadaće. Dobivamo:

$$\max yb \geq (\frac{1}{z}y)b > \alpha = \max yb$$

što je kontradikcija. Dakle, vrijedi

$$\max yb = \min cx\,,$$

odnosno minimalna vrijednost cx primarne zadaće jednaka je maksimalnoj vrijednosti yb dualne zadaće. Čime je teorem dokazan. $\square$

Poglavlje 4

Primjeri

4.1 Zabavni park

Primjer 4.1.1. *Stavljeni ste u poziciju upravitelja zabavnog parka koji mora odlučiti koliko zaposlenika zaposliti, a da pritom minimizira troškove. Nažalost, zakon vas ograničava: zaposlenici moraju raditi pet dana zaredom, a po dva dana odmarati. Po tome, imate 7 tipova zaposlenika: tip A se odmara nedjeljom i ponedjeljkom, tip B se odmara ponedjeljkom i utorkom itd. Kako bi zabavni park normalno funkcionirao, ponedjeljkom su potrebna barem 22, utorkom barem 17, srijedom barem 13, četvrtkom barem 14, petkom barem 15, subotom barem 18 i nedjeljom barem 24 zaposlenika. Tjedna plaća za jednog zaposlenika iznosi 1548,00 kuna. [7]*

Rješenje. Uz dane uvjete primjera, moramo odrediti minimalan broj zaposlenika, kako bismo minimizirali troškove plaća. Za početak uvedimo varijable odluke. Ako bismo varijablom x_i označili broj zaposlenika koji rade na i-ti dan, onda ne bismo mogli obuhvatiti uvjet da svaki zaposlenik mora raditi pet dana zaredom. Zato ćemo varijable odluke odabrati drugačije. Neka je x_i broj zaposlenika koji svoj radni tjedan započinju i-ti dan. Sukladno informacijama iz Tablice 4.1 i Tablice 4.2, u kojima su dana ograničenja, i prethodnim razmatranjima, dobivamo sljedeći matematički model zadaće linearnog programiranja:

$$\text{Mimimizirati } x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 \text{ s obzirom na}$$
$$\begin{aligned} x_3 + x_4 + x_5 + x_6 + x_7 &\geq 22\ , \\ x_1 + x_4 + x_5 + x_6 + x_7 &\geq 17\ , \\ x_1 + x_2 + x_5 + x_6 + x_7 &\geq 13\ , \\ x_1 + x_2 + x_3 + x_6 + x_7 &\geq 14\ , \\ x_1 + x_2 + x_3 + x_4 + x_7 &\geq 15\ , \\ x_1 + x_2 + x_3 + x_4 + x_5 &\geq 18\ , \end{aligned}$$

$$x_2 + x_3 + x_4 + x_5 + x_6 \geq 24 \ ,$$
$$x_i \geq 0, \, i = 1, ..., 7.$$

Tip zaposlenika	Ne radi	Pon	Uto	Sri	Čet	Pet	Sub	Ned
Tip A	ned/pon	0	1	1	1	1	1	0
Tip B	pon/uto	0	0	1	1	1	1	1
Tip C	uto/sri	1	0	0	1	1	1	1
Tip D	sri/čet	1	1	0	0	1	1	1
Tip E	čet/pet	1	1	1	0	0	1	1
Tip F	pet/sub	1	1	1	1	0	0	1
Tip G	sub/ned	1	1	1	1	1	0	0

Tablica 4.1: Tablica zadaće linearnog programiranja

Dan	Pon	Uto	Sri	Čet	Pet	Sub	Ned
Potrebno zaposlenika	≥ 22	≥ 17	≥ 13	≥ 14	≥ 15	≥ 18	≥ 24

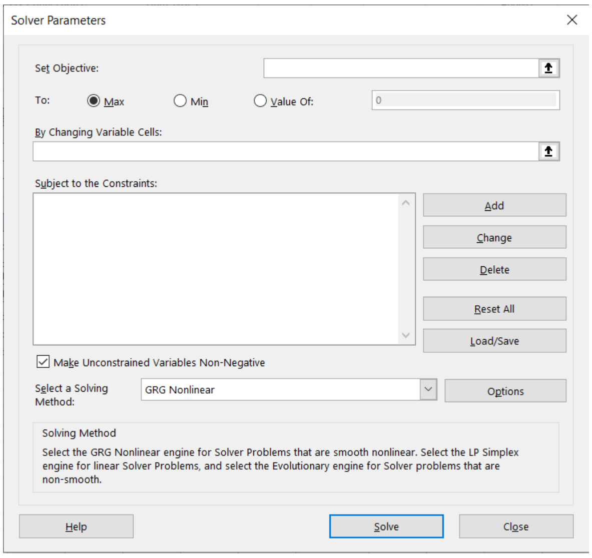
Tablica 4.2: Tablica ograničenja

Zadaću linearnog programiranja riješit ćemo korištenjem programa MS Excel, programa za proračunske tablice. MS Excel sadrži alat *Solver* koji pronalazi optimalna rješenja za gotovo sve zadaće linearnog programiranja. Opiširnije kako koristiti alat pogledajte u [8]. Na početku kreiramo Radni list i u njemu dvije tablice prikazane na Slici 4.1. U jednoj tablici nabrojimo varijable odluke i dane koeficijente uz varijable funkcije cilja. Vrijednosti varijabla $x_1, ..., x_n$ postavimo na nulu. Polje, koje je označeno rozom bojom, predstavlja minimalnu vrijednost zadaće linearnog programiranja, a određena je s vrijednosti funkcije cilja. Na početku, njena vrijednost je nula. U drugoj tablici nabrojimo problemska ograničenja. U rozom obojanom stupcu, funkcijom SUMPRODUCT odredili smo vrijednost ograničenja za zadane vrijednosti varijabla odluke. Na početku poprimaju vrijednost 0. Nakon što smo ispunili tablice, u zaglavlju programa MS Excel odabiremo karticu Data i u njoj izaberemo alat *Solver*. Na Slici 4.2 prikazan je dijaloški okvir *Solver Parameters*. Slijedi njegovo ispunjavanje. U polje *Set Objective* upisuje se adresa ćelije u kojoj je dana funkcija cilja (u našem slučaju I6). U retku *To* odabiremo opciju *Min* budući da želimo minimizirati vrijednost funkcije cilja. Raspon ćelija, koje predstavljaju vrijednosti varijabla odluke, odabiremo u polju *By Changing Variable Cells*. Ograničenja upisujemo u polje *Subject to the Constraints*. Klikom na dugme *Add* otvara se dijaloški okvir *Add Constraint*, prikazan na Slici 4.3. U polje *Cell Reference* dijaloškog okvira upisujemo adresu vrijednosti ograničenja, koja je u drugoj tablici označena rozom bojom. U

	Varijable odluke							
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
	0	0	0	0	0	0	0	Min
c	1	1	1	1	1	1	1	0

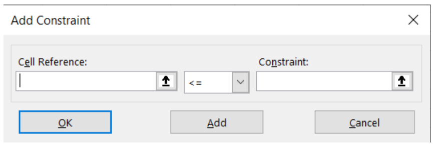
	Ograničenja									
Tip zaposlenika	A	B	C	D	E	F	G		Nejednakost	
Ne radi	ned/pon	pon/uto	uto/sri	sri/čet	čet/pet	pet/sub	sub/ned			
Ponedjeljak	0	0	1	1	1	1	1	0	>=	22
Utorak	1	0	0	1	1	1	1	0	>=	17
Srijeda	1	1	0	0	1	1	1	0	>=	13
Četvrtak	1	1	1	0	0	1	1	0	>=	14
Petak	1	1	1	1	0	0	1	0	>=	15
Subota	1	1	1	1	1	0	0	0	>=	18
Nedjelja	0	1	1	1	1	1	0	0	>=	24

Slika 4.1: Tablica funkcije cilja i tablica ograničenja



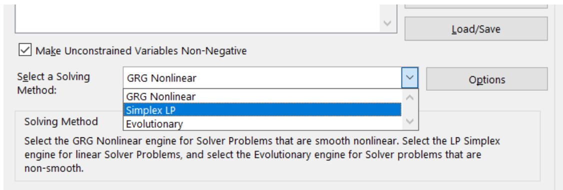
Slika 4.2: Dijaloški okvir *Solver Parameters*

srednjem polju odabiremo znak nejednakosti, a u posljednjem polju *Constraint* upisujemo slobodne koeficijente koji su dani u zadnjem desnom stupcu druge tablice, prikazane na Slici 4.1. Budući da se radi o broju zaposlenika, moramo unijeti jedan dodatan uvjet, a to je da vrijednosti varijabla odluke budu cijeli brojevi. Istaknimo ovdje da se može dogoditi da zadaća nema rješenja u skupu cijelih brojeva, ali da rješenje ipak postoji u skupu realnih brojeva. Nakon što unesemo sva ograničenja, vraćamo se na dijaloški okvir *Solver*



Slika 4.3: Dijaloški okvir *Add Constraint*

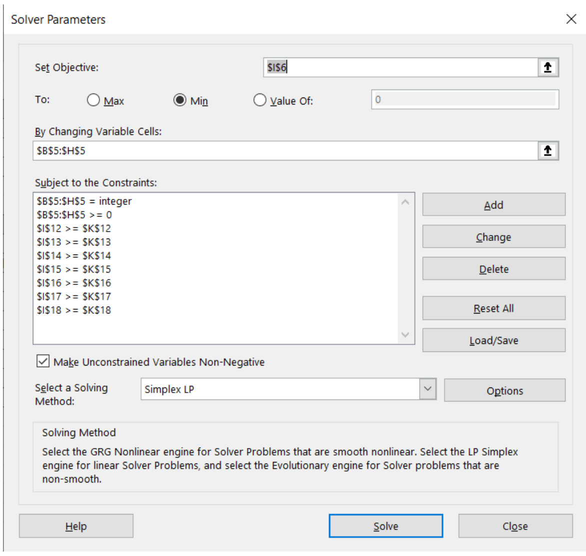
Parameters. Nenegativna ograničenja možemo definirati na dva načina. Upisivanjem u polje *Subject to the Constraints* ili odabirmo *Make Uncostreined Variables Non-Negative*. Kako bismo riješili našu zadaću linearnog programiranja, preostaje nam odrediti jednu od metoda rješavanja, prikazanu na Slici 4.4. U našem slučaju odabiremo *Simplex LP*. Nakon



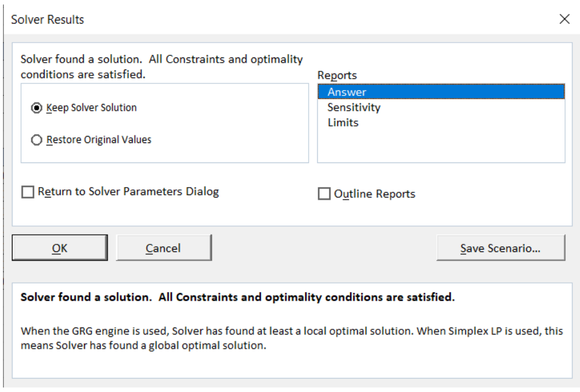
Slika 4.4: Odabir metode rješavanja

što smo ispunili sva potrebna polja dana u dijaloškom okviru *Solver Parameters*, dijaloški okvir poprima sljedeći izgled, prikazan na Slici 4.5. Klikom na dugme *Solve* pokreće se program i otvara novi dijaloški okvir *Solver Results*. U našem slučaju rješenje postoji, stoga možemo odabrati između dviju opcija: da zadržimo postignuto rješenje ili da vratimo početne vrijednosti. U ovom primjeru odabrali su prvu opciju *Keep Solver Solution*. Budući da smo odabrali prvu opciju, u tablicama kreiranima na početku, u polja označena rozom bojom, te u polja koja pokazuju vrijednosti varijabla, unesene su optimalne vrijednosti. Prikazano na Slici 4.7. Ako pogledamo Sliku 4.6, u polju *Reports* smještenom na desnoj strani dijaloškog okvira *Solver Results*, može se odabrati jedan od izvještaja Solvera. U našem slučaju, odabrali smo opciju odgovora *Answer*.

U izvješću, prikazanom na Slici 4.8 podatci su grupirani u tri dijela. U prvom dijelu *Objective Cell* podatci su vezani uz određivanje vrijednosti funkcije cilja. Iz njega možemo iščitati da se određivao minimum te da minimalna vrijednost zadaće linearnog programiranja iznosi 25. U terminima primjera, zaključujemo da upravitelj zabavnog parka mora zaposliti barem 25 zaposlenika kako bi bili zadovoljeni traženi uvjeti. U drugom dijelu



Slika 4.5: Ispunjen dijaloški okvir *Solver Parameters*



Slika 4.6: Dijaloški okvir *Solver Results*

izvješća *Variable Cells* dani su podatci o optimalnim vrijednostima varijabla odluke. Treći dio izvješća *Constraints* sadrži podatke o ograničenjima. Na kraju, preostaje nam odrediti minimalan iznos novca koji upravitelj zabavnog parka mora izdvojiti za svoje zaposlenike na tjednoj bazi. Budući da tjedna plaća za jednog zaposlenika iznosi 1548,00 kn, a prema prethodno riješenoj zadaći linearnog programiranja je određeno da mora zaposliti najmanje 25 zaposlenika, slijedi da je minimalan trošak plaće:

	A	B	C	D	E	F	G	H	I	J	K
1											
2											
3		Varijable odluke									
4		x_1	x_2	x_3	x_4	x_5	x_6	x_7			
5		0	3	5	7	4	6	0	Min		
6	c	1	1	1	1	1	1	1	25		
7											
8											
9		Ograničenja									
10	Tip zaposlenika	A	B	C	D	E	F	G		Nejednakost	
11	Ne radi	ned/pon	pon/uto	uto/sri	sri/čet	čet/pet	pet/sub	sub/ned			
12	Ponedjeljak	0	0	1	1	1	1	1	22	>=	22
13	Utorak	1	0	0	1	1	1	1	17	>=	17
14	Srijeda	1	1	0	0	1	1	1	13	>=	13
15	Četvrtak	1	1	1	0	0	1	1	14	>=	14
16	Petak	1	1	1	1	0	0	1	15	>=	15
17	Subota	1	1	1	1	1	0	0	19	>=	18
18	Nedjelja	0	1	1	1	1	1	0	25	>=	24
19											
20											

Slika 4.7: Tablica funkcije cilja i tablica ograničenja nakon *Solvera*

$$25 \cdot 1548,00 = 38700,00 \text{ kuna.}$$

U ovom primjeru vidjeli smo praktičan način uporabe računalnog programa za rješavanje zadaće linearnog programiranja. Njihova primjena dolazi do izražaja kod zadaća linearnog programiranja zadanih s mnogo varijabla odluke i problemskih ograničenja. Dodatno o uporabi *Solvera* možete pronaći u [1].

Microsoft Excel 16.0 Answer Report
Worksheet: [vjezbe2 (1).xlsx]Sheet1
Report Created: 29-Sep-19 8:22:31 AM
Result: Solver found a solution. All Constraints and optimality conditions are satisfied.
Solver Engine
Engine: Simplex LP
Solution Time: 0.063 Seconds.
Iterations: 1 Subproblems: 2
Solver Options
Max Time Unlimited, Iterations Unlimited, Precision 0.000001
Max Subproblems Unlimited, Max Integer Sols Unlimited, Integer Tolerance 1%, Assume NonNegative

Objective Cell (Min)

Cell	Name	Original Value	Final Value
$I$6	c Min	25	25

Variable Cells

Cell	Name	Original Value	Final Value	Integer
$B$5	x_1	0	0	Integer
$C$5	x_2	3	3	Integer
$D$5	x_3	5	5	Integer
$E$5	x_4	7	7	Integer
$F$5	x_5	4	4	Integer
$G$5	x_6	6	6	Integer
$H$5	x_7	0	0	Integer

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$I$12	Ponedjeljak Mi	22	$I$12>=$K$1	Binding	0
$I$13	Utorak Min	17	$I$13>=$K$1	Binding	0
$I$14	Srijeda Min	13	$I$14>=$K$1	Binding	0
$I$15	Četvrtak Min	14	$I$15>=$K$1	Binding	0
$I$16	Petak Min	15	$I$16>=$K$1	Binding	0
$I$17	Subota Min	19	$I$17>=$K$1	Not Binding	1
$I$18	Nedjelja Min	25	$I$18>=$K$1	Not Binding	1
$B$5	x_1	0	$B$5>=0	Binding	0
$C$5	x_2	3	$C$5>=0	Binding	0
$D$5	x_3	5	$D$5>=0	Not Binding	5
$E$5	x_4	7	$E$5>=0	Not Binding	7
$F$5	x_5	4	$F$5>=0	Not Binding	4
$G$5	x_6	6	$G$5>=0	Not Binding	6
$H$5	x_7	0	$H$5>=0	Binding	0
$B$5:$H$5=Integer					

Slika 4.8: Izvještaj

4.2 Ispitivanje javnog mnijenja

Primjer 4.2.1. *Politolog je dobio sredstva za projekt istraživanja trendova među biračima. Sredstva uključuju* \$3200 *za provedbu kućnih anketa među biračima nekoliko dana uoči izbora. Ankete provode studenti preddiplomskog studija, poslijediplomskog studija i profesori fakulteta. Svaki student preddiplomskog studija provodi anketu na 18 kućanstava i za taj posao dobiva* \$100. *Svaki student poslijediplomskog studija provodi anketu na 25 kućanstava i za taj posao dobiva* \$150. *Svaki profesor provodi anketu na 30 kućanstava i za taj posao dobiva* \$200. *S obzirom na ograničeni vozni park, broj anketara ograničen je na najviše njih 20. Koliko studenata preddiplomskog, odnosno poslijediplomskog studija i profesora treba angažirati da bi se maksimizirao broj anketiranih kućanstava? Koliki je taj broj? [3]*

Rješenje. Sukladno postavljenom pitanju, cilj politologa je maksimiziranje broja anketiranih kućanstava. Kako je broj anketiranih kućanstava s obzirom na anketara (studenti preddiplomskog i diplomskog studija te profesori) različit, politolog mora odlučiti koliko anketara svake skupine odabrati. Iz tog razloga, uvodimo sljedeće varijable odluke:

- x_1 = broj studenata preddiplomskog studija
- x_2 = broj studenata poslijediplomskog studija
- x_3 = broj profesora

Sumiranjem ciljeva i ograničenja, danih u tekstu primjera, odredimo funkciju cilja i problemska ograničenja dane zadaće linearnog programiranja. Promotrimo Tablicu 4.3. Koristeći varijable odluke i informacije iz Tablice 4.3, možemo oblikovati funkciju cilja:

$$C = 18x_1 + 25x_2 + 30x_3.$$

Cilj je pronaći vrijednost varijabla odluke koje daju maksimalnu vrijednost funkcije cilja. Nadalje, utvrdimo problemska ograničenja. Iz teksta problema i Tablice 4.3 možemo iščitati dva važna ograničenja. Ograničenje vezano uz broj anketara i ograničenje vezano uz iznos ankete po kućanstvu.

$$\begin{aligned} x_1 + x_2 + x_3 &\leq 20 \ , \\ 100x_1 + 150x_2 + 200x_3 &\leq 3200 \ . \end{aligned}$$

Budući da ne možemo imati negativan broj anketara, uvodimo nenegativna ograničenja: $x_1 \geq 0$, $x_2 \geq 0$ i $x_3 \geq 0$ te za promatrani problem dobivamo sljedeći matematički model (primarna zadaća):

	Preddipl.	Poslijedipl.	Profesori	Ograničenja
Broj anketara	1	1	1	≤ 20
Broj kućanstava	18	25	30	MAX
Iznos po poslu	\$100	\$150	\$200	$\leq$ \$3200

Tablica 4.3: Tablica zadaće linearnog programiranja.

$$\begin{gathered}
\text{Maksimizirati } C = 18x_1 + 25x_2 + 30x_3 \text{ s obzirom na} \\
x_1 + x_2 + x_3 \leq 20 \ , \\
100x_1 + 150x_2 + 200x_3 \leq 3200 \ , \\
x_1, \ x_2, \ x_3 \geq 0 \ .
\end{gathered}$$

S obzirom na primarnu zadaću, dualna zadaća je dana s:

$$\begin{gathered}
\text{Minimiziraj } C = 20y_1 + 3200y_2 \text{ s obzirom na} \\
y_1 + 100y_2 \geq 18 \ , \\
y_1 + 150y_2 \geq 25 \ , \\
y_1 + 200y_2 \geq 30 \ , \\
y_1, \ y_2 \geq 0 \ .
\end{gathered}$$

Algebarski, primjenom simpleks tablice, riješimo dualnu zadaću linearnog programiranja. Budući da su $n = 2$ i $m = 3$, simpleks tablica T, određena s A, b i c poprimit će dimenziju 4×6. Problemska ograničenja, pomoću poravnavajućih varijabli, pretvorimo u jednadžbe. Slijedi:

$$A = \begin{bmatrix} 1 & 100 & -1 & 0 & 0 \\ 1 & 150 & 0 & -1 & 0 \\ 1 & 200 & 0 & 0 & -1 \end{bmatrix}, \ b = \begin{bmatrix} 18 \\ 25 \\ 30 \end{bmatrix} \text{ i } c = \begin{bmatrix} 20 & 3200 & 0 & 0 & 0 \end{bmatrix} .$$

Uvrštavanjem u simpleks tablicu T, dobivamo:

$$T = \begin{bmatrix} A & b \\ c & 0 \end{bmatrix} = \left[\begin{array}{ccccc:c} 1 & 100 & -1 & 0 & 0 & 18 \\ 1 & 150 & 0 & -1 & 0 & 25 \\ 1 & 200 & 0 & 0 & -1 & 30 \\ \hdashline 20 & 3200 & 0 & 0 & 0 & 0 \end{array} \right] .$$

Simpleks tablica T sadržavat će tri bazične varijable, budući da je dualna zadaća zadana s tri problemska ograničenja. Organizirajmo simpleks tablicu T tako da bazične varijable stavimo ispred nebazičnih. Nakon toga provedimo algoritam (opisan u Poglavlju 2) kako bismo dobili potpuno smanjenu simpleks tablicu.

$$\left[\begin{array}{ccc:cc:c} 1 & 100 & -1 & 0 & 0 & 18 \\ 1 & 150 & 0 & -1 & 0 & 25 \\ 1 & 200 & 0 & 0 & -1 & 30 \\ \hdashline 20 & 3200 & 0 & 0 & 0 & 0 \end{array}\right] \sim \left[\begin{array}{ccc:cc:c} 1 & 100 & -1 & 0 & 0 & 18 \\ 0 & 50 & 1 & -1 & 0 & 7 \\ 0 & 100 & 1 & 0 & -1 & 12 \\ \hdashline 0 & 1200 & 20 & 0 & 0 & -360 \end{array}\right] \sim$$
$$\left[\begin{array}{ccc:cc:c} 1 & 200 & 0 & -1 & 0 & 30 \\ 0 & -50 & 0 & -1 & 1 & -5 \\ 0 & 100 & 1 & 0 & -1 & 12 \\ \hdashline 0 & -800 & 0 & 0 & 20 & -600 \end{array}\right] \sim \left[\begin{array}{ccc:cc:c} 1 & 0 & 0 & -5 & 4 & 10 \\ 0 & -50 & 0 & -1 & 1 & -5 \\ 0 & 0 & 1 & -2 & 1 & 2 \\ \hdashline 0 & 0 & 0 & 16 & 4 & -520 \end{array}\right] \sim$$
$$\left[\begin{array}{ccc:cc:c} 1 & 0 & 0 & -5 & 4 & 10 \\ 0 & 1 & 0 & \frac{1}{50} & -\frac{1}{50} & \frac{1}{10} \\ 0 & 0 & 1 & -2 & 1 & 2 \\ \hdashline 0 & 0 & 0 & 16 & 4 & -520 \end{array}\right].$$

U posljednjoj simpleks tablici slijedi $r = \begin{bmatrix}16 & 4\end{bmatrix}$. Budući da su svi elementi od r pozitivni, simpleks metoda staje. Zaključujemo da je optimalno rješenje dualne zadaće $y_1^* = 10$ i $y_2^* = \frac{1}{10}$. Uočavamo da je samo prva nejednakost u dualnoj zadaći stroga:

$$y_1^* + 100y_2^* = 10 + 10 = 20 > 18.$$

Iz toga, prema Propoziciji 3.1.5, možemo zaključiti da je $x_1^* = 0$. Nadalje, $y_1^* > 0$ i $y_2^* > 0$ pa se obje nejednakosti, opet primjenom Propozicije 3.1.5, u uvjetima primarne zadaće postižu. Dakle, nužno vrijedi

$$\begin{aligned} x_2^* + x_3^* &= 20 \\ 150x_2^* + 200x_3^* &= 3200 \end{aligned} \quad ,$$

a iz tog sustava dobivamo $x_2^* = 16$ i $x_3^* = 4$. Primijetimo da te vrijednosti odgovaraju vrijednostima komponenti vektora r. Dakle, optimalno rješenje primarne zadaće je $x_1^* = 0$, $x_2^* = 16$ i $x_3^* = 4$.

Zaključujemo, kako bismo maksimizirali broj anketiranih kućanstava, moramo angažirati 16 studenata poslijediplomskog studija, 4 profesora i niti jednog studenta preddiplomskog studija. Prema teoremu dualnosti (Teorem 3.1.1) slijedi da je maksimalan broj anketiranih kućanstava jednak vrijednosti funkcije cilja dualne zadaće. Dakle, 520 je maksimalan broj anketiranih kućanstava.

4.3 Raspodjela resursa

Primjer 4.3.1. *Tvrtka proizvodi bicikle s tri brzine, pet brzina i deset brzina. Svaki bicikl prolazi kroz tri pogona, proizvodnje, bojenja i metalizacije, te konačnog sklapanja. Relevantni podatci dani su u Tablici 4.4. Koliko mnogo bicikla svakog tipa tvrtka treba dnevno*

	Trobrzinski	Petbrzinski	Desetbrzinski	Ograničenja
Proizvodnja	3	4	5	≤ 120
Bojenje i metalizacija	5	3	5	≤ 130
Sklapanje	4	3	5	≤ 120
Profit po biciklu ($)	80	70	100	

Tablica 4.4: Tablica zadaće linearnog programiranja.

proizvesti kako bi maksimizirala svoj profit? Koliki je taj profit? [3]

Rješenje. Sukladno postavljenom pitanju, cilj tvrtke je maksimiziranje profita. Kako je profit s obzirom na model bicikla različit, tvrtka mora odlučiti koliko bicikla svake vrste odabrati. Iz tog razloga, uvodimo sljedeće varijable odluke:

- x_1 = broj bicikala s tri brzine,
- x_2 = broj bicikala s pet brzina,
- x_3 = broj bicikala s deset brzina.

Sumiranjem ciljeva i ograničenja, danih u Tablici 4.4, odredimo funkciju cilja i problemska ograničenja dane zadaće linearnog programiranja. Koristeći varijable odluke, možemo oblikovati funkciju cilja:

$$P = 80x_1 + 70x_2 + 100x_3 \ .$$

Cilj je pronaći vrijednost varijabla odluke koje daju maksimalnu vrijednost funkcije cilja. Nadalje, utvrdimo problemska ograničenja. Iz Tablice 4.4 možemo iščitati tri važna ograničenja. Ograničenje vezano uz proizvodnju, ograničenje vezano uz bojenje i metalizaciju te ograničenje vezano uz sklapanje.

$$\begin{aligned} 3x_1 + 4x_2 + 5x_3 &\leq 120 \ , \\ 5x_1 + 3x_2 + 5x_3 &\leq 130 \ , \\ 4x_1 + 3x_2 + 5x_3 &\leq 120 \ . \end{aligned}$$

Budući da ne možemo proizvesti negativan broj bicikala, uvodimo nenegativna ograničenja: $x_1 \geq 0$, $x_2 \geq 0$ i $x_3 \geq 0$, te za promatrani problem raspodjele resursa dobivamo sljedeći matematički model (primarna zadaća):

$$\begin{gathered}\text{Maksimizirati } P = 80x_1 + 70x_2 + 100x_3 \text{ s obzirom na}\\ 3x_1 + 4x_2 + 5x_3 \leq 120\ ,\\ 5x_1 + 3x_2 + 5x_3 \leq 130\ ,\\ 4x_1 + 3x_2 + 5x_3 \leq 120\ ,\\ x_1,\ x_2, x_3 \geq 0\ .\end{gathered}$$

Primjer riješimo promatranjem njegove dualne zadaće. S obzirom na primarnu zadaću, dualna zadaća je dana s:

$$\begin{gathered}\text{Minimizirati } P = 120y_1 + 130y_2 + 120y_3 \text{ s obzirom na}\\ 3y_1 + 5y_2 + 4y_3 \geq 80\ ,\\ 4y_1 + 3y_2 + 3y_3 \geq 70\ ,\\ 5y_1 + 5y_2 + 5y_3 \geq 100\ ,\\ y_1,\ y_2, y_3 \geq 0\ .\end{gathered}$$

Algebarski, primjenom simpleks tablice, riješimo dualnu zadaću linearnog programiranja. Budući da su $n = 3$ i $m = 3$, simpleks tablica T, određena s A, b i c poprimit će dimenziju 4×7. Problemska ograničenja, pomoću poravnavajućih varijabli, pretvorimo u jednadžbe. Slijedi:

$$A = \begin{bmatrix} 3 & 5 & 4 & -1 & 0 & 0 \\ 4 & 3 & 3 & 0 & -1 & 0 \\ 5 & 5 & 5 & 0 & 0 & -1 \end{bmatrix},\ b = \begin{bmatrix} 80 \\ 70 \\ 100 \end{bmatrix} \text{ i } c = \begin{bmatrix} 120 & 130 & 120 & 0 & 0 & 0 \end{bmatrix}.$$

Uvrštavanjem u simpleks tablicu T, dobivamo:

$$T = \begin{bmatrix} A & b \\ c & 0 \end{bmatrix} = \left[\begin{array}{cccccc:c} 3 & 5 & 4 & -1 & 0 & 0 & 80 \\ 4 & 3 & 3 & 0 & -1 & 0 & 70 \\ 5 & 5 & 5 & 0 & 0 & -1 & 100 \\ \hdashline 120 & 130 & 120 & 0 & 0 & 0 & 0 \end{array}\right].$$

Simpleks tablica T sadržavat će tri bazične varijable, budući da je dualna zadaća zadana s tri problemska ograničenja. Organizirajmo simpleks tablicu T tako da bazične varijable stavimo ispred nebazičnih. Nakon toga provedimo algoritam (opisan u Poglavlju 2) kako bismo dobili potpuno smanjenu simpleks tablicu.

$$\left[\begin{array}{ccc:ccc:c} 3 & 5 & 4 & -1 & 0 & 0 & 80 \\ 4 & 3 & 3 & 0 & -1 & 0 & 70 \\ 5 & 5 & 5 & 0 & 0 & -1 & 100 \\ \hdashline 120 & 130 & 120 & 0 & 0 & 0 & 0 \end{array}\right] \sim \left[\begin{array}{ccc:ccc:c} 3 & 5 & 4 & -1 & 0 & 0 & 80 \\ 4 & 3 & 3 & 0 & -1 & 0 & 70 \\ 2 & 0 & 1 & 1 & 0 & -1 & 20 \\ \hdashline 120 & 130 & 120 & 0 & 0 & 0 & 0 \end{array}\right] \sim$$

$$\left[\begin{array}{ccc:ccc:c} -5 & 5 & 0 & -5 & 0 & 4 & 0 \\ -2 & 3 & 0 & -3 & -1 & 3 & 10 \\ 2 & 0 & 1 & 1 & 0 & -1 & 20 \\ \hdashline 120 & 130 & 120 & 0 & 0 & 0 & 0 \end{array}\right] \sim \left[\begin{array}{ccc:ccc:c} 1 & -1 & 0 & 1 & 0 & -\frac{4}{5} & 0 \\ -2 & 3 & 0 & -3 & -1 & 3 & 10 \\ 2 & 0 & 1 & 1 & 0 & -1 & 20 \\ \hdashline 120 & 130 & 120 & 0 & 0 & 0 & 0 \end{array}\right] \sim$$

$$
\left[\begin{array}{ccc|ccc|c}
1 & -1 & 0 & 1 & 0 & -\frac{4}{5} & 0 \\
0 & 1 & 0 & -1 & -1 & \frac{7}{5} & 10 \\
0 & 2 & 1 & -1 & 0 & \frac{3}{5} & 20 \\
\hline
120 & 130 & 120 & 0 & 0 & 0 & 0
\end{array}\right] \sim
\left[\begin{array}{ccc|ccc|c}
1 & 0 & 0 & 0 & -1 & \frac{3}{5} & 10 \\
0 & 1 & 0 & -1 & -1 & \frac{7}{5} & 10 \\
0 & 0 & 1 & 1 & 2 & -\frac{11}{5} & 0 \\
\hline
120 & 130 & 120 & 0 & 0 & 0 & 0
\end{array}\right] \sim
$$

$$
\left[\begin{array}{ccc|ccc|c}
1 & 0 & 0 & 0 & -1 & \frac{3}{5} & 10 \\
0 & 1 & 0 & -1 & -1 & \frac{7}{5} & 10 \\
0 & 0 & 1 & 1 & 2 & -\frac{11}{5} & 0 \\
\hline
0 & 0 & 0 & 10 & 10 & 10 & -2500
\end{array}\right].
$$

U posljednjoj simpleks tablici slijedi $r = \begin{bmatrix} 10 & 10 & 10 \end{bmatrix}$. Budući da su svi elementi od r pozitivni, simpleks metoda staje. Zaključujemo da je optimalno rješenje dualne zadaće $y_1^* = 10$, $y_2^* = 10$ i $y_3^* = 0$. Uočimo da niti jedna nejednakost u dualnoj zadaći nije stroga:

$$
\begin{aligned}
3y_1^* + 5y_2^* + 4y_3^* &= 30 + 50 + 0 = 80 \geq 80 \,, \\
4y_1^* + 3y_2^* + 3y_3^* &= 40 + 30 + 0 = 70 \geq 70 \,, \\
5y_1^* + 5y_2^* + 5y_3^* &= 50 + 50 + 0 = 100 \geq 100 \,.
\end{aligned}
$$

Nadalje, $y_1^* \geq 0$, $y_2^* \geq 0$ i $y_3^* \geq 0$ pa se sve nejednakosti, primjenom Propozicije 3.1.5, u uvjetima primarne zadaće postižu. Dakle, nužno vrijedi

$$
\begin{aligned}
3x_1^* + 4x_2^* + 5x_3^* &= 120 \,, \\
5x_1^* + 3x_2^* + 5x_3^* &= 130 \,, \\
4x_1^* + 3x_2^* + 5x_3^* &= 120 \,,
\end{aligned}
$$

a iz tog sustava dobivamo $x_1^* = 10$, $x_2^* = 10$ i $x_3^* = 10$. Primijetimo da te vrijednosti odgovaraju vrijednostima komponenti vektora r. Dakle, optimalno rješenje primarne zadaće je $x_1^* = 10$, $x_2^* = 10$ i $x_3^* = 10$.

Zaključujemo, kako bismo maksimizirali profit, dnevno moramo proizvesti 10 trobrzinskih bicikala, 10 petbrzinskih bicikala i 10 desetbrzinskih bicikala. Prema teoremu dualnosti (Teorem 3.1.1) slijedi da je maksimalan profit jednak vrijednosti funkcije cilja dualne zadaće. Dakle, 2500 je maksimalan profit.

Bibliografija

[1] B. Plazibat, L. Reić, *Operacijska istraživanja u MS Excelu*, Sveučilište u Splitu, 2015.

[2] G. Strang, *Linear algebra and its applications*, Thomson Brooks/cole, 2006.

[3] K. E. Byleen, R. A. Barnett, M. R. Ziegler, *Primjenjena matematika za poslovanje, ekonomiju, znanost o živom svijetu i humanističke znanosti*, Mate, 2006.

[4] L. Brickmann, *Mathematical introduction to linear programming and game theory*, Springer, 1989.

[5] L. Čaklović, *Geometrija linearnog programiranja*, Element, 2010.

[6] P. R. Thie, *An introduction to linear programming and game theory*, John Whiley And Sons, 1979.

[7] web stranici pristupljeno 4. 9. 2019.
`https://web.math.pmf.unizg.hr/nastava/rp2p/zadaci1819/RP2_2vjezbe.pdf`

[8] web stranici pristupljeno 5. 9. 2019.
`https://www.excel-easy.com/data-analysis/solver.html`

Sažetak

U ovom radu proučavani su osnovni problemi linearnog programiranja. Cilj je bio, za promatranu funkciju cilja, pronaći optimalnu (minimalnu ili maksimalnu) vrijednost na skupu područja rješenja zadanog linearnim nejednakostima. Rješavanju problema linearnog programiranja, pristupili smo na tri načina: geometrijski, algoritamski (simpleks metoda) i algebarski (teorija dualnosti). Optimalnu vrijednost problema linearnog programiranja, kod geometrijske metode, određivali smo na dva načina: korištenjem linije konstantne vrijednosti i provjeravanjem vrijednosti funkcije cilja u vrhovima područja rješenja. Kako geometrijska metoda nije pogodna za rješavanje velikih problema, tada smo problemu pristupili algoritamski, simpleks metodom. Korake simpleks metode organizirali smo simpleks tablicom. Teorem dualnosti je dokazan konstruktivno, koristeći simpleks metodu, te geometrijski pomoću teorije nejednakosti. Sve metode u radu su potkrijepljene primjerima.

Summary

This thesis studies the main problems of linear programming. The aim of the observed cost function was to find the optimal (minimum and maximum) value of the solution area set defined by linear inequalities. We approached the problem of linear programming in three methods: geometric, algorithmic (simplex method) and algebraic (dualty theorem). We determined linear programming optimal value, according to geometric method, in two ways: using constant value line and by verifying the cost function value at the corners of the solution area. As the geometric method is not suitable for solving big problems, we approached the problem in algorithmic mode, using simplex method. Simplex method steps were organized by simplex tableau. The duality theorem is proved constructively, using the simplex method, and geometrically using the theory of inequality. All presented methods in thesis are supported by examples.

Životopis

Antonio Lukačević rođen je 28. lipnja 1994. godine u Sisku. Odrastao je u mjestu Gušće, u srcu Donje Posavine, gdje je pohađao i završio osnovnu školu. U Sisku je 2009. godine upisao Ekonomsku školu, smjer: ekomonist. Tijekom srednjoškolskog obrazovanja prepoznao je zanimanje za matematikom, a znanje nadopunjavao pohađanjem izborne nastave matematike. Svoje obrazovanje nastavio je na Prirodoslovo-matematičkom fakultetu u Zagrebu, gdje je 2013. godine upisao preddiplomski sveučilišni studij Matematike, smjer: nastavnički. Po završetku preddiplomskog studija, na istom fakultetu je 2017. godine upisao diplomski studij Matematike, smjer: nastavnički.