

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**LİNEER KESİRLİ TAŞIMA PROBLEMLERİNE BİR
ÇÖZÜM ÖNERİSİ**

Matematikçi Seda Esfer GÜR

**FBE Matematik Mühendisliği Anabilim Dalı Matematik Mühendisliği Programında
Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı: Prof. Dr. Mustafa Sivri

İSTANBUL, 2009

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ	iii
ŞEKİL LİSTESİ.....	iv
ÇİZELGE LİSTESİ	v
ÖNSÖZ	vi
1. GİRİŞ	1
2. LİNEER KESİRLİ TAŞIMA PROBLEMİ.....	2
2.1 Lineer Kesirli Taşıma Problemi	2
2.1.1 Lineer kesirli fonksiyonlu programlama.....	4
2.2 Lineer Kesirli Taşıma Problemlerine Bir Çözüm Önerisi	6
2.2.1 Klasik taşıma probleminin tanımı.....	6
2.2.2 Klasik taşıma probleminin algoritması	8
2.2.3 Üç boyutlu taşıma probleminin tanımı	9
2.2.4 Üç boyutlu taşıma probleminin algoritması.....	11
3. UYGULAMA.....	13
3.1 Klasik Taşıma Problemi Uygulaması	13
3.2 Üç Boyutlu Taşıma Problemi Uygulaması	16
4. UYGULAMA PROGRAMI	38
4.1 Programın Tanımı Ve İçeriği	38
4.2 Programın Çözümü	39
5. SONUÇ	66
KAYNAKLAR	68
ÖZGEÇMİŞ.....	69

SİMGE LİSTESİ

$Z(x)$	Amaç fonksiyonu
z	Başlangıç çözümünden elde edilen amaç fonksiyonunun değeri
$\bar{z}$	İterasyon sonucu elde edilen amaç fonksiyonunun değeri
min	Minimum
max	Maksimum
m	Kaynakların sayısı
n	Hedeflerin sayısı
p	Taşıma şekli
c_{ij}	Klasik taşıma probleminde z^1 'in fiyatları
d_{ij}	Klasik taşıma probleminde z^2 'nin fiyatları
c_{ijk}	Üç boyutlu taşıma probleminde z^1 'in fiyatları
d_{ijk}	Üç boyutlu taşıma probleminde z^2 'nin fiyatları
a_i	Her kaynaktaki taşınabilir miktar
b_j	Her hedefe taşınabilir miktar
c_k	Her taşıma şekli için taşınabilir miktar

ŞEKİL LİSTESİ

Şekil 4.1	38
Şekil 5.1	66
Şekil 5.2	67

ÇİZELGE LİSTESİ

Çizelge 3.1.....	13
Çizelge 3.2.....	15
Çizelge 3.3.....	16
Çizelge 3.4.....	16
Çizelge 3.5.....	20
Çizelge 3.6.....	23
Çizelge 3.7.....	26
Çizelge 3.8.....	29
Çizelge 3.9.....	32
Çizelge 3.10.....	35
Çizelge 3.11	37

ÖNSÖZ

Lineer kesirli programlama, sınırlı kaynakların kullanımını optimum kılmak için tasarlanmış bir matematiksel modelleme yöntemidir. Genellikle mühendislik, finans, ekonomi alanlarında kullanılmaktadır. Yöntemin kullanılabilirliği bilgisayar yazılımlarındaki gelişmelerle daha da artmıştır. Bu çalışmamızda lineer kesirli programlama tekniği[5], klasik ve üç boyutlu taşıma problemine uygulanmıştır.

Bu tezin hazırlanmasında yardımlarını esirgemeyen değerli Hocam Sayın Prof. Dr. Mustafa SİVRİ'ye saygılarımı sunar, teşekkürü bir borç bilirim.

LİNEER KESİRLİ TAŞIMA PROBLEMLERİNE BİR ÇÖZÜM ÖNERİSİ

Seda Esfer GÜR
Matematik Mühendisliği, Yüksek Lisans Tezi

Çalışmamızda lineer programlamanın özel bir hali olan lineer kesirli programlama[5], klasik taşıma problemlerine ve üç boyutlu taşıma problemlerine uygulanmıştır ve bununla ilgili örnekler verilmiştir.

Anahtar Kelimeler: Üç boyutlu lineer kesirli taşıma problemi, klasik taşıma problemi, lineer kesirli programlama

JÜRİ:

1. Prof. Dr. Mustafa SİVRİ
2. Prof. Dr. Müfit GİRESUNLU
3. Yrd. Doç. Dr. Nuran GÜZEL

Kabul tarihi: 15.04.2009

Sayfa Sayısı: 67

A SOLUTION OFFER TO LINEAR FRACTIONAL TRANSPORTATION PROBLEMS

Seda Esfer GÜR
Mathematical Engineering, M.S. Thesis

In this study, linear fractional functionals programming[5], a special form of linear programming, is applied to both classical transportation problem and three dimensional transportation problem and related examples are given.

Keywords: Three dimensional linear fractional functionals transportation programming, classic transportation programming, linear fractional programming

JURY:

1. Prof. Dr. Mustafa SİVRİ (Supervisor)
2. Prof. Dr. Müfit GİRESUNLU
3. Yrd. Doç. Dr. Nuran GÜZEL

Date: 15.04.2009

Page: 67

1. GİRİŞ

Taşıma problemi, ilk olarak 1941 yılında F.L. Hitchcock tarafından formülize edilmiş ve daha sonra T.C. Koopmans tarafından geliştirilmiştir. Problemin ilk doğrusal programlama modeli G.B. Dantzing tarafından kurulmuştur. 1953 yılında W.W. Cooper ve A. Charnes atlama taşı yöntemini geliştirmişlerdir.

Taşıma modelleri özde doğrusal programlamanın özel bir durumudur. Genelde doğrusal programlama optimal bir üretim programını belirlemeyi amaçlarken, taşıma modeli de üretilen ürünün tüketim merkezlerine taşımanın toplam maliyetini minimum yapan optimal dağıtım programını belirlemeyi amaçlar. Bu nedenle, taşıma modeli bir ürünün çeşitli üretim merkezlerinden çeşitli tüketim merkezlerine dağıtımını sağlayan farklı taşıma programları arasından en düşük yani minimum maliyetlisini belirlemeyi amaçlayan bir yöntem olarak tanımlanabilir. (Sarıslan, 1986)

Dünya pazarındaki rekabetin ve uluslararası ticaretin her geçen gün arttığı günümüzde, taşımacılık maliyetleri çok kritik bir unsur haline gelmiştir. Dolayısıyla bir çok kısıtın varlığında çözülmesi gereken bu problemi çözecek farklı yaklaşımlara ihtiyaç vardır.

Bu amaçla hazırladığımız çalışmamızın 2. bölümünde lineer kesirli taşıma problemi ve taşıma problemini çözerken faydalandığımız teoriler ve varsayımlar verilmiştir. Çalışmamızın 3. bölümünde ise klasik ve üç boyutlu taşıma problemlerine birer örnek verilmiştir. 4. bölümde ise çözüm önerisi olarak sunduğumuz yöntemin MATLAB programında hazırlanan üç boyutlu taşıma probleminin bilgisayar programı anlatılmış ve program sunulmuştur.

5. bölüm sonuç bölümü olmakla beraber önceki bölümlerde anlatılmış klasik taşıma problemi ve üç boyutlu taşıma problemlerinin çözümü program çıktısı olarak verilmiştir.

2. LİNEER KESİRLİ TAŞIMA PROBLEMİ

2.1 Lineer Kesirli Taşıma Problemi

Lineer kesirli programlama problemleri uygulama alanlarının çokluğu nedeniyle araştırmacılar tarafından büyük ilgi görmüştür. Kesirli programlamanın bir örneği J.R ISBELL VE W.H MARLOW tarafından verilmiştir. (Swarup, 1964)

Daha önceki bir çalışmada CHARNES VE COOPER lineer kesirli fonksiyonlu programlama problemini iki lineer programlama problemi haline getirerek çözmüşlerdir.

$$\text{Max } R(x) = (c'X + \alpha)/(d'X + \beta),$$

$$\begin{array}{ll} \text{Kısıtlar} & \begin{array}{l} AX = b, \\ X \geq 0, \end{array} \end{array} \quad (2.1)$$

- i. X, c ve $d, n \times 1$ vektörlerdir,
- ii. $A, m \times n$ matristir, $A = \|a_{ij}\|, \quad (i=1, \dots, m; j=1, \dots, n)$
- iii. $b, m \times 1$ vektör,
- iv. c', d' c ve d vektörlerinin transpozesidir
- v. α, β sabit skalerlerdir.

(1) Denkleminin kısıtları sürekli olduğundan çözüm kümesi :

$$S = \{X \mid AX = b, X \geq 0\} \text{ kümesi boş olmayan sınırlı bir kümedir.}$$

Bu özellikten dolayı $Y = tX$ dönüşümü yapılarak iki lineer programlama problemi haline şu şekilde gelmiştir:

$$\text{Max } c'Y + \alpha t \quad (2.2)$$

$$\text{Kısıtlar } AY - bt = 0,$$

$$\begin{array}{l} d'Y + \beta t = 1, \\ Y, t \geq 0, \end{array}$$

$$\text{Max } -c'Y - \alpha t, \quad (2.3)$$

$$\text{Kısıtlar } AY - bt = 0$$

$$-d'Y - \beta t = 1,$$

$$Y, t \geq 0.$$

A'nın m kolonu lineer bağımsızdır.

Teorem 1 : $\max R(x)$, $AX=b$ ve $X \geq 0$ 'ın uygun çözümler bölgesinde oluşacaktır.

Yardımcı Teorem: Her Y, t aşağıdaki kısıtları sağlar:

$$AY - bt = 0,$$

$$d'Y + \beta t = \gamma,$$

$$Y, t \geq 0, \gamma \neq 0.$$

İspat: $\text{Max } -c'Y - \alpha t$ ya da $c'Y + \alpha t$ (2.2) ya da (2.3)'ün uygun çözümler kümesinde oluşacaktır.

(2.2) denkleminin bağı olarak $c'Y + \alpha t$ 'yi ele alınsın.

(2.2) Denkleminin herhangi bir uygun çözümler kümesi düzgün S kümesinde sürekli bir $t > 0$ kesinlikle vardır. Artık (2.2) denklemi $(n+1)$ değişken ve $(m+1)$ kısıt içerir. Uygun çözümler kümesinde temel değişkenlerin sayısı $(m+1)$ olacak ve temel değişkenlerden bir tanesi her zaman t olacaktır (S düzgün kümedir). Uygun çözümler kümesinde diğer temel değişkenler $y_1, y_2, \dots, y_m$ olsun.

Bunlar (2.1) denkleminin ilk m değişkeni olsun. B matrisi;

$$B = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mm} \end{bmatrix}$$

$x_1, x_2, \dots, x_m$ ((2.1)'in uygun çözüm kümesini sağlayan $x_k = 0, k = m+1, \dots, n$) değişkenlerine karşılık gelen katsayılarından oluşmuştur.

Buradan (2.2)'nin her temel uygun çözümünün düzgün S kümesiyle uyduğu görülür, (2.1)'in uygun çözümler kümesi $Y=tX$ ile ilişkilidir. (Y^*, t^*) 'in $c'Y + \alpha t$ 'nin optimal çözümü olduğunu

(2.2)'den biliyoruz. O zaman (2.1)'e bağılı olarak $Y^*/t^* = X^*$, $R(x)$ 'in optimal çözümüdür. (Y^*, t^*) (2.2)'nin uygun çözümü ve X^* , (2.1)'in uygun çözümü olacaktır.

2.1.1 Lineer kesirli fonksiyonlu programlama*

Paydanın tüm uygun çözümler kümesi için pozitif olduğu varsayımı altında lineer programlama problemi aşağıdaki gibi olsun:

$$\text{Max } \left(\frac{c'X + \alpha}{d'X + \beta} \right) = R(X)$$

$$\begin{aligned} \text{Kısıtlar } & AX = b \\ & X \geq 0 \end{aligned}$$

$X \geq 0$ olmasının nedeni negatif miktarda kaynağın taşınmasının mümkün olmamasıdır. *

X_B başlangıç uygun çözümümüz olsun, öyle ki;

$$BX_B = b, \text{ ya da}$$

$$X_B = B^{-1}b, \quad B = (b_1, b_2, \dots, b_m),$$

$$X_B \geq 0 \text{ dır.}$$

$z^1 = c'_B X_B + \alpha$ ve $z^2 = d'_B X_B + \beta$ olsun. c'_B ve d'_B elemanları sırasıyla amaç fonksiyonunun payının ve paydasının temel değişkenleriyle alakalı katsayılar vektörüdür. Ek olarak temel uygun çözüm için şu varsayım yapılsın:

$$u_j = B^{-1}a_j,$$

$$z_j^1 = c'_B u_j,$$

$$z_j^2 = d'_B u_j,$$

B 'de olmayan A 'nın her a_j kolonu olarak bilinir.

Elde edilmiş $z = \frac{z^1}{z^2}$ değeriyle başka bir uygun çözüm bulmak istenir. B 'nin sadece bir

* Swarup K., Linear Fractional Functionals Programming, 1964

* J. Reeb & S. Leavengood, Transportation Problem: A Special case for Linear Programming

kolonunda deęiřmiř bu uygun çözümler dikkate alınır.

Yeni uygun çözümler kümesi $\hat{X}_B$ ile gösterilsin. O zaman $\hat{X}_B = \hat{B}^{-1}b$ ve $\hat{B} = (\hat{b}_1, \hat{b}_2, \dots, \hat{b}_m)$ dir. b_γ ları çıkartıp a_j ile yer deęiřtirerek B matrisinden yeni tekil olmayan matris elde edilir. Yeni $\hat{B}$ matrisinin kolonları řöyle verilmiřtir:

$$\begin{aligned}\hat{b}_i &= b_i, \\ \hat{b}_\gamma &= a_j.\end{aligned}\quad (i \neq \gamma)$$

Orijinal řartlarla ve u_{ij} ile yeni temel deęiřkenler elde edilmiřtir:

$$\begin{aligned}\hat{x}_{Bi} &= x_{Bi} - x_{B\gamma}(u_{ij}/u_{\gamma j}), \\ \hat{x}_{B\gamma} &= x_{B\gamma}/u_{\gamma j} = \theta \quad (\text{farz edilsin}) \\ a_j &= \sum_{i=1}^{i=m} u_{ij} b_i.\end{aligned}\quad (i \neq \gamma)$$

Asıl amaç, elde edilmiř amaç fonksiyonunun deęeriyle yeni bir uygun çözüm bulmaktır. Yeni bir uygun çözümün bulunmasıyla z 'nin geliştirilip geliştirilmeyeceęine karar verilmelidir.

Orijinal temel uygun çözüm $z = z^1 / z^2$ dir. Yeni amaç fonksiyonu;

$$\bar{z} = \bar{z}^1 / \bar{z}^2 \text{ olsun.}$$

$$\bar{z}^1 = z^1 + \theta(c_j - z_j^1),$$

$$\bar{z}^2 = z^2 + \theta(d_j - z_j^2) \text{ dir, } z_j^1 \text{ ve } z_j^2 \text{ orijinal temel uygun çözüme aittir.}$$

Amaç fonksiyonunun deęeri ařaęıdaki kořullardan birini saęlıyorsa deęiřecektir:

$$\bar{z} > z \text{ ya da}$$

$$[z^1 + \theta(c_j - z_j^1)]/[z^2 + \theta(d_j - z_j^2)] > z^1 / z^2, \text{ ya da}$$

$$[z^1 + \theta(c_j - z_j^1)]/[z^2 + \theta(d_j - z_j^2)] - z^1 / z^2 > 0, \text{ ya da}$$

$$z^2 [z^1 + \theta(c_j - z_j^1)] - z^1 [z^2 + \theta(d_j - z_j^2)] > 0$$

(z^2 ve $\bar{z}^2$ pozitifdir, çünkü amaç fonksiyonun paydası tüm uygun çözümler için pozitifdir.)

$$z^2 (c_j - z_j^1) - z^1 (d_j - z_j^2) > 0$$

(Dejenere olmayan durumda θ pozitifdir, eğer $\theta = 0$, ise $\bar{z} = z$ dir)

$\Delta_j = z^2(c_j - z_j^1) - z^1(d_j - z_j^2)$ olsun. $\Delta_j = z^2(c_j - z_j^1) - z^1(d_j - z_j^2) > 0$ oldukça iterasyon devam edecek ve tüm Δ_j değerleri için $\Delta_j = z^2(c_j - z_j^1) - z^1(d_j - z_j^2) < 0$ koşulu sağlandığında iterasyon son bulacaktır. Optimal çözüme ulaşılacaktır.

2.2 Lineer Kesirli Taşıma Problemlerine Bir Çözüm Önerisi

2.2.1 Klasik taşıma probleminin tanımı

Amaç fonksiyonu:

$$\min Z(x) = \frac{\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}}{\sum_{i=1}^m \sum_{j=1}^n d_{ij} x_{ij}} \quad (2.4)$$

$$\sum_{j=1}^n x_{ij} = a_i \quad i = 1, \dots, m \quad (2.5)$$

$$\sum_{i=1}^m x_{ij} = b_j \quad j = 1, \dots, n \quad (2.6)$$

$$x_{ij} \geq 0 \quad i = 1, \dots, m \quad j = 1, \dots, n \quad (2.7)$$

$$a_i, b_j > 0 \quad (2.8)$$

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j = T \quad (2.9)$$

m : kaynakların sayısı

n : hedeflerin sayısı

a_i : her kaynaktaki taşınabilir miktar $i = 1, \dots, m$

b_j : her hedefe gerekli miktar $j = 1, \dots, n$

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \geq 0 \text{ ve } \sum_{i=1}^m \sum_{j=1}^n d_{ij} x_{ij} > 0$$

(2.5), (2.6), (2.7), bağıntılarını sağlayan $X = \{x_{ij} \mid i = 1, \dots, m; j = 1, \dots, n\}$ matrisi taşıma planı olarak adlandırılır ve (2.4)'ü sağlıyorsa X optimumdur denir.

(2.9) koşulu sağlandığında, dengeli taşıma problemi denir. (2.9) bağıntısı çözümün olması için gerekli ve yeterli koşuldur. Kısıtlı sistemin matrisinin derecesi $m+n$ dejenere olmayan taşıma problemi (2.4) - (2.9), en az $m+n-1$ sıfır olmayan elemanı olduğunu gösterir.

Başlangıç çözümümüz için $z = z^1 / z^2$ olsun. Amaç, z değerinin küçülterek optimal

$\bar{z} = \bar{z}^1 / \bar{z}^2$ çözüme ulaşmaktır.

$$\bar{z}^1 = z^1 + \theta(c_{ij} - z_{ij}^1)$$

$$\bar{z}^2 = z^2 + \theta(d_{ij} - z_{ij}^2) \text{ dir.}$$

$$z > \bar{z} \Rightarrow \frac{z^1}{z^2} > \frac{\bar{z}^1}{\bar{z}^2}$$

$$\frac{z^1}{z^2} > \frac{z^1 + \theta(c_{ij} - z_{ij}^1)}{z^2 + \theta(d_{ij} - z_{ij}^2)}$$

$$z^1 z^2 + z^1 \theta(d_{ij} - z_{ij}^2) > z^2 z^1 + z^2 \theta(c_{ij} - z_{ij}^1)$$

$$z^1 z^2 + z^1 \theta(d_{ij} - z_{ij}^2) - z^2 z^1 - z^2 \theta(c_{ij} - z_{ij}^1) > 0 \text{ gerekli sadeleştirmeler yapıldıktan sonra;}$$

$$z^1 (d_{ij} - z_{ij}^2) - z^2 (c_{ij} - z_{ij}^1) > 0 \text{ olur. Denklemini (-) ile çarparsak}$$

$$z^2 (c_{ij} - z_{ij}^1) - z^1 (d_{ij} - z_{ij}^2) < 0 \text{ olur. } \Delta_{ij} = z^2 (c_{ij} - z_{ij}^1) - z^1 (d_{ij} - z_{ij}^2) < 0 \text{ oldukça iterasyon}$$

devam edecektir.

2.2.2 Klasik taşıma probleminin algoritması

(2.9) bağıntısına bağlı olarak $m+n-1$ pozitif elemana sahip çözümü sağlayacak şekilde bir başlangıç çözümü oluşturulur.

$$I_x = \{(i, j) \mid x_{ij} > 0, x_{ij} \in X\} \text{ olsun.}$$

Simpleks çarpanlarımız:

$$u_i^1, u_i^2 \quad i = 1, \dots, m \quad \quad \quad v_j^1, v_j^2 \quad j = 1, \dots, n$$

şöyle tanımlanır:

$$c_{ij} = u_i^1 + v_j^1 \quad (2.10)$$

$$d_{ij} = u_i^2 + v_j^2 \quad (2.11)$$

$(i, j) \in I_x$ ve

$$c'_{ij} = c_{ij} - (u_i^1 + v_j^1) \quad \forall (i, j) \quad (2.12)$$

$$d'_{ij} = d_{ij} - (u_i^2 + v_j^2) \quad \forall (i, j) \quad (2.13)$$

(2.10) - (2.11) sistemi bağımsız olarak çözülebilir, bu sistem $m+n-1$ tane denkleme ve $m+n$ değişkene sahip olduğundan keyfi olarak $u_1^1 = 0, v_1^1 = 0$ seçilebilir ve diğer çarpanlar bulunur.

$u_i^1, u_i^2, v_j^1, v_j^2$ 'yi bularak bu değerleri temel olmayan değişkenleri c'_{ij} ve d'_{ij} 'yi bulmak için kullanılır.

$\Delta_{ij} = z^2 (c_{ij} - (u_i^1 + v_j^1)) - z^1 (d_{ij} - (u_i^2 + v_j^2)) > 0$ koşulu optimal çözüme ulaşıldığını gösterir.

$\Delta_{ij} = z^2 (c_{ij} - (u_i^1 + v_j^1)) - z^1 (d_{ij} - (u_i^2 + v_j^2)) < 0$ koşulunu sağlayan X^* değeri için çözüm araştırılmaya devam edilir.

$\Delta_{ij} = z^2 (c_{ij} - (u_i^1 + v_j^1)) - z^1 (d_{ij} - (u_i^2 + v_j^2)) < 0$ koşulunu sağlayan değerler içinden

$$\Delta_{i_0 j_0} = \min \{\Delta_{ij} \mid \Delta_{ij} < 0\} \text{ seçilir.}$$

Seçilen $x_{ij} = t, t > 0$ değeriyle tabana sokularak daha ucuz bir dağıtıma atlanır. x_{ij} boş

hücrelerine $x_{ij} = t > 0$ değeri yazılırsa dengeler bozulur. Sistem denge haline (2.9) koşulu sağlayacak şekile getirilir.

Elde edilen t değerlerinin minimumu seçilerek sisteme atanır ve oluşan $\bar{z} = \frac{\bar{z}^1}{\bar{z}^2}$ değerinin başlangıç değerinden $z = \frac{z^1}{z^2}$ 'den küçük olduğu kontrolü yapılarak iterasyonlara devam edilir.

$\forall i, j$ için $\Delta_{ij} = z^2 (c_{ij} - (u_i^1 + v_j^1)) - z^1 (d_{ij} - (u_i^2 + v_j^2)) > 0$ optimallik koşulu sağlanıncaya kadar devam edilir.

2.2.3 Üç boyutlu taşıma probleminin tanımı

Amaç fonksiyonu:

$$\min Z(x) = \frac{\sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^p c_{ijk} x_{ijk}}{\sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^p d_{ijk} x_{ijk}} \quad (2.14)$$

$$\sum_{j=1}^n \sum_{k=1}^p x_{ijk} = a_i \quad i = 1, \dots, m \quad (2.15)$$

$$\sum_{i=1}^m \sum_{k=1}^p x_{ijk} = b_j \quad j = 1, \dots, n \quad (2.16)$$

$$\sum_{i=1}^m \sum_{j=1}^n x_{ijk} = c_k \quad k = 1, \dots, p \quad (2.17)$$

$$x_{ijk} \geq 0 \quad i = 1, \dots, m \quad j = 1, \dots, n \quad k = 1, \dots, p \quad (2.18)$$

$$a_i, b_j, c_k > 0 \quad (2.19)$$

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j = \sum_{k=1}^p c_k = T \quad (2.20)$$

m: kaynakların sayısı

n: hedeflerin(pazarın) sayısı

p: taşıma şekli sayısı

a_i : her kaynaktaki taşınabilir miktar $i = 1, \dots, m$

b_j : her hedefe gerekli miktar $j = 1, \dots, n$

c_k : her taşıma şekliyle taşınması gereken miktar $k = 1, \dots, p$

$$\sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^p c_{ijk} x_{ijk} \geq 0 \text{ ve } \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^p d_{ijk} x_{ijk} > 0$$

(2.15), (2.16), (2.17), (2.18) bağıntılarını sağlayan $X = \{x_{ijk} \mid i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, p\}$ matrisi taşıma planı olarak adlandırılır ve (2.14)'ü sağlıyorsa X optimumdur denir.

(2.20) koşulu sağlandığında, dengeli taşıma problemi denir. (2.20) bağıntısı çözümün olması için gerekli ve yeterli koşuldur. Kısıtlı sistemin matrisinin derecesi $m+n+p$ dejenere olmayan taşıma problemi (2.14) – (2.20)'nin en az $m+n+p-2$ sıfır olmayan elemanı olduğunu gösterir. (Moanta, 2006)

Başlangıç çözümümüz için $z = z^1 / z^2$ olsun. Amacımız z değerimizi küçülterek optimal

$\bar{z} = \bar{z}^1 / \bar{z}^2$ çözüme ulaşmaktır.

$$\bar{z}^1 = z^1 + \theta(c_{ijk} - z_{ijk}^1)$$

$$\bar{z}^2 = z^2 + \theta(d_{ijk} - z_{ijk}^2) \text{ 'dir.}$$

$$z > \bar{z} \Rightarrow \frac{z^1}{z^2} > \frac{\bar{z}^1}{\bar{z}^2}$$

$$\frac{z^1}{z^2} > \frac{z^1 + \theta(c_{ijk} - z_{ijk}^1)}{z^2 + \theta(d_{ijk} - z_{ijk}^2)}$$

$$z^1 z^2 + z^1 \theta(d_{ijk} - z_{ijk}^2) > z^2 z^1 + z^2 \theta(c_{ijk} - z_{ijk}^1)$$

$$z^1 z^2 + z^1 \theta(d_{ijk} - z_{ijk}^2) - z^2 z^1 - z^2 \theta(c_{ijk} - z_{ijk}^1) > 0 \text{ gerekli sadeleştirmeler yapıldıktan sonra;}$$

$$z^2 (c_{ijk} - z_{ijk}^1) - z^1 (d_{ijk} - z_{ijk}^2) < 0 \text{ olur. } \Delta_{ijk} = z^2 (c_{ijk} - z_{ijk}^1) - z^1 (d_{ijk} - z_{ijk}^2) < 0 \text{ oldukça}$$

iterasyon devam edecektir. Tüm Δ_{ijk} değerleri $\Delta_{ijk} > 0$ olduğunda optimum değere ulaşılmış olacaktır.

2.2.4 Üç boyutlu taşıma probleminin algoritması

(2.20) bağıntısına bağlı olarak $m+n+p-2$ pozitif elemana sahip çözümü sağlayacak şekilde bir başlangıç çözümü oluşturulur.

$$I_x = \{(i, j, k) \mid x_{ijk} > 0, x_{ijk} \in X\} \text{ olsun.}$$

Simpleks çarpanlar:

$$u_i^1, u_i^2 \quad i = 1, \dots, m \quad \quad \quad v_j^1, v_j^2 \quad j = 1, \dots, n \quad \quad \quad w_k^1, w_k^2 \quad k = 1, \dots, p$$

şöyle tanımlanır:

$$c_{ijk} = u_i^1 + v_j^1 + w_k^1 \quad (2.21)$$

$$d_{ijk} = u_i^2 + v_j^2 + w_k^2 \quad (2.22)$$

$(i, j, k) \in I_x$ ve

$$c'_{ijk} = c_{ijk} - (u_i^1 + v_j^1 + w_k^1) \quad \forall (i, j, k) \quad (2.23)$$

$$d'_{ijk} = d_{ijk} - (u_i^2 + v_j^2 + w_k^2) \quad \forall (i, j, k) \quad (2.24)$$

(2.21) - (2.22) sistemi bağımsız olarak çözülebilir, bu sistem $m+n+p-2$ tane denkleme ve $m+n+p$ değişkene sahip olduğundan keyfi olarak $u_1^1 = 0, v_1^1 = 0$ seçilebilir ve diğer çarpanlar bulunur.

$u_i^1, u_i^2, v_j^1, v_j^2, w_k^1, w_k^2$ 'yi bularak bu değerleri temel olmayan değişkenler için kullanarak c'_{ijk} ve d'_{ijk} 'yi bulmak için kullanılır. (Moanta, 2006)

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ koşulu optimal çözüme ulaşıldığını gösterir. $\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) < 0$ koşulunu sağlayan X^* değeri için çözüm araştırılmaya devam edilir.

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) < 0$ koşulunu sağlayan değerler içinden $\Delta_{i_0 j_0 k_0} = \min \{\Delta_{ijk} \mid \Delta_{ijk} < 0\}$ seçilir.

Seçilen $x_{ijk} = t$, $t > 0$ değeriyle tabana sokularak daha ucuz bir dağıtıma atlanır. x_{ijk} boş hücresine $x_{ijk} = t > 0$ değeri yazılırsa dengeler bozulur. Sistem denge haline (2.20) koşulu sağlayacak şekile getirilir.

Elde edilen t değerlerinin minimumu seçilerek sisteme atanır ve oluşan $\bar{z} = \frac{\bar{z}^1}{\bar{z}^2}$ değerinin

başlangıç değeri $z = \frac{z^1}{z^2}$ 'den küçük olduğu kontrolü yapılarak iterasyonlara devam edilir.

Tüm Δ_{ijk} 'lar için $\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik koşulu sağlanıncaya kadar iterasyonlara devam edilir.

3. UYGULAMA

3.1 Klasik Taşıma Problemi Uygulaması

$m=3, n=3$

$$a_1=12 \quad b_1=10$$

$$a_2=13 \quad b_2=20$$

$$a_3=25 \quad b_3=20$$

Çizelge 3.1

1 10 7	3 2 8	7 3
8 4	2 13 5	6 8
10 11	9 5 3	5 20 7

$m+n-1$ denklemden yola çıkarak, $3+3-1=5$ atamayla bir başlangıç çözümü ele alınır.

$$z^1 = 1*10 + 3*2 + 2*13 + 9*5 + 5*20 = 187$$

$$z^2 = 10*7 + 2*8 + 13*5 + 5*3 + 20*7 = 306$$

$$\bar{z} = \frac{z^1}{z^2} = 0.6111$$

Simpleks çarpanlar bulunur:

$$u_1^1 + v_1^1 = 1$$

$$u_1^2 + v_1^2 = 7$$

$$u_1^1 + v_2^1 = 3$$

$$u_1^2 + v_2^2 = 8$$

$$u_2^1 + v_2^1 = 2$$

$$u_2^2 + v_2^2 = 5$$

(3.1)

$$u_3^1 + v_2^1 = 9$$

$$u_3^2 + v_2^2 = 3$$

$$u_3^1 + v_3^1 = 5$$

$$u_3^2 + v_3^2 = 7$$

(3.1) denklem sisteminden simpleks çarpanlar şöyledir:

$$u_1^1 = 0$$

$$v_1^1 = 1$$

$$u_1^2 = 0$$

$$v_1^2 = 7$$

$$u_2^1 = -1$$

$$v_2^1 = 3$$

$$u_2^2 = -3$$

$$v_2^2 = 8$$

$$u_3^1 = 6$$

$$v_3^1 = -1$$

$$u_3^2 = -5$$

$$v_3^2 = 12$$

Δ_{ij} optimallik koşulunun sağlanıp sağlanmadığına bakılır:

$$z^2(7 - (u_1^1 + v_1^1)) - z^1(3 - (u_1^2 + v_1^2)) = 6z^2 + 4z^1 = 2584$$

$$z^2(8 - (u_2^1 + v_2^1)) - z^1(4 - (u_2^2 + v_2^2)) = 8z^2 = 2448$$

$$z^2(6 - (u_2^1 + v_3^1)) - z^1(8 - (u_2^2 + v_3^2)) = 8z^2 + z^1 = 2635$$

$$z^2(10 - (u_3^1 + v_1^1)) - z^1(11 - (u_3^2 + v_1^2)) = 3z^2 - 9z^1 = -765 \Rightarrow$$

$\min(\Delta_{ij}) = -765$ olduğundan u_3, v_1 kutusuna t değişkeni atanır. Tablonun yeni hali:

1 10-t 7	3 2+t 8	7 3
8 4	2 13 5	6 8
10 t 11	9 5-t 3	5 20 7

$10-t=0, 5-t=0$ denklemlerinde $t=\{10, 5\}$ olur. $\min(t)=5$ olduğundan $t=5$ dir.

Çizelge 3.2

1 5 7	3 7 8	7 3
8 4	2 13 5	6 8
10 5 11	9 3	5 20 7

$$z^1 = 1*5 + 3*7 + 2*13 + 10*5 + 5*20 = 202$$

$$z^2 = 5*7 + 7*8 + 13*5 + 5*11 + 20*7 = 351$$

$$\bar{z} = \frac{z^1}{z^2} = 0.5755$$

Simpleks çarpanlar bulunur:

$$\begin{aligned}
 u_1^1 + v_1^1 &= 1 & u_1^2 + v_1^2 &= 7 \\
 u_1^1 + v_2^1 &= 3 & u_1^2 + v_2^2 &= 8 \\
 u_2^1 + v_2^1 &= 2 & u_2^2 + v_2^2 &= 5 \\
 u_3^1 + v_1^1 &= 10 & u_3^2 + v_1^2 &= 11 \\
 u_3^1 + v_3^1 &= 5 & u_3^2 + v_3^2 &= 7
 \end{aligned} \tag{3.2}$$

(3.2) denklem sisteminden simpleks çarpanlar şöyledir:

$$\begin{aligned}
 u_1^1 &= 0 & v_1^1 &= 1 & u_1^2 &= 0 & v_1^2 &= 7 \\
 u_2^1 &= -1 & v_2^1 &= 3 & u_2^2 &= -3 & v_2^2 &= 8 \\
 u_3^1 &= 9 & v_3^1 &= -4 & u_3^2 &= 4 & v_3^2 &= 3
 \end{aligned}$$

Δ_{ij} optimallik koşulunun sağlanıp sağlanmadığına bakılır:

$$z^2(7 - (u_1^1 + v_1^1)) - z^1(3 - (u_1^2 + v_1^2)) = 11z^2 = 3861$$

$$z^2(8 - (u_2^1 + v_1^1)) - z^1(4 - (u_2^2 + v_1^2)) = 8z^2 = 2808$$

$$z^2(6 - (u_2^1 + v_3^1)) - z^1(8 - (u_2^2 + v_3^2)) = 11z^2 - 8z^1 = 2245$$

$$z^2(10 - (u_3^1 + v_1^1)) - z^1(11 - (u_3^2 + v_1^2)) = -3z^2 + 9z^1 = 765$$

Tüm Δ_{ij} değerleri $\Delta_{ij} \geq 0$ olduğundan optimal çözüme ulaşılmıştır.

Çizelge 3.3

iterasyon	$\bar{z}$
1	0.6111
2	0.5755

3.2 Üç Boyutlu Taşıma Problemi Uygulaması

m=3, n=3, p=3 ve

$$a_1=12 \quad b_1=40 \quad c_1=24$$

$$a_2=25 \quad b_2=5 \quad c_2=20$$

$$a_3=23 \quad b_3=15 \quad c_3=16$$

Çizelge 3.4

11 12 12	6 4	8 14
4 12 21	7 22	8 14
12 9	5 2	6 6
14 10	11 18	3 8
9 7	10 10	7 2
5 7 13	4 24	9 26
15 5	4 15	14 22
21 12	23 4	15 16
16 9 13	19 14	18 7 16

m+n+p-2 denkleminde yola çıkarak, 3+3+3-2=7 atamayla bir başlangıç çözümü alınır.

$$z^1 = 11*12 + 4*12 + 5*7 + 10*5 + 7*8 + 16*9 + 18*7 = 591$$

$$z^2 = 12*12 + 12*21 + 7*13 + 5*10 + 8*2 + 9*13 + 7*16 = 782$$

$$\bar{z} = \frac{z^1}{z^2} = 0.7558$$

Simpleks arpanlar bulunur:

$$\begin{array}{ll}
 u_1^1 + v_1^1 + w_1^1 = 11 & u_1^2 + v_1^2 + w_1^2 = 12 \\
 u_2^1 + v_1^1 + w_1^1 = 4 & u_2^2 + v_1^2 + w_1^2 = 21 \\
 u_2^1 + v_2^1 + w_2^1 = 10 & u_2^2 + v_2^2 + w_2^2 = 10 \\
 u_2^1 + v_3^1 + w_2^1 = 7 & u_2^2 + v_3^2 + w_2^2 = 2 \\
 u_3^1 + v_1^1 + w_2^1 = 5 & u_3^2 + v_1^2 + w_2^2 = 13 \\
 u_3^1 + v_1^1 + w_3^1 = 16 & u_3^2 + v_1^2 + w_3^2 = 13 \\
 u_3^1 + v_3^1 + w_3^1 = 18 & u_3^2 + v_3^2 + w_3^2 = 16
 \end{array} \tag{3.3}$$

(3.3) denklem sisteminden ařağıdaki sonular elde edilir:

$$\begin{array}{lll}
 u_1^1 = 0 & v_1^1 = 0 & w_1^1 = 11 \\
 u_2^1 = -7 & v_2^1 = 5 & w_2^1 = 12 \\
 u_3^1 = -7 & v_3^1 = 2 & w_3^1 = 23
 \end{array}
 \quad
 \begin{array}{lll}
 u_1^2 = 0 & v_1^2 = 0 & w_1^2 = 12 \\
 u_2^2 = 9 & v_2^2 = 11 & w_2^2 = -10 \\
 u_3^2 = 23 & v_3^2 = 3 & w_3^2 = -10
 \end{array}$$

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik kořulunu saėlayıp saėlamadıėına bakılır.

$$z^2 (6 - (u_1^1 + v_2^1 + w_1^1)) - z^1 (4 - (u_1^2 + v_2^2 + w_1^2)) = -10z^2 + 19z^1 = 3409$$

$$z^2 (8 - (u_1^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_1^2 + v_3^2 + w_1^2)) = -5z^2 + 1z^1 = -3319$$

$$z^2 (7 - (u_2^1 + v_2^1 + w_1^1)) - z^1 (22 - (u_2^2 + v_2^2 + w_1^2)) = -2z^2 + z^1 = -973$$

$$z^2 (8 - (u_2^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_2^2 + v_3^2 + w_1^2)) = 2z^2 + 10z^1 = 7474$$

$$z^2 (12 - (u_3^1 + v_1^1 + w_1^1)) - z^1 (9 - (u_3^2 + v_1^2 + w_1^2)) = 8z^2 - 5z^1 = 3301$$

$$z^2(5-(u_3^1+v_2^1+w_1^1))-z^1(2-(u_3^2+v_2^2+w_1^2))=-4z^2+44z^1=22876$$

$$z^2(6-(u_3^1+v_3^1+w_1^1))-z^1(6-(u_3^2+v_3^2+w_1^2))=32z^1=18912$$

$$z^2(14-(u_1^1+v_1^1+w_2^1))-z^1(10-(u_1^2+v_1^2+w_2^2))=2z^2-20z^1=-10256$$

$$z^2(11-(u_1^1+v_2^1+w_2^1))-z^1(18-(u_1^2+v_2^2+w_2^2))=-6z^2-17z^1=-14739$$

$$z^2(3-(u_1^1+v_3^1+w_2^1))-z^1(8-(u_1^2+v_3^2+w_2^2))=-11z^2-15z^1=-17467$$

$$z^2(9-(u_2^1+v_1^1+w_2^1))-z^1(7-(u_2^2+v_1^2+w_2^2))=4z^2-8z^1=-1600$$

$$z^2(4-(u_3^1+v_2^1+w_2^1))-z^1(24-(u_3^2+v_2^2+w_2^2))=-6z^2=-4692$$

$$z^2(9-(u_3^1+v_3^1+w_2^1))-z^1(26-(u_3^2+v_3^2+w_2^2))=2z^2-10z^1=-4346$$

$$z^2(15-(u_1^1+v_1^1+w_3^1))-z^1(5-(u_1^2+v_1^2+w_3^2))=-8z^2-15z^1=-15121$$

$$z^2(4-(u_1^1+v_2^1+w_3^1))-z^1(15-(u_1^2+v_2^2+w_3^2))=-24z^2-14z^1=-27042 \Rightarrow$$

$$z^2(14-(u_1^1+v_3^1+w_3^1))-z^1(22-(u_1^2+v_3^2+w_3^2))=-11z^2-29z^1=-25741$$

$$z^2(21-(u_2^1+v_1^1+w_3^1))-z^1(12-(u_2^2+v_1^2+w_3^2))=5z^2-13z^1=-3773$$

$$z^2(23-(u_2^1+v_2^1+w_3^1))-z^1(4-(u_2^2+v_2^2+w_3^2))=2z^2+6z^1=5110$$

$$z^2(15-(u_2^1+v_3^1+w_3^1))-z^1(16-(u_2^2+v_3^2+w_3^2))=-3z^2-14z^1=-10620$$

$$z^2(19-(u_3^1+v_2^1+w_3^1))-z^1(14-(u_3^2+v_2^2+w_3^2))=-2z^2+10z^1=4346$$

$\min(\Delta_{ijk}) = -27042$ olduğundan u_1, v_2, w_3 kutusuna t değişkeni atanır. Tablonun yeni hali:

11 12-t 12	6 4	8 14
4 12+t 21	7 22	8 14
12 9	5 2	6 6

14 10	11 18	3 8
9 7	10 5-t 10	7 8 2
5 7+t 13	4 24	9 26

15 5	4 t 15	14 22
21 12	23 4	15 16
16 9-t 13	19 14	18 7 16

$2-t=0$, $5-t=0$, $9-t=0$ denklemlerinden $t=\{12, 5, 9\}$ tür. $\min(t)=5$ olduğundan $t=5$ olur.

Çizelge 3.5

11 7 12	6 4	8 14
4 17 21	7 22	8 14
12 9	5 2	6 6

14 10	11 18	3 8
9 7	10 10	7 8 2
5 12 13	4 24	9 26

15 5	4 5 15	14 22
21 12	23 4	15 16
16 4 13	19 14	18 7 16

$$z^1 = 11*7 + 4*17 + 5*12 + 7*8 + 16*4 + 4*5 + 18*7 = 471$$

$$z^2 = 7*12 + 17*21 + 12*13 + 8*2 + 5*15 + 4*13 + 7*16 = 852$$

$\bar{z} = 0,5528$, $z > \bar{z}$ koşulu sağlandı.

Simpleks çarpanlar bulunur:

$$u_1^1 + v_1^1 + w_1^1 = 11$$

$$u_1^2 + v_1^2 + w_1^2 = 12$$

$$u_2^1 + v_1^1 + w_1^1 = 4$$

$$u_2^2 + v_1^2 + w_1^2 = 21$$

$$u_2^1 + v_3^1 + w_2^1 = 7$$

$$u_2^2 + v_3^2 + w_2^2 = 2$$

(3.4)

$$u_3^1 + v_1^1 + w_2^1 = 5$$

$$u_3^2 + v_1^2 + w_2^2 = 13$$

$$u_1^1 + v_2^1 + w_3^1 = 4$$

$$u_1^2 + v_2^2 + w_3^2 = 15$$

$$u_3^1 + v_1^1 + w_3^1 = 16$$

$$u_3^2 + v_1^2 + w_3^2 = 13$$

$$u_3^1 + v_3^1 + w_3^1 = 18$$

$$u_3^2 + v_3^2 + w_3^2 = 16$$

(3.4) denklem sisteminden aşağıdaki sonuçlar elde edilir:

$$\begin{array}{lll}
 u_1^1 = 0 & v_1^1 = 0 & w_1^1 = 11 \\
 u_2^1 = -7 & v_2^1 = -19 & w_2^1 = 12 \\
 u_3^1 = -7 & v_3^1 = 2 & w_3^1 = 23
 \end{array}
 \qquad
 \begin{array}{lll}
 u_1^2 = 0 & v_1^2 = 0 & w_1^2 = 12 \\
 u_2^2 = 9 & v_2^2 = 25 & w_2^2 = -10 \\
 u_3^2 = 23 & v_3^2 = 3 & w_3^2 = -10
 \end{array}$$

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik koşulunu sağlayıp sağlamadığına bakılır.

$$z^2 (6 - (u_1^1 + v_2^1 + w_1^1)) - z^1 (4 - (u_1^2 + v_2^2 + w_1^2)) = -14z^2 + 33z^1 = 27471$$

$$z^2 (8 - (u_1^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_1^2 + v_3^2 + w_1^2)) = -5z^2 + 1z^1 = -3789$$

$$z^2 (7 - (u_2^1 + v_2^1 + w_1^1)) - z^1 (22 - (u_2^2 + v_2^2 + w_1^2)) = 22z^2 + 24z^1 = 30048$$

$$z^2 (8 - (u_2^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_2^2 + v_3^2 + w_1^2)) = 2z^2 + 10z^1 = 6414$$

$$z^2 (12 - (u_3^1 + v_1^1 + w_1^1)) - z^1 (9 - (u_3^2 + v_1^2 + w_1^2)) = 8z^2 + 26z^1 = 19062$$

$$z^2 (5 - (u_3^1 + v_2^1 + w_1^1)) - z^1 (2 - (u_3^2 + v_2^2 + w_1^2)) = 20z^2 + 58z^1 = 44358$$

$$z^2 (6 - (u_3^1 + v_3^1 + w_1^1)) - z^1 (6 - (u_3^2 + v_3^2 + w_1^2)) = 32z^1 = 15072$$

$$z^2 (14 - (u_1^1 + v_1^1 + w_2^1)) - z^1 (10 - (u_1^2 + v_1^2 + w_2^2)) = 2z^2 - 20z^1 = -7716$$

$$z^2 (11 - (u_1^1 + v_2^1 + w_2^1)) - z^1 (18 - (u_1^2 + v_2^2 + w_2^2)) = 18z^2 - 3z^1 = 13923$$

$$z^2 (3 - (u_1^1 + v_3^1 + w_2^1)) - z^1 (8 - (u_1^2 + v_3^2 + w_2^2)) = -11z^2 - 15z^1 = -16437$$

$$z^2 (9 - (u_2^1 + v_1^1 + w_2^1)) - z^1 (7 - (u_2^2 + v_1^2 + w_2^2)) = 4z^2 - 8z^1 = -360$$

$$z^2 (10 - (u_2^1 + v_2^1 + w_2^1)) - z^1 (10 - (u_2^2 + v_2^2 + w_2^2)) = 24z^2 + 14z^1 = 27042$$

$$z^2 (4 - (u_3^1 + v_2^1 + w_2^1)) - z^1 (24 - (u_3^2 + v_2^2 + w_2^2)) = 18z^2 - 17z^1 = 21930$$

$$z^2 (9 - (u_3^1 + v_3^1 + w_2^1)) - z^1 (26 - (u_3^2 + v_3^2 + w_2^2)) = 2z^2 - 10z^1 = -3006$$

$$z^2 (15 - (u_1^1 + v_1^1 + w_3^1)) - z^1 (5 - (u_1^2 + v_1^2 + w_3^2)) = -8z^2 - 15z^1 = -13881$$

$$z^2(14 - (u_1^1 + v_3^1 + w_3^1)) - z^1(22 - (u_1^2 + v_3^2 + w_3^2)) = -11z^2 - 29z^1 = -23031 \Rightarrow$$

$$z^2(21 - (u_2^1 + v_1^1 + w_3^1)) - z^1(12 - (u_2^2 + v_1^2 + w_3^2)) = 5z^2 - 13z^1 = -1863$$

$$z^2(23 - (u_2^1 + v_2^1 + w_3^1)) - z^1(4 - (u_2^2 + v_2^2 + w_3^2)) = 26z^2 + 20z^1 = 31572$$

$$z^2(15 - (u_2^1 + v_3^1 + w_3^1)) - z^1(16 - (u_2^2 + v_3^2 + w_3^2)) = -3z^2 - 14z^1 = -9150$$

$$z^2(19 - (u_3^1 + v_2^1 + w_3^1)) - z^1(14 - (u_3^2 + v_2^2 + w_3^2)) = 22z^2 + 24z^1 = 30048$$

$\min(\Delta_{ijk}) = -23031$ olduğundan u_1, v_3, w_3 kutusuna t değişkeni atanır. Tablonun yeni hali:

11 7-t 12	6 4	8 14
4 17+t 21	7 22	8 14
12 9	5 2	6 6

14 10	11 18	3 8
9 7	10 10	7 8-t 2
5 12+t 13	4 24	9 26

15 5	4 5 15	14 t 22
21 12	23 4	15 16
16 4-t 13	19 14	18 7 16

7-t=0, 8-t=0 ve 4-t=0 denklemlerinden $t=\{7, 8, 4\}$ tür. $\text{Min}(t)=4$ olduğundan $t=4$ olur.

Çizelge 3.6

11 3 12	6 4	8 14	14 10	11 18	3 8
4 21 21	7 22	8 14	9 7	10 10	7 4 2
12 9	5 2	6 6	5 16 13	4 24	9 26

15 5	4 5 15	14 4 22
21 12	23 4	15 16
16 13	19 14	18 7 16

$$z^1 = 11*3 + 4*21 + 5*16 + 7*4 + 4*5 + 14*4 + 18*7 = 427$$

$$z^2 = 3*12 + 21*21 + 16*13 + 4*2 + 5*15 + 4*22 + 7*16 = 968$$

$$\bar{z} = 0,4411$$

Simpleks çarpanlar bulunur:

$$u_1^1 + v_1^1 + w_1^1 = 11$$

$$u_1^2 + v_1^2 + w_1^2 = 12$$

$$u_2^1 + v_1^1 + w_1^1 = 4$$

$$u_2^2 + v_1^2 + w_1^2 = 21$$

$$u_2^1 + v_3^1 + w_2^1 = 7$$

$$u_2^2 + v_3^2 + w_2^2 = 2$$

(3.5)

$$u_3^1 + v_1^1 + w_2^1 = 5$$

$$u_3^2 + v_1^2 + w_2^2 = 13$$

$$u_1^1 + v_2^1 + w_3^1 = 4$$

$$u_1^2 + v_2^2 + w_3^2 = 15$$

$$u_1^1 + v_3^1 + w_3^1 = 14$$

$$u_1^2 + v_3^2 + w_3^2 = 22$$

$$u_3^1 + v_3^1 + w_3^1 = 18$$

$$u_3^2 + v_3^2 + w_3^2 = 16$$

(3.5) denklem sisteminden aşağıdaki sonuçlar elde edilir:

$$u_1^1 = 0 \quad v_1^1 = 0 \quad w_1^1 = 11$$

$$u_1^2 = 0 \quad v_1^2 = 0 \quad w_1^2 = 12$$

$$u_2^1 = -7 \quad v_2^1 = 3 \quad w_2^1 = 1$$

$$u_2^2 = 9 \quad v_2^2 = -33 \quad w_2^2 = 19$$

$$u_3^1 = 4 \quad v_3^1 = 13 \quad w_3^1 = 1$$

$$u_3^2 = -6 \quad v_3^2 = -26 \quad w_3^2 = 48$$

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik koşulunu sağlayıp sağlamadığına bakılır.

$$z^2 (6 - (u_1^1 + v_2^1 + w_1^1)) - z^1 (4 - (u_1^2 + v_2^2 + w_1^2)) = -8z^2 - 25z^1 = -18419$$

$$z^2 (8 - (u_1^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_1^2 + v_3^2 + w_1^2)) = -16z^2 - 28z^1 = -27444$$

$$z^2 (7 - (u_2^1 + v_2^1 + w_1^1)) - z^1 (22 - (u_2^2 + v_2^2 + w_1^2)) = -34z^1 = -14518$$

$$z^2 (8 - (u_2^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_2^2 + v_3^2 + w_1^2)) = -9z^2 - 19z^1 = -16825$$

$$z^2 (12 - (u_3^1 + v_1^1 + w_1^1)) - z^1 (9 - (u_3^2 + v_1^2 + w_1^2)) = -3z^2 - 3z^1 = -4185$$

$$z^2 (5 - (u_3^1 + v_2^1 + w_1^1)) - z^1 (2 - (u_3^2 + v_2^2 + w_1^2)) = -13z^2 - 29z^1 = -24967$$

$$z^2 (6 - (u_3^1 + v_3^1 + w_1^1)) - z^1 (6 - (u_3^2 + v_3^2 + w_1^2)) = -22z^2 - 26z^1 = -32398 \Rightarrow$$

$$z^2 (14 - (u_1^1 + v_1^1 + w_2^1)) - z^1 (10 - (u_1^2 + v_1^2 + w_2^2)) = 13z^2 + 9z^1 = 16427$$

$$z^2 (11 - (u_1^1 + v_2^1 + w_2^1)) - z^1 (18 - (u_1^2 + v_2^2 + w_2^2)) = 7z^2 - 33z^1 = 6888$$

$$z^2 (3 - (u_1^1 + v_3^1 + w_2^1)) - z^1 (8 - (u_1^2 + v_3^2 + w_2^2)) = -11z^2 - 15z^1 = -17053$$

$$z^2 (9 - (u_2^1 + v_1^1 + w_2^1)) - z^1 (7 - (u_2^2 + v_1^2 + w_2^2)) = 15z^2 + 21z^1 = 23487$$

$$z^2 (10 - (u_2^1 + v_2^1 + w_2^1)) - z^1 (10 - (u_2^2 + v_2^2 + w_2^2)) = 13z^2 - 15z^1 = 6179$$

$$z^2(4 - (u_3^1 + v_2^1 + w_2^1)) - z^1(24 - (u_3^2 + v_2^2 + w_2^2)) = -4z^2 - 44z^1 = -22660$$

$$z^2(9 - (u_3^1 + v_3^1 + w_2^1)) - z^1(26 - (u_3^2 + v_3^2 + w_2^2)) = -9z^2 - 39z^1 = -25365$$

$$z^2(15 - (u_1^1 + v_1^1 + w_3^1)) - z^1(5 - (u_1^2 + v_1^2 + w_3^2)) = 14z^2 + 43z^1 = 31913$$

$$z^2(14 - (u_1^1 + v_3^1 + w_3^1)) - z^1(22 - (u_1^2 + v_3^2 + w_3^2)) = 27z^2 + 45z^1 = 45351$$

$$z^2(23 - (u_2^1 + v_2^1 + w_3^1)) - z^1(4 - (u_2^2 + v_2^2 + w_3^2)) = 26z^2 + 20z^1 = 33708$$

$$z^2(15 - (u_2^1 + v_3^1 + w_3^1)) - z^1(16 - (u_2^2 + v_3^2 + w_3^2)) = 8z^2 + 15z^1 = 14149$$

$$z^2(16 - (u_3^1 + v_1^1 + w_3^1)) - z^1(13 - (u_3^2 + v_1^2 + w_3^2)) = 11z^2 + 29z^1 = 23031$$

$$z^2(19 - (u_3^1 + v_2^1 + w_3^1)) - z^1(14 - (u_3^2 + v_2^2 + w_3^2)) = 11z^2 - 5z^1 = 8513$$

$\min(\Delta_{ijk}) = -32398$ olduğundan u_3, v_3, w_1 kutusuna t değişkeni atanır. Tablonun yeni hali:

11 3-2t 12	6 4	8 14
4 21+t 21	7 22	8 14
12 9	5 2	6 t 6

14 10	11 18	3 8
9 7	10 10	7 4-t 2
5 16+t 13	4 24	9 26

15 5	4 5 15	14 4+2t 22
21 12	23 4	15 16
16 13	19 14	18 7-2t 16

$3-2t=0$, $4-t=0$, denklemlerinden $t=\{1.5, 4\}$ tür. $\min(t)=1.5$ olduğundan $t=1.5$ olur.

Çizelge 3.7

11	6	8	14	14	11	3
12	4	14	10	18	8	
4	7	8	9	10	7	2.5
22.5	22	14	7	10	2	
12	5	6	5	4	9	
9	2	1.5	17.5	24		26
		6	13			

15	4	14
5	5	7
21	23	15
12	4	16
16	19	18
13	14	4
		16

$$z^1 = 4 * 22,5 + 6 * 1,5 + 5 * 17,5 + 7 * 2,5 + 4 * 5 + 14 * 7 + 18 * 4 = 394$$

$$z^2 = 22,5 * 21 + 1,5 * 6 + 17,5 * 13 + 2,5 * 2 + 5 * 15 + 7 * 22 + 4 * 16 = 1007$$

$$\bar{z} = 0.3913$$

Simpleks çarpanlar bulunur:

$$u_2^1 + v_1^1 + w_1^1 = 4$$

$$u_2^2 + v_1^2 + w_1^2 = 21$$

$$u_3^1 + v_3^1 + w_1^1 = 6$$

$$u_3^2 + v_3^2 + w_1^2 = 6$$

$$u_2^1 + v_3^1 + w_2^1 = 7$$

$$u_2^2 + v_3^2 + w_2^2 = 2$$

$$u_3^1 + v_1^1 + w_2^1 = 5$$

$$u_3^2 + v_1^2 + w_2^2 = 13$$

(3.6)

$$u_1^1 + v_2^1 + w_3^1 = 4$$

$$u_1^2 + v_2^2 + w_3^2 = 15$$

$$u_1^1 + v_3^1 + w_3^1 = 14$$

$$u_1^2 + v_3^2 + w_3^2 = 22$$

$$u_3^1 + v_3^1 + w_3^1 = 18$$

$$u_3^2 + v_3^2 + w_3^2 = 16$$

(3.6) denklem sisteminden aşağıdaki sonuçlar elde edilir:

$$\begin{array}{llll}
 u_1^1 = 0 & v_1^1 = 0 & w_1^1 = 0 & u_1^2 = 0 \quad v_1^2 = 0 \quad w_1^2 = 25 \\
 u_2^1 = 4 & v_2^1 = -8 & w_2^1 = 1 & u_2^2 = -4 \quad v_2^2 = -20 \quad w_2^2 = 19 \\
 u_3^1 = 4 & v_3^1 = 2 & w_3^1 = 12 & u_3^2 = -6 \quad v_3^2 = -13 \quad w_3^2 = 35
 \end{array}$$

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik koşulunu sağlayıp sağlamadığına bakılır.

$$z^2 (11 - (u_1^1 + v_1^1 + w_1^1)) - z^1 (12 - (u_1^2 + v_1^2 + w_1^2)) = 11z^2 + 13z^1 = 16199$$

$$z^2 (6 - (u_1^1 + v_2^1 + w_1^1)) - z^1 (4 - (u_1^2 + v_2^2 + w_1^2)) = 14z^2 + z^1 = 14492$$

$$z^2 (8 - (u_1^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_1^2 + v_3^2 + w_1^2)) = 6z^2 - 2z^1 = 5254$$

$$z^2 (7 - (u_2^1 + v_2^1 + w_1^1)) - z^1 (22 - (u_2^2 + v_2^2 + w_1^2)) = 11z^2 - 21z^1 = 2803$$

$$z^2 (8 - (u_2^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_2^2 + v_3^2 + w_1^2)) = 2z^2 - 6z^1 = -350$$

$$z^2 (12 - (u_3^1 + v_1^1 + w_1^1)) - z^1 (9 - (u_3^2 + v_1^2 + w_1^2)) = 8z^2 + 10z^1 = 11996$$

$$z^2 (5 - (u_3^1 + v_2^1 + w_1^1)) - z^1 (2 - (u_3^2 + v_2^2 + w_1^2)) = 9z^2 - 3z^1 = -7881$$

$$z^2 (14 - (u_1^1 + v_1^1 + w_2^1)) - z^1 (10 - (u_1^2 + v_1^2 + w_2^2)) = 13z^2 + 9z^1 = 16637$$

$$z^2 (11 - (u_1^1 + v_2^1 + w_2^1)) - z^1 (18 - (u_1^2 + v_2^2 + w_2^2)) = 18z^2 - 19z^1 = 10640$$

$$z^2 (3 - (u_1^1 + v_3^1 + w_2^1)) - z^1 (8 - (u_1^2 + v_3^2 + w_2^2)) = -2z^1 = -788$$

$$z^2 (9 - (u_2^1 + v_1^1 + w_2^1)) - z^1 (7 - (u_2^2 + v_1^2 + w_2^2)) = 4z^2 + 8z^1 = 7180$$

$$z^2 (10 - (u_2^1 + v_2^1 + w_2^1)) - z^1 (10 - (u_2^2 + v_2^2 + w_2^2)) = 13z^2 - 15z^1 = 7181$$

$$z^2 (4 - (u_3^1 + v_2^1 + w_2^1)) - z^1 (24 - (u_3^2 + v_2^2 + w_2^2)) = 7z^2 - 31z^1 = -5165$$

$$z^2 (9 - (u_3^1 + v_3^1 + w_2^1)) - z^1 (26 - (u_3^2 + v_3^2 + w_2^2)) = 2z^2 - 26z^1 = -8230 \Rightarrow$$

$$z^2 (15 - (u_1^1 + v_1^1 + w_3^1)) - z^1 (5 - (u_1^2 + v_1^2 + w_3^2)) = 3z^2 + 30z^1 = 14841$$

$$z^2 (14 - (u_1^1 + v_3^1 + w_3^1)) - z^1 (22 - (u_1^2 + v_3^2 + w_3^2)) = 5z^2 + 19z^1 = 12521$$

$$z^2(23 - (u_2^1 + v_2^1 + w_3^1)) - z^1(4 - (u_2^2 + v_2^2 + w_3^2)) = 15z^2 + 7z^1 = 17863$$

$$z^2(15 - (u_2^1 + v_3^1 + w_3^1)) - z^1(16 - (u_2^2 + v_3^2 + w_3^2)) = -3z^2 + 2z^1 = -2233$$

$$z^2(16 - (u_3^1 + v_1^1 + w_3^1)) - z^1(13 - (u_3^2 + v_1^2 + w_3^2)) = 16z^1 = 26304$$

$$z^2(19 - (u_3^1 + v_2^1 + w_3^1)) - z^1(14 - (u_3^2 + v_2^2 + w_3^2)) = 11z^2 - 5z^1 = 9107$$

$\min(\Delta_{ijk}) = -8230$ olduğundan u_3, v_3, w_2 kutusuna $2t$ değişkenini atamak gerekir. Tablonun yeni hali:

11	6	8	14	14	11	3
12	4	14	10	18	8	
4	7	8	9	10	7	2.5-t
22.5+t	21	22	7	10	2	
12	5	6	5	4	9	2t
9	2	1.5-t	17.5-t	13	24	26

15	4	14
5	5	7
21	23	15
12	4	16
16	19	18
13	14	4
		16

$1.5-t=0$, $17.5-t=0$ denklemlerinden $t=1.5$ 'tir.

Çizelge 3.8

11	6	8	14	11	3
12	4	14	10	18	8
4	7	8	9	10	7
24	21	22	7	10	2
12	5	6	5	4	9
9	2	6	16	13	3
				24	26

15	4	14
5	5	7
21	23	15
12	4	16
16	19	18
13	14	4
		16

$$z^1 = 4 * 24 + 5 * 16 + 7 * 1 + 9 * 3 + 4 * 5 + 14 * 7 + 18 * 4 = 400$$

$$z^2 = 24 * 21 + 16 * 13 + 1 * 2 + 3 * 26 + 5 * 15 + 7 * 22 + 4 * 16 = 1085$$

$$\bar{z} = 0.3687$$

Simpleks çarpanlar bulunur:

$$u_2^1 + v_1^1 + w_1^1 = 4$$

$$u_2^2 + v_1^2 + w_1^2 = 21$$

$$u_2^1 + v_3^1 + w_2^1 = 7$$

$$u_2^2 + v_3^2 + w_2^2 = 2$$

$$u_3^1 + v_1^1 + w_2^1 = 5$$

$$u_3^2 + v_1^2 + w_2^2 = 13$$

$$u_3^1 + v_3^1 + w_2^1 = 9$$

$$u_3^2 + v_3^2 + w_2^2 = 26$$

(3.7)

$$u_1^1 + v_2^1 + w_3^1 = 4$$

$$u_1^2 + v_2^2 + w_3^2 = 15$$

$$u_1^1 + v_3^1 + w_3^1 = 14$$

$$u_1^2 + v_3^2 + w_3^2 = 22$$

$$u_3^1 + v_3^1 + w_3^1 = 18$$

$$u_3^2 + v_3^2 + w_3^2 = 16$$

(3.7) denklem sisteminden aşağıdaki sonuçlar elde edilir:

$$\begin{array}{llll}
 u_1^1 = 0 & v_1^1 = 0 & w_1^1 = 2 & u_1^2 = 0 & v_1^2 = 0 & w_1^2 = 51 \\
 u_2^1 = 2 & v_2^1 = -6 & w_2^1 = 1 & u_2^2 = -30 & v_2^2 = 6 & w_2^2 = 19 \\
 u_3^1 = 4 & v_3^1 = 4 & w_3^1 = 10 & u_3^2 = -6 & v_3^2 = 13 & w_3^2 = 9
 \end{array}$$

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik koşulunu sağlayıp sağlamadığına bakılır.

$$z^2 (11 - (u_1^1 + v_1^1 + w_1^1)) - z^1 (12 - (u_1^2 + v_1^2 + w_1^2)) = 9z^2 + 39z^1 = 25365$$

$$z^2 (6 - (u_1^1 + v_2^1 + w_1^1)) - z^1 (4 - (u_1^2 + v_2^2 + w_1^2)) = 10z^2 + 53z^1 = 3050$$

$$z^2 (8 - (u_1^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_1^2 + v_3^2 + w_1^2)) = 2z^2 + 50z^1 = 22170$$

$$z^2 (7 - (u_2^1 + v_2^1 + w_1^1)) - z^1 (22 - (u_2^2 + v_2^2 + w_1^2)) = 9z^2 + 5z^1 = 11765$$

$$z^2 (8 - (u_2^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_2^2 + v_3^2 + w_1^2)) = 20z^1 = 8000$$

$$z^2 (12 - (u_3^1 + v_1^1 + w_1^1)) - z^1 (9 - (u_3^2 + v_1^2 + w_1^2)) = 6z^2 + 36z^1 = 20910$$

$$z^2 (5 - (u_3^1 + v_2^1 + w_1^1)) - z^1 (2 - (u_3^2 + v_2^2 + w_1^2)) = 5z^2 + 49z^1 = 25025$$

$$z^2 (6 - (u_3^1 + v_3^1 + w_1^1)) - z^1 (6 - (u_3^2 + v_3^2 + w_1^2)) = -4z^2 + 52z^1 = 16460$$

$$z^2 (14 - (u_1^1 + v_1^1 + w_2^1)) - z^1 (10 - (u_1^2 + v_1^2 + w_2^2)) = 13z^2 + 9z^1 = 17705$$

$$z^2 (11 - (u_1^1 + v_2^1 + w_2^1)) - z^1 (18 - (u_1^2 + v_2^2 + w_2^2)) = 16z^2 + 7z^1 = 20160$$

$$z^2 (3 - (u_1^1 + v_3^1 + w_2^1)) - z^1 (8 - (u_1^2 + v_3^2 + w_2^2)) = -2z^2 + 24z^1 = 7430$$

$$z^2 (9 - (u_2^1 + v_1^1 + w_2^1)) - z^1 (7 - (u_2^2 + v_1^2 + w_2^2)) = 6z^2 - 18z^1 = -690$$

$$z^2 (10 - (u_2^1 + v_2^1 + w_2^1)) - z^1 (10 - (u_2^2 + v_2^2 + w_2^2)) = 13z^2 - 15z^1 = 8105$$

$$z^2 (4 - (u_3^1 + v_2^1 + w_2^1)) - z^1 (24 - (u_3^2 + v_2^2 + w_2^2)) = 5z^2 - 5z^1 = -3425$$

$$z^2 (15 - (u_1^1 + v_1^1 + w_3^1)) - z^1 (5 - (u_1^2 + v_1^2 + w_3^2)) = 5z^2 + 4z^1 = 7025$$

$$z^2 (14 - (u_1^1 + v_3^1 + w_3^1)) - z^1 (22 - (u_1^2 + v_3^2 + w_3^2)) = 9z^2 - 33z^1 = -3435$$

$$z^2(23 - (u_2^1 + v_2^1 + w_3^1)) - z^1(4 - (u_2^2 + v_2^2 + w_3^2)) = 17z^2 - 19z^1 = 10845$$

$$z^2(15 - (u_2^1 + v_3^1 + w_3^1)) - z^1(16 - (u_2^2 + v_3^2 + w_3^2)) = z^2 - 24z^1 = -10685 \Rightarrow$$

$$z^2(16 - (u_3^1 + v_1^1 + w_3^1)) - z^1(13 - (u_3^2 + v_1^2 + w_3^2)) = 2z^2 - 10z^1 = -1830$$

$$z^2(19 - (u_3^1 + v_2^1 + w_3^1)) - z^1(14 - (u_3^2 + v_2^2 + w_3^2)) = 11z^2 - 5z^1 = 9935$$

$\min(\Delta_{ijk}) = -10685$ olduğundan u_2, v_3, w_3 kutusuna t değişkeni atanır. Tablonun yeni hali:

11	6	8
12	4	14
4 24 21	7 22	8 14
12 9	5 2	6 6

14	11	3
10	18	8
9 7	10 10	7 1-t 2
5 16 13	4 24	9 3+t 26

15	4 5 5	14 7 22
21 12	23 4	15 t 16
16 13	19 14	18 4-t 16

$1-t=0, 4-t=0$ denklemlerinden $t=\{1, 4\}$ tür. $\min(t)=1$ olduğundan $t=1$ olur.

Tablonun yeni hali:

Çizelge 3.9

11	6	8	14	11	3
12	4	14	10	18	8
4	7	8	9	10	7
24	21	22	7	10	2
12	5	6	5	4	9
9	2	6	16	13	4
				24	26

15	4	14
5	5	7
15	15	22
21	23	15
12	4	1
16	19	18
13	14	3
		16

$$z^1 = 4 * 24 + 5 * 16 + 9 * 4 + 4 * 5 + 14 * 7 + 15 * 1 + 18 * 3 = 399$$

$$z^2 = 24 * 21 + 16 * 13 + 4 * 26 + 5 * 15 + 7 * 22 + 1 * 16 + 3 * 16 = 1109$$

$$\bar{z} = 0.3589$$

Simpleks çarpanlar bulunur:

$$u_2^1 + v_1^1 + w_1^1 = 4$$

$$u_2^2 + v_1^2 + w_1^2 = 21$$

$$u_3^1 + v_1^1 + w_2^1 = 5$$

$$u_3^2 + v_1^2 + w_2^2 = 13$$

$$u_3^1 + v_3^1 + w_2^1 = 9$$

$$u_3^2 + v_3^2 + w_2^2 = 26$$

$$\begin{array}{ll}
u_1^1 + v_2^1 + w_3^1 = 4 & u_1^2 + v_2^2 + w_3^2 = 15 \\
u_1^1 + v_3^1 + w_3^1 = 14 & u_1^2 + v_3^2 + w_3^2 = 22 \\
u_2^1 + v_3^1 + w_3^1 = 15 & u_2^2 + v_3^2 + w_3^2 = 16 \\
u_3^1 + v_3^1 + w_3^1 = 18 & u_3^2 + v_3^2 + w_3^2 = 16
\end{array} \tag{3.8}$$

(3.8) denklem sisteminden aşağıdaki sonuçlar elde edilir:

$$\begin{array}{llllll}
u_1^1 = 0 & v_1^1 = 0 & w_1^1 = 3 & u_1^2 = 0 & v_1^2 = 0 & w_1^2 = 27 \\
u_2^1 = 1 & v_2^1 = -6 & w_2^1 = 1 & u_2^2 = -6 & v_2^2 = 6 & w_2^2 = 19 \\
u_3^1 = 4 & v_3^1 = 4 & w_3^1 = 10 & u_3^2 = -6 & v_3^2 = 13 & w_3^2 = 9
\end{array}$$

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik koşulunu sağlayıp sağlamadığına bakılır.

$$z^2 (11 - (u_1^1 + v_1^1 + w_1^1)) - z^1 (12 - (u_1^2 + v_1^2 + w_1^2)) = 8z^2 + 15z^1 = 14857$$

$$z^2 (6 - (u_1^1 + v_2^1 + w_1^1)) - z^1 (4 - (u_1^2 + v_2^2 + w_1^2)) = 9z^2 + 29z^1 = 21552$$

$$z^2 (8 - (u_1^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_1^2 + v_3^2 + w_1^2)) = 1z^2 + 26z^1 = 11483$$

$$z^2 (7 - (u_2^1 + v_2^1 + w_1^1)) - z^1 (22 - (u_2^2 + v_2^2 + w_1^2)) = 9z^2 + 5z^1 = 11976$$

$$z^2 (8 - (u_2^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_2^2 + v_3^2 + w_1^2)) = 20z^1 = 7980$$

$$z^2 (12 - (u_3^1 + v_1^1 + w_1^1)) - z^1 (9 - (u_3^2 + v_1^2 + w_1^2)) = 5z^2 + 12z^1 = 10333$$

$$z^2 (5 - (u_3^1 + v_2^1 + w_1^1)) - z^1 (2 - (u_3^2 + v_2^2 + w_1^2)) = 4z^2 + 25z^1 = 14411$$

$$z^2 (6 - (u_3^1 + v_3^1 + w_1^1)) - z^1 (6 - (u_3^2 + v_3^2 + w_1^2)) = -5z^2 + 28z^1 = 5627$$

$$z^2 (14 - (u_1^1 + v_1^1 + w_2^1)) - z^1 (10 - (u_1^2 + v_1^2 + w_2^2)) = 13z^2 + 9z^1 = 18008$$

$$z^2 (11 - (u_1^1 + v_2^1 + w_2^1)) - z^1 (18 - (u_1^2 + v_2^2 + w_2^2)) = 16z^2 + 7z^1 = 20537$$

$$z^2 (3 - (u_1^1 + v_3^1 + w_2^1)) - z^1 (8 - (u_1^2 + v_3^2 + w_2^2)) = -2z^2 + 24z^1 = 7358$$

$$z^2 (9 - (u_2^1 + v_1^1 + w_2^1)) - z^1 (7 - (u_2^2 + v_1^2 + w_2^2)) = 7z^2 + 6z^1 = 10157$$

$$z^2(10 - (u_2^1 + v_2^1 + w_2^1)) - z^1(10 - (u_2^2 + v_2^2 + w_2^2)) = 14z^2 + 9z^1 = 19117$$

$$z^2(7 - (u_2^1 + v_3^1 + w_2^1)) - z^1(2 - (u_2^2 + v_3^2 + w_2^2)) = z^2 + 24z^1 = 10685$$

$$z^2(4 - (u_3^1 + v_2^1 + w_2^1)) - z^1(24 - (u_3^2 + v_2^2 + w_2^2)) = 5z^2 - 5z^1 = 3550$$

$$z^2(15 - (u_1^1 + v_1^1 + w_3^1)) - z^1(5 - (u_1^2 + v_1^2 + w_3^2)) = 5z^2 + 4z^1 = 7141$$

$$z^2(14 - (u_1^1 + v_3^1 + w_3^1)) - z^1(22 - (u_1^2 + v_3^2 + w_3^2)) = 10z^2 - 9z^1 = 7499$$

$$z^2(23 - (u_2^1 + v_2^1 + w_3^1)) - z^1(4 - (u_2^2 + v_2^2 + w_3^2)) = 18z^2 + 5z^1 = 21957$$

$$z^2(16 - (u_3^1 + v_1^1 + w_3^1)) - z^1(13 - (u_3^2 + v_1^2 + w_3^2)) = 2z^2 - 10z^1 = -1772 \Rightarrow$$

$$z^2(19 - (u_3^1 + v_2^1 + w_3^1)) - z^1(14 - (u_3^2 + v_2^2 + w_3^2)) = 11z^2 - 5z^1 = 10204$$

$\min(\Delta_{ijk}) = -1772$ olduğundan u_3, v_1, w_3 kutusuna t değışkeni atanır. Tablonun yeni hali:

11	6	8
12	4	14
4 24	7	8
21	22	14
12	5	6
9	2	6

14	11	3
10	18	8
9	10	7
7	10	2
5 16-t	4	9 4+t
13	24	26

15	4 5	14 7
5	15	22
21	23	15 1
12	4	16
16 t	19	18 3-t
13	14	16

16-t=0 , 3-t=0 denklemlerinden t={ 16, 3 }'tür. Min(t)=3 olduğundan t=3 olur.

Çizelge 3.10

11	6	8	14	14	11	3
12	4	14	10	18	8	
4	7	8	9	10	7	
24	21	22	7	10	2	
12	5	6	5	4	9	7
9	2	6	13	24	26	

15	4	14
5	5	7
21	23	15
12	4	1
16	19	18
3	14	16

$$z^1 = 4 * 24 + 5 * 13 + 9 * 7 + 4 * 5 + 16 * 3 + 14 * 7 + 15 * 1 = 405$$

$$z^2 = 24 * 21 + 13 * 13 + 7 * 26 + 5 * 15 + 3 * 13 + 7 * 22 + 1 * 16 = 1139$$

$$\bar{z} = 0.3556$$

Simpleks çarpanlar bulunur:

$$u_2^1 + v_1^1 + w_1^1 = 4$$

$$u_2^2 + v_1^2 + w_1^2 = 21$$

$$u_3^1 + v_1^1 + w_2^1 = 5$$

$$u_3^2 + v_1^2 + w_2^2 = 13$$

$$u_3^1 + v_3^1 + w_2^1 = 9$$

$$u_3^2 + v_3^2 + w_2^2 = 26$$

$$\begin{array}{ll}
u_1^1 + v_2^1 + w_3^1 = 4 & u_1^2 + v_2^2 + w_3^2 = 15 \\
u_1^1 + v_3^1 + w_3^1 = 14 & u_1^2 + v_3^2 + w_3^2 = 22 \\
u_2^1 + v_3^1 + w_3^1 = 15 & u_2^2 + v_3^2 + w_3^2 = 16 \\
u_3^1 + v_1^1 + w_3^1 = 16 & u_3^2 + v_1^2 + w_3^2 = 13
\end{array} \tag{7}$$

(3.8) denklem sisteminden aşağıdaki sonuçlar elde edilir:

$$\begin{array}{llll}
u_1^1 = 0 & v_1^1 = 0 & w_1^1 = 3 & u_1^2 = 0 \quad v_1^2 = 0 \quad w_1^2 = 27 \\
u_2^1 = 1 & v_2^1 = -6 & w_2^1 = -1 & u_2^2 = -6 \quad v_2^2 = 6 \quad w_2^2 = 9 \\
u_3^1 = 6 & v_3^1 = 4 & w_3^1 = 10 & u_3^2 = -4 \quad v_3^2 = 13 \quad w_3^2 = 9
\end{array}$$

$\Delta_{ijk} = z^2 (c_{ijk} - (u_i^1 + v_j^1 + w_k^1)) - z^1 (d_{ijk} - (u_i^2 + v_j^2 + w_k^2)) > 0$ optimallik koşulunu sağlayıp sağlamadığına bakılır.

$$z^2 (11 - (u_1^1 + v_1^1 + w_1^1)) - z^1 (12 - (u_1^2 + v_1^2 + w_1^2)) = 8z^2 + 15z^1 = 15187$$

$$z^2 (6 - (u_1^1 + v_2^1 + w_1^1)) - z^1 (4 - (u_1^2 + v_2^2 + w_1^2)) = 9z^2 + 29z^1 = 21996$$

$$z^2 (8 - (u_1^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_1^2 + v_3^2 + w_1^2)) = z^2 + 26z^1 = 11669$$

$$z^2 (7 - (u_2^1 + v_2^1 + w_1^1)) - z^1 (22 - (u_2^2 + v_2^2 + w_1^2)) = 9z^2 + 5z^1 = 12276$$

$$z^2 (8 - (u_2^1 + v_3^1 + w_1^1)) - z^1 (14 - (u_2^2 + v_3^2 + w_1^2)) = 20z^1 = 8100$$

$$z^2 (12 - (u_3^1 + v_1^1 + w_1^1)) - z^1 (9 - (u_3^2 + v_1^2 + w_1^2)) = 3z^2 + 22z^1 = 12327$$

$$z^2 (5 - (u_3^1 + v_2^1 + w_1^1)) - z^1 (2 - (u_3^2 + v_2^2 + w_1^2)) = 2z^2 + 35z^1 = 16453$$

$$z^2 (6 - (u_3^1 + v_3^1 + w_1^1)) - z^1 (6 - (u_3^2 + v_3^2 + w_1^2)) = -7z^2 + 38z^1 = 7417$$

$$z^2 (14 - (u_1^1 + v_1^1 + w_2^1)) - z^1 (10 - (u_1^2 + v_1^2 + w_2^2)) = 15z^2 - z^1 = 16680$$

$$z^2 (11 - (u_1^1 + v_2^1 + w_2^1)) - z^1 (18 - (u_1^2 + v_2^2 + w_2^2)) = 18z^2 - 3z^1 = 19287$$

$$z^2 (3 - (u_1^1 + v_3^1 + w_2^1)) - z^1 (8 - (u_1^2 + v_3^2 + w_2^2)) = 24z^1 = 5670$$

$$z^2 (9 - (u_2^1 + v_1^1 + w_2^1)) - z^1 (7 - (u_2^2 + v_1^2 + w_2^2)) = 9z^2 - 4z^1 = 8631$$

$$z^2(10 - (u_2^1 + v_2^1 + w_2^1)) - z^1(10 - (u_2^2 + v_2^2 + w_2^2)) = 16z^2 - z^1 = 17819$$

$$z^2(7 - (u_2^1 + v_3^1 + w_2^1)) - z^1(2 - (u_2^2 + v_3^2 + w_2^2)) = 3z^2 + 14z^1 = 9087$$

$$z^2(4 - (u_3^1 + v_2^1 + w_2^1)) - z^1(24 - (u_3^2 + v_2^2 + w_2^2)) = 5z^2 - 5z^1 = 3670$$

$$z^2(15 - (u_1^1 + v_1^1 + w_3^1)) - z^1(5 - (u_1^2 + v_1^2 + w_3^2)) = 5z^2 + 4z^1 = 7315$$

$$z^2(14 - (u_1^1 + v_3^1 + w_3^1)) - z^1(22 - (u_1^2 + v_3^2 + w_3^2)) = 10z^2 - 9z^1 = 7745$$

$$z^2(23 - (u_2^1 + v_2^1 + w_3^1)) - z^1(4 - (u_2^2 + v_2^2 + w_3^2)) = 18z^2 + 5z^1 = 22527$$

$$z^2(19 - (u_3^1 + v_2^1 + w_3^1)) - z^1(14 - (u_3^2 + v_2^2 + w_3^2)) = 9z^2 + 5z^1 = 12276$$

$$z^2(18 - (u_3^1 + v_3^1 + w_3^1)) - z^1(16 - (u_3^2 + v_3^2 + w_3^2)) = -2z^2 + 10z^1 = 1772$$

Tüm $\Delta_{ijk} > 0$ olduğu için optimal çözüme ulaşıldı.

İterasyonlardan elde edilen sonuçlar aşağıdaki gibidir:

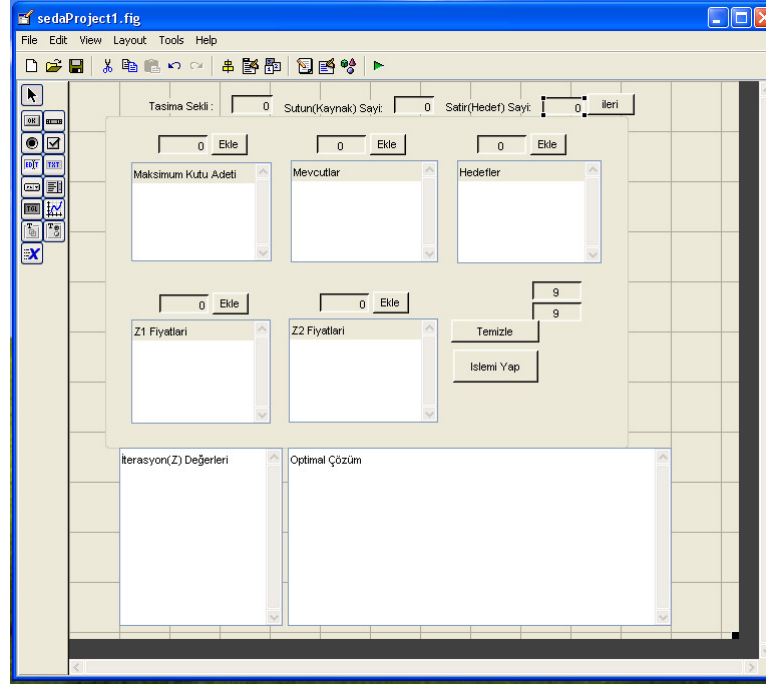
Çizelge 3.11

iterasyonlar	$\bar{z}$
1	0.7558
2	0.5528
3	0.4411
4	0.3913
5	0.3687
6	0.3598
7	0.3556

4. UYGULAMA PROGRAMI

4.1 Programın Tanımı Ve İçeriği

Çözümü incelenen Kanti-Swarup algoritmasının uygulaması MATLAB programı kullanılarak hazırlanmıştır. Programın ara yüzü aşağıdadır:



Şekil 4.1

Program görsel bir program olup üç boyutlu taşıma problemleri için kullanılabilir. Programdaki veri giriş alanları aşağıdaki gibi tanımlanmıştır:

- Taşıma Şekli: Kaç farklı yolla taşıma işleminin yapıldığını gösterir. Bu çalışmada incelenen problem için her zaman 3 olarak girilmelidir. Üç boyutlu taşıma problemi uygulamasında 3 farklı yolla örneğin kara, tren ve deniz yoluyla taşıma şeklinin incelendiği düşünülebilir.
- Kaynak Sayısı: Mevcut malların bulunduğu depo sayısıdır.
- Hedef Sayısı: Malların ulaştırılacağı pazar/hedef yer sayısıdır.
- Taşınabilecek Miktar: Her bir taşıma şekli için taşınması gereken miktar.

- Depolar: Her depodaki mal miktarı.
- Hedefler: Pazara taşınması gereken mal miktarı.
- Z1 Fiyatları: Birinci amacımızın fiyatlarını gösterir.
- Z2 Fiyatları: İkinci amacımızın fiyatlarını gösterir.
- Temizle Butonu: Tüm veri girişi yapılmış alanları aynı anda temizler. Ayrıca kutuların içlerine tıklandığında veriler silinir.
- İşlemi Yap Butonu: Taşıma işlemini gerçekleştirir ve iterasyon değerlerini ve taşınan miktarın hesaplanıp ekranda gösterilmesini sağlar.

4.2 Programın Çözümü

```
function varargout = tez(varargin)

% TEZ Application M-file for tez.fig

% FIG = TEZ launch tez GUI.

% TEZ('callback_name', ...) invoke the named callback.

% Last Modified by GUIDE v2.5 04-Jan-2009 17:22:45

if nargin == 0 % LAUNCH GUI

fig = openfig(mfilename,'reuse');

% Use system color scheme for figure:

set(fig,'Color',get(0,'defaultUicontrolBackgroundColor'));

% Generate a structure of handles to pass to callbacks, and store it.

handles = guihandles(fig);

guidata(fig, handles);

if nargout > 0

varargout{1} = fig;

end
```

```

elseif ischar(varargin{1}) % INVOKE NAMED SUBFUNCTION OR CALLBACK

try

[varargout{1:nargout}] = feval(varargin{:}); % FEVAL switchyard

catch

disp(lasterr);

end

end

%| ABOUT CALLBACKS:

%| GUIDE automatically appends subfunction prototypes to this file, and

%| sets objects' callback properties to call them through the FEVAL

%| switchyard above. This comment describes that mechanism.

%| Each callback subfunction declaration has the following form:

%| <SUBFUNCTION_NAME>(H, EVENTDATA, HANDLES, VARARGIN)

%| The subfunction name is composed using the object's Tag and the

%| callback type separated by '_', e.g. 'slider2_Callback',

%| 'figure1_CloseRequestFcn', 'axis1_ButtondownFcn'.

%| H is the callback object's handle (obtained using GCBO).

%| EVENTDATA is empty, but reserved for future use.

%| HANDLES is a structure containing handles of components in GUI using

%| tags as fieldnames, e.g. handles.figure1, handles.slider2. This

%| structure is created at GUI startup using GUIHANDLES and stored in

%| the figure's application data using GUIDATA. A copy of the structure

%| is passed to each callback. You can store additional information in

%| this structure at GUI startup, and you can change the structure

```

```

%| during callbacks. Call guidata(h, handles) after changing your
%| copy to replace the stored original so that subsequent callbacks see
%| the updates. Type "help guihandles" and "help guidata" for more
%| information.

%| VARARGIN contains any extra arguments you have passed to the
%| callback. Specify the extra arguments by editing the callback
%| property in the inspector. By default, GUIDE sets the property to:
%| <MFILENAME>(<SUBFUNCTION_NAME>', gcbo, [], guidata(gcbo))
%| Add any extra arguments after the last argument, before the final
%| closing parenthesis.

% -----

function varargout = pushbutton1_Callback(h, eventdata, handles, varargin)

% Stub for Callback of the uicontrol handles.pushbutton1.

    disp('pushbutton1 Callback not implemented yet.')

    pdeg=get(handles.pdegEdit1,'String');

    mdeg=get(handles.mdegEdit1,'String');

    ndeg=get(handles.ndegEdit1,'String');

    % set(handles.pdegEdit1,'Enable','off');

    % set(handles.mdegEdit1,'Enable','off');

    % set(handles.ndegEdit1,'Enable','off');

    sonuc=str2double(pdeg)*str2double(mdeg)*str2double(ndeg);

    % set(handles.pushbutton1,'Visible','off');

    set(handles.z1Edit1,'String',sonuc);

    set(handles.z2Edit1,'String',sonuc);

```

```

% -----

function varargout = listBox1_Callback(h, eventdata, handles, varargin)

% Stub for Callback of the uicontrol handles.listBox1.

disp('listBox1 Callback not implemented yet.')

%-----

function varargout = listeyeEkle(edit1,list1,addedit1,error,errorStr)

%TÜM LİSTELER İÇİN LİSTEYE EKLEME İŞLEMİ

deg=str2double(get(edit1,'String'));

a = get(list1,'String');

temp = char(a);

b = get(addedit1,'String');

c = strvcats(temp,b);

aa = get(list1,'UserData');

deg=deg+1;

if aa < deg

    set(list1,'String',cellstr(c));

    set(error,'String','');

    aa=aa+1;

    set(list1,'UserData',aa);

else

    set(error,'String',errorStr);

end

% -----

```

```
function varargout = pushbutton12_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.pushbutton12.
```

```
listeyeEkle(handles.pdegEdit1,handles.kutuAdetList1,handles.addKutuEdit1,handles.error1,'  
Kutu Adetini Geçemezsiniz');
```

```
% -----
```

```
function varargout = ndegEdit1_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.ndegEdit1.
```

```
disp('ndegEdit1 Callback not implemented yet.')
```

```
% -----
```

```
function varargout = pushbutton5_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.pushbutton5.
```

```
disp('pushbutton5 Callback not implemented yet.')
```

```
listeyeEkle(handles.ndegEdit1,handles.mevcutList1,handles.addMevcutEdit1,handles.error1,'  
Sutun Adetini Geçemezsiniz');
```

```
% -----
```

```
function varargout = pushbutton6_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.pushbutton6.
```

```
disp('pushbutton6 Callback not implemented yet.')
```

```
listeyeEkle(handles.mdegEdit1,handles.hedefList1,handles.addHedefEdit1,handles.error1,'Sat  
ir Adetini Geçemezsiniz');
```

```
% -----
```

```
function varargout = Z1List_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.Z1List.
```

```
disp('Z1List Callback not implemented yet.')
```

```
silme_fonksiyonu(h,'Z1 Fiyatlari');
```

```
% -----
```

```
function varargout = Z2List_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.Z2List.
```

```
disp('Z2List Callback not implemented yet.')
```

```
silme_fonksiyonu(h,'Z2 Fiyatlari');
```

```
% -----
```

```
function varargout = edit10_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.edit10.
```

```
disp('edit10 Callback not implemented yet.')
```

```
% -----
```

```
function varargout = edit11_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.edit11.
```

```
disp('edit11 Callback not implemented yet.')
```

```
% -----
```

```
function varargout = pushbutton8_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.pushbutton8.
```

```
disp('pushbutton8 Callback not implemented yet.')
```

```
listeyeEkle(handles.z1Edit1,handles.Z1List,handles.addZ1Edit1,handles.error1,'Z1 Adetini  
Geçemezsiniz');
```

```
% -----
```

```
function varargout = z1Edit1_Callback(h, eventdata, handles, varargin)
```

```
% Stub for Callback of the uicontrol handles.z1Edit1.
```

```
disp('z1Edit1 Callback not implemented yet.')
```

```
% -----
```

```

function varargout = z2Edit1_Callback(h, eventdata, handles, varargin)

% Stub for Callback of the uicontrol handles.z2Edit1.

disp('z2Edit1 Callback not implemented yet.')

% -----

function varargout = pushbutton9_Callback(h, eventdata, handles, varargin)

% Stub for Callback of the uicontrol handles.pushbutton9.

disp('pushbutton9 Callback not implemented yet.')

listeYEkle(handles.z2Edit1,handles.Z2List,handles.addZ2Edit1,handles.error1,'Z2 Adetini
Geçemezsınız');

% -----

function varargout = islemYapButton1_Callback(h, eventdata, handles, varargin)

%İŞLEMİ YAPAN BUTON

% Stub for Callback of the uicontrol handles.islemYapButton1.

disp('islemYapButton1 Callback not implemented yet.')

    pdeg=str2double(get(handles.pdegEdit1,'String'));

% 'pdeg' DEĞİŞKENİ TAŞIMA ŞEKLİ SAYISINI TUTAR

    mdeg=str2double(get(handles.mdegEdit1,'String'));

% 'mdeg' DEĞİŞKENİ HEDEF SAYISINI TUTAR

    ndeg=str2double(get(handles.ndegEdit1,'String'));

% 'ndeg' DEĞİŞKENİ KAYNAK SAYISINI TUTAR

    kutuAdet=str2double(get(handles.kutuAdetList1,'String'));

    mevcut=str2double(get(handles.mevcutList1,'String'));

    hedef=str2double(get(handles.hedefList1,'String'));

    z1diziStr=str2double(get(handles.Z1List,'String'));

```

```

z2diziStr=str2double(get(handles.Z2List,'String'));

%-----BAŞLANGIÇ ÇÖZÜMÜ-----

ii=2;

for k=1 :pdeg
    for i=1 :mdeg
        for j=1 :ndeg
            z1dizi(i,j,k)=z1diziStr(ii);
            z2dizi(i,j,k)=z2diziStr(ii);
            ii=ii+1;
        end
    end
end

disp(z1dizi);
disp(z2dizi);

for i=1 : ndeg
    mevcutKalan(i)=mevcut(i+1);
end

for k=1 : pdeg
    kutuAdetKalan(k)=kutuAdet(k+1);
end

for j=1 : mdeg
    hedefKalan(j)=hedef(j+1);
end

disp('kutuAdetKalan');
```

```

disp(kutuAdetKalan);

disp('mevcutKalan');

disp(mevcutKalan);

disp('hedefKalan');

disp(hedefKalan);

kutuXsutun=pdeg*ndeg;

ortadizi=zeros(mdeg,ndeg,pdeg);

j=1;

k=1;

sonuc=0;

o=1;

doluYerSay=mdeg+ndeg+pdeg-2;

dolusayisi=0;

for i=1 :mdeg

    while o <= kutuXsutun

        xx=ceil( o / ndeg );

        if xx<=pdeg

            k=xx;

        else

            k=pdeg;

        end

        j= mod(o, ndeg);

        if j==0

            j=ndeg;

```

```

end

karsilsir(1)=kutuAdetKalan(k);

karsilsir(2)=hedefKalan(i);

karsilsir(3)=mevcutKalan(j);

sonuc=min(karsilsir);

if (sonuc<= kutuAdetKalan(k) & sonuc>0)

    kutuAdetKalan(k)=kutuAdetKalan(k)-sonuc;

    hedefKalan(i)=hedefKalan(i)-sonuc;

    mevcutKalan(j)=mevcutKalan(j)-sonuc;

    ortadizi(i,j,k)=sonuc;

    dolusayisi=dolusayisi+1;

    o=o+1;

end

if sonuc ==0

    o=o+ndeg;

end

end

o=1;

end

for k =1 : pdeg

    if( kutuAdetKalan(k) > 0 )

        for j =1 : ndeg

            if (mevcutKalan(j)> 0)

                for i =1 : mdeg

```

```

if (hedefKalan(i)> 0)

    if( ortadizi(i,j,k) == 0 )

        karsilsir(1)=kutuAdetKalan(k);

        karsilsir(2)=hedefKalan(i);

        karsilsir(3)=mevcutKalan(j);

        sonuc=min(karsilsir);

        if (sonuc<= kutuAdetKalan(k) & sonuc>0)

            kutuAdetKalan(k)=kutuAdetKalan(k)-sonuc;

            hedefKalan(i)=hedefKalan(i)-sonuc;

            mevcutKalan(j)=mevcutKalan(j)-sonuc;

            ortadizi(i,j,k)=sonuc;

            dolusayisi=dolusayisi+1;

        end

    end

end

end

end

end

end

end

if(doluYerSay == dolusayisi )

    disp('Orta Dizi');

    disp(ortadizi);

```

```

else

    disp('Hedef Dolu yer sayısına ulasilamadi!..');

end

%-----BAŞLANGIÇ ÇÖZÜMÜ SONU-----

%-----2.İşlem(OPTİMAL ÇÖZÜM) -----

sonsuzsongu=10000;

%ortadizi=zeros(mdeg,ndeg,pdeg);

for dongu=1 : sonsuzsongu

    % Alt Zarar Islem (satir,sutun,kutu)

    denklemZ1=zeros(doluYerSay,doluYerSay+2);

    denklemDiziZ1=zeros(doluYerSay,doluYerSay);

    denkZ1=zeros(doluYerSay);

    %disp('Z1 Denklem  ');

    %disp(denklemZ1);

    denklemZ2=zeros(doluYerSay,doluYerSay);

    denklemDiziZ2=zeros(doluYerSay,doluYerSay);

    denkZ2=zeros(doluYerSay);

    %disp('Z2 Denklem  ');

    %disp(denklemZ2);

    z1=0;

    z2=0;

    ii=1;

    kZ1olmaz=0;

    jZ1olmaz=0;

```

```

kZ2olmaz=0;

jZ2olmaz=0;

detControlZ1=1;

detControlZ2=1;

%repeat

while (1==1)

for k=1 :pdeg

    for i=1 :mdeg

        for j=1 :ndeg

            if ortadizi(i,j,k) ~= 0

                z1 =z1+ ortadizi(i,j,k)*z1dizi(i,j,k);

                z2 =z2+ ortadizi(i,j,k)*z2dizi(i,j,k);

                denklemZ1(ii,j)=1;

                denklemZ1(ii,i+ndeg)=1;

                denklemZ1(ii,k+mdeg+ndeg)=1;

                denkZ1(ii,1)=z1dizi(i,j,k);

                denklemZ2(ii,j)=1;

                denklemZ2(ii,i+ndeg)=1;

                denklemZ2(ii,k+mdeg+ndeg)=1;

                denkZ2(ii,1)=z2dizi(i,j,k);

                ii=ii+1;

            if(kZ1olmaz==0)

                kZ1olmaz=detControlZ1;%k

            end

```

```

        if(jZ1olmaz==0)

            jZ1olmaz=detControlZ1+ndeg;%j

        end

        if(kZ2olmaz==0)

            kZ2olmaz=detControlZ2;%k

        end

        if(jZ2olmaz==0)

            jZ2olmaz=detControlZ2+ndeg;%j

        end

    end

end

end

end

i=1;

ii=1;

%disp('denklemZ1');

%disp(denklemZ1);

%disp('denklemZ2');

%disp(denklemZ2);

for j=1 : doluYerSay+2

    if kZ1olmaz ~= j & jZ1olmaz ~= j

        for k=1 : doluYerSay

            denklemDiziZ1(k,i)=denklemZ1(k,j);

        end
    end
end

```

```

        i=i+1;

    end

    if kZ2olmaz ~= j & jZ2olmaz ~= j

        for k=1 : doluYerSay

            denklemDiziZ2(k,ii)=denklemZ2(k,j);

        end

        ii=ii+1;

    end

end

%disp('denklemDiziZ1');

%disp(denklemDiziZ1);

%disp('denklemDiziZ2');

%disp(denklemDiziZ2);

det1=det(denklemDiziZ1);

det2=det(denklemDiziZ1);

detControlZ1=detControlZ1+1;

detControlZ2=detControlZ2+1;

if( det1 ~=0 && det2 ~=0)

    break;

end

end

d1= inv(denklemDiziZ1)*denkZ1;

d2= inv(denklemDiziZ2)*denkZ2;

disp('Z1/Z2 Sonuç :');

```

```

disp(z1/z2);

disp('Z1');

disp(z1);

disp('Z2');

disp(z2);

a = get(handles.ZetList,'String');

temp = char(a);

b = num2str(z1/z2);

c = strvcac(temp,b);

set(handles.ZetList,'String',cellstr(c));

disp('v-u-w degerleri ');

disp(d1);

disp(d2);

denklemler=zeros(2,doluYerSay+2);

k1=1;

k2=1;

for k=1 : doluYerSay+2

    if (kZ1olmaz==k | jZ1olmaz==k)

        denklemler(1,k)=0;

    else

        denklemler(1,k)=d1(k1,1);

        k1=k1+1;

    end

    if (kZ2olmaz==k | jZ2olmaz==k)

```

```

denklemler(2,k)=0;

else

denklemler(2,k)=d2(k2,1);

k2=k2+1;

end

end

boslar=zeros(mdeg,ndeg,pdeg);

doluyer=mdeg+ndeg+pdeg;

doluYerSay=doluyer-2;

bosSize=mdeg*ndeg*pdeg-doluYerSay;

xx=1;

for i=1 :pdeg

    for j=1 :mdeg

        for k=1 :ndeg

            if (ortadizi(j,k,i) == 0)

                ds2=z2*(z1dizi(j,k,i)-
(denklem(1,k)+denklemler(1,j+ndeg)+denklemler(1,i+mdeg++ndeg)));

                ds1=z1*(z2dizi(j,k,i)-
(denklem(2,k)+denklemler(2,j+ndeg)+denklemler(2,i+mdeg+ndeg)));

                % DELTA DEĞERLERİNİN HESAPLANMASI...

                boslar(j,k,i)=ds2-ds1;

            end

        end

    end

end

end

```

```

disp(boslar);

enkucukyer=[1 1 1];

enkucuk=boslar(1,1,1);

for k=1 : pdeg
    for i=1 : mdeg
        for j=1 : ndeg
            if enkucuk>boslar(i,j,k);
                enkucuk=boslar(i,j,k);
                enkucukyer(1)=i;
                enkucukyer(2)=j;
                enkucukyer(3)=k;
            % EN KÜÇÜK BOŞ ALAN DEĞERİ BULUNUYOR....
        end
    end
end

if(boslar(enkucukyer(1),enkucukyer(2),enkucukyer(3))>=0)
    break ;
else
    % T NİN ATANACAĞI YERİN BULUNMASI

    disp( 'Setçilen t Yeri Kutu Satir Sutun =' );

    disp( enkucukyer(1) );

    disp( enkucukyer(2) );

    disp( enkucukyer(3) );

```

```

end

disp(enkucukyer);

ortadizi(enkucukyer(1),enkucukyer(2),enkucukyer(3))=-1;

tlidenklem=zeros(doluyer,doluYerSay+1);

aradenklem=zeros(doluyer,doluYerSay+1,3);

ii=1;

jj=1;

kk=1;

liste=zeros(doluYerSay+1,3);

for j=1 : ndeg
    for k=1 : pdeg
        for i=1 : mdeg
            if (ortadizi(i,j,k)>0 )
                aradenklem(jj,1,ii)=i;
                aradenklem(jj,2,ii)=j;
                aradenklem(jj,3,ii)=k;

                liste(kk,1)=i;
                liste(kk,2)=j;
                liste(kk,3)=k;

                kk=kk+1;

                jj=jj+1;
            end
        end
    end
    if(ortadizi(i,j,k)==-1)
        tlidenklem(ii,doluYerSay+1)=-1;
    end
end

```

```

        end

    end

end

ii=ii+1;

end

jj=1;

for i=1 : mdeg

    for k=1 : pdeg

        for j=1 : ndeg

            if (ortadizi(i,j,k)>0 )

                aradenklem(jj,1,ii)=i;

                aradenklem(jj,2,ii)=j;

                aradenklem(jj,3,ii)=k;

                jj=jj+1;

            end

            if(ortadizi(i,j,k)==-1)

                tlidenklem(ii,doluYerSay+1)=-1;

            end

        end

    end

    ii=ii+1;

end

jj=1;

for k=1 : pdeg

```

```

for i=1 : mdeg
    for j=1 : ndeg
        if (ortadizi(i,j,k)>0 )
            aradenklem(jj,1,ii)=i;
            aradenklem(jj,2,ii)=j;
            aradenklem(jj,3,ii)=k;

            jj=jj+1;
        end

        if(ortadizi(i,j,k)==-1)
            tlidenklem(ii,doluYerSay+1)=-1;
        end
    end
end

end

ii=ii+1;

end

%disp(aradenklem);

for ii=1 : doluyer
    for jj=1 : doluYerSay+1

        if( aradenklem(jj,1,ii)>0 & aradenklem(jj,2,ii)>0 & aradenklem(jj,3,ii)>0 )

            for kk=1 : doluYerSay+1

                if(aradenklem(jj,1,ii)==liste(kk,1) & aradenklem(jj,2,ii)==liste(kk,2) &
aradenklem(jj,3,ii)==liste(kk,3))

                    tlidenklem(ii,kk)=1;
                end
            end
        end
    end
end

```

```

        end

    end

end

end

disp('Rank Denkleminin Baslik Bilgisi');

disp(liste);

%disp(tlidenklem);

disp('Rankli t denklemi');

tt=rref(tlidenklem);

disp(tt);

i=1;

    for kk=1 : doluYerSay+1

        if liste(kk,1)>0

            if(tt(kk,doluYerSay+1)<0)

dizienkucubul(i)=ortadizi(liste(kk,1),liste(kk,2),liste(kk,3))/abs(tt(kk,doluYerSay+1));

                i=i+1;

            end

        end

    end

end

%disp(dizienkucubul);

% T'NİN DEĞERİ BURADA BULUNUYOR...

tdegeri=min(dizienkucubul) ;

%disp(tdegeri);

```

```

for kk=1 : doluYerSay+1

    if liste(kk,1)>0

        if (ortadizi(liste(kk,1),liste(kk,2),liste(kk,3))>0 )

            % TABLONUN YENİ HALİ BULUNUYOR...
            ortadizi(liste(kk,1),liste(kk,2),liste(kk,3))=ortadizi(liste(kk,1),liste(kk,2),liste(kk,3))+tt(kk,doluYerSay+1)*tdegeri;

            end

        end

    end

    ortadizi(enkucukyer(1),enkucukyer(2),enkucukyer(3))=tdegeri;

    disp('tdegeri');

    disp(tdegeri);

    disp('Orta Dizi');

    disp(ortadizi);

    %a = get(handles.ortaList1,'String');

    %temp = char(a);

    %c=num2str(ortadizi);

    %c = strvcats(temp,b);

    %set(handles.ortaList1,'String',cellstr(c));

end

disp('Enson');

%-----2.İşlem(OPTİMAL ÇÖZÜM) SONU -----

%  disp('Orta Dizi');

%  disp(ortadizi);

str=";
```

```

for k=1 : pdeg

    str=strvcat(str,strvcat(num2str(k), ' . Kutu'));

    str=strvcat(str,num2str(ortadizi(:,k)));

end

set(handles.ortaList1,'String',str);

% --- Executes on selection change in ZetList.

function ZetList_Callback(hObject, eventdata, handles)

% hObject    handle to ZetList (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns ZetList contents as cell array

%    contents{get(hObject,'Value')} returns selected item from ZetList

% --- Executes on button press in cleanerButton1.

function cleanerButton1_Callback(hObject, eventdata, handles)

% hObject    handle to cleanerButton1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

silme_fonksiyonu(handles.kutuAdetList1,'Tasinacak Miktar');

silme_fonksiyonu(handles.mevcutList1,'Mevcutlar');

silme_fonksiyonu(handles.hedefList1,'Hedefler');

silme_fonksiyonu(handles.Z1List,'Z1 Fiyatlari');

silme_fonksiyonu(handles.Z2List,'Z2 Fiyatlari');

silme_fonksiyonu(handles.ZetList,'iterasyon(Z) Degerleri');

```

```

silme_fonksiyonu(handles.ortaList1,'Optimal Çözüm');

% --- Executes during object creation, after setting all properties.

function ZetList_CreateFcn(hObject, eventdata, handles)

% hObject    handle to ZetList (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.

%    See ISPC and COMPUTER.

if ispc

    set(hObject,'BackgroundColor','white');

else

    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));

end

% --- Executes on selection change in ortaList1.

function ortaList1_Callback(hObject, eventdata, handles)

% hObject    handle to ortaList1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns ortaList1 contents as cell array

%    contents{get(hObject,'Value')} returns selected item from ortaList1

% --- Executes during object creation, after setting all properties.

function ortaList1_CreateFcn(hObject, eventdata, handles)

% hObject    handle to ortaList1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

```

```

% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.

%    See ISPC and COMPUTER.

if ispc

    set(hObject,'BackgroundColor','white');

else

    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));

end

function pdegEdit1_Callback(hObject, eventdata, handles)

% hObject    handle to pdegEdit1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of pdegEdit1 as text

%    str2double(get(hObject,'String')) returns contents of pdegEdit1 as a double

% --- Executes on selection change in kutuAdetList1.

function kutuAdetList1_Callback(hObject, eventdata, handles)

% hObject    handle to kutuAdetList1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

    silme_fonksiyonu(hObject,'Tasinacak Miktar');

% --- Executes during object deletion, before destroying properties.

function kutuAdetList1_DeleteFcn(hObject, eventdata, handles)

% hObject    handle to kutuAdetList1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

```

```

% handles    structure with handles and user data (see GUIDATA)

% --- Executes on selection change in mevcutList1.

function mevcutList1_Callback(hObject, eventdata, handles)

% hObject    handle to mevcutList1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

silme_fonksiyonu(hObject,'Mevcutlar');

% Hints: contents = get(hObject,'String') returns mevcutList1 contents as cell array
%      contents{get(hObject,'Value')} returns selected item from mevcutList1

function silme_fonksiyonu(hObject,str)

    set(hObject,'Value',1);

    set(hObject,'String',str);

    set(hObject,'UserData',1);

% --- Executes on selection change in hedefList1.

function hedefList1_Callback(hObject, eventdata, handles)

% hObject    handle to hedefList1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

silme_fonksiyonu(hObject,'Hedefler');

% Hints: contents = get(hObject,'String') returns hedefList1 contents as cell array
%      contents{get(hObject,'Value')} returns selected item from hedefList1

```

5. SONUÇ

Dünya pazardaki rekabetin ve uluslararası ticaretin her geçen gün arttığı günümüzde, taşımacılık maliyetleri çok kritik bir unsur haline gelmiştir. Özellikle ihracatla ve taşımacılıkla uğraşan firmalar için bir aracın kapasitesinden en iyi şekilde faydalanmak ve kullanılan araç sayısını azaltmak çok önemlidir. Dolayısıyla bir çok kısıtın varlığında çözülmesi gereken bu problemi çözecek farklı yaklaşımlara ihtiyaç vardır. Bu amaçla lineer kesirli programlama tekniği[5] klasik ve üç boyutlu taşıma problemine uygulanmıştır.

Klasik taşıma problemimizin program sonucu:

Untitled

Sutun(Mevcut) Sayı: 3 Satır(Hedef) Sayı: 3

50 Ekle

Maksimum Kutu Adeti

50

20 Ekle

Mevcutlar

10
20
20

25 Ekle

Hedefler

12
13
25

5 Ekle

Z1 Fiyatları

1
3
7
8
2
6
10

7 Ekle

Z2 Fiyatları

7
8
3
4
5
8
11

Temizle

İşlemi Yap

İterasyon(Z) Değerleri

0.61111
0.5755

Şekil 5.1

Üç boyutlu taşıma probeminin sonucu aşağıdaki gibidir:

Untitled

Kutu Sayı : 3 Sütun(Mevcut) Sayı: 3 Satır(Hedef) Sayı: 3 İleri

0 Ekle 0 Ekle 0 Ekle

Maksimum Kutu Ağırlığı
24
20
16

Mevcutlar
40
5
15

Hedefler
12
25
23

0 Ekle 0 Ekle

Z1 Degerleri
11
6
8
4
7
8
12

Z2 Degerleri
12
4
14
21
22
14
9

Temizle

İşlemi Yap

Listbox
0.75575
0.55282
0.44112
0.39126
0.36866
0.35978
0.35558

1
.Kutu
0 0 0
24 0 0
0 0 0
2
.Kutu
0 0 0
0 0 0
13 0 7
3
.Kutu
0 5 7
0 0 1
3 0 0

Şekil 5.2

KAYNAKLAR

- [1] Dereli T., Baykasoğlu A., Daş G., Göçken T., (2004), “Üç Boyutlu Paketleme Problemlerine Analitik Yaklaşımlar”, YA/EM'2004 - Yöneylem Araştırması/Endüstri Mühendisliği - XXIV Ulusal Kongresi, Gaziantep – Adana
- [2] Karayalçın İ., (1979), Harekat Araştırması, Fatih Yayınevi, İstanbul,130
- [3] Moanta D. (2007), “Some Aspects on Solving a Linear Fractional Transportation Problem”, Journal of Applied Quantitative Methods Vol.2 No.3,343-345
- [4] Sarıalan H., (1986), Kaynak Dağılımında Doğrusal Programlama, Ankara Üniversitesi Siyasal Bilgiler Fakültesi Yayınları, Ankara,207
- [5] Swarup K., (1964), Linear Fractional Functionals Programming, University of Delhi, India,1028-1033
- [6] Taha Hamdy A., (2003), Yöneylem Araştırması, Literatür Yayıncılık, İstanbul,11

ÖZGEÇMİŞ

Doğum tarihi 10.08.1984

Doğum yeri İstanbul

Yüksek Lisans 2006-2009

Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Matematik Müh. Anabilim Dalı

Lisans 2002-2006

Yıldız Teknik Üniversitesi Fen-Edebiyat Fak.
Matematik Bölümü

2004-2008

Anadolu Üniversitesi İşletme Fakültesi
İşletme Bölümü

Lise 1998-2002

Hüseyin Avni Sözen Anadolu Lisesi

Çalıştığı kurum(lar)

2008-Devam ediyor Akbank T.A.Ş