Zeki Optimizasyon Teknikleri
Intelligent Optimization Techniques

Hazırlayan: M. Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Bu dersin sunumları, "Ron Larson, David C.Falvo, Elementary Linear Algebra, Houghton Mifflin Harcourt Publishing Company, 2009." kitabı kullanılarak hazırlanmıştır.

İçerik

- Doğrusal programlama problemi
- Geometrik gösterim
- Simpleks algoritması

Doğrusal programlama problemi

- Doğrusal programlamada (DP), **amaç fonksiyonu** ve **kısıtlar** karar değişkenlerinin **doğrusal fonksiyonudur**.
- DP, ekonomistler tarafından 1930'larda optimizasyon problemlerinin çözümünde kullanılmıştır.
- George B. Dantzig 1947 yılında DP problemini formüle etti ve **simpleks çözüm yöntemini geliştirdi.**
- Kuhn & Tucker, DP'ya önemli teorik katkılar sağlamıştır.
- Charnes & Cooper endüstriyel uygulamalar için çalışmalar yapmıştır.
- DP, **karmaşık durumlarda optimal karar** vermemizi sağlayan önemli bir gelişme olarak kabul edilir.
- DP problemlerinin çözümünde **simpleks algoritması en etkin ve popüler yöntemdir.**

Doğrusal programlama problemi

Scalar form

- Genel doğrusal programlama problemi aşağıdaki gibi ifade edilir:

$$\text{Minimize } f(x_1, x_2, \ldots, x_n) = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\
&\ \vdots \\
a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m
\end{aligned}$$

$$\begin{aligned}
x_1 &\geq 0 \\
x_2 &\geq 0 \\
&\ \vdots \\
x_n &\geq 0
\end{aligned}$$

c_j, b_j, a_{ij} **katsayılar** ve x_j **karar** değişkenleridir.

Doğrusal programlama problemi

Matris formu

$$\text{Minimize } f(\mathbf{X}) = \mathbf{c}^T\mathbf{X}$$
$$\mathbf{aX} = \mathbf{b}$$
$$\mathbf{X} \geq \mathbf{0}$$

$$\mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix}, \quad \mathbf{b} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{Bmatrix}, \quad \mathbf{c} = \begin{Bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{Bmatrix}$$

$$\mathbf{a} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

- Bir DP probleminin karakteristik özellikleri:
  - **Amaç fonksiyonu** minimize/maksimize edilecektir.
  - Tüm **kısıtlar** eşitlik şeklindedir.
  - Tüm **karar değişkenleri** pozitiftir.

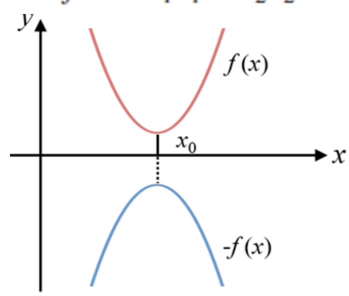
Doğrusal programlama problemi

- Herhangi bir DP problemi aşağıdaki dönüşümler yapılarak standart formda yazılabilir:

1- Bir $f(x)$ fonksiyonunu maksimize etmek, $-f(x)$ fonksiyonunu minimize etmeye eşittir.

- Bir $\boldsymbol{x_0}$ noktası, $\boldsymbol{f(x)}$ fonksiyonunun minimum değeri ise, aynı nokta $-\boldsymbol{f(x)}$ fonksiyonunun maksimum değeridir.

$$\text{minimize } f = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$
$$\text{maximize } -f = -c_1x_1 - c_2x_2 - \cdots - c_nx_n$$



Doğrusal programlama problemi

2- Bir değişken pozitif veya negatif değer alabilir ve iki pozitif değişkenin farkı ile gösterilebilir.

- Bir x_j değişkeni sınırsızdır, x_{j1} ve x_{j2} değişkenleri pozitiftir.

$$x_j = x_{j1} - x_{j2}$$

$$x_{j1} \geq 0 \text{ ve } x_{j2} \geq 0$$

3- Bir kısıt "küçük eşit" veya "büyük eşit" şeklinde ise, **değişken eklenerek** (slack/surplus variable) **eşit ile gösterilebilir.**

$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n \leq b_k$$
$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n + x_{n+1} = b_k$$

$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n \geq b_k$$
$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n - x_{n+1} = b_k$$

İçerik

- Doğrusal programlama problemi
- Geometrik gösterim
- Simpleks algoritması

Geometrik gösterim

- İki değişkenli DP problemi basit **grafik metotla çözülebilir.**
- **Geometrik gösterim DP probleminin karakteristiğini yansıtır.**

Örnek:

- **3 farklı makine** ile **2 farklı parça** üretiliyor. Her makinenin haftalık maksimum kullanım süresi, her makinede parçaların işlem süreleri, her parçanın kar miktarı tabloda verilmiştir.
- Haftalık **karı maksimize etmek için üretilecek parça sayılarını bulalım.**

Kullanılan makine	**Gerekli işlem süresi (dk)**		**Makinenin haftalık kullanım süresi (dk)**
	Parça 1	**Parça 2**	
Makine 1	10	5	2500
Makine 2	4	10	2000
Makine 3	1	1,5	450
Parçanın kar miktarı (TL)	**50**	**100**	

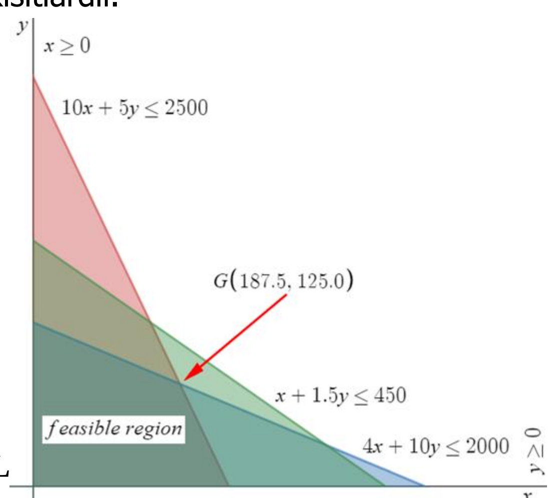
Geometrik gösterim

Kullanılan makine	Gerekli işlem süresi (dk)		Makinenin haftalık kullanım süresi (dk)
	Parça 1	Parça 2	
Makine 1	10	5	2500
Makine 2	4	10	2000
Makine 3	1	1,5	450
Parçanın kar miktarı (TL)	50	100	

Örnek - devam

- Haftalık üretilen parça sayıları x (Parça 1) ve y (Parça 2) olsun.
- Maksimum süre sınırları kısıtlardır.

$$10x + 5y \leq 2500$$
$$4x + 10y \leq 2000$$
$$x + 1.5y \leq 450$$
$$x \geq 0$$
$$y \geq 0$$



- Toplam kar fonksiyonu

$f(x, y) = 50x + 100y$

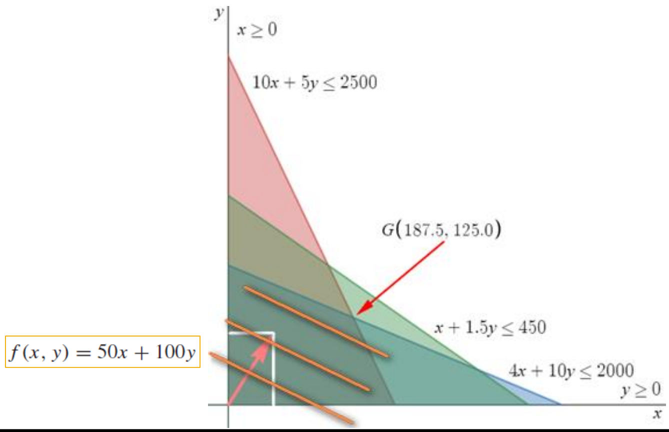
$\boldsymbol{x^*} = 187.5$, $\boldsymbol{y^*} = 125.0$

$f(187.5, 125.0) =$ **21875 TL**

Geometrik gösterim

Örnek - devam

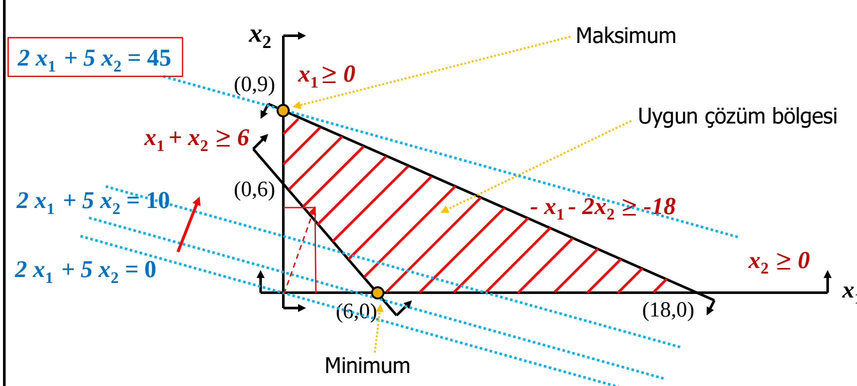
- **Optimum kar için amaç fonksiyonu kullanılabilir.**
- **Amaç fonksiyonu**, katsayı vektörü yönünde **hareket ettirilir.**
- Feasible bölgeye **ilk girdiği nokta minimum**, **son çıktığı nokta maksimum** noktadır.



Geometrik gösterim

Örnek:

$$
\begin{aligned}
&\textit{Max} && 2x_1 + 5x_2 \\
&\textit{Kısıtlar} && x_1 + x_2 \geq 6 \\
& && -x_1 - 2x_2 \geq -18 \\
& && x_1,\ x_2 \geq 0
\end{aligned}
$$



Geometrik gösterim

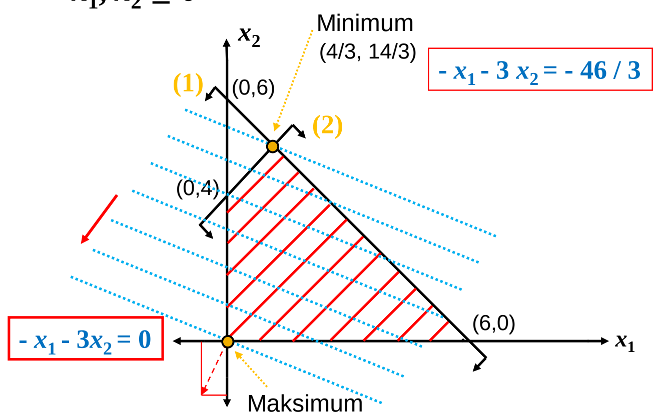
Örnek:

Min $- x_1 - 3x_2$

Kısıtlar $x_1 + x_2 \leq 6$ (1)

$- x_1 + 2x_2 \leq 8$ (2)

$x_1, x_2 \geq 0$



Geometrik gösterim

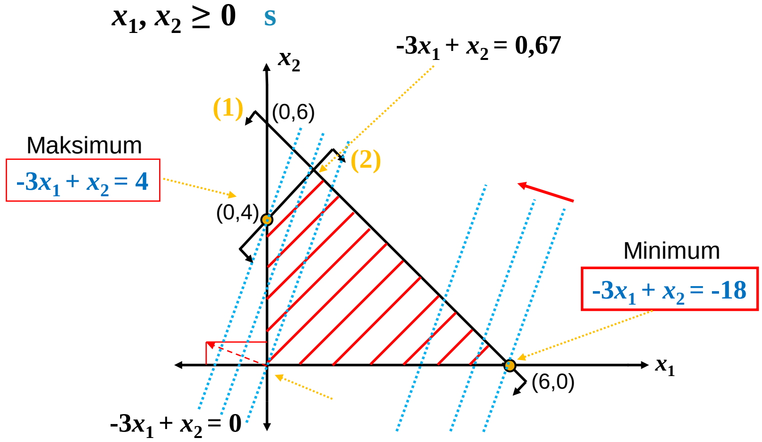
Örnek:

Min $- 3x_1 + x_2$

Kısıtlar $x_1 + x_2 \leq 6$ (1)

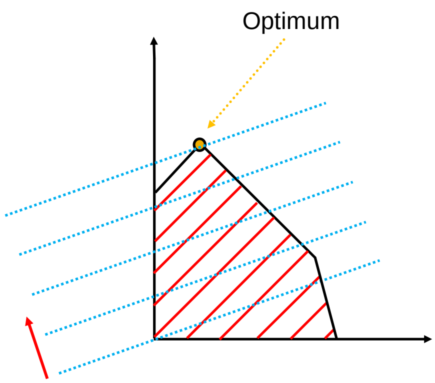
$- x_1 + 2x_2 \leq 8$ (2)

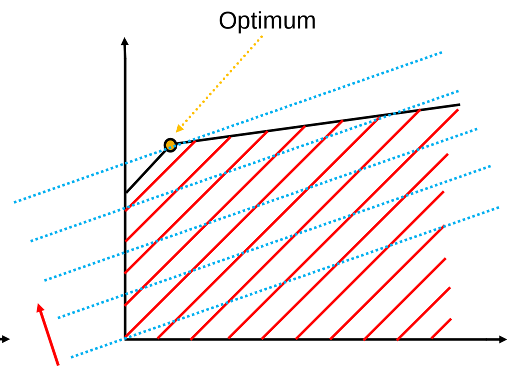
$x_1, x_2 \geq 0$ s



Geometrik gösterim

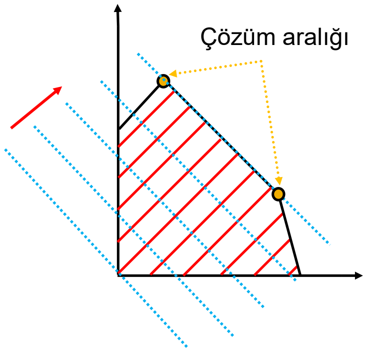
- Uygun çözüm bölgesinde **optimum çözüm bir tane** olabilir.

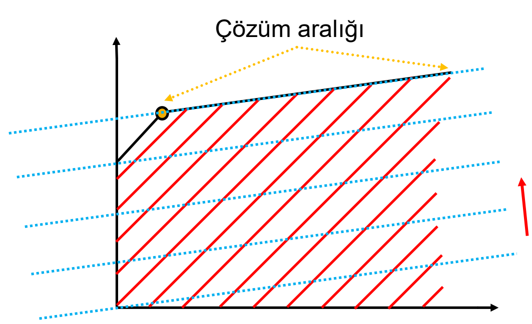




Geometrik gösterim

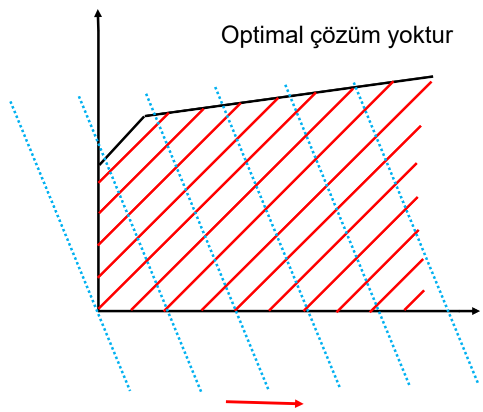
- Uygun çözüm bölgesinde **optimum çözüm sonsuz adet** olabilir.





Geometrik gösterim

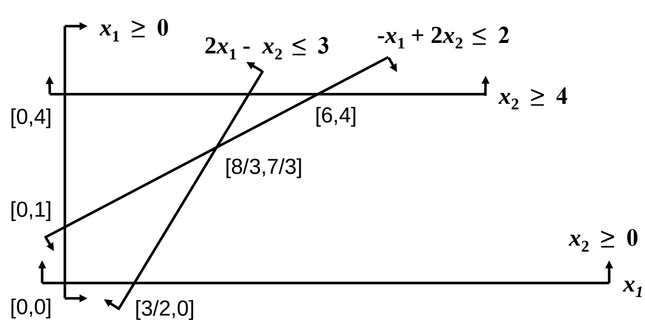
- Uygun çözüm bölgesinde **optimum çözüm olmayabilir.**



Geometrik gösterim

- **Uygun çözüm bölgesi olmayabilir.**

$$
\begin{array}{ll}
\textit{Min} & -x_1 - 3x_2 \\
\textit{Kısıtlar} & -x_1 + 2x_2 \le 2 \\
 & 2x_1 - x_2 \le 3 \\
 & x_2 \ge 4 \\
 & x_1, x_2 \ge 0
\end{array}
$$



Geometrik gösterim

Örnek:

Max

$0.3x + 0.2y$

Kısıtlar

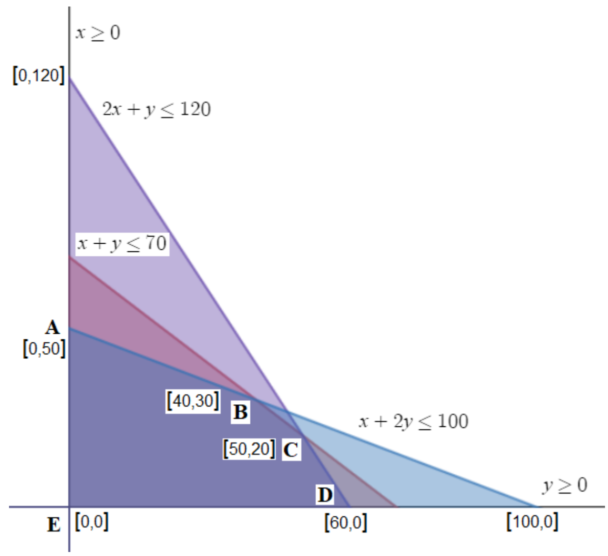
$x, y \geq 0$

$x + y \leq 70$

$x + 2y \leq 100$

$2x + y \leq 120$

A = 10
B = 18
***C = 19**
D = 18
E = 0



Geometrik gösterim

Örnek - devam

Max

$0.3x + 0.2y$

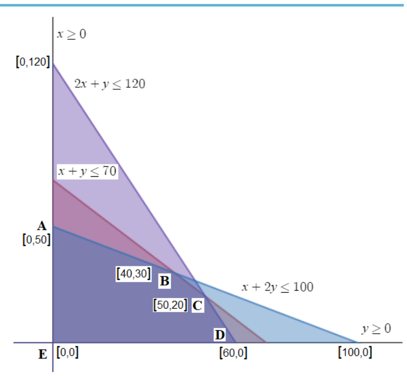
Kısıtlar

$x, y \geq 0$ (1, 2)

$x + y \leq 70$ (3)

$x + 2y \leq 100$ (4)

$2x + y \leq 120$ (5)



KÖŞE	*KOOR.*	*KÖŞE*	*KOOR.*
(1, 2)+	0, 0	(2, 5)+	60, 0
(1, 3)	0, 70	(3, 4)+	40, 30
(1, 4)+	0, 50	(3, 5)+*	50, 20
(1, 5)	0, 120	(4, 5)	140/3, 80/3
(2, 3)	70, 0		
(2, 4)	100, 0		

(1, 2) = 0
(1, 4) = 10
(2, 5) = 18
(3, 4) = 18
*(3, 5) = 19

Geometrik gösterim

Örnek:

Maks

$f(x, y, z) = 4x + 3y + 7z$

Kısıtlar

$x + 3y + 2z \geq 120$ (1)

$2x + y + 3z \geq 120$ (2)

$x \geq 0$ (3)

$y \geq 0$ (4)

$z \geq 0$ (5)

KÖŞE	*x, y, z*	*4x + 3y + 7z*
1, 2, 3+	0,120/7, 240/7	~291
1, 2, 4	-120, 0, 120	Uygun değil
1, 2, 5+	48, 24, 0	264
1, 3, 4+	0, 0, 60	420
1, 3, 5+	0, 120, 0	360
1, 4, 5+*	120, 0, 0	480
2, 3, 4+	0, 0, 40	280
2, 3, 5+	0, 120, 0	360
2, 4, 5+	60, 0, 0	240
3, 4, 5+	0, 0, 0	0

Optimum çözüm $= f(120, 0, 0) = 480$

İçerik

- Doğrusal programlama problemi
- Geometrik gösterim
- Simpleks algoritması

Simpleks algoritması

- İki değişkenli doğrusal programlama problemlerinde grafiksel çözüm yöntemi uygundur.
- **Çok sayıdaki değişken** ve **kısıt** olan problemlerde **simpleks yöntemi** (George Dantzig, 1946) daha uygundur.
- Simpleks yöntemi **feasible bölgenin köşe noktalarını** test ederek amaç fonksiyonu için **optimum değeri** bulur.
- Standart form: **Tüm kısıtlar eşitlik** şeklindedir ve **tüm değişkenler pozitiftir.**
- DP problemi, simpleks algoritmasıyla çözülmeden **önce kısıtlar standart formda** yazılmalıdır.

$$3x_1 + 2x_2 \leq 2 \qquad 3x_1 + 2x_2 + e_1 = 2 \qquad e_1 \geq 0$$

$$3x_1 + 2x_2 \geq 2 \qquad 3x_1 + 2x_2 - e_2 = 2 \qquad e_2 \geq 0$$

- $\boldsymbol{e_1}$ slack variable, $\boldsymbol{e_2}$ surplus variable olarak adlandırılır.

Simpleks algoritması

Standart form

max $z = c_1x_1 + c_2x_2 + \cdots + c_nx_n$

kısıtlar

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &\leq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &\leq b_m \end{aligned}$$

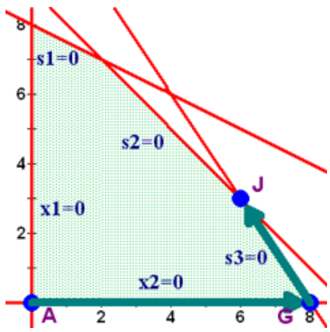
standart form

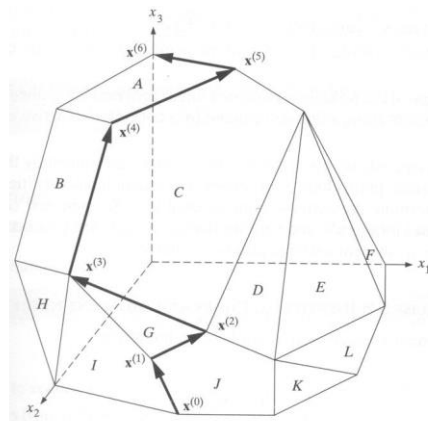
$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n + s_1 \qquad\qquad &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \qquad + s_2 \qquad &= b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \qquad\qquad + s_m &= b_m \end{aligned}$$

$$x_i \geq 0 \qquad b_i \geq 0 \qquad s_i \geq 0$$

Simpleks algoritması

- **Simpleks algoritması,** n boyutlu uzayda uygun çözüm bölgesindeki **köşe noktaları test ederek optimum çözümü bulur.**





Simpleks algoritması

- Bir DP probleminde, n adet değişken ve m adet eşitlik varsa ve $n \geq m$ ise, temel çözüm aşağıdaki gibi bulunur.
  - **(n - m) değişken sıfıra eşitlenir.** Bu değişkenler temel olmayan değişkenler olarak adlandırılır.
  - **Kalan m değişken için problem çözülür.** Bu değişkenler temel değişkenler olarak adlandırılır.
  - Başlangıç temel değişkenler vektörü **temel başlangıç çözüm** olarak adlandırılır.
  - Optimum çözüm elde edilinceye kadar **değişkenler vektörü iteratif şekilde yeniden hesaplanır.**
- Her adımdan sonra aşağıdaki **üç durum test edilir:**
  - Çözümde **iyileştirme** yapılabilir mi?
  - Temel olmayan **hangi değişken temel** olarak **alınır?**
  - **Hangi** temel **değişken temel olmayan yapılır?**

Simpleks algoritması

- Aşağıdaki kısıtlar için **max** $z = 4x_1 + 6x_2$ başlangıç çözümünü bulalım **($x_1 \geq 0$, $x_2 \geq 0$)**.

$$\begin{aligned} -x_1 + x_2 &\leq 11 \\ x_1 + x_2 &\leq 27 \\ 2x_1 + 5x_2 &\leq 90 \end{aligned}$$

- s_1, s_2 ve s_3 değişkenleriyle kısıtlar standart forma dönüştürülür.

$$\begin{aligned} -x_1 + x_2 + s_1 \qquad\qquad &= 11 \\ x_1 + x_2 \qquad + s_2 \qquad &= 27 \\ 2x_1 + 5x_2 \qquad\qquad + s_3 &= 90 \end{aligned}$$

- İki değişken **temel olmayan** seçilir (x_1 ve x_2 seçilsin.).
- Uygun **başlangıç çözümü:**

$$(x_1, x_2, s_1, s_2, s_3) = (0, 0, 11, 27, 90)$$

Simpleks algoritması

Simpleks tablosu ve pivoting işlemi – Örnek

- Simpleks yöntemi, **simpleks tablosunda satır işlemleri yaparak** optimum çözümü elde eder.

max $\quad z = 4x_1 + 6x_2$

$$-c_1x_1 - c_2x_2 - \cdots - c_nx_n + (0)s_1 + (0)s_2 + \cdots + (0)s_m + z = 0$$

kısıtlar

$$\boxed{\begin{aligned} -x_1 + x_2 &\leq 11 \\ x_1 + x_2 &\leq 27 \\ 2x_1 + 5x_2 &\leq 90 \end{aligned}} \qquad \boxed{\begin{aligned} -x_1 + x_2 + s_1 \qquad\qquad &= 11 \\ x_1 + x_2 \qquad + s_2 \qquad &= 27 \\ 2x_1 + 5x_2 \qquad\qquad + s_3 &= 90 \end{aligned}}$$

Optimum için:
Amaç fonksiyonu katsayıları ≥ 0 olmalı.

x_1	x_2	s_1	s_2	s_3	b	*Basic Variables*
−1	1	1	0	0	11	s_1
1	1	0	1	0	27	s_2
2	5	0	0	1	90	s_3
−4	−6	0	0	0	0	

↑ *Current z-value*

Başlangıç çözümü

$x_1 = 0,\ x_2 = 0,\ s_1 = 11,\ s_2 = 27,\ s_3 = 90$

$(x_1, x_2, s_1, s_2, s_3) = (0, 0, 11, 27, 90) \quad \boxed{z = 0}$

Simpleks algoritması

Simpleks tablosu ve pivoting işlemi - Örnek

- Amaç fonksiyonu katsayılarında negatif varsa iyileştirme yapılır.
- Her iyileştirme adımında bir değişken temel yapılır, bir temel değişken çıkartılır.
  - **Temel değişken** kümesine girecek değişken, **en küçük katsayıya sahip olandır** (pivot sütun).
  - **Temel olmayan** yapılacak değişken, b_i/a_{ij} (≥ 0) değerini **minimum yapan** değişkendir (pivot satır).
  - Pivot satır ve pivot sütunun **ortak elemanı** pivot olarak adlandırılır.

x_1	x_2	s_1	s_2	s_3	b	Basic Variables
−1	1	1	0	0	11	s_1 ← Departing
1	1	0	1	0	27	s_2
2	5	0	0	1	90	s_3
−4	−6	0	0	0	0	
	↑ Entering					

$$\frac{11}{1} = 11, \quad \frac{27}{1} = 27, \quad \frac{90}{5} = 18$$

Simpleks algoritması

Simpleks tablosu ve pivoting işlemi

x_1	x_2	s_1	s_2	s_3	b	
−1	1	1	0	0	11	s_1 ←
1	1	0	1	0	27	s_2
2	5	0	0	1	90	s_3
−4	−6	0	0	0	0	
	↑					

- s_1 yerine x_2 temel değişken alınır.
- İyileştirilmiş çözüm, **Gauss-Jordan eliminasyonu** ile bulunur.
- **Pivot satır ile diğer satırlar arasında** işlem yapılır.

Before Pivoting   After Pivoting

$$\begin{array}{c} R_1 \\ R_2 \\ R_3 \\ R_4 \end{array} \begin{bmatrix} -1 & 1 & 1 & 0 & 0 & 11 \\ 1 & 1 & 0 & 1 & 0 & 27 \\ 2 & 5 & 0 & 0 & 1 & 90 \\ -4 & -6 & 0 & 0 & 0 & 0 \end{bmatrix} \longrightarrow \begin{bmatrix} -1 & 1 & 1 & 0 & 0 & 11 \\ 2 & 0 & -1 & 1 & 0 & 16 \\ 7 & 0 & -5 & 0 & 1 & 35 \\ -10 & 0 & 6 & 0 & 0 & 66 \end{bmatrix} \begin{array}{l} \\ -R_1 + R_2 \\ -5R_1 + R_3 \\ 6R_1 + R_4 \end{array}$$

x_1	x_2	s_1	s_2	s_3	b	Basic Variables
−1	1	1	0	0	11	x_2
2	0	−1	1	0	16	s_2
7	0	−5	0	1	35	s_3
−10	0	6	0	0	66	

$$(x_1, x_2, s_1, s_2, s_3) = (0, 11, 0, 16, 35)$$

$$\boxed{z = 4x_1 + 6x_2 = 4(0) + 6(11) = 66}$$

Simpleks algoritması

Simpleks tablosu ve pivoting işlemi - Örnek

- s_3 yerine x_1 temel değişken alınır ve Gauss-Jordan eliminasyonu yapılır.

x_1	x_2	s_1	s_2	s_3	b		
-1	1	1	0	0	11	x_2	$11/(-1) = -11$
2	0	-1	1	0	16	s_2	$16/2 = 8$
7	0	-5	0	1	35	$s_3 \leftarrow$	$35/7 = 5$
-10 $\uparrow$	0	6	0	0	66		

$$\begin{bmatrix} -1 & 1 & 1 & 0 & 0 & 11 \\ 2 & 0 & -1 & 1 & 0 & 16 \\ 7 & 0 & -5 & 0 & 1 & 35 \\ -10 & 0 & 6 & 0 & 0 & 66 \end{bmatrix} \rightarrow \begin{bmatrix} -1 & 1 & 1 & 0 & 0 & 11 \\ 2 & 0 & -1 & 1 & 0 & 16 \\ 1 & 0 & -\frac{5}{7} & 0 & \frac{1}{7} & 5 \\ -10 & 0 & 6 & 0 & 0 & 66 \end{bmatrix} \begin{array}{l} \\ \\ \frac{1}{7}R_3 \\ \\ \end{array}$$

$$\rightarrow \begin{bmatrix} 0 & 1 & \frac{2}{7} & 0 & \frac{1}{7} & 16 \\ 0 & 0 & \frac{3}{7} & 1 & -\frac{2}{7} & 6 \\ 1 & 0 & -\frac{5}{7} & 0 & \frac{1}{7} & 5 \\ 0 & 0 & -\frac{8}{7} & 0 & \frac{10}{7} & 116 \end{bmatrix} \begin{array}{l} R_1 + R_3 \\ R_2 - 2R_3 \\ \\ R_4 + 10R_3 \end{array}$$

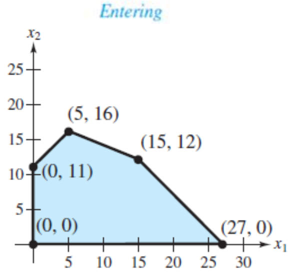
Simpleks algoritması

Simpleks tablosu ve pivoting işlemi - Örnek

- s_2 yerine s_1 temel değişken alınır ve Gauss-Jordan eliminasyonu yapılır.

x_1	x_2	s_1	s_2	s_3	b	Basic Variables
0	1	$\frac{2}{7}$	0	$\frac{1}{7}$	16	x_2
0	0	$\frac{3}{7}$	1	$-\frac{2}{7}$	6	$s_2 \leftarrow$
1	0	$-\frac{5}{7}$	0	$\frac{1}{7}$	5	x_1
0	0	$-\frac{8}{7}$ $\uparrow$ Entering	0	$\frac{10}{7}$	116	

x_1	x_2	s_1	s_2	s_3	b	Basic Variables
0	1	0	$-\frac{2}{3}$	$\frac{1}{3}$	12	x_2
0	0	1	$\frac{7}{3}$	$-\frac{2}{3}$	14	s_1
1	0	0	$\frac{5}{3}$	$-\frac{1}{3}$	15	x_1
0	0	0	$\frac{8}{3}$	$\frac{2}{3}$	132	$\leftarrow$ Maximum z-value



$$(x_1, x_2, s_1, s_2, s_3) = (15, 12, 14, 0, 0)$$

$$\boxed{z = 4x_1 + 6x_2 = 4(15) + 6(12) = 132}$$

$$\begin{array}{ccccccc} (0, 0) & & (0, 11) & & (5, 16) & & (15, 12) \\ z = 0 & \rightarrow & z = 66 & \rightarrow & z = 116 & \rightarrow & z = 132 \end{array}$$

Simpleks algoritması

Örnek:

$$\begin{aligned} \text{Maximize} \quad & Z = 4x_1 - x_2 + 2x_3 \\ \text{subject to} \quad & 2x_1 + x_2 + 2x_3 \le 6 \\ & x_1 - 4x_2 + 2x_3 \le 0 \\ & 5x_1 - 2x_2 - 2x_3 \le 4 \\ & x_1, x_2, x_3 \ge 0 \end{aligned}$$

$$\Downarrow$$

$$\begin{aligned} \text{Maximize} \quad & Z = 4x_1 - x_2 + 2x_3 \\ \text{subject to} \quad & 2x_1 + x_2 + 2x_3 + \boxed{x_4} = 6 \\ & x_1 - 4x_2 + 2x_3 + \boxed{x_5} = 0 \\ & 5x_1 - 2x_2 - 2x_3 + \boxed{x_6} = 4 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \ge 0 \end{aligned}$$

- x_4, x_5, x_6 slack variables.

Simpleks algoritması

Örnek - devam

- **Başlangıç çözümü (Z = 0):** (x_1, x_2, x_3, x_4, x_5, x_6) = (0, 0, 0, 6, 0, 4).
- Temel değişkenler (x_4, x_5, x_6), temel olmayan değişkenler (x_1, x_2, x_3).

$$\begin{array}{llllllll} (Z) & -4x_1 + x_2 & -\ 2x_3 + & 0x_4 & +\ 0x_5 & +\ 0x_6 & +Z & = 0 \\ (x_4) & 2x_1 + x_2 & +\ 2x_3 + & 1x_4 & +\ 0x_5 & +\ 0x_6 & & = 6 \\ (x_5) & x_1 - 4x_2 & +\ 2x_3 + & 0x_4 & +\ 1x_5 & +\ 0x_6 & & = 0 \\ (x_6) & 5x_1 - 2x_2 & -\ 2x_3 + & 0x_4 & +\ 0x_5 & +\ 1x_6 & & = 4 \end{array}$$

$$\frac{b_4}{c_{41}} = \frac{6}{2} = 3 \qquad \frac{b_5}{c_{51}} = \frac{0}{1} = 0 \qquad \frac{b_6}{c_{61}} = \frac{4}{5} = 0.8$$

- x_5 yerine x_1 **temel değişken** yapılır.
- **Gouss-Jordan eliminasyonu** yapılır.

Simpleks algoritması

Örnek - devam

$$
\begin{array}{ll}
(Z) & 0x_1 - 15x_2 + 6x_3 + 0x_4 + 4x_5 + 0x_6 + Z = 0 \\
(x_4) & 0x_1 + 9x_2 - 2x_3 + 1x_4 - 2x_5 + 0x_6 = 6 \\
(x_1) & 1x_1 - 4x_2 + 2x_3 + 0x_4 + 1x_5 + 0x_6 = 0 \\
(x_6) & 0x_1 + 18x_2 - 12x_3 - 0x_4 - 5x_5 + 1x_6 = 4
\end{array}
$$

$$\frac{b_4}{c_{42}} = \frac{6}{9} = 0.667 \qquad \frac{b_6}{c_{62}} = \frac{4}{18} = 0.222$$

- x_6 yerine x_2 **temel değişken** yapılır.
- **Gouss-Jordan eliminasyonu** yapılır.

Simpleks algoritması

Örnek - devam

$$
\begin{array}{ll}
(Z) & 0x_1 + 0x_2 - 4x_3 + 0x_4 - \frac{1}{6}x_5 + \frac{5}{6}x_6 + Z = \frac{10}{3} \\
(x_4) & 0x_1 + 0x_2 + 4x_3 + 1x_4 + \frac{1}{2}x_5 - \frac{1}{2}x_6 = 4 \\
(x_1) & 1x_1 + 0x_2 - \frac{2}{3}x_3 + 0x_4 - \frac{1}{9}x_5 + \frac{2}{9}x_6 = \frac{8}{9} \\
(x_2) & 0x_1 + 1x_2 - \frac{2}{3}x_3 + 0x_4 - \frac{5}{18}x_5 + \frac{1}{18}x_6 = \frac{2}{9}
\end{array}
$$

- x_4 yerine x_3 **temel değişken** yapılır.
- **Gouss-Jordan eliminasyonu** yapılır.

Simpleks algoritması

Örnek - devam

$$
\begin{array}{llcl}
(Z) & 0x_1 + 0x_2 + 0x_3 + 1x_4 + \frac{1}{3}x_5 + \frac{1}{3}x_6 + Z & = & \frac{22}{3} \\
(x_3) & 0x_1 + 0x_2 + 1x_3 + \frac{1}{4}x_4 + \frac{1}{8}x_5 - \frac{1}{8}x_6 & = & 1 \\
(x_1) & 1x_1 + 0x_2 + 0x_3 + \frac{1}{6}x_4 - \frac{1}{36}x_5 + \frac{5}{36}x_6 & = & \frac{14}{9} \\
(x_2) & 0x_1 + 1x_2 + 0x_3 + \frac{1}{6}x_4 - \frac{7}{36}x_5 - \frac{1}{36}x_6 & = & \frac{8}{9}
\end{array}
$$

- ***Z* = 22/3** için **yeni temel çözüm:** $(x_1, x_2, x_3, x_4, x_5, x_6)$ = (14/9, 8/9, 1, 0, 0, 0).
- Tüm **katsayılar pozitif** olduğundan **optimum çözümdür.**

$$Z = \frac{22}{3} = 7.333 \qquad x_1 = \frac{14}{9} = 1.556 \qquad x_2 = \frac{8}{9} = 0.889 \qquad x_3 = 1$$

Simpleks algoritması

Örnek – devam – simpleks tablosu ile gösterimi

- Tüm satırlar için pivot işleminden sonra yeni canonical form:

Iteration	Basis	Z	x_1	x_2	x_3	x_4	x_5	x_6	b_r	$\frac{b_r}{c_{rs}}$
1	Z	1	-4	1	-2	0	0	0	0	--
	x_4	0	2	1	2	1	0	0	6	3
	x_5	0	1	-4	2	0	1	0	0	0
	x_6	0	5	-2	-2	0	0	1	4	$\frac{4}{5}$

Pivotal Row

Pivotal Column

Pivotal Element

Iteration	Basis	Z	x_1	x_2	x_3	x_4	x_5	x_6	b_r	$\frac{b_r}{c_{rs}}$
2	Z	1	0	-15	6	0	4	0	0	--
	x_4	0	0	9	-2	1	-2	0	6	1/3
	x_1	0	1	-4	2	0	1	0	0	--
	x_6	0	0	18	-12	0	-5	1	4	2/9

Simpleks algoritması

Örnek – devam – simpleks tablosu ile gösterimi

- Tüm satırlar için pivot işleminden sonra yeni canonical form:

Iteration	Basis	Z	x_1	x_2	x_3	x_4	x_5	x_6	b_r	$\frac{b_r}{c_{rs}}$
3	Z	1	0	0	-4	0	$-\frac{1}{6}$	$\frac{5}{6}$	$\frac{10}{3}$	--
	x_4	0	0	0	4	1	$\frac{1}{2}$	$-\frac{1}{2}$	4	1
	x_1	0	1	0	$-\frac{2}{3}$	0	$-\frac{1}{9}$	$\frac{2}{9}$	$\frac{8}{9}$	--
	x_2	0	0	1	$-\frac{2}{3}$	0	$-\frac{5}{18}$	$\frac{1}{18}$	$\frac{2}{9}$	--
4	Z	1	0	0	0	1	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{22}{3}$	Z
	x_3	0	0	0	1	$\frac{1}{4}$	$\frac{1}{8}$	$-\frac{1}{8}$	1	x_3
	x_1	0	1	0	0	$\frac{1}{6}$	$-\frac{1}{36}$	$\frac{2}{9}$	$\frac{14}{9}$	x_1
	x_2	0	0	1	0	$\frac{1}{6}$	$-\frac{7}{36}$	$-\frac{1}{36}$	$\frac{8}{9}$	x_2

Simpleks algoritması

Örnek:

- Bir firma, TV, radyo ve gazete reklamı verecektir. Yayın organları için **maliyet** ve **tahmini hedef kitlesi** tabloda verilmiştir.

	Television	*Newspaper*	*Radio*
Cost per advertisement	\$2000	\$600	\$300
Audience per advertisement	100,000	40,000	18,000

- **Gazete**, bir firma için **haftalık reklam sayısını 10** ile sınırlamıştır.
- **Radyo reklamı sayısı**, toplam reklam sayısının **yarısından fazla olmayacak.**
- **TV reklam sayısı**, toplam reklam sayısının **%10**'undan az olmayacak.
- Haftalık **toplam reklam bütçesi \$18,200.**
- **Her yayın organı için kaç tane reklam** verilirse, **toplam ulaşılan hedef kitle sayısı maksimum olur?**

Simpleks algoritması

Örnek – devam

	Television	*Newspaper*	*Radio*
Cost per advertisement	\$2000	\$600	\$300
Audience per advertisement	100,000	40,000	18,000

$$z = 100{,}000x_1 + 40{,}000x_2 + 18{,}000x_3 \qquad \textbf{Objective function}$$

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

$$\begin{aligned} 2000x_1 + 600x_2 + 300x_3 &\leq 18{,}200 \\ x_2 &\leq 10 \\ x_3 &\leq 0.5(x_1 + x_2 + x_3) \\ x_1 &\geq 0.1(x_1 + x_2 + x_3) \end{aligned}$$

$$\left.\begin{aligned} 20x_1 + 6x_2 + 3x_3 &\leq 182 \\ x_2 &\leq 10 \\ -x_1 - x_2 + x_3 &\leq 0 \\ -9x_1 + x_2 + x_3 &\leq 0 \end{aligned}\right\} \quad \textbf{Constraints}$$

Simpleks algoritması

Örnek – devam

$$\left.\begin{aligned} 20x_1 + 6x_2 + 3x_3 &\leq 182 \\ x_2 &\leq 10 \\ -x_1 - x_2 + x_3 &\leq 0 \\ -9x_1 + x_2 + x_3 &\leq 0 \end{aligned}\right\} \quad \textbf{Constraints}$$

$$z = 100{,}000x_1 + 40{,}000x_2 + 18{,}000x_3 \qquad \textbf{Objective function}$$

x_1	x_2	x_3	s_1	s_2	s_3	s_4	b	*Basic Variables*
20	6	3	1	0	0	0	182	s_1 ← *Departing*
0	1	0	0	1	0	0	10	s_2
−1	−1	1	0	0	1	0	0	s_3
−9	1	1	0	0	0	1	0	s_4
−100,000	−40,000	−18,000	0	0	0	0	0	
↑ *Entering*								

x_1	x_2	x_3	s_1	s_2	s_3	s_4	b	*Basic Variables*
1	$\frac{3}{10}$	$\frac{3}{20}$	$\frac{1}{20}$	0	0	0	$\frac{91}{10}$	x_1
0	1	0	0	1	0	0	10	s_2 ← *Departing*
0	$-\frac{7}{10}$	$\frac{23}{20}$	$\frac{1}{20}$	0	1	0	$\frac{91}{10}$	s_3
0	$\frac{37}{10}$	$\frac{47}{20}$	$\frac{9}{20}$	0	0	1	$\frac{819}{10}$	s_4
0	−10,000	−3000	5000	0	0	0	910,000	
	↑ *Entering*							

Simpleks algoritması

Örnek – devam

$$\left.\begin{aligned} 20x_1 + 6x_2 + 3x_3 &\le 182 \\ x_2 &\le 10 \\ -x_1 - x_2 + x_3 &\le 0 \\ -9x_1 + x_2 + x_3 &\le 0 \end{aligned}\right\}$$ **Constraints**

$z = 100{,}000x_1 + 40{,}000x_2 + 18{,}000x_3$ **Objective function**

x_1	x_2	x_3	s_1	s_2	s_3	s_4	b	*Basic Variables*
1	0	$\frac{3}{20}$	$\frac{1}{20}$	$-\frac{3}{10}$	0	0	$\frac{61}{10}$	x_1
0	1	0	0	1	0	0	10	x_2
0	0	$\frac{23}{20}$	$\frac{1}{20}$	$\frac{7}{10}$	1	0	$\frac{161}{10}$	$s_3 \leftarrow$ *Departing*
0	0	$\frac{47}{20}$	$\frac{9}{20}$	$-\frac{37}{10}$	0	1	$\frac{449}{10}$	s_4
0	0	-3000	5000	10,000	0	0	1,010,000	
		↑ *Entering*						

x_1	x_2	x_3	s_1	s_2	s_3	s_4	b	*Basic Variables*
1	0	0	$\frac{1}{23}$	$-\frac{9}{23}$	$-\frac{3}{23}$	0	4	x_1
0	1	0	0	1	0	0	10	x_2
0	0	1	$\frac{1}{23}$	$\frac{14}{23}$	$\frac{20}{23}$	0	14	x_3
0	0	0	$\frac{8}{23}$	$-\frac{118}{23}$	$-\frac{47}{23}$	1	12	s_4
0	0	0	$\frac{118{,}000}{23}$	$\frac{272{,}000}{23}$	$\frac{60{,}000}{23}$	0	1,052,000	

$z = 1{,}052{,}000$ **Maximum weekly audience**

Simpleks algoritması

Örnek – devam

$$\left.\begin{aligned} 20x_1 + 6x_2 + 3x_3 &\le 182 \\ x_2 &\le 10 \\ -x_1 - x_2 + x_3 &\le 0 \\ -9x_1 + x_2 + x_3 &\le 0 \end{aligned}\right\}$$ **Constraints**

$z = 100{,}000x_1 + 40{,}000x_2 + 18{,}000x_3$ **Objective function**

x_1	x_2	x_3	s_1	s_2	s_3	s_4	b	*Basic Variables*
1	0	0	$\frac{1}{23}$	$-\frac{9}{23}$	$-\frac{3}{23}$	0	4	x_1
0	1	0	0	1	0	0	10	x_2
0	0	1	$\frac{1}{23}$	$\frac{14}{23}$	$\frac{20}{23}$	0	14	x_3
0	0	0	$\frac{8}{23}$	$-\frac{118}{23}$	$-\frac{47}{23}$	1	12	s_4
0	0	0	$\frac{118{,}000}{23}$	$\frac{272{,}000}{23}$	$\frac{60{,}000}{23}$	0	1,052,000	

$z = 1{,}052{,}000$ **Maximum weekly audience**

$x_1 = 4, x_2 = 10, x_3 = 14$

Media	*Number of Advertisements*	*Cost*	*Audience*
Television	4	$8000	400,000
Newspaper	10	$6000	400,000
Radio	14	$4200	252,000
Total	28	$18,200	1,052,000

	Television	*Newspaper*	*Radio*
Cost per advertisement	$2000	$600	$300
Audience per advertisement	100,000	40,000	18,000

Simpleks algoritması

Minimizasyon probleminin çözümü

- Minimizasyon problemi, maksimizasyon problemine dönüştürülerek çözülür (augmented matrisin transpozu alınır.).
- x **değişkenleri**, alt satırda slack değişkenlerin sütunundadır.

min $w = c_1x_1 + c_2x_2 + \cdots + c_nx_n$

kısıtlar

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &\geq b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &\geq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &\geq b_m \end{aligned}$$

$x_i \geq 0 \quad b_i \geq 0$

$$\left[\begin{array}{cccc:c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ & & & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} & b_m \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ c_1 & c_2 & \cdots & c_n & 0 \end{array}\right]$$

max $z = b_1y_1 + b_2y_2 + \cdots + b_my_m$

kısıtlar

$$\begin{aligned} a_{11}y_1 + a_{21}y_2 + \cdots + a_{m1}y_m &\leq c_1 \\ a_{12}y_1 + a_{22}y_2 + \cdots + a_{m2}y_m &\leq c_2 \\ &\vdots \\ a_{1n}y_1 + a_{2n}y_2 + \cdots + a_{mn}y_m &\leq c_n \end{aligned}$$

$y_1 \geq 0, y_2 \geq 0, \ldots, y_m \geq 0$

$$\left[\begin{array}{cccc:c} a_{11} & a_{21} & \cdots & a_{m1} & c_1 \\ a_{12} & a_{22} & \cdots & a_{m2} & c_2 \\ & & & & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} & c_n \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ b_1 & b_2 & \cdots & b_m & 0 \end{array}\right]$$

Simpleks algoritması

Örnek:

min $w = 3x_1 + 2x_2$

kısıtlar

$$2x_1 + x_2 \geq 6$$
$$x_1 + x_2 \geq 4$$
$$x_1 \geq 0 \quad x_2 \geq 0$$

$$\left[\begin{array}{cc:c} 2 & 1 & 6 \\ 1 & 1 & 4 \\ \cdots & \cdots & \cdots \\ 3 & 2 & 0 \end{array}\right] \dashrightarrow \left[\begin{array}{cc:c} 2 & 1 & 3 \\ 1 & 1 & 2 \\ \cdots & \cdots & \cdots \\ 6 & 4 & 0 \end{array}\right]$$

max $z = 6y_1 + 4y_2$

kısıtlar

$$2y_1 + y_2 \leq 3$$
$$y_1 + y_2 \leq 2$$
$$y_1 \geq 0 \quad y_2 \geq 0$$

y_1	y_2	s_1	s_2	b	Basic Variables	
2	1	1	0	3	s_1	← Departing
1	1	0	1	2	s_2	
−6	−4	0	0	0		
↑ Entering						

y_1	y_2	s_1	s_2	b	Basic Variables	
1	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{3}{2}$	y_1	
0	$\frac{1}{2}$	$-\frac{1}{2}$	1	$\frac{1}{2}$	s_2	← Departing
0	−1	3	0	9		
	↑ Entering					

Simpleks algoritması

Örnek:

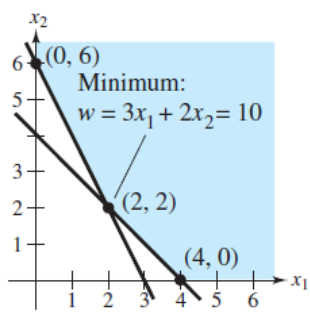
max $z = 6y_1 + 4y_2$

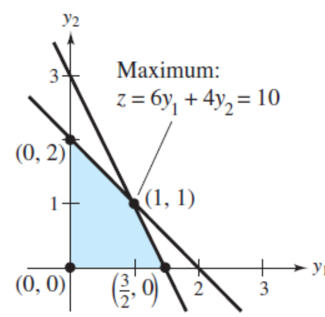
kısıtlar

$$2y_1 + y_2 \leq 3$$
$$y_1 + y_2 \leq 2$$
$$y_1 \geq 0 \quad y_2 \geq 0$$

y_1	y_2	s_1	s_2	b	Basic Variables
1	0	1	−1	1	y_1
0	1	−1	2	1	y_2
0	0	2	2	10	
		↑ x_1	↑ x_2		

$$\boxed{z = 10 \quad w = 10 \quad x_1 = 2 \quad x_2 = 2}$$





Ödev

- Bir yazılım firmasının **nitelikleri farklı iki personeli vardır.**
- Her iki personel ile ayrı ayrı **toplam çalışma saati** üzerinden anlaşma yapmıştır **(birinci personel ile 400 saat, ikinci personel ile 300 saat).**
- Firma, **iki farklı yazılım projesi** işi almıştır. **Birinci proje 200 saat**, **ikinci proje 300 saat** toplam çalışma süresi gerektirmektedir.
- **İki personelin** bu **iki farklı proje** için **ücret/saat maliyetleri** tabloda verilmiştir.
- Bu iki personel ile iki projeyi **minimum maliyetle** geliştirebilmek için **her iki personelin iki projede kaçar saat çalışması gerektiğini** simpleks algoritması ile bulunuz.

	Proje 1	**Proje 2**
Personel 1	30 TL	25 TL
Personel 2	36 TL	30 TL