

VILNIAUS UNIVERSITETAS  
MATEMATIKOS IR INFORMATIKOS FAKULTETAS

Laboratorinis darbas 4

Tiesinis programavimas

Nikita Gainulin

VILNIUS 2024

# Turinys

<b>1</b>	<b>Įvadas</b>	<b>2</b>
<b>2</b>	<b>Nagrinėjamas uždavinys</b>	<b>2</b>
<b>3</b>	<b>Tiesinis optimizavimas ir uždavinio sprendimo algoritmas</b>	<b>2</b>
3.1	Tiesinis optimizavimas . . . . .	2
3.2	Simplekso metodas . . . . .	2
3.3	Algoritmo kodas . . . . .	4
<b>4</b>	<b>Rezultatai ir jų analizė</b>	<b>4</b>
4.1	Minimumo taškai . . . . .	4
4.2	Funkcijos reikšmės . . . . .	4
4.3	Greitis . . . . .	4
4.4	Efektyvumas . . . . .	4
<b>5</b>	<b>Išvada</b>	<b>4</b>
<b>6</b>	<b>Priedas</b>	<b>4</b>

## 1 Įvadas

Savo ankstesniame laboratoriniame darbe gilinausi į netiesinį programavimą. Kaip nustačiau ankstesniame laboratoriniame darbe, netiesinio optimizavimo metodų tikslas - optimizuoti uždavinį, kurio tikslo funkcija arba vienas iš apribojimų yra netiesinio tipo. Šiame laboratoriniame darbe gilinsiuosi į tiesinį optimizavimą, kur visos modelio funkcijos bus tiesinės.

## 2 Nagrinėjamas uždavinys

Šiame laboratoriniame darbe apskaičiuosime mažiausią tikslo funkcijos reikšmę, optimalų sprendinį ir bazę šioms dviem sistemoms:

$$\begin{aligned} \min & 2x_1 - 3x_2 - 5x_4 \\ -x_1 + x_2 - x_3 - x_4 & \leq 8 \\ 2x_1 + 4x_2 & \leq 10 \\ x_3 + x_4 & \leq 3 \\ x_i & \geq 0 \end{aligned} \tag{1}$$

$$\begin{aligned} \min & 2x_1 - 3x_2 - 5x_4 \\ -x_1 + x_2 - x_3 - x_4 & \leq 1 \\ 2x_1 + 4x_2 & \leq 5 \\ x_3 + x_4 & \leq 7 \\ x_i & \geq 0 \end{aligned} \tag{2}$$

## 3 Tiesinis optimizavimas ir uždavinio sprendimo algoritmas

### 3.1 Tiesinis optimizavimas

Kaip jau buvo matyti iš pateikto uždavinio, tiesinio optimizavimo metodas puikiai tinka jam spręsti ir reikiamiems atsakymams pateikti. Visi apribojimai ir pati tikslo funkcija yra tiesiniai. Kalbant apie minėtą tiesinį optimizavimo metodą, šiame laboratoriniame darbe naudosiu **simplekso metodą**.

### 3.2 Simplekso metodas

Paprastai, išgirdę sąvoką „simplekso metodas“, galime pagalvoti apie mums jau pažįstamą Nelderio-Medo metodą, kuris veikia su mažų matmenų figūromis. Tačiau, kaip jau žinome, šis metodas netinka nei tiesiniams, nei netiesiniams optimizavimo uždaviniams spręsti, nes jis neturi galimybės savarankiškai tvarkytis su apribojimais. Netiesiniam optimizavimui jis yra svarbi kito, baudų metodo, dalis. Tačiau tiesinio optimizavimo atveju simplekso metodas taikomas visiškai kitaip ir niekuo nepanašus į tradicinį Nelderio-Medo metodą.

Simplekso metodą pademonstruosiu optimizuodamas 1-ąją sistemą (1). Pirmasis simplekso metodo žingsnis yra paversti mūsų sistemos apribojimus iš nelygybių į lygybes įvedant vadinamuosius laisvuosius kintamuosius. Tokių kintamųjų kiekį lemia tai, kiek nelygybių yra sistemoje (išskyrus teigiamų reikšmių apribojimą). Mūsų atveju jų yra 3, todėl įvedame šiuos laisvuosius kintamuosius:

$s_1, s_2$  ir  $s_3$ . Kitas žingsnis - padauginti mūsų tikslo funkciją iš -1, nes pagal numatytuosius nustatymus simplekso lentelė dirba su maksimizavimu, todėl turime pakeisti indeksus, kad išlaikytume minimizavimą. Taigi, mūsų naujoji sistema su lygybėmis atrodo taip:

$$\begin{aligned} \max & -2x_1 + 3x_2 + 5x_4 \\ -x_1 + x_2 - x_3 - x_4 + s_1 &= 8 \\ 2x_1 + 4x_2 + s_2 &= 10 \\ x_3 + x_4 + s_3 &= 3 \\ x_i, s_i &\geq 0 \end{aligned} \quad (3)$$

Po pirmiau minėtų transformacijų naująją sistemą reikia perrašyti į standartinę matricinę formą  $AX = B$ , kur  $A$  - apribojimų indeksų matrica,  $X$  - nežinomų ir laisvųjų kintamųjų matrica,  $B$  - dešinėsios pusės reikšmių matrica. Štai kaip atrodo mūsų naujosios sistemos standartinė forma:

$$\underbrace{\begin{pmatrix} x_1 & x_2 & x_3 & x_4 & s_1 & s_2 & s_3 \\ -1 & 1 & -1 & -1 & 1 & 0 & 0 \\ 2 & 4 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}}_A \underbrace{\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ s_1 \\ s_2 \\ s_3 \end{pmatrix}}_X = \underbrace{\begin{pmatrix} 8 \\ 10 \\ 3 \end{pmatrix}}_B$$

Dabar galime sukurti pradinę simplekso lentelę:

		$C_j$	-2	3	0	5	0	0	0
$C_b$	$B$	$X_b$	$x_1$	$x_2$	$x_3$	$x_4$	$s_1$	$s_2$	$s_3$
0	$s_1$	8	-1	1	-1	-1	1	0	0
0	$s_2$	10	2	4	0	0	0	1	0
0	$s_3$	13	0	0	1	1	0	0	1
$Z_j - C_j$			2	-3	0	-5	0	0	0

1 lentelė: Pradinė simplekso lentelė, kur  $C_j$  - maksimizavimo tikslo funkcijos indeksai,  $B$  - bazinės reikšmės,  $C_b$  - bazinių reikšmių indeksai maksimizavimo tikslo funkcijoje,  $X_b$  - atitinka standartinės formos  $B$  matricai

Kaip pastebėjote, turime nepažįstamą eilutę  $Z_j - C_j$ . Šioje eilutėje nustatomas pagrindinis stulpelis ir ji bus naudinga vėliau, kai pradėsime skaičiuoti optimalią vertę. Kol kas pakanka atkreipti dėmesį į šią formulę, pagal kurią apskaičiuojama  $Z_j - C_j$ :

$$Z_j - C_j \equiv C_b X_j - C_j \quad (4)$$

Kadangi iš pradžių  $C_b$  yra 0, visa apatinė  $Z_j - C_j$  eilutė yra lygi  $-C_j$ . Toliau šioje eilutėje mums reikia rasti mažiausią neigiamą reikšmę, šiuo atveju -5. Tai yra pagrindinis stulpelis  $x_4$ . Dabar mums reikia pasirinkti pagrindinę eilutę. Tai atliekama visas  $X_b$  reikšmes dalijant iš pagrindinės eilutės reikšmių atitinkamai. Gauname štai tokias reikšmes:

$$\frac{8}{-1} = -8$$

$$\frac{10}{0} = \infty$$

$$\frac{13}{1} = 13$$

Iš apskaičiuotų santykinų dydžių turime pasirinkti tą, kuris duoda mažiausią teigiamą reikšmę, kuri mūsų atveju yra 13. Taigi, kai mes gavome pagrindinius stulpelį ir eilutę, galime pradėti konstruoti sekančią simplekso lentelę:

		$C_j$	-2	3	0	5	0	0	0
$C_b$	$B$	$X_b$	$x_1$	$x_2$	$x_3$	$x_4$	$s_1$	$s_2$	$s_3$
0	$s_1$								
0	$s_2$								
0	$s_3$								
$Z_j - C_j$									

Tikslo funkcijos indeksai  $C_j$  eilutėje nesikeis, todėl galime drąsiai juos perrašyti.

### 3.3 Algoritmo kodas

## 4 Rezultatai ir jų analizė

### 4.1 Minimumo taškai

### 4.2 Funkcijos reikšmės

### 4.3 Greitis

### 4.4 Efektyvumas

## 5 Išvada

## 6 Priedas