Ime, priimek / Name : Urban Potocnik

04: Linearno programiranje v IKT
Linear programming for ICT

1 Uvod / Introduction

Pri operacijskih raziskavah pogosto naletimo na probleme, ki jih lahko rešujemo s pomočjo linearnega programiranja.

Glavni izziv LP problemov ni proces reševanja, temveč formuliranje problema, kriterijske funkcije ter omejitev.

2 Linearni optimizacijski problem

Imamo naslednji optimizacijski problem:

Najdi realne vrednosti spremenljivk x in y, pri katerih je vrednost optimizacijskega kriterija z maksimalna, z upoštevanjem podanih omejitev.

Maksimize $z = x + 2y$

Subject to:

$2x + y <= 20$

$-4x + 5y <= 10$

$-x + 2y >= -2$

$x >= 0$

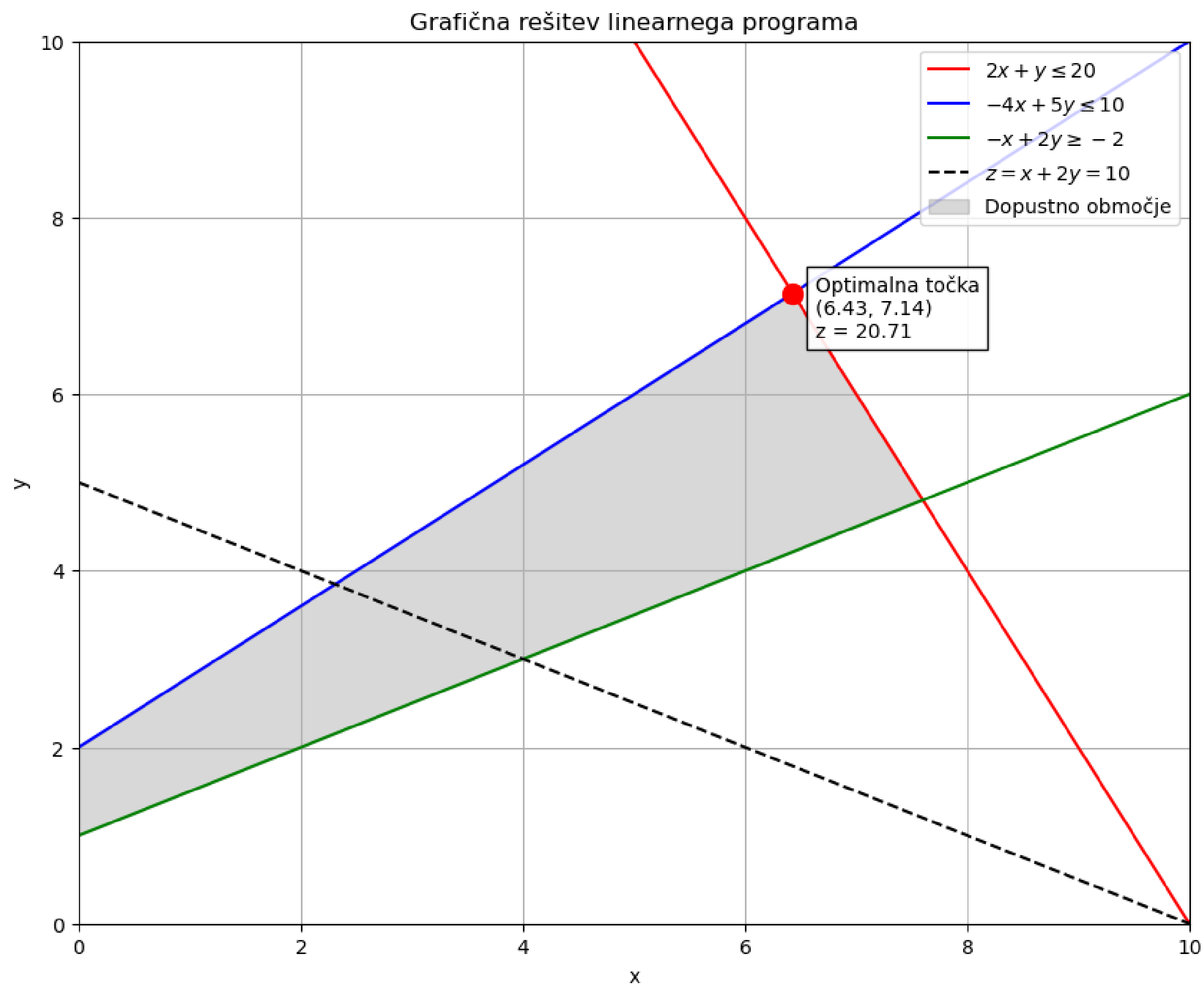
$y >= 0$

2.1 Vizualizacija in grafično reševanje

Izriši omejitve in množico dopustnih rešitev. Izriši premico s konst.vrednostmi kriterija z.

Najdi približne optimalne vrednosti x,y na grafu, ter ugotovi, koliko je takrat kriterij z:

Vstavi graf :

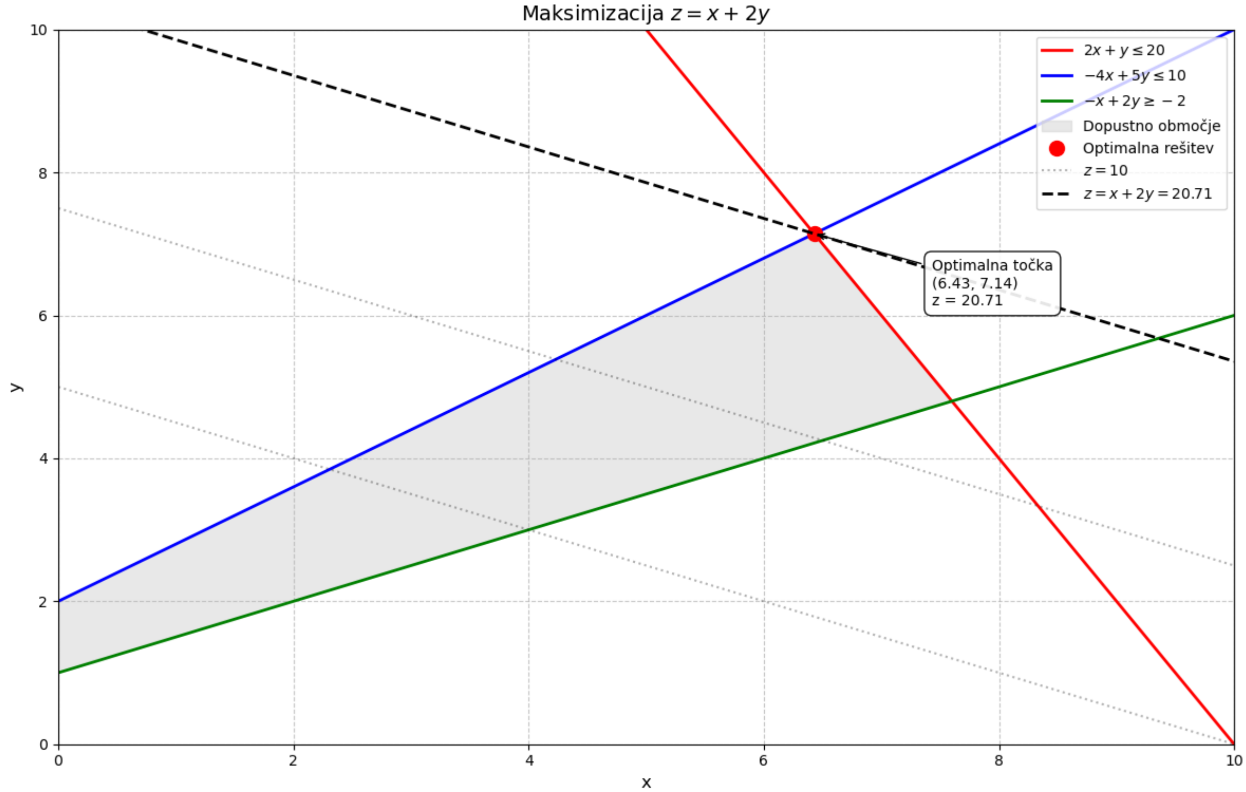


2.2 Reševanje s scipy.optimize.linprog

Vstavi rezultat: Optimalna vrednost: z = 20.7143
Optimalne vrednosti: x = 6.4286, y = 7.1429

Nariši rešitev:

Grafična rešitev linearnega programa
Maksimizacija $z = x + 2y$



2.3 Spremeni kriterij in nariši novo rešitev

Nov kriterij:

Max: z = 5 x + y

Rešitev: Optimalna vrednost: z = 45.2000 Optimalne vrednosti: x = 8.4000, y = 3.2000

3 Optimizacija z LPSolve

LP Solve rešuje probleme linearnega programiranja. V program zapišemo kriterijsko funkcjo (kateri določimo min / max), omejitve ter po potrebi še definiramo tip naših spremenljivk (bin / int). Program nato sam izračuna vse možne rešitve in jih prikaže.

https://sourceforge.net/projects/lpsolve/

3.1 Problem optimalnega števila naročnikov

Zapiši problem v LP Solve in poišči rešitev.

Opis problema 1:

Nov mobilni operater se pripravlja za vstop na tržišče. Ker še nima svoje infrastrukture oddajnikov, se je z že obstoječim operaterjem dogovoril za najem njegovega omrežja. Zakupil je mesečno kapaciteto 1000 SMS sporočil, ter 2000 minut pogovora.

Svojim naročnikom bo operater ponudil paket za fizične osebe z mesečno naročnino 20€/mesec ter paket za pravne osebe in podjetja z mesečno naročnino 40€/mesec. Predhodne tržne raziskave so pokazale, da fizične osebe v povprečju pošljejo do 40 SMS sporočil in opravijo za 20 minut pogovorov, pravne pa 10 SMS sporočil ter 50 minut pogovorov.

Koliko katerih uporabnikov mora pridobiti operater, da bo optimalno izkoristil zakupljene kapacitete in hkrati imel maksimalni zaslužek?

```
### Reševanje z LPSolve
> How to use LPSolve for solving this problem.

Spremenljivki, ki jih moramo določiti, sta naslednji  / We have the following variables:
* število naročnikov, fizičnih oseb /Number of subscribers, persons :  n_f
* število naročnikov pravnih oseb /Number of subscribers, business users:  n_p. Oboje je celo
število (integer).

Kriterijska funkcija je maksimizacija zaslužka od prodaje paketov, ki jo dobimo iz cene in
prodanih paketov. V LPSolve
to zapišemo na naslednji način:

> We optimize criterion, which is maximum revenue from selling subscriptions, given that the
cost is 20 for persons and 40 for companies.

<code>/* Objective function */ <br>
max: + 20 n_f + 40 n_p;  </code>

Podati moramo še omejitve spremenljivk glede na število zakupljenih enot (1000 SMS, 2000
minut), s pomočjo povprečne porabe
enot pri posameznem tipu paketov. To zapišemo z:
> We need to define bounds (constraints) according to the number of leased resources (1000 SMS,
2000 minutes)

<code> /* Variable bounds */ <br>
  40 n_f + 10 n_p <= 1000;
  20 n_f + 50 n_p <= 2000; </code>

Podamo še, da so iskane spremenljivke celoštevilske:
> The variables (numbers of packages) are integers:

<code> int n_f, n_p; </code>
```

3.1.1 Rezultat optimizacije:

Status: Optimal

Optimal number of personal packages: 21.0

Optimal number of business packages: 15.0

Maximum revenue: 1020.0

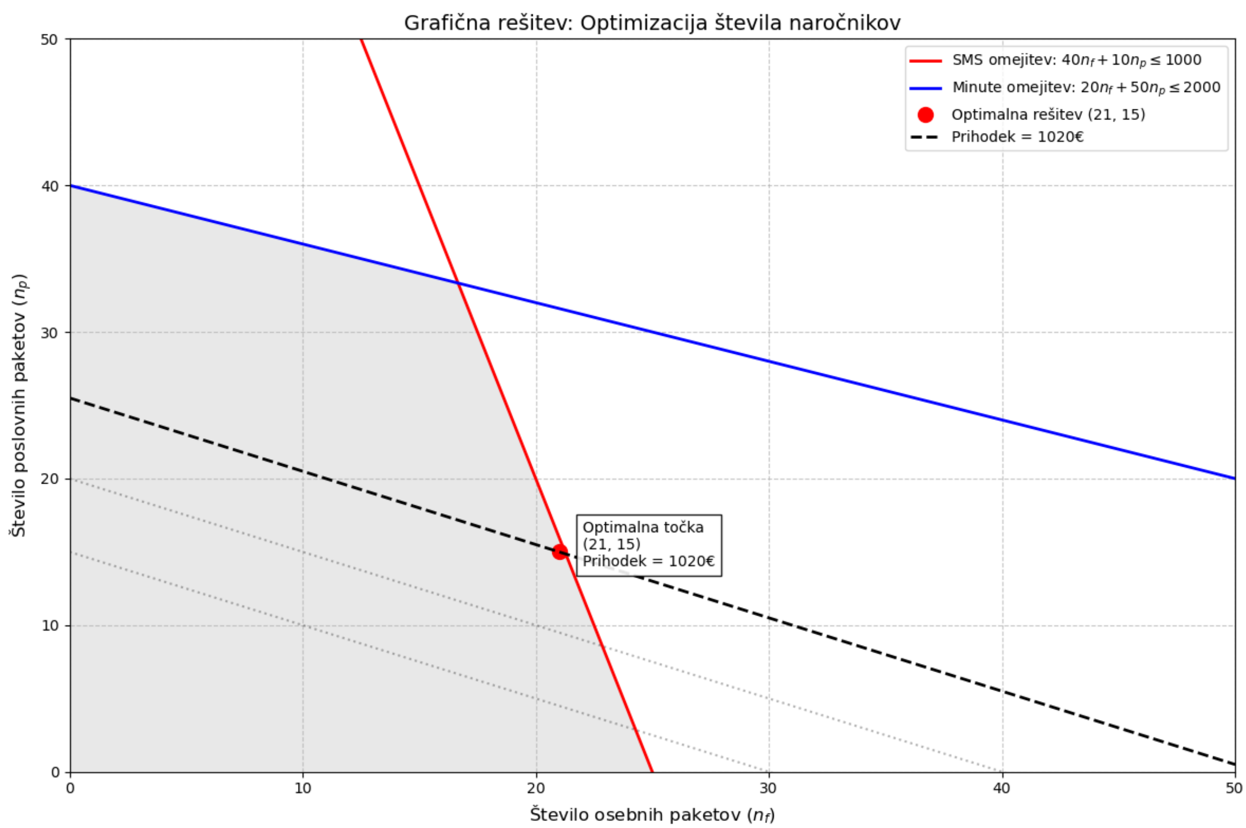
Resource usage:

SMS used: 990/1000 (10 remaining)

Minutes used: 1920/2000 (80 remaining)

3.1.2 Grafično reševanje problema (dodatna naloga)

Vizualiziraj problem v python matplotlib.



3.2 Ohlapne spremenljivke

Z uporabo ohlapnih spremenljivk spremenimo omejitve problema iz neenačb v enačbe. To sicer pomeni, da uvedemo toliko dodatnih spremenljik, kot imamo omejitev, vendar v zameno zato dobimo pri rešitvi še en dodaten podatek, koliko nam še manjka do dejanske meje.

V primeru prvega problema tako nazaj dobimo podatek koliko SMS sporočil in minut pogovora ostane ostane neizkoriščenih.

Spreminjaj ceno paketa za fizične in pravne osebe, ter preveri zaslužek, število prodanih paketov, ter preostanek sms in minut

Rezultat:

	Primer 1	2	3	4
Cena fizične	20	25	15	30
Cena pravne osebe	40	35	45	30
Zaslužek	1020	1050	990	900
Paketov fizične	21	25	16	30
Paketov pravne	15	10	18	0
Preostanek SMS	10	0	20	0
Preostanek minut	80	50	20	1400

3.3 Modifikacija problema

Dodamo dve spremenljivki N_sms, N_min : povesta količino zakupljenega prometa operaterja operaterja (število paketov 10 sms, paketov 10 minut). Cena paketa SMS je 1 Eur, in cena paketa minut je 3 eur.

Operater za zakup minut in sms ne želi porabiti več kot 1000 Eur.

Prilagodite problem, zapis problema:

V osnovnem problemu je operater najemal fiksne količine SMS-ov (1000) in minut (2000) ter optimiziral število prodanih naročnin (osebnih in poslovnih) za maksimiranje prihodkov. V modificirani različici problema operater ne najema neposredno določenega števila SMS-ov in minut, temveč kupi pakete, kjer:

1 paket SMS = 10 SMS-ov (cena: 1 €),

1 paket minut = 10 minut (cena: 3 €).

Operater ima omejen budget 1000 € za nakup paketov in želi maksimizirati dobiček, torej razliko med prihodki od prodaje naročnin in stroški nakupa virov.

Opišite rešitev, koliko bi operater zaslužil v tem primeru, koliko bi zakupil minut in sms, in kakšne naročnine bi prodal. Lahko spreminjate tudi ceno paketov.

Optimalna rešitev pri osnovnih cenah (20€ osebni, 40€ poslovni)

Kupljeni viri:

SMS: 100 paketov (1000 SMS) za 100 €

Minute: 300 paketov (3000 minut) za 900 €

Skupni stroški: 1000 € (porabljen celoten budget)

Prodane naročnine:

25 osebnih paketov (25 × 20€ = 500 €)

0 poslovnih paketov

Skupni prihodek: 500 €

Dobiček:

500€ (prihodki)−1000€ (strosˇki)=−500€(izguba)500€ (prihodki)−1000€ (strosˇki)=−500€ (izguba)

Poraba virov:

SMS: 1000/1000 (100% izkoriščenost)

Minute: 500/3000 (samo 17% izkoriščenost)

3.4 Iskanje najcenejše poti v grafu

Definirajte kriterijsko funkcijo in potrebne omejitve za izračun najkrajše poti od izvora do ponora. V LP Solve preizkusite

problem ILP (cela števila oziroma binarna števila) problem.

```
Zapis problema / Problem statement:

Iščemo najkrajšo pot iz vozlišča A v vozlišče H, kjer števila na povezavah predstavljajo
dolžino poti. Vrednost Vsake spremenljivke bo pomenila, ali je povezava med vozliščema
vključena v najkrajšo pot ali ne, torej bodo spremenljivke binarne. Definiramo jih torej kot:

<code>bin ab, ac .... ;</code>

Kriterijska funkcija je minimum vsote dolžin vključenih povezav, zapisati moramo vsoto vseh
možnih povezav na način:

<code>min: + 42 ab + 7 ac + 18 ad + ......; </code>

Omejitve spremenljivk zapišemo za vsako vozlišče. Za vozlišče A velja, da mora iz njega voditi
natanko 1 pot, torej:

<code> +ab + ac + ad = 1;</code>

Za druga vozlišča pa velja, da zapišemo + in - glede na usmeritve puščic (poti), ter mora biti
vsota 0 (po eni povezavi pridemo, po drugi gremo).
Pri zadnjem pa je vsota -1.

<code> -ab -bc -bf + bh = 0; </code>
```

3.4.1 Poišči zaporedje vozlišč z najkrajšo (najcenejšo) potjo od A do H

Zaporedje vozlišč v najkrajši poti: A -> D -> E -> G -> F -> B -> H

3.4.2 Poišči najdaljšo pot od A do I

Spremeni formulacijo, tako da poiščeš najdaljšo pot od A do I.

A -> B -> F -> B -> H -> I

Skupna dolžina: 100 enot