

# **ÇOK KULLANICILI OFDM İÇİN KAYNAK TAHSİSİ**

## **RESOURCE ALLOCATION FOR MULTIUSER OFDM**

**CAN TURGU**

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

ELEKTRİK ve ELEKTRONİK Mühendisliği Anabilim Dalı İçin Öngördüğü

YÜKSEK LİSANS TEZİ

olarak hazırlanmıştır.

2008

## ÇOK KULLANICILI OFDM İÇİN KAYNAK TAHSİSİ

**Can Turgu**

### ÖZ

Çok kullanıcıli bir sistemde, kullanıcılar aynı kanalı aynı anda OFDMA (Dikgen Frekans Bölüşümlü Çoğullama) tekniği ile paylaşabilirler. Bu senaryoda, ortak kullanılan kanal kaynaklarının (alt-taşıyıcılar ve güç) kullanıcılar arasında nasıl dağıtılacağı problemi OFDMA kaynak tahsisi olarak adlandırılmaktadır. Yapısı gereği doğrusal-olmayan bu problem aynı zamanda çokdorukludur ancak eniyilik bozulmadan doğrusal ve dışbükey hale getirilebilir.

Problem, ayrık doğası gereği tamsayı programlama ile çözülmeyi gerektirmektedir. Literatürde tamsayı programlama için değişik öneriler bulunmaktadır. Tez kapsamında Dallon ve Sınırla ile Dallon ve Kes Algoritmaları kaynak tahsisi probleminin eniyi çözümünü bulmak için kullanılmıştır.

Problemi çözenin diğeri bir yolu da değişkenlerin ayrık yapısını sürekli olacak şekilde gevşeterek ayrık probleme bir alt-sınır oluşturduktan sonra, yuvarlama veya en-yakın tamsayı çözümü bulma ile eniyiye yakın olurlu sonucu elde etmektir. Literatürde bu yolu izleyen çok sayıda çalışma bulunmaktadır. Tezde yapılan çalışmalarda, Lagrange Gevşetmesi ve Su-doldurma Algoritmalarının bir melezi incelenmiştir. Ayrıca bizim önerdiğimiz algoritma ile çok daha hızlı ve eniyiye yakın sonuçlar elde edilebilmektedir.

Tamsayı programlamada eniyiye yakın sonuç veren bir yöntem de Genetik Algoritmalar'dır. Tezde, bu algoritmalar OFDMA kaynak tahsisi problemine uyarlanarak başarılı sonuçlar elde edilmiştir.

Ayrıca, tez çalışmaları dahilinde MIMO (Çok Girişli Çok Çıkışlı)-OFDM kanallarda kaynak tahsisi ile ilgili bir çalışma yapılmış ve gelecekte yapılabilecek çalışmalar için önemli bilgiler toplanmıştır.

**Anahtar Kelimeler:** OFDM, OFDMA, çok-kullanıcıli sistemler, kaynak tahsisi, doğrusal programlama, tamsayı programlama, genetik algoritma.

**Danışman:** Yrd. Doç. Dr. Cenk TOKER, Hacettepe Üniversitesi, Elektrik ve Elektronik Mühendisliği Bölümü

## RESOURCE ALLOCATION FOR MULTIUSER OFDM

**Can Turgu**

### ABSTRACT

In a multiuser system, the user can share the common channel using the OFDMA (Orthogonal Frequency Division Multiple Access) technique. In this scenario, the issue of allocating the shared channel resources (subcarrier and power) among the users is called as the OFDMA resource allocation problem. By nature, this problem is non-linear and multimodal, however, it can be reformulated as a linear and convex problem after some manipulations.

Due to the discrete nature of the problem, it should be solved by Integer Programming. In the literature, there are several proposals for Integer Programming. In the context of this thesis, the Branch and Bound, and Branch and Cut Algorithms are utilized.

Another way of solving the problem is finding a lower-bound to the original problem by relaxing the discrete variables as continuous variables and, after that obtaining the feasible sub-optimum solution through rounding or finding the closest integer solution using the relaxation solutions. In literature, there are many studies in this direction. In the thesis, a hybrid of the Lagrange Relaxation and a variant of the Waterfilling Algorithm is investigated. Moreover, we propose an algorithm which can reach to a better sub-optimum solution within a much shorter time.

Another method for Integer Programming, which can give suboptimum solutions is the Genetic Algorithms. In the thesis, these algorithms are adapted to the resource allocation problem and successful results are obtained.

Apart from that, a study for resource allocation in MIMO (Multi-Input Multi-Output)-OFDM channels is also realized and important data is collected for possible future work.

**Keywords:** OFDM, OFDMA, multiuser systems, resource allocation, linear programming, integer programming, genetic algorithms.

**Advisor:** Assist. Prof. Dr.. Cenk TOKER, Hacettepe University, Department of Electrical and Electronics Engineering.

## TEŐEKKÜR

Tez alıřmalarım boyunca bana s¼rekli destek olup motivasyonumu kaybetmememi saęlayan tez danıřmanım Sayın Yrd.Do.Dr. Cenk Toker'e teőekk¼r ederim.

Bilgi ve deneyimlerini paylařan deęerli j¼ri ¼yelerine teőekk¼r ederim.

Bana g¼sterdikleri yakınlık ve yardımlarından dolayı t¼m Hacettepeli alıřma arkadaşlarıma teőekk¼r ederim.

Son olarak aileme bana verdikleri destek ve g¼sterdikleri sabırdan dolayı teőekk¼r ederim.

## İÇİNDEKİLER DİZİNİ

	<u>Sayfa</u>
ÖZ .....	i
ABSTRACT .....	ii
TEŞEKKÜR .....	iii
İÇİNDEKİLER DİZİNİ .....	iv
ŞEKİLLER DİZİNİ .....	vi
1 GİRİŞ .....	1
1.1 Çok Taşıyıcılı Kiplenim .....	1
1.2 Çoklu Erişim ve Kaynak Tahsisi .....	4
2 OFDMA SİSTEMLERDE KAYNAK TAHSİSİ .....	8
2.1 Sistem Modeli .....	10
2.2 Problem Tanımı .....	12
2.2.1 Tek Kullanıcılı Kanal İçin Kaynak Tahsisi .....	12
2.2.2 OFDMA Sistemlerde Kaynak Tahsisi .....	13
2.3 Literatürde Önerilen Diğer Yöntemler .....	15
3 DOĞRUSAL VE TAMSAYI PROGRAMLAMA .....	18
3.1 Doğrusal Programlamada Temel Çözümler .....	18
3.1.1 Simpleks Yöntemi .....	21
3.1.2 Doğrusal Tamsayı Programlama .....	26
4 TAMSAYI PROGRAMLAMA İLE KAYNAK TAHSİSİ .....	41
4.1 Kaynak Tahsisi Probleminin Doğrusallaştırılması .....	41
4.2 Tamsayı Programlama İle Kaynak Tahsisi .....	47
4.2.1 Önerilen Algoritma .....	47
5 GENETİK ALGORİTMA İLE KAYNAK TAHSİSİ .....	49
5.1 Genetik Algoritma .....	49
5.2 Genetik Algoritmanın OFDMA Sistemlere Uygulanması .....	52
6 ÇOK KULLANICILI MIMO OFDM SİSTEMLERDE KAYNAK TAHSİSİ ...	53
6.1 Sistem Modeli .....	53
6.2 Problem Tanımı .....	55
6.3 Tamsayı Programlama .....	56
6.3.1 Doğrusallaştırma .....	56

6.3.2	Tamsayı Programlama Algoritması .....	57
7	BENZETİM SONUÇLARI .....	60
7.1	OFDMA Sistemlerde Kaynak Tahsisi Benzetimleri .....	60
7.1.1	Tek Kullanıcı OFDM Sistemlerde Kaynak Tahsisi Benzetimleri .....	60
7.1.2	Çok kullanıcı OFDMA Sistemlerde Kaynak Tahsisi Benzetimleri .....	63
7.2	Çok Kullanıcı MIMO OFDM Sistemlerde Kaynak Tahsisi Benzetimleri ....	76
8	SONUÇLAR .....	79
	KAYNAKLAR .....	81

## ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
1.1 OFDM Blok Şeması. [1]. .....	3
1.2 Çoktaşıyıcı Kiplenim [7].....	5
2.1 OFDMA sistem modeli [7].....	10
3.1 Olurlu çözüm kümesi [12]. .....	20
3.2 $\mathbf{Ax} = \mathbf{b}$ denklem kümesinin doğrusal bağımsız 2, 4 ve 5'inci sütunlarının kullanılarak bir temel çözümün elde edilmesi. ....	21
3.3 Tamsayı Programlama ve gevşetilmiş problemin eniyi çözümleri [13]. ....	27
3.4 Dallar ve Sınırlar Algoritmasında olurlu kümenin ağaç yapısı ile dallandırılması [40]. .....	28
3.5 Olurlu çözüm kümesi [13]. .....	30
3.6 Ağaç yapısı[13]. .....	31
3.7 Problem 1 ve Problem 2 [13]. .....	31
3.8 Problem 3 ve Problem 4 [13]. .....	32
3.9 Problem 5 ve Problem 6 [13]. .....	32
3.10 Çözüm Kümesi [13]. .....	33
3.11 Birinci kesme [13]. .....	34
3.12 İkinci kesme [13]. .....	34
3.13 Olurlu çözüm kümesi. ....	36
3.14 Birinci kesme.....	37
3.15 İkinci kesme. ....	38
3.16 Son durum .....	39
4.1 Bit tahsis vektörünün açılımı. Vektör alt-vektörlerden oluşmaktadır. ....	42
5.1 Genetik Algoritma akış şeması.....	49
5.2 Tek noktalı çiftleştirme. ....	51
5.3 İki noktalı çiftleştirme. ....	51
7.1 Ağaçgözlü Algoritma ile Kaynak Tahsisi .....	61

7.2	Tek kullanıcıli OFDM senayosunda Açgözlü Algoritma ve Statik Metod ile Kaynak Tahsisi ( $N = 32, R = 64$ ) .....	62
7.3	Sürekli uzayda OFDMA kaynak tahsisi ( $K = 2, N = 32, M = 8, R_k = 32, \forall k$ ) .....	65
7.4	Farklı BER değeri için sürekli ve ayrik uzayda OFDMA kaynak tahsisi ( $K = 4, N = 64, M = 6, R_k = 32, \forall k$ ) Sürekli uzay için Lagrange Çarpanları, ayrik uzay için Dallon ve Sınırla Yöntemi kullanılmıştır. ....	66
7.5	Farklı kullanıcı sayıları için sürekli ve ayrik uzayda OFDMA kaynak tahsisi $BER = 10^{-3}, N = 64, M = 6, R_k = 32, \forall k$ ). .....	67
7.6	Statik Tahsis, Dallon ve Sınırla, Dallon ve Kes ile kaynak tahsisi $K = 4, N = 64, M = 6, R_k = 32, \forall k$ ). .....	68
7.7	Wong MAO, Önerilen Algoritma ile kaynak tahsisi ( $K = 4, N = 64, M = 6, R_k = 32, \forall k$ ) Tamsayı Programlama için Dallon ve Sınırla Algoritması kullanılmıştır. ....	70
7.8	Wong MAO, Önerilen Algoritma ve Statik tahsis yöntemleri ile kaynak tahsisi. $BER = 10^{-3}, N = 64, M = 6, R_k = 32, \forall k$ ). .....	71
7.9	Wong MAO, Önerilen Algoritma yöntemleri ile kaynak tahsisi. $K = 4, BER = 10^{-3}, M = 6, R_k = 32, \forall k$ ). .....	72
7.10	Önerilen Algoritma ve Genetik Algoritma ile kaynak tahsisi. $K = 4, N = 64, M = 6, R_k = 32, \forall k$ ). .....	74
7.11	Önerilen Algoritma ve Genetik Algoritma ile kaynak tahsisi. $BER = 10^{-3}, N = 64, M = 6, R_k = 32, \forall k$ ). .....	75
7.12	Genetik Algoritma nesil güç dağılımı .....	76
7.13	Verilen SER için ihtiyaç duyulan toplam güç, ( $K = 3, N = 8, M = 3, R_k = 6, \forall k$ ) .....	77
7.14	Verilen SER için ihtiyaç duyulan toplam güç, ( $K = 3, N = 8, M = 3, SER = 10^{-3}$ ) .....	78



## SİMGELER VE KISALTMALAR DİZİNİ

ARQ	: Automatic Repeat reQuest
BER	: Bit Error Rate
CCI	: Co-Channel Interference
CDMA	: Code Division Multiple Access
CSI	: Channel State Information
DVB	: Digital Video Broadcast
FDMA	: Frequency Division Multiple Access
HIPERLAN2	: High Performance LAN/2
(I)FFT	: (Inverse) Fast Fourier Transform
i.i.d.	: independent and identically distributed
LAN	: Local Area Network
M-QAM	: M-ary Quadrature Amplitude Modulation
MA	: Margin Adaptive
MAC	: Media Access Control
MIMO	: Multiple Input Multiple Output
MUD	: Multiuser Detection
OFDM	: Orthogonal Frequency Division Multiplexing
OFDMA	: Orthogonal Frequency Division Multiple Access
OSI	: Open Systems Interface
PSD	: Power Spectral Density
QoS	: Quality of Service
RA	: Rate Adaptive
SER	: Symbol Error Rate
SINR	: Signal to Interference and Noise Ratio
SISO	: Single Input Single Output
SNR	: Signal to Noise Ratio
SQP	: Sequential Quadratic Programming
SVD	: Singular Value Decomposition

## SÖZLÜK DİZİNİ

Alt-kanal	: Sub-channel
Alt-taşıyıcı	: Sub-carrier
Ana	: Primal
Artık Değişken	: Slack Variable
Aşağı Yönlü	: Downlink
Çifteş	: Dual
Çoklu Ortam	: Multimedia
Çokyollu	: Multipath
Çokyüzlü	: Polytope
Dallan ve Kes	: Branch and Cut
Dallan ve Sınırla	: Branch and Bound
Dışbükey	: Convex
Dikgen	: Orthogonal
Doğrusal	: Linear
Doğrusal Programlama	: Linear Programming
Doğrusal Bağımsız	: Linearly Independent
Döngüsel Önek	: Cyclic Prefix
Düğüm	: Node
Düzlem Kesme	: Cutting Plane
Enbüyük	: Maximum
Enbüyütme	: Maximization
Enküçük	: Minimum
Enküçültme	: Minimization
Eniyi	: Optimum
Eniyiye yakın	: Sub-Optimum
Evreuyumlu	: Coherent
Gevşetme	: Relaxation
Gürbüz	: Robust
Hüzme Oluşturmak	: Beamforming
İçsel Çarpım	: Scalar Product
İlinti	: Correlation

İşleç	: Operator
İşlenen	: Operand
İşlev	: Function
Kesme	: Cut
Kısıt	: Constraint
Kipçözücü	: Demodulator
Kiplenim	: Modulation
Maliyet	: Cost (Objective)
Olurlu	: Feasible
Olursuz	: Infeasible
Özdeğer	: Eigenvalue
Özvektör	: Eigenvector
Pivotlama	: Pivoting
Simpleks	: Simplex
Tahsis	: Allocation
Tamsayı Programlama	: Integer Programming
(Tam) Kerte	: (Full) Rank
Tekil	: Singular
Temel Olurlu	: Basic Feasible
Uyarlamalı	: Adaptive
Uygunluk	: Fitness
Yinelemeli	: Recursive

## 1. GİRİŞ

Sivil telsiz iletişim sistemler, başlangıçta sadece ses ve yazı tabanlı mesajlaşma için kullanılırken günümüzde çoklu ortam (multimedia) iletişimi için de kullanılmaktadırlar. Kullanıcıların her geçen gün artan daha yüksek veri hızı ve daha yüksek veri hızında iletişim talepleri için yüksek başarıma sahip çeşitli sistemler geliştirilmiştir ve geliştirilmeye devam edilmektedirler. Ancak tüm sistemlerde, kullanıcıların ortaklaşa erişmeye çalıştıkları iletişim kanalı, bu gelişmenin önünde bir engel teşkil etmektedir.

### 1.1 Çok Taşıyıcılı Kiplenim

Çok taşıyıcılı kiplenimde mevcut geniş bant-aralığı çok sayıda birbirinden bağımsız darbantlı alt-kanallara bölünmektedir. Böylece girişimden arınmış, yavaş iletişim yapan alt-kanalların toplamdaki veri hızı çok yüksek olabilmektedir. Tek-taşıyıcılı sistemin tersine güçlü bir kanal eşitleyicisine ihtiyaç yoktur fakat geniş bant aralığını, daha dar parçalara bölmek için ek işlemlere ihtiyaç vardır.

Veri hızının  $R$  bps ve mevcut bantgenişliğinin  $B$  Hz olduğu durumlarda (bir iletişim sisteminde) kanalın ilinti bantgenişliğinin ( $B_c$ ) mevcut bantgenişliğinden daha dar olduğu durumlarda ( $B_c < B$ ) kanal frekans seçicidir. Eğer mevcut bantgenişliği  $N$  eşit parçaya bölünürse ( $B_N = B/N$ ), ve  $N$  yeteri kadar büyük seçilirse,  $B_N \ll B_c$  olacaktır. Bu durumda ise alt-kanallar göreceli olarak frekans-düz sönümlenmeye maruz kalacaktır.

Pratikte, farklı alt-kanallar Ayırık Fourier Dönüşümü (Discrete Fourier Transform, DFT) ile oluşturulabilmektedir. DFT'nin etkin bir gerçekleştirilme yöntemi ise Hızlı Fourier Dönüşümüdür (Fast Fourier Transform, FFT).

$x[n]$ ,  $0 \leq n \leq N-1$  ayırık zamanda bir diziyi temsil etsin. Bu dizinin N-noktalı DFT'si

$$DFT\{x[n]\} = X[i] \triangleq \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j \frac{2\pi ni}{N}}, 0 \leq i \leq N-1$$

, ters DFT ise

$$IDFT\{X[i]\} = x[n] \triangleq \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} X[i] e^{j \frac{2\pi ni}{N}}, 0 \leq n \leq N-1$$

olarak ifade edilmektedir.

Bilindiği gibi zaman boyutundaki iki sinyalin evrişimi, sinyallerin Fourier dönüşümü sonrası frekans boyutundaki çarpımlarıyla ilişkilidir. Bu ilişki benzer bir şekilde DFT için de geçerlidir, ancak bu sefer döngüsel evrişim kullanılır.

$$y[n] = h[n] * x[n] = x[n] * h[n] = \sum_k h[k]x[n - k]$$

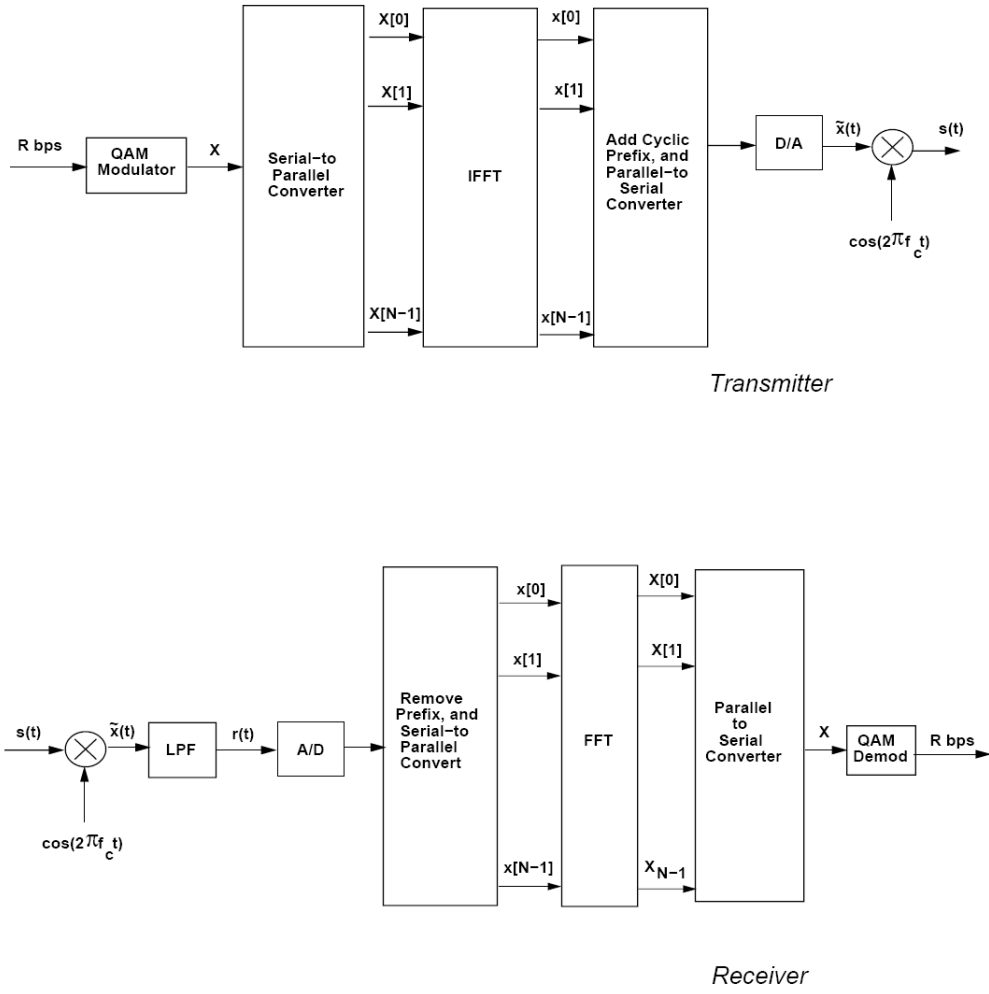
$$y[n] = h[n] \otimes x[n] = x[n] \otimes h[n] = \sum_k h[k]x[n - k]_N$$

$[n - k]_N$ ,  $[n - k]$  modulo  $N$  anlamına gelir. Dolayısıyla  $x[n - k]_N$ 'nin aslında  $x[n - k]$ 'nin periyodik hali olduğu açıktır. Frekans boyutundaki çarpma şu şekilde ifade edilebilir

$$DFT\{y[n] = x[n] \otimes h[n]\} = X\{i\}H\{i\}, 0 \leq i \leq N - 1.$$

Bir iletişim sisteminde veri bitleri  $x[n]$ 'lerin doğrudan kanal  $h[n]$ 'e verilmeleri doğrusal evrişime neden olur ve DFT'nin frekans boyutundaki çarpma özelliğini bozar. Bu sebeple  $x[n - k]$  periyodik veya döngüsel hale dönüştürülmelidir. Bu amaçla veri paketinin başına döngüsel önek (cyclic prefix, cp) eklenmektedir.

Dikgen Frekans Bölüşümlü Çoğullama'da (Orthogonal Frequency Division Multiplexing, OFDM) [1], [2] ise QAM modülasyonu ve DFT-IDFT işlemleri için FFT-IFFT çifti kullanılmaktadır. OFDM şeması Şekil 1.1'de gösterilmiştir.



Şekil 1.1: OFDM Blok Şeması. [1].

Şekil 1.1'deki işlemler matrisler vasıtasıyla açıklanabilmektedir. İlk olarak, vericide gönderilecek bitler IFFT dönüşümüne tabi tutulurlar:

$$\mathbf{X} = \mathbf{Q}\mathbf{x}$$

$$\mathbf{Q} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & \dots & W_N^{N-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \dots & W_N^{(N-1)^2} \end{bmatrix}$$

$$W_N = e^{-j\frac{2\pi}{N}}, \quad \mathbf{Q}^{-1} = \mathbf{Q}^H.$$

$x[i] \rightarrow x_i$  ifade deęişiklięi kullanılarak kanaldan alınan sinyal řu řekilde yazılabilir

$$\begin{bmatrix} y_{N-1} \\ y_{N-2} \\ \vdots \\ y_0 \end{bmatrix} = \begin{bmatrix} h_0 & h_1 & \cdots & h_\mu & 0 & \cdots & 0 \\ 0 & h_0 & \cdots & h_{\mu-1} & h_\mu & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_0 & \cdots & h_{\mu-1} & h_\mu \end{bmatrix} \begin{bmatrix} x_{N-1} \\ \vdots \\ x_0 \\ x_{-1} \\ \vdots \\ x_{-\mu} \end{bmatrix} + \begin{bmatrix} v_{N-1} \\ v_{N-2} \\ \vdots \\ v_0 \end{bmatrix} \quad (1.1)$$

$$\mathbf{y} = \mathbf{H}\mathbf{x} + v.$$

Fakat dongüsel önekteki semboller paketin sonundaki sembollerle aynıdır. Dolayısıyla

$$\begin{bmatrix} y_{N-1} \\ y_{N-2} \\ \vdots \\ \vdots \\ \vdots \\ y_0 \end{bmatrix} = \begin{bmatrix} h_0 & h_1 & \cdots & h_\mu & 0 & \cdots & 0 \\ 0 & h_0 & \cdots & h_{\mu-1} & h_\mu & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_0 & \cdots & h_{\mu-1} & h_\mu \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ h_2 & h_3 & \cdots & h_{\mu-2} & \cdots & h_0 & h_1 \\ h_1 & h_2 & \cdots & h_{\mu-1} & \cdots & 0 & h_0 \end{bmatrix} \begin{bmatrix} x_{N-1} \\ x_{N-2} \\ \vdots \\ \vdots \\ \vdots \\ x_0 \end{bmatrix} + \begin{bmatrix} v_{N-1} \\ v_{N-2} \\ \vdots \\ v_0 \end{bmatrix} \quad (1.2)$$

$$\mathbf{y} = \hat{\mathbf{H}}\mathbf{x} + v.$$

olmaktadır. Bu durumda (1.2) ifadesindeki  $H$  matrisi dongüsel hale gelmiştir. Vericideki IFFT ve alıcıdaki FFT dönüşümleri ( $\mathbf{X} = \mathbf{Q}\mathbf{x}$  ve  $\mathbf{Y} = \mathbf{Q}\mathbf{y}$  olmak üzere)  $y = \hat{H}x + v$  ifadesinde yerlerine konursa

$$\mathbf{Y} = \mathbf{\Lambda}\mathbf{X} + v_{\mathbf{Q}}$$

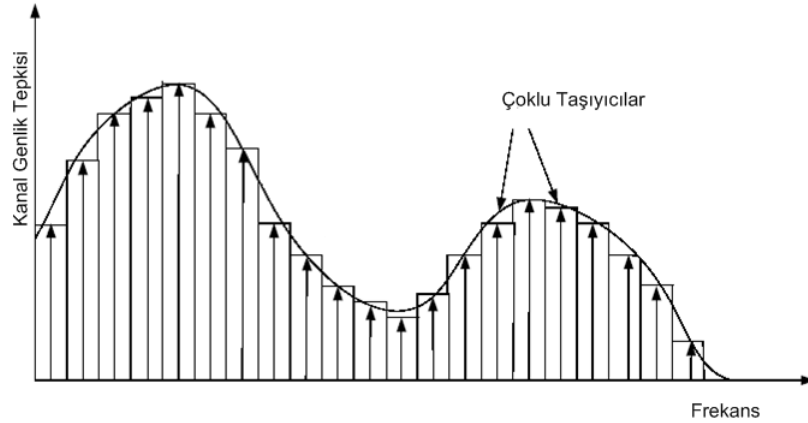
elde edilir. Burada elemanları  $\sigma_i$  olan  $\mathbf{\Lambda}$  köşegen bir matristir.  $\sigma_i$  katsayıları ise kanal katsayıları ile ilişkilidir.

## 1.2 Çoklu Eriřim ve Kaynak Tahsisi

Bir telsiz iletişim sisteminde ortak kanala erişmek isteyen kullanıcılar, kanalı paylaşmak durumundadırlar. Çok kullanıcılı sistemlerde kanala erişim için Frekans Bölüşümlü Çoklu Eriřim (Frequency Division Multiple Access, FDMA) [3], Zaman Bölüşümlü Çoklu Eriřim (Time Division Multiple Access, TDMA) [3] ve Kod

Bölüşümlü Çoklu Erişim (Code Division Multiple Access, CDMA) [4] [5], Uzay Bölüşümlü Çoklu Erişim (Space Division Multiple Access, SDMA) [6] [2] gibi yöntemler kullanılmaktadır. FDMA'de kanal, frekans alt-bantlarına bölünerek kullanıcılara paylaştırılmakta iken, TDMA'de kanal zaman bölgesinde parçalara bölünmekte ve kullanıcılara zaman aralıkları atanmaktadır. CDMA'de ise kullanıcılar kanala birbirinden farklı kodları kullanarak erişmektedirler. SDMA'de alıcı ve vericilerde birden fazla anten kullanılmakta ve uzay boyutunun vasıtasıyla kullanıcılar birbirinden ayrılarak sisteme erişimleri sağlanmaktadır. Alıcı ve vericide birden fazla antenin bulunduğu sistemler Çok Girişli Çok Çıkışlı (Multiple Input Multiple Output, MIMO) [6] sistemler olarak adlandırılmaktadır.

Bu tezde temel olarak FDMA ele alınacaktır. Şekil 1.2'de çok taşıyıcı (multicarrier) kiplenim kullanılmasıyla kanalın parçalara bölünmesi gösterilmektedir. Şekildeki alt-kanallar farklı kullanıcılara atanarak sisteme erişimleri sağlanmaktadır.



Şekil 1.2: Çoktaşıyıcı Kiplenim [7].

OFDM ile kanalın parçalara ayrılması işlemi gerçekleştirilebilmektedir. OFDM, IEEE 802.11a/g kablosuz yerel alan bağı (Local Area Network, LAN) [8], sayısal video yayını (Digital Video Broadcasting, DVB) [9], ve yüksek performanslı radyo yerel alan ağı (High Performance Radio LAN/2, HIPERLAN2) [10], WiMAX (Worldwide Interoperability for Microwave Access) [11] gibi birçok kablosuz uygulamada kullanılmaktadır. OFDM'de FDM'den farklı olarak alt-kanalların frekansları birbirine dik olacağından alt-kanallar arası girişimin önüne geçilmektedir. Çok kullanıcı iletiminde kullanılan Çoklu Erişimli Dikgen Frekans Bölüşümlü Çoğullama'da



(Orthogonal Frequency Division Multiple Access, OFDMA) ise kanal birbirine dikgen alt-kanallara bölünmekte ve alt-kanallar belirli kurallara göre kullanıcılara paylaştırılmaktadır. Dolayısıyla OFDMA'de, bir OFDM çerçevesinde birden fazla kullanıcının verisi taşınabilmektedir.

Baz istasyonu ve kullanıcılar arası iletişimde toplam iletim gücü ve bantgenişliği sınırlı kaynaklardır. Dolayısıyla bu kaynakların eniyi dağıtımı kritik bir önem taşımaktadır.

Önceleri kaynak tahsisi statik olarak yapılmıştır. Statik kaynak tahsisinde kullanıcıların sisteme erişimleri için TDMA veya FDMA kullanılmaktadır. Statik kaynak tahsisinde kullanıcıların kanal durumu gözetilmeden, önceden belirlenmiş zaman aralıklarında veya frekans bantlarında iletişim yapmaktadırlar.

Uyarlamalı kaynak tahsisinde ise kaynaklar kullanıcılara kanal durumlarına göre dağıtılmaktadır. Coğrafik olarak farklı yerlerde bulunan kullanıcılardan birisi için kötü olan bir alt-kanal (düşük kanal katsayılı dolayısıyla iletişim için daha fazla güce ihtiyaç duyan), bir diğeri için iyi olabilmektedir. Dolayısıyla bu önbilgi ile yapılan kaynak tahsisinin statige oranla daha iyi bir başarıma sahip olması beklenmektedir.

Güç paylaşımli kaynak tahsisinde amaç her kullanıcı için önceden belirlenen belirli bir veri hızına göre vericiden gönderilen toplam gücü enküçültmektir. Burada birinci kısıt kullanıcılar arası girişimi engellemek için her alt-taşıyıcının sadece bir kullanıcı tarafından kullanılabilmesidir. İkinci kısıt ise her kullanıcının veri hızının belirli bir değerin altına düşmemesidir. İkinci kısıt kullanıcıların iletişim kalitelerini doğrudan etkilemektedir.

Çok kullanıcıli MIMO OFDM sistemlerde ise kaynak tahsisine yeni bir boyut (uzay boyutu) katılarak kullanıcılar arası girişim makul seviyelere indirilebilmekte ve böylece kullanıcıların aynı alt-taşıyıcıları paylaşabilme imkanı doğmaktadır.

Tezin ikinci bölümünde OFDMA sistemlerde kaynak tahsisi problemi tek kullanıcıli ve çok kullanıcıli senaryolar için matematiksel olarak ifade edilmiştir. Kaynak tahsisi problemi, tamsayı kısıtların gevşetilmesiyle dışbükey (convex) hale getirilerek problem için bir alt sınır oluşturulmuştur. Literatürde konuyla ilgili önerilen diğeryöntemler hakkında da bilgi verilmiştir.

Üçüncü bölümde, ikinci bölümde formülize edilen kaynak tahsisini çözmekte kullanılacak olan çeşitli doğrusal ve tamsayı programlama teknikleri gösterilmiştir.

Doğrusal programlama için Simpleks Algoritması [12], tamsayı programlama için ise Dallan ve Sınırla [13] (Branch and Bound) Düzlem Kesme [13] (Cutting Plane), ve Dallan ve Kes [14] (Branch and Cut) Algoritmaları kullanılmıştır.

Tezin dördüncü bölümünde tamsayı programlama teknikleri kaynak tahsisi problemine uygulanmıştır. Problemin eniyi çözümü Dallan ve Sınırla ile Dallan ve Kes yöntemleriyle bulunurken en iyiye yakın bir performans gösteren ve tamsayı programlama yöntemlerine göre daha hızlı çalışan, bizim ortaya attığımız bir algoritma önerilmiştir.

Beşinci bölümde ise kaynak tahsisi problemi Genetik Algoritmalar kullanılarak çözülmüş ve sonuçlar gerekirci kaynak tahsisi yöntemleri ile kıyaslanmıştır.

Sunulan tezin altıncı bölümünde ise çok kullanıcı MIMO OFDM sistemlerde kaynak tahsisi problemi incelenmiştir. Doğrusal olmayan kaynak tahsisi problemi doğrusal hale getirilip Dallan ve Sınırla yöntemiyle çözülmüştür.

Son olarak çeşitli benzetim çalışmaları yapılarak, sonuçlar yorumlanıp tez tamamlanmıştır.

## 2. OFDMA SİSTEMLERDE KAYNAK TAHSİSİ

Bir iletişim sisteminin başlıca amacı vericiden alıcıya iletilmek istenen tüm bilginin (bitlerin), toplam iletim gücünün iletişim kanalı kaynakları (başlıcaları zaman, frekans ve uzay olan) arasında en verimli şekilde paylaşarak, en hızlı ve güvenilir biçimde kullanıcıya ulaştırılmasıdır. Sistemde birden fazla kullanıcı bulunması durumunda bu kaynakların kullanıcılar arasında belirli kriterlere göre paylaşılması problemi ortaya çıkmaktadır.

OFDMA sistemlerde kanal, frekans boyutunda parçalanarak birbirine dik alt-taşıyıcılar oluşturulmaktadır. Bu alt-taşıyıcıların kullanıcılara nasıl dağıtılacağı ve alt-taşıyıcılara ne kadar bit ve güç atanacağı aynı zamanda da servis kalitesi (Quality of Service, QoS) kriterlerinin (bit hata oranı, bit error rate, BER, veri hızı istemi) sağlanması kaynak tahsisi problemi olarak anılmaktadır. Temel olarak kaynak tahsisi problemi hangi kaynağın eniyileştirileceğine göre ikiye ayrılmaktadır:

1) Güç enküçültülmesi tabanlı kaynak tahsisi (Güç Payı Uyarlamalı, Margin Adaptive, MA): Sistem tasarımında amaç her kullanıcı için önceden belirlenen belirli bir veri hızına göre vericiden gönderilen toplam gücü enküçültmektir [26], [27], [28], [36], [37].

2) Kapasite enbüyütülmesi tabanlı kaynak tahsisi (Hız Uyarlamalı, Rate Adaptive, RA): Burada ise kaynak tahsisi probleminde amaç önceden belirlenen güç kısıtına göre sistemin toplam kapasitesini enbüyütmektir [17], [19], [21], [22], [23], [24].

Kaynak tahsisinde önemli bir parametre de servis kalitesidir. Servis kalitesi, tasarlanan sistemden her kullanıcıya sunulması gereken en düşük başarıımı belirten göstergedir. Tezin kapsamında servis kalitesi olarak aşağıdaki kriterler kullanılmıştır:

a) Bit hata oranı: Bit hata oranı kullanıcının veri alışverişindeki ortalama hata oranını vermektedir. BER genellikle her kullanıcı için önceden belirlenir, ve servis tipine (veri, ses, görüntü vb.) göre sabit seçilir.

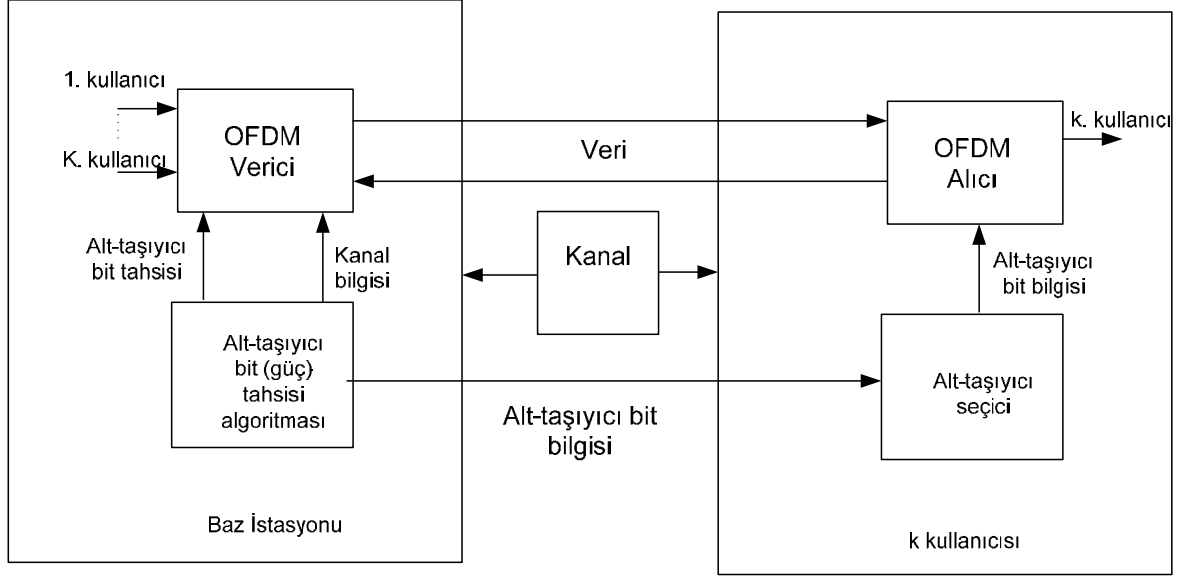
b) Servisin kullanılabilirliği (Availability of Service): Servis kullanılabilirliği sistemden bantgenişliği talep eden kullanıcıya istediği bantgenişliğinin atanıp atanmayacağı veya kullanıcıların beklemesi ve sisteme girmekten vazgeçmeleri konularını belirlemektedir. Örneğin sistemin toplam kapasitesini enbüyütmek adına kötü kanal şartlarına sahip kullanıcılara bant genişliği atanmaması o kullanıcıların servisten yararlanmasını engelleyecektir.

Servis kalitesi beraberinde adil olma (fairness) kavramını doğurmaktadır. Adil bir sistemde, sistemin toplam gücünün enküçültülmesi (MA kaynak tahsisi) veya toplam kapasitenin enbüyütülmesinden (RA kaynak tahsisi) ödün vermek uğruna sistemin bir servisini talep eden her kullanıcıya kaynak tahsis edilmektedir. Adil olmayan sistemler ise, kanal kazançları çok düşük olduğu için sistem başarımını olumsuz etkileyen kullanıcılara kaynaklarını tahsis etmeyerek toplam sistem başarımında adil sistemlere göre daha başarılı sonuçlar doğurmaktadır.

Tez kapsamında ağırlıklı olarak MA kaynak tahsisi problemi incelenmiştir. MA kaynak tahsisi probleminde maliyet işlevi vericiden gönderilen toplam güç iken kısıtlar kullanıcıların kanalda aynı frekans bantlarını kullanmamaları ve kullanıcıların iletişimlerinin en az talep ettikleri veri hızında olmasıdır. Bu model tüm kullanıcılara servis sağlaması bakımından adil bir sistemdir.

RA kaynak tahsisi probleminde ise maliyet işlevi sistemin toplam kapasitesi iken kısıtlar sistemin toplam gücü ve kullanıcıların kanalda aynı frekans bantlarını kullanmamalarıdır. Bu haliyle adil olmayan bir sistem olan RA kaynak tahsisi modeline kullanıcıların iletişimlerinin talep ettikleri veri hızında olması kısıtı da eklenerek adil bir sistem oluşturulabilir. İleride gösterileceği gibi literatürde MA kaynak tahsisi adil olarak yapılırken, RA kaynak tahsisi modellemelerinde kaynak tahsisi adil olmayan biçimde de yapılmıştır [20]. Görüldüğü gibi RA ve MA kaynak tahsisi problemleri arasında yakın bir ilişki vardır.

## 2.1 Sistem Modeli



Şekil 2.1: OFDMA sistem modeli [7].

Şekil 2.1’de blok şeması verilen aşağı yönlü (downlink) OFDMA sisteminde verici (baz istasyon) ve alıcılarda (kullanıcılar) tek anten bulunmaktadır. Kullanıcılara ait sinyaller baz istasyonda farklı alt-taşıyıcılara bindirilerek iletişim sağlanmaktadır. Kullanıcılar arası girişimi engellemek için bir alt-taşıyıcının en fazla bir kullanıcıya atanabileceği varsayılmaktadır. Her alt-taşıyıcının bantgenişliğinin kanalın evreuyumlu bantgenişliğinden daha küçük olduğu, dolayısıyla alt-taşıyıcıların frekans-düz sönümlenmeye maruz kaldıkları ve tüm kullanıcıların kanal katsayılarının verici tarafından hatasız olarak bilindiği kabul edilmektedir.

Verici, bu kanal bilgisini kullanarak tüm kullanıcılar için alt-taşıyıcı, bit ve güç tahsisi işlemini gerçekleştirmektedir. Alt-taşıyıcılara atanan bit miktarına göre uyarlamalı kipleme uygun kiplenim şeklini seçmekte ve algoritmanın o alt-taşıyıcı için atadığı güce göre verici gücünü ayarlamaktadır.

Sistemde  $K$  tane kullanıcı ve  $N$  tane alt-taşıyıcı vardır. Bir kullanıcıya ( $k$ ), bir OFDM çerçevesi başına atanan toplam bit sayısı  $R_k$  bit ile gösterilmektedir. Kullanıcı  $k$ ’nın  $n$ ’inci alt-taşıyıcısına atanan bit miktarı  $c_{k,n}$  ile temsil edilmektedir. OFDMA sistemlerde kullanıcılar arası girişimi engellemek için aynı alt-taşıyıcının birden fa-

zla kullanıcı tarafından kullanılmasına izin verilmemektedir. Dolayısıyla  $n$ 'inci alt-taşıyıcıda  $k$ 'inci kullanıcı için  $c_{k,n} \neq 0$  ise  $c_{k',n} = 0, \forall k' \neq k$  olmaktadır. Yani  $n$ 'inci alt-taşıyıcı  $k$ 'inci kullanıcıya atanmışsa ve üzerinden veri iletimi gerçekleştiriliyorsa ( $c_{k,n} \neq 0$ ) diğer kullanıcılar o alt-taşıyıcıdan veri gönderemez ( $c_{k',n} = 0$ ). Bit sayısı  $c_{k,n}$ ,  $D = \{0, 1, \dots, M\}$  kümesi içinden seçilmekte ve uyarlamalı kipleme bit sayısına göre  $2^M$  elemanlı bir sinyal yıldız kümesinin ilgili elemanını alt-taşıyıcıya yerleştirmektedir. Verici tarafından bir alt-taşıyıcıya sembol başına atanabilecek en yüksek bit sayısı  $M$  ile gösterilmektedir.

Vericide kaynak tahsisi işlemi tamamlandıktan sonra Ters Hızlı Fourier Dönüşümü (Inverse Fast Fourier Transform, IFFT) ile semboller zaman bölgesine geçirilerek her bir OFDM çerçevesine alt-taşıyıcılar arası dikgenliği garantilemek için dögüsel önek (cyclic prefix) eklenmektedir.

Alıcıda, alınan sinyalden dögüsel önekler çıkarıldıktan sonra zaman bölgesindeki sinyallerin Hızlı Fourier Dönüşümü (Fast Fourier Transform, FFT) alınarak alt-taşıyıcılardaki sinyaller elde edilmektedir. Daha sonra kipçözücüler bit tahsisine göre alıcıda sezilen sembolleri oluşturmaktadır.

Frekans seçici ve rasgele kanalda her alt-taşıyıcı ve kullanıcı için kanal katsayıları farklı olacaktır. Tezde kullanıcı  $k$ 'nın  $n$ 'inci alt-taşıyıcısındaki kanal katsayısının büyüklüğü  $\alpha_{k,n}$  olarak kabul edilmektedir. Ayrıca tek-yanlı izgesel güç yoğunluğu (Power Spectral Density, PSD) her kullanıcı ve alt-taşıyıcı için  $N_0 = 1$  olarak kabul edilmektedir. Bir kullanıcının  $c$  tane bittten oluşan bir sembolü  $P_e$  hata olasılığı ile alabilmesi için sembol başına ihtiyaç duyulan güç  $f_k(c)$  ile gösterilmektedir. Sembol başına ihtiyaç duyulan  $f_k(c)$  gücü, kullanıcı için uygun görülen QoS durumuna göre ayarlanabilmektedir. Kullanıcı  $k$  için  $n$ 'inci alt-taşıyıcıda M-QAM modülasyonu (Karesel Genlik Modülasyonu, M-ary Quadrature Amplitude Modulation, M-QAM) için

$$f_k(c_{k,n}) = \frac{N_0}{3} \left[ Q^{-1}\left(\frac{P_e}{4}\right) \right]^2 (2^{c_{k,n}} - 1) \quad (2.1)$$

olarak ifade edilir. Burada  $P_e$  hedef BER değerini temsil etmekte,  $Q$  işlevi ise

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-t^2/2} dt \quad (2.2)$$

ile verilmektedir. Sonuçta alıcının talep ettiği QoS'i (burada en yüksek hata olasılığı) sağlayabilmek için verici tarafından  $k$  kullanıcısının  $n$ 'inci alt-taşıyıcısına atanması

gereken güç

$$P_{k,n} = \frac{f_k(c_{k,n})}{\alpha_{k,n}^2} \quad (2.3)$$

olarak bulunur.

## 2.2 Problem Tanımı

MA kaynak tahsisi algoritmalarında amaç, kullanıcılar için önceden belirlenen veri hızlarına göre tüm alt-taşıyıcılara ve kullanıcılara atanan toplam gücü ( $P'_T$ ,  $P_{k,n}$  değerlerinin toplamı) enküçültecek  $c_{k,n}$  bit tahsisi değerlerini bulmaktır. Bölüm 2.1'deki tartışma takip edilerek eniyileme problemi aşağıdaki gibi tanımlanabilir. Enküçültülecek maliyet işlevi  $P'_T$  ve kısıtlar :

$$\begin{aligned} \text{maliyet işlevi} \quad P'_T &= \min_{c_{k,n} \in D} \sum_{n=1}^N \sum_{k=1}^K \frac{1}{\alpha_{k,n}^2} f_k(c_{k,n}) \\ \text{kısıtlar} \quad \forall k \in \{1, \dots, K\} &\text{ için } \sum_{k=1}^N c_{k,n} = R_k \\ c_{k',n} &= 0, \text{ eğer } c_{k,n} > 1, \forall n \in \{1, \dots, N\} \subset \mathbb{Z}, \\ &\forall k' \neq k, k \in \{1, \dots, K\}, c_{k,n} \in \{0, \dots, M\} \subset \mathbb{Z} \end{aligned} \quad (2.4)$$

şeklindedir. Burada birinci kısıt ile her kullanıcının, ihtiyaç duyduğu veri hızına erişiminin sağlanması garanti altına alınmaktadır. Bunun doğal sonucu olarak problem tanımındaki sistemin adil bir sistem olduğu görülmektedir. İkinci kısıt ise aynı alt-taşıyıcının aynı OFDM çerçevesinde sadece tek bir kullanıcıya atanmasını sağlamaktadır.

### 2.2.1 Tek Kullanıcılı Kanal İçin Kaynak Tahsisi

Tek kullanıcı olması durumunda sistemdeki varolan alt-taşıyıcıların tamamı o kullanıcıya atanmaktadır. Dolayısıyla kullanıcı istediği alt-taşıyıcıya istediği kadar güç ve bit yüklemesi yapabilmektedir. Tek kullanıcılı OFDM sistemlerde MA kaynak tahsisi problemi aşağıdaki gibi ifade edilir

$$\begin{aligned} \text{maliyet işlevi} \quad P_T^* &= \min_{c_n \in D} \sum_{n=1}^N \frac{1}{\alpha_n^2} f(c_n) \\ \text{kısıtlar} \quad \sum_{n=1}^N c_n &= R. \end{aligned} \quad (2.5)$$

Burada  $\alpha_n$ ,  $n$ 'inci alt-taşıyıcının kanal katsayısı,  $D = \{0, 1, \dots, M\}$  ve  $M$  alt-taşıyıcı başına atanabilecek en yüksek bit sayısıdır. Bu problemin eniyi çözümü Açgözlü (Greedy) Algoritma [15] ile bulunabilmektedir. Açgözlü Algoritma her

iterasyonda her alt-taşıyıcıya bir bit daha eklendiğinde ihtiyaç duyulan gücü hesaplamaktadır. En düşük güce ihtiyaç duyan alt-taşıyıcı bir bit daha taşımayı haketmektedir. Bit tahsisi işlemleri  $R$  yineleme sonra sonlanmaktadır. Ağgözlü Algoritma'nın yapısı aşağıdaki gibidir

$$\begin{aligned}
&\forall n \text{ için } c_n = 0 \text{ ve } \Delta P_n = [f(1) - f(0)]/\alpha_n^2 \\
&\text{Aşağıdaki işlemler } R \text{ defa tekrarlanacaktır:} \\
&\tilde{n} = \arg \min_n \Delta P_n; \\
&c_{\tilde{n}} = c_{\tilde{n}} + 1 \\
&\Delta P_n = [f(c_{\tilde{n}} + 1) - f(c_{\tilde{n}})]/\alpha_n^2
\end{aligned} \tag{2.6}$$

$R$  iterasyon sonunda bit tahsisi işlemi tamamlanmış olur.

### 2.2.2 OFDMA Sistemlerde Kaynak Tahsisi

Ağgözlü Algoritma tek kullanıcı sistemler için eniyi çözümü verirken çok kullanıcı sistemlerde eniyi çözümü vermekten uzak kalmaktadır. Bu durum OFDMA'de aynı alt-taşıyıcının birden fazla kullanıcı tarafından kullanılmasına izin verilmemesinden kaynaklanmaktadır. Aynı alt-taşıyıcının birden fazla kullanıcı için iyi (yüksek kanal katsayılı) olduğu durumlarda sadece bir kullanıcı o alt-taşıyıcıyı kullanabilmektedir. Bu durum diğer kullanıcı için önemli bir kapasite kaybına neden olabilmektedir. Bu nedenle benzer şartlar altında hangi kullanıcının ilgili alt-taşıyıcıyı kullanmasına izin verileceği önemli bir problemdir.

Wong [15], tamsayı (bit ve kanalların ayrık olması) kısıtlarını gevşeterek problemi dışbükey hale getirmiştir. Problemin gevşetilmiş hali ileride kullanılacak tamsayı programlama metodlarında başarımın üst sınır referansı olarak alınacaktır.

Tamsayı kısıtlarının gevşetilmesindeki amaç problemi ayrık uzaydan sürekli uzaya taşımaktır. Atanan bitleri gösteren  $c_{k,n}$ 'nin alacağı değerler  $[0, M]$  aralığında bir gerçek sayı olarak seçilmektedir. Alt-taşıyıcıların kimler tarafından paylaşıldığını gösteren kısıt ise  $\rho_{k,n} \in [0, 1]$  değişkeni ile ifade edilecektir.  $\rho_{k,n} = 1$  durumu  $k$ 'ninci kullanıcının  $n$ 'inci alt-taşıyıcıyı tamamen kullandığını,  $\rho_{k,n} = 0$  ise hiç kullanmadığını göstermektedir. İki veya daha çok kullanıcı için  $\rho_{k,n} \in (0, 1)$  ve  $\sum_{k \in K'} \rho_{k,n} = 1$ ,  $k \in K' \subseteq \{1, \dots, K\}$  koşulu ise ilgili kullanıcıların  $n$ 'inci alt-taşıyıcıyı zaman paylaşımı ile kullandığını, yani  $k$ 'ninci kullanıcının tüm OFDMA çerçevelerinin  $\rho_{k,n}$  oranındakilerde  $n$ 'inci alt-taşıyıcıya sahip olduğu, diğer  $(1 - \rho_{k,n})$  çerçevelerde ise  $K'$  kümesindeki diğer kullanıcıların bu alt-taşıyıcıyı kullandığını gösterir. Böylece gevşetilmiş OFDMA kay-



nak tahsisi problemi aşağıdaki gibi ifade edilir:

$$\begin{aligned}
\text{maliyet işlevi} \quad P_T &= \min_{\substack{c_{k,n} \in [0, M] \\ \rho_{k,n} \in [0, 1]}} \sum_{n=1}^N \sum_{k=1}^K \frac{\rho_{k,n}}{\alpha_{k,n}^2} f_k(c_{k,n}) \\
\text{kısıtlar} \quad \sum_{n=1}^N \rho_{k,n} c_{k,n} &= R_k, \text{ tüm } k \in \{1, \dots, K\} \text{ için} \\
\sum_{k=1}^K \rho_{k,n} &= 1, \text{ tüm } n \in \{1, \dots, N\} \text{ için.}
\end{aligned} \tag{2.7}$$

Alt-taşıyıcının kullanıcılar arası paylaşımını gösteren  $\rho_{k,n}$  parametresi için aşağıdaki gösterim doğrudur

$$\rho_{k,n} = \begin{cases} 1, & c_{k,n} \neq 0 \\ 0, & c_{k,n} = 0 \end{cases}. \tag{2.8}$$

O halde  $c_{k,n}$ 'nin tamsayı olma kuralı korunursa ve  $\rho_{k,n}$ 'nin sadece (2.8)'deki 1 veya 0 değerlerini almasına izin verilirse gevşetilmiş kaynak tahsisi problemi (2.7) ile tamsayı kaynak tahsisi problemi (2.4) aynı sonucu verecektir. Problem (2.7) ile problem (2.4)'deki maliyet işlevleri aynı olmasına rağmen (2.7)'de eniyileme işlemi sürekli değişkenler nedeniyle daha büyük bir uzay üzerinden yapıldığı için ayrık değişkenler kabul eden (2.4) problemine bir alt-sınır oluşturmaktadır.

Problem (2.7)'de kullanılan  $f_k(c)$  işlevi dışbükey olmasına rağmen  $\rho f_k(c)$  şeklindeki maliyet işlevi,  $\rho$  ve  $c$  değişkenleri üzerinden dışbükey değildir [15]. Çünkü maliyet işlevinin Hessian matrisinin özdeğerlerine bakıldığında negatif değer ortaya çıkmakta, bu da dışbükeylikte Hessian matrisinin yarı kesim artı olma kuralını ihlal etmektedir. Problemi dışbükey hale getirmek için Wong tarafından  $r_{k,n} = c_{k,n} \rho_{k,n}$  olacak şekilde yeni bir değişken tanımlanmıştır. Yeni değişkenin tanımlanmasıyla Problem (2.7) aşağıdaki gibi ifade edilebilir:

$$\begin{aligned}
\text{maliyet işlevi} \quad P_T &= \min_{\substack{r_{k,n} \in [0, M \rho_{k,n}] \\ \rho_{k,n} \in [0, 1]}} \sum_{n=1}^N \sum_{k=1}^K \frac{\rho_{k,n}}{\alpha_{k,n}^2} f_k\left(\frac{r_{k,n}}{\rho_{k,n}}\right) \\
\text{kısıtlar} \quad \sum_{n=1}^N r_{k,n} &= R_k, \text{ tüm } k \in \{1, \dots, K\} \text{ için} \\
\sum_{k=1}^K \rho_{k,n} &= 1, \text{ tüm } n \in \{1, \dots, N\} \text{ için.}
\end{aligned} \tag{2.9}$$

Problem (2.9) dışbükeydir, dolayısıyla dışbükey eniyileme yöntemleriyle çözülebilmektedir. Wong, Problem (2.9)'u çözmek için Lagrange çarpanlarını [16] kullanmıştır. Fakat bu çözümde aynı alt-taşıyıcı birden fazla kullanıcıya zaman

paylaşımı ile atanmış ve/veya kullanıcılara atanan bitler kesirli değerler almış olabilir. Wong, gevşetilmiş problemde elde edilen çözümü (2.4) problemine uygulayabilmek için MAO (Multiuser Adaptive OFDM, Çok Kullanıcı Uyarlamalı OFDM) yöntemini önermiştir. Bu yöntemde ilk adımda alt-taşıyıcıların sadece tek bir kullanıcıya atanmasını sağlamıştır. Eğer aynı alt-taşıyıcı birden fazla kullanıcı tarafından kullanılıyor ise o alt-taşıyıcıyı en çok kullanan, yani o alt-taşıyıcı için en yüksek  $\rho_{k,n}$  değerine sahip kullanıcı, alt-taşıyıcıyı tüm zamanlarda kullanırken diğerleri o alt-taşıyıcıdan düşürülmektedir. Alt-taşıyıcı tahsisi yapıldıktan sonra ise her kullanıcı için açgözlü algoritma ile kaynak tahsisi işlemi yapılmıştır. Doğal olarak Wong'un önerdiği yöntem eniyiye yakın bir çözümdür.

### 2.3 Literatürde Önerilen Diğer Yöntemler

Literatürde önerilen metodlar incelendiğinde genel olarak problem alt-kanal tahsisi ve güç-bit tahsisi olmak üzere iki aşamaya ayrılmıştır. Genelde amaç eniyi çözümden ödün vermek pahasına hızlı ve uygulanabilir çözümler elde etmektir.

Ermelova, [17] OFDM için tasarlanmış olan Sıralı Alt-taşıyıcı Seçme Algoritması'nı [18] (Ordered Subcarrier Selection Algorithm, OSSA) OFDMA için kullanmıştır. Önerilen yöntem temel olarak, her iterasyonda rasgele seçilen bir kullanıcıya OSSA algoritmasını uyguladıktan sonra başka bir kullanıcıyı seçmekte ve seçilen yeni kullanıcı için de OSSA algoritmasını uygulamaktadır. Algoritma RA kategorisindedir.

Yine RA eniyilemeyi hedefleyen Giovanidis [19] tarafından önerilen algortmada kullanıcılar gruplara bölünmektedir. Eşit güç dağıtımını kullanılarak toplam veri hızını enbüyültecek alt-taşıyıcı seçimi kombinasyonları aranmaktadır.

Jang'ın [20] önerdiği RA yönteminde öncelikle Sinyalin Gürültüye ve Girişime Oranı (Signal to Interference and Noise Ratio, SINR) gözönüne alınarak alt-taşıyıcı tahsisi yapılmaktadır. Alt-taşıyıcı tahsisinden sonra ise güç ve bit dağıtımını yapılmaktadır.

Kim [21] tarafından önerilen RA yöntemde Karush-Kuhn-Tucker koşulu kullanılarak öncelikle alt-taşıyıcı tahsisi yapıldıktan sonra güç tahsisi kullanıcılara tek tek su doldurma yöntemiyle yapılmaktadır.

Rhee, [22] yönteminde OFDMA için kapasite enbüyültmesi problemini dışbükey hale getirmiştir. Dışbükey hale getirilmiş problemin çözümünü daha da hızlandırmak amacıyla önerilen eniyiye yakın yöntemde alt-taşıyıcılara eşit güç

dağıtılarak alt-taşıyıcı tahsisi yapılmıştır.

Wang, [23] kapasite enbüyütmesi probleminin çiftesini formülize etmiş, çiftes problemi kullanarak alt-taşıyıcı tahsisini yaptıktan sonra ise açgözlü algoritma kullanarak güç tahsisini gerçekleştirmiştir.

Shen tarafından önerilen RA algoritmada ise [24] sistemin kapasitesi enbüyütülürken kaynak tahsisinin adil bir şekilde yapılması için probleme yeni kısıtlar getirilmiştir. Algoritmanın daha hızlı çalışması için öncelikle toplam güç kullanıcılar arasında eşit olarak paylaştırılmış, alt-taşıyıcı tahsisi yapıldıktan sonra Lagrange çarpanları kullanılarak halihazırda kullanılan alt-taşıyıcı dağılımına göre eniyi güç dağıtımı yapılmıştır .

OFDMA sistemlerde kaynak tahsisi, açıksistemler arabağlaşımı (Open System Interface, OSI) modeli gözönüne alındığında fiziksel katmanda (PHY) gerçekleştirilmektedir. IEEE 802.16 sistemindeki OFDMA sistemini inceleyen Einhaus, [25] kaynak tahsisi işlemine Ortam Erişim Kontrolü (MAC) tabakasını da dahil etmiştir. Dolayısıyla sadece bit katarlarını değil aynı zamanda verilerin paket-çerçeve formunda iletimi ile de ilgilenmiştir (Örneğin Hata Durumunda Yineleme, ARQ). Bu modelin tez kapsamında incelenen modellerden oldukça farklı olduğu gözlenmiştir. Önerilen algoritma RA kategorisindedir.

Zhang [26] tarafından önerilen MA algoritma, ilk aşamada aynı alt-taşıyıcının birden fazla kullanıcı tarafından kullanılmasına izin verdikten sonra açgözlü algoritma ile kaynak tahsisini yapmaktadır. Bu işlem sonucunda aynı alt-taşıyıcıların kullanıcılar arasında paylaşımı söz konusu değilse algoritma sonlanır. Aksi halde her alt-taşıyıcının sadece bir kullanıcı tarafından kullanılmasını sağlayarak eniyiye yakın bir çözüm elde edilmektedir.

Kıvanç [27] tarafından önerilen MA yöntemde ilk olarak her kullanıcıya atanacak olan alt-taşıyıcı sayısı kullanıcıların sinyal gürültü oranlarına bakılarak atanmaktadır. Alt-taşıyıcı sayıları belirlendikten sonra da toplam gücü enküçültecek şekilde bit tahsisi yapılmaktadır.

Choe, [28] açgözlü algoritmayı OFDMA'e uyarlamıştır. Her iterasyonda seçilen alt-taşıyıcı için fazladan bir bit eklendiğinde en az güce ihtiyaç duyan kullanıcı o alt-taşıyıcı kullanmayı hak etmektedir. Alt-taşıyıcı tahsisi tamamlandıktan sonra her kullanıcı için Blockwise Loading Algorithm (BLA) ile bit tahsisi yapılmaktadır. BLA

alt-taşıyıcıları gruplayarak aynı gruptaki alt-taşıyıcılar için aynı tip kiplenimi kullanan eniyiye yakın bir yöntemdir.

Munz, [29] kaynak tahsisi problemi için bir çoklu su doldurma algoritması geliştirmiştir. Han [30] ise oyun teorisini kaynak tahsisi problemine uygulamıştır.

Wang [31], Tang [32] ve Reddy [33], [34] kaynak tahsisi problemini çeşitli evrimsel algoritmalar kullanarak çözmüşlerdir. Evrimsel algoritmalar ile kaynak tahsisi ileriki bölümlerde anlatılacaktır.

Kim [35], doğrusal olmayan kaynak tahsisi problemini doğrusal hale getirmiştir. Mao [36], [37] ise problemi matris, vektörler şeklinde ifade edip Dallan ve Sınırla Algoritmasını kullanmıştır. Dallan ve Sınırla Algoritması ile kaynak tahsisi probleminde eniyi çözümler elde edilebilmektedir. Dallan ve Sınırla Algoritması Kıvanç tarafından da uygulanmış fakat kaynak tahsisi problemi makul sürelerde çözülememiştir.

Görüldüğü gibi literatürde OFDMA sistemlerde kaynak tahsisi problemi için çok farklı alternatifler sunulmaktadır. Önerilen yöntemlerin büyük çoğunluğu eniyi çözümü sağlamamaktadır. Tez kapsamında eniyi çözümü verecek yöntemler aranmıştır. Bunun için tamsayı programlama yöntemlerinin uygun olduğu görülmüştür. Çünkü tamsayı programlama teknikleri bize sağlam (robust) sonuçlar sunmakta ve eniyi çözümü vermektedir. OFDMA kaynak tahsisi probleminin çözümü için Dallan ve Sınırla [36] ile Dallan ve Kes yöntemleri uygulanmıştır. Bunun dışında tamsayı kısıtların gevşetilmesi işlemi yapıldıktan sonra doğrusal programlama yöntemi olan Simpleks Algoritması kullanılarak elde edilen alt-taşıyıcı dağılımından sonra Ağgözlü Algoritma kullanılarak eniyiye yakın bir çözüm de önerilmiştir. Ayrıca Genetik Algoritmaların tahsis problemine uyarlanması incelenmiş, son olarak da MIMO kanallarda kaynak tahsisi için bir önçalışma yapılmıştır.

### 3. DOĞRUSAL VE TAMSAYI PROGRAMLAMA

Pek çok mühendislik uygulamasında karşımıza çıkan doğrusal programlama problemlerinde maliyet işlevi ve kısıtlar doğrusal bir yapıya sahiptir (Eniyileme işlemindeki değişkenlerin kabul ettiği değerler tamsayı ise problem doğrusal tamsayı programlama olarak adlandırılır). Doğrusal programlamada maliyet işlevi ve kısıtlar standart biçimde [38]

$$\begin{array}{ll} \text{maliyet işlevi} & \mathbf{c}^T \mathbf{x} \\ \text{kısıtlar} & \mathbf{Ax} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{array} \quad (3.1)$$

$\mathbf{c} \in \mathbb{R}^n$ ,  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{b} \in \mathbb{R}^m$  olacak şekilde ifade edilebilir.

Eşitsizlik kısıtı  $\mathbf{Ax} \leq \mathbf{b}$  artık değişkenlerin yardımıyla eşitlik kısıtına çevrilebilmektedir. Bunun için ilk olarak vektör ve matrislerden oluşan  $\mathbf{c}^T \mathbf{x}$  ile  $\mathbf{Ax} \leq \mathbf{b}$  ve  $\mathbf{x} \geq \mathbf{0}$  ifadeleri skalar olarak aşağıdaki gibi yazılır

$$\begin{array}{ll} \text{maliyet işlevi} & c_1x_1 + c_2x_2 + \dots c_nx_n \\ \text{kısıtlar} & a_{11}x_1 + a_{12}x_2 + \dots a_{1n}x_n \leq b_1 \\ & \vdots \\ & a_{m1}x_1 + a_{m2}x_2 + \dots a_{mn}x_n \leq b_m \\ & x_1 \geq 0, x_2 \geq 0, \dots x_n \geq 0. \end{array} \quad (3.2)$$

Aynı ifade ( $x_{n+1}, x_{n+2}, \dots, x_{n+m}$ ) artık değişkenlerin katkısıyla

$$\begin{array}{ll} \text{maliyet işlevi} & c_1x_1 + c_2x_2 + \dots c_nx_n \\ \text{kısıtlar} & a_{11}x_1 + a_{12}x_2 + \dots a_{1n}x_n + x_{n+1} = b_1 \\ & a_{11}x_1 + a_{12}x_2 + \dots a_{1n}x_n + x_{n+2} = b_2 \\ & \vdots \\ & a_{m1}x_1 + a_{m2}x_2 + \dots a_{mn}x_n + x_{n+m} = b_m \\ & x_1 \geq 0, x_2 \geq 0, \dots x_n \geq 0 \\ & x_{n+1} \geq 0, x_{n+2} \geq 0, \dots x_{n+m} \geq 0. \end{array} \quad (3.3)$$

olarak (dolayısıyla maliyet işlevi  $\mathbf{c}^T \mathbf{x}$  ve kısıtlar  $\mathbf{Ax} = \mathbf{b}$ ,  $\mathbf{x} \geq \mathbf{0}$  şeklinde) ifade edilir. Yeni formda, maliyet işlevi ifadesi değişmezken eşitsizlik kısıtı  $\left[ \mathbf{A} \ \mathbf{I} \right]$  haline dönüşmektedir (Burada  $\mathbf{I}$ ,  $m \times m$  boyutlarında birim matristir.).

#### 3.1 Doğrusal Programlamada Temel Çözümler

$$\mathbf{Ax} = \mathbf{b} \quad (3.4)$$

eşitliğinde ( $\mathbf{A}$  matrisinin kertesini  $m$  olmak koşuluyla)  $\mathbf{A}$  matrisinden birbirinden bağımsız  $m$  tane sütun seçilir. Gösterim kolaylığı için ve genelliği kaybetmeden bu vektörlerin  $\mathbf{A}$  matrisinin ilk  $m$  sütunu olduğu varsayılabilir. Doğrusal bağımsız olan bu

$m$  sütundan oluşturulan  $m \times m$  boyutlarındaki tam kerte (full rank) matris  $\mathbf{B}$  olarak adlandırılın. Açıkça,  $\mathbf{B}$  matrisi tekil olmayan bir matristir. Dolayısıyla

$$\mathbf{B}\mathbf{x}_B = \mathbf{b} \quad (3.5)$$

eşitliği biricik olarak çözülebilir. Sonuç olarak  $\mathbf{A}\mathbf{x} = \mathbf{b}$  denkleminin çözümlerinden birisi,  $\mathbf{x}$  vektörünün ilk  $m$  elemanı  $\mathbf{x}_B$ 'ye eşitlenip kalan elemanları sıfır yapılarak bulunabilir,  $\mathbf{x} = \begin{bmatrix} \mathbf{x}_B & \mathbf{0}_{1 \times (n-m)} \end{bmatrix}^T$ .

Yukarıdaki fikri genellersek, bilinmeyen sayısı  $n$  olan  $m$  ( $m \leq n$ ) tane doğrusal eşitlikten oluşan bir denklem sisteminde  $\mathbf{B}$  matrisi,  $\mathbf{A}$  matrisinin  $m$  adet doğrusal bağımsız sütunundan oluşturulmuş  $m \times m$  boyutlarında tekil olmayan bir matris olsun.  $\mathbf{A}\mathbf{x} = \mathbf{b}$  eşitliğinde,  $\mathbf{x}$  vektörünün  $\mathbf{B}$  matrisiyle ilişkisi bulunmayan  $n-m$  bileşeninin 0'a eşitlenerek elde edilen çözüm, (3.4) denklem sistemi için *temel çözüm* olarak adlandırılır. Bu sistemde  $\mathbf{x}$  vektörünün  $\mathbf{B}$  matrisiyle ilişkisi bulunan elemanları da *temel değişken* olarak adlandırılır. Ayrıca

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \quad (3.6)$$

eşitlik sistemini sağlayan  $\mathbf{x}$  vektörü *temel olurlu çözümdür*.

Hem kısıtları sağlayan hem de (3.1)'deki maliyet işlevi  $\mathbf{c}^T \mathbf{x}$ 'in eniyi değerini veren  $\mathbf{x}$  vektörü *eniyi olurlu çözüm* olarak adlandırılır. Bu çözüm aynı zamanda temel değişken özelliğini de taşıyorsa *eniyi temel olurlu çözüm* olarak adlandırılır.

Örneğin, aşağıdaki eşitsizliklerle sınırlandırılmış iki boyutlu uzayı göz önüne alalım

$$x_1 + \frac{8}{3}x_2 \leq 4$$

$$x_1 + x_2 \leq 2$$

$$2x_1 \leq 3$$

$$x_1 \geq 0$$

$$x_2 \geq 0.$$

Artık deęişkenlerle eşitsizlik sistemi eşitlik formuna çevrilirse

$$x_1 + \frac{8}{3}x_2 + x_3 = 4$$

$$x_1 + x_2 + x_4 = 2$$

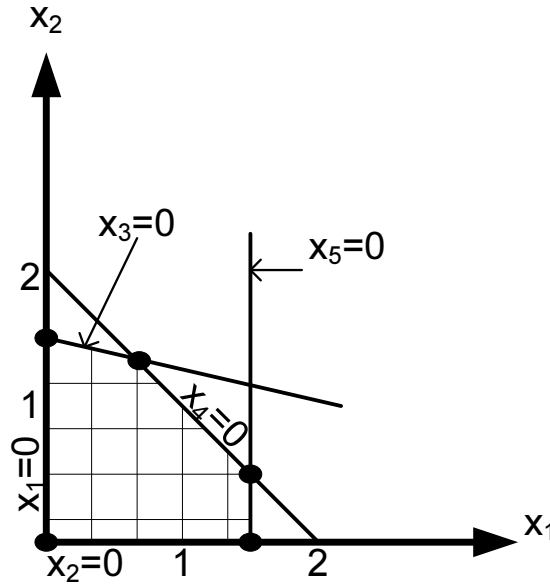
$$2x_1 + x_5 = 3$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0$$

olur. Bu eşitlikler matris-vektör biçiminde aşağıdaki gibi ifade edilebilir  $\mathbf{Ax} = \mathbf{b}$ ,

$$\mathbf{A} = \begin{bmatrix} 1 & 8/3 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 2 & 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 4 \\ 2 \\ 3 \end{bmatrix}.$$

$\mathbf{Ax} = \mathbf{b}$  denklem kümesinin oluşturduğu olurlu kümede, Şekil 3.1'de görüldüğü gibi 5 tane tepe noktası vardır. Kısıtların oluşturduğu olurlu bölge de Şekil 3.1'de gösterilmektedir.



Şekil 3.1: Olurlu çözüm kümesi [12].

Yukarıdaki örnekteki denklem sisteminde  $\mathbf{A}$  matrisinin kertesinin 3 olması nedeniyle 3 adet temel deęişken vardır. Dolayısıyla herhangi 2 deęişkeni 0'a eşitleyerek düzlemde kenarlar boyunca bir tepe noktadan başka bir tepe noktaya geçiş

sağlanabilmektedir. Örneğin  $(0,0)$  noktasındayken  $x_1 = 0$  ve  $x_3 = 0$  doğrularının kesişmesiyle oluşan tepe noktasına ulaşmak için  $\mathbf{A}$  matrisinin birinci ve üçüncü sütunları silinerek

$$\mathbf{B} = \begin{bmatrix} \frac{8}{3} & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

elde edilir.

$$\begin{bmatrix} | & | & | & | & | \\ \cancel{a_1} & a_2 & \cancel{a_3} & a_4 & a_5 \\ | & | & | & | & | \end{bmatrix} \begin{bmatrix} 0 \\ x_2 \\ 0 \\ x_4 \\ x_5 \end{bmatrix} = \mathbf{b} \Rightarrow \begin{bmatrix} x_2 \\ x_4 \\ x_5 \end{bmatrix} = \mathbf{B}^{-1} \mathbf{b}$$

$$\mathbf{B} = \begin{bmatrix} | & | & | \\ a_2 & a_4 & a_5 \\ | & | & | \end{bmatrix}$$

Şekil 3.2:  $\mathbf{Ax} = \mathbf{b}$  denklem kümesinin doğrusal bağımsız 2, 4 ve 5'inci sütunlarının kullanılarak bir temel çözümün elde edilmesi.

Yukarıdaki  $\mathbf{B}$  matrisinin tersi  $\mathbf{b}$  vektörüyle çarpılırsa

$$\begin{bmatrix} x_2 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 0.3 \\ 3 \end{bmatrix}$$

vektörü elde edilir. Bu değerler  $x_1 = 0$ ,  $x_2 = 1.5$ ,  $x_3 = 0$ ,  $x_4 = 0.5$ ,  $x_5 = 3$  noktasını göstermektedir. Böylece tepe noktası  $x_1 = 0$ ,  $x_2 = 0$ 'dan başka bir tepe noktası  $x_1 = 0$ ,  $x_2 = 1.5$ 'e geçiş sağlanmıştır.

### 3.1.1 Simpleks Yöntemi

Standart biçimdeki (3.1) doğrusal programlama probleminde  $\mathbf{A}$  matrisinin  $m \times n$  boyutlarında ve  $m$  kertesine sahip olduğu durum için doğrusal programlamanın temel teoremine göre [12]:



- i) Eğer olurlu bir çözüm varsa, temel olurlu bir çözüm de vardır.  
 ii) Eğer eniyi olurlu bir çözüm varsa eniyi temel olurlu çözüm de vardır.

Bu teorem sayesinde eniyileme problemi için sürekli uzayda sonsuz sayıda elemana sahip olurlu kümeyi ( $\mathbf{Ax} \leq \mathbf{b}$ 'nin tanımladığı çokyüzlünün her noktası) taramak yerine sadece sonlu sayıya sahip temel çözümler (çokyüzlünün sadece tepe noktaları) taranarak eniyi çözüm elde edilebilmektedir. Doğrusal programlamada temel olurlu kümenin oluşturduğu çokyüzlünün tepelerinden oluşan değişken sayısı  $n$  ve kısıt sayısı  $m$  olan bir problem için en çok (temel çözümler  $n$  adet sütundan  $m$  tanesi seçilerek oluşturulduğundan)

$$\binom{n}{m} = \frac{n!}{m!(m-n)!}$$

tane temel çözüm vardır.

Doğrusal programlamanın temel teoremi sayesinde eniyileme problemi çözümü daha basit bir hale gelmesine rağmen pratikte değişken sayısı ve kısıt sayısı arttıkça tepe sayısındaki artış kaba kuvvet taramayı imkansız hale getirmektedir. Temel teoremden yola çıkan Simpleks Yöntemi ise temel olurlu çözümleri etkin olarak taramayı hedeflemektedir. Bunun için her adımda maliyet işlevinin değerini düşürecek şekilde bir temel olurlu çözümden başka bir temel olurlu çözüme gitmektedir.

Bir temel çözümden maliyet işlevi değerini düşürecek şekilde başka bir temel çözüme gitme işlemine pivotlama işlemi denilmektedir. Pivotlama işlemi aşağıdaki örnek ile açıklanacaktır [39].

$$\begin{array}{ll} \text{maliyet işlevi} & 2x_1 + 3x_2 - 4x_3 - 5x_4 + x_5 - 2x_6 \\ \text{kısıtlar} & x_1 - 2x_2 + 2x_5 + 7x_6 = 3 \\ & 3x_2 + x_4 - x_5 + 4x_6 = 5 \\ & x_3 - 3x_5 + 2x_6 = 4 \\ & x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0, x_6 \geq 0 \end{array} \quad (3.7)$$

olarak tanımlanmış doğrusal programlama probleminde bir tepe noktası için temel değişkenler  $x_1, x_3, x_4$  ve bir temel çözüm  $x_1 = 3, x_2 = 0, x_3 = 4, x_4 = 5, x_5 = 0, x_6 = 0$  şeklindedir. Simpleks Yöntemi'nde, kısıtlar ve maliyet işlevi değerleri bir tablo içerisine yerleştirilmekte ve bir tepe noktasından diğerine yer değiştirme işlemi bu tablo üzerinden yapılmaktadır. Yukarıdaki örnek için ilgili tablo aşağıdadır :

$$\left[ \begin{array}{cccccc|c} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & b \\ 1 & -2 & 0 & 0 & 2 & 7 & 3 \\ 0 & 3 & 0 & 1 & -1 & 4 & 5 \\ 0 & 0 & 1 & 0 & -3 & 2 & 4 \end{array} \right].$$

Tabloda ilk 6 sütun sırasıyla 6 değişkene denk gelmektedir. Görüldüğü gibi temel değişkenler  $x_1$ ,  $x_3$  ve  $x_4$ 'e karşılık gelen 1, 3 ve 4'üncü sütunlar  $3 \times 3$  boyutlarında bir birim matris oluşturmaktadır.  $x_2 = x_5 = x_6 = 0$  ve bu birim matris yardımıyla ilgili tepeye karşılık gelen temel çözüm elde edilir.

Sonraki adımda örneğin  $x_5$  ile  $x_4$  değişkenlerine pivot işlemi yapılarak  $x_5$ 'in temel değişken yapılması istensin. Bu durumda aşağıda belirtilen Gauss eleme yöntemindeki temel satır değiştirme işlemleri gerçekleştirilerek

$$1.\text{satır} \leftarrow 1.\text{satır} + (2 \times) 2.\text{satır}$$

$$3.\text{satır} \leftarrow 3.\text{satır} + (-3 \times) 2.\text{satır}$$

$$2.\text{satır} \leftarrow (-1 \times) 2.\text{satır}$$

şu tablo elde edilir

$$\left[ \begin{array}{cccccc|c} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & b \\ 1 & 4 & 0 & 2 & 0 & 15 & 13 \\ 0 & -3 & 0 & -1 & 1 & -4 & -5 \\ 0 & -9 & 1 & -3 & 0 & -10 & -11 \end{array} \right].$$

Yeni tablodaki temel değişkenler  $x_1$ ,  $x_3$ ,  $x_5$  iken temel çözüm  $x_1 = 13$ ,  $x_2 = 0$ ,  $x_3 = -11$ ,  $x_4 = 0$ ,  $x_5 = -5$ ,  $x_6 = 0$  olacaktır. Örnekten görüldüğü üzere temel değişken olmayan  $x_5$  değişkeni üzerinden pivotlama işlemi sonucu  $\mathbf{x} = [3 \ 0 \ 4 \ 5 \ 0 \ 0]^T$  tepe noktasından komşu  $\mathbf{x} = [13 \ 0 \ -11 \ 0 \ -5 \ 0]^T$  tepe noktasına geçilmiştir. Benzer şekilde  $x_1$  ve  $x_4$  değişkenleri ile diğer komşu tepe noktalara geçilebilir. Böylece toplam  $n - m = 3$  komşuluğa da ulaşılabilir. Böylece toplam

Tablodaki elemanlar  $y_{ij}$ ,  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, n + 1$  ( $m$  kısıt ve  $n$  değişken sayıları) şeklinde gösterilsin. Bu durumda tabloda  $k$ 'inci satırda sıfırdan farklı katsayıya sahip  $x_p$  temel değişkeni ile temel olmayan  $x_q$  değişkenlerinin yerlerini değiştirmek için yapılması gereken satır işlemleri şunlardır

$$y_{ij} = y_{ij} - \frac{y_{iq}}{y_{kq}} y_{kj}, \quad i \neq k, \quad j = 1, 2, \dots, n + 1 \quad (3.8)$$

$$y_{kj} = \frac{y_{kj}}{y_{kq}}, \quad j = 1, 2, \dots, n + 1.$$

Simpleks Algoritması'nda pivot işlemi sonrası oluşan yeni maliyet değerinin eski maliyet değerine göre daha küçük olması beklenmektedir. Bunu sağlamak için ilk

önce ilk  $m$  eleman temel değişkenler olacak şekilde tablo sütun değiştirmeleri ile yeniden düzenlenir:

$$\left[ \begin{array}{cccccc|c} x_1 & x_2 & x_3 & x_{m+1} & x_n & & b \\ 1 & 0 & 0 & y_{1,m+1} & y_{1n} & & y_{1,n+1} \\ 0 & 1 & 0 & y_{2,m+1} & y_{2n} & & y_{2,n+1} \\ \vdots & & & & & & \\ 0 & 0 & 1 & y_{m,m+1} & y_{mn} & & y_{m,n+1} \end{array} \right]. \quad (3.9)$$

İlk  $m$  sütundaki birim matris ilk  $m$  değişkenin temel değişken olduklarını göstermektedir. Bunu takiben temel ve temel olmayan değişkenleri kapsayan çözüm

$$\mathbf{x}^T = \begin{bmatrix} \mathbf{x}_B \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} y_{1,n+1} \\ y_{2,n+1} \\ \vdots \\ y_{m,n+1} \end{bmatrix} \quad (3.10)$$

olacaktır.

Tabloda görülen temel çözüm için maliyet vektörü ise  $\mathbf{c}_B = [c_1 \ c_2 \ \dots \ c_m]^T$  ile tanımlanırsa

$$z = \sum_{i=1}^n c_i x_i = \mathbf{c}_B^T \mathbf{x}_B = \sum_{i=1}^m c_i y_{i,n+1} \quad (3.11)$$

olacaktır. Örnek (3.7)'deki maliyet değeri ise  $-35$ 'tir.

Eğer (3.9)'da temel olurlu çözüm yerine başka bir olurlu çözümü kullanırsak  $x_{m+1}, \dots, x_n$  değişkenlerinin tamamı sifıra eşit olmayacaktır. Bu durumda ilk  $m$  değişken diğerleri cinsinden aşağıdaki gibi ifade edilebilir

$$\begin{aligned} x_1 &= y_{1,n+1} - \sum_{j=m+1}^n y_{1j} x_j \\ x_2 &= y_{2,n+1} - \sum_{j=m+1}^n y_{2j} x_j \\ &\vdots \\ x_m &= y_{m,n+1} - \sum_{j=m+1}^n y_{mj} x_j. \end{aligned} \quad (3.12)$$

Eğer (3.12), (3.11) içinde kullanılırsa maliyet fonksiyonu:

$$z = z_b + x_{m+1}(c_{m+1} - w_{m+1}) + x_{m+2}(c_{m+2} - w_{m+2}) + \dots + x_n(c_n - w_n) \quad (3.13)$$

olarak elde edilir. Burada  $w_k$

$$w_k = \sum_{i=1}^n y_{ik} c_i$$

olarak tanımlanmıştır. (3.13)'den görüleceği gibi sıfırdan küçük olan  $(c_k - w_k)$  değerlerine karşılık gelen  $x_k$  değerlerini arttırmak (3.13)'deki yeni maliyet işlevini azaltacaktır. Simpleks Algoritması'ndaki değişkenlerin sıfırdan büyük olma şartı bundan kaynaklanmaktadır. Yeni maliyet fonksiyonundan ortaya çıkan başka bir sonuç ise tüm  $(c_k - w_k)$  değerleri sıfırdan büyük olduğunda maliyet fonksiyonunun en küçük değerine ulaşılmış olacaktır, çünkü pozitif  $(c_k - w_k)$  değerleri ile maliyet işlevi daha fazla düşürülemez.

Azaltılmış maliyet katsayısı  $r_k$  şöyle tanımlansın

$$r_k = c_k - w_k. \quad (3.14)$$

Bu durumda  $r_k < 0$  olan herhangi bir değişkenin ( $x_k$ ) değeri arttırılarak maliyet işlevi değeri azaltılabilir. Seçilen değişkenin indis değeri  $q$  olsun

$$q = \arg \min r_k.$$

Seçilen  $x_q$  değişkeninin değeri arttırılırken (dolayısıyla maliyet işlevinin değeri düşürülürken) aynı zamanda kısıtlar da sağlanmalıdır. Örneğin temel değişken  $x_p$

$$x_p = y_{p,n+1} - \sum_{j=m+1}^n y_{pj} x_j \quad (3.15)$$

olarak da ifade edilebilir. Değişken  $x_q$ 'nin değeri arttıkça  $x_p - y_{pq} x_q$  kadar değişmektedir. Eğer  $y_{pq} < 0$  ise  $x_q$  arttıkça  $x_p$  üzerindeki  $x_1 \geq 0$  kısıtı sağlanamamaktadır. Bu durum her değişken için geçerli ise problem sınırsız olup bir çözümü bulunamayacaktır. Eğer  $y_{pq} > 0$  ise  $x_q$  arttıkça  $x_p$  azalacaktır. Değişken  $x_q$ 'nin alabileceği en büyük değer  $y_{p,n+1}/y_{pq}$  olacaktır. Çünkü  $x_q$  değişkeni  $x_p = 0$  olana kadar artabilir ( $x_i \leq 0$  olamaz). Bu durum diğer değişkenler için de geçerli olduğundan  $x_q$  değişkeninin uyması gereken kısıt

$$x_q \leq \min \frac{y_{i,n+1}}{y_{i,q}}, i = 1, 2, \dots, m.$$

olacaktır.

Pozitif olan  $y_{iq}$  değişkenleri için

$$p = \arg \min_i \frac{y_{i,n+1}}{y_{i,q}}$$

olarak tanımlansın. Eğer

$$x_q = \frac{y_{p,n+1}}{y_{pq}}$$

ile tanımlanırsa  $x_p$  değişkeni sifıra eşitlenirken  $x_q$  değişkeni temel değişken olacaktır. Böylece pivot işlemi maliyet fonksiyonunun değerini düşürecek şekilde gerçekleştirilmiş olur ve  $x_p$  temel değişkenlikten çıkararak  $x_q$  yeni temel değişken olmuştur.

### 3.1.2 Doğrusal Tamsayı Programlama

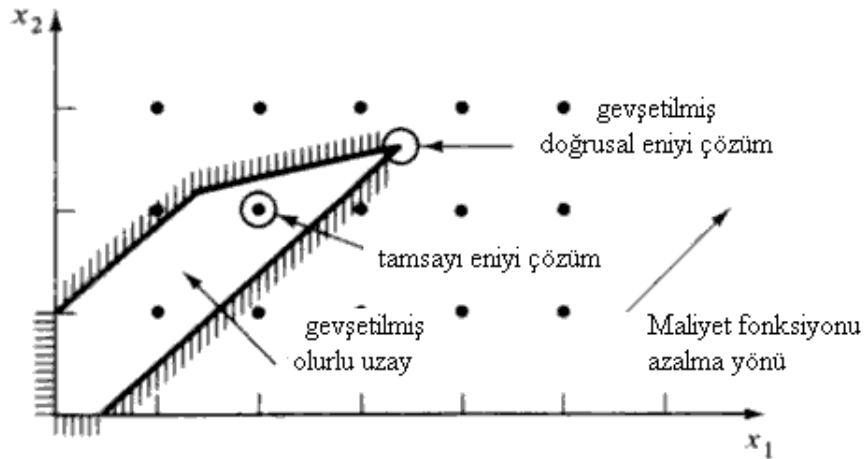
Tüm değişkenlerin tamsayı değerler kabul ettiği doğrusal programlama problemleri doğrusal Tamsayı Programlama olarak adlandırılır. Doğrusal Tamsayı Programlama problemleri matematiksel olarak aşağıdaki gibi ifade edilmektedir:

$$\begin{array}{ll} \text{maliyet işlevi} & \mathbf{c}^T \mathbf{x} \\ \text{kısıtlar} & \mathbf{Ax} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \\ & \mathbf{x}: \text{tamsayı} \end{array} \quad (3.16)$$

Tüm değişkenlerin 0 ya da 1 olma zorunluluğu olan doğrusal tamsayı problemleri ise *0-1 programlama* ya da *ikili eniyileme* olarak tanımlanmaktadır.

Tanım kümesinin ayrık doğası nedeniyle dışbükey olmayan Doğrusal Tamsayı Programların çözümü doğrusal programların çözümlerine göre daha zor ve karmaşıktır. Tamsayı Programlama'da eniyi çözüm kaba kuvvet yöntemiyle bulunabilir. Kaba kuvvet yöntemi çözüm uzayındaki her elemanı tek tek deneyerek eniyi çözüme ulaşmaktadır. Fakat bu yöntemin pratikte özellikle büyük olurlu kümelerde çok yavaş kalacağı açıktır.

Tamsayı Programlama'daki kısıtların tamsayı olma zorunluluğunun kaldırılmasına *gevşetme* denilmektedir. Gevşetme sonucu bulunan sonuç, eniyi olurlu tamsayı çözüm için bir alt-sınır oluşturmaktadır. Çünkü eniyileme için kullanılan uzay tamsayı uzaydan daha geniş bir alanı taramakta ve muhtemelen tamsayı uzay tarafından olursuz tanımlanan bölgeleri olurlu kabul etmektedir. Bu durum Şekil 3.3'de gösterilmiştir.



Şekil 3.3: Tamsayı Programlama ve gevşetilmiş problemin eniyi çözümleri [13].

Şekil 3.3’de bir eniyileme probleminin olurlu tamsayı eniyi çözümü ile tamsayı kısıtının gevşetildiği durumdaki eniyi çözüm gösterilmiştir.

Kısıtları sağlayan olurlu bir çözüm bulabilmek için doğrusal programlama sonucu elde edilen çözüm en yakın tamsayıya yuvarlanarak yeni bir tamsayı çözüm bulunabilir. Fakat pratikte elde edilen yeni çözüm kısıtları sağlayamayabilir.

Eğer kısıtları gevşetilmiş tamsayı programlama probleminin eniyi olurlu çözümündeki tüm değişkenler tamsayı ise tamsayı eniyileme problemi de çözülmüş olur. Buradan hareketle çeşitli eniyileme yöntemleri geliştirilmiştir. Bunlardan en sık kullanılanları Dallar ve Sınırla, Düzlem Kesme ve bunların melezi olan Dallar ve Kes yöntemleridir.

### Dallar ve Sınırla Yöntemi

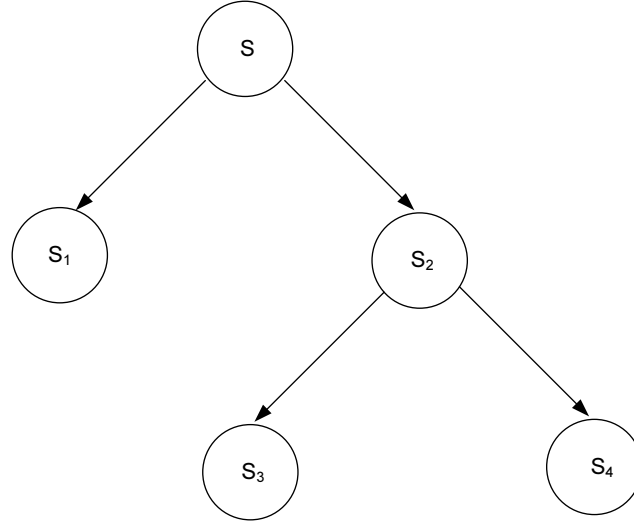
Dallar ve Sınırla Yöntemi "Böl ve Yönet" ilkesi ile problemi alt problemlere ayırır ve alt problemdeki uygun noktaları tarayıp olurlu kümedeki tüm noktalara bakmaya gerek duymadan eniyi çözümü sağlar.

Burada anlatılacak Dallar ve Sınırla Algoritması’nda eniyilenecek problemin doğrusal programlama yapısında olduğu varsayılmaktadır. Yinelemeli bir yapıya sahip Dallar ve Sınırla Yöntemi’nde her yineleme temelde dallanma işlemi, sınırlama ve işlem yapılacak noktanın seçilmesi olmak üzere üç aşamadan oluşmaktadır.

Dallandırma işlemi ile problem alt problemlere bölünmektedir. Çeşitli kısıtlar gevşetilerek gerçekleştirilen doğrusal programlama sonucunda dallandırma işlemi

gerçekleştirilir. Kısıtları sağlayan her çözüm bir aday çözümdür. Eğer alt problemde olurlu bir çözüm elde edilemezse o noktanın dallandırılmasına izin verilmez. Çünkü o noktadan sonra yapılacak dallandırmalar da olurlu olmayacaktır. Diğer noktalar için ise algoritma sonlanana kadar dallandırma işlemi devam etmektedir.

Dallan ve Sınırla Yöntemi ağaç yapısını kullanmaktadır. Şekil 3.4'de tipik bir ağaç yapısı gösterilmiştir:



Şekil 3.4: Dallan ve Sınırla Algoritmasında olurlu kümenin ağaç yapısı ile dallandırılması [40].

Burada  $S$  olurlu kümeyi göstermektedir. Yöntemde  $S$  kümesi dallara ayrılmaktadır. Dallandırma için iki farklı seçenek vardır. İlk seçenekte herhangi bir noktadan daha aşağı seviyede incelenmemiş noktalar varsa o noktaların seviyesine inilerek en sona gidene kadar dallandırma işlemine devam edilmektedir. Diğer seçenekte ise herhangi bir noktadayken, o seviyedeki noktaların maliyet fonksiyonlarına bakılarak en düşük maliyete sahip nokta seçilir ve dallandırma işlemi o nokta üzerinden yapılır.

Çeşitli tamsayı kısıtlarının gevşetilmesiyle oluşturulan alt kümelerin çözülmesi sonucu ortaya çıkan çözüm eğer tamsayı değil ise problem için bir alt sınır oluşturmaktadır. Başlangıçta üst sınır  $\infty$  olarak alınır. Algoritma çalışırken bulunan ilk tamsayı çözüm olurlu ise üst sınır olur. Yinelemeler sırasında herhangi bir aday çözümün maliyet işlevi değeri üst sınırdan daha yüksek ise o noktadan sonra dallandırma yapılmaz. Aksi halde algoritma çalışmaya devam eder. Örneğin yukarıda

şekilde  $S_4$  noktasında bir tamsayı çözüm bulunmuş olsun. Bu çözüm problemin üst sınırı olmaktadır.  $S_1$  noktasında ise tamsayı olmayan bir çözüm bulunsun. Eğer  $S_1$  noktasındaki çözüm ( $S_1$  noktasındaki alt sınır)  $S_4$  noktasının çözümden daha yüksek bir maliyet işlevine sahipse  $S_1$  noktasında dallandırma yapılmamaktadır. Çünkü doğrusal maliyet işlevi nedeniyle bu yönde ilerlemenin maliyeti azaltmayacağı açıktır.

Dallan ve sınırla Yöntemi'nde, problem ilk haliyle ana problem adını almaktadır :

Ana Problem

$$\begin{aligned} \text{maliyet işlevi} & \quad z = \mathbf{c}^T \mathbf{x} \\ \text{kısıtlar} & \quad \mathbf{Ax} \leq \mathbf{b} \\ & \quad \mathbf{x} \geq \mathbf{0}, \mathbf{x} \text{ tamsayı} \end{aligned} \quad (3.17)$$

Ana problemin en iyi tamsayı çözümü  $\mathbf{x}^*$  olsun. Ana problemdeki tamsayı kısıtının gevşetilmesiyle oluşan yeni problemin çözümü ise  $\mathbf{x}^0$  ile gösterilsin. Eğer  $\mathbf{x}^0$  çözümündeki her eleman tamsayı ise  $\mathbf{x}^0$  ve  $\mathbf{x}^*$  birbirlerine eşit olacaktır. Aksi durumda  $\mathbf{x}^0$  çözümünün maliyeti  $\mathbf{c}^T \mathbf{x}^0$  ana problem için bir alt sınır olacaktır. Bu durumda örneğin  $\mathbf{x}^0$  vektörünün  $i$  'ninci elemanı  $x_i^0$  kesirli bir sayı olsun. Ana problem  $x_i^0$  üzerinden aşağıdaki gibi iki alt probleme bölünebilir:

Problem 1

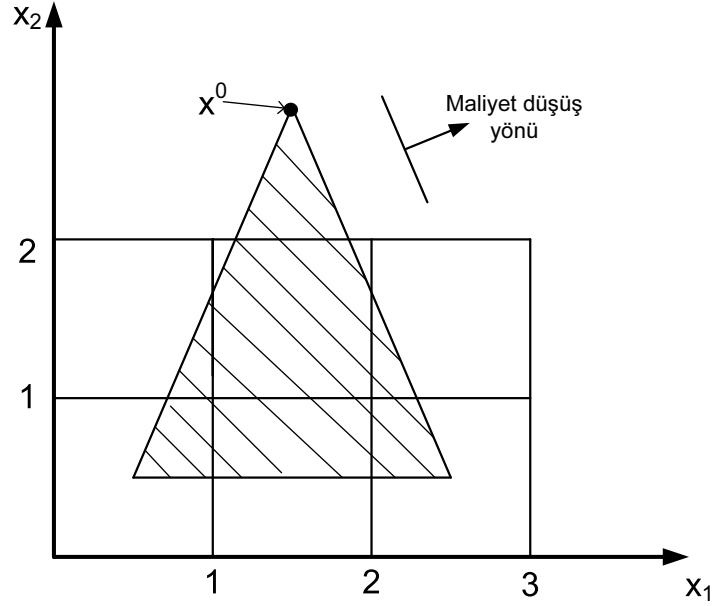
$$\begin{aligned} \text{maliyet işlevi} & \quad \mathbf{c}^T \mathbf{x} \\ \text{kısıtlar} & \quad \mathbf{Ax} \leq \mathbf{b} \\ & \quad \mathbf{x} \geq \mathbf{0}, \mathbf{x} : \text{ tamsayı} \\ & \quad x_i \leq \lfloor x_i^0 \rfloor \end{aligned} \quad (3.18)$$

Problem 2

$$\begin{aligned} \text{maliyet işlevi} & \quad \mathbf{c}^T \mathbf{x} \\ \text{kısıtlar} & \quad \mathbf{Ax} \leq \mathbf{b} \\ & \quad \mathbf{x} \geq \mathbf{0}, \mathbf{x} : \text{ tamsayı} \\ & \quad x_i \geq \lfloor x_i^0 \rfloor + 1 \end{aligned} \quad (3.19)$$

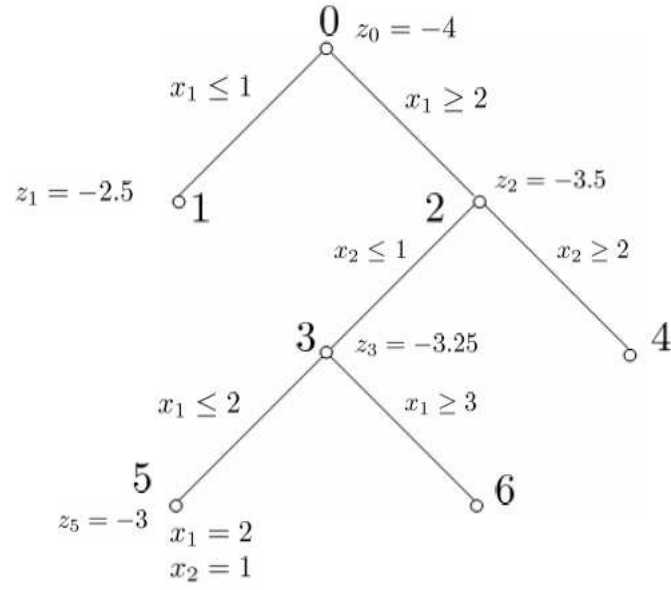
$\lfloor \cdot \rfloor$  işlevi işlenenin tamsayı kısmını döndürmektedir. Böylece Ana Problem, 2 alt probleme bölünmüş olmaktadır.



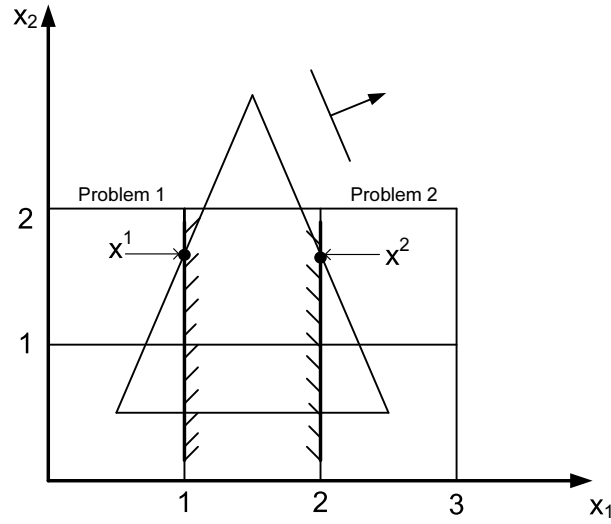


Şekil 3.5: Olurlu çözüm kümesi [13].

Örneğin amacımız  $x_1$  ve  $x_2$  değişkenlerinden oluşan  $-(x_1 + x_2)$  maliyet işlevinin en küçük değerini bulmak olsun. Kısıtlar ile olurlu alan Şekil 3.5'teki üçgenin içinde kalan alana sınırlandırılmıştır. Tamsayı kısıtının gevşetilmesiyle bulunan eniyi çözüm  $x_1 = 1.5$ ,  $x_2 = 2.5$  noktasıdır. Bu aşama Şekil 3.6'daki ağacın tepesi, yani "0" düğümüdür.  $x^0$  çözüm noktasında kesirli bir değere sahip  $x_1$  değişkeni üzerinden  $x_1 \leq 1$ ,  $x_2 \geq 2$  kısıtları kullanılarak Problem 1 ve Problem 2 oluşturulur. Bu adımdan sonra bir doğrusal programlama problemi olan Problem 1 çözülerek problem tekrar dallandırılır. İlk adımda  $x_1$  değişkeni gevşetilerek 1 ve 2 düğümleri oluşturulmuştur. Problem 1 ve 2'nin olurlu kümesi Şekil 3.7'de gösterilmektedir.

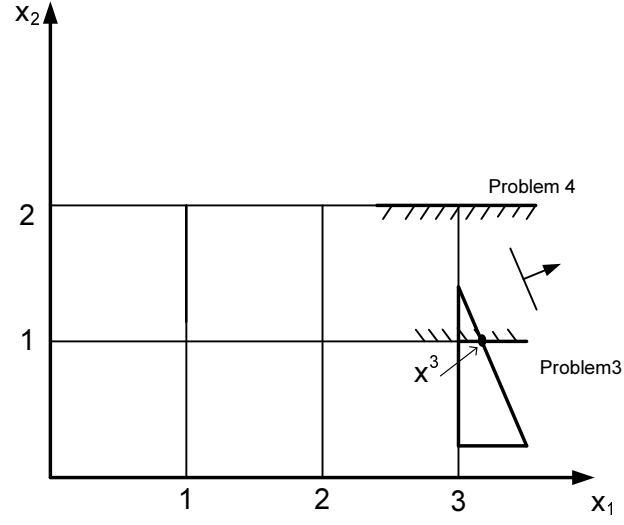


Şekil 3.6: Ağaç yapısı[13].



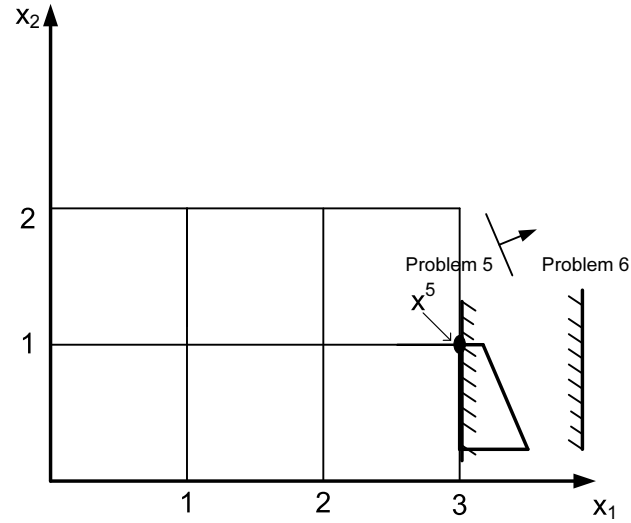
Şekil 3.7: Problem 1 ve Problem 2 [13].

Daha düşük bir maliyet işlevi değerine sahip olan 2 düğümünden bu sefer  $x_2$  değişkenine göre dallandırma işlemine devam edilerek Problem 3 ve Problem 4 elde edilir.



Şekil 3.8: Problem 3 ve Problem 4 [13].

Şekil 3.8'de görüldüğü gibi olurlu bir çözüme sahip olmayan 4 düğümünü dallandırma sonucunda oluşacak düğümler de olursuz olacağından bu noktada dallandırma işlemine devam edilmez. Dolayısıyla 3 noktasında dallandırma işlemi yapılarak Problem 5 ve Problem 6 elde edilir.



Şekil 3.9: Problem 5 ve Problem 6 [13].

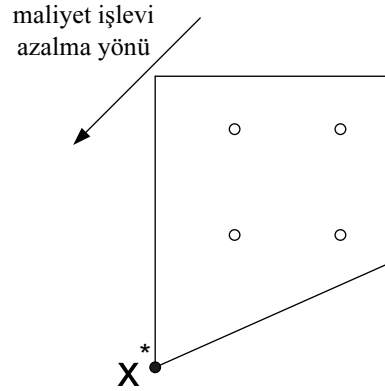
Algoritma 5 düğümünde olurlu bir çözüm bulmuştur. Bu düğümde daha düşük bir maliyet fonksiyonu değerine sahip bir nokta bulunmadığı için algoritma bu noktada sonlandırılır.

Dallan ve Sınırla Yöntemi'nde sıklıkla doğrusal programlamaya ihtiyaç duyulduğundan yöntemde kullanılan doğrusal programlama algoritmasının hızı kritik bir önem taşımaktadır.

### Düzlem Kesme Algoritması

Tamsayı programlama problemlerini çözmek için literatürde önerilen başka bir yöntem de Düzlem Kesme Algoritması'dır. Düzlem Kesme Algoritması, çıkış noktasını Simpleks Algoritması'ndan almaktadır. Algoritma her iterasyonda probleme yeni doğrusal kısıtlar ekleyerek, olurlu kümeyi tamsayı sonuçları koruyarak küçültür. Sonucu iterasyonda ise kısıtların eklendiği problemin çözümü ile orijinal tamsayı problemin çözümleri çakışmaktadır.

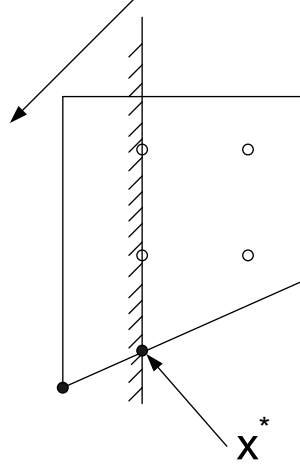
Örneğin, Şekil 3.10'da dörtgenin içindeki alan gevşetilmiş olurlu kümeyi göstermektedir. Kümede 4 adet tamsayı bulunmaktadır.



Şekil 3.10: Çözüm Kümesi [13].

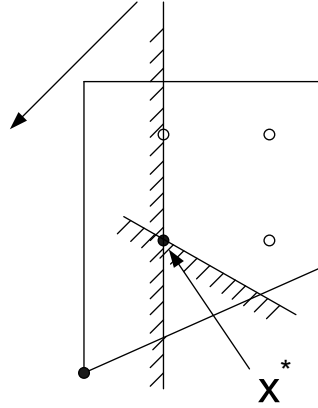
Herhangi bir tamsayı olurlu noktayı dışarıda bırakmadan olurlu kümeyi küçültme işlemi doğrusal kısıt düzlem kesme (veya sadece kesme) olarak adlandırılmaktadır. Şekil 3.11'de gösterilen birinci kesme sonucu doğrunun sol tarafındaki bölge çözüm uzayından çıkarılmıştır. Görüldüğü gibi dört adet tamsayı nokta halen

çözüm uzayı içindedir. Ancak kesme kısıtının probleme dahil edilmesiyle elde edilen problemin sonucu hala tamsayı değildir.



Şekil 3.11: Birinci kesme [13].

Şekil 3.12’de gösterilen ikinci kesme sonucu oluşan problemin sonucu ile orijinal Tamsayı Programlama problemin sonuçları çakışmaktadır. Böylece algoritma sona erer. Her iki kesme sonucunda da hiç bir tamsayı olurlu çözüm dışarıda kalmamıştır.



Şekil 3.12: İkinci kesme [13].

Bir kesme yöntemi olan Gomory Kesmeleri’nin çalışması aşağıda açıklanmaktadır.

Simpleks metodunda da kullanılan, (3.9)'daki tablo ifadesi kullanılırsa;  $i \in \{1, 2, \dots, m\}$  olmak üzere en son yinelemedeki tabloda

$$x_i + \sum_{j=m+1}^n y_{ij}x_j = y_{i,n+1} \quad (3.20)$$

denklemin geçerliliğini korumaktadır. Problemin tanımındaki kısıtlar  $\mathbf{x}$ 'in elemanlarının negatif olmayan tamsayı değerler almasını zorunlu kıldığı için

$$\sum_{j=m+1}^n \lfloor y_{ij} \rfloor x_j \leq \sum_{j=m+1}^n y_{ij}x_j \quad (3.21)$$

eşitsizliği sağlanmak zorundadır. Eşitsizlik (3.21), (3.20) eşitliği ile birleştirilirse

$$x_i + \sum_{j=m+1}^n \lfloor y_{ij} \rfloor x_j \leq y_{i,n+1} \quad (3.22)$$

elde edilir. Tamsayı kısıtı sebebiyle yukarıdaki eşitsizliğin sol tarafı da tamsayı olmak zorundadır. Dolayısıyla sağ tarafın kesirli kısmı alınsa da eşitsizlik bozulmaz

$$x_i + \sum_{j=m+1}^n \lfloor y_{ij} \rfloor x_j \leq \lfloor y_{i,n+1} \rfloor. \quad (3.23)$$

Bu eşitsizlik, (3.20) denkleminde çıkarılırsa

$$\sum_{j=m+1}^n (y_{ij} - \lfloor y_{ij} \rfloor) x_j \geq y_{i,n+1} - \lfloor y_{i,n+1} \rfloor \quad (3.24)$$

elde edilir.

Kesirli kısmı göstermek için  $f_{ij}$

$$f_{ij} = y_{ij} - \lfloor y_{ij} \rfloor \quad (3.25)$$

terimi tanımlansın. Açıkça  $f_{ij}$

$$0 \leq f_{ij} \leq 1 \quad (3.26)$$

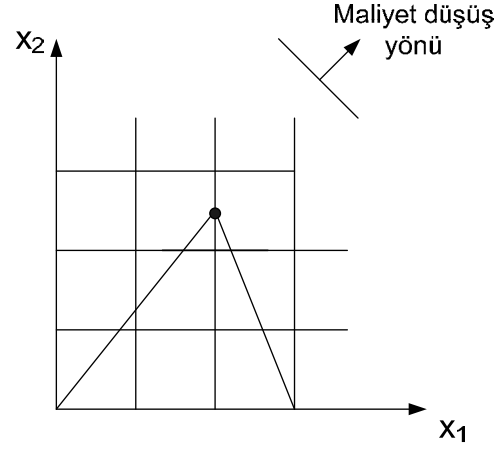
aralığını sağlayacaktır. Eşitsizlik (3.24) ile (3.25) eşitliği birleştirilirse Gomory kesmesi olarak adlandırılan

$$\sum_{j=m+1}^n f_{ij}x_j \geq f_{i,n+1} \quad (3.27)$$

eşitsizliği edilir. Bu eşitsizlik her iterasyonda ana probleme bir kısıt olarak eklenmektedir.

Örnek olarak aşağıdaki tamsayı programlama problemi çözülecektir [41]

$$\begin{aligned}
 \min \quad & z = -(x_1 + x_2) \\
 \text{kısıtlar} \quad & -5x_1 + 4x_2 \leq 0 \\
 & 5x_1 + 2x_2 \leq 15 \\
 & x_1, x_2 \geq 0, \text{ tamsayı.}
 \end{aligned} \tag{3.28}$$



Şekil 3.13: Olurlu çözüm kümesi.

Yukarıdaki problemde tamsayı kısıtların gevşetildiği ve arttıran artık değişkenler eklenerek eşitlik kısıtlı hale dönüştürüldüğü doğrusal programda,  $y_1, y_2$  artık değişkenler olmak üzere

$$x_1 = 2, x_2 = 2.5, y_1 = 0, y_2 = 0$$

eniye çözümü elde edilir. Simpleks tablosunun son hali aşağıdaki gibidir

$$\begin{bmatrix}
 x_1 & x_2 & y_1 & y_2 & b_i \\
 0 & 1 & 1/6 & 1/6 & 5/2 \\
 1 & 0 & -1/15 & -1/15 & 2 \\
 0 & 0 & -1/10 & -3/10 & -9/2 & z
 \end{bmatrix}. \tag{3.29}$$

Tablodaki ilk satırı kapsayan denklem

$$x_2 + \frac{y_1}{6} + \frac{y_2}{6} = 5/2 \tag{3.30}$$

şeklinindedir. Gerekli manipülasyonlar ile sağ taraf tamsayı ve kesirli olmak üzere ikiye ayırılırsa

$$\frac{y_1}{6} + \frac{y_2}{6} = 1/2 + (2 - x_2)$$

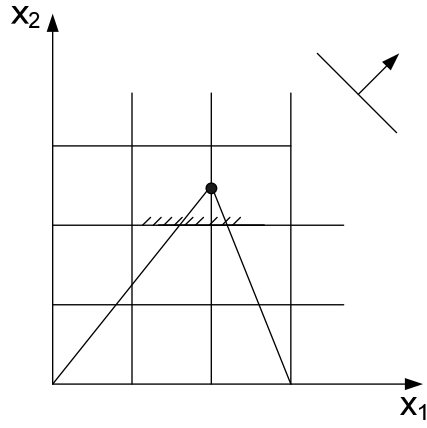
olur. Sağ taraf negatif olamayacağından

$$1/2 + (2 - x_2) \geq 0$$

olur. Eşitsizlik (3.30) ile birleştirilirse

$$\frac{y_1}{6} + \frac{y_2}{6} \geq 0$$

eşitsizliği elde edilir. Bu eşitsizlik  $x_2 \leq 2$  eşitsizliği ile aynı etkiyi göstermektedir ve tabloda  $y_3$  değişkeni ile gösterilecektir. Böylece ilk kesme elde edilmiş olur.



Şekil 3.14: Birinci kesme

$$\left[ \begin{array}{cccccc} x_1 & x_2 & y_1 & y_2 & y_3 & b_i \\ 0 & 1 & 1/6 & 1/6 & 0 & 5/2 \\ 1 & 0 & -1/15 & -1/15 & 0 & 2 \\ 0 & 0 & 1/6 & -1/6 & -1 & 1/2 \\ 0 & 0 & -1/10 & -3/10 & 0 & -9/2 & z \end{array} \right] \quad (3.31)$$

Daha sonra tabloya temel değişken olarak  $y_1$  veya  $y_2$  getirilebilir. Eğer  $y_2$  temel değişken olarak tabloya girerse tablo

$$\left[ \begin{array}{cccccc} x_1 & x_2 & y_1 & y_2 & y_3 & b_i \\ 0 & 1 & 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 1/5 & -2/5 & 11/5 \\ 0 & 0 & 1 & 1 & -6 & 3 \\ 0 & 0 & 0 & -1/5 & -4/5 & -21/5 & z \end{array} \right]$$



haline gelir. Bu tablonun en iyi çözümü ise

$$x_1 = 11/5, x_2 = 2, y_1 = 3, y_2 = 0, y_3 = 0$$

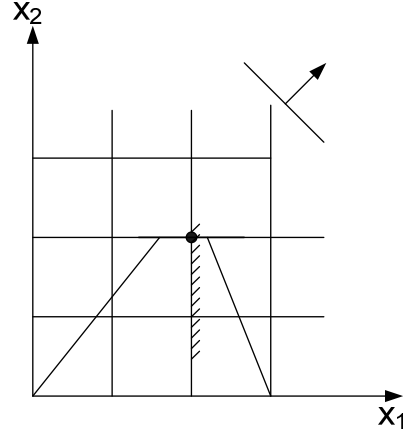
olur. Bu sefer çözümünde tamsayı olmayan değer içeren 2. satır seçilsin;

$$x_1 + y_2/5 - 2y_3/5 = 11/5$$

denkleminde gerekli manipülasyonlar yapılırsa

$$y_2/5 + 3y_3/5 \geq 1/5$$

eşitsizliği elde edilir. Bu eşitsizlik  $x_1 \leq 2$  eşitsizliğine denktir. Böylece Şekil 3.15'te verilen ikinci kesme elde edilir.



Şekil 3.15: İkinci kesme.

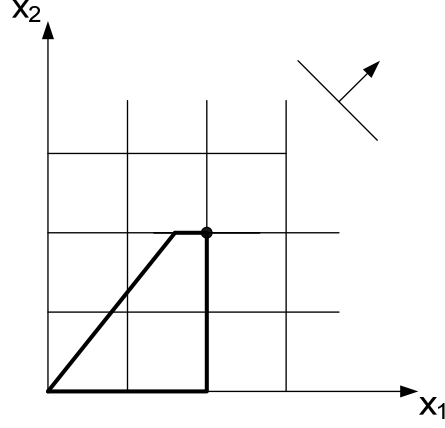
Tablonun son hali aşağıdaki gibidir

$$\begin{bmatrix} x_1 & x_2 & y_1 & y_2 & y_3 & y_4 & b_i \\ 0 & 1 & 0 & 0 & 1 & 0 & 2 \\ 1 & 0 & 0 & 0 & -1 & 1 & 2 \\ 0 & 0 & 1 & 0 & -9 & 5 & 2 \\ 0 & 0 & 0 & 1 & 3 & -5 & 1 \\ 0 & 0 & 0 & 0 & -3/5 & -1 & -4 \end{bmatrix} z$$

Bu tablonun en iyi çözümü ise

$$x_1 = 2, x_2 = 2, y_1 = 2, y_2 = 1, y_3 = 0, y_4 = 0$$

olmaktadır. Görüldüğü gibi tüm değişkenler tamsayı değerler almaktadır. Dolayısıyla eniyi tamsayı sonuç bulunarak problem çözülmüş olur.



Şekil 3.16: Son durum

Pratikte karşılaşılan problemlerde, en iyi olurlu tamsayı sonucu elde edebilmek için çok fazla kesme yapılması gerektiği görülmüştür. Kesme sırasında kesirli değerler kullanıldığından bu da ayrı bir problem oluşturmaktadır. Düzlem Kesme Yöntemi'nin tek başına kullanılması eniyi tamsayı çözümü kısa sürede bulmada etkin bir metod değildir.

### Dallan ve Kes Algoritması

Düzlem Kesme Algoritması tek başına kullanıldığında eniyi olurlu çözüme ulaşmak oldukça uzun sürmektedir. Aslında, literatürde de düzlem kesme algoritmasının tek başına kullanılmadığı görülmüştür. Dallan ve Kes Algoritması ise hem Dallan ve Sınırla hem de Düzlem Kesme Algoritmaları'nı içeren melez bir algoritmadır. Düzlem Kesme Yöntemi olurlu uzayı daraltarak Dallan ve Sınırla Algoritmasının hızlanmasını sağlamayı amaçlamaktadır. Algoritma ilk aşamada Düzlem Kesme ile Dallan ve Sınırla yöntemlerinde olduğu gibi öncelikle değişkenlerin tamsayı olma kısıtını gevşeterek bir ilk sonuç bulmaktadır. Dallan ve Kes algoritması da bir örnekle açıklanacaktır. Örnekteki maliyet fonksiyonu ve kısıtlar aşağıdaki gibidir [14].

## Problem 0

$$\begin{array}{ll}
\text{maliyet işlevi} & -5x_1 - 6x_2 \\
\text{kısıtlar} & x_1 + 2x_2 \leq 7 \\
& 2x_1 - x_2 \leq 3 \\
& x_1, x_2 \geq 0, \text{ tamsayı}
\end{array} \tag{3.32}$$

Öncelikle tamsayı kısıtı kaldırılarak ilk çözüm elde edilir. Bu çözüm  $-26.2$  maliyet değerine sahip olan  $x_1 = 2.6, x_2 = 2.2$  noktasıdır. Bu noktadan sonra problem  $x_2$  kısıtı üzerinden ikiye ayrılır.

## Problem 1

$$\begin{array}{ll}
\text{maliyet işlevi} & -5x_1 - 6x_2 \\
\text{kısıtlar} & x_1 + 2x_2 \leq 7 \\
& 2x_1 - x_2 \leq 3 \\
& x_2 \geq 3 \\
& x_1, x_2 \geq 0.
\end{array} \tag{3.33}$$

## Problem 2

$$\begin{array}{ll}
\text{maliyet işlevi} & -5x_1 - 6x_2 \\
\text{kısıtlar} & x_1 + 2x_2 \leq 7 \\
& 2x_1 - x_2 \leq 3 \\
& x_2 \leq 2 \\
& x_1, x_2 \geq 0.
\end{array} \tag{3.34}$$

Problem 1'in çözümü  $-23$  maliyet değerine sahip  $x_1 = 1, x_2 = 3$  noktalarıdır. Böylece tamsayı olurlu bir çözüm bulunmuştur. Fakat Problem 2 de incelenmeden bu çözümün eniyi çözüm olup olmadığı belli olmayacaktır.

Problem 2'nin çözümü ise  $-24.5$  maliyet değerine sahip  $x_1 = 2.5, x_2 = 2$  noktasıdır. Bu noktadaki çözümdeki değerlerin tamamı tamsayı değildir. Dallon ve Sınırla Algoritması bu noktayı tekrar dallandırmak isteyecektir. Fakat Dallon ve Kes algoritması probleme

$$x_1 + 2x_2 \leq 6$$

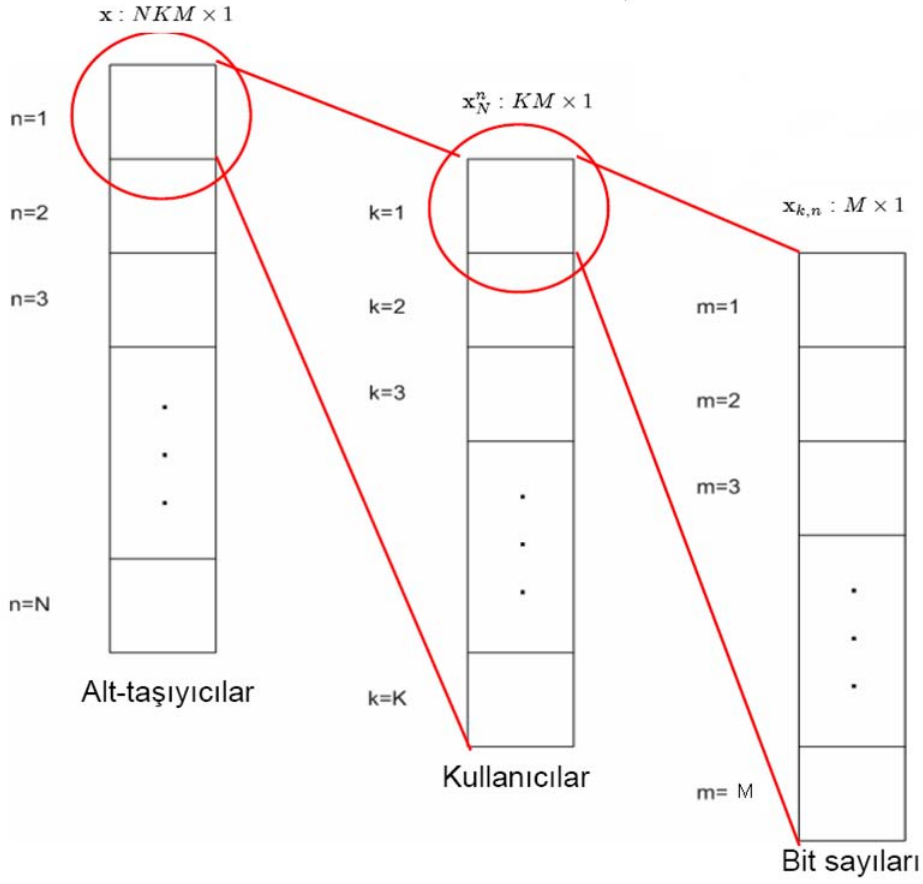
kısıtını ekler. Kısıt çözüm uzayındaki tüm tamsayılar tarafından sağlandığı için geçerli bir kısıttır. Problem 2'ye bu kısıtın eklenmesi ile elde edilen yeni problemin çözümüm  $-22.6$  maliyet değerine sahip  $x_1 = 2.4, x_2 = 1.8$  noktasıdır. Halen bir tamsayı olurlu bir çözüm elde edilemediği halde bu noktanın maliyet değeri Problem 1'in maliyet değerinden daha fazla olduğu için bu noktadan aşağıdaki seviyelerde bir tamsayı çözüm bulunsa bile bulunan çözüm eniyi çözüm olmayacaktır. Dolayısıyla en iyi çözüm Problem 1'de bulunmuştur. Algoritma burada sonlandırılır.

## 4. TAMSAYI PROGRAMLAMA İLE KAYNAK TAHSİSİ

Bu bölümde öncelikle, doğrusal olmayan OFDMA sistemlerde kaynak tahsisi problemi Mao tarafından önerilen yöntemle [36] doğrusal hale getirilecektir. OFDMA sistemlerde kaynak tahsisi problemi aslen doğrusal olmayan bir problem olmasına rağmen eniyiliği yitirilmeden doğrusallaştırılabilmektedir. Kim ve Lee kaynak tahsisi problemini [35]'de doğrusal hale getirmiş ve Mao ile Wang [36] ise doğrusallaştırılmış problemi matris ve vektörler ile ifade etmişlerdir. Tezin devam eden kısmında bu formülasyondan yararlanılacaktır.

### 4.1 Kaynak Tahsisi Probleminin Doğrusallaştırılması

Bölüm 2.2.2'de açıklandığı üzere güç fonksiyonu  $f(c_{k,n})$  doğrusal olmayan bir fonksiyon olmasının yanı sıra  $c_{k,n}$  sadece tamsayı değerler alabilmektedir. Ayrıca belirli bir BER ve  $c_{k,n}$  değeri için ihtiyaç duyulan güç miktarı da belirlidir. Buradan hareketle  $c_{k,n}$  değerlerinin olası tüm değerleri kullanılarak sabit bir güç vektörü ve buna karşılık gelen değişkenlerin oluşturduğu bir alt-taşıyıcı-kullanıcı-bit vektörü Mao [36] tarafından önerilen modeldeki gibi tanımlanabilir. Öncelikle değişkenleri içeren  $\mathbf{x}$  vektörünün yapısını ele alalım. Vektörün yapısı Şekil 4.1'de gösterilmiştir.



Şekil 4.1: Bit tahsisi vektörünün açılımı. Vektör alt-vektörlerden oluşmaktadır.

Şekilde 4.1'de en sağdaki  $M \times 1$  boyutundaki alt-taşıyıcı-kullanıcı-bit vektörü  $\mathbf{x}_{k,n}$ ,  $n$ 'inci alt-taşıyıcıda  $k$  kullanıcıya kaç bit atandığı bilgisini taşımaktadır. Bu vektör  $n$ 'inci alt-taşıyıcıdaki tüm kullanıcılar için birleştirilirse,  $KM \times 1$  boyutlarındaki alt-taşıyıcı vektörü  $\mathbf{x}_N^n$  meydana gelmektedir. Alt-taşıyıcı vektörleri de her frekans parçası için birleştğinde karar değişkeni olan bit-tahsis vektörü  $\mathbf{x}$ 'i oluşturmaktadır.

Alt-taşıyıcı-kullanıcı-bit vektörü  $\mathbf{x}_{k,n}$  şöyle tanımlanmıştır

$$\mathbf{x}_{k,n} = \begin{bmatrix} x_{k,n,1} \\ x_{k,n,2} \\ \vdots \\ x_{k,n,M} \end{bmatrix} \quad (4.1)$$

$$k = 1, 2, \dots, K, \quad n = 1, 2, \dots, N, \quad \mathbf{x}_{k,n} \in \{0, 1\}^{M \times 1}.$$

Burada  $x_{k,n,m}$  sadece 1 veya 0 değerlerini alabilmektedir. Örneğin  $x_{k,n,m}$ 'nin 1 olması

$k$  kullanıcısının  $n$ 'inci alt-taşıyıcısına  $m$  adet bit atandığı anlamına gelmektedir. Alt-taşıyıcı-kullanıcı-bit vektöründe en fazla bir tane 1 değeri olabilir.  $\mathbf{x}_{k,n}$  vektöründe tüm değerlerin 0 olması  $k$  kullanıcısı için  $n$ 'inci alt-taşıyıcıya hiç bit atanmayacağı anlamına gelmektedir. Dolayısıyla  $\mathbf{x}_{k,n}$  vektörünün değerler kümesi

$$\mathbf{x}_{k,n} \in \left\{ \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 1 \end{pmatrix} \right\}_{M \times M+1}$$

olmaktadır.

Her bir alt-taşıyıcı için alt-taşıyıcı vektörü  $\mathbf{x}_N^n = [\mathbf{x}_{1,n}^T \ \mathbf{x}_{2,n}^T \ \dots \ \mathbf{x}_{K,n}^T]^T$  ( $n = 1, 2, \dots, N, x_N^n \in \{0, 1\}^N$ ) ise  $n$ 'inci alt-taşıyıcı için hangi kullanıcıya ne kadar bit atandığı bilgisini taşımaktadır. Bir alt-taşıyıcı sadece bir kullanıcıya atanabildiğinden  $\mathbf{x}_N^n$  vektörü içinde de en fazla bir tane 1 değeri olabilir. Bu durumda,  $\mathbf{x}_N^n$  vektörünün değerler kümesi

$$\mathbf{x}_N^n \in \left\{ \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 1 \end{pmatrix} \right\}$$

olmaktadır.

Bit-tahsis vektörü olarak tanımlanan  $\mathbf{x}$  ise  $NKM \times 1$  boyutlarındadır ve aşağıdaki gibidir

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_N^1 \\ \mathbf{x}_N^2 \\ \vdots \\ \mathbf{x}_N^N \end{bmatrix}$$

$$\mathbf{x} \in \{0, 1\}^{NKM \times 1}.$$

Bit-tahsis vektörü  $\mathbf{x}$ ,  $c_{k,n}$  ile ilgili tüm bilgileri bünyesinde barındırmaktadır. Böylece tüm bit tahsis olasılıkları bir vektör değişkeni ile temsil edilebilmektedir. Bit-tahsis vektörünün elemanları sadece 0 veya 1 değerlerini alabildikleri için anahtarlama elemanları olarak da düşünülebilirler.

Örneğin, alt-taşıyıcı sayısı  $N = 3$ , kullanıcı sayısı  $K = 2$ , ve herhangi bir alt-taşıyıcı için bir kullanıcıya atanabilecek en çok bit miktarı  $M = 2$  için örnek bit tahsis vektörü aşağıda gösterilmiştir

$$\mathbf{x} = \begin{bmatrix} 0 \\ 1 \\ - \\ 0 \\ 0 \\ - \\ 0 \\ 0 \\ - \\ 0 \\ 1 \\ - \\ 0 \\ 0 \\ - \\ 0 \\ 0 \end{bmatrix} .$$

Örnek bit-tahsis vektörü incelendiğinde birinci alt-taşıyıcıda ilk kullanıcıya 2 bit atandığı görülmektedir. İkinci alt-taşıyıcıda ise ikinci kullanıcıya 2 bit atanmıştır. Üçüncü alt-taşıyıcıda ise hiç bir kullanıcıya bit atanmamıştır. Örnekte her alt-taşıyıcı sadece bir kullanıcıya tahsis edilmiştir. Dolayısıyla aynı taşıyıcının birden fazla kullanıcı tarafından kullanılmaması kuralı korunmaktadır.

Maliyet işlevi ise bit-tahsis vektörü ile uyumlu olacak ve dolayısıyla vektörel hale getirilecek şekilde düzenlenmektedir. Herhangi bir  $\mathbf{x}$  bit tahsis vektörünün maliyetini hesaplamak için alt-taşıyıcı-kullanıcı-güç vektörü  $\mathbf{p}_{k,n}$  tanımlanmıştır. Bu vektörün  $\mathbf{x}_{k,n}$  vektörüyle içsel çarpımı sonucunda  $(\mathbf{p}_{k,n}^T \mathbf{x}_{k,n})$ ,  $n$ 'inci alt-taşıyıcıda  $k$

kullanıcısının  $m$  adet bit göndermek için ihtiyaç duyduğu güç bulunur

$$\mathbf{p}_{k,n} = \begin{bmatrix} p_{k,n,1} \\ p_{k,n,2} \\ \vdots \\ p_{k,n,M} \end{bmatrix} \quad (4.2)$$

$k = 1, 2, \dots, K$ ,  $n = 1, 2, \dots, N$ ,  $\mathbf{p}_{k,n} \in \mathbb{R}^{M \times 1}$ .

Alt-taşıyıcı-kullanıcı güç vektörünün elemanları  $p_{k,n,m}$  ise aşağıdaki gibi tanımlanmıştır:

$$p_{k,n,m} = \frac{f(m)}{\alpha_{k,n}^2}. \quad (4.3)$$

Alt-taşıyıcı güç vektörü  $\mathbf{p}_N^n = [\mathbf{p}_{1,n}^T \ \mathbf{p}_{2,n}^T \ \dots \ \mathbf{p}_{K,n}^T]^T$  ise  $n$ 'inci alt-taşıyıcıdaki güç tahsisi bilgisini taşımaktadır. Bit-tahsisi vektöründe olduğu gibi alt-taşıyıcı vektörü tüm alt-taşıyıcılara yayılarak  $NKM \times 1$  boyutlarındaki güç vektörü

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_N^1 \\ \mathbf{p}_N^2 \\ \vdots \\ \mathbf{p}_N^N \end{bmatrix} \quad (4.4)$$

olarak elde edilmektedir. Bit-tahsisi vektörünün devriği ile güç vektörünün çarpımı  $\mathbf{p}^T \mathbf{x}$ ,  $\mathbf{x}$  vektörünün maliyetinin değerini vermektedir.

İncelenen  $\mathbf{x}$  çözümünün veri hızı kısıtını sağlayıp sağlamadığını göstermek için  $\mathbf{A}_u$  matrisi aşağıdaki gibi tanımlanmıştır

$$\mathbf{A}_u = \begin{bmatrix} \mathbf{a}_{K1} & \mathbf{a}_{K1} & \cdots & \mathbf{a}_{K1} \\ \mathbf{a}_{K2} & \mathbf{a}_{K2} & \cdots & \mathbf{a}_{K2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{a}_{KK} & \mathbf{a}_{KK} & \cdots & \mathbf{a}_{KK} \end{bmatrix}, \mathbf{A}_u \in \mathbb{Z}^{K \times KMN}. \quad (4.5)$$

Burada

$$\begin{aligned} \mathbf{a}_{K1} &= [\mathbf{a}_u \ \mathbf{0}_u \ \cdots \ \mathbf{0}_u], \mathbf{a}_{K1} \in \mathbb{Z}^{1 \times KM} \\ \mathbf{a}_{K2} &= [\mathbf{0}_u \ \mathbf{a}_u \ \cdots \ \mathbf{0}_u], \mathbf{a}_{K2} \in \mathbb{Z}^{1 \times KM} \\ &\vdots \\ \mathbf{a}_{KK} &= [\mathbf{0}_u \ \mathbf{0}_u \ \cdots \ \mathbf{a}_u], \mathbf{a}_{KK} \in \mathbb{Z}^{1 \times KM} \end{aligned} \quad (4.6)$$



ve

$$\mathbf{a}_u = \begin{bmatrix} 1 & 2 & \cdots & M \end{bmatrix}, \mathbf{a}_u \in \mathbb{Z}^{1 \times M} \quad (4.7)$$

$$\mathbf{0}_u = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}, \mathbf{a}_u \in \{0\}^{1 \times M}.$$

Veri hızı kısıtı matrisi  $\mathbf{A}_u$   $K$  satırdan oluşmaktadır. Her satır bir kullanıcıya atanan toplam veri hızını göstermektedir. Kullanıcılara atanan toplam veri hızları kısıtının değeri ise aşağıdaki  $\mathbf{r}$  vektörüyle belirtilmiştir:

$$\mathbf{r} = \begin{bmatrix} R_1 & R_2 & \cdots & R_K \end{bmatrix}^T, \mathbf{r} \in \mathbb{Z}^{K \times 1}. \quad (4.8)$$

Sonuçta veri-hızı kısıtları  $\mathbf{A}_u \mathbf{x} = \mathbf{r}$  ile ifade edilebilmektedir.

Aynı alt-taşıyıcının birden fazla kullanıcı tarafından kullanılmaması kısıtı ise  $\mathbf{A}_c$  matrisi ile gösterilmektedir

$$\mathbf{A}_c = \begin{bmatrix} \mathbf{1}_c & \mathbf{0}_c & \cdots & \mathbf{0}_c \\ \mathbf{0}_c & \mathbf{1}_c & \cdots & \mathbf{0}_c \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_c & \mathbf{0}_c & \cdots & \mathbf{1}_c \end{bmatrix}, \mathbf{A}_c \in \{0, 1\}^{N \times KMN} \quad (4.9)$$

Burada

$$\mathbf{1}_c = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}, \mathbf{1}_c \in 1^{1 \times KM} \quad (4.10)$$

$$\mathbf{0}_c = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}, \mathbf{0}_c \in 0^{1 \times KM} \quad (4.11)$$

ve, bu kısıta karşılık gelen vektör ise

$$\mathbf{s} = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^T, \mathbf{s} \in 1^{N \times 1} \quad (4.12)$$

olarak tanımlandığında alt-taşıyıcı paylaşımı kısıtı  $\mathbf{A}_c \mathbf{x} \leq \mathbf{s}$  olarak yazılabilmektedir.

Örneğin, her iki kısıt grubunun matrisleri birleştirildiğinde kısıtlar matrisinin yapısı  $N = 2, K = 2, M = 2$  için aşağıdaki gibidir

$$\begin{bmatrix} 1 & 2 & 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 & 0 & 1 & 2 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}. \quad (4.13)$$

Sonuç olarak eniyileme problemi:

$$\begin{aligned} \min \quad & \mathbf{p}^T \mathbf{x} \\ \text{kısıtlar} \quad & \mathbf{A}_u \mathbf{x} \geq \mathbf{r} \\ & \mathbf{A}_c \mathbf{x} \leq \mathbf{s} \end{aligned} \quad (4.14)$$

haline gelmektedir. Görüldüğü gibi kısıt matrisleri sadece  $K, N$  ve  $M$  değerlerine bağlıdır. Kanal katsayılarının değişmesi durumunda maliyet işlevi vektörü  $\mathbf{p}$ 'nin aldığı değerler değişirken kısıt matrisleri sabit kalacaktır.

## 4.2 Tamsayı Programlama İle Kaynak Tahsisi

Tamsayı programlamada, (kapalı bir olurlu küme için) çözüm uzayındaki eleman sayısı sonludur. Çözüm olasılıklarından birisi olarak ayrık uzaydaki her eleman tek tek denenerek eniyi çözüme ulaşılabilir. Fakat OFDMA sistemlerdeki kaynak tahsisinde beklendiği gibi bu yaklaşım oldukça yavaş çalışmakta ve pratikte kullanılmaktan uzak olmaktadır. Eniyileme problemindeki  $\mathbf{x}$  bit-tahsisi vektörü  $KMN$  adet elemandan oluşmaktadır. Her bir eleman sadece 0 veya 1 değerini alabildiğinden bütün uzayı taramak için  $2^{KMN}$  adet olası çözümü incelemek gerekmektedir.

Kaynak tahsisi probleminde aynı alt-taşıyıcının birden fazla kullanıcı tarafından kullanılmaması kısıtı uzayı küçültmektedir. Çünkü  $N$ 'inci alt-taşıyıcıda bit-tahsisi vektörünün alabileceği farklı değer sayısı  $(KM + 1)$  tanedir (farklı değer sayısı  $KM$  yerine  $KM + 1$  olmaktadır. Çünkü  $n$ 'inci alt-taşıyıcıda hiç bir kullanıcıya bit atanmaması durumu da olası bir çözümün parçası olmaktadır). Dolayısıyla toplamda  $(KM + 1)^N$  adet olasılık vardır. Böylece uzaydaki olurlu değişkenleri bulabilmek için daha az noktaya bakmak yeterli olacakken halen kaba kuvvet yöntemiyle tüm noktaları taramak etkili bir yöntem olmaktan uzak olacaktır.

Tezde eniyileme problemi (4.14), Bölüm 3.1.2'de anlatılan Dallon ve Sınırla ile Dallon ve Kes Algoritmaları ile çözümlenerek eniyi çözüme ulaşılmıştır. Elde edilen sonuçlar Bölüm 7'de tartışılmaktadır.

### 4.2.1 Önerilen Algoritma

Yapılan benzetim çalışmalarında (4.14) maliyet işleminin enküçültülmesinde, tamsayı çözüm ile, tamsayı kısıtlarının gevşetildiği doğrusal çözüm arasında bir benzerlik olduğu görülmüştür. Buradan hareketle Simpleks Algoritması ile bulunan doğrusal çözümden tamsayı çözüme geçiş için çeşitli yöntemler önerilmiştir.

Simpleks Algoritması ile bulunan doğrusal çözümün tüm elemanların tamsayı olma olasılığı vardır. Bu durumda doğrudan eniyi tamsayı olurlu çözüm bulunmuş olur. Aksi halde sürekli değere sahip bir çözüm ile aynı alt-taşıyıcının birden fazla kullanıcı tarafından kullanılması durumu ortaya çıkmaktadır. Bu tezde önerilen algoritma, ilk adımda Simpleks Algoritması ile bulunan çözümde paylaşılan alt-taşıyıcıların listesini çıkartarak her alt-taşıyıcının sadece bir alt-kullanıcıya atanmasını sağlamaktadır. Bunun için iki farklı yöntem önerilmektedir.

İlkinde, hangi kullanıcı paylaşılan alt-taşıyıcıdan daha çok pay alıyorsa, o alt-taşıyıcıya o kullanıcıya alt-taşıyıcı atanmaktadır. Önerilen ikinci yöntemde çakışma olan alt-taşıyıcının hangi kullanıcıya atanacağını belirlemede kanal katsayıları etkili olmaktadır. Kullanıcı  $i$  ile kullanıcı  $j$  arasındaki  $n$ 'inci alt-taşıyıcıdaki çakışma durumunda aşağıdaki

$$\frac{\alpha_{i,n}^2}{\sum_{n=1}^N \alpha_{i,n}^2} - \frac{\alpha_{j,n}^2}{\sum_{n=1}^N \alpha_{j,n}^2} \quad (4.15)$$

ifadesi kullanılmaktadır. İfadenin değeri pozitif ise çakışma olan  $n$ 'inci alt-taşıyıcı  $i$ 'inci kullanıcıya atanmaktadır. Aksi halde  $j$ 'inci kullanıcı o alt-taşıyıcıyı kullanmayı haketmektedir. Yapılan benzetim çalışmalarında iki yöntemin çok yakın sonuçlar verdiği gözlenmiştir.

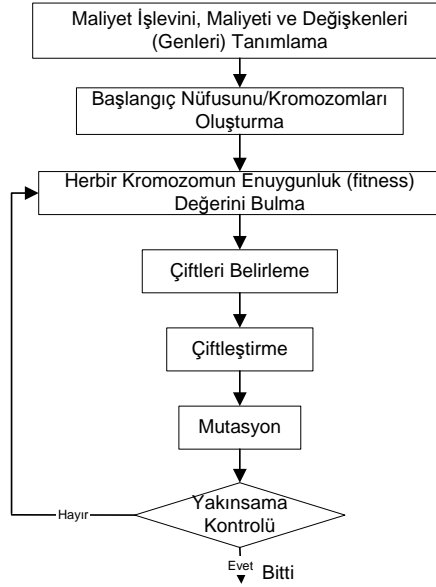
Böylece kullanıcılara alt-taşıyıcı tahsisi işlemi sonlanmıştır. Bu adımdan sonra her kullanıcı için Ağgözlü Algoritma kullanılarak bit tahsisi işlemi gerçekleştirilmektedir. Sonuçlar kısmında görüleceği gibi önerilen algoritma, eniyiye yakın çözümü, eniyi çözüme oranla çok az hata ile hızlı bir şekilde üretmektedir.

## 5. GENETİK ALGORİTMA İLE KAYNAK TAHSİSİ

OFDMA sistemlerde kaynak tahsisi probleminin arama uzayı boyutlarının yüksek olması nedeniyle, genetik algoritmaların kullanılması uygundur. Bu bölümde genetik algoritmalar tanıtıldıktan sonra Wang tarafından önerilen genetik algoritma tabanlı OFDMA kaynak tahsisi yöntemi gösterilecektir.

### 5.1 Genetik Algoritma

Genetik algoritma doğadaki evrim mekanizmasının modellenmesiyle ortaya çıkan sezgisel bir eniyileme yöntemidir. Holland [42] tarafından geliştirilen genetik algoritma bir ilk nüfusu oluşturup, o nüfustan maliyet fonksiyonunu küçülten bireyler oluşturarak her yinelemede eniyi çözüme yaklaşmayı amaçlamaktadır. Algoritmanın akış şeması Şekil 5.1'de gösterilmiştir.



Şekil 5.1: Genetik Algoritma akış şeması

Birinci adımda maliyet işlevi ve değişkenler tanımlanmaktadır. Genetik algoritmalarda olası her bir aday çözüm bir kromozom olarak adlandırılmaktadır. Eğer eniyileme problemindeki değişken sayısı  $N_{var}$  ise bir kromozom,  $N_{var}$  adet gen olarak adlandırılan elemandan oluşan bir satır vektörü olacaktır

$$kromozom = [p_1 \ p_2 \ \cdots \ p_{N_{var}}]. \quad (5.1)$$

Kromozomda  $p_k$  ile gösterilen skalar değişkenlerin değerleri 0 ve 1'lerden oluşacak şekilde seçilebileceği gibi alternatif olarak değişkenler tamsayı olarak da ifade edilebilirler. (Bu tez kapsamında kullanılmamakla birlikte gen değerleri kesirli veya gerçek sayı olarak da alınabilmektedir.)

İkinci adımda, belirlenen nüfus sayısında, yani  $N_{pop}$  adet kromozom rasgele oluşturularak  $N_{pop} \times N_{var}$  boyutlarında ilk nüfus matrisi üretilmiş olur. Nüfus matrisinin her bir satırı bir kromozoma yani bir aday çözüme denk gelmektedir. Her aday çözümün maliyet değerleri uygunluk işlevi kullanılarak teker teker hesaplanır (Uygunluk işlevi terimi genetik algoritma terminolojisinde sıklıkla eniyileme terminolojisindeki maliyet işlevi terimi yerine kullanılır). Uygunluk değeri o kromozomun ulaşılmak istenilen en iyi maliyet değerine ne kadar yaklaşıldığı bilgisini taşımaktadır.

Üçüncü adımda doğal seçilim işlemi gerçekleştirilmektedir. Doğal seçimde kromozomlar maliyet işlevlerinin değerine göre değerlendirilmekte ve belirlenen kriterlere göre seçilen kromozomların çiftleşmesi gerçekleştirilmektedir. Genetik algoritmalarda nüfustaki toplam kromozom sayısının sabit olması gerektiğinden dolayı kötü genlere yani yüksek maliyet işlevi değerine sahip kromozomlar nüfustan çıkarılarak üremeleri engellenmektedir. Nüfustan atılacak kromozomlar ile ilgili kriterler için literatürde önerilen çeşitli yöntemler aşağıdaki gibidir:

1. Değerlerine göre sıralanan ve en iyi ile en kötü kromozomun alındığı üstten ve alttan seçim, (bazı durumlarda en iyi kromozom bir yerel en küçüğe takılmış olabilir.)

2. Rasgele seçim,

3. Kromozomlar arasında en iyiye daha fazla seçilme olasılığı veren ağırlıklandırılmış seçim,

4. İki ayrı küme oluşturularak bu kümelerin içinden birer tane kromozom seçilmesini amaçlayan turnuva seçimi.

Dördüncü adımda ise kromozomları çiftleştirme işlemi yapılmaktadır. Çiftleştirme yaptırılacak ebeveynler yukarıdaki kriterlere göre seçilerek ve çaprazlanarak yeni çocuklar oluşturulmaktadır. Sıkça kullanılan çiftleştirme işlemlerinden olan tek noktalı çiftleştirme işleminde, iki adet kromozom belirlendikten sonra aralarında bir nokta seçilip o noktadan çaprazlama işlemi yapılmaktadır. İki noktalı çiftleştirme işleminde ise, seçilen iki adet kromozom iki

adet nokta belirlendikten sonra bu noktalar arasında çaprazlanmaktadır. Çiftleştirme işlemi sonucu oluşan nüfusun eski nüfusa göre daha düşük maliyet değerlerine sahip olması beklenmektedir.



Şekil 5.2: Tek noktalı çiftleştirme.



Şekil 5.3: İki noktalı çiftleştirme.

Beşinci adımda ise genetik algoritmanın yerel en küçük noktalardan birinde takılmasını engellemek amacıyla mutasyon işlemi gerçekleştirilmektedir. Mutasyon sırasında seçilen kromozomların bir ya da birden fazla elemanı rasgele olarak değiştirilmektedir. Mutasyon sırasında düşük maliyet işlevi değerine sahip kromozomların genleri değiştirilerek öncekinden daha yüksek maliyet işlevi değerine sahip yeni bireyler elde edilebilir. Bu istenmeyen durumu engellemek için elitizm işlemi kullanılmaktadır. Elitizm ile düşük maliyet işlevi değerine sahip kromozomlar hiç bir değişikliğe uğramadan doğrudan yeni nüfusun içinde yerlerini almaktadırlar.

Son adımda, eğer istenilen sonlandırma kriterleri sağlanmadıysa algoritma tekrar ikinci adımdan başlayarak çalışmaya devam eder. Aksi halde algoritma sonlandırılır. Algoritmayı sonlandırma kriteri olarak belirli bir iterasyon sayısı belirlenebilir. Alternatif olarak oluşan yeni nüfuslarda daha düşük maliyet işlevi değerleri bulunamadığı zaman algoritma sonlandırılabilir.

## 5.2 Genetik Algoritmanın OFDMA Sistemlere Uygulanması

OFDMA sistemlerde kaynak tahsisi probleminin karmaşıklığı, arama uzayının geniş olması nedeniyle yüksek olduğu için yapısı genetik algoritmaların kullanılması için uygundur. Wang tarafından önerilen genetik algoritma temelli yöntemde kromozomların her bir geni bir alt-taşıyıcıya denk gelmektedir. Dolayısıyla kromozomlar  $1 \times N$  boyutlarındadır. Kromozomların her bir geni 1 ile  $K$  arasındaki tamsayı değerleri alabilmektedir. Kullanıcı sayısı 3 ve alt-taşıyıcı sayısı 6 olan bir OFDMA sistem için örnek bir kromozom aşağıdaki gibidir:

$$kromozom = [3 \ 1 \ 2 \ 2 \ 1 \ 3]$$

Kromozomun ilk geninin üç olması birinci alt-taşıyıcının üçüncü kullanıcıya atandığı anlamına gelmektedir. Benzer şekilde ikinci alt-taşıyıcı da birinci kullanıcıya atanmıştır. İlk nüfus rasgele oluşturulan kromozomların bir matriste toplanmasıyla elde edilir. Kromozomlar, sadece alt-taşıyıcıların kullanıcılara tahsisi bilgisini içermektedir. Bit tahsisi işleminin yapılması için Wang aç gözlü algoritmayı kullanmıştır. Böylece her kromozom için ayrı ayrı aç gözlü algoritma kullanılarak kromozomların maliyet işlevi değerleri hesaplanabilir. Uygunluk işlevi olarak önceki metodlarda önerildiği gibi alt-taşıyıcılara atanan toplam güç seçilmiştir.

Maliyet işlevi değerleri hesaplandıktan sonra en düşük maliyet işlevi değerlerine sahip  $P$  adet kromozomun elitizm işlemi ile mutasyona uğramaması ve nüfustan çıkarılmamaları ve mutasyona uğramamaları sağlanır.

Çiftleştirme işlemi çift noktalı çiftleştirme yöntemiyle gerçekleştirilmiştir.

Mutasyon işlemi ise uyarlamalı olarak değişmektedir. Eğer peş peşe yaratılan nüfuslarda daha iyi bir çözüm bulunamıyorsa daha çok kromozom mutasyon işlemine tabi tutulmaktadır. Belirli bir iterasyon sayısına ulaştıktan sonra ise algoritma sonlandırılmaktadır.

Genetik algoritma ile OFDMA kaynak tahsisindeki ilgili parametreler ve değerleri benzetim sonuçları bölümünde açıklanacaktır.

## 6. ÇOK KULLANICILI MIMO OFDM SİSTEMLERDE KAYNAK TAHSİSİ

Çok kullanıcıli OFDM sistemlerde çoklu anten kullanımı (MIMO, Multi-Input Multi-Output), SISO sistemlerin aksine aynı frekans alt-kanalının birden fazla kullanıcı tarafından kullanılmasına izin vermektedir. Böylece alıcı ve vericide tek anten bulunan (SISO) sistemlere göre daha düşük güçle veya daha yüksek veri hızında iletişim yapabilme imkanı ortaya çıkmaktadır. Fakat aynı frekans alt-taşıyıcısının ortak kullanımı beraberinde ortak kanal girişimini meydana getirmektedir. Literatürde [43] kullanıcıları gruplayarak düşük korelasyonlu kullanıcıların aynı alt-taşıyıcıyı kullanmasını önermiştir. Aynı taşıyıcı ortaklaşa kullanıldığında Sinyalin Girişime ve Gürültüye Oranı (Signal-to-Interference and Noise-Ratio, SINR) azalacak bu da QoS'de problemlere yol açacaktır. Burada önerilen algoritma kaynak tahsisi sırasında QoS'i göz önüne alırken aynı zamanda SINR'ı da kullanarak CCI problemini de ele almaktadır.

### 6.1 Sistem Modeli

Bu çalışmada, her birinin  $M_T^k$  adet verici antene sahip olduğu  $K$  kullanıcıdan ve  $M_R$  adet alıcı anteni olan bir baz istasyondan oluşan bir MIMO (Multi-Input Multi-Output) senaryo incelenmektedir. OFDM iletişim tekniğinin kullanıldığı ve kanal bantgenişliği  $N$  alt-taşıyıcıya bölüldüğü varsayılmaktadır. Kanal durum bilgisinin (CSI, Channel State Information) vericide tam olarak bilindiği kabul edilmekte ve  $k$ -ıncı kullanıcının  $n$ -inci alt-taşıyıcısında gözlenen  $M_R \times M_T^k$  MIMO kanal matrisi  $\mathbf{H}_n^k$  ile gösterilmektedir.

İletişim kanalı, Tekil Değer Ayrışımı (SVD, Singular Value Decomposition) kullanılarak  $\mathbf{H}_n^k = \sum_{i=1}^{\min(M_T^k, M_R)} \mathbf{u}_i^k(n) s_i^k(n) (\mathbf{v}_i^k(n))^H$  biçiminde parçalanabilir. Burada,  $i = 1, 2, \dots, \min(M_T^k, M_R)$  için  $\{\mathbf{u}_i^k(n)\}$  ve  $\{\mathbf{v}_i^k(n)\}$  sırasıyla  $M_R \times 1$  ve  $M_T^k \times 1$  birimcil vektörlerken,  $\{s_i^k(n)\}$   $\mathbf{H}_n^k$ 'nin tekil değerleridir. Dış ortamlarda en büyük  $s_i$  'nin kanalın neredeyse tüm enerjisini içinde topladığı bulunmuştur [44]. Buradan hareketle, önerilecek teknikte  $k$ 'inci kullanıcının  $n$ 'inci alt-taşıyıcısındaki MIMO kanalın en yüksek tekil değerine karşılık gelen özkipini (eigenmode) açığa çıkaran bir hüzmeye oluşumu (beamforming) varsayılmıştır. Dolayısıyla bundan sonra  $\{\mathbf{u}_i^k(n)\}$ ,  $\{\mathbf{v}_i^k(n)\}$  ve  $\{s_i^k(n)\}$ 'deki  $i$  indisi düşürülecek ve enyüksek tekil değer ve karşılık gelen özvektörler ele alınacaktır.

Her kullanıcının sinyali,  $M_T^k$  antenden kanala göndermeden önce en yüksek tekil değere karşılık gelen  $\mathbf{v}_k$  vektörüyle çarpılarak gönderilen sinyalin gücü tek



bir boyutta toplanmakta ve Sinyal-Gürültü-Oranı (SGO) en yüksek seviyeye çıkartılmaktadır. Böylece baz istasyon tarafından  $n$ 'inci alt-taşıyıcıda alınan toplam sinyal

$$\begin{aligned}\mathbf{r}(n) &= \sum_{k=1}^K \mathbf{u}_k(n) s_k(n) \sqrt{p_k(n)} x_k(n) + \mathbf{n}(n) \\ &= \mathbf{U}(n) \mathbf{S}(n) \mathbf{P}(n)^{1/2} \mathbf{x}(n) + \mathbf{n}(n)\end{aligned}\quad (6.1)$$

olarak yazılabilir. Yukarıdaki denklemde,  $n$ 'inci alttaşıyıcı için,  $M_R$  alıcı anteni tarafından alınan sinyal  $M_R \times 1$  boyutlarında bir vektör olan  $\mathbf{r}(n)$  ile,  $k$ 'inci kullanıcının gönderdiği veri sembolü,  $x_k(n)$  ile, bu kanalın maksimum tekil değeri  $s_k(n)$  ve bu tekil değere karşılık gelen birimcil vektör  $\mathbf{u}_k(n)$  ile temsil edilmektedir. Kullanıcı  $k$ 'nin  $n$ 'inci alttaşıyıcıdan gönderdiği toplam güç ise  $p_k(n)$  ile gösterilmekte, ayrıca bağımsız ve özdeş dağılımlı (independent and identically distributed, i.i.d.) sıfır ortalama ve birim varyansa sahip karmaşık değerli Gauss gürültü  $\mathbf{n}(n)$  ile temsil edilmektedir. Denklem (6.1)'in ikinci satırındaki  $\mathbf{U}(n) = [\mathbf{u}_1(n) \mathbf{u}_2(n) \cdots \mathbf{u}_K(n)]$ ,  $\mathbf{S}(n) = \text{diag}[s_1(n), s_2(n), \cdots, s_K(n)]$ ,  $\mathbf{P}(n)^{1/2} = \text{diag}[\sqrt{p_1(n)}, \sqrt{p_2(n)}, \cdots, \sqrt{p_K(n)}]$  ve  $\mathbf{x}(n) = [x_1(n) x_2(n) \cdots x_K(n)]^T$  tanımları kullanılarak, alıcıdaki uyumlu süzgeç çıkışı

$$\begin{aligned}\mathbf{y}(n) &= \mathbf{U}^H(n) \mathbf{r}(n) \\ &= \mathbf{U}^H(n) \mathbf{U}(n) \mathbf{S}(n) \mathbf{P}(n)^{1/2} \mathbf{x}(n) + \mathbf{U}^H(n) \mathbf{n}(n)\end{aligned}\quad (6.2)$$

olarak yazılabilir. Burada dikkat edilmesi gereken nokta,  $\mathbf{U}(n)$  matrisinin sütunlarının her kullanıcının MIMO kanalının en yüksek tekil değerlerine karşılık gelen  $\{\mathbf{u}_k(n)\}$  birimcil vektörlerinden oluşmasıdır. Dolayısıyla  $\mathbf{U}^H(n) \mathbf{U}(n) = \begin{bmatrix} \mathbf{u}_1^H(n) & \mathbf{u}_2^H(n) & \cdots & \mathbf{u}_K^H(n) \end{bmatrix} \begin{bmatrix} \mathbf{u}_1(n) & \mathbf{u}_2(n) & \cdots & \mathbf{u}_K(n) \end{bmatrix}$  eşitliğinde  $\mathbf{u}_k^H(n) \mathbf{u}_k(n) = 1$ , iken  $\mathbf{u}_k^H(n) \mathbf{u}_l(n)$   $k$  kullanıcısı ile  $l$  kullanıcısı arasındaki ilintiyi göstermektedir. İlinti matrisi  $\mathbf{R}(n) = \mathbf{U}^H(n) \mathbf{U}(n)$  ve  $\rho_{k,l}(n) = \mathbf{u}_k^H(n) \mathbf{u}_l(n)$  olarak tanımlanırsa

$$\mathbf{R}(n) = \begin{bmatrix} 1 & \rho_{1,2}(n) & \cdots & \rho_{K,K}(n) \\ \rho_{2,1}(n) & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \rho_{K,K}(n) & \cdots & \cdots & 1 \end{bmatrix}\quad (6.3)$$

olarak yazılabilecek ve uyumlu süzgeç çıkışı

$$\mathbf{y}(n) = \mathbf{R}(n) \mathbf{S}(n) \mathbf{P}(n)^{1/2} \mathbf{x}(n) + \mathbf{U}^H(n) \mathbf{n}(n)\quad (6.4)$$

olarak yazılabilecektir. Aslında problem bu haliyle çok-kullanıcıli sezim (MUD, Multi-User Detection) formasyonuna sahiptir ve bu çalışmada da varsayıldığı gibi kaynak tahsisi yapıldıktan sonra, standart MUD teknikleriyle [5] çözülebildiği varsayılmaktadır.

## 6.2 Problem Tanımı

Denklem (6.4)'deki tanımdan  $k$ 'inci kullanıcının  $n$ 'inci alt-taşıyıcıdaki SINR'ı

$$SINR_k(n) = \frac{P_k(n) |s_k(n)|^2}{N_k(n) + \sum_{l=1, l \neq k}^M P_l(n) |s_l(n)|^2 \rho_{k,l}(n)^2} \quad (6.5)$$

şeklinde tanımlanabilir. Burada,  $P_k(n)$ ,  $k$ 'inci kullanıcının  $n$ 'inci alt-kanaldan gönderdiği güçtür ve önerilecek eniyileme tekniğinin değişkenlerinden birisidir. Benzer şekilde problemin bir diğer değişkeni ise  $k$ 'inci kullanıcının  $n$ 'inci alt-taşıyıcıdan gönderdiği bit sayısı  $c_k(n)$ 'dir.

Kaynak tahsisi probleminin maliyet işlevi gönderilen toplam gücün enazaltılması olarak seçilmiştir

$$\min: \sum_{n=1}^N \sum_{k=1}^K P_k(n). \quad (6.6)$$

Kullanıcı  $k$ 'nın ihtiyaç duyduğu veri hızı  $R_k$  bit/çerçeve (OFDM frame) olarak alınır ise problemin kısıtı ise

$$\sum_{n=1}^N c_k(n) = R_k, \quad k = 1, 2, \dots, K \quad (6.7)$$

olacaktır.

$M$ -QAM kipleniminde ortalama sembol hata oranı (SER,  $\bar{P}_e$ ) aşağıdaki gibi bir üst sınırla sınırlandırılabilir

$$\begin{aligned} \bar{P}_e &\leq 4Q \left( \sqrt{\frac{3E_s}{(M-1)I_0}} \right) = 4Q \left( \sqrt{\frac{3|s|^2 P_s}{(M-1)I}} \right) \\ &= 4Q \left( \sqrt{\frac{3 \times SINR}{(M-1)}} \right). \end{aligned} \quad (6.8)$$

Buradan yola çıkarak  $k$ 'inci kullanıcı  $n$ 'inci alt-taşıyıcıda  $M_k(n)$ -QAM kiplenimini kullanıyorsa  $SINR_k(n)$  tekrar yazılabilir

$$\begin{aligned} SINR_k(n) &= \frac{1}{3} \left[ Q^{-1} \left( \frac{\bar{P}_e}{4} \right) \right]^2 (M_k(n) - 1) \\ &= f(\bar{P}_e)(2^{c_k(n)} - 1). \end{aligned} \quad (6.9)$$

Denklemler (6.5) ile (6.9) birleştirilirse

$$P_k(n) = f(\bar{P}_e) \cdot (2^{c_k(n)}) \cdot \left\{ N_k(n) + \sum_{l=1, l \neq k}^M P_l(n) |s_l(n)|^2 \rho_{k,l}(n)^2 \right\} \frac{1}{|s_k(n)|^2} \quad (6.10)$$

bulunur. Denklem (6.10)'da  $P_l(n)$  'yi de açtıktan sonra.  $|s_k(n)|^2 \ll |s_l(n)|^2 \rho_{k,l}(n)^2$  varsayımı kabul edilirse

$$P_k(n) \approx f(\bar{P}_e) \times (2^{c_k(n)} - 1) \frac{N_k(n)}{|s_k(n)|^2} + f(\bar{P}_e) \times (2^{c_k(n)} - 1) \frac{1}{|s_k(n)|^2} \cdot \sum_{l=1, l \neq k}^K \left( \frac{|s_l(n)|^2}{|s_k(n)|^2} \right) \rho_{k,l}(n)^2 \times f(\bar{P}_e) (2^{c_l(n)} - 1) N_l(n) \quad (6.11)$$

elde edilir. Denklem (6.11)'deki değişkenler  $c_k(n)$  ve  $c_l(n)$  'dir ve  $P_k(n)$  bu değişkenler cinsinden yazılabildiği için eniyilemenin bu değişkenler üzerinden yapılması yeterli olacaktır.

### 6.3 Tamsayı Programlama

Denklem (6.11)'de tanımlanan  $P_k(n)$  terimi doğrusal ve doğrusal olmayan bileşenlerden oluşmaktadır. Maliyet işlevini oluşturan  $P_k(n)$  'nin doğrusal bileşeni birinci satırdaki ifade iken,  $c_k(n)$  ve  $c_l(n)$ 'ler arası çarpım nedeniyle geri kalan kısmı doğrusal olmayan bileşeni oluşturmaktadır. Maliyet işlevi doğrusal olan tamsayı programlama problemleri için verimli teknikler bulunmaktadır. Dolayısıyla problemin doğrusallaştırılmasında fayda vardır. Problemin yeni değişkenler katılarak doğrusallaştırılması aşağıda tartışılmaktadır.

#### 6.3.1 Doğrusallaştırma

**Önerme [45]:** n-adet değişkenin  $\{x_j\}_{j=1}^n$  çarpımı

$$y = \prod_{j \in S} x_j \quad (6.12)$$

olsun. Burada  $S : \{1, \dots, n\}$  ve  $x_j \in \{0, 1\}$ ,  $j \in S$ 'dir. Denklem (6.12)'e

$$\sum_{j \in S} x_j - y \leq s - 1 \quad (6.13)$$

$$-\sum_{j \in S} x_j + sy \leq 0, \quad (6.14)$$

eşitsizlikleriyle de eşdeğer olarak ifade edilebilir. Burada  $s$ ,  $S$  kümesinin eleman sayısıdır. Denklem (6.12)'de doğrusal olmayan terimler 2 boyutlu olduğu için  $s$  değeri 2 'dir ve her doğrusal olmayan terimi yeni bir değişken  $y$  değişkeni ile (6.13), (6.14) eşitsizlikleriyle doğrusal hale getirilebilir.  $NKM \times 1$  boyutlarındaki  $\mathbf{x}$  vektöründe toplam, (6.11)'deki her alt-taşıyıcı için  $(KM(KM - M)) / 2$  adet doğrusal olmayan terim vardır.  $(N(KM(KM - M)) / 2) \times 1$  boyutlarındaki  $\mathbf{y}$  vektörü  $k < l$  ;  $k, l = \{1, \dots, K\}$  ve  $m_k, m_l \in \{1, \dots, M\}$  olacak şekilde  $\mathbf{y}_i(n) = x_{k,n,m_k} \times x_{l,n,m_l}$  ikili çarpımlarından oluşmaktadır.  $n$ 'inci alt-taşıyıcıdaki  $y_i(n)$  terimine karşılık gelen  $k$  ve  $l$  kullanıcıları için iki kullanıcı güçleri arasındaki girişimden kaynaklanan güç  $M^{(K-2)} \rho_{k,l}(n)^2 \left[ \frac{N_k(n)}{|s_l(n)|^2} + \frac{N_l(n)}{|s_k(n)|^2} \right]$  olacak şekilde  $(N(KM \times (KM - M)) / 2) \times 1$  boyutlarındaki  $\mathbf{p}_{\text{çapraz}}$  güç vektörü oluşturulabilir. Denklem (6.11)'nin doğrusal kısmını oluşturan ilk satırındaki terimler de

$$p_{k,n,m} = f(\bar{P}_e) \times (2^m - 1) \frac{N_k(n)}{|s_k(n)|^2} \quad (6.15)$$

tanımı kullanılarak

$$\mathbf{p}_{k,n} = \left[ p_{k,n,1} \quad p_{k,n,2} \quad \dots \quad p_{k,n,M} \right]^T \quad (6.16)$$

$$\mathbf{p}_N^n = \left[ \left[ \mathbf{p}_{1,n}^T \quad \mathbf{p}_{2,n}^T \quad \dots \quad \mathbf{p}_{K,n}^T \right]^T \right] \quad (6.17)$$

$$\mathbf{p}_{\text{doğrusal}} = \left[ (\mathbf{p}_N^1)^T \quad (\mathbf{p}_N^2)^T \quad \dots \quad (\mathbf{p}_N^N)^T \right]^T \quad (6.18)$$

$NKM \times 1$  boyutlarındaki vektörle ifade edilebilir. Doğrusal ve doğrusal olmayan güç ifadeleri  $\mathbf{p}_{\text{doğrusal}}$  ve  $\mathbf{p}_{\text{çapraz}}$  birleştirilerek  $(NKM + (N(KM \times (KM - M)) / 2)) \times 1$  boyutlarındaki

$$\tilde{\mathbf{p}} = \left[ \mathbf{p}_{\text{doğrusal}}^T \quad \mathbf{p}_{\text{çapraz}}^T \right]^T \quad (6.19)$$

vektörü oluşturulur. Burada  $\tilde{\mathbf{p}}$  vektöründeki değerler sabittir.

### 6.3.2 Tamsayı Programlama Algoritması

Bir kullanıcının hangi alt-taşıyıcıya ne kadar bit atayacağını ve hangi alt-taşıyıcının hangi kullanıcıya atanacağı bilgisi problemin değişkenidir. Bu değişkenler aşağıdaki gibi bir vektör formatında yazılabilir. Belirli bir  $k$  kullanıcısı için yine belirli bir  $n$  alt-taşıyıcıda gönderilecek bit sayısı

$$\mathbf{x}_{k,n} = \left[ x_{k,n,1} \quad x_{k,n,2} \quad \dots \quad x_{k,n,M} \right]^T \quad (6.20)$$

vektörü tarafından verilmektedir. Burada, bu kanaldan eğer  $m$ -bit gönderilecekse  $\mathbf{x}_{k,n}$  vektörünün  $2^m$ 'inci terimi bir, geri kalanı sıfır değerini alır. Sonraki adımda  $\mathbf{x}_{k,n}$  vektörleri kullanıcı indisi  $k$ 'ya göre bir araya getirilerek belirli bir  $n$  alt-taşıyıcısı için  $\mathbf{x}_N^n = \left[ \mathbf{x}_{1,n}^T \quad \mathbf{x}_{2,n}^T \quad \cdots \quad \mathbf{x}_{K,n}^T \right]^T$  vektörü oluşturulur. Takip eden adımda tüm alt-taşıyıcıların  $\mathbf{x}_N^n$  vektörleri birleştirilerek

$$\mathbf{x} = \left[ (\mathbf{x}_N^1)^T \quad (\mathbf{x}_N^2)^T \quad \cdots \quad (\mathbf{x}_N^N)^T \right]^T \quad (6.21)$$

bit tahsis vektörü [36] 'da olduğu gibi oluşturulur. Son olarak eniyileme algoritmasının değişkenlerini içeren vektör  $\mathbf{z} = [\mathbf{x}^T \mathbf{y}^T]^T$  olarak tanımlanır.

Böylece (6.11) ifadesi şu şekilde doğrusal hale getirilmiş olur

$$\text{Maliyet Fonksiyonu: } \mathbf{P}^T \mathbf{z} \quad (6.22)$$

$$\text{Kısıtlamalar: } \mathbf{A} \mathbf{z} \leq \mathbf{c}. \quad (6.23)$$

Kısıtlamalar matrisi  $\mathbf{A} = \begin{bmatrix} -\mathbf{A}_{c1} & \mathbf{A}_{c2} & \mathbf{A}_{c3} \end{bmatrix}$  şeklindedir. Buradaki alt-matrislerin yapıları ise şöyledir: Veri hızı kısıtlaması için kullanılan  $\mathbf{A}_{c1}$  matrisi.

$$\mathbf{A}_{c1} = \begin{bmatrix} \mathbf{a}_{K1} & \cdots & \mathbf{a}_{K1} & \mathbf{0}_{1xc} \\ \mathbf{a}_{K2} & \cdots & \mathbf{a}_{K2} & \mathbf{0}_{1xc} \\ \vdots & \ddots & \vdots & \vdots \\ \mathbf{a}_{KK} & \cdots & \mathbf{a}_{KK} & \mathbf{0}_{1xc} \end{bmatrix}$$

Burada  $\mathbf{a}_{Kk} = \left[ \mathbf{0}_{1 \times (k-1)M} \quad \mathbf{a}_k \quad \mathbf{0}_{1 \times (K-k)M} \right] \in \mathbb{Z}^{1 \times KM}$ ,  $\mathbf{a}_k = \left[ 1 \quad 2 \quad \cdots \quad M \right] \in \mathbb{Z}^{1 \times M}$ ,  $k = 1, 2, \dots, K$ ,  $c = (NKM \times (KM - M)) / 2$  olarak tanımlanmıştır, ve  $\mathbf{0}_{1 \times L}$   $1 \times L$  boyutlarında tüm sıfırlardan oluşan bir vektördür.

Kullanıcı  $k$  'nın  $n$ 'inci alt-taşıyıcısında kiplenimi tek yerden yapılmalıdır. Örnek vermek gerekirse  $k$  kullanıcısı  $n$  'inci alt-taşıyıcısında hem 1 hem de 2 bit göndermek yerine ya 1 bit ya da 2 bit göndermelidir. Bu kısıt için  $\mathbf{A}_{c2}$  matrisi

$$\mathbf{A}_{c2} = \begin{bmatrix} \mathbf{1}_{1 \times M} & \mathbf{0}_{1 \times M} & \cdots & \mathbf{0}_{1 \times M} & \mathbf{0}_{1xc} \\ \mathbf{0}_{1 \times M} & \mathbf{1}_{1 \times M} & \cdots & \mathbf{0}_{1 \times M} & \mathbf{0}_{1xc} \\ \vdots & & \ddots & \vdots & \vdots \\ \mathbf{0}_{1 \times M} & \mathbf{0}_{1 \times M} & \cdots & \mathbf{1}_{1 \times M} & \mathbf{0}_{1xc} \end{bmatrix}$$

olarak tanımlanmıştır. Burada  $\mathbf{1}_{1 \times M}$ ,  $1 \times M$  boyutlarında tümü birlerden oluşan bir vektördür.

Doğrusal olmayan terimleri doğrusallaştırmada kullanılan (6.13) ve (6.14) kısıtlamaları düzensiz bir yapıya sahip seyrek  $\mathbf{A}_{c3}$  matrisiyle ifade edilebilir. Yukarıdaki yapılan tanımlar ile (6.22)-(6.23)'deki doğrusal forma dönüşen problem, tamsayı programlama tekniklerinden olan Dallar ve Sınırla Yöntemi ile çözülebilmektedir. Böylece çok kullanıcı MIMO OFDM sistemlerde kaynak tahsisi problemi doğrusal hale getirilmiştir. Fakat  $|s_k(n)|^2 \ll |s_l(n)|^2 \rho_{k,l}(n)^2$  varsayımının alt-taşıyıcı ve kullanıcı sayısı arttıkça geçerliliğini kaybettiği gözlenmiştir. Dolayısıyla önerilen modelleme pratikte efektif bir metod olmayacaktır.

## 7. BENZETİM SONUÇLARI

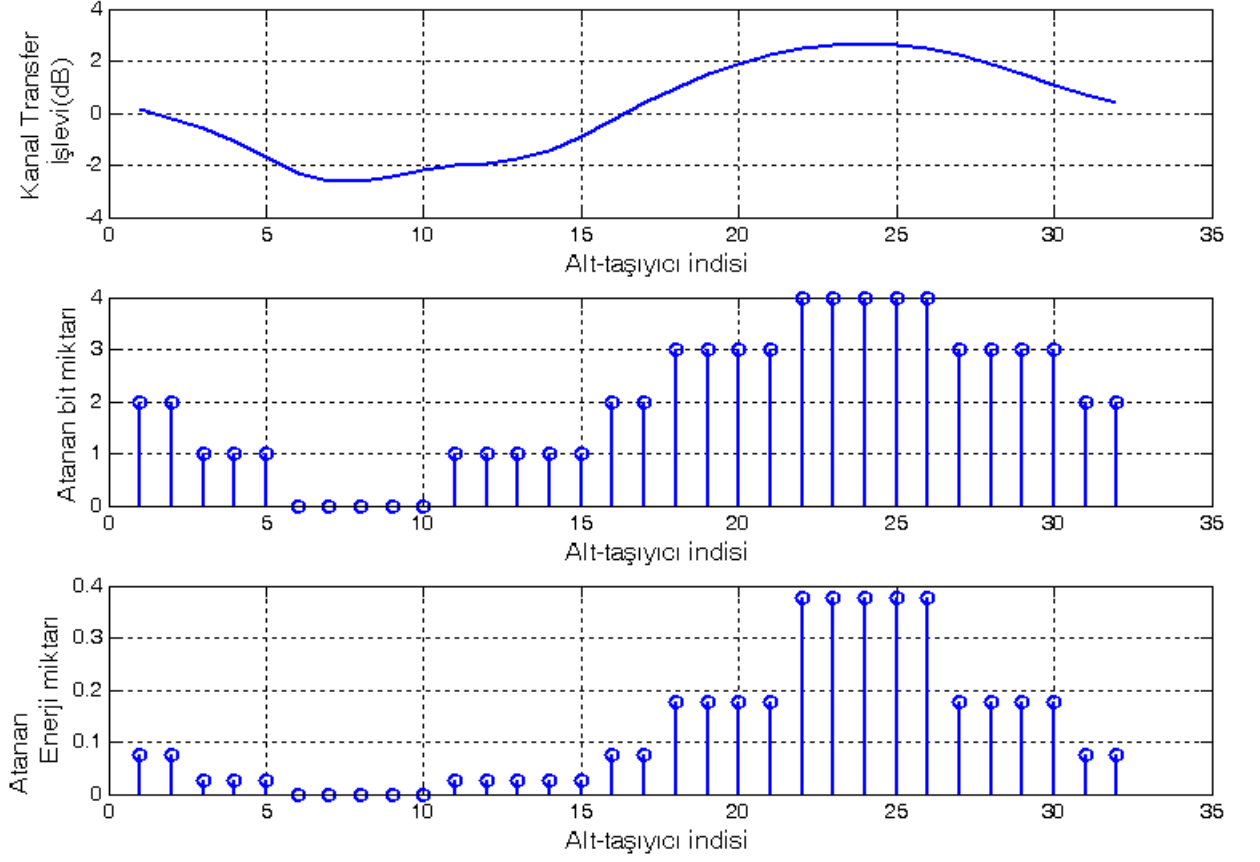
Bu bölümde tezin değişik yerlerinde bahsi geçen kaynak tahsisi yöntemlerinin benzetim sonuçları ve birbirleriyle kıyaslamaları verilecektir. Benzetimlerde incelenecek parametreler; kullanıcı sayısı  $K$ , alt-taşıyıcı sayısı  $N$ , bir alt-taşıyıcıya atanabilecek en yüksek bit miktarı  $M$ , bit hata olasılığı BER, ve  $k$  kullanıcısının ihtiyaç duyduğu bit sayısı  $R_k$ 'dir. Benzetim çalışmaları OFDMA ve çok kullanıcı MIMO OFDM olmak üzere iki ana gruba ayrılmıştır.

### 7.1 OFDMA Sistemlerde Kaynak Tahsisi Benzetimleri

OFDMA sistemlerdeki kaynak tahsisi algoritmaları benzetimlerinde kullanılan çokyollu (multipath) kanalın beş adet sıfır ortalamaya sahip Gauss rasgele değişkenden oluştuğu varsayılmıştır. Bu beş örneğin toplam değışintisi bire normalize edilmektedir. Böylece kanal, alınan sinyalin Sinyal-Gürültü-Oranı'ını etkilemeyecektir. Kanal katsayıları  $\alpha_{k,n}$ , bu katsayıların  $N$  adet alt-taşıyıcı üzerinden hızlı Fourier dönüşümleri alınmasıyla oluşturulmuştur.

#### 7.1.1 Tek Kullanıcı OFDM Sistemlerde Kaynak Tahsisi Benzetimleri

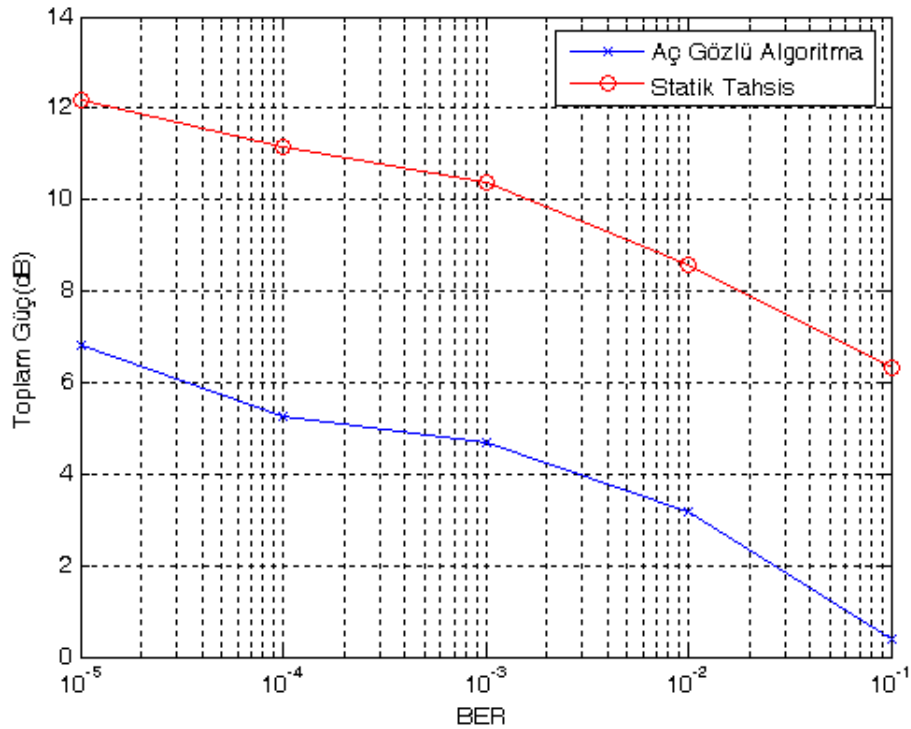
Kullanılan modelde tek kullanıcı olduğunda, kullanıcının istediği alt-taşıyıcıyı serbestçe kullanma hakkı bulunmaktadır. Örnek sistemde  $N = 32$  adet alt-taşıyıcı bulunmaktadır. Kullanıcının ihtiyaç duyduğu bit miktarı ise  $R = 64$  olarak seçilmiştir. Toplam gücü enküçültecek MA kaynak tahsisi Ağgözlü Algoritma ile gerçekleştirilmiştir.



Şekil 7.1: Açgözlü Algoritma ile Kaynak Tahsisi

Şekil 7.1’de rasgele bir kanalın aktarım işlevinin genliği görülmektedir. Şekilden Açgözlü Algoritma’nın, bitleri yüksek kanal katsayılı alt-taşıyıcılara atama eğiliminde olduğu görülmektedir. Yine ilginç bir sonuç da enerjinin büyük kısmının iyi alt-taşıyıcılara atanırken, kötü alt-taşıyıcılara enerji atanmaması, dolayısıyla burardan iletişim yapılmamasıdır.





Şekil 7.2: Tek kullanıcı OFDM senaryosunda Açık Gözlü Algoritma ve Statik Metod ile Kaynak Tahsisi ( $N = 32$ ,  $R = 64$ )

Şekil 7.2'deki benzetimde ise Şekil 7.1 için kullanılan senaryoda ( $N = 32$ ,  $R = 64$ ) Açık Gözlü Algoritma ve statik tahsis karşılaştırılmaktadır. Statik kaynak tahsisinde kanal durumuna bakılmaksızın her alt-taşıyıcıya eşit enerji ve 2 bit atanmıştır. Sonuçlar 200 farklı kanalın ortalaması alınarak elde edilmiştir. Şekilden de görüldüğü gibi tek kullanıcı OFDM sistemlerde Açık Gözlü Algoritma statik kaynak tahsisine göre çok daha başarılı sonuçlar vermektedir.

Ancak, tek kullanıcı sistemler için eniyi sonucu veren Açık Gözlü Algoritma, OFDMA sistemlerde eniyi çözümü vermeyecektir. Çünkü (bir alt-taşıyıcı birden fazla kullanıcı için yüksek kanal katsayısına sahip olabilir) birden fazla kullanıcı OFDMA sistemlerde tüm kullanıcılar için Açık Gözlü Algoritma kullanıldığında o alt-taşıyıcı birden fazla kullanıcıya atanabilir. Fakat OFDMA sistemlerde alt-taşıyıcıların paylaşılmasına izin verilmediğinden, bir çıkar çatışması yaşanacak ve bazı kullanıcıların talep ettikleri alt-taşıyıcılara bit tahsisi yapılmayacaktır. Bu nedenle, bu durumu gözönüne alarak eniyi çözümü veren gelişmiş algoritmalara ihtiyaç duyulmaktadır.

### 7.1.2 Çok kullanıcıli OFDMA Sistemlerde Kaynak Tahsisi Benzetimleri

OFDMA sistemlerdeki kaynak tahsisi benzetimleri çalışmalarında sunulacak sonuçların ilkinde tamsayı kısıtları gevşetilerek dışbükey hale getirilen problemin çözümü ile elde edilen başarımların grafikleri verilecektir. Sonra tamsayı kısıtlı problemin tamsayı programlama teknikleri ile çözümünü içeren benzetimler sunulacaktır. Bunu takiben eniyiye yakın sonuçlar veren algoritmaların benzetim çalışmaları sunulacak, başarımları kıyaslanarak algoritmaların birbirlerine karşı avantajları ve dezavantajları tartışılacaktır.

Tez kapsamında yapılan çalışmalarda çok kullanıcıli OFDMA sistemlerde kaynak tahsisi problemi ile ilgili tamsayı kısıtların gevşetilmesiyle oluşturulan dışbükey kaynak tahsisi problemi (2.9) Lagrange çarpanları kullanılarak [15] çözülmüştür. Bu çözüm tamsayı kısıtları sağlamasa da tamsayı kısıtlı problemle (2.4) kıyaslama açısından anlamlıdır. Dışbükey problem, Sıralı Karesel Programlama (SQP, Sequential Quadratic Programming) ile de çözülmüştür. Sıralı Karesel Programlama metodu düşük kullanıcı ve alt-taşıyıcı sayıları için aynı problemi Lagrange çarpanları yöntemine göre çok daha kısa sürede çözmekte iken kullanıcı sayıları ve alt-taşıyıcı sayıları arttıkça performansı nümerik sorunlar nedeniyle düşmektedir.

Tezde, daha sonra tamsayı kısıtlı kaynak tahsisi problemi Dallon ve Sınırla yöntemi ile çözülmüştür [36]. Dallon ve Sınırla yöntemine alternatif olarak Dallon ve Kes yöntemi de problemin çözümünde kullanılmıştır. Her iki yöntem de eniyi tamsayı sonucu vermektedir.

Problemin eniyi çözümü tamsayı programlama yöntemleri ile bulunduğundan sonra tez kapsamında çeşitli eniyiye yakın sonuçlar veren algoritmalar da incelenmiştir. Wong, MAO [15] (Multiuser Adaptive OFDM, Çok kullanıcıli Uyarlanabilir OFDM) isimli bir algoritma geliştirmiştir. Algoritmada ilk adımda Lagrange çarpanları yinelemeli olarak ayarlanarak bir çözüme ulaşılmaya çalışılmaktadır. Daha sonra ulaşılan alt-taşıyıcı tahsisi bilgisi kullanılarak bit tahsisi Açgözlü algoritma ile yapılmaktadır. Ayrıca ilk adımda kaynak tahsisini Simpleks Algoritması ile yapıp daha sonra bit tahsisini Açgözlü Algoritma ile yapan bir algoritma önerilmiştir. Son olarak OFDMA sistemlerde kaynak tahsisi problemi Genetik Algoritma ile çözülmüştür [31]. Eniyiye yakın çözümler veren algoritmaların benzetimlerini içeren grafiklerde eniyi çözümü veren tamsayı programlama yöntemi sonuçları da gösterilmektedir. Böylece

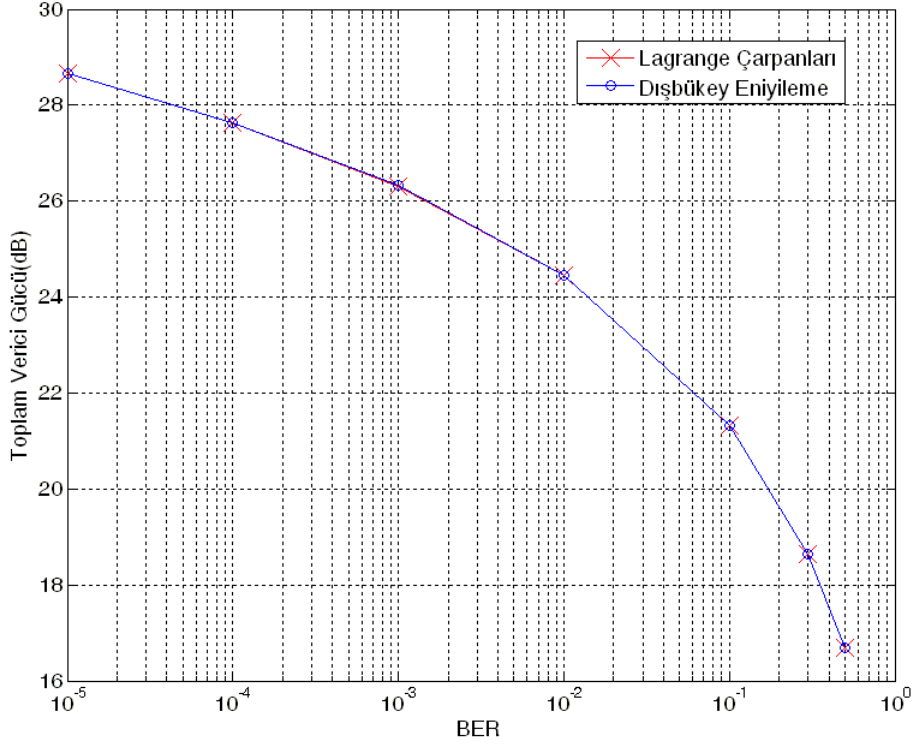
eniyye yakın algoritmalarının ne kadar başarılı (eniyye ne kadar yakın sonuçlar verdikleri) oldukları hakkında fikir edinilebilmektedir.

Benzetim çalışmalarında algoritmalar her parametre için 200 defa çalıştırılıp bulunan değerlerin ortalamaları alınarak sonuçlar elde edilmiştir. Temel olarak, toplam verici gücü çeşitli QoS (BER) ve kullanıcı sayısı değerleri için incelenmiştir.

Sürekli uzayda yapılan gevşetilmiş kaynak tahsisi, beklenildiği gibi ayrık uzayda yapılan eniyi kaynak tahsisine göre daha düşük maliyet işlevi değerleri sağlamaktadır. Burada sürekli uzayda kaynak tahsisi (2.9) probleminin çözümü anlamına gelmekte iken ayrık uzaydaki kaynak tahsisi (2.4) veya aynı problemin vektör-matris şeklinde ifadesi olan (4.14) probleminin çözümü anlamına gelmektedir. Sürekli uzayda yapılan kaynak tahsisi ayrık uzayda yapılan kaynak tahsisi için bir alt sınır oluşturmaktadır.

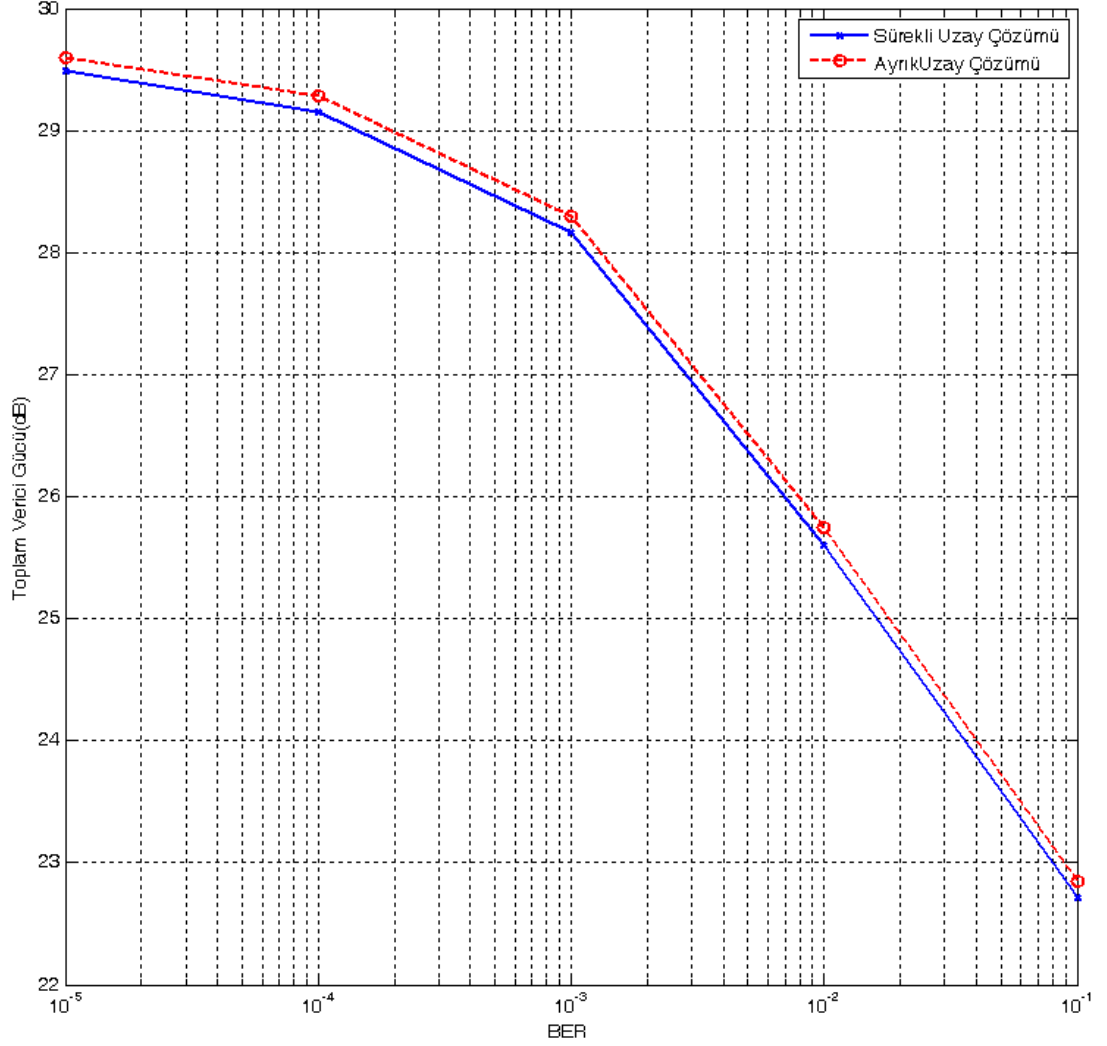
Farklı yöntemler kıyaslanırken, karşılaştırmaların adil ve anlamlı olabilmesi için algoritmalar aynı kanal ve parametreler üzerinden çalıştırılmıştır.

Şekil 7.3'te Dışbükey Eniyileme ile Lagrange Çarpanları Metodu farklı BER değerlerine göre kıyaslanmaktadır. Senaryoda kullanıcı sayısı  $K = 2$ , alt-taşıyıcı sayısı  $N = 32$ , bir alt-taşıyıcıya atanabilecek en çok bit miktarı  $M = 8$  ve kullanıcıların talep ettikleri bit miktarı çerçeve başına  $R_k = 32, \forall k$ 'dir. Dışbükey Eniyileme SQP yöntemini kullanırken Lagrange Çarpanları Metodu, [15]'de anlatılan yöntemi kullanmaktadır. Aynı problemi eniyiledikleri için beklenildiği gibi iki yöntem de aynı sonucu vermektedirler. Şekil 7.3 incelendiğinde, daha düşük BER değerleri için (daha düşük BER değeri daha iyi bir QoS anlamına gelmektedir.) daha fazla güce ihtiyaç duyulduğu görülmektedir. Dışbükey Eniyileme'nin Lagrange Çarpanları Metodu'na göre avantajı aynı sonucu daha çabuk bulmasıdır. Fakat Şekil 7.3'de kullanılan kullanıcı sayısı ve alt-taşıyıcı miktarları arttırıldığında Dışbükey Eniyileme sonucu Lagrange Çarpanları Metodu'na göre çok daha kötü bir performans göstermektedir. Dışbükey eniyileme her iterasyonda gradyan ve Hessian işlemleri yapılmaktadır. Gradyanın alındığı noktanın bileşenleri sıfıra yaklaştığı zaman gradyan sonsuza gitmektedir. Bu durumu engellemek için gradyan sınırlandırıldığında ise Dışbükey Eniyileme'nin yakınsama süresi oldukça artmaktadır.



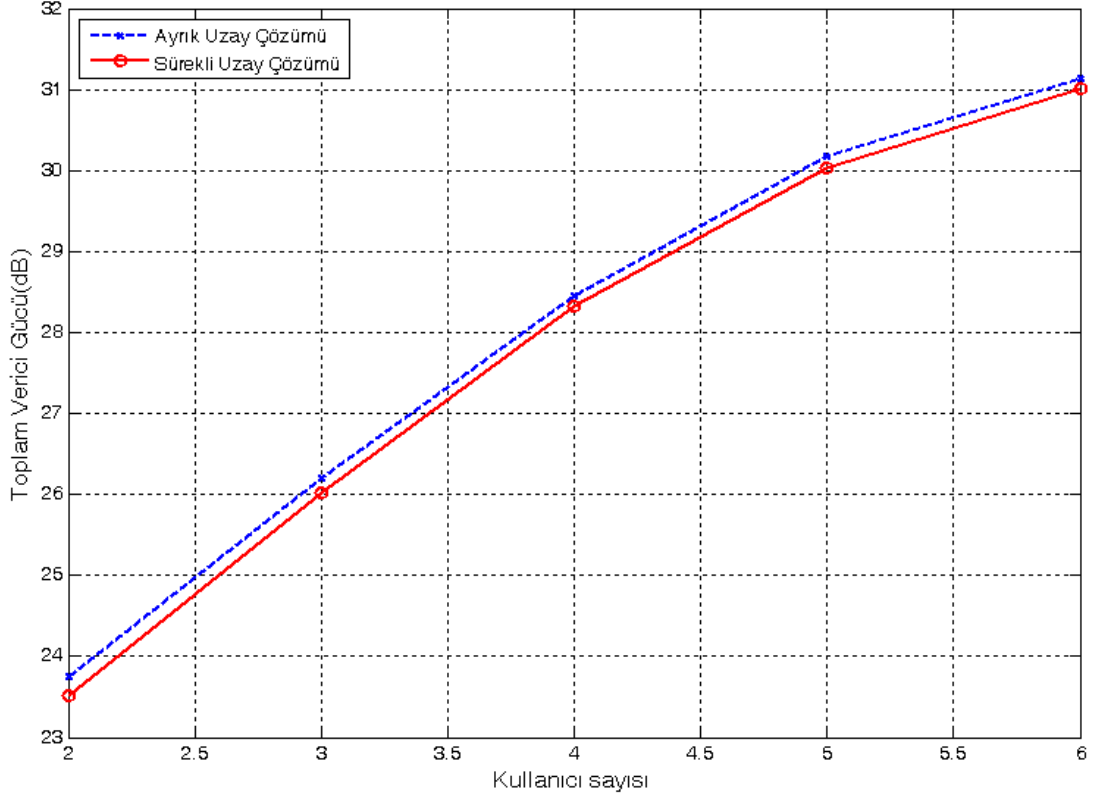
Şekil 7.3: Sürekli uzayda OFDMA kaynak tahsisi ( $K = 2$ ,  $N = 32$ ,  $M = 8$ ,  $R_k = 32, \forall k$ )

Şekil 7.4'de farklı BER değerleri için tamsayı kısıtlarının gevşetildiği sürekli uzaydaki ve tamsayı kısıtların varlığını koruduğu ayrık uzaydaki benzetimler karşılaştırılmaktadır. Sürekli uzaydaki problemin (2.9) çözümü için, Lagrange çarpanları kullanılırken ayrık uzay için Dallar ve Sınırla Algoritması kullanılmıştır. Senaryoda kullanıcı sayısı  $K = 4$ , alt-taşıyıcı sayısı  $N = 64$ , bir alt-taşıyıcıya atanabilecek en çok bit miktarı  $M = 6$  ve kullanıcıların talep ettikleri bit miktarı çerçeve başına  $R_k = 32$ 'dir. Lagrange Çarpanları Yöntemi ile sürekli uzayda eniyi çözüme ulaşılırken ayrık uzayda da Dallar ve Sınırla Yöntemi eniyi çözümü sağlamaktadır. Sürekli uzayda yapılan eniyileme, ayrık uzayda yapılan eniyilemeye göre daha geniş bir uzay üzerinden yapıldığı için ayrık zamana göre bir alt-sınır oluşturmaktadır.



Şekil 7.4: Farklı BER değerleri için sürekli ve ayrık uzayda OFDMA kaynak tahsisi ( $K = 4$ ,  $N = 64$ ,  $M = 6$ ,  $R_k = 32, \forall k$ ) Sürekli uzay için Lagrange Çarpanları, ayrık uzay için Dallar ve Sınırla Yöntemi kullanılmıştır.

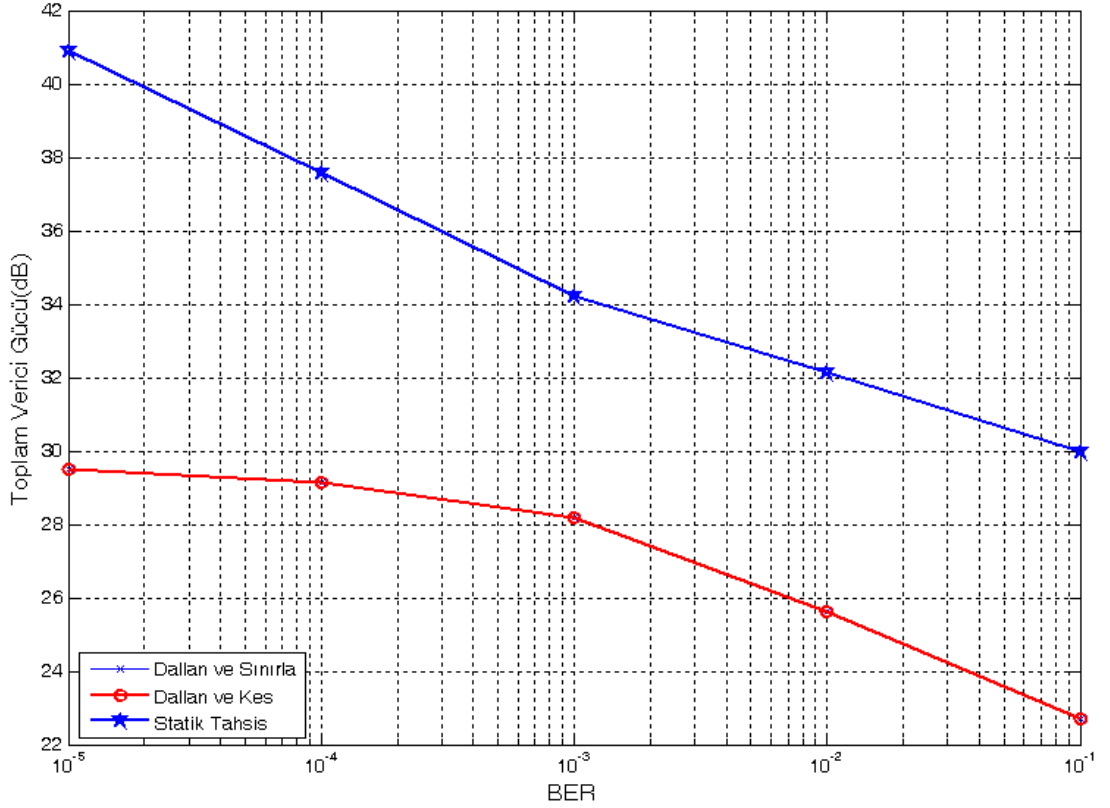
Şekil 7.5'te ise sürekli uzay ve ayrık uzaydaki problemler için farklı kullanıcı sayılarını içeren benzetim sonuçları görülmektedir. Benzetimdeki senaryo Şekil 7.4'deki senaryo ile aynıdır. Her nokta için seçilen BER değeri ise  $10^{-3}$ 'tür. Burada da sürekli uzaydaki problem Lagrange Çarpanları Yöntemi ile çözülürken ayrık uzaydaki problem Dallar ve Sınırla yöntemi ile çözülmektedir. Beklenildiği gibi kullanıcı sayısı arttıkça verici daha fazla güce ihtiyaç duymakta ve sürekli uzaydaki çözüm ayrık uzaydaki çözüme göre bir alt sınır oluşturmaktadır.



Şekil 7.5: Farklı kullanıcı sayıları için sürekli ve ayrık uzayda OFDMA kaynak tahsisi  $BER = 10^{-3}$ ,  $N = 64$ ,  $M = 6$ ,  $R_k = 32, \forall k$ ).

Şekil 7.6'da yapılan benzetim çalışmasında hepsi de tamsayı sonuçlar veren Dallan ve Sınırla, Dallan ve Kes Algoritmaları ile Statik Kaynak Tahsisi farklı BER değerleri için kıyaslanmaktadır. Senaryoda kullanıcı sayısı  $K = 4$ , alt-taşıyıcı sayısı  $N = 64$ , bir alt-taşıyıcıya atanabilecek en çok bit miktarı  $M = 6$  ve kullanıcıların talep ettikleri bit miktarı çerçeve başına  $R_k = 32, \forall k$ 'dir. Statik kaynak tahsisinde her kullanıcıya blok halinde ve eşit miktarda alt-taşıyıcı (16'şar) atanmaktadır. Alt-taşıyıcılar kullanıcılara eşit olarak paylaştırıldıktan sonra bitler de alt-taşıyıcılara eşit olarak atanmaktadır. Dolayısıyla her alt-taşıyıcıya 2 adet bit atanmıştır. Görüldüğü gibi atama işlemleri sırasında kullanıcıların kanal durumları gözetilmemektedir. Dallan ve Sınırla Algoritması ile Dallan ve Kes Algoritması kıyaslandığında, beklenildiği gibi ikisinin de aynı sonucu verdikleri görülmektedir. Çalışma süreleri kıyaslandığında, Dallan ve Kes Algoritması Dallan ve Sınırla Algoritması'na göre önemli bir avantaj sağlamamaktadır. Gomory kesmelerini kullanan Dallan ve Kes Algoritması'nda,

alternatif kesmeler kullanılarak daha farklı performanslar edilebileceği öngörülmektedir. Statik Kaynak Tahsisi diğer iki yöntemle kıyaslandığında statik tahsisin diğer iki yönteme göre oldukça yüksek güç değerlerine ihtiyaç duyduğu görülmektedir. Böylece uyarlamalı algoritmaların, statik tahsise göre çok daha başarılı sonuçlar verdikleri gerçeği ortaya çıkmaktadır. Öte yandan statik kaynak tahsisi kanal bilgisine ihtiyaç duymamaktadır. Dolayısıyla vericinin kanal hakkında hiç bir bilgiye sahip olmaması durumunda kullanılması gerekli olabilir.



Şekil 7.6: Statik Tahsis, Dallan ve Sınırla, Dallan ve Kes ile kaynak tahsisi  $K = 4$ ,  $N = 64$ ,  $M = 6$ ,  $R_k = 32, \forall k$ ).

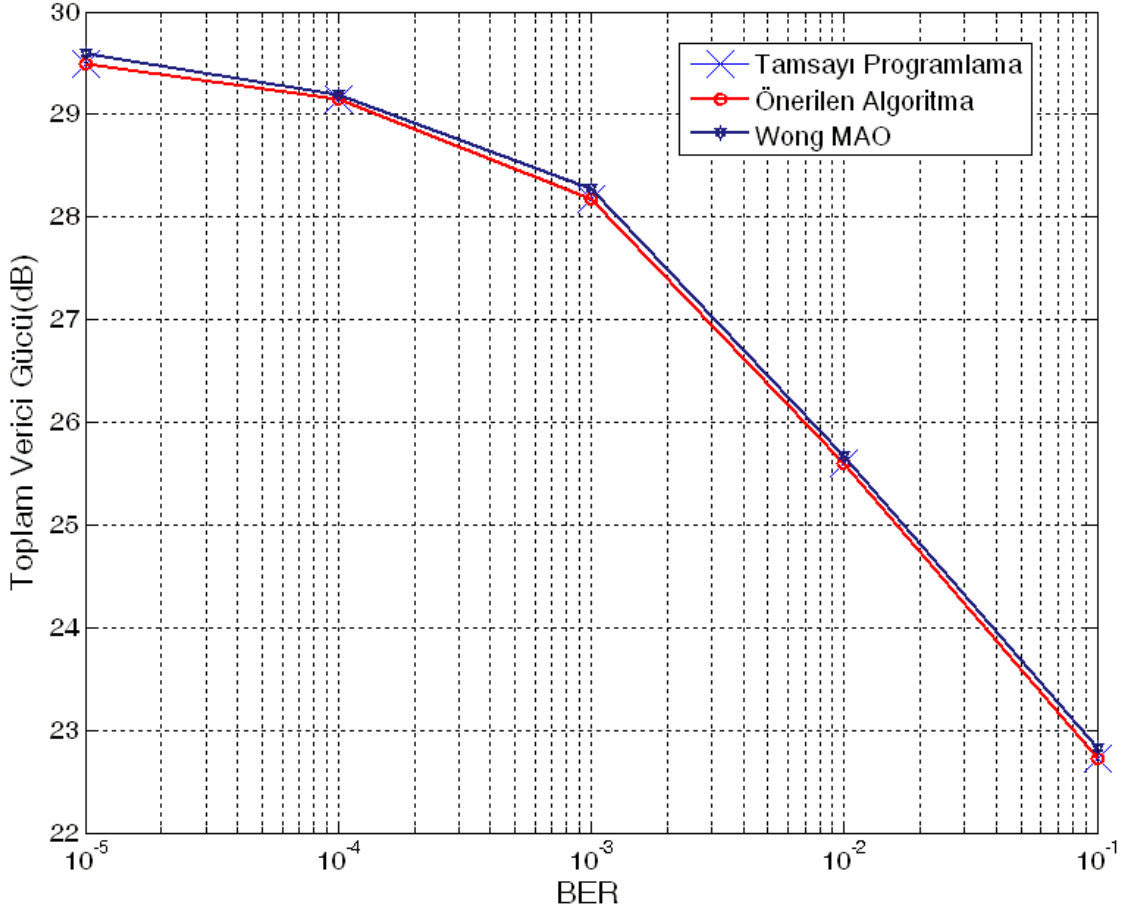
Yapılan çalışmalarda tamsayı programlama ile tamsayı kısıtların gevşetilmesi sonucu oluşan doğrusal problemlerin çözümlerinin birbirlerine oldukça yakın oldukları görülmüştür. Burada hareketle iki aşamalı bir Simpleks Açgözlü Algoritma önerilmiştir. Önerilen algoritmada ilk adımda doğrusallaştırılmış problemde tamsayı kısıtlar gevşetilerek oluşturulan problem Simpleks yöntemiyle çözülmektedir. Simpleks yöntemiyle bulunan çözüm değişkeninin her elemanının tamsayı olması durumu Sim-

pleks yöntemi ile tamsayı programlama metodlarının çözümlerinin çakıştığı anlamına gelmektedir. Dolayısıyla algoritma sonlandırılır. Aksi halde birden fazla kullanıcıya atanmış alt-taşıyıcılar için kullanıcılar rekabete girmektedir. Aynı alt-taşıyıcıyı talep eden birden fazla kullanıcı arasından en yüksek kanal katsayısına sahip olan kullanıcı o alt-taşıyıcıya sahip olmakta iken, diğer kullanıcılar o alt-taşıyıcıdan mahrum bırakılır. Alt-taşıyıcı tahsisi bu şekilde tamamlandıktan sonra her kullanıcı için bit tahsisi Açgözlü Algoritma ile yapılarak algoritma tamamlanır.

Wong, MAO (Multiuser Adaptive OFDM, Çok kullanıcılı Uyarlamalı OFDM) isimli bir algoritma geliştirmiştir. Algoritmada ilk adımda Lagrange çarpanları yinelemeli olarak ayarlanarak bir çözüme ulaşmaya çalışılmaktadır. Daha sonra ulaşılan alt-taşıyıcı tahsisi bilgisi kullanılarak bit tahsisi Açgözlü Algoritma ile yapılmaktadır.

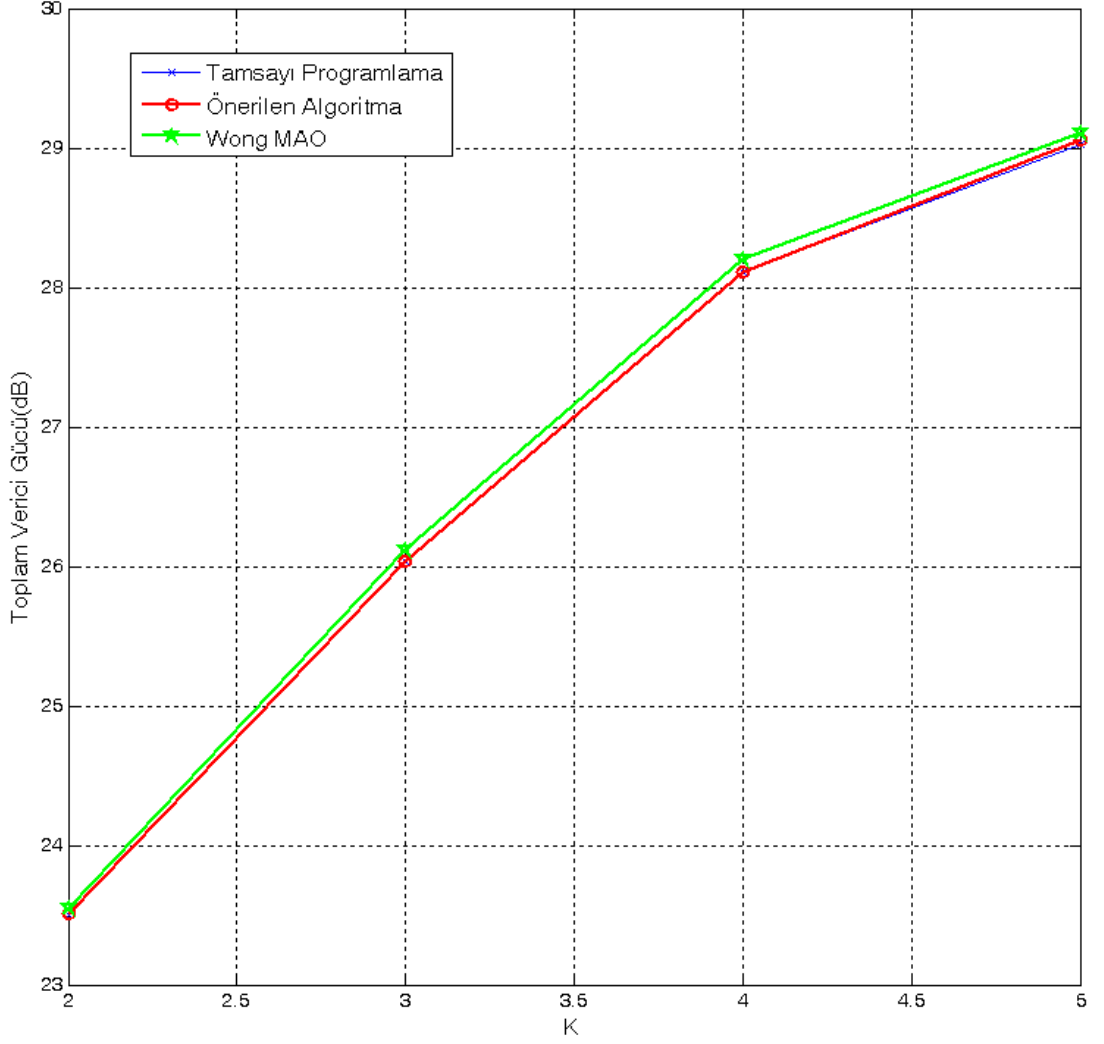
Şekil 7.7’de farklı BER değerleri için Wong MAO Algoritması ve Önerilen Algoritma kıyaslanmaktadır. İncelenen senaryoda ise kullanıcı sayısı  $K = 4$ , alt-taşıyıcı sayısı  $N = 64$ , bir alt-taşıyıcıya atanabilecek en çok bit miktarı  $M = 6$  ve kullanıcıların sembol başına ihtiyaç duydukları bit miktarı  $R_k = 32, \forall k$  olarak seçilmiştir. Dallar ve Sınırla Algoritması ile çözülen Tamsayı Programlama eğrisi ise eniyi sonucu verdiği için eniyiye yakın çözümler için bir alt-sınır oluşturmaktadır. Hem Wong MAO hem de Önerilen Algoritma eniyi tamsayı çözüme yakın değerler vermektedirler. Önerilen Algoritma Wong MAO yöntemine göre daha basit olmasında rağmen Wong MAO’dan kısmen daha iyi başarıyı sağlayabilmektedir. Wong MAO yöntemi hesaplaması uzun olan Lagrange çarpanlarını kullandığı için efektif bir yöntem değildir. Ayrıca Wong, her iterasyonda Lagrange çarpanlarının ”yeteri kadar küçük” olacak şekilde arttırılmasını önermiştir. Dolayısıyla adım adım arttırılan Lagrange çarpanlarında, adımların büyüklüğü yüksek olursa algoritma daha hızlı çalışacak fakat daha kötü sonuçlar üretecektir. Dolayısıyla adımların seçilmesi algoritmanın başarımında kritik bir rol oynarken, adım boyu için matematiksel bir ifade önerilmemiştir. Öte yandan Açgözlü Simpleks Algoritma’da böyle bir sorun yokken, aynı zamanda kaynak tahsisinin Simpleks yöntemi yerine İç Nokta Metodları (Interior Point Methods) kullanılırsa algoritma daha da hızlı çalışabilme potansiyeline sahiptir.





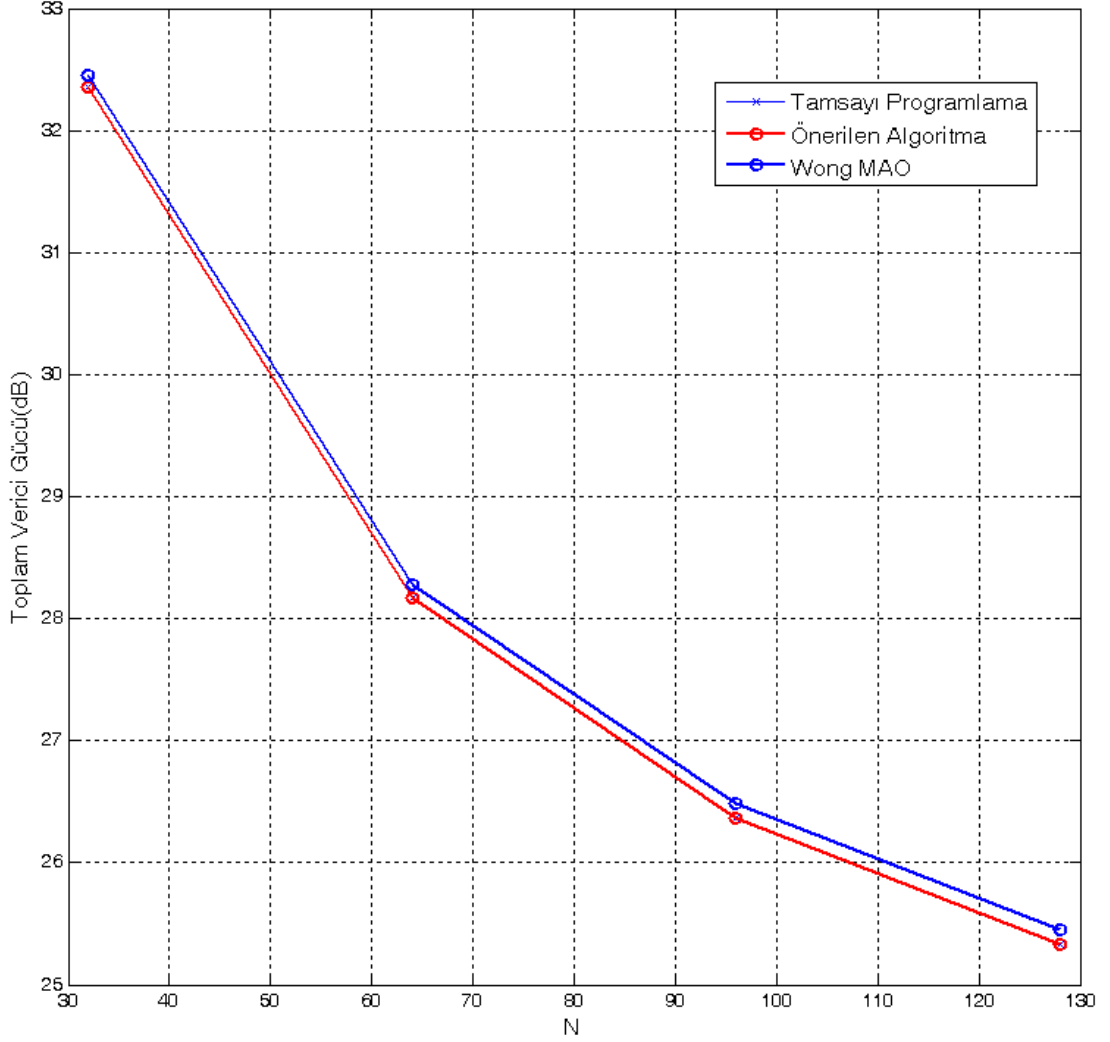
Şekil 7.7: Wong MAO, Önerilen Algoritma ile kaynak tahsisi ( $K = 4$ ,  $N = 64$ ,  $M = 6$ ,  $R_k = 32, \forall k$ ) Tamsayı Programlama için Dallon ve Sımla Algoritması kullanılmıştır.

Şekil 7.8'de farklı kullanıcı sayıları için Wong MAO ve Önerilen Algoritma kıyaslanmaktadır. İncelenen senaryoda kullanıcı sayısı  $K = 4$ , alt-taşıyıcı sayısı  $N = 64$ , bir alt-taşıyıcıya atanabilecek en çok bit miktarı  $M = 6$  ve kullanıcıların sembol başına ihtiyaç duydukları bit miktarı  $R_k = 32, \forall k$  olarak seçilmiştir. Tamsayı programlama yöntemi olarak Dallon ve Sımla Algoritması seçilmiştir. Wong MAO ve Önerilen Algoritma, bu durumda da birbirlerine yakın sonuçlar vermektedirler.



Şekil 7.8: Wong MAO, Önerilen Algoritma ve Statik tahsis yöntemleri ile kaynak tahsisi.  $BER = 10^{-3}$ ,  $N = 64$ ,  $M = 6$ ,  $R_k = 32, \forall k$ .

Şekil 7.9’da farklı alt-taşıyıcı sayılarının etkileri incelenmiştir. Benzetimlerde kullanılan alt-taşıyıcı sayıları 32, 64, 96 ve 128’dir. Şekilden gözlendiği gibi kullanıcıların talep ettiği veri hızı  $R_k$  aynı kalmak üzere alt-taşıyıcı sayısı arttıkça alt-taşıyıcılara atanan toplam güç miktarı azalmaktadır. Çünkü artan alt-taşıyıcı sayısı beraberinde kullanıcılara daha çok yüksek katsayılı kanallara sahip olma imkânını getirmektedir. Öte yandan alt-taşıyıcı sayısı azaldıkça her bir alt-taşıyıcıya daha fazla bit atanması gerekliliği ortaya çıkmaktadır.

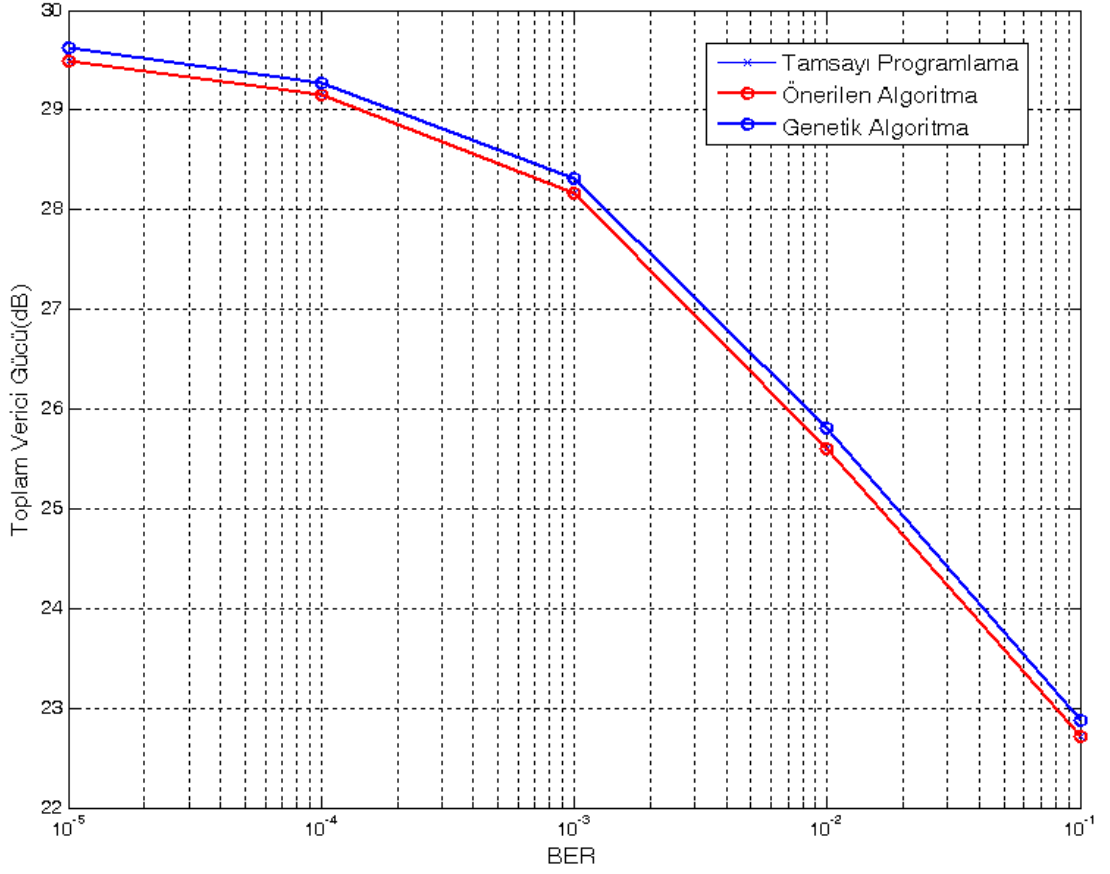


Şekil 7.9: Wong MAO, Önerilen Algoritma yöntemleri ile kaynak tahsisi.  $K = 4$ ,  $BER = 10^{-3}$ ,  $M = 6$ ,  $R_k = 32, \forall k$ .

OFDMA sistemlerde kaynak tahsisi, Wang [31] tarafından gerekirci yöntemlere bir alternatif olarak genetik algoritmalar ile çözülmüştür. Algoritmada nüfusun boyutu 36 olarak seçilmiştir. Dolayısıyla nüfusta 36 adet aday çözüm vardır. En düşük uygunluk işlevi değerlerine sahip 6 bireye elitizm işlemi uygulanarak bir sonraki nesilde de varlıklarını deęişmeden korumaları sağlanmıştır. Böylece her yeni nesilde, en düşük uygunluk işlevi değeri en kötü olasılıkla bir önceki jenerasyona eşit olacaktır. Nüfusun bireyleri arasındaki çiftleşme olasılığı 0.9 olarak belirlenmiştir. Çaprazlama işlemi ise iki noktalı çaprazlama şeklinde olmaktadır. Mutasyon olasılığı ise uyarlamalı

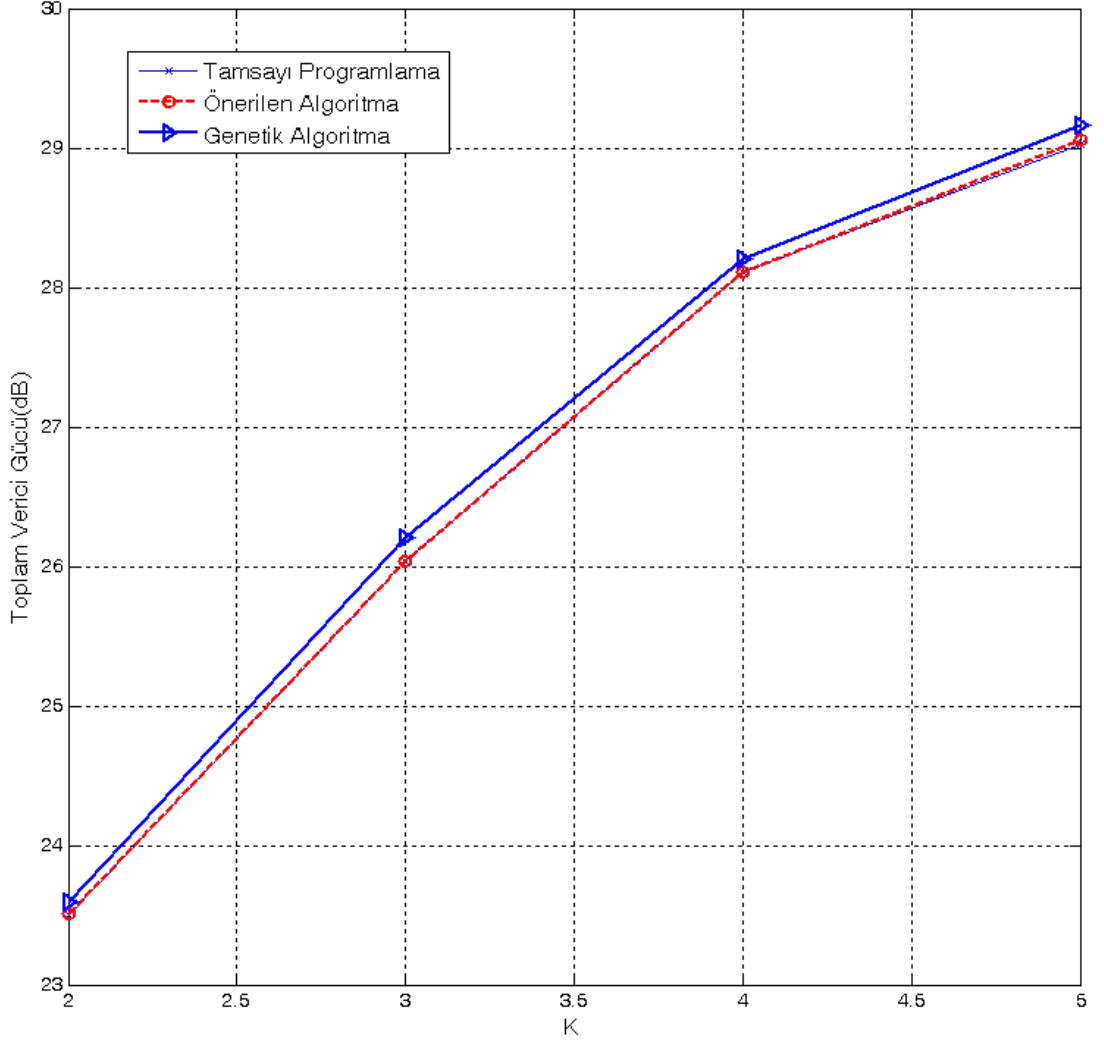
olarak değişmektedir. Başlangıç olarak mutasyon olasılığı 0.1'e ayarlanmaktadır. Eğer ard arda gelen beş nesil boyunca daha iyi uygunluk işlevi değerleri elde edilemezse mutasyon olasılığı 0.2'ye, on nesil boyunca daha iyi bir çözüm bulunamazsa 0.4'e ve on beş nesil boyunca daha iyi bir çözüm bulunamazsa 0.5'e ayarlanmaktadır. Algoritma 200 nesil sonunda sona ermektedir. Algoritmanın durma kriteri toplam iterasyon sayısının 200 olması ile belirlendiği için algoritma sonlu bir süre içinde duracaktır. Son nesildeki en düşük uygunluk işlevi değerine sahip birey algoritmanın bulduğu çözüm olarak atanmaktadır. Bulunan çözümün, genetik algoritmaların genel karakteristiği gereği eniyi çözüm olacağına bir garantisi yoktur.

Şekil 7.10'da görüldüğü gibi genetik algoritma kullanılarak gerçekleştirilen kaynak tahsisi en iyiye yakın değerler vermiştir. İncelenen senaryoda ise kullanıcı sayısı  $K = 4$ , alt-taşıyıcı sayısı  $N = 64$ , bir alt-taşıyıcıya atanabilecek en çok bit miktarı  $M = 4$  ve kullanıcıların sembol başına ihtiyaç duydukları bit miktarı  $R_k = 32, \forall k$  olarak seçilmiştir. Önerilen Algoritma ile Genetik Algoritma'nın performansları birbirine benzerdir. Genetik algoritmanın ne kadar iyi bir sonuç bulacağı tam olarak belli olmasa da her iterasyonda, tamsayı kısıtlarını sağlayan olurlu çözümleri bünyesinde barındırmaktadır. Dolayısıyla pratikte algoritmanın durma kriteri olarak belli bir süre belirlenirse, süre sonunda (eniyi çözüme göre ne kadar başarısız olduğu bilinmese de) halihazırda bir tamsayı olurlu çözüm bulunmaktadır.



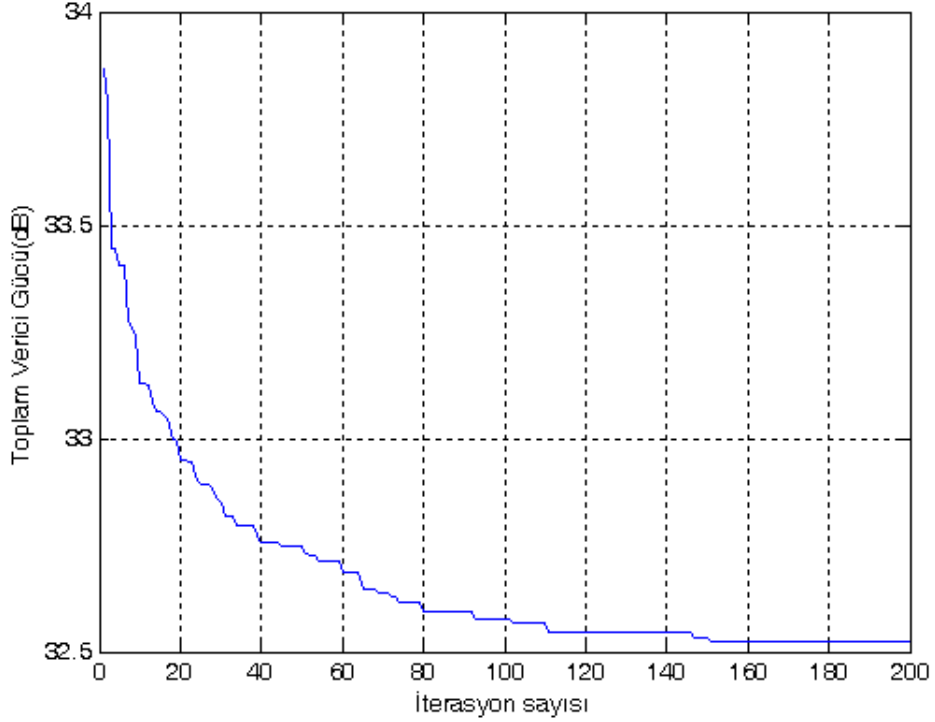
Şekil 7.10: Önerilen Algoritma ve Genetik Algoritma ile kaynak tahsisi.  $K = 4$ ,  $N = 64$ ,  $M = 6$ ,  $R_k = 32, \forall k$ .

Şekil 7.11’de ise farklı kullanıcı sayıları için Önerilen Algoritma ve Genetik Algoritma kıyaslanmıştır. Kullanılan senaryo Şekil 7.10’daki senaryo ile aynıdır. Tamsayı programlama ise Dallar ve Sınırla Algoritması ile yapılmıştır. Farklı kullanıcı sayıları için de her iki algoritma yakın sonuçlar ortaya çıkartmaktadır. Genetik algoritma parametrelerinin daha farklı seçilmesi (benzetimlerde [31] tarafından parametreler kullanılmıştır.) veya farklı sezgisel yöntemlerin kullanılmasıyla eniyi çözüme daha da yaklaşılacağı öngörülmektedir.



Şekil 7.11: Önerilen Algoritma ve Genetik Algoritma ile kaynak tahsisi.  $BER = 10^{-3}$ ,  $N = 64$ ,  $M = 6$ ,  $R_k = 32, \forall k$ .

Şekil 7.12’de Genetik Algoritma’daki iterasyonlar boyunca kullanıcılara dağıtılan gücün değişimi gösterilmektedir. İncelenen senaryoda ise kullanıcı sayısı  $K = 4$ , alt-taşıyıcı sayısı  $N = 64$ , bir alt-taşıyıcıya atanabilecek en çok bit miktarı  $M = 6$  ve kullanıcıların sembol başına ihtiyaç duydukları bit miktarı  $R_k = 32, \forall k$  olarak seçilmiştir.



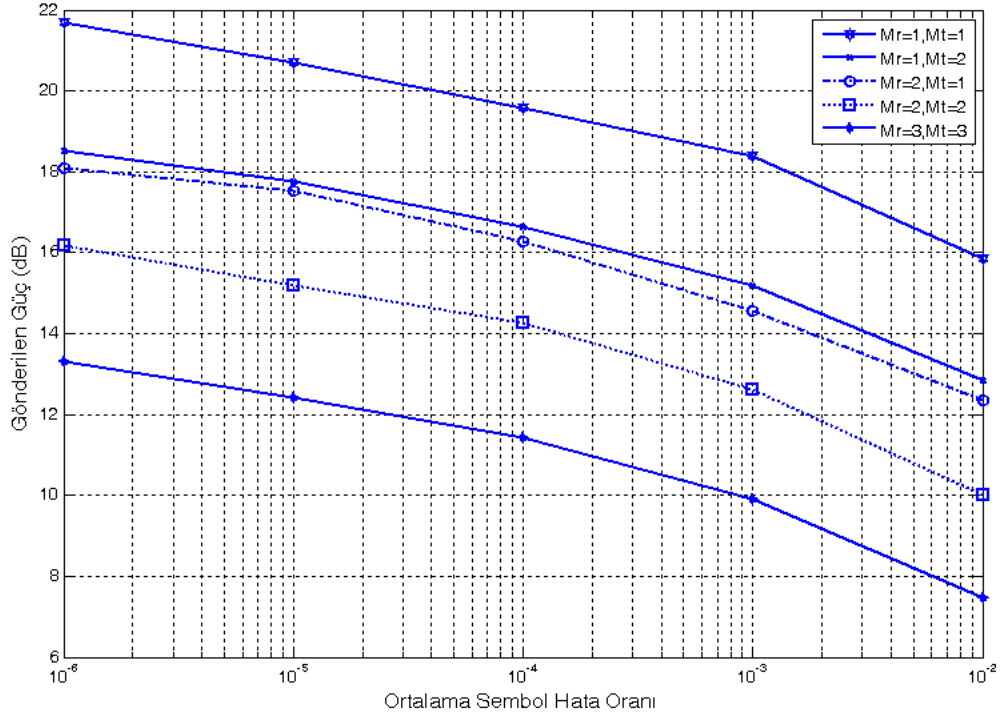
Şekil 7.12: Genetik Algoritma nesil güç dağılımı

## 7.2 Çok Kullanıcı MIMO OFDM Sistemlerde Kaynak Tahsisi Benzetimleri

Benzetim çalışmasında  $K = 3$  kullanıcı ve baz istasyondan oluşan bir telsiz iletişim sistemi senaryosu gözönüne alınmıştır. Verici ve alıcıdaki anten sayısı parametrik bırakılmış, her verici ile alıcı arasındaki kanalın frekans tepkisi, beş adet çokluyol(multipath) bileşenini temsilen, beş örnek uzunluğundaki sıfır ortalamalı, beyaz Gauss rasgele değişkenlerden oluşan normu bir olacak şekilde normalize edilmiş bir vektörün hızlı Fourier dönüşümü alınarak hesaplanmıştır. Toplam alt-taşıyıcı sayısı  $N = 8$ , ve her kullanıcının sembol başına ihtiyaç duyduğu bit miktarı  $R_k = 6, \forall k$  olarak seçilmiştir. Ayrıca bir kullanıcı bir alt-taşıyıcısından sembol başına en fazla  $M = 3$  bit gönderebilmektedir. Kaynak tahsisi yapıldıktan sonra bir çok-kullanıcı sezim tekniği kullanılarak kullanıcılar arası girişimin kaldırıldığı varsayılmaktadır.

Şekil 7.13'te değişik  $M_R$  ve  $M_T$  anten sayıları için verilen bir hedef sembol hatası oranında (QoS) ihtiyaç duyulan toplam güç incelenmektedir. Görülebileceği gibi anten sayısı  $M_R = 1$  ve  $M_T = 1$  olduğunda en kötü başarımlar sağlanmaktadır. Bunun

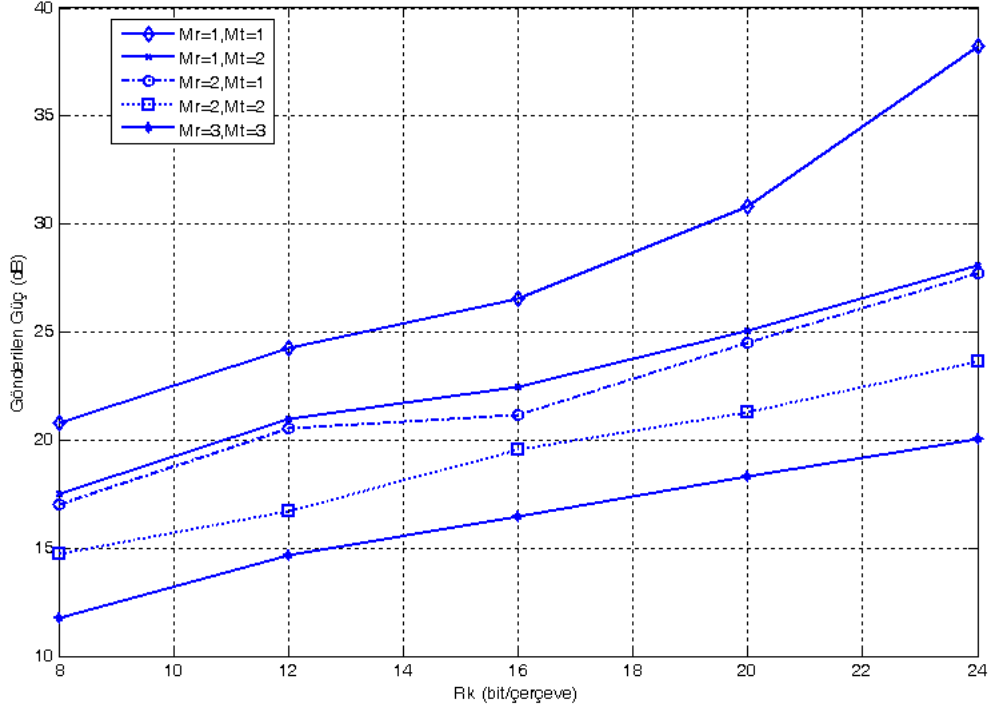
nedeni MIMO'nun sağladığı uzaysal ayrıma sahip olmayan bu senaryoda kullanıcılar-arası girişimin yüksek olması nedeniyle kullanıcıların aynı alt-taşıyıcıyı paylaşmak yerine sınırlı sayıda alt-taşıyıcıdan daha fazla bit göndermeyi tercih etmeleridir. Bunun doğal bir sonucu yükselen güçtür. Öte yandan eklenen her antenle MIMO tekniğinin sağladığı uzaysal ayırım sayesinde kullanıcılar-arası girişimin azalma potansiyeli nedeniyle aynı alt-taşıyıcıyı birden fazla kullanıcı kullanabilecektir. Bu şekilde kullanıcılar veri hızı kısıtını daha çok taşıyıcıya yayarak daha az güce gereksinim duyacaklardır. Şekil 7.13'de gözlenebilen bu durum sonucunda ( $M_R = 1, M_T = 1$ )'e kıyasla ( $M_R = 2, M_T = 2$ ) senaryosunda 6dB, ( $M_R = 3, M_T = 3$ ) senaryosunda 9dB'ye yakın bir kazanç ortaya çıkmaktadır.



Şekil 7.13: Verilen SER için ihtiyaç duyulan toplam güç, ( $K = 3, N = 8, M = 3, R_k = 6, \forall k$ )

Kullanıcıların farklı veri hızı talebinde buldukları Şekil 7.14'de ise yine alıcı ve verici anten sayısındaki artış ile daha düşük güç tüketimi sağlanabilmektedir. Kullanılan senaryo ise bir önceki senaryo ile aynıdır. Şekilde yatay eksen her kullanıcının talep ettiği için bit sayısını vermektedir.





Şekil 7.14: Verilen SER için ihtiyaç duyulan toplam güç, ( $K = 3$ ,  $N = 8$ ,  $M = 3$ ,  $SER = 10^{-3}$ )

Benzetim sonuçları incelendiğinde, eniyiye yakın Wong MAO, Önerilen Algoritma ve genetik algoritma ile kaynak tahsisi birbirlerine yakın sonuçlar verdikleri gözlenmektedir. Alıcı ve vericide birden fazla anten kullanılması (MIMO) ise aynı veri hızındaki iletişimin çok daha düşük güçlerde yapılmasına imkan vermektedir.

## 8. SONUÇLAR

OFDMA sistemlerde kaynak tahsisi problemi RA ve MA olmak üzere ikiye ayrılmaktadır. RA problemde amaç güç kısıtına göre veri hızını yükseltmek iken MA problemde amaç veri hızı kısıtına göre gücü azaltmaktır. OFDMA sistemlerde aynı alt-taşıyıcı birden fazla kullanıcı tarafından kullanılmadığı için yapısında çok-dorukluluk ve yüksek işlemsel karmaşıklık barındırmaktadır.

OFDMA sistemlerde kaynak tahsisi problemi oldukça güncel bir problem olup halen aktif olarak araştırılmakta ve yeni algoritmalar önerilmektedir.

Literatürde önerilen kaynak tahsisi yöntemlerinin büyük çoğunluğu eniyiden ödün vermek pahasına hızlı algoritmalar geliştirmeyi hedeflemiştir. Bunun içinde iki aşamalı algoritmalar geliştirilmiştir. İlk aşamada alt-taşıyıcı tahsisi yapılırken ikinci aşamada bit tahsisi yapılmaktadır.

Wong, OFDMA sistemlerde MA kaynak tahsisi problemini modellemiş ve dışbükey olmayan kaynak tahsisi problemini tamsayı kısıtları gevşeterek dışbükey hale getirmiştir. Bunun dışında kaynak tahsisi için eniyiye yakın çözüm üreten bir yöntem (Wong MAO) önermiştir.

Kaynak tahsisi probleminin en iyi çözümü çeşitli tamsayı programlama teknikleri kullanılarak çözülebilmesine rağmen bu teknikler parametre boyutları arttıkça (kullanıcı sayısı, alt-taşıyıcı sayısı, kullanıcıların talep ettikleri veri hızı) verimsiz hale gelmektedir. Spesifik olarak Dallan ve Sınırla ile Dallan ve Kes algoritmaları kıyaslandığında Dallan ve Kes algoritmasının aynı sonuçları veren Dallan ve Sınırla algoritmasına göre işlem karmaşıklığı bakımından kayda değer bir gelişme sağlamadığı görülmüştür.

Tezde Simpleks ve Ağgözlü Algoritmalarını kullanan hızlı bir algoritma önerilmiştir. Yöntem eniyi çözümü bulamasa da eniyiye oldukça yakın sonuçlar sağlamakta ayrıca Wong MAO algoritmasına üstünlük sağlamaktadır. Algoritmanın çalışma süresi ise Dallan ve Sınırla algoritmasının yarısından daha azdır.

Deterministik yöntemler dışında sezgisel yöntemler de incelenmiştir. Genetik algoritma, OFDMA sistemlerde kaynak tahsisi probleminde başarılı sonuçlar sağlamaktadır. Genetik algoritma, iterasyonlarının herhangi bir anında mutlaka tüm kısıtları sağlayan bir çözüm sağlamaktadır. Bu bir avantaj olarak görülebilir.

Problemin tamsayı elemanlara sahip çözüm uzayından yola çıkarak daha efek-

tif kaynak tahsisi yöntemleri geliştirilebileceği öngörülmektedir. Bu sayede şu anda dezavantaj olan aynı alt-taşıyıcının birden fazla kullanıcı tarafından kullanılamaması problemi avantaj haline dönüştürülebilir.

MIMO sistemlerin, kaynak tahsisinde SISO sistemlere göre kayda değer bir avantaj sağladığı görülmüştür. Fakat MIMO sistemlerin pratikte uygulanmalarında bir takım problemler olmaktadır. (örneğin mobil istasyonlara konabilecek anten miktarı)

İleride teknolojinin gelişmesi ile daha yüksek işlem yapma kapasitesine sahip mikroişlemciler ile halihazırda bulunan tamsayı programla teknikleri daha yüksek parametrelere sahip kaynak tahsisi problemlerini de çözebilecek hale geleceklerdir. Kaynak tahsisi problemi doğrusal vektör (maliyet işlevi) ve matris (kısıtlar) şeklinde standart biçimde ifade edilebildiği için ileride önerilecek yeni tamsayı programlama teknikleri ile de çözülebilecektir.

**KAYNAKLAR**

- [1] Goldsmith, A., 2005, Wireless Communications, Cambridge University Press.
- [2] Tse, D. and Viswanath, P., 2005, Fundamentals of Wireless Communication, Cambridge University Press, Cambridge.
- [3] Haykin, S., 2001, Communication Systems, John Wiley and Sons.
- [4] Kühn, V., 2006, Wireless Communications over MIMO Channels, John Wiley and Sons, England.
- [5] Verdu, 1998, Multiuser Detection, Cambridge University Press.
- [6] Paulraj, A., Nabar, R., and Gore, D., 2003, Introduction to Space-Time Wireless Communications, Cambridge University Press, Cambridge.
- [7] Shen, Z., 2006, Multiuser Resource Allocation in Multichannel Wireless Communication Systems, Doktora Tezi, University of Texas.
- [8] IEEE 802.11, <http://grouper.ieee.org/groups/802/11>
- [9] ETSI DVB Standard, <http://www.dvb.org>
- [10] ETSI LAN2 Standard, <http://www.etsi.org/website/technologies/hiperlan.aspx>
- [11] Andrews, J.G., 2007, Fundamentals of WiMAX, Prentice Hall.
- [12] Luenberger, D., 2003, Linear and Nonlinear Programming, Springer.
- [13] Papadimitriou, C., Steiglitz, K., 1998, Combinatorial Optimization: Algorithms and Complexity, Dover Publications.
- [14] Mitchell, J., 1998, Branch-and-Cut Algorithms for Integer Programming.
- [15] Wong, C., Cheng, R.S., Lateief, K.B., Murch, R.D., Oct 1999, Multiuser OFDM with Adaptive Subcarrier, Bit, and Power Allocation, IEEE Journal on Selected Areas in Communications, vol.17, no.10, pp.1747-1758.
- [16] Luenberger, D., 1969, Optimization by Vector Space Methods, Wiley.

- [17] Ermolova, N., Makarevitch, B., 2007, Low complexity adaptive power and sub-carrier allocation for OFDMA, *IEEE Transactions on Wireless Communications*, vol.6, no.2, pp.433-437.
- [18] Dardari, D., 2004, Ordered Subcarrier Selection Algorithm for OFDM-Based High-Speed WLANs, *IEEE Transactions on Communications*, Vol 3, No 5.
- [19] Giovanidis, A., Sezgin, A., Monich, U., Donghee K., 2007, Dynamic User Grouping and Shared Frequency Resource Assignment Strategies for OFDMA, *IEEE Vehicular Technology Conference* pp.2746-2750.
- [20] Jang, J., Lee, K., 2003, Transmit Power Adaptation for Multiuser OFDM Systems, *IEEE Journal on Selected Areas in Communications*, vol.21, no.2, pp. 171-178.
- [21] Kim, K., Han, Y., Kim, S., 2005, Joint Subcarrier and Power Allocation in Uplink OFDMA Systems, *IEEE Communications Letters*, vol.9, no.6, pp. 526-528.
- [22] Rhee, W.; Cioffi, J.M., 2000, Increase in Capacity of Multiuser OFDM System Using Dynamic Subchannel Allocation, *IEEE Vehicular Technology Conference* vol.2, pp.1085-1089.
- [23] Wang, L., Niu, Z., 2005, Computationally Efficient Resource Allocation Algorithm for Multiuser OFDM Systems, *Asia-Pacific Conference on Communications*, pp.416-420.
- [24] Shen, Z., Andrews, J.G.; Evans, B.L., 2005, Adaptive Resource Allocation in Multiuser OFDM Systems with Proportional Rate Constraints, *IEEE Transactions on Wireless Communications*, vol.4, no.6, pp. 2726-2737.
- [25] Einbaus, M., Klein, O., Walke, B., Halfmann, R., 2007, MAC Level Performance Evaluation of Downlink Resource Allocation Strategies for an OFDMA System Based on IEEE 802.16, *IEEE Vehicular Technology Conference*, pp.2796-2800.
- [26] Zhang G, 2004, Subcarrier and Bit Allocation for Real-Time Services in Multiuser OFDM Systems, *IEEE International Conference on Communications* , vol.5, no., pp. 2985-2989.

- [27] Kivanc, D., Li G, Liu, H., 2003, Computationally Efficient Bandwidth Allocation and Power Control for OFDMA, *IEEE Transactions on Wireless Communications*, vol.2, no.6, pp. 1150-1158.
- [28] Choe, K., Lim Y., Park S., 2004, Subcarrier Adaptation for Multiuser OFDM Systems, *IEEE Global Telecommunications Conference*, vol.2, no., pp. 1230-1233.
- [29] Munz, G., Pfletschinger, S., Speidel, J., 2002, An Efficient Waterfilling Algorithm for Multiple Access OFDM, *IEEE Global Telecommunications Conference*, vol.1, no., pp. 681-685.
- [30] Han, Z., Ji, Z., Liu, K.J.R., 2005, Fair Multiuser Channel Allocation for OFDMA Networks Using Nash Bargaining Solutions and Coalitions, *IEEE Transactions on Communications*, vol.53, no.8, pp. 1366-1376.
- [31] Wang Y., Chen, F., Wei, G., 2005, Adaptive Subcarrier and Bit Allocation for Multiuser OFDM System Based on Genetic Algorithm,” *International Conference on Communications, Circuits and Systems*, vol.1, no., pp. 242-246.
- [32] Tang, Z., Zhu, Y., Wei, G., Zhu, J., 2007, Cross-Layer Resource Allocation for Multiuser OFDM Systems Based on ESGA, *IEEE Vehicular Technology Conference*, pp.1573-1577.
- [33] Reddy, Y.B., Phoha, V.V., 2007, Genetic Algorithm Approach for Resource Allocation in Multi-User OFDM Systems, *International Conference on Communication Systems Software and Middleware*, pp.1-6, 7-12.
- [34] Reddy, Y.B., Gajendar, N., Taylor, P., 2007, Computationally Efficient Resource Allocation in OFDM Systems: Genetic Algorithm Approach, *International Conference on Information Technology*, pp.36-41.
- [35] Kim, I., Lee, H., Kim, N., Lee, Y.H., 2001, On the Use of Linear Programming for Dynamic Subchannel and Bit Allocation in Multiuser OFDM, *IEEE Global Telecommunications Conference* pp.3648-3652.
- [36] Mao, Z., Wang, X.M., Lin, J., 2005, Fast Optimal Radio Resource Allocation in OFDMA System Based on Branch-and-Bound Method, *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing* pp.348-351.

- [37] Lin, J., Mao, Z., 2006, A Real-Time Radio Resource Allocation Scheme in OFDMA System, Canadian Conference on Electrical and Computer Engineering, pp.1757-1760.
- [38] Nocedal, J., Wright, S., 2006, Numerical Optimization, Springer.
- [39] Moon, T., Stirling, W., 2000, Mathematical Methods and Algorithms for Signal Processing, Prentice Hall.
- [40] Bertsimas, D., Weismantel, R., 2005, Optimization over Integers, Dynamic Ideas.
- [41] Cutting Plane Techniques, <http://www.cs.uky.edu/~lewis/cs-heuristic/text/integer/cutting.html>
- [42] Holland, J., 1975, Adaptation in Natural and Artificial Systems, University of Michigan Press.
- [43] Zhang, Y., Letaief, K., 2005, An Efficient Resource Allocation for Spatial Multiuser Access in MIMO/OFDM, IEEE Transactions on Communications, vol.53, pp.107-116.
- [44] Kim, K., Cioffi, J., 1999, Transmission Optimization with a Space-Time Filter at low SNR Environment, IEEE Global Telecommunications Conference, pp.889-893.
- [45] Li, D., Sun, X., 2006, Nonlinear Integer Programming, Springer.

## ÖZGEÇMİŞ

Adı Soyadı : Can Turgu

Doğum Yeri : Ankara

Doğum Yılı : 1983

Medeni Hali : Bekar

### Eğitim ve Akademik Durumu

Lise 1994-2001 : Yıldırım Beyazıt Anadolu Lisesi, ANKARA

Lisans 2001-2005 : Gazi Üniversitesi  
Elektrik ve Elektronik Mühendisliği Bölümü,  
Ankara

Yabancı Dil: İngilizce

### İş Tecrübesi

2008: Sistaş AŞ, Ankara

2006-2008 : Araştırma Görevlisi, Hacettepe Üniversitesi Elektrik Elektronik Mühendisliği, Ankara

2005: A-pro Mühendislik , Ankara