

**T.C.
MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**BİR ÜRÜN GELİŞTİRME PROBLEMİNE
HEDEF PROGRAMLAMANIN UYGULANMASI**

YÜKSEK LİSANS TEZİ

Lale KARAYEĞİT

Matematik Anabilim Dalı

Matematik Programı

Tez Danışmanı: Yrd. Doç. Dr. Gülay İlonca TELSİZ KAYAOĞLU

ŞUBAT 2010

ÖNSÖZ

Bu çalışma süresince üzerimde emeđi olan, bilgi ve deneyimiyle bana her zaman destek olan tez danışmanım, deđerli arkadaşım ve hocam Yrd. Doç. Dr. Gülay İlonu TELSİZ KAYAOĐLU'na ve manevi desteđinden dolayı hocam Yrd. Doç. Dr. Özlem YILMAZ'a tüm kalbimle teşekkür ederim.

Lale KARAYEĐİT

Şubat 2010

İÇİNDEKİLER

ÖNSÖZ	i
ÖZET	iii
SUMMARY	iv
GİRİŞ	1
1. DOĞRUSAL PROGRAMLAMA	
1.1.Giriş	5
1.2.Doğrusal Programlamanın Tanımı	6
1.3.Doğrusal Programlamanın Yapısal Görünümü	7
1.4.Doğrusal Programlamanın Çözüm Yöntemleri	12
2.HEDEF PROGRAMLAMA	
2.1.Giriş	19
2.2.Hedef Programlamanın Gelişimi	20
2.3.Hedef Programlamanın Tanımı	22
2.4.Hedef Programlamanın Yapısı	23
2.5.Hedef Programlamanın Sınıflandırılması	25
2.6.Hedef Programlama Modeli	29
2.7.Doğrusal Hedef Programlama	33
3.UYGULAMA	
3.1.Giriş	40
3.2.Modelin Kurulması	47
3.3.Modelin Çözümü	51
SONUÇLAR VE ÖNERİLER	55
KAYNAKLAR	56
ÖZGEÇMİŞ	60

ÖZET

Bu çalışmada; bir ürünü, hedefler doğrultusunda geliştirmek için hangi hammaddeden ne kadar kullanılması gerektiği, öncelikli doğrusal hedef programlama yardımıyla bulunmuştur.

Birinci ve ikinci bölümlerde doğrusal programlamanın ve hedef programlamanın metodolojisi anlatılmıştır.

Üçüncü bölümde ise bir ürün geliştirme problemi sunulmuş ve bu problemin öncelikli doğrusal hedef programlama modeli kurulmuştur. Kurulan model, WINQSB paket programı kullanılarak çözülmüştür.

Anahtar Kelimeler: Hedef Programlama, Doğrusal Hedef Programlama, Ürün Geliştirme

SUMMARY

In this study; necessity to which amount of which raw material should be used to develop a product in the direction of the goals, is found by the help of pre-emptive linear goal programming.

In the first and second sections, the methodology of linear programming and goal programming are explained.

In the third section, a product development problem is introduced and the pre-emptive linear goal programming model of this problem is constructed. That constructed model was solved by using the package program which named WINQSB.

Keywords: Goal Programming, Linear Goal Programming, Product Development

GİRİŞ

İnsanlar yüzyıllar boyunca karşılaştıkları problemlere en iyi çözümü getirmek gibi bir ihtiyaçla karşılaşmışlardır. Matematikte de en iyi çözümün arandığı maksimum ve minimum problemlerinin incelenmesi çok eski zamanlara dayanmaktadır. Asırlar boyunca, karşılaşılan problemler kendilerine özgü yöntemlerle çözülmüş ve tüm problemlerin çözümüne yol gösterecek yaklaşımlar oluşmamıştır. Yaklaşık olarak üçyüz yıl önce, matematiksel analizin oluşumu döneminde, daha genel çözüm metodlarının ilk örnekleri ortaya konulmaya başlanmıştır [1].

Optimizasyon (en iyileme) kavramı, “bir probleme en iyi mümkün çözüm bulma süreci olarak” olarak tanımlanmaktadır [2]. Matematikte, bu süreç genellikle bir fonksiyonun değerinin verilen kısıtlar altında maksimize ya da minimize edilmesinden oluşur.

Yunan tarihçisi Herodotus'a göre, Mısırlılar Nil nehrinin her yıl taşması sonucu arazi sınırlarının yeniden belirlenmesi ve yeni sınırlara göre vergilendirme işleminin *en iyi* yolla yapılabilmesi için çaba sarfetmişlerdir. Bu çabalar, ölçme ve karar verme aracı olarak düzlem geometrisinin temel kavramlarının oluşturulmasına yol açmıştır [3]. Mısırlılar, Nil nehrinin bahar dönemlerindeki yıllık taşmalarında nehir kıyısından toplu halde uzaklaşıp sular çekildiğinde yine büyük topluluklar halinde geri dönüyorlardı. Çekilme işlemi çok kısa sürede yapılamamaktaydı. Bunun için günlerce önceden halk uyarılmalıydı. Bu amaçla, Mısırlılar *en iyi* çekilme zamanını hesaplayabilmek için bir tür takvim bile geliştirmişlerdi. Söz konusu takvimi de sayma ve geometri konusundaki birikimlerini kullanarak yapmışlardı [4].

Newton ve Leibniz tarafından Kalkülüs'ün (Calculus) 17. yüzyılda geliştirilmesi optimizasyon teorisinin gelişiminde önemli bir kilometre taşı olmuştur. Kalkülüs, hem matematiksel bir fonksiyonun hem de fonksiyon oluşturabilen bağımsız değişkenlerin maksimum veya minimum cinsinden optimal koşullarının elde edilmesine olanak sağlamaktadır. Kalkülüs'ün kullanımı düzgün-davranışlı fonksiyonlarla sınırlandırılmıştır. Ancak, Kalkülüs uygulamalarında karşılaşılan

cebirsel problemlerin çözümü bazen güç olabilmektedir. Dolayısıyla, Kalkülüs pragmatik anlamda gerçek dünya problemlerinin optimizasyonunda yeterli ve güçlü bir araç olamamaktadır [3].

J.L. Lagrange'ın 1788 yılında Lagrange çarpanları yöntemini bilim dünyasının hizmetine sunması önemli bir adım olmuştur. 1939'da W. Karush'un kısıtlandırılmış problemler için optimallik koşullarını bulması optimizasyon teorisinde yeni bir atılım olmuştur. II. Dünya Savaşı'nın başlamasıyla 1942'de İngiltere ve Amerika Birleşik Devletleri'nin Yöneylem Araştırması gruplarını oluşturması optimizasyon dünyası için bir dönüm noktası olmuştur. Sezgisel optimizasyon araçlarından olan yapay sinir ağları 1943'de W. McCulloch ve W. Pitts tarafından çalışıldı. Ertesi yıl ise, J. Von Neumann ve O. Morgenstern tarafından "Oyun Teorisi ve Ekonomik Davranış" adlı eserle oyun kuramı tanıtıldı [5].

II. Dünya Savaşı'ndan sonra yeni sınıf optimizasyon teknikleri geliştirildi. Sözkonusu teknikler daha karmaşık problemlere başarıyla uygulandı. Bunda, yüksek hızlı dijital bilgisayarların geliştirilmesi ve optimum değerlerin elde edilmesi için nümerik tekniklere matematiksel analizin uygulanması son derece etkili olmuştur. Nümerik teknikler Kalkülüs'ün bir takım zorluklarını ortadan kaldırmıştır [3].

Lineer programların çözümü için Simpleks yöntem 1947'de G.B. Dantzig tarafından geliştirildi. Bu, optimizasyon dünyasında gerçekten bir devrim sayılmaktadır. R. Bellman 1950'de dinamik programlama modelini ve çözümünü geliştirdi. 1951'de H. Kuhn ve A. Tucker daha önce Karush'un önerdiği kısıtlandırılmış problemler için optimallik koşullarını tekrar formüle ederek doğrusal olmayan programlama modelleri üzerinde çalıştılar. Yine aynı yıl, J. Von Neumann, G. Dantzig ve A. Tucker primal-dual lineer programlama modellerini geliştirdiler. Yine önemli bir katkı 1955'de stokastik programlama adı altında G. B. Dantzig tarafından yapıldı. Kuadratik programlama 1956'da M. Frank ve P. Wolfe tarafından geliştirildi. 1958'deki önemli bir katkı R. Gomory tarafından tamsayılı programlama olarak adlandırıldı. A. Charnes ve W. Cooper şans kısıtlı programlama modellerini 1959'da optimizasyon dünyasına armağan ettiler. 1960'da sezgisel optimizasyon araçlarından birisi olan yapay zeka ve yöneylem araştırması ilişkilerini içeren çalışmalar yapıldı. Hedef programlama modeli yine A. Charnes ve W. Cooper tarafından 1965 yılında geliştirildi. 1975'de çok amaçlı karar verme teorisinin temelleri M. Zeleny, S. Zionts,

J. Wallenius, W. Edwards ve B. Roy tarafından atıldı. L. Khachian lineer programlama modellerinin çözümü için farklı bir algoritma olan elips yöntemini 1979'da geliřtirildi. 1984'te, N. Karmarkar lineer programlama için alternatif bir çözüm algoritması olan içnokta algoritmasını geliřtirdi. 1992'de J.H. Holland tarafından bir sezgisel optimizasyon tekniđi olarak kabul edilen genetik algoritma geliřtirildi. Çađdař optimizasyon dünyasında da her geen gün artan bir ivmeyle önemli katkılar yapılmakta ve bilimin hizmetine sunulmaktadır [6].

Optimizasyon modelleri yukarıda da belirtildiđi gibi matematiksel teknikler kullanılmaktadır. Daha özel anlamda, optimizasyon modelleme geleneksel olarak *matematik programlama* olarak adlandırılmaktadır [7]. Diđer bir ifadeyle, matematik programlama, optimizasyon modelinin kurulması ve çözümün elde edilmesi işlemine verilen genel isimdir. Geçmiřten gelen bir gelenekle günümüzde de "matematik programlama" ve "optimizasyon" kavramları eşanlamlı olarak kullanılmaktadır.

Matematik programlama modelleri çeřitli kriterlere göre sınıflandırılabilir. Matematik programlar fonksiyonlarının tipine göre, yukarıda deđinildiđi gibi, birinci dereceden fonksiyonlardan oluşuyorlarsa *lineer programlama*, diđer durumlarda ise *lineer olmayan programlama* şeklinde sınıflandırılırlar. Karar deđişkenlerinin tipine göre, sadece tam sayılı deđişkenlerden oluşan problemlere *tam sayılı programlama* adı verilir.

Hem sürekli hem de tam sayılı deđişken içeren modeller ise *karma tam sayılı programlama* adını alırlar. En az bir tane rassal parametre içeren programlar ise *stokastik programlar* olarak nitelendirilirler. Aksi halde ise model *deterministik* olarak isimlendirilir. Optimizasyon probleminin çözümü zamanın bir fonksiyonu ise, problem *dinamik programlama* olarak adlandırılmaktadır. Dinamik programlama da kendi içerisinde deterministik ve stokastik olarak sınıflandırılabilir. Birden fazla amaç fonksiyonuyla başa çıkmak için geliřtirilen ve çok kriterli karar verme aracı olan *hedef programlama*, birbirleriyle çelişebilen amaçları hep birlikte göz önüne almakta ve amaçlardan sapmaları minimize ederek çözüme ulaşmaktadır. *Konveks* ve *kesirli programlama* türleri de yine yaygın olarak kullanılabilen optimizasyon modellerindendir.

Burada sadece bazılarında söz edilen matematik programlama türlerinin çözümleri için farklı matematiksel yöntemler geliştirilmiştir. Örneğin, lineer programlar için geliştirilen Simpleks yöntem tüm lineer modelleri çözüme potansiyeline sahipken lineer olmayan programlama modellerinin hepsini çözebilen genel bir çözüm yolu geliştirilememiştir. Lineer olmayan modeller için önerilen algoritmalar bazı özellikleri taşıyan tiplere uygulanabilmektedir. Söz gelimi, eşitlik kısıtlı lineer olmayan modellere Lagrange çarpanları kullanılırken eşitsizlik kısıtlı problemlere de Kuhn-Tucker koşulları uygulanmaktadır.

Bu çalışmada, matematik programlama modellerinden doğrusal programlama ve hedef programlama incelenmiş ve bir doğrusal hedef programlama problemi çözülmüştür.

1. DOĐRUSAL PROGRAMLAMA

1.1. GİRİŐ

Dođrusal eŐitsizlikler sistemi Őeklindeki bir problemin incelenmesi Fourierin alıŐmalarına kadar dayanmaktadır ve bu tanınmıŐ matematiki anısına Fourier-Motzkin eliminasyon yntemi Őeklinde isimlendirilmiŐtir.

1920'lerde Sovyet Rusya'da tm ekonomi planlaması konuları pratikte n plana gemiŐken teorik olarak tm ekonominin nasıl planlanabileceđini gstermek iin yapılan teorik alıŐmalar arasında Leonid Kantorovich'in katkısı ilk defa bir dođrusal programlama probleminin aıkca ortaya ıkarılmasına yol amıŐtır. Ne yazık ki teorinin pratik planlamaya uygulanmasının imknsızlıđı ve ideolojik nedenler dolayısıyla Kantorovich'in bu alıŐmasının nemi ancak II. Dnya SavaŐından sonra anlaŐılabilmıŐtir.

İkinci Dnya SavaŐı sırasında BirleŐik Amerika'da ortaya ıkan lojistik tahsis sorunlarını incelemek iin kurulan bir araŐtırma grubu, grup baŐkanı olan George Dantzig etrafında, bu trl sorunların zlmesi iin dođrusal programlama probleminin tanımlanması geređini ortaya ıkartmıŐlar ve bu trl dođrusal programlama problemlerinin zm iin simpleks algoritması adını verdikleri bir zm sistemi ortaya atmıŐlardır. zelikle bu matematik modelin ve zm algoritmasının, maliyetleri ve getirileri planlama suretiyle harp masraflarının kısılmasına yol atıđı aıkca grldđ iin bu teorik ve pratik geliŐmeler 1947'ye kadar devlet sırası olarak saklı kalmıŐtır. 1947'de John von Neumann, zellikle oyunlar teorisi ile de ilgileniyorken, dualite teorisini geliŐtirmiŐtir.

Bu zamana kadar dođrusal programlamaya yaptıkları katkılar nedeni ile Kantorovich, Dantzig ve von Neumann'a 1975'de Nobel Ekonomi dl verilmiŐtir.

1947'den sonra özellikle geliştirilen bilgisayar uygulamaları ile birlikte özellikle büyük özel sanayi birimleri ve büyük devlet projeleri için birçok doğrusal programlama problem tanımlanmış ve simpleks algoritması ile çözülüp pratikte kullanılmaya başlanmıştır. Örneğin petrol rafine şirketlerinin günlük üretim planlamaları ve çok girdili ve çok çıktılı üretim karışımı planlamaları için doğrusal programlama çözümlerini devamlı olarak kullanmaya başlamışlardır.

1979'da Leonid Hacıyan doğrusal programlama probleminin polinom zaman içinde çözülebileceğini ilk defa ispat etmiştir.

Doğrusal programlamanın pratik yararlılığı bu yöntemin ilk kullanılma problemlerinden biri olan ve Dantzig tarafından ortaya atılan 70 kişinin 70 göreve, karar verici kuruma en iyi sonuç çıkaracak şekilde, tahsis edilmesi örneğinin biraz daha ayrıntılarına bakmak suretiyle anlaşılabilir. Eğer çözüm için her mümkün tahsisi teker teker elden geçirip her biri için amaca yaptığı katkıyı bulmak deneyimine girilirse, bu kadar çok büyük sayıda permutasyonun elden geçirilmesinin imkânsız olduğu anlaşılacaktır. Eğer bu problem doğrusal programlama şeklinde belirtilip en iyi çözümü bulma kararı verirse, en zor ve zaman alıcı çabanın probleminin çözümünde değil problemin programa girdisinin hazırlanmasında olduğu anlaşılır. Bu problemin bilgisayarla simpleks algoritması kullanılarak çözülmesi göz kırpmaya bile almaz. Doğrusal programlama kuramı arkasında bulunan teori, kontrol edilmesi gereken mümkün en iyi çözüm sayısını çok etkili şekilde azaltmaktadır.

1.2. DOĞRUSAL PROGRAMLAMAMANIN TANIMI

Doğrusal programlama; kaynakların optimal dağılımının, kaynakların seçenekli dağılımının, optimal üretim bileşiminin, minimum maliyeti veren girdi bileşiminin, en uygun karın ve en az maliyetinin belirlenmesinde kullanılmaktadır. Ayrıca işletmede; karşılaşılabilecek darboğazların giderilmesinde, seçenekli üretim tekniklerinin kullanılmasının getirisini belirlemede, kıt kaynakların etkin kullanımı ve bunların gölge fiyatlarının belirlenmesiyle en uygun çözümlere ulaştıracak politikaları saptamada doğrusal programlama modelleri kullanılır.

Doğrusal programlama değişkenlere ve kısıtlayıcılara bağlı kalarak amaç fonksiyonunu en uygun (maksimum veya minimum) kılmaya çalışır. Buna göre doğrusal programlama değişkenlere ve kısıtlayıcı şartlara bağlı kalarak amaca en iyi ulaşma tekniğidir. Temel olarak doğrusal programlama, verilen optimallik ölçütüne bağlı kalarak, kıt kaynakların optimal şekilde dağıtımını içeren deterministik matematiksel bir tekniktir.

Tanımdaki en iyileme iki yolla gerçekleştirilir:

i) Amaç fonksiyonu değerinin en büyüklenmesi (maksimizasyon).

ii) Amaç fonksiyonu değerinin en küçüklenmesi (minimizasyon).

1.3. DOĞRUSAL PROGRAMLAMANIN YAPISAL GÖRÜNÜMÜ

Bir doğrusal programlama problemi, gerçekleştirilmesi arzu edilen, açık ve ölçülebilir biçimde belirlenen bir doğrusal amaç fonksiyonuyla bu amaç fonksiyonunun gerçekleşme derecesini yani alabileceği sayısal değeri sınırlayan, doğrusal eşitlik ve eşitsizlikler biçiminde ifade edilen kısıtlıklardan oluşur.

1.3.1. Amaç Fonksiyonu

Doğrusal programlamada amaç, matematiksel biçimde formüle edilen ve ifade ettiği sayısal değeri en büyükleme yada en küçükleme gibi iki durumu gerçekleştirilmeyi hedefleyen bir tek doğrusal fonksiyondur. Doğrusallıkla değişkenlerin birinci dereceden olması ve çarpımlarının bulunmaması kastedilmektedir.

Karar değişkenleri x_1, x_2, \dots, x_n ; katsayılar C_1, C_2, \dots, C_n ile gösterilirse amaç

fonksiyonu $Z = C_1x_1 + C_2x_2 + \dots + C_nx_n$ ya da genel olarak $Z = \sum_{j=1}^n C_j x_j$

($j=1,2,\dots,n$) biçiminde ifade edilebilir.

Z_{enb} amacın en büyükleme, Z_{enk} amacın en küçükleme yönelik olduğunu ifade etmektedir.

1.3.2. Kısıtlayıcı Fonksiyonlar

Kısıtlayıcı fonksiyonlar sadece kaynakların sınırlarını değil, gereksinim ve yönetim kararlarını ifade etmekte de kullanılırlar.

$a_{11}, a_{12}, \dots, a_{mn}$ ve b_1, b_2, \dots, b_m sabit sayılar olmak üzere kısıtlayıcılar karar değişkenlerinin doğrusal fonksiyonları olarak;

$$\left. \begin{array}{l} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \quad (\leq = \geq) \quad b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \quad (\leq = \geq) \quad b_2 \\ \dots\dots\dots \\ \dots\dots\dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \quad (\leq = \geq) \quad b_m \end{array} \right\} \quad (1.1)$$

biçiminde ya da genel olarak $i = 1, 2, \dots, m$ olmak üzere ; $\sum_{j=1}^n a_{ij}x_j \leq b_i$,

$\sum_{j=1}^n a_{ij}x_j \geq b_i$, $\sum_{j=1}^n a_{ij}x_j = b_i$ biçiminde gösterilebilir.

En büyükleme problemlerinin kısıtlayıcı fonksiyonları genellikle \leq , en küçükleme problemlerinin kısıtlayıcı fonksiyonları \geq işaretleriyle gösterilir. Öte yandan $=$ işaretli kısıtlayıcı fonksiyonlara hem en büyükleme hem de en küçükleme problemlerinde rastlanır.

1.3.3. Negatif Olmama Koşulu

Doğrusal programlama problemlerinde yer alan değişkenlerin (x_j 'lerin) bu tekniğin gereği olarak negatif değerler alınmaması istenilir. Yani; $x_1 \geq 0$ $x_2 \geq 0$... $x_n \geq 0$ veya kısaca $x_j \geq 0$, $j = 1, 2, \dots, n$ dir.

Maksimize durum için;

$$\text{Amaç Fonksiyonu: } Z_{\text{enb}} = \sum_{j=1}^n C_j x_j$$

$$\text{Kısıtlayıcı fonksiyonlar: } \sum_{j=1}^n a_{ij} x_j \leq b_i,$$

Negatif olmama koşulu: $j = 1, 2, \dots, n$ olmak üzere $x_j \geq 0$ biçimindedir.

Minimize durum için ise, amaç fonksiyonu $Z_{\text{enk}} = \sum_{j=1}^n C_j x_j$, kısıtlayıcı fonksiyonlar

$\sum_{j=1}^n a_{ij} x_j \geq b_i$, negatif olmama koşulu $j = 1, 2, \dots, n$ olmak üzere $x_j \geq 0$ biçimindedir.

Doğrusal programlamanın matematiksel model olarak açıklanması sayısal sonuçlarda karar vericinin kişisel etki ve kararlarını ortadan kaldırır.

Burada;

Z : Amaç fonksiyonu,

i = 1, 2,m : Üretim bölümlerinin veya üretim kaynaklarının sayısı,

j = 1, 2,n : Ürün çeşidi (faaliyet sayısı),

x_j : Belirli bir zaman döneminde üretilen j inci ürünün üretim miktarı veya faaliyet düzeyi,

C_j : x_j de oluşan bir birimlik artışın Z de oluşturacağı değişiklik,

a_{ij} : j inci üründen bir birim üretmek için i inci kaynaktan tüketilen kaynak miktarı,

b_i : i inci sınırlı kaynak miktarıdır.

Doğrusal programlamada, girdi katsayılarının oluşturduğu matrisin özel bir yeri vardır.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & \cdots & a_{mn} \end{bmatrix} \rightarrow \text{Girdi katsayılarının oluşturduğu matris}$$

Üretim teknolojisini gösteren bu matrise **teknoloji matrisi** denir. Gerçekten bu matrisin elemanları olan a_{ij} ler her üründen bir birim üretmek için gerekli kaynak miktarını, başka bir deyişle üretim teknolojisini ifade etmektedir. Matrisin satırları bakımından incelendiğinde, her bir satırın n çeşit ürünün her birinden birer birim üretmek için gerekli olan belirli bir kaynak gereksinimine karşılık geldiği görülebilir. Matrisin sütunları da belirli bir üründen bir kısım üretmek için gerekli olan kaynak gereksinimini gösterir.

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ \vdots \\ b_m \end{bmatrix} \rightarrow \text{Kaynakların sınırlarını açıklayan kaynak miktarı vektörü}$$

$$C = [C_1 \quad C_2 \quad \cdots \quad \cdots \quad C_n] \rightarrow \text{Amaç fonksiyonu katsayıları satır vektörü}$$

$$x = [x_1 \quad x_2 \quad \cdots \quad \cdots \quad x_n]^T \rightarrow \text{Karar değişkenleri matrisi}$$

Bu durumda doğrusal programlama problemi aşağıdaki biçimde yazılabilir:

Amaç Fonksiyonu :

$$Z = [C_1 \quad C_2 \quad \cdots \quad \cdots \quad C_n] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_n \end{bmatrix} = Cx$$

Kısıtlayıcı Fonksiyonlar :

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_n \end{bmatrix} (\leq = \geq) \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ \vdots \\ b_m \end{bmatrix}$$

Negatif Olmama Koşulu:

$$x \geq 0$$

Doğrusal programlama gerçeğe uygun durumlara, varsayımlarının fazla katı olmaması nedeni ile kolaylıkla uyarlanabildiği için oldukça geniş bir kullanım alanı bulmuştur. Denilebilir ki bir kaynak dağılım probleminin olduğu her alanda doğrusal programlama kullanılabilir.

Doğrusal programlama alanlarına ;

- Portföy yönetimi problemleri,
- Taşımacılık problemleri,
- Tarımsal problemleri,
- Üretim stok kontrol problemleri,
- Mamul karışım problemleri,
- Makin-işgücü atama problemleri,
- İşgücü-programlama problemleri,
- Pazarlama problemleri,
- Doğal kaynakların ülke ihtiyaçlarına uygun şekilde tahsisi problemleri,

gibi birbirinden çok farklı alanlar örnek verilebilir.

1.4. DOĞRUSAL PROGRAMLAMANNIN ÇÖZÜM YÖNTEMLERİ

1.4.1. Grafikle Çözüm Yöntemi

Grafik çözüm tekniği ancak üç değişkenli problemlerin çözümünde bir karışıklığa neden olmadan kullanılabilir. Söz konusu teknikte başvurulacak işlemler:

- Kısıtlayıcıların grafiğini teşkil etmek,
- Uygun çözüm alanını belirlemek,
- Amaç fonksiyonunun grafiğini çizmek,
- En iyi veya optimum çözümü bulmak.

Çözüm hazırlığı x_1 , x_2 eksen takımı üzerinde, birer doğrudan oluşan eşitsizlikler, doğru çizimindeki yöntemlerle, yani iki noktası belli edilerek çizilir ve eşitsizliği sağlayan alan taranır. Böylece bütün şartları sağlayan alan-çözümler alanı veya poligonu saptanmış olur. Bütün teknik olarak olurlu çözümler bu poligon içindedir. Bundan sonra bu olurlu çözümlerden amacı (burada toplam karı) maksimum kılan çözüm aranacaktır. Bunun içinde parametrik formdaki kar denklem yine x_1 - x_2 koordinat sisteminde, parametrenin herhangi bir değeri ve genellikle $K_t=0$ değeri için çizilir. Bu doğru K_t ye pozitif değerler verilerek ötelenir. Bu doğrulardan herhangi birisi üzerinde bulunan ve çözüm poligonu içinde kalan noktalar eş-kar getiren olurlu çözümleri verecektir. Bu kar doğruları K_t nin artan değerleri için çizilmeye devam edilir. Bu doğrulardan çözümler poligonun en son terk edenin son çıkışma noktası veya noktaları karı maksimum kılan çözüm veya çözümleri vermektedir. Açık olarak çözüm kesişim noktalarından birisine paralel bu doğrunun çözümler poligonuna bir kenar teşkil eden parçasıdır. Eksenleri x_1 ve x_2 olan iki boyutlu bir grafik üzerinde çalışacağız.

İlgi alanımız $x_1 \geq 0$ ve $x_2 \geq 0$ negatif olmama şartlarının sağlandığı negatif olmayan bölgeden ibarettir.

Bir doğrusal programlama ister tekli ister çoklu çözüm durumu geçerli olsun, optimum çözüm uygun çözüm alanının sınırlı sayıdaki köşelerinde aranmalıdır. Bu köşelerden en az bir tanesi optimum çözümü taşıyacaktır. Uygun çözüm alanında sonsuz sayıda uygun çözüm bulunduğuna göre bunların arasında optimum olanını

seçmek ilk bakışta çok zor gibi görünebilir. Ne var ki uygun çözüm alanının bir özelliği optimum çözümün bulunmasını büyük ölçüde kolaylaştıracaktır. Bu özellik uygun çözüm alanının bir konveks noktalar kümesi olmasıdır. Konveks küme öyle bir kümedir ki içinde yer alan herhangi iki noktayı birleştiren doğrudan bütünüyle bu küme içindedir.

Doğrusal olan bir amaç fonksiyonunun konveks bir küme üzerinde maksimum veya minimum değerine bu kümenin bir köşe noktasından ulaşacağı, durum böyle olunca da herhangi bir doğrusal programlama modelinin optimum çözümünün uygun çözüm alanının köşe noktalarından birinde yer alacağı matematiksel olarak ispatlanmıştır.

1.4.2. SİMPLEKS ÇÖZÜM YÖNTEMİ

Grafikle çözümün uygulanamadığı çok değişkenli doğrusal programlama problemlerinin çözümünde yaygın biçimde kullanılan yöntem simpleks yöntemidir.

George B. Dantzig tarafından geliştirilen bu yöntem tekrarlı bir yöntem olduğundan simpleks algoritma olarak da adlandırılmaktadır.

1.4.2.1. Standart Biçim

Daha önce açıklandığı gibi problemin belirlenmesinden sonra yapılması gerekli en önemli iş, problemi en iyi biçimde temsil eden ve çözümü kolay olan bir modelin kurulmasıdır. Bilindiği gibi, doğrusal programlama problemleri farklı biçimlerde gösterilmektedir. Amaç fonksiyonu Z_{enb} ve Z_{enk} , kısıtlayıcı fonksiyonları eşitsizlik (\leq veya \geq) ya da eşitlik biçiminde tanımlanabilir.

Standart Biçim; Bir doğrusal programlama modeli aşağıdaki özelliklere sahipse, standart biçimde olduğu söylenir.

1. Tüm karar değişkenleri negatif değildir.
2. Amaç fonksiyonu en büyükleme veya en küçükleme tipindedir.
3. Tüm kısıtlayıcı fonksiyonlar (negatif olmama koşulu dışında) = işaretlidir.

4. Kısıtlayıcı fonksiyonların sağ taraf sabitleri negatif değildir.

Simpleks yöntem bir eşitlikler sistemine, standart işlemlerin tekrar tekrar uygulanmasıyla çözüm arayan bir süreçtir. Bu nedenle yöntemin en önemli adımı kısıtlayıcı fonksiyonların eşitlik biçiminde yazılmasıdır. Eşitsizlik biçimindeki bir kısıtlayıcının eşitsizliğin yönü bakımından iki türlü olduğu bilinmektedir. Eşitsizlikler

$$\sum_{j=1}^n a_{ij}x_{ij} \leq b_i \text{ veya } \sum_{j=1}^n a_{ij}x_{ij} \geq b_i \text{ biçimindedir.}$$

(\leq) işaretli eşitsizlikleri eşitlik biçimine dönüştürmek için bunların sol taraflarına negatif olmayan birer değişken eklenir. “Aylak değişken” adı verilen bu değişkenler $x_{n+1}, x_{n+2}, \dots, x_{n+m}$ ile gösterilir. (\geq) işaretli eşitsizlikler ise, sol taraflarından negatif olmayan birer değişken çıkartılmasıyla eşitlik biçimine dönüştürülür. Eşitsizliğin iki tarafı arasındaki farkı gösteren bu değişkene “artık değişken” denir. Bu değişkenler de aylak değişkenler gibi $x_{n+1}, x_{n+2}, \dots, x_{n+m}$, sembolleriyle gösterilirler. Yukarıda açıklandığı gibi, negatif olmama koşulu karar değişkenlerinin yanı sıra aylak ve artık değişkenlere de uygulanmaktadır. Bunun nedeni, kısıtlayıcı fonksiyonlardaki (\geq) ve (\leq) şartlarının gerçekleşmesini sağlamaktır.

1.4.2.2. Standart Biçimin Matris Gösterimi

Standart biçimdeki doğrusal programlama modelinin kısıtlayıcı fonksiyonları matrislerle şöyle gösterilir.

$$\begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1n} & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2n} & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{m1} & a_{m2} & \cdot & \cdot & \cdot & a_{mn} & 0 & 0 & \cdot & \cdot & \cdot & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \\ x_{n+1} \\ x_{n+2} \\ \cdot \\ \cdot \\ x_{n+m} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ b_m \end{bmatrix}$$

Kısıtlayıcıların eşitlik biçimine dönüştürülmesiyle , modelin yürürlükteki n karar değişkenine m değişken eklenmiş yani bilinmeyen sayısı n 'den $(n+m)$ ' ye çıkartılmıştır. $(n+m)$ bilinmeyene karşılık denklem sayısı m olduğundan , herhangi bilinmeyen sifıra eşitlenip diğer m bilinmeyen eşitlikler sisteminin birlikte çözülmesiyle elde edilir. Bu yolla ulaşılan çözüme temel çözüm, temel çözümde değerleri sıfırdan farklı olan değişkenlere ise temel değişken denir. Çözüm değerleri sıfır olan değişkenler temel olmayan değişkenlerdir.

Temel çözüm sayısı sonlu bir sayıdır ve herhangi m sayıdaki değişkeni göz önünde tutarak elde edilir. m kısıtlayıcı, n değişkenin bulunduğu standart biçimindeki problemin temel çözümlerinin sayısı $\binom{n}{m} = \frac{n!}{m!(n-m)!}$ dir.

İçeriğindeki değişkenlerin tümü pozitif (>0) olan temel “çözüme temel uygun çözüm” denir. Temel değişkenlerden bir yada bir kaçının sifıra eşit olması durumundaki çözüme “bozuk çözüm” denir. Temel uygun çözümlerin sayısı da yukarıdaki bağıntıyla hesaplanan sayıyla sınırlıdır.

Ancak negatif olmayan artık değişkenler için negatif değerler bulunması, bu temel çözümün uygun olmadığına işaret eder. Bu nedenle, negatif olmama koşulunu gerçekleyen diğer bir başlangıç çözümünün araştırılması zorunludur. Başlangıç çözümünü araştırmanın yolu, önceden olduğu gibi katsayılar matrisinin yanında bir birim matris oluşturmaktır. Birim matris oluşturmak için artık değişkenlerle eşitlik biçimine dönüştürülen kısıtlayıcı fonksiyonlara negatif olmayan birer “yapay değişken” eklenir. Yapay değişkenler, bir başlangıç uygun çözüme ulaşmak amacıyla (\geq) işaretli kısıtlayıcılara eklenen değişkenlerdir ve A_i ile gösterilirler.

Bilindiği gibi aylak ve artık değişkenlerin amaç fonksiyonundaki katsayıları sıfırdır. Yapay değişkenler için durum farklıdır. İlk temel uygun çözümün bulunmasına yardımcı olmalarına karşın hiçbir fiziki ve ekonomik anlamı olmayan yapay değişkenlerin en iyi çözüme girmeleri engellenmelidir. Bunu sağlamak için yapay değişkenlerin amaç fonksiyonundaki katsayılarının çok büyük olduğu düşünülür. Büyük değerli katsayılar genellikle M ile gösterilir. Bu uygulanacak simpleks yöntemine de “Charnes’in M yöntemi” denir. Bu yöntem hem enküçükleme hem de en büyükleme için kullanılır [10].

Simpleks Yönteminde kullanılan ilave deęişkenler, eşitsizlięin yönüne göre ařaęıda tablo biçiminde özetlenmiştir:

Kısıtın Eşitsizlik Yönü	Deęişken	a Katsayısı	C Katsayısı	
			Z enb	Z enk
\leq	Aylak	1	0	0
\geq	Artık	-1	0	0
	Yapay	1	-M	+M
=	Yapay	1	-M	+M

Tablo 1.1. Simpleks Yönteminde kullanılan ilave deęişkenler tablosu

1.4.2.3. Başlangıç Çözüm Tablosu

Standart biçimin oluşturulmasından sonra en iyi çözümün araştırılması işlemine geçilebilir. Simpleks yöntemin ardışık tekrarları başlangıç çözüm tablosu adı verilen bir tablonun düzenlenmesinden sonra başlar. Başlangıç çözüm tablosu, ařaęıdaki tablo esasına göre düzenlenir.

	2	3			4				5	
	↓								↓	
1 →	TDV	x_1	x_2	...	x_n	x_{n+1}	x_{n+2}	...	x_{n+m}	ÇV
	0 x_{n+1}	a_{11}	a_{12}	...	a_{1n}	1	0	...	0	b_1
	0 x_{n+2}	a_{21}	a_{22}	...	a_{2n}	0	1	...	0	b_2

	0 x_{n+m}	a_{m1}	a_{m2}	...	a_{mn}	0	0	...	1	b_m
6 →	Z_j	0	0	...		0	0	...	0	0
7 →	$Z_j - C_j$	$-C_1$	$-C_2$...	$-C_n$	0	0	...	0	-

Tablo 1.2. Simpleks Başlangıç Çözüm Tablosu

1. Deęişkenler satırı: Tablonun ilk satırıdır. Standart biçimin tüm deęişkenleri önce karar deęişkenleri, sonra dięer deęişkenler olmak üzere bu satırda gösterilir.

2. Temel deęişkenler sütunu: Tablonun ilk sütunudur. Tablodaki çözüme karşılık gelen temel çözümün deęişkenleri ile bu deęişkenlerin amaç fonksiyonu katsayılarını gösterir.

Başlangıçta sadece aylak deęişkenler temelde bulduklarından, C_j ler sıfıra eşittir. Bu sütunun önemli özellięi başlangıç çözümünün sıfır olduğunu göstermesidir. Yani,

başlangı simpleks tablosunda aylak değişkenler temel değişkenler, karar değişkenler ise temel olmayan değişkenlerdir. Yukarıda açıklandığı gibi, temelde bulunmayan değişkenlerin özüm değerleri sıfırdır. Temel olmayan değişkenler için sıfır konulmasıyla temel değişkenler ve değerleri,

$$x_1, x_2, \dots, x_n = 0 \text{ ve } x_{n+1} = b_1, x_{n+2} = b_2, \dots, x_{n+m} = b_m$$

olur. Bu çözüm koordinat sisteminin orjin noktasına karşılık gelir.

3.Gövde: Problemin orijinal karar değişkenlerinin kısıtlayıcı fonksiyonlardaki katsayılarından (a_{ij} , $i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$) oluşan $m \times n$ matristir.

4. Birim matris: Aylak değişkenlerin kısıtlayıcı fonksiyon katsayılarının oluşturduğu $m \times m$ birim matristir.

5. Çözüm vektörü: Temeldeki değişkenlerin çözüm değerlerini gösteren $m \times 1$ sütun vektördür. Başlangıçta, kısıtlayıcı fonksiyonların sağ taraf sabitlerinden oluşur.

6. Z_j satırı: Yürülükteki temelde bulunan değişkenlerin amaç fonksiyonu katsayıları ile x_j sütunundaki katsayıların karşılıklı çarpımlarının toplamından oluşur. Buna göre örneğin,

$$Z_1 = 0(a_{11}) + 0(a_{21}) + \dots + 0(a_{m1}) = 0 \text{ olur.}$$

Temel değişkenlerde ortaya çıkan değişiklikler nedeniyle, amaç fonksiyonu değerinin değişimi Z_j sembolünde toplanır. Ayrıca, bu satırın son elemanı amaç fonksiyonunun o simpleks tablosu için aldığı değeri gösterir. Başlangıç tablosunda bu değer genellikle sıfıra ($0(x_{n+1})+0(x_{n+2})+\dots+0(x_{n+m})=0$) eşittir.

7. $Z_j - C_j$ satırı: Tablonun son satırıdır. Elemanları, Z_j ile o sütunla ilgili değişkenin amaç fonksiyonu katsayısı arasındaki farka eşittir. $Z_j - C_j$ farkları x_j değişkeninin temele alınmasının amaç fonksiyonunda yol açacağı değişikliği ters işaretlerle gösterir.

Başlangıç çözümünün bulunmasından sonra sıra amaç fonksiyonunun değerini artıracak diğer temel değişkenlerin araştırılmasına gelir. Bunun için, temelde bulunan değişkenlerden bir tanesinin temelden çıkartılması, yerine bu adımda temelde

bulunmayan deęişkenlerden bir tanesinin alınması gerekir. Amaç fonksiyonuna marjinal katkısı en büyük olan deęişken çözüme giren ilk deęişken olacaktır. Marjinal katkının büyüklüğü deęişkenlerin amaç fonksiyonundaki katsayılarının incelenmesiyle belirlenir.

Yeni bir deęişken temele girdiğine göre, yürürlükteki temelde bulunan deęişkenlerden bir tanesinin temeli terketmesi gerekir. Bu yolla temeldeki deęişken sayısının aynı kalması sağlanır.

Temeli terkedecek deęişken ölçütü, çözüm vektörü elemanlarının çözüme girmesine karar verilen deęişkene ait sütunun karşılıklı elemanlarının oranına dayanır. Bu oranlar arasından (negatif ve sıfır olanlar dışında) en küçük olana sahip deęişken temelden ayrılır.

$Z_j - C_j$ satırındaki tüm deęerler, en büyükleme problemlerinde ≥ 0 , en küçükleme problemlerinde ise ≤ 0 ise en iyi çözüm elde edilmiş olur.

Simpleks yönteminde, temele giren deęişkenin bulunduğu sütuna *anahtar sütun* denir. Temeli terkeden deęişkenin bulunduğu satıra *anahtar satır* denir. Anahtar sütun ile anahtar satırın kesiştiği gözedeki deęere *anahtar sayı* denir.

Anahtar sayının belirlenmesinden sonra anahtar sütun birim sütun vektöre dönüştürülebilir. Bunu sağlamak için GAUSS-JORDAN eleme yöntemindeki eleme işlemine denk olan anahtar işlemlerden yararlanır. Bu yonteme yapılacak ilk iş anahtar satır elemanlarının anahtar sayıya bölerek anahtar satırın yeni elemanlarının hesaplanmasıdır. Bu yolla anahtar sayın bulunduğu gözedeki sayı 1 yapılır.

Diđer bütün satır elemanlarının (Z_j ve $Z_j - C_j$ satırları dışında) yeni deęerleri aşağıdaki formülle bulunur:

$$\left(\text{Eski Satır Elemanları} \right) - \left(\begin{array}{c} \text{Eski Satırla Anahtar} \\ \text{Sütunun Kesiştiği} \\ \text{Gözdeki Sayı} \end{array} \right) \times \left(\begin{array}{c} \text{Anahtar Satırın} \\ \text{Yeni Elemanları} \end{array} \right)$$

2. HEDEF PROGRAMLAMA

2.1. GİRİŞ

Karar problemleri amaç sayısına göre “tek amaçlı” ve “çok amaçlı” olmak üzere iki gruba ayrılmaktadır. Tek amaçlı problemlerle çok sık karşılaşılma ile birlikte bu problemlere yönelik en önemli eleştiri amaç sayısının birle eşit olmasıdır. Bu eleştiri tek amaçlı problemlerin çözümünü kolay olması yüzünden amaç sayısı birden çok olsa da karar vericinin amaç sayısını zorlamalarla birle indirmesidir. Bunun için karar verici amaçlarını ortak bir yönleri bakımından tek bir amaç olarak ifade edebilmelidir. Gerçek hayat problemlerinde bunu gerçekleştirmek her zaman mümkün değildir. Bu gibi durumlarda, genellikle en önemli görünen amaç öncelikle dikkate alınır diğerler amaçlar probleme ya kısıt olarak eklenir ya da başka bir optimizasyon problemi olarak değerlendirilir. Bir başka yaklaşım ise, çok sayıdaki amacın tek bir amaca dönüştürülmesidir. Birbiriyle çelişen amaçların aynı ölçekte değerlendirilmesi, dolayısıyla tek bir amaç fonksiyonu olarak ele alınması oldukça zordur. Problemdle çelişen amaçlar aynı ölçekte değerlendirilmiş olsalar da , çelişen bu amaçları optimum kılan tek bir çözümün bulunması olanaksız olabileceği gibi bulunan çözüm uygun olmayabilir[14]. Böyle durumlarda her amacın önem derecesini temel alan uzlaşık çözümler bulunabilir [15].

Burada, çok amaçlı durumları içeren problemler için hedef programlama tekniği tanıtılacaktır. Hedef programlamada ana düşünce, orjinali çok amaçlı olan problemi tek amaçlı olan probleme dönüştürmektir. Hedef programlama ile belirlenen hedeflerden olabilecek istenmeyen sapmalar en küçüklenir. Bu yüzden hedef programlama orantısız ve genellikle birbirleri ile çelişen hedeflerin bulunduğu problemlerde başarı ile uygulanabilmektedir [16].

2.2. HEDEF PROGRAMLAMANNIN GELİŐİMİ

Hedef programlama modeli, çok amaçlı matematiksel programlama modellerinden en çok bilinenlerden biridir. Hedef programlamadaki temel düşünce, her bir kriter için bir hedef belirlemektir.

Karar vericiler, doğrusal programlama ile çözülemeyecek olan çok hedefli bazı sorun yada durumlarla karşı karşıya kalabilir. Böyle bir durumda karar verici hangi hedefi tercih edecek ya da nasıl bir karar verecektir. İşte, hedef programlama , bu tür durumlarda kullanılabilen bir tekniktir.

Hedef programlamadaki ana fikir, önceden tanımlanmış hedefler çerçevesinde bir ya da daha çok amaç fonksiyonunun çözümünün bulunmasıdır.

İngilizce karşılığı “goal programming” olan yöntem Türkçeye “hedef programlama” veya “amaç programlama” olarak çevrilir. Ignizio (objektive) amacı “karar vericilerin arzu ettiği genel bir ifadenin yansıması” şeklinde tanımlarken hedefi, “istenilen bir seviye ile belirlenmiş bir amaç” olarak tanımlanmaktadır. Dervitsiotis (goal) ise amacı, “hareket etmek istenilen yön” olarak ifade ederken hedefi, “verilen bir zamanda ilerlemek istenilen yönün neresinde bulunduğunu gösteren bir terim” olarak tanımlamaktadır.

Hedef programlamada, hedef, istenen bir seviye ile belirlenmiş bir amaç olarak ve amacı da karar vericinin arzu ettiği genel bir ifadenin yansıması şeklinde tanımlanması uygun olacaktır.

Hedef programlama modeli 1950’ lere dayanmasına rağmen 1970’ lerin ortasından bu yana etkin bir şekilde kullanılmaya başlanılmıştır.

Hedef programlamanın ilk çıkışı, 1955 yılında Charnes ve arkadaşları tarafından yapılan çalışmaya dayanır. Hedef programlamanın ilk tanımı ise Charnes ve Cooper tarafından yapılmıştır [17].1970’lerin ortalarına kadar literatürde kısıtlı sayıda hedef programlama uygulamasına rastlanmaktadır. Daha sonra Lee ve Ignizio’nun

çalışmalarına dayanan gelişmeler olmuştur [18,19]. Bu çalışmalarda hedef programlama uygulamalarının ve teknik gelişmelerinin artışı öne çıkmıştır.

Romeo, Scnieederjans ve Tamiz' in yaptığı çalışmalarda birçok etkili uygulama alanının olduğu ortaya konulmuştur.

Hedef programlama ilk kez 1952'de fikir olarak ortaya atılmasına rağmen, algoritmik olarak 1961'de Charnes ve Cooper tarafından geliştirilmiştir. Algoritma daha sonra 1965'te Ijiri, 1972'de Lee ve 1976'da Ignizio ve 1984'de Schniederjans tarafından geliştirilmiştir [17,20].

Hedef programlama konusundaki ilk bilgisayar yazılımı, 1962 yılında anten sistemlerinin tasarımı ile ilgili olarak doğrusal olmayan hedef programlama problemlerinin çözümü amacı ile Ignazio tarafından yazılmıştır. Ignizio, 1967'de doğrusal hedef programlama için ardışık (iteratif) doğrusal programlama yöntemiyle yeni bir bilgisayar yazılımı geliştirmiştir. Daha sonra 1968 yılında Veikko Jaaskelinen doğrusal hedef programıyla ilgili bir bilgisayar yazılımı geliştirmiştir.1960'ların sonunda 1970'lerin başında Ignizio tamsayılı ve doğrusal olmayan hedef programlama modellerini de içeren algoritmalar ve bilgisayar yazılımlarını geliştirmeye devam etmiştir .

Gerçek hayata uyarlanabilirliği bakımından çok etkin olan hedef programlama,çok amaçlı karar verme metotları içinde en çok uygulama alanı olan tekniktir. Son yıllarda pek çok alanda ortaya çıkan problemlerin çözümünde bu teknikten faydalanılmaktadır.

Hedef programlama tekniğinden yararlanan alanlardan bazıları aşağıda verilmiştir [19];

Ulaştırma Problemleri, Üretim Planlaması, Orman Kaynaklarının Planlaması, Satın Alma, Enerji Planlaması, İşgücü Planlaması, Sağlık Planlaması, Akademik Kaynak Dağılımı Yerel Yönetimlerin Ekonomik Planlaması Okul Otobüs Servisleri Planlaması, Hastanelerde Kaynak Planlaması Proje Seçimi ve Yönetimi Genel Bütçe Planlaması Portföy Seçimi Finansal Planlama Pazarlama Stok Yönetimi, Maliyet Tahmin Tekniklerinin Geliştirilmesi, Zaman Standartlarının Geliştirilmesi, Beslenme (diyet) veya Yem Karışım Problemleri, Yatırım Planlaması, Öğrenci Başarısının

Kestirimi, Yükleme Problemleri, Performans Değerlendirme, Toplam Kalite Yönetimi, İş Değerlendirme, Tarımsal Üretim ve Yönetim.

2.3. HEDEF PROGRAMLAMAMANIN TANIMI

Hedef programlama yöntemi, doğrusal programlamanın daha fonksiyonel bir türüdür. Doğrusal programlama ile ancak tek bir hedef ve tek bir ölçekle ifade edilebilen problemlerin çözümü yapılabilmektedir. Hedef programlama ile aynı anda birden fazla ve farklı ölçekli hedefler saptanabilir ve belirlenen kısıtlar altında bu hedeflere ulaşılmaya çalışılır.

Hedef programlama tekniği ile problem çözümünde bir maksimum ya da minimum sonuç elde edilmez. Bu teknik ile amaçlar ile belirlenen hedeflerden sapmalar minimize edilmeye çalışılır [21].

Hedef programlamada mümkün olduğu kadar bütün hedeflere en iyi şekilde ulaşılmak istenir. Bu nedenle belirlenen her hedefin tek tek ele alınıp bu hedeflerin elde edilemeye ölçüsüyle ilgilenilir. Hedefler arasındaki sapmaların kısıtlamalar kümesine uygun olarak minimize edilmesine çalışılır.

Hedef programlama, doğrusal programlamanın iyi geliştirilmiş ve test edilmiş tekniklerini kullanırken, çok sayıda hedefin aynı anda incelenmesine de olanak sağlar. Bu yöntemin önemli bir üstünlüğü de işletmelerin, karar verme sürecinde çok sayıda amaç ve hedef birleştirmelerine izin vermesidir.

Hedef programlamanın teknik avantajlarından biri de hiçbir hedef gerçekleştirilebilir olmasa bile, her zaman bir çözüm sağlamasıdır. Hedef programlama, planlanmış amaçları tatmin etmek ve bunlara uygun ve gerçekçi yaklaşmak için kullanılır. Ayrıca hedef programlama sadece amaçların niteliklerini değil, bunların karşılaştırılmalarını ve hedef değerlerini de bildirir.

Hedef programlamanın amacı, tüm kısıtları sağlayan ve mümkün olduğu kadar tüm hedeflere ulaşan bir çözüm bulmaktır.

2.4. HEDEF PROGRAMLAMAMANIN YAPISI

Hedef programlama modelini oluşturan bileşenler şunlardır:

Amaçlar; kriterlerin karar vericilerin arzuları doğrultusunda yönlendirilmiş şekli olarak tanımlanabilir. Karar vericilerin isteklerini genel olarak belirten kavramdır. Örneğin karın maksimize edilmesi, personel değişim hızının minimuma inmesi, yoksulluğun ortadan kaldırılması karar vericilerin amaçları olabilir.

Karar Değişkenleri; modelde karar verici tarafından değeri belirlenmek istenen bilinmeyenlere karar değişkeni adı verilir. Karar değişkenleri x_i ler ile ifade edilir. Kontrol ya da yapısal değişkenler olarak da bilinirler. Karar değişkenleri, doğrusal programlama problemlerinde tanımlanan değişkenlerin aynısıdır. Örneğin üretilen ürün miktarı, istihdam edilecek işçi sayısı gibi.

Sistem Kısıtları; teknolojik, yapısal veya sistem kısıtlayıcıları probleme ilişkin geliştirilen hedef programlamada da tam olarak sağlanması gereken ve hiçbir sapmaya izin verilmeyen kısıtlayıcılarıdır. Sistem kısıtları, incelenen problemin doğası gereği oluşan kısıtlardır. Söz konusu bu kısıtlar, eldeki kısıtlı kaynakları ifade ederler. Doğrusal programlama problemlerindeki kısıtlara karşılık gelirler. Bunlar mutlak olan ve değişmelere kesinlikle izin verilmeyen kısıtlardır Doğrusal programlamadaki gibi formüle edilirler ve öncelikle bunların gerçekleştirilmesi gerekir.

Hedefler; amaçların daha da somutlaşarak belirli değerlere dönüşmüş şekilleri olarak tanımlanabilir. Hedefler, ulaşılmak istenilen düzeyin sayısal bir değer olarak ifade edilmiş halidir. Herhangi bir hedef fonksiyonunun değerinin belirli bir değer altında kalmaması, üstüne çıkmaması ya da belirlenen değere eşit olması şeklinde olabilir.

Hedef kısıtları; karar vericinin ulaşmayı istediği veya gerekli gördüğü hedefler, hedef programlamaya, hedef kısıtlayıcıları olarak aktarılır. Bunlar sistem kısıtları kadar katı ve değişmez değildir yani sistem kısıtlarına göre daha esneklerdir. Ulaşılmak istenen hedef değerlerini gösteren fonksiyonlardır. Sistem kısıtları

sağlandıktan sonra hedef kısıtlarının sağlanması süreci başlar. Hedef tam anlamıyla sağlanmışsa sapma sıfırdır. Hedefe ulaşılamamışsa negatif sapma, hedefin üzerinde bir başarı sağlanmışsa pozitif sapma meydana gelir. Pozitif sapmalar d_i^+ , negatif sapmalar d_i^- ile gösterilir. Hedef kısıtlayıcısı “ \geq ” yönünde ise d_i^+ istenen değişken ve d_i^- istenmeyen sapma değişkenlerdir. Hedef kısıtlayıcısı “ \leq ” yönünde ise d_i^- istenen değişken ve d_i^+ istenmeyen sapma değişkenlerdir. Hedef kısıtlayıcısı “ $=$ ” durumunda ise d_i^+ ve d_i^- her ikisi de istenmeyen sapma değişkenlerdir.

Amaç fonksiyonları; herhangi bir amaç için belirlenen hedeften sapmaları en küçükleyen fonksiyona amaç fonksiyonu adı verilir. Hedef programlamada , amaç fonksiyonun optimal değeri, sistem ve hedef kısıtlayıcılarının belirlediği çözüm alanı içinde aranır.

Başarı fonksiyonu; tüm amaç fonksiyonlarının belirli bir öncelik ve/veya ağırlığa göre toplama şeklinde yazılmasıyla oluşturulur. Başarı fonksiyonunun oluşturulmasındaki temel ilke, çok amaçlı modeli tek amaçlı bir modele indirgemektedir. Böylelikle, asıl amaç hedeflerden olabilecek istenmeyen sapmalar toplamını en küçükleme olacaktır.

Sapma değişkenleri; hedeflenen başarı ile gerçekleşen başarı arasındaki farklara karşılık gelir. Diğer bir ifadeyle sapma, eşitsizlik şeklinde oluşturulmuş bir amaç fonksiyonunda, sağ taraf yani hedef değerinin altında kalma ya da üstüne çıkma durumları olarak tanımlanabilir. Sapma değişkenleri, hedeflerin üstünde veya altında elde edilen faaliyetlerin miktarını belirleyen değişkenlerdir. Sapma değişkenleri negatif değerler alamazlar ve bir hedefin aynı anda hem üstünde hem de altında olmayacaklarından birinin değeri daima sıfır olur. Her bir hedef için negatif sapma ve pozitif sapma olarak adlandırılan iki adet sapma değişkeni tanımlanır. Hedef tam anlamıyla sağlanmışsa her iki sapma değişkeninin değeri sıfırdır. Hedefe ulaşılamamışsa negatif sapma, hedefin üzerinde bir başarı sağlanmışsa pozitif sapma meydana gelir.

Hedef Programlama tekniğinde amaç, hedeflerden sapmaların minimum olmasıdır. Bu nedenle, bütün sistem sapma değişkenlerinin değerlerini minimum yapabilmek üzerine kurulmuştur. Sapma değişkenlerinin problemin kuruluş aşamasında doğru olarak yerleştirilmesi, problemin doğru sonuçlandırılabilmesi için çok önemlidir.

2.5. HEDEF PROGRAMLAMAMININ SINIFLANDIRILMASI

Hedef programlama modeli gerek varsayımları gerekse modelin yapısal özellikleri nedeniyle çeşitli şekillerde sınıflandırılabilir.

2.5.1. Amaç Fonksiyonlarının Öncelik Seviyelerine Göre

Amaç fonksiyonlarına göre hedef programlama iki guruba ayrılır. Bunlar; Doğrusal Hedef Programlama ve Doğrusal Olmayan Hedef Programlama'dır.

2.5.1.1. Doğrusal Hedef Programlama

Amaç fonksiyonu,sistem ve hedef kısıtlayıcıları denklemlerinin tümünün doğrusal olduğu hedef programlama türüdür. Doğrusal hedef programlama modeli grafik yöntemle ya da değiştirilmiş simpleks yöntemleriyle çözülebilir.

2.5.1.2. Doğrusal Olmayan Hedef Programlama

Amaç fonksiyonu,sistem ya da hedef kısıtlayıcılarından her hangi birini doğrusal olmadığı hedef programlama türüdür. Doğrusal olmayan hedef programlama modelleri çeşitli teknikler ile doğrusal hale dönüştürüldükten sonra çözümlenebilir.

Doğrusal Olmayan Hedef Programlama problemlerinin çözümü için iki temel algoritma vardır. Bunlar, İteratif Doğrusal Olmayan Hedef Programlama ve Stewart - Griffith Doğrusallaştırarak Doğrusal Olmayan Hedef Programlama problemlerinin çözümüdür [17].

2.5.2. Karar Değişkenlerinin Alabilecekleri Değerlerine Göre

Karar değişkenlerine göre hedef programlama türleri ; Sürekli değerler alabilen hedef programlama , tamsayılı hedef programlama ve 0-1 hedef programlamadır.

2.5.2.1. Sürekli Hedef Programlama

Modeldeki değişkenlerin tümü (temel ve temel olmayan değişkenler) süreklilik gösterdiği durumlarda sürekli hedef programlama söz konusudur.

2.5.2.2. Tamsayılı Hedef Programlama

En az bir karar deęişkeninin tamsayı deęerler alması istenildięi durumlarda tamsayılı hedef programlama kullanılır. Bu tür hedef programlama modellerinin çözümü için Gomory'nin Kesme Düzlemi Yöntemi , White'ın Dal Sınır Yöntemi, minimum – maximax yöntemi gibi çeşitli çözüm algoritmaları geliştirilmiştir [21].

2.5.2.3. 0-1 Hedef Programlama

Herhangi bir faaliyet için yalnızca iki karar alternatifinin bulunduğu ve bunlardan birisine karar verilmesi gerektięi durumlarda 0-1 tamsayılı programlama modeli kullanılır. Genelde 0 faaliyetin yapılmayacağı, 1 ise yapılacağı şeklinde deęerlendirilir [22].

2.5.3. Katsayıların Özelliklerine Göre Hedef Programlama Türleri

Katsayıların özelliklerine göre üç tür hedef programlama teknięi vardır. Bunlar; deterministik, stokastik ve belirsiz (bulanık) hedef programlama teknikleridir [17].

2.5.3.1. Deterministik Hedef Programlama

Modeldeki sabit ve deęişkenlerin katsayılarının tümünün tam olarak bilindięi ve modele dahil edildięi hedef programlama türüdür.

2.5.3.2. Stokastik Hedef Programlama

Stokastik hedef programlama konusu yeni bir çalışma alanı olup , yeni çözüm yöntemleri geliştirilmeye devam edilmektedir.

2.5.3.3. Bulanık Hedef Programlama

Karar verici problemdeki bazı parametreleri, kısıtlayıcıları vb. kesin olarak ifade edememekte veya problemde çevre etkilerinden kaynaklanan belirsizlikler bulunmaktadır. Bu tür belirsizlikleri yok sayarak problemi tüm parametreleri

kesinmiş gibi modellemek yerine bu tür problemler bulanık matematiksel yöntemler kullanılarak çözülmektedir.

1970' te Bellman ve Zade bulanık amaç (G), bulanık kısıtlar (C), bulanık karar (D) gibi üç temel kavramı ortaya atmıştır [21].

Hedef programlamadaki hedef değeri kesin olarak modele yerleştirilmesi zorunluluğu bulanık hedef programlama yaklaşımı ile esnetilebilmekte ve bu durumda karar vericiye esneklik sağlamaktadır.

2.5.4. Hedeflerin Özelliklerine Göre Hedef Programlama Türleri

Hedeflerin özelliklerine göre hedef programlama tekniği aşağıdaki gibi sınıflandırılabilir.

2.5.4.1. Tek Hedefli Programlama

Ele alınan problemin tek hedefi olduğundan, karar vericinin istediği bu hedefe ulaşmaktır. Tek hedefi içeren problemlerin, modelinin kurulması ve onun çözümü ele alındığında, en basit hedef programlama problemi olduğunu görürüz.

2.5.4.2. Eşit Ağırlıklı Çok Hedefli Programlama

Burada, hedeflerin göreceli olarak önemleri birbirine eşittir ve bütün hedefler eşanlı olarak doyurulmaya çalışır. Herhangi bir hedefin diğer hedeflere göre bir önceliği söz konusu değildir.

Probleme ilişkin hedefler eşit önemli (ağırlıklı) ise, istenmeyen sapma değişkenleri toplama biçiminde ifade edilen amaç fonksiyonu, minimum kılınmaya çalışılır.

2.5.4.3. Ağırlıklı Çok Hedefli Programlama

Bu yöntemde belirlenen hedeflere önem düzeylerine göre ağırlık değerleri verilerek, hedefler tek bir amaç fonksiyonu olarak ifade edilir.

Böyle bir yaklaşım genellikle, sapma değişkenlerinin ölçü birimleri birbirinden farklı olduğu durumlarda tercih edilir. Ayrıca, karar verici hedefler arasındaki kendisi için önem farklılıklarını belirtmek için de ağırlıklandırma yoluna başvurulabilir.

2.5.4.4. Öncelikli Çok Hedefli Programlama

Karar vericilere göre, herhangi bir hedefin öneminin, diğer hedeflerden daha önemli olduğu durumlar söz konusu olabilir. Böyle durumlarda da hedef programlama modeli kullanılabilir. Burada, hedeflere ilişkin hiyerarşik bir yapının karar verici tarafından ortaya konulması ve hedeflerin en önemliden daha az önemliye doğru sıralanması söz konusudur. En öncelikli hedef sağlanmadan diğer hedeflerin sağlanması karar vericiler için anlamsızdır.

Hedeflerin karşılaştırılması sonunda göreceliği bir önem sırası ortaya çıktığında “Öncelikli Hedef Programlama” dan söz edilmektedir.

Öncelikli hedef programlama yönteminde, amaç fonksiyonunu oluşturmak için ulaşılması istenen hedeflerin hiyerarşik bir yapıda verilmesi gerekir. Karar verici, tercihini kullanarak hedeflerin en önemliden daha az önemliye doğru sıralamasını yapar. Bu sıralama işlemi sayısal veya sözel yapılabilir.

Öncelikli çok hedefli programlama yöntemi önem derecelerine göre hedeflerin önceliklendirilmesiyle başlar. Model daha sonra, yüksek öncelikli hedefin optimum değerinin düşük öncelikli hedef tarafından kötüleştirilmesine izin verilmeyecek şekilde her seferinde bir hedefi optimum kılar.

2.5.4.5. Ağırlıklı-Öncelikli Çok Hedefli Programlama

Ağırlıklı - Öncelikli hedef programlamada hedefler, önceliklerine göre gruplandırılır. Hedefler en yüksek önceliklerden başlayarak sağlanmaya çalışılır. Sonra ikinci öncelikte düzeydeki hedefler daha sonra üçüncü öncelik düzeyindeki hedefler olmak üzere hedeflerin tümü sırasıyla sağlanmaya çalışılır.

Bazı hedef programlama problemlerinde aynı hedefe ilişkin iki veya daha fazla sapma değişkeni, aynı öncelik düzeyinde amaç fonksiyonunda yer alabilir. Böyle bir durumda, sapma değişkenlerinin önceliği aynı P_i ise, bu sapma değişkenlerde ağırlıklar kullanılarak hangi sapmanın daha önemli olduğu belirlenir.

2.6. HEDEF PROGRAMLAMA MODELİ

Hedef programlama modeli çok amaçlı programlama modellerinin özel bir türüdür. Çok amaçlı programlama modelleri; optimizasyon düşüncesine dayanır ve kendi aralarında çelişen amaçları kısıtlayıcı kümesine göre eşanlı olarak doyuran bir çözüm vektörünü belirlemeyi amaçlar. Hedef programlama modelinde ise, karar vericinin doyurucu bulduğu bir çözüm belirlenmeye çalışılır. Bu noktadan hareketle, hedef programlama modelinin optimizasyon düşüncesinden daha çok bir doyum düşüncesine dayandığını söyleyebiliriz.

Her bir hedef için ilgili karar değişkenlerinin kullanılmasıyla bir hedef fonksiyonu oluşturulur. Hedefler, herhangi bir hedef fonksiyonunun değerinin belirli bir değerin altında kalmaması, üstüne çıkmaması veya belirlenen değere eşit olması şeklinde olabilir.

Sistem kısıtlarını ifade eden hedef fonksiyonlarında bu hedef değerleri kendiliğinden ortaya çıkar. Ancak diğer amaçlar için karar vericinin makul hedefler belirlemesi beklenir. Sözgelimi i . amaç için belirlenen hedef b_i olmak üzere, her hedef fonksiyonu aşağıdaki durumlardan birine uyacak şekilde ifade edilir.

$$f_i(x) \leq b_i \quad (2.1)$$

$$f_i(x) \geq b_i \quad (2.2)$$

$$f_i(x) = b_i \quad (2.3)$$

Hedef fonksiyonlarının erişim düzeylerinden ne kadar uzaklaşıldığının ölçülmesini sağlayan sapma değişkenleri, negatif ve pozitif sapma olarak ikiye ayrılır. d_i^- ile ifade edilen negatif sapma değişkeninin değerinin pozitif olması, ilgili hedefin belirlenen erişim düzeyinin altında bir değere ulaştığını gösterir, d_i^+ ile ifade edilen pozitif sapma değişkeninin değerinin sıfırdan büyük olması durumu ise; ilgili hedef için belirlenen erişim düzeyinin aşıldığını gösterir.

Negatif ve pozitif sapma deęişkenlerinin sıfır'a eřit olması ise ilgili hedef için belirlenen erişim düzeyine tam olarak ulařıldığını gösterir.

Bir hedeften hem negatif yönlü hem de pozitif yönlü sapma olması mümkün deęildir. Sapmalar tek yönlüdür. Hedeften eşanlı olarak tek bir sapma söz konusu olduęu için, sapma deęişkenlerinin negatif olmaması gerekir. Bu sapma deęişkenleri ařaęıdaki gibi tanımlanabilir:

$$d_i^+ = \begin{cases} f_i(x) - b_i & ; f_i(x) > b_i \\ 0 & ; f_i(x) \leq b_i \end{cases} \quad (2.4)$$

$$d_i^- = \begin{cases} b_i - f_i(x) & ; f_i(x) < b_i \\ 0 & ; f_i(x) \geq b_i \end{cases} \quad (2.5)$$

İstenmeyen sapma deęişkenleri yukarıda (2.1), (2.2) ve (2.3) ile açıklanan durumlara baęlı olarak ařaęıdaki gibi belirlenir.

- (2.1) řeklinde ise d_i^+
- (2.2) řeklinde ise d_i^-
- (2.3) řeklinde ise d_i^- ve d_i^+

Sapma deęişkenlerinin ait oldukları hedef fonksiyonlarının sol tarafına eklenmesi ile elde edilen hedef kısıtları, herhangi bir i. amaç için ařaęıdaki gibidir.

$$f_i(x) + d_i^- - d_i^+ = b_i \quad (2.6)$$

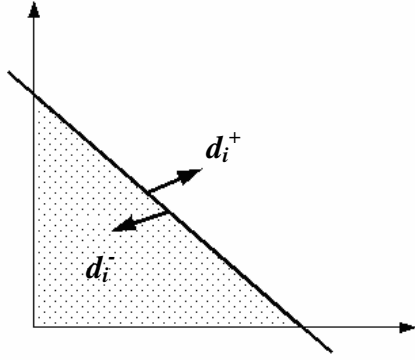
Hedef programlama modelinde, hedefler için belirlenen erişim düzeylerinden meydana gelebilecek sapmaların minimizasyonu istenir. Buna göre, yukarıda belirtilen tüm ilişkiler özet olarak ařaęıdaki tabloda verilmiřtir

Hedef Tipi	Hedef Programlama Formu	Minimize Edilecek Sapma Deęişkenleri
$f_i(x) \leq b_i$	$f_i(x) + d_i^- - d_i^+ = b_i$	d_i^+
$f_i(x) \geq b_i$	$f_i(x) + d_i^- - d_i^+ = b_i$	d_i^-

$f_i(x) = b_i$	$f_i(x) + d_i^- - d_i^+ = b_i$	$d_i^- - d_i^+$
----------------	--------------------------------	-----------------

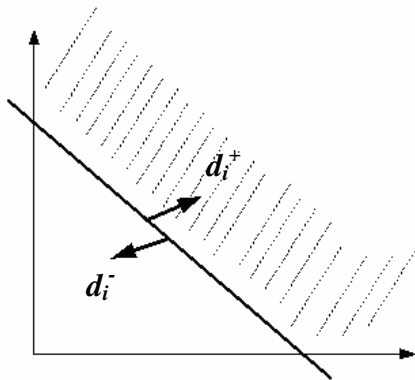
Tablo 2.1. Hedef tipi formülasyonları

Birinci durumda, negatif sapma değişkeni olan d_i^- 'nin 0 'dan olabildiğince büyük olması ve pozitif sapma değişkeni olan d_i^+ 'nin 0 değerini alması istenir (Şekil 2.1) .

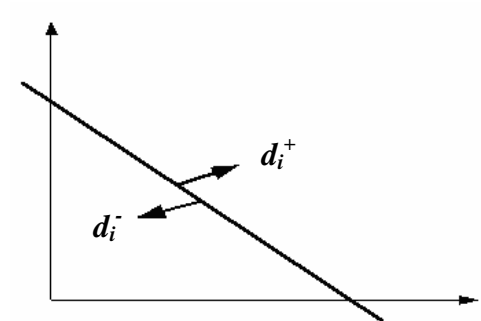


Şekil 2.1 Küçük-eşit biçimindeki hedef

İkinci durumda, negatif sapma değişkeni olan d_i^- 'nin 0 ' a yaklaşmasını ve hatta 0 değerini alması, pozitif sapma değişkeni olan d_i^+ 'nın da olabildiğince 0 'dan büyük olması istenir (Şekil 2.2) . Üçüncü durumda, hedef fonksiyonunun belirlenen erişim düzeyine eşit olması istersek, negatif ve pozitif sapma değişkenininin 0'a olabildiğince yakın olması istenir (Şekil 2.3).



Şekil 2.2 Büyük-eşit biçimindeki hedef



Şekil 2.3 Eşit biçimindeki hedef

2.6.1. Hedef Programlama Modelinde Kullanılan Amaç Fonksiyonlarının Farklı Tipleri

2.6.1.1. Eşit Ağırlıklı Amaç Fonksiyonu

$$\text{Min } Z = \sum_{i=1}^n d_i^- + d_i^+ \quad (i = 1,2,3,\dots,n) \quad (2.13)$$

Bu eşitlikte Z ,negatif ve pozitif sapmaların toplamının minimumudur.Bu tür amaç fonksiyonu, sapma değişkenleri için herhangi bir ağırlıklandırma ve ya öncelik söz konusu olmadığında kullanılır.

2.6.1.2. Öncelikli Amaç Fonksiyonu

$$\text{Min } Z = \sum_{i=1}^n P_k (d_i^- + d_i^+) \quad (k = 1,2,3,\dots,k) \quad (i = 1,2,3,\dots,n) \quad (2.14)$$

Bu amaç fonksiyonunda ; k tane hedefin her biri için P_k öncelikleri kullanılır.Hedefler önceliklerine göre sıralanmak istendiğinde bu tip amaç fonksiyonu kullanılmaktadır.Bu ilişki matematiksel $P_1 > P_2 > P_3 > \dots > P_{n-1} > P_n$ şeklinde ifade edilir.Amaç fonksiyonunun oluşturulabilmesi için en önemliden daha az önemliye doğru sıralanan hedefler, ilk önce birinci öncelikli hedefin karşılanmasını daha sonra sırasıyla diğer hedeflerin karşılanması gerektirir.

2.6.1.3. Ağırlıklı – Öncelikli Amaç Fonksiyonu

$$\text{Min } Z = \sum_{i=1}^n W_k P_k (d_i^- + d_i^+) \quad (k = 1,2,3,\dots,k) \quad (i = 1,2,3,\dots,n) \quad (2.15)$$

Bu amaç fonksiyonunda ise; hedefler önceliklerine göre sıralanır ve sapma değişkenleri ağırlıklandırılır. Ağırlıklandırma $W_k \in [0,1]$ ile gösterilir ve k'ncı seviyede i' inci hedeften oluşan sapmaya ilişkin matematiksel ağırlık olarak ifade edilir. Burada , W_k ile ifade edilen ağırlıkların toplam olarak 1'e eşit olması gerekir.

Yukarıda belirtilen amaç fonksiyonlarından hangisinin uygulama kullanılacağı , problemin durumuna göre belirlenir.

Eğer, problemde hedeflere herhangi bir öncelik sıralaması yapma ihtiyacı duyulmuyorsa birinci amaç fonksiyonu kullanılır. Fakat, hedeflerin önceliklerine göre sıralanması istenirken, sapma değişkenleri için bir sıralama yapmak istenmiyorsa ikinci amaç fonksiyonu kullanılır.

Üçüncü amaç fonksiyonu ise hem hedeflerin hem de sapma değişkenlerinin farklılaştırılması gerekiyorsa kullanılır.

2.7. DOĞRUSAL HEDEF PROGRAMLAMA

Üçüncü bölümde ele alınan problem, bir doğrusal hedef programlama problemidir. Bu nedenle doğrusal hedef programlama daha ayrıntılı incelenecektir.

2.7.1. Doğrusal Hedef Programlamanın Varsayımları

Lee, doğrusal hedef programlamanın varsayımlarını dört ana başlık altında incelemektedir [. Bu varsayımlar oransallık, toplanabilirlik, sınırlılık, bölünebilirlik ve belirlilik varsayımlarıdır.

2.7.1.1. Doğrusallık Varsayımı

Doğrusallık varsayımı ile girdiler ile çıktılar arasında aynı yönlü doğrusal bir ilişkinin olduğu ifade edilir. Diğer bir ifadeyle girdiler artarken-azalırken çıktılarda aynı oranda artar-azalır.

2.7.1.2. Toplanabilirlik Varsayımı

Çeşitli faaliyetler tarafından kullanılan kaynakların toplam kullanımı ve elde edilen toplam katkı, her bir faaliyet tarafından ayrı ayrı kullanılan kaynakların toplamı ve bunların ayrı ayrı yarattıkları katkıların toplamına eşit olmalıdır. Bu eşitlik sağlanıyor ise toplanabilirlik varsayımı da sağlanıyor demektir.

2.7.1.3. Sınırlılık Varsayımı

Sınırlı kaynaklarla karar verme durumunda olan kişi, kontrol etme şansına sahip olduğu ekonomik değerlerin sınırlarını belirten kısıtlayıcı koşullar altında karar almak durumundadırlar.

Çözülmesi istenilen problem, mevcut sınırlı kaynakların optimal bir şekilde dağılımını sağlamaktadır. Problemin çözümünde kullanılacak kaynaklar sonludur. Bu nedenle probleme giren kaynaklar kısıtlanır.

2.7.1.4. Negatif Olmama Varsayımı

Doğrusal programlama gerçek problemlere uygulanır. Bu nedenle değişkenler negatif olamazlar.

Doğrusal hedef programlama yöntemi, ancak modelde kullanılan değişkenlerin tümünü pozitif değerler aldığı durumlarda kullanılabilir. Problemin çözülebilmesi için modeldeki tüm değişkenlerin, yani karar ve sapma değişkenlerinin değerleri sıfır veya sıfırdan büyük olması gereklidir. Doğrusal hedef programlama modelinde, doğrusal programlama modellerinde olduğu gibi negatif olmama koşulu bir varsayım olmasına rağmen başlangıçta bu varsayımı sağlamayan durumlar varsa diğer bir ifade ile herhangi bir değişken negatif değer alabiliyorsa, bu değişken ancak negatif olmayan iki değişkenin farkı olarak yazılarak modele alınabilir. Çözümde bu farkın oluşturduğu yeni değişkenler kullanılır. Çözüm sonucunda bulunan değerler yerine konularak değişkenin gerçek değeri bulunur.

2.7.1.5. Amaçlara Öncelik Verilmesi

Doğrusal hedef programlamanın en önemli özelliklerinden biri belirlenen hedefler arasında, önceliklere göre bir sıralama yapma olanağının bulunmasıdır. Doğrusal hedef programlama modelinde her bir amaca veya amaç grubuna belli bir öncelik

verilebilir. İlk amacın önceliği P_1 ile gösterilir. Daha sonraki amaçlar da sırasıyla (P_2, P_3, \dots, P_K) öncelikleri ile tanımlanır.

Model daha sonra, yüksek öncelikli hedefin optimum değerinin küçük öncelikli hedef tarafından kötüleştirilmesine izin vermeyecek şekilde, her seferinde bir hedef değerini optimum kılar.

2.7.1.6. Amaçların Ağırlıklandırılması

Doğrusal hedef programlama modelindeki sapmaların önem dereceleri birbirinden farklı olabilir. Böyle durumlarda sapmalara ağırlık değerleri verilebilir. Bu ağırlıklar her bir sapmanın diğerlerine oranla göreceli olarak önemini gösterir. Model de tek bir amaç fonksiyonu, problemin hedeflerin temsil eden fonksiyonların ağırlıklandırılmış toplamı haline getirilir.

Amaçların ağırlıklandırılması yaklaşımı özellikle sapmaların boyutları birbirinden farklı olduğu durumlarda kullanılır. Bu yaklaşımın belli başlı iki zorluğu bulunmaktadır. Birinci hedeflerin ağırlıklandırılması işleminin zor olması , ikincisi ise ağırlıkların hem hedeflerin görece önemlerinin hem de sapmalar arasındaki boyutsal ilişkilerin ikisini birden açıklamasıdır.

2.7.2. Doğrusal Programlama ile Doğrusal Hedef Programlamanın Karşılaştırılması

2.7.2.1. Problemin Amacı

Doğrusal Programlama modelinde amaç, belli doğrusal eşitlik veya eşitsizliklerden oluşan kısıtlayıcı koşullar altında doğrusal bir amaç fonksiyonunu optimize etmektedir. Doğrusal Hedef Programlama modelinde ise amaç, belirlenen hedeflerden sapmaların toplamını minimize etmektir. Doğrusal Hedef Programlama modelinde, doğrusal programlamadan farklı olarak tek bir amaç yerine birbiri ile çelişebilen birden çok amaç aynı modelde bulunabilir.

Doğrusal Hedef Programlama, geleneksel tek amaç fonksiyonlu doğrusal programlamanın karşılayamadığı, kısıtlara ilişkin hedef sapmalarını, hedeflere ilişkin öncelikleri belirlediği gibi, amaç fonksiyonundaki değişkenlerin aynı ölçü biriminde olması koşulunu da aramamaktadır.

Doğrusal Hedef Programlama, karar vericiler açısından daha esnek bir yapıya sahip bir tekniktir. Sadece tek bir amaç yerine birden çok ve birbiri ile çelişebilen amaçları sağlamaya çalıştığı için karar vericilere daha çok amacı eşanlı olarak sağlama olanağı verir.

2.7.2.2. Kısıtlar

Doğrusal Programlama modelinde bütün kısıtlar eşit öneme sahiptir ve hepsi eşanlı olarak mutlaka sağlanmak zorundadır. Doğrusal Hedef Programlama modelinde ise hedefler modelde hedef kısıtları olarak yer alırlar ve belirlenen öncelik sıralarına yada ağırlıklarına göre sağlanmaya çalışılır. Doğrusal Hedef Programlama probleminin amaç fonksiyonunda aynı zamanda öncelikli hale getirilebilen sapma değişkenlerinin ağırlıklı toplamının minimize edilmesi istenir. Söz konusu hedefleri öncelikli hale getirme işlemi tercihi bir şekilde yapılır. Burada sapma değişkenleri için ağırlıklar (bir anlamda hedeflerin ağırlıkları), kendine denk düşen hedef değerlerinden her birim sapma için göreceli bir cezayı yansıtır. Ayrıca doğrusal programlama modelinde tek bir amaç mutlak olarak sağlanmaya çalışılırken, doğrusal hedef programlama modelinde en öncelikli hedeften başlanarak tüm hedefler sağlanılmaya çalışılır ama bütün amaçların sağlanması gerekmez.

2.7.2.3. Sapma Değişkenleri

Doğrusal Hedef Programlamada bulunan pozitif ve negatif sapma değişkenleri, doğrusal programlamadaki atıl ve artık kapasite değerlerini gösteren boş ve artık değişkenlere karşılık gelir. Sapma değişkenleri her bir hedef kısıtlaması için dahil edilir ve o hedefin olası eksik veya fazla başarısını yansıtır.

Hedef tam anlamıyla gerçekleşmiş ise sapma değeri sıfır olur. Hedef ulaşılamamışsa sapma negatif, hedef üzerinde bir başarı sağlanmışsa pozitif sapma söz konusu olur.

2.7.2.4. Amaç Fonksiyonu

Doğrusal programlamada amaç fonksiyonu maksimizasyon veya minimizasyon şeklinde olabilirken doğrusal hedef programlamada ise amaç fonksiyonu sadece minimizasyon şeklinde olabilir. Doğrusal hedef programlamada karar vericiler tarafından belirlenen hedefler, birer kısıt olarak modele girer. Kaynaklar üzerindeki sınırlamaları yansıtan kısıtlamalar ise modele aynen herhangi bir doğrusal programlama modeline katılacağı gibi dahil edilir.

Doğrusal hedef programlamanın amaç fonksiyonunda karar değişkenleri olan X_i ler bulunmaz. Amaç fonksiyonunda sadece minimize edilmesi istenilen hedeflerden sapmaları ifade eden d_i^- ve d_i^+ (sapma değişkenleri) yer alır.

2.7.2.5. Hedefler

Doğrusal programlama amaç fonksiyonu, kar maksimizasyonu veya maliyet minimizasyonu olarak düzenlendiğinden daha önceden herhangi bir hedef belirlenmez. Doğrusal Hedef Programlamada ise gerçekleştirilmesi istenilen hedef değerlerinin önceden belirlenmesi gereklidir. Doğrusal programlamada sağ yan sabitleri maksimizasyon problemlerinde üst sınır, minimizasyon problemlerinde ise alt sınır olarak işlem görür.

2.7.2.6. Değişkenler

Doğrusal programlamada karar değişkenlerinin tümü sıfır ya da sıfırdan büyük değerler almak zorundadır. Doğrusal hedef programlama modelinde de doğrusal programlama modeline benzer şekilde bütün değişkenleri yani hem karar değişkenleri (X_i) hem de sapma değişkenleri (d_i^+, d_i^-) ya sıfıra eşit ya da sıfırdan büyük değerler alması gerekir. Ayrıca, aynı hedefe ait sapma değişkenlerinin en az birinin sıfır olma zorunluluğu vardır. Diğer bir ifade ile aynı hedefe ait sapma değişkenlerinin ikisi birden sıfırdan farklı olamaz . Yani (d_i^+, d_i^-) ifadesinin değeri

sıfıra eşit olma durumundadır. Bu durum modele bir kısıt olarak eklenir. Bu kısıt iki değişkenden en az birinin sıfır olması gerektiğini gösterir.

2.7.2.7. Çözüm

Doğrusal Hedef Programlama modelinin çözümü sırasında doğrusal programlama modelinde olduğu gibi simpleks algoritması kullanılır. Doğrusal hedef programlama modelinde temel çözüme girecek değişken, yine doğrusal programlama modelinde olduğu gibi en yüksek önceliğe sahip olan ve en yüksek gelişmeyi sağlayacak değişken olmalıdır.

Ancak Doğrusal Hedef Programlama modelinde hedeflerin tümü her problem için gerçekleştirilemeyebilir. Doğrusal programlama modelinde ise bunun aksine ulaşılmak istenilen çözüm, optimum çözümdür. Yapılan çözümler sonucu doğrusal programlama modeli ile optimum çözüm bulunamaması olasılığı var iken Doğrusal Hedef Programlama Modelinde hiçbir hedef gerçekleşme bile mutlaka uygun bir çözüm bulunur.

2.7.2.8. Öncelikler

Doğrusal programlama modelinde kısıtlar arasındaki herhangi bir öncelik durumu söz konusu değildir. Tüm kısıtlar eşit önceliklidir ve mutlaka sağlanmalıdır. Doğrusal hedef programlamada hedefler önemlerine göre listelenir. En önemli hedef ilk sırada yer alırken en az önemli hedef son sırada yer alır. P_1 en önemli hedefin öncelik düzeyini, ikinci derecede önemli hedefin öncelikli düzeyini gösterir. Birinci hedeften başlayarak sırası ile tüm amaçlar sağlanmaya çalışılarak çözüme devam edilir.

2.7.2.9. Ağırlıklar

Doğrusal Programlama birden çok amaç ve buna bağlı olarak birden çok hedef olmadığı için hedefler arasında bir ağırlık farklılığı da söz konusu değildir.

Doğrusal hedef programlama yönteminde ise istenmeyen her bir sapmaya belirli bir ağırlık verilir. Bu ağırlıklar her bir sapmanın nisbi önemini gösterir.

Özetle, doğrusal programlama ile doğrusal hedef programlama karşılaştırmasının temel noktaları Tablo 2.2.'de gösterilmiştir.

Maddeler	Doğrusal Programlama	Doğrusal Hedef Programlama
Amaç	Optimum	Tatmin edici
Hedefler	Öncelikli bir tane (en büyük veya en küçük yapılacak)	Her bir hedef için belirlenmiş bir değer
Kantitatif ifade	Doğrusal	Doğrusal
Yapı	Tek amaç , çok kısıt	Tek amaç,çok hedef,çok kısıt
Amaç fonksiyonu	Karar değişkenleri	Sapma değişkenleri
Kısıtlar	Tümü eşit önemli	Öncelikli, ağırlıklı

Tablo 2.2. : Doğrusal Programlama ile Doğrusal Hedef Programlama arasındaki temel farklar

3. UYGULAMA

3.1. GİRİŞ

Bu çalışmada amaç, öncelikle kalori değeri düşük olan, lif, bitkisel protein, kalsiyum ve demir bakımından zengin; ayrıca yağ, şeker, karbonhidrat miktarları düşük olan bir “Müsli” elde etmektir. Ayrıca bu ürünün mineraller ve Omega-3 bakımından da mümkün olduğunca zengin olması hedeflenmiştir (Müsli; tahıllı gıdalar, yulaf, buğday, pirinç gibi gıdalar ve bunlardan elde edilen ürünlerdir).

Ulaşılmak istenen birden fazla hedef olduğu için bu problemin çözümünde “hedef programlama” kullanılmıştır. Hedeflerin birbirlerine göre öncelikleri olduğundan “öncelikli hedef program” kullanılmıştır. Modelin çözüm aşamasında ise birinci bölümde anlatılan simpleks yöntemini temel alan WINQSB programı kullanılmıştır.

3.1.1. Lifin Yararları

Lifler sindirim sistemimiz tarafından sindirilemeyen ve genelde bitkisel hücrelerin duvarlarını oluşturan selüloz, hemiselüloz, pectin, lignin gibi maddelerden oluşur. Lifli (posalı) besinlerin ortak özellikleri, su emmeleri ve şişmeleridir. Yüksek miktarda su tutabilen pectin, psyllium, gums gibi lifler çözünebilir lifler olarak adlandırılırlar.

Lifli besinler bağırsaklarda sindirilemedikleri gibi ayrıca suda tutarak hacim oluştururlar. bağırsakların içi dolu iken içindekileri uzaklaştırmak için çalışma hızı ve itme gücü artar. Lifsiz besinler bazen bağırsaklarda 24 saatten fazla atılmadan dururlarsa da lifli besinler 24 saat içinde vücut dışına atılırlar. Bu çalışma ve bağırsaklardaki doluluk barsak kanserlerinin oluşumunu engellediği bilinmektedir.

En çok lif içeren besinlerin başında tahıllar gelir. Yulaf en çok çözünebilir lif içeren tahıldır. Yapılan metabolik çalışmalar besinlerine yulaf, arpa gibi tahılları ekleyenlerde ve yağ alımını durduranlarda kan kolesterol seviyesinin %15 lere kadar düştüğünü göstermiştir. Bu beslenme rejimine pectin ve bazı bitkisel reçineleri katanlarda %10 daha da düşebilmektedir. Ancak bu durumda sindirim sistemine ait bazı şikayetler görülebilmektedir. Beslenmede liflerin çoğalması kalp hastalıkları ve kalp krizi riskinde ciddi azalmalara neden olmaktadır.

Kolesterol üzerindeki liflerin etkisinin ne miktarlarda görüleceği konusunda çalışmalar sürmektedir. Günde 1 kase (56 gr.) yulaf alınmasının bu etkiyi sağladığı gösterilmiştir [29]. Ayrıca, yapılan birçok araştırmada, diyetle eklenen 10 gr lifli gıdanın kalp krizi ölümlerini yüzde 27 oranında azalttığı bulundu [30].

3.1.2. Vücutun Protein Gereksinimi

Yetişkin bir insan vücudunun %16'sı proteinden oluşur. Proteinler, vücutta sudan sonra en çok bulunan maddelerdir ve vücudumuzdaki biyolojik olayların yapıtaşlarıdır, Proteinin fonksiyonları yaşam sürecini olanaklı kılar.

Proteinler, amino asitlerden yapılmış çok büyük moleküllerdir, Birçok aminoasit vücutta üretilebilir, ancak 8 adet aminoasit vücutta üretilemez, bu aminoasitleri besinlerden almamız gerekir. 1 gram protein 4 kalori enerji sağlamaktadır. Sağlıklı bir beslenme için günlük enerji ihtiyacının % 10'unun proteinlerden karşılanması gerekmektedir.

Proteinin Fonksiyonları;

- Hareket
- Asit baz dengesi
- Sıvı dengesi
- Mesaj-sinyal iletme (besin fonksiyonlarına yardım)
- Bağışıklık sisteminin devamlılığı
- Büyüme, farklılaştırma
- İnsülin hücrelerimize glikoz-şeker alımını sağlar

Günlük protein gereksinimimizin 1/3'ünün hayvansal, 2/3'ünün bitkisel protein

kaynaklarından karşılanması gerekmektedir, Günlük Protein Gereksinimi Vücut Ağırlığının 0,8 g/kg'dır [31].

Protein miktarının yetersiz olması halinde, vücut kendi dokularını kullanır. Büyüme durur, ağırlık azalır, Protein yapısında olan doku, hücre, enzim gibi elemanların etkinliği azalır, Hastalıklara yakalanma riski artar, Özellikle büyüme çağında oluşacak protein eksikliği önemli sorunlar oluşturmaktadır.

3.1.3. Yağlar

Yağlarda proteinler gibi hücrelerin kendilerini yenilemelerine yardımcı olurlar. Yağlar günlük besinin en fazla %30'unu oluşturmaktadır. Ayrıca günlük olarak alınan doymamış yağ oranı günde alınan yağ oranının %10'unu geçmemelidir . Yağlar karbonhidratlara göre çok daha fazla kalori içerirler. Her gramı 9 kaloridir. Teorik olarak hiç yağ yemeden yaşanabilir. Çünkü vücut gereksinimi olan yağları karbonhidratlardan yapabilir. Ancak bazı yağların eksikliği sorun yaratabilir. Örneğin linoleik asit denen bir yağı vücut sentezleyemez. Bu yağın dışardan alınması gerekmektedir. Eksikliği halinde bol miktarda kolesterol damarlarda birikir.

Fındık, ceviz, badem gibi besin maddelerinin içinde bol miktarda posa, kolesterol düşürücü bitkisel sterol, folik asit, magnezyum, bitkisel Omega-3 yağları vardır. Bu bileşimler, özellikle içerdikleri doymamış yağlar nedeniyle kötü kolesterolü azaltma, iyi kolesterolü yükseltmede önemli bir rol üstlenmektedirler. 1999 yılında, hastalar üzerinde yapılan klinik çalışmanın sonucunda, günde 1 gram omega-3 yağ asidi alımının kalp krinden ölüm riskini azalttığı gözlenmiştir [32].

3.1.4. Fazla Şeker ve Karbonhidrat Tüketiminin Zararları

Vücut, kan şekerinin tümünü aynı anda enerjiye çevirememektedir. Kan şekeri düzeyi normalin üzerine çıktığında; pankreastan salınan insülin hormonu fazla şekerin depolanması için karaciğer, kas ve diğer hücreleri uyarır. Glikozun bir kısmı, kas ve karaciğerde glikojen şeklinde depolanır. İhtiyacından fazla enerji tüketimi durumunda vücut, bir kısım glikozu vücut yağına çevirir. Dolayısıyla obezite ve beraberindeki 40'ı aşkın hastalık için davetiye çıkartılmış olmaktadır. Bu nedenle karbonhidratları azı karar çoğu zarar mantığı ile değerlendirmekte yarar vardır. Son zamanlarda şeker kullanımının hızla artmasıyla birlikte kalp - damar hastalıkları,

diyabet, kanser, sindirim sistemi hastalıkları ve romatizmal hastalıkların görülme sıklıklarında artışlar olmaktadır.

Beyin, sinir sistemi ve alyuvarlar normal koşullarda enerji ihtiyaçlarını mutlak surette karbonhidratlardan karşılamak durumundadır. Bazı karbonhidratlar besinlerde doğal olarak bulunurlar (meyvelerde fruktoz, sütte laktoz, tahıllarda nişasta gibi). Bazıları ise sonradan ilave edilirler (sofra şekeri ve şeker içeren besinler). Kaynağı ne olursa olsun, vücut gerçekte bu farkı anlamaz. Karbonhidratlar büyük oranda bitkisel kaynaklı besinlerden alınmaktadır. Bu karbonhidratlar vücudumuzda yapıtaşı olan glikoza dönüşür ve kan şekerinin esas kaynağını oluştururlar. O nedenle Dünya Sağlık Örgütü günlük enerjimizin %55-60'ının karbonhidratlardan karşılanması gerektiğini vurgulamaktadır.

3.1.5. Kalsiyumun Gerekliliği

Kalsiyum vücudumuzda en fazla bulunması gereken mineraldir, ve bu yüzden bol miktarda almalıyız. Vücudumuzdaki kalsiyum miktarı vücut ağırlığımızın %1,5 yada %2 sidir. Bu demek oluyor ki kilosu 60 kg. olan bir bayanın vücudunda yaklaşık 1,2 kg. kalsiyum bulunmaktadır. Kalsiyum miktarının %99'u kemiklerimizde ve dişlerimizde bulunmaktadır, geriye kalan %1 lik kısım ise kanda, kaslarda ve diğer vücut dokularında bulunmaktadır. Kemikler canlı dokulardır. Dişler ise kireçlenmiş dokulardır ve yapılarında kemikler de olduğu gibi kalsiyum bulunmaktadır. Bu yüzden de kalsiyum vücut için son derece önemlidir.

1940'lı yıllarda, günlük kalsiyum gereksinimi hastalıkların önlenmesi için bir yol olarak kabul edilmekteydi. Fakat 50 yıl sonra kalsiyumun uzun tedavi sürecinde, ciddi hastalıkların ortaya çıkmasını önlemede ve sağlığımızı korumak için çok önemli olduğu kanıtlanmıştır. 1997'de Ulusal Bilim Akademisi yetişkinlerin günlük kalsiyum ihtiyacını 800 mg'dan 1000 mg'a çıkartmıştır [33].

3.1.6. Demir Eksikliği

Dünya üzerinde pek çok insanı etkileyen önemli bir sağlık sorunu olan demir eksikliğini, sağlıklı yiyecekler ile yeterli ve dengeli beslenerek önleyebilirsiniz.

Demir yetersizliği anemisi dünyada iki milyar insanı etkileyen önemli bir halk sağlığı sorunu. Bu sorun özellikle 0-5 yaş grubu çocukları, adölesan dönemindeki kızlar ve doğurganlık çağındaki kadınları etkiliyor.

Ülkemizde okul öncesi çocukların yarısı, okul çağı çocukların üçte biri, gebe ve emziren kadınların yarısında ne yazık ki kansızlık görülüyor. Büyüme ve gelişme geriliği, deri renginin solukluğu, huzursuzluk, iştahsızlık, kaşık tırnak, baş dönmesi, baş ağrısı, uykusuzluk, davranış bozuklukları, enfeksiyonlara karşı direnç azalması, üşüme, titreme, konsantrasyon bozukluğu, pika ve geofaji (buz ve toprak yeme) demir eksikliğinin başlıca belirtileri arasında sayılabilir.

Yetişkin bir bireyin vücudunda ortalama 3-5 gr civarında demir bulunmaktadır. Bunun çoğunluğu da kanda, kırmızı hücrelerin hemoglobin kısmındadır. Hücrelerdeki demir besin öğelerinin enerjiye dönüşmesinde görevli enzim sisteminin bir kısmıdır. Kırmızı kan hücrelerindeki demirin görevi ise oksijen taşımaktır.

3.1.7. Potasyum

Potasyum suyun bedenden atılmasını sağlar. Kalp kasıyla diğer kasların kasılmasında rolü büyüktür. Buğday, taze fasulye, ıspanak, patates, kuru kayısı, erik ve bademde bulunur. Günlük ihtiyaç 2000-3000 mg.'dir.

Potasyum (K) hücreler içinde bulunan katyon madenlerin en başında yer alır. Hücre dışı sıvılarda en çok sodyum, hücre içinde en çok potasyum bulunur. Potasyum kasların işlevinde çok önemli bir yer tutar. Özellikle kalp kasının düzenli çalışması vücuttaki, kalp kası hücresindeki ve kandaki potasyum düzeyleriyle çok yakından ilgilidir.

Potasyum, hücreler içindeki ozmotik basıncın korunmasına katkıda bulunarak hücre içindeki sıvının dışı kaçmasının önlenmesini sağlamaktadır. Hücre içinde fazla düzeyde potasyum bulunması aynı zamanda hücrelerdeki protein sentezi için de gereklidir.

Besinler yoluyla günde 4 mg kadar potasyum alınmaktadır. Yukarıdaki tabloda çeşitli besinlerin potasyum yönünden değerleri verilmiştir.

Potasyumun vücuttan en önemli atılım yolu böbrekler aracılığıylaadır. Sağlıklı bir biçimde çalışan böbrekler, vücuttaki fazla potasyumu kolaylıkla atabilirler.

3.1.8. Çinko

Bir çok besinde fazlasıyla bulunan çinko minerali bağışıklık sisteminde anahtar rolü oynar, zinde yapar, verimli yapar. Akyuvarların, antikorların oluşmasında payı vardır. Bağışıklık sisteminin bu askerleri bizi hastalığa neden olan virüslerden koruduğu gibi zehirli maddeleri de zehirsiz hale getirmede yardımcı olur. Bağışıklık sisteminin düzenli çalışabilmesi için vücutta bol miktarda çinko bulunması gerekiyor. Yaraların iyileşmesi, görme duyusunun güçlenmesi diyabet hastalığı, böbrek hastaları, çinko eksikliği tehlikesiyle karşı karşıyadırlar. Çinko eksikliği sizi enfeksiyon hastalıklarına karşı savunmasız bırakır. Ayrıca tat ve koku duyularını da zayıflatır. Yetişkinler için günde önerilen miktar 15 mg dır.

Özellikle enfeksiyonlara karşı zayıf olma çinko azlığının en önemli belirtisidir. Bunun dışında çocuklarda öğrenme yeteneğinin sınırlı olması, büyümedeki aksaklıklar, saç dökülmesi, kısırlık, soğuk eller ve ayaklar, koklama ve tat alma duyusunun sınırlı olması, tırnaklarda beyaz lekeler de çinko azlığının diğer belirtileridir.

3.1.9. Magnezyum

Magnezyum hayati önem taşıyan 11 mineralden birisi, belki de en önemlisidir. Vücudun kendisi bu minerali üretmediği için magnezyumun besinler yoluyla alınması gerekir. Magnezyum özellikle strese ve migrene karşı iyidir ve kalbi korur. Astım ve alerjik nezleyi hafifletir. Ayrıca cildi düzgünleştirir, saçı güzelleştirir,

tırnakları kuvvetlendirir. 300 enzimi çalıştırır ve bununla metabolizmayı etkilemiş olur.

Magnezyum toprakta ve deniz suyunda saklıdır. Vücudumuzda da sürekli doldurulması gereken bir magnezyum rezervi vardır. Yani bu mineralin sayısız fonksiyonlarını yerine getirebilmesi için vücuda devamlı olarak verilmelidir.

Yanlış beslenme, toprakta bu mineralin giderek azalması magnezyumun vücut tarafından yeteri kadar alınmamasına neden olur. Vücut bu minerali yeteri kadar almadığı takdirde mide, bağırsak bölgesindeki, idrar yollarında, baldırlardaki kramplar, kalp ritmindeki bozukluklar, boyunda ve omuzlarda kasılmalar veya sinirlilik, ellerde uyuşukluk ve karıncalanma, migren, dikkat azlığı, gürültüye karşı hassasiyet magnezyum eksikliğinin işaretleridir.

Bir kadının günde en az 300 mg magnezyuma ihtiyacı vardır [30]. Magnezyumun fazlasının zararı yoktur, çünkü fazlası vücuttan çıkar. Olabilecek yan etkiler, hafif ishal, hafif mide rahatsızlıkları.

3.1.10. Keten Tohumunun Yararları

Keten Tohumu; bağışıklık sistemini güçlendirir. Kalp-damar hastalıklarından korur. LDL kolesterol ve trigliserit seviyesini, yüksek tansiyonu düşürür. Romatizmal hastalıkları önler. Sinir sistemini ve hafızayı güçlendirir. Kan şekerini dengeler. Konsantrasyon bozukluğuna, yaşlanmaya bağlı dikkat dağınıklığına karşı iyi gelir. Ruhsal bozukluklara karşı iyi gelir.

Omega-3, omega-6 yağ asitlerinin iyi bir kaynağı olan keten tohumu, yüksek oranda çözünür ve çözünmez lif içerir, göğüs, kolon, prostat kanserine karşı koruyucu olan lignanların kaynağıdır. Uzmanlar günde 1 kaşık (yaklaşık 3 gr.) keten tohumu tüketilmesinin yeterli olduğunu belirtmektedir [34].

3.2. MODELİN KURULMASI

Bu çalışmanın uygulama kısmında, ABD Tarım Bakanlığı'nın Ulusal Beslenme Veritabanı (USDA - United States Department of Agriculture, National Nutrient Database) verilerinden yararlanılmıştır. Modelin kurulması aşamasında kısıtlar bilimsel veriler ve piyasada tercih edilen müsli çeşitlerinin besin değerlerinin ortalaması gözönüne alınarak oluşturulmuştur.

Hedef programlama modelinin unsurlarının oluşturulması aşağıda adım adım açıklanmıştır.

3.2.1. Karar Değişkenlerinin Belirlenmesi

Bu çalışmada, ele alınan 11 çeşit besin maddesinin, 100 gr. ürünün içeriğinde kaç gr. bulunması gerektiği karar değişkenleri olarak ele alınmıştır. Bu besin maddeleri seçilirken de piyasada tercih edilen müsli çeşitlerinin içeriği gözönüne alınmıştır.

x_1 : Yulaf

x_2 : Buğday Patlağı

x_3 : Keten Tohumu

x_4 : Kuru Kayısı

x_5 : Kuru Elma

x_6 : Kuru İncir

x_7 : Kuru Üzüm

x_8 : Badem

x_9 : Ay Çekirdeği

x_{10} : Fındık

x_{11} : Ceviz

3.2.2. Hedef ve Kısıtların Belirlenmesi

3.2.2.1. Hedef Kısıtlayıcılarının belirlenmesi

Enerji (kcal)

Ürünün enerjisinin referans alınan ortalama değerden daha düşük (veya eşit) olması hedeflenmiştir.

$$3,89x_1 + 3,22x_2 + 5,34x_3 + 2,41x_4 + 2,43x_5 + 2,49x_6 + 2,99x_7 + 5,97x_8 + 5,84x_9 + 6,49x_{10} + 6,54x_{11} \leq 358$$

Lif (cg)

Kısım 3.1.1.'de belirtilen nedenlerden dolayı, ürünün 100 gramında bulunan lif miktarının 10 gr'ın üzerinde olması hedeflenmiştir.

$$10,6x_1 + 15x_2 + 27,3x_3 + 7,3x_4 + 8,7x_5 + 9,8x_6 + 3,70x_7 + 11,8x_8 + 8,6x_9 + 9,4x_{10} + 6,7x_{11} \geq 1000$$

Protein (cg)

Ürünün 100 gramında bulunan protein miktarının referans alınan ortalama değerden daha çok olması hedeflenmiştir.

$$16,89x_1 + 15x_2 + 18,29x_3 + 3,39x_4 + 0,93x_5 + 3,3x_6 + 3,07x_7 + 22,09x_8 + 20,78x_9 + 15,03x_{10} + 15,23x_{11} \geq 1130$$

Yağ (cg)

Ürünün 100 gramında bulunan yağ miktarının referans alınan ortalama değerden daha az olması hedeflenmiştir.

$$6,9x_1 + 1,6x_2 + 42,16x_3 + 0,51x_4 + 0,32x_5 + 0,93x_6 + 0,46x_7 + 52,83x_8 + 51,46x_9 + 62,4x_{10} + 65,21x_{11} \leq 650$$

Karbonhidrat (cg)

Ürünün 100 gramında bulunan karbonhidrat miktarının referans alınan ortalama değerden daha az olması hedeflenmiştir.

$$66,27x_1 + 77x_2 + 28,88x_3 + 62,64x_4 + 65,89x_5 + 63,87x_6 + 79,18x_7 + 19,27x_8 + 20x_9 + 17,6x_{10} + 13,71x_{11} \leq 6370$$

Şeker (cg)

Ürünün 100 gramında bulunan şeker miktarının referans alınan ortalama değerden daha az olması hedeflenmiştir.

$$1,55x_3 + 53,44x_4 + 57,19x_5 + 47,92x_6 + 59,19x_7 + 4,9x_8 + 2,62x_9 + 4,89x_{10} + 2,61x_{11} \leq 349$$

Kalsiyum (mg./100)

Ürünün 100 gramında bulunan kalsiyum miktarının, referans alınan ortalama değerden daha çok olması hedeflenmiştir.

$$54x_1 + 40x_2 + 255x_3 + 55x_4 + 14x_5 + 162x_6 + 50x_7 + 266x_8 + 78x_9 + 123x_{10} + 98x_{11} \geq 2000$$

Demir (mg./100)

Ürünün 100 gramında bulunan demir miktarının, referans alınan ortalama değerden daha çok olması hedeflenmiştir.

$$4,72x_1 + 5,73x_3 + 2,66x_4 + 1,4x_5 + 2,03x_6 + 1,88x_7 + 4,51x_8 + 5,25x_9 + 4,38x_{10} + 2,91x_{11} \geq 840$$

Diğer Mineraller (mg)

Potasyum :

$$4,29x_1 + 4,75x_2 + 8,13x_3 + 11,62x_4 + 4,50x_5 + 6,80x_6 + 7,49x_7 + 7,46x_8 + 6,45x_9 + 7,55x_{10} + 4,41x_{11} \geq 2000$$

Çinko:

$$3,97x_1 + 3,07x_2 + 4,34x_3 + 0,39x_4 + 0,20x_5 + 0,55x_6 + 0,22x_7 + 3,54x_8 + 5,00x_9 + 2,50x_{10} + 3,09x_{11} \geq 15$$

Magnezyum:

$$177x_1 + 133x_2 + 392x_3 + 32x_4 + 16x_5 + 68x_6 + 32x_7 + 286x_8 + 325x_9 + 173x_{10} + 158x_{11} \geq 200$$

3.2.2.2. Sistem Kısıtlayıcılarının Belirlenmesi

Omega-3 Kısıtı

Kısım 3.1.3.de belirtilen nedenlerden, ürünün 100 gramında bulunan omega-3 miktarının 1 gr.in üstünde olması istenmektedir.

$$22,81x_3 + 9,080x_{11} \geq 100$$

Keten Tohumu Kısıtı

Kısım 3.1.7’de belirtilen nedenlerden dolayı keten tohumu miktarının en az 3 gr. olması istenmektedir.

$$x_3 \geq 3$$

Yulaf Kısıtı

Kısım 3.1.1’de belirtilen nedenlerden dolayı yulaf miktarının en az 56 gr. olması istenmektedir.

$$x_1 \geq 56$$

Gramaj Kısıtı

Tüm veriler 100 gr. ürün için olduğundan, ürünün içeriğindeki maddelerin toplam ağırlığı 100 gr. olmalıdır.

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} = 100$$

3.3.MODELİN ÇÖZÜMÜ

Kısım 3.2.2.de hedef ve kısıtları verilen problemin öncelikli hedef programlama ile çözümünü gerçekleştirecek formülasyon aşağıdaki gibidir:

1.öncelikli hedef : Kalorinin ele alınan değerden düşük olması

2.öncelikli hedef : Lif miktarının 10 gr in üstünde olması

3.öncelikli hedef : Proteinin yüksek ; karbonhidrat , yağ , şeker oranlarının düşük olması

4.öncelikli hedef : Kalsiyum ve demirin yüksek olması

5. öncelikli hedef: Diğer minerallerin yüksek olması

$$\text{Min } P_1(d_1^+) + P_2(d_2^-) + P_3(d_3^- + d_4^+ + d_5^+ + d_6^+) + P_4(d_7^- + d_8^-) + P_5(d_9^- + d_{10}^- + d_{11}^-)$$

$$3,89x_1 + 3,22x_2 + 5,3x_3 + 2,41x_4 + 2,43x_5 + 2,49x_6 + 2,83x_7 + 5,97x_8 + 5,84x_9 + 6,49x_{10} +$$

$$6,54x_{11} - d_1^+ + d_1^- = 358$$

$$10,6x_1 + 15x_2 + 27,3x_3 + 7,3x_4 + 8,7x_5 + 9,8x_6 + 3,70x_7 + 11,8x_8 + 8,6x_9 + 9,4x_{10} +$$

$$6,7x_{11} - d_2^+ + d_2^- = 1000$$

$$16,89x_1 + 15x_2 + 18,29x_3 + 3,39x_4 + 0,93x_5 + 3,3x_6 + 3,07x_7 + 22,09x_8 + 20,78x_9 +$$

$$15,03x_{10} + 15,23x_{11} - d_3^+ + d_3^- = 1130$$

$$6,9x_1 + 1,6x_2 + 42,16x_3 + 0,51x_4 + 0,32x_5 + 0,93x_6 + 0,46x_7 + 52,83x_8 + 51,46x_9 +$$

$$62,4x_{10} + 65,21x_{11} - d_4^+ + d_4^- = 650$$

$$66,27x_1 + 77x_2 + 28,88x_3 + 62,64x_4 + 65,89x_5 + 63,87x_6 + 79,18x_7 + 19,27x_8 + 20x_9 +$$

$$17,6x_{10} + 13,71x_{11} - d_5^+ + d_5^- = 6370$$

$$1,55x_3 + 53,44x_4 + 57,19x_5 + 47,92x_6 + 59,19x_7 + 4,9x_8 + 2,62x_9 + 4,89x_{10} + 2,61x_{11} - d_6^+$$

$$+ d_6^- = 349$$

$$54x_1 + 40x_2 + 255x_3 + 55x_4 + 14x_5 + 162x_6 + 50x_7 + 266x_8 + 78x_9 + 123x_{10} + 98x_{11} - d_7^+$$

$$d_7^- = 32000$$

$$4,72x_1 + 5,73x_3 + 2,66x_4 + 1,4x_5 + 2,03x_6 + 1,88x_7 + 4,51x_8 + 5,25x_9 + 4,38x_{10} + 2,91x_{11} -$$

$$d_8^+ + d_8^- = 840$$

$$4,29x_1 + 4,75x_2 + 8,13x_3 + 11,62x_4 + 4,50x_5 + 6,80x_6 + 7,49x_7 + 7,46x_8 + 6,45x_9 + 7,55x_{10}$$

$$+ 4,41x_{11} - d_9^+ + d_9^- = 2000$$

$$3,97x_1 + 3,07x_2 + 4,34x_3 + 0,39x_4 + 0,20x_5 + 0,55x_6 + 0,22x_7 + 3,54x_8 + 5,00x_9 + 2,50x_{10} +$$

$$3,09x_{11} - d_{10}^+ + d_{10}^- = 15$$

$$177x_1 + 133x_2 + 392x_3 + 32x_4 + 16x_5 + 68x_6 + 32x_7 + 286x_8 + 325x_9 + 173x_{10} + 158x_{11} -$$

$$d_{11}^+ + d_{11}^- = 200$$

$$22,81x_3 + 9,080x_{11} \geq 100$$

$$x_3 \geq 3$$

$$x_1 \geq 56$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} = 100$$

Problemin öncelikleri, hedef ve kısıtları ihtiyaca göre değiştirilebilir. Yukarıda formüle edilen problemin WINQSB Paket Programı Veri girişi tablosu ve çözüm tablosu aşağıdadır:

Variable -->	Yulaf	Bugday	Keten	Kuru Kayısı	Kuru Elma	Kuru İncir	Kuru Üzüm	Badem	Ayçekirdeği	Fındık	Ceviz	d1+
Min:Hedef 1												
Min:Hedef 2												
Min:Hedef 3												
Min:Hedef 4												
Min:Hedef 5												
enerji	3.89	3.22	5.34	2.41	2.43	2.49	2.99	5.97	5.84	6.49	6.54	
lif	10.6	15.00	27.30	7.30	8.70	9.80	3.70	11.80	8.60	9.40	6.70	
protein	16.89	15.00	18.29	3.39	0.93	3.30	3.07	22.09	20.78	15.03	15.23	
yag	6.90	1.60	42.16	0.51	0.32	0.93	0.46	52.83	51.46	62.40	65.21	
karbonhidrat	66.27	77.00	28.88	62.64	65.89	63.87	79.18	19.27	20.00	17.60	13.71	
seker			1.55	53.44	57.19	47.92	59.19	4.90	2.62	4.89	2.61	
kalsiyum	54.00	40.00	255.00	55.00	14.00	162.00	50.00	266.00	78.00	123.00	98.00	
demir	4.72		5.73	2.66	1.40	2.03	1.88	4.51	5.25	4.38	2.91	
potasyum	4.29	4.75	8.13	11.62	4.50	6.80	7.49	7.46	6.45	7.55	4.41	
cinko	3.97	3.07	4.34	0.39	0.20	0.55	0.22	3.54	5.00	2.50	3.09	
magnezyum	177.00	133.00	392.00	32.00	16.00	68.00	32.00	286.00	325.00	173.00	158.00	
Omega-3			22.81								9.08	
yulaf	1											
keten			1									
gramaj	1	1	1	1	1	1	1	1	1	1	1	
LowerBound	0	0	0	0	0	0	0	0	0	0	0	
UpperBound	M	M	M	M	M	M	M	M	M	M	M	
VariableType	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous

Variable -->	d2+	d2-	d3+	d3-	d4+	d4-	d5+	d5-	d6+	d6-	d7+	d7-
Min:Hedef 1												
Min:Hedef 2			1									
Min:Hedef 3				1	1		1			1		
Min:Hedef 4												
Min:Hedef 5												
enerji												
lif	-1	1										
protein			-1	1								
yag					-1	1						
karbonhidrat							-1	1				
seker									-1	1		
kalsiyum												-1
demir												
potasyum												
cinko												
magnezyum												
Omega-3												
yulaf												
keten												
gramaj												
LowerBound	0	0	0	0	0	0	0	0	0	0	0	
UpperBound	M	M	M	M	M	M	M	M	M	M	M	
VariableType	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous

Variable -->	d7-	d8+	d8-	d9+	d9-	d10+	d10-	d11+	d11-	Direction	R. H. S.	
Min:Hedef 1												
Min:Hedef 2												
Min:Hedef 3												
Min:Hedef 4	1		1									
Min:Hedef 5					1		1		1			
enerji										=	358.00	
lif										=	1000.00	
protein										=	1130.00	
yag										=	650.00	
karbonhidrat										=	6370.00	
seker										=	349.00	
kalsiyum	1		1							=	32000.00	
demir		-1	1		-1	1				=	840.00	
potasyum										=	2000.00	
cinko							-1	1		=	15.00	
magnezyum									-1	1	=	200.00
Omega-3										>=	100.00	
yulaf										>=	56.00	
keten										>=	3.00	
gramaj										=	100.00	
LowerBound	0	0	0	0	0	0	0	0	0			
UpperBound	M	M	M	M	M	M	M	M	M			
VariableType	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous	Continuous			

Tablo 3.1. Problemin WINQSB Paket Programında Veri Girişi Tablosu

	Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Allowable Min. c(j)	Allowable Max. c(j)
1	Yulaf	65,01	0	0	0	0	0
2	Bugday	0	0	0	0	0	M
3	Keten Tohumu	3,69	0	0	0	0	0
4	Kuru Kayisi	24,74	0	0	0	0	0
5	Kuru Elma	0	0	0	0	0	M
6	Kuru İncir	0	0	0	0	0	M
7	Kuru Üzüm	4,82	0	0	0	0	0
8	Badem	0	0	0	0	0	M
9	Ayçekirdegi	0	0	0	0	0	M
10	Findik	0	0	0	0	0	M
11	Ceviz	1,74	0	0	0	0	0
	Goal	Value	(Min.) =	0			
	Goal	Value	(Min.) =	0			
	Goal	Value	(Min.) =	1.351,06			
	Goal	Value	(Min.) =	26.207,95			
	Goal	Value	(Min.) =	1.359,82			

Tablo 3.2. Problemin WINQSB Paket Programında Çözüm Tablosu

Bu sonuca göre, 100gr. ürünün içeriğinde 65.01gr. yulaf, 3.69gr. keten tohumu, 24.74gr. kuru kayısı, 4.82gr. kuru üzüm ve 1.74gr. ceviz bulunduğunda hedeflerden sapma miktarları minimum olmaktadır.

Birinci ve ikinci hedefler tam olarak sağlanmıştır. Bu hedeflerden sapma miktarı sıfırdır.

Protein, yağ, karbonhidrat ve şeker hedeflerinden toplam sapma yani üçüncü hedeften sapma 13.51 gr. dır.

Dördüncü ve beşinci hedeften sapma miktarları sırasıyla 0.26 gr. ve 1.36 gr. dır.

SONUŞ VE ÖNERİLER

Türkiye nüfusunun yüzde 27'sinin obezite sorunu yaşamaktadır. Dünya Sağlık Örgütü verilerine göre, 2010 yılında Türkiye'de her 10 kadından 6'sının ve her erkekten yüzde 3'ünün obez olmasının beklenmektedir. Obezite, yeni neslin en büyük problemidir.

Obezitenin bir sonucu olarak, koroner kalp hastalıkları, diyabet, hipertansiyon, bazı kanser türleri ve birçok metabolik sorun bireylerin yaşam kalitesini etkilemektedir.

Sağlıklı bir yaşamın temeli yeterli ve dengeli beslenme kurallarının yaşam biçimi haline getirilmesi ile mümkündür. Sağlıklı bir yaşam, kaliteli yaşama ve sağlıklı bir geleceğe zemin hazırlar.

Tüm bunları gözönünde bulundurarak bu çalışmada, dengeli beslenmeye katkı sağlayacak bir ürün geliştirilmesi problemi üzerinde çalışılmıştır.

İstenirse probleme maliyet kısıtı da eklenerek, aynı zamanda maliyeti en küçükleyecek bir çözüm aranabilir.

KAYNAKLAR

- [1] **Tikhomirov V. M.**, 1991. Stories About Maxima and Minima (Mathematical World- Volume 1), American Mathematical Society, USA.
- [2] **Clapham C., Nicholson J.**, 1996. The Concise Oxford Dictionary of Mathematics, 2nd Edition, Oxford University Press, New York.
- [3] **Gottfried, B. S., Weisman, J.**, 1973. Introduction to Optimization Theory, Prentice Hall Inc., New Jersey.
- [4] **Cooke, R.**, 1997. The History of Mathematics: A Brief Course, John Wiley & Sons Inc., Canada.
- [5] **Neumann, J., Morgenstern, O.**, 1944. Theory of Games and Economic Behavior, Princeton University Press.
- [6] **Gass, S.**, 2002. Great Moments in History, OR/MS Today, V. 29, N. 31.
- [7] **Sinha, S. M.**, 2006. Mathematical Programming: Theory and Methods, Elsevier Science & Technology.
- [8] **Çetin, E.**, Matematik Programlama, Bilim ve Teknik Üniversitesi (<http://www.biltek.tubitak.gov.tr/gelisim/matematik/kuralim.htm>) .
- [9] **Schrijver, A.**, 1998. Theory of Linear and Integer Programming, John Wiley.
- [10] **Cinemre, N.**, 2004. Doğrusal programlama, Üçüncü Baskı., Beta yayınları, İstanbul.
- [11] **Bulutay, T.**, 1965. Doğrusal Programlamaya Giriş, Ankara.

- [12] **Charnes A., Cooper W.W., Henderson A.**, 1961. An Introduction to Linear Programming, John Wiley & Sons, New York.
- [13] **Akbilgiç, O.**, 2005. Çok Amaçlı Karar Verme Teknikleri, MSGSÜ Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, İstanbul.
- [14] **Cinemre, N.**, 2004. Yöneylem Araştırması, İkinci Baskı, Beta yayınları, İstanbul.
- [15] **Baray Ş. A., Esnaf Ş.**, 2000. Taha, A. H. Operation Research an Introduction, 6.basımdan çeviri, Literatür Yayınevi, İstanbul.
- [16] **Kocadağlı, O.**, 2005. Doğrusal Hedef Programlama ile Bütçeleme, VII. Ulusal Ekonometri ve İstatistik Sempozyumu, İstanbul.
- [17] **Kosmidou K., Zopounidis C.**, 2004. Goal Programming Techniques for Bank Asset Liability Management, Kluwer Academic Publishers.
- [18] **Lee, S. M.**, 1972. Goal Programming for Decision Analysis, Auerback, Philadelphia.
- [19] **Ignizio, J. P.**,1978. A Review of Goal Programming: A Tool for Multiobjective Analysis, The Journal of the Operational Research Society, V. 29, N. 11, 1109-1119.
- [20] **Schniederjans, M. J.**, 1984. Linear Goal Programming, Petrocelli Books, New Jersey.
- [21] **Alp, S.**, 2008. Doğrusal Hedef Programlama Yöntemi Kullanılarak Kentiçi Otobüsle Toplutaşıma Sitemi İçin Bir Model Oluşturulması Ve Uygulanması, Marmara Üniversitesi Fen Bilimleri Enstitüsü Doktora Tezi, İstanbul.
- [22] **Erdoğan, Ş., Koç. E., Koç, O. İ.**, 2001. 0-1 Tamsayılı Hedef Programlama ve Diyet Problemine Uygulaması, 5. Ulusal Ekonometri ve İstatistik Sempozyumu, Çukurova Üniversitesi.

- [23] **Kazan, A.**, 2002. Eşit Öncelikli Hedef Programlama Modeli ile Bir Dengeli Beslenme Probleminin Çözümlemesi, *Mevzuat Dergisi*, 5(50).
- [24] **Gülenç, İ. F., Karabulut, B.**, 2005. Doğrusal Hedef Programlama ile Bir Üretim Planlama Probleminin Çözümü, *Kocaeli Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, Sayı 9, 1, 55-68.
- [24] **Anderson, A. M., Earle M. D.**, 1983. Diet Planning in the Third World by Linear and Goal Programming, *J. Operational Research Society*, V. 34, 9-16.
- [25] **Romero, C., Rehman, T.**, 1984. A Note on Diet Planning in the Third World by Linear and Goal Programming, *V.35, N.6*, 555-558.
- [26] **Lee, C. W., Kwak, N. K.**, 1999. Information Resource Planning for a Healty-care System using on AHP-based Goal Programming Method, *The Journal of the Operational Research Society*, V.50, N.12, 1191-1198.
- [27] **Yu, X., Luo, C.**, 2009. An Improved Graph Method for Linear Goal Programming, *Journal of Mathematics Research*, V.1, N.1, 31-34.
- [28] **Bal, H.**, 1986. Çoklu Amaçların Çözümlemesinde Amaç Programlaması ile Genelleştirilmiş Ters Yaklaşımı ve Yem Sanayiinde bir Uygulama, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*.
- [29] **Davidson, M. H.**, 1991. The Hypocholesterolemic Effects of Beta-Glucan in Oatmeal and Oat Bran: A Dose-Controlled Study, *JAMA*, 266(8), 1833-1839.
- [30] **Pereira, M. A., O'Reilly, E., Augustsson, K.**, 2004. Dietary Fiber and Risk of Coronary Heart Disease: A Pooled Analysis of Cohort Studies. *Arch Intern Med.*, 164(4), 370-376.
- [31] **Bilsborough, S., Neil, M.**, 2006. A Review of Issues of Dietary Protein Intake in Humans, *International Journal of Sport Nutrition and Exercise Metabolism* (16), 129-152.

- [32] **Lancet**, 1999. "Dietary supplementation with n-3 polyunsaturated fatty acids and vitamin E after myocardial infarction: results of the GISSI-Prevenzione trial, V.354, 447-455.
- [33] **Standing Committee on the Scientific Evaluation of Dietary Reference Intakes, Food and Nutrition Board, Institute of Medicine**, 1997. Dietary Reference Intakes for Calcium, Phosphorus, Magnesium, Vitamin D and Fluoride. Washington, DC: National Academy Press.
- [34] **Thompson, L., Cunnane, S.**, 1995. Flaxseed in Human Nutrition, First Printing, Amer Oil Chemists Society, Champaign, Illinois, U.S.A.

ÖZGEÇMİŞ

Kişisel

Adı Soyadı :Lale Karayeğit
Doğum Tarihi :07-03-1977
Doğum Yeri :İstanbul

Eğitim

1998-1994 : Mimar Sinan Üniversitesi, Fen Edebiyat Fakültesi,
Matematik Bölümü
1991-1994 : Otakçılar Lisesi

İş Deneyimi

2005-... : Bayrampaşa Tuna Lisesi, Matematik Öğretmeni
2001-2004 : Ş.Ö.M.G.İ.Ö.O., Matematik Öğretmeni
1998-2001 : Özel Evrim Lisesi, Matematik Öğretmeni