STANDART FORMU YAPAY DEĞİŞKEN İÇEREN DOĞRUSAL PROGRAMLAMA MODELİNİN ÇÖZÜMÜ İÇIN İKI EVRELI SİMPLEKS ALGORİTMASI

Yard.Doç.Dr. Hasan DURUCASU
Anadolu Üniversitesi İktisadi ve İdari
Bilimler Fakültesi İşletme Bölümü
Sayısal Yöntemler Anabilim Dalı

SUMMARY

The solution of linear programming problems that are faced in business applications in real life is required to use an effective resolution algorithm. Simplex algorithm that is used to solve linear programming problems is a set of repeated simple operation steps until the best solution is reached. In this study, therefore the mathematical details of operational steps in the simplex algorithm are examined. Then the two phase version that is related to a special case of simplex approach is presented and a computer program based on that approach is developed and listed.

GİRİŞ

Günümüzde işletme alanındaki çok sayıda araştırmacı, doğrusal programlama problemlerinin çözümü için geliştirilmiş kullanıma hazır paket programlardan yararlanmaktadır. Kullanıma hazır bilgisayar paket programlarını günümüz araştırmacısına hesaplama kolaylığı sağlamasına karşın, kanımızca simpleks algoritmasının matematik temelinin işletmecilik araştırmacısı tarafından tam olarak kavranması, onun bilgisayar çözüm çıktılarından en fazla yararı elde edilebilmesinin ön koşulunu oluşturmaktadır.

SİMPLEKS YÖNTEMİNE İLİŞKİN MATEMATİK AYRINTILAR

Simpleks yaklaşımı öncelikle işletmecilik probleminin matematik olarak belirtilmesini zorunlu kılmaktadır. Bilindiği gibi doğrusal programlama modelinin standart formunun matris biçiminde ifadesi, $b \geq 0$ olmak üzere,

$AX = b$ sınırlayıcı koşulları ve
$X \geq 0$ negatif olmama kısıtı altında
$x_0 = CX$amaç fonksiyonu biçimindedir.

Yukarıda verilen standart formda, C amaç fonksiyonu katsayılar matrisini göstermektedir. C amaç fonksiyonu katsayılar matrisinde temel değişkenlerin katsayıları C_B modelde yer alan diğer değişkenlere ilişkin katsayılar C_R matrisi olarak gösterildiğinde, bölümlendirilmiş matris yazılımı kullanılarak $C=(C_B\text{-}C_R)$ yazılabilir.

Benzer biçimde modeldeki X gösteriminde x_B temel değişkenleri; x_R temel çözüm dışı diğer değişkenleri simgelediğinde, X matrisi bölümlendirilmiş matris biçiminde $X = \begin{bmatrix} x_B \\ x_R \end{bmatrix}$ olarak yazılır.

Matrislerin bölümlendirilmiş matrisler olarak yazılması yaklaşımı A teknolojik matrisine uygulandığında, A'nın temel değişkenlere ilişkin katsayıları B, temel değişken olmayan diğer değişkenlerin katsayıları R alt matrisleri biçiminde ifade edilebilir.
Bu durumda yukarıdaki model, bölümlendirilmiş matris yazılımı kullanılarak, $x_B, x_R \geq 0$ kısıtları altında

$$[B\ R]\begin{bmatrix} x_B \\ x_R \end{bmatrix} = b \quad (1)$$

$$X_0 = [C_B\ C_R]\begin{bmatrix} x_B \\ x_R \end{bmatrix} \quad (2)$$

biçiminde yazılır. (1)'den

$$Bx_B + Rx_R = b \quad (3)$$

yazılır.

Benzer biçimde (2) $\quad X_0 = C_B x_B + C_R x_R \quad$ (4)

olarak düzenlenir.

Standart modelin son ifadesinde, $|B| \neq 0$ olduğunda (3) ifadesinin her iki yanı, soldan B^{-1} ile çarpılarak

önce $\quad B^{-1}Bx_B + B^{-1}Bx_R = B^{-1}b$

sonra $\quad x_B = B^{-1}b - B^{-1}Rx_R \quad$ (5)

elde edilir.

(4) ve (5)'ten

$$x_0 = C_B(B^{-1}b - B^{-1}Rx_R) + C_R x_R$$

$$x_0 = C_B B^{-1} - C_B B^{-1}Rx_R + C_R x_R$$

$$x_0 = C_B B^{-1} - (C_B B^{-1}R - C_R)x_R \quad (6)$$

yazılır.

Sıralanagelen işlemler bir tabloda özetlendiğinde,

$\mathbf{x}_0$	$\mathbf{x}_B$	$\mathbf{x}_R$	Sağ Taraf
1	0	$\mathbf{C}_B\mathbf{B}^{-1}\mathbf{R}-\mathbf{C}_R$	$\mathbf{C}_B\mathbf{B}^{-1}\mathbf{b}$
0	1	$\mathbf{B}^{-1}\mathbf{R}$	$\mathbf{B}^{-1}\mathbf{b}$

biçiminde simpleks yönteminin matrissel çözüm tablosu elde edilir.

(5)'te temel dışı değişkenleri gösteren $\mathbf{x}_R = \mathbf{0}$ olduğunda $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b}$ temel çözümdür. $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$ çözümü ise temel uygun çözümdür. Bu durumda, (6) denklemi $x_0 = \mathbf{C}_B\mathbf{B}^{-1}\mathbf{b}$ biçimine girer.

$\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$ için bulunan mevcut çözüm, en iyi çözüm olarak benimsenemiyorsa, temel değişken kümesinden bir değişkenin kümenin dışına atılıp, yerine temel değişken kümesinde yer almıyan bir değişkenin yerleştirilmesiyle çözümün iyileştirilmesi gerekir.

En iyi çözümün elde edilebilmesi için öncelikle temel çözüm kümesine sokulması gereken temel çözüm kümesi dışı değişkenin belirlenmesi gerekir. Bu amaçla $\mathbf{x}_R \neq \mathbf{0}$ olduğunda, (6)'da yer alan $\mathbf{B}^{-1}\mathbf{R}$ çarpım matrisinin j. sütunu olan

$$\left(\mathbf{B}^{-1}\mathbf{R}\right)_j = \mathbf{y}_j \qquad (7)$$

olarak ele alındığında, $\left(\mathbf{C}_B\mathbf{B}^{-1}\mathbf{R}\right)_j = \mathbf{C}_B.\mathbf{y}_j = \mathbf{z}_j$ olarak konulabilir. Son ifadede yer alan $\mathbf{y}_j$ sütun matrisi elemanları cinsinden $\mathbf{y}_j = \begin{bmatrix} \mathbf{y}_{1j} \\ \mathbf{y}_{2j} \\ \cdots \\ \mathbf{y}_{nj} \end{bmatrix}$ biçiminde açık olarak yazılabilir. $\mathbf{y}_j$ sütun matrisinin her bir elemanının genel ifadesi ise, s temel değişken kümesini göstermek üzere $\mathbf{y}_{sj}$ biçimindedir.

Böylece (6) $$x_0 = \mathbf{C}_B\mathbf{B}^{-1}\mathbf{b} - \sum_j (\mathbf{z}_j - \mathbf{c}_j)\mathbf{x}_j$$

veya $$x_0 = \mathbf{C}_B\mathbf{B}^{-1}\mathbf{b} + \sum_j (\mathbf{c}_j - \mathbf{z}_j)\mathbf{x}_j \qquad (8)$$

olarak yazılır.

(8)'de yer alan $\mathbf{c}_j - \mathbf{z}_j$ terimi, temel çözüm dışı $\mathbf{x}_j$ değişkenlerinden birinin temel çözüm kümesine sokulmasının x_0 amaç fonksiyonun değerine getireceği farklılaşmanın bir ifadesidir. Bu nedenle, enbüyükleme amaçlı problemlerde, $\mathbf{c}_j - \mathbf{z}_j$ pozitif değerlerinden en büyüğünü; enküçükleme amaçlı problemlerde $\mathbf{c}_j - \mathbf{z}_j$ negatif değerlerinden en küçüğünü veren $\mathbf{x}_j$ değişkeni temel çözüm kümesine sokulacaktır. Özetle temel değişken kümesine sokulacak değişken $\mathbf{x}_k$ olarak adlandırıldığında;

enbüyükleme amaçlı modelde $\mathbf{x}_k = \underset{j}{\mathbf{enb}}\left\{\mathbf{c}_j - \mathbf{z}_j \,\middle|\, \mathbf{c}_j - \mathbf{z}_j > \mathbf{0}\right\}$ (9),

enküçükleme amaçlı modelde $\mathbf{x}_k = \underset{j}{\mathbf{enk}}\left\{\mathbf{c}_j - \mathbf{z}_j \,\middle|\, \mathbf{c}_j - \mathbf{z}_j < \mathbf{0}\right\}$ (10)

olmalıdır.

(9) denklemi göz önünde bulundurularak, tüm j'ler için $\mathbf{c}_j - \mathbf{z}_j \leq 0$ koşulu sağlandığında enbüyük amaç fonksiyonu değerine erişildiği, aksine tüm j'ler için $\mathbf{c}_j - \mathbf{z}_j \leq 0$ eşitsizliği sağlanamıyorsa en büyük çözüme henüz erişilememiş olduğu sonucuna varılır.

Benzer biçimde (10)'dan tüm j'ler için $\mathbf{c}_j - \mathbf{z}_j \geq 0$ olduğunda enküçük amaç fonksiyonu değerinin elde edildiği görülür. Tüm j'ler için $\mathbf{c}_j - \mathbf{z}_j \geq 0$ olmadığında ise en küçük amaç fonksiyonu değerinin henüz elde edilememiş olduğu anlaşılır.

$\mathbf{x}_k$ değişkeninin temel çözüm kümesine sokulmasını, bu kümeden çıkarılacak $\mathbf{x}_r$ değişkeninin saptanması işlemi izler. Temel çözüm kümesine giren değişken $\mathbf{x}_k$ olduğunda, (5) denklemi önce,

$$\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} - (\mathbf{B}^{-1}\mathbf{R})_k \mathbf{x}_k,$$

sonra (7) dikkate alınarak $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} - \mathbf{y}_k \mathbf{x}_k$ biçiminde yazılır.

Son matris denklemin s. satırı ise $(\mathbf{x}_B)_s = (\mathbf{B}^{-1}\mathbf{b})_s - \mathbf{y}_{sk}\mathbf{x}_k$ biçimindedir. Modelde yer alan temel ya da temel çözüm dışı tüm değişkenlerin negatif olamama kısıtı son ifadeye uygulandığında $(\mathbf{x}_B)_s = (\mathbf{B}^{-1}\mathbf{b})_s - \mathbf{y}_{sk}\mathbf{x}_k \geq 0$ elde edilir. Son ifadeden $0 \leq \mathbf{x}_k \leq \dfrac{(\mathbf{B}^{-1}\mathbf{b})_s}{\mathbf{y}_{sk}}$ yazılır. Öte yandan temel çözüm kümesinden çıkarılacak değişkenin değerinin sıfır düzeyinde olması gerektiği bilinmektedir. Bu varsayım son eşitsizlikle birlikte, x_r çözüm kümesinden çıkacak değişkeni gösterdiğinde

$$x_r = \underset{s}{\text{enk}} \left\{ \frac{(B^{-1}b)_s}{y_{sk}} \middle| \frac{(B^{-1}b)_s}{y_{sk}} > 0 \right\} \qquad (11)$$

biçiminde formüle edilir. Son ifade negatif oranlar dikkate alınmaksızın pozitif oranların en büyüğüne karşılık gelen değişkenin temel çözüm kümesinden çıkarılacağını belirtmektedir. Uygulamada elle yapılan hesaplamalarda basitlik sağlama amacıyla (12)'nin

$$x_r = \underset{s}{\text{enk}} \left\{ \frac{b_s}{y_{sk}} \middle| \frac{b_s}{y_{sk}} > 0 \right\}$$ biçimi kullanılır.

x_r değişkenin temel çözüm kümesinden çıkarılıp yerine x_k değişkenin temel çözüm kümesine sokulmasıyla yeni bir tablo düzenlenip, yukarıda verilen hesaplamalar tekrar edilir.

İKİ AŞAMALI SİMPLEKS YÖNTEMİ

Yineleme temelli simpleks algoritması bir uygun başlangıç çözümünden daha iyi çözümler elde etmeyi amaçlamaktadır. Bu nedenle başlangıç çözümü uygun çözüm olmadığında alışılmış simpleks algoritması kullanılamamaktadır.

Bilindiği gibi pozitif değerli bir yapay değişkenin bir çözüm kümesinde var olması, bu çözümün uygun çözüm olamıyacağının bir göstergesidir. Öte yandan işletmecilik uygulamasında karşılaşılan kısıtlamalar başlangıç çözüm kümesinde çoğu kez yapay değişkenin varlığını zorunlu kılabilmektedir. Doğrusal programlama modelinin standart formunun yapay değişkenler içerdiği böylesi durumlarda simpleks yönteminin iki aşamalı versiyonu kullanılmaktadır.

Simpleks yönteminin iki aşamalı versiyonunda, çözüm iki aşamada bulunmaktadır: I. aşama'da herhangi bir yapay değişkenin temel çözüm kümesinde yer almadığı bir çözüm bulunur. Tüm yapay değişkenler temel çözüm kümesi dışında bırakıldığında bulunan çözüm, tanım dolayısıyla uygun çözüm kümesini oluşturur. Uygun çözüm kümesi bulunduğunda, II. aşama, uygun çözüm kümesinde yer alan çözümlerden birini en elverişli çözüm olarak değerlendirip benimser.

I. AŞAMA:

Bu aşamada özgün Z amaç fonksiyonu ihmal edilir. Amaç fonksiyonu olarak tüm yapay değişkenlerin toplamı olarak tanımlanan yeni Q amaç fonksiyonu kullanılır.

Yeni amaç fonksiyonu Q, problemin özgün kısıtları uyarınca enküçüklenir. Böylesi bir işlemin uygulanmasının nedeni, tüm yapay değişkenler temel çözüm kümesi dışında tutulduğunda, yani temel uygun çözüm bulunduğunda Qenk'ün sıfır olacağı gerçeğidir.

Problemin özgün kısıtları uyarınca enküçüklenen Q'nun değeri olan Q* sıfıra eşit olarak bulunduğunda yöntemin II. aşamasına geçilir. Tersine Q*'ın sıfırdan büyük olması tüm yapay değişkenlerin uygun çözüm kümesi dışında bırakılamadığının göstergesidir ve bu durumda problemin reel çözümü yoktur.

II. AŞAMA:

Özgün amaç fonksiyonu Z ve onun özgün katsayıları, I. aşamanın son simpleks tablosunun $\mathbf{A}$ teknolojik matrisi ve $\mathbf{B}$ sağ taraf vektörü kullanılarak II. aşamanın başlangıç tablosuna ilişkin z_j ve $z_j - c_j$ satırları hesaplanır. Ardışık hesaplamalara elverişli çözüm bulunana değin devam edilir.

BİLGİSAYAR PROGRAMI

Yukarıda yer alan iki aşamalı simpleks algoritmasından hareketle en büyükleme amaçlı modellerin çözümü için geliştirilmiş bilgisayar programı izleyen kesimde verilmiştir.

```
100 REM ENBÜYÜKLEME AMAÇLI DOĞRUSAL MODELE İLİŞKİN
110 REM 2 EVRELİ SİMPLEKSİ TEMEL ALAN PROGRAM
120 REM KARAR DEĞİŞKENİ SAYISI: M
121 READ M
```

```
130 REM DENKLEM SAYISI            : N
131 READ N
140 H=N+M: K=H+N
150 DIM B(N,1), B1(N,1), B2(N,1), F(1,K+1), F1(1,K+1), A(N,K+1), A1(N,K+1)
151 DIM A2(N,K+1), B3(N,1), C(N,1), A3(N,K+1), TOP(1,K+1), SD(1,K+1)
171 FOR I=1 TO N: READ B(I,1): NEXT I
181 FOR J=2 TO K+1: IF J<=M+1 THEN READ SD(1,J): NEXT J
183 FOR I=1 TO N
184 FOR J=1 TO K: A(I,J)=0: NEXT J
185 NEXT I
190 FOR I=1 TO N
201 READ D$
205 FOR J=2 TO K+1
211 IF J<=M+1 THEN READ A(I,J): GO TO 320
220 IF J=I+M+1 AND (D$="KE" OR D$="ke") THEN A(I,J)= 1
230 IF J=I+M+1 AND (D$="BE" OR D$="be") THEN A(I,J)=-1
240 IF J=I+M+1 AND (D$="ES" OR D$="es") THEN A(I,J)= 1
250 IF B(I,1)<0 THEN A(I,J)=-A(I,J)
320 NEXT J
330 IF B(I,1)<0 THEN B(I,1)=-B(I,1)
340 NEXT I
346 READ G$
350 FOR J=1 TO K+1
360 IF G$="EB" OR G$="eb" AND J>H+1 THEN F(I,J)=-1
370 IF G$="EK" OR G$="ek" AND J>H+1 THEN F(I,J)= 1
380 IF J>H+1 THEN F(I,J)= 0
390 NEXT J
410 FOR J=1 TO K+1
420 IF J=1 THEN LPRINT "    X";
430 IF J>1 THEN LPRINT USING " # # # # #";J-1;
440 IF J=K+1 THEN LPRINT "    ST";
450 NEXT J
453 LPRINT
455 FOR J=1 TO K+2: LPRINT "=======";: NEXT J: LPRINT
470 FOR J=1 TO K+1
480 IF J=1 THEN LPRINT "    1";
490 IF J>1 THEN LPRINT USING "# # # . # #";F(1,J);
500 NEXT J
505 LPRINT
520 FOR J= 1 TO K+1: LPRINT "-------";: NEXT J: LPRINT
530 FOR I=1 TO N
540 FOR J=1 TO K+1
550 IF J=1 THEN LPRINT "   0";
560 IF J>1 THEN LPRINT USING "# # # . # #";A(I,J);
570 NEXT J
580 LPRINT "  = ";: LPRINT USING "# # # . # #";B(I,1)
590 NEXT I
```

```
600 FOR J=1 TO K+1: LPRINT "*******";: NEXT J: LPRINT
610 FOR J=1 TO K+1
620 FOR I=1 TO N
630 TOP(1,J)=TOP(1,J)+A(I,J)
640 NEXT I
650 NEXT J
660 ST=0
670 FOR I=1 TO N
680 ST=ST+B(I,1)
690 NEXT I
750 FOR J=2 TO K+1
760 F(1,J)=TOP(1,J)+F(1,J)
770 NEXT J
790 FOR J=1 TO K+1
800 IF J=1 THEN LPRINT "    1";
810 IF J>1 THEN LPRINT USING "# # # . # #";F(1,J);
820 IF J=K+1 THEN LPRINT USING "# # # . # #";ST
830 NEXT J
840 FOR J=1 TO K+1: LPRINT "-------";: NEXT J: LPRINT
850 FOR I=1 TO N
860 FOR J=1 TO K+1
870 IF J=1 THEN LPRINT "    0";
880 IF J>1 THEN LPRINT USING "# # # . # #";A(I,J);
890 NEXT J
900 LPRINT " = ";: LPRINT USING "# # # . # #";B(I,1)
910 NEXT I
920 FOR J=1 TO K+1: LPRINT "*******";: NEXT J: LPRINT
930 KONT=0
940 FOR J=2 TO K+1
950 IF F(1,J)>1E-09 THEN KONT=1
960 NEXT J
970 IF KONT=1 THEN 1020
980 IF KONT=0 THEN 2880
1020 FOR J=2 TO K+1
1030 IF J=2 THEN S=F(1,J) : T=J
1040 IF F(1,J)>S THEN S=F(1,J) : T=J
1050 NEXT J
1060 KONT=0
1070 FOR I=1 TO N
1080 IF A(I,T)>0 THEN C(I,1)=B(I,1)/A(I,T) ELSE C(I,1)=1E+37
1090 NEXT I
1100 FOR I=1 TO N
1110 IF C(I,1)>0 THEN KONT=1
1120 NEXT I
1130 IF KONT = 1 THEN 1150
1140 IF KONT=0 THEN 3020
1150 FOR I=1 TO N
```

```
1160 IF C(I,1)>0 THEN T1=I: I=N
1170 NEXT I
1180 FOR I=T1 TO N
1190 IF I=T1 THEN C1=C(I,1)
1200 IF C1>0 THEN 1210 ELSE 1230
1210 IF C(I,1)<=C1 THEN C1=C(I,1): T1=I
1220 IF C(I,1)>C1 THEN C1=C1
1230 NEXT I
1240 FOR I=1 TO N
1250 B1(T1,1)=B(T1,1)/A(T1,T)
1260 NEXT I
1270 FOR J=2 TO K+1
1280 A1(T1,J)=A(T1,J)/A(T1,T)
1290 NEXT J
1300 FOR I=1 TO N
1310 FOR J=2 TO K+1
1320 IF A(I,T)=0 OR A(I,T)=A(T1,J) THEN B2(T1,1)=B1(T1,1)
1330 B2(I,1)=(B1(T1,1)*A(I,T))-B(I,1)
1340 IF I=T1 THEN B2(I,1)=B1(T1,1)
1350 IF A(I,T)=0 OR A(I,T)=A(T1,J) THEN A2(T1,J)=A1(T1,1)
1360 A2(I,J)=(A1(T1,J)*A(I,T))-A(I,J)
1370 IF I=T1 THEN A2(I,J)=A1(T1,J)
1380 IF B2(I,1)<0 THEN A2(I,J)=-A2(I,J)
1390 NEXT J
1400 IF B2(I,1)<0 THEN B2(I,1)=-B2(I,1)
1410 NEXT I
1420 FOR J=2 TO K+1
1430 A3(T1,J)=A2(T1,J)*F(1,T)
1440 F1(1,J)=F1(1,J)-A3(T1,J)
1450 NEXT J
1460 B3(T1,1)=B2(T1,1)*F(1,T)
1470 ST=ST-B3(T1,1)
1480 FOR I=1 TO N
1490 IF I=T1 THEN A5(I,1)=T ELSE A5(I,1)=0
1500 IF I=T1 THEN A(I,1)=0
1510 A7(I,1)=A5(I,1)+A(I,1)
1520 NEXT I
1540 FOR J=1 TO K+1
1550 IF J=1 THEN LPRINT "    I";
1560 IF J>1 THEN LPRINT USING "# # # . # #";F1(1,J);
1570 IF J=K+1 THEN LPRINT USING "    # # # . # #";ST
1580 NEXT J
1595 FOR J=1 TO K+1: LPRINT "-------";: NEXT J: LPRINT
1600 FOR I=1 TO N
1610 FOR J=1 TO K+1
1620 IF J=1 AND I = T1 THEN LPRINT USING "   # #";T-1;
1630 IF J=1 AND I<>T1 THEN LPRINT USING "   # #";A7(I,1)-1;
```

```
1640 IF J>1 THEN LPRINT USING "# # # . # #";A2(I,J);
1650 NEXT J
1660 LPRINT " = ";: LPRINT USING "# # # . # #";B2(I,1)
1670 NEXT I
1680 FOR J=1 TO K+1: LPRINT "*******";: NEXT J: LPRINT
1690 FOR I=1 TO N
1700 FOR J=2 TO K+1
1710 A(I,J)=A2(I,J)
1720 A(I,1)=A7(I,1)
1730 B(I,1)=B2(I,1)
1740 F(1,J)=F1(1,J)
1750 NEXT J
1760 NEXT I
1770 GOTO 930
2880 IF ST<1E-08 THEN ST=0 ELSE GOTO 3040
2890 LPRINT: LPRINT
2920 LPRINT "*** ENBÜYÜK ÇÖZÜM BULUNDU ***"
2925 LPRINT: LPRINT
2930 FOR I=1 TO N
2940 FOR J=2 TO K+1
2950 IF A(I,1)=J AND I>0 THEN Z=SD(1,J)*B2(I,1)+Z
2960 NEXT J
2970 LPRINT "X(";A(I,1)-1;")";" = ";B(I,1): LPRINT
2990 NEXT I
3000 LPRINT "S O N U Ç = ";Z
3010 GOTO 9999
3020 LPRINT "SONSUZ ÇÖZÜM VAR"
3030 GOTO 3050
3040 LPRINT "ÇÖZÜM BULUNAMADI"
9000 DATA 2,3
9001 DATA 18,3,5
9002 DATA 40,60
9003 DATA "KE",3,2
9004 DATA "BE",1,1
9005 DATA "ES",-1,3
9006 DATA "EB"
9999 END
```

Yukarıda verilen program Enb $x_0 = 40x_1 + 60x_2$

$$
\begin{aligned}
3x_1 + 2x_2 &\leq 18 \\
x_1 + x_2 &\geq 3 \\
-x_1 + 3x_2 &= 5 \\
x_1, x_2 &\geq 0
\end{aligned}
$$

doğrusal modelinin çözümüne ilişkin olarak

```
  x     1     2     3     4     5     6     7     8     ST
==============================================================
  1  0.00  0.00  0.00  0.00  0.00 -1.00 -1.00 -1.00
--------------------------------------------------------------
  0  3.00  2.00  1.00  0.00  0.00  0.00  0.00  0.00 =  18.00
  0  1.00  1.00  0.00 -1.00  0.00  0.00  0.00  0.00 =   3.00
  0 -1.00  3.00  0.00  0.00  1.00  0.00  0.00  0.00 =   5.00
**************************************************************
  1  3.00  6.00  1.00 -1.00  1.00 -1.00 -1.00 -1.00    26.00
--------------------------------------------------------------
  0  3.00  2.00  1.00  0.00  0.00  0.00  0.00  0.00 =  18.00
  0  1.00  1.00  0.00 -1.00  0.00  0.00  0.00  0.00 =   3.00
  0 -1.00  3.00  0.00  0.00  1.00  0.00  0.00  0.00 =   5.00
**************************************************************
  1  2.00 -6.00  0.00  0.00 -2.00  0.00  0.00  0.00    16.00
--------------------------------------------------------------
 -1  3.67  0.00  1.00  0.00 -0.67  0.00  0.00  0.00 =  14.67
 -1  1.33  0.00  0.00 -1.00 -0.33  0.00  0.00  0.00 =   1.33
  2 -0.33  1.00  0.00  0.00  0.33  0.00  0.00  0.00 =   1.67
**************************************************************
  1  0.00 -6.00  0.00  1.50 -2.00  0.00  0.00  0.00    14.00
--------------------------------------------------------------
 -1  0.00  0.00  1.00  2.75  0.25  0.00  0.00  0.00 =  11.00
  1  1.00  0.00  0.00 -0.75  0.00  0.00  0.00  0.00 =   1.00
  2  0.00  1.00  0.00 -0.25  0.25  0.00  0.00  0.00 =   2.00
**************************************************************
  1  0.00 -6.00 -0.55  0.00 -2.14  0.00  0.00  0.00     8.00
--------------------------------------------------------------
  4  0.00  0.00  0.36  1.00  0.09  0.00  0.00  0.00 =   4.00
  1  1.00  0.00  0.27  0.00  0.07  0.00  0.00  0.00 =   4.00
  2  0.00  1.00  0.09  0.00  0.27  0.00  0.00  0.00 =   3.00
**************************************************************

*** ENBÜYÜK ÇÖZÜM BULUNDU ***

X( 4 )= 4
X( 1 )= 4
X( 2 )= 3


S O N U Ç =  340
```

çıktısını vermektedir.

SONUÇ

Simpleks algoritması doğrusal programlama problemlerinin çözümüne sistematik bir yaklaşım sağlamaktadır. Algoritmanın işlemleri yinelemeye yönelik temel özelliği onun programlara esas olarak benimsenmesine yol açmaktadır. Temel çözüm kümesine sokulacak ve bu kümeden çıkarılacak değişkenlerin saptanması etkin ve kullanımı kolay ölçütlerle sağlanmaktadır. Yinelemeleri kontrol eden karar ölçütlerinin değişikliğe uğratılmasını temel alan daha etkin hesaplama algoritmalarından hareket eden güçlü programlar geniş modellerin çözümü için kullanılmaktadır.

KAYNAKLAR

1. ACAR Ahmet , Linear Programming For Managerial Decisions, METU, Ankara, (1993).

2. CLAYCOMBE W.W., SULLIVAN W.G., Foundations of Mathematical Programming, Reston Publishing Company Inc., Reston (Virginia), (1975).

3. ESİN A., Yöneylem Araştırmasında Yararlanılan Karar Yöntemleri, AİTİA Yayını, Ankara, (1981).

4. HAMDY A. Taha, Operations Research, MacMillan Pub. Co. Inc., New York, (1976).

5. KAUFMAN A., Methodes et Modeles de La Recherche Operationnelle, 2, 2, Dunod, Paris, (1970).

6. NICHOLSON Robert H., Mathematics for Business and Economics, McGraw-Hill Book Company, Singapore, (1986).

7. TÜTEK Hülya H., GÜMÜŞOĞLU Şevkinaz, Sayısal Yöntemler-Yönetsel Yaklaşım, İstanbul, (1994).

8. ÖZTÜRK A., Yöneylem Araştırması, Uludağ Üniversitesi Yayını, Bursa, (1984).

9. YILMAZ Zekai, Sayısal Yöntemler, Uludağ Üniversitesi Yayını, Bursa, (1988).