G.Ü. İ.İ.B.F. Dergisi, 1/2003, 37-52

Çok Amaçlı Atama Problemlerine Bir Çözüm Önerisi

İbrahim GÜNGÖR*

ABSTRACT

In multicriteria assignment problems, goals might have very different features. In this study, an algorithm is suggested for multicriteria assignment problems that have secondary goals in addition to cost minimization and profit maximization. The fundamental logic of the algorithm is the solution of a classic assigment problem in which C_{ij} coefficients are modified to become suitable to goals, constraints and to the results of sensitivity analysis of C_{ij} coefficients with Hungarian algorithm. This logic can also be used for the solution of other assignment problems that have different goals. The aplicability of the algorithm is seen in the solution of multicriteria assignment problem that is constituted for work distribution problems of the firms which are joined to a contract.

1. GİRİŞ

Genel olarak, n sayıdaki işçinin m sayıdaki işe atanması şeklinde tanımlanabilen klasik atama probleminin doğrusal modeli aşağıdaki gibi yazılabilir (Doğan 1994: 173):

Burada; c_{ij}, i işçisinin j işine atanması maliyetini ifade eder. x_{ij} karar değişkeni olup, i işçisi j işine atanırsa $x_{ij}=1$, atanmazsa $x_{ij}=0$ değeri alır.

$$\text{Min.}Z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}\, x_{ij}$$

Kısıtlar:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i=1,2,...,m \qquad \text{(bir işçi sadece bir işe atanmalı)}$$

$$\sum_{i=1}^{m} x_{ij} = 1, \quad j=1.2,....n \qquad \text{(bir işe sadece bir işçi atanmalı)}$$

$$x_{ij} = 0 \text{ veya } 1$$

Hedef programlama probleminin doğrusal modeli ise aşağıdaki gibi yazılabilir (Lai ve Hwang 1994: 32):

* Yrd.Doç.Dr., Süleyman Demirel Üniversitesi İ.İ.B.F. İşletme Bölümü Öğretim Üyesi.

$$\text{Min.}Z = P_1(d_1^+, d_1^-) + P_2(d_2^+, d_2^-) + \ldots\ldots + P_k(d_k^+, d_k^-)$$

Kısıtlar:

$f_t(x) - d_t^+ + d_t^- = b_t$, $t=1,2\ldots,k$ (amaçlarla ilgili kısıtlar)

$g_i(x) \geq 0$, $i=1,2,\ldots,m$ (diğer kısıtlar)

$d_1^+ . d_1^- = 0$ $x, d_t^+, d_t^- \geq 0$ $t=1,2\ldots,k$

Yukarıdaki atama ve hedef programlama modeli dikkate alındığında, çok amaçlı bir atama probleminin hedef programlama modeli aşağıdaki gibi yazılabilir:

$$\text{Min.}Z = P_1(d_1^+,d_1^-) + P_2(d_2^+,d_2^-) + \ldots\ldots + P_k(d_k^+,d_k^-)$$

Kısıtlar:

$f_t(x) - d_t^+ + d_t^- = b_t$, $t=1,2\ldots,k$ (amaçlarla ilgili kısıtlar)

$\sum_{j=1}^{n} x_{ij} = 1$, $i=1,2,\ldots,m$ (bir işçi sadece bir işe atanmalı)

$\sum_{i=1}^{m} x_{ij} = 1$, $j=1,2,\ldots,n$ (bir işe sadece bir işçi atanmalı)

$d_1^+ . d_1^- = 0$; $x_{ij} = 0$ veya 1; $d_t^+, d_t^- \geq 0$

Burada; d_t^+, t amaçtan pozitif sapma değerini, d_t^- ise negatif sapma değerini gösterir. Bu sapma değişkenlerinden biri pozitif değer alırken diğerinin sıfır değer alması gerektiği için $d_1^+ . d_1^- = 0$ şartı da vardır. k tane amaç bulunan problemde P_t, t amacın öncelik seviyesini gösterir.

Hedef programlamada birinci öncelikli amaç $[P_1(d_1^+,d_1^-)]$ temeldir. Birinci öncelikli amaçla belirtilen istek maksimum düzeyde karşılanır. Sonra, birinci öncelikli amacın karşılanmışlık seviyesi düşürülmeden ikinci öncelikli amaçla belirtilen istek maksimum düzeyde karşılanır. Bu öncelik mantığı tüm amaçlar için ardışık olarak aynıdır. Yani, t öncelikli amaç t+1 öncelikli amaçtan önce dikkate alınır ve t amacın karşılanmışlık seviyesi, t+1 amaçla belirtilen isteği maksimize ederken bozulmaz (Kuruüzüm 1989: 40).

Çok amaçlı bir atama probleminin, 0-1 doğrusal hedef programlama modelinin kurulması ve optimum çözümünün bulunması çok fazla işlem ve zaman gerektirdiği (özellikle çok sayıda iş ve işçinin bulunduğu problemlerde) ortadadır. Örneğin; 30 iş ve işçinin bulunduğu bir atama probleminin hedef programlama modelinde; (30).(30)=900 tane 0-1 karar değişkeni, [(amaç sayısı).(2)] tane sapma değişkeni, (2).(30)=60 normal kısıt ve amaç sayısı kadar amaçlarla ilgili kısıtların yer alması gerekecektir.

Bu çalışmanın amacı, birden çok amacın bulunduğu atama problemlerinin optimum çözümünü macar (Kuhn 1955) algoritması ile bulabilmek için, bu problemlerin klasik atama problemi olarak nasıl düzenlenebileceğini araştırmaktır.

Yapılan literatür taraması sonucunda, çok amaçlı yapı gösteren atama problemleri ile ilgili çok az çalışma yapıldığı görülmüştür [Klingman ve Phillips (1984), Mazzola ve Neebe (1986), Phillips (1987), Aboudi ve Nemhauser (1991), Epstein ve Dickman (1994), Caron ve diğerleri (1999)]. Birden çok amacın bulunduğu atama problemlerinde amaçlar çok farklı özellikte olabilmektedir. Yapılan çalışmaların birbirinden farkı, ya farklı özellikte amaçları olan bir atama problemini incelemeleri ya da farklı çözüm prosedürlerinin araştırılması şeklindedir. Bu çalışmaların analizi dördüncü bölümde geniş olarak ele alınmaktadır.

2. ÖNERİLEN ALGORİTMA

Bu algoritma, maliyetin minimize veya karın maksimize edilmesiyle birlikte, ikinci derecede öncelikli başka amaçların da bulunduğu çok amaçlı atama problemlerinin çözümü için geliştirilmiştir. Algoritmadaki temel mantık, C_{ij} katsayılarının amaçlara, kısıtlara ve C_{ij} katsayılarının duyarlılık analizi sonuçlarına uygun şekilde yeniden düzenlenerek oluşturulan klasik atama probleminin macar algoritması ile çözülebilirliliğini sağlamaktır. Bu mantık, farklı özellikteki çok amaçlı atama problemlerinin çözümü için de kullanılabilmektedir. Algoritmanın adımları aşağıdaki gibidir:

1- Atama problemindeki C_{ij} katsayılarından P_1 değeri,

$P_1 = \min. |C_{ij} - C_{ij}|$ $(C_{ij} \neq C_{ij})$ (i=1,...,m) (j=1,...,n) olarak belirlenir.

2- $P_1 > P_2 > .... > P_k$ (k = amaç sayısı) olacak şekilde, amaç sayısı kadar P_t katsayıları belirlenir.

3- İkinci derecede öncelikli amaçtan itibaren t. öncelikli amacı ilgilendiren C_{ij} katsayılarına P_t değeri, söz konusu amacı yerine getirtmeye uygun olacak şekilde eklenir veya çıkarılır (t=2,...,k).

4- Adım 3'de belirlenen C_{ij} katsayıları dikkate alınarak, atama probleminin macar algoritması ile optimum çözümü bulunur.

Önerilen algoritmada, birinci derecede öncelikli amaca göre çözülen problemin C_{ij} katsayılarının duyarlılık analizi sonuçları dikkate alınmaktadır. C_{ij} Katsayılarının duyarlılık analizi ile, bu katsayıların optimum çözümü değiştirmeyen değişim aralıkları hesaplanabilir (Öztürk 1999: 247).

Optimum çözümü değiştirmek için C_{ij} katsayısına eklenebilecek en küçük sayı E_{ij} ve çıkarılabilecek en küçük sayı F_{ij} olsun. Optimum çözümde temelde yer almayan X_{ij} değişkenlerine ilişkin C_{ij} katsayılarının duyarlılık analizi sonuçlarına göre, minimizasyon amaçlı problemde $F_{ij}=0$ veya maksimizasyon amaçlı problemde $F_{ij}=0$ değerleri ortaya çıkarsa, bu katsayılara ilişkin değişkenlerin optimum çözümdeki C_{ij} - Z_{ij} değerleri de sıfır olacağından, o problemde seçenekli optimum (alternative optumum) çözüm var demektir (Taha, 1997, s:102). Yani, aynı maliyeti veya karı veren başka çözümler var demektir.

Minimizasyon amaçlı atama probleminde temel olmayan değişkenlerle ilgili ($F_{ij} \neq 0$ olan F_{ij}'lere ilişkin) C_{ij} katsayısından, $0 \leq V < F_{ij}$ olan bir V sayısının çıkarılması veya eklenmesi; temel değişkenlerle ilgili ($E_{ij} \neq 0$ olan E_{ij}'lere ilişkin) C_{ij} katsayısına, $0 \leq V < E_{ij}$ olan bir V sayısının eklenmesi veya çıkarılması durumunda bulunacak optimum çözüm, orijinal C_{ij} katsayılarına göre bulunan seçenekli optimum çözümlerden biri olacaktır. Buna göre; $D=\min(F_{ij}, E_{ij})$ ($F_{ij} \neq 0$ ve $E_{ij} \neq 0$) olan bir D değerinden küçük bir sayının, problemdeki tüm C_{ij} değerlerinden veya bir kısmından çıkarılması veya eklenmesi halinde bulunacak optimum çözüm de, orijinal C_{ij} katsayılarına göre bulunan seçenekli optimum çözümlerden biri olacaktır.

Bu bilgi, çok amaçlı atama problemlerinde birinci derecede öncelikli amacın dışındaki amaçların olabildiğince gerçekleşmesini sağlamak için, bu amaçları ilgilendiren C_{ij} katsayılarının yeniden düzenlenmesinde kullanılabilir. Ancak, D değerinin bulunması için, sadece birinci derecede öncelikli amacın dikkate alınması ile çözülen problemin C_{ij} katsayılarının duyarlılık analizi yapılması gerekir. Bu durumda çok amaçlı atama probleminin çözümü için iki aşamalı bir çözüm işlemi gerekecektir. Tek aşamada optimum çözüme ulaşabilmek için D değeri yerine kullanılabilecek bir P_1 değeri;

$$P_1 = \min. |C_{ij} - C_{ij}| \qquad (C_{ij} \neq C_{ij}) \quad (i=1,...,m) \quad (j=1,...,n)$$

olarak belirlenebilir. Burada $P_1 \leq \min(F_{ij}, E_{ij})$ olacağından, D yerine P_1 değerinin kullanılması aynı sonucu verecektir. Önerilen algoritmanın birinci adımındaki P_1 değerlerinin belirlenmesinde bu bilgi kullanılmıştır.

İkinci adımda belirlenen P_t değerlerinin birbirinden ardışık uzaklıklarının eşit veya farklı olması ve uzaklık miktarının kaç olduğu önemli değildir. Yani; $(P_t > P_{t-1})$ olması yeterli olup, $(P_t - P_{t-1})$ farkının miktarı ve bu farkların eşit olup olmaması önemli değildir. Çünkü, atama probleminde (X_{ij}=0 veya 1) şartı vardır. Başka bir deyişle, toplam atama maliyeti hesaplanırken C_{ij} katsayıları ya sıfır ile, ya da bir ile çarpılabilecektir. Toplam maliyete katkıları eşit olan iki katsayının (C_{ij}, C_{vw}) birinden ε gibi çok küçük bir sayının çıkarılması halinde ($C_{ij} - \varepsilon$), optimum çözüme X_{ij} değişkeni yerine X_{vw} değişkeni girecektir (satırlardaki ve sütunlardaki değişkenlerin toplamının bire eşit olması gereği sağlanıyorsa). Bu nedenle, algoritmanın ikinci adımında belirlenen P_t katsayıları üçüncü adımda belirtildiği şekilde ilgili C_{ij} katsayılarından çıkarılır veya toplanır ise, bulunacak optimum çözümde öncelik sırasına göre bütün amaçlar olabildiğince sağlanmış olacaktır.

Çok amaçlı atama probleminin çözümündeki temel mantık; t derecede öncelikli amacın olabildiğince sağlanması için, kendinden daha öncelikli olan amaçların dikkate alınması ile bulunacak seçenekli optimal çözümlerden en uygun olanının seçilmesidir (Hiller ve Lieberman 1990: 271). Önerilen algoritmada da bu mantık bulunmaktadır.

Bu algoritma ile bulunan minimum maliyet değeri, orijinal C_{ij} katsayılarının değerinde bazı değişiklikler yapılmış olmasından dolayı gerçek atama maliyetini vermeyecektir. Gerçek atama maliyeti, optimum çözümdeki temel değişkenlerin orijinal C_{ij} katsayılarının toplanması ile bulunabilir.

3. UYGULAMA

Karayolları Isparta Teşkilatı, her yıl ihtiyacı olduğu kadar otomobili çeşitli işlerde bir dönem boyunca kullanmak üzere kiralıyor. Kiralama işi kapalı zarf usulü ile yapılmaktadır. Geçmiş yıllarda yapılan bir ihale için verilen teklifler (1000TL/Km) olarak Tablo.1'de verilmiştir. İhaleye 9 iş açılmasına karşın, 11 otomobil sahibi girmiştir. Bu nedenle Tablo.1'e 10 ve 11 numaralı suni işler eklenmiş ve bu işler için verilen tekliflere sıfır değeri verilmiştir.

İhaleyi yapan teşkilat, aşağıda öncelik sırasına göre verilen 6 farklı amaca olabildiğince uygun olmak üzere, bir işi bir otomobil sahibine vermek istemektedir.

Öncelik sırasına göre amaçlar:

1. Toplam taşıma maliyetini minimize etmek,
2. 1,4,8,10 numaralı otomobil sahipleri geçmiş yıllarda denenmiş ve beğenilmiş olduklarından, bunlara verilen iş sayısını maksimize etmek,
3. 1,2,3,4 numaralı işler ile ilgili yöneticiler 1,2,5,8,10 numaralı otomobil sahiplerinden biriyle çalışmak istediğinden; 1,2,3,4 numaralı işlerden

1,2,5,8,10 numaralı otomobil sahiplerine verilenlerin sayısını maksimize etmek,

4. 7,8,9 numaralı işler ile ilgili yöneticiler 6 veya 7 numaralı otomobil sahipleriyle çalışmak istemediğinden; 7,8,9 numaralı işlerden 6 ve 7 numaralı otomobil sahiplerine verilenlerin sayısını minimize etmek,

Tablo 1. Uygulama Probleminin Verileri

	İhaleye açılan işler											
İhaleye giren otomobil sahipleri	i \ j	1	2	3	4	5	6	7	8	9	10	11
	1	100	160	120	140	150	150	160	150	150	0	0
	2	120	120	140	150	150	160	150	160	120	0	0
	3	150	190	150	120	160	200	160	140	160	0	0
	4	110	120	170	160	130	160	140	180	170	0	0
	5	100	140	110	170	180	180	160	110	180	0	0
	6	130	180	180	180	190	150	120	140	170	0	0
	7	150	150	140	160	130	160	150	120	120	0	0
	8	150	120	160	160	140	180	160	150	140	0	0
	9	120	140	130	160	140	150	180	130	140	0	0
	10	100	120	140	140	160	160	180	140	160	0	0
	11	130	150	120	120	160	180	140	160	120	0	0

5. 9 ve 11 numaralı otomobil sahipleri geçmiş yıllarda denenmiş ve bazı küçük sorunlar ortaya çıkmış olduğundan, bu firmalara verilen iş sayısını minimize etmek,

6. 2 numaralı iş ile ilgili yönetici 4 numaralı otomobil sahibi ile çalışmak istediğinden, bu isteği maksimize etmek.

Önerilen algoritmanın birinci adımındaki P_1 değeri,

P_1 = min. $|C_{ij} - C_{ij}| = |C_{1.1} - C_{1.5}| = |100-110| = 10$ ($C_{ij} \neq C_{ij}$) olarak bulunur. Bu değer, birinci öncelikli amacın dikkate alınması ile çözülen problemin Ek.1'deki duyarlılık analizi sonuçlarıyla karşılaştırıldığında, $P_1 \leq \min(F_{ij}, E_{ij})$ ($F_{ij} \neq 0$ ve $F_{ij} \neq 0$) olduğu görülmektedir. ($P_1=10$)

Problemde 6 farklı amaç bulunduğundan, algoritmanın ikinci adımındaki diğer P_t sayıları; $P_2=5$, $P_3=4$, $P_4=3$, $P_5=2$, $P_6=1$ olarak belirlenebilir.

Üçüncü adıma göre:

İkinci öncelikli amaç için;	$C_{ij} + 5$	(i=1,4,8,10 ve j=10,11)
Üçüncü öncelikli amaç için;	$C_{ij} - 4$	(i=1,2,5,8,10 ve j=1,2,3,4)
Dördüncü öncelikli amaç için;	$C_{ij} + 3$	(i=6,7 ve j=7,8,9)
Beşinci öncelikli amaç için;	$C_{ij} - 2$	(i=9,11 ve j=10,11)
Altıncı öncelikli amaç için;	$C_{4.2} - 1$	

işlemleri yapılarak Algoritmanın üçüncü adımına uygun olarak düzenlenen atama tablosu Tablo.2' de verilmiştir. Tablodaki koyu renkli hücreler, düzenleme yapılan hücreleri göstermektedir. Amaçların öncelik sevileri ve amaçlara ilişkin katsayı düzenlemeleri yapılırken gereken titizliğin gösterilmesi gerekmektedir.

Tablo.2'deki veriler, klasik atama problemi olarak ele alınıp Macar Algoritması ile çözüldüğünde aşağıdaki sonuçlar elde edilmiştir (çözüm için QSA paket programı kullanılmıştır):

Min.Z=1079 $X_{1.3}=1$ $X_{2.9}=1$ $X_{3.4}=1$ $X_{4.5}=1$ $X_{5.8}=1$

$X_{6.7}=1$ $X_{7.11}=1$ $X_{8.2}=1$ $X_{9.6}=1$ $X_{10.1}=1$ $X_{11.10}=1$

İterasyon sayısı = 5 olarak gözlenmiştir.

Buradaki MinZ değeri değiştirilmiş C_{ij} katsayılarından hesaplandığı için gerçek taşıma maliyetini göstermeyeceğinden, gerçek taşıma maliyetinin optimum çözümde yer alan X_{ij} değişkenlerinin orijinal C_{ij} katsayılarından hesaplanması gerekmekte ve bu değer,

Min.Z*=120+120+120+130+110+120+0+120+150+100+0=1090

olmaktadır.

Tablo 2. Katsayıları Amaçlara Uygun Şekilde Değiştirilmiş Atama Tablosu

	İhaleye açılan işler											
İhaleye giren otomobil sahipleri	i \ j	1	2	3	4	5	6	7	8	9	10	11
	1	96	156	116	136	150	150	160	150	150	5	5
	2	116	116	136	146	150	160	150	160	120	0	0
	3	150	190	150	120	160	200	160	140	160	0	0
	4	110	119	170	160	130	160	140	180	170	5	5
	5	96	136	106	166	180	180	160	110	180	0	0
	6	130	180	180	180	190	150	123	143	173	0	0
	7	150	150	140	160	130	160	153	123	123	0	0
	8	146	116	156	156	140	180	160	150	140	5	5
	9	120	140	130	160	140	150	180	130	140	-2	-2
	10	96	116	136	136	160	160	180	140	160	5	5
	11	130	150	120	120	160	180	140	160	120	-2	-2

Optimum çözüm sonuçlarına göre amaçların karşılanmışlık seviyesi incelendiğinde:

- İkinci öncelikli amacın tam olarak karşılandığı (1,4,8,10 numaralı otomobil sahiplerinden hiç birine 10 ve 11 numaralı suni işler verilmemiştir),

- Üçüncü öncelikli amacın, ($X_{1.3} = X_{8.2} = X_{10.1} = 1$ ve $X_{3.4} = 1$olduğundan) ¾ oranında karşılanabildiği,

- Dördüncü öncelikli amacın, ($X_{6.7} = 1$ ve $X_{7.11} = 1$ olduğundan) ½ oranında karşılanabildiği,

- Beşinci öncelikli amacın, ($X_{11.10} = 1$ ve $X_{9.6} = 1$ olduğundan) ½ oranında karşılanabildiği,

- Altıncı öncelikli amacın ise hiç karşılanamadığı ($X_{4.2} = 0$) görülmektedir.

Çok sayıda amaç içeren bu atama problemi, *0-1 tamsayılı doğrusal hedef programlama modeli olarak da düzenlenebilir:*

$\mathbf{Min.Z}= P_1 d_1^+ + P_2 d_2^+ + P_3 d_3^- + P_4 d_4^+ + P_5 d_5^- + P_6 d_6^-$

Kısıtlar:

$100X_{1,1}+160X_{1,2}+120X_{1,3}+140X_{1,4}+150X_{1,5}+150X_{1,6}+160X_{1,7}+150X_{1,8}+150X_{1,9}+120X_{2,1}$
$+120X_{2,2}+140X_{2,3}+150X_{2,4}+150X_{2,5}+160X_{2,6}+150X_{2,7}+160X_{2,8}+120X_{2,9}+150X_{3,1}+190X_{3,2}$
$+150X_{3,3}+120X_{3,4}+160X_{3,5}+200X_{3,6}+160X_{3,7}+140X_{3,8}+160X_{3,9}+110X_{4,1}+120X_{4,2}+170X_{4,3}$
$+160X_{4,4}+130X_{4,5}+160X_{4,6}+140X_{4,7}+180X_{4,8}+170X_{4,9}+100X_{5,1}+140X_{5,2}+110X_{5,3}+170X_{5,4}$
$+180X_{5,5}+180X_{5,6}+160X_{5,7}+110X_{5,8}+180X_{5,9}+130X_{6,1}+180X_{6,2}+180X_{6,3}+180X_{6,4}+190X_{6,5}$
$+150X_{6,6}+120X_{6,7}+140X_{6,8}+170X_{6,9}+150X_{7,1}+150X_{7,2}+140X_{7,3}+160X_{7,4}+130X_{7,5}+160X_{7,6}$
$+150X_{7,7}+120X_{7,8}+120X_{7,9}+150X_{8,1}+120X_{8,2}+160X_{8,3}+160X_{8,4}+140X_{8,5}+180X_{8,6}+160X_{8,7}$
$+150X_{8,8}+140X_{8,9}+120X_{9,1}+140X_{9,2}+130X_{9,3}+160X_{9,4}+140X_{9,5}+150X_{9,6}+180X_{9,7}+130X_{9,8}$
$+140X_{9,9}+100X_{10,1}+120X_{10,2}+140X_{10,3}+140X_{10,4}+160X_{10,5}+160X_{10,6}+180X_{10,7}+140X_{10,8}$
$+160X_{10,9}+130X_{11,1}+150X_{11,2}+120X_{11,3}+120X_{11,4}+160X_{11,5}+180X_{11,6}+140X_{11,7}+160X_{11,8}$
$+120X_{11,9}-\mathbf{d_1^+}+\mathbf{d_1^-} = \mathbf{0}$

$X_{1,10}+X_{1,11}+X_{4,10}+X_{4,11}+X_{8,10}+X_{8,11}+X_{10,10}+X_{10,11}-\mathbf{d_2^+}+\mathbf{d_2^-} = \mathbf{0}$

$X_{1,1}+X_{1,2}+X_{1,3}+X_{1,4}+X_{2,1}+X_{2,2}+X_{2,3}+X_{2,4}+X_{5,1}+X_{5,2}+X_{5,3}+X_{5,4}+X_{8,1}+X_{8,2}+X_{8,3}+X_{8,4}+X_{10,1}$
$+X_{10,2}+X_{10,3}+X_{10,4}-\mathbf{d_3^+}+\mathbf{d_3^-}=\mathbf{4}$

$X_{6,7}+X_{6,8}+X_{6,9}+X_{7,7}+X_{7,8}+X_{7,9}-\mathbf{d_4^+}+\mathbf{d_4^-}=\mathbf{0}$

$X_{9,10}+X_{9,11}+X_{11,10}+X_{11,11}-\mathbf{d_5^+}+\mathbf{d_5^-}=\mathbf{2}$

$X_{4,2}-\mathbf{d_6^+}+\mathbf{d_6^-}=\mathbf{1}$

$X_{1,1}+X_{1,2}+X_{1,3}+X_{1,4}+X_{1,5}+X_{1,6}+X_{1,7}+X_{1,8}+X_{1,9}+X_{1,10}+X_{1,11}=\mathbf{1}$
$X_{2,1}+X_{2,2}+X_{2,3}+X_{2,4}+X_{2,5}+X_{2,6}+X_{2,7}+X_{2,8}+X_{2,9}+X_{2,10}+X_{2,11}=\mathbf{1}$
$X_{3,1}+X_{3,2}+X_{3,3}+X_{3,4}+X_{3,5}+X_{3,6}+X_{3,7}+X_{3,8}+X_{3,9}+X_{3,10}+X_{3,11}=\mathbf{1}$
$X_{4,1}+X_{4,2}+X_{4,3}+X_{4,4}+X_{4,5}+X_{4,6}+X_{4,7}+X_{4,8}+X_{4,9}+X_{4,10}+X_{4,11}=\mathbf{1}$
$X_{5,1}+X_{5,2}+X_{5,3}+X_{5,4}+X_{5,5}+X_{5,6}+X_{5,7}+X_{5,8}+X_{5,9}+X_{5,10}+X_{5,11}=\mathbf{1}$
$X_{6,1}+X_{6,2}+X_{6,3}+X_{6,4}+X_{6,5}+X_{6,6}+X_{6,7}+X_{6,8}+X_{6,9}+X_{6,10}+X_{6,11}=\mathbf{1}$
$X_{7,1}+X_{7,2}+X_{7,3}+X_{7,4}+X_{7,5}+X_{7,6}+X_{7,7}+X_{7,8}+X_{7,9}+X_{7,10}+X_{7,11}=\mathbf{1}$
$X_{8,1}+X_{8,2}+X_{8,3}+X_{8,4}+X_{8,5}+X_{8,6}+X_{8,7}+X_{8,8}+X_{8,9}+X_{8,10}+X_{8,11}=\mathbf{1}$
$X_{9,1}+X_{9,2}+X_{9,3}+X_{9,4}+X_{9,5}+X_{9,6}+X_{9,7}+X_{9,8}+X_{9,9}+X_{9,10}+X_{9,11}=\mathbf{1}$
$X_{10,1}+X_{10,2}+X_{10,3}+X_{10,4}+X_{10,5}+X_{10,6}+X_{10,7}+X_{10,8}+X_{10,9}+X_{10,10}+X_{10,11}=\mathbf{1}$
$X_{11,1}+X_{11,2}+X_{11,3}+X_{11,4}+X_{11,5}+X_{11,6}+X_{11,7}+X_{11,8}+X_{11,9}+X_{11,10}+X_{11,11}=\mathbf{1}$
$X_{1,1}+X_{2,1}+X_{3,1}+X_{4,1}+X_{5,1}+X_{6,1}+X_{7,1}+X_{8,1}+X_{9,1}+X_{10,1}+X_{11,1}=\mathbf{1}$
$X_{1,2}+X_{2,2}+X_{3,2}+X_{4,2}+X_{5,2}+X_{6,2}+X_{7,2}+X_{8,2}+X_{9,2}+X_{10,2}+X_{11,2}=\mathbf{1}$
$X_{1,3}+X_{2,3}+X_{3,3}+X_{4,3}+X_{5,3}+X_{6,3}+X_{7,3}+X_{8,3}+X_{9,3}+X_{10,3}+X_{11,3}=\mathbf{1}$
$X_{1,4}+X_{2,4}+X_{3,4}+X_{4,4}+X_{5,4}+X_{6,4}+X_{7,4}+X_{8,4}+X_{9,4}+X_{10,4}+X_{11,4}=\mathbf{1}$
$X_{1,5}+X_{2,5}+X_{3,5}+X_{4,5}+X_{5,5}+X_{6,5}+X_{7,5}+X_{8,5}+X_{9,5}+X_{10,5}+X_{11,5}=\mathbf{1}$
$X_{1,6}+X_{2,6}+X_{3,6}+X_{4,6}+X_{5,6}+X_{6,6}+X_{7,6}+X_{8,6}+X_{9,6}+X_{10,6}+X_{11,6}=\mathbf{1}$
$X_{1,7}+X_{2,7}+X_{3,7}+X_{4,7}+X_{5,7}+X_{6,7}+X_{7,7}+X_{8,7}+X_{9,7}+X_{10,7}+X_{11,7}=\mathbf{1}$
$X_{1,8}+X_{2,8}+X_{3,8}+X_{4,8}+X_{5,8}+X_{6,8}+X_{7,8}+X_{8,8}+X_{9,8}+X_{10,8}+X_{11,8}=\mathbf{1}$

$X_{1,9}+X_{2,9}+X_{3,9}+X_{4,9}+X_{5,9}+X_{6,9}+X_{7,9}+X_{8,9}+X_{9,9}+X_{10,9}+X_{11,9}=\mathbf{1}$
$X_{1,10}+X_{2,10}+X_{3,10}+X_{4,10}+X_{5,10}+X_{6,10}+X_{7,10}+X_{8,10}+X_{9,10}+X_{10,10}+X_{11,10}=\mathbf{1}$
$X_{1,11}+X_{2,11}+X_{3,11}+X_{4,11}+X_{5,11}+X_{6,11}+X_{7,11}+X_{8,11}+X_{9,11}+X_{10,11}+X_{11,11}=\mathbf{1}$

$\mathbf{X_{ij} = 0}$ veya $\mathbf{1}$; $\mathbf{d_j^+ , d_j^- \geq 0}$

Bu problem LINDO paket programı ile çözülmüş ve optimum çözüm sonuçları Tablo 3.'de verilmiştir.

Atama modeli ve hedef programlama modeli ile yapılan çözümler karşılaştırıldığında görüleceği gibi, iki modelle yapılan çözüm de aynı sonucu vermektedir. Ancak, problemin çözümü atama modeli ile 5 iterasyon sürmesine karşılık, hedef programlama modeli ile 31 iterasyon sürmüştür.

Tablo 3. Uygulama Probleminin Optimum Çözüm Sonuçları

VARIABLE	VALUE	REDUCED COST
d_1	1090.000	0.000000
d_3^-	1.000000	0.000000
d_4	1.000000	0.000000
d_5^-	1.000000	0.000000
d_6^-	1.000000	0.000000
$X_{1,3}$	1.000000	0.000000
$X_{2,9}$	1.000000	0.000000
$X_{3,4}$	1.000000	0.000000
$X_{4,5}$	1.000000	0.000000
$X_{5,8}$	1.000000	0.000000
$X_{6,7}$	1.000000	0.000000
$X_{7,11}$	1.000000	0.000000
$X_{8,2}$	1.000000	0.000000
$X_{9,6}$	1.000000	0.000000
$X_{10,1}$	1.000000	0.000000
$X_{11,10}$	1.000000	0.000000
NO. ITERATIONS=	31	

İki modelin aynı paket program ile çözüm olanağı bulunamadığından, çözüm sürelerinin karşılaştırılması yapılmadı. Ancak, iterasyon sayıları arasındaki fark ve atama modelinin bilinen çözüm üstünlükleri dikkate alındığında, atama modelinin çözüm süresinin önemli bir oranda düşük olacağı söylenebilir. Atama modelinin elle çözümü bile rahatça yapılabilir olmasına karşılık, hedef programlama modelinin elle çözümünün oldukça zor olacağı ortadadır. Ayrıca, ele alınan problemin atama modeli olarak düzenlenmesinin, hedef programlama modeli olarak düzenlenmesinden daha kolay ve daha az zaman gerektirdiği de gözlenmiştir.

4. YAPILAN ÇALIŞMALARIN ANALİZİ VE ÖNERİLER

Çok amaçlı atama problemleri ile ilgili az sayıda çalışma yapıldığı görülmektedir [(Klingman ve Phillips 1984), (Mazzola ve Neebe 1986), (Phillips 1987), (Aboudi ve Nemhauser 1991), (Epstein ve Dickman 1994), (Caron ve diğerleri, 1999)]. Yapılan çalışmaların bir birinden farklı tarafı, ya farklı özellikte amaçları olan bir atama problemini incelemeleri ya da farklı çözüm prosedürlerinin araştırılması şeklindedir. Ancak, bu çalışmada ele alınan özellikte amaçlar içeren atama problemleri ile ilgili bir çalışma bulunamadı.

Epstein ve Dickman (1994)'in yaptığı çalışmada üç farklı $C_k(ij)$ katsayısının bulunduğu bir çok amaçlı atama problemi ele alınmaktadır:

$$C_1 = \begin{bmatrix} 2 & 4 & 2 \\ 1 & 2 & 1 \\ 3 & 1 & 0 \end{bmatrix} \qquad C_2 = \begin{bmatrix} 3 & 1 & 4 \\ 2 & 5 & 3 \\ 1 & 2 & 1 \end{bmatrix} \qquad C_3 = \begin{bmatrix} 4 & 2 & 2 \\ 3 & 2 & 3 \\ 2 & 5 & 2 \end{bmatrix}$$

Üç işçinin üç makineye atanması ile ilgili bu problemde; C_1, işçilerin makinelere atama maliyet matrisi, C_2, belli bir işi makinede bitirme zamanı matrisi ve C_3, birim zamanda yapılan hata sayıları matrisidir. Önem sırasına göre amaçlar; 1) maliyeti minimize etmek, 2) bitirme zamanını minimize etmek, 3) hata sayılarını minimize etmek.

Epstein ve Dickman bu problemin optimum çözümünü, macar algoritmasını kullanarak üç aşamada yapmışlar. İlk aşamada, birinci öncelikli katsayılar matrisi dikkate alınarak seçenekli optimum çözümler bulunmuştur. İkinci aşamada, birinci aşamada bulunan seçenekli optimum çözümlerde yer alan değişkenler ve bu değişkenlerin ikinci öncelikli katsayıları dikkate alınarak (birinci çözümdeki seçenekli optimum çözümlerde yer almayan değişkenler sıfır kabul edilerek) ikinci öncelikli amaca göre seçenekli optimum çözümler bulunmuştur. Aynı şekilde, üçüncü öncelikli amaç da ikinci öncelikli amaca göre bulunan seçenekli optimum çözümlerde yer alan değişkenler ve bu değişkenlerin üçüncü öncelikli katsayıları dikkate alınarak üçüncü öncelikli amaç minimize edilmiş ve çok amaçlı atama probleminin optimum çözümü bulunmuştur. Yani, bir öncelikli amaca göre bulunan seçenekli optimum çözümlerin kullanılması ile bir sonraki önceliğe sahip olan amaç optimize edilerek optimum çözüme ulaşılmaktadır. Üç amacın bulunduğu bu problem için yapılan üç aşamalı çözüm aşağıdaki gibi olmaktadır:

$$C_1^* = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix} \qquad C_2^* = \begin{bmatrix} 0 & - & 0 \\ 0 & 1 & 0 \\ - & 0 & 0 \end{bmatrix} \qquad C_3^* = \begin{bmatrix} 2 & - & \mathbf{0} \\ \mathbf{0} & - & 0 \\ - & \mathbf{0} & 0 \end{bmatrix}$$

Optimum çözüme göre, 1,2,3 numaralı işçiler sırayla 3,1,2 numaralı makinelere atanacaktır. Bu çözüm prosedürüne göre macar algoritması kullanılmakta fakat amaç sayısı kadar yeniden çözüm gerekmektedir.

Epstein ve Dickman' in ele aldığı böyle bir atama probleminin, bu çalışmada önerilen algoritmada kullanılan "problemin genel yapısını bozmadan katsayıların amaçlara uygun şekilde yeniden düzenlenmesi" yoluyla tek aşamada optimum çözümü bulunabilir. Buna göre, üç ayrı katsayılar matrisi aşağıdaki şekilde birleştirilmiş tek matris halinde düzenlenebilir:

$$C = (C_1 . N_1 . 10) + (C_2 . N_2 . 10) + (C_3)$$

$$N_3 = (C_3\text{'deki en büyük katsayının basamak sayısı}) - 1$$

$$N_2 = N_3 + (C_2\text{'deki en büyük katsayının basamak sayısı})$$

$$N_1 = N_2 + (C_1\text{'deki en büyük katsayının basamak sayısı})$$

$$C = \begin{bmatrix} 200 & 400 & 200 \\ 100 & 200 & 100 \\ 300 & 100 & 0 \end{bmatrix} + \begin{bmatrix} 30 & 10 & 40 \\ 20 & 50 & 30 \\ 10 & 20 & 10 \end{bmatrix} + \begin{bmatrix} 4 & 2 & 2 \\ 3 & 2 & 3 \\ 2 & 5 & 2 \end{bmatrix} = \begin{bmatrix} 234 & 412 & 242 \\ 123 & 252 & 133 \\ 312 & 125 & 12 \end{bmatrix}$$

Bu şekilde düzenlenen katsayı matrisi dikkate alınarak kurulacak atama probleminin optimum çözümünde bütün amaçlar önem sırasına göre dikkate alınmış olacaktır. Çünkü, minimizasyon amacının olduğu bu problemde birinci derecede öncelikli amaca ilişkin katsayılar yüz, ikinci derecede öncelikli amaca ilişkin katsayılar on, üçüncü derecede öncelikli amaca ilişkin katsayılar bir ile çarpılarak önem derecelerine göre dikkate alınabilmesi sağlanmıştır. Klasik atama problemi haline getirilmiş bu problemin macar algoritması ile bulunan optimum çözümü aşağıdadır (optimum çözümde yer alan değişkenlerin yerleri koyu renkli sıfırlarla belirtilmiştir):

$$C = \begin{bmatrix} 0 & 57 & \mathbf{0} \\ \mathbf{0} & 8 & 2 \\ 308 & \mathbf{0} & 0 \end{bmatrix}$$

Burada elde edilen optimum çözüm (1,2,3 numaralı işçiler sırayla 3,1,2 numaralı makinelere atanacaktır), Epstein ve Dickman' in bulduğu optimum çözüm ile aynıdır.

Epstein ve Dickman, yaptıkları çalışmanın Klingman ve Phillips (1984)'in yaptığı çalışmada kullandığı metodun bir adaptasyonu olduğunu ifade etmişlerdir. Klingman ve Phillips, askeri personelin çeşitli görevlere atanması ile ilgili 780000 değişkenli 11 amaçlı bir problemi, klasik atama problemi olarak değil hedef programlama modeli olarak ele alıp optimum çözümünü araştırmışlardır.

Phillips (1987) de birden çok katsayı matrisinin bulunduğu çok amaçlı atama problemini ele almış, fakat problemi hedef programlama modeli olarak kurarak, öncelikli amaçlarda ağırlıkların nasıl belirlenmesi gerektiğini araştırmıştır. Problemin optimum çözümünü, hedef programlama modelinin çözümüyle elde etmiştir.

Caron ve diğerleri (1999), kıdem ve iş öncelikli kısıtlar içeren atama problemlerinin çözümünü araştırmışlardır. Örneğin, kıdem öncelikli bir problem aşağıdaki gibi gösterilmiştir:

$$a_{ij} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Dört işçinin üç işe atanacağı bu problemde; a_{ij} , i işçisinin kalifiye özelliği açısından j işine atanması uygunsa $a_{ij}=1$, değilsi $a_{ij}=0$ olarak düzenlenmiştir. Bir işçi ancak kalifiye özelliği açısından uygun olduğu bir işe atanabilir. Atamada, işçilerin kalifiye seviyeleri de dikkate alınmalı ve kalifiye seviyesine göre öncelik verilmelidir. Yani, daha öncelikli bir işçinin atanmasına uygun bir iş var iken, bu işçi boşta bırakılıp daha az öncelikli bir işçinin bir işe atanması istenmemektedir. Ayrıca, işlere atanan toplam işçi sayısı da maksimize edilmek istenmektedir.

Caron ve diğerleri, bu yapıdaki bir probleme iş öncelikli durumu da eklemişler ve hastanelerde yedek hemşirelerin çeşitli servislerde geçici olarak boşalmış (hastalanma, izin vs sebeplerle) hemşirelik işlerine atanması sorununun çözümünü ele almışlardır. Optimum çözümün bulunması için C_{ij} katsayılarında, ele alınan problemdeki kıdem ve iş önceliği ile ilgili isteğe ve atanan hemşire sayısının maksimum olmasını sağlamaya uygun şekilde değiştirmeler yaparak problemi macar algoritması ile çözülebilen klasik atama problemi olarak düzenlemişlerdir. Mazzola ve Neebe (1986) ve Aboudi ve Nemhauser (1991), benzer özellikteki problemlerin doğrusal programlama modelini kurmuşlar ve çözümde dal-sınır algoritması veya kesme düzlemi algoritması kullanmışlardır. Birden çok amaç içeren böyle bir problem de, çok amaçlı atama problemleri sınıfında düşünülebilir. Ancak, daha önce ele alınan problemlerden oldukça farklı bir yapıya sahiptir. Bu nedenle, literatürde bu yapıdaki problemler, "Kıdem ve İş Öncelikli Kısıtlı Atama Problemi" veya "Yanlı-Kısıtlı Atama Problemi" (Assignment Problem With Side-Konstraints) şeklinde farklı isimlerle ifade edilmektedir [Mazzola ve Neebe (1986), Aboudi ve Nemhauser (1991), Caron G. ve diğerleri (1999)].

5. SONUÇ

Gerçek hayatta karşılaşılan atama problemlerinin bir kısmında birden çok amaç bulunabilmektedir. Çok amaçlı atama problemi olarak ele alınan bu problemlerde amaçlar çok farklı özellikte olabilmektedir.

Her çeşit çok amaçlı atama probleminin 0-1 doğrusal hedef programlama modelinin kurulabileceği ve bu modelin uygun bir algoritma ile optimum çözümünün bulunabileceği söylenebilir. Ancak, birden çok amaç içeren bir atama probleminin klasik atama problemi olarak düzenlenmesi ve macar algoritması ile çözümün araştırılması, hedef programlama modeline göre daha az işlem ve çözüm zamanı gerektirmektedir.

Bu çalışmada, maliyetin minimize veya karın maksimize edilmesiyle birlikte, ikinci derecede öncelikli başka amaçların da bulunduğu çok amaçlı atama problemlerinin çözümü için bir algoritma önerilmiştir. Algoritmadaki temel mantık, C_{ij} katsayılarının amaçlara, kısıtlara ve C_{ij} katsayılarının duyarlılık analizi sonuçlarına uygun şekilde yeniden düzenlenerek oluşturulan klasik atama probleminin macar algoritması ile çözülebilirliliğini sağlamaktır. Bu mantık, farklı özellikte amaçlar içeren diğer atama problemlerinin çözümü için de kullanılabilmektedir.

Kapalı zarf usulü ile yapılan ve çok sayıda işin bulunduğu bir ihalede, işlerin firmalara dağıtılması sorununun çözümünde, önerilen algoritmanın uygulanabilirliği gözlenmiştir. Ayrıca, birden çok cij katsayı matrisinin dikkate alındığı çok amaçlı atama probleminin çözümünde de, bu algoritmadaki mantığın kullanılmasıyla tek aşamada optimum çözümün bulunabileceği görülmüştür.

KAYNAKÇA

ABOUDI, R. ve NEMHAUSER, G.L. (1991), "Some Facets For An Assignment Problem With Side Constraints", Operations Research, 39, pp. 244-250.

CARON, G. ve HANSEN, P. ve JAUMARD, B. (1999), "The Assignment Problem With Seniority and Job Priority Constraints", Operations Research, 47, pp. 449-453.

DOĞAN, İ. (1994), Yöneylem Araştırması Teknikleri ve İşletme Uygulamaları, Marmara Üniversitesi İktisadi ve İdari Bilimler Fakültesi yayını, İstanbul.

EPSTEIN, S. ve DİCKMAN, B. (1994), "Multicriteria Assignment Problems With Preemptive Priorities", The Mid-Atlantic Journal of Business, 3, pp. 113-120.

HİLLER, F.S. ve LIEBERMAN, G.J. (1990), Introduction to Operations Research, Fifth Edition, McGraw-Hill.

KLİNGMAN, D. ve PHİLLİPS, N.V. (1984), "Topological and Computational Aspects of Preemptive Multicriteria Military Personnel Assignment Problems", Management Science, 30, pp. 1362-1375.

KUHN, H. W. (1955), "The Hungarian Method For The Assignment Problem", Naval Res. Logistics Quart., 2, pp. 83-97.

KURUÜZÜM, O. (1989), Proses Endüstrisinde Proses Kontrolü Problemine Hedef Programlama İle Yaklaşım ve Alternatif Bir HP Algoritması Önerisinin Bir Uygulama Üzerinde Değerlendirilmesi, Doktora Tezi, İ.T.Ü.

LAI, Y.J. ve HWANG, C.L. (1994), Fuzzy Multiple Objective Decision Making, Methods and Applications, Sipringer-Verlag Berlin Heidelberg.

MAZZOLA, J. ve NEEBE, A.W. (1986), "Resource-Constrained Assignment Scheduling", Operations Research, 34, pp. 560-572.

ÖZTÜRK, A. (1999), Yöneylem Araştırması, Altıncı baskı, Ekin Kitabevi, Bursa.

PHILLIPS, N.V. (1987), "A Weighting Function For Pre-emptive Multicriteria Assignment Problems", Journal of the Operational Research Society, 38, pp. 797-802.

TAHA, H.A. (1997), Operations Research an Introduction, A Viacom Company.

Ek 1. Uygulama probleminde birinci öncelikli amaca göre bulunan optimum çözümdeki amaç katsayılarının duyarlılık analizi sonuçları

I	II	III	IV	I	II	III	IV	I	II	III	IV
X1,1	100	S	0	X4,9	170	S	50	X8,5	140	S	10
X1,2	160	S	40	X4,10	0	S	0	X8,6	180	S	30
X1,3	120	S	0	X4,11	0	S	0	X8,7	160	S	20
X1,4	140	S	20	X5,1	100	S	10	X8,8	150	S	30
X1,5	150	S	20	X5,2	140	S	30	X8,9	140	S	20
X1,6	150	S	0	X5,3	110	S	0	X8,10	0	0	S
X1,7	160	S	20	X5,4	170	S	60	X8,11	0	S	0
X1,8	150	S	30	X5,5	180	S	60	X9,1	120	S	20
X1,9	150	S	30	X5,6	180	S	40	X9,2	140	S	20
X1,10	0	S	0	X5,7	160	S	30	X9,3	130	S	10
X1,11	0	0	S	X5,8	110	0	S	X9,4	160	S	40
X2,1	120	S	20	X5,9	180	S	70	X9,5	140	S	10
X2,2	120	S	0	X5,10	0	S	10	X9,6	150	0	S
X2,3	140	S	20	X5,11	0	S	10	X9,7	180	S	40
X2,4	150	S	30	X6,1	130	S	50	X9,8	130	S	10
X2,5	150	S	20	X6,2	180	S	80	X9,9	140	S	20
X2,6	160	S	10	X6,3	180	S	80	X9,10	0	S	0
X2,7	150	S	10	X6,4	180	S	80	X9,11	0	S	0
X2,8	160	S	40	X6,5	190	S	80	X10,1	100	0	S
X2,9	120	0	S	X6,6	150	S	20	X10,2	120	S	0
X2,10	0	S	0	X6,7	120	20	S	X10,3	140	S	20
X2,11	0	S	0	X6,8	140	S	40	X10,4	140	S	20
X3,1	150	S	50	X6,9	170	S	70	X10,5	160	S	30
X3,2	190	S	70	X6,10	0	S	20	X10,6	160	S	10
X3,3	150	S	30	X6,11	0	S	20	X10,7	180	S	40
X3,4	120	0	S	X7,1	150	S	50	X10,8	140	S	20
X3,5	160	S	30	X7,2	150	S	30	X10,9	160	S	40
X3,6	200	S	50	X7,3	140	S	20	X10,10	0	S	0
X3,7	160	S	20	X7,4	160	S	40	X10,11	0	S	0
X3,8	140	S	20	X7,5	130	0	S	X11,1	130	S	30
X3,9	160	S	40	X7,6	160	S	10	X11,2	150	S	30
X3,10	0	S	0	X7,7	150	S	10	X11,3	120	0	S
X3,11	0	S	0	X7,8	120	S	0	X11,4	120	S	0
X4,1	110	S	10	X7,9	120	S	0	X11,5	160	S	30
X4,2	120	0	S	X7,10	0	S	0	X11,6	180	S	30
X4,3	170	S	50	X7,11	0	S	0	X11,7	140	S	20
X4,4	160	S	40	X8,1	150	S	50	X11,8	160	S	40
X4,5	130	S	0	X8,2	120	S	0	X11,9	120	S	0
X4,6	160	S	10	X8,3	160	S	40	X11,10	0	S	0
X4,7	140	S	0	X8,4	160	S	40	X11,11	0	S	0
X4,8	180	S	60								

I- Değişkenler (X_{ij}); **II-** Amaç katsayıları (C_{ij}); **III-** Optimum çözümün değişmesi için amaç katsayısına eklenmesi gereken en küçük sayılar (E_{ij}); **IV-** Optimum çözümün değişmesi için amaç katsayısından çıkarılması gereken en küçük sayılar (F_{ij})