ÇOKLU-AMAÇLARIN ÇÖZÜMLEMESİNDE AMAÇ PROĞRAMLAMASI İLE GENELLEŞTİRİLMİŞ TERS YAKLAŞIMI VE YEM SANAYİİNDE BİR UYGULAMA

403

Doktora Tezi

Hasan BAL

Mart 1986

Gazi Üniversitesi
Fen Bilimleri Enstitüsü

Bu tezin Doktora tezi olarak uygun olduğunu onaylarım.

Doç Dr. Fevzi Kutay

Danışman

Görevli Sınav Komisyonu

Prof.Dr.Özkan ÜNVER

Prof.Dr.Alptekin ESİN

Doç.Dr.Fevzi KUTAY

Komisyon Başkanı

Prof. Dr. Özkan Ünver

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez yazım yönetmeliğine uygundur.

GOAL PROGRAMMING AND GENERALIZED INVERSE APPROACHES IN THE MULTI-OBJECTIVE ANALYSIS AND APPLICATION IN FEED INDUSTRY

ABSTRACT

This study aimed to optimize the nutritional balance of selected feed mixes which were described in the Turkish Feed Regulation, by means of goal programming and generalized inverse approaches. Crude protein, crude cellulose, crude ash, calcium and phosphorus contents of some feedstuffs for livestocks and in addition to these nutrient metabolical energy for poultry were considered according to the data obtained from General Directorate of Feed Industry.

Goal programming is a relatively new tool that has been proposed as a model and approachfor the analysis of problems involving multiple and conflicting objectives. This method modification and extension of linear programming which can be used to optimize the nutritional balance of a selected mix of feedstuffs. It achieves this through replacement of cost minimization in the objective function by the total deviation of the goal from pre-specified levels required for optimum balance.

Following the goal programming approach to multiple goals we try to secure further perspective and insight by

approaching this same topic in a different way by using the generalized inverse of a matrix. Goal programming and generalized inverse approaches both derive solutions which minimize the "distance" from the set of goals. The distance in the goal programming approach is measured by the sum of absolute value of the deviations from each goal. But the distance in the generalized approach is measured by the square root of the sum of squares of the deviation from each goal.

In some cases the difference between distance functions in the two approaches gives us an advantage in favour of generalized inverse.

Solutions to the two approches were obtained at the Computing Center of the Faculty of Art and Sciences of Gazi University by using the programs we prepared.

Ö Z E T

Bu çalışma, amaç programlaması ve genelleştirilmiş ters yaklaşımları kullanılarak yem yönetmeliğinde belirtilen esaslara uygun besin değeri yüksek yem karışımlarını hazırlama amacını taşımaktadır. Yem Sanayii Genel Müdürlüğü verilerinden yararlanılarak büyükbaş hayvanlar için seçilen yemin ham protein, ham selüloz, ham kül, kalsiyum ve fosfor, kanatlı hayvanlar için de bu besinlere ek olarak metabolik enerji içerikleri dikkate alınmıştır.

Amaç programlaması çoklu çelişen amaçlı problemlerin çözümlemesinde bir model ve yaklaşım olarak önerilen yeni bir tekniktir. Doğrusal programlamanın genişlemesi olan bu yöntem seçilmiş bir yem karmasının beslenme dengesini eniyilemede kullanılmaktadır. Bu, maliyeti enküçüklemek yerine önceden belirlenen düzeylerden amaçların sapmaları toplamını en aza indirerek en iyi dengeyi sağlamaktadır.

Ayrıca bir matrisin genelleştirilmiş tersi kullanılarak farklı bir şekilde aynı konuya yaklaşılarak çoklu hedefler ele alınmıştır. Hem amaç proğramlaması hem de genelleştirilmiş ters yaklaşımları hedefler kümesinden "uzaklığı" enküçükleyen çözümler sağlamaktadır. Amaç programlamlaması yaklaşımında uzaklık; her hedeften sapmaların mutlak değerleri toplamı, genelleştirilmiş ters yaklaşımında ise her hedeften sapmaların kareleri toplamının karekökü ile ölçülmektedir.

İki yaklaşımın uzaklık fonksiyonları arasındaki fark bir çok durumda, genelleştirilmiş ters lehine bize bir avantaj sağlamaktadır.

Her iki yaklaşımın çözümleri Gazi Üniversitesi Fen-Edebiyat Fakültesi komputer merkezinde hazırladığımız programlar kullanılarak elde edilmiştir.

TEŞEKKÜR

Doktora Tez çalışmamın tüm aşamalarında beni yönlendiren ve yardımlarını esirgemeyen değerli hocam ve danışmanım Doç.Dr. Fevzi Kutay'a, Bilgisayar Programlarında yardımcı olan Dr. Müslüm Ekni ve Salih Çelebioğlu'na ve Uygulamada kullanılan verilerin sağlanmasında yardımcı olan, başta Ziraat Mühendisi Yavuz Koca olmak üzere Yem Sanayii Genel Müdürlüğü Yetkilileri ve Ankara Yem Sanayii fabrikası ilgililerine teşekkür ederim.

Hasan Bal

İÇİNDEKİLER

Sayfa No:

Sayfa No:

I. BÖLÜM

GİRİŞ

Yöneylem araştırması ve yönetim bilimlerinde kullanıldığı anlamda bir karar problemi" belli bir seçenek kümesinden en az bir amaç veya ölçüte" en uygun seçeneklerin belirlenmesi olarak tanımlanabilir.

Bir karar probleminde karar verici tek bir amacı göz önüne alıyorsa standart eniyileme yöntemleri (Doğrusal Programlama) en iyi kararın verilmesini mekanik bir arama şekline dönüştürebilmektedir (örneğin, kârın enbüyüklenmesi gibi). Birden fazla amacın tanımlandığı karar durumlarında ise amaç vektörünün eniyilenmesi açık bir anlam taşımamakta ve giderek yaygınlaşan bir görüşe göre gerçek bir karar problemi ancak böyle durumlarda ortaya çıkmaktadır [1].

1.1. ÇOK AMAÇLI MATEMATİKSEL PROGRAMLAMA PROBLEMİ

Çok Amaçlı Matematiksel Programlama, matematiksel programlama çatısı altında çoklu amaçları açıkça ve aynı anda gözönüne alma işlemidir.

Çok amaçlı matematiksel programlama problemi (Ç.A.M.P) matematiksel olarak şu şekilde ifade edilebilir:

$$A\bar{x} \; \{\leqq, =, \geqq\} \; \bar{b} \qquad (1.1.1)$$

kısıtlarına göre

$$z = [\, C_1(\bar{x}),\ C_2(\bar{x}), \ldots,\ C_k(\bar{x})]^T \qquad (1.1.2)$$

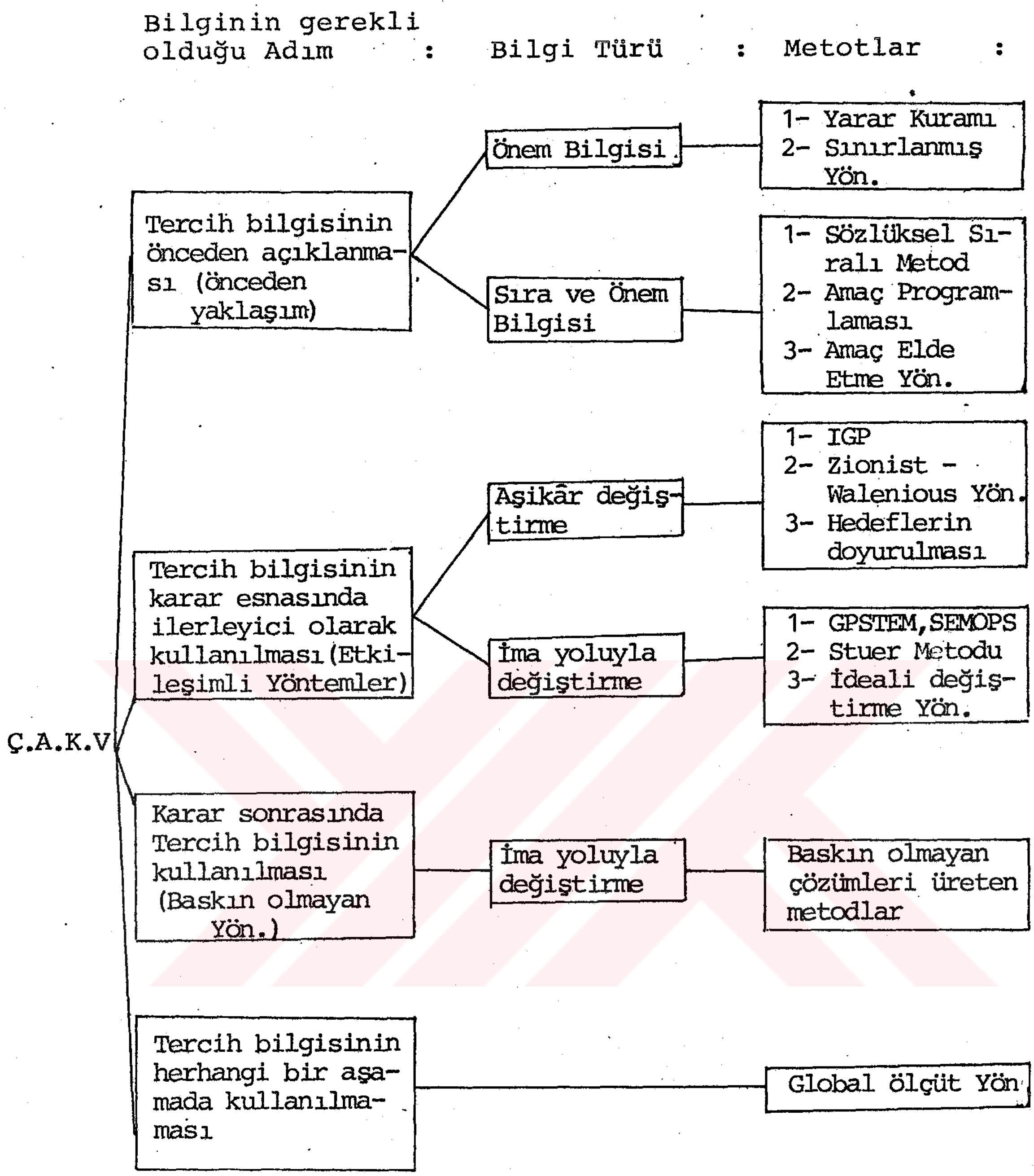
amaç vektörünü eniyileyen $\bar{x}$ çözümünü araştırmaktır. Birçok hallerde kısıtlayıcılara $\bar{x} \geqq \bar{0}$ şartı, yani değişkenlerin negatif olmama şartı da eklenir. Yukarıdaki problem amaç vektörünün enbüyüklenmesi durumunda literatürde vektör enbüyüklenmesi problemi olarak da bilinir. Yani bu problemde amaç, bütün ölçütleri aynı anda enbüyükleyen (veya enküçükleyen) $\bar{x}$ çözümünü bulmaktır. Her amaç fonksiyon için eniyi çözümün ortak olduğu aşikâr durum dışında, açıktır ki, bütün amaçları aynı anda enbüyükleyen (veya enküçükleyen) çözümü bulmak mümkün değildir.[2]

Çok amaçlı karar verme problemlerinde temel hedef, karar vericiye kendi tercih bilgisi yardımıyla çelişen amaçları uzlaştıran bir çözüm bulmada yardım etmektir[3].

Çok amaçlı (ölçütlü) karar problemlerinde bir seçeneğin değerlendirilmesinde karar vericinin seçeneğe ilişkin değer yargıları önem kazanmaktadır. Yine bu problemlerin bir diğer karakteristiği çoklu amaçların aynı birimle ölçülmemiş olmasıdır.

1.2. ÇOK AMAÇLI KARAR VERME YÖNTEMLERİ

Ç.A.M.P. probleminin çözümünde karar vericinin ve çözücünün rolü çok önemlidir. Karar verme sürecinde karar vericiden birtakım tercih bilgileri isteniyor olabilir. Hwang ve diğerleri[4]'ne göre karar verici tarafından çözücüye aktarılan tercih bilgisinin türüne göre, çok amaçlı karar vermeyi dört kategoride incelemek mümkündür.



Şekil 1.2.1. Çok Amaçlı Karar Verme Yöntemlerinin Sınıflandırılması

Bunlar Şekil 1.2.1'de gösterildiği gibi (1) Karar alıcının tercih bilgisini önceden açıklaması (önceden yaklaşım), (2) Tercih bilgisini karar esnasında kullanması (ilerleyerek yaklaşım), (3) Tercih bilgisini karar sonrası açıklaması (sonradan yaklaşım), (4) Tercih bilgisinin herhangi bir aşamada kullanılmamasıdır.

Bu yaklaşımların birtakım avantaj ve dezavantajları mevcuttur[4, 5]. Örneğin, önceden yaklaşımların en büyük dezavantajı, karar verici için, gerekli tercih bilgisini belirlemesi zorluğudur.

Karmaşık bir ortamdaki çoklu amaçlarla, özellikle aynı birimle ölçülemeyen amaçlarla, uğraşmak zorunluluğu yöneylem araştırmasında yeni bir yaklaşımın ortaya çıkmasına neden olmuştur[6]. Şu ana kadar çok amaçlı matematiksel programlama problemlerinin tümü için "eniyi" bir tek yaklaşım bulunamadığı gibi, muhtemelen gelecekte de olmıyacağı ifade edilmektedir. [7] Bununla beraber, ÇAMP'nın lokomotifi olarak adlandırılan bir yaklaşımı sözkonusudur; en fazla eleştirenleri bile bugün bu yaklaşımın ÇAMP'nın "en çalışkan" yöntemi olduğunu kabul etmişlerdir.[8, 9]

1.3. ÇOK AMAÇLI KARAR VERMEDE AMAÇ PROGRAMLAMASI

Yönetimde birçok problemler için verilen kısıtlayıcılar içinde belirlenen hedefleri kesin olarak sağlamak mümkün olmamaktadır. Bu durumda böyle hedeflerin önemlerine göre elde edilme derecesinin enbüyüklenmesi problemi sözkonusudur. Yönetim için bir hedefin diğerinden önemli olduğu pekçok durum vardır[10]. İşte Amaç Programlaması, bu tür problemlerin çözümü için en uygun, esnek ve güçlü çok amaçlı bir karar verme yöntemidir.

1.4. TARİHÇE

Amaç programlaması olarak bilinen yöntem önceleri 1950'lerde regresyon çözümlemesine bir alternatif olarak[11] ve ardından da çözülemez durumdaki doğrusal programlama problemlerinin çözümünde bir araç olarak Charnes ve Cooper[12] tarafından ortaya atılmıştır.

1965'li yıllardan sonra da çoklu çelişen amaçlardan oluşan karmaşık karar problemleri için enuygun ve güçlü bir teknik olmuştur. Ijiri[13] önemlerine göre çoklu hedefleri çözümlemede ilk defa öncelikli olma faktörünü ve aynı öncelik düzeyindeki hedeflere ilişkin sapmaların ağırlıklandırılması kavramını ortaya atmıştır.

1972 yılında Lee[14] tarafından bu alanda ilk kitabın yayınlanması amaç programlamasının popüler olmasını sağlamıştır.

Dyer[15] Frank-Wolf yönteminin uygulaması olan etkileşimli yaklaşım ile amaç programlamasını birlikte ele alarak etkileşimli amaç programlaması kavramını geliştirdi.

Ignizio[16] Birleşik Devletler uzay programında çalışırken karmaşık, büyük çaplı ve çok sayıda çelişen amaçlardan oluşan problemlerle karşılaştı. Bu problemlerin tamamının etkin olarak amaç programlaması ile çözülebildiğini gördü. Bugün yazar, bu konunun tek otoritesi olarak kabul edilmektedir. Amaç programlamasının bugünkü durumuna gelmesini en çok ona borçluyuz.

Stokastik modeller içinde amaç programlaması ilk olarak Contini[17] tarafından incelendi. Zanakis ve Maret[18] Markov modelinin ve amaç programlamasının birlikte ele alınmasıyla insangücü planlaması problemlerinde daha gerçekçi ve daha esnek sonuçlar alınabileceğini ileri sürmüşlerdir.

Amaç programlaması çok yaygın uygulama alanı potansiyeline sahip bir yöntemdir. Lin[19],1980 yılına kadar amaç programlamasının çeşitli alanlara uygulanışını gösteren önemli çalışmaları liste halinde sunmuştur.

Son zamanlarda da Anderson ve Earle[20] amaç programlamasının diyet planlaması problemlerinde doğrusal programlamadan daha üstün çözümler sağladığını göstermişlerdir.

2. BÖLÜM

AMAÇ PROGRAMLAMASI

2.1. AMAÇ PROGRAMLAMASININ TANIMI

Bir organizasyonun amaçları yönetimin tipine, karakterine ve felsefesine göre farklılık gösterir. Genellikle kârın enbüyüklenmesi başlıca amaç gibi görünür. Çünkü işletmeler ekonomik amaçlarla kurulur ve çalıştırılır. Ancak modern işletmecilikte ekonomik olmayan amaçlar da vardır ve kârın enbüyüklenmesine bu amaçlarda özen gösterilir. Bu tür amaçlar daha çok kamu kesimi ve amacı sadece kâr olmayan işletmelerde daha fazla gözlenir. Örneğin, bir termik santralın kurulmasında ve bir hava limanı yapılmasında birbiri ile çelişen farklı amaçlar sözkonusudur. Birden çok amaçlı problemler klâsik tekniklerle çözümlenemezler. Örneğin, DP tekniği uygulanmak istenirse amaç fonksiyonundan başka, diğer amaçları da yan kısıtlayıcı gibi modele eklemek gerekir. Amaç Programlaması bu nedenlerle ortaya konmuştur. DP'deki enbüyüklenme ve enküçüklenme çabalarının tersine AP'de , amaçlar arasındaki sapmalar , kısıtlamalar kümesine uygun olarak enküçüklenmeye çalışılır. AP, belli karar çerçevesinde farklı ve çelişen amaçların eniyilenmesini, araştıran matematiksel bir modeldir.

AP 'de en küçüklenmeye çalışılan sapmalar doğrusal programlamanın simpleks algoritmasında "aylâk" değişkenler olarak adlandırılırdı. Sapmalar her hedeften negatif ve pozitif sapma olarak iki yönde temsil edilir, AP'de asıl gaye atanan öncelik ve göreli önemlerine göre bu sapmaları en aza indirmektir.

Amaç programlaması, tüm amaçları hedeflere dönüştüren bir modeldir. Bu dönüştürme; her amacın sağ tarafına istenen düzeyin atanmasıyle yapılır. İstenen düzey ile çözüm düzeyi arasındaki sapmaların veya, L_1, L_2* normlarına göre ölçülmüş uzaklıkların enküçüklenmesiyle çözüme ulaşılmaya çalışılır.

Amaç programlaması doğrusal programlamanın özel bir genişlemesi olarak kabul edilebilir[21]. Yeni bir teknik olduğundan gerçek potansiyeli henüz ortaya konamamıştır. Ancak gerçek potansiyelinin en az doğrusal programlamanınki kadar olacağı tahmin edilmektedir.[14]

Bir amaç programlaması problemi (doğrusal olmasa bile) doğrusal programlama problemi olarak formüle edilebilir. Böylece, problemin çözümü için mevcut birçok doğrusal programlama tekniklerinden yararlanılabilir.[22]

Amaç Programlaması'nın tek dezavantajı, karar vericinin bilgisi olmadığında da hedefleri seçmek zorunda kalmasıdır[23]. Amaç programlamasında karar vericinin hedeflerini seçmesine ek olarak bu hedeflerin sıralanışını da göz önüne alması mümkündür. Dikkat edildiği gibi amaçlar için, birinin diğerinden öncelikli olması bilgisinin yeterli oluşu, yani amaçlara sayısal bir ağırlık vermek zorunda olmaması bu yöntemin bir avantajıdır.

---

* L_1-Normu: $\sum_{i=1}^{m} |Ax_i-b_i|$, L_2-Normu: $\sqrt{\sum_{i=1}^{m} (Ax_i-b_i)^2}$

2.2. AMAÇ PROGRAMLAMASININ FORMÜLASYONU

Daha önce de incelendiği gibi hedefler üç yapıda olabilir, bunlar

$f(\bar{x}) \geq \bar{b}$,

$f(\bar{x}) \leq \bar{b}$,

ve

$f(\bar{x}) = \bar{b}$,

dir.

Mümkün olduğu kadar bütün hedeflere eniyi şekilde ulaşmak istediğimizden, her amacın elde edilememe ölçüsüyle ilgileneceğiz. Bu, istenen düzeyden (her b_i değerinden) istenmeyen sapmalardır.

Bu sapmalar $\bar{d} = \bar{b} - f(\bar{x})$ ile ifade edilir. Böyle sapma ya pozitif, ya da negatif değerli olacağından, sapmayı

$$d_i = n_i + p_i \ , \quad i = 1,\ldots, m \qquad (2.2.1)$$

ile ifade edebiliriz. Burada

$$n_i \ p_i = 0 \qquad (2.2.2)$$

ve

$$n_i, \ p_i \geq 0 \ , \ i = 1,\ldots,m \qquad (2.2.3)$$

şartlarını sağlamalıdır.

Tipik olarak n_i negatif sapma ve p_i de pozitif sapma olarak bilinir. Dikkat edilirse çözüm düzeyi ile istenen düzey arasındaki sapmaların mutlak değerleri ile ilgileniyoruz, bir diğer anlamda hedeften sapmaların mutlak değerleri enküçüklenmeye çalışılacaktır. Çeşitli yapıdaki hedefler ile sapma değişkenleri arasındaki ilişki aşağıdaki tabloda görülmektedir.

Tablo 2.1. Hedefler ve Hedef Sapmaları

Hedefin Başlangıç Şekli	Eşitlik Haline Dönüştürülmüş Şekli	Minimum Yapılacak Sapma Değişkenler
$f_i(\bar{x}) \geqq b_i$	$f_i(\bar{x}) + n_i - p_i = b_i$	n_i
$f_i(\bar{x}) \leqq b_i$	$f_i(\bar{x}) + n_i - p_i = b_i$	p_i
$f_i(\bar{x}) = b_i$	$f_i(\bar{x}) + n_i - p_i = b_i$	$n_i + p_i$

2.3. TEMEL TANIMLAR [7,16]

Tanım 2.3.1. Amaçlar. Karar değişkenleri veya kontrol edilebilen değişkenlerin matematiksel fonksiyonlarıyla temsil edilir. Böyle fonksiyonlar çoğunlukla karar alıcının arzu veya isteklerini simgeler.

Bir amaç fonksiyonunun sağ taraf sabiti kesin olarak belirlenemez. En yaygın amaç fonksiyonlar,

Minimum $f(\bar{x})$ veya Maksimum $f(\bar{x})$

biçimindedir.

Tanım 2.3.2. İstenen düzey. Bir amaca ulaşmada kabul edilebilir özel bir değerdir. Böylece, istenen düzey bir amaca ulaşıp ulaşmadığımızı ölçmek için kullanılabilir.

Tanım 2.3.3. Hedef. İstenen düzeyle birlikte bir amaç, hedef olarak adlandırılır. Örneğin, biz kârı maksimum yapmak istediğimizde kâr bir amaçtır, ancak bunun yerine kâr düzeyinin en az 1000 TL'ye ulaşmasını arzularsak, bir hedef kurmuş oluruz.

Tanım 2.3.4. Başarma Vektörü. AP ' de başarma vektörü a, öncelik sayısına göre sıralanmış K-boyutlu bir vektördür.

Tanım 2.3.5. Amaç Programı. J karar değişkeni ve 2m sapma değişkenli m-sayıda doğrusal fonksiyondan oluşmaktadır.

Tanım 2.3.6. Temel Çözüm. (J + 2m) - m sayıda değişken sıfır alınarak m-sayıdaki hedef çözülür. Bu çözüm temel çözüm olarak adlandırılır.

Tanım 2.3.7. Eniyi Çözüm. AP'da $\bar{x}^*$ çözümüne karşı gelen $\bar{a}^*$ vektörü, diğer mümkün $\bar{x}$ çözümüne karşı gelen $\bar{a}$ vektörüne tercih ediliyorsa, $\bar{x}^*$ çözümü en iyi çözümdür. Burada $\bar{a}^*$ ve $\bar{a}$'nın bütün elemanlarının kendileri pozitif olmak üzere, $\bar{a}^* - \bar{a}$ vektörünün sıfırdan farklı ilk bileşeni negatif ise $\bar{a}^*$, $\bar{a}$ vektörüne tercih edilmektedir.

2.4. A.P.'DA BİR ÇÖZÜMÜN DEĞERLENDİRİLMESİ

Başarı fonksiyonu n_i ve p_i sapma değişkenleri i. hedefe ulaşılamamanın ölçüsünü gösterir. Çok amaçlı modelde en iyi çözümü belirlemek için çözümleri karşılaştırabilme gereği de duyarız. Herhangibir aşikâr olmayan çok amaçlı modelde bir çözümün iyiliğinin ölçüsü kişinin tamamıyla kendi isteğine bağlıdır. Ancak bir çözümün iyiliğini ölçmede aşağıdaki kriterlere de başvurulabilir[7]:

(1) Ağırlıklı hedef sapmalarının toplamını enküçükleleyebilmesi, bu görüş Charnes ve Cooper tarafından"önerilen ağırlıklı amaç programlaması yöntemine dayanmaktadır.

(2) Hedef sapma değişkenlerinin birtakım doğrusal olmayan şekillerinin enküçüklenmesi, burada hedef sapmalarının ağırlıklandırılması yerine, önem derecelerine göre üsleri ağırlıklandırılır.

(3) Hedef sapmalarının enbüyük olanının enküçüklenmesini öngören görüş.

(4) Hedef sapmaları ve ağırlıklı hedef sapmalarına ilişkin sıralı kümenin sözlüksel sıra minimumunun elde edilebilmesi. Bu düşünce öncelik düzeyli olarak bilinen amaç programlamasının gelişmesine öncülük etmiştir.

(5) Yukarıdaki yaklaşımlardan herhangi birisinin aracılığıyla bulunan başlangıç çözümün komşuluğundaki baskın olmayan (etkin) çözümler kümesini elde etme. Bu görüş amaç programlaması aracılığıyla etkin çözümlerin alt kümesini belirlemeye öncülük etmiştir.[24]

Yukarıdaki yaklaşımlar başarı vektörü aracılığıyla simgelenebilir ve ölçülebilir. Örneğin, ağırlıklı amaç programlaması aşağıdaki şekilde ifade edilebilir:

$$f_i(\bar{x}) \ \left(\begin{matrix} \leq \\ = \\ \geq \end{matrix} \right) \ b_i \ , \quad i \in F \qquad (2.4.1)$$

$$C_i(\bar{x}) + n_i - p_i = b_i \quad i \in C \qquad (2.4.2)$$

kısıtlarına göre

$$\bar{a} = \sum_{i \varepsilon C} (u_i \, n_i + w_i \, p_i) \qquad (2.4.3)$$

eşitliğini enküçükleyen $\underline{x}^*$ çözümünün bulunmasıdır.

Burada $\bar{a}$: verilen programın başarılma (gerçekte başarılamama) ölçüsüdür. u_i: i-inci hedefin negatif sapmasına verilen ağırlıktır. w_i: i-inci hedefin pozitif sapmasına verilen ağırlıktır. F: mutlak (fiziksel) kısıtlardır. C: hedefler kümesidir (yani istenen düzey ile ele alınan amaçlar kümesi).

2.5. AMAÇ PROGRAMLAMASININ TÜRLERİ

Doğrusal Programlamanın çeşitleri gibi amaç programlamasının da birçok çeşitleri vardır [25]. Bunlar

1) Doğrusal Amaç Programlaması (D.A.P),
2) Doğrusal olmayan Amaç Programlaması (D.O.A.P),
3) Tam Sayılı A.P (T.S.A.P),
4) 0-1 amaç programlaması,
5) Stokastik amaç programlaması.

Tezimde yukardaki amaç programlaması türlerinden sadece doğrusal amaç programlaması üzerinde durulacaktır.

2.6. DOĞRUSAL AMAÇ PROGRAMLAMASI (D.A.P).

Doğrusal amaç programlamasının formülasyonu aşağıdaki şekilde verilebilir.

$$\sum_{j=1}^{j} C_{ij} x_i + n_i - p_i = b_i, \quad i=1,2,\ldots,m \qquad (2.6.1)$$

$$\bar{x},\ \bar{n},\ \bar{p}\ \geq 0 \qquad (2.6.2)$$

doğrusal kısıtları altında

$$\bar{a} = \{\rho_1(\bar{n},\bar{p}),\ \rho_2(\bar{n},\bar{p}),\ldots,\ \rho_k(\bar{n},\bar{p})\} \qquad (2.6.3)$$

minimum olacak şekilde $\bar{x} = \{x_1, x_2, \ldots, x_j\}$ noktasının bulunmasıdır. Burada

x_j : karar (veya kontrol) değişkeni,

$\bar{a}$: başarma vektörü

C_{ij} : teknolojik katsayılar,

b_i : i-inci mutlak kısıt veya hedefe ilişkin sağ taraf sabiti,

n_i : i-inci kısıt veya hedefle ilgili negatif sapma,

p_i : i-inci kısıt veya hedefle ilgili pozitif sapma,

$\rho_k(\bar{n},\bar{p})$: k-inci öncelik düzeyiyle ilgili sapma değişkenlerinin bir doğrusal fonksiyonudur.

2.7. DOĞRUSAL AMAÇ PROGRAMLAMA MODELİNİN ÇÖZÜMÜ

Amaç Programlamasının amacı, verilen hedefler kümesine mümkün olduğu kadar ulaşabilmektedir. Çözüme, verilen karar ortamı içinde ve karar vericinin öncelik yapısına göre ulaşılmalıdır. AP'da iki temel çözüm yöntemi vardır. Bunlar

1- Grafik Yöntemi,

2- Simpleks Yöntemi.

2.8. GRAFİK YÖNTEMİYLE ÇÖZÜM

Grafiksel çözüm üç ve daha fazla değişkenden oluşan problemler için uygun olmamasına rağmen, büyük çaplı problemler için kullanılan yöntemin anlaşılmasına yardımcı olur.

Grafiksel yaklaşım aşağıdaki adımlardan oluşur:

(1) Karar değişkenleri aracılığıyla tüm hedefleri (amaçları) koordinat düzleminde gösteririz.

(2) En öncelikli amaçların çözümlerini belirleriz.

(3) İkinci dereceden öncelikli amacın (veya amaçlar kümesinin) en iyi çözümü belirlenir. Bu adımda belirlenen en iyi çözüm kendisinden daha öncelikli amaçlar için ulaşılan düzeyi bozmamalıdır.

(4) (3) ncü adım bütün öncelik düzeyleri için tekrarlanır.

(5) Amaçların hepsi karar alıcı için aynı önemde ise, amaçlar ile istenen düzey, $\bar{b}_i$, vektörü arasındaki en yakın uzaklığı veren çözüm araştırılır.

Örnek 2.7.1.

$10x_1 + 15x_2 \leqq 40$ (1. öncelikli)

$100x_1 + 140x_2 \geqq 1400$ (2. öncelikli)

$x_2 \geqq 8$ (3. öncelikli)

Yukarıdaki amaçlara öncelik yapılarına göre mümkün olduğu kadar en iyi şekilde ulaşmaya çalışacağız, sisteme sapma değişkenleri eklenirse sistem

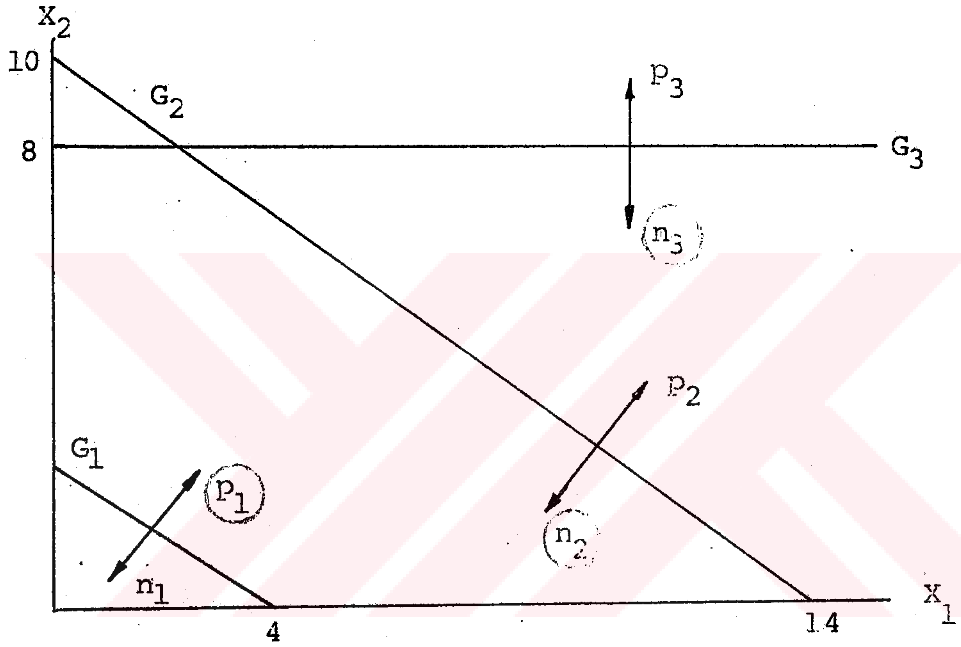
$$G_1 : 10x_1 + 15x_2 + n_1 - p_1 = 40$$

$$G_2 : 100x_1 + 140x_2 + n_2 - p_2 = 1400$$

$$G_3 : \qquad x_2 + n_3 - p_3 = 8$$

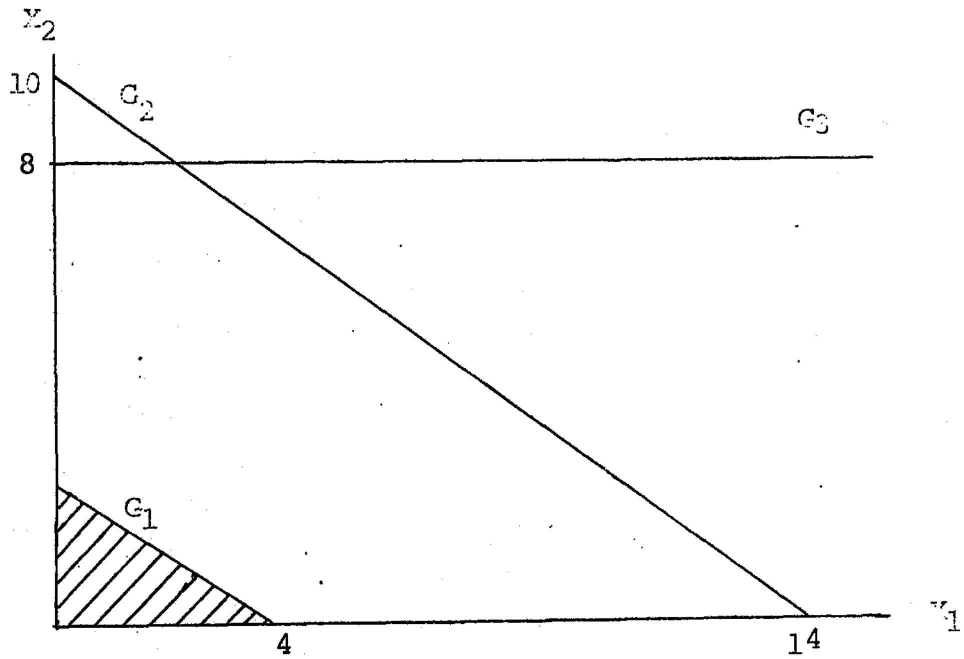
biçimine dönüşecektir. Bu aşamadan sonra

$\bar{a} = \{(p_1), (n_2), (n_3)\}$'i, minimum yapan (x_1, x_2) çözümü araştırılacaktır. Burada $\bar{x}$, $\bar{p}$, $\bar{n} \geqslant 0$ olacaktır.



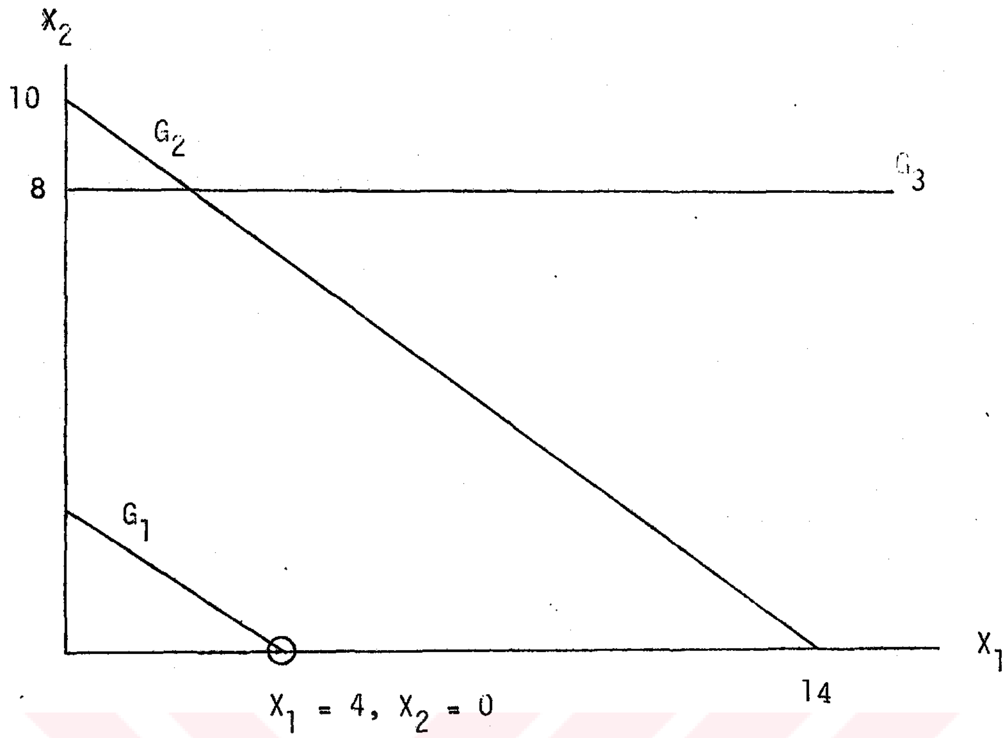
Grafik I. Örnek 2.7.1'in Grafiksel Gösterimi.

Grafik I'de de görüldüğü gibi kısıtlayıcılar sapma değişkenleri vasıtasıyla hedeflere dönüştürüldü. Şekilde daire içerisine alınan sapma değişkenler öncelik yapılarına göre minimum yapılacaktır. Bu düşünce altında birinci öncelikli p_1 sapmasını minimum yapan çözümler araştırılacaktır.



Grafik II. 1'inci Öncelik Düzeyinde Çözümler

Birinci öncelik düzeyinde p_1 sapmasını minimum yapacak en iyi çözüm ya $\bar{x}_1^* = (4,0)$ noktası, veya $\bar{x}_2^* = (0, \frac{40}{15})$ noktasıdır. Bundan sonraki aşamada bu iki noktadan ikinci öncelik düzeyindeki n_2 sapmasını minimum yapan çözüm araştırılacaktır. $\bar{x}_1^* = (4,0)$ noktası için p_1 sapması $p_1 = 1000$ olacaktır, $\bar{x}_2^* = (0, \frac{40}{15})$ noktası için ise $n_2 = 1026.7$ olacaktır. Öyleyse minimum sapma 1000 olacağından birinci ve ikinci öncelik düzeyi için en iyi çözüm $\bar{x}_1^* = (4,0)$ noktası olacaktır. Bu aşamadan sonra üçüncü öncelik düzeyi gözönüne alındığında $\bar{x}_2^* = (0, \frac{40}{15})$ noktası n_3 sapmasını minimum yapmasına karşılık, bu çözüm ikinci öncelik düzeyinin erişilen minimum sapmasını bozmuş olacağından, bu öncelik düzeyi içinde $\bar{x}_1^* = (4,0)$ noktası en iyi çözüm olacağından, her üç öncelik düzeyi gözönüne alınırsa problemin en iyi çözümü, $\bar{a}^* = (0, 1000, 8)$ vektörünü öncelik sırasına göre enküçük yapan çözüm, $\bar{x}_1^* = (4,0)$ noktasıdır. Bu nokta Grafik III'de gösterilmiştir.



Grafik III. Birinci, İkinci ve Üçüncü Öncelik Düzeyleri için En İyi Çözüm

Bu çözümle birinci amacın istenen düzeyine tamamen ulaşıldığı halde ikinci ve üçüncü öncelikli amaçların istenen düzeylerine tamamen ulaşılamamıştır.

2.9. SIRALI DOĞRUSAL AMAÇ PROGRAMLAMASI

Hedefler önemlerine göre sıralandırılmışsa modeli aşağıdaki şekilde verebiliriz:

$$f_i(\bar{x}) + n_1 - p_i = b_i$$

$$\bar{x},\ \bar{p},\ \bar{n} \geqq 0$$

kısıtlarına göre

$$\bar{a} = \{a_1,\ a_2, \ldots\ a_k\}$$

nın sıralı minimumunu belirleyen $\bar{x}$ çözümünün bulunması modeline sözlüksel sıralı doğrusal amaç programlaması modeli diyeceğiz[26].

Sözlüksel sıralı D.A.P.'da amaçlar en önemliden en az önemliye doğru bir sıraya dizilir. Tüm seçenekler arasında birinci amacı eniyileyen seçenekler dışında kalanlar elenir, geriye kalanlardan ikinci amaca göre en iyi seçenekler dışındakiler elenir ve bu eleme tek bir seçenek kalıncaya kadar devam edilir.

Sözlüksel sıralı (veya, kısaca sıralı) doğrusal amaç programlaması literatürde öncelikli yapıya sahip amaç programlaması olarak da yaygın bir şekilde kullanılmaktadır.

Öncelikli olma kavramı doğrusal programlamada da söz konusudur. Birinci öncelik tüm kısıtların sağlanmasına ayrılmışken, ikinci öncelik ise bu kısıtlar sağlandıktan sonra amaç fonksiyonunun düşünülmesi, göz önüne alınmasıdır[27].

Çözüm yöntemlerine geçmeden önce biraz sapma değişkenlerinin ağırlıklandırılmasından söz edelim.

2.10. SAPMALARIN AĞIRLIKLANDIRILMASI

Burada göz önüne alınacak ölçüt, bir sapma değişkenindeki ne kadarlık bir artma miktarı, kendisiyle aynı öncelikte diğer bir değişken sapmasının bir birim azalmasıyle dengelenebileceğidir[13]. Örneğin, satış hedefi aynı önem dereceli iki farklı üründen oluşmuşsa, aynı öncelik düzeyinde iki sapma değişkeni sözkonusudur. Bununla birlikte bu iki ürünün kâr katkı oranı birbirinden farklı olabilir, işte bu farklılıktan dolayı kârı yüksek olan ürüne daha fazla ağırlık verilmelidir. Bu nedenle sapma değişkenlerinin farklı ağırlıklarını belirlerken kullanılacak ölçüt pişmanlık veya fırsat maliyetinin minimum edilmesidir.[14]

2.11. VARSAYIMLAR

Amaç programlaması doğrusal programlamanın bir genişlemesi olduğundan, amaç fonksiyonu, kısıtlayıcılar ve hedef ilişkileri doğrusaldır.

Karar değişkenlerinin bölünebilir olması, yani kesirli karar değişkenleri uygun çözüm olarak kabul edilebilmesidir.

Deterministiktir, modelin katsayıları (a_{ij}, b_i, P_j) sabitler olmalıdır.

2.12. AMAÇ PROGRAMLAMASININ ALGORİTMA İLE ÇÖZÜMÜ

Bugüne kadar birçok amaç programlaması algoritmaları geliştirilmiştir. AP'yi kullanmanın en büyük sınırlaması şimdiye kadar etkin bir algoritmaya sahip olamayışıdır[28]. Bilindiği gibi ilk AP problemi Charnes ve Cooper[12] tarafından verilmiştir. Bunlar problemi, tek amacın öncelik yapısı göz önüne alınarak mevcut lineer programlama yönteminden yararlanarak çözmüşlerdir. Lee[14], Ignizio[16] amaçlar kümesinin öncelik yapılarını gözönüne alarak simpleks yöntemine dayanan algoritmalar geliştirdiler. Ancak bu algoritmaların küçük çaplı problemler için bile hesaplamaları çok zaman alıcı olmaktadır. Bu algoritmalar literatürde Geliştirilmiş Simpleks Yöntemi veya Çok Aşamalı Simpleks Yöntemi olarak da kullanılmaktadır. Dauer ve Krueger[29] genel amaç programlaması problemleri için ardışık bir algoritma geliştirdiler. Ignizio ve Perlis[30] bu algoritmaya benzer, ardışık doğrusal amaç programlaması problemlerini çözmek için bir diğer algoritma geliştirdiler. Bu algoritma öncelik yapılarına göre alt problemlerinin çözümüne doğrusal programlama yöntemleri kullanarak ulaşmaya çalışır. Arthur ve Ravindran[31,32] ise bu yaklaşımı öncelik yapılarına göre alt problemlerin çözümünde de amaç programlaması yöntemini kullanan etkin bir algoritma geliştirdiler.

Son zamanlarda Schneiderjans ve Kwak[33] genel amaç programlaması modellerini çözmede yeni bir yaklaşım önermişlerdir. Bu algoritma Lemke tarafından geliştirilen Dual Simpleks Yöntemi'nden yararlanan bir algoritmadır. Bu yöntem, çözümünde pozitif sapma değişkenlerinin fazla olduğu modeller için hesaplanma zamanını kısaltmada çok etkindir.

Son olarak da Ignizio[11], amaç programlamasının duali aracılığıyla diğer bir algoritma geliştirmiştir.

Bu algoritmalar arasında gerek Schneiderjans ve Kwak, gerek Arthur ve Ravindran'ın geliştirdiği algoritmalar daha etkin olmasına karşın, duyarlık analizi yapabilmeye Çok Aşamalı Simpleks Algoritması kadar uygun değildir. Yine Ignizio ve Perlis'in geliştirdikleri algoritma, çözüm için doğrusal programlama yöntemlerinden yararlanmasına rağmen, dualite ve duyarlık analizinin uygulanışında Çok Aşamalı Simpleks Algoritması kadar uygun değildir.

Bu nedenle biz çözüm yöntemi olarak Ignizio'nun geliştirdiği Çok Aşamalı Simpleks Yöntemi'ni kullanacağız. Bu yöntemle elde edilen Çözüm ile Ignizio ve Perlis'in geliştirdikleri algoritma vasıtasıyla elde edilen çözüm aynıdır. Büyük çaplı problemler için Ignizio ve Perlis'in hazırladıkları algoritma, mevcut doğrusal programlama bilgisayar paket programları aracılığıyla çözüme ulaşabilmektedir, ve bilgisayar zamanını daha az kullanma imkânı sağlamaktadır. Bunun yanında Çok Aşamalı modelin formülasyonu statiktir, önceden belirlenmiştir. Halbuki bireysel doğrusal programlamadan oluşan ardışık doğrusal amaç programlaması (Perlis ve Ignizio SLGP modeli) modeli önceden belirlenemez.[30]

2.13. ÇOK AŞAMALI SİMPLEKSİN TEORİK TEMELLERİ[34]

Çok Aşamalı Simpleks, İki Aşamalı Simpleks Yönteminin açıkça bir genişlemesidir. Bu yöntemle çözülebilecek Sıralı amaç programlaması (S.A.P) aşağıdaki şekilde verilebilir.

$$\sum_{j=1}^{n} c_{ij}\, x_j + n_i - p_i = b_i, \quad i = 1,2,\ldots,m, \; j=1,2,\ldots,n \tag{2.13.1}$$

$$\bar{x},\ \bar{p},\ \bar{n} \geq 0 \tag{2.13.2}$$

kısıtlayıcılarına göre

$$\bar{a} = (a^1,\ a^2,\ldots\ a^k,\ldots\ a^k) \tag{2.13.3}$$

eşitliğini sıralı minimum yapan $\bar{x}$ çözümlerinin bulunmasıydı.

Başarı Vektörü, $\bar{a}$ şu şekilde ifade edilebilir:

$$\bar{a} = \{ \sum_{i=1}^{m} (w_i\ p_i+u_i n_i),\ldots,\ \Sigma(w_i\ p_i+u_i n_i)\}. \tag{2.13.4}$$

Matris gösterimiyle (S.A.P) modelinin yazılımı

$$[C,\ I,\ -I] \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} = \bar{b} \qquad (2.13.4)$$

$$\bar{x},\ \bar{n},\ \bar{p} \geq 0 \qquad (2.13.5)$$

kısıtlayıcılarına göre

$$\bar{a} = [u\ \ w] \begin{pmatrix} \bar{n} \\ \bar{p} \end{pmatrix} \qquad (2.13.6)$$

eşitliğini sıralı minimum yapan $\bar{x}$ çözümünün bulunmasıdır. Burada I, m x m-boyutlu birim matrisi, U ve W ise k x m-boyutlu matrislerdir.

Bu yapıdaki $\bar{a}$, başarı vektörü, temel değişkenlerin başlangıç kümesi olan negatif sapma değişkenleri içerecektir. Bu yüzden $\bar{a}$, hem temel, hem de temelde olmayan değişkenlerin fonksiyonu olacaktır. Biz $\bar{a}$'nin sadece temelde olmayan değişkenlerle ifade edilmiş olmasını istiyoruz. (2.13.4)'den n'yi çözer, (2.13.6)'da yerine koyarsak $\bar{a}$ vektörü şöyle olacaktır:

$$\bar{a} = [-uc \quad 0 \quad u + w] \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} + u\bar{b}. \qquad (2.13.7)$$

Bu hatırlatmalardan sonra Çok Aşamalı Simpleks Yöntemi'nin teorik temellerine geçelim. m x m-boyutlu B temel matrisi [C, I, -I]'ya ilişkin m-doğrusal bağımsız sütunlardan oluşmuş olsun [C, I, -I]'nın herhangi bir temelde olmayan ϕ vektörü, B'nin sütunlarının doğrusal bir bileşimi olarak yazılabilir, yani $\phi_q = B\Phi$, q = 1,...,n + m

B bir temeldir, tekil olmadığına göre $\Phi = \bar{B}^{1} \phi_q$ olur.

Şimdi hedeflerin çok aşamalı primal kümesini gözönüne alalım:

$$[C \quad I \quad -I] \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} = [B, N] \; \bar{X}' = [B \quad N] \begin{pmatrix} x_B \\ x_N \end{pmatrix} = \bar{b}$$

Burada $\bar{x}' = \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix}$ ' dır.

N, m x n + m-boyutlu temelde olmayan değişkenlerin katsayılar matrisi; x_B; temel değişkenler vektörü; x_N: m + n temelde olmayan değişkenler vektörü, temel değişken tanımıyle, $x_N = 0$, ve böylece $Bx_B = \bar{b}$ olur.

Bundan başka B tekil olmadığından, tersi vardır ve $x_B = B^{-1}\bar{b}$ dir. Yukarıdakine benzer şekilde $\bar{a}$ vektörü de

$$\underset{\cdot\cdot}{a} = [w_B \quad w_N] \begin{pmatrix} x_B \\ x_N \end{pmatrix} = [w_B \; x_B]$$

şeklinde ayrıştırılabilir. Sonuç olarak her Φ vektörü ile ilgili $z_q = w_B \Phi_q$ vektörü tanımlanabilir.

Ardında SAP ' de ayrılacak değişken veya mümkünlük şartı incelenecektir. Kural, verilen Φ_{iq} vektörü için x_{B_i} ile Φ_{iq}'nin en küçük pozitif oranının seçimidir. Aşağıdaki türetme bu özel seçimi doğrulayacaktır. $B\Phi_q = \phi_q$ olduğu ifade edilmişti. λ herhangi bir gerçel sayı olsun, $\lambda B\Phi_q = \lambda\phi_q$'dır. Ek olarak, $Bx_B = \bar{b}$ olduğundan $B\,x_B - \lambda\Phi_q + \lambda\Phi_q = \bar{b}$ olur. Bu işlemi B^{-1} ile çarpar gerekli düzenlemeler yapılırsa, yeni m + 1 - boyutlu vektör:

$$\begin{bmatrix} x_B - \lambda\Phi_q \\ \lambda \end{bmatrix} = \begin{bmatrix} B^{-1}\,\bar{b} - B^{-1}\,\lambda\phi_q \\ \lambda \end{bmatrix} = \begin{bmatrix} B^{-1}\bar{b} - \lambda\Phi_q \\ \lambda \end{bmatrix}$$

Burada $\lambda = x'_q$, S.A.P'de temelde olmayan, çözüm olarak elde edilen giren değişkendir. Mümkün bir temel çözüm elde etmek için, eski temel değişkenlerin birisi sıfır olmalıdır. Bununla birlikte mümkünlüğün değişmemesi için $x'_q = \lambda \geqq 0$ ve $x_B - \lambda\Phi_q \geqq 0$ olmalıdır. Öyleyse $\Phi_q > 0$ olan durumları göz önüne almak gereklidir. Yani $\Phi_{iq} > 0$ olmak üzere

$$x_{B_i} - \lambda\Phi_{iq} \geqq 0, \quad i = 1, \ldots m$$

veya

$$\lambda \leqq x_{B_i}/\Phi_{iq},$$

olmalıdır. Bunun için $\lambda = \min_i \{x_{B_i}/\Phi_{iq},\ \Phi_{iq} > 0\} = B^{-1}b_i/\Phi_{i_q} = x_{B_i}/\Phi_{iq}$ olarak seçilmesi uygundur. Seçilen bir q için, $\Phi_{i'q} > 0$ ve $i = i'$ bize minimum λ'yı verir.

ϕ' ve θ' sırayla başlangıç genişletilen tablo ve gösterge satırlara karşı getirilir. Burada $q=1,\ldots,n+2m'$-dir. O zaman, öncekine benzer biçimde $\phi'_q = B^{-1}\,\phi'_q$ ve matris şekliyle:

$$\Phi' = B^{-1}\,[C \quad I \quad -I] = [B^{-1}C \quad B^{-1} \quad -B^{-1}]$$

$z'_q = w_B\,\Phi'_q = w_B\,B^{-1}\,\phi'_q$ olarak tanımlanması

$$z' = w_B B^{-1}[C \quad I \quad -I] = [w_B\,B^{-1}C \quad w_B\,B^{-1} \quad -w_B\,B^{-1}]$$

olmasını gerektirir. Böylece

$$\theta' = \{\theta'_q\} = \{z'_q - w'_q\}$$

$$= [\, w_B\ B^{-1}C \quad w_B\ B^{-1} \qquad -w_B\ B^{-1}] - [\, o \quad u \quad w]$$

$$= [\, w_B\ B^{-1}C \quad w_B\ B^{-1} - u \quad -w_B\ B^{-1} - w\,]$$

SAP tablosu ve algoritmasıyla bu sonuçların bağlantısını kurmak için herhangi bir iterasyonda ilk (primal) şekli teşkil eden denklemleri gözönüne alalım: $Bx_B = \bar{b}$ ve $w_B\ x_B = \bar{a} \Rightarrow \bar{a} - w_B\ x_B = \bar{0}$. Yukarıdaki denklemleri birleştirirsek

$$\begin{bmatrix} B & O \\ -w_B & I \end{bmatrix} \begin{bmatrix} x_B \\ \bar{a} \end{bmatrix} = \begin{bmatrix} \bar{b} \\ \bar{0} \end{bmatrix}$$

eşitliğini elde ederiz.

Eniyileme şartı ve giren değişkenin seçimi:

$z_q = w_B\ \Phi_q$ olarak tanımlamıştık, burada $\Phi_q = B^{-1}\phi q$ dır. Eğer x_q temele girecekse çözüm

$$\begin{bmatrix} x_B - \lambda\Phi_q \\ \lambda \end{bmatrix}$$

dır. Burada λ'nın seçimi eski temel değişkenin sıfır olmasına, temelden ayrılmasına neden olur. $\bar{a} = w_B\ x_B$ önceki başarı vektörü olsun ve $\bar{\bar{a}}$, $\bar{x}_q$ temele girdikten sonra yeni başarı vektörüdür. O zaman

$$\bar{\bar{a}} = [\, w_B \quad w_q\,] \begin{pmatrix} x_B - \lambda\Phi_q \\ \lambda \end{pmatrix}$$

$$= w_B\ x_B - \lambda(w_B\ \Phi_q - w_q)$$

$$= \bar{a} - \lambda(z_q - w_q)$$

veya

$$\hat{\underline{a}} - \underline{a} = -\lambda(z_q - w_q).$$

Burada w_q, q-inci temelde olmayan değişkenin ağırlık vektörüdür. $\lambda \geqq 0$ olduğundan $z_q - w_q \geqq 0$ ise $\hat{\bar{a}}$ sıralı olarak $\bar{a}$ den daha azdır. Bu şu demektir; sıralı minimum probleminde, çözümün değeri artmıştır (düzelmiştir) anlamındadır. $z_q - W_q$'-nun maksimumu seçilirse, çoğunlukla başarı vektörünün değeri en fazla değişir. $z_q = W_q$ ise hiçbir değişiklik olmayacak, $z_q - W_q \leqq 0$ ise $\hat{\bar{a}} \geqq \bar{a}$ ve x_q temele girmişse çözüm kötüdür. Her q için $z_q - w_q \leqq 0$ ise çözüm eniyidir.

Bu sonucu tablo ve çözüm algoritmasıyla ilişkilendirmek için, problemin çözümü $(x_B, \bar{a})^T$'dir ve

$$\begin{bmatrix} B & 0 \\ -w_B & I \end{bmatrix}$$

nın tersi ile çarpılarak elde edilir. Bunun için

$$\begin{bmatrix} x_B \\ a \end{bmatrix} = \begin{bmatrix} B^{-1} & 0 \\ w_B B^{-1} & I \end{bmatrix} \begin{bmatrix} b \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} B^{-1} & 0 \\ w_B B^{-1} & I \end{bmatrix} \begin{bmatrix} C & I & -I \\ 0 & -u & -w \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{bmatrix}$$

$$= \begin{bmatrix} B^{-1} C & B^{-1} & -B^{-1} \\ w_B B^{-1} C & w_B B^{-1} - u & -w_B B^{-1} - w \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{bmatrix}$$

bulunur ki, bu da tabloya eşdeğerdir.

$\bar{B}^1$ biliniyorsa, herhangi bir iterasyon için bütün elemanları elde etmek mümkündür. Başlangıçta $w_B = U$ ve $B = I$ dır, böylece yukarıdakiler

$$\begin{bmatrix} \bar{b} \\ \\ u\bar{b} \end{bmatrix} = \begin{bmatrix} c & I & -I \\ \\ uc & 0 & -u - w \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{bmatrix}$$

biçimine indirgenir, bu başlangıç çok aşamalı ilk tabloya karşı gelir.

Mümkünlük ve Eniyileme Sorununa gelince; tablonun indeks satırları eniyilemeyi gösterir.

Her vektör sıralı olarak negatif ise çözüm eniyidir. Eniyi olmayan bir çözümü düzeltmek için: $\theta_q^k = z_q^k - w_q^k$'nın maksimum değeri θ_q^1, $\theta_q^2, \ldots,$ $\theta_q^k \geqq 0$ elemanlardan temele girecek değişkeni gösterir, çünkü bu işlem çoğunlukla a_k'yı en fazla miktarda değiştirir. Evvelki önceliklerin düşünülmesi, bütün θ vektörleri sıralı (lexicographical) olarak negatif olmadıkça optimalitenin elde edilememesi yüzünden, önemlidir. Sonuç olarak primal eniyileme $w_B\ \bar{B}^1\ C \leqq 0$, $w_B\ \bar{B}^1 - u \leqq 0$ ve $-w_B\ \bar{B}^1 - w \leqq 0$ ile gösterilir. Mümkünlük $x_B \geqq 0$ veya $\bar{x}$, $\bar{n}$, $\bar{p} \geqq 0$ ile gösterilir. İlk formülasyon çoğunlukla mümkünlüğü gösterir ve araştırma eniyileme içindir.

2.14. ÇOK AŞAMALI SİMPLEKS TABLOSUNUN KURULUŞU

Problemleri elle hızlı çözmenin anahtarı Çok Aşamalı Simpleks tablosunun uygun olarak kuruluşunda yatmaktadır. Yöntemin genel tablosu Tablo 2-14.1'de gösterilmiştir.

Bu tablonun elemanları aşağıdaki gibi tanımlanabilir.

BAŞLIKLAR:

P_k : k-inci öncelik düzeyi, $k = 1,2,\ldots K$

V : problem değişkenleri (karar ve sapma değişkenleri). V'nin sağındaki değişkenler (x_j ve p_i) temelde olmayan değişkenlerin başlangıç kümesidir. V'nin altındaki değişkenler (n_i) temel değişkenlerin başlangıç kümesidir.

$\bar{b}$: $\bar{b}$'nin altındaki elemanlar her amacın sağ taraf sabit değerler yani b_i'lerdir.

ELEMANLAR:

$j = 1,2,\ldots,J$; $i = 1,2,\ldots,m$; $s = 1,2,\ldots,S$;
$k = 1,2,\ldots,K$

e_{is} = s-inci temel dışı değişken altındaki i-inci satır elemanıdır, yani e_{is}, i-inci amaçta s-inci temel dışı değişkenin katsayısıdır.

W_{ks} : s-inci temel dışı değişken ile ilgili k-inci önceliğin ağırlık faktörü.

U_{ik} : i-inci temel değişkenle ilgili K önceliğinin ağırlıklandırma faktörü.

$I_{k,s}$: s-inci temel dışı değişken altındaki k önceliğin indeks numarası.

a_k : k öncelikli başarı seviyesi, burada $\bar{a} = \{a_1, a_2,\ldots, a_k\}$ biçimindedir.

Tablo 2.14.1

Başlangıç Çok Aşamalı Simpleks Tablosu

	P_k	$w_{k,1} \dots w_{k,j}$	$w_{k,j+1} \dots w_{k,j+m}$		
SOL KISIM	$\vdots$	$\vdots$	$\vdots$		ÜST KISIM
	P_1	$w_{1,1} \dots w_{1,j}$	$w_{1,j+1} \dots w_{1,j+m}$		
$P_k \dots P_1$	V	$x_1 \dots x_j$	$P_1 \dots P_m$	$\bar{b}$	
$U_{1,k} \dots U_{1,1}$	n_1	$e_{1,1} \dots e_{1,j}$	$e_{1,j+1} \dots e_{1,j+m}$	b_1	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$U_{m,k} \dots U_{m,1}$	n_m	$e_{m,1}$ $e_{m,j}$	$e_{m,j+1} \dots e_{m,j+m}$	b_m	
(İNDEKS)	P_1	$I_{1,1} \dots I_{1,j}$	$I_{1,j+1} \dots I_{1,j+m}$	a_1	
GÖSTERGE	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
SATIRLARI	P_k	$I_{k,1} \dots I_{k,j}$	$I_{k,j+1} \dots I_{k,j+m}$	a_k	

$I_{k,s}$ ve a_k dışındaki bütün elemanlar başlangıç matematiksel karar modelinden basit olarak elde edilir. $I_{k,s}$ ve a_k aşağıdaki gibi hesaplanmalıdır:

$$I_{k,s} = \sum_{i=1}^{m} (e_{i,s} \cdot U_{i,k}) - W_{k,s} \tag{2.14.1}$$

$$a_k = \sum_{i=1}^{m} (b_i \cdot U_{ik}) \tag{2.14.2}$$

Başlangıç tablosunda temel değişkenler, bazı özel durumlar dışında, daima negatif sapma değişkenleri (n_i) nin kümesidir. Temel değişken sıra ile şimdiki çözümdeki bir değişkendir. Yani, m-amaçların olduğu sistemde sadece m-tane değişken temel olabilir. Simpleks algoritmasının iterasyonları, şimdiki çözümü daha iyiye götürüyorsa, temelde olan bir değişkenin temel olmayan bir değişkenle basit olarak değişiminden ibarettir.

Sadece birtek öncelik düzeyi kurulmuşsa bu tablo, doğrusal programlamanın simpleks tablosuna benzer olacak. Doğrusal amaç programlama modellerinin herhangi bir aşamasında k önceliği için $a_1, a_2, \ldots, a_k$ ile verilen vektör başarı düzeyini gösterir. Bu vektörün sıfır değeri bu amaçların hepsinin seviyesine ulaşılmış anlamındadır.[16]

Genel tabloda gösterge satırları mevcut çözümün eniyi olup olmadığını göstermeye yarar, eğer eniyi değilse çözümün iyileştirilmesi için temel ve temel dışı değişkenleri arasında uygun değişiklik yapılır. Şimdilik k öncelik düzeyine ulaşmayla ilgileniyorsak sadece $P_1, P_2, \ldots, P_k$ ile ilgili gösterge satırlarının hesaplanmış olması gerekir.[16]

Örnek 2.14.1

$$x_1 + x_2 + n_1 - p_1 = 10$$
$$x_1 + n_2 - p_2 = 4$$
$$5x_1 + 3x_2 + n_3 - p_3 = 55$$
$$x_1 + x_2 + n_4 - p_4 = 12$$

ve

$$\bar{x}, \bar{p}, \bar{n} \geq 0$$

kısıtlayıcılarına göre

$\bar{a} = \{(3p_1 + 4p_2), (n_3), (p_4)\}$'yi minimum yapan $\bar{x} = (x_1, x_2)$ çözümlerini bulalım.

Bu örnek için tablo aşağıda verilecektir, karışıklık olmaması için sıfır elemanları üst blok ve sol blok kısmına konmamıştır.

Tablo 2.14.2: Üç öncelik düzeyi için başlangıç tablosu

			P_3						1	
			P_2							
			P_1			3	4			
P_3	P_2	P_1	V	x_1	x_2	P_1	P_2	P_3	P_4	$\bar{b}$
			n_1	1	1	−1	0	0	0	10
			n_2	1	0	0	−1	0	0	4
	1		n_3	5	3	0	0	−1	0	55
			n_4	1	1	0	0	0	−1	12
			P_1	0	0	−3	−4	0	0	0
			P_2	5	3	0	0	−1	0	55
			P_3	0	0	0	0	0	−1	0

Başlangıç tablosunun yorumu aşağıdaki şekilde verilecektir.

$n_1 = 10$

$n_2 = 4$ (Temel değişkenler)

$n_3 = 55$

$n_4 = 12$

Öteki değişkenler (temel dışı değişkenler) sıfırdır.

P_1 ; $a_1 = 0$ olduğundan tamamen başarılmış (sağlanmış),
P_2 ; $a_2 = 55$ olduğundan tamamıyle başarılamamış,
P_3 ; $a_3 = 0$ olduğundan tamamen başarılmış.

2.15. ALGORİTMA

Başlangıç tablosu oluşturulduktan sonra, Çok Aşamalı Simpleks algoritmasının gerçek adımlarına geçeceğiz. Bu adımlar vasıtasıyla, eğer sistem çözülebilir ise doğrusal amaç programlaması problemleri için eniyi çözüme ulaşılır.

1. Adım: Mutlak kısıtlayıcılar ve amaçlar uygun dönüştürme ile hedeflere çevrilir. Bunlar vasıtasıyla başlangıç tablosu oluşturulur ve sadece birinci öncelik düzeyi için gösterge satırı hesaplanır. k=1 alıp 2 nci adıma gideriz.

2. Adım: Eniyiliğin kontrol edilmesi: a_k'yı inceleriz. a_k sıfır ise 6 ncı adıma gider, değilse k-inci gösterge satırında her pozitif değerli gösterge sayısını ($I_{k,s}$) inceleriz. Aynı sütunda, yüksek düzeyde, negatif olmayan gösterge sayısı için enbüyük pozitif $I_{k,s}$ bulunur, bu sütunu s' olarak belirleriz. $I_{k,s}$ seçiminde eşitlik durumu keyfi olarak önemsenmeyebilir (veya çiğnenebilir). Eğer böyle $I_{k,s}$ bulunmamışsa 6 ncı adıma, aksi halde 3 ncü adıma gideriz.

3. Adım: Temele girecek yeni değişkenin belirlenmesi: s' sütununda temel olmayan değişken temele girecek yeni değişkendir.

4. Adım: Ayrılacak değişkenin belirlenmesi; $b_i/e_{i,s}$ oranının, $e_{i,s} \geqq 0$ olmak üzere, en küçüğünün bulunduğu satır belirlenir. Bu satırı i' olarak alalım, i'-üncü satırla ilgili temel değişken ayrılacak değişkendir.

5. Adım: Yeni tablonun oluşturulması;

(a) Bütün $e_{i,s}$, b_i, $I_{k,s}$ elemanları boş olan yeni bir tablo kurarız. Önceki tablonun s' sütununda temel dışı değişken ile önceki tablonun i'-üncü satırındaki temel değişken yer değiştirirler.

(b) Yeni tablonun i'-satırı ($e_{i',s'}$ dışında) önceki tablonun i' satırı $e_{i',s'}$ ile bölünerek elde edilir.

(c) Yeni tablonun s' sütunu, ($e_{i',s'}$ hariç) önceki tablonun s' sütunu ($-e_{i',s'}$) ile bölünerek elde edilir.

(d) $e_{i',s'}$ konumunda yeni elemanı, $e_{i',s'}$'nin tersidir, diğer elemanlar aşağıdaki şekilde hesaplanır: $\hat{b}_i$ ve $\hat{e}_{i,s}$ hesaplanacak yeni elemanlar ve b_i ile $e_{i,s}$ bu elemanların önceki tablodan elde edilen değerleri olsun. O zaman i' satır ve s' sütunlarında olmayan bu elemanlar:

$$\hat{e}_{i,s} = e_{i,s} - \frac{(e_{i',s})(e_{i,s'})}{e_{i',s'}} \qquad (2.15.1)$$

$$\hat{b}_i = b_i - \frac{(b_{i'})\ (e_{i,s'})}{e_{i',s'}} \qquad (2.15.2)$$

(e) Tablo kurmada bir diğer adım $I_{k,s}$ ve a_k'nın yeni değerlerini bulmaktır. Bu değerler k-inci öncelik düzeyi ve daha büyük öncelik düzeylerinin hepsi için hesaplanmalıdır. Bu değerler

$$I_{k,s} = \sum_{i=1}^{m} (e_{is} \cdot u_{ik}) - w_{ks} \qquad (2.15.3)$$

$$a_k = \sum_{i=1}^{m} (b_i \cdot u_{ik}) \qquad (2.15.4)$$

ile hesaplanır.

(f) 2 nci adıma dön.

6. Adım: Daha düşük öncelikli düzeylerin değerlendirilmesi. $k = k + 1$ alalım, eğer $k > K$ (toplam öncelik sayısı) ise çözüm eniyi, $k < K$ ise P_k için gösterge satırı hesaplar ve 2 nci adıma gideriz.

Sayfa 31'deki Örnek 2.14.1'i alarak yukarıda verdiğimiz algoritma ile çözümü arayalım.

Tablo 2.15.1

Örnek 2.14.1'in birinci öncelik düzeyi için başlangıç tablosu

			P_3						1	
			P_2							
			P_1			3	4			
P_3	P_2	P_1	V	x_1	x_2	P_1	P_2	P_3	P_4	$\bar{b}$
			n_1	1	1	-1	0	0	0	10
	1		n_2	1	0	0	-1	0	0	4
			n_3	5	3	0	0	-1		55
			n_4	1	1	0	0	0	-1	12
			P_1	0	0	-3	-4	0	0	0

(1) Adım: k=1 için başlangıç tablosu tablo 2.15.1 gibi düzenlenir.
(2) Adım: $a_1 = 0$ olduğundan 6'ncı adıma gideriz.

(6) Adım: $k = k + 1 = 1 + 1 = 2$, $k \leq K$, yani $k = 2$, $K = 3$ olduğundan (2.15.3) ve (2.15.4) denklemlerini kullanarak P_2 önceliği için gösterge satırını hesaplarız. Yeni Tablo 2.15.2'deki gibi olacaktır.

Tablo 2.15.2. Birinci ve ikinci öncelik düzeyi için 2 nci Tablo

			P_3						1	
			P_2							
			P_1			3	4			
P_3	P_2	P_1	V	x_1	x_2	P_1	P_2	P_3	P_4	b
			n_1	1	1	-1	0	0	0	10
			n_2	1	0	0	-1	0	0	4
	1		n_3	5	3	0	0	-1	0	55
			n_4	1	1	0	0	0	-1	12
			P_1	0	0	-3	-4	0	0	0
			P_2	5	3	0	0	-1	0	55

2. Adım: $a_2 = 55$ bu yüzden 2 nci gösterge satırında pozitif değerli sayıları inceleyeceğiz. ($I_{i,s}$ değeri). Burada en büyük $I_{k,s}$ değeri araştırılacak, bu da $I_{i,1}$ enbüyük değerdir (+5) ve $I_{i,1}$ üzerinde negatif değerli gösterge sayısı olmadığından $s' = 1$ olur. 3 ncü adıma gideriz.

3. Adım: $s' = 1$ olduğundan x_1 giren değişkendir.

4. Adım: $\frac{b_i}{e_{i,s}}$ oranlarının pozitif en küçüğü araştırılır.

$b_1/e_{1,1} = 10/1 = 10$

$b_3/e_{2,1} = 4/1 = 4$

$b_3/e_{3,1} = 55/5 = 11$

$b_4/e_{4,1} = 12/1 = 12$

Minimum pozitif oran $b_2/e_{2,1} = 4$ olduğundan, bu oranla ilgili değişken n_2, yani $i' = 2$ öyleyse çıkacak değişken n_2'dir.

5. Adım (a): Yeni tablo x_1 ile n_2'nin konumlarını değiştirmektir.

5. Adım (b): Yeni tablonun $i' = 2$ satırı Tablo 2.15.2'de 2. satırın $e_{2,1} = 1$ ile bölünmesinden elde edilir.

5. (c): Yeni tablonun $s' = 1$ sütunu, $e_{2,1}$ dışında Tablo 2.15.2'deki 1. kolonunun $-e_{2,1} = -1$ ile bölünmesinden elde edilir. $e_{2,1}$'de yeni eleman önceki tablonun 1 nci sütun 2 nci satırındaki elemanın tersidir. 5(a), 5(b) ve 5(c)'deki sonuçlar. Tablo 2.15.3 te görülebilir.

5. (d): kalan $e_{i,s}$ ve b_i elemanları (2.15.1) ve (2.15.2) denklemleri kullanılarak hesaplanır.

5. (e): $I_{k,s}$'ların hepsini ve a_k değerlerini ($k = 1,2$ için) hesaplarız. Bu hesaplamalarda da (2.15.3) ve (2.15.4) denklemlerinden yararlanılır. 5(a)'dan 5(e)'ye kadar hesaplanan elemanların oluşturduğu tablo Tablo 2.15.3'de gösterilmiştir. 2. Adıma geçeriz.

Tablo 2.15.3

İlk değişiklik yapılıp yeni elemanları hesaplanarak oluşturulan tablo

			P_3						1	
			P_2							
			P_1			3	4			
P_3	P_2	P_1	V	n_2	x_2	P_1	P_2	P_3	P_4	$\bar{b}$
			n_1	-1	1	-1	1	0	0	6
			x_1	1	0	0	-1	0	0	4
	1		n_3	-5	3	0	5	-1	0	35
			n_4	-1	1	0	1	0	-1	8
			P_1	0	0	-3	-4	0	0	0
			P_2	-5	3	0	5	-1	0	35

2. Adım: a_2 = 35 olduğundan 2. öncelik düzeyi tamamiyle elde edilememiştir. Biz 2 nci gösterge satırındaki tüm elemanları inceleyeceğiz. $I_{2,4}$ = 5 enbüyük değer olarak göze çarpar. Ancak bu elemanın üzerinde negatif bir gösterge sayısı vardır. Burada 1 inci öncelik düzeyinin erişildiği seviyeyi bozmaksızın böyle bir değişikliği yapamayız. Öyleyse $I_{2,2}$ = 3 alınır. Bu değişiklik 1 inci öncelik düzeyinin erişilen seviyesini bozmaz, bu yüzden s' = 2 alır, 3 ncü adıma gideriz.

3. Adım: x_2 temele girecek değişkendir.

4. Adım: $b_i/e_{i,s}$ pozitif oranlarını hesaplarız.

$$\frac{b_1}{e_{1,2}} = \frac{b}{1} = 6 \qquad \text{(minimum)}$$

$$\frac{b_3}{e_{1,3}} = \frac{35}{3} = 11.67$$

$$\frac{b_4}{e_{14}} = \frac{8}{1} = 8$$

Böylece i' = 1 ve n_1 temelden çıkacak değişkendir.

5. Adım: x_2 ve n_1'in yerdeğiştireceği yeni tablo bütün elemanlarıyle 5(b) ... 5(e) adımlarının herbirindeki gibi hesaplanır. Bu hesaplamaları Tablo (2.15.4)'e taşırsak

Tablo 2.15.4

İkinci değişiklik yapıldıktan sonra yeni elemanların oluşturulması

			P_3						1	
			P_2							
			P_1			3	4			
P_3	P_2	P_1	V	n_2	n_1	P_1	P_2	P_3	P_4	$\bar{b}$
			x_2	-1	1	-1	1	0	0	6
			x_1	1	0	0	-1	0	0	4
	1		n_3	-2	-3	3	2	-1	0	17
			n_4	0	-1	0	0	0	-1	2
			P_1	0	0	-3	-4	0	0	0
			P_2	-2	-3	3	2	-1	0	17

Adım (2): $a_2 = 17$ olduğundan, 2 nci öncelikli amacın düzeyine halâ ulaşılamamıştır. Bununla birlikte, $I_{2,s}$'nin bütün pozitif elemanlarının üzerinde kendisinden daha öncelikli $I_{1,5}$ satırında negatif elemanlar olduğundan 6 ncı adıma gideriz.

6. Adım: $k = k + 1 = 2 + 1 = 3 \quad k = K \ (3 = 3)$ olur, P_k için gösterge satırı oluşturulur. Yeni tablo 2.15.5 Tablosu aşağıdaki gibi olacaktır.

Tablo 2.15.5

Örnek 2.14.1 in son en iyi tablosu

			P_3						1	
			P_2							
			P_1			3	4			
P_3	P_2	P_1	V	n_2	n_1	P_1	P_2	P_3	P_4	$\bar{b}$
			x_2	-1	1	-1	1	0	0	6
			x_1	1	0	0	0	0	0	4
	1		n_3	-2	-3	3	2	-1	0	17
			n_4	0	-1	1	0	0	-1	2
			P_1	0	-0	-3	-4	0	0	0
			P_2	-2	-3	3	2	-1	0	17
			P_3	0	0	0	0	0	-1	0

2. Adım: $a_3 = 0$ olduğundan (6) ya gideriz.

6. Adım: $k = k + 1 = 3 + 1 = 4$ olduğundan çözüm eniyidir.

∴Bu örneğin çözümü $\bar{x}^* = (x_1, x_2) = (4, 6)$ olur. Tablodan da görüleceği gibi, $\bar{a} = (0, 17, 0)$ olur. Bu 1 inci ve 3 üncü öncelik düzeylerinin tamamen, 2 nci öncelik düzeyinin ise kısmen elde edildiğini ifade eder. Birinci öncelik düzeyi ile ilgili amaçlar mutlak ise (yani kısıtlayıcılar) çözümün tamamlanabilir olması için a_1'de de sıfır düzeyi elde edilebilmelidir.

AP
1) Kıt kaynakların en iyi dağıtımında,
2) Planlama ve düzenlemede,
3) Politika Analizlerinde kullanılır.

AP Duyarlık çözümlemesine imkân verir,Bu yüzden girdi bileşimlerine göre çıktıda olabilecek değişmelerin izlenmesi mümkündür. Ayrıca AP modeli belli şartlarla, kısıtlamalarla ve girdilerle amaca ulaşma derecesini de belirler. Çeşitli kısıtlama, girdi ve hedeflerin önceliği bileşimlerine göre bir Simulasyon analizi yapılmasını kolaylaştırır.[35 ,36]

AP birçok amacı aynı modelde temsil edebilmek, amaçların hedeflerine ulaşıp ulaşmadığını gösterebilmek bakımından çok kullanışlı bir yöntemdir.

2.16. SIRALI MİNİMUM ETKİN BİR ÇÖZÜMDÜR.

Algoritma ile başarı vektörü, $\bar{a}$ sıralı minimumunu bulduk. Böyle bir çözümün arzu edilmeyen özelliği a_t $(t \neq k)$ için önemli bir iyileşme olmasına rağmen, önemsiz de olsa a_k'daki bir artmaya karşı tolerans tanımamasıdır[24]. Bu sakıncayı ortadan kaldırmak için amaçların uygun olarak seçilen istenen düzeyi aracılığıyla etkin bir çözüm elde edilebilir. Diğer herhangi bir çözümün bu çözüme baskın gelemeyeceğini ispatlayan bir teorem vereceğiz.

Doğrusal amaç programlaması için

$$a_k(\bar{x}) \leqq a_k(\bar{x}^o), \quad k = 1,\dots K \tag{2.16.1}$$

ve

$$a_k(\bar{x}) < a_k(\bar{x}^o) \quad \text{en az bir } k \text{ için} \tag{2.16.2}$$

olacak şekilde $\bar{x} \varepsilon X$ bulunamıyorsa, $\bar{x}^o$ etkin bir çözümdür.

Doğrusal Amaç programlaması modelini fiziksel amaçlar ile mutlak olmayan amaçları ayrıştırarak şöyle ifade edebiliriz:

$$A\bar{x} + n^r - \bar{p}^r = \bar{b}, \quad r \,\varepsilon\, \{1,\dots m\} \tag{2.16.3}$$

(Mutlak amaçlar: kısıtlayıcılar)

$$C\bar{x} + \bar{n}^s - \bar{p}^r = z^o, \quad s \,\varepsilon\, \{1,\dots, s\} \tag{2.16.4}$$

(Mutlak olmayan amaçlar).

D.A.P bu iki sistemi sağlayan

$$\bar{a} = \{g \quad (\bar{n}, \bar{p})\} \tag{2.16.5}$$

ifadesinin enküçüklenmesi idi.

Teorem 2.16.1[24] z_o (2.16.3) i sağlıyan herhangi bir çözüm için ulaşılamıyacak seviyede yeterince büyük istenen düzey olsun, $\bar{a}^o$ da doğrusal amaç programlaması probleminin sıralı minimum çözümü olsun, $\bar{a}$ da problemin herhangi bir diğer çözümü olsun. Bu durumda $\bar{a}$, $\bar{a}^o$, a baskın (dominate) olamaz.

İspat:

$\bar{a}^o$'ın bileşenleri $a_1^o, a_2^o, \dots, a_k^o$ ve $\bar{a}$'nın bileşenleri $a_1, a_2, \dots, a_k$ olsun, ilk önce kısıtlayıcıların orijinal kümesi sağlanmışsa hem a_1^o, hem de a_1 sıfır değerli olmalıdır. Böylece a_2^o ve a_2'yi inceleyeceğiz.

a_2^o, eşdeğer tek amaç doğrusal programlama probleminde eniyi çözümü temsil ettiğinden $a_2 < a_2^o$ olamaz. Ayrıca, $a_2 > a_2^o$ ise $\bar{a}$, $\bar{a}^o$'a baskın olamaz. Şimdi de a_3^o ile a_3'ü karşılaştıralım.

a_3^o, herhangi bir büyük öncelikli düzeyi azaltmaksızın elde edilebilir a_3'ün eniyi değerini temsil ettiğinden, a_3 aşikâr olarak a_3^o'dan küçük olamaz. a_2^o için başka eniyi çözümler yoksa, $\bar{a}_o$ tüm kalan düzeylerinin değerleri sabit olmalıdır (bir noktaya yakınsadık) ve ispat tamamlanır. Aksi durumda ise a_4^o ile a_4'ü inceleriz.

a_4^o ile a_4'ün incelenmesi a_3^o ve a_3'ün incelenmesine eşdeğerdir ve gerçekte tüm a_k^o ve a_k $(k \geq 3)$'ya uygulanabilir. Sonuç olarak, diğer çözümler $\bar{a}^o$'a baskın olamaz ve böylece $\bar{a}_o$ etkin bir çözümdür. İspat böylece tamamlanmış olur.

3. BÖLÜM
D U A L İ T E

3.1. SIRALI DOĞRUSAL AMAÇ PROGRAMLAMASINDA DUAL

Doğrusal programlama (DP) nın duali eniyi şekilde geliştirilmiştir. Bu dualin varlığı ve ondan yararlanma DP nın potansiyelinin çoğunu ortaya koymuştur. Karşılık olarak, doğrusal amaç programlamasında dual'in teorik yapısı ve özellikleri halâ inceleniyor, sonuçları bugün tam anlamıyle bilinemiyor.

Amaç programlamasının gerçek potansiyeli, belki dualinin en iyi şekilde incelenmesinden ortaya çıkacaktır[34]. Sıralı doğrusal amaç programlama modelleri için dual ilk önce 1970'lerin başlarında IGNIZIO[16] tarafından ele alınmıştır.

Daha önce tartıştığımız gibi, doğrusal amaç programlaması problemlerinin çözümleri için yaygın olarak kullanılan algoritmalar, Çok Aşamalı Simpleks ile Ardışık Doğrusal Amaç Programlaması (veya Ardışık İşlemler) idi. Markowski ve Ignizio bu iki modelin matematiksel duali ve biribiriyle olan ilişkilerini sunan bir makale yayınladılar. Çok aşamalı tabloyu, ardışık tabloya, ardışık tabloyu da çok aşamalı tabloya dönüştüren algoritmalar geliştirdiler. Bu sonuçlar, özellikle duyarlık çözümlemesinde çok önemlidir. Ardışık işlemlerle duyarlık çözümlemesi yapmak zor olacağından son dual tablosu, uygun bir algoritma ile çok aşamalı son dual tablosuna dönüştürülür, bu tablo yardımıyla duyarlık çözümlemesi çalışmaları yapılabilir.[37]

Sıralı doğrusal amaç programlamasının duali aracılığıyla, hesaplama bakımından etkin çözüm verecek algoritmalar ile S.D.A.P modelinin çözümü mümkündür.[11]

Bilindiği gibi doğrusal programlamanın duali bir doğrusal programlama problemidir, oysa amaç programlamasının duali bir amaç programlaması problemi değil, çoklu sağ taraf sabitlerine sahip bir doğrusal programlama problemidir. Bu yüzden bu dual problemi çok boyutlu dual (ÇBD) olarak adlandırılır. Dualin çok boyutlu olması problemin öncelikli düzey sayısından kaynaklanmaktadır.

Hatırlanacağı gibi matris gösterimiyle sıralı doğrusal amaç programlamasını şu şekilde ifade etmiştik:

$$[\,C \quad I \quad -I\,] \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} = \ddot{b} \tag{3.1.1}$$

$$\bar{x},\ \bar{p},\ \bar{n} > 0 \tag{3.1.2}$$

kısıtlarına göre

$$\bar{a} = [\,u \quad w\,] \begin{pmatrix} \bar{n} \\ - \\ \bar{p} \end{pmatrix} \tag{3.1.3}$$

nin sıralı minimum yapılması. Bu şekilde $\bar{a}$, başlangıç temel değişkenleri de içerecektir. Yani $\bar{a}$ hem temel, hem de temelde olmayan değişkenlerin bir fonksiyonudur. Biz, sadece temelde olmayan değişkenlerle $\bar{a}$'yi ifade etmek istiyoruz. Öyleyse amaç fonksiyonda bulunan başlangıç temel değişkenleri (n_i) elimine edeceğiz. (3.1.1) denklemini n_i için çözer ve (3.1.3)' de yerine koyarsak, yeni amaç fonksiyon şu şekilde olacaktır: Yani, (3.1.1) denklemi $C_{\bar{x}} - I_{\bar{n}} - I_{\bar{p}} = \bar{b}$ şeklinde yazılabilir. Buradan $\bar{n}$ çözülürse,

$\bar{n} = \bar{b} - C_{\bar{x}} + I_{\bar{p}}$ olacaktır. Bunu (3.1.3)'de yerine koyarsak

$$\bar{a} = [u \quad w] \begin{pmatrix} \bar{b} - c\bar{x} + \bar{p} \\ \bar{p} \end{pmatrix}$$

$$= u\,\bar{b} - uc\bar{x} + up + w_{\bar{p}} = u\bar{b} - uc_{\bar{x}} + u_{\bar{p}} + w_{\bar{p}}$$

$$= [-uc \quad 0 \quad u + w] \begin{pmatrix} \bar{x} \\ n \\ p \end{pmatrix} + u\,.\,\bar{b}$$

Öyleyse çok aşamalı ilk (primal) model şöyle olacaktır:

(3.1.1) ifadesi (-1) ile çarpılırsa

$$[-C \quad -I \quad I] \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} = -b \qquad (3.1.4)$$

$$\bar{x},\ \bar{n},\ \bar{p} \geqq 0 \qquad (3.1.5)$$

kısıtlarına göre

$$\bar{a} = [-uc \quad 0 \quad u + w] \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} + u\bar{b} \qquad (3.1.6)$$

nın sıralı minimum yapılmasıdır.

3.2. ÇOK BOYUTLU DUAL

Ignizo [16] tarafından gösterildiği gibi her sıralı doğrusal amaç programlamasının primali için çok boyutlu olarak bilinen dual problemi vardır.

Yukarıda verilen çok aşamalı primalin çok boyutlu dualinin genel şekli aşağıdaki gibi verilir:

$$[-C \quad -I \quad I]^T Y^T \leq [-UC \quad 0 \quad U+W]^T \tag{3.2.1}$$

Y kısıtlanmamış değişken

kısıtlayıcılarına göre

$$\bar{\alpha}^T = -\bar{b}^T Y^T + [U\bar{b}]^T \tag{3.2.2}$$

eşitliğini bileşenlerine göre maksimum yapan Y değerlerinin araştırılması. Bu modeli daha açık yazılımıyla K öncelik sayısı olmak üzere

$$\begin{bmatrix} -C^{(1,K)} \\ -I^{(1,K)} \\ I^{(1,K)} \end{bmatrix} Y \leq \begin{bmatrix} (-U^1C^1)^T \\ 0 \\ (U^1+W^1)^T \end{bmatrix}, \ldots, \begin{bmatrix} -(U^KC^K) \\ 0 \\ (U^K+W^K)^T \end{bmatrix} \tag{3.2.3}$$

kısıtlarına göre

$$\text{Sıralı Max } \alpha = -b^{(1,K)}Y + \{U^1b^1, \ldots, U^Kb^K\} \tag{3.2.4}$$

olacak şekilde Y değerlerinin bulunması olarak da ifade edebiliriz. α: her öncelik düzeyinin başarısını gösteren K-boyutlu vektördür.

$$Y^T = \begin{bmatrix} Y_1^1 & \ldots & Y_1^k & \ldots & Y_1^K \\ & & \vdots & & \\ & & Y_i^k & & \\ & & \vdots & & \\ Y_m^1 & \ldots & Y_m^k & \ldots & Y_m^K \end{bmatrix} = [Y^1 \ \ldots \ Y^k \ \ldots \ Y^K] \tag{3.2.5}$$

Burada Y^T, mxK-boyutlu bir matristir, Y_i^k çok aşamalı primalin i-inci hedefi ve k-inci öncelikle ilgili dual değişkendir.

Sözkonusu çok boyutlu dualin kısıtlarını eşitliğe dönüştürürsek

$$\text{Amaç fonk : } \max \bar{\alpha} = -Y\bar{b}^T + U\bar{b}^T \quad (3.2.6)$$

$$\text{Kısıtlar: } [-C \quad I]^T Y^T + S^T = [-uc \quad u + w]^T \quad (3.2.7)$$

olur. Burada

$$S \ , Y \geq 0$$

$$S^T = \begin{bmatrix} s_1^1 & \dots & s_1^k & \dots & s_1^K \\ \vdots & & & & \\ s_{n+m}^1 & & s_{n+m}^k & & s_{n+m}^K \end{bmatrix} = (\bar{s}_1, \dots, \bar{s}_{n+m}) \quad (3.2.8)$$

S^T, n + m x k boyutlu bir matris ve s_q^k : q-inci sıralı dual ve k-inci öncelikle ilgili aylâk dual değişkendir.

ÖRNEK 4.1

Sıralı doğrusal amaç programlamasının aşağıdaki şekilde verilen ilk (primal) modelini düşünelim.

$$\text{Amaç fonk : } \min \bar{a} = \{(P_1 + P_2), (3n_3 + 4n_4)\}$$

$$\begin{aligned} \text{Kısıtlar : } \quad x_1 + x_2 + n_1 - p_1 &= 12 \\ 2x_1 + x_2 + n_2 - p_2 &= 20 \\ 16x_1 + 10x_2 + n_3 - p_3 &= 160 \\ 3x_1 + 5x_2 + n_4 - p_4 &= 60 \\ \bar{x}, \bar{n}, \bar{p} &\geq 0. \end{aligned}$$

Eğer burada

$$n_3 = 160 - 16x_1 - 10x_2 + p_3$$

$$n_4 = 60 - 3x_1 - 5x_2 + p_4$$

alırsak, başarı vektöründen çıkarılmış temel değişkenlerden oluşan ilk modelimizin duali şöyle olacaktır.

Amaç fonk : Sıralı max $\bar{\alpha} = (-12 \quad -20 \quad -160 \quad -60)Y + \{0,\ 720\}$

Kısıtlayıcılar:

$$\begin{bmatrix} -1 & -2 & -16 & -3 \\ -1 & -1 & -10 & -5 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} Y \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} -60 \\ -50 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 4 \end{bmatrix}.$$

Burada Y, kısıtsız ve çok boyutludur. Bu örnek için

$$Y = \begin{bmatrix} y_1^1 \\ y_2^1 \\ y_3^1 \\ y_4^1 \end{bmatrix}, \begin{bmatrix} y_1^2 \\ y_2^2 \\ y_3^2 \\ y_4^2 \end{bmatrix},$$

ve

$$C = \begin{bmatrix} 1 & 1 \\ 2 & 1 \\ 16 & 10 \\ 3 & 5 \end{bmatrix}, \quad b = \begin{bmatrix} 12 \\ 20 \\ 160 \\ 60 \end{bmatrix}$$

dir. Yine

$U^1 = (0, 0)$, $w^1 = (1, 1)$

$U^2 = (3, 4)$, $w^2 = (0, 0)$

$$b^1 = \begin{bmatrix} 12 \\ 20 \end{bmatrix}, \qquad b^2 = \begin{bmatrix} 160 \\ 60 \end{bmatrix}$$

$$C^{(1)} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}, \qquad C^2 = \begin{bmatrix} 16 & 10 \\ 3 & 5 \end{bmatrix}$$

$U^1 b^1 = \{0\}$, $U^2 . b^2 = \{720\}$

olacaktır.

3.3. ÇOK BOYUTLU DUAL ALGORİTMASI

Çok boyutlu dual bir "doğrusal programlama" problemi olduğundan, çözümü geleneksel simplekse dayalı bir algoritma yardımıyla yapılabilir. Yani, yapacağımız ilgili doğrusal programlama modellerinin serisine Simpleks algoritmasını uygulamaktır. Sıra halindeki her model

(1) sağ taraf sabitlerinin değişmesi,

(2) belirli kısıtlayıcıların, öncelikli doğrusal problem de elde edilen çözümlere bağlı olarak, çıkarılması dışında bilinen yöntemlere eşdeğerdir. Sağ taraf sabitlerinin değişmesi öncelik düzeyinin farklılığından doğmaktadır. Kısıtlayıcıların çıkarılması ise "aylâklığın tamlayanı"na ilişkin özelliğin bir sonucudur [11]. Buna göre geçersiz kısıtlayıcılar sözkonusu olduğunda sonraki problemde bu kısıtlayıcılar çıkarılarak dual çözüm yapılır.

3.4. ALGORİTMA

1. Adım: (3.2.3) ve (3.2.4)' deki gibi olarak Ç.B.D. problemi kurulur. k=1 alalım.

2. Adım: Burada sadece k'ıncı sağ taraf vektörün-den oluşan geleneksel doğrusal programlama modelini kurar, uygun bir simpleks algoritması kullanarak çözüm yaparız. k = K ise 4'üncü adıma, değilse 3'üncü adıma gideriz.

3. Adım: Önceden çözülen doğrusal programlama modeli için, geçersiz kısıtlayıcıları çıkarırız, modelde bu tür kısıtlayıcılar yoksa 4'üncü adıma, aksi halde k = k + 1 alır, 2'nci adımdan devam ederiz.

4. Adım: k'ıncı sağ taraf sabiti ve çok boyutlu dual için eniyi olanı şimdiki çözümdür. Sıralı doğrusal amaç programlamasına ilişkin eniyi çözüm k'ıncı dual model için başlangıç temel değişkenlerle ilgili olan gölge fiyatlar tarafından belirlenir.

Örnek 4.1. Önceki sayısal örnekte kurulmuş olan Ç.B.D. problemi için yukarıdaki algoritmayı uygulamalıyız. Başlangıç DP modeli (K = 1 için) çözülmüş olmalıdır. Yani

$$\begin{bmatrix} -1 & -2 & -16 & -3 \\ -1 & -1 & -10 & -5 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} Y^1 \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad , Y^1 \text{ kısıtsız}$$

kısıtlarına göre

$$\max.\alpha^1 = -12y_1^1 - 20y_2^1 - 160y_3^1 - 60y_4^1 + \{0\}.$$

Y^1 kısıtsız olmasına rağmen 3, 4, 5 ve 6 ncı kısıtlayıcılar Y^1'in pozitif olmasını gerektirir. Son iki kısıtlayıcı y_3^1 ve y_4^1 sıfır olduğuna göre, kalan doğrusal programlama modelini çözersek (grafikle de çözülebilir),

$Y^1 = (0, 0, 0, 0)$ olduğunu görürüz. Buradan da

$\alpha^1 = 0 + 0 = 0$

olur. Bu çözüm için 7'nci ve 8'inci kısıtlar gereksiz olduklarından, bunlar k=2 için DP modelinden çıkarılacaklardır. Yeni model

Amaç fonk: max $\alpha^2 = -12y_1^2 - 20y_2^2 - 160y_3^2 - 60y_4^2 + \{720\}$

Kısıtlar:

$$\begin{bmatrix} -1 & -2 & -16 & -3 \\ -1 & -1 & -10 & -5 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} Y^2 \leq \begin{bmatrix} -60 \\ -50 \\ 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 4 \end{bmatrix}$$

Y^2 Kısıtsız

olacaktır.

Yine Y^2 kısıtsız olmasına rağmen, 3., 4., 5., ve 6'ncı kısıtlar Y^1'in pozitif olmasını gerektirir. Bu problem Büyük-M ve üst sınırlı düzeltilmiş simpleks yöntemleriyle kolaylıkla çözülebilir. Biz Büyük M-yöntemiyle[38] çözelim. Bu yöntemde problem bir max. problemi ise M(> 0) çok büyük sayı olmak üzere yapay değişkenlere -M değeri verilir. Problem bir min. problemi ise bu değişkenlere M değeri verilir. Bu değerler amaç fonksiyonunun mahiyetine göre ulaşmak istediğimiz amaca karşıt fiyatlar olduğundan, yapay değişkenlerin temelden çıkmaları beklenerek çözüme ulaşılır. Çözüm algoritması olarak da k=1 için çok aşamalı simpleks algoritması adımları problemin maksimum oluşu gözetilerek uygulanabilir.

Kısıtlayıcılara yapay ve aylâk değişkenlerin eklenmesiyle model

$$y + 2y_2 + 16y_3 + 3y_4 - S_1 + S_3 = 60$$
$$y_1 + y_2 + 10y_3 + 5y_4 - S_2 + S_4 = 50$$
$$y_3 + S_5 = 3$$
$$y_4 + S_4 = 4$$
$$y_i \geqq 0, \; S_i \geqq 0$$

kısıtlayıcılarına göre

$$\max \alpha^2 = -12y_1^2 - 20y_2^2 - 160y_3 - 60y_4 + 720$$

olacaktır.

Tablo 3.4.1

Başlangıç Tablosu, Büyük- M Yöntemi

		$\bar{x}_B$	-12 y_1	-20 y_2	-160 y_3	-60 y_4	0 S_1	0 S_2	S_3	S_4	S_5	S_6
← -M	S_3	60	1	2	16	3	-1	0	1	0	0	0
-M	S_4	50	1	1	10	5	0	-1	0	1	0	0
0	S_5	3	0	0	1	0	0	0	0	0	1	0
0	S_6	4	0	0	0	1	0	0	0	0	0	1
			2M-12	3M-20	26M-160	-8M-60	-M	-M				

Tablo 3.4.2

İkinci Tablo

		x_B	y_1	y_2	$\overset{-M}{s_3}$	$\overset{-60}{y_4}$	s_1	s_2	y_3	s_4	s_5	s_6
-160	y_3	$\frac{60}{16}$	$\frac{1}{16}$	$\frac{2}{16}$	$\frac{1}{16}$	$\frac{3}{16}$	$-\frac{1}{16}$	0	1	0	0	0
← -M	s_4	$\frac{200}{16}$	$\frac{6}{16}$	$-\frac{4}{16}$	$-\frac{10}{16}$	$\frac{50}{16}$	$\frac{10}{16}$	-1	0	1	0	0
0	s_5	$\frac{12}{16}$	0	0	0	0	0	0	0	0	1	0
0	s_6	4	0	0	0	1	0	0	0	0	0	1

$$\frac{50M}{16} - 60$$
↑

Tablo 3.4.3

Üçüncü Tablo

			y_1	y_2	$\overset{-M}{s_3}$	$\overset{-M}{s_4}$	s_1	s_2	y_3	y_4	s_5	s_6
-160	y_3	3	$\frac{32}{800}$	$\frac{112}{800}$	$\frac{20}{800}$	$-\frac{3}{50}$	$-\frac{1}{10}$	$\frac{3}{50}$	1	0	0	0
← -60	y_4	4	$\frac{6}{50}$	$\frac{-4}{50}$	$-\frac{10}{50}$	$\frac{16}{50}$	$\frac{10}{50}$	$-\frac{16}{50}$	0	1	0	0
0	s_5	$\frac{12}{16}$	0	0	0	0	0	0	0	0	1	0
0	s_6	0	$-\frac{6}{50}$	$\frac{4}{50}$	$\frac{10}{50}$	$\frac{-16}{50}$	$\frac{-10}{50}$	$\frac{16}{50}$	0	0	0	1
			$\frac{8}{5}$	-2,4	-M-8	-M	-9/6	-46/5				
			↑									

Tablo 3.4.4

Son optimal Tablo

					$-M$	$-M$	0	0				
		$\bar{x}_B$	y_4	y_2	S_3	S_4	S_1	S_2	y_3	y_1	S_5	S_6
-160	y_3	$5/3$	$-\frac{1}{3}$	$\frac{1}{6}$	$\frac{9}{100}$	$-\frac{1}{6}$	$-\frac{1}{6}$	$\frac{1}{6}$	1	0	0	0
-12	y_1	$\frac{200}{6}$	$\frac{50}{6}$	$\frac{-4}{6}$	$\frac{-10}{6}$	$\frac{16}{6}$	$\frac{10}{6}$	$\frac{-16}{6}$	0	1	0	0
0	S_5	$\frac{12}{16}$	0	0	0	0	0	0	0	0	1	0
0	S_6	4	0	0	0	0	0	0	0	0	0	1
			$-\frac{40}{3}$	$-\frac{4}{3}$	$-M$	$-M$	$-\frac{20}{3}$	$-\frac{16}{3}$				

Bütün $c_j - z_j$'ler negatif ve problem bir maksimum problemi olduğundan en iyi çözüme erişilmiştir.

Primal modelin en iyi çözüm tablosundaki çözüm vektöründeki değerler, dual modelin en iyi çözüm tablosunda $c_j - z_j$ satırındaki aylâk değişkenlerin (mutlak değeri) değerlerini verir.[39]

Buna göre bu problemde aylâk (slack) değişkenler S_1 ve S_2'dir. S_1'in gölge fiyatı $-\frac{20}{3}$, S_2'nin gölge fiyatı $-\frac{16}{3}$, öyleyse

$$x_1 = \frac{20}{3}$$

$$x_2 = \frac{16}{3}$$

noktası primalin en iyi çözümüdür. $\alpha^2 = \frac{-2000+2160}{3} = 53.3$ olacaktır.

Çok boyutlu dualin çözümünde birinci öncelikteki kısıtlayıcılar her zaman sağlanır. α^1, birinci öncelikteki kısıtlayıcıların (hedeflerin) elde edilme derecesinin ölçüsüdür. α^2, ise ikinci öncelikteki kısıtların elde edilme derecesinin ölçüsüdür. $\alpha^1 = 0$ çıktığına göre birinci öncelik düzeyindeki kısıtlar sağlanmış, ancak $\alpha^2 = 53.3$ olduğuna göre ikinci öncelik düzeyindeki kısıtlar sağlanamamıştır, denilebilir.

İlk modelde az sayıda değişken fakat çok sayıda kısıtlayıcı varsa problemin çözümü güçleşecektir. Dualin önemli oluşu bu tür problemlerin çözümünü kolaylaştırmasıdır.[40]

3.5. DUALITY ÖZELLİKLERİ

Teorem 3.5.1[34]

$\bar{x}$, $\bar{n}$, $\bar{p}$ primal için herhangi bir mümkün çözüm ve Y de dual için mümkün bir çözüm ise $\bar{a} \leq \bar{\alpha}$ dır.

İspat: $\bar{x}$, $\bar{n}$, $\bar{p}$ primal de mümkün bir çözüm olduğundan (2-13.4)'ü ele alalım. Bu ifadeyi -Y ile çarparsak

$$[\,Y \quad -C \quad -I \quad I\,] \begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} = -Y\bar{b} \qquad (3.5.1)$$

olur. Benzer olarak Y de mümkün bir çözüm olduğundan

$$[\,Y \quad -C \quad -I \quad I\,] \leq [\,-UC \quad 0 \quad u + w\,] \qquad (3.5.2)$$

Bunu $\begin{pmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{pmatrix} \geq 0$ ile çarparsak ve geçişliliği kullanırsak

$$-Y\bar{b} = [Y \quad -C \quad -I \quad I] \begin{bmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{bmatrix} \leq [-UC \quad 0 \quad U + W] \qquad (3.5.3)$$

her ifadeye $U\bar{b}$'yi ilâve edersek istenen sonuç sağlanmış olur.

Teorem 3.5.2[34] $\bar{a}^* = \bar{\alpha}^*$ olacak şekilde, $\bar{x}^*$, $\bar{n}^*$, $\bar{p}^*$ primalin ve $\bar{Y}^*$ da dualin herhangi bir mümkün çözümü ise çözümler kendi problemlerinde optimaldir.

İspat: $\bar{x}$, $\bar{n}$, $\bar{p}$ primalin mümkün bir çözümü olsun (varsayalım),

$$[-UC \quad 0 \quad U + W] \begin{bmatrix} \bar{x} \\ \bar{n} \\ \bar{p} \end{bmatrix} \geq -Y^*\bar{b} \qquad (3.5.6)$$

$$[-UC \quad 0 \quad U + W] \begin{bmatrix} \bar{x}^* \\ \bar{n}^* \\ \bar{p}^* \end{bmatrix} = \bar{a}^* - U\bar{b}. \qquad (3.5.7)$$

Bunun için primal bir minimizasyon problemi olduğundan $\bar{x}^*$, $\bar{n}^*$, $\bar{p}^*$ optimal olmalıdır. $\hat{Y}$ de dualin mümkün bir çözümü olsun

$$-\hat{Y}\bar{b} \leq [-UC \quad 0 \quad U + W] \begin{bmatrix} \bar{x}^* \\ \bar{n}^* \\ \bar{p}^* \end{bmatrix} = -Y^*\bar{b} = \alpha^* - U\bar{b} \qquad (3.5.8)$$

olur. Bu yüzden maksimizasyon probleminde Y* optimaldir.

Teorem 3.5.3[34] Primal optimal ise $\bar{a} = \bar{\alpha}$ olacak şekilde dual içinde optimal bir çözüm vardır.

İspat: X_B^*, primalin temel optimal çözümü olsun. B de uygun temel matris olsun. X_B^* optimal olduğundan

$$W_B^* \; B^{-1*} \; C \leq 0, \tag{3.5.9}$$

$$W_B^* \; B^{-1*} \leq U, \tag{3.5.10}$$

ve

$$-W_B^* \; B^{-1*} \leq W \tag{3.5.11}$$

$$Y^* = -W_B^* \; B^{-1*} + U \tag{3.5.12}$$

tanımlayalım.

$$Y^* [-C \quad -I \quad I] = [\; W_B^* \; B^{-1} \quad C - UC \quad W_B \; B^{-1*} - U \quad -W_B^* \; B^{-1} + U] =$$

$$[-UC \quad 0 \quad U + W] \tag{3.5.13}$$

optimallik şartıyle. Böylece Y^*,(3.2.1)'i sağladığından mümkün çözümdür. Optimallikte:

$$\bar{a}^* = W_B^* \; B^{-1*} \; \bar{b} = W_B^* \; B^{-1*} \; \bar{b} - U\bar{b} + U\bar{b}$$

$$= [\; W_B^* \; B^{-1*} - U] \; \bar{b} + U\bar{b} = Y^* \bar{b} + U\bar{b} = \bar{\alpha}^* \tag{3.5.13}$$

Teorem 3.5.4 [34,41] Bir doğrusal amaç programlaması probleminde primal ve dual için eniyi çözümlerin q-inci dual aylâk değişken vektörü ile q-inci primal değişkenin herbir elemanının çarpımları sıfırdır. Benzer olarak i-inci primal aylâk değişken ve i-inci dual değişkenin herbir elemanının çarpımı da sıfırdır.

$(q = 1,2,\ldots,n+m)$ ve $(i=1,2,\ldots,m)$

Aylâklığın tamlayanı olarak adlandırılan bu teoremin ispatı için ilgili referanslara başvurulabilir.

4. BÖLÜM
GENELLEŞTİRİLMİŞ TERS

4.1. GENELLEŞTİRİLMİŞ TERS YAKLAŞIMI

Çoklu amaçlar çok ölçütlü eniyileme veya amaç programlaması teknikleri aracılığıyla da ele alınabilir. Amaç programlaması gibi çok ölçütlü eniyileme tekniklerinin tümü öngörücü yöntemlerdir. Bir dereceye kadar karar vericinin model üzerindeki kontrolüne engel olmaktadırlar. Bir eniyileme probleminde ölçütlerin (amaçların) ustalıkla kullanılması ele alınan algoritmanın sağladığı çözümler üzerinde karar vericiye bir takım imkânlar verebilir, ancak bu dolaylı bir işleyiştir ve yürütülmesi karmaşık bir yoldur. Oysa çoklu ölçütleri analiz ederken karar vericiye model üzerinde birtakım serbestlikler tanıyan yaklaşımlar da vardır. Bu yaklaşımlardan biri de "Genelleştirilmiş ters" tekniğidir.[42]

Tek ölçütlü, Çok ölçütlü ve amaç programlaması yaklaşımlarının hepsi uzaklık fonksiyonu modeli olarak düşünülebilir[43]. Amaç programlaması mutlak değer normunun (L_1 metriği*) enküçüklenmesi özelliğine sahip bir yaklaşım iken, genelleştirilmiş ters (GT) yaklaşımı da Öklid normunun (L_2-metriğinin) enküçüklenmesini sağlamaktadır.[13]

4.2. TARİHÇE

GT kavramı ilk olarak 1903 yılında Fredholm tarafından ele alınmıştır[44]. Bu konudaki ilk yazılı yayın Moore tarafından 1920 yılında gerçekleştirilmiştir. 1955 yılında Penrose genelleştirilmiş ters (GT) in temel özelliklerini kurmuştur. Ijiri[13] GT kavramının iş planlaması problemlerine uygulamasını gerçekleştirerek doğrusal denklem sistemlerinin çözümünde önemli rol almasını sağlamıştır.

Bir doğrusal denklem sistemi genel olarak şöyle ifade edilebilir.

$$A\underline{x} = \underline{b} . \qquad (4.2.1)$$

Burada A, m x n - boyutlu bir matris $\underline{b}$, m-boyutlu bir vektör ve x , n-boyutlu bir vektördür. (4.2.1)'deki orijinal problem eşitsizliklerden oluşmuşsa aylâk değişkenler vasıtasıyla eşitliklere dönüştürülebilir. Burada A matrisi tekil olmayan (non-singular) bir matris ise (4.2.1) sisteminin çözümü $\underline{x} = A^{-1}\bar{b}$ ile bulunur. Eğer A matrisi tekil ise veya kare değil ise A^{-1} şeklinde bir tersi yoktur. Bu durumda A^{-1}'in tüm özelliklerini içinde bulunduran yeni bir kavram ortaya çıkmıştır. Genelleştirilmiş tersler.

4.3. GENELLEŞTİRİLMİŞ TERS TANIMI

Eğer $\hat{x}$, $A\bar{x} = b$ 'nin bir çözümü ise, bu denklemler sisteminin bütün çözümleri

$$x = \hat{x} + A^{o}z \qquad (4.3.1)$$

ile verilebilir. Burada A^{o}, A'nın çekirdek uzayına karşı gelen temel matristir (yani $A^{o}A = 0$) ve A'nın rankı r ise A^{o},(n-r)xr tipinde bir matristir. A, r tane doğrusal bağımsız sütun veya satır bulundurmaktadır. Z ise (n-r)-boyutlu keyfi bir vektördür. (4.3.1)'deki keyfi Z vektörünün bileşenleri ustalıkla kullanılırsa (4.2.1)'in tüm çözümlerini verecektir. Modelde A^o gibi bir serbestlik derecesi bulundurması ve $\hat{x}$ gibi bir çözüm elde edebilmesi genelleştirilmiş terslerin kullanılmasıyla mümkün olmuştur. (4.3.1)'deki ilk terim, yani $\hat{x}$ çözümü; A'nın genelleştirilmiş tersi b vektörüyle çarpılarak elde edilir.

4.4. BİR MATRİSİN GENELLEŞTİRİLMİŞ TERSİ

Herhangi bir A matrisi için Penrose aşağıdaki dört özelliği sağlıyan tek bir B matrisinin varolduğunu göstermiştir.[42]

(1) $$A\,B\,A = A \qquad (4.4.1)$$

(2) $$B\,A\,B = B \qquad (4.4.2)$$

(3) $$(AB)^T = AB \qquad (4.4.3)$$

(4) $$(BA)^T = BA. \qquad (4.4.4)$$

Böyle bir B matrisi A'nın genelleştirmiş tersi olarak adlandırılır. A tekil değilse $B = A^{-1}$ olur ki bu dört özelliğin sağlandığı kolaylıkla görülür. Bundan sonra A'nın genelleştirilmiş tersi için A^+ kullanacağız.

Genelleştirilmiş terslerin Penrose tanımlaması ile ilgili ispatsız bazı teoremler vereceğiz.

Tanım: 4.4.1[13]

Her A_{mxn}: matrisi için, A matrisinin genelleştirilmiş tersi

$$A^+ \Leftrightarrow \begin{array}{rll} \text{i)} & AA^+ \ \text{Simetrik}, & (AA^+ = (AA^+)^T) \\ \text{ii)} & A^+A \ \text{Simetrik}, & ((A^+A)^T = A^+A) \\ \text{iii)} & AA^+A = A & \\ \text{iv)} & A^+AA^+ = A^+ & \end{array}$$

dir.

Teorem 4.4.1[44] Her $\bar{b}$ vektörü için $\|A\bar{x} - \bar{b}\|^2 = \|\bar{e}\|^2 = \bar{e}'\bar{e}$ yi minimum yapan $\bar{x}$ vektörleri arasında $A^+\bar{b}$ vektörü minimum normludur.

Bu teorem enküçük kareler çözümünün normunu (öklid) minimal yapan özelliğinden dolayı uygulamalarımızda önemli bir rol alacaktır, bu yüzden bunun ispatı üzerinde duracağız.

İspat:

$(A\bar{x} - \bar{b}) = (A\bar{x} - AA^+\bar{b}) + (AA^+\bar{b} - \bar{b})$ yazılabilir.

$(A\bar{x} - AA^+\bar{b}) \in R(A)$, $-(I - AA^+)\bar{b} \in R(A)^{\perp}$ olduklarından,

$$\| A\bar{x} - \bar{b}\|^2 = \|A\bar{x} - AA^+\bar{b}\|^2 + \|AA^+\bar{b} - \bar{b}\|^2$$

olacaktır. Bu eşitliğin sağ tarafındaki, ikinci ifade sabit olduğundan $\|A\bar{x} - AA^+\bar{b}\|$ ifadesini enküçükleyen, $\bar{x} = A^+\bar{b}$ çözümü bu eşitliği minimum yapan enküçük kareler çözümüdür.

Genelleştirilmiş tersin hesaplanması ile ilgili çeşitli algoritmalar vardır, biz hesaplamalarda bize yardımcı olabilecek bazı teoremleri vereceğiz.

Teorem 4.4.2[45]

$A \in \mathbb{C}^{mxn}$ ve $\mathrm{Rank}(A) = 1$ ise $A^+ = \frac{1}{\alpha} A^*$ dir.

Burada $\alpha = \mathrm{Tr}\, A^*A = \sum_{i,j} |\alpha_{ij}|^2$, dir (A^* = A'nın transpozu).

Teorem 4.4.3[45]

$A \in \mathbb{C}^{mxn}$ ve $\mathrm{Rank}(A) = r$ olsun. $\{V_1, V_2, \ldots, V_r\}$, $R(A^*)$ nın temeli ve $\{w_1, w_2, \ldots, w_{n-r}\}$, $N(A)$ için bir temel ise

$A^+ = [v_1, v_2, \ldots v_r, 0, \ldots 0]\,[Av_1\ Av_2 \ldots Av_r\ w_1\ w_2 \ldots w_{n-r}]^{-1}$

dir.

Önerme 4.4.1[45]

$A \in \mathbb{C}^{mxn}$ ise, $A = BC$ ve $\text{rank}(A) = \text{rank}(A) = \text{rank}(B) = \text{rank}(C) = r$ olacak şekilde $B \in \mathbb{C}^{mxr}$ ve $C \in \mathbb{C}^{rxn}$ matrisleri vardır.

Teorem 4.4.4[45]

$A \in \mathbb{C}^{mxn}$, $B \in \mathbb{C}^{mxr}$, $C \in \mathbb{C}^{rxn}$ olacak şekilde $A = BC$ ise ve $\text{rank}(A) = \text{rank}(B) = \text{rank}(C) = r$ ise

$$A^+ = C^*(C\,C^*)^{-1}\,(B\,B^*)^{-1}\,B^* \quad \text{dır.}$$

4.4.3 Teoremi yardımıyle bir A matrisinin genelleştirilmiş tersi A matrisinden bağımsız olarak hesaplanabilir.

Örnek 4.4.1[45]

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 2 & 2 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$R(A^*)$, $\left\{ \begin{matrix} 1 \\ 1 \\ 2 \end{matrix} \quad \begin{matrix} 0 \\ 2 \\ 2 \end{matrix} \quad \begin{matrix} 1 \\ 0 \\ 1 \end{matrix} \quad \begin{matrix} 1 \\ 0 \\ 1 \end{matrix} \right\}$ sistemi tarafından gerilir.

$R(A^*)$ in temelini teşkil eden bir alt küme:

$$\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \quad \text{dir.}$$

$$A \cdot \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 2 \\ 2 \end{bmatrix} \quad \text{ve} \quad A \cdot \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ 1 \\ 1 \end{bmatrix}$$

olacağından

$$A^+ \begin{bmatrix} 3 \\ 2 \\ 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \quad A^+ \begin{bmatrix} 3 \\ 4 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

olacaktır.

Şimdi $R(A)^+ = N(A^*)$ ın bir temeli hesaplanabilir.

$A^*\bar{x} = 0$ sistemi çözülürse

$x_1 + x_3 + x_4 = 0$

$x_1 + 2x_2 = 0$

$2x_1 + 2x_2 + x_3 + x_4 = 0$

$x_3 = a$ ve $x_4 = b$ alarak sistem çözülürse

$x_1 = -a - b$

$x_2 = \dfrac{a + b}{2}$, Çözüm $x_o = \begin{bmatrix} -a-b \\ \frac{a+b}{2} \\ a \\ b \end{bmatrix} = a \begin{bmatrix} -1 \\ \frac{1}{2} \\ 1 \\ 0 \end{bmatrix} + b \begin{bmatrix} -1 \\ -\frac{1}{2} \\ 0 \\ 1 \end{bmatrix} = x_3 \begin{bmatrix} -1 \\ \frac{1}{2} \\ 1 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} -1 \\ \frac{1}{2} \\ 0 \\ 1 \end{bmatrix}$

olur.

$x_3, x_4 \in \mathbb{C}$, o zaman

$$A^+ \begin{bmatrix} -1 \\ 1/2 \\ 1 \\ 0 \end{bmatrix} = A^+ \begin{bmatrix} -1 \\ 1/2 \\ 0 \\ 1 \end{bmatrix} = 0 \,.$$

Bunların hepsini birleştirirsek

öyleyse

$$A^{+}\begin{bmatrix} 3 & 3 & -1 & -1 \\ 2 & 4 & 1/2 & 1/2 \\ 2 & 1 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

$$A^{+} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 3 & 3 & -1 & -1 \\ 2 & 4 & 1/2 & 1/2 \\ 2 & 1 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix}^{-1}$$

4.4.4 Teoreminden yararlanarak bir matrisin genelleştirilmiş terslerini bulalım.

Örnek 4.4.2

$$A = \begin{bmatrix} 1 & 2 & 1 & 4 & 1 \\ 2 & 4 & 0 & 6 & 6 \\ 1 & 2 & 0 & 3 & 3 \\ 2 & 4 & 0 & 6 & 6 \end{bmatrix}$$ matrisinin (GT) sini hesaplayalım.

1) Elemanter satır işlemlerini kullanarak

$$E_A = \begin{bmatrix} 1 & 2 & 0 & 3 & 3 \\ 0 & 0 & 1 & 1 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

E_A'da birim vektörlerine karşı gelen A vektörleri

$$B = \begin{bmatrix} 1 & 1 \\ 2 & 0 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}$$ olacaktır.

E_A nın sıfır olmıyan diğer elemanları

$$C = \begin{bmatrix} 1 & 2 & 0 & 3 & 3 \\ 0 & 0 & 1 & 1 & -2 \end{bmatrix}$$ olacaktır.

rank(A) = rank(B) = Rank(C) = 2

$A^+ = C^*(C\,C^*)^{-1}(B^*B)^{-1}B^*$ olacağından

$$A^+ = \frac{1}{1161}\begin{bmatrix} 27 & 6 & 3 & 6 \\ 54 & 12 & 6 & 12 \\ 207 & -40 & -20 & -40 \\ 288 & -22 & -11 & -22 \\ -333 & 98 & 44 & 98 \end{bmatrix}$$ olacaktır.

4.5. GENELLEŞTİRİLMİŞ TERSLERİN ÖZELLİKLERİ[13]

Genelleştirilmiş ters aşağıdaki özelliklere sahiptir.

1. $A^{++} = A$;

2. $(A^*)^+ = (A^+)^*$;

3. A tekil olmayan matris ise $A^+ = A^{-1}$;

4. $(A^*A)^+ = A^+A^*$;

5. A, A^*A, A^* ve A^+A matrislerinin rankı A^+A matrisinin izine eşittir;

6. $A = 0 \Rightarrow A^{+} = 0$;

7. A tam sütun ranklı ise $A^{+} = (A^{*}A)^{-1}A^{*}$;

8. A tam satır ranklı ise $A^{+} = A^{*}(AA^{*})^{-1}$;

9. AA^{+} matrisi $\mathbb{R}^{n}$ nin R(A) üzerine dik izdüşüm dönüşümünün matrisi, $I - AA^{+}$ ise $\mathbb{R}^{n}$ nin $R(A)^{\perp}$ üzerine dik izdüşüm dönüşümünün matrisidir;

10. B_{mxr}, $¢_{rxr}$ ve D_{rxn} matrislerinin rankı, $1 \leq r \leq \min(m, n)$ olmak üzere, r ye eşit ise $(BCD)^{+}=D^{+}C^{+}B^{+}$;

11. U ve V Unitary matris olmak üzere $(UAV)^{+}=V^{*}A^{+}U^{*}$;

12. A normal ise $A^{+}A = AA^{+}$ ve $(A^{n})^{+} = (A^{+})^{n}$;

13. A idempotent ise $A^{+} = A$

14. $\lambda \in ¢$ olmak üzere $\lambda \neq 0$ ise $(\lambda A)^{+} = \lambda^{+}A^{+}$ ve $\lambda = 0$ ise $(\lambda A)^{+} = 0$ dır. ($\lambda^{+} = \lambda^{-1}$ anlamındadır)

15. $R(A^{*}) = R(A^{+})$

4.6. BİR MATRİSİN ÇEKİRDEK UZAYI

Herhangi bir $A \in ¢^{mxn}$ matrisi için

$N(A) = \{\bar{x} \in C^{n} : A\bar{x} = \bar{0}\}$ ile tanımlanan kümeye A matrisinin çekirdek (veya boş uzayı) uzayı denir.[44]

Homojen olmıyan doğrusal

$$A\bar{x} = \bar{b} \tag{4.6.1}$$

denkleminin genel çözümü, herhangibir $\bar{x}_{o}$ belirli çözümü ile

$$A\bar{x} = \bar{0} \tag{4.6.2}$$

homojen denkleminin genel çözümünün toplamı olacağından,[46] çekirdek uzayının yararı büyüktür.

Bir matrisin çekirdek uzayının tabanı, matrisi elemanter satır işlemleriyle köşegenleştirerek elde edilebilir. Rankı r olan A_{mxn} matrisi için önce uygun satır ve sütun değişiklikleriyle $\begin{bmatrix} I_r : H \\ \cdots \\ 0 \end{bmatrix}$ şeklinde bir D matrisi, elde edilir, burada I_r : r - boyutlu birim matris ; 0 : (m-r)xn-boyutlu sıfır matrisi; H: rx(n-r)-boyutlu kalanlar matrisidir. Bu A matrisi, E; elemanter satır işlemlerinin matrisi, P de sütun değişikliklerine ilişkin permütasyon matrisi olmak üzere A = (EDP) özelliğini koruyan bir matristir.

$$\begin{bmatrix} -H \\ \cdots\cdots \\ I_{n-r} \end{bmatrix} \tag{4.6.3}$$

matrisi oluşturularak, köşegenleştirme işlemlerinde sütun değişikliği yapılmışsa, bu matrisin uygun satırları değiştirilerek çekirdek uzayının matrisi A^o elde edilir ki, bu da

$$P^* \begin{bmatrix} -H \\ \cdots \\ I_{n-r} \end{bmatrix} \tag{4.6.4}$$

matrisidir. Çünkü,

$$AA^o = EDP\,P^* \begin{bmatrix} -H \\ \cdots \\ I_{n-r} \end{bmatrix} = E \begin{bmatrix} I_r & : & H \\ \cdots & \cdots & \cdots \\ & 0 & \end{bmatrix} \begin{bmatrix} -H \\ \cdots \\ I_{n-r} \end{bmatrix} = E0 = 0$$ [13]

öyleyse A^o, A matrisinin çekirdek uzayının tabanını verecektir.

Örnek 4.6.1

$$A = \begin{bmatrix} 8 & 8 & 16 & 8 \\ 4 & 4 & 8 & 4 \\ 3 & 3 & 7 & 5 \end{bmatrix}$$ matrisi köşegenleştirilerek,

D matrisi şu şekilde elde edilecektir.

$$\begin{bmatrix} 8 & 8 & 16 & 8 \\ 4 & 4 & 8 & 4 \\ 3 & 3 & 7 & 5 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 \\ 3 & 3 & 7 & 5 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 2 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 2 & 1 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \rightarrow \left[\begin{array}{cc:cc} 1 & 0 & \frac{3}{2} & 1 \\ 0 & 1 & \frac{1}{2} & 0 \\ \hdashline 0 & 0 & 0 & 0 \end{array}\right]$$

I. Adım: 1. satır 8 e bölündü, ve -4 ile çarpılarak 2 satıra ilâve edildi.

II.Adım: 2. satır ile 3. satır yer değiştirdi ve birinci satır -3 ile çarpılarak 2. ci satıra eklendi.

III.Adım: 2. sütun ile 4 cü sütun değiştirildi.

IV.Adım: 2. ci satır 2 ile bölündü, ve -1 ile çarpılarak 1.ci satıra eklendiğinde son matris elde edilir ki, bu matris $\left[\begin{array}{c:c} I_r & H \\ \hdashline \multicolumn{2}{c}{0} \end{array}\right]$ şeklindeki D matrisimizdir.

$\begin{bmatrix} -H \\ I_{n-r} \end{bmatrix}$ matrisini elde etmek kolaydır. Bu matrisin de 2. satırı ile 4.cü satırı yer değiştirilirse elde edilen matris A^o matrisidir.

$$A^o = \begin{bmatrix} -\frac{3}{2} & -1 \\ 0 & 1 \\ 1 & 0 \\ -\frac{1}{2} & 0 \end{bmatrix} ,$$ $AA^o = 0$ olduğu kolaylıkla görülebilir.

Genelleştirilmiş ters yardımıyle $A\ddot{x} = \bar{b}$ sisteminin bulunan $\hat{x}$ çözümü belirli bir çözümdür. A nın Çekirdek uzayındaki herhangi bir $\bar{x}_o$ vektörü ile $\hat{\bar{x}} = (A^+\bar{b})$ vektörünün toplamı da bir çözümdür.

$$A(\hat{\bar{x}} + \bar{x}_o) = A\hat{\bar{x}} + A\bar{x}_o = A\hat{\bar{x}} + \bar{0} = \bar{b} + \bar{0} = \bar{b} \qquad (4.6.5)$$

Buradan da $\hat{\bar{x}}$, $A\bar{x} = \bar{b}$ sisteminin bir çözümü ise, $\hat{\bar{x}} + \bar{x}_o$ vektörü de bu sistemin bir çözümü olacaktır. Öyleyse $A\bar{x} = \bar{b}$ sisteminin bütün çözümleri şu şekilde ifade edilebilir.

$$\bar{x} = A^{+}\bar{b} + A^{o}\bar{z} \tag{4.6.6}$$

$\bar{z}$ vektörü keyfi bir vektördür, A^{o}, A'nın Çekirdek uzayı A^{+} ise A'nın genelleştirilmiş tersidir.

Bir A matrisi için Çekirdek uzayının tabanı A^{o}, tek değildir. Gerekli görülürse A^{o} yerine tek olarak belirlenebilen n x n boyutlu $(I - A^{+}A)$ matrisi alınabilir. Çünkü A nın çekirdek uzayındaki herhangi $\bar{x}_o \in N(A)$ vektörü keyfi seçilen $\bar{z}$'ler için

$$x_o = (I - A^{+}A)\bar{z} \tag{4.6.7}$$

olarak ifade edilebilir.[13]

Bu kez (4.6.6) yerine çözüm vektörü

$$\bar{x} = A^{+}\bar{b} + (I - A^{+}A)\bar{z} \tag{4.6.8}$$

olacaktır.

4.7. BİR MATRİSİN GÖRÜNTÜ UZAYI

Herhangi bir $A \in \mathbb{R}^{mxn}$ matrisi için

$$R(A) = \{\bar{b} \in \mathbb{R}^{m} : \forall \bar{x} \in \mathbb{R}^{n} \text{ için } \bar{b} = A\bar{x}\}$$

ile tanımlanan kümeye A matrisinin görüntü uzayı veya A nın açıklığı adı verilir. Genelleştirilmiş terslerin bulunmasında R(A)'ya ilişkin taban kullanılarak daha az işlemle sonuca gidilebilir, bu yüzden R(A) nın yararı büyüktür.[44]

$A\bar{x} = \bar{b}$ sisteminin çözülebilir olması için gerek ve yeter şart $\bar{b}$ vektörünün A nın sütunlarının doğrusal bileşimi olarak ifade edilebilmesidir. Bir diğer ifade ile $\bar{b} \in R(A)$ ise sistem çözülebilirdir.[47] Bir matrisin sütun uzayı ile satır uzayının boyutu eşittir, bu boyut rank olarak adlandırılır.[48,49]

A nın açıklığı R(A) ile A nın transpozunun açıklığı $R(A^*)$ daki vektörler arasında birebirlik bir dönüşüm olduğundan bu iki alt uzay eşdeğerdir.

R(A*) içindeki herhangi bir vektör, $\begin{bmatrix} I_r \\ \cdots \\ H^* \end{bmatrix}$ matrisinin sütunlarının doğrusal bileşimi olarak ifade edilebilir. Yani bu matris R(A*) ın bir tabanını oluşturur. R(A*) ile N(A) alt uzayları biribirine dik uzaylardır. Aynı şekilde R(A) ile N(A*) alt uzayları da biribirine dik uzaylardır.[13]

4.8. GENELLEŞTİRİLMİŞ TERSLERİN ÇOKLU AMAÇLARIN ÇÖZÜMÜNDE KULLANILMASI

Örnek 4.8.1[13]

$$\begin{aligned} x_1 + 0.5x_2 &= 2,5 \\ 3x_1 + 2x_2 + x_3 &= 12 \\ 5x_1 + x_4 &= 10 \end{aligned}$$

$$x_1,\ x_2,\ x_3,\ x_4 \geq 0$$

Yukarıdaki örneği genelleştirilmiş ters yaklaşımıyle çözmeye çalışalım.

Çözüm:

$$A = \begin{bmatrix} 1 & 0.5 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 5 & 0 & 0 & 1 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad \text{ve} \quad b = \begin{bmatrix} 2,5 \\ 12 \\ 10 \end{bmatrix}$$

A nın rankı 3 olduğundan tam satır ranklı bir matristir.
$A^{+} = A^{*}(AA^{*})^{-1}$ olacaktır.

$$A^{+} = \frac{1}{62}\begin{bmatrix} 16 & -2 & 10 \\ 92 & 4 & -20 \\ -232 & 60 & 10 \\ -80 & 10 & 12 \end{bmatrix}$$

Çekirdek uzayının mümkün bir tabanı

$$A^{o} = \frac{1}{62}\begin{bmatrix} 2 \\ -4 \\ 2 \\ -10 \end{bmatrix} \quad , \quad \text{burada} \quad AA^{o} = 0$$

dır.

Buradan da $A\bar{x} = \bar{b}$ sisteminin çözüm kümesi

$$\bar{x} = \frac{1}{62}\begin{bmatrix} 16 & -2 & 10 \\ 92 & 4 & -20 \\ -232 & 60 & 10 \\ -80 & 10 & 12 \end{bmatrix}\bar{b} + \frac{1}{62}\begin{bmatrix} 2 \\ -4 \\ 2 \\ -10 \end{bmatrix} z \qquad (4.8.1)$$

olacaktır, z keyfi bir skalerdir.

$x_i \geqq 0$ şartı arandığından çözümün

$A^{+}\bar{b} + A^{o}z \geqq 0$ veya

$A^{o}z \geqq - A^{+}\bar{b}$ şartını sağlaması gereklidir.

$$\bar{b} = \begin{matrix} 2,5 \\ 12 \\ 10 \end{matrix}$$ olduğundan

$$\bar{x} = \frac{1}{62}\begin{bmatrix} 116 \\ 78 \\ 240 \\ 40 \end{bmatrix} + \frac{1}{62}\begin{bmatrix} 2 \\ -4 \\ 2 \\ -10 \end{bmatrix} z \qquad (4.8.2)$$

$x_i \geqq 0$ kısıtı yüzünden z yi tamamen keyfi seçemeyiz.

$$-58 \leqq z \leqq 4 \qquad (4.8.3)$$

aralığındaki bütün z ler için $x_i \geqq 0$ şartı sağlanmış olacaktır. $z = 0$ için, $\bar{x} = (116/62,\ 78/62,\ 240/62,\ 40/62)$ çözüm kümesi 4.4.1 Teoremine göre minimum normlu veya $||A\bar{x} - \bar{b}||$ yi minimum yapan bir çözümdür.

$\bar{b}$ hedef vektörü $\bar{b} = \begin{bmatrix} 3,5 \\ 12 \\ 10 \end{bmatrix}$ şeklinde olsaydı çözüm nasıl olacak?

Çözüm:

$$\bar{x} = \frac{1}{62}\begin{bmatrix} 112 \\ 170 \\ 8 \\ -40 \end{bmatrix} + \frac{1}{62}\begin{bmatrix} 2 \\ -4 \\ 2 \\ -10 \end{bmatrix} z, \qquad (4.8.4)$$

böyle bir durumda $x_i \geqq 0 \Leftrightarrow z = -4$ olmasıdır. Bunun için, $\bar{x}$, üzerinde pozitiflik kısıtından dolayı-sonsuz sayıda çözümler olmasına karşın-sadece bir tek çözüm sözkonusudur. Bu da

$$\bar{x} = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad \text{veya} \quad \bar{x} = (2,\ 3,\ 0,\ 0) \quad \text{dir.}$$

4.9. GT - YAKLAŞIMIYLA AP - YAKLAŞIMININ KARŞILAŞTIRILMASI

Anderson ve Earle[20] Amaç programlaması ve doğrusal programlama yolu ile elde edilen çözümlerin bir karşılaştırmasını yaparak beslenme problemleri için, birtakım eleştirilere rağmen[50] amaç programlamasının doğrusal programlamadan daha üstün çözümler sağladığını göstermişlerdir.

Genelleştirilmiş ters yaklaşımı yardımıyla sağlanan minimum normlu enküçük kareler çözümünün, Amaç programlaması yaklaşımıyla elde edilen çözüm kadar etkin olacağı kanısındayız. Bu yüzden beslenme problemlerinde Anderson ve Earle'nın önerdiği yaklaşım yerine Genelleştirilmiş ters yaklaşımının kullanılmasının bazı durumlarda daha uygun olacağını söyleyebiliriz.

Örnek 4.9.1

$$x_1 + 0.5x_2 = 2.5$$
$$3x_1 + 2x_2 \leqslant 12$$
$$5x_1 \leqslant 10$$

Amaçlarını en iyi şekilde doyuran çözüm, amaç programlaması yardımıyla araştırılırsa (tüm amaçlar aynı önemde)

$$S = \text{Amaç fonksiyon} = \min \Sigma |Ax_i - b|$$
$$= |x_1 + 0.5x_2 - 2.5| + |3x_1 + 2x_2 - 12| + |5x_1 - 10| \quad \text{ve}$$

hedefler,

$$x_1 + 0.5x_2 = 2.5$$
$$3x_1 + 2x_2 \leqslant 12$$
$$5x_1 \leqslant 10 ,$$

iki değişkenli olduğundan grafik yoluyla çözüm yapılabilir.

Bütün uç noktalar (mümkün temel çözüm) ve Amaç fonksiyon değeri;

$x_1 = (2.5,0)$ $\quad S_1 = |2.5-2.5|+|7.5-12|+|12.5-10| = 6.5$

$x_2 = (2,3),$ $\quad S_2 = |3.5-2.5| +|6+6-12|+|10-10| = 1$

$x_3 = (4,0),$ $\quad S_3 = |4-2.5|+|12-12|+|20-10| = 11,5$

$x_4 = (2,1),$ $\quad S_4 = |2.5-2.5|+|8-12|+|10-10| = 4$

$x_5 = (0,6),$ $\quad S_5 = |3-2.5|+|12-12|+|10-0| = 10.5$

$x_6 = (0,5),$ $\quad S_6 = |2.5-2.5|+|10-12|+|10-0| = 12$

kolaylıkla da görüleceği gibi en az sapmayı veren çözüm (2,3) noktasıdır. Öyleyse, Amaç programlaması için en iyi çözüm (2,3) noktasındadır.

Şimdi aynı problem için, genelleştirilmiş ters yaklaşımı yardımıyle çözüm bulalım.

S = Amaç fonk = Min $\|A\bar{x} - \bar{b}\|$ olacaktır.

$$A = \begin{bmatrix} 1 & 0,5 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 5 & 0 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 2.5 \\ 12 \\ 10 \end{bmatrix}$$

ve Sayfa 72 den de hatırlanacağı gibi

$$\bar{x} = \frac{1}{62} \begin{bmatrix} 116 \\ 78 \\ 240 \\ 40 \end{bmatrix} + \begin{bmatrix} 2 \\ -4 \\ 2 \\ 10 \end{bmatrix} z$$

olacaktır. Buna göre z = o alınırsa 6T'nin çözümü $x_1=116/62$, $x_2 = 78/62$ $x_3 = 240/62$, $x_4 = 40/62$ olacaktır.

GT çözümü; bu amaçları eniyi doyuran çözümdür.

Tablo 4.9.1

AP ile GT nın karşılaştırılması

	AP kullanıldığında hedeften sapma miktarı (%)	GT kullanıldığında hedeften sapma miktarı (%)
1. Hedef	$\frac{1}{2.5} = 0{,}40$	0
2. Hedef	$\frac{0}{12} = 0$	0,3225
3. Hedef	$\frac{0}{10} = 0$	0,0645
Toplam sapma:	0,40	0,3870

Tablodan da görüldüğü gibi amaç programlaması kullanıldığında hedef ($\bar{b}$) ile çözüm düzeyi arasındaki sapma miktarı, Genelleştirilmiş ters kullanıldığında oluşacak sapma miktarından daha büyüktür. Öyleyse GT yaklaşımı amaçları doyurmada çoğunlukla AP den daha duyarlı bir yaklaşımdır.

GT nin diğer bir avantajı $A\bar{x} = \bar{b}$ denklem sistemi tutarlı olmasa bile (çözümsüz ise) mümkün olduğu kadar en küçük kareler çözümünü sağlamasıdır. Karar değişkenlerinin pozitiflik şartının aranmadığı problemler için de etkin bir yöntemdir [13]. Ayrıca denklem sistemlerinin eşitlik durumunda da çok etkindir.

4.10. HEDEFLERİN AĞIRLIKLANDIRILMASI VE SIRAYA KONMASI

Yönetici için bazı amaçlar diğerlerinden daha önemli olabilir. Amaçların veya hedeflerin aynı birimle ölçülüyor olmaması durumunda sıraya konması veya ağırlıklandırılması suretiyle çözüm araştırılmalıdır.

$A\bar{x} = \bar{b}$, $\bar{b}$ hedeflerinin sıralanışına göre $A_i\bar{x} = b_i$ olarak ayrıştırılmış olduğunu varsayalım, b_k en-önemli hedefimiz olsun, b_{k-1} ikinci dereceden önemli,...,b_1 en az derecede önemli hedef olsun. A_i matrisi, b_i hedefine karşı gelen satır matrisi olacaktır.

Hedeflerin önem derecelerine göre sıralı çözümü araştırılacağına göre, ilk planda en önemli hedefi sağlayan çözümler kümesi

$$\bar{x} = A_k^{+} b_k + A_k^{o} \bar{z}_k \quad (\bar{z}_k \in \mathbb{R}^{n-rk}) \tag{4.10.1}$$

bulunacaktır. Sonra ikinci öncelikli hedefleri doyuran çözümler araştırılır. Mümkünse

$$A_{k-1} \bar{x} = b_{k-1} \tag{4.10.2}$$

denklemini sağlayan $\bar{x}$ çözümü bulunur. Bununla birlikte bu denklemi sağlayan $\bar{x}$ çözümünün (4.10.1) denklemini de sağlayan çözüm olması gerekir, yani

$$A_{k-1} \bar{x} = A_{k-1} A_k^{+} b_k + A_{k-1} A_k^{o} \bar{z}_k = b_{k-1} \tag{4.10.3}$$

veya

$$A_{k-1} A_k^{o} \bar{z}_k = b_{k-1} - A_{k-1} A_k^{+} b_k \tag{4.10.4}$$

bu ifadenin sağ tarafı sabit olduğuna göre, $\bar{z}_k$ sabit değildir ve

$$\bar{z}_k = (A_{k-1} A_k^{o})^{+} (b_{k-1} - A_{k-1} A_k^{+} b_k) + (A_{k-1} A_k^{o})^{o} \bar{z}_{k-1} \tag{4.10.5}$$

dir. $\bar{z}_k$ nın bu değeri (4.10.1) de yerine konursa en büyük öncelikli hedeflerin düzeyinden uzaklığı minimum yaptıktan sonra ikinci dereceden öncelikli hedeflerin düzeyinden uzaklığı (ℓ_2-metriği) nı minimum yapan çözüm elde edilir, yani

$$\bar{x} = A_k^+ b_k A_k^o (A_{k-1}A_k^o)^+ (b_{k-1} - A_{k-1}A_k^+ b_k) + A_k^o (A_{k-1}A_k^o)^o \bar{z}_{k-1} \qquad (4.10.6)$$

elde edilecektir.

Aynı mantıkla en az önemli hedefe kadar aynı işlemler sürdürülür.

Aynı öncelik düzeyindeki hedeflerin ağırlıklandırılmasına gelince,

$$d^2 = \sum_{j=1}^{s} \alpha_j \, (A_j\bar{x} - b_j)^2 = \sum_{j=1}^{s} (\sqrt{\alpha_j} A_j \bar{x} - \sqrt{\alpha_j} b_j)^2 \qquad (4.10.7)$$

fonksiyonunu minimize etmek amaç olduğuna göre A_j deki ve b_j deki her elemanı $\sqrt{\alpha_j}$ ile çarpar, önceki bölümlerde olduğu gibi genelleştirilmiş ters aracılığıyle çözüm bulmaya çalışırız.

Örnek 4.10.1[13]

$$\begin{aligned} x_1 + 0.5x_2 &= 4 \\ 3x_1 \qquad &= 12 \\ 5x_1 \qquad &= 10 \\ x_1, x_2 &\geqq 0 \end{aligned}$$

İlk kısıt veya hedef öncelikli hedefimiz olsun. 2. ile 3. hedefler de ikinci öncelikli olmak üzere bu sistemi genelleştirilmiş ters tekniği ile çözelim. $x_1 + 0.5x_2 = 4$

eşitliğini sağlayan çözüm kümesi

$$\bar{x} = \begin{bmatrix} 0.8 \\ 0.4 \end{bmatrix} 4 + \begin{bmatrix} -0.5 \\ 1 \end{bmatrix} z = \begin{bmatrix} 3.2 \\ 1.6 \end{bmatrix} + \begin{bmatrix} -0.5 \\ 1 \end{bmatrix} z \qquad (4.10.8)$$

olacaktır. Şimdi ikinci öncelikteki hedefler kümesinin çözümü; 1. ci hedefin çözümünü de sağlayacak şekilde araştırılırsa ve $\alpha_2 = 4$ alınırsa

$$\begin{bmatrix} 6 & 0 \\ 5 & 0 \end{bmatrix} \bar{x} = \begin{bmatrix} 24 \\ 10 \end{bmatrix} \qquad (4.10.9)$$

ifadesinde $\bar{x}$ yerine (4.10.8) eşitliği yazılırsa

$$\begin{bmatrix} 6 & 0 \\ 5 & 0 \end{bmatrix} \left[\begin{bmatrix} 3.2 \\ 1.6 \end{bmatrix} + \begin{bmatrix} -0.5 \\ 1 \end{bmatrix} z \right] = \begin{bmatrix} 24 \\ 10 \end{bmatrix} \qquad (4.10.10)$$

veya

$$\begin{bmatrix} -3.0 \\ -2.5 \end{bmatrix} z = \begin{bmatrix} 4.8 \\ -6 \end{bmatrix} \qquad (4.10.11)$$

eşitliği elde edilecektir.

(4.10.11) eşitliğinin genelleştirilmiş ters yardımıyla çözümü:

$$\begin{bmatrix} -3.0 \\ -2.5 \end{bmatrix}^{+} = \frac{1}{15.25} \begin{bmatrix} -3 & -2.5 \end{bmatrix} \qquad (4.10.12)$$

olduğundan

$$z = \frac{1}{15.25} \begin{bmatrix} -3 & -2.5 \end{bmatrix} \begin{bmatrix} 4.8 \\ -6 \end{bmatrix} = \frac{12}{305} \qquad (4.10.13)$$

$$-1.6 \leq z \leq 6.4 \qquad (4.10.14)$$

z'nin bu değeri (4.10.8) de yerine konursa

$$x_1 = \frac{194}{61}, \quad x_2 = \frac{100}{61}$$ çözümleri elde edilir.

Genelleştirilmiş terslerin uygulama alanlarından bazıları şunlardır: İzdüşüm teorisi, Lineer Denklem Sistemleri, Doğrusal Programlama, Çok Amaçlı Programlama, Stokastik Modeller, Mühendislik, Fizik ve İstatistik.

5. BÖLÜM

UYGULAMA

Eski çağlardan beri insanlar çeşitli gereksinmelerinin karşılanmasında hayvanlardan faydalanmışlar ve hayvancılık insanların en önemli uğraşlarından biri olagelmiştir. Hayvansal ürünlerin insan beslenmesindeki rolü dikkate alındığında hayvancılığın gelecekte de önemli bir çalışma alanı olacağı söylenebilir.

Hayvancılıkta beklenen gelişmenin sağlanabilmesi bir yandan verim yeteneği yüksek hayvanların yetiştirilmesine diğer taraftan bu hayvanların çevre koşullarının iyileştirilmesine bağlı görülmektedir. Çevre koşulları denildiğinde öncelikle hayvanların beslenmesi, yani yem sorunu akla gelmektedir[51].

Yem: Madde ve enerji bakımından hayvanın yaşama ve verim ihtiyaçlarını karşılamak amacıyla ve belli sınır ve şartlarda yedirildiği zaman hayvanın sağlığına zararlı olmayan maddeler ve bunların karışımlarıdır[52,53].

Hayvanların uygun şekilde beslenebilmesi, gereksinme duydukları besin maddelerinin yeterli miktarda ve dengeli verilmesi ile mümkün olur. Hayvanların gereksinme duydukları besin maddelerini tam ve dengeli şekilde verebilmek için karma yemler hazırlanmaktadır. Yurdumuzda hazırlanan karma yemlerin olanaklar ölçüsünde ucuz olması esas amaç olmaktadır. Oysa hazırlanan yemin ucuz olması kadar besin değeri bakımından da kaliteli olması çok önemlidir.

Karmaya girecek yem miktarlarının hesaplanmasında çeşitli yöntemler kullanılabilir[54]. Bunlar: Klasik Yöntem, Doğrusal Programlama, Benzetim Yöntemi, Karesel Proğramlamadır. Bu yöntemler, belli bir amaca göre kısıtlayıcıları sağlayan en iyi çözümü araştırmaya dayanır. Oysa biz amaçlarıda hedeflere dönüştürerek, hedeflerden en az sapmayı sağlayan yöntemlerden Amaç Programlaması ve Genelleştirilmiş Ters Yaklaşımını kaliteli (besin değeri yüksek) karma yemlerin hazırlanmasında etkin bir araç olarak kullanabiliriz.

Amaç programlaması insan beslenmesi problemlerinde de kullanılan[20] ve dengeli beslenmeyi sağlayan çok kullanışlı bir yöntemdir. Amaç proğramlaması gibi hedeflerden sapmaları enküçüklemeye çalışan Genelleştirilmiş ters yaklaşımı özellikle kısıtların eşitlik ve denklem sisteminin tutarsız olması durumunda AP ye yeğlenir. Bu yüzden biz karma yemlerin hazırlanışında bazı yemler için iki yöntemi karşılaştırdık. Böyle problemlerde Genelleştirilmiş ters yaklaşımını da etkin olarak kullanmak mümkündür.

Bu yöntemlerin karma yem hazırlamada kullanılabilmesi için karmaya girebilecek yemlerin; (i) besin maddeleri içerikleri (ii) fiyatları (iii) nitelikleri (kısıtlayıcıları) bilinmesi gerekmektedir.

Araştırmada kullanılan yem fiyatları ile ilgili bilgiler 31 Eylül 1985 tarihinde Türk A.Ş. Ankara Yem fabrikası kayıtlarından alınmıştır.

Tablo 5.1.1: Önemli Yem Maddelerinin Ortalama Besin İçerikleri

YEM MADDELERİ	Ham Protein (%)	Ham Selüloz (%)	Ham Yağ (%)	Ham Kül (%)	ME Kcal/kg	ND/100 kg	% Ca	% P	% METHİONİN	% LİSİN
Akdarı	10	10	5	2	3000	75	0,02	0,30	0,15	0,20
Arpa	10	7	3	4	2760	75	0,07	0,40	0,14	0,25
Ayçiçek T.K.(Ext.)	30	20	2	7	1900	60	0,44	0,83	0,70	1,12
Balık Unu	60	-	5	11	2900	67	4,50	2,50	1,80	5,00
Bakla	25	7	1,5	4	2400	70	0,16	0,40	0,25	1,52
Bira Posası	25	19	5	4	1900	52	0,25	0,50	0,57	0,72
Bitkisel ve Hayvansal Yağ	-	-	99,4	-	8000	-	-	-	-	-
Buğday	11	4	5	4	3000	15	0,08	0,25	0,20	0,40
Çavdar	12	3	2	2	2660	70	0,07	0,40	0,16	0,40
Et-Kemik Unu	44	-	11	22	2400	60	14,00	6,00	0,62	3,25
Fındık Küspesi (Exp.)	40	7	7	6	2650	68	0,20	0,60	0,54	1,07
Fındık İçi Kabuğu	7	21	13	3	2000	50	-	-	-	-
İspirto Mayası	39	6	2	8	2200	50	0,10	1,40	0,64	3,6
İstiridye Kabuğu	-	-	-	85	-	-	33,00	-	-	-
Kalsiyum Karbonat	-	-	-	100	-	-	40,00	-	-	-
Kalsiyum Fosfat	-	-	-	100	-	-	25,00	18,00	-	-
Kan Unu	75	-	1	4	2900	56	0,28	0,28	1,12	6,0
Kemik Unu(	6	-	5	80	1900	18	24,00	10,00	-	-
Kepek (Buğday)	13	11	4	6	1900	50	0,10	1,20	0,20	0,60
Keten T.K.(Exp.)	34	10	8,6	6,6	2200	74	0,40	0,90	0,50	1,30
Kireç Taşı (Mermer)	-	-	-	85	-	-	33,00	-	-	-
Kuru Ot	10	27	2,5	7	-	30	0,50	0,25	-	-
Bayat Ekmek	8	4	-	4	-	70	0,2	-	-	-
Mısır	9	3	4	2	3370	80	0,02	0,30	0,15	0,25
Mısır Grizi	18	8	4	3	2800	62	0,04	0,22	0,40	0,50
Mısır Kepeği	10	12	8	4	2500	55(?)	?	?	?	?
Mısır Özü Küspesi (Exp.)	23	11	8	8,5	2500	62(?)	1,30	0,90	0,35	0,90
Mısır Proteini	50	2	5	2	3800	80	0,10	0,40	1,35	1,00
Mısır Unu	7	1	3	1	3100	77(?)	-	-	-	-
Pamuk T.K.(Exp.)	32	16	2	7	2600	55	0,20	1,20	0,43	1,35
Pancar Posası (Kuru)	8	20	0,7	9	625	50	0,83	0,10	0,01	0,60
Pirinç Kepeği	13	8	16	8	2500	50	0,10	1,35	0,17	0,50
Razmol	12	9	5	2,5	2000	55	0,04	0,25	0,20	0,60
Selektör Altı	14	8	5	10	2300	50	0,10	0,85	0,17	0,40
Sorohum Darı	10	8	3	2	2300	75	0,03	0,30	0,18	0,27
Soya Küspesi	42	7	7	6	2300	70	0,15	0,65	0,59	2,56
Susam Küspesi (Exp.)	42	7	7	12	2500	70	0,20	1,30	1,48	1,37
Süpürge Tohumu	10	7	5	3	2000	71	0,02	0,30	0,15	0,20
Şeker Pancarı Melası	9	-	-	9,5	2000	50	0,10	0,02	-	-
Üre	262	4	4	8	-	-	-	-	-	-
Yer Fıstığı Küspesi(Exp.)	45	11	6	5,5	2500	79	0,17	0,55	0,41	1,55
Yonca Unu (İkinci elek)	17	20	3	11	1500	45	1,41	0,24	0,29	0,73
Yulaf	10	10	5	3	1540	65	0,11	0,33	0,20	0,70
D.C.P.	-	-	-	1	-	-	22	18	-	-

Kaynak: T.A.Ş. Yem Sanayii Genel Müdürlüğü, Teknoloji Böl. 16.4.1982.

Karmaya girebilecek yemlerin besin maddeleri içerikleri Yem Sanayii Genel Müdürlüğü Teknoloji bölümünden sağlanmıştır. Yemlerin nitelikleri (kısıtlayıcıları) ise[55] nolu kaynaktan alınmıştır.

5.1. AP İLE SÜT YEMİNİN HAZIRLANMASI

$$0{,}10\ X_1 + 0{,}30\ X_2 + 0{,}13\ X_3 + 0{,}12\ X_4 + 0{,}08\ X_5 \geqslant 160$$

$$0{,}07\ X_1 + 0{,}20\ X_2 + 0{,}11\ X_3 + 0{,}09\ X_4 + 0{,}04\ X_5 \leqslant 150$$

$$0{,}04\ X_1 + 0{,}07\ X_2 + 0{,}06\ X_3 + 0{,}025\ X_4 + 0{,}04\ X_5 + X_6 \leqslant 80$$

$$0{,}004\ X_1 + 0{,}0083 X_2 + 0{,}012 X_3 + 0{,}0025 X_4 + 0{,}002 X_5 \geqslant 6$$

$$0{,}0007 X_1 + 0{,}0044 X_2 + 0{,}001 X_3 + 0{,}0004 X_4 + 0{,}0008 X_5 + 0{,}4 X_6 = 7$$

$$X_1 + X_2 + X_3 + X_4 + X_5 + X_6 = 984$$

$$54{,}02\ X_1 + 51{,}46\ X_2 + 41{,}38\ X_3 + 37{,}59\ X_4 + 34{,}97\ X_5 + 2{,}6 X_6 \leqslant 45000$$

kısıtlarına göre

$$\bar{a} = \{n_1 + p_2 + p_3 + n_4 + p_5 + n_5 + p_6 + n_6,\ p_7\} \qquad (5.1.2)$$

ifadesinin sıralı minimumu AP ile çözülürse

SÜT Y HEDEF NO	2 a Vektörü bileşeni	6 Çözüm Sayısı	7 Pozitif Sapma	7 Negatif Sapma
1	0.000	0.000	63.378	0.000
2	0.000	661.027	0.000	5.260
3		0.000	0.000	21.591
4		0.000	.113	0.000
5		313.371	0.000	0.000
6		9.602	0.000	0.000
7			0.000	0.000

çözümü elde edilecektir. Burada $\bar{a}$ = (0, 0) olması karma yemin tüm gereksinimleri karşılıyan yem olduğu anlaşılıyor. p_1 = 63.378 olması en iyi çözümdeki ham protein değerinin, sınır değeri olan 160 tan p_1 kadar fazla olduğunu gösterir. Aynı şekilde n_2 = 5.260 olması da en iyi çözümdeki ham selüloz değerinin n_2 kadar az olacağı anlamındadır, $n_5 = p_5 = 0$, $n_6 = p_6 = 0$ ve $n_7 = p_7 = 0$ olması 5.nci, 6.ncı ve 7.nci hedeflerin tam sağlandığını ifade eder.
Bu problemde i = 1,2..6 ya kadar olan hedeflere 1.nci öncelik son hedefe de ikinci öncelik verilmiştir. Tüm hedeflere $u_i = w_i = 1$ ağırlığı verilmiştir. Aşağıdaki tabloda daha açık olarak en iyi çözüm değerleri verilmiştir.

Tablo 5.1.1: AP ile Elde Edilen En İyi Süt Yeminin Nitelikleri.

	EN İYİ ÇÖZ. MİKTARI	ALT SINIR	ÜST SINIR	SINIR.DEG. SAPMA	AP İLE % SAPMA	FİYATLAR
HAM.PROTEİN	223.378	160	-	63.378(p_1)	0.3961	
HAM SELÜLOZ	144.740	-	150	5.260(N_2)	0,035	
HAM KÜL	68.409	-	90	21.591(N_3)	0,2399	
FOSFOR	6.113	6	-	0,113(p_4)	0,022	
KALSİYUM	7	7	-	0	-	-
AĞIRLIK	984	-	984	0	-	-
FİYAT	45.000	-	45.000	0	-	-
ARPA	0,000	-	-	-	-	54,02
AY.KÜSPESİ	661,027	-	-	-	-	51,46
KEPEK	0,000	-	-	-	-	41,38
RAZMOL	0,000	-	-	-	-	37,59
BAYAT EKMEK	313,371	-	-	-	-	34,97
KAL.KARBONAT	9,602	-	-	-	-	2,6

Yeme 16 kg.lık vitamin önkarışımı ilave edileceğinden modele 984 kg yem elde edilecek şekilde sınırlama getirilmek şartıyla bir ton yem hazırlanması düşünülmüştür. Bu yüzden vitamin ve izmineral miktarları çözüm sırasında dikkate alınmamıştır (Diğer karmalar hazırlanırkende bu sınırlama göz önüne alınmıştır).

5.2. GENELLEŞTİRİLMİŞ TERS İLE SÜT YEMİNİN HAZIRLANMASI

(5.1.1) hedefleri eşitliğe dönüştürülürse

$$A=\begin{bmatrix} 0,10 & 0,30 & 0,13 & 0,12 & 0,08 & 0 & -1 & 0 & 0 & 0 \\ 0,07 & 0,20 & 0,11 & 0,09 & 0,04 & 0 & 0 & 1 & 0 & 0 \\ 0,04 & 0,07 & 0,06 & 0,025 & 0,04 & 1 & 0 & 0 & 1 & 0 \\ 0,004 & 0,0083 & 0,012 & 0,0025 & 0,002 & 0 & 0 & 0 & 0 & -1 \\ 0,0007 & 0,0044 & 0,001 & 0,0004 & 0,0008 & 0,4 & 0 & 0 & 0 & 0 \\ 1. & 1. & 1. & 1. & 1. & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

ve A rankı 6 olduğundan tam satır ranklı bir matristir. Süt yemi için çekirdek uzay matrisi (A^o) ise

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1.0002505 & -0,99098196 & -0,9994989 & +1.001002 \\ 0,0002505 & -0,009018 & -0,000501 & 0,001002 \\ 0,01998 & 0,2207215 & 0,0500401 & 0,03991984 \\ -0,02999 & -0,1603608 & -0,0700201 & -0,04995992 \\ -0,0002405 & -0,0213427 & -0,0195191 & 0,014038 \\ 0,001995 & 0,006318 & 0,01001 & 0,000498 \end{bmatrix}$$

olacaktır.

$A \; A^o = 0$ olduğu kolaylıkla görülebilir.

(5.1.1) probleminin GT ile çözümü

$$\bar{\bar{X}}_o = A^+ \quad \bar{b} \quad + \quad A^o \quad \bar{\bar{z}}$$

olduğuna göre,

$$X_o = \begin{bmatrix} 191,811 \\ 200,409 \\ 194,778 \\ 192,348 \\ 189,920 \\ 14,026 \\ -16,938 \\ 50,045 \\ 30,274 \\ -0,367 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1,0002505 & -0,99098196 & -0,9994989 & -1,001002 \\ 0,0002505 & -0,009018 & -0,000501 & 0,001002 \\ 0,01998 & 0,2207215 & 0,0500401 & 0,0399198 \\ -0,02999 & -0,1603607 & -0,0700201 & -0,0499599 \\ -0,0002405 & -0,0213427 & -0,0195191 & 0,014038 \\ 0,0019995 & 0,006318 & 0,01001 & 0,000498 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}$$

$x_i \leqslant 0$ ($i = 7, 8, 9, 10$) olması, gereksinimin karşılanamadığını ifade ettiğine göre çekirdek uzay yardımıyla $x_i \geqslant 0$ ($i = 7, 8, 9, 10$) durumuna getirmek mümkündür.

$z_1 = 0$, $z_2 = 76,739$, $z_3 = 0$ $z_4 = 0$ alınırsa (İkinci öncelikteki maliyet hedefi de göz önüne alınarak),

$$X_o = [\,191,811 \quad 277,148 \quad 194,778 \quad 192,848 \quad 113,873 \quad 13,334 \quad 0 \quad 37,739 \quad 28,686 \quad 0,1178\,]^T$$

(FİYATI: 43.949 TL)

çözümü elde edilecektir. Bu çözüm $x_i \geqslant 0$ ($i = 1,2,\ldots,10$) olduğundan gereksinimleri karşılayan çözümdür. Son dört çözüm elemani ise hedeften sapma miktarlarıdır. Bu sapma miktarları Amaç proğramlaması ile elde edilen sapma miktarları (% sapmalar) ile karşılaştırıldığında aşağıdaki tablo oluşacaktır.

Tablo 5.1.2: Süt Yemi İçin AP ile GT nin Karşılaştırılması

BESİNLER	AP ile elde edilen çözüm düzeyinin hedeften sapması (%)	Genelleştirilmiş Ters ile elde edilen çözüm düzeyinin hedeften sapması (%)
HAM PRO.	0,3961	0
HAM SELÜ.	0,0350	0,2515
HAM KÜL	0,2399	0,3181
FOSFOR	0,0220	0,0189
KALSİYUM	0	0
TOPLAM SAPMA	0,6930	0,5885

Tablodan da görüldüğü gibi GT ile elde edilen çözümün verdiği sapma, AP nin verdiği sapmadan oldukça küçük kalmaktadır. AP karmayı 45.000 TL ye mal ederken GT karmayı 43.949 TL ye mal etmektedir. GT nin bir diğer avantajı, $Z_i (i = 1, 2, 3, 4)$ değerleri uygun seçilerek yeni karma oluşturulabilir. Oysa AP ile bunu yapmak mümkün değildir. Demekki eşit sayıda aynı yönlü kısıt var olduğunda da en iyi çözümü veren yöntem GT dir.

5.2. BESİ YEMİNİN AP İLE HAZIRLANMASI

Süt yeminde kullanılan maddeler bu yemde de kullanıldığından ve sadece sınır değerleri ve yönü değişeceğinden, bu yem için problemin modelini vermeyeceğiz. Sınırlamanın değerleri ve yönü Tablo 5.2 de görülmektedir.

$\bar{a} = \{n_1+n_2+p_3+n_4+p_4+p_5+n_5+p_6,\ p_7\}$ ifadesinin (5.1.1) kısıtlarına (sağ taraf sabitleri dışında) göre AP ile sıralı minimum çözümü

BESİ Y. HEDEF NO:	2 ā VEK BİLEŞENİ	6 X ÇÖZÜMÜ	7 P POZİTİF SAPMA	7 N NEGATİF SAPMA
1	0.000	0.000	75.976	0.000
2	0.000	675.523	6.764	0.000
3		0.000	0.000	14.068
4		0.000	.190	0.000
5		291.491	0.000	0.000
6		16.986	0.000	0.000
7		16.98	0.000	0.000

olacaktır. Bu çözüm için 1.nci ve 2.nci öncelik seviyelerine ulaştığı söylenebilir. Karmada altı maddeden üçü bulunmaktadır. Bu çözümün nitelikleri Tablo 5.2.1 de özetlenmiştir.

Tablo 5.2.1: AP ile Elde Edilen Eniyi Besi Yeminin Nitelikleri

BESİNLER	En İyi Çözüme Giren Miktar	Alt Sınır	Üst Sınır	Sınır Değerinden Sapma Miktarı	Fiyat
HAM PROTEİN	225,976	150	-	75,976 (P_1)	
HAM SELÜLOZ	146,764	140	-	6,764 (p_2)	
HAM KÜL	75,932	-	90	14,068 (N_3)	
FOSFOR	6,190	6	-	0,190 (p_4)	
KALSİYUM	10,000	10	-	0,000	
AĞIRLIK	984		984	0,000	
FİYAT	45,000	-	45000	0,000	
ARPA	0,000	-	-	-	54,02
AY KÜSPESİ	675,523	-	-	-	51,46
KEPEK	0,000	-	-	-	41,38
RAZMOL	0,000	-	-	-	37,59
B.EKMEK	291,491	-	-	-	34,97
KAL.KARBONAT	16,986	-	-	-	2,6

BESİ YEMİNİN GT İLE HAZIRLANMASI

Bu yem için A ve A^o matrisleri

$$A = \begin{bmatrix} 0.10 & 0.30 & 0.13 & 0.12 & 0.08 & 0 & -1 & 0 & 0 & 0 \\ 0,07 & 0,20 & 0,11 & 0,09 & 0,04 & 0 & 0 & -1 & 0 & 0 \\ 0,04 & 0,07 & 0,06 & 0,025 & 0,04 & 1 & 0 & 0 & 1 & 0 \\ 0,004 & 0,0083 & 0,012 & 0,0025 & 0,002 & 0 & 0 & 0 & 0 & -1 \\ 0,0007 & 0,0044 & 0,001 & 0,0004 & 0,0008 & 0,4 & 0 & 0 & 0 & 0 \\ 1. & 1. & 1. & 1. & 1. & 1. & 0 & 0 & 0 & 0 \end{bmatrix}$$

(A nın rankı 6 dır)

$$A^o = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1,0002505 & -0,99098195 & -0,9994989 & -1,001002 \\ 0,0002505 & -0,009018 & -0,000501 & 0,001002 \\ 0,01998 & 0,2207215 & 0,0500401 & 0,03991984 \\ 0,02999 & 0,1603607 & 0,0700201 & 0,04995992 \\ -0,00024048 & -0,0213427 & -0,0195191 & 0,01403808 \\ 0,00199949 & 0,006318 & 0,01001 & 0,000498 \end{bmatrix}$$

olacaktır.

$$\bar{X}_o = A^+ \bar{b} + A^o \bar{Z}$$

olacağından

$$X_o = \begin{bmatrix} 190,998 \\ 197,001 \\ 193,224 \\ 191,784 \\ 189,549 \\ 21,445 \\ -8,502 \\ -41,132 \\ 23,155 \\ -0,423 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1,0002505 & -0,99098196 & -0,9994989 & -1,001002 \\ 0,0002505 & -0,009018 & -0,000501 & 0,001002 \\ 0,01998 & 0,2207215 & 0,0500401 & 0,03991984 \\ 0,02999 & 0,1603607 & 0,0700201 & 0,04995992 \\ -0,00024048 & -0,0213427 & -0,0195191 & 0,01403808 \\ 0,00199949 & 0,006318 & 0,0100001 & 0,000498 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}$$

$z_1 = 0$, $z_2 = 191,2739$, $z_3 = 0$, $z_4 = 0$ alınırsa.

$$X_o = [190,998 \quad 388,275 \quad 193,224 \quad 191,784 \quad 0 \quad 19,720 \quad 33,716 \quad -10,459 \quad 19,072 \quad 0,785]^T.$$

$x_8 \leqslant 0$ olması ham selüloz gereksiniminin karşılanmadığını ifade etmektedir. Bu çözüm AP nin çözümü ile karşılaştırıldığında yine AP den daha az sapma içerecektir. Bu Tablo 5.2.2.de görülmektedir.

Tablo 5.2.2: Besi Yemi İçin AP ile GT nin Karşılaştırılması

BESİNLER	AP Sapması	GT Sapması
H.Protein	0,5065	0,2247
H.Selüloz	0,0483	0,0747
H.Kül	0,1563	0,2119
Fosfor	0,0316	0,1308
Toplam Sapma	0,7427	0,6421

Her ne kadar GT nin sapması az sa da ham selüloz gereksinimini karşılamadığı için AP burada daha üstün dür.

5.3. BUZAĞI-KUZU YEMİNİN AP İLE HAZIRLANMASI

$$
\begin{aligned}
0,1\ X_1 + 0,3\ X_2 + 0,13\ X_3 + 0,32\ X_4 &\geqslant 170 \\
0,07X_1 + 0,2\ X_2 + 0,11\ X_3 + 0,16\ X_4 &\leqslant 110 \\
0,04X_1 + 0,07X_2 + 0,06\ X_3 + 0,07\ X_4 &\leqslant 100 \\
0,0007X_1 + 0,004X_2 + 0,001X_3 + 0,002X_4 &\geqslant 10 \qquad (5.3.1) \\
0,004X_1 + 0,01\ X_2 + 0,012X_3 + 0,012X_4 &\geqslant 6 \\
X_1 + X_2 + X_3 + X_4 &= 982 \\
54,02\ X_1 + 51,46\ X_2 + 41,38\ X_3 + 47,65\ X_4 &\leqslant 50.000
\end{aligned}
$$

kısıtlarına göre

$\bar{a} = \{n_1+p_2+p_3+n_4+n_5+n_6+p_6,\ p_7\}$ ifadesinin AP

ile sıralı minimum, çözümünü bulmak mümkün olmamıştır. Oysa bu problem GT ile kolaylıkla çözülebilir.

BUZAĞI - KUZU YEMİNİN GT İLE HAZIRLANMASI

(5.3.1) hedefleri eşitlik haline getirilirse bu yeme ait A matrisi,

$$
\begin{bmatrix}
0,1 & 0,3 & 0,13 & 0,32 & -1 & 0 & 0 & 0 & 0 \\
0,07 & 0,2 & 0,11 & 0,16 & 0 & 1 & 0 & 0 & 0 \\
0,04 & 0,07 & 0,06 & 0,07 & 0 & 0 & 1 & 0 & 0 \\
0,0007 & 0,004 & 0,001 & 0,002 & 0 & 0 & 0 & -1 & 0 \\
0,004 & 0,01 & 0,012 & 0,012 & 0 & 0 & 0 & 0 & -1 \\
1. & 1. & 1. & 1. & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

ve çekirdek uzayı matrisi

$$A^{o} = \begin{bmatrix} -1 & -1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0,2 & 0,03 & ,22 \\ -0,13 & -0,04 & -0,09 \\ -0,03 & -0,02 & -0,03 \\ 0,0033 & 0,0003 & 0,0013 \\ 0,006 & 0,008 & 0,008 \end{bmatrix}$$

olacaktır.

GT ile (5.3.1) Sisteminin Çözümü

$X_o = A^{+} b \qquad + \qquad A^{o}$ Z olacağından

$$X_o = \begin{bmatrix} 250.2572 \\ 241,2562 \\ 249,0997 \\ 241,3869 \\ 37,029 \\ -21,792 \\ 41,258 \\ -8,127 \\ 3,299 \end{bmatrix} + \begin{bmatrix} -1 & -1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0,20 & 0,03 & ,22 \\ -0,13 & -0,04 & -0,09 \\ -0,03 & -0,02 & -0,03 \\ 0,003 & ,0003 & ,0013 \\ ,006 & ,008 & ,008 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}$$

olacaktır.

$Z_1 = -167,6307,\quad Z_2 = 0\quad Z_3 = 0$ alınırsa

$X_o = [\,417,888\quad 73,6254\quad 249,0997\quad 241,3869\quad 3,5028\quad 0\quad 46,2869\quad -8,6881\quad 2,2932]^T$

$X_8 \leqslant 0$ almasıbu yemin kalsiyum bakımından yetersiz olduğunu göstermektedir, bu yüzden yeme oldukça ucuz olan kalsiyum karbonat veya kireç taşı ilavesi yapılabilir.

Tablo 5.3.1: GT İle Elde Edilen En İyi Buzağı-Kuzu Yeminin Nitelikleri

	En İyi Çözüm-Deki Değer	Alt Sınır	Üst Sınır	Hedeften Sapma	Fiyat
Protein	173,5028	170	-	3,5028	
Selüloz	110,	-	110	0	
Kül	53,7131	-	100	46,2869	
Kalsiyum	1,3199	10	-	8,6801	
Fosfor	8,2932	6	-	-	
Ağırlık	982	982	982	-	
Arpa	417,2869	-	-	-	54,02
Ay.Küspesi	73,6254	-	-	-	51,46
Kepek	249,0997	-	-	-	41,38
P.Küspesi	241,3869	-	-	-	47,65
Toplam					48172

Bu karma yem kalsiyum bakımından gereksinimi karşılamadığından vitamin önkarışımıyla beraber oldukça ucuz olan mermer tozu veya kalsiyum karbonat ilavesi düşünülmelidir.

5.4. AP İLE ETLİK PİLİÇ YEMİNİN HAZIRLANMASI

$$
\begin{array}{l}
0{,}30\, X_1 + 0{,}40\, X_2 + 0{,}42\, X_3 + 0{,}60\, X_4 + 0{,}09\, X_6 + 0{,}09\, X_7 + 0{,}44\, X_8 \geqslant 200 \\
0{,}20\, X_1 + 0{,}07\, X_2 + 0{,}07\, X_3 + 0\, X_4 + 0{,}03\, X_6 + 0\, X_8 \leqslant 75 \\
0{,}07\, X_1 + 0{,}06\, X_2 + 0{,}06\, X_3 + 0{,}11\, X_4 + X_5 + 0{,}02\, X_6 + 0{,}095\, X_7 + 0{,}22\, X_8 + 0{,}55\, X_9 \leqslant 80 \\
0{,}0044 X_1 + 0{,}0002 X_2 + 0{,}0015 X_3 + 0{,}045\, X_4 + 0{,}22\, X_5 + 0{,}0002 X_6 + 0{,}0010 X_7 + 0{,}14\, X_8 + 0{,}33\, X_9 \leqslant 12 \\
0{,}0044 X_1 + 0{,}0002 X_2 + 0{,}0015 X_3 + 0{,}045\, X_4 + 0{,}22\, X_5 + 0{,}0002 X_6 + 0{,}0010 X_7 + 0{,}14\, X_8 + 0{,}33\, X_9 \geqslant 6 \\
0{,}0083 X_1 + 0{,}006\, X_2 + 0{,}0065 X_3 + 0{,}025\, X_4 + 0{,}18\, X_5 + 0{,}003\, X_6 + 0{,}0002 X_7 + 0{,}06\, X_8 \geqslant 6
\end{array}
$$

$$X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8 + X_9 = 979$$

$$1.9\, X_1 + 2.65\, X_2 + 2.3\, X_3 + 2.9\, X_4 + 3.37\, X_6 + 2\, X_7 + 2.4\, X_8 \geqslant 2950$$

$$51{,}46X_1 + 52{,}33X_2 + 73{,}19X_3 + 219{,}69X_4 + 170{,}44X_5 + 60{,}96X_6 + 26X_7 + 154{,}35X_8 + 2{,}4X_9 = 60.000$$

kısıtlarına göre

$$\bar{a} = \{3n_1 + p_2 + p_3 + p_4 + 1.5n_5 + 2n_6 + p_7,\ p_8,\ 10\, p_9\}$$

ifadesini sıralı minimum yapan çözüm

Hedef No:	3 A Vektörünün Bileşeni	9 X Çözümü	9 P Pozitif Sapma	9 N Negatif Sapma
1	0,0000	17,3758	0,0000	0,0000
2	0,0000	304,9561	0,0000	32,3156
3	0,0000	0,0000	0,0000	23,2256
4		6,7026	0,0000	0,0000
5		0,0000	5,9999	0,0000
6		595,4080	0,0000	0,0000
7		0,0000	0,0000	0,0000
8		34,5376	0,0000	0,0000
9		20,0200	0,0000	0,0000

olacaktır. Bu karma hazırlanırken, 1.ci, 2.nci, 3.cü, Tablo 5.4.1: AP ile Elde Edilen En İyi Etlik Piliç Yeminin Nitelikleri

	En İyi Çöz. Miktarı	Alt Sınır	Üst Sınır	Sınır Değ. Sapma Mik.	Fiyat
Ham Protein	200	200	-	0	-
H.Selüloz	42,68	-	75	32,315(N)	-
H.Kül	57,37	-	80	22,62(N)	-
Kalsiyum	12	-	12	0	-
Kalsiyum		6	-	6 (p)	-
Fosfor	6	6	-	0	-
M.E.	2950	2950	-	0	-
Ağırlık	979	979	979	-	-
Ayçiçeği-Küspesi	17,3558	-	-	-	51,46
Fındık Küspesi	304,9561	-	-	-	52,33
Soya Küspesi	0,000	-	-	-	73,19
Balık Unu	6,7026	-	-	-	219,69
Kalsiyum Fosfat	0,000	-	-	-	170,44
Mısır	595,4080	-	-	-	60,96
Pancar Melas	0,000	-	-	-	26,00
Et-Kemik Unu	34,5376	-	-	-	154,35
Mermer Tozu	20,0200	-	-	-	2,40
Fiyat					60,000

4.cü, 5.ci, 6.ncı ve 7.nci hedeflere birinci öncelik (Fiziksel kısıtlar), bunlarla aynı birimle ölçülemeyen metabolik enerjiye 2.nci öncelik, ve maliyete üçüncü öncelik verilerek çözüme ulaşılmıştır. Dikkat edildiği gibi her üç öncelik düzeyinin seviyesine ulaşılmış ve bu yem gereksinimi karşılayan ucuz bir karma niteliği taşımaktadır.

5.5. PİLİÇ BÜYÜTME YEMİNİN AP İLE HAZIRLANMASI

$$\begin{array}{l}
0.10\,X_1 + 0{,}13\,X_2 + 0{,}32\,X_3 + 0{,}40\,X_4 + 0{,}60\,X_5 + 0\,X_6 + 0{,}09\,X_7 + 0{,}09\,X_8 + 0\,X_9 \geqslant 160 \\
0{,}07\,X_1 + 0{,}11\,X_2 + 0{,}16\,X_3 + 0{,}07\,X_4 + 0\,X_5 + 0\,X_6 + 0{,}03\,X_7 + 0\,X_8 + 0 \cdot X_9 \leqslant 80 \\
0{,}04\,X_1 + 0{,}06\,X_2 + 0{,}07\,X_3 + 0{,}06\,X_4 + 0{,}11\,X_5 + 1\,X_6 + 0{,}02\,X_7 + 0{,}095\,X_8 + 0{,}85X_9 \leqslant 80 \\
0{,}0007X_1 + 0{,}001\,X_2 + 0{,}002\,X_3 + 0{,}0002X_4 + 0{,}045\,X_5 + 0{,}25\,X_6 + 0{,}0002X_7 + 0{,}001\,X_8 + 0{,}33X_9 \geqslant 6 \\
0{,}0007X_1 + 0{,}001\,X_2 + 0{,}002\,X_3 + 0{,}0002X_4 + 0{,}045\,X_5 + 0{,}25\,X_6 + 0{,}0002X_7 + 0{,}001\,X_8 + 0{,}33X_9 \leqslant 15 \\
0{,}004\,X_1 + 0{,}012\,X_2 + 0{,}012\,X_3 + 0{,}006\,X_4 + 0{,}025\,X_5 + 0{,}18\,X_6 + 0{,}003\,X_7 + 0{,}0002X_8 + 0\,X_9 \geqslant 6
\end{array}$$

$$X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8 + X_9 = 982{,}5$$

$$2.76X_1 + 1{,}9X_2 + 2{,}6X_3 + 2{,}65X_4 + 2{,}9X_5 + 3{,}37X_7 + 2 \cdot X_8 \geqslant 2700$$

$$54{,}02X_1 + 41{,}38X_2 + 47{,}65X_3 + 52{,}33X_4 + 219{,}69X_5 + 26X_6 + 170{,}44X_7 + 60{,}96X_8 + 2{,}4X_9 = 68000$$

kısıtlarına göre

$$\bar{a} = \{1{,}05n_1 + p_2 + p_3 + n_4 + p_5 + 1{,}15n_6 + 1{,}5p_7 + n_7,\ 1.25n_8,\ p_9\}$$

ifadesini minimum yapan çözüm

Piliç Büyütme Y. Hedef No	3 a: Başarı Vektörü Bileşeni	9 X Çözümü	9 P Pozitif Sapması	9 N Negatif Sapması
1	0,000	0,000	0,000	0,000
2	0,000	0,000	0,000	66,483
3	0,000	0,000	0,000	17,987
4		2,774	,726	0,000
5		138,657	0,000	8,274
6		0,000	30,093	0,000
7		444,092	0,000	0,000
8		396,978	0,000	0,000
9		0,000	0,000	0,000

olacaktır. Bu karma hazırlanırken, 1.nci, 2.nci, 3.cü, 4.ncü, 5.nci, 6.nci ve 7.nci hedefler birinci öncelikle, metabolik enerji hedefi olan 8.nci hedefe 2.nci öncelikte, maliyet ise üçüncü öncelikte düşünülmüştür. Bu karma ucuz ve gereksinimi karşılayan bir yem türüdür.

Tablo 5.5.1: AP İle Elde Edilen En İyi Piliç Büyütme Yeminin Nitelikleri

	Optimal Çözüm Miktarı	Alt Sınır	Üst Sınır	Sınır Değeri Sapması	FİYAT
Ham Protein	160	160	-	0	
Ham Selüloz	13,517	-	80	66,483	
Ham Kül	62,013	-	80	17,987	
Kalsiyum	6,72	-	15	8,25	
Kalsiyum	6,72	6	-	0,72	
Fosfor	36,093	6	-	30,	
M.Enerji	2700	2700	-	0	
Fiyat	68,000	68,000	-	0	
Ağırlık	982,5	982,5	982,5	0	
Arpa	0,000	-	-	-	54,02
Kepek	0,000	-	-	-	41,38
P.T.Küspesi	0,000	-	-	-	47,65
Fındık	2,774	-	-	-	52,33
Balık Unu	138,657	-	-	-	219,69
D.C.P.	0,000	-	-	-	26
Mısır	444,092	-	-	-	170,44
Melas	396,978	-	-	-	60,96
Kireç Taşı	0,000	-	-	-	2,4

Tablo 5.5.2: Yem Sanayii Türk A.Ş. Ankara ve Çankırı Fabrikalarınca Uygulanan Karma Yem Satış Fiyatları (TL/kg) ve Hazırladığımız Karma Yem Maliyetleri ile Karşılaştırılması

KARMA YEM CİNSİ	15-11-1986 FİYATI	AP İLE HAZIRLANAN KARMA YEMİN MALİYETİ	GT İLE HAZIRLANAN KARMA YEMİN MALİYETİ
Süt Yemi	60,000	45,000	43,949
Besi Yemi	60,000	45,000	45,553
Buzağı-Kuzu Yemi	61,000	-	48,172
Etlik Piliç Yemi	100,000	60,000	-
Piliç Büyütme Yemi	95,000	68,000	-

SONUÇ

Amaç programlaması şimdiye kadar beslenme problemlerinde etkin olarak yaygın bir şekilde kullanılmıştır. Ancak bu yöntemin de bazı durumlarda bir takım komplikasyonlar yaratacağı göz önünde tutulmalıdır. Örneğin, pivot seçiminde iki veya daha fazla değişken aynı minimum oranı veriyorsa çözüme ulaşmak güçleşeceğinden bu yöntem gibi hedefler arasındaki sapma kavramının enküçüklenmesine dayanan genelleştirilmiş ters yaklaşımı daha uygun çözümler sağlamaktadır.

Tablo 5.1.2, Tablo 5.2.2 ve Tablo 5.5.1 göz önüne alınırsa GT süt yemi probleminde çok büyük avantaj sağlamıştır. Hem besin değeri yüksek hem de daha ucuz karma hazırlanmasını sağlamıştır. AP ise besi yeminde GT'ye üstünlük sağlamıştır. İki yöntem Buzağı-Kuzu Yemi probleminin çözümünde de karşılaştırılmak istenmiştir. AP ile bu problem çözülemezken GT ile bu problem kolaylıkla çözülmüştür. Ayrıca GT karar vericiye model üzerinde birtakım serbestlikler de vermiştir.

Tablo 5.1.2 göz önüne alınırsa AP ile hazırlanan karmada ham protein hedefinin sapmasının oldukça büyük olduğu görülmektedir. Oysa bu Karma GT ile hazırlandığında bu hedefe tam olarak erişildiği görülür. Yani sapma sıfır düzeyindedir. Ham kül sapması her iki yöntemin hazırladığı karmada biri birine yakındır. Sadece Ham selüloz sapması GT karmasında büyük kalmaktadır. AP karması hedeflerden toplam olarak % 69 luk, GT karması ise % 59 luk sapma göstermiştir. Bu yüzden GT karması besin değeri bakımından daha yüksektir, aynı karma AP karmasından maliyet bakımından da daha ucuzdur.

Tablo 5.2.2. de de karşılaştırıldıklarında GT yem karması daha az sapma gösterirken ham selüloz gereksinimi karşılamadığı görülmüştür. Bu yüzden AP yem karması, sapması büyük olmasına rağmen standartlara uygundur.

Buzağı Kuzu yeminde AP ile karma yem hazırlamak mümkün olmadığından GT yem karması ile karşılaştırma imkanı elde edilememiştir, ne var ki GT yem karması kalsiyum bakımından standartları sağlıyamamıştır. Bu yüzden bu karmaya vitamin ve öz mineral karışımıyla beraber oldukça ucuz olan kalsiyum karbonat veya kireç taşı ilavesi düşünülmelidir.

Ayrıca AP karmaları tek çözümden oluşmasına karşın GT karmaları birden fazla çözüm sağladıklarından karar verici için kullanılması daha uygundur.

Etlik piliç yemi karması ve piliç büyütme yemi karması hazırlanırken sadece AP yöntemi kullanılmıştır. Hem besin değeri bakımından hem de ucuz maliyet bakımından bu iki karma yemi hazırlamada AP yöntemi, varsayımlarımız altında standartlara uygun tatmin edici sonuçlar vermiştir.

KAYNAKLAR

1. Yılmaz,M.R.,"Çok Ölçütlü Karar Verme Yöntemlerine Eleştirisel Bir Bakış", Yöneylem Araştırması, Bildiriler (1978)

2. Belenson, S.M. and Kapur,K.C., "An Algorithm For Solving Multicriterion Linear Programming Problems with Examples", Opl. Res. Q, 24, (1974)

3. Iserman,H., "The Enumeration of The Set of all Efficient Solutions for A Linear Multiple Objective Program", Opl. Res. Q., 28, 3 (1977)

4. Hwang,C.L. et al., "Mathematical Programming With Multiple Objectives: A Tutorial", Comput. and Opr. Res., 7, 1 (1980)

5. Evans, G.W., "An Overview of Techniques to Solving Multiobjective Mathematical Programs",Mng. Sci., 30,11 (1984)

6. Daellenbach,H.G., George,J.A. and McNickle,D.C., Introduction to Operations Research Techniques, Second Edition, Allyn and Bacon Inc. Boston (1983)

7. Ignızıo, J.P., "Generalized Goal Programming", Comput. and Opr. Res., 10, 4 (1983)

8. Ignızıo,J.P., "Integer Goal Programming via Goal Aggregation",Large Scale Syst, 8 (1985)

9. Ignızıo, J.P., "An Approach to Certain Large Scale Multiobjective Integer Programming", Large Scale Syst., 4, 2 (1983)

10. Levin,R.I., Kırkpatrick,C.A., and Rubin,D.S., Quantitative Approaches to Management, Fifth Edition, McGraw-Hill Book Co.,New York (1982)

11. Ignızıo,J.P., "An Algorithm for Soving the Linear Goal Programming Problem by Solving its Dual", J. Opl. Res. Soc., 36, 6 (1985)

12. Charnes,A., and Cooper,W.W., Management Models and Industrial Application of Linear Programming, Vol I, Vol II, Wiley, New York (1961)

13. Ijırı, Y., Management Goals and Accounting for Control, Northolland Publishing Co., Amsterdam (1965)

14. Lee, S.M., Goal Programming for Decision Analysis, Auerbach, Philadelphia, Pa (1972)

15. Dyer,J.S., "Interactive Goal Programming", Mng. Sci., 19,1 (1972)

16. Ignızıo,J.P., Goal Programming and Extensions, D.C. Heat and Company Lexington, Massachusetts (1976)

17. Contini,B., "A Stochastic Approach to Goal Programming", Ops. Res., 16 (1968)

18. Zanakis,S.H., and Maret,M.W., "A Markovian Goal Programming to Aggregate Manpower Planning", J.Opl. Res.Soc., 32 (1981)

19. Lin, W.T., "A Survey of Goal Programming Applications", Omega, 8 (1980)

20. Anderson,A.M., and Earle,M.D., "Diet Planning in The Third World by Linear and Goal Programming", J.Opl. Res.Soc., 34,1 (1983)

21. Lee, S.M. and Clayton,E.R., "A Goal Programming Model for Academic Resource Allocation", Mng.Sci., 18, 8 (1972)

22. Countney,J.F., Klastorin,T.D., and Ruesli,T.W., "A Goal Programming Approach to Urban-Suburban Location Preferences",Mng.Sci., 18, 6 (1972)

23. Masud,A.S. and Hwang,C.L., "Interactive Sequential Goal Programming", J. Opl. Res. Soc., 32, 5, 1981

24. Ignızıo,J.P., "The Determination of A Subset of Efficient Solution via Goal Programming", Comput. and Opr. Res., 3, 1 (1981)

25. Ignızıo,J.P., "A Review of Goal Programming:A Tool for Multiobjective Analysis", J.Opl. Res. Soc. 29, 11 (1978)

26. Ignızıo, J.P., "An Approach to Modelling and Analysis of Multiobjective Generalized Networks.", Eue.J.Opl. Res., 12, 4(1983)

27. Sutcliffe, C., Board,I., and Cheshise,P., "Goal Programming and Allocating Children to Secondary Schools in Reading", J. Opl. Res. Soc., 35, 8 (1984)

28. Olson,D.L. "Comparison of Four Goal Programming Algorithms", J. Opl. Res. Soc., 35, 4 (1984)

29. Dauer,J.P., and Kruger,R.S. "An Iterative Approach to Goal Programming",Opl. Res. Q., 28, 3 (1977)

30. Ignızıo,J.P., and Perlis,J.H., "Sequential Linear Goal Programming Implementation Via MPSX",Comput. and Opr. Res., 6 (1979)

31. Arthur,J.L., and Ravindran,A., "PAGP: A Partitioning Algorithm for (Linear) Goal Programming Problems", ACM Trans. Math. Software, 6, 3, (1980)

32. Arthur, J.L., and Ravindran, A., "An Efficient Goal Programming Algorithm Using Constraint Partitioning and Variable Elimination", Mng. Sci., 24,8,(1978)

33. Schneiderjans, M.S., and Kwak, N.K., "An Alternative Solution Method for Goal Programming Problems: A Tutorial",J. Opl. Res. Soc., 33 (1982)

34. Markowski,C.A. and Ignızıo,J.P., "Theory and Properties of the Lexicographic Linear Goal Programming",Large Scale Syst.,5, 2 (1983)

35. Lee,S.M. and Moore, L.J., Introduction to Decision Sciences, Petrocell, Charter, New York (1975)

36. Sengupta,S., "Goal Programming to A Type of Quality Control Problem", J. Opl. Res. Soc., 32 (1981)

37. Markowski, C.A. and Ignızıo, J.P., "Duality and Transformations in Multiphase and Sequential Linear Goal Programming", Comput. and Opr. Res., 10, 4 (1983)

38. Kara,I., Yöneylem Araştırması, E.İ.T.İ.A., Eskişehir (1980)

39. Esin,A., Yöneylem Araştırmasında Yararlanılan Karar Yöntemleri. A.İ.T.İ.A Yayını, Ankara (1981)

40. Avralıoğlu, Z., Doğrusal Programlama ve Tarımsal İşletmelerde bir Uygulama, A.İ.T.İ.A Yayını, Ankara, (1980)

41. Gass, T., Linear Programming, Fifth Edition, Mc-Graw-Hill Book Co., New York (1985)

42. Jack,W., and Buchanan,J.T., "An Alternative to Optimization: Two Methods for Linear Systems", J. Opl. Res. Soc., 36, 5 (1985)

43. Romero,C., "Multiobjective and Goal Programming Approaches as A Distance Function Model", J.Opl.Res. Soc., 36, 3 (1985)

44. Ben-Israel A., and Greville, T.N.E., Generalized Inverses: Theory and Applications, Wiley, New York (1974)

45. Campbel,S.L. and Meyer,C.D., Generalized Inverses of Linear Transformation, Pitman Pub. Lim., London, (1979)

46. Hacısalihoğlu, H.H., Lineer Cebir, İkinci Baskı, F.Ü. Fen Fak., Ankara (1982)

47. Strang, G., Linear Algebra and Its Applications, Second Edition, Academic Press, Inc., New York (1980)

48. Lang,S., Linear Algebra, Second Edition, Addison-Wesley Pub. Co., New York (1972)

49. Lipschutz,S., Theory and Problems of Linear Algebra, Schaum's Outline Series, SI Edition, McGraw-Hill Book Co., New York (1968)

50. Remore,C., and Rehman,T.A., "A Note on Diet Planning in the Third World by Linear and Goal Programming", J.Opl. Res. Soc., 35, 6 (1984)

51. Çakır,C., Doğrusal Programlama Yöntemiyle En Düşük Maliyetli Karma Yem formüllerinin Saptanması üzerine Bir Araştırma (Doçentlik Tezi), Bornova (1978)

52. Yem Kanunu, T.C. Resmi Gazete 7 Haziran 1973, s.14557

53. Yem Yönetmeliği, T.C. Resmi Gazete, 5 Ağustos 1974, s.14967

54. Akyıldız,R., Karma Yemler Endüstrisi, San Matbaası, Ankara, (1979)

55. Tebliğ, T.C. Resmi Gazete, 6-Kasım-1982, s. 17860

EKLER

```
C       BU PROMRAM ÇOK AŞAMALI SİMPLEKS YÖNTEMİ İLE BİRDEN FAZLA AMAÇLI
C       DOGRUSAL AMAC PROGRAMLAMASI PROBLEMLERİNİ ÇÖZMEYE YARAR.
C       PROGRAM MAXIMUM 10 KARAR DEĞİŞKENİ (SAPMA DEĞİŞKENİ DIŞINDA),
C       10 AMAC VE 5 ÖNCELİK DÜZEYİNE SAHİP PROBLEMLER İÇİN HAZIRLANMIŞTIR
Ç       DAHA BÜYÜK PROBLEMLER İÇİN BOYUTLAR UYGUN OLARAK BÜYÜTÜLEBİLİR.
        COMMON TL (10,10), TT (5,20),TE (20,20),TI (5,20),
        TB (10), TA (5), JÇOL (20,2), JROW (20,2),
        NOBJ,NPRI,NVAR,NCOL,NROW
        DATA IMAX/120/
        DATA  NPRB/0/
C       AMAÇ SAYISI, ÖNCELİK SAYISI VE DEĞİŞKEN SAYISININ OKUNMASI.
   25   READ ( 5, 30,END = 90)  NOBJ, NVAR , NPRI
C       WRİTE (1, 30) NOBJ,NPRI,NVAR
        NCOL = NOBJ + NVAR
C       SÜTUN BAŞLIKLARI 2 İLE  KARAR DEĞİŞKENİ, 3 İLE P SAPMA DEĞİŞKENİ
C       4 İLE N SAPMA DEĞİŞKENİ KODLANDI.
        DO 1 NV=1, NVAR
        JCOL (NV,1) = 2
    1   JCOL (NV,2) = NV
        DO 2 NO=1, NOBJ
        NC=NO+NVAR
        JCOL (NC, 1) = 3
        JCOL (NC, 2) = NO
        JROW (NO, 1) = 4
    2   JROW (NO, 2) = NO
C       KATSAYILAR MATRISININ OKUNMASI
        DO 7 NO = 1, NOBJ
    7   READ (5,31) (TE (NO,NV), NV =1, NVAR)
C   7   WRITE (1,31) (TE (NO,NV), NV = 1, NVAR)
        DO 3 NOR = 1, NOBJ
        DO 3 NO = 1, NOBJ
        NOC = NO  +NVAR
        TE (NOR,NOC) = 0
        IF (NO. EQ. NOR) TE (NOR,NOC) = -1
    3   CONTİNUE
C       B VEKTORUNUN OKUNMASI
        READ (5,31) (TB (NO), NO =1, NOBJ)
        DO 6 NP = 1,NPRI
```

```
      DO 4 NO = 1, NOBJ
    4 TL (NO,NP) =0
      DO 5 NC =1, NCOL
    5 TT (NP,NC) =0
    6 CONTINUE
C     ÜST BLOK VE SOL BLOK MATRİSİNDEKİ SIFIRDAN FARKLI ELEMANLARIN
C     BELİRLENMESİ HER PROBLEM İÇİN AYRI AYRI BELİRLENMELİ
      TT (1,3)  = 3.
      TT (1,4)  = 4.
      TT (3,6)  = 1.
      TL (3,2)  = 1.
      NPRB=NPRB+1
      WRITE (1,34) NPRB
      NPRT=0
    8 NROW=0
      ITE =0
    9 IF (NROW.EQ.NPRI) GO TO 11
      NROW = NROW+1
C     OPTİMUMUN GÖSTERGESİ OLAN SATIRLARIN HESAPLANMASI.
      I= NROW
      DO 42 NP=1, NROW
      TA (NP)=0
      DO 18 NO=1,NOBJ
   18 TA (NP)=TA(NP)+TB(NO)*TL (NO,NP)
      DO 28 NC =1,NCOL
      TI (NP,NC)= -TT (NP,NO)
      DO 28 NO=1,NOBJ
      TI (NP,NC)= TI (NP,NC)+TE (NO,NC)*TL (NO,NP)
   28 CONTINUE
   42 CONTINUE
C     TEMELE GİRECEK DEĞİŞKEN İLE ÇIKACAK DEĞİŞKENİN BELİRLENMESİ,
   10 CALL  TEST (NEVC,NDVR)
      ITE=ITE+1
      IF (ITE.GE. IMAX) GO TO 19
```

```
      IF (NEVC.LE. 0) GO TO 9
C     YENİ TABLODAKİ ELEMANLARIN HESAPLANMASI
      CALL PERM (NEVC,NDVR)
      GO TO 10
C     EN İYİ ÇÖZÜMÜN ÇIKTISINI HESAPLAMA
   11 CALL POUT (NPRT,NCPL)
      IF (NPRT.NE. 0) GO TO 25
      GO TO 8
   19 WRITE (1,38) IMAX
      GO TO 25
   30 FORMAT (5I5)
   31 FORMAT (10F8.4/)
   34 FORMAT (1H1,////, YUMURTA YEMİ PROBLEMİ,14)
   38 FORMAT (//,** ALGORITMA ,I5, ITERASYONDA BİTMEDİ**)
   90 STOP
      END
TEMELE GİRECEK VE ÇIKACAK DEĞİŞKENİ BELİRLER.
      SUBROUTINE TEST (NEVC,NDVR)
      COMMON TL (10, 10), TT (5,20), TE (20,20), TI (5,20)
     *TB(10) ,TA(5), JCOL (20,2), JROW (20,2),
     *NOBJ, NPRI,NVAR, NROW
      VEVC=0
      NEVC=0
      L=NROW -1
      DO 3 NC=1,NCOL
      IF (TI(NROW,NC).LE.0) GO TO 3
      IF(NROW .EQ. 1) GO TO 2
      DO 1 N=1,L
      IF(TI(N,NC) .LT. 0) GO TO 3
    1 CONTINUE
    2 IF (TI(NROW,NC) .LE.VEVC) GO TO 3
      NEVC=NC
      VEVC=TI (NROW,NC)
    3 CONTINUE
      IF (NEVC .EQ. 0) RETURN
      NDVR=0
      DO 7 NR=1,NOBJ
```

```
      IF (TE(NR,NEVC) .LE.0) GO TO 7
      V=TB(NR)/TE(NR,NEVC)
      IF (NDVR .EQ. 0) GO TO 6
      IF (V-VDVR)  6,4,7
    4 DO 5 NP=1,NPRI
      IF(TL(NR,NP) -TL (NDVR,NP) ) 5,5,6
    5 CONTINUE
    6 VDVR= V
      NDVR=NR
    7 CONTİNUE
      IF (NDVR .GE. 1) RETURN
      WRITE (1, 8) P
    8 FORMAT (//,**PROGRAM DURDU**)
      RETURN
      END
C     BU ALTPROGRAM YENİ TABLO OLUŞTURARAK TABLODAKİ ELEMANLARI HE-
      SAPLAR
      SUBROUTINE PERM (NEVC,NDVR)
      COMMON TL (10, 10), TT (5,20), TE(20.20), TI(5,20)
     *TB(10), JCOL (20,2). JROW (20,2),
     *NOBJ, NPRI,NVAR,  NCOL, NROW
      DO 1 I=1,2
      J=JCOL(NEVC,1)
      JCOL(NEVC,I)=JROW (NDVR,I)
    1 JROW(NDVR,I)=J
      DO 2 NP=1,NPRI
      TEMP=TL(NDVR,NP)
      TL(NDVR,NP)=TT(NP,NEVC)
    2 TT(NP,NEVC)=TEMP
      PIV=TE(NDVR,NEVC)
      PIB=TB(NDVR)
      DO 31 NO=1,NOBJ
      IF(NO .EQ.NDVR ) G O TO 31
      PIX=TE(NO,NEVC)/PIV
      TB(ND)=TB(NO)-PIX*PIB
      DO 3 NC=1,NCOL
      IF(NC .EQ. NEVC) GO TO 3
```

```
      TE(NO,NC)=TE(NO,NC)-TE(NDVR,NC)*PIX
 3    CONTINUE
31    CONTINUE
      DO 4 NC=1,NCOL
 4    TE(NDVR,NC)=(TE(NDVR,NC)/PIV)
      DO 5 NO=1,NOBJ
 5    TE(NO, NEVC)=(-TE(NO,NEVC)/PIV)
      TB(NDVR)=(TB(NDVR)/PIV)
      TE(NDVR,NEVC)=(1/PIV)
      DO 53 NP=1,NROW
      TA(NP)=0
      DO 12 NO=1, NOBJ
12    TA(NP)=TA(NP)+TB(NO)*TL(NO,NP)
      DO 52NC=1,NCOL
      TI(NP,NC)=-TT(NP,NC)
      DO 52 NO=1,NOBJ
      TI(NP,NC)=TI(NP,NC)+TE(NO,NC)*TL(NO,NP)
52    CONTINUE
53    CONTINUE
      RETURN
      END
      SUBROUTINE POUT (NPRT,NCPL)
      DIMENSION WOUT (20,4)
      COMMON TL (10,10), TT(5,20), TE(20, 20), TI(5,20),TB(10),
     *TA(5),JCOL(20,2),JROW(20,2),
     *NOBJ, NPRI,NVAR,NCOL,NROW
      NPRT=NPRT+1
      DO 12 I=1,20
      DO 12 J=1,4
 12   WOUT (I,J)= 0
      DO 13 NP= 1,NPRI
 13   WOUT (NP,1)=(TA(NP))
      DO 14 NO=1,NOBJ
      IC=JROW(NO,1)
      IR=JROW(NO,2)
 14   WOUT(IR,IC)=(TB(NO))
 14   WOUT(IR,IC)=(TB(NO))
```

```
      WRİTE(1,33)NPRI,NVAR,NOBJ,NOBJ
      I=MAXO (NPRI,NVAR,NOBJ,)
      DO 20 K=1,I
      IF(K .GT. NPRI) GO TO 16
      IF(K .GT. NVAR) GO TO 15
      WRITE(1,34)K,(WOUT(K,J),J=1,4)
      GO TO 20
   15 WRITE(1,35)K,WOUT(K,1),(WOUT(K,J),J=3,4)
      GO TO 20
    6 IF(K .GT. NVAR) GO TO 17
      WRITE(1,36) K,(WOUT(K,J),J=2,4)
      GO TO 20
   17 WRITE(1,37) K,(WOUT(K,J),J=3,4)
   12 CONTINUE
   33 FORMAT(/, SATIRNA , I6,ASTAR SAYISI,I6,XSTAR SAYISI ,I6,
     *PSTAR SAYISI,I6,NSTAR SAYISI,I6/)
   34 FORMAT(I6,4F18.3)
   35 FORMAT(I6,F18.3,18X,2F18.3)
   36 FORMAT(I6,18X,3F18.3)
   37 FORMAT(I6X,2F18.3)
      RETURN
      END
```

SAYFA 31 DEKİ ÖRNEK 2,14,1 IN DATALARI VE ÇÖZÜMÜ

```
4  3  2
1.      1.
1.      0.
5.      3.
1.      1.
10.     4.      55.     12.
```

YUMURTA YEMİ PROBLEMİ 1

SATIRNO	3ASTAR SAYISI	2XSTAR SAYISI	4PSTAR SAYISI	4NSTAR SAYISI
1	0.0000	4.0000	0.0000	0.0000
2	17.0000	6.0000	0.0000	0.0000
3	0.0000		0.0000	17.0000
4			0.0000	2.0000

```
1 REM BU PROGRAM TAM SATIR RANKLI (MATRİS SUTUN RANKLI İSE TRANSPOZU
ALINACAK MATRİSLERİN GENELLEŞTİRİLMİŞ TERSİNİ BULMAYA YARAR
2 PRINT
3 PRINT
4 READ M,N
6 DIM A(M,M),B(M,M),C(M,N),CT(N,M),GA(N,M)
8 LPRINT "C MATRİSİ"
10 FOR I=1 TO M
15 FOR J=1 TO N
20 READ C(I,J)
22 LPRINT C(I,J)
25 NEXT J
35 NEXT I
37 REM MATRİSİN TRANSPOZUNU ALMA
40 FOR I=1 TO N
45 FOR J=1 TO M
50 CT(I,J)=C(J,I)
55 NEXT J
60 NEXT I
61 REM MATRISIN TRANSPOZU İLE CARPIMI
65 FOR I=1 TO M
70 FOR J=1 TO M
80 A(I,J) = 0
85 FOR K=1 TO N
90 A(I,J)=A(I,J)+C(I,K)*CT(K,J)
95 NEXT K
100 NEXT J
105 NEXT I
110 REM CARPIM MATRİSİNİN TERSİNİ BULMA
170 FOR I=1 TO M
180 B(I,I)=1
190 NEXT I
200 FOR J=1 TO M
210 T=A (J,J)
220 FOR K=1 TO M
230 B(J,K)=B(J,K)/T
240 A(J,K)=A(J,K)/T
```

```
250 NEXT K
260 FOR I=1 TO M
270 P=A(I,J)
280 IF I=J THEN GOTO 320
290 FOR K=1 TO M
300 B(I,K)=B(I,K)-P*B(J,K)
310 A(I,K)=A(I,K)-P*A(J,K)
315 NEXT K
320 NEXT I
330 NEXT J
450 FOR I=1 TO N
460 FOR J=1 TO M
470 GA(I,J)=0
480 FOR K=1 TO M
485 REM C MATRİSİ A MATRISI ILE SAĞDAN ÇARPIMI
490 GA(I,J)=GA(I,J)+CT(I,K)*B(K,J)
500 NEXT K
510 NEXT J
513 LPRINT
515 NEXT I
520 LPRINT "GA MATRISI"
530 FOR I=1 TO N
540 FOR J=1 TO M
550 LPRINT GA(I,J)
560 NEXT J
570 LPRINT
580 NEXT I
590 STOP
```

ÖZGEÇMİŞ

Hasan Bal 1954 yılında Elazığ ili Karakoçan ilçesi Karapınar köyünde doğdu. İlk, orta ve lise öğrenimini Karakoçan'da tamamladı. 1979 yılında Hacettepe Üniversitesi Matematik bölümünde mezun olduktan sonra, 1980 yılında A.İ.T.İ.A.'ya Asistan olarak girdi. 1982 yılında, Uygulamalı İstatistik master programını bitirdi. Halen G.Ü.Fen.Ed.Fak. İstatistik Bölümünde Araştırma görevlisi olarak çalışmaktadır. İstatistik ve Betimsel İstatistik gibi dersleri vermiştir.