

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

**İLERİ NESİL KABLOSUZ SİSTEMLERDE DİNAMİK KAYNAK
TAHSİSİ VE KANAL KESTİRİMİ**

DOKTORA TEZİ

Mustafa Karakoç

Anabilim Dalı: Elektronik ve Haberleşme Mühendisliği

Danışman: Doç. Dr. Adnan KAVAK

KOCAELİ, 2008

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

**İLERİ NESİL KABLOSUZ SİSTEMLERDE DİNAMİK KAYNAK
TAHSİSİ VE KANAL KESTİRİMİ**

DOKTORA TEZİ

Mustafa KARAKOÇ

Tezin Enstitüye Verildiği Tarih: 02 Mayıs 2008

Tezin Savunulduğu Tarih: 09 Haziran 2008

Tez Danışmanı

Doç.Dr. Adnan KAVAK


(.....)

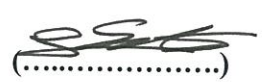
Üye

Doç.Dr. A.Arif ERGİN


(.....)

Üye

Prof.Dr. Sarp ERTÜRK


(.....)

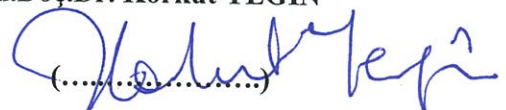
Üye

Prof.Dr. Hasan DİNÇER


(.....)

Üye

Yrd.Doç.Dr. Korkut YEĞİN


(.....)

KOCAELİ, 2008

ÖNSÖZ VE TEŞEKKÜR

Kablosuz haberleşme sistemi kullanıcı sayılarının kablolu haberleşme sistemini kullananların sayılarına eşitlendiği günümüzde, kullanıcıların ihtiyaçlarını verimli bir şekilde karşılayacak sistemlere gereksinim vardır. Sistemin gereksinim duyduğu bant genişliği bu ölçüde artmış ve dolayısıyla mevcut kaynağın en verimli bir şekilde kullanılması çok önemli hale gelmiştir. Bu tez çalışmasının temelini oluşturan üçüncü nesil kablosuz haberleşme sistemleri, kullanıcıların temel ihtiyaçları olan daha fazla hız, daha kaliteli iletişim ve daha çok kullanıcıyı istenilir servis kalitesinde destekleme olanağı sağlamayı amaçlamaktadır.

Bu tez çalışmasında üçüncü nesil kablosuz haberleşme sistemlerinde verimli kaynak tahsisi yapmak amacı ile kullanılan, birbirine dik değişken yayılım faktörü kodlarının performansı üzerinde durulmuştur. Önerilen algoritmalar ile kullanıcılara yönelik kod tahsisi ve tekrar tahsisi mümkün kılınmış, performans artışı gözlenmiş ve kod ağacının optimum kullanımı sağlanmıştır.

Bu çalışmanın her safhasında özveri ile yaklaşan danışmanım Sayın Doç. Dr. Adnan KAVAK'a (KO.Ü.), değerli yorumları ile beni yönlendiren ve aynı zamanda tez izleme jürisinde olan hocalarım Sayın Doç. Dr. A. Arif ERGİN (G.Y.T.E.) ve Sayın Prof. Dr. Sarp ERTÜRK'e (KO.Ü.), hocam Sayın Doç. Dr. Yaşar BECERİKLİ'ye (KO.Ü.), başta Alev SÖKE olmak üzere değerli yardımlarını esirgemeyen tüm arkadaşlarıma, bu noktaya gelişimde her zaman yanımda olan maddi ve manevi desteklerini esirgemeyen aileme teşekkürü bir borç bilirim.

İÇİNDEKİLER

ÖNSÖZ ve TEŞEKKÜR.....	i
İÇİNDEKİLER	ii
ŞEKİLLER DİZİNİ.....	v
TABLolar DİZİNİ	vii
SİMGELER ve KISALTMALAR	viii
ÖZET	xiii
İNGİLİZCE ÖZET.....	xiv
1. GİRİŞ	1
1.1. Kablosuz Haberleşme Sistemlerinin Gelişimi	1
1.1.1. Birinci nesil haberleşme sistemleri	1
1.1.2. İkinci nesil haberleşme sistemleri	2
1.1.3. Üçüncü nesil haberleşme sistemleri	6
1.2. Çalışmanın Amacı.....	9
1.3. Literatürde Yapılan Çalışmaların Özetleri	11
1.4. OVFS Kod Ağacı	16
1.5. Tezin Organizasyonu	17
2. GENİŞBANT KOD BÖLÜŞÜMLÜ ÇOKLU ERİŞİM (WIDEBAND CODE DIVISION MULTIPLE ACCESS, WCDMA).....	19
2.1. Giriş.....	19
2.2. Sistem Genel Görünüşü	20
2.3. WCDMA Temel Özellikleri.....	21
2.4. Protokol Mimarisi	23
2.4.1. Mantıksal kanallar	25
2.4.2. İletim kanalları	26
2.5. Fiziksel Kanallar	28
2.5.1. UL fiziksel kanalları.....	28
2.5.2. DL fiziksel kanalları.....	32
2.6. Çoklu Oranlı Kullanıcı Veri İletimi	36
2.6.1. İletim formatı takibi	41
2.6.2. Kanal kodlama	41
2.7. Yayılım ve Modülasyon.....	42
2.7.1. UL yayılımı	42
2.7.1.1. PRACH yayılımı	44
2.7.1.2. PCPCH yayılımı.....	45
2.7.2. DL yayılımı	46
2.7.2.1. Senkronizasyon kodları.....	48
2.7.2.2. Modülasyon.....	48
2.8. Verici Çeşitlemesi	49
2.9. Hava Arayüzü Prosedürleri	50
2.9.1. DL yayılımı	50
2.9.2. Hücreler arası geçiş.....	51
2.9.2.1. Frekanslar arası aktarımlar.....	53
2.9.2.2. GMS ve WCDMA arası aktarımlar	54

2.9.3. Güç kontrolü	56
2.9.4. UL senkronizasyon iletim şeması	56
2.9.5. Paket datalar	58
2.10. WCDMA Gelişimi, Release 2000 ve Sonrası	59
3. KOD SINIRLI ALGORİTMALAR	60
3.1. Giriş	60
3.2. Genetik Algoritma Tabanlı Dinamik OVSF Kod Tahsisi	61
3.2.1. Basit genetik algoritma (SGA)	62
3.2.2. Dominance ve diploidy tabanlı genetik algoritma (D&D-GA)	73
3.3. Benzetilmiş Tavlama Tabanlı Dinamik OVSF Kod Tahsisi	77
3.3.1. Metallerde tavlama	77
3.3.2. Fiziksel sistemler ve benzetilmiş tavlama	78
3.3.3. SA algoritmasına genel bakış	79
3.3.4. SA algoritması parametreleri	82
3.3.4.1. Çözümlerin gösterimi	82
3.3.4.2. Maliyet fonksiyonu	83
3.3.4.3. Komşuluk hareketi	83
3.3.4.4. Soğutma çizelgesinin seçimi	84
3.3.4.5. İç döngü ve dış döngü koşulunun belirlenmesi	89
3.4. Uyarlamalı Benzetilmiş Tavlama Tabanlı Dinamik OVSF Kod Tahsisi	90
4. SİNYAL-PARAZİT ORANI (SIR) SINIRLI ALGORİTMALAR	94
4.1. Giriş	94
4.2. Uyarlanabilir Anten Dizisi Tabanlı Dinamik Kod Tahsis Algoritması (AAA-DCA)	95
4.3. Çoklu Yol Yayılım Etkisinin Gözönünde Bulundurulması	97
4.3.1. Sinyal modeli	97
4.3.2. Parazit analizi ve sistem çıkış denklemi	100
4.4. Çoklu Yol Yayılım Etkisinin İhmali	107
4.4.1. Sinyal modeli	107
4.4.2. Parazit analizi ve sistem çıkış denklemi	107
4.5. Aşağı Bağlantı (DL) Işıma Örüntüsü Algoritmaları	109
4.6. OVSF Kod Tahsis Veritabanı	111
5. KOD SINIRLI ve SIR SINIRLI DİNAMİK OVSF KOD TAHSİS ALGORİTMALARININ PERFORMANSI	114
5.1. Giriş	114
5.2. Simülasyon Parametreleri ve Metodoloji	115
5.2.1. Simülasyon parametreleri	116
5.2.2. Simülasyon metodolojisi	117
5.2.2.1. Kod sınırlı algoritmalar için simülasyon metodolojisi	117
5.2.2.2. SIR sınırlı algoritmalar için simülasyon metodolojisi	118
5.3. Sistem Performansı	120
5.3.1. Bloklanma olasılığı	120
5.3.2. Spektral verimlilik	122
5.3.3. Çağrı model etkisi	124
5.4. Algoritmik Parametrelerin Etkisi	125
5.4.1. Farklı çaprazlama ve mutasyon oranlarının etkisi	125
5.4.2. Farklı seçim tekniklerinin etkisi	128
5.4.3. Farklı çaprazlama tekniklerinin etkisi	130

5.4.4. Farklı mutasyon tekniklerinin etkisi	132
5.4.5. Komşuluk hareketinin etkisi	134
5.4.6. Soğutma çizelgesinin etkisi.....	135
5.4.6.1. Genel sistem performansı üzerine etkisi	135
5.4.6.2. Yakınsama noktasının etkisi	136
5.4.6.3. İç döngünün etkisi	137
5.4.6.4. Başlangıç ve son sıcaklığın etkisi	139
5.4.7. Farklı AAA topolojilerinin etkisi	140
5.4.8. Açısal yayılımın etkisi	143
5.4.9. DL ışıma örüntüsü algoritmalarının etkisi	144
5.4.10. Anten elemanları arasındaki mesafenin etkisi.....	146
5.5. Maliyet Analizi	147
6. SONUÇLAR ve ÖNERİLER.....	149
KAYNAKLAR	152
EKLER.....	158
ÖZGEÇMİŞ	179

ŞEKİLLER DİZİNİ

Şekil 1.1: GSM ve GPRS sistemi.....	5
Şekil 1.2: UMTS sistem mimarisi.....	8
Şekil 1.3: Kod ağacı ve onun matrissel ifadesi	10
Şekil 1.4: OVSF kod ağacı ($SF=8$ için)	17
Şekil 2.1: WCDMA ile frekans kullanımı	21
Şekil 2.2: Hava arayüz protokol mimarisi.....	23
Şekil 2.3: Mantıksal kanal yapısı	24
Şekil 2.4: İletim ve mantıksal kanallar arasındaki bağlantıların adreslenmesi	27
Şekil 2.5: UL DPDCH/DPCCH için çerçeve yapısı	29
Şekil 2.6: RACH erişim yuvası sayısı ve yerleşim yerleri.....	30
Şekil 2.7: Rasgele erişim iletiminin yapısı.....	31
Şekil 2.8: Rasgele erişim mesaj kısmının radyo çerçeve yapısı.....	31
Şekil 2.9: CPCH rasgele erişim iletiminin yapısı.....	32
Şekil 2.10: DPCH DL için çerçeve yapısı.....	33
Şekil 2.11: Birincil CCPCH çerçeve yapısı	34
Şekil 2.12: İkincil CCPCH çerçeve yapısı	35
Şekil 2.13: Senkronizasyon kanal yapısı.....	35
Şekil 2.14: PDSCH çerçeve yapısı.....	36
Şekil 2.15: UL için çoklu oranlı bilgi transfer şeması	39
Şekil 2.16: DL için çoklu oranlı bilgi transfer şeması	40
Şekil 2.17: Bütün sekiz durumlu PCCC turbo kodlama	42
Şekil 2.18: Karmaşık yayılım devresi IQ/kod çoğullama	43
Şekil 2.19: UL DPCCH ve DPDCH'ler için yayılım.....	44
Şekil 2.20: PRACH'nin mesaj kısmının yayılımı	45
Şekil 2.21: PCPCH'nin mesaj kısmının yayılımı.....	45
Şekil 2.22: SCH hariç bütün DL fiziksel kanalların yayılımı	47
Şekil 2.23: SCH ve P-CCPCH'nin yayılımı ve modülasyon	48
Şekil 2.24: Modülasyon prensibi	49
Şekil 2.25: Sıkıştırılmış mod yapısı	54
Şekil 2.26: WCDMA ve GSM çerçeve yapıları arasındaki ölçüm zamanlaması ilişkisi	55
Şekil 2.27: RACH kanalı üzerinde paket iletimi.....	58
Şekil 3.1: GA bloğu kullanılarak gerçekleştirilen dinamik OVSF kod tahsis sisteminin akış diyagramı.....	62
Şekil 3.2: Örnek OVSF kod ağacı (başlangıç kromozomu) $SF=16$, $U=5$, $S=10$	63
Şekil 3.3: Başlangıç kromozomundan (Ch_{ini}) yola çıkarak başlangıç popülasyonunun oluşturulması	64
Şekil 3.4: OVSF kod tahsis problemine özel tanımlanan uygunluk fonksiyonunun sözde kodu.....	66
Şekil 3.5: Tek noktalı çaprazlama tekniğinin OVSF kod ağacına uygulanması.....	69
Şekil 3.6: Yer değiştirme mutasyon tekniğinin OVSF kod ağacına uygulanması.....	71
Şekil 3.7: "Punnet Square" olarak adlandırılan üçlü kodlama.....	75

Şekil 3.8: D&D-GA için çaprazlama operatörü	76
Şekil 3.9: SA algoritması kullanılarak OVSF kod ağacının optimizasyonunu gerçekleştiren akış diyagramı.....	80
Şekil 3.10: Çeşitli soğutma çizelgelerinin soğuma eğrileri ve formülleri.....	86
Şekil 3.11: Lundy&Mees ve orantısal azalımlı soğutma çizelgeleri.....	89
Şekil 3.12: Uyarlamalı benzetilmiş tavlama genetik algoritma ile OVSF kod ağacının optimizasyonuna ait akış diyagramı	92
Şekil 4.1: AAA-DCA algoritması kullanılarak OVSF kod ağacının optimizasyonunu gerçekleştiren akış diyagramı.....	96
Şekil 4.2: (a) İki katmanlı CDMA hücresele ağ yapısında parazit doğrultuları, (b) ULA topolojisi, (c) UCA topolojisi	97
Şekil 5.1: Farklı trafik yüklerinde bloklanma olasılıklarının karşılaştırılması.	121
Şekil 5.2: Farklı trafik yüklerinde spektral verimliliğin karşılaştırılması.....	123
Şekil 5.3: Farklı çağrı modellerinin kod bloklanma olasılığı üzerindeki etkisi	125
Şekil 5.4: SA algoritmasında tekrar tahsis edilen kullanıcıların soğutma çizelgesi üzerindeki yakınsama noktasının tespiti	137
Şekil 5.5: SA algoritmasında tekrar tahsis edilen kullanıcıların iç döngü sayısına göre değişimi	138
Şekil 5.6: SA algoritmasında tekrar tahsis edilen kullanıcıların başlangıç ve son sıcaklıklarına göre değişimi	140
Şekil 5.7: Farklı trafik yükleri ve farklı anten topolojilerinin etkisi	141
Şekil 5.8: Farklı trafik yükleri ve açısal yayılımının etkisi	144
Şekil 5.9: Farklı trafik yükleri ve DL ışıma örüntüsü algoritmalarının etkisi.....	145
Şekil 5.10: ULA ve UCA topolojileri için anten elemanları arası mesafe.....	146
Şekil 5.11: Farklı trafik yükleri ve anten elemanları arasındaki mesafenin etkisi ...	147

TABLÖLER DİZİNİ

Tablo 2.1: WCDMA Parametreleri	20
Tablo 2.2: 3GPP radyo erişim ağı özellikleri	22
Tablo 2.3: Mantıksal kontrol kanalları	25
Tablo 2.4: Mantıksal trafik kanalları (kullanıcı kanalları)	26
Tablo 2.5: Ortak iletim kanalları	27
Tablo 2.6: Birincil ve ikinci CPICH	34
Tablo 2.7: Hata düzeltme kodlama parametreleri	41
Tablo 2.8: DL fiziksel kanallarında TX çeşitleme modlarının uygulanması	49
Tablo 2.9: Hücreler arası geçiş ölçümleri	52
Tablo 3.1: Fiziksel sistemler ve kombinasyonel optimizasyon problemleri arasındaki benzerlik	78
Tablo 4.1: RMD Gösterimi	111
Tablo 5.1: SGA, D&D-GA ve ASAGA için farklı çaprazlama oranlarının etkisi... ..	126
Tablo 5.2: SGA, D&D-GA ve ASAGA için farklı mutasyon oranlarının etkisi	127
Tablo 5.3: Farklı seçim tekniklerinin tekrar tahsis edilen kullanıcı sayıları cinsinden sistem performansı üzerine etkisi	129
Tablo 5.4: Farklı çaprazlama tekniklerinin tekrar tahsis edilen kullanıcı sayıları cinsinden sistem performansı üzerine etkisi	131
Tablo 5.5: Farklı mutasyon tekniklerinin tekrar tahsis edilen kullanıcı sayıları cinsinden sistem performansı üzerine etkisi	133
Tablo 5.6: Farklı komşuluk hareketlerinin tekrar tahsis edilen kullanıcı sayıları cinsinden SA algoritması üzerine etkisi	135
Tablo 5.7: Soğutma çizelgesinin tekrar tahsis edilen kullanıcı sayıları cinsinden SA algoritması üzerine etkisi	136

SİMGELER VE KISALTMALAR

Simgeler

a_{yeni}	:Yeni kullanıcı kanal vektörü
b_k	:DL ışıma örüntüsü
$b_{k,i,q}^{(j)}(l)$	:j. sınıf kullanıcıya ait sıfır ortalamalı, istatistik olarak bağımsız ve birim varyanslı bilgi sembollerini içeren dizi
Bps	:Bir saniyede bit cinsinden iletilen veri hızı (Bit per second)
$c_{k,i,q}^{(j)}(l,m)$	: l . sembolünün çip yayılım kodu
Ch_{ini}	:Başlangıç kromozomu
D	:Komşu hücreler arası dik uzaklık
f	:Çoklu yol indeksi
$f(j)$	:Popülasyondaki j. kromozomun uygunluk değeri
F_c	:Çaprazlama operatörünün uygunluk değeri
F_k	:Mutasyon operatörünün uygunluk değeri
$g^{(\eta)}(l)$	:Çoklu yollardan kaynaklanan istenilir sinyale bileşenine ait iç parazit bileşeni
G	:Güç kuvvetlendirici
H	:Transpose, kompleks eşlenik alma
Hz	:Hertz
I	:Birim matris
$I_{k,i,q,f,\gamma}^{(j,\eta)}$	:İntegral terimi
m	:Anten sayısı
M	:Sıcaklık azalma adım aralığı
n	:Popülasyondaki kromozom sayısı
$n_0^{(\eta)}(l)$	:Aynı hücrede bulunan diğer kullanıcılardan kaynaklanan parazit sinyalleri
$n_k^{(\eta)}(l)$	:MS ile aynı frekans kanalını kullanan diğer hücrelerdeki kullanıcılardan gelen parazit sinyaller
$n_T^{(\eta)}(l)$	:Gürültü bileşeni
N_j	:j. sınıf kullanıcı sayısı
p_c	:Çaprazlama oranı
p_m	:Mutasyon oranı
P	:Popülasyon
P_{th}	:SIR eşik değeri
Q_j	:j. sınıf paralel kod kanalı
R	:Birim veri transfer hızı, anten yarıçapı
R_j	:j. sınıf veri transfer hızı
$s^{(\eta)}(l)$	:İstenilir sinyal bileşeni

$S(i)$	:Kromozomdaki i. kullanıcının veri transfer hızı
T	:Sıcaklık
T_b	:Başlangıç Sıcaklığı
T_c	:Çip periyodu
T_k	:Son Sıcaklık
T_w	:Sembol periyodu
TP	:Geçici popülasyon
U	:Kromozomdaki aktif kullanıcılara ait kodların sayısı
$\mathbf{w}_{k,i}^{(j)}$	:j. sınıftaki i. kullanıcı için DL ısıma örüntüsü
W	:Bantgenişliği
$\alpha_{k,f}^0$	:MS'e f. çoklu yol doğrultusunda gelen tüm sinyallerin etkisine maruz kaldığı zayıflama ve gölgeleme
β	:Kazanç Faktörü
$\{\beta_{k,f}\}$	:Kanal zayıflama parametreleri
Δ	:Anten elemanları arası mesafe
ΔC	:Maliyetler arası fark
$\Delta\theta$	:Açısal yayılım
λ	:Ortalama çağrı gelme oranı
$\{\theta_{k,f}\}$	:Kanal geliş açısı
$\{\tau_{k,f}\}$	:Kanal çoklu yol yayılım gecikmesi
$\psi_{k,i}^{(j)}$	:Bernoulli rasgele değişkeni
$(\cdot)^*$	:Karmaşık eşlenik alma işlemi
$\mathbf{v}_{\gamma}^{*(\eta)}$	:EGC katsayısı

Kısaltmalar

1G	:Birinci Nesil Haberleşme Sistemleri (First Generation Communications Systems)
2G	:İkinci Nesil Haberleşme Sistemleri (Second Generation Communications Systems)
3G	:Üçüncü Nesil Haberleşme Sistemleri (Third Generation Communications Systems)
3GPP	:3G Ortaklık Projeleri (Third Generation Partnership Project)
8-PSK	:8-Phase Shift Keying
AAA	:Uyarlamalı anten dizisi (Adaptive Antenna Array, AAA)
AICH	:Acquisition Gösterge Kanalı
AMPS	:Advanced Mobile Phone Service
ASAGA	:Uyarlamalı Benzetilmiş Tavlamalı Genetik Algoritma (Adaptive Simulated Annealing Genetic Algorithm)
ATM	:Asenkron Transfer Modu (Asynchronous Transfer Mode)
BCCH	:Tümegönderim Kontrol Kanalı (Broadcast Control Channel)
BCH	:Tümegönderim Kanalı (Broadcast Channel)
BER	:Bit Hata Oranı
BLER	:Blok Hata Oranı
BMC	:Tüme/Çoğa Gönderim Kontrol (Broadcast/Multicast Control)

BP	:Bloklanma oranı
BSC	:Baz İstasyonu Kontrol Ünitesi (Base Station Controller)
BTS	:Baz Alıcı Verici İstasyonu (Base Transceiver Station)
CCA	:Geleneksel Kod Tahsisi (Conventional Code Assignment)
CCCH	:Ortak Kontrol Kanalı (Common Control Channel)
CCPCH	:Birincil ve İkincil Ortak Kontrol Fiziksel Kanal
CDMA	:Kod Bölüşümlü Çoklu Erişim (Code Division Multiple Access)
CN	:Çekirdek Ağ (Core Network)
CPCH	:Ortak Paket Kanalı (Common Packet Channel)
CPICH	:Birincil ve İkincil Ortak Kontrol Pilot Kanal
CRC	:Döngüsel Yedekleme Kontrolü (Cyclic Redundancy Control)
CS	:Devre Anahtarlama (Circuit Switched)
DCA	:Dinamik Kod Tahsisi (Dynamic Code Assignment)
DCCH	:Adanmış Kontrol Kanalı (Dedicated Control Channel)
DCH	:Adanmış kanal (Dedicated Channel)
D&D-GA	:Dominance ve diploidy tabanlı Genetik Algoritma (Dominance and Diploidy Based Genetic Algorithm)
DL	:Aşağı Bağlantı (Downlink)
DOA	:Geliş açısı
DPCCH	:Adanmış Fiziksel Kontrol Kanalı (Dedicated Physical Control Channel)
DPCH	:Adanmış Fiziksel Kanal (Dedicated Physical Channel)
DPDCH	:Adanmış Fiziksel Data Kanalı (Dedicated Physical Data Channel)
DSCH	:DL Paylaşımlı Kanal (Downlink Shared Channel)
DTCH	:Adanmış Trafik Kontrol Kanalı (Dedicated Traffic Control Channel)
DTX	:Süreksiz İletim (Discontinuous Transmission)
EDGE	:GSM Gelişimi için Artırılmış Data Oranı (Enhanced Data Rate for GSM Evolution)
EGC	:Eşit kazançlı birleştirme (Equal Gain Combining)
ETSI	:Avrupa Haberleşme Standartları Enstitüsü (European Telecommunications Standards Institute)
FACH	:İleri Erişim Kanalı (Forward Access Channel)
FBI	:Geribildirim Bilgisi (Feedback Information)
FCCH	:Frekans Düzeltme Kanalı (Frequency Correction Channel)
FDD	:Frekans Bölüşümlü Çoğullama (Frequency Division Duplex)
GA	:Genetik Algoritma (Genetic Algorithm)
GGSN	:Geçiş GPRS Destek Birimi (Gateway GPRS Support Node)
GMSK	:Gaussian Minimum Shift Keying
GPRS	:Genel Paket Radyo Servisi (General Packet Radio Service)
GSM	:Global Systems for Mobile Communications
GSMA	:GSM Grubu (GSM Association)
HLR	:Ana Konum Kayıt Defteri (Home Location Register)
IMT-2000	:Uluslararası Mobil Haberleşme-2000 (International Mobile Telecommunications-2000)
ITU	:Uluslararası Haberleşme Birliği (International Telecommunication Union)
L&M	:Lundy & Mees
MAC	:Ortam Erişim Kontrol (Medium Access Control)
MAI	:Çoklu erişim girişimi (Multiple Access Interference)

MS	:Mobil kullanıcı (Mobile Station)
MSC	:Mobil Anahtarlama Merkezi (Mobile Switching Center)
NMT	:Nordic Mobile Telephony
Node-B	:3G baz istasyonu
OCCCH	:ODMA Ortak Kontrol Kanalı (ODMA Common Control Channel)
ODCCH	:ODMA Adanmış Kontrol Kanalı (ODMA Dedicated Control Channel)
ODTCH	:ODMA Adanmış Trafik Kanalı (ODMA Dedicated Traffic Channel)
OMC	:Çalışma ve Bakım Merkezi (Operation and Maintenance Center)
OVSF	:Birbirine Dik Değişken Yayılım Faktörü (Orthogonal Variable Spreading Factor)
PCCC	:Paralel Bağlı Katmanlı Kod
PCCH	:Bilgilendirme Kontrol Kanalı (Paging Control Channel)
PCH	:Bilgilendirme Kanalı (Paging Channel)
PCPCH	:Fiziksel Ortak Paket Kanalı (Physical Common Packet Channel)
PICH	:Sayfa Gösterge Kanalı (Paging Identify Channel)
PDCP	:Paket Data Yakınsama Protokolü (Packet Data Convergence Protocol)
PRACH	:Fiziksel Rasgele Erişim Kanalı (Physical Random Access Channel)
PS	:Paket Anahtarlama (Packet Switched)
PSC	:Birincil Senkronizasyon Kodu (Primary Synchronization Channel)
PSK	:Faz Kaydırmalı Anahtarlama (Phase Shift Keying)
QoS	:Servis kalitesi (Quality of Service)
RACH	:Rasgele Erişim Kanalı (Random Access Channel)
RF	:Radyo Frekans
RMD	:Kaynak yönetim veritabanı (Resource Manager Database)
RLC	:Radyo Link Kontrol (Radio Link Control)
RNC	:Radyo Ağ Kontrol Ünitesi (Radio Network Controller)
RRC	:Radyo Kaynak Kontrol (Radio Resource Control)
RSCP	:Alınan Sinyal Kodu Gücü
RSSI	:Alınan Sinyal Gücü Göstergesi
SA	:Benzetilmiş Tavlama (Simulated Annealing)
SCH	:Senkronizasyon Kanalı (Synchronization Channel)
SE	:Spektral verimlilik
SF	:Yayılım Faktörü (Spreading Factor)
SGA	:Basit Genetik Algoritma (Simple Genetic Algorithm)
SGSN	:Sunucu GPRS Destek Birimi (Serving GPRS Support Node)
SIM	:Subscriber Identity Module
SIR	:Sinyal Parazit Oranı (Signal to Interference Ratio)
STTD	:Uzay Zaman Blok Kodlama
SSC	:İkincil Senkronizasyon Kodu (Secondary Synchronization Channel)
TACS	:Total Access Communication Systems
TDD	:Zaman Bölüşümlü Çoğullama (Time Division Duplex)
TDMA	:Zaman Bölüşümlü Çoklu Erişim (Time Division Multiple Access)
TFCI	:İletim Format Kombinasyon Belirteci (Transport Format Combination Indicator)
TPC	:İletim Güç Kanalı (Transmit Power Channel)
TS	:Zaman Yuvası (Time Slot-TS)
TSTD	:Zaman Anahtarlama Verici Çeşitlemesi

UCA	:Düzgün Dairesel Anten Dizisi (Uniform Circular Array)
UE	:Kullanıcı Ekipmanı (User Equipment)
UL	:Yukarı Bağlantı (Uplink)
ULA	:Düzgün Doğrusal Anten Dizisi (Uniform Linear Array)
UMTS	:Ortak Mobil Haberleşme Sistemi (Universal Mobile Telecommunications System)
USTS	:UL Senkronizasyon İletim Şeması (Uplink Synchronization Transmission Channel)
UTRAN	:UMTS Karasal Radyo Erişim Ağı (UMTS Terrestrial Radio Access Network)
WCDMA	:Genişbant Kod Bölüşümlü Çoklu Erişim (Wideband Code Division Multiple Access)
VLR	:Ziyaretçi Konum Kayıt Defteri (Visitor Location Register)

İLERİ NESİL KABLOSUZ SİSTEMLERDE DİNAMİK KAYNAK TAHSİSİ VE KANAL KESTİRİMİ

Mustafa KARAKOÇ

Anahtar Kelimeler: Benzetilmiş Tavlama, Genetik Algoritma, Kaynak Yönetimi, OVFS, Uyarlamalı Anten Dizileri, Uzaysal Çeşitlilik, WCDMA

Özet: Kablosuz haberleşme sistemlerine olan talep her yıl artmaktadır. Kullanıcılar daha hızlı servislere ve gerçek zamanlı uygulamalara ihtiyaç duydukları servis kalitesinde talep gösterirler. Fakat tüm bu servislerin kullanımına olanak sağlayan mevcut bant genişliği gibi kaynaklar her zaman aynı artışı göstermezler. Mevcut bant genişliğine göre mümkün olan en fazla kullanıcıya istenilir kalite seviyesinde kaynak sağlamak çok önemlidir.

Haberleşme sistemlerinin gelişimi kullanıcıların taleplerine uygun bant genişliğini sağlayacak şekilde sürmektedir. Bu tez çalışmasının temelini oluşturan üçüncü nesil kablosuz haberleşme sistemleri, kullanıcıların taleplerine uygun bant genişliğini sağlamak ve yönetmek amacı ile birbirine dik değişken yayılım faktörü kodlarını kullanırlar. Bu kodların sayısı, sabit bantgenişliğinden dolayı sınırlı olduğundan en verimli bir şekilde kullanılması önem kazanmaktadır. Literatürde önerilen çeşitli yöntemler ile bu kodların kullanımı optimize edilmiştir. Önerilen tüm yöntemler kod sayısı sınırı altında sunulmuş ve kod sınırlı algoritmalar olarak tanımlanmıştır.

Bu tez çalışmasında kod sınırlı algoritmalar olarak literatürde optimizasyon amaçlı olarak kullanılan genetik algoritma, benzetilmiş tavlama ve onların birleşimi ile oluşan uyarlamalı benzetilmiş tavlama genetik algoritma gibi evrimsel hesaplama teknikleri ilk defa kullanılmış ve mevcut kod sınırlı algoritmalara göre performans artışı sağladığı görülmüştür. Ayrıca ilk defa akıllı anten teknolojilerine dayalı sinyal parazit oranı (SIR) sınırlı kod tahsis algoritmaları sunulmuş, böylece kod sayısı sınırı ortadan kaldırılmıştır. Kod sayısından bağımsız optimizasyon, sistem performansını büyük ölçüde artırmıştır. Önerilen algoritmalar ile kullanıcılara yönelik kod tahsisi ve tekrar tahsisi sağlanmış, performans artışı gözlenmiş ve kod ağacının optimum kullanımı gerçekleştirilmiştir.

DYNAMIC CODE ASSIGNMENT AND CHANNEL ESTIMATION FOR NEXT GENERATION WIRELESS SYSTEMS

Mustafa KARAKOÇ

Keywords: Adaptive Antenna Arrays, Genetic Algorithm, OVSF, Resource Management, Simulated Annealing, Spatial Diversity, WCDMA

Abstract: The demand for wireless communications is increasing exponentially every year. Subscribers demand on faster services and real time applications with requested quality of service. However the number of subscriber who prefers these mentioned services are also very important. But the available resources such as bandwidth, which facilitate to use of these services are not increasing at the same rate. It is very important that the radio resources should be used in an efficient way allowing the networks to support as many users as possible with satisfying quality of service requirement.

The evolution of communications systems continuous to provide suitable bandwidth to user demands. Third generation wireless communications systems which form the basis for this thesis. It uses orthogonal variable spreading factor (OVSF) codes to provide and manage the bandwidth according to subscribers' demand. Because of the limited bandwidth, the number of these codes is limited and efficient use of these codes is important. In the literature, there are several methods which optimize the use of these codes. All the proposed methods are limited under the number of codes and therefore referred as code limited algorithms.

In this thesis, evolutionary computation algorithms which are genetic algorithm, simulated annealing and combination of these algorithms known as adaptive simulated annealing based genetic algorithm are applied first time for the purpose of optimization of OVSF codes. It is observed that system performance is increased when compared to existing code limited algorithms. Furthermore, signal to interference ration (SIR) limited code assignment algorithm that is based on using an adaptive antenna array is proposed for the first time in the literature, so code number limitation is eliminated. The optimization which is independent from the number of code increases the system performance on a large scale. Code assignment and reassignment is provided, and performance increase is observed by optimum use of code tree with the proposed algorithm.

1. GİRİŞ

Kablosuz haberleşme sistemlerine olan ilgi her geçen gün katlanarak artsa da ihtiyaç duyulan servislerin sağlanması için gerekli olan bantgenişliği yani kaynak, daima aynı oranda sabit kalmaktadır. Analog haberleşme tekniğine dayanan ses haberleşmesi ile gelişimine başlayan kablosuz haberleşme sistemleri, daha sonra sayısal ses haberleşmesi ve sabit düşük hızlı veri transferi gibi servislere destek vermeye devam ederek gelişimini dosya transferi, garanti edilmiş servis kalitesinde çoklu ortam uygulamaları, yüksek hızlı video hizmetleri gibi servislerle sürdürmüştür (Molish, 2005). Burada saydığımız servisler bize göstermektedir ki zamanla son kullanıcıların tercihi sesten daha ziyade yüksek hızda ve istenen servis kalitesinde (quality of service, QoS) veri transferi olarak değişmiştir. Kablosuz haberleşme sistemlerinde bir servisin başarısı, o servisi kullanan kullanıcı sayısı ve her bir kullanıcının aldığı servisin QoS değeri ile doğru orantılıdır. Dolayısıyla artan kullanıcı sayısı ve daha hızlı servis gereksinimi nedeniyle bu alanda yapılan araştırmalar, mevcut bantgenişliğini en verimli şekilde kullanmaya yönelik yöntemlerin belirlenmesine odaklanmıştır.

1.1. Kablosuz Haberleşme Sistemlerinin Gelişimi

Kablosuz haberleşme sistemlerinin gelişimi, 1950’li yılların sonunda başlasa da ilk ticari sistemler, 1970’li yılların sonunda ve 1980’li yılların başında hayata geçmiştir. Bu bölümde kablosuz haberleşme sistemlerinin gelişimi ve hayatımıza etkileri üzerine kısa bir özet verilmektedir.

1.1.1. Birinci nesil haberleşme sistemleri

Hücreli teknolojilerin başladığı 1970’lerin sonundan itibaren kablosuz haberleşme sistemleri, gelişimini hem teknoloji hem de kullanım alanı yönünden sürekli olumlu

yönde arttırmıştır. Hücresel haberleşme sistemlerinin gelişim sürecindeki ilk ticari hücresel ağ, Tokyo’da 1979 yılında Japonya hizmetine sunulmuştur. Hemen ardından Avrupa’da Nordic Mobile Telephony (NMT), Amerika’da Advanced Mobile Phone Service (AMPS) ve İngiltere’de Total Access Communication Systems (TACS) tarafından hücresel teknolojiler hizmete sunulmuştur. Hizmete sunulan bu hücresel teknolojiler, “Birinci Nesil Haberleşme Sistemleri (First Generation Communications Systems, 1G)” olarak adlandırılmıştır (Nathan, 2003). 1G sistemler, analog haberleşme tekniğine dayanan ses servislerini sunan teknolojilerdir. Yukarıda tanımlanan tüm 1G sistemler, başlangıçta kendine öz standartlar ve parametreler kullandıklarından sistemler arası geçişe imkan tanınmamaktadır. Dolayısıyla serbest dolaşım (roaming) gibi geçişler söz konusu değildir. Nüfusa göre abone sayısı (penetrasyon) da oldukça düşüktür. Bu değerler, İsveç gibi mobil teknolojileri sıkı takip eden, mobil teknolojilerin standarda kavuşmasında aktif rol alan ülkelerde, % 5-7 seviyelerinde iken Portekiz gibi gelişmiş teknolojiyi son kullanıcısının hizmetine sunan ülkelerde ise % 0,5-1 seviyelerindedir.

1.1.2. İkinci nesil haberleşme sistemleri

1G sistemleri sunan hücresel teknolojilerin farklı standartlara sahip olması sebebiyle, aynı yerde farklı hücresel haberleşme sistemleri arasında ortaya çıkan uyumsuzluk, ortak bir altyapısı olan, standartlaşmış bir sistemin oluşturulmasını zorunlu kılmıştır. 1990’ların başında sayısal haberleşme teknolojisi hız kazanmış, beraberinde o dönem için yeni nesil teknolojiler olarak adlandırılan ve “İkinci Nesil Haberleşme Sistemleri (Second Generation Communications Systems, 2G)” olarak bilinen bir teknolojiyi ortaya çıkarmıştır. Sayısal terimi, konuşmacının sesinin örneklenip filtrelenerek işlenmesi ile insan kulağında aynı etkiyi oluşturacak bir etki yaratması olarak tanımlanabilir (Steele ve diğ., 2001). Bu işlem, birçok kullanıcının aynı anda mevcut haberleşme sistemini kullanmasına olanak tanımaktadır. Global Systems for Mobile Communications (GSM), TDMA IS-136, CDMA IS-135 gibi hücresel teknolojiler öne çıkan 2G sistemleri olarak bilinmektedir.

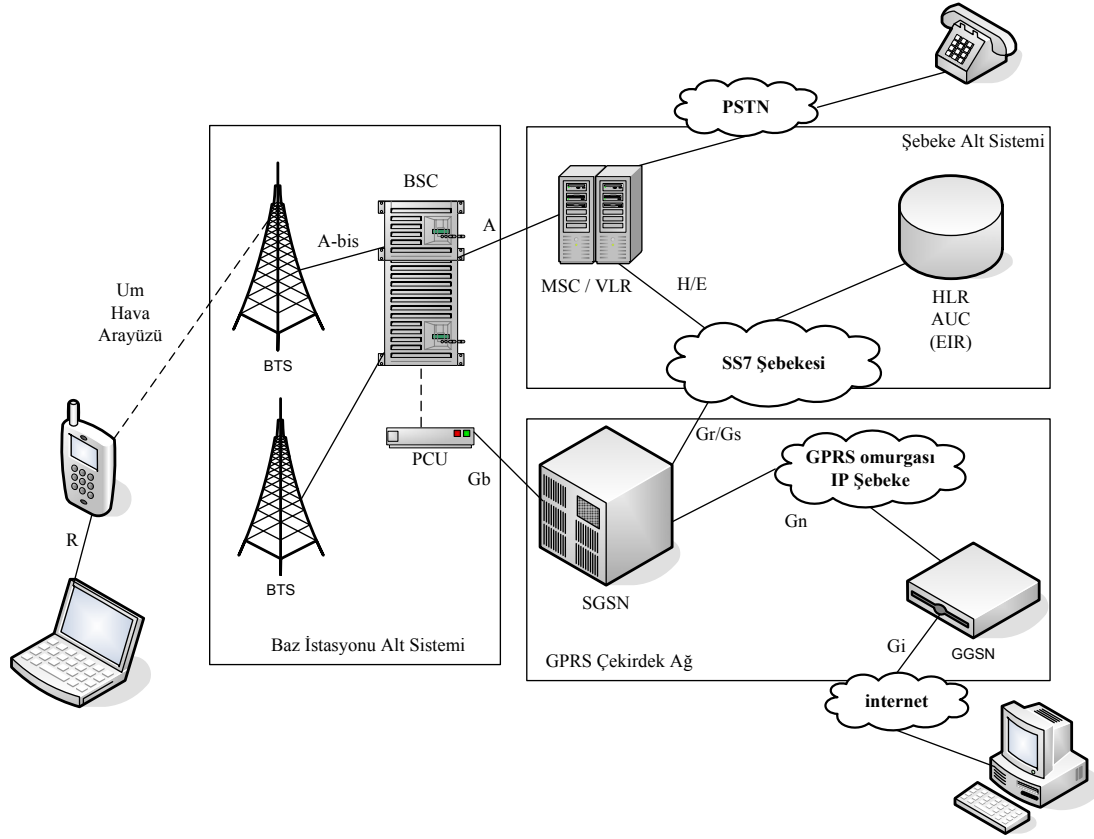
GSM, günümüzde en çok kullanılan kablosuz haberleşme teknolojisi olarak bilinmektedir. İlk olarak 1980 yılında geliştirilmiştir. 1982'den 1985'e kadar GSM grubu (GSM Association, GSMA) yapının analog bir sistem mi yoksa sayısal bir sistem mi olacağını tanımlamaya çalışmıştır. Yapılan farklı testlerden sonra yapının sayısal bir sistem olmasına ve darbant zaman bölmeli çoklu erişim (Time Division Multiple Access, TDMA) standardı üzerine inşa edilmesine karar verilmiştir. Modülasyon tekniği olarak Gaussian Minimum Shift Keying (GMSK) seçilmiştir. GSM sisteminin teknik özellikleri, 1987 yılında tamamlanmış ve 1991 yılı itibariyle ilk ticari sayısal hücresel sistem Finlandiya'da kurulmuştur (Godara, 2002). GSM, şu ana kadar dünyada en popüler ve en geniş kullanıcı kitlesine sahip kablosuz haberleşme teknolojisidir. 2007 sonu itibariyle kablosuz haberleşme kullanıcılarının sayısı, 3 milyar kişi civarındadır ve bunun büyük bir bölümü GSM kullanıcısıdır (Global Mobile Market, 2007). Zamanla sistem, kullanıcıların talepleri doğrultusunda şekillenmiş, sadece sesli aramanın yanı sıra SMS, düşük hızlı veri transferi, roaming, ön ödemeli servisler gibi farklı kavramları da kullanılır hale getirmişlerdir. 2G hücresel sistemler ailesinde yer alan GSM'in 1G sistemlere göre sahip olduğu en önemli avantaj, daha yüksek ses kalitesi ile daha ucuz aramaları gerçekleştirme imkanı sağlamasıdır. Operatörler için sunduğu en büyük avantaj ise standart yapıya sahip bir sistem olduğundan farklı üreticilere ait ürünlerin bir arada kullanılabilmesine imkan sağlamasıdır.

GSM sistemi, 850 MHz, 900 MHz, 1800 MHz ve 1900 MHz frekanslarında çalışsa da, operatörler en çok 900 MHz ve 1800 MHz için bu teknolojiyi tercih etmektedirler. 900 MHz için yukarı bağlantı (Uplink, UL) frekans bandı 890-915 MHz, aşağı bağlantı (Downlink, DL) frekans bandı 935-960 MHz aralığındadır. Bu da hem UL hem de DL yönünde 25 MHz'lik bir frekans bandında her biri 200 KHz olan 125'er adet kanal bulunduğunu göstermektedir.

Genel GSM sistem mimarisi, 2 ana bileşenden oluşmaktadır. Bunlar; sistemin sabit ağ yapısı ve bu ağ'dan faydalanan abonelerdir. Şekil 1.1'de de görüleceği üzere sistemin sabit ağ yapısı da kendi arasında 3 alt sistemden oluşmaktadır. Bunlar;

- **Radyo Ağı-Baz İstasyonu Alt Sistemi (Radio Network-Base Station Subsystem):** Bu alt sistem, baz alıcı verici istasyonu (Base Transceiver Station, BTS) ve baz istasyonu kontrol ünitesi (Base Station Controller, BSC) yapılarından oluşmaktadır. BTS, hücre merkezine konumlandırılır ve hücredeki abonelerin haberleşmelerini sürdürmeleri amacıyla sinyalleşme ve veri trafiği için radyo kanallarını organize eder. BSC ise belli sayıdaki BTS'in bağlı olduğu yapıdır. BTS'lerin kontrolü, alarm yönetimi, frekans paylaşımı ve senkronizasyon burada sağlanır.
- **Anahtarlama ve Yönetim Alt Sistemi (Switching and Management Subsystem):** Bu alt sistem, mobil anahtarlama merkezi (Mobile Switching Center, MSC) ve veritabanlarından oluşur. Anahtarlama, yönlendirme, veri senkronizasyonu, üyelik, adres bilgileri, kullanıcıya özgü servisler ve servis sağlama için ihtiyaç duyulan bilginin saklandığı veritabanları, bu alt sistemin birer üyesidir. Birden fazla BSC, bir MSC tarafından kontrol edilir. Sabit ve mobil ağlar arasındaki trafik geçişi buradan sağlanır. Abone ile ilgili kimlik tanımlamaları, kullandıkları servislere dair bilgilerin kontrolü buradaki veritabanlarında tutulur (Home Location Register, HLR). Ayrıca abonenin mevcut konum bilgisi, servis aldığı MSC bilgileri de bu alt sistemin görevidir (Visitor Location Register, VLR).
- **Çalışma ve Yönetim Alt Sistemi (Operation and Management Subsystem):** GSM sisteminin çalışması, bu alt sistemin kontrolü altında gerçekleşir. Çalışma ve Bakım Merkezi (Operation and Maintenance Center, OMC) tarafından sisteme ait kontrol fonksiyonları tanımlanır ve yönetilir. OMC'den hem BSC'ye hem de MSC'ye erişim vardır. Abonelik işlemleri, ücretlendirme ve istatistikler burada tutulur. Güvenlik ve performans yönetimi de bu alt sistem tarafından yürütülür.

GSM sistemleri, daha önce de bahsettiğimiz gibi zamanla kullanıcıların talepleri doğrultusunda şekillenmiştir. Burada en büyük çıkışı yapan talep her ne kadar SMS olsa da veri transferi, kablosuz haberleşme sistemlerinin yol haritasını belirleyecek en önemli etken olarak kabul edilmektedir. Bu bağlamda GSM sistemi, Genel Paket Radyo Servisi (General Packet Radio Service, GPRS) ve GSM Gelişimi için Artırılmış Data Oranı (Enhanced Data Rate for GSM Evolution, EDGE) gibi iki kavramı hayata geçirmiş ve veri transferini etkin kılmıştır (Prasad ve diğ., 2003).



Şekil 1.1: GSM ve GPRS sistemi

GPRS, ses dışında veri transferi için kullanılan katma değerli bir servistir. Bu servis, mevcut devre anahtarlamalı GSM hava ara yüzünün paket anahtarlamalı yapı kullanılarak genişletilmesi ile mümkün kılınmıştır. Bazı ekipmanların sisteme dahil edilmesi ve ses tabanlı servis yazılımlarının ses ve veriyi bir arada destekleyecek şekildeki yazılımlara yükseltilmesi ile veri transferi operatörler tarafından da desteklenmiştir. Ancak yine de ses trafiği devre anahtarlamalı yapıda, veri trafiği paket anahtarlamalı yapıdadır. Paket anahtarlamalı yapı, kullanıcı sadece veri transferi yapacağı zaman aktif olur. Transfer edilecek veri büyüklüğü, sistemde aynı servisi alan kullanıcı sayısına bağlı olarak değişmektedir. Aynı anda sekiz zaman yuvası da (time slot, *TS*) veri iletimine ayrıldığında GPRS ile teorik olarak ulaşılacak maksimum veri transfer hızı 171,2 Kbps olarak verilmektedir. Bu transfer hızı ile düşük de olsa internet erişimi sağlanmış olur. Dolayısıyla mobil terminal internet sunucusu gibi davranır. Düşük hız GPRS için tanımlanan en büyük sınırlayıcı etkidir (Eberspacher ve diğ., 2001).

GPRS servisinin sahip olduđu sınırlamalar, EDGE teknolojisinin tanımlanması ile bir ölçüde azaltılmıştır. EDGE, TDMA tabanlı bir teknolojidir. GPRS servisinin uygulanabildiği her teknoloji EDGE için de uygulanabilir. EDGE, 3G'nin bir alternatifi değildir. Ancak geçiş süresince tamamlayıcı etkisi çok büyük olmuştur. Bu yüzden EDGE sistemi 2,5G olarak tanımlanmaktadır. EDGE'de teorik olarak 500 Kbps'a yakın değerlerde veri transfer hızı gerçekleştirilir. Bu transfer hızı, 8-PSK (Phase Shift Keying) olarak tanımlanan modülasyon tekniği ile sağlanır ve GPRS'nin desteklediği GMSK modülasyon tekniği ile beraber kullanılır. Veri transferi, EDGE teknolojisinde her *TS* için teorik olarak 8,8-59,2 Kbps/*TS* arasında değişen hıza sahiptir ve bu hız arttıkça GMSK'den 8-PSK'ye doğru bir geçiş sağlanmış olur. Ancak yüksek hızlı uygulamaların artmasından dolayı veri transfer hızının istenilir QoS değerinde sağlanması gerekliliği ortaya çıkmıştır. Bu da 3G sistemlerin gelişimini tetikleyen önemli unsurlardan biridir. 3G sistemlerin gelişimi etkileyen en önemli sınırlayıcı etkenler şunlardır:

- Sınırlı veri transfer hızı: 2G sistemler temel olarak abonelerine ses servisi sunmak amacı ile tasarlanmıştır. Bu nedenle 2G sistemlerin sağlayabileceği veri transfer hızı düşüktür. Bu, 2G sistemler için en önemli sınırlayıcı etkindir.
- Paket anahtarlama servislerin düşük verimliliği: Evde ya da ofiste sabit internet erişimini kullanmanın yanı sıra mobil olarak da internet erişimi gereği sürekli artmaktadır. Ancak kablosuz internet uygulamalarının 2G sistemler tarafından sağlanması verimli bir çözüm değildir.

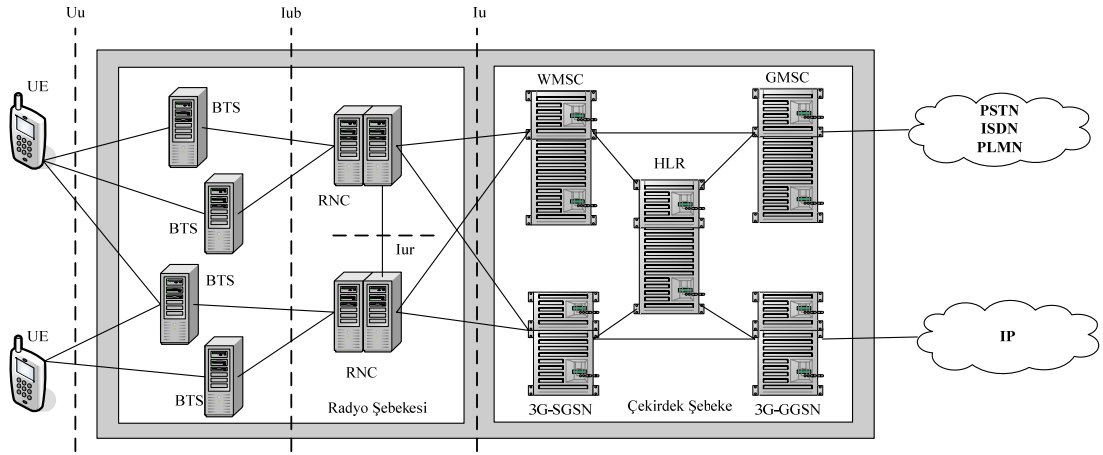
1.1.3. Üçüncü nesil haberleşme sistemleri

“Üçüncü Nesil Haberleşme Sistemleri (Third Generation Communications Systems, 3G)” aboneye yüksek hızlı veri transferi ve çoklu ortam uygulamaları sunmak amacıyla geliştirilmiş bir teknolojidir. Uluslararası Haberleşme Birliği (International Telecommunication Union, ITU) tarafından belirlenen Uluslararası Gezgin Haberleşme (International Mobile Telecommunications, IMT-2000) standartları ile tanımlanan 3G sistemler, 144 Kbps'den 2 Mbps'ye kadar yüksek hızlı veri transferi gerçekleştirme yeteneğine sahiptir. Kod Bölüşümlü Çoklu Erişim (Code Division

Multiple Access, CDMA) ve Ortak Mobil Haberleşme Sistemi (Universal Mobile Telecommunications System, UMTS) IMT-2000 standartlarına uygun 3G sistemlerdir (Walke ve diğ., 2003).

CDMA 2000 sistemi 1X, 1XEV-DO, 1XEV-DV ve 3X gibi çeşitli biçimlere sahiptir. 1XEV özelliği, 3GPP tarafından geliştirilmiştir ve yüksek hızlı paket veri hava ara yüzü olarak da bilinmektedir. Bu teknoloji ile birlikte çok düşük bir bant genişliği işgal edilerek (1,25 MHz/taşıyıcı) 140 Kbps veri transfer hızlarında 3G servisler verilebilir. Aynı zamanda 1XEV faz-1 olarak da adlandırılan 1XEV-DO, 2,4 Mbps veri transfer hızlarında servis sunmak için ses ve veriyi farklı farklı taşıyıcı kanallarda tutmaktadır. EV-DV ya da diğer adıyla faz-2, 3-5 Mbps arasında veri transfer hızlarını destekleyebilmektedir. Bununla birlikte CDMA 2000 3X, ITU tarafından onaylanan bir 3G (IMT-2000) standardıdır ve çoklu kullanıcı uygulamalarını desteklemektedir. Bu teknoloji ayrıca 5MHz'lik bir bant genişliğini ve 2-6 Mbps arasında veri transfer hızlarını da destekleyebilmektedirler (Zigangirov, 2004).

UMTS ise IMT-2000 standartlarına uygun bir diğer 3G teknolojisidir ve temelinde Genişbant Kod Bölüşümlü Çoklu Erişim (Wideband Code Division Multiple Access, WCDMA) metodu vardır. WCDMA, NTT DoCoMo operatörü tarafından geliştirilen bir çoklu erişim tekniğidir. Bu teknik, 3G standartlarını belirlemek amacı ile IMT-2000 ana başlığı altında ITU'de de kullanılmıştır. WCDMA, Frekans Bölüşümlü Çoğullama (Frequency Division Duplex, FDD) ve Zaman Bölüşümlü Çoğullama (Time Division Duplex, TDD) gibi temel modları destekleyebilme, farklı veri iletim hızları oranlarını sağlayabilme, asenkron girişim engelleme, uyarlanabilir güç kontrolü, kapsama ve kapasite hassasiyeti gibi bazı önemli özelliklere sahiptir. Öte yandan UMTS, biri 1900 MHz frekansında UL için, diğeri 2100 MHz frekansında DL için 5 MHz genişliğinde bantgenişliği işgal etmektedir.



Şekil 1.2: UMTS sistem mimarisi

UMTS sistem mimarisi, birbiri ile etkileşimli üç alt sistemden oluşur. Bunlar;

- Çekirdek Ağ (Core Network, CN),
- UMTS Karasal Radyo Erişim Ağı (UMTS Terrestrial Radio Access Network, UTRAN)
- Kullanıcı Ekipmanı (User Equipment, UE).

CN'in temel fonksiyonu kullanıcı trafiğinin yönlendirilmesi, iletimi ve anahtarlama olarak tanımlanabilir. CN aynı zamanda veritabanı ve ağ yönetim fonksiyonlarını üstlenmiştir. UMTS için CN mimarisinin temeli GPRS-GSM CN mimarisine dayanır. Sadece yazılımsal farklılıklar vardır. CN, devre anahtarlama (Circuit Switched, CS) ve paket anahtarlama (Packet Switched, PS) olmak üzere iki moda sahiptir. CS ekipmanları, MSC, VLR ve geçiş MSC birimlerinden oluşurken PS ekipmanları sunucu GPRS destek birimi (Serving GPRS Support Node, SGSN) ve geçiş GPRS destek birimi (Gateway GPRS Support Node, GGSN) ünitelerinden oluşur. Asenkron transfer modu (Asynchronous Transfer Mode, ATM) UMTS CN tanımlı bir ekipmandır. RAN bileşenleri ise, BTS ya da diğer adıyla Node-B ve Radyo Ağ Kontrol ünitelerinden oluşur (Radio Network Controllers, RNCs). BTS'in ana fonksiyonu kapalı çevrim güç kontrolü, fiziksel kanal kodlama, modülasyon/demodülasyon, hava ara yüz iletimi/alımı ve hata kontrolü olarak verilir. RNC'nin ana fonksiyonları ise radyo kaynak kontrol ve yönetimi, güç kontrolü, kanal tahsis, giriş kabul kontrolü ve şifreleme olarak belirlenmiştir. UE, mobil

terminal ve 3G servis desteęi saęlayabilecek bir SIM (Subscriber Identity Module) karttan oluşur. Tipik bir 3G mimarisi Şekil 1.2’de verilmiştir. (Holma ve Toksala, 2006).

1.2. Çalışmanın Amacı

Mevcut 2G sistemler tarafından desteklenen servisler, ses ve sabit-düşük hızlı veri transferi ile sınırlıdır. Dosya transferi ya da garanti edilmiş servis kalitesinde çoklu ortam uygulamaları, 3G sistemler tarafından sağlanmaktadır (Dahlman ve dię., 1998). Temel ses iletişiminden başlayarak, mesajlaşma, dosya transferi ve çoklu ortam uygulamaları için yüksek hızlı veri transferine kadar farklı talepleri karşılayan 3G sistemler, farklı veri transfer hızı ihtiyacı bulunan kullanıcıların taleplerini karşılamak ve oluşturdukları trafik ile baş edebilmek için, kullanıcılara ihtiyaç duyduğu veri transfer hızını sağlamaktadır. Birbirine dik deęişken yayılım faktörü (Orthogonal Variable Spreading Factor, OVSF) kodları, 3G sistemlerde farklı veri transfer hızı talebini karşılamak amacıyla ihtiyaç duyulan farklı bant genişliklerinin sağlanması için kullanılır. Her BTS, DL iletimi için OVSF kod ağacını yönetir. OVSF kod ağacının sınırlı kapasitede olmasından dolayı, bu kaynağın verimli bir şekilde kullanılması ve yönetilmesi önemlidir.

Bu çalışmanın amacı, OVSF kod ağacında yeteri kadar kod olmasına rağmen daęınık kodlamadan dolayı sisteme dahil edilemeyen kullanıcılar için ağacın optimize edilerek bu kullanıcıların taleplerinin karşılanmasını sağlamaktır. Bununla birlikte BTS’de uyarlamalı anten dizileri (Adaptive Antenna Arrays, AAAs) kullanılmasıyla mobil kullanıcının uzaysal çeşitlilik bilgisinden faydalanarak aynı kodun standartlarca belirlenen bazı sınırlamalar altında birden fazla kullanıcıya tahsis edilmesi hipotezi de tez süresince geliştirilerek burada sunulmuştur. Böylece kod tahsisi için boş (tahsis edilmemiş) kod gereksinimi ortadan kalkmış olur. Yapılan bu tez çalışmasının bilime sağladığı katkı ve yenilikler aşağıda maddeler halinde sunulmaktadır:

- OVFS kod ağacının dinamik olarak optimizasyonunda evrimsel algoritmalar kullanılmıştır. Bunlar; Genetik Algoritma (Genetic Algorithm, GA), Benzetilmiş Tavlama (Simulated Annealing, SA) ve Uyarlamalı Benzetilmiş Tavlama Genetik Algoritma (Adaptive Simulated Annealing Based Genetic Algorithm, ASAGA)'dır. Dinamik optimizasyonun gerçekleştirildiği bu yapıda, OVFS kod ağacındaki boş kodları bir araya getirmek üzere sisteme özel bir maliyet fonksiyonu tanımlanmış ve bu fonksiyon her iterasyonda yeni değerler alarak onun minimize edilmesi üzerine odaklanmıştır. Şekil 1.3'te görüldüğü gibi OVFS kod ağacı, ilk defa bu tez çalışmasında matris formunda kullanılmıştır. Ağaçtaki boş ve dolu kodların miktarı bilgisinden yararlanılarak maliyet fonksiyonu hesabı matematiksel olarak ifade edilmiştir.

Şekil 1.3: Kod ağacı ve onun matrissel ifadesi

standartlarca belirlenen seviyelerde daha güçlü olmasıdır (3GPP2 C.S0024-B, 2006). Böylelikle OVFS kod atama stratejisi, bu tez çalışmasında ilk defa SIR sınırlı yapı halinde ele alınarak sunulmuştur.

1.3. Literatürde Yapılan Çalışmaların Özetleri

Artan veri transfer hızı talebinin neden olabileceği kod bloklanma problemi ile baş edebilmek ve buna bağlı olarak çağrının bloklanmasının önüne geçebilmek için araştırmacılar, OVFS kod ağacının dinamik kullanımı üzerine yoğunlaşmışlardır. Bu konuda yapılan bazı çalışmalar aşağıda özetlenmiştir.

Chao ve diğ. (2005), yapmış oldukları çalışmalarında kod talebinde bulunan kullanıcıya birden fazla OVFS kod ataması yaparak verimli bir kod tahsisi yapmayı hedeflemişlerdir. Burada kullanıcının talebini karşılamak üzere atanan OVFS kodu, talebin üzerinde ise burada arta kalan kod miktarından kaynaklanan verimsiz atamanın önüne geçmek için birden fazla kod kullanıcıya atanmış ve bant genişliğinin verimli kullanımı ile donanım karmaşıklığı arasında bir maliyet hesabı yapılmıştır.

Tseng ve Chao (2002), önerdikleri kalabalık olanın ilk kullanımı (crowded first) ve kod atamasına OVFS kod ağacının solundan başlama (leftmost) algoritmalarını kod tahsis ve tekrar tahsis amacı ile denemişlerdir. %80 trafik yüklü ve SF=256 olan kod ağacında rasgele atama (random assign) adı verilen teknik ile bu algoritmaları karşılaştırarak kod bloklanmasında % 77, tekrar tahsis edilen kullanıcı sayısında % 81'lik bir iyileşme sağlamışlardır.

OVFS kod ağacının döndürülerek kullanımı Chen ve Chang (2004) tarafından denenmiş, bloklanma olasılığında ve kod tekrar tahsis maliyetinde düşüşler gözlenmiştir. Döndürülmüş OVFS kod ağacında her dala ait iki torun kod, komşu dala ait torun kodlar ile yer değiştirmektedir. Chen ayrıca Lin ile yapmış olduğu çalışmada (2006), döndürülmüş OVFS kod ağacını doğrusal bölümlendirme ve doğrusal olmayan bölümlendirme gibi iki farklı yöntemde değerlendirerek

performans artışı gözlenmiştir. Doğrusal bölümlendirmede OVFS kod ağacındaki tüm kodlar döndürülmüş, doğrusal olmayan bölümlendirmede ise tüm kodlar gelen kullanıcı veri transfer oranı doğrultusunda parça parça döndürmüşlerdir.

Minn ve Siu'nun OVFS kodlarının dinamik olarak yer değiştirmesini içeren çalışmalarında (2000), yeniden tahsis esnasında minimum yer değiştirme esas alınarak kod ataması yapılmaktadır. Bu çalışma, literatürde OVFS kod yeniden tahsisi için referans alınan temel bir eserdir.

OVFS kod ağacında yeni kullanıcının talebi atanırken ilk olarak en sağdan atama stratejisinin kullanımı Gong ve diğ. (2005) tarafından önerilmiştir. Çalışma sonuçlarında daha az tekrar tahsis edilen kullanıcı sayısına erişildiği belirtilmiştir.

Assarut ve diğ. (2001), yapmış oldukları çalışmada OVFS kod ağacını önceden alt ağaçlara bölmüşler ve her hız için ona karşılık gelen alt ağacın kullanımını önermişlerdir. Ayrıca kullanıcı tarafından talep edilen veri hızlarının istatistiksel dağılımına bakarak alt ağaçların büyüklüklerini değiştirmişlerdir. Çok kullanılan kod için rezerve edilen ağaç dalları sayısı daha fazla, az kullanılan kod içinse daha azdır. Bu yöntemle bloklanma olasılığının azaldığını belirtmişlerdir.

Minn ve Siu'nun önerdiği dinamik kod tahsis algoritması ile Assarut ve diğ.'nin önerdiği önceden rezerve edilmiş kod alt ağaçları Sekine ve diğ. (2003) yapmış oldukları Hibrit OVFS kod tahsis algoritması çalışmasında birleştirilmiştir. Burada alt ağaçlar, her veri transfer hızı için eşit olarak bölünmüştür. Eğer ilgili alt ağaçta boş kod varsa atama yapılır, yoksa dinamik olarak kod tahsis algoritması çalıştırılır.

Perez ve diğ. (2006) yapmış oldukları çalışmada OVFS kod atama stratejisi, çağrının gelme aşaması ve kod tahsis aşaması olarak incelenmiş, her aşama için çoklu kodların kullanımı felsefesine dayanan algoritmaların çalıştırılması ile kod atama işlemi tamamlanmıştır. Önerilen bu çalışma ile daha az bloklanma oranı ve daha hızlı bir kod atama sürecinin elde edildiği belirtilmiştir.

Bir diğ er alternatif algoritma da Rouskas ve Skoutas (2002/1) tarafından  onerilen aŗađı bađlantı i in WCDMA OVSF sistemlerde kod tahsis/tekrar tahsis algoritmasıdır ve bu  alıŗma Tseng ve Chao'nun (2002) kullandıkları kalabalık olanın ilk kullanımı (crowded first) algoritmasına benzer niteliktedir.

Peng ve diđ. (2005) yapmıŗ oldukları  alıŗma, Assarut ve diđ.'nin (2001)  alıŗmasına benzer niteliktedir. Burada alt ađa  oluŗumunda kullanılan algoritma ile daha iyi performans sergilendiđi ifade edilmiŗtir.

Tsai ve Lin (2005),  alıŗmalarında kullandıkları    farklı algoritma ile istenilir performans artıŗını elde ettiklerini belirtmiŗlerdir. Bu  alıŗmada  onerilen algoritmalar seyrek dalın ilk kullanımı (sparse first), seyrek olan dallardan atamaya sađ dal ile baŗlanması (sparse first/rightmost) ve geliŗtirilmiŗ seyrek olan dallardan atamaya sađ dal ile baŗlanması (modified sparse first/rightmost) olarak verilmiŗtir. Genel olarak seyrek dalın  nce kullanılması prensibine dayanan bu  alıŗmada kod ađacı her kod tahsisi sonrasında tekrar incelenerek seyrek kodların listesi tutulmaktadır. Geliŗtirilmiŗ algoritmada ise bu belirtilen kriter dıŗında veri transfer hızının istatistik bilgileri de maliyet fonksiyonunda kullanılmıŗtır.

Tseng ve diđ. (2001), yapmıŗ oldukları  alıŗmalarda OVFS kod tahsisi ve tekrar tahsisi konusunda yapılan ilk  alıŗmaya imza atmıŗlardır. Her kod tahsisinden sonra OVFS kod ađacının optimizasyonu ile kod bloklanmasında azalma ve yer deđiŗtirme maliyetinde d ŗuŗ g zlenmiŗtir. Bu  alıŗmada rasgele kod tahsisi, OVFS kod ađacının solundan baŗlama (leftmost) ve kalabalık olanın ilk kullanımı (crowded first) algoritmaları denenerek performans artıŗı g zlenmiŗtir.

Chen ve Hwang'ın (2006)  alıŗmaları, OVFS kod tahsis literat r ne kalabalık gruba ilk atanma (crowded-group first strategy) ve kalabalık dala ilk atama (crowded-branch first strategy) isimli algoritmaları kazandırmıŗtır.  onerilen bu algoritmalar, OVFS kod ađacının y netiminde, kod tahsisinde ve kodun tekrar tahsisinde kullanılmıŗ ve hem s re hem de uygun kodun bulunması a ısından olduk a verimli bir maliyet fonksiyonu sergilemiŗtir. Ayrıca  nerdikleri yeni performans kriterleri ile

algoritmaların sağladıkları faydaları belirtmişlerdir. Bu kriterler, bloklanan kodun ağırlığı ve tekrar tahsis edilen kodun olasılığı cinsinden verilmektedir.

Rouskas ve Skoutas (2002/2), daha önce önerdikleri ve gelen yeni çağrıya göre minimum yer değiştirmeye neden olacak olası değişimleri hesaba katan çalışmalarına ek olarak bu çalışmalarında (2005) tekrarlamalı olarak istatistiksel veri kullanıp gelen çağrıyı, olası minimum yer değiştirmeye neden olacak bir şekilde sunmuşlardır. (2002/1)'de yeni kullanıcı için atanacak olan kod, ileride bloğa neden olma olasılığı olan ancak atama maliyeti minimum olan kodun atanmasını ele almaktadır. (2005)'te ise durum kod atama ve kodun yer değiştirmesi işlemlerinden sonra da tekrarlamalı olarak devam etmektedir.

Hou ve Lee (2004), çalışmalarında OVFSF kod ağacının Rayleigh zayıflaması ve çoklu erişim girişimi (multiple access interference, MAI) gibi parametrelere bağlı olarak uyarlamalı olarak değiştirmişlerdir. Burada sisteme ekledikleri OVFSF kontrol birimi ve ortam karakteristiğinden dolayı değişen sinyal seviyesi problemleri ile baş edebilmek için OVFSF kod ağacının boyutu literatür tarafından belirlenen sınırlar çerçevesince (Shueh ve Chen, 2003) değiştirilmiştir. Sistemin karmaşıklığı oldukça yüksek ve gerçek zamanlı uygulamalar için biraz yavaş olsa da ortamın dinamiklerini göze alarak ağacı değiştirmeleri sistem performansını artırmıştır.

Shueh ve Chen (2003), yapmış oldukları çalışmada OVFSF kod tekrar tahsisinin sistem performansını %20 artırmaya karşın sisteme getirdiği hesaplama yükünün %60 arttığına dikkat çekmiştir. Burada, sağlanan faydanın sisteme getirdiği yük ile dengeli olmasını sağlayacak bir algoritma kullanılarak tekrar tahsis hesaplama yükü çok fazla olacak kodlar, direkt bloklanmaktadır.

Assarut ve diğ. (2002), çalışmalarında geleneksel, dinamik ve alan bölmeli kod tahsis algoritmaları arasındaki karşılaştırmayı incelemiş ve alan bölmeli algoritmanın daha iyi performans sergilediğini belirtmişlerdir. Performans kriteri olarak hesaplama süresi ve bloklanma oranı ele alınmıştır.

Yu ve diğ. (2005), önermiş oldukları çalışmada OVFS kod yapısı kullanılarak çoklu yol girişim etkisini minimize etmişlerdir. Bilinen kullanıcılara ait kanal özellikleri kullanılarak bilinmeyen kullanıcıların kanalları, OVFS kod yapısına ve kurallarına uygun şekilde modellenmiştir. Ortamdaki anlık değişimler başlangıçta bilinen kullanıcılara göre belirlense de daha sonra bu değişim bir modele oturtulmuştur. Böylece girişime neden olan kullanıcıların, ortak kod kanallarına ve ortak kod kanallarının da kullanıcıya etkileri en aza indirgenmiştir.

Dell'Amico ve diğ. (2004), OVFS kod ağacının verimli kullanımı için önerdikleri farklı kod ağacı bölme algoritmalarında, minimum maliyetli kod ağacının seçilmesi ve dinamik kod ağacı bölme metotları üzerinde çalışmışlardır. Bu algoritmaların geleneksel ve dinamik OVFS kod tahsis algoritmalarına göre daha az bloklanma olasılığı oluşturduğu ve spektral verimliliği artırdığı gözlenmiştir. Ayrıca önerilen bu algoritmaların hesaplama karmaşıklığının dinamik kod tahsis metotlarına göre daha az olduğu ifade edilmiştir.

Chen ve Hwang (2004), önerdikleri dinamik kod tahsis metodunda yeni gelen kullanıcı için tahsis edilecek OVFS kanallama kodunun, kod ağacındaki olası en kalabalık daldan atanması yöntemini ele almışlardır. Burada ilgili veri transfer hızı ile aynı hıza sahip OVFS kod ağacının ilgili katmanında bir arama yapılarak en kalabalık olan dal seçilmekte ve boş uygun kod varsa bu dala kod tahsis talebinde bulunan kullanıcının OVFS kanallama kodu atanmaktadır.

Ferng ve diğ. (2005), önermiş oldukları çalışmada OVFS kod ağacını bölerek yüksek veri transfer hızı talebinde bulunan kullanıcı ile düşük veri transfer hızı talebinde bulunan kullanıcıyı aynı ağaçta farklı yerlere atamışlardır. Bununla birlikte yüksek veri transfer hızı talebinde bulunan kullanıcının bloklanma olasılığını en aza indirmek için belli bir eşik değer üstünde veri transfer hızına sahip kullanıcıların kodlarını basit bölme algoritmalarına tabi tutmuş performans artışını gözlemlemişlerdir.

Jungnickel ve diğ. (2004), önerdikleri çalışmada çoklu giriş çoklu çıkış (multiple input multiple output, MIMO) kanalları için her kanala özgü olacak şekilde OVFS

kod ağacı tanımlamış ve uzaysal çoğullamayı bu kanal yapısına uygun şekilde modellemişlerdir. Böylelikle sistemde oluşan performans değişimi ve hesaplama karmaşıklığını incelemişlerdir.

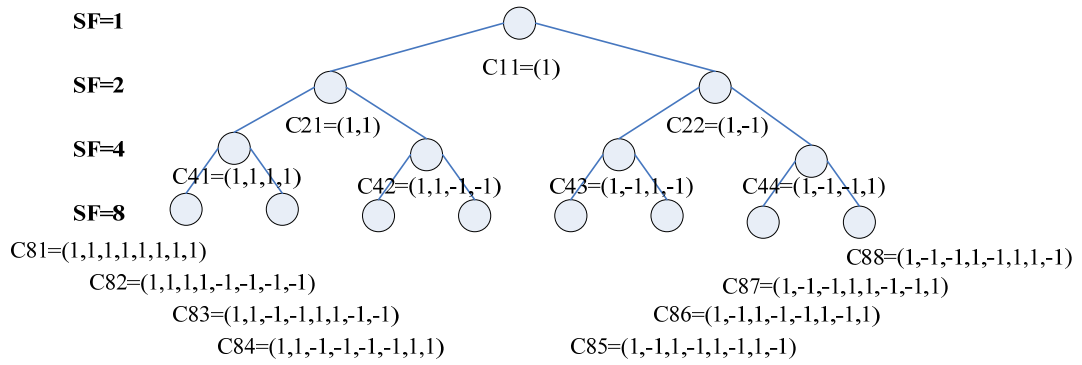
1.4. OVFS Kod Ağacı

Bir WCDMA sisteminde, kullanıcıya ait bilgi sembollerine iki işlem uygulanır. Bunlardan birincisine kanallama (channelization) adı verilir ve her bir bit, kod dizisi haline dönüştürülür. Oluşturulan bu kod dizisinin uzunluğu yayılım faktörü (spreading factor, SF) olarak adlandırılır ve genel olarak ikinin üstel katları şeklindedir. Uygulanan ikinci işlem, sinyalin yayılımı (scrambling) amaçlıdır. Bu kod, yayılım sinyaline uygulanır ve tek bir kullanıcıdan iletilen ardışık bilgi sembollerinin ayrıştırılmasında kullanılır.

OVFS kanallama kodları, WCDMA sisteminde farklı uygulamaların ihtiyaç duyduğu değişken veri transfer oranlarına ait bantgenişliğini karşılamak amacı ile değişken veri oranlarını destekler. OVFS kodları mobil kullanıcının farklı fiziksel kanalları arasındaki diklik özelliğini korur. Bu kodlar, kod ağacı biçiminde Şekil 1.4'deki gibi ifade edilmektedir. OVFS kod ağacındaki kanallama kodları, $C_{SF,k}$ şeklinde belirtilen özgün bir gösterime sahiptir. Burada SF yayılım faktörünü, k ise $1 \leq k \leq SF$ arası değere sahip kod numarasını göstermektedir. Örneğin $SF=1$ için tek bir kod, $SF=2$ için iki farklı veri transfer oranında (R ve 2R) üç farklı kod (iki adet R hızında, bir adet 2R hızında), $SF=4$ için üç adet birbirinden farklı veri transfer oranları (R, 2R, 4R) ve yedi farklı kod (dört adet R hızında, iki adet 2R hızında, bir adet 4R hızında) bulunmaktadır. Genellemek gerekirse SF büyüklüğündeki bir kod ağacı için $(\log_2 SF)+1$ adet birbirinden farklı veri transfer oranları ve $(2SF-1)$ adet farklı kod bulunmaktadır.

Ağacın türetilmesinde uygulanan kural gereği, SF yayılım faktöründe her bir $\{x\}$ kodu için $2SF$ yayılım faktöründe iki çocuk kod elde edilmektedir. Bu kodlardan ilk olanı sol alt dal ile ifade edilir ve ata kodun iki kere tekrarlanmasıyla oluşur $\{x,x\}$. Diğeri ise sağ alt dal ile ifade edilir ve ata koda onun dönüştürülmüş halinin

eklenmesi ile elde edilir $\{x, -x\}$. Aynı katmanda olan tüm kodlar birbiri ile diktir. Bununla beraber farklı katmanlardaki kodlar da aralarında çocuk-ata ilişkisi olmadığı sürece birbiri ile diktir. Kod ağacında bir seviye aşağıya inildiğinde kod uzunluğu iki kat artarken veri transfer hızı da aynı oranda azalır. Aynı ağacı kullanan iki ayrı mobil kullanıcıya birbiri ile dik olmayan OVFS kanallama kodları verilemez. UTRAN FDD standartlarına göre (3GPP2 C.S0024-B, 2006) UL boyunca 4–256 arası, DL boyunca 4–512 arası OVFS kod ağacı tahsisi yapılır (Holma ve Toksala, 2006).



Şekil 1.4: OVFS kod ağacı (SF=8 için)

1.5. Tezin Organizasyonu

Bu tez çalışması Bölüm 1.2’de de bahsedildiği gibi maksimum sayıda kod tahsisi sağlayabilmek amacı ile OVFS kod ağacını optimize etmeyi amaçlamıştır. Bu bağlamda yapılan çalışmalar çeşitli konferanslarda sunulmuş ve dergilerde yayınlanmış ya da yayınlanmak üzere kabul edilmiştir. Tezin organizasyonu, her bölümde yapılan çalışmalar ve bu çalışmaların sonucu kabul edilen makaleleri belirtmektedir. Bu tez çalışması aşağıda kısaca açıklanan 6 bölümden oluşmaktadır.

Bölüm 1: Giriş bölümünde kablosuz haberleşme sistemlerinin tarihsel gelişimi, tez çalışmasına konu olan problemin tanımı, çalışmanın amacı ve bu çalışmanın sağladığı faydalar, literatürde bu problemin çözümü üzerine yapılan çalışmalar, OVFS kod ağacının tanımı ve önemi ile tez organizasyonu hakkında bilgi sunulmaktadır.

Bölüm 2: 3G sistemler için WCDMA standardının kısa bir özeti verilmiştir. WCDMA özellikleri, protokol mimarisi, fiziksel kanallar, çoklu oranlı kullanıcı veri transferi ile ilgili kavramlar burada ele alınmaktadır.

Bölüm 3: OVSF kod ağacı kod tahsis ve tekrar tahsis algoritmalarından kod sınırlı yapıda olanları içermektedir. Genetik Algoritma (GA), Benzetilmiş Tavlama (SA), Uyarlamalı Benzetilmiş Tavlamalı Genetik Algoritma (ASAGA) tabanlı dinamik kod tahsis yöntemleri bu bölümde sunulmuştur.

Bölüm 4: OVSF kod ağacı kod tahsis ve tekrar tahsis algoritmalarından sinyal gürültü oranı (SIR) sınırlı yapıda olanları içermektedir. BTS’de uyarlamalı anten dizisinin (AAA) kullanımı ile mobil kullanıcının (MS) uzaysal çeşitlilik bilgisi OVSF kanallama kodu atamasında kullanılmıştır. Çoklu yol yayılım etkisinin sistem analizinde hesaba katıldığı ve çoklu yoldan bağımsız sistem analizinin ele alındığı SIR sınırlı algoritmalar bu bölümde sunulmuştur.

Bölüm 5: Kod sınırlı ve SIR sınırlı dinamik OVSF kod tahsis algoritmalarının sistem performansları, algoritmik parametrelerin etkisi ve maliyet analizi cinsinden performansları incelenmiştir. Ayrıca sunulan bu algoritmaların literatür tarafından referans alınan geleneksel kod tahsis (CCA) (Okawa ve Adachi, 1998) ve dinamik kod tahsis (DCA) (Minn ve Sui, 2000) algoritmaları ile karşılaştırmaları yapılmıştır.

Sonuçlar ve Öneriler bölümünde, yapılan çalışmalardan elde edilen sonuçlar genel hatlarıyla değerlendirilerek çalışmanın bilime ve endüstriye sağlayabileceği katkılar tartışılmıştır. Daha sonra yapılabilecek çalışmalar için önerilerde bulunmaktadır.

2. GENİŞBANT KOD BÖLÜŞÜMLÜ ÇOKLU ERİŞİM (WIDEBAND CODE DIVISION MULTIPLE ACCESS, WCDMA)

2.1. Giriş

Bu bölüm 3G sistemler için WCDMA standardının kısa bir özetini içermektedir. Bu standart IMT-2000, 3GPP ve UMTS olarak bilinen ana yapıların bir parçası olarak da düşünülebilir. 2G sistemlerin, özellikle GSM'in başarısı daha verimli sistemlerin gelişimi için cesaret verici bir etken olmuştur. Burada sistemin temel amacı maddeler halinde aşağıdaki gibi sıralanabilir;

- Daha iyi spektral verimlilik
- Daha yüksek veri transfer hızının elde edilmesi (2Mbps civarında); 200 KHz yerine 5MHz'lik bantgenişliğine sahip bir kanalın seçilmesi ile
- Farklı veri transfer hızı gereksiniminde olan video, resim, yazı ve ses iletimini içeren çoklu ortam uygulamalarının desteklenmesi
- 2G sistemleri de kapsayan uyumluluk

Avrupa'da yukarıda saydığımız amaçlara uygun bir sistem geliştirmek için 1990'lı yıllarda ilk olarak Avrupa Birliği Yardım Araştırma Programı altında, daha sonra daha resmi bir yapı şeklinde Avrupa Haberleşme Standartları Enstitüsü (European Telecommunications Standards Institute, ETSI) programı altında araştırma aktivitelerine başlamıştır. Bu bağlamda farklı çalışma grupları oluşturulmuş ve 1998 yılındaki final sürecinde iki taslak üzerinde mutabakata varılarak standartlaşma sağlanmıştır. Bunlar; genişbant CDMA ki aynı zamanda FDD mod olarak bilinir ve temel sistemi ifade eder. Bir diğeri ise T/CDMA ki aynı zamanda TDD mod olarak bilinir ve yüksek hızlı uygulamaları destekler. FDD ve TDD modlar genel olarak WCDMA adı altında toplanmıştır. Sistem temeli FDD modda oluşturulduğundan WCDMA teknolojisi incelenirken bu mod temel alınmıştır. Aynı zamanda WCDMA

genel olarak radyo erişim teknolojileri için bir kısaltma olarak kullanılırken, UMTS ifadesi CN’i de içine alan tüm sistemi ifade etmek için kullanılır (Chaudry ve diğ. 1999).

Tablo 2.1: WCDMA Parametreleri

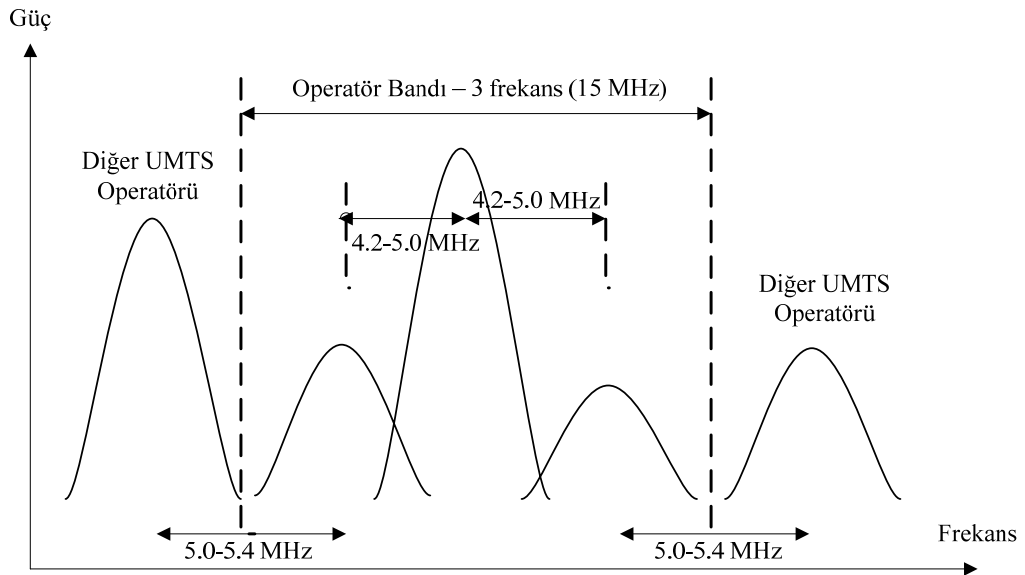
Kanal bantgenişliği	5 MHz
Çoğullama modu	FDD ya da TDD
DL radyo frekans (radio frequency RF) kanal modu	Doğrudan Yayılım
Çip oranı	3.84 Mcps
Çerçeve uzunluğu	10 ms
Yayılım modülasyonu	Dengeli QPSK (DL) Çift kanal QPSK (DL) Karmaşık yayılım devresi
Data modülasyonu	QPSK (DL) BPSK (UL)
Kanal kodlama	Evrişimli ve turbo kodlar
DL için kanal çoğullama	Data ve kontrol kanallarının zamanla çoğullanması
UL için kanal çoğullama	Kontrol ve pilot kanallarının zamanla çoğullanması, Data ve kontrol kanallarının I&Q çoğullanması
Çoklu oran	Değişken yayılım ve çoklu kod
Yayılım faktörü	4-256 (UL), 4-512 (DL)
Yayılım (DL)	Kanal ayırımı için OVFSF kodları Kullanıcı ve hücre ayırımı için Gold kodları (uzunluk: $2^{18}-1$, kısaltılma çevrimi: 10 ms)
Yayılım (UL)	Kullanıcı ayırımı için 2^{41} uzunluğunda Gold kodları ve OVFSF kodları (I ve Q kanallarında farklı zaman gecikmeleri, kısaltılma çevrimi: 10 ms)

2.2. Sistem Genel Görünüşü

WCDMA hava arayüzü, aynı zamanda UTRAN olarak ta adlandırılır, UE ile Node-B arasındaki iletişim ortamı olarak ifade edilir. Tablo 2.1, WCDMA parametrelerini listelemektedir. Bu arayüz CDMA erişim tekniğini temel olarak kullanır (Zigangirov, 2004). Sistemin çip oranı 3.84 Mcps’dir. WCDMA çerçeve uzunluğu 10 ms’dir ve her çerçeve 15 yuvadan (slot) oluşur (3.84 Mcps çip oranında 2560 çip/slot). Yayılım kodlarının (spreading codes, SF) aralığı UL için 4–256, DL için 4–512 arasındadır. DL için Gold adı verilen kodlar, farklı hücreleri ayırtırmak için kullanılan 10 ms periyoda (3.84 Mcps için 38400 çip) sahip olan kodlardır. Bu kodun kendi uzunluğu $2^{18}-1$ çip’tir. UL için ise yine 10 ms periyoda sahip Gold kodlar ya da alternatif

olarak kullanılabilecek 256 çip periyoda sahip kısa kodlar farklı kullanıcıları ayırtmak için kullanılırlar. Aynı zamanda, yayılım kodları kullanıcı tarafından talep edilen bantgenişliğinin belirlenmesinde, SF kodları farklı UE'ler ve/veya Node-B'ler arasındaki sinyalin ayırt edilmesinde kullanılır.

Kanal bantgenişliği girişim (interference) senaryolarına bağlı olarak komşu kanalın negatif etkisinden taşıyıcı kanalın korumasını sağlayacak şekilde 4.2 MHz ile 5.4 MHz arasında seçilir. Şekil 2.1, üç hücreli 15 MHz bantgenişliğine sahip bir operatör için bu örneği tanımlamaktadır. Operatörler arası girişimi minimize etmek için farklı operatörlerin taşıyıcı kanalları arasındaki mesafe, bir operatörün kendi taşıyıcı kanalları arasındaki mesafeye göre daha fazla tutulur (Holma ve Toksala, 2000).



Şekil 2.1: WCDMA ile frekans kullanımı

2.3. WCDMA Temel Özellikleri

Tablo 2.2.'de özetlenen WCDMA özellikleri 3GPP standartları tarafından belirlenmiş hava arayüz tanımlamalarıdır.

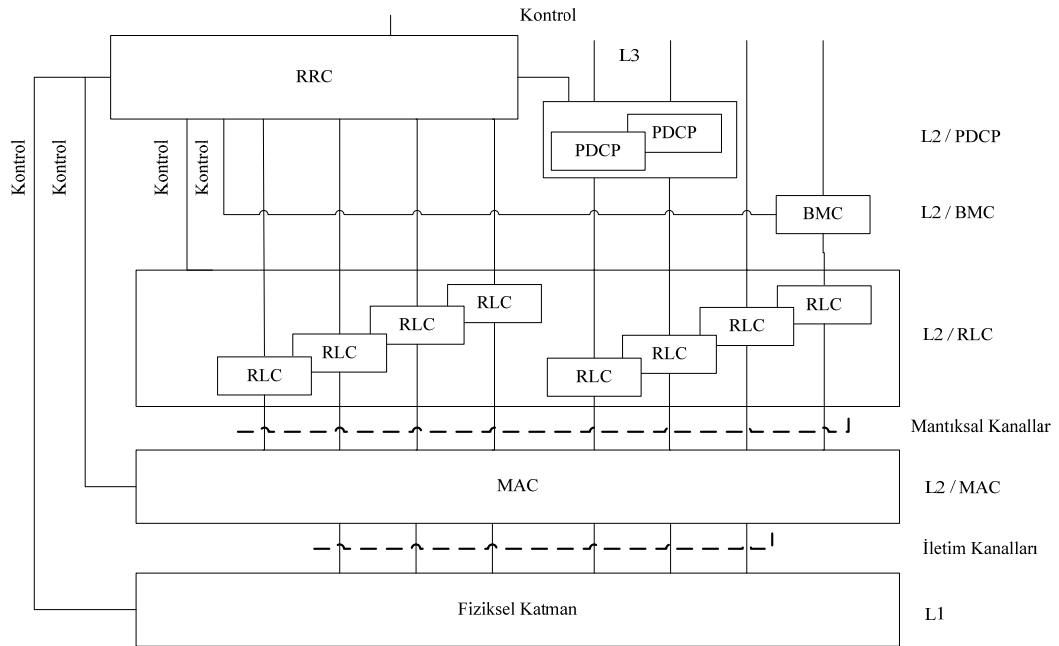
Tablo 2.2: 3GPP radyo erişim ağı özellikleri

No	İsim	Faaliyet Alanı
TS 25.201	Fiziksel katman – Genel tanımlar	Fiziksel katman içeriklerine dair genel tanımları içerir.
TS 25.211	Fiziksel kanallar ve taşıma kanallarının fiziksel kanallar üzerinde gönderimi (FDD)	FDD modda fiziksel kanallar ve 1. katman taşıma kanallarının karakteristiğini oluşturur ve tanımlar. • Taşıma kanalları • Fiziksel kanallar ve yapıları • Aynı linkteki farklı fiziksel kanallar arasındaki bağıl zamanlama • UL ve DL arasındaki bağıl zamanlama • Taşıma kanallarının fiziksel kanallar üzerinde gönderimi
TS 25.212	Çoğullama ve kanal kodlama (FDD)	FDD modda çoğullama, kanal kodlama ve serpiştirmeyi tanımlar. • Taşıma kanallarının kodlanması ve çoğullanması • Kanal kodlama alternatifleri • Fiziksel katman kontrol bilgisi için kodlama • Farklı serpiştirmeler • Oran eşleştirme • Fiziksel kanal bölümlenmesi ve gönderimi
TS 25.213	Yayılım ve modülasyon (FDD)	FDD modda yayılım ve modülasyonun karakteristiğini tanımlar. • Yayılım • Kanallama ve yayılım kodlarının üretimi • Senkronizasyon kodlarının üretimi • Modülasyon
TS 25.214	Fiziksel kanal prosedürleri (FDD)	FDD modda fiziksel kanal prosedürlerini tanımlar. • Hücre arama prosedürleri • Güç kontrol prosedürleri • Rasgele erişim prosedürü
TS 25.215	Fiziksel kanal ölçümleri (FDD)	FDD modda fiziksel kanal ölçümlerini tanımlar. • 1. katman tarafından icra edilen ölçümler • Daha yüksek katmanlara ölçümlerin raporlanması • Frekans geçişi ölçümleri • Sessiz mod ölçümleri

2.4. Protokol Mimarisi

Şekil 2.2, hava arayüz protokol mimarisini göstermektedir. Bu hava arayüzü üç protokol katmanından oluşmuştur.

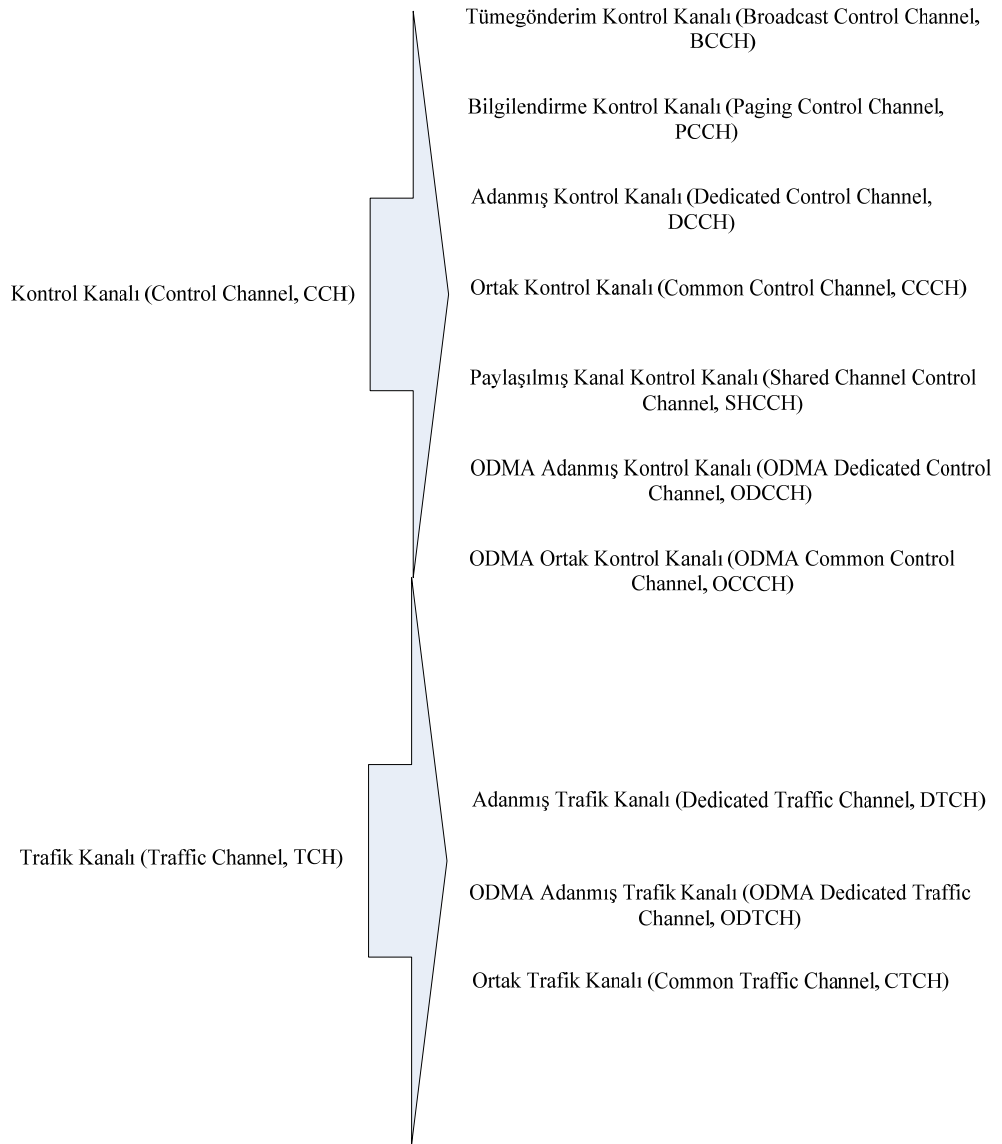
- Fiziksel katman (Katman 1, L1)
- Data link katmanı (Katman 2, L2)
- Ağ katmanı (Katman 3, L3)



Şekil 2.2: Hava arayüz protokol mimarisi

Fiziksel katman, data link katmanının Ortam Erişim Kontrol (Medium Access Control, MAC) alt katmanı ile ağ katmanının Radyo Kaynak Kontrol (Radio Resource Control, RRC) alt katmanı arasındaki bir arayüz olarak görev yapar. Bu katman MAC için farklı iletim kanallarını sunar. İletim kanalı hava arayüzü üzerinden iletilecek olan bilginin nasıl transfer edileceğini karakterize eder. MAC ise data link katmanının Radyo Link Kontrol (Radio Link Control, RLC) alt katmanı için farklı mantıksal kanalları sunar. Her bir mantıksal kanal da iletilecek bilginin tipine bağlı olarak şekillenir.

Yukarıda belirtildiği gibi data link katmanı MAC, RLC, Paket Data Yakınsama Protokolü (Packet Data Convergence Protocol, PDCP) ve Tüme/Çoğa Gönderim Kontrol (Broadcast/Multicast Control, BMC) alt katmanlarından oluşur. Ağ katmanı ve RLC, kontrol amaçlı ve trafik amaçlı olarak bölünmüşlerdir. PDCP ve BMC sadece kullanıcı amaçlı olarak belirlenmiştir. Kontrol amaçlı yapılar ise RRC ve data link katmanı arayüzleridir (ETSI TS 125 201).



Şekil 2.3: Mantıksal kanal yapısı

2.4.1. Mantıksal kanallar

MAC katmanı mantıksal kanallar üzerinde veri transfer servislerini sağlar. Mantıksal kanallardan oluşacak olan bir kanal dizilimi MAC tarafından sunulan veri transfer servislerinin çeşitliliğine bağlı olarak şekillenir. Her mantıksal kanal tipi iletilecek bilginin tipine bağlı olarak belirlenmiştir. Mantıksal kanal tipleri aşağıdaki gibi verilen iki ana gruptan oluşur (ETSI TS 125 201). Bunlar;

- Kontrol amaçlı bilginin iletimi için mantıksal kanallar (Tablo 2.3.)
- Trafik amaçlı bilginin iletimi için mantıksal kanallar (Tablo 2.4.)

Tablo 2.3: Mantıksal kontrol kanalları

Tümegönderim kontrol kanalı (Broadcast control channel, BCCH)	DL yönünde sistem kontrol bilgisinin tümegönderiminden sorumludur.
Bilgilenme kontrol kanalı (Paging control channel, PCCH)	DL yönünde bilgilendirme transferinden sorumludur ve şu hallerde kullanılır; • Hücre içinde hangi mobil kullanıcıların olduğunu tespit etmek için • Hücredeki mobil kullanıcıların uyku moduna geçişi esnasında
Ortak kontrol kanalı (Common control channel, CCCH)	Şebeke ve mobil kullanıcı arasında iki yönlü olarak kontrol bilgisinin transferinden sorumludur ve şu hallerde kullanılır; • Mobil kullanıcı, şebeke ile RRC bağlantısına sahip olmadığı zaman • Mobil kullanıcı hücre değiştirdikten sonra yeni hücreye erişim sağlamak için ortak iletim kanallarını kullanacağı zaman
Adanmış kontrol kanalı (Dedicated Control Channel, DCCH)	Şebeke ve mobil kullanıcı arasında noktadan noktaya iki yönlü adanmış kontrol bilgilerinin iletilmesinde kullanılır. Bu kanal kurulum esnasında RRC bağlantısı kanalıyla oluşturulur.
ODMA ortak kontrol kanalı (ODMA common control channel, OCCCH)	Mobil kullanıcılar arasında kontrol bilgilerinin iletimi için oluşturulan iki yönlü kanaldır.
ODMA adanmış kontrol kanalı (ODMA dedicated Control Channel, ODCCH)	Mobil kullanıcılar arasında adanmış kontrol bilgilerinin iletimi için oluşturulan iki yönlü kanaldır. Bu kanal kurulum esnasında RRC bağlantısı kanalıyla oluşturulur.

Tablo 2.4: Mantıksal trafik kanalları (kullanıcı kanalları)

Adanmış trafik kanalı (Dedicated traffic channel, DTCH)	Noktadan noktaya kanaldır. Kullanıcı bilgilerinin iletimi için bir mobil kullanıcıya adanmıştır. DTCH hem UL'de hem de DL'de vardır.
ODMA adanmış trafik kanalı (ODMA dedicated traffic channel, ODTCH)	Noktadan noktaya kanaldır. Mobil kullanıcılar arasında kullanıcı bilgilerinin iletimi için bir mobil kullanıcıya adanmıştır.

2.4.2. İletim kanalları

İletim kanalı hava arayüzü üzerinden iletilecek olan verinin karakteristiğinin ne ve nasıl olacağını tanımlayan kanaldır. İki farklı iletim kanalı vardır (ETSI TS 125 211). Bunlar;

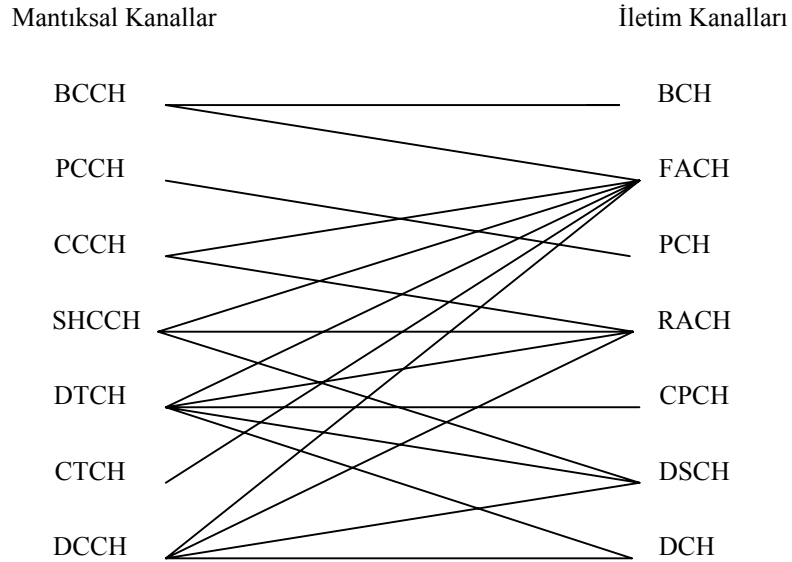
- Adanmış kanallar
- Ortak kanallar (Tablo 2.5.)

Adanmış iletim kanalları olarak bir tane kanal vardır. Bu kanal, adanmış kanal (Dedicated channel, DCH) olarak adlandırılır ve DL ya da UL boyunca aktif haldedir. DCH, tüm hücreye/tüm hücreden iletilir. Görevi, hızlı veri transfer oranlarının değişimi (her 10 ms'de), hızlı güç kontrolü ve mobil kullanıcının adreslenmesi şeklindedir. Şekil 2.4, iletim ve mantıksal kanallar arasındaki bağlantıları adreslemeyi göstermektedir ve aşağıdaki bağlantılar mevcuttur.

- BCCH kanalı BCH'e bağlanır, aynı zamanda FACH'a bağlanabilir.
- PCCH kanalı PCH'e bağlanır.
- CCCH kanalı RACH ve FACH'a bağlanır.
- SHCCH kanalı RACH, USCH/FACH ve DSCH'ye bağlanır.
- DTCH kanalı ya RACH ya da FACH'ye, RACH ve DSCH'ye, DCH ve DSCH'ye, DCH'ye, FDD mod için de CPCH'ye bağlanır.
- CTCH kanalı FACH'ye bağlanır.
- DCCH kanalı ya RACH ve FACH'ye, RACH ve DSCH'ye, DCH ve DSCH'ye, DCH'ye ve CPCH'ye bağlanır.

Tablo 2.5: Ortak iletim kanalları

Tümegönderim kanalı (Broadcast channel, BCH)	DL iletim kanalı hücreye özgü bilginin tümegönderildiği sistemdir. BCH daima tüm hücreye düşük sabit bit oranı ile gönderilir.
İleri erişim kanalı (Forward access channel, FACH)	DL iletim kanalıdır. Sadece ışına örüntülü (beamforming) anten kullanan hücrelere ya da tüm hücrelere gönderilir.
Bilgilenme kanalı (Paging channel, PCH)	DL iletim kanalıdır. PCH daima tüm hücreye gönderilir. PCH'in iletimi, verimli bir eylemsizlik modu prosedürünü oluşturmak için bir fiziksel katman sinyalinin iletimi ile ve bilgilendirme belirteci ile ifade edilir.
Rasgele erişim kanalı (Random access channel, RACH)	UL iletim kanalıdır. RACH daima tüm hücre içinden alınır. RACH sınırlı boyuta sahip bir veri alanı ile, bir çarpışma riski ile ve açık çevrim güç kontrolünün kontrolü ile karakterize edilir
Ortak paket kanalı (Common packet channel, CPCH)	UL iletim kanalıdır. CPCH, veri trafiğinin iletimi için kullanılan içerik tabanlı rasgele erişim kanalıdır. CPCH, UL CPCH için güç kontrolünü sağlamak amacıyla DL yönünde adanmış bir kanal ile ifade edilir.
DL paylaşılmış kanal (Downlink shared channel, DSCH)	DL iletim kanalı farklı mobil terminaller tarafından paylaşılır. DSCH, DCH ile ifade edilir.



Şekil 2.4: İletim ve mantıksal kanallar arasındaki bağlantıların adreslenmesi

2.5. Fiziksel Kanallar

İletim kanalları, fiziksel kanallar tarafından belirlenen veri transfer oranlarına göre adreslenmiş ve kodlanmış kanallardır. Dolayısıyla iletim kanalları ile fiziksel kanallar birbirleri ile direk ilişkili kanallardır. Fiziksel kanallar radyo çerçeveleri (radio frame) ve zaman yuvalarından (time slot, *TS*) oluşurlar. Bir radyo çerçevesinin uzunluğu 10 ms ve her bir çerçeve de 15 zaman yuvasından oluşmaktadır. Zaman yuvası, bit adını verdiğimiz en küçük bilgi birimlerini içeren bir alandır. Bir zaman yuvasındaki bitlerin sayısı fiziksel kanalın durumuna bağlıdır. Fiziksel kanalın sembol oranına bağlı olarak radyo çerçevelerinin ya da zaman yuvalarının konfigürasyonu değişir. Veri transfer oranını belirleyen temel fiziksel kaynak, kod/frekans düzlemi tarafından belirlenir. Buna ek olarak, UL’de farklı bilgi yığınları I ve Q dalları üzerinden iletilebilir. Sonuç olarak fiziksel kanal, taşıyıcı frekans, kod ve kısmi faz (0 yada $\pi/2$) bilgisine göre şekillenir.

2.5.1. UL fiziksel kanalları

Bu amaçla kullanılan iki UL adanmış fiziksel kanal ve iki ortak fiziksel kanal vardır (ETSI TS 125 201). Bunlar;

- UL adanmış fiziksel data kanalı (Dedicated physical data channel, DPDCH) ve UL adanmış fiziksel kontrol kanalı (Dedicated physical control channel, DPCCH)
- Fiziksel rasgele erişim kanalı (Physical random access channel, PRACH) ve fiziksel ortak paket kanalı (Physical common packet channel, PCPCH)

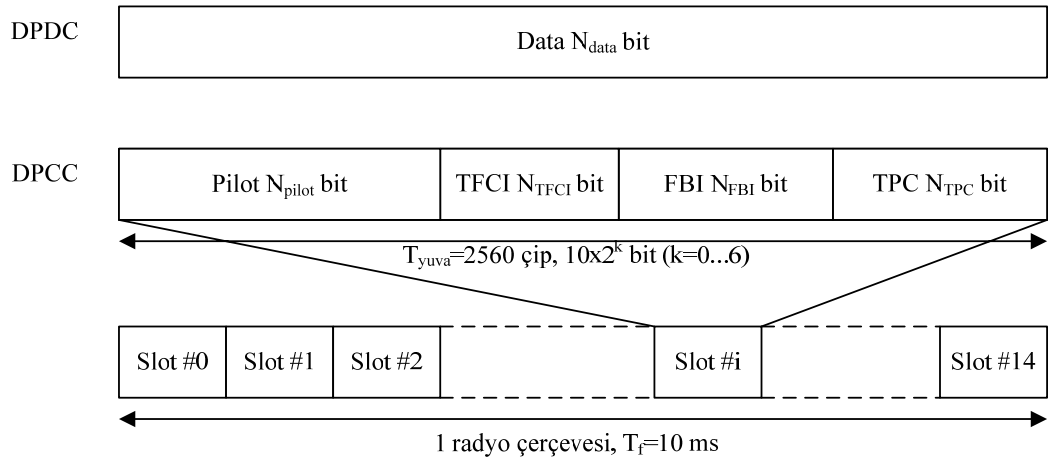
UL DPDCH kanalı MAC ve yukarısı katmanlar (örneğin adanmış iletim kanalı – dedicated transport channel, DTCH-) tarafından üretilen adanmış verinin taşınması için kullanılır. Burada her bir fiziksel katman bağlantı içeriğinde sıfır, bir ya da birkaç tane UL DPDCH kanalı bulunur. UL DPDCH kanalı fiziksel katmanda üretilen kontrol bilgilerinin iletilmesi amacıyla kullanılır. Kontrol bilgilerinin içeriğinde faz uyumlu alım için kanal kestirimini destekleyecek ve değeri bilinen

pilot bitleri, iletim güç kontrol (transmit power control, TPC) komutları, geribildirim bilgisi (feedback information, FBI) ve seçmeli nakil format kombinasyon belirteci (transport format combination indicator, TFCI) bulunur. TFCI bilgisi, UL DPDCH kanalı üzerinde çoğullanmış farklı iletim kanallarının anlık durumu hakkında alıcıyı bilgilendirir. Her bir fiziksel katman bağlantı için, ortamda sadece bir tane UL DPCCH kanalı bulunur.

Şekil 2.5. UL adanmış fiziksel kanallarının çerçeve yapısını gösterir. Her çerçeve uzunluğu 10 ms'dir ve 15 yuva bir çerçeveyi oluşturur. Ayrıca bir yuva uzunluğu $T_{slot}=2560$ çip'ten oluşmaktadır ve bir güç kontrol periyoduna karşılık gelir. Aynı şekildeki k parametresi her bir UL DPDCH/DPCCH kanalı üzerindeki bitlerin sayısını gösterir. Bu değer aynı zamanda SF ile ilintilidir (ETSI TS 125 211).

$$SF = 256 / 2^k \quad (2.1)$$

UL DPDCH kanalı SF dağılımı, 256'dan 4'e kadar değişen bir değer sergiler. Aynı fiziksel katman bağlantısı üzerinde bulunan bir UL DPDCH ve UL DPCCH kanalı genel olarak farklı veri iletim oranlarında ve dolayısıyla farklı SF değerinde olabilir.



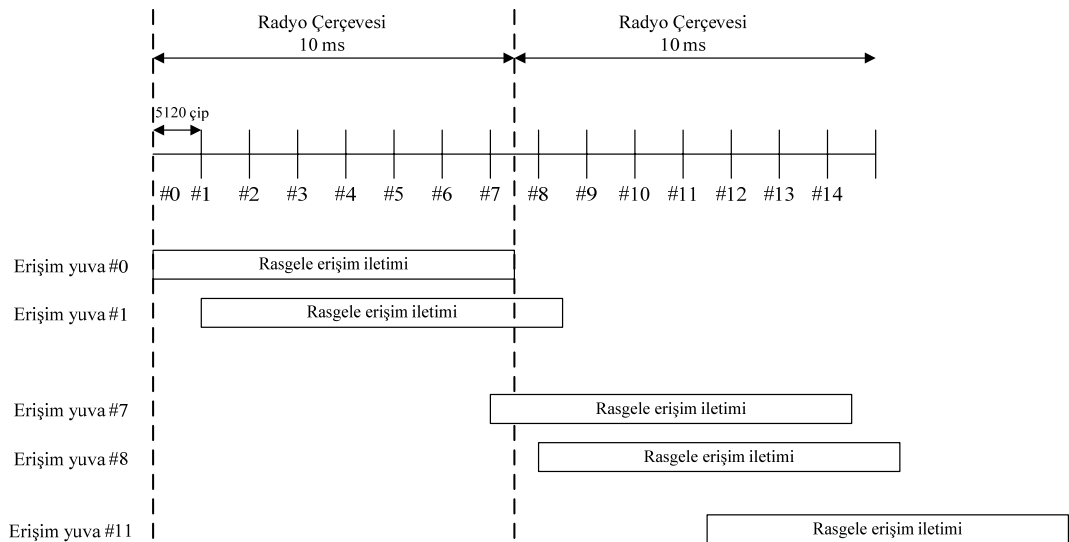
Şekil 2.5: UL DPDCH/DPCCH için çerçeve yapısı

Değişken veri transfer oranlarına sahip çoklu paralel servisler (=adanmış mantıksal trafik ve kontrol kanalları) her bir DPDCH çerçevesi içinde zamanla çoğullanabilir. DPDCH bit oranı çerçeveden çerçeveye değişir. Genelde her bir bağlantı için sadece bir DPDCH kanalı atanır ve servisler bu DPDCH kanalını paylaşarak taleplerini

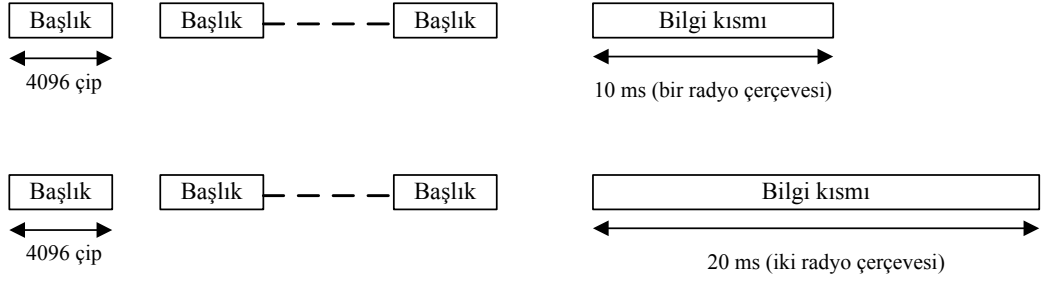
yürütürler. Bununla birlikte birden fazla DPDCH kanalı da bir bağlantı esnasında atanabilir. Çoklu kod transferi kullanıldığı zaman paralel DPDCH kanalları farklı kanallama kodlarını kullanarak iletim yapabilirler.

PRACH, RACH taşımak üzere kullanılır. Rasgele erişim transferi yerleştirilmiş hızlı kazanım göstergeleri ile ALOHA yaklaşımına dayanır. Mobil istasyon (mobile station, MS) erişim yuvalarıyla belirtilen belirli sayıda iyi tanımlanmış zaman gecikmelerle iletimi başlatabilir. Her iki çerçeve başına 15 erişim yuvası bulunur ve 5120 çip aralıklı yerleştirilir. Şekil 2.6 erişim yuvası sayısını ve birbirlerine uzaklıklarını gösterir. Erişim yuvalarının varsayılan hücrede ne yapabildiğine dair bilgi yüksek tabakalarla verilir. Rasgele erişim transferinin yapısı Şekil 2.7 de verilmektedir. Rasgele erişim transferi 4096 çip uzunluğundaki bir veya birçok başlangıç eki (preamble) ve 10 ya da 20ms uzunluğundaki mesajlardan oluşur. MS, mesaj bölümünün ağa uzaklığını belirli imzalar (signature) kullanarak gösterir (ETSI TS 125 211).

Rasgele erişim veri dizisinin başlangıç eki bölümü bir imzanın 256 kez tekrarından oluşur. 16 uzunluğunda Hadamard kodu setine dayanan 16 farklı imza bulunur.

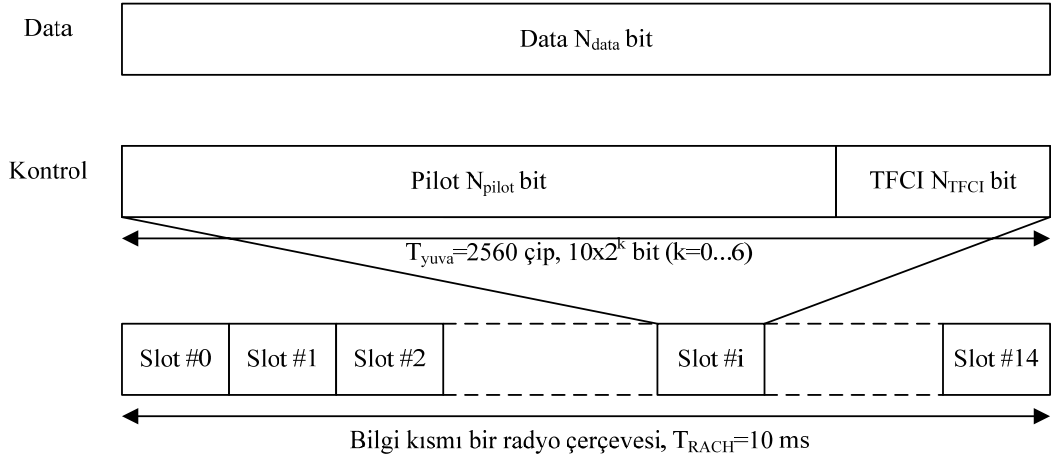


Şekil 2.6: RACH erişim yuvası sayısı ve yerleşim yerleri



Şekil 2.7: Rasgele erişim iletiminin yapısı

Şekil 2.8 radyo çerçevesinin rasgele erişim mesaj bölümünün yapısını gösterir. Radyo çerçevesinin 10 ms mesaj bölümü her biri $T_{yuva} = 2560$ çip uzunluğundaki 15 yuvaya ayrılır. Her yuva, MAC bilgisi taşıyan veri bölümü ve fiziksel katman kontrol bilgisi taşıyan kontrol bölümü olmak üzere iki bölümden oluşur. Veri ve kontrol bölümleri paralel olarak iletim yapar. 20 ms uzun mesaj bölümü iki ardışık mesaj radyo çerçevesinden oluşur. Veri bölümü 10×2^k bitten oluşur, burada $k=0, 1, 2, 3$ tür. Bu, mesaj veri bölümü için sırasıyla 256, 128, 64 ve 32 yayılım faktörüne karşılık gelir (ETSI TS 125 211).

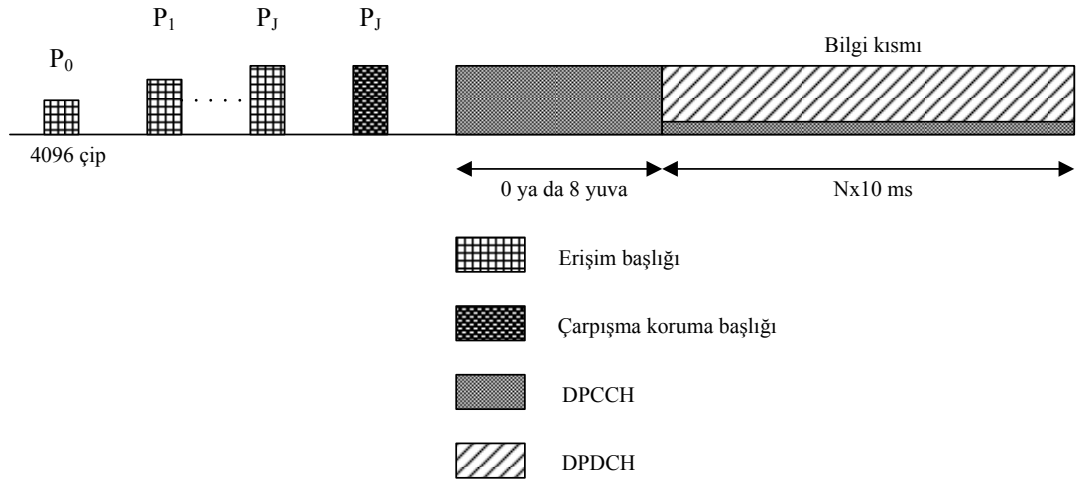


Şekil 2.8: Rasgele erişim mesaj kısmının radyo çerçeve yapısı

Kontrol bölümü evre uyumlu algılama için kanal tahminini desteklemek üzere bilinen sekiz pilot bitten ve iki TFCI bitten oluşur. Bu, mesaj kontrol bölümü için 256 yayılım faktörüne karşılık gelir.

PCPCH, CPCH iletim kanalını taşımak üzere kullanılır. CPCH iletimi hızlı hedef (acquisition) göstergesi ile birlikte DSMA-CD yaklaşımına dayanır. MS, mevcut

hücrenin alınan BCH'sinin çerçeve sınırına bağlı olarak belirli sayıda iyi tanımlanan zaman gecikmesinde (time-offset) iletimi başlatabilir. CPCH rastgele erişim iletiminin yapısı Şekil 2.9'da gösterilir. CPCH rasgele erişim iletimi, 4096 çip uzunluğunda bir veya birden fazla erişim başlığı, 4096 çip uzunluğunda bir çarpışma koruma başlığı (CD-P), bir DPCCH güç kontrolü başlığı (0 yuva ya da 8 yuva uzunluğunda) ve $N \times 10$ ms uzunluğunda değişken bir mesajdan oluşur (ETSI TS 125 211).



Şekil 2.9: CPCH rasgele erişim iletiminin yapısı

2.5.2. DL fiziksel kanalları

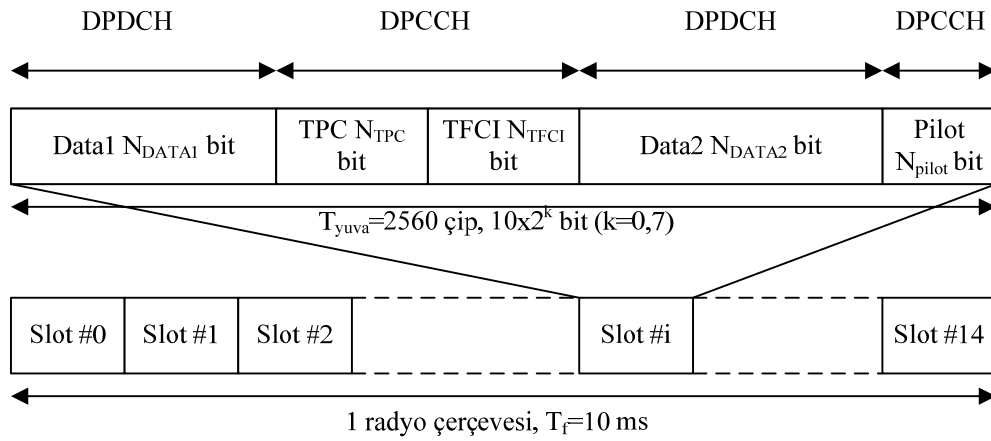
Biri paylaşılan ve beşi ortak kontrol kanalı olmak üzere adanan fiziksel kanallar DL yönünde bulunur:

- DL adanan fiziksel kanal (DPCH)
- Fiziksel DL paylaşım kanal (DSCH)
- Birincil ve ikincil ortak pilot kanalları (CPICH)
- Birincil ve ikincil ortak kontrol fiziksel kanalları (CCPCH)
- Senkronizasyon kanalı (SCH)

Şekil 2.10 DPCH'nin çerçeve yapısını gösterir. DPCH de adanmış iletim kanalı, fiziksel katmanda üretilen kontrol bilgisi ile zamanda çoğullanmış olarak iletilir

(bilinen pilot bitler, güç kontrol komutları ve seçimli iletim biçim kombinasyonu göstergesi). TFCI iletildiğinde DPCH birçok servisi bir arada bulundurabilir. TFCI iletilmediğinde ise DPCH sabitlenmiş hızda bir servisi bulundurabilir. TFCI'nın iletilip iletilmeyeceğini ağ belirler (ETSI TS 125 211).

Toplam bit hızı fiziksel bir DL kanalı için maksimum bit hızını aşan bir iletimde, çoklu kod geçişi uygulanır (örneğin, aynı SF 'yi kullanan birçok paralel DL DPCH iletilir). Bu durumda fiziksel katman kontrol bilgisi sadece ilk DPCH DL'de uygulanır.



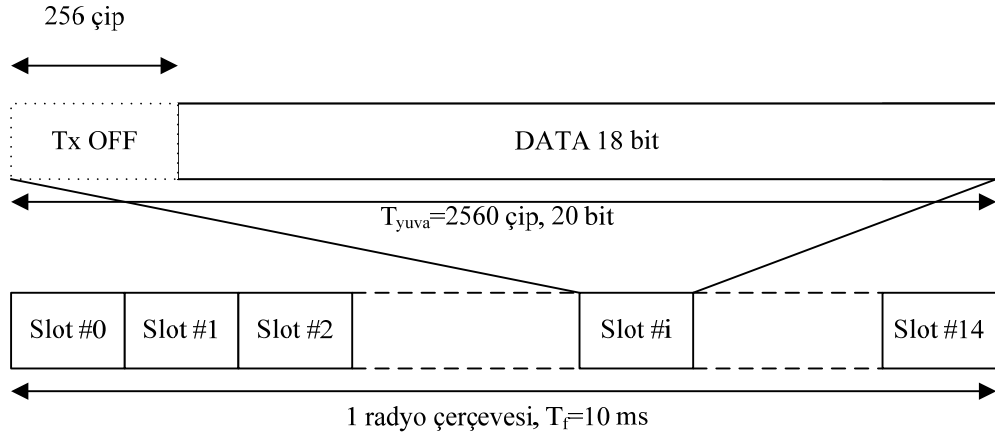
Şekil 2.10: DPCH DL için çerçeve yapısı

Ortak pilot kanalı (CPICH) daha önceden tanımlanan bir bit/sembol sıralaması taşıyan sabit hızlı (30 Kbps, $SF=256$) bir DL fiziksel kanalıdır. Tablo 2.6'da gösterildiği gibi birincil ve ikincil CPICH olmak üzere iki tür ortak pilot kanalı bulunur.

Birincil CCPCH, BCH'yi taşımak üzere kullanılan sabit hızlı (30 Kbps, $SF=256$) fiziksel DL kanallarıdır. Ortak fiziksel kontrol kanalları iç döngü güç kontrollü değildir. Şekil 2.11 birincil CCPCH'nin çerçeve yapısını gösterir. Birincil CCPCH her yuvanın ilk 256 çipi boyunca iletilmez. Bunun yerine bu periyotta birincil ve ikincil SCH'ler iletilir (ETSI TS 125 211).

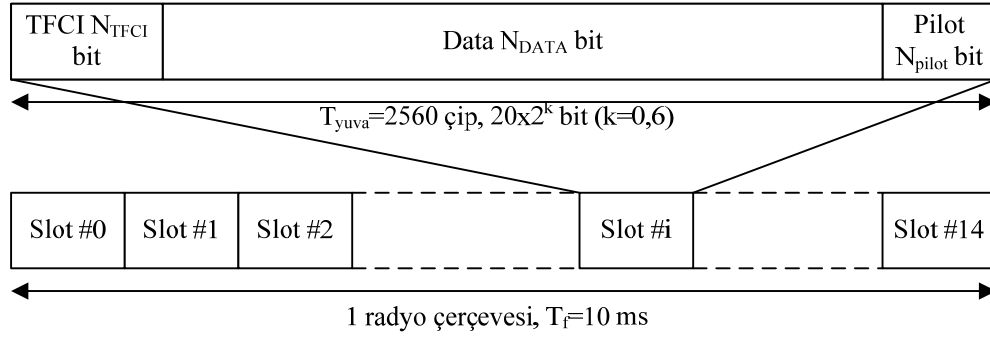
Tablo 2.6: Birincil ve ikinci CPICH

Birincil CPICH	- Her zaman aynı kanallama kodunu kullanır. - Birincil yayılım kodu yoluyla yayılır. - Hücre başına bir tanedir. - Bütün hücreye yayını yapar. - Birincil CPICH; SCH, birincil CCPCH, AICH, PICH için faz referansıdır. Ayrıca tüm diğer DL fiziksel kanalları için varsayılan faz referansıdır.
İkincil CPICH	- Hücre başına sıfır, bir veya birdir. - Hücresinin sadece bir kısmına iletebilir. - İkincil bir CPICH ikincil CCPCH ve DL DPCH için referans olabilir. Bu durumda MS, yüksek katman sinyalleme yoluyla bilgilendirilir.



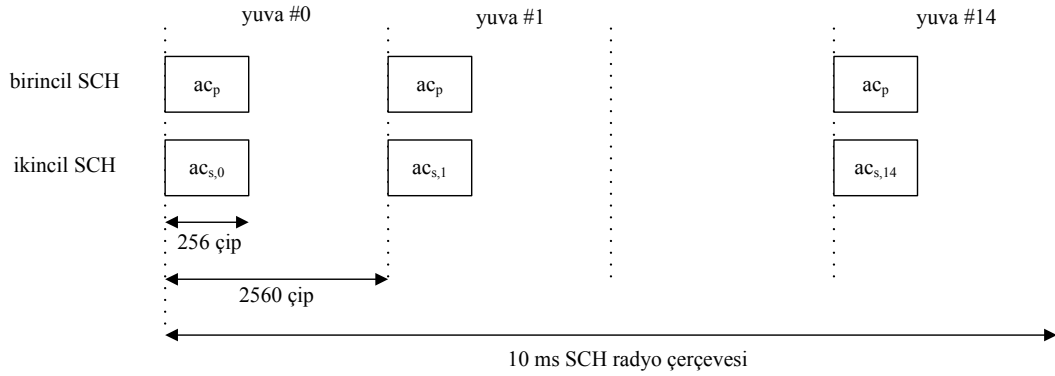
Şekil 2.11: Birincil CCPCH çerçeve yapısı

İkincil CCPCH'nin çerçeve yapısı Şekil 2.12'de verilmektedir. İkincil CCPCH, FACH ve PCH taşımak üzere kullanılır. Birincil ve ikincil CCPCH arasındaki temel fark, birincil CCPCH önceden tanımlanan sabit bir hıza sahipken ikincil CCPCH'nin değişken hızı desteklemesidir. Ayrıca, birincil CCPCH tüm hücre üzerinden kesintisiz olarak iletilirken, ikincil CCPCH sadece veriler uygun olduğunda iletilir ve adanmış bir fiziksel kanal ile aynı şekilde sınırlı bir doğrultuda iletebilir (sadece FACH taşıyan ikincil CCPCH için geçerlidir) (ETSI TS 125 211).



Şekil 2.12: İkincil CCPCH çerçeve yapısı

Şekil 2.13 hücre taraması için kullanılan senkronizasyon kanalının (SCH) yapısını gösterir. SCH iki alt kanal, birincil ve ikincil SCH'dan oluşur. Birincil SCH 256 chip uzunluğunda modüle edilmiş bir koddan oluşur, Şekil 2.13'de ise c_p ile ifade edilen birincil senkronizasyon kodu (primary synchronization code, PSC) her yuvada iletilir. PSC sistemdeki her hücre için aynıdır (ETSI TS 125 211).

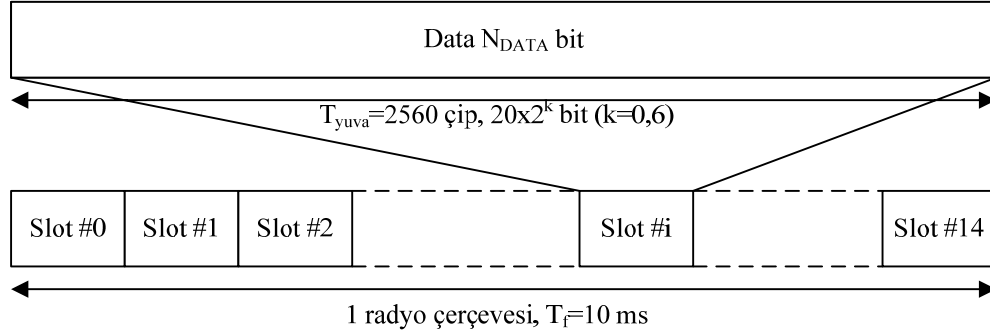


Şekil 2.13: Senkronizasyon kanal yapısı

İkincil SCH, 15 yuva ile tekrarlanan 256 chip uzunluğunda ardışık modüle edilmiş iletim kodu ile birincil SCH ve paralel iletilen ikincil senkronizasyon kodlarından (SSC) oluşur. SSC $c_s^{i,k}$ ile ifade edilir, burada $i=1, 2, \dots, 64$ yayılım kod grubunu, ve $k=0, 1, 2, \dots, 14$ yuva sayısını verir. Her SSC 25 uzunluğundaki 16 farklı kod setinden seçilir. İkincil SCH üzerindeki bu sıralama hücrenin DL yayılım kodunun hangi kod grubuna ait olduğunu gösterir.

Fiziksel DL paylaşım kanalı, DL paylaşım bilgisini taşımak için kullanılır. Kod çoğullamayı temel alan kullanıcılar tarafından paylaşılır. PDSCH yapısı Şekil

2.14’de gösterilir. DSCH her zaman DCH ile bağlantılı olduğundan, PDSCH de her zaman bir DL DPCCH ile bağlantılıdır. PDSCH yayılım faktörleri 256’den 4 e kadar değişebilir. Yayılım faktörü ve diğer fiziksel katman parametreleri sadece çerçeve temelinde değişiyorsa, TFCI PDSCH’nin anlık parametrelerini MS’i bilgilendirmek için kullanılır (ETSI TS 125 211).



Şekil 2.14: PDSCH çerçeve yapısı

Acquisition gösterge kanalı (AICH) PRACH ya da PCPCH üzerindeki Acquisition göstergelerini taşımak üzere kullanılan bir fiziksel kanaldır. Sayfa gösterge kanalı (PICH) sayfa göstergelerini taşıyan sabit hızlı ($SF=256$) bir fiziksel kanaldır. PICH her zaman bir PCH iletim kanalının gösterildiği bir ikincil CCPCH ile bağlantılıdır.

2.6. Çoklu Oranlı Kullanıcı Veri İletimi

WCDMA, farklı veri hızları ve farklı özellikte servis parametresi kullanan farklı türlerdeki servislerin iletimine olanak sağlayan esnek çoklu oranlı iletim şemasına sahiptir. Örneğin, kanal kodlama türü, boşluk verme derinliği ve veri hızı istenilen servis özelliğine göre değişebilir.

Şekil 2.15 ve Şekil 2.16 sırasıyla UL ve DL için çoklu oranlı iletim ve çoğullama şemalarını gösterir. İletim kanalı verisi şifrelenmiştir, bu yüzden fiziksel kanallara eşleştirilir ve radyo iletim bağı üzerinden iletilir. Kanal kodlama şeması hata izleme, hata düzeltme, hız eşleştirme, boşluk verme ve fiziksel kanala haritalanan iletim kanallarının bir birleşimidir (ETSI TS 125 212).

Veri, iletim kanalının özgün ve 10, 20, 40, ya da 80 ms olabilen her iletim süresi aralığında taşıma blok setleri halinde kodlama/çoğullama birimine ulaşır. Çoklu oranlı iletim aşağıdaki basamaklardan oluşur.

- Her iletim bloğuna döngüsel yedekleme kontrolünün (CRC) eklenmesi
- İletim bloğunun birleştirilmesi ve kod bloğunun bölünmesi
- Kanal kodlama
- Hız eşleştirme
- Süreksiz iletim (DTX) göstergesi bitlerinin eklenmesi
- Boşluk verme
- Radyo çerçevelerinin bölünmesi
- İletim kanallarının çoğullanması
- Fiziksel kanalların bölünmesi
- Fiziksel kanallara eşleştirme

İletim bloklarında hata izleme CRC yoluyla gerçekleştirilir. CRC 24, 16, 12, 8 ya da 0 bittir ve daha yüksek katman sinyali, her iletim kanalı için kullanılması gereken CRC uzunluğunun kaç olduğunu belirler.

CRC'nin ilave edilmesinden sonra, iletim bloğunun birleştirilmesi ve kod bloğunun bölünmesi gerçekleşir. Bütün iletim blokları seri olarak birleştirilir. İletim süresi aralığındaki bit sayısı kod bloğunda kullanılanın maksimum büyüklüğünden daha fazlaysa, iletim bloklarının birleştirilmesinden sonra kod bloğu bölünmesi gerçekleşir. Kod bloklarının maksimum büyüklüğü kullanılan katmanlı kodlama, turbo kodlama ya da kodlama kullanılmamasına bağlıdır. Maksimum kod bloğu büyüklükleri:

- Katmanlı kodlama: 504
- Turbo kodlama: 5114
- Kanal kodlaması yok: sınırsız

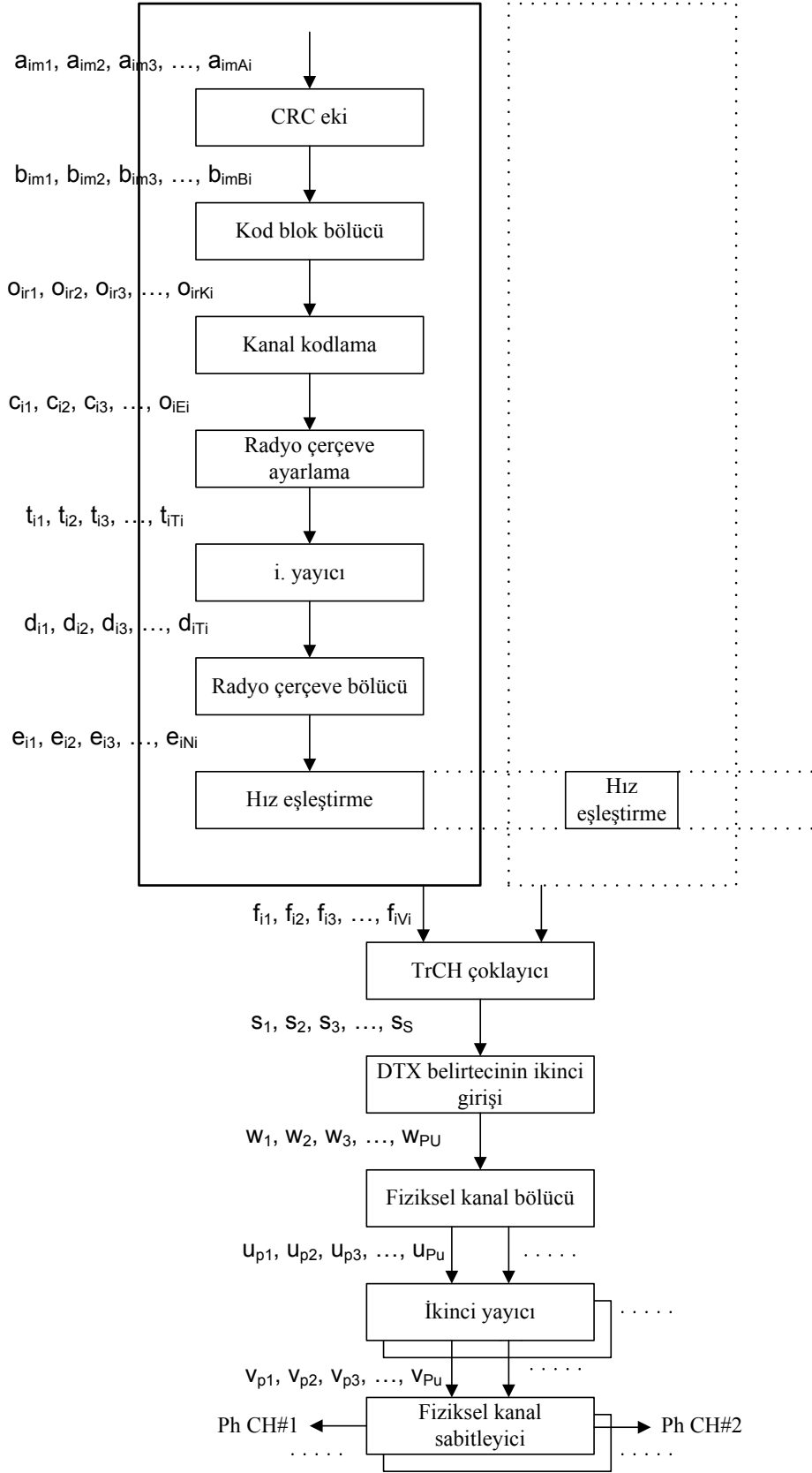
Radyo çerçevesi büyüklüğü denkleştirmesi, çıkışı aynı büyüklükte olan ardışık radyo çerçevelerine bölünebileceğini göstermek üzere, giriş bit sıralamasını doldurmaktır. Radyo çerçevesi büyüklüğü denkleştirmesi sadece UL'de gerçekleşir. DL'de hız eşleştirme çıkış bloğu uzunluğu radyo çerçevesi bölümü için zaten uygundur.

İletim süresi aralığı 10 ms'den uzun olduğunda, giriş biti sıralaması bölünür ve ardışık radyo çerçeveleri üzerine eşleştirilir. Bu, spektrum etkinliğini geliştirerek birçok radyo çerçevesi üzerinden iletme olanak sağlar.

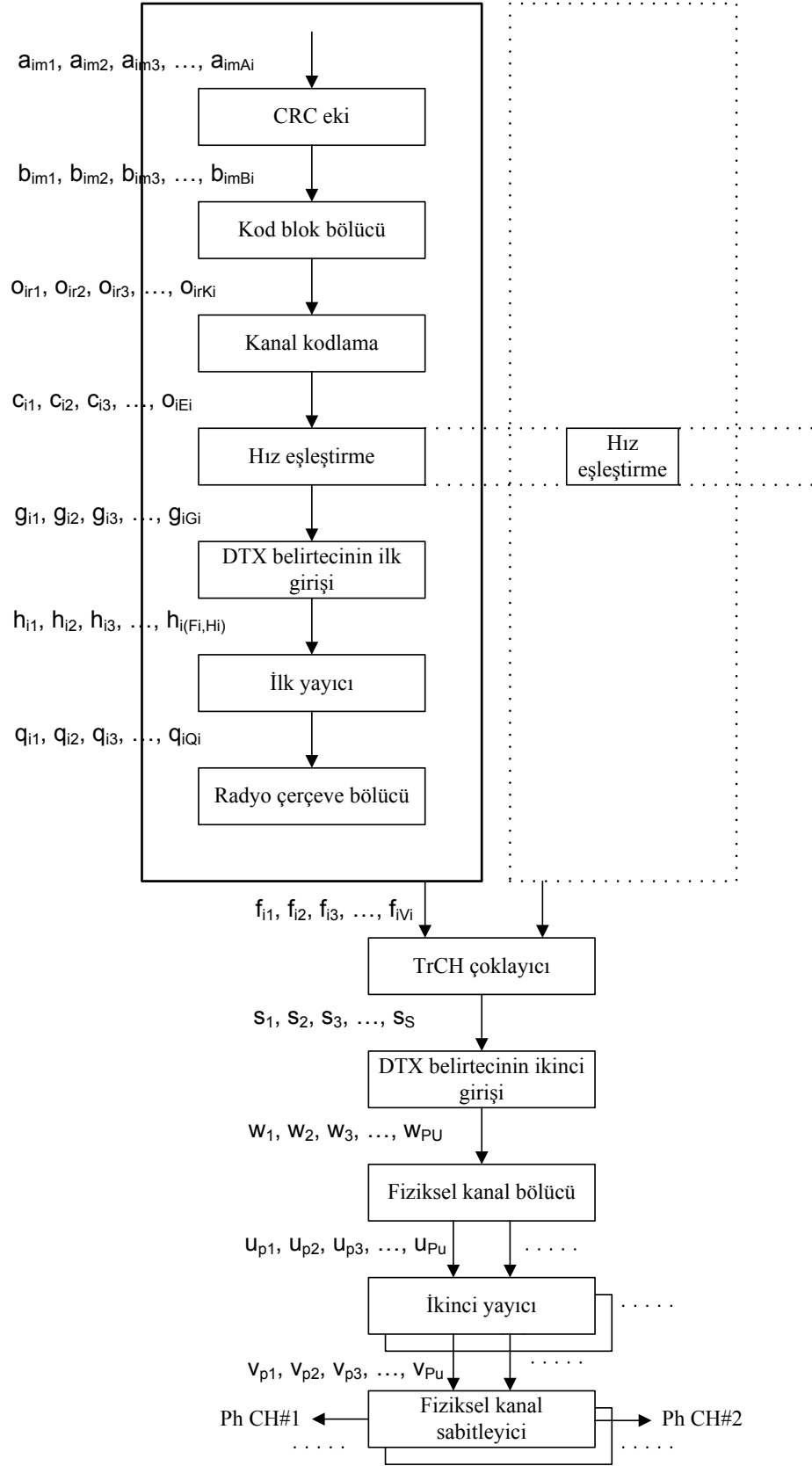
WCDMA esnek veri hızları sağladığından, iletim kanalı üzerindeki bitlerin sayısı farklı iletim süresi aralıkları arasında değişebilir. Hız eşleştirme bu sembol hızı sonucunu, fiziksel bir kanalın olası sınırlı sembol hızı serisine uyarlar. Hız eşleştirme iletim kanalındaki bitlerin, yarı-durağan olan ve sadece daha yüksek katmanlı sinyalleme yoluyla değişebilen hız eşleştirme özneliğine göre tekrarlandığı ya da delindiği anlamına gelir.

Bitlerin sayısı maksimumdan düşükse DL'de iletim kesilir (örneğin, DTX radyo çerçevelerini bitlerle doldurmak için kullanılır). DTX gösterge bitlerinin kesim noktası radyo çerçevesinde kullanılan iletim kanallarının sabit veya esnek pozisyonuna bağlıdır. Bağlantı sırasında her iletim kanalı için sabit ya da esnek pozisyonlardan hangisinin kullanıldığı ağa bağlıdır. DTX gösterge bitleri sadece iletimin ne zaman kesileceğini gösterir, iletilmezler.

Sonucu iletmek için bir ya da daha fazla fiziksel kanal kullanılabilir. Birden fazla fiziksel kanal kullanıldığında fiziksel kanal bölümü, bitleri farklı kanallar boyunca böler. İkinci boşluk vermeden sonra, fiziksel kanal eşleştirme gerçekleşir (ETSI TS 125 212).



Şekil 2.15: UL için çoklu oranlı bilgi transfer şeması



Şekil 2.16: DL için çoklu oranlı bilgi transfer şeması

2.6.1. İletim formatı takibi

İletim formatı takibi, iletim formatı birleşimi göstergeli (TFCI) veya göstergesiz olarak gerçekleştirilebilir. Bir TFCI iletilirse, alıcı iletim formatı birleşimini TFCI'dan takip eder. TFCI iletilmediğinde, sözde kör iletim format takibi kullanılabilir (yani, alıcı kısmı iletim formatı birleşimini bazı bilgileri, örneğin DPDCH nin DPCCH veya CRC kontrol sonuçlarının güç oranından elde edilen, kullanır).

2.6.2. Kanal kodlama

Tablo 2.7. farklı iletim kanalı türlerinin kanal kodlama parametrelerini listeler. Aşağıdaki kanal kodlama şemaları uygulanabilir:

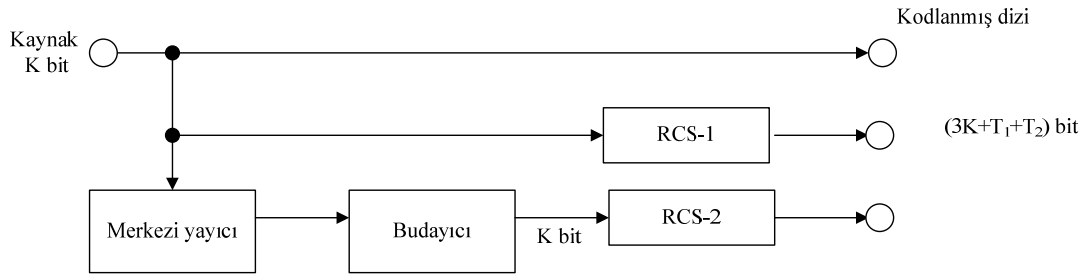
- 9 uzunluk ve 1/3 ya da 1/2 kodlama oranı ile sınırlı katmanlı kodlama
- Turbo kodlama
- Kanal kodlaması yok

Tablo 2.7: Hata düzeltme kodlama parametreleri

İletim kanal tipi	Kodlama tipi	Kodlama oranı
BCH	Katmanlı kodlama	1/2
PCH		
RACH		
CPCH, DCH, DSCH, FACH	Turbo kodlama, Kodlama yok	1/3, 1/2

Turbo kodlama şeması sekiz-durum öge kodlayıcılı paralel bağlı katmanlı bir koddur (PCCC). Kaydırmalı kaydedicinin başlangıç değeri PCCC kodlayıcısının hepsinin sıfır olması gerektiğini gösterir. PCCC kodlayıcısının çıkışı istenen kod hızı ile ilgili kodlanmış bitler üretmek üzere delinir. 1/3 oranı için, sistematik veya parity olan bitlerin hiçbirisi delinmez (ETSI TS 125 212).

Şekil 2.17 turbo kod iç boşluk vericisi ile birlikte bütün sekiz-durum PCCC turbo kodlama şemasını gösterir. Turbo kod iç boşluk vericisi ana boşluk verici üretimi ve budamadan oluşur. Gelişi güzel verilen K blok uzunluğu için, 134 ana boşluk verici setinden bir ana boşluk verici seçilir. Ana boşluk verici üretiminden sonra, bitler ana boşluk vericiyi K blok uzunluğuna ayarlamak üzere budanır. T_1 ve T_2 kuyruk bitleri sırasıyla RSC-1 ve RSC-2 kodlayıcı ögeleri için ilave edilir (ETSI TS 125 212).



Şekil 2.17: Bütün sekiz durumlu PCCC turbo kodlama

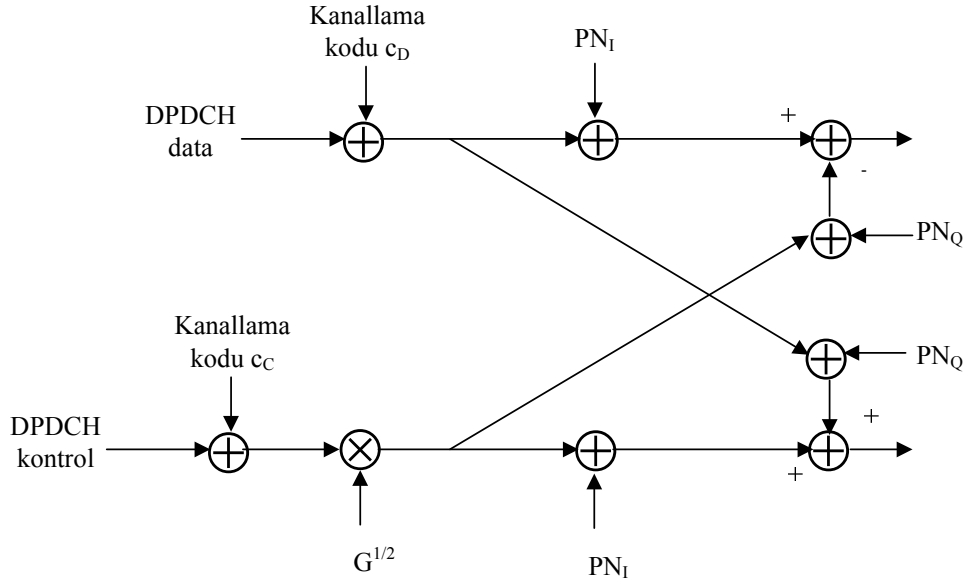
2.7. Yayılım ve Modülasyon

WCDMA, OVSF kodları ve sözde rastlantısal yayılım kodları iki katmanlı bir kod yapısını uygular. Yayılım bütün veri sembollerini belirli bir sayıda çipe dönüştüren kanallama kodları kullanılarak gerçekleştirilir, böylece sinyalin bandgenişliği artar. Yayılım, UL'de kullanıcı ayrımı ve DL'de hücre ayrımı için kullanılır (ETSI TS 125 213).

2.7.1. UL yayılımı

UL'de hem kısa hem de uzun yayılım kodları kullanılır. Kısa kodlar ileri seviyede çok kullanıcılı alıcı tekniklerinin uygulanmasını kolaylaştırmak için kullanılır; aksi takdirde, uzun yayılım kodları kullanılabilir. Kısa kodlar 256 uzunluğundaki S(2) kodları ve uzun kodlar 2^{41} uzunluğundaki Gold kodları olarak tanımlanabilir, fakat Gold kodlar 10 ms'lik bir çerçeve döngüsü oluşturmak üzere kesilmiştir.

UL’de kullanılan I-Q/kod çoğullama iki kanalın paralel iletimini sağlar ve bu nedenle module edilmiş sinyal gecikmesine ve ilgili tepe gücü ortalama güç oranına (tepe faktörü) dikkat edilmelidir. Şekil 2.18’de gösterilen karmaşık yayılım devresi kullanılarak, iletilen güç kuvvetlendirici (G) etkisi genel olarak QPSK iletimi ile aynı kalır (ETSI TS 125 213).

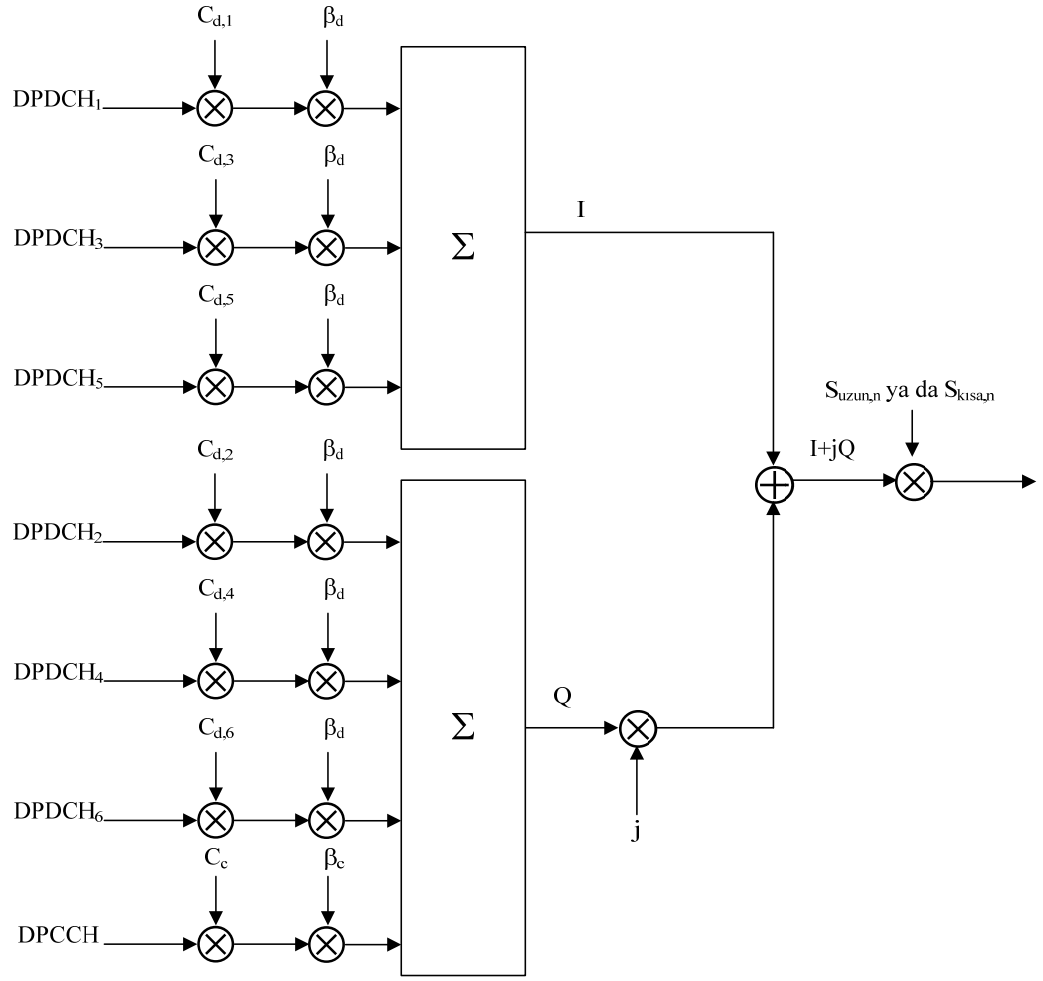


Şekil 2.18: Karmaşık yayılım devresi IQ/kod çoğullama

Ayrıca, DPDCH ve DPCCH arasındaki güç farkına bakılmaksızın sabit kalma durumu, sinyal zarfı değişimleri G 'nin tüm değerleri için QPSK iletimine benzer.

I-Q/kod karmaşık yayılım sonuçlarıyla çoğullama çözümü, güç arttırıcı çıkışında gerilemeye yol açar, böylece güç farkının bir fonksiyonu olarak etki yapar. Ayrıca, elde edilen çıkış gerilemesi bir QPSK sinyali için elde edilen gerileme ile aynıdır.

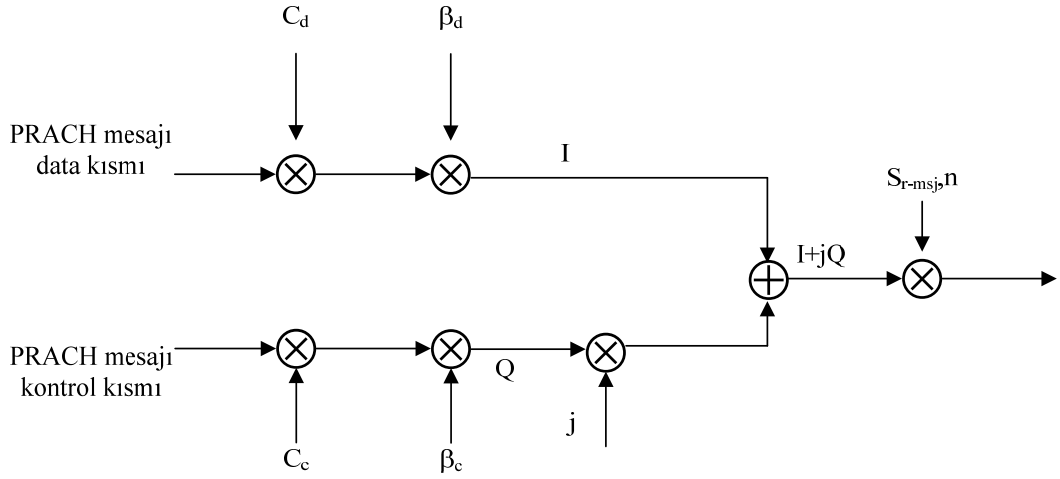
Şekil 2.19 farklı kanallama kodları yoluyla yayılan DPCCH ve DPDCH'lerin UL yayılımını gösterir. Bir DPCCH ve altıya kadar paralel DPDCH aynı anda iletilebilir. Kanallamadan sonra, gerçek-değerli yayılım sinyalleri DPCCH ve DPDCH için farklı olan fakat bütün DPDCH'ler için aynı olan kazanç faktörü ile ölçülür. Sinyalin gerçekten-karmaşığa dönüşmesinden sonra, uzun veya kısa olabilen karmaşık değerli karıştırma kodu ile karıştırılır (ETSI TS 125 213).



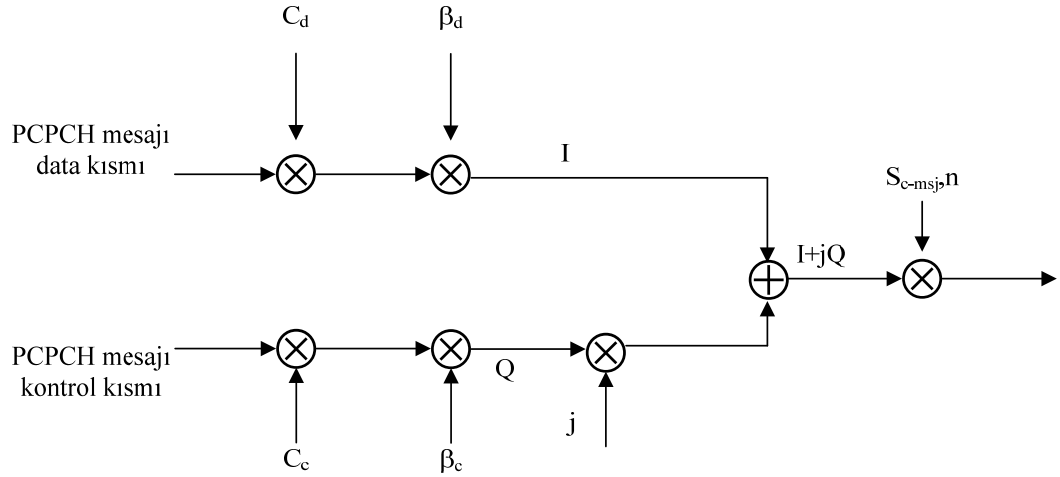
Şekil 2.19: UL DPCCH ve DPDCH'ler için yayılım

2.7.1.1. PRACH yayılımı

PRACH, başlık ve mesaj kısımlarından oluşur. Başlık karmaşık değerli bir koddur ve mesaj bölümü ise veri ve kontrol kısımlarından oluşur. Şekil 2.20 PRACH'nin mesaj bölümünün yayılım ve karıştırmasını gösterir. Kanallamadan sonra, yayılım sinyalleri kazanç faktörü (β) ile çarpılır. Karıştırma 10 ms uzunluğunda karmaşık değerli karıştırma kodu (C) yoluyla gerçekleştirilir (ETSI TS 125 213).



Şekil 2.20: PRACH'nin mesaj kısmının yayılımı



Şekil 2.21: PCPCH'nin mesaj kısmının yayılımı

2.7.1.2. PCPCH yayılımı

PCPCH başlık ve mesaj kısımlarından oluşur. Başlık kısmı karmaşık değerli bir koddan, mesaj kısmı veri ve kontrol bilgilerinden oluşur. Şekil 2.21 PCPCH'nin mesaj kısmının yayılımını gösterir. Kontrol ve veri bilgileri farklı kanallama kodlarıyla yayılır. Kanallamadan sonra, yayılım sinyalleri kazanç faktörü ile ölçülür. Karıştırma 10 ms uzunluğunda karmaşık değerli karıştırma kodu yoluyla gerçekleştirilir (ETSI TS 125 213).

2.7.2. DL yayılımı

DL’de UL ile aynı dik kanallama kodları kullanılır. Karıştırma için 2^{18} uzunluğunda Gold kodlar kullanılır fakat 10 ms (örneğin, 384,000 çip) çerçeveli bir döngü oluşturmak için kesilir. Karmaşık değerli kod oluşturmak üzere I ve Q kanallarında farklı süre kaymalarına sahip aynı kesikli kod kullanılır.

$2^{18}-1$ karıştırma kodu oluşturmak mümkündür, fakat bunların sadece 8191’i kullanılır. Her hücreye bir birincil karıştırma kodu tahsis edilir. Hücre arama süresini kısaltmak için birincil karıştırma kodları 512 sete bölünür. Böylece, MS’nin maksimum 512×10 ms uzunluğundaki kodları araması gerekir.

Birincil karıştırma kodlarına ek olarak, 15 ikincil karıştırma kod seti vardır. İkincil karıştırma kod setleri dik kanallama kodları yeterli olmadığında kullanılır. Bu uyarlanabilir antenler kullanıldığında olabilir. İkincil karıştırma kodunun, kod kanalları arasındaki dikliğe zarar verdiğine dikkat edilmelidir. Bununla birlikte uyarlanabilir antenlerle sunulan, uzaysal yalıtım ile dengelenmiştir.

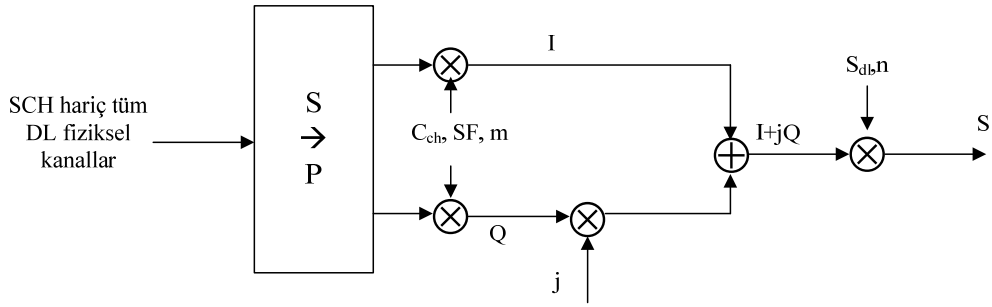
Birincil CCPCH ve birincil CPICH her zaman birincil karıştırma kodu kullanılarak iletilir. Diğer DL fiziksel kanalı ya birincil karıştırma kodu ya da hücrenin birincil karıştırma kodu ile bağlantılı setin ikincil karıştırma kodu ile iletilebilir.

Şekil 2.22 SCH hariç bütün DL fiziksel kanalların yayılım işlemlerini gösterir. Yayılmayan fiziksel kanal gerçek değerli sembollerin sıralamasından oluşur. AICH hariç tüm kanallar için semboller üç değer alabilir +1, -1, ve 0, 0 DTX’i gösterir. Her AICH için, sembol değerleri iletilecek acquisition göstergelerinin tam birleşimine bağlıdır.

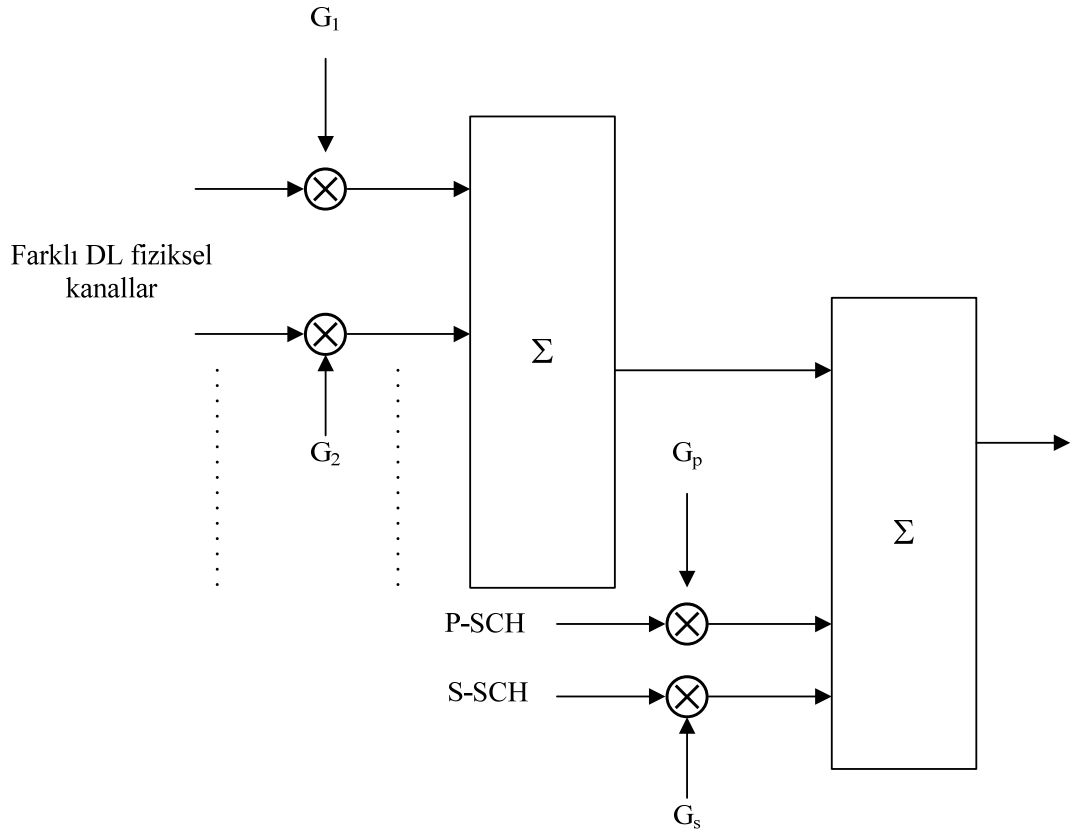
Her iki ardışık sembol çifti önce seriden paralele dönüştürülür ve ardından I ve Q dallarına eşleştirilir. Tek ve çift numaralandırıldığında sırasıyla I ve Q dallarına eşlenir. AICH hariç tüm kanallar için sıfır sembolü her çerçevedeki ilk sembol olarak tanımlanır. AICH için 0 sembolü her erişim yuvasındaki ilk sembol olarak tanımlanır. *I* ve *Q* dalları daha sonra aynı $C_{ch,SF,m}$ gerçek değerli kanallama kodu

yoluyla $\hat{c}$ hızına yayılır. I ve Q dallarındaki ger $\hat{c}$ ek deęerli $\hat{c}$ iplerin dizilişı, daha sonra $\hat{c}$ iplerin tek karmaşıık deęerli dizilişı olarak kabul edilir. $\hat{c}$ iplerin bu dizilişı karmaşıık deęerli karışırma kodu $S_{dl,n}$ yoluyla karışırılır. P-CCPCH durumunda karışırma kodu P-CCPCH $\hat{c}$ er $\hat{c}$ eve sınırı ile hizalanır (örneęin, yayılan P-CCPCH $\hat{c}$ er $\hat{c}$ evesinin ilk karmaşıık $\hat{c}$ ipi, karışırma kodunun sıfır $\hat{c}$ ip sayısı ile $\hat{c}$ arpılır). Dięer DL kanalları durumunda, karışırma kodu P-CCPCH'ye uygulanan karışırma kodu ile hizalanır. Bu durumda karışırma kodunun karışırılacak fiziksel kanalın $\hat{c}$ er $\hat{c}$ eve sınırı ile hizalanması gerekmez.

Şekil 2.23 farklı DL kanallarının nasıl birleştirildięini gösterir. Şekil 2.22'de S noktasına karşıılık gelen her karmaşıık deęerli yayılım kanalı bir aęırlık faktörü G_t ile ayrı olarak aęırlıklandırılır. Bütün DL fiziksel kanallar daha sonra karmaşıık ekleme kullanılarak birleştirilir (ETSI TS 125 213).



Şekil 2.22: SCH hariç bütün DL fiziksel kanalların yayılımı



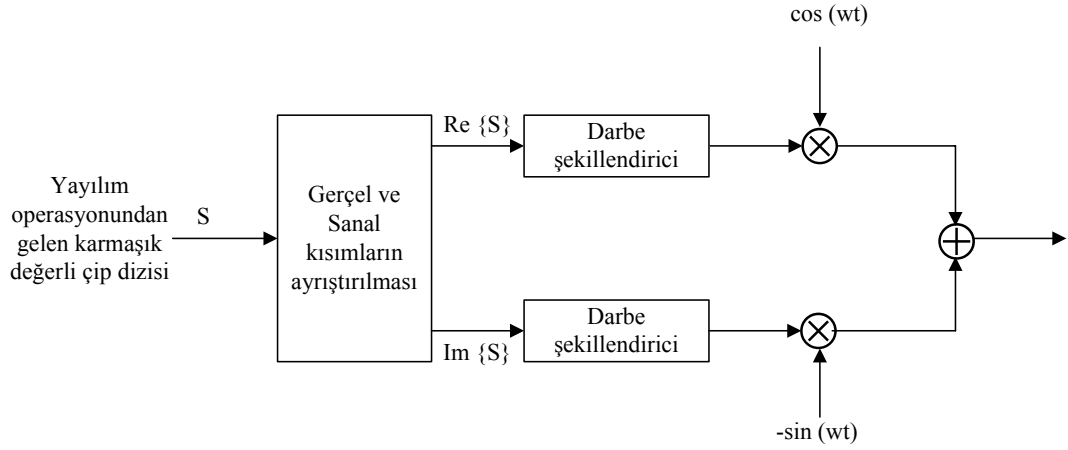
Şekil 2.23: SCH ve P-CCPCH'nin yayılımı ve modülasyon

2.7.2.1. Senkronizasyon kodları

Birincil senkronizasyon kodu (PSC), CPSC genelleştirilmiş hiyerarşik Golay dizilimi gibi yapılanmıştır. PSC ayrıca, düzensiz otomatik korelasyon özellikleri elde etmek için seçilir.

2.7.2.2. Modülasyon

Karmaşık değerli chip sıralaması QPSK yayılım işlemiyle oluşur. Şekil 2.24 UL ve DL'de kullanılan modülasyon prensibini gösterir. Darbe şekillendirme 0.22 azalma faktörü ile root-raised cosine'dir. Bu yapı mobil ve baz istasyonları için aynıdır (ETSI TS 125 213).



Şekil 2.24: Modülasyon prensibi

Tablo 2.8: DL fiziksel kanallarında TX çeşitleme modlarının uygulanması

Kanal	Açık çevrim mod		Kapalı çevrim mod
	TSTD	STTD	
P-CCPCH		X	
SCH	X		
S-CCPCH		X	
DPCH		X	X
PICH		X	
PDSCH (DPCH ile sunulur)		X	X
AICH		X	

2.8. Verici Çeşitlemesi

WCDMA'nın DL'de iki tür çeşitleme mekanizması kullanılabilir. Bunlar açık çevrim ve kapalı çevrimdir. Açık çevrim DL verici çeşitlemesi olarak uzay/zaman blok kodlama temelli verici çeşitlemesi (STTD) uygular. STTD şifreleme BTS'de isteğe bağlı olmasına rağmen, MS'de zorunludur. Diğer bir açık döngü verici çeşitlemesi olan zaman anahtarlama verici çeşitlemesi (TSTD), SCH kanalına uygulanabilir. Bu çeşitleme mekanizması da BTS'de isteğe bağlıdır, fakat MS'de zorunludur. Her çeşitleme yöntemi her fiziksel kanalla kullanılamaz. Tablo 2.8 farklı DL fiziksel kanallarındaki açık ve kapalı döngü verici çeşitleme modlarının olası uygulamalarını özetler. STTD'nin eş zamanlı kullanımı ve DPCH ve PDSCH üzerindeki kapalı döngü modları izinli değildir.

2.9. Hava Arayüzü Prosedürleri

Bir radyo sisteminin çalışması için farklı hava arayüzü prosedürleri gerekir (örneğin, bağlantı sağlamak ve radyo kaynaklarının minimum tüketim ile sürdürmek). Bu bölümde aşağıdaki ara yüz prosedürleri ele alınmaktadır:

- Hücre arama işlemi
- Aktarım
- Güç kontrolü
- UL senkronize iletim şeması (USTS)

2.9.1. DL yayılımı

DL araması boyunca, MS için bir hücrede arama yapılır ve DL karıştırma kodu ve o çerçevenin ortak kanal çerçeve senkronizasyonunu belirlenir. Tüm ortak fiziksel kanalların radyo çerçeve zamanlaması P-CCPCH'nin zamanlaması ile ilgili olduğundan, sadece P-CCPCH'nin zamanlamasını bulmak yeterlidir. Hücre araması genel olarak üç basamakta gerçekleşir. Bunlar, yuva senkronizasyonu, çerçeve senkronizasyonu ve kod grubu tanımlaması ve karıştırma kodu tanımlamasıdır. 3GPP şartnamesi TS25.214'ün örnek bir prosedürü aşağıdaki gibi açıklanır (ETSI TS 125 214):

- Yuva Senkronizasyonu: Hücre arama prosedürünün ilk basamağı boyunca MS hücrede yuva senkronizasyonunu elde etmek için SCH'nin birincil senkronizasyon kodunu kullanır. Bu bütün hücrelerde ortak olan birincil senkronizasyon koduna uyan tek bir uyumlu filtre ile yapılabilir. Hücrenin yuva zamanlaması uyumlu filtre çıkışındaki tepe noktalar izlenerek elde edilebilir.
- Çerçeve senkronizasyonu ve kod grubu tanımlaması: Hücre arama prosedürünün ikinci basamağında, MS birinci basamakta bulunan hücrenin çerçeve senkronizasyonunu bulmak ve kod grubunu tanımlamak için SCH'nin ikincil senkronizasyon kodunu kullanır. Bu, alınan sinyallerin tüm olası ikincil senkronizasyon kodu sıralamaları ile ilişkilendirme ve maksimum korelasyon

değerini tanımlama yoluyla yapılabilir. Sıralamaların döngüsel kayması benzersiz olduğundan, kod grubu ve çerçeve senkronizasyonu belirlenir.

- Karıştırma kodunun tanımlanması: Hücre araması prosedürünün üçüncü ve son basamağında, MS hücre tarafından bulunan tam birincil karıştırma kodunu belirler. Birincil karıştırma kodu tipik olarak ikinci basamakta tanımlanan tüm kodlar arasındaki kod grubu ile CPICH üzerinden sembol korelasyonu yoluyla tanımlanır. Birincil karıştırma kodu tanımlandıktan sonra, birincil CCPCH izlenebilir ve sistem hücreye özgü BCH bilgisi okunabilir.

MS durumunda yukarıdaki 2 ve 3. basamaklarda aranan karıştırma kodları hakkında alınan bilgi sadeleştirilir.

2.9.2. Hücreler arası geçiş

WCDMA birçok farklı türde geçişe sahiptir.

- Yumuşak, daha yumuşak ve sert hücreler arası geçiş
- Frekanslar arası aktarım
- FDD ve TDD modları arasında aktarım
- WCDMA ve GSM arasında aktarım

Hücreler arası geçiş (Handover) algoritması, hücreler arası geçiş kararını vermek için farklı türlerde ölçüm bilgilerine ihtiyaç duyar. Tablo 2.9 hücreler arası geçiş amaçlı gerçekleştirilebilecek ölçümleri listeler. Gerçek hücreler arası geçiş algoritma uyarlaması cihaz üreticilerine bırakılmıştır (ETSI TS 125 215).

WCDMA'daki BTS'lerin senkronize olması gerekir ve böylece BTS için, GPS gibi, hiçbir dış senkronizasyon kaynağı gerekmez. Asenkron BTS'ler, hücreler arası geçiş algoritmaları tasarlanırken ve konum bilgisi servisleri de değerlendirilmelidir.

Tablo 2.9: Hücreler arası geçiş ölçümleri

Alınan sinyal kodu gücü (RSCP)	Bir kodda alınan güç Birincil CPICH'nin pilot bitleri üzerinde ölçülür. RSCP nin referans noktası MS'deki anten konnektörüdür.
TDD alınan sinyal kodu gücü	Bir kodda alınan güç bir TDD hücresinden PCCPCH'de ölçülür. RSCP nin referans noktası MS'deki anten konnektörüdür.
Radyo bağlantısı birleşiminden sonra alınan sinyal kodu gücü	Bir kodda alınan güç radyo bağlantı birleşiminden sonra DPCCCH pilot bitlerinde ölçülür. RSCP'nin referans noktası MS'deki anten konnektörüdür.
SIR	Sinyal parazit oranı, $(RCP/ISCP) \times (SF/2)$ olarak tanımlanır. Burada ISCP, parazit sinyal kodu gücü pilot bitlerde alınan sinyalde ölçülen parazittir. Sadece parazitin dik olmayan kısmının paraziti ölçümde bulunur. SF, kullanılan yayılım faktörüdür. SIR RL birleşiminden sonra DPCCCH'de ölçülmelidir. SIR'nin referans noktası MS'nin anten konnektörüdür.
Alınan sinyal güç belirteci (RSSI)	Alınan sinyal güç belirteci, ilgili kanal bant genişliğindeki alınan geniş bant gücü olarak tanımlanır. Ölçümler DL taşıyıcı üzerinde gerçekleştirilmelidir. RSSI'nin referans noktası MS'de anten konnektörüdür.
GSM taşıyıcı RSSI	Alınan sinyal gücü göstergesi, ilgili kanal bant genişliğindeki alınan geniş bant gücü olarak tanımlanır. Ölçümler GSM BCCH taşıyıcı üzerinde gerçekleştirilmelidir. RSSI'nin referans noktası MS'deki anten konnektörüdür.
CPICH E_c/N_0	Çip başına alınan enerji bandındaki güç yoğunluğuna bölünür. E_c/N_0 RSCP/RSSI ile aynıdır. Ölçümler birincil CPICH'de gerçekleştirilmelidir. E_c/N_0 'nun referans noktası MS'deki anten konnektörüdür.
İletim kanalı blok hata oranı (BLER)	İletim kanalının blok hatası hızının (BLER) kestirimi. BLER kestirimi radyo bağlantı birleşiminden sonra her iletim bloğu üzerindeki CRC'nin değerlendirmesine dayanmalıdır. Dış döngüdeki güç kontrolünün hızlı güç kontrolü için SIR'da hedef ayarlamasında kullanılır.
Fiziksel kanal bit hata oranı (BER)	Fiziksel kanal bit hata hızı (BER) radyo bağlantı birleşiminden sonra DPDCH verilerinin kanal deşifresinden önce ortalama bit hata hızının bir kestirimidir.
Mobil istasyon iletim gücü	Toplam MS gücü bir taşıyıcı üzerinden iletir. UE'nin iletildiği gücün referans noktası MS deki anten konnektörü olabilir

Hücreler arası geçişe girmeden önce, MS DL SCH iki BTS'den elde edilen zamanlama farklılıklarını ölçer. MS zamanlama farklılıklarını çalışan BTS'e geri bildirir. Yeni bir DL hücreler arası geçiş bağlantısının zamanlaması, bir sembolün çözünürlüğü ile ayarlanır (örn., iki BTS'den alınan adanmış DL sinyalleri bir

sembolün doğruluğu ile senkronizedir). Bu, mobil RAKE alıcısının iki BTS'den alınan makro çeşitleme enerjisini toplanmasına olanak sağlar. Adanan DL kanallarının zamanlama ayarları bir sembolün çözünürlüğü ile DL kodlarının dikliği kaybedilmeden yürütülebilir.

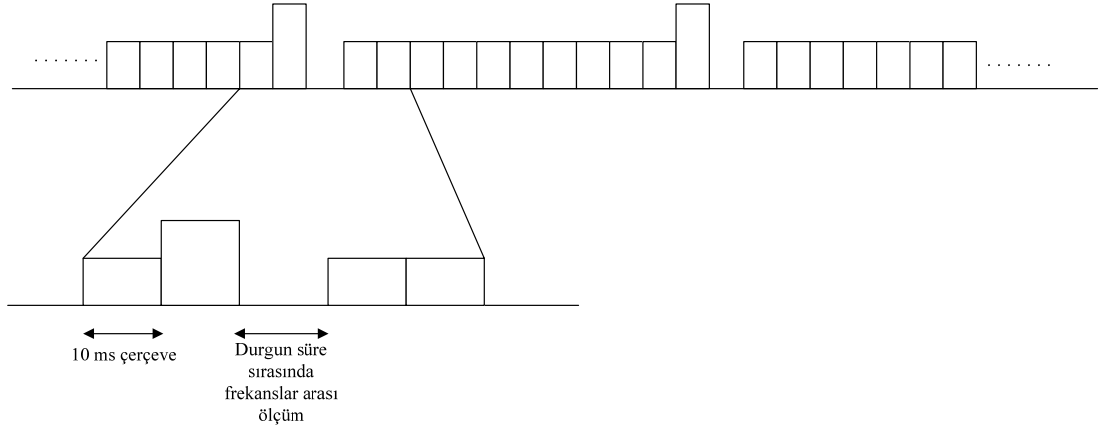
2.9.2.1. Frekanslar arası aktarımlar

Frekanslar arası aktarımlar hiyerarşik hücre yapılarının kullanımı için gereklidir. Hiyerarşik hücre yapıları, makro, mikro ve bina içi hücrelerdir. Birçok taşıyıcı ve frekanslar arası aktarımlar, etkin noktadaki yüksek kapasiteyi korumak için de ayrıca kullanılabilir. Frekanslar arası aktarımlar ayrıca GSM ya da IS-95 gibi ikinci nesil aktarımlar için de gerekir (ETSI TS 125 215). Frekanslar arası aktarımları tamamlamak üzere, kullanılan mevcut frekansta bağlantı korunurken diğer frekanslarda da ölçüm yapabilmek için etkili yöntemler gerekir. WCDMA da frekanslar arası ölçümler için iki yöntem değerlendirilmiştir:

- Çift bantlı alıcı
- Sıkıştırılmış mod

Çift bantlı alıcı yaklaşımı özellikle, mobil terminal, anten çeşitlemesi uyguluyorsa uygun olarak değerlendirilebilir. Frekanslar arası ölçümler sırasında, ölçümler için bir alıcı dalı diğer bir frekansa bağlanırken, diğeri mevcut frekansı almaya devam eder. Ölçümler sırasındaki kazanç çeşitlemesi kaybının daha yüksek DL iletim gücü ile düzeltilmesi gerekir. Dual alıcı yaklaşımının avantajı, mevcut frekans bağlantısında herhangi bir kesinti olmamasıdır. Hızlı kapalı döngü güç kontrolü her zaman çalışır.

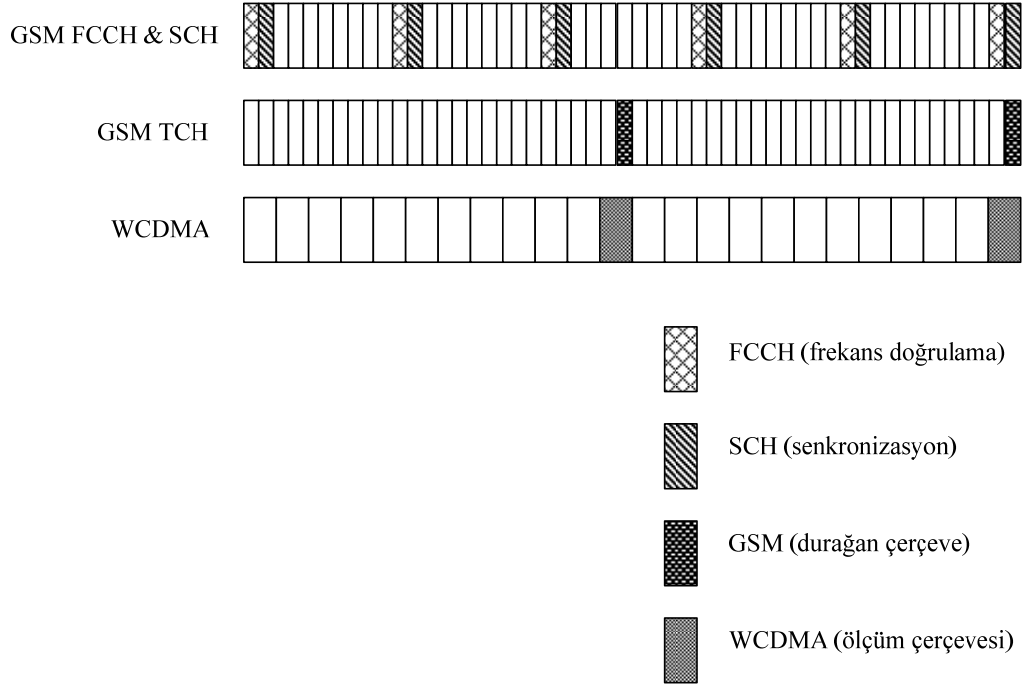
Şekil 2.25'te verilen sıkıştırılmış mod yaklaşımı (yuvalanmış mod olarak da adlandırılır) çift bantlı alıcısı olmayan MS'ler için gereklidir. 10 ms bir çerçeve sırasında iletilen bilgi normal olarak ya kod delinerek ya da FEC hızı değiştirilerek sıkıştırılır.



Şekil 2.25: Sıkıştırılmış mod yapısı

2.9.2.2. GSM ve WCDMA arası aktarımlar

WCDMA sistemi ve GSM sistemi arasındaki aktarım, günümüzde tüm dünyayı kapsamayı hedefleyen, WCDMA çerçeve zamanlaması tanımında dikkate alınan temel tasarım kriterlerinden biridir. GSM uyumlu çoklu çerçeve yapısı ve süper çerçeve yapısı, GSM sisteminde olduğu gibi sistemler arası ölçümler için benzer zamanlamaya olanak sağlar. Görünüşe bakılırsa, sistemler arası aktarım sistem içi parazit noktasından daha az önemli olduğundan, gereken ölçüm aralığının bir GSM sisteminde kullanılan GSM terminali kadar sık olması gerekmez. Aksine, zamanlamanın uyumluluğu önemli olduğundan, WCDMA’da çalışılırken, çok modlu bir terminal GSM taşıyıcı üzerindeki senkronizasyon çerçevesinde bulunan senkronizasyon veri dizisinden istenilen bilgiyi bir frekans düzeltme veri dizisinin yardımıyla yakalayabilir. Bu yolla bir GSM ve WCDMA taşıyıcıları arasındaki bağlı zamanlama iki asenkronize GSM taşıyıcısı arasındaki zamanlamaya benzer olarak elde edilir. WCDMA kanalları ve GSM kanalları arasındaki zamanlama ilişkisi Şekil 2.27’de verilmiştir (Holma ve Toksala, 2000).



Şekil 2.26: WCDMA ve GSM çerçeve yapıları arasındaki ölçüm zamanlaması ilişkisi

Burada, GSM trafik kanalı ve WCDMA kanalları benzer 120 ms çoklu çerçeve yapısını kullanırlar. GSM frekans düzeltme kanalı (FCCH) ve GSM senkronizasyon kanalı (SCH) çerçevelerde verilen sekiz GSM yuvasından birini kullanır. Şekil 2.26'da verildiği gibi FCCH için FCCH çerçevesi ile birlikte bir zaman yuvası her zaman önceki SCH için SCH çerçevesi ile birlikte bir zaman yuvası kullanılır.

Bir WCDMA terminali ölçümleri ya DL iletimlerinde kesintilerin bulunduğu sıkıştırılmış moddaki ölçüm aralıklarını isteyerek ya da ölçümleri uygun bir ölçüm şekli ile bağımsız olarak gerçekleştirerek yapabilir. Sıkıştırılmış mod yerine bağımsız ölçümlerle çift bantlı alıcı yaklaşımı kullanılır çünkü GSM alıcı dalı WCDMA alıcı dalından bağımsız olarak çalışabilir.

Sistemler arasında sorunsuz karşılıklı çalışma için, WCDMA baz istasyonunun çalışılan alandaki mevcut GSM frekansı terminalini uyarmasına olanak sağlamak üzere sistemler arasında bilgi değişimi olmalıdır. Ayrıca, her yöne doğru 2 Mbps'e ulaşan UMTS veri hızlarına kıyasla GSM'deki düşük veri hızı kapasitesi dikkate alındığında mevcut servisin elde edildiği gerçek aktarım için daha entegre çalışma gereklidir. Beklenildiği gibi GSM sistemi alanda hücre tanımlamasını basitleştirmek

için WCDMA yayılım kodlarını gösterir. Daha sonra GSM’de bulunan ölçüm denemeleri GSM modunda çalışırken WCDMA’yı ölçmek için de kullanılabilir.

WCDMA, GSM’de olduğu gibi senkronizasyonu bulmak için herhangi bir süper çerçeve yapısını kullanmaz. Bu yüzden GSM modunda çalışan terminal WCDMA baz istasyonu yayılım kodu zamanlaması elde edildiğinde, WCDMA çerçeve senkronizasyonu bulunur. Baz istasyonu yayılım kodu 10 ms periyoduna sahiptir ve çerçeve zamanlaması WCDMA ortak kanalları ile senkronizedir.

2.9.3. Güç kontrolü

WCDMA hızlı kapalı çevrim ve açık çevrim güç kontrolü prosedürlerine sahiptir. Hızlı güç kontrolü 1.5 KHz hızla çalışır (yuva başına bir komut). Tanımlanmış güç kontrolü adımı büyüklüğü 1dB’dir, fakat tanımlanmış adım büyüklüklerinin katları da kullanılabilir. Güç kontrol komutları sadece her ikinci yuvadan gönderilebilir.

Hücreler arası geçiş durumunda, her komutun güvenilirliği dikkate alınarak farklı baz istasyonlarından güç kontrolü komutları birleştirilir. Sıkıştırılmış modda, hızlı güç kontrolü, daha hızlı yaklaşmak üzere sıkıştırılan bir çerçeveden sonra kısa bir periyot için daha büyük adım büyüklüğü kullanır. Açık döngü kontrolü RACH ya da CPCH de iletimi başlatmadan önce kullanılır. Güç kontrolünün kesinliğinin ± 9 dB olması gerekir.

2.9.4. UL senkronizasyon iletim şeması

UL senkronize iletim şeması (USTS), MS’e uygulanabilecek alternatif bir teknolojidir. USTS, UL hücre içi paraziti hücreye MS’den gelen dik sinyalleri aldirarak azaltabilir. MS’den gelen dik sinyalleri almak için:

- Bir hücrede adanan tüm fiziksel kanallara aynı yayılım kodları tahsis edilir

- Bir hücrede tüm MS'ler arasında adanan fiziksel kanalların hepsi farklı kanallama kodlarına tahsis edilir ve yayılım faktörü ve kanallama kodu kod sayısı ağdan her MS'ye ulaştırılır
- Bir MS'deki DPDCH ve DPCCH için kanallama kodları, tepe gücü ortalama güç oranını azaltmak üzere bir MS'deki OVSF kod ağacının hem üst yarı kısmı hem de alt yarı kısmından seçilebilir.
- Kanalizasyon kodları doluysa ek yayılım kodları tahsis edilebilir.
- Her MS'nin sinyal iletim süresi ayarlanır.

İletim süresi kontrolü iki adımda yürütülür. İlk adım başlangıç senkronizasyonu ve ikinci adım izleme işlemidir:

- Başlangıç senkronizasyonu: İletim süresini FACH üzerinden başlangıç süresi kontrol mesajı yoluyla ayarlamadır. Hücre MS'den RACH üzerinden gelen sinyali aldığı anda, hücre alınan süre ve referans süresi arasındaki zaman farkını $1/8$ chip süresi biriminde ölçer. Başlangıç senkronizasyonu için süredeki farkı içeren mesaj, MS'ye FACH yoluyla ulaştırılır.
- İzleme işlemi (kapalı döngü zamanlama kontrolü): İletim süresini DPCCH üzerinden zaman hizalama biti (TAB) yoluyla ayarlama. UE mesaja göre iletimini ayarlar. Hücre referans süresini MS'den alınan sinyal zamanlaması ile periyodik olarak karşılaştırır. Alınan sinyal zamanı referans süresinden önce ise, $TAB=0$ olur. Alınan süre referans süresinden sonra ise $TAB=1$ olur. TAB her 20 ms'lik süre kontrol periyodunda TPC ile yer değiştirir, her iki çerçevenin son TPC biti TAB ile yer değiştirir. MS'de ve TAB'da sıfır-bir kararı yürütülmelidir ve 0 olarak değerlendirildiğinde, iletim $1/8$ chip geciktirilirken, 1 olarak değerlendirilirse, iletim $1/8$ chip ilerletilir.

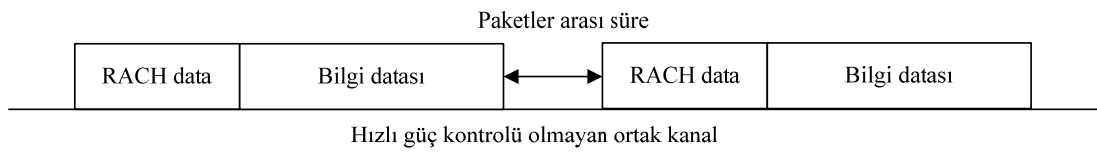
2.9.5. Paket datalar

WCDMA üç farklı türde paket data iletimi olasılığına sahiptir.

- Ortak kanallar
- Adanan kanallar
- Paylaşılan kanallar

Paket veri iletimi için ortak kanalların kullanılmasının avantajı çok kısa bağlantı kurma süresi olarak verilebilir. Ortak kanallar RACH (UL) ve FACH (DL) kısa paketler için kullanılır. Hücreler arası geçiş bu kanallar için kullanılamaz. Ayrıca, bu işlemde sadece açık döngü güç kontrolü vardır. Bu yüzden RACH ve FACH paket iletimi sadece sınırlı kapasite kullanan kısa paketlerle sınırlandırılmalıdır. Şekil 2.27 RACH üzerindeki paket iletimini gösterir (Holma ve Toksala, 2000).

Ortak paket kanalı (CPCH) kısa ve orta büyüklükteki veri paketlerini iletmek için kullanılır. Prosedür RACH'e benzer, fakat kanal farklı kullanıcılar arasındaki zaman bölümünde paylaşılabilir ve hızlı güç kontrolü kullanılabilir. Hücreler arası geçiş, CPCH için kullanılamaz.



Şekil 2.27: RACH kanalı üzerinde paket iletimi

Adanan bir kanalda geniş ya da daha sık paketler iletilir. Hücreler arası geçiş ve hızlı güç kontrolü kullanılır ve bu yüzden radyo performansı ortak kanallarda iletimden daha iyidir. Veri hızı iletim süresine göre değişebilir. Dolayısıyla, DL'deki OVFSF kodlar en yüksek bit hızına göre tahsis edilir. Eğer birden fazla yüksek hız paket erişim bağlantısı bulunursa, uygun OVFSF kodu yeterli olmayabilir.

OVFSF kodların bitmesi sorununun üstesinden gelmek için, bu kodların evrimli kullanımı ve optimizasyonu çok önemlidir.

2.10. WCDMA Gelişimi, Release 2000 ve Sonrası

WCDMA araştırma ve yeni gelişmelere dayanarak ilerlemeye devam ediyor. R-2000 özellikleri tüm IP çekirdeği ağı yoluyla servislerin provizyonuna fırsat veren etkili IP desteği sağlar. Başlık sıkıştırması ve servis kalitesini (QoS) de içeren teknik konular çözülecektir. Sonraki 3GPP standardı yaklaşık 10Mps'den daha da yüksek veri hızı sağlar. Yüksek-hızlı paket erişimi (HSPA) belirlemek için bir çalışma başlamıştır. Aşağıdaki geliştirmeler önerilir:

- Uyarlamalı modülasyon ve kodlama (AMCS);
- Hibrit otomatik tekrarlı isteği (HARQ)
- Daha küçük çerçeve büyüklüğü;
- Program mekanizmasının konumu;
- Hızlı hücre sitesi seçimi;
- Erişim kontrol kanalı ile bağlantılı UL DCH (ACCH)

3. KOD SINIRLI ALGORİTMALAR

3.1. Giriş

OVSF kod ağacı, WCDMA sistemlerde, farklı veri transfer hızı talebini desteklemek amacıyla, farklı bantgenişliklerinin sağlanması için kullanılır. Bir OVSF WCDMA sisteminde, her BTS DL veri transferi için bir kod ağacını yönetir. Birbirine dik kodların sayısı sınırlı olduğundan BTS, sistem performansını arttırmak için kod ağacının verimli bir şekilde kullanılmasından ve yönetiminden sorumludur. Birinci bölümde bahsedilen literatür çalışmalarında, OVSF kodlarının kullanımı ve kod ağacının yönetimi için en uygun kod tahsis yöntemleri araştırılmış ve kod tahsis algoritmaları, geleneksel kod tahsisi (CCA) (Okawa ve Adachi, 1998) ve dinamik kod tahsisi (DCA) (Minn ve Siu, 2000) ana başlıkları altında incelenmiştir. CCA algoritmasında gelen bir çağrı talebi için eğer OVSF kod ağacında uygun bir kod varsa bu kod tahsil edilir, aksi takdirde kapasite yeterli olsa bile çağrı reddedilir. CCA algoritmasında kod bloklanması çok önemli bir problemdir. Bu problemle başa çıkabilmek için araştırmalar OVSF kod ağacının dinamik olarak kullanımı üzerine yoğunlaşmıştır. DCA algoritmasında kapasitenin yeterli olduğu durumlarda yeni kullanıcının kod tahsis talebini karşılamak için kod ağacındaki aktif kullanıcılar çeşitli yöntemlerle yer değiştirirler.

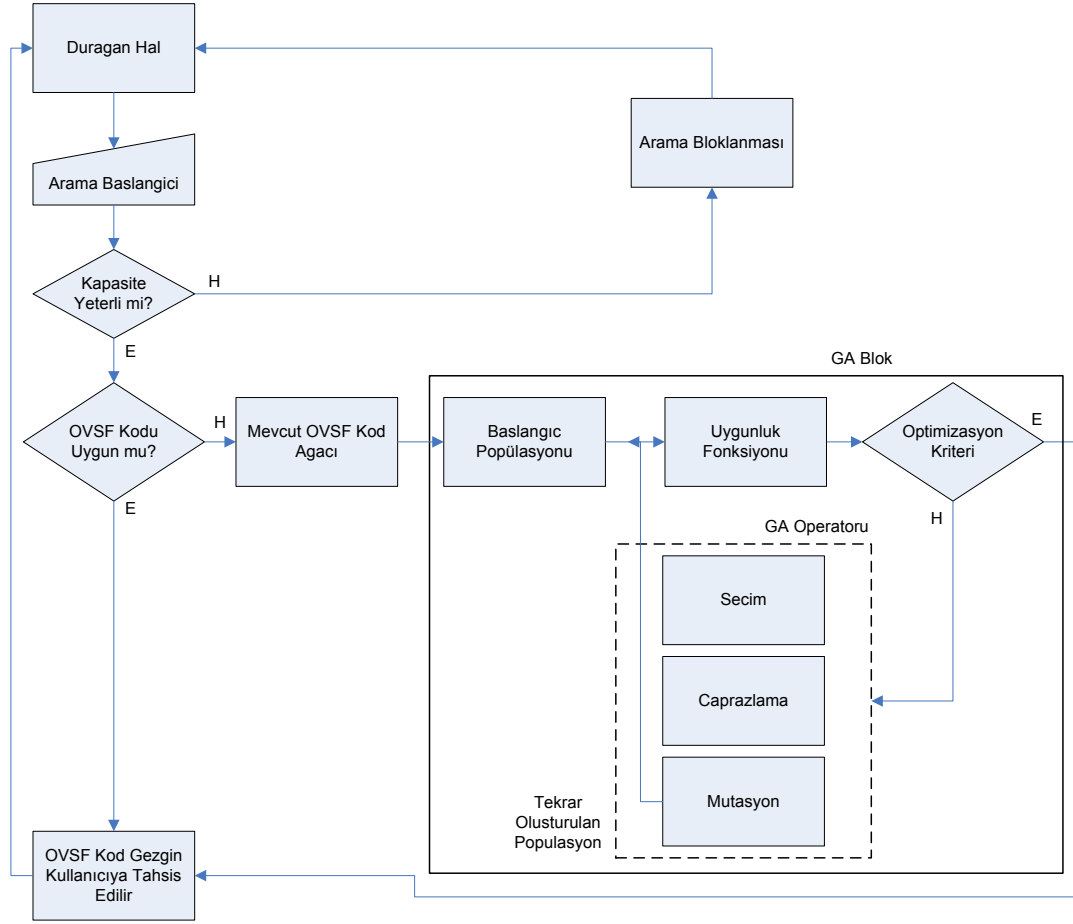
Bu bölümde; OVSF kod ağacının dinamik olarak yönetimi için mevcut kod ağacının trafik yoğunluğuna göre değişen bir popülasyon kullanan Genetik Algoritma (Genetic Algorithm, GA), Benzetilmiş Tavlama (Simulated Annealing, SA) ve Uyarlamalı Benzetilmiş Tavlamalı Genetik Algoritma (Adaptive Simulated Annealing Genetic Algorithm, ASAGA) yöntemleri ele alınmıştır. Ayrıca, basit GA'nın (simple GA, SGA) verimliliğini arttırabilmek ve hızlı değişen trafik yoğunluğuna uyum sağlayabilmek için “diploid” yapı uygulanmış ve bu amaçla

baskın (dominance) ve diploidy GA (dominance & diploidy GA, D&D-GA) tabanlı dinamik kod tahsis algoritması önerilmiştir.

3.2. Genetik Algoritma Tabanlı Dinamik OVFS Kod Tahsisi

GA'nın temel ilkeleri ilk kez Michigan Üniversitesinde John Holland tarafından ortaya atılmıştır. Holland (1975), daha sonra yaptığı çalışmaları “Adaptation in Natural and Artificial Systems” adlı kitabında bir araya getirmiştir. Holland ilk olarak evrim yasalarını GA içinde optimizasyon problemleri için kullanmıştır (Goldberg 1989). Bu algoritmalar daha sonra, optimizasyon problemleri için çok önemli bir araç haline gelmiştir. Günümüzde de GA pek çok değişik uygulamada kullanılmaktadır. Bunun nedeni, GA'nın problemlere tek bir çözüm üretmek yerine farklı çözümlerden oluşan bir çözüm kümesi üretmesidir. Böylelikle, arama uzayında aynı anda birçok nokta değerlendirilmekte ve sonuçta bütünsel çözüme ulaşma olasılığı yükselmektedir. Çözüm kümesindeki çözümler, birbirinden tamamen bağımsızdır. Her biri çok boyutlu uzay üzerinde bir vektördür (Bingöl ve diğ., 2000).

GA kullanılarak gerçekleştirilen dinamik OVFS kod tahsisi için temel akış diyagramı Şekil 3.1'de verilmiştir. Burada durağan halde sistem, OVFS kanallama kodu isteğinde bulunacak kullanıcıyı beklemektedir. Veri transfer ihtiyacı olan mobil kullanıcı, ihtiyacı olan OVFS kanallama kodunu kaynak yöneticisine bildirir. Eğer sistemde bu veri transfer hızını karşılayacak kadar boş kapasite (boş kodların hızları toplamı) varsa OVFS kod ağacı yapısına uygun olacak şekilde –diklik prensibine uygun– mobil kullanıcıya OVFS kanallama kodu tahsis edilir. Kapasite yeterli, ancak ihtiyaç duyulan veri transfer hızına uygun boş OVFS kanallama kodu yoksa bu durumda GA bloğu yürürlüğe girer. GA bloğunda sistem performansını arttırmak için SGA ya da D&D-GA algoritmaları kullanılmaktadır. Sistem performansı, GA bloğu vasıtasıyla OVFS kanallama kodu tahsis edilen kullanıcı sayısı ile doğru orantılıdır. Şekil 3.1'deki GA akış diyagramının sözde kodu (pseudo code), Ek-A'da verilmiştir.

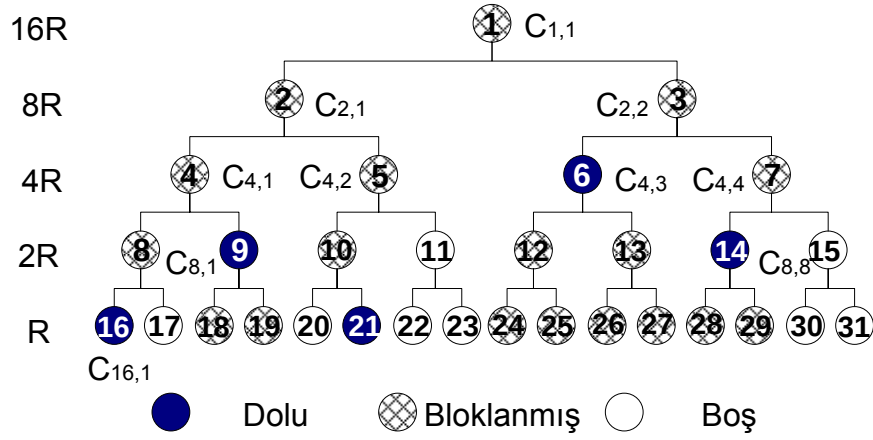


Şekil 3.1: GA bloğu kullanılarak gerçekleştirilen dinamik OVSF kod tahsis sisteminin akış diyagramı

3.2.1. Basit genetik algoritma (SGA)

GA'yı kullanan ağaç yapısının daha iyi anlaşılması için Şekil 3.2'de gösterilen örnek OVSF kod ağacı kullanılmıştır. Burada GA bloğuna girecek mevcut OVSF kod ağacı, başlangıç kromozomu (Ch_{ini}) olarak adlandırılmıştır (Karakoç ve Kavak, 2007). Başlangıç kromozomu, bu ağaçtaki aktif kullanıcıların indis bilgileri ile ifade edilmektedir ($Ch_{ini}=[6 \ 9 \ 14 \ 16 \ 21]$). Diğer bir deyişle kullanılan dalların indis bilgileri, mevcut OVSF kod ağaç indisleri ile ifade edilmekte ve bu indisler popülasyonda bir kromozoma karşılık gelmektedir. İndis değerleri 0, $SF-1$ arasında değişmekte ve yukarıdan aşağıya, soldan sağa doğru artmaktadır. Şekil 3.2'ye göre başlangıç kromozomu $Ch_{ini}=[6 \ 9 \ 14 \ 16 \ 21]$ 'dir ve bu kromozomdaki kullanıcıların

veri transfer hızları da başlangıç kromozomundaki sıraya uygun olarak [4R 2R 2R R R] şeklindedir. Burada R, birim veri transfer hızını göstermektedir.



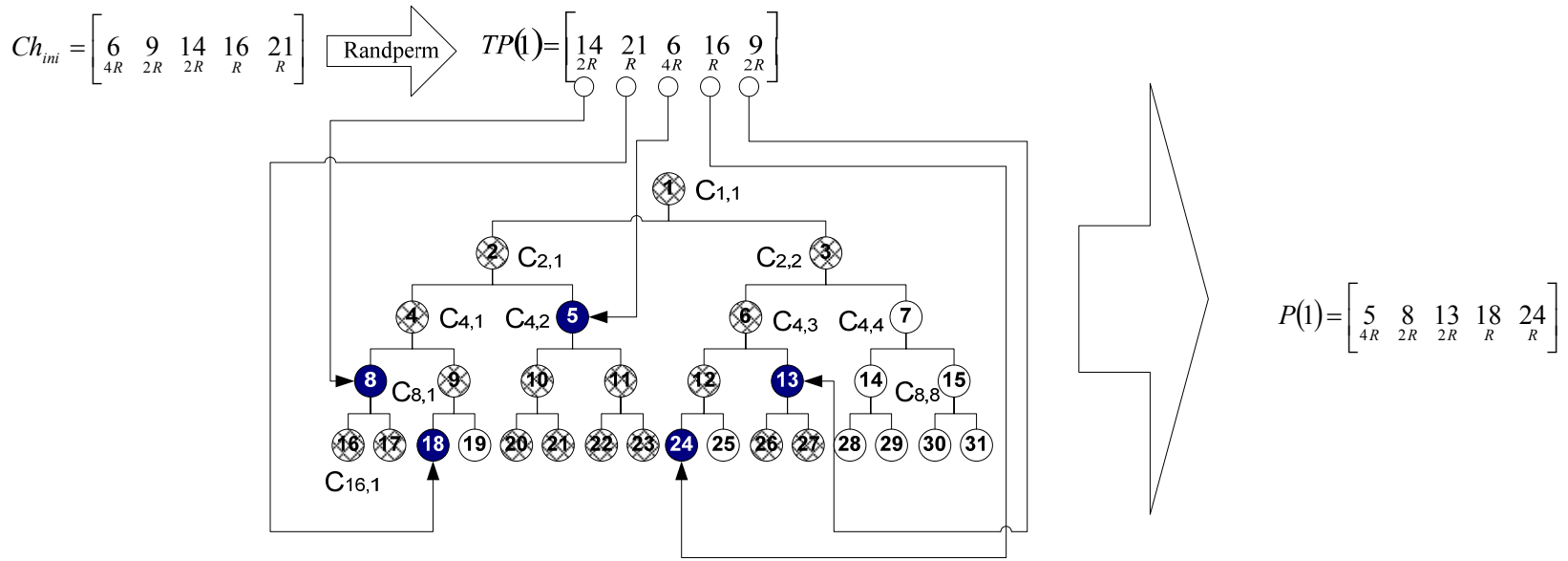
Şekil 3.2: Örnek OVFS kod ağacı (başlangıç kromozomu) $SF=16$, $U=5$, $S=10$

SGA, n adet kromozom içeren bir P popülasyonunun sabit veri transfer hızlarına sahip rasgele değerler alması ile başlar. SGA kullanılarak gerçekleştirilen OVFS kod tahsis algoritmasında, popülasyon büyüklüğü (kromozom sayısı) trafik yoğunluğuna bağlı olarak

$$n = SF - \sum_{i=1}^U S(i) \quad (3.1)$$

formülü ile ifade edilmektedir. Burada, SF , U ve S değerleri sırasıyla yayılım faktörü, kullanıcı sayısı ve onların veri transfer hızları ile ifade edilmektedir. Şekil 3.2'de SF değeri 16, U değeri 5 ve S değeri de 10R'dir. Dolayısıyla başlangıç popülasyonunun büyüklüğü (n) $16-10=6$ adet kromozomdan oluşmaktadır. Bu n kromozomdan her biri, rasgele permütasyon kullanılarak elde edilen OVFS kod ağacının kodlanmış bilgisini içerir, her biri olası bir çözümdür ve farklı bir OVFS kod ağacını temsil etmektedir. Popülasyondaki her kromozom farklı indisler ile ifade edilir, ancak indisler farklı olsa da hepsinin veri transfer hızlarının sayıları ve büyüklükleri başlangıç kromozomunki ile aynıdır. Yine de her bir kromozomdaki aktif kullanıcıların sayısı ve bu aktif kullanıcıların veri transfer hızları Ch_{ini} ile aynıdır. Örneğin;

Şekil 3.3: Başlangıç kromozomundan (Ch_{ini}) yola çıkarak başlangıç popülasyonunun oluşturulması



Şekil 3.3, başlangıç kromozomundan yola çıkarak başlangıç popülasyonunun her bir kromozomunun nasıl oluşturulduğunu göstermektedir. Buradaki TP (geçici popülasyon), başlangıç popülasyonu oluşturulurken kullanılan bir ara adımdır. Bu popülasyonun her bir kromozomu, başlangıç kromozomundaki indislerin rasgele yer değişimi ile elde edilir. Daha sonra TP 'deki her bir kromozom, sırasıyla sadece veri transfer hızlarına bakılarak yeniden kod ağacına soldan sağa doğru ve diklik prensibine uygun olarak yerleştirilir ve böylece başlangıç popülasyonu oluşturulur. Şekil 3.3'te verilen örnek başlangıç popülasyonundaki birinci kromozomun ($P(1)$) nasıl elde edildiği gösterilmektedir. Buna göre rasgele elde edilen $TP(1)$ 'in ilk olarak birinci sıradaki $2R$ veri transfer hızı, daha sonra ikinci sıradaki R veri transfer hızı ve sırasıyla kromozomdaki tüm aktif kullanıcıların veri transfer hızları sırasıyla OVSF kod ağacına soldan sağa doğru yerleştirilir. Daha sonra bu yeni ağacın indisleri, başlangıç popülasyonunun birinci kromozomu $P(1)$ olarak alınır. Buradan sistemdeki aktif kullanıcıların sayısında (örneğe göre $U=5$), onların veri transfer hızında (örneğe göre $[4R \ 2R \ 2R \ R \ R]$) ve sistemin toplam trafik yoğunluğunda (örneğe göre $S=10$) bir değişiklik olmadığı açıkça görülmektedir. Değişiklik, sadece kullanıcıların ağaçta yerleştirildikleri yerlerde yani indis bilgilerindedir. Bu, işlem kromozomun her bir elemanı ağaca yerleştirilene kadar devam eder. Şekil 3.3'te elde edilen başlangıç popülasyonunun ilk kromozomuna ek olarak, popülasyonu oluşturan diğer kromozomlar aşağıda verilmiştir.

$$\begin{aligned}
Ch_{ini} &= \begin{bmatrix} 6 & 9 & 14 & 16 & 21 \\ 4R & 2R & 2R & R & R \end{bmatrix} \Rightarrow TP(2) = \begin{bmatrix} 9 & 6 & 16 & 14 & 21 \\ 2R & 4R & R & 2R & R \end{bmatrix} \Rightarrow P(2) = \begin{bmatrix} 5 & 8 & 13 & 24 & 28 \\ 4R & 2R & 2R & R & R \end{bmatrix} \\
Ch_{ini} &= \begin{bmatrix} 6 & 9 & 14 & 16 & 21 \\ 4R & 2R & 2R & R & R \end{bmatrix} \Rightarrow TP(3) = \begin{bmatrix} 16 & 9 & 14 & 6 & 21 \\ R & 2R & 2R & 4R & R \end{bmatrix} \Rightarrow P(3) = \begin{bmatrix} 6 & 9 & 10 & 16 & 28 \\ 4R & 2R & 2R & R & R \end{bmatrix} \\
Ch_{ini} &= \begin{bmatrix} 6 & 9 & 14 & 16 & 21 \\ 4R & 2R & 2R & R & R \end{bmatrix} \Rightarrow TP(4) = \begin{bmatrix} 21 & 9 & 6 & 16 & 14 \\ R & 2R & 4R & R & 2R \end{bmatrix} \Rightarrow P(4) = \begin{bmatrix} 5 & 9 & 13 & 16 & 24 \\ 4R & 2R & 2R & R & R \end{bmatrix} \\
Ch_{ini} &= \begin{bmatrix} 6 & 9 & 14 & 16 & 21 \\ 4R & 2R & 2R & R & R \end{bmatrix} \Rightarrow TP(5) = \begin{bmatrix} 6 & 14 & 16 & 21 & 9 \\ 4R & 2R & R & R & 2R \end{bmatrix} \Rightarrow P(5) = \begin{bmatrix} 4 & 10 & 12 & 22 & 23 \\ 4R & 2R & 2R & R & R \end{bmatrix} \\
Ch_{ini} &= \begin{bmatrix} 6 & 9 & 14 & 16 & 21 \\ 4R & 2R & 2R & R & R \end{bmatrix} \Rightarrow TP(6) = \begin{bmatrix} 21 & 16 & 9 & 6 & 14 \\ R & R & 2R & 4R & 2R \end{bmatrix} \Rightarrow P(6) = \begin{bmatrix} 5 & 9 & 12 & 16 & 17 \\ 4R & 2R & 2R & R & R \end{bmatrix}
\end{aligned}$$

Popülasyondaki her bir kromozom için daha sonra bir uygunluk değeri (fitness value) belirlenir. Her biri problem için olası bir çözüm olan kromozomların bir sonraki

iterasyon içinde var olması ya da elenmesi konusu uygunluk fonksiyonunun değerine göre karar verilir. Uygunluk fonksiyonu, SGA için çözümü aranan her problemde mutlaka tanımlanması gereken ve her probleme özgü tanımlanan tek kısımdır. Burada da uygunluk fonksiyonu, OVFS kod tahsis/tekrar tahsis problemi için özel olarak tanımlanmıştır. Uygunluk değeri, uygunluk fonksiyonundan dönen bir değerdir. j . kromozomun uygunluk değeri $f(j)$ ile gösterilir ve Ch_{ini} 'ye göre j . popülasyonun her bir kromozomun yer değiştirme miktarları toplamını verir.

$$f(j) = \frac{1}{\sum_{i=1}^U [(Ch_{ini}(j) - P(j, i)) \times S(i)]} \quad (3.2)$$

burada, j kromozom sayısıdır, ve $j=1, \dots, n$ ile ifade edilir. Yukarıda tanımlanan uygunluk fonksiyonunun sözde kodu Şekil 3.4'te verilmiştir.

```

Parameters: u: number of user, s: user's data bit rate
1: procedure fitness_function(code_tree, new_code_tree)
2:      /*Fitness function calculates the amount of replacement of codes in the tree*/
3:      for j=1 to n do
4:          for i=1 to u do
5:              f(j)=abs(code_tree(j)-new_code_tree(j,i))*s(i)
6:          end /*for i= 1 to u do*/
7:          cost=[cost f(j)];
8:      end /*for j=1 to n do*/
9: end /*procedure fitness_function(code_tree, new_code_tree)*/

```

Şekil 3.4: OVFS kod tahsis problemine özel tanımlanan uygunluk fonksiyonunun sözde kodu

Şekil 3.3'te verilen başlangıç kromozomundan popülasyondaki birinci kromozomun elde edilmesi işleminde uygunluk fonksiyonu tarafından hesaplanan değer

$$\sum_{t=1}^U (Ch_{ini}(t) - P(i, t)S(i, t)) = (6-5) \times 4R + (9-8) \times 2R + (14-13) \times 2R + (18-16) \times R + (24-21) \times R$$

$$= 4R + 2R + 2R + 2R + 3R = 13R \text{ [birim/yerdeğiştirme]}$$

Ancak minimizasyon problemi ilgilendiğimiz için bu değer,

$$C_i = \frac{1}{\sum_{t=1}^U (Ch_{ini}(t) - P(i, t)S(i, t))} = 0,0769 \text{ [birim / yer değiştirme]'dir.}$$

Popülasyondaki kromozomların uygunluk değerlerine göre seçim operatörü (selection), n kromozomlu yeni bir popülasyon üretir. Seçim işleminin amacı, uygunluk değerleri yüksek kromozomların nesiller aracılığıyla devamlılıklarının sağlanmasıdır. Seçim işlemi, arama uzayındaki aramanın yönlendirilmesinde ve iyi çözümlerin elde edilmesinde büyük bir öneme sahiptir. Seçim operatörü kullanılarak oluşturulan bu yeni popülasyon, orijinal popülasyonda uygunluk değerleri yüzdesi iyi olan kromozomların çeşitli tekniklere göre elde edilmesi ile meydana gelir. Bu çalışmada başlangıçta seçim operatörü olarak rulet tekerleği tekniği kullanılmıştır (Baker, 1985). Rulet tekerleğinde her bir kromozomun seçilme olasılığı ($Pr(j)$), onun uygunluk değeri ile doğru orantılıdır ($f(j)$),

$$P_{selected}(j) = \begin{cases} P(1) & \text{if } Pr(1) > Pr(2), Pr(3), \dots, Pr(j) \\ \dots & \\ P(j) & \text{if } Pr(j) > Pr(1), \dots, Pr(j-2), Pr(j-1) \end{cases} \quad (3.3)$$

Rulet tekerleği tekniğinde, popülasyonu oluşturan her bir kromozom, uygunluk değeri yüzdesi ile orantılı olarak seçilmektedir. Orijinal popülasyonun her bir kromozomunun uygunluk değeri, rulet tekerleği üzerinde ayrılan parça ile doğru orantılıdır. Yani uygunluk değeri yüksek olan kromozomlar rulet tekerleğinde daha büyük bir dilime sahiptir ve bir sonraki popülasyonda seçilme olasılıkları daha yüksektir. Bir sonraki popülasyon için seçim yapılırken rulet tekerleği üzerinde rasgele bir nokta seçilir ve bu noktaya karşılık gelen uygunluk değerine ait kromozom, popülasyona yerleştirilir. Bu işlem, popülasyon tamamlanıncaya kadar yani n defa tekrar eder. Burada uygunluk değeri probleme özgü tanımlanan uygunluk fonksiyonuna göre 0-1 arasındadır ve 1'e yaklaştıkça değeri artmış sayılır ve dolayısıyla rulet tekerleği üzerinde daha büyük bir alan işgal etmiş olur.

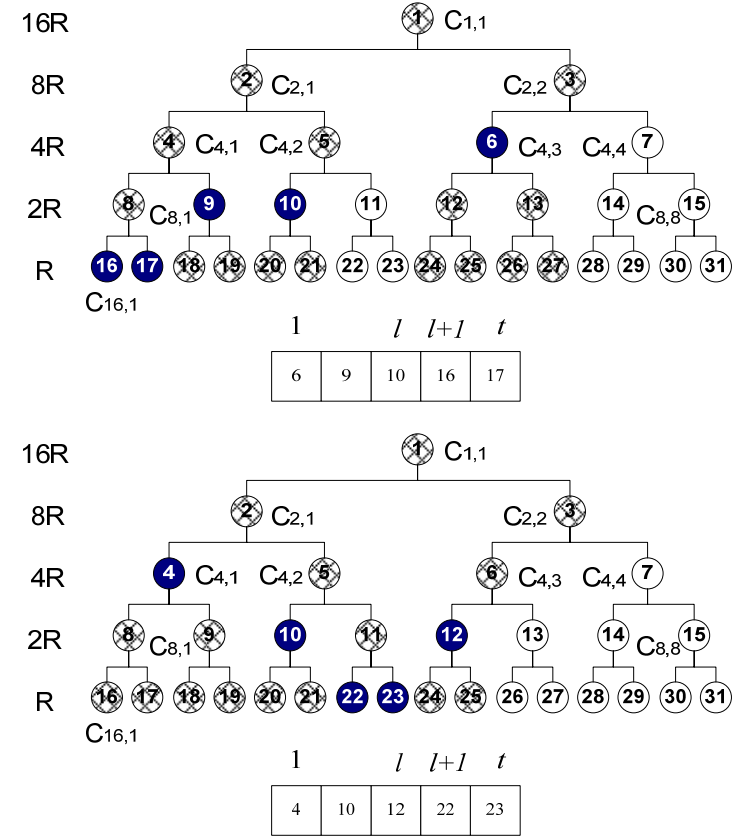
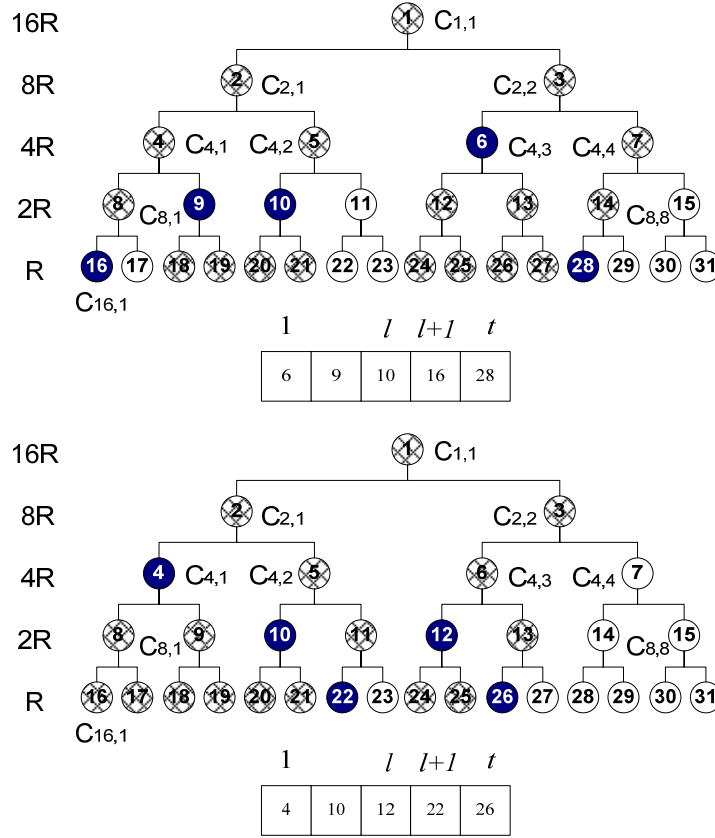
SGA'nın problem çözmedeki gücü, çaprazlama operatöründen (crossover) kaynaklanmaktadır. Kromozomlar arasındaki bilgi değişimini gerçekleştirerek daha iyi kromozomlar üretmek için kullanılır (Eshelman ve diğ., 1989). Sadece seçim işlemi uygulandığında, nesillerdeki kromozomlar hep birbirlerine benzer olacaktır. Bunu önlemek için çaprazlama operatörü kullanılmaktadır. Çaprazlama operatörü,

önceden belirlenen çaprazlama oranına (p_c) göre popülasyonun tamamına ya da bir kısmına uygulanır. Çaprazlama işlemine tabi tutulacak kromozom sayısı (j)

$$j = n \times p_c \quad (3.4)$$

ile tespit edilir. Burada, p_c çaprazlama oranını verir. Bu işlem sonunda j , tek ise $(j-1)/2$, çift ise $j/2$ kadar rasgele seçilmiş kromozom çifti çaprazlamaya uğrar. Daha sonra çaprazlanacak kromozom çiftleri, popülasyondan rasgele seçilir ve çaprazlama işlemi sadece bu çiftler arasında yapılır. Bu çalışmada, başlangıçta tek noktalı çaprazlama tekniği uygulanmıştır. Bu teknikte, her bir kromozom çifti için rasgele bir l noktası seçilir. Daha sonra birinci kromozomun $[1;l]$ arası genleri ile ikinci kromozomun $[l+1;t]$ arası genleri çaprazlanmış birinci kromozomu, yine ikinci kromozomun $[1;l]$ arası genleri ile birinci kromozomun $[l+1;t]$ arası genleri çaprazlanmış ikinci kromozomu oluşturur. Şekil 3.5'te, çaprazlama operatörü açık olarak ifade edilmiştir.

Şekil 3.5: Tek noktalı çaprazlama tekniğinin OVSF kod ağacına uygulanması

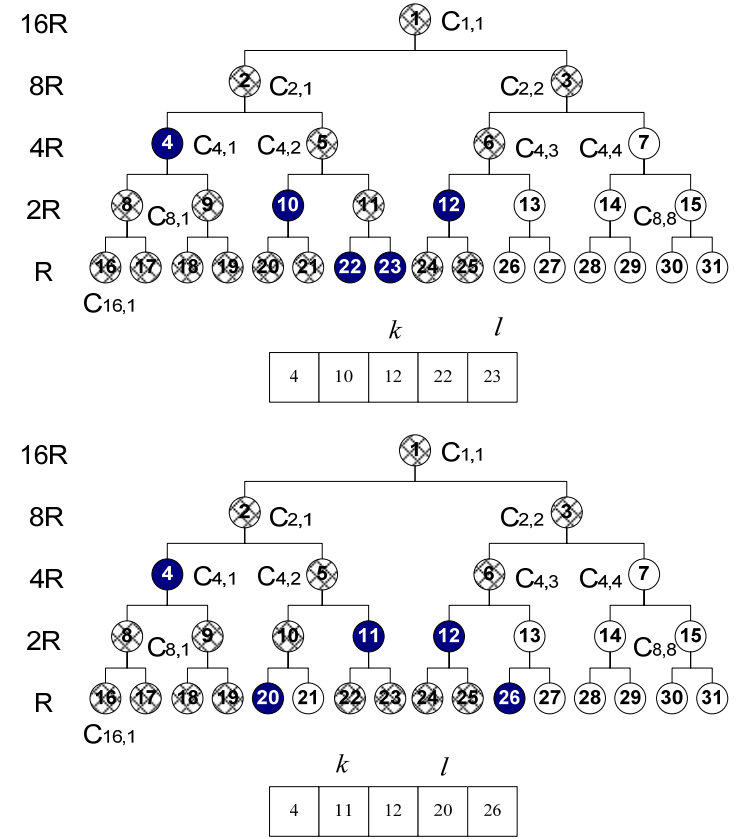
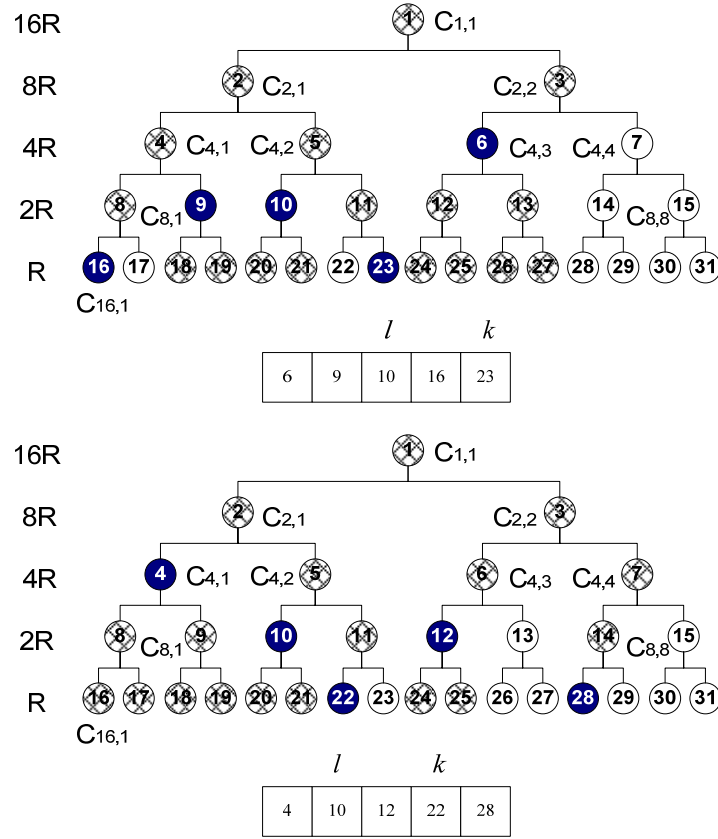


Çaprazlama operatöründen sonra mutasyon operatörü (mutation) kullanılır. Bu işlem, popülasyon içindeki kromozomların çoğunluğunun yerel minimuma takılmasını engeller ve aynı zamanda genetik çeşitliliği artırır. Yanlışlıkla ya da kazara kaybolan gen değerlerinin yeniden ortaya çıkarılmasını, genetik yığılmanın önlenmesini ve nüfusun çoğunluğuyla çözüme yakınsadığında bütünsel en iyi çözümü bulmak için nüfus çevresinde rasgele küçük adımlarla arama yapılmasını sağlar. Bunun için 0-1 arasında rasgele bir sayı seçilir. Eğer bu sayı mutasyon oranından (p_m) küçükse, o zaman mutasyon işlemi başlatılır; aksi takdirde bir sonraki kromozom için başka rasgele bir sayı alınır. Mutasyon işleminde kromozomdan rasgele seçilmiş iki gen (l, k) kendi aralarında yer değiştirmektedir. Şekil 3.6’de yer değiştirme mutasyonunun nasıl yapıldığı görülmektedir. Bu tekniğin adı yer değiştirme mutasyonu olarak adlandırılmaktadır ve

$$\begin{aligned} \text{if } rand(1) < p_m \quad & p(j, l_{new}) = p(j, k_{old}) \text{ and } p(j, k_{new}) = p(j, l_{old}) \\ \text{else} \quad & p(j, l_{new}) = p(j, l_{old}) \text{ and } p(j, k_{new}) = p(j, k_{old}) \end{aligned} \quad (3.5)$$

ile ifade edilir. Yer değiştirme işlemi esnasında aktif kullanıcının veri transfer hızı dikkate alınır. Bu çalışmada bilindiği üzere normalde kromozomu oluşturan aktif kullanıcıları OVSF kod ağacına yerleştirirken en büyük veri transfer hızına sahip kullanıcıdan başlayarak en küçük olana kadar işlemi devam ettirmekteyiz. Ancak yer değiştirme mutasyon operatöründe örneğin 4R veri transfer hızına sahip bir kullanıcı ile R veri transfer hızına sahip kullanıcı yer değiştirdiğinde önce R hızlı kullanıcı daha sonra da 4R hızlı kullanıcı diklik prensibine uygun olacak şekilde OVSF kod ağacına yerleştirilir.

Şekil 3.6: Yer değiştirme mutasyon tekniğinin OVSF kod ağacına uygulanması



SGA için bahsedilen tüm bu işlem basamakları sonunda popülasyondaki her kromozomun uygunluk değeri kontrol edilir. Uygunluk değeri, en optimumun olan kromozomun temsil ettiği OVFS kod ağacı, mevcut OVFS kod ağacı olarak alınır ve mobil kullanıcı tarafından ihtiyaç duyulan veri transfer hızı talebi bu ağaca işlenir. Aksi taktirde SGA ile dinamik OVFS kod tahsisi, ya optimizasyon kriteri sağlanana kadar ya da daha önceden belirlenen algoritmanın durma değerine erişilene kadar devam eder. Burada belirlenen optimizasyon kriteri, ihtiyaç duyulan kodun mobil kullanıcıya tahsis edilip edilmediğidir. Algoritmanın durma değeri de 1000 olarak set edilmiştir. Yani SGA algoritması, optimizasyon testini sağlayan OVFS kod ağacını bulmak için 1000 defa çalışmaktadır. Belirlenen bu rakam sonunda uygun kod ağacı elde edilemezse arama bloklanır. Her ne kadar değişken trafik yoğunluğuna bağlı olarak uygunluk fonksiyonunun parametreleri değişse de, SGA her zaman olası en iyi sonucu elde etmek için çalışmaktadır. Kısaca özetlemek gerekirse;

- Popülasyon, OVFS kod ağacındaki aktif kullanıcıların sayısına bağlı olduğu için büyüklüğü dinamik olarak her adımda değişebilmektedir. Bu değer sabit bir rakamla ifade edilmesi mümkün değildir.
- Uygunluk fonksiyonu, OVFS kod tahsis/tekrar tahsis probleminin amacına uygun olacak şekilde probleme özgü tanımlanmıştır ve popülasyonun her bir kromozomunun başlangıç kromozomuna göre toplam yer değiştirme miktarları ile ifade edilir.
- Seçim operatörü için ilk aşamada rulet tekerleği seçim tekniği kullanılmıştır. Rulet tekerleğinde, her kromozomun uygunluk değeri ile tekerlekte işgal ettiği alan orantılıdır. Bu orana bağlı olarak da seçilme olasılığı değişmektedir. Seçim işlemi, popülasyonu oluşturan kromozom sayısı tamamlanana kadar devam eder. Seçim operatörü olarak daha sonra, turnuva, en iyi yüzde ve rasgele teknikleri de rulet tekerleği ile birlikte incelenecektir.
- Çaprazlama, yeni çözümler üretmek ve kromozomlar arasında bilgi değişimi yapmak amacı ile gerçekleştirilen güçlü bir operatördür. İlk aşamada tek noktalı çaprazlama tekniği seçilmiştir. Burada kromozom herhangi bir noktasından ayrıştırılarak ona eş olarak seçilen diğer bir kromozomun ayrıştırılmış aynı noktası ile birleştirilir. Yani ilk kromozomun başlık kısmı ile ikinci kromozomun kuyruğu, ikinci kromozomun başlık kısmı ile ilk kromozomun kuyruğu

birleştirilir. Çaprazlama operatörü olarak daha sonra iki noktalı, düzenli ve aritmetik çaprazlama teknikleri tek noktalı çaprazlama tekniği ile birlikte incelenecektir.

- Mutasyon işlemi, popülasyondaki kromozomların tekdüzeliğinden kurtulması amacı ile gerçekleştirilir. Mutasyon operatörü olarak başlangıçta yer değiştirme tekniği kullanılmıştır. Bu teknikte, kromozom içindeki rasgele seçilen herhangi iki gen kendi aralarında yer değiştirerek OVSF kod ağacını tekrar inşa ederler. Mutasyon operatörü olarak daha sonra öteleme mutasyon tekniği de ele alınacaktır.

SGA, yukarıda genel olarak tanımlanan OVSF kod ağacının yer değiştirmesi işlemini gerçekleştirirken bazı zorluklarla karşı karşıya kalır. En çok karşılaşılan zorluk, mevcut trafik yoğunluğu için OVSF kod ağacını optimize ederken karşılaşılan erken yakınsamadır. SGA, normal çalışma durumunda tüm popülasyonu tek bir kromozom üzerine yoğunlaştırarak yakınsar. Eğer, çaprazlama operatöründen sonra elde edilen yeni kromozom, mevcut kullanıcıların veri transfer hızını sağlayamazsa ya da GA operatörü optimum kromozoma yakınsamaya çalışırken ortama yeni bir kullanıcı gelir ya da ortamdan herhangi bir kullanıcı giderse, GA bu beklenmedik değişime uyum sağlayamaz. Bu sorunun üstesinden gelebilmek için “dominance ve diploidy” yaklaşımı ele alınmıştır.

3.2.2. Dominance ve diploidy tabanlı genetik algoritma (D&D-GA)

SGA, doğada bulunan en basit genetik özelliğe sahip yarı kromozomlu ya da tek sıralı kromozomu incelemektedir. Bu basit model, ilgilendiğimiz probleme ilişkin tüm bilgiyi tek sıralı bir dizide içermektedir. Bunun yanı sıra doğal hayatta yaşam formunu sürdürebilmek için daha karmaşık yapıları kromozom yapılarının oluşturulmasına ihtiyaç vardır. Diploid formda ilgilendiğimiz aynı probleme ilişkin her bir bilgi, bir ya da birden fazla kromozom çiftinde taşınır (Weile ve Michielssen, 2001).

Diploid kromozom, zamanla değişen ortamlarda kromozomların hayatta kalmasına yardımcı olur. Sahip olunan iki ayrı gen, bize ilgilendiğimiz probleme ilişkin iki ayrı sonuç bilgisini sunmaktadır ve bu bilgi bir sonraki nesile aktarılabilir. Bu iki ayrı kromozomdan bir tanesi baskın iken diğeri ise çekiniktir.

Diploid yapının en belirgin avantajı, tek sıralı kromozoma göre popülasyonda olası sonuçlar için daha fazla çeşitlilik sağlamasıdır. Mevcut OVSF ağaç yapısı için geçersiz ancak daha sonra olası ağaç yapıları için faydalı olabilecek kromozomlar çekinik pozisyonda ele alınır.

Diploid yapı, farklı durumlar arasında çok hızlı geçişin sağlandığı bağlantılı uygulamalar için faydalı olmaktadır. GA'da, diploid yapının daha karmaşık yapıya olduğu açıktır. Burada bir yerine iki ayrı kromozom oluşturulup bunlardan bir tanesi de baskın özellikte olacak şekilde işaretlenerek alınacaktır.

Baskınlık haritasının gelişimi, kodlamanın üçlü tanımlanması ile yani kromozomdaki her genin üç farklı değerden birini alması ile elde edilir (Goldberg, 1989). Bu üçlü yapı, her bir gen için probleme göre $\{-1, 0, 1\}$ değerlerinden birinin alınması ile elde edilir. Üçlü diploid kromozomların kodları, iki kurala göre çözülür. Bu kurallardan birincisinde;

- $0 \rightarrow 0$ 'ı kodlar, ayrıca hem $(-1) \rightarrow 1$ 'i hem de $1 \rightarrow 1$ 'i kodlar
- 1 0 'a göre, 0 da (-1) 'e göre daha baskındır.

Diğer bir yapı ise Şekil 3.7'de gösterilen ve "punnett square" olarak adlandırılan yapıdır. Bu kuralın en büyük avantajı, değişen ortam şartlarına göre baskın karaktere sahip olan kromozomun kolayca işaretlenerek yer değiştirebilmesidir. Karakterleri temsil eden genler, baskın ya da çekinik formda 1 ya da -1 ile ifade edilebileceklerinden hiçbir karakter, yalnız baskın öğeleri içeremez. Dahası diploid yapısı bilgidan koruma metotları ile yani karakter için faydalı ama kromozom dizisi için ifade edilmesi gereksiz yapıyı oluşturarak popülasyonu destekler. Üçlü kodlama, Şekil 3.7'de gösterilen ve çizgili ya da noktalı karelerden oluşan iki basit Punnett Square yapısının birleşiminden oluşmaktadır. Aktif haldeki Punnett Square yapısının

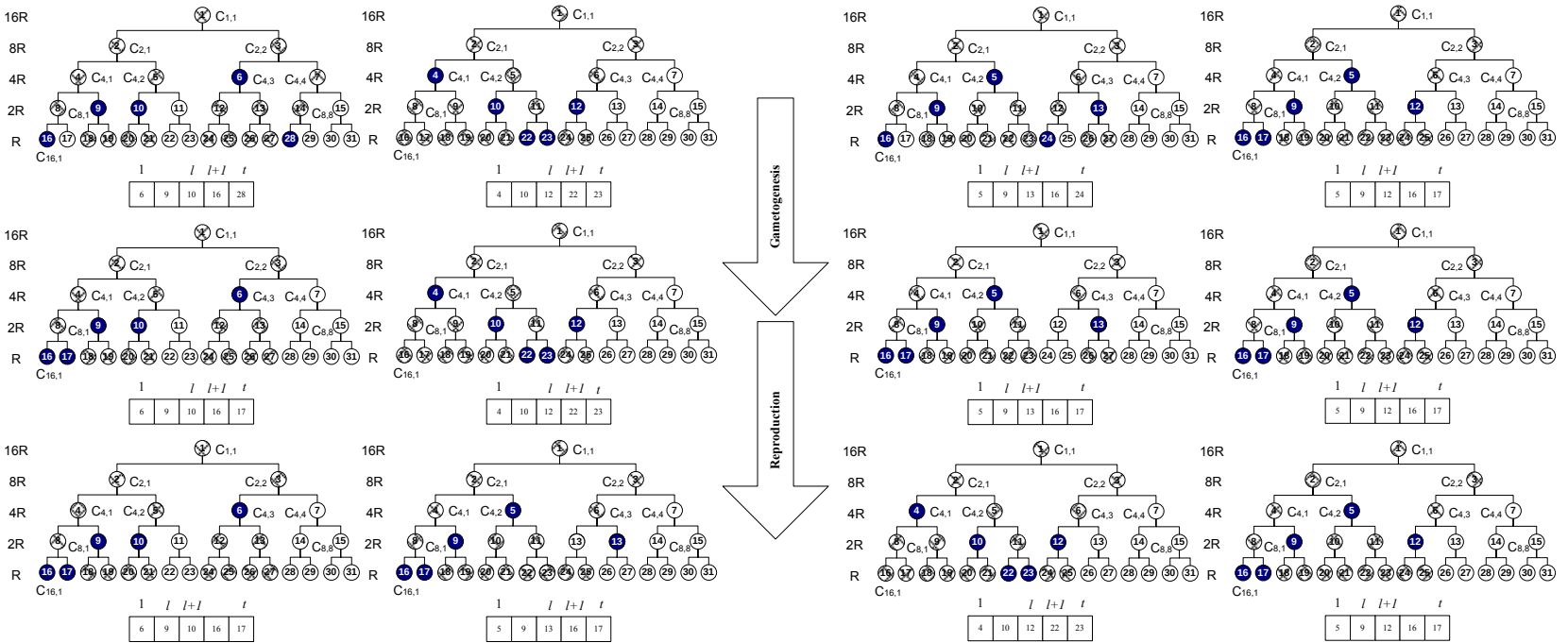
hangisi olduğunu belirlemek için ya 0 baskın (çizgili yapı) ya da 1 baskın (noktalı yapı) seçilerek genler işaretlendirilebilir. Mevcut popülasyon, ilgilendiğimiz OVFSF kod ağacındaki hızları içermiyorsa ya da GA yapısı en uygun OVFSF kod ağacını belirlemek için çalışırken kullanıcı sayısında değişme oluyorsa bu durumda baskın durum yer değiştirerek arama uzayında mevcut probleme uygun bir yapıdan başlanarak optimizasyon yapılır.

	-1	0	1
-1	-1,-1 (1)	-1,0 (0)	-1,1 (1)
0	-1,0 (0)	0,0 (0)	0,1 (1)
1	-1,1 (1)	0,1 (1)	1,1 (1)

Şekil 3.7: “Punnet Square” olarak adlandırılan üçlü kodlama

SGA’dan üçlü “dominance ve diploidy” yapısına geçmek için SGA yapısında bazı değişiklikler yapmak gerekmektedir. İlk olarak başlangıç popülasyonu olarak bir değil iki ayrı popülasyon oluşturulur ($2 \times n$). Daha sonra SGA’da belirtildiği gibi uygunluk değerlerine bağlı olarak popülasyon iyileştirilir. Bilindiği üzere seçim operatörü uygunluk değerine göre iyi kromozomları seçmektedir. Bu yapıda değişiklik yoktur. Çaprazlama işlemine geçtiğimizde; bu aşama SGA’ya göre biraz daha karmaşık yapıdadır. Her bir kromozom, gamet hücresi (gametogenesis) ve üreme (reproduction) aşamalarından geçirilir. Gamet hücresi aşaması doğada görülen mayoz işleminin aynısıdır. Özellikle çaprazlama operatörüne giren iki ayrı anne baba kromozom, gamet oluşturmak için SGA çaprazlama tekniklerini kullanarak hibritleştirilirler. Daha sonra üreme aşamasında rasgele oluşturulan iki gamet, çocuk kromozomları üretir. İki aşamadan oluşan bu faz Şekil 3.8’de ele alınmıştır. Son olarak D&D-GA’da gerçekleştirilen mutasyon işlemi de SGA’daki yapıya çok benzerdir. Böylelikle bir işlem adımı boyunca GA bloğu D&D-GA yaklaşımı çerçevesinde çalışmış olur. Bu süreç optimizasyon kriteri sağlanana kadar devam eder.

Şekil 3.8: D&D-GA için çaprazlama operatörü. Gamet hücresi aşamasında kromozomlar tek noktali çaprazlama tekniği uygulanarak hibrit yapılır. Üreme aşaması ise hibrit hale getirilen kromozomlardan birer tanesinin alınması ile gerçekleştirilir.



3.3. Benzetilmiş Tavlama Tabanlı Dinamik OVSF Kod Tahsisi

Benzetilmiş tavlama (SA) algoritması, pek çok değişkene sahip fonksiyonların en büyük veya en küçük değerlerinin bulunması ve özellikle pek çok yerel minimum değere sahip doğrusal olmayan fonksiyonların en küçük değerlerinin bulunması için tasarlanmıştır (Kirkpatrik ve diğ., 1983). Diğer optimizasyon teknikleri (genetik algoritma, parçacık sürü, tabu arama vb.) gibi en iyi çözümün en kısa zamanda üretimini sağlar. Bu sebeple, özellikle matematiksel modellerle gösterilemeyen kombinasyonel problemlerin optimizasyonunda tercih edilir. SA algoritmasının uzun süre tek bir arama bölgesine takılıp kalmaması algoritmanın avantajlarından biridir. Bununla birlikte algoritmanın arama uzayında arama işlemine tek bir noktadan başlaması ve minimum arayışını sadece bu noktanın yakın çevresinde gerçekleştirmesi, hesaplama karmaşıklığı açısından sisteme daha az yük getirirse de optimuma ulaşma süresi daha uzun olmaktadır.

3.3.1. Metallerde tavlama

Tavlama terimi, bir metalin belirli bir süre için sıcaklığının artırılıp, kristalleşinceye kadar yavaş yavaş soğutulduğu ısı bir işlemi anlatır. Tavlama sonucunda meydana gelen kristalleşme, metalin mekanik özelliklerini iyileştiren moleküler yapı değişikliklerine neden olur. Bu değişiklikler, metale ait metal parçacıklarının azalan yeni enerji durumuna göre kendilerini yeniden düzenlemeleridir. Metallerde tavlamanın amacı; stresi azaltmak, yumuşaklığı, dayanıklılığı ve esnekliği arttırmak ve belirli bir mikro yapıyı oluşturmaktır.

Tavlama işlemi iki aşamada gerçekleşmektedir; birinci aşama istenen sıcaklığa kadar ısıtma, ikinci aşama belirli bir sıcaklıkta tutma ve soğutmadır (Masri ve diğ., 1999). Tavlama işleminin ilk aşaması olan ısıtma işleminde metal istenen sıcaklığa kadar ısıtılır. Metal parçacıkları yüksek sıcaklıklarda oldukça yüksek enerjiye ve serbestliğe sahiptir. Sıcaklığın yavaş yavaş düşürülmesiyle gerçekleştirilen soğutma işlemi sırasında parçacıkların enerjileri azalır. Her yeni enerji duruma uygun olarak, yeni bir dengeye girmek üzere kendilerini tekrar düzenlerler. Soğutma işlemi

genellikle oda sıcaklığına kadar olur. Bu işlemlerde zaman önemli bir parametredir. Soğutma, çok hızlı olursa kristal yapı içerisinde düzensizlikler ve bozulmalar görülür. Metal parçacıkları, enerjinin en düşük olduğu duruma ulaşamaz ve enerjinin en yüksek olduğu kristalimsi durumda soğutma işlemi sona erer. Bu da, metalde şekil bozukluğuna hatta çatlamalara neden olur. Gerçek tavlama süresi, gerekli değişim hareketleri için yeterli derecede uzun olmalıdır. Mükemmel bir kristal yapının elde edilmesi ancak bu süre içinde metalin yavaş soğutulması ile gerçekleştirilir. Bunun nedeni, ısı enerjinin sağladığı hareket etme ve yer değiştirme kolaylığının, metal parçacıkların yerel en iyi şekillendirmeden kaçınmalarına, bunun yerine bütünsel en iyi şekillendirmeyi oluşturmalarına imkan vermesidir. Ayrıca tavlama sıcaklığı da bu süreçte önemli bir konudur. Sıcaklık artırılarak tavlama işlemi hızlandırılabilir (Hondroudakis ve diğ., 2006).

3.3.2. Fiziksel sistemler ve benzetilmiş tavlama

SA algoritması, Kirkpatrick ve arkadaşları (1983) tarafından geliştirilmiş yerel bir arama algoritmasıdır. Katı bir maddenin enerji durumunu en aza indiren fiziksel sistemlerdeki tavlama süreci ile kombinasyonel optimizasyon problemlerinin çözüm süreci arasındaki benzerlik üzerine kurulmuştur.

Tablo 3.1: Fiziksel sistemler ve kombinasyonel optimizasyon problemleri arasındaki benzerlik

Fiziksel Sistemler	Optimizasyon Problemleri
Durum	Yapılandırma
Enerji	Maliyet fonksiyonu
Son durum	En iyi yapılandırma
Soğutma	Tekrarlamalı işlem
Ölçülü tavlama	Benzetilmiş tavlama

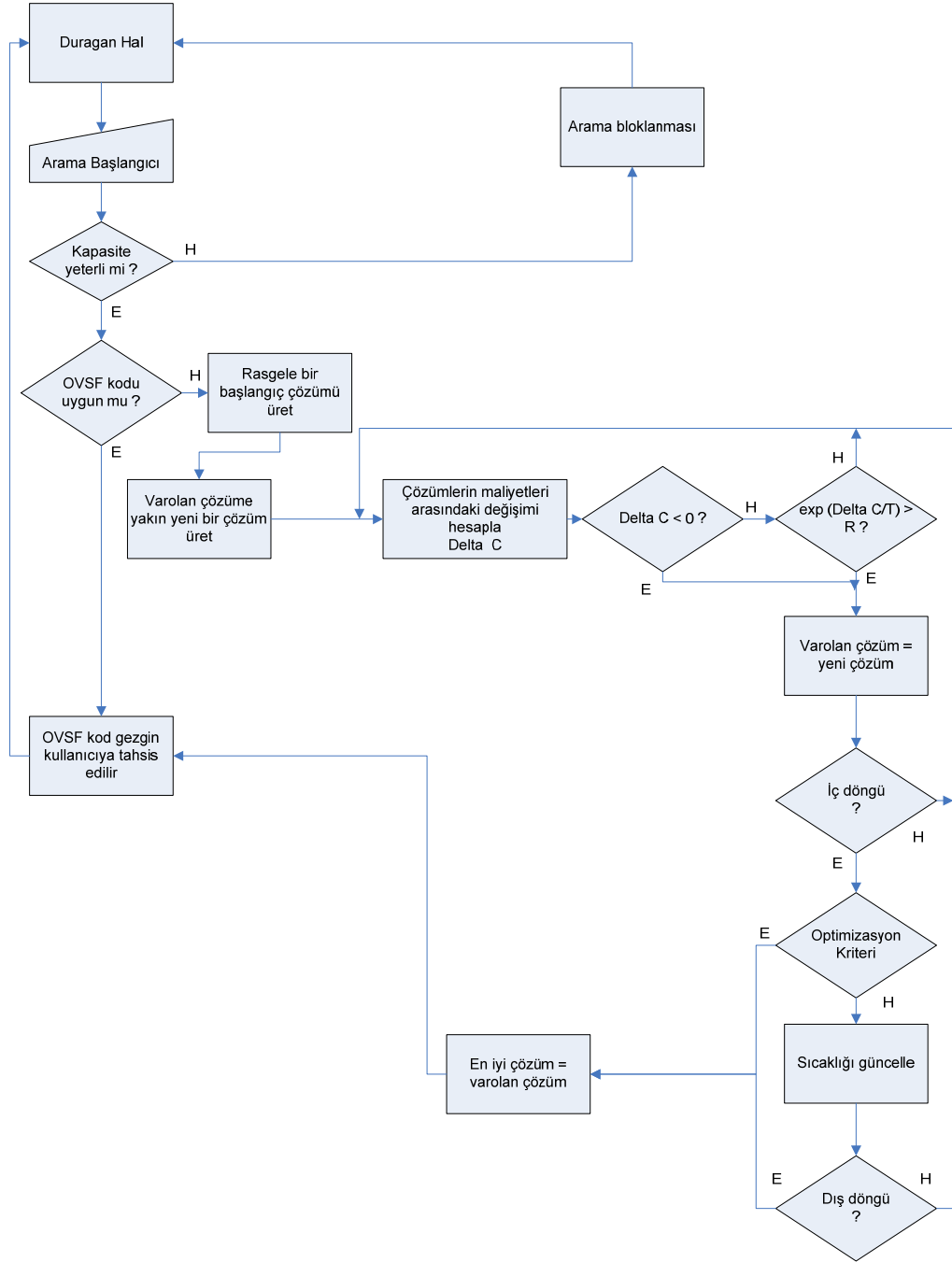
Fiziksel sistemlerde gerçekleştirilen tavlama işlemi, kombinasyonel optimizasyon problemleri için model oluşturmak amacıyla kullanılmaktadır. Fiziksel sistemler ve kombinasyonel optimizasyon problemleri arasındaki benzerlik, Tablo 3.1’de gösterilmiştir (Lutfiyya ve diğ., 1992). Buna göre sistemdeki enerji durumlarının toplamı, problemin yapılandırılmasına benzetilir. Sistemin sahip olduğu enerji ise problem için optimizasyonu istenen maliyet fonksiyonuna benzetilir. Son durum,

optimizasyon problemlerinde maliyet fonksiyonundan elde edilen en iyi yapılandırmaya karşılık gelir. En iyi yapılandırmanın diğer bir deyişle en iyi çözümün elde edilme süreci, fiziksel sistemlerde son duruma karar verilmesi ile belirlenir. Fiziksel sistemlerdeki soğutma süreci, optimizasyon problemlerindeki en iyi çözümün bulunması için yapılan tekrarlamalı işlemlere karşılık gelmektedir. Fiziksel sistemlerdeki bu süreç, optimizasyon problemlerinde benzetilmiş tavlama olarak bilinmektedir. Kontrol parametresi olarak da sıcaklık kullanılmaktadır.

3.3.3. SA algoritmasına genel bakış

SA algoritması, optimizasyon problemlerinin çözümü için kullanılan yerel bir arama algoritmasıdır. Bu teknikte amaç, problem için uygun çözümler ararken zayıf bir yerel çözümde takılıp kalmayı engellemektir. Burada SA algoritması ile yeni gelen kullanıcıya ihtiyaç duyduğu veri transfer hızını sağlayacak uygun kanallama kodu minimum maliyet ile tahsis edilmeye çalışılır.

SA algoritması ile OVSF kod ağacının optimizasyonunu gösteren akış diyagramı Şekil 3.9’da verilmiştir. OVSF kod ağacının SA algoritması ile optimizasyonu, ancak kod ağacında yeni kullanıcının ihtiyacı olan veri transfer hızı kadar boş kapasite olup, uygun boş OVSF kodu olmaması durumunda gerçekleştirilir. Arama süreci ilk olarak fiziksel sistemlerde herhangi bir duruma karşılık gelen rasgele seçilmiş bir başlangıç çözümüyle başlar. Kod ağacının kapasitesi yeterli olsa da uygun OVSF kanallama koduna sahip olmayan kod ağacındaki aktif kullanıcıların indis bilgilerinin kullanımı, arama sürecini belirler. OVSF kod ağacındaki aktif kullanıcıların indis bilgisinin belirlenmesi Şekil 3.2’de verilmiştir.



Şekil 3.9: SA algoritması kullanılarak OVFS kod ağacının optimizasyonunu gerçekleştiren akış diyagramı

Daha sonra, önceden belirlenmiş bir komşuluk hareketinin kullanılmasıyla arama uzayında var olan çözüme yakın, olası yeni bir çözüm üretilir. Burada belirlenen komşuluk hareketinde amaç, yeni bir çözüm oluşturulması ve oluşturulan yeni çözümün (yeni OVFS kod ağacının) mevcut çözüme göre minimum yer değiştirmesiyle sağlanmasıdır. Bunun için Denklem 3.2’de GA için verilen uygunluk

fonksiyonu kullanılır. Mevcut ve yeni OVSF kod ağaçlarının maliyetleri arasındaki değişim miktarı

$$\Delta C = C_i - C_{i-1} \quad (3.6)$$

ile hesaplanır. Burada ΔC , iki çözümün maliyetleri arasındaki değişim miktarını göstermektedir. C_{i-1} , $(i-1)$. ötelemedeki çözümü (SA algoritması ile OVSF kod tahsisi yapılacak mevcut kod ağacını) temsil etmektedir. Genel bir yerel arama süreci için maliyette bir azalma gerçekleşiyorsa, yani ΔC değeri sıfırdan küçükse yeni çözüm var olan çözümle değiştirilir. Aksi durumda, yeni çözüm Denklem 3.7’de ifade edilen eşitsizliğe göre belirli bir olasılıkla kabul edilir. Bu eşitsizliğin doğru olmama durumunda yeni bir çözüm üretmek için SA algoritmasının başına dönülür.

$$\exp(-\Delta C / T) > R \quad (3.7)$$

Burada R , $[0,1]$ arasında üretilmiş düzgün dağılımlı rasgele bir sayıyı ve T , sıcaklık olarak bilinen kontrol parametresini göstermektedir. Yeni çözümün varılan çözüm olarak kabul edildiği bu adımdan sonra algoritmanın iç döngü koşulunu sağlayıp sağlamadığı kontrol edilir. İç döngü koşulu, her sıcaklık değerinde kaç kez olası yeni çözüm üretileceğine karar verir. Eğer iç döngü koşulu sağlanıyorsa, sıcaklık bir sonraki adımda azaltılmak suretiyle önceden belirlenmiş bir soğutma çizelgesine göre belirli bir güncelleme kuralı ile güncellenir. Aksi halde komşuluk hareketinin kullanılmasıyla varılan çözüme yakın yeni bir çözüm üretilerek bu adıma kadar olan işlemler tekrarlanır. Bu adımdan sonra dış döngünün sağlanıp sağlanmadığı kontrol edilir. Dış döngü, algoritmanın sonlandırma koşulunu içerir. Algoritmanın sonlandırma koşulu, yeni kullanıcının ihtiyaç duyduğu veri transfer hızı için OVSF kod ağacında uygun boş kanallama kodunun sağlanmasıdır. Eğer tanımlanan bu koşul sağlanıyorsa, en iyi çözüm yerel en küçük maliyet miktarına sahip varılan çözümdür ve algoritma bu çözümle sonlandırılır. Aksi halde algoritma, yeni bir çözüm üreterek bu adıma kadar olan işlemleri tekrarlar. Algoritmanın sonlandırılması, bu verilen koşulun yerine getirilmesine ya da önceden belirlenmiş dış döngü sayısına göre gerçekleştirilir.

Başlangıçta T , yüksek bir sıcaklıktır. Sıcaklığın yüksek olmasından dolayı yeni çözümlerin kabul edilme olasılığı oldukça yüksektir. Ayrıca başlangıçta sıcaklığın yüksek olması, arama uzayında büyük adımlar atılmasına ve arama uzayının keşfedilmesine imkan verir. ΔC 'nin artan değerleri de arama uzayında farklı bölgelerde arama yapıldığını göstermektedir. Ancak T ve ΔC azalmaya başladıkça, yeni çözümlerin kabul edilme olasılığı azalmakta ve böylece sistem daha kararlı hale gelmektedir. Yani arama bütünsellikten yerelliğe doğru kaymaktadır. SA algoritması kullanılarak gerçekleştirilen OVFS kod tahsis yönteminin sözde kodu Ek-A'da verilmiştir.

3.3.4. SA algoritması parametreleri

SA algoritması bir optimizasyon problemine uygulanmadan önce aşağıda sıralanan parametrelerin belirlenmesi gerekmektedir (Shahookar ve Mazumder, 1991) (Pham ve Karaboğa, 2000).

- Çözümlerin gösterimi
- Maliyet fonksiyonu
- Yakın çözümleri üreten mekanizmanın tanımlanması (komşuluk hareketi)
- Soğutma çizelgesinin seçimi
- İç döngü ve dış döngü koşulunun belirlenmesi

3.3.4.1. Çözümlerin gösterimi

Çözümlerin gösterimi, problemden probleme değişmektedir. Gösterim tekniği, SA algoritmasının performansını etkileyen önemli parametrelerden biridir. Çözümlerin gösterimi iki şekilde yapılır. Birincisi, ikili sayıların kullanıldığı ikili dizilerden oluşan gösterimdir. İkinci gösterim şekli ise tamsayı ve ya gerçel sayıların oluşturduğu dizilerdir. OVFS kod ağacı, aktif kullanıcıların indis bilgilerinden oluştuğu için çözümlerin gösterimi bu algorithmada tamsayı şeklindedir.

3.3.4.2. Maliyet fonksiyonu

SA algoritması ve çözümü aranan optimizasyon problemi arasındaki tek bağıdır. Olası çözümü bir giriş olarak alır ve çözümün ne derece karşılandığının ölçüsünü gösteren bir sayı üretir. Üretilen sayının aralığı, problemde problemde değişmektedir. Burada maliyet fonksiyonu olarak toplam yer değiştirme miktarı ele alınmıştır. Burada ifade edilen maliyet fonksiyonu GA'daki uygunluk fonksiyonuna karşılık gelmektedir.

3.3.4.3. Komşuluk hareketi

Komşuluk hareketi, SA algoritmasında önemli bir adımdır ve arama uzayında var olan çözüme en yakın olası yeni bir çözüm üretmek için kullanılır. Önceden kararlaştırılmış iç döngü koşulu sağlanıncaya kadar komşuluk hareketi adımı tekrarlanır ve olası yeni çözümler üretilir. Yeni çözümler üretmek için kullanılan iki komşuluk hareketi tekniği aşağıda işleyişleriyle birlikte sıralanmıştır (Leung ve diğ., 2001):

- Yerdeğiştirme komşuluk hareketi (swap neighborhood move),
- Öteleme komşuluk hareketi (shift neighborhood move).

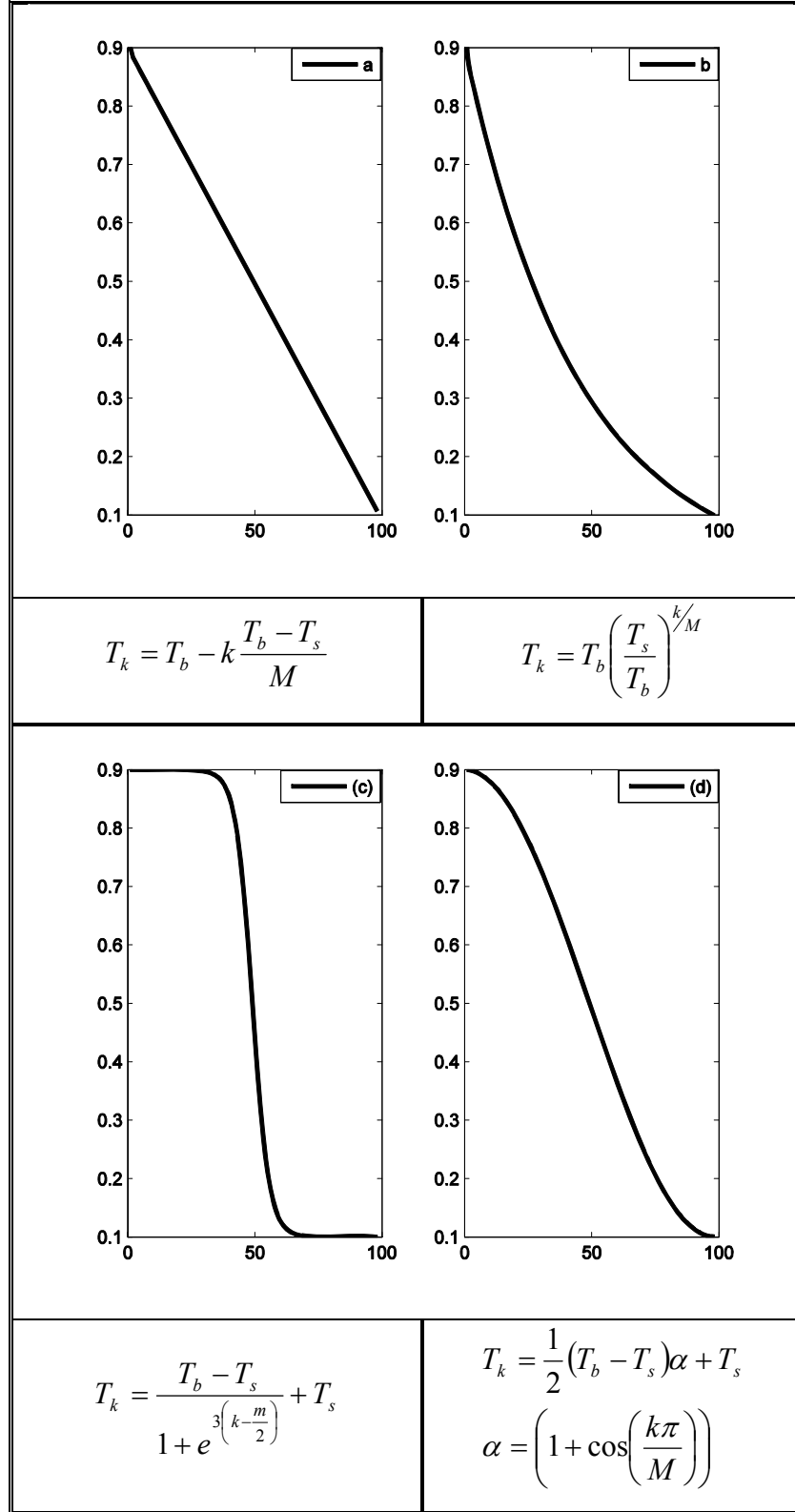
Yerdeğiştirme komşuluk hareketi, olası çözümü gösteren OVFS kod ağacı dizisi içinde yer değiştirmesi istenen rasgele iki elemanın seçilip, birbirlerinin yerine geçmeleri ile yapılır. Böylelikle, yeni olası çözümü gösteren bir OVFS kod ağacı dizisi elde edilir.

Ötelemeli komşuluk hareketi, OVFS kod ağacı dizisinden rasgele bir aktif kullanıcının indisi seçilip, bu indis bilgisinin yine rasgele seçilmiş olan başka bir aktif kullanıcının indis bilgisinin önüne yerleştirilmesi ve dolayısıyla diğer indis değerlerinin sağa doğru kayması ile elde edilir. Böylelikle, yeni olası çözümü gösteren yeni bir OVFS kod ağacı çıkmış olur.

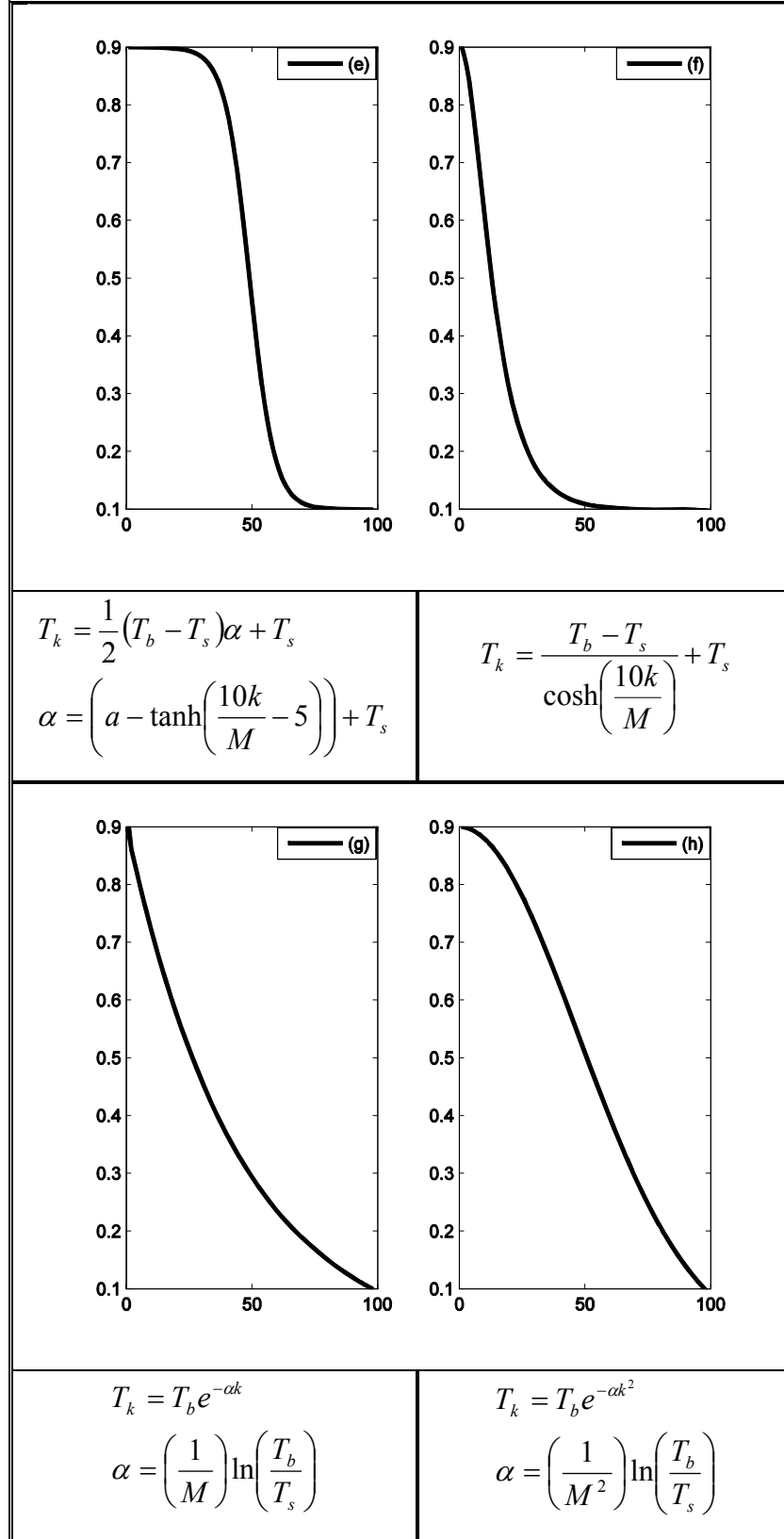
OVSF kod ağacının SA algoritması ile optimizasyonunda kullanılan komşuluk hareketi, GA mutasyon operatörüne benzemektedir. Komşuluk hareketi ile iç döngüde diğer olası çözümler araştırılır. Yani mevcut arama uzayında uygunluk değeri minimum olan noktanın seçilerek bu noktanın etrafında uygunluk değeri daha küçük olan başka yerel minimum noktaların olup olmadığına bakılır. Ayrıca bu minimum noktanın sistem için gerek koşulu (uygun boş OVSF kanallama kodu) sağlayıp sağlamadığı incelenir. Eğer OVSF kod ağacının diklik prensibine bağlı kalınarak uygun OVSF kanallama kodu sağlanıyorsa, talepte bulunan kullanıcıya bu kod tahsis edilir.

3.3.4.4. Soğutma çizelgesinin seçimi

SA algoritmasının performansı, genellikle her tekrarlama sıcaklık değerini güncelleyen soğutma çizelgesinin seçimine bağlıdır. Ayrıca soğutma işleminde kullanılan başlangıç sıcaklığı, son sıcaklık ve algoritmanın tekrarlama sayısının doğru seçimi de çözümün yakınsama hızı ve kesinliği açısından büyük bir önem taşımaktadır. Literatürde pek çok farklı soğutma çizelgesi vardır ve her biri sıcaklık güncellemek için farklı soğutma kuralı kullanır. Bunlardan bazıları Şekil 3.10'da verilmiştir (Leung ve diğ., 2001).



Şekil 3.10: Çeşitli soğutma çizelgelerinin soğuma eğrileri ve formülleri



Şekil 3.10 (devamı): Çeşitli soğutma çizelgelerinin soğuma eğrileri ve formülleri

Sıcaklık değerini güncelleyen bu çizelgelerden ikisi aşağıda işleyişleriyle birlikte verilmiştir (Lai ve Chen, 1997);

- Orantısal azalımlı soğutma çizelgesi,
- Lundy & Mees soğutma çizelgesi.

Orantısal azalımlı soğutma çizelgesinde; k . ve $(k+1)$. adımlardaki T_k ve T_{k+1} sıcaklıkları aralarındaki ilişki, belirli bir α katsayısıyla belirlenir. Her bir adımda belirlenen sıcaklık, aşağıdaki formülde ifade edilen eşitlikle yeniden hesaplanarak güncellenmektedir.

$$T_{k+1} = \alpha T_k \quad (3.8)$$

α katsayısı, algoritmanın her tekrarında başlangıç sıcaklığından son sıcaklık değerine ulaşana dek Denklem 3.9’da ifade edilen eşitlikte hesaplanan adımlarla azalmaktadır.

$$\alpha = M \sqrt{\frac{T_s}{T_b}} \quad (3.9)$$

Burada T_s son sıcaklığı, T_b başlangıç sıcaklığını ve M algoritmanın tekrarlama sayısını göstermektedir. α katsayısı, $(0, 1)$ arasında değişmektedir.

Lundy&Mees soğutma çizelgesinde; algoritmanın k . ve $(k+1)$. adımlarındaki T_k ve T_{k+1} sıcaklıkları, β katsayısıyla ilişkilendirilmiştir. Algoritmanın her tekrarında sıcaklık, hesaplanan β ’nın kullanımıyla bir önceki sıcaklığa göre azaltılır. Sıcaklığı güncellemek için

$$T_{k+1} = \frac{T_k}{1 + \beta T_k} \quad (3.10)$$

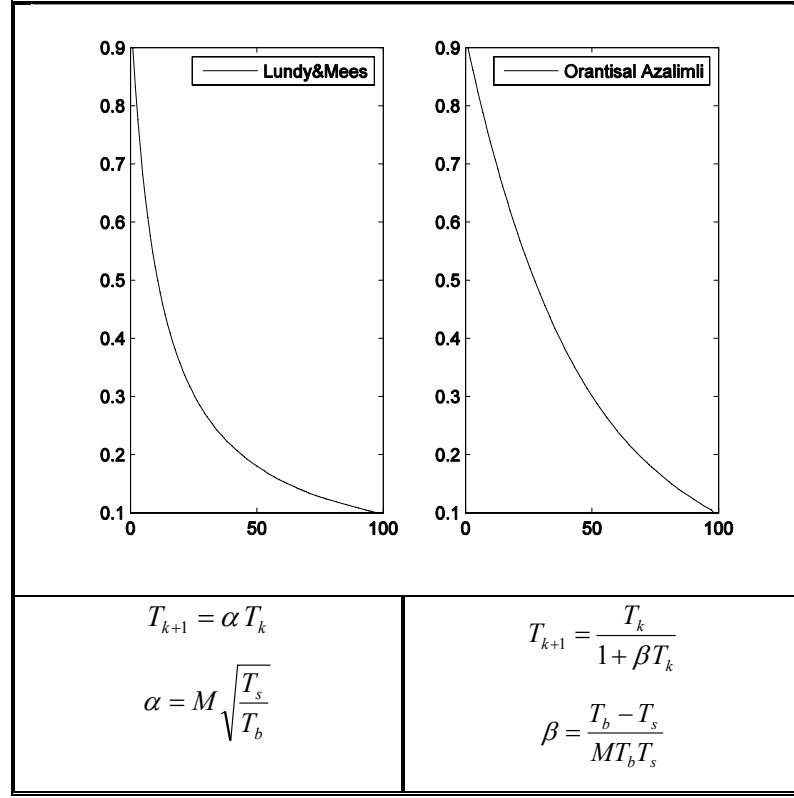
kullanılır. β katsayısı, algoritmanın tekrar sayısı ile başlangıç ve son sıcaklığa bağlı olarak Denklem 3.11’de ifade edilen eşitlikle hesaplanmaktadır.

$$\beta = \frac{T_b - T_s}{MT_b T_s} \quad (3.11)$$

Bu eşitlikte, T_b başlangıç sıcaklığını T_s son sıcaklığı ve M algoritmanın tekrarlama sayısını göstermektedir. β katsayısı da 0'dan büyük bir değerdir.

Şekil 3.10'da verilen soğutma çizelgeleri için $x=0$ noktası üzerindeki y değeri, başlangıç sıcaklığını (T_b) ve $x=M$ noktası üzerindeki y değeri de, son sıcaklığı (T_s) vermektedir. Şekil 3.10'da tüm figürler için $M=100$ alınmıştır. Şekillerin başlangıç ve bitiş noktaları arasındaki izlediği yola göre bütünsellikten yerelliğe doğru nasıl hareket ettiği hakkında bilgi sahibi olabiliriz. Örneğin, Şekil 3.10a'da doğrusal bir azalma durumu söz konusudur. Şekil 3.10f'de bütünsellikten hemen yerelliğe geçilip yerelde aramanın daha fazla olacağı hakkında bir yorum, Şekil 3.10h'de ise bütünsel olarak daha fazla arama yapıldığı yorumu yapabiliriz.

OVSF kod ağacının SA algoritması ile optimizasyonunda karşımıza iki problem çıkmaktadır. Bunlardan biri T_s 'nin belirlenmesi, ikincisi ise M 'nin belirlenmesidir. Çalışmanın başında da belirttiğimiz üzere amaç, ihtiyaç duyulan veri transfer hızına uygun OVSF kanallama kodunu, kod ağacında oluşturmaktır. Algoritmanın çalışma prensibine göre oluşturulan yeni ağacın uygunluk değerinin mevcut ağaca göre daha küçük olacağı açıktır. Ancak minimum olacak diye bir garanti yoktur. SA, yeni kullanıcıya ihtiyaç duyduğu OVSF kanallama kodunu tahsis edecek şekilde kod ağacını oluşturursa, o zaman algoritmanın çalışması bitmiş demektir. Yani yeter koşul sağlandığı zaman algoritma durur. Bu da soğutma çizelgesinin her zaman tamamlanmayacağı anlamına gelmektedir. SA algoritmasında kullanılan Lundy&Mees ve orantısal azalımlı soğutma çizelgelerinin grafikleri Şekil 3.11'de verilmiştir.



Şekil 3.11: Lundy&Mees ve orantısal azalımlı soğutma çizelgeleri

3.3.4.5. İç döngü ve dış döngü koşulunun belirlenmesi

SA algoritmasının arama süreci boyunca iki döngü kullanır. Bunlar; iç döngü ve dış döngüdür. İç döngü koşulu, her sıcaklık değerinde kaç kez olası yeni çözüm üretileceğine karar verir. Dış döngü ise algoritmanın sonlandırma koşuludur. İç döngü koşulu sağlandığında sıcaklık güncellenir ve dış döngü koşulu yeniden kontrol edilir. Dış döngü sağlandığında ise SA durur.

SA algoritması uygulanırken iç döngü koşulunun sağlanıp sağlanmadığını kontrol edebilmek için aşağıda sıralanan koşullardan biri kullanılabilir (Shahookar ve Mazumder, 1991):

- Olası en çok yeni çözüm üretme sayısı.
- Olası yeni çözümler üretirken maliyet fonksiyonunda iyileşme olmaması hali.
- Yeteri kadar yeni çözüm üretiminin gerçekleştirilmesi.

Bütün bir arama sürecini sonlandırmak için dört çeşit durdurma koşulu kullanılabilir. Bunlar (Leung ve diğ., 2001);

- Algoritmanın tekrarlama sayısı,
- Maliyet miktarları arasındaki değişim miktarı,
- Sistemin son sıcaklığı,
- Yakınsama koşulunun sağlanması.

Bu koşulların, hesaplama sürelerine ve çözüm kalitelerine bakılarak içlerinden çalışılan problem için en uygun olanı seçilebilir. OVFS kod ağacının SA algoritması ile optimizasyonunda, iç ve dış döngü için yukarıda bahsedilen koşullardan bazıları kullanılmıştır. İç döngü için, yeteri kadar yeni çözüm üretiminin gerçekleştirilmesi durumu ele alınmıştır. Her sıcaklık değeri için beş tane daha yakın komşulukta OVFS kod ağacı üretilerek bunların içlerinden uygunluk değeri en küçük olan iç döngüden çıkar. Yani iç döngü ile yerelde yakın komşuluklarda yapılan aramalar sonucunda en iyi uygunluk değerini veren OVFS kod ağacı döngüden çıkar. Dış döngü içinse yakınsama koşulunu sağlaması göz önünde bulundurulmuştur. Burada, belirli bir veri transfer hızında OVFS kanallama koduna ihtiyaç duyan kullanıcıya, ihtiyaç duyduğu OVFS kodu, kod ağacında sağlandığı anda dış döngü durur. Bununla birlikte dış döngünün yanıt vermediği durumlar söz konusudur. Örneğin arama uzayında SA algoritması ile yapılan tüm aramalar sonucunda ihtiyaç duyulan veri transfer hızına uygun boş OVFS kanallama koduna sahip bir kod ağacı oluşturulamaz ise bu durumda dış döngü 1000 kere bu arama işlemini sürdürür. Sonuçta uygun boş OVFS kanallama koduna sahip bir kod ağacı üretebiliyorsa bu durumda kod, talepte bulunan kullanıcıya tahsis edilir. Aksi takdirde 1000 dış döngü sonucunda bir yanıt alınamıyorsa arama bloklanır ve talep reddedilir.

3.4. Uyarlamalı Benzetilmiş Tavlamalı Genetik Algoritma Tabanlı Dinamik OVFS Kod Tahsisi

Yapılan çeşitli çalışmalar sonucunda gerek SA algoritmasının gerekse GA'nın verimli ve güvenilir sonuçlar elde ettiği gözlenmiştir. Problemin, kullanılan tekniğe

göre uygun bir şekilde yorumlandığında amaca uygun sonuçlar verdiği açıktır. Ancak GA için erken (premature) yakınsama ve SA için arama süresinin çok uzun olması dezavantaj olarak görülmektedir. Erken yakınsama dominance&diploidy kavramı ile ortadan kaldırılrsa da, sisteme getirilecek olan işlem yükü olumsuz etki yaratmaktadır (Karakoç ve Kavak, 2007). Bundan dolayı SA ve GA'nın bir arada kullanılması düşüncesi ortaya atılmıştır. Literatürde SA ile GA'yı bir arada kullanan çeşitli çalışmalar mevcuttur. Bu iki algoritmanın birleştirilmesi çeşitli metotlarla yapılmaktadır. Ancak amaç, her iki algoritmanın da özüne zarar vermeden daha iyi performans sergileyen bir yapının oluşturulmasıdır. Bu metotlardan bazılarında GA'nın çaprazlama operatörü yerine SA algoritması kullanılmış, bazılarında mutasyon operatörü yerine SA algoritması kullanılmıştır (Tan ve diğ., 1995). Literatürdeki bazı çalışmalarda GA'nın çaprazlama ve mutasyon operatörleri kullanılarak elde edilen kromozomlar, SA algoritması sonuçları ile karşılaştırılmış ve uygunluk değeri en küçük olan kromozom olası çözüm olarak ele alınmış ve onun çevresinde yerel aramalar yapılmıştır (Leung ve diğ., 2003). Li ve diğ., (1995) tarafından yapılan çalışmada belirtildiği gibi GA tarafından üretilen her bir kromozoma da SA uygulanabilir. OVFS kod ağacının optimizasyonunda, GA'nın çaprazlama ve mutasyon operatörü ile elde edilen kromozomların SA algoritması ile değerlendirilmesi ve uygunluk değeri daha iyi olan kromozomun olası çözüm kabul edilerek işlenmesi söz konusudur. Oluşturulan bu yapının akış diyagramı Şekil 3.12'de verilmektedir.

Şekilden de görüldüğü gibi, çaprazlama ve mutasyon operatörleri tarafından elde edilen kromozomlar, sahip oldukları uygunluk değerleri cinsinden karşılaştırılırlar. Eğer $\Delta C < 0$ ise bu durumda $F_c > F_m$ durumundadır ve olası çözüm olarak mutasyon operatöründen dönen kromozomlar alınır. $\Delta C > 0$ olması durumunda ise $F_m > F_c$ koşulu sağlanmış demektir. Bu durumda belli bir olasılık dahilinde çaprazlama operatöründen dönen kromozom olası çözüm olarak alınır, aksi takdirde mutasyon operatörüne geri dönülür. Bazı çalışmalarda bu geri dönüş rotası çaprazlama operatörü girişi olarak belirlenmiştir. İç döngü koşulu sağlanana kadar algoritma mutasyon operatörüne geri dönerek uygunluk değeri daha iyi olan kromozomu elde etmeye çalışır. İç döngü koşulu sağlandığında ise optimizasyon koşulunun sağlanıp sağlanmadığı test edilir. Akış diyagramında verilen optimizasyon koşulu blokları, yeni kullanıcıya ihtiyacı olan veri transfer hızını karşılayacak OVSF kanallama kodunun sağlanıp sağlanmadığını test etmek için oluşturulmuştur. Eğer koşul sağlanıyorsa, algoritma görevini yapmış demektir ve OVSF kanallama kodu mobil kullanıcıya tahsis edilir ve algoritma sonlandırılır. Tersisi durumda algoritma, akış diyagramında verilen görevi yerine getirmek üzere bir sonraki bloğa gider. Bu işlem dizisi, yeni kullanıcıya ihtiyaç duyduğu uygun OVSF kanallama kodu tahsis edilene kadar ya da dış döngü koşulu sağlanana kadar devam eder. Dış döngü koşulu sağlandığında yine uygun OVSF kanallama kodunun atanıp atanmayacağı kontrol edilir, eğer atanmıyorsa çağrı bloklanır. Uyarlamalı benzetilmiş tavlama genetik algoritma (ASAGA) yapısının sözde kodu Ek-A'da verilmektedir.

ASAGA, dominance & diploidy kavramı ile de beraber kullanılabilir. Burada, D&D-GA'da belirtilen reproduction aşamasında hibritleşen çiftlerden hangilerinin alınması gerektiği SA ile belirlenebilir. Ancak bu çalışmada, yukarıda belirtilen çaprazlama ve mutasyon operatörleri tarafından elde edilen kromozomların SA algoritması ile işlenerek uygun olanının seçimine dayanan teknik kullanılmıştır.

4. SİNYAL-PARAZİT ORANI (SIR) SINIRLI ALGORİTMALAR

4.1. Giriş

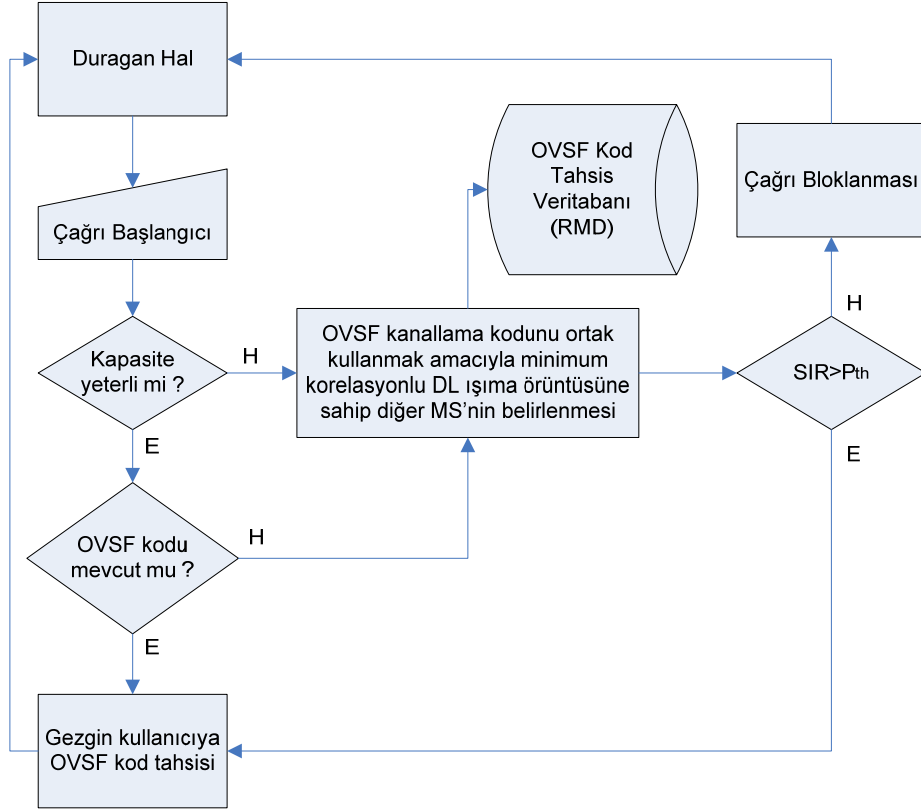
WCDMA sistemlerde değişken oranlı veri transferini gerçekleştirmek için OVSF kodları kullanılır. OVSF kod ağacının sınırlı yapıda olmasından dolayı, bu kaynağın verimli bir şekilde kullanılması ve yönetilmesi önemlidir. Bu bağlamda ilk bölümde geleneksel (CCA) ve dinamik (DCA) kod tahsis algoritmaları, üçüncü bölümde de evrimsel tabanlı dinamik kod tahsis algoritmaları (SGA, D&D-GA, SA-DCA, ASAGA) incelenmiştir. Bu algoritmaların hepsi kod sınırlı yapıdadır. Yani yeni kullanıcıya tahsis edebilecek maksimum OVSF kanallama kodu, kod ağacındaki boş kodların veri transfer hızları toplamı ile sınırlıdır. BTS’de uyarlamalı anten dizisinin (AAA) kullanımı ile mobil kullanıcının (MS) uzaysal çeşitlilik bilgisi, OVSF kanallama kodu atamasında kullanılabilir. Farklı uzaysal koordinatlarda bulunan kullanıcıların DL ışıma örüntülerinin farklılığından dolayı bu kullanıcılar aynı kodu paylaşabilirler. Burada gerek koşul, istenilir kullanıcı tarafından alınan sinyal gücünün standartlarca belirlenen seviyelerde parazit sinyal gücünden daha güçlü olmasıdır. Böylece OVSF kanallama kodu atama stratejisi, SIR sınırlı yapı haline dönüşür (Karakoç ve Kavak, 2007).

SIR sınırlı uyarlanabilir anten dizisi tabanlı dinamik kod tahsisi (AAA-DCA) algoritmasını oluşturan yapının genel faydası üç ayrı başlıkta ele alınmıştır. İlk olarak BTS’de üç sektörlü düzgün doğrusal anten dizisi (ULA) ya da düzgün dairesel anten dizisi (UCA) kullanıldığında meydana gelen ve karışık ses veri trafik senaryolarını içeren WCDMA sistemin DL kapasite artışı ele alınmıştır. Ayrıca hem ses hem de veri kullanıcılarından kaynaklanan hücreiçi (intracell) ve hücrelerarası (intercell) parazit modellenmiştir. Bu modelde DL kanal vektörü oluşturulurken, çözümlenebilir çokluyol (multipath) etkileri de göz önünde bulundurulmuştur. İkinci olarak karışık trafik senaryosu için sistem çıkış eşitliği ele alınmıştır. Burada elde

edilen değer, aynı OVSF kanallama kodunun birden fazla kullanıcıya tahsis edilemeyeceğini belirleyen eşik değeridir. Son olarak da farklı faktörlerin (anten dizisi topolojisi, çoklu yol açısal yayılımı, ışınma örüntüsü algoritması) etkisi ele alınmıştır. Türetilmiş parazit, sistem çıkış modelinin kullanılması ve Monte Carlo simülasyonu ile WCDMA sistem kapasitesi, farklı faktörler (ULA ve UCA topolojileri (değişken elemanlar arası mesafe Δ , çoklu yol açısal yayılımlar)) ve ışınma örüntüsü algoritmaları (ortalama DOA, baskın DOA ve ağırlıklı ortalama DOA) ışığında ele alınmıştır.

4.2. Uyarlanabilir Anten Dizisi Tabanlı Dinamik Kod Tahsis Algoritması (AAA-DCA)

Şekil 4.1, BTS’de kaynak yönetimini gerçekleştiren AAA-DCA algoritmasına ait akış diyagramını göstermektedir. Akış diyagramına göre durağan halde herhangi bir çağrı talebi yoktur. İhtiyaç duyulan veri transfer hızındaki OVSF kanallama kodu, çağrı başlangıcı ile kaynak yöneticisine çağrı istem işlemci sinyali ile gelir. İlk olarak, OVSF kod ağacında kullanılmayan kodların hızlarının toplamı belirlenir. Bu toplam sistem kapasitesi olarak adlandırılır. Eğer OVSF kod ağacı kapasitesi, yeni kullanıcıya OVSF kanallama kodu tahsis edecek kadar boş dala sahipse ihtiyaç duyulan veri transfer hızını karşılayabilecek OVSF kanallama kodu, ilgili katmandaki kullanılmayan kodlar arasında araştırılır. Eğer çağrı OVSF kod ağacının yeterli kapasiteye sahip olmamasından ya da ihtiyaç duyulan veri transfer hızına uygun OVSF kanallama kodunun bulunmamasından dolayı gerçekleştirilemiyorsa, algoritmanın DL ışınma örüntüsü vektör kod arama kısmı icra edilir.



Şekil 4.1: AAA-DCA algoritması kullanılarak OVSF kod ağacının optimizasyonunu gerçekleştiren akış diyagramı

İhtiyaç duyulan OVSF kanallama kodunu ortak kullanabilmek için yeni kullanıcının kanal vektörü ($\mathbf{a}_{yeni}$) ile minimum ilintiye sahip DL ışıma örüntüsünü ($\mathbf{b}_k$) kullanan aktif kullanıcı ve ya kullanıcıların OVSF kodu araştırılır.

$$k = \arg \left\{ \min \left\{ \left\| \mathbf{b}_k^H \mathbf{a}_{yeni} \right\| \right\} \right\} \quad (4.1)$$

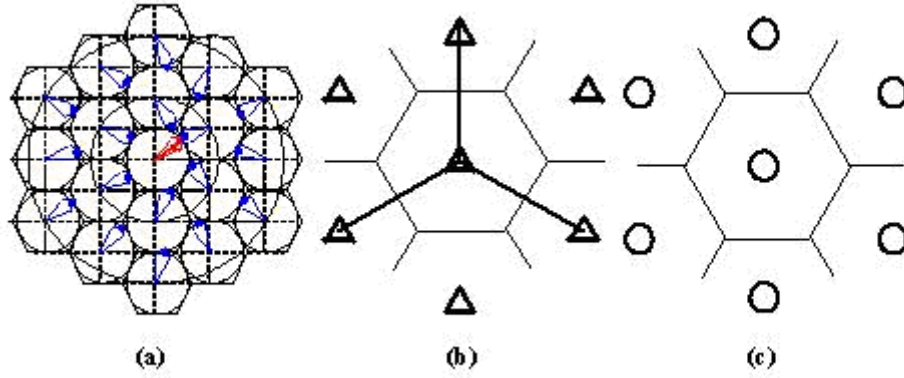
Burada, $k=1, \dots, B$; B mevcut aktif kullanıcıların DL ışıma örüntülerinin sayısı, H ise kompleks eşlenik ile transpoze alma işlemidir. Bu bölümde yapılan araştırmalar aynı DL ışıma örüntüsünü kullanan birden fazla kullanıcı varsa, en az kullanıcı tarafından kullanılan kodun yeni kullanıcıya tahsis edilmesi gerektiğini göstermektedir. Yeni kullanıcıya ihtiyaç duyduğu OVSF kodunun tahsis edilmesi için ilk koşul, standartlarca belirlenmiş anlık SIR seviyesinin P_{th} aşılmış olmasıdır.

AAA-DCA algoritması, ilk olarak çoklu yol etkisi göz önünde bulundurularak modellenmiştir. Daha sonra bu etki, sistemde ihmal edilerek yeni bir sistem çıkış denklemi oluşturulmuştur.

4.3. Çoklu Yol Yayılım Etkisinin Gözönünde Bulundurulması

4.3.1. Sinyal modeli

Bu bölümde CDMA hücresel ağının ilk iki katmanı ele alınmıştır (Kavak, 2001). Hücrede bulunan AAA, Şekil 4.2’de görüldüğü gibi ya üç sektörlü ULA ya da sektörlere ayrılmamış UCA topolojisini içerir.



Şekil 4.2: (a) İki katmanlı CDMA hücresel ağ yapısında parazit doğrultuları, (b) ULA topolojisi, (c) UCA topolojisi

Hücredeki her BTS, her biri yönsüz doğrusal anten elemanı ile donatılmış çeşitli sayıdaki MS'lere servis hizmeti sunar. Hücredeki k . baz istasyonu tarafından R_j bit/saniye veri transfer hızına sahip, j . sınıf (veri transfer sınıfı) N_j adet kullanıcıya servis hizmetinin sunulduğu basit bir senaryoyu ele alalım. Burada CDMA kullanıcıları hücre içinde düzgün olarak dağılsın. DL için yayılım faktörü (işlemci kazancı) $SF=W/R$ ile ifade edilir. R birim hızdaki veri transfer hızını, W ise tahsis edilmiş spektral bantgenişliğini göstermektedir. SF işlemci kazancı, aynı zamanda sembol periyodun çip periyoduna oranı olarak da adlandırılabilir ($SF=T_w/T_c$). j . sınıf bir kullanıcı için çoklu-kod kanal iletim yönteminin kullanıldığını varsayalım (Torrieri, 1992). Buna göre her bir kullanıcıya ait R_j veri transfer veri yığınları, Q_j adet paralel kod kanalına yayılır. Burada $Q_j=R_j/R$ ile verilir ($j=1, 2, \dots, (\log_2 SF)+1$). Farz edelim ki $b_{k,i,q}^{(j)}(l)$, j . sınıf kullanıcıya ait sıfır ortalamalı, istatistik olarak bağımsız (i.i.d.) ve birim varyanslı bilgi sembollerini içeren bir dizi olsun ve bu dizi k . hücredeki i . kullanıcının q . paralel kod kanalı vasıtasıyla iletilsin. Bunun yanı sıra

$c_{k,i,q}^{(j)}(l,m)$ ile ifade edilen terim bu kullanıcının l . sembolünün çip yayılım kodu olsun. Burada $m=0, 1, \dots, Q-1$ 'dir. İfade edilen bu duruma karşılık gelen sürekli zaman yayılım sinyali

$$\tilde{c}_{k,i,q}^{(j)}(l;t) = \sum_{m=0}^{\left(\frac{SF}{Q_j}\right)-1} c_{k,i,q}^{(j)}(l;m)p(t-mT_c) \quad (4.2)$$

ile yazılabilir. Burada $p(t)$, birim enerjili çip darbe şekillendirici filtredir. BTS tarafından j . sınıf i . kullanıcı için iletilen temel bant sinyal

$$s_{k,i}^{(j)}(t) = \sum_{q=1}^{Q_j} \sum_{l=-\infty}^{\infty} b_{k,i,q}^{(j)}(l)\tilde{c}_{k,i,q}^{(j)}(l;t) \quad (4.3)$$

ile verilir.

BTS'ler anten dizileri ile donatıldıklarından $\mathbf{w}_{k,i}^{(j)}$ ile ifade edilen terim, j . sınıftaki i . kullanıcı için DL ışıma örüntüsü (beamforming) ağırlık vektörü olarak ifade edilir. İletim amacı ile hücredeki her bir kullanıcıya ait CDMA dalga formu, kendisine karşılık gelen ilgili ağırlık vektörü ile çarpılarak aynı hücredeki diğer kullanıcıların sinyalleri ile toplanır. Dolayısıyla BTS'den ilgili MS'e doğru iletilen tüm sinyaller aynı iletim kanalı üzerinden yayılırlar. Böylece tüm sinyallerde aynı zayıflama ve yol kaybı etkisi görülür. Yönsüz anten elemanına sahip olan MS'ler sadece kendilerine ait olan sinyalleri değil, aynı hücrede bulunan diğer kullanıcılardan kaynaklanan sinyalleri (intracell interference) ve MS ile aynı frekans kanalını kullanan diğer hücrelerdeki kullanıcılardan gelen sinyalleri (intercell interference) de alacaktır. Farz edelim ki BTS k 'dan MS'e doğru çoklu yol indeksleri f ve çoklu yolların sayıları F_k ile ifade edilsin. k . hücrede, her bir çoklu yol bileşenine ait kanal parametreleri yani $\{\theta_{k,f}\}$ geliş açısı $\{\tau_{k,f}\}$ çoklu yol yayılım gecikmesine göre $\{\beta_{k,f}\}$ zayıflama parametreleri de (büyük ölçekli ve küçük ölçekli zayıflamaları içeren) istenilir kullanıcının geliş açısına göre belirlensin. Bu bileşenlerden $\{\beta_{k,f}\}$ zayıflama parametresi

$$\beta_{k,f} = \frac{\alpha_{k,f}^0}{(r_k^0)^2} \quad (4.4)$$

ile belirlenir. Burada r_k^0 , MS'in hücre merkezine olan uzaklığıdır, $\alpha_{k,f}^0$ parametresi de, MS'e f . çoklu yol doğrultusunda gelen tüm sinyallerin etkisine maruz kaldığı zayıflama (fading) ve gölgelenme (shadowing) etkisidir. Bu etkinin oluşturduğu değer, sıfır ortalamalı karmaşık Gauss dağılımlı rasgele değişkendir. Bu değişkenin genliği $\|\alpha_{k,f}^0\|$, f . çoklu yol doğrultusundaki ortalama karekök değeri $E\{\|\alpha_{k,f}^0\|^2\}$ log-normal dağılımlı Rayleigh dağılımı ile modellenmiş, çoklu yol zayıflaması ile ifade edilir. Yani $10\log_{10} E\{\|\alpha_{k,f}^0\|^2\}$ sıfır ortalamalı ve varyansı σ_s^2 ile verilen normal bir dağılımdır (Naguib ve diğ., 1994).

MS tarafından alıcı girişinde alınan temel bant sinyal

$$\mathbf{x}_k^{(j)}(t) = \sum_{i=1}^{N_j} \sqrt{P_{k,i}^{(j)}} \sum_{f=1}^{F_k} \psi_{k,i}^{(j)} \beta_{k,f} s_{k,i}^{(j)}(t - \tau_{k,f}) \mathbf{a}^H(\theta_{k,f}) \mathbf{w}_{k,i}^{(j)} \quad (4.5)$$

ile ifade edilir. Burada, $\psi_{k,i}^{(j)}$ Bernoulli rasgele değişkeni ile modellenmiş, v_j ile verilen olasılığı sabit ve tipik olarak ses için 0.4, veri için 0.5 değeri esas alınan ses aktiflik faktörüdür (Evans ve Everitt, 1999). Farz edelim ki her hücre, anlık mükemmel güç kontrol mekanizmasını uygulasin. Yani diğer bir deyişle BTS tarafından aynı sınıftaki tüm kullanıcılara aynı güç seviyesi iletilsin. Böylelikle $P_{k,i}^{(j)}$ ile ifade edilen güç değeri i kullanıcı indeksinden bağımsız hale gelir ve $P_k^{(j)}$ kısaltmasıyla tekrar yazılarak j . sınıf kullanıcı için k . BTS tarafından iletilen güç seviyesi olarak adlandırılır.

Denklem 4.5'te tek hücre için vermiş olduğumuz açılımı, iki katmanlı K+1 adet hücreye sahip CDMA hücresel ağı için genişlettiğimizde

$$x(t) = \sum_{k=1}^K \sum_{j=1}^J x_k^{(j)}(t) + n(t) \quad (4.6)$$

elde ederiz. Burada J toplam sınıf indisini verir ($j=1,2,\dots,J$). $n(t)$ ise sıfır ortalamalı ve varyansı σ_n^2 olan karmaşık gürültü terimidir. Burada aynı hücrede bulunan diğer kullanıcılardan kaynaklanan ve MS ile aynı frekans kanalını kullanan diğer hücrelerdeki kullanıcılardan kaynaklanan gürültü bileşenleri ele alınmamıştır.

4.3.2. Parazit analizi ve sistem çıkış denklemi

Merkezi hücrede bulunan ($k=0$) MS'in alıcı biriminde, her paralel kod kanalı için alınan temel bant sinyalin, kendisine karşılık gelen kod ile F_0 elemanlı RAKE alıcısında ilintilendirildiği varsayılır. Herhangi bir zaman anında BTS, farklı sınıflardaki kullanıcılara iletim hizmeti sunar. Burada R veri transfer hızına sahip olan kullanıcı için $j=1$ iken, bu değer $2R$ veri transfer hızında $j=2$, $4R$ veri transfer hızında $j=3$, $SF \times R$ veri transfer hızında ise $j=(\log_2 SF)+1$ değerindedir. Farz edelim ki MS, her bir çoklu yol sinyaline ait yayılım gecikmelerini bilmektedir, yani τ_γ , $\gamma = 1,2,\dots,F_0$. Verilen bir zaman aralığında η . sınıf (η notasyonu, istenilir MS'e ait sınıfı belirtmek amacı ile kasıtlı olarak kullanılmıştır) bir MS için, kodu l . sembol periyodu boyunca $\tilde{c}_{d,h}^{(\eta)}(l;t-\tau_\gamma)$ olan d . MS alıcısının h . kod kanalının γ . çoklu yolu için korelatör çıkışı

$$z_\gamma^{(\eta)}(l) = \frac{1}{T_c} \int_{(l-1)T_w+\tau_\gamma}^{lT_w+\tau_\gamma} x(t) \tilde{c}_{d,h}^{(\eta)*}(l;t-\tau_\gamma) dt \quad (4.7)$$

ile verilir. Burada T_c , çip periyodunu vermektedir. $()^*$ ifadesi de karmaşık eşlenik alma işlemini belirtmektedir. Analizin açıkça gösterilebilmesi adına bundan sonraki denklem türetimleri l . MS'in ilk kod kanalı için yapılacaktır. Bundan dolayı 4.7 no'lu denklemde yer alan d ve h terimleri l olarak alınır. Birleştirme (combining) metodu olarak da eşit kazançlı birleştirme (equal gain combining-EGC) uyumlu RAKE yapısı uygulanacaktır (Kaizhi ve diğ., 2003). Sonuç istatistiği,

$$z^{(\eta)}(l) = \sum_{\gamma=1}^{F_0} v_{\gamma}^{*(\eta)} z_{\gamma}^{(\eta)}(l) = s^{(\eta)}(l) + n_0^{(\eta)}(l) + n_k^{(\eta)}(l) + n_T^{(\eta)}(l) \quad (4.8)$$

ile verilir. Burada, γ terimi RAKE alıcı için alt alıcı dal indis numarasını ifade eder.

$v_{\gamma}^{*(\eta)}$ EGC katsayısı olarak adlandırılır ve

$$v_{\gamma}^{*(\eta)} = \frac{h_{\gamma}^{(\eta)}}{\|h_{\gamma}^{(\eta)}\|} \quad (4.9)$$

ile ifade edilir. Burada $h_{\gamma}^{(\eta)} = \beta_{\gamma} \mathbf{a}^H(\theta_{0,\gamma}) \mathbf{w}_{0,1}^{(\eta)}$ eşitliği ile verilir. EGC katsayısının (ağırlık faktörü değerlerinin), pilot kanal vasıtası ile alıcıda bilindiği ya da tahmin edildiği varsayılır (Alouni ve Simon, 2001). 4.8 no'lu denkleme göre $s^{(\eta)}(l)$ istenilir sinyal bileşeni, $n_0^{(\eta)}(l)$ aynı hücrede bulunan diğer kullanıcılardan kaynaklanan parazit sinyalleri (intracell interference), $n_k^{(\eta)}(l)$ MS ile aynı frekans kanalını kullanan diğer hücrelerdeki kullanıcılardan gelen parazit sinyaller (intercell interference) ve $n_T^{(\eta)}(l)$ ise gürültü terimini ifade etmektedir. Burada $k=1, 2, \dots, K$ ile verilir. 4.7 no'lu denklemde verilen integral sonunda elde edilen çarpımın basitleştirilmesi için

$$I_{k,i,q,f,\gamma}^{(j,\eta)} = \frac{1}{T_c} \int_{(l-1)T_w + \tau_{\gamma}}^{lT_w + \tau_{\gamma}} b_{k,i,q}^{(j)}(l) \tilde{c}_{k,i,q}^{(j)}(l; t - \tau_f) \tilde{c}_{1,1}^{(\eta)*}(l; t - \tau_{\gamma}) dt \quad (4.10)$$

integral terimi tanımlanmıştır. $I_{k,i,q,f,\gamma}^{(j,\eta)}$ integral terimi, sıfır ortalamalı ve varyansı SF olan ilintisiz (uncorrelated) rasgele değişkendir .

4.8 no'lu denklemde ifade edilen istenilir sinyale ait bileşen

$$s^{(\eta)}(l) = s_{istenilir}^{(\eta)}(l) + s_{iç parazit}^{(\eta)}(l)$$

$$s^{(\eta)}(l) = \psi_1^{(\eta)} b_{1,1}^{(\eta)}(l) L \sqrt{\widetilde{P}_0^{(\eta)}} \sum_{f=1}^{F_0} |\beta_f| \left| \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,1}^{(\eta)} \right| + \psi_1^{(\eta)} \sqrt{\widetilde{P}_0^{(\eta)}} \sum_{\gamma=1}^{F_0} \sum_{\substack{f=1 \\ f \neq \gamma}}^{F_0} v_{\gamma}^{*(\eta)} \beta_f \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,1}^{(\eta)} I_{1,1,f,\gamma}^{(\eta,\eta)} \quad (4.11)$$

ile verilir. Burada ilk terim ($s_{istenilir}^{(\eta)}(l)$), aynı çoklu yol için ($\gamma=f$) sinyal bileşenini ifade eder. İkinci terim ($s_{iç parazit}^{(\eta)}(l)$) ise çapraz çoklu yollardan kaynaklanan ($\gamma \neq f$) içparazit bileşenini ifade eder. 4.11 no'lu denklemin türetilmesinde, aynı kullanıcıya ait tüm kod kanallarının aynı vektör kanalı vasıtasıyla gittiği ve aynı kullanıcıya ait farklı kod kanallarından kaynaklanan içparazit terimlerinin göz ardı edildiği kabul edilmiştir. 4.8 no'lu denklemde ifade edilen hücreiçi ve hücrelerarası parazit değerleri tam olarak aşağıdaki denklemlerde verilmiştir.

$$n_0^{(\eta)}(l) = \sum_{j=1}^{(\log_2 SF)+1} n_{0,j}^{(\eta)}(l) = \sum_{j=1}^{(\log_2 SF)+1} \sqrt{\widetilde{P}_0^{(j)}} \sum_{\gamma=1}^{F_0} v_{\gamma}^{*(\eta)} \sum_{i=1}^{N_j} \psi_i^{(j)} \sum_{f=1}^{F_0} \beta_f \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,i}^{(j)} \sum_{q=1}^{Q_j} I_{0,i,q,f,\gamma}^{(j,\eta)} \quad (4.12)$$

Burada η . sınıfa ait iç parazit hesabında $i=2$ 'den başlayıp N_{η} sayısal değerine kadar artmaktadır.

$$n_k^{(\eta)}(l) = \sum_{j=1}^{(\log_2 SF)+1} n_{k,j}^{(\eta)}(l) = \sum_{j=1}^{(\log_2 SF)+1} \sum_{\gamma=1}^{F_0} v_{\gamma}^{*(\eta)} \sum_{k=1}^K \sum_{i=1}^{N_j} \psi_{k,i}^{(j)} \sqrt{\widetilde{P}_k^{(j)}} \sum_{f=1}^{F_0} \beta_{k,f} \mathbf{a}^H(\theta_{k,f}) \mathbf{w}_{k,i}^{(j)} \sum_{q=1}^{Q_j} I_{k,i,q,f,\gamma}^{(j,\eta)} \quad (4.13)$$

ve gürültü terimi ise

$$n_T^{(\eta)}(l) = \sum_{\gamma=1}^{F_0} v_{\gamma}^{*(\eta)} \frac{1}{T_c} \int_{(l-1)T_w + \tau_{\gamma}}^{lT_w + \tau_{\gamma}} n(t) c_{1,1}^{(\eta)*}(l; t - \tau_{\gamma}) dt \quad (4.14)$$

İstenilir MS kullanıcılarının sahip olduğu bit enerjisinin, parazit ve gürültü toplamına oranını hesaplamak için istenilir sinyal ve parazit sinyal bileşenlerine ait güç ifadesi varyanslarının hesaplanması gerekir.

$$\text{var}\{s^{(\eta)}(l)\} = \underbrace{\psi_{0,1}^{(\eta)} \tilde{P}_0^{(\eta)} L^2 R_0^{(\eta)}}_{s^{(\eta)}(l)} + \underbrace{\psi_{0,1}^{(\eta)} \tilde{P}_0^{(\eta)} L(F_0 - 1) R_{0,i\text{ç parazit}}^{(\eta)}}_{g^{(\eta)}(l)} \quad (4.15)$$

ile verilir. Burada $s^{(\eta)}(l)$ istenilir sinyal bileşeni, $g^{(\eta)}(l)$ ise çoklu yollardan kaynaklanan istenilir sinyale bileşenine ait iç parazit bileşenidir. $R_0^{(\eta)}$ ve $R_{0,i\text{ç parazit}}^{(\eta)}$ olarak tanımlanan terimler ise

$$R_0^{(\eta)} = \left(\sum_{f=1}^{F_0} |\beta_{0,f}| \left| \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,1}^{(\eta)} \right| \right)^2 \quad (4.16)$$

ve

$$R_{0,i\text{ç parazit}}^{(\eta)} = \sum_{f=1}^{F_0} |\beta_{0,f}|^2 \left| \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,1}^{(\eta)} \right|^2 \quad (4.17)$$

ile verilir. Hücreiçi parazit, hücrelerarası parazit ve gürültü etkilerinden kaynaklanan parazit gücü ise

$$g_0^{(\eta)} = \text{var}\{n_0^{(\eta)}(l)\} = \sum_{j=1}^{(\log_2 SF)+1} Q_j L \tilde{P}_0^{(j)} F_0 \sum_{i=1}^{N_j} \psi_i^{(j)} \sum_{f=1}^{F_0} |\beta_f|^2 \left| \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,i}^{(j)} \right|^2 \quad (4.18)$$

Burada yine η . sınıfa ait iç parazit gücü hesabında $i=2$ 'den başlayıp N_η sayısal değerine kadar artmaktadır. Hücrelerarası parazit gücü ise

$$g_k^{(\eta)} = \text{var}\{n_k^{(\eta)}(l)\} = \sum_{j=1}^{(\log_2 SF)+1} Q_j L \sum_{k=1}^K \tilde{P}_k^{(j)} F_k \sum_{i=1}^{N_j} \psi_{k,i}^{(j)} \sum_{f=1}^{F_k} |\beta_{k,f}|^2 \left| \mathbf{a}^H(\theta_{k,f}) \mathbf{w}_{k,i}^{(j)} \right|^2 \quad (4.19)$$

verilir. Gürültü gücü ise

$$g_n^{(\eta)} = \text{var}\{n_T^{(\eta)}(l)\} = L F_0 \sigma_n^2 \quad (4.20)$$

ile verilir.

İstenilir sinyal ve parazit sinyallere ait varyansların türetilmesinde esas aldığımız yaklaşım, zayıflamaların birbirinden bağımsız olmalarıdır. 4.18 ve 4.19 no'lu denklemlerde verilen varyanslar, rasgele değişkenlerdir ve aktiflik faktörüne, ilgili

sinyalin ışıma örüntüsü vektörlerine ve kablosuz kanalda oluşan zayıflama ve yansıma etkilerine bağlıdır.

Bu eşitliklerden yola çıkarak η . sınıfa ait bir MS için bit enerjisinin, parazit ve gürültü toplamına oranı

$$\left(\frac{E_b}{N_0 + I_0} \right)_\eta = \frac{s^{(\eta)}(l)}{g^{(\eta)}(l) + g_0^{(\eta)}(l) + g_k^{(\eta)}(l) + g_n^{(\eta)}} \frac{\psi_{0,1}^{(\eta)} \tilde{P}_0^{(\eta)} L^2 R_0^{(\eta)}}{\psi_{0,1}^{(\eta)} \tilde{P}_0^{(\eta)} L (F_0 - 1) R_{0,i\check{c} parazit}^{(\eta)} + \sum_{j=1}^{(\log_2 SF)+1} \left[\mathcal{Q}_j L \tilde{P}_0^{(j)} F_0 \sum_{i=1}^{N_j} \psi_i^{(j)} \sum_{f=1}^{F_0} |\beta_f|^2 \left| \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,i}^{(j)} \right|^2 + \mathcal{Q}_j L \sum_{k=1}^K \tilde{P}_k^{(j)} F_k \sum_{i=1}^{N_j} \psi_{k,i}^{(j)} \sum_{f=1}^{F_k} |\beta_{k,f}|^2 \left| \mathbf{a}^H(\theta_{k,f}) \mathbf{w}_{k,i}^{(j)} \right|^2 \right] + L F_0 \sigma_n^2} \quad (4.21)$$

ψ , burada aktiflik faktörü olduğu için ya 0 (konuşma yok, dinleme modunda) ya da 1 (konuşma modunda, dinleme yok) değerini alacaktır.

Dolayısıyla konuşma modunda analize katkısı olacağından 1 alınır. Gerekli terimler kısaltıldığında

$$\left(\frac{E_b}{N_0 + I_0} \right)_\eta = \frac{L}{(F_0 - 1) \frac{R_{0,i\check{c} parazit}^{(\eta)}}{R_0^{(\eta)}} + G_0^{(\eta)} + G_k^{(\eta)} + \frac{F_0 \sigma_n^2}{\tilde{P}_0^{(\eta)} R_0^{(\eta)}}} \quad (4.22)$$

elde edilir. Burada

$$G_0^{(\eta)} = \frac{1}{\tilde{R}_0^{(\eta)}} g_0^{(\eta)}(l) = \frac{1}{\tilde{R}_0^{(\eta)}} \sum_{j=1}^{(\log_2 SF)+1} \mathcal{Q}_j \tilde{P}_0^{(j)} F_0 \sum_{i=1}^{N_j} \sum_{f=1}^{F_0} |\beta_f|^2 \left| \mathbf{a}^H(\theta_{0,f}) \mathbf{w}_{0,i}^{(j)} \right|^2 \quad (4.23)$$

$$G_k^{(\eta)} = \frac{1}{\tilde{R}_0^{(\eta)}} g_k^{(\eta)}(l) = \frac{1}{\tilde{R}_0^{(\eta)}} \sum_{j=1}^{(\log_2 SF)+1} \mathcal{Q}_j \sum_{k=1}^K \tilde{P}_k^{(j)} F_k \sum_{i=1}^{N_j} \sum_{f=1}^{F_k} |\beta_{k,f}|^2 |\mathbf{a}^H(\theta_{k,f}) \mathbf{w}_{k,i}^{(j)}| \quad (4.24)$$

ve gürültü terimi için de

$$\tilde{R}_0^{(\eta)} = \frac{F_0}{\tilde{P}_0^{(\eta)} R_0^{(\eta)}} \quad (4.25)$$

verilir.

Sistemin çıkış ihtimalinin olasılıksal olarak en az herhangi bir sınıfa ait kullanıcı için belirlenmiş bit hata oranının (bit error rate, BER) literatürde belirlenmiş eşik değerinden daha üst seviyede olması istenir.

$$\begin{aligned} P_{out} &= \Pr(BER_1 > p_{0,1} \text{ or } BER_2 > p_{0,2} \text{ or } \dots \text{ or } BER_j > p_{0,j}) \\ &= 1 - \Pr(BER_1 \leq p_{0,1} \text{ and } BER_2 \leq p_{0,2} \text{ and } \dots \text{ and } BER_j \leq p_{0,j}) \end{aligned} \quad (4.26)$$

Burada $p_{0,j}$. sınıf bir kullanıcı olduğunda, sistemin kabul edilebilir performansı için ihtiyaç duyulan BER eşik değeridir. Yani

$$P_{out} = 1 - \Pr \left\{ \begin{aligned} &(F_0 - 1) \frac{R_{0,iç \text{ parazit}}^{(1)}}{R_0^{(1)}} + G_0^{(1)} + G_k^{(1)} + \frac{\sigma_n^2}{\tilde{R}_0^{(1)}} \leq \frac{L}{S_1} \\ &\text{and} \\ &(F_0 - 1) \frac{R_{0,iç \text{ parazit}}^{(2)}}{R_0^{(2)}} + G_0^{(2)} + G_k^{(2)} + \frac{\sigma_n^2}{\tilde{R}_0^{(2)}} \leq \frac{L}{S_2} \\ &\text{and} \\ &\dots\dots \\ &\text{and} \\ &(F_0 - 1) \frac{R_{0,iç \text{ parazit}}^{(j)}}{R_0^{(j)}} + G_0^{(j)} + G_k^{(j)} + \frac{\sigma_n^2}{\tilde{R}_0^{(j)}} \leq \frac{L}{S_j} \end{aligned} \right. \quad (4.27)$$

olmalıdır.

4.4. Çoklu Yol Yayılım Etkisinin İhmali

4.4.1. Sinyal modeli

CDMA hücresel ağının ilk iki katmanı çoklu yol yayılım etkisinin ihmal edildiği durumda, ULA ve UCA topolojileri için ele alınmıştır.

BTS k 'dan MS'e doğru giden yol bileşeninin kanal parametreleri olarak θ_k geliş açısı, τ_k çoklu yol yayılım gecikmesi, β_k zayıflama parametreleri (büyük ölçekli ve küçük ölçekli zayıflamaları içeren) istenilir kullanıcının geliş açısına göre belirlenir. β_k zayıflama parametresi

$$\beta_k = \frac{\alpha_k^0}{(r_k^0)^2} \quad (4.28)$$

ile ifade edilir. Burada r_k^0 MS'in hücre merkezine olan uzaklığıdır. α_k^0 parametresi de MS'e gelen sinyalin maruz kaldığı zayıflama (fading) ve gölgelenme (shadowing) etkisidir.

MS tarafından alıcısı girişinde alınan tekli yol temel bant sinyal

$$\mathbf{x}_k^{(j)}(t) = \sum_{i=1}^{N_j} \sqrt{P_{k,i}^{(j)}} \psi_{k,i}^{(j)} \beta_k s_{k,i}^{(j)}(t - \tau_k) \mathbf{a}^H(\theta_k) \mathbf{w}_{k,i}^{(j)} \quad (4.29)$$

ile ifade edilir.

4.4.2. Parazit analizi ve sistem çıkış denklemi

τ_γ yayılım gecikmesi, çoklu yol etkisinin ihmal edildiği durumda bir değere sahiptir. Verilen bir zaman aralığında η . sınıf bir MS için kodu l . sembol periyodu boyunca

$\tilde{c}_{d,h}^{(\eta)}(l;t-\tau_\gamma)$ olan d . MS alıcısının h . kod kanalının γ . yayılım gecikmesi için korelatör çıkışı

$$z_\gamma^{(\eta)}(l) = \frac{1}{T_c} \int_{(l-1)T_w + \tau_\gamma}^{lT_w + \tau_\gamma} x(t) \tilde{c}_{d,h}^{(\eta)*}(l;t-\tau_\gamma) dt = s^{(\eta)}(l) + n_I^{(\eta)}(l) + n_T^{(\eta)}(l) \quad (4.30)$$

ile verilir. Burada $s^{(\eta)}(l)$ istenilir sinyal bileşeni, $n_I^{(\eta)}(l)$ MAI sinyalleri ve $n_T^{(\eta)}(l)$ ise gürültü terimi olarak ifade edilir. Denklem 4.30'da verilen integral sonunda

$$I_{k,i,q,1,\gamma}^{(j,\eta)} = \frac{1}{T_c} \int_{(l-1)T_w + \tau_\gamma}^{lT_w + \tau_\gamma} b_{k,i,q}^{(j)}(l) \tilde{c}_{k,i,q}^{(j)}(l;t-\tau_\gamma) \tilde{c}_{1,1}^{(\eta)*}(l;t-\tau_\gamma) dt \quad (4.31)$$

elde edilir. Ortamda çoklu yol etkisi göz önünde bulundurulmadığından ve kod gecikmesi tüm kodlar için aynı zaman değerinde olduğundan farklı kodlar arasındaki diklik özelliği bozulmamış olur. Dolayısıyla $q=1$ ve i =istenilir kullanıcı olma durumunda $I_{k,i,q,1,\gamma}^{(j,\eta)}$ değeri SF, diğer durumlarda ise $I_{k,i,q,1,\gamma}^{(j,\eta)}$ değeri sıfır olur.

Burada istenilir sinyal bileşeni, $q=1$ ve i =istenilir kullanıcı olma durumunda

$$s^{(\eta)}(l) = \psi_1^{(\eta)} b_{1,1}^{(\eta)}(l) L \sqrt{\tilde{P}_0^{(\eta)}} \mathbf{a}^H(\theta_{0,1}) \mathbf{w}_{0,1}^{(\eta)} \quad (4.32)$$

olarak ifade edilir. Farklı OVFSF kanallama kodlarının birbirine dik olmasından dolayı, MAI bileşeni olarak sadece farklı katmanlarda, istenilir kullanıcı ile aynı koda sahip kullanıcıların etkisi göz önünde bulundurulur.

$$n_I^{(\eta)}(l) = \sum_{j=1}^{(\log_2 SF)+1} \sum_{k=1}^K \sum_{\substack{i=1 \\ i \neq \text{desired}}}^{N_j} \psi_{k,i}^{(j)} \sqrt{\tilde{P}_k^{(j)}} \mathbf{a}^H(\theta_k) \mathbf{w}_{k,i}^{(j)} \sum_{q=1}^{Q_j} I_{k,i,q,1,\gamma}^{(j,\eta)} \quad (4.33)$$

ve gürültü bileşeni olarak ta

$$n_T^{(\eta)}(l) = \frac{1}{T_c} \int_{(l-1)T_w + \tau_\gamma}^{lT_w + \tau_\gamma} n(t) c_{1,1}^{(\eta)*}(l;t-\tau_\gamma) dt \quad (4.34)$$

ifadesi alınır.

İstenilir MS kullanıcısının sahip olduğu bit enerjisinin, parazit ve gürültü toplamına oranını hesaplamak için istenilir sinyal ve parazit sinyal bileşenlerine ait güç ifadesinin varyanslarının hesaplanması gereklidir.

$$\text{var}\{s^{(\eta)}(l)\} = \tilde{P}_0^{(\eta)} L^2 |\mathbf{a}^H(\theta_{0,1}) \mathbf{w}_{0,1}^{(\eta)}|^2 \quad (4.35)$$

$$\text{var}\{n_l^{(\eta)}(l)\} = \sum_{j=1}^{(\log_2 SF)+1} Q_j L^2 \sum_{k=1}^K \tilde{P}_k^{(j)} \sum_{\substack{i=1 \\ i \neq \text{desired}}}^{N_j} |\mathbf{a}^H(\theta_{k,1}) \mathbf{w}_{k,i}^{(j)}|^2 \quad (4.36)$$

BTS tarafından tüm MS'lere aynı çıkış gücü değeri ile sinyal iletimi yapıldığı varsayıldığında η . sınıfa ait bir MS için bit enerjisinin parazit ve gürültü toplamına oranı

$$\left(\frac{E_b}{N_0 + I_0} \right)_\eta = \frac{|\mathbf{a}^H(\theta_{0,1}) \mathbf{w}_{0,1}^{(\eta)}|^2}{\sum_{j=1}^{(\log_2 SF)+1} Q_j \sum_{\substack{i=1 \\ i \neq \text{desired}}}^{N_j} |\mathbf{a}^H(\theta_{k,1}) \mathbf{w}_{k,i}^{(j)}|^2} \quad (4.37)$$

ile verilir.

4.5. Aşağı Bağlantı (DL) Işıma Örüntüsü Algoritmaları

DL sırasında, UL boyunca elde edilen sinyal yayılım bilgilerine dayanarak, her kullanıcı için anten dizisinde ışıma örüntüsü ağırlık vektörünün tahminine ihtiyaç vardır. FDD esnasında, UL ve DL frekansları farklı olduğundan sinyal zayıflamaları da farklı olacaktır ve bu nedenle DL iletimi için UL boyunca elde edilen ışıma örüntüsü ağırlıklarının kullanımı doğru değildir. DL ışıma örüntüsünü belirlemek için kullanılabilecek algoritmalarından biri geri besleme metodudur (Rappaport ve Liberty, 1999). Burada hücre içindeki tüm MS'lere periyodik aralıklarla pilot sinyalleri ya da önceden belirlenmiş tonda sinyaller gönderilir. Bir diğer metotta ise UL boyunca elde edilen ölçümler, DL sinyalini ve parazit kovaryans matrisini elde etmek için kullanılır. Bu sinyaller, daha sonra DL ağırlık vektörlerini tahmin etmek için kullanılacak olan temel bilgilerdir (Hochwald ve Marzetta, 2001). Bu metotla

elde edilen çalışmalar sonucunda, FDD frekans farkının az olduğu durumlarda UL ve DL için benzer zayıflama istatistiklerinin (kovaryans matrisi) elde edildiği görülmüştür. Bu nedenle uygulamalarda, UL kovaryans matrisinin frekans dönüşümü kullanılarak DL kovaryans matrisi tahmin edilebilir (Liang ve Chin, 2001). Ancak kovaryans matris tabanlı ışıma örüntüsü algoritmaları, dönüşümden kaynaklanan tahmin hatalarından çok fazla etkilenir. FDD sistemler için uygulanan bir başka ışıma örüntüsü metodunda, çoklu yolların geliş (direction of arrival, DOA) açılarının kullanımı söz konusudur. Deneysel olarak gösterilmiştir ki MS'in küçük hareketleri esnasında (örneğin kısmi hareketin 2λ olduğu durumlar gibi) sinyalin yayılım geometrisi, çok fazla değişiklik göstermez ve bu nedenle UL ve DL için çoklu yollara ait DOA açıları hemen hemen aynı alınır (Kavak ve diğ., 2001). SIR tabanlı AAA-DCA algoritmasında, DL ağırlık vektörlerini tahmin etmek için üç farklı DOA tabanlı DL ışıma örüntüsü metodu kullanılmıştır. Bunlar; ortalama DOA (dominant DOA), baskın DOA (mean DOA) ve ağırlıklı ortalama DOA (weighted-mean DOA) metotlarıdır. k . hücrede bulunan her bir sınıfa ait kullanıcılar için DL ağırlık vektörü, doğrultu vektörü

$$\mathbf{w}_{k,i}^{(j)} = \mathbf{a}(\hat{\theta}_{k,f}) \quad (4.38)$$

ile ifade edilir. Burada $\hat{\theta}_{k,f}$ k . hücredeki f . çoklu yola karşılık gelen DAO açısıdır ve aşağıdaki gibi elde edilir: Ortalama DOA metodunda, çoklu yollara ait minimum ve maksimum DOA açılarının ortalaması alınır.

$$\hat{\theta}_{k,f} = \frac{1}{2} \{ \min(\theta_{k,f}) + \max(\theta_{k,f}) \} \quad (4.39)$$

Baskın DOA metodunda en güçlü sinyalin elde edildiği çoklu yola karşılık gelen DOA açısı alınır.

$$\hat{\theta}_{k,f} = \max_{\theta_{k,f}} (\beta_{k,f}) \quad (4.40)$$

Ağırlıklı ortalama DOA metodunda ise her bir çoklu yol DOA açıları, kendilerine karşılık gelen kısmi zayıflama katsayıları ile ağırlıklandırılarak elde edilir.

$$\hat{\theta}_{k,f} = \frac{\theta_{k,1}\beta_{k,1} + \dots + \theta_{k,f}\beta_{k,f}}{\beta_{k,1} + \dots + \beta_{k,f}} \quad (4.41)$$

FDD frekans farkı, mutlak taşıyıcı frekansa göre çok daha küçük bir değerde olduğundan, her ne kadar başlangıçta UL ve DL zayıflama katsayıları, her ne kadar birbirinden bağımsız olsalar da aynı davranışı ve istatistiği gösterirler (örneğin kovaryans matris) (Kavak ve diğ., 2001). Diğer bir deyişle karmaşık zayıflamaya neden olan dağıtıcı nesnelerin yansıma katsayısı frekans çevriminden ihmal edilecek kadar az etkilendiğinden çoklu yolların kuvveti (karmaşık zayıflamaların genliği), UL'den DL'ye iletim boyunca korunmaktadır.

4.6. OVFS Kod Tahsis Veritabanı

OVFS kod tahsis veritabanı, kod ağacındaki tüm aktif kullanıcılara ait bilgilerin saklandığı veritabanıdır (resource manager database, RMD). Kullanıcıya kod tahsisi yapıldığı zaman, bu kullanıcıya ait tüm bilgiler veritabanında yerini alır ve konuşma sonlandırılana kadar burada tutulur. SIR tabanlı dinamik OVFS kod tahsis yönteminin en önemli kısmıdır. Burada, istenilir kullanıcıya ait DL yönünde DOA açısı ve çokluyol açı bilgileri, DL ağırlık vektörü, her bir DL DOA açısına ait (istenilir+çoklu yol) doğrultu vektörü, veri transfer hızı, OVFS kod ağacında işgal ettiği konum bilgisi ve birim kod olarak her bir koda ait indis bilgileri saklanır. Aşağıdaki tablo RMD ile ilgili gösterimi vermektedir.

Tablo 4.1: RMD Gösterimi

Kullanıcı Sayısı	Konuşma Zamanı	DL-DOA	DL-w	DL-a	Hız	OVFS	Birim Kod
1	...	...	...	...	...	...	...
2	...	...	...	...	...	...	...

Tablodaki konuşma zamanı, yani çağrının hatta kalma süresi, 0.25 birim zaman ortalama değerli Exponentially dağılım ile modellenmektedir. Burada verilen terimler, saniye cinsinden ifade edilmektedir. Süresi dolan aktif kullanıcı konuşmasını tamamlamış sayılır ve RMD'den kaydı silinir.

DL-DOA sütunu, istenilir kullanıcıya ait tüm DL-DOA açılarını saklar. Bu açılar kullanıcının istenilir açısı ve ona bağlı çoklu yol açılarıdır. Çoklu yol açılarının sayıları, rasgele değişkenli olarak modellenmekle birlikte bu çalışmada daha önceden belirlenen bir sabit değere ayarlanmıştır. Çoklu yol etkisi göz önünde bulundurulduğu durumda, her bir kullanıcı için, 1 adet istenilir sinyal DL-DOA açısı ve 3 adet de çoklu yol DL-DOA açısı daha önceden belirlenmiştir. Çoklu yol etkisi ihmal edildiğinde ise DL-DOA açısı 1 adet istenilir sinyal olarak tanımlanır. ULA anten düzeneği içinde kullanıcılara ait tüm DL-DOA açıları $[0^\circ-120^\circ]$ açı aralığında, UCA anten düzeneği içinde ise $[0^\circ-360^\circ]$ açı aralığındadır.

DL-w sütununda, DL ışıma örüntüsü algoritmalarından faydalanılır. UL boyunca kullanıcıya ait UL-DOA açıları, ortalama DOA, baskın DOA ve ağırlıklı ortalama DOA algoritmaları ile değerlendirilerek DL yönünde ağırlık vektörü elde etmek için kullanılır. Bu ağırlık vektörünün boyutu $[m \times 1]$ 'dir. m anten sayısını gösterir.

DL-a sütunu, her bir kullanıcıya ait DL DOA açısının (istenilir+çoklu yol) doğrultu vektörü bilgisini içeren bir matristir. Bu matrisin ilk sütunu, kullanıcının direkt geliş açısı (istenilir DOA) bilgisini içeren doğrultu vektörüdür. Burada yer alan bilgilerin boyutu $[m \times F_0 + 1]$ 'dir. F_0 çoklu yolların sayısını ifade eder. Çoklu yol etkisi ihmal edildiğinde boyut $[m \times 1]$ 'lik bir vektör halini alır.

Hız sütunu, DL yönünde veri transfer isteğinde bulunan kullanıcının, bu transferi hangi hızla yapacağına dair hız bilgisi içerir. OVFSF kod ağacında, kullanıcının hangi katmanda yer alacağına dair bilgi bu sütunda verilir. Veri transfer hızı arttıkça OVFSF kod ağacında yukarılara doğru çıkılırken, o kullanıcıya tahsis edilen OVFSF kanallama kodunun boyutu düşer. Aynı şekilde veri transfer hızı azaldıkça, OVFSF kod ağacında aşağılara doğru inilirken, o kullanıcıya tahsis edilen OVFSF kanallama kodunun boyutu artar.

OVFSF sütununda kod tahsisi yapılmış bir kullanıcının OVFSF kod ağacında hangi katmanda ve hangi dalda olduğuna dair bilgi verilmektedir. Bu sütunda yer alan [6 1] bilgisi, SF=256 olduğu durumda kullanıcının OVFSF kod ağacında, yukarıdan aşağıya 6. katmanda ve soldan sağa 1. dalda yer aldığını işaret etmektedir. Aynı şekilde bu

bilgilerden faydalanarak DL yönünde talep edilen veri transfer hızının $2^{[(\log_2 SF)+1]-6} = 8R$ olduğunu ve buradan da işgal ettiği kodların numaralarının

$$\underbrace{[1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]}_{8 \text{ adet}}$$

ile belirlendiği görülmektedir

Birim kod sütununda ise yukarıda belirtmiş olduğumuz 8 adet dal numarası saklanır. Burada saklanan bilgi, veri transfer hızının büyüklüğüne göre belirlenmiş bir satır vektörüdür.

5. KOD SINIRLI VE SIR SINIRLI DİNAMİK OVFS KOD TAHSİS ALGORİTMALARININ PERFORMANSI

5.1. Giriş

Çalışmanın temel amacı, OVFS kanallama kodu talebinde bulunan kullanıcıya ihtiyacı olan kodun sağlanmasıdır. Bölüm 3 ve Bölüm 4’te tanımlanan algoritmalar, genel olarak sözü edilen amaca hizmet etseler de yapısal olarak farklılıklar içerirler. Özetlemek gerekirse;

- OVFS kod ağacında, veri transferi talebinde bulunan kullanıcıya tahsis edilecek kadar boş kanallama kodunun olmadığı, ancak ağaçtaki boş kodların kapasitesinin ihtiyaç duyulan kanallama kodundan büyük ya da eşit olduğu durumda, kod sınırlı dinamik OVFS kod tahsis algoritması, kod ağacında yer değiştirme işlemi ile uygun kanallama kodu tahsis etmeye çalışır. Bu amaçla, GA tabanlı SGA ve D&DGA kod tahsis algoritmaları, SA ve ASAGA kod tahsis algoritması incelenmiştir.
- OVFS kod ağacında, veri transferi talebinde bulunan kullanıcıya tahsis edilecek kadar boş kanallama kodu yok ve/veya ağaçtaki boş kodların kapasitesi ihtiyaç duyulan kanallama kodundan küçük ise, SIR sınırlı dinamik OVFS kod tahsis algoritması, kod ağacında SIR analizi uygun kanallama kodu tahsis etmeye çalışır. Bu nedenle AAA tabanlı kod tahsis algoritması, çoklu yol etkisi göz önünde bulundurularak ve ihmal edilerek incelenmiştir. Burada sistem için gerek koşul, aynı kodu kullanacak olan kullanıcıların birbirine olan girişim etkisinin literatürce belirlenen eşik seviyesinden küçük olmasıdır. Bu durumda aynı OVFS kanallama kodu, birden fazla kullanıcı tarafından ortak olarak kullanılabilir.

5.2. Simülasyon Parametreleri ve Metodoloji

Bu tez çalışmasında, performans ölçütü olarak çeşitli karşılaştırma ve iyileştirme parametreleri ele alınmıştır. Sistem performansı ve maliyet analizi başlıkları altında, Bölüm 3 ve Bölüm 4'te bahsedilen algoritmalar karşılaştırmalı olarak incelenmiştir. Bu bölümlerde, belirtilen performans ölçütü cinsinden hangi algoritmanın daha iyi performans sergilediği verilmiştir. İyileştirme parametreleri ise algoritmik parametrelerin etkisi başlığı altında ele alınmıştır. Bu bölümde, incelenen algoritmanın performansının artırması ve muhtemel değişkenlerin genel sistem üzerine etkisi incelenmiştir.

Sistem performansının ele alındığı bölümde ilk olarak bloklanma oranı incelenmiştir. Bloklanma oranı, gelen toplam veri transfer isteklerinden kaç tanesine uygun OVSF kanallama kodunun tahsis edilip, kaç tanesinin reddedildiği bilgisini verir. Daha sonra spektral verimlilik ele alınmıştır. Spektral verimlilik, ihtiyaç duyulan toplam veri transfer hızı isteğinin ne kadarının tahsis edilip, ne kadarının tahsis edilemediği bilgisini verir. Bloklanma oranı kişi sayısı üzerinden ifade edilirken, spektral verimlilik veri transfer hızı büyüklüğü cinsinden ifade edilmektedir. Çağrı model etkisinde ise, OVSF kod ağacının daha önceden belirlenmiş olasılıklara sahip farklı çağrı modelleri altında tepkisi incelenmiştir.

Algoritmik parametreler başlığı altında ise her algoritmaya özgü çeşitli parametreler ele alınmıştır. Kod sınırlı algoritmalar için farklı seçim, çaprazlama ve mutasyon teknikleri, farklı çaprazlama ve mutasyon oranları, soğutma çizelgesindeki her bir parametre değişiminin sistem performansı üzerindeki etkileri ele alınmıştır. SIR sınırlı algoritmalar için ise farklı AAA topolojileri, farklı anten sayıları, anten elemanları arasındaki mesafe, açısal yayılımdaki değişim ve farklı DL ışıma örüntüsü algoritmalarının sisteme etkileri incelenmiştir.

Maliyet analizi bölümünde ise, kod sınırlı yapıdaki tekrarlamalı algoritmaların her bir iterasyondaki toplam toplama ve çarpma işlemi sayıları hesaplanmıştır.

5.2.1. Simülasyon parametreleri

Simülasyonlarda kullanılan parametreler; OVSF ile ilgili, kod sınırlı algoritmalar ile ilgili ve SIR sınırlı algoritmalar ile ilgili parametreler olarak sınıflandırılmıştır.

İlk olarak OVSF ile ilgili parametrelerde; çağrı gelme işleminin ortalama geliş oranı λ , 4 ile 64 arama/birim zaman olarak değişen Poisson dağılımı ile çağrının hatta kalma süresi, 0.25 birim zaman ortalama değerli Exponentially dağılım ile modellenmiştir. Maksimum yayılım faktörü (SF) 256 olarak alınmıştır. Çağrı talebinde bulunan mobil kullanıcının veri transfer hızı, R ile $SF \times R$ arasında rasgele olarak değişmektedir.

Kod sınırlı algoritmalar ile ilgili parametrelerde; GA için kromozom gösterimi tamsayıdır. Popülasyon büyüklüğü mevcut trafik yoğunluğuna bağlı olarak değişmektedir. Ortamdaki aktif kullanıcıların sayısı ve onların veri transfer hızları popülasyon büyüklüğünü vermektedir. Ayrıca rulet tekerleği, tek noktalı çaprazlama ve yer değiştirme teknikleri sırasıyla seçim, çaprazlama ve mutasyon operatörleri olarak kullanılmıştır. Çaprazlama (p_c) ve mutasyon (p_m) oranları, sırasıyla 0.8 ve 0.2 olarak ele alınmıştır. SA için başlangıç sıcaklığı olarak 0.9 seçilmiştir, son sıcaklık değeri ise 0.1'dir. Yerdeğiştirme ve ötelemeli olmak üzere iki komşuluk hareketi uygulanmıştır. Soğutma çizelgeleri olarak da orantısal azalimli (proportional decrease) ve Lundy & Mees soğutma çizelgeleri kullanılmıştır. Dış döngü için belirlenen maksimum iterasyon sayısı 1000'dir. İç döngü de beş defa tekrarlanacaktır. ASAGA için tanımlanan parametreler GA ve SA için tanımlananlar ile aynıdır.

SIR sınırlı algoritmalar ile ilgili parametrelerde ise iki katmanlı CDMA hücresel ağ yapısı kullanılmıştır. Kullanıcılar, tüm hücre için düzgün dağılımlı olarak konumlandırılmıştır. En kötü senaryo için kullanıcı hücreler arası birleşim noktasında alınmaktadır. Bir hücre içinde tüm sınıflardan belli sayılarda kullanıcılar bulunmaktadır. Bu kullanıcılar, ULA anten dizisi için $[30^\circ-150^\circ]$ açı aralığı içinde, UCA anten dizisi için $[0^\circ-360^\circ]$ açı aralığı içinde dağılmışlardır. Gürültü seviyesi istenilir kullanıcıya göre 10 dB daha aşağıda modellenmiştir. Her kullanıcı için hücre

içinde daha önceden belirlenmiş sayıda çoklu yol açısı vardır. Bu açılar, $\Delta\theta$ açısal yayılımı içinde sınırlandırılmışlardır. Aktivite oranı, 0.1 olarak alınmıştır.

5.2.2. Simülasyon metodolojisi

Tüm kod sınırlı ve SIR sınırlı algoritmalarda; ortalama çağrı gelme oranları, çağrının hatta kalma süresi, veri transfer hızı dağılımı ve maksimum yayılım faktörü değerleri her simülasyonda belirtilen özelliklere göre giriş parametresi olarak verilmiştir. OVFS kod tahsis yapılan ve yapılamayan (bloklanmış) mobil kullanıcıların sayısı, onların veri transfer hızları, yer değiştirme sonucu kod tahsis yapılan kullanıcıların sayısı ve veri transfer hızları matris formunda saklanmaktadır. Simülasyonlar, aynı giriş parametreleri için 10 defa tekrarlanmıştır. Elde edilen sonuçlar, bu 10 ayrı simülasyonun ortalaması alınarak elde edilmiştir.

5.2.2.1. Kod sınırlı algoritmalar için simülasyon metodolojisi

Kod sınırlı algoritmalar için simülasyonlar, 10000 kullanıcının kod tahsis talebinde bulunması ile sınırlandırılmıştır. Her bir simülasyon, aynı giriş parametreleri için 10 defa ayrı ayrı gerçekleştirilmiştir. Burada verilen sonuçlar, bu simülasyonlardan elde edilen sonuçların ortalama değeridir. Her bir simülasyon için aşağıdaki adımlar uygulanmıştır.

- Çağrı talebi geldiğinde ilk olarak, sistem kapasitesi kontrol edilir. Eğer kapasite yeterli ise, ihtiyaç duyulan veri transfer hızını karşılayabilecek OVFS kanallama kodu, ilgili katmanda kullanılmayan kodlar arasında araştırılır. Uygun kod varsa tahsis edilir.
- OVFS kod ağacının yeterli kapasiteye sahip ancak ihtiyaç duyulan veri transfer hızına uygun OVFS kanallama kodu olmadığı durumda çağrı gerçekleştirilemiyorsa, kod sınırlı algoritmalar devreye girer. OVFS kod ağacının kapasitesi yeterli değilse de çağrı bloklanır.

- Kod sınırlı algoritmalar, veri transfer hızı talebinde bulunan kullanıcıya ihtiyacı olan OVFS kanallama kodunu en düşük maliyet ile tahsis etmeye çalışır.

5.2.2.2. SIR sınırlı algoritmalar için simülasyon metodolojisi

SIR sınırlı algoritmalarda, hücreiçi parazit ve hücrelerarası parazite neden olan kullanıcıların hücredeki dağılımlarından dolayı analizini sağlıklı yapabilmek için, simülasyonlar (farklı anten dizi topolojileri, anten dizisi elemanlar arası mesafe, açısal yayılım, ışıma örüntüsü algoritmaları) Monte Carlo simülasyon modeli ile iyileştirilmiştir. Sistemde farklı veri transfer hızlarına ait rasgele sayıda kullanıcı bulunmaktadır. Gürültü seviyesi, her sınıftaki istenilir sinyal bileşenlerinden 10 dB daha düşüktür. İhtiyaç duyulan $[E_b/(N_0+I_0)]$ seviyeleri, literatürce belirlenmiş olan standartlara uygun olarak belirlenmiştir. Anten sayısı 1 olduğunda ($m=1$) sistem adaptiflikten çıkar ve yönsüz anten sistemi halini alır. Simülasyonlar 10000 kullanıcının kod tahsis talebinde bulunması ile sınırlandırılmıştır. Her bir simülasyon 10 defa ayrı ayrı gerçekleştirilmiş olup, elde edilen sonuçlar bu değerlerin ortalamasıdır. Her bir simülasyon için aşağıdaki adımlar uygulanmıştır.

- Çağrı talebi geldiğinde ilk olarak, sistem kapasitesi kontrol edilir. Eğer kapasite yeterli ise, ihtiyaç duyulan veri transfer hızını karşılayabilecek OVFS kanallama kodu, ilgili katmanda kullanılmayan kodlar arasında araştırılır. Uygun kod varsa tahsis edilir.
- OVFS kod ağacının yeterli kapasiteye sahip olmamasından dolayı ya da ihtiyaç duyulan veri transfer hızına uygun OVFS kanallama kodunun bulunmamasından dolayı çağrı gerçekleştirilemiyorsa, algoritmanın DL ışıma örüntüsü vektör kod arama kısmı icra edilir. Bu kısımda ihtiyaç duyulan OVFS kanallama kodunu ortak kullanmak için yeni kullanıcının kanal vektörü ($\mathbf{a}_{yeni}$) ile minimum ilintiye sahip DL ışıma örüntüsünü ($\mathbf{b}_k$) kullanan aktif kullanıcıların, OVFS kodu araştırılır.
- En uygun ışıma örüntüsüne ait kullanıcıyı belirlemek için;
- İstenilir kullanıcının uzaysal konumu (DOA açısı) düzgün dağılımlı rasgele değişkenle modellenmiş istenilir kullanıcının uzaysal konumu (DOA açısı)

belirlenir. Parazit analizi için en kötü senaryo durumunda kullanıcıların hücrelerin birleşim noktasında yani bir hücrenin kenarında bulunduğu varsayılır. İstenilir kullanıcıya ait DOA açıları, ULA anten topolojisi için $[30^\circ-150^\circ]$ açı aralığında, UCA anten topolojisi için $[0^\circ-360^\circ]$ açı aralığında belirlenir.

- İstenilir kullanıcıya ait DOA açısı belirlendikten sonra, dış hücrelerde (iki katmanlı CDMA hücresel ağı kullanılmış ve toplam 18 tane dış hücre göz önünde bulundurulmuştur) istenilir kullanıcı ile aynı frekans aralığını kullanan parazit kullanıcılardan kaynaklanan etkiyi belirlemek için onlara ait DL-DOA açıları (istenilir+çoklu yol) belirlenir. Dış hücrelerin DL-DOA açıları da düzgün dağılımlı rasgele değişken modeli ile belirlenmiştir. Bu açıların nasıl elde edileceği Ek-B’de ele alınmıştır.
- İstenilir kullanıcıya ait DOA bilgisi ile DL yönünde belirtilen açısal yayılım çerçevesinde ($\Delta\theta$) çoklu yol açıları verilmiştir. Simülasyonlarda, tüm sınıflara ait kullanıcıların çoklu yolların sayısı $F_o=F_k=3$ olarak seçilmiştir. Çoklu yol etkisinin göz ardı edildiği simülasyonlarda da $F_o=F_k=1$ olarak belirlenmiştir.
- Baz istasyonundan istenilir kullanıcıya doğru, iki katmanlı 19 hücreli CDMA ağ yapısında, kullanıcıya ait DL zayıflama katsayıları ve çoklu yol DL-DOA açıları kullanılarak kanal vektörü belirlenir. Bundan dolayı zayıflama katsayısını belirlemek için 57 adet ($K+1 \times F=19 \times 3$) rasgele değişken belirlenir. Bu değişkenler, standart sapması 10dB olan Rayleigh rasgele değişken modeli ile belirlenmiştir. Zayıflama katsayıları çoklu yol etkisinin göz ardı edildiği durumda 19 adet rasgele değişkenden oluşmaktadır.
- Bölüm 4’te verilen DL ışıma örüntüsü algoritmaları kullanılarak her sınıftaki istenilir kullanıcılar için DL iletim ağırlık vektörü belirlenmiştir. DL zayıflamalar hakkında ön bilgi sahibi olmadığımız için UL zayıflama katsayıları kullanılarak bu değerler bağımsız olarak üretilmiştir. Daha önce de ifade edildiği gibi UL ve DL zayıflamaları birbiri ile ilintisiz olduğundan çoklu yolların şiddetleri cinsinden aynı davranış korunur.
- İstenilir kullanıcı, hücredeki kullanıcı yoğunluğu ve dış hücrelerden gelen kullanıcılar dikkate alınarak, istenilir kullanıcının çoklu yollarından kaynaklanan iç parazit, hücreiçi parazit ve hücrelerarası parazit değerleri ilgili denklemler kullanılarak sayısal değerler halinde belirlenir.

- Son olarak sistem çıkış gücü, çoklu yol etkisinin incelendiği durum için 4.27 no'lu denkleme göre, çoklu yol etkisinin göz ardı edildiği durum için 4.37 no'lu denkleme göre belirlenmiştir.

Sistem çıkış gücü, literatür tarafından belirlenmiş eşik seviyesinin üstünde ise aynı kod birden fazla kullanıcıya ortak kullanım için tahsis edilir. Aksi halde diğer kullanıcının kodunun ortak kullanımı için bu analiz tekrar yapılır. Eğer hiçbir kullanıcının kodunu ortak kullanacak şekilde çıkış gücü sağlanamıyorsa bu durumda veri transfer isteğinde bulunan kullanıcının kod tahsis talebi reddedilir ve çağrı blok olur.

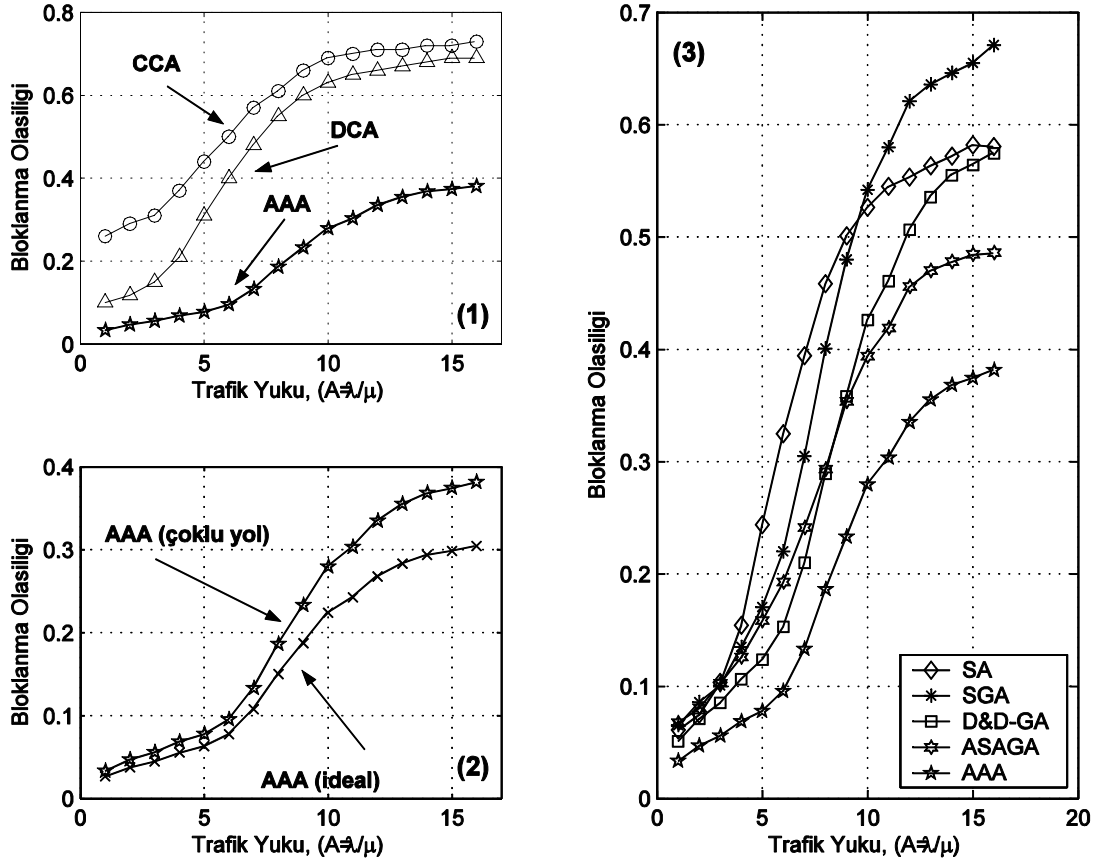
5.3. Sistem Performansı

5.3.1. Bloklanma olasılığı

Sistem performansını incelerken, ilk olarak kod bloklanma olasılığı (BP) ele alınmıştır. Burada veri transfer isteğinde bulunan kullanıcıların (N_{talep}) OVSF kanallama kodu tahsis edilemeyen kullanıcılara (N_{blok}) oranı verilir.

$$pr(BP) = \frac{N_{blok}}{N_{talep}} \quad (5.1)$$

Şekil 5.1, farklı trafik yükleri için bloklanma olasılığını ele almıştır. Şekillerden açıkça görülmektedir ki trafik yükündeki artış, beraberinde bloklanma olasılığındaki artışı da getirmiştir. Bu da sistem performansını olumsuz yönde etkilemektedir.



Şekil 5.1: Farklı trafik yüklerinde bloklanma olasılıklarının karşılaştırılması.

Şekil 5.1 (1)'de AAA literatürde en çok kullanılan CCA ve DCA ile karşılaştırılmıştır. Şekilden performans artışı açıkça görülmektedir. Örneğin trafik yükü 10 olduğunda, AAA için bloklanma olasılığı % 27.9, DCA için bloklanma olasılığı % 63 ve CCA için bloklanma olasılığı % 69 civarındadır. Bu değerler, temel alındığında AAA, DCA'ya göre % 125.8'lik performans artışı sergilerken, bu değer CCA'ya göre % 147.3 civarındadır. Aynı parametreler altında, çoklu yol etkisinin ihmal edildiği AAA ile F_0 adet çoklu yola sahip AAA arasındaki performans karşılaştırması, Şekil 5.1 (2)'de yapılmıştır. Çoklu yolların ortamda oluşturduğu etki, sistemde % 20 performans kötüleşmesi olarak kendini göstermiştir. Trafik yükü 10 iken ideal AAA için bloklanma olasılığı % 22.3 civarındadır. Şekil 5.1 (3)'te ise SIR sınırlı AAA-DCA algoritması, kod sınırlı SGA, D&D-GA, SA ve ASAGA algoritmalarının bloklanma olasılıkları cinsinden karşılaştırmaları verilmiştir. Burada, AAA-DCA algoritmasının sistemde daha az bloklanmaya yol açtığı söylenebilir. Sayısal olarak karşılaştırdığımızda trafik yükü 10 iken AAA-DCA

algoritması için bloklanma olasılığı % 27,9, ASAGA için % 39.4, D&D-GA için % 42.6, SA için % 52.6 ve SGA için % 54.2 civarındadır. Bu sonuçlar ışığında, AAA-DCA algoritmasının ASAGA algoritmasına göre % 41.2'lik, D&D-GA algoritmasına göre de % 52.6'lık bir performans artışı sergilediği görülmüştür. Elde edilen bu karşılaştırmalı değerlerden sonra, SIR sınırlı yapının kod sınırlı yapıya göre bloklanma olasılığı cinsinden daha iyi sistem performansı sergilediği söylenebilir.

Şekil 5.1'i oluşturan parametrik değişkenler şöyledir: $SF:256$, komşuluk:ötelemeli, soğutma çizelgesi:Lundy & Mees, $p_c:0.8$, $p_m:0.2$, seçim:rulet tekerleği, çaprazlama:tek noktalı, mutasyon:ötelemeli, $m:3$ ULA, $\Delta\theta:20^\circ$, $F_0:0$ ve 3, $\Delta:0.5\lambda$.

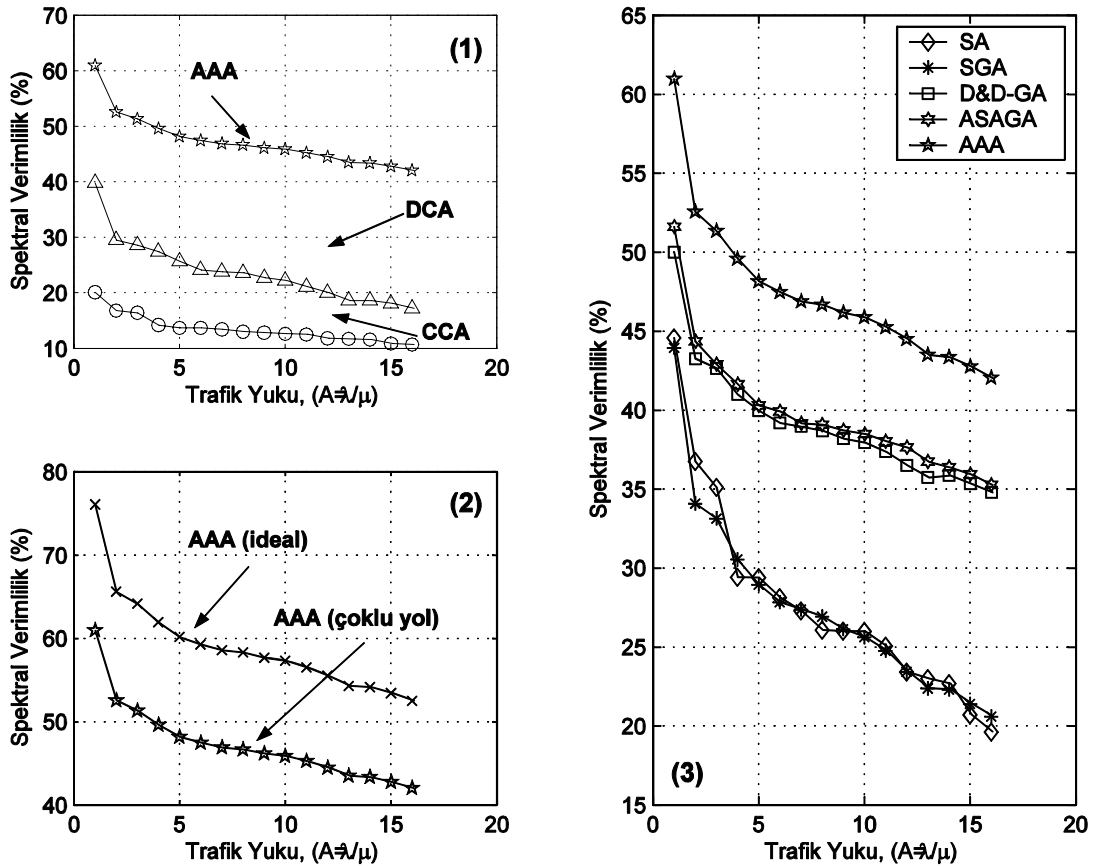
5.3.2. Spektral verimlilik

Sistem performansının karşılaştırılması amacı ile kullanılan bir diğer yöntem de sistemin spektral verimliliğinin (SE) incelenmesidir. Spektral verimlilik, ihtiyaç duyulan toplam veri transfer hızı (R_{talep}) isteğinin ne kadarının tahsis edilip (R_{tahsis}), ne kadarının tahsis edilemediğini vermektedir. Bloklanma oranı kişi sayısı üzerinden ifade edilirken spektral verimlilik veri transfer hızı büyüklüğü cinsinden ifade edilmektedir.

$$SE = \frac{R_{tahsis}}{R_{talep}} \times 100 \quad (5.2)$$

Şekil 5.2, tüm algoritmalar için farklı trafik yüklerinde mevcut spektrumun ne derece verimli kullanıldığını göstermektedir. Genel olarak şekillerden AAA kullanımının mevcut spektrumun verimli bir şekilde sarf edilmesine ne derece olumlu etkisi olduğu açıkça görülmektedir. Burada trafik yükündeki artış ve spektral verimlilik ters orantılı olarak değişmektedir. Örneğin Şekil 5.2 (1)'de AAA, literatürde en çok kullanılan CCA ve DCA ile karşılaştırılmıştır. Spektrum cinsinden performans artışı açıkça görülmektedir. Trafik yükü 10 olduğunda AAA mevcut spektrumun % 45.8'ini verimli bir şekilde kullanırken bu değer yine DCA için % 22.2 ve CCA için ise % 12.6 civarındadır. Karşılaştırma açısından spektral verimlilikler incelendiğinde DCA algoritması, AAA-DCA algoritmasına göre mevcut spektrumu % 50.9 daha

kötü kullanırken, CCA algoritması % 72.1 oranında daha kötü kullanmaktadır. Şekil 5.2 (2)'de ise ideal AAA ve F_0 adet çoklu yola sahip AAA yapıları kullanılarak oluşturulan DCA algoritmaları arasındaki performans karşılaştırması verilmiştir. Çoklu yolların ortamda oluşturduğu etki, sistemde mevcut spektrumun % 20 oranında daha kötü kullanımına neden olmuştur. Trafik yükü 10 iken ideal AAA için spektral verimlilik % 72.1 civarındadır.



Şekil 5.2: Farklı trafik yüklerinde spektral verimliliğin karşılaştırılması

Şekil 5.2 (3)'te ise SIR sınırlı AAA-DCA algoritmasının, kod sınırlı SGA, D&D-GA, SA ve ASAGA algoritmaları ile karşılaştırmaları spektral verimlilik cinsinden ele alınmıştır. Açıkça görülmektedir ki AAA-DCA algoritması diğer algoritmalara göre mevcut spektrumu daha iyi kullanmaktadır. Spektral verimlilik değerlerini sayısal olarak karşılaştırdığımızda trafik yükü 10 iken spektral verimlilik değeri, AAA-DCA algoritması için % 45.8, SGA algoritması için % 25.6, D&D-GA algoritması için % 37.9, SA algoritmasının için % 26 ve ASAGA algoritması için de % 38.5 civarındadır.

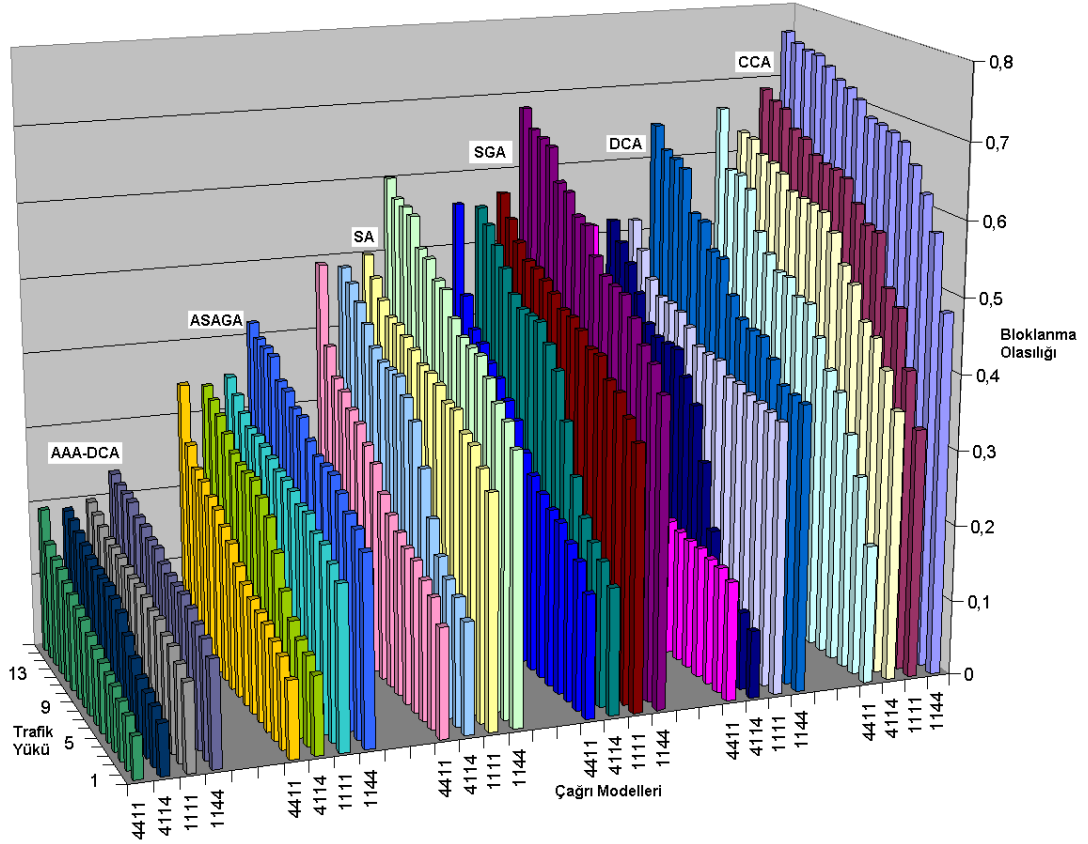
Spektral verimlilik cinsinden AAA-DCA algoritması SGA algoritmasına göre % 44.1, D&D-GA algoritmasına göre % 17.2, SA algoritmasına göre % 43.5, ASAGA algoritmasına göre % 15.9 civarında bir performans artışı sergilemektedir. Bu sonuçlara bakarak SIR sınırlı yapının sistem performansının bloklanma olasılığında olduğu gibi kod sınırlı yapıya göre spektral verimlilik cinsinden de daha iyi olduğu görülmektedir.

Şekil 5.2'yi oluşturan parametrik değişkenler şöyledir: $SF:256$, komşuluk:ötelemeli, soğutma çizelgesi:Lundy & Mees, $p_c:0.8$, $p_m:0.2$, seçim:rulet tekerleği, çaprazlama:tek noktalı, mutasyon:ötelemeli, $m:3$ ULA, $\Delta\theta:20^\circ$, $F_0:0$ ve 3, $\Delta:0.5\lambda$.

5.3.3. Çağrı model etkisi

Şu ana kadar yapılan simülasyonlarda, veri transfer hızı talebinde bulunan kullanıcıların hızları, R ile $SF \times R$ arasında değişen değerlere sahip eşit olasılıkla rasgele üretilmiş, ama formatlı sayılardan oluşmaktadır. Bu bölümde, bilinen olasılıklara göre daha önce hazırlanmış farklı çağrı modellerinin genel sistem performansı üzerindeki etkisi incelenecektir. Bu amaçla R, 2R, 4R ve 8R veri transfer hızlarına sahip çağrı modelleri, P_R , P_{2R} , P_{4R} ve P_{8R} olasılıkları ile ele alınarak ($P_R:P_{2R}:P_{4R}:P_{8R}$) ile gösterilir. Örneğin (4:4:1:1) çağrı modelinin anlamı, veri transfer isteğinde bulunacak olan mobil kullanıcıların %40'ının R hızında, %40'ının 2R hızında, %10'unun 4R hızında ve %10'unun 8R hızında olmasıdır. Bunun yanında (1:1:1:1) çağrı modelinin anlamı, tüm çağrıların eşit hızda ve %25 olasılıkla olmasıdır.

Şekil 5.3 farklı çağrı modelleri için kod bloklanma olasılığını göstermektedir. Burada en iyi performans, R:2R:4R:8R için 4:4:1:1 ve 4:1:1:4 çağrı modellerinde elde edilmiştir. Algoritma cinsinden karşılaştırma yaptığımızda en iyi sonucun AAA-DCA ile elde edildiği görülmüştür. Düşük veri transfer hızına sahip kullanıcıların fazlalığı, kod bloklanma olasılığını arttıracığından elde edilen bu sonuçlar doğaldır. Bununla birlikte 1:1:4:4 çağrı modelinde nispeten daha yüksek veri transfer hızı olacağından diğer modellere göre daha fazla bloklanma olasılığı olmaktadır.



Şekil 5.3: Farklı çağrı modellerinin kod bloklanma olasılığı üzerindeki etkisi

5.4. Algoritmik Parametrelerin Etkisi

5.4.1. Farklı çaprazlama ve mutasyon oranlarının etkisi

Bu bölümde, SGA, D&D-GA ve ASAGA algoritmalarının farklı çaprazlama ve mutasyon oranlarında değişen trafik yüklerine göre etkileri incelenmiştir.

Tablo 5.1’de sunulan veriler, sabit mutasyon oranı ($p_m=0.2$) ve farklı çaprazlama oranları ($p_c=0.2, 0.4, 0.5, 0.8$) için, değişen trafik yüklerine göre ortalama tekrar tahsis edilen kullanıcı sayısını göstermektedir. Tablodan görüldüğü üzere çaprazlama oranı arttıkça ortalama tekrar tahsis edilen kullanıcı sayısı her iki yaklaşım için de artış göstermektedir. Bunun nedeni; çaprazlama oranının çaprazlama işlemine tabi tutulacak kromozom sayısı ile doğru orantılı olmasıdır. Çaprazlama oranı arttıkça,

çaprazlama işlemi için popülasyon içerisinde o kadar çok kromozom seçilmekte ve daha verimli bir tekrar tahsis oluşturma işlemi gerçekleştirilmektedir. Tablo 5.1, değişen trafik yükleri için ASAGA ve D&D-GA algoritmaları SGA algoritmasından her zaman daha iyi sonuçlar verdiğini göstermektedir. Bununla birlikte ASAGA ve D&D-GA algoritmalarının tekrar kod tahsis ettikleri kullanıcı sayıları yaklaşık olarak benzer değerlerdedir.

Tablo 5.1: SGA, D&D-GA ve ASAGA için farklı çaprazlama oranlarının etkisi

		Trafik Yüğü (λ)							
		4	8	12	16	20	24	28	32
SGA	$p_m: 0.2$ $p_c: 0.8$	2.2	2.9	2.9	3.4	3.9	4.4	5.7	5.9
	$p_m: 0.2$ $p_c: 0.6$	0.8	1.6	2.4	3.2	3.2	3	4.6	4.8
	$p_m: 0.2$ $p_c: 0.4$	0.6	0.8	1.6	1.8	2.6	3	3.8	4
	$p_m: 0.2$ $p_c: 0.2$	0	0.8	1.2	1.6	2.4	2.4	2.4	3.2
D&D-GA	$p_m: 0.2$ $p_c: 0.8$	4	4.3	5.1	5.3	5.5	5.9	6.3	7.8
	$p_m: 0.2$ $p_c: 0.6$	1.8	3.6	3.6	5.3	5.5	5.7	6.1	7.3
	$p_m: 0.2$ $p_c: 0.4$	1.4	3	3.2	4.5	5	5.5	5.8	7.3
	$p_m: 0.2$ $p_c: 0.2$	1	1.8	2.5	2.7	3.3	3.6	4	7.2
ASAGA	$p_m: 0.2$ $p_c: 0.8$	3.72	4.51	5.05	5.49	6.24	6.72	7.78	8.86
	$p_m: 0.2$ $p_c: 0.6$	3.27	3.95	4.43	4.81	5.48	5.89	6.82	7.76
	$p_m: 0.2$ $p_c: 0.4$	2.82	3.41	3.82	4.15	4.72	5.08	5.89	6.70
	$p_m: 0.2$ $p_c: 0.2$	2.59	3.13	3.50	3.80	4.33	4.66	5.39	6.14
		36	40	44	48	52	56	60	64
SGA	$p_m: 0.2$ $p_c: 0.8$	6.6	7.4	9	10.2	11	13	13.9	16
	$p_m: 0.2$ $p_c: 0.6$	5.6	6.4	7.2	8.6	10	11	11.2	13.6
	$p_m: 0.2$ $p_c: 0.4$	5.6	5.6	6.4	6.4	8	8	8.8	10.4
	$p_m: 0.2$ $p_c: 0.2$	4	4.8	5.6	5.6	7.2	8	8.8	9.6
D&D-GA	$p_m: 0.2$ $p_c: 0.8$	9	10.4	14.5	15.3	15.8	18.1	19.4	22.8
	$p_m: 0.2$ $p_c: 0.6$	8.8	10	11.7	12.6	15.8	17.1	18.9	19.8
	$p_m: 0.2$ $p_c: 0.4$	8.5	9.4	11.7	12	14.4	15.3	18	19.8
	$p_m: 0.2$ $p_c: 0.2$	8	9	11.7	11.7	12.6	14	17.1	18.9
ASAGA	$p_m: 0.2$ $p_c: 0.8$	10.08	11.72	15.45	16.68	17.96	20.88	22.51	26.02
	$p_m: 0.2$ $p_c: 0.6$	8.84	10.28	13.55	14.63	15.75	18.31	19.73	22.82
	$p_m: 0.2$ $p_c: 0.4$	7.63	8.87	11.69	12.62	13.59	15.80	17.03	19.69
	$p_m: 0.2$ $p_c: 0.2$	6.99	8.13	10.72	11.57	12.46	14.49	15.61	18.05

Tablo 5.2’de sunulan veriler ise sabit çaprazlama oranı ($p_c=0.8$) ve farklı mutasyon oranları ($p_m=0.2, 0.1, 0.05, 0.01$) için, değişen trafik yüklerine göre ortalama tekrar tahsis edilen kullanıcı sayısını göstermektedir. Mutasyon oranı azaldıkça ortalama tekrar tahsis edilen kullanıcı sayısının her iki yaklaşım için de bir artış gösterdiği

Tablo 5.2’den görülmektedir. Mutasyon operatörü kromozomların çoğunluğunun yerel minimum değere takılmasını engellemek için kullanılan bir operatör olduğundan mutasyon oranı 0.1 ile 0.001 değerleri arasında çok küçük bir değer olarak kullanılmaktadır. Bununla beraber mutasyon oranını belirleyen kesin bir koşul veya kriter yoktur. Tablo 5.2, Tablo 5.1’de gördüğümüz algoritmik performanslar ile karşılmaktadır. Buna göre ASAGA ve D&D-GA algoritmaları SGA algoritmasından her zaman daha iyi sonuçlar vermektedir.

Tablo 5.2: SGA, D&D-GA ve ASAGA için farklı mutasyon oranlarının etkisi

		Trafik Yüğü (λ)							
		4	8	12	16	20	24	28	32
SGA	$p_m: 0.2 \ p_c: 0.8$	2.2	2.9	2.9	3.4	3.9	4.4	5.7	5.9
	$p_m: 0.1 \ p_c: 0.8$	2.7	3.2	3.4	4	4.4	5.2	6.2	6.5
	$p_m: 0.05 \ p_c: 0.8$	3	3.7	4	4.4	4.8	6	6.8	7.1
	$p_m: 0.01 \ p_c: 0.8$	3.3	4	4.8	5	5.5	6.4	7.5	7.9
D&D-GA	$p_m: 0.2 \ p_c: 0.8$	4	4.3	5.1	5.3	5.5	5.9	6.3	7.8
	$p_m: 0.1 \ p_c: 0.8$	4.2	4.5	5.4	5.6	5.7	6.6	7.1	8.5
	$p_m: 0.05 \ p_c: 0.8$	4.6	5	5.7	5.9	6.8	7.6	8	9.5
	$p_m: 0.01 \ p_c: 0.8$	4.8	5	6.1	6.8	7.9	8.2	9.1	10
ASAGA	$p_m: 0.2 \ p_c: 0.8$	3.72	4.51	5.05	5.49	6.24	6.72	7.78	8.86
	$p_m: 0.1 \ p_c: 0.8$	4.02	4.87	5.45	5.93	6.74	7.25	8.40	9.57
	$p_m: 0.05 \ p_c: 0.8$	4.29	5.19	5.81	6.32	7.19	7.73	8.95	10.20
	$p_m: 0.01 \ p_c: 0.8$	4.51	5.46	6.12	6.65	7.56	8.13	9.42	10.73
		36	40	44	48	52	56	60	64
SGA	$p_m: 0.2 \ p_c: 0.8$	6.6	7.4	9	10.2	11	13	13.9	16
	$p_m: 0.1 \ p_c: 0.8$	7.1	8.9	9.8	10.6	12.5	14.3	13.9	16.6
	$p_m: 0.05 \ p_c: 0.8$	8	9.4	10	11.5	13.6	14.5	15	16.9
	$p_m: 0.01 \ p_c: 0.8$	8.8	9.8	10.5	12.2	14	14.9	15.6	17.1
D&D-GA	$p_m: 0.2 \ p_c: 0.8$	9	10.4	14.5	15.3	15.8	18.1	19.4	22.8
	$p_m: 0.1 \ p_c: 0.8$	10	10.4	15.4	16.3	18.2	20.1	21.7	23.5
	$p_m: 0.05 \ p_c: 0.8$	10.5	11	16.2	17.2	19.2	21.6	22.6	25.8
	$p_m: 0.01 \ p_c: 0.8$	10.5	11.4	17.4	19.3	20.2	22.1	23.7	26.6
ASAGA	$p_m: 0.2 \ p_c: 0.8$	10.08	11.72	15.45	16.68	17.96	20.88	22.51	26.02
	$p_m: 0.1 \ p_c: 0.8$	10.89	12.67	16.69	18.02	19.40	22.56	24.31	28.11
	$p_m: 0.05 \ p_c: 0.8$	11.61	13.50	17.78	19.20	20.68	24.04	25.91	29.96
	$p_m: 0.01 \ p_c: 0.8$	12.21	14.20	18.71	20.20	21.76	25.30	27.26	31.52

5.4.2. Farklı seçim tekniklerinin etkisi

Bu bölümde farklı seçim tekniklerinin sistem performansı üzerindeki etkisi incelenecektir. Seçim işlemi, genetik bir operatördür ve bir sonraki popülasyonu oluşturacak kromozomları üretmek için mevcut popülasyondan ebeveynlerin (anne-baba) seçilmesi işlemini gerçekleştirir. Burada seçilen kromozom, bir sonraki adımda yine seçim operatörüne girmeden önce çaprazlama ve mutasyon operatörleri tarafından kullanılır. Bu çalışmada rulet tekerleği, turnuva, en iyi yüzde ve rasgele seçim teknikleri gibi farklı seçim tekniklerinin kullanımı ile tekrar tahsis edilen kullanıcıların sayısı ve bunun genel sistem performansı üzerindeki etkisi incelenmiş ve elde edilen sonuçlar Tablo 5.3'te SGA, D&D-GA ve ASAGA için verilmiştir. Sistem performansını karşılaştırmak için tekrar tahsis edilen kullanıcıların sayısı ele alınmıştır.

Bu bölümde incelenen farklı seçim teknikleri şu şekilde açıklanabilir: Rulet tekerleği tekniğinde kromozomun seçilme olasılığı, uygunluk değeri ile ilişkilidir. Yani uygunluk değeri büyük olan bir sonraki popülasyonda yeniden yer alır. Turnuva tekniğinde, tamsayı olarak rasgele belirlenmiş N adet rulet tekerleği seçim işlemi sonucunda elde edilen kromozom altkümesi incelenerek içlerinden en iyinin bir sonraki nesle aktarımı yapılır. Yapısal olarak rulet tekerleğine göre N kat daha fazla hesaplama yükü getirdiği açıktır. En iyi yüzde tekniğinde, kromozomlar popülasyonun uygunluk değerleri cinsinden en iyi ξ yüzdelik kısmından rasgele olarak seçilirler. Burada ξ , 0 ile 100 arasında seçilebilir. Bu çalışmada 0–30 arasında rasgele bir değerde seçim işlemi ele alınmıştır. Son olarak rasgele seçim operatöründe ise kromozomlar uygunluk değerine bakılmaksızın rasgele olarak elde seçilirler (Baker, 1985).

Simülasyon sonuçlarına göre en iyi performansın turnuva tekniği için elde edildiği, onu en iyi yüzde, rulet tekerleği ve rasgele seçim tekniklerinin izlediği görülmüştür. Bu sonuçlar SGA, D&D-GA ve ASAGA için geçerlidir. Turnuva tekniği, rulet tekerleği tekniği ile N defa elde edilen seçilmiş kromozomları bir kez daha seçme işlemine tabi tuttuğu için en iyi sonucu vermektedir. En iyi yüzde tekniği de, rasgele belirlenmiş belli bir yüzdedeki uygunluk değerine sahip kromozomları aldığından,

rulet tekerleği tekniğine göre daha iyi sonuç vermektedir. Çalışmalarda sıklıkla tercih edilen rulet tekerleği tekniği, turnuva ve en iyi yüzde tekniklerine göre tekrar tahsis edilen kullanıcı sayısı cinsinden daha kötü sonuçlar vermektedir. Rasgele tekniği ise popülasyondaki kromozomları belli bir kritere göre almadığından performansı en düşük olan seçim tekniğidir.

Tablo 5.3: Farklı seçim tekniklerinin tekrar tahsis edilen kullanıcı sayıları cinsinden sistem performansı üzerine etkisi

		Trafik Yüğü (λ)							
		4	8	12	16	20	24	28	32
SGA	Rulet Tekerleği	2.2	2.9	2.9	3.4	3.9	4.4	5.7	5.9
	Turnuva	3.8	4.6	4.6	6.6	8.4	8.2	9	10.8
	En iyi yüzde	2.3	3.4	3.9	5	5.2	6.8	7.1	7.8
	Rasgele	0.8	1.6	2.6	3	3.4	4	5.4	5.4
D&D-GA	Rulet Tekerleği	4	4.3	5.1	5.3	5.5	5.9	6.3	7.8
	Turnuva	5.6	6.4	7.4	7.4	8.3	9.2	10.2	11.9
	En iyi yüzde	4	4.9	5.8	6.7	7.6	7.7	8	8.5
	Rasgele	1.8	2.7	3	3.6	3.6	4.5	5.4	6.4
ASAGA	Rulet Tekerleği	3.72	4.51	5.05	5.49	6.24	6.72	7.78	8.86
	Turnuva	5.75	6.92	7.65	8.91	11.07	11.41	12.43	14.74
	En iyi yüzde	3.8	5.21	6.22	7.46	8.46	9.5	9.78	10.58
	Rasgele	1.49	2.64	3.58	4.21	4.66	5.57	7	7.66
		36	40	44	48	52	56	60	64
SGA	Rulet Tekerleği	6.6	7.4	9	10.2	11	13	13.9	16
	Turnuva	11.4	12	12.6	14.4	14.4	21.6	25.2	28.2
	En iyi yüzde	8.2	9.1	9.6	10.8	11.7	14.7	15.6	16.9
	Rasgele	6	6.2	7.2	7.8	8.6	9.2	10.4	14.4
D&D-GA	Rulet Tekerleği	9	10.4	14.5	15.3	15.8	18.1	19.4	22.8
	Turnuva	12.8	15.2	17.7	18.5	20.4	23	26.2	34.2
	En iyi yüzde	9.5	10.5	14.8	15.3	17.2	19.2	21.7	23.4
	Rasgele	8.2	10.2	12.9	13.8	14.7	17.4	18.6	20.3
ASAGA	Rulet Tekerleği	10.08	11.72	15.45	16.68	17.96	20.88	22.51	26.02
	Turnuva	15.72	18.01	20.15	21.72	23.34	30.06	34.84	42.17
	En iyi yüzde	11.55	12.99	16.11	17.15	19.32	22.85	25.22	27.08
	Rasgele	9.17	10.59	13.01	13.8	15.26	17.02	18.92	23.29

Tablo 5.3. açıkça göstermektedir ki değişen trafik yükleri ve farklı seçim teknikleri için ASAGA ve D&D-GA algoritmaları SGA algoritmasından her zaman daha iyi sonuçlar vermektedir. Ayrıca her seçim tekniğinin hesaplama yükleri de ayrı ayrı ele alınmıştır. Turnuva, rulet tekerleği ve en iyi yüzde seçim tekniklerinin hesaplama yükleri $O(2n)$ iken, rasgele seçim tekniğinin hesaplama yükü $O(n)$ civarındadır.

5.4.3. Farklı çaprazlama tekniklerinin etkisi

Çaprazlama operatörü, iki kromozomun (anne-baba) bir araya gelerek yeni kromozomlar (çocuklar) oluşturması amacıyla kullanılır. Çaprazlama tekniğinin uygulanmasındaki amaç daha iyi özellikli kromozomların oluşturulmasıdır. Çaprazlama tekniği dışarıdan girilen çaprazlama oranına bağlı olarak gerçekleştirilir. SGA, D&D-GA ve ASAGA için çaprazlama oranı (p_c) 0.8 alınarak çeşitli çaprazlama teknikleri incelenmiştir. Bunlar tek noktalı, iki noktalı, aritmetik ve düzenli çaprazlama teknikleridir (Eshelman ve diğ., 1989). Tek noktalı çaprazlama tekniğinde, rasgele seçilen noktadan itibaren kromozomların genleri kendi aralarında yer değiştirerek yeni kromozomları oluşturur. İki noktalı çaprazlamada, rasgele seçilen nokta sayısı ikidir. Düzenli çaprazlama tekniği belli bir olasılıkla rasgele belirlenmiş genleri birinci kromozomdan, kalan genleri de kalan genleri de ikinci kromozomdan alarak yeni kromozomları oluşturur. Bu teknik ile anne ve baba kromozomlara ait genler parça parça değil (tek veya iki noktalı çaprazlama operatöründeki gibi) gen bazında alınarak yeni kromozomlar oluşturulur. Bazı problemler için bu bir avantaj gibi görünse de bazı durumlarda sistem yapısını etkilemekte ve yapının bozulmasına neden olmaktadır. Bununla birlikte, bu teknik OVFS kod ağacının tekrar tahsis edilmesi yapısını etkilememektedir. Son olarak aritmetik çaprazlama tekniğinde anne ve baba kromozomun doğrusal kombinasyonu ile yeni kromozomlar oluşturulmaktadır. Bu doğrusal kombinasyon

$$cocuk1 = a * anne + (1 - a) * baba$$

$$cocuk2 = (1 - a) * anne + a * baba \quad (5.3)$$

şeklinde. Burada a , bir ağırlık faktörüdür ve tekrar kod tahsis etme başarısına göre kromozom içinde hareket eder. Farklı çaprazlama oranlarının sistem üzerindeki etkisini karşılaştırmak için, diğer operatörlerle (seçim, mutasyon) aynı yapıda özelliklerin karşılaştırılması amaçlanmıştır. Bu amaçla Tablo 5.4’de, farklı çaprazlama tekniklerinin kullanımı ile tekrar kod tahsis edilen kullanıcıların sayısı incelenmiştir.

Tablo 5.4: Farklı çaprazlama tekniklerinin tekrar tahsis edilen kullanıcı sayıları cinsinden sistem performansı üzerine etkisi

		Trafik Yüğü (λ)							
		4	8	12	16	20	24	28	32
SGA	Tek Nokta	2.2	2.9	2.9	3.4	3.9	4.4	5.7	5.9
	İki Nokta	1.6	1.8	2.7	3.1	3.6	4	4.5	5.4
	Düzenli	2.6	3.4	4.4	4.6	6.8	7.4	8.2	8.8
	Aritmetik	3.5	4.8	6.7	8.6	9.5	11.4	12.4	13.1
D&D-GA	Tek Nokta	4	4.3	5.1	5.3	5.5	5.9	6.3	7.8
	İki Nokta	2.2	2.8	3.2	3.8	4.7	5.6	5.9	7.2
	Düzenli	4.9	5.8	6.3	6.8	6.8	7.7	8.7	9.5
	Aritmetik	5.2	5.9	6.9	7.7	8.3	8.8	9.7	10
ASAGA	Tek Nokta	3.72	4.51	5.05	5.49	6.24	6.72	7.78	8.86
	İki Nokta	1.85	2.19	2.92	3.41	4.07	4.66	5.1	6.17
	Düzenli	3.39	4.28	5.18	5.48	6.8	7.54	8.44	9.13
	Aritmetik	4.18	5.29	6.79	8.12	8.85	9.93	10.88	11.34
		36	40	44	48	52	56	60	64
SGA	Tek Nokta	6.6	7.4	9	10.2	11	13	13.9	16
	İki Nokta	5.7	6.3	8.1	9.9	10.8	12.7	14.4	15.3
	Düzenli	9.6	10	10.8	12	14.4	15.6	18	19.2
	Aritmetik	14.5	15.2	16.3	17.8	19	19.4	20	20.9
D&D-GA	Tek Nokta	9	10.4	14.5	15.3	15.8	18.1	19.4	22.8
	İki Nokta	8.6	9.8	10.4	10.8	11.7	12.6	15.3	18.9
	Düzenli	10.6	14.5	15.4	16.5	17.1	19.9	20.9	26.6
	Aritmetik	11.5	16.2	19.8	23.2	24.6	26.5	26.8	30.4
ASAGA	Tek Nokta	10.08	11.72	15.45	16.68	17.96	20.88	22.51	26.02
	İki Nokta	6.85	7.66	9.1	10.33	11.23	12.65	14.83	16.91
	Düzenli	10.07	11.83	12.69	13.89	15.63	17.49	19.34	22.3
	Aritmetik	12.82	15.68	17.88	20.14	21.44	22.4	22.9	24.77

Farklı çaprazlama teknikleri kullanılarak elde edilen sonuçlara göre, en iyi performans aritmetik çaprazlama tekniği kullanıldığında gerçekleştirilmiştir. Onu düzenli, tek noktalı ve iki noktalı çaprazlama tekniklerinin izlediği görülmektedir. Bu sonuçlar, SGA, D&D-GA ve ASAGA için geçerlidir. Aritmetik çaprazlama tekniği ile üretilen yeni kromozomlar, anne-baba kromozomların doğrusal bir kombinasyonu ile oluşmaktadır ve sayısal olarak daha fazla kullanıcı için kod tekrar tahsisi imkanı sağlamıştır. Düzenli çaprazlama tekniği, anne ve babadan rasgele seçilen genlerle yeni kromozomları oluşturduğundan aritmetik çaprazlama tekniğine göre daha az kullanıcıya kod tekrar tahsisi imkanı sağlamıştır. Bu iki çaprazlama tekniğini, tek noktalı ve iki noktalı çaprazlama teknikleri izlemektedir. İki noktalı çaprazlama tekniği özellikle OVSF kod ağacı yapısını etkilediğinden kötü sonuçlar vermektedir.

Yerdeğiştirme işlemi için GA bloğuna giren hem kullanıcı sayısının hem de her bir kullanıcının veri transfer hızının değişmediği açıkça görülmektedir. Ancak bu kullanıcıların sadece ilgili dallardaki yerleri değişebilir. Tekrar tahsis edilen kullanıcıların sayısı, genel sistem performansına olan etkisi ile doğru orantılı olduğundan tüm bu yargılar genel sistem performansı için de geçerlidir.

Tablo 5.4. açıkça göstermektedir ki değişen trafik yükleri ve farklı çaprazlama teknikleri için ASAGA ve D&D-GA algoritmaları, SGA algoritmasından her zaman daha iyi sonuçlar vermektedir. Her bir çaprazlama tekniğinin hesaplama yükü de ayrıca hesaplanmıştır. Tüm çaprazlama operatörleri için bu değer aynı ve $O(n)$ 'dir.

5.4.4. Farklı mutasyon tekniklerinin etkisi

Bu bölüm, farklı mutasyon tekniklerinin genel sistem performansı üzerindeki etkisini incelemektedir. Mutasyon genetik operatörü kullanılarak bir kromozomdaki bir ya da birden fazla genin yeri, başlangıç durumuna göre değiştirilir. Bu işlem genetik çeşitliliği arttırmak ve kromozomların yerel minimuma takılmasını önlemek amacıyla yapılır. Çaprazlama operatöründe olduğu gibi mutasyon operatörü de kullanıcıya bağlı olarak girilen çaprazlama oranına bağlı olarak gerçekleştirilir. Bu değer genellikle küçük değerlerde olması uygundur (0.01, 0.05, 0.1). Eğer mutasyon oranı yüksek bir değer seçilirse arama durumu kuralsız rasgele arama halini alır.

Bu çalışmada, yer değiştirme ve öteleme mutasyon teknikleri, GA bloğuna uygulanmış ve sistem üzerindeki etkileri gözlenmiştir. Yer değiştirme mutasyon tekniğinde, kromozomdan seçilen rasgele iki gen arasında yer değiştirme işlemi gerçekleştirilir. Öteleme mutasyon tekniğinde ise seçim kriterine göre sağa ya da sola öteleme işlemi gerçekleştirilir. Farklı mutasyon tekniklerinin sistem üzerindeki etkisinin sayısal sonuçları, Tablo 5.5'te SGA, D&D-GA ve ASAGA için tekrar tahsis edilen kullanıcı sayıları cinsinden verilmiştir.

Tablo 5.5: Farklı mutasyon tekniklerinin tekrar tahsis edilen kullanıcı sayıları cinsinden sistem performansı üzerine etkisi

		Trafik Yüğü (λ)							
		4	8	12	16	20	24	28	32
SGA	Yer deęiřtirme	2.2	2.9	2.9	3.4	3.9	4.4	5.7	5.9
	Öteleme	2.8	3.4	3.7	4.2	5.6	6	6.8	7.6
D&D-GA	Yer deęiřtirme	4	4.3	5.1	5.3	5.5	5.9	6.3	7.8
	Öteleme	4.8	5.6	6.2	6.4	7.8	8.2	9	9.8
ASAGA	Yer deęiřtirme	3.01	3.65	4.06	4.49	4.37	4.88	5.98	6.96
	Öteleme	3.72	4.51	5.05	5.49	6.24	6.72	7.78	8.86
		36	40	44	48	52	56	60	64
SGA	Yer deęiřtirme	6.6	7.4	9	10.2	11	13	13.9	16
	Öteleme	8.2	9.1	9.6	10.8	12.2	13.6	14.6	17.1
D&D-GA	Yer deęiřtirme	9	10.4	14.5	15.3	15.8	18.1	19.4	22.8
	Öteleme	10.6	12.2	14.6	16.2	17	18.6	20.8	24.4
ASAGA	Yer deęiřtirme	8.34	10.76	14.91	15.75	16.44	20.14	21.21	24.33
	Öteleme	10.08	11.72	15.45	16.68	17.96	20.88	22.51	26.02

Farklı trafik yükleri altında yapılan simülasyonlarda, öteleme mutasyon teknięinin, yer deęiřtirme mutasyon teknięine göre daha fazla performans artışı saęladığı gözlenmiřtir. Dolayısıyla bu artış, sistem performansını da doğru orantılı olarak etkilemiřtir. Aynı durum, SGA, D&D-GA ve ASAGA için de geçerlidir. Öteleme mutasyon teknięi, mutasyon işlemi süresince OVSF kod ağaç yapısını daha az etkilediğinden, bu davranış simülasyon sonuçlarına yansımıştır. Seçim ve çaprazlama tekniklerinin etkilerinin incelendiğı bölümlere paralel olarak, trafik yüğü ile tekrar tahsis edilen kullanıcı sayısı ve toplam sistem performansı, doğru orantılıdır. Bununla birlikte ASAGA ve D&D-GA hem trafik yüküne göre hem de kullanılan mutasyon tekniklerine göre SGA'ya göre daha fazla kullanıcıya kod tahsis etmiřtir.

Tablo 5.5, deęişen trafik yükleri ve farklı mutasyon teknikleri için ASAGA ve D&D-GA algoritmalarının SGA algoritmasından her zaman daha iyi sonuçlar verdiğini açıkça göstermektedir. Mutasyon operatörlerinin hesaplama yükünü elde etmek için her teknik, bir kromozoma uygulanacak şekilde ele alındı. Yer deęiřtirme ve öteleme mutasyon operatörleri için hesaplama yükü, aynı ve $O(n)$ 'dir. Ancak öteleme mutasyon operatörü, tüm kromozom üzerinde öteleme yapacağından sisteme olan

etkisi daha fazla olacaktır. Bu durum da sonuca yansıyacak ve daha fazla hesaplama yükü oluşturacaktır.

5.4.5. Komşuluk hareketinin etkisi

Bu bölümde, farklı komşuluk hareketlerinin genel sistem performansı üzerindeki etkisi incelenmektedir. Komşuluk hareketi, önceden kararlaştırılmış iç döngü koşulu sağlanıncaya kadar arama uzayında var olan çözüme en yakın, olası yeni çözümler üretmek için kullanılmaktadır. Bu çalışmada, yeni elde edilen çözümler için gerek koşul olan uygun OVSF kodunun sağlanıp sağlanmadığı incelenmiştir. SA algoritmasındaki komşuluk hareketi, GA'daki mutasyon operatörüne benzer yapıdadır. Burada yerdeğiştirme (swap) ve öteleme (shift) komşuluk hareketleri, belirtilen başlangıç ve son sıcaklıklar (0.9'dan 0.1'e kadar) ile uygulanmıştır. Yerdeğiştirme komşuluk hareketinde, rasgele seçilen iki gen arasında yer değiştirme işlemi yapılır. Öteleme komşuluk hareketinde ise seçim koşuluna göre sağa ya da sola öteleme işlemi yapılır. Bu çalışmada da iç döngü denge koşulu beş olarak belirlenmiş, soğutma çizelgesi de Lundy & Mees olarak seçilmiştir.

Farklı komşuluk hareketlerinin genel sistem performansı üzerindeki etkisinin sayısal sonuçları, Tablo 5.6'da tekrar tahsis edilen kullanıcıların sayıları cinsinden verilmiştir. Farklı trafik yükleri altında yapılan simülasyonlarda, öteleme komşuluk hareketinin, yerdeğiştirme komşuluk hareketine göre daha fazla performans artışı sağladığı gözlenmiştir. Dolayısıyla bu artış, genel sistem performansını da doğru orantılı olarak etkilemiştir. Öteleme komşuluk hareketi, yaptığı işlem süresince OVSF kod ağacı yapısını daha az etkileyerek çalıştığından, bu davranış simülasyon sonuçlarına da yansımıştır. Tekrar kod tahsis edilen kullanıcı sayısı ve toplam sistem performansı, doğru orantılıdır. Bununla birlikte trafik yükü ile tekrar tahsis edilen kullanıcı sayısı da doğru orantılıdır.

Tablo 5.6: Farklı komşuluk hareketlerinin tekrar tahsis edilen kullanıcı sayıları cinsinden SA algoritması üzerine etkisi

	Trafik Yüğü (λ)							
	4	8	12	16	20	24	28	32
Yer deęiřtirme	2.24	2.36	3.21	3.14	4.27	4.44	5.47	6.15
Öteleme	2.32	3	3.36	3.7	4.84	5.02	5.88	6.5
	36	40	44	48	52	56	60	64
Yer deęiřtirme	6.81	8.1	8.68	11.59	13.99	15.09	16.73	19.62
Öteleme	7.36	8.92	11.5	12.38	14.16	16.22	18.12	20.56

5.4.6. Soęutma çizelgesinin etkisi

Soęutma çizelgesinin genel sistem performansı üzerindeki etkisini belirlemek için, orantısal azalımlı ve Lundy & Mees soęutma çizelgeleri kullanılmıřtır. Arama uzayında adım aralıęının belirlenmesi amacı ile soęutma çizelgesi kullanılmakta ve her bir adımda bu deęer kullanılan teknięe göre yinelenmektedir. Burada kullanılan bařlangıç sıcaklıęı, son sıcaklık ve algoritmanın tekrarlama sayısının doęru seęimi, çözüme yakınsama açısından büyük bir önem taşımaktadır. Simülasyonlarda bařlangıç ve son sıcaklık sırasıyla 0.9 ve 0.1 olarak alınmıřtır. Aynı zamanda komşuluk hareketi olarak da öteleme komşuluk hareketi kullanılmıřtır.

5.4.6.1. Genel sistem performansı üzerine etkisi

Soęutma çizelgesinin genel sistem performansı üzerindeki etkisinin sayısal sonuçları Tablo 5.7’de verilmektedir. Farklı trafik yükleri ve aynı komşuluk hareketi altında yapılan simülasyonlarda Lundy & Mees soęutma çizelgesinin orantısal azalımlı soęutma çizelgesine göre daha fazla performans artışı saęladığı gözlemlenmiřtir. Trafik yükü arttıkça, tekrar tahsis edilen kullanıcı sayısı da artmaktadır. Lundy & Mees soęutma çizelgesi, yapısal olarak orantısal azalımlı soęutma çizelgesine göre biraz daha hızlı bir şekilde yerel deęerlere ulařtıęından, bu sistem performansını olumlu olarak etkilemiřtir. OVFS kod ağacının optimizasyonunda temel gereksinim, yeni kullanıcıya ihtiyaç duyduęu veri transfer hızını saęlayacak uygun kanallama kodunun belirlenmesi olduęundan, soęutma çizelgesinin belli bir noktasında

yakınsama koşulu sağlanır. Dolayısıyla soğutma çizelgesinin seçimi, bu yakınsama noktasının belirlenmesinde önemlidir.

Tablo 5.7: Soğutma çizelgesinin tekrar tahsis edilen kullanıcı sayıları cinsinden SA algoritması üzerine etkisi

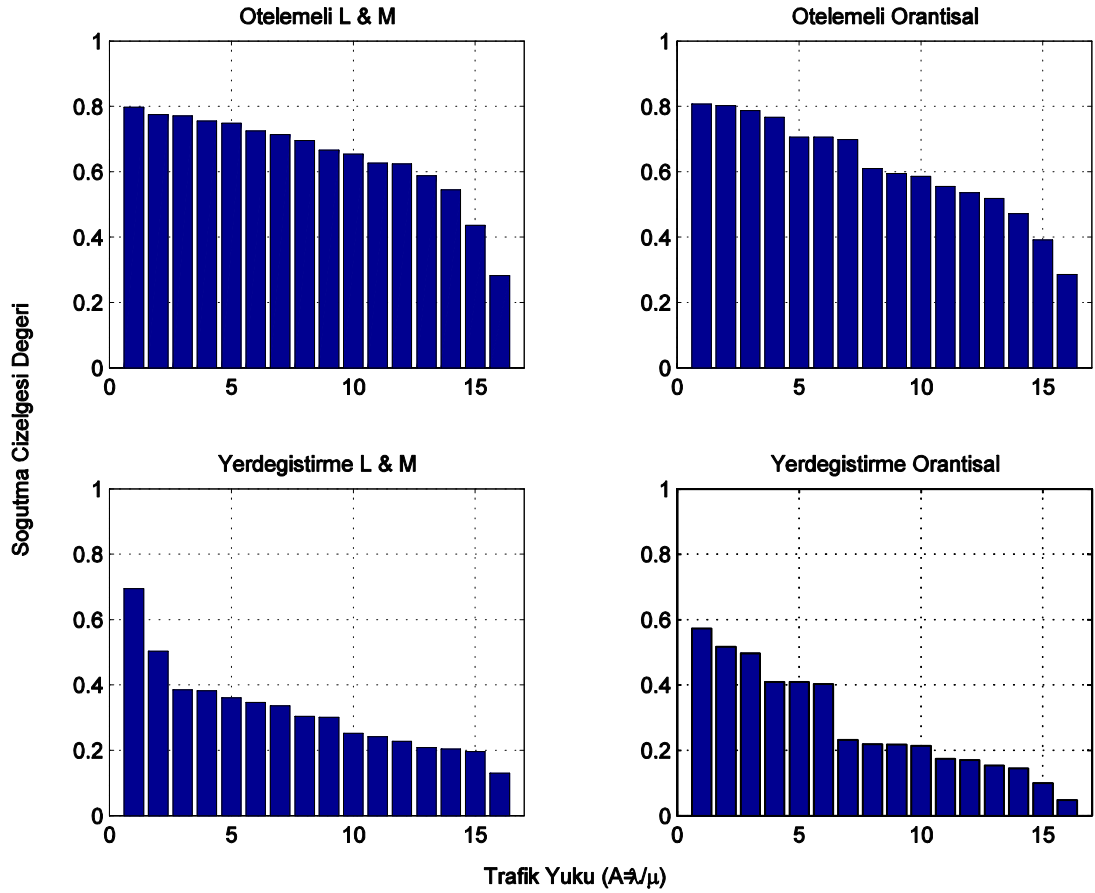
	Trafik Yüğü (λ)							
	4	8	12	16	20	24	28	32
L&M	2.32	3	3.36	3.7	4.84	5.02	5.88	6.5
Orantısal	2.19	3.19	3.4	3.6	4.11	4.54	4.89	5.17
	36	40	44	48	52	56	60	64
L&M	7.36	8.92	11.5	12.38	14.16	16.22	18.12	20.56
Orantısal	6.24	7.23	9.29	10.42	12.34	16.0	17.66	19.78

5.4.6.2. Yakınsama noktasının etkisi

OVSF kod ağacının SA algoritması ile optimizasyonunda amaç, daha önce de belirtildiği gibi yeni gelen kullanıcının veri transfer hızı ihtiyacına uygun OVSF kanallama kodunu sağlayacak olan dalın kapasite dahilinde tespitidir. Bu bağlamda uygun OVSF kanallama kodu sağlandığı zaman, SA algoritması görevini yapmış demektir. Bu da soğutma çizelgesinin herhangi bir noktasında meydana gelebilir. Bu alt bölümde SA algoritmasının soğutma çizelgesinin hangi bölümünde yakınsadığı incelenecektir. Orantısal azalımlı ve Lundy & Mees soğutma çizelgelerinin farklı trafik yükleri altında tekrar tahsis ettikleri kullanıcıların soğutma çizelgesinin hangi anlık değer olduğu ele alınacaktır.

Şekil 5.4, farklı trafik yüklerinde tekrar tahsis edilen kullanıcıların, soğutma çizelgesinin hangi noktasında yakınsadıklarını göstermektedir. Alınan bu değerler, ilgili trafik yükünde yeniden kod tahsis edilen kullanıcıların ortalamasıdır. Şekilden de anlaşılacağı üzere tüm komşuluk hareketleri ve soğutma çizelgeleri için trafik yükü arttıkça, tekrar tahsis edilen kullanıcılar için uygun OVSF kanallama kodunu içeren boş dalların tespiti zorlaşmaktadır. Bu etki, yerdeğiştirme komşuluk hareketinde daha büyüktür. Öteleme komşuluk hareketi ile yapılan aramada daha hızlı yakınsama olduğu görülmektedir. Burada yanlış anlaşılması gereken husus, boş dalların tespitinin zor olması, bulunamaması anlamına gelmemesidir. Yani yapılan bu çalışmanın trafik yükü arttıkça tekrar tahsis edilen kullanıcı sayısının

artması ile bir çelişki oluşturduğu manasına gelmemelidir. Soğutma çizelgesinin sonlarına doğru yakınsama meydana gelmektedir. Bu da sadece simülasyon süresini etkilemektedir.

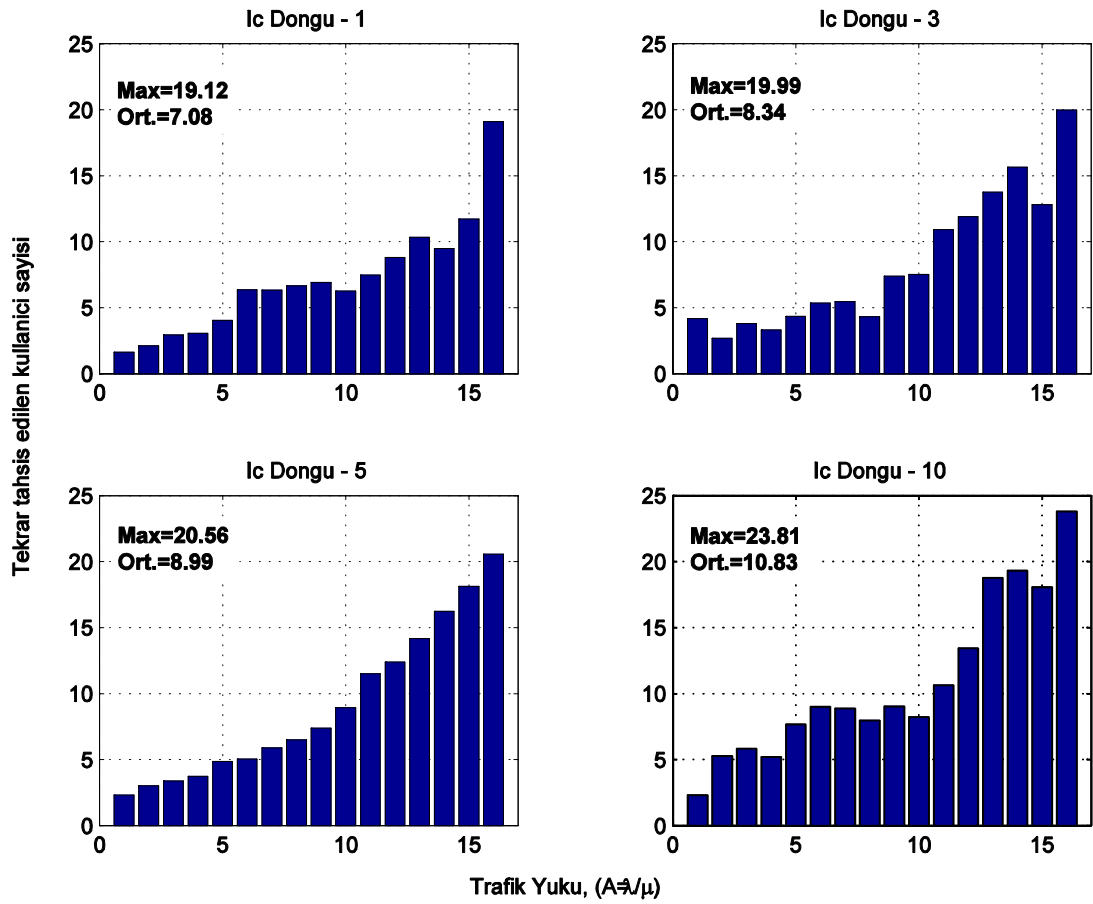


Şekil 5.4: SA algoritmasında tekrar tahsis edilen kullanıcıların soğutma çizelgesi üzerindeki yakınsama noktasının tespiti

5.4.6.3. İç döngünün etkisi

İç döngü koşulu, her sıcaklık değerinde kaç kez olası yeni çözüm üretileceğine karar verir. SA algoritması uygulanırken iç döngü koşulunun sağlanıp sağlanmadığını kontrol edebilmek için dikkate alınan koşul, yeteri kadar yeni çözüm üretiminin gerçekleştirilip gerçekleştirilmediğidir. Yeni çözüm, elde edilen yerel minimum değerın yakınlarında arama yapılması demektir. Bu bağlamda, denge koşulu olarak tanımlanan koşul ile 1, 3, 5 ve 10 defa yerel arama alanında arama yaptırılarak iç döngü, dengeye getirilir. İç döngünün sayısındaki değişim, sistem performansını da

etkileyecektir. Şekil 5.5, SA algoritması için tekrar tahsis edilen kullanıcı sayısının iç döngüdeki değişimden nasıl etkilendiğini göstermektedir. Tekrar tahsis edilen kullanıcı sayısı ve trafik yükündeki artış ile iç döngü sayısındaki artış doğru orantılı olarak değişmektedir. İç döngü koşulu, daha önce de belirtildiği gibi belli sayıda yerel aramanın tekrarlanmasıdır. İç döngüye giren olası çözüm, OVSF kod ağacının diklik prensibine bağlı ve belli bir uygunluk değerine sahiptir. İç döngü için verilen bu tekrarlama süreci içindeki her bir adımda, ağaç yapısındaki değişim de kontrol edilmektedir. Dolayısıyla uygunluk değeri, daha küçük olan ancak OVSF kod ağacı diklik prensibine bağlı olmayan ağaç yapıları dikkate alınmaz. Bununla birlikte yapılan yerel aramada, uygunluk değeri büyük olan zaten göz ardı edilmektedir.



Şekil 5.5: SA algoritmasında tekrar tahsis edilen kullanıcıların iç döngü sayısına göre değişimi

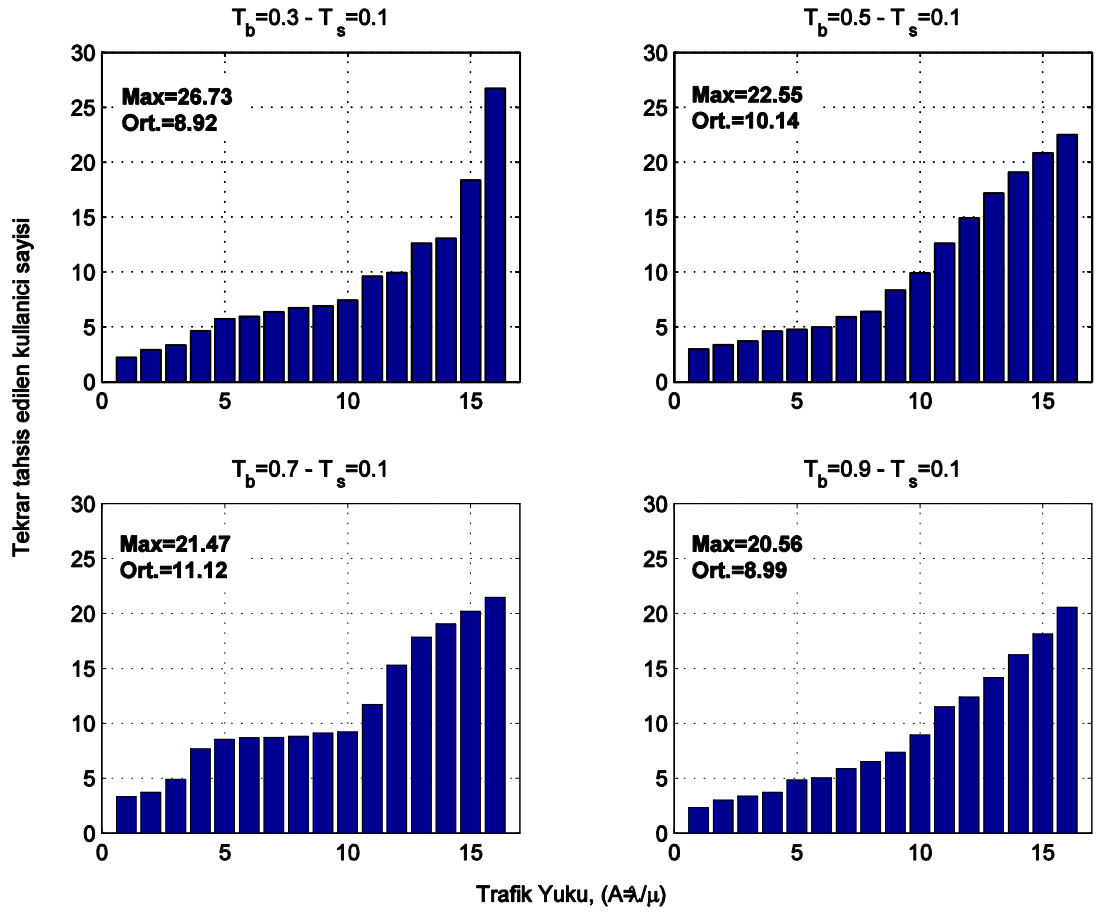
Literatürde yapılan ön çalışmalar iç döngü arttıkça yerelde daha fazla arama yapıldığından ve dolayısıyla daha iyi sonuç elde etme olasılığından bahsetmektedir (Leung ve diğ., 2001). Artış miktarı problemin tanımına ve problemin yapısını

etkileyen diğer sınırlayıcı bileşenlere bağlı olarak değişse de bu çalışmada farklı iç döngü değerleri kullanılarak elde edilen sonuçlar, literatüre paralel bir anlam içermektedir.

5.4.6.4. Başlangıç ve son sıcaklığın etkisi

Başlangıç ve son sıcaklıklar, arama uzayında yapılan aramalardaki adım aralıklarının belirlenmesinde önemli bir ölçüttür. Soğutma çizelgesinin seçimi konusunda, farklı soğutma çizelgeleri üzerinde yapılan simülasyonlarda başlangıç sıcaklığı 0.9 ve son sıcaklık 0.1 için grafiksel gösterimler ele alınmıştır. Farklı başlangıç ve son sıcaklıkların OVSF kod ağacının SA algoritması ile optimizasyonunda komşuluk hareketi olarak öteleme tekniği, soğutma çizelgesinin seçiminde de Lundy & Mees çizelgesi seçilmiş ve elde edilen simülasyon sonuçları Şekil 5.6’da verilmiştir.

Şekil 5.6, SA algoritması için tekrar tahsis edilen kullanıcı sayısının başlangıç ve son sıcaklık arası değişimlerden nasıl etkilendiğini göstermektedir. Tekrar tahsis edilen kullanıcı sayısı ve trafik yükündeki değişim ile başlangıç sıcaklığındaki değişim ters orantılı olarak değişmektedir. Dış döngünün herhangi bir adımında sistem için yakınsama koşulu sağlanabileceği gibi, belirtilen adım sayısı (1000 adım) sonunda da bu koşul tekrar kontrol edilebilir. Başlangıç sıcaklığı değeri arttıkça, adımlar arası mesafe artar.

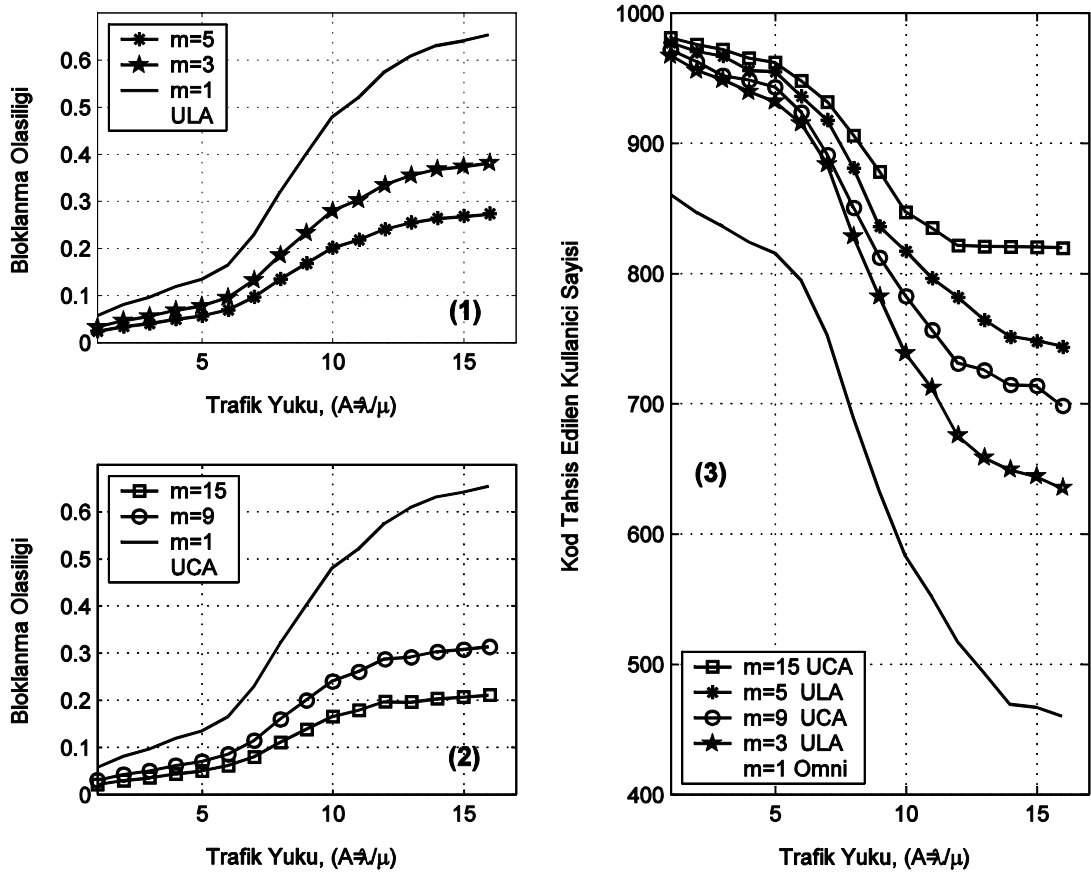


Şekil 5.6: SA algoritmasında tekrar tahsis edilen kullanıcıların başlangıç ve son sıcaklıklarına göre değişimi

5.4.7. Farklı AAA topolojilerinin etkisi

AAA-DCA algoritmasında kapasite artışının belirlenmesi amacı ile çoklu yol etkisi göz önünde bulundurularak, farklı anten dizi topolojileri ile sistem kapasitesi karşılaştırılmıştır. Tez çalışmasının bu kısmında, BTS’de farklı sayıda anten elemanı ve farklı anten topolojileri olduğunda, sistem performansının nasıl değiştiği sorusuna cevap aranmaktadır. Şekil 5.7’de, farklı AAA topolojilerine ait bloklanma olasılıkları ve OVFS kanallama kodu tahsis edilen kullanıcı sayıları, farklı anten topolojilerinde $\Delta\theta=20^\circ$ ve baskın DOA DL ışıma örüntüsü algoritması kullanılarak ele alınmıştır. Şekil 5.7 (1), ULA topolojisi için, 5.7 (2) ise UCA topolojisi için bloklanma olasılığını vermektedir. Şekil 5.7 (3)’te ise her iki topoloji için kod tahsis edilen kullanıcı sayısı, farklı trafik yüklerinde ve farklı sayıda anten elemanları ile incelenmiştir. Bu amaçla karşılaştırılan topolojiler, üç sektörlü ULA dizisi $m=3$ ve

$m=5$ anten elemanı, UCA dizisi $m=9$ ve $m=15$ anten elemanı ve $m=1$ yönsüz anten elemanıdır. Şekil 5.7'den açıkça görülmektedir ki ULA ve UCA topolojileri tekli yönsüz doğrusal anten sistemine göre çok daha fazla kapasite artışı sergilemektedir ve her iki topoloji için sistem kapasitesindeki artış anten dizisi eleman sayısı ile doğru orantılı olarak değişmektedir. Bunun yanı sıra, UCA anten dizisinin ULA ile karşılaştırıldığında daha fazla kullanıcıya kod tahsis ettiği görülmektedir.



Şekil 5.7: Farklı trafik yükleri ve farklı anten topolojilerinin etkisi

Şekil 5.7 (1) ULA topolojisinde, farklı anten elemanlarının sistem kapasitesine etkisi bloklanma olasılığı cinsinden ele alınmıştır. Şekilden de görülmektedir ki, uyarlamalı antenli sistemler yönsüz tek elemanlı anten yapılarına göre çok daha fazla kullanıcıya kod tahsis etmektedirler. Sayısal olarak ele aldığımızda trafik yükü 10 için, yönsüz anten elemanının bloklanma olasılığı % 36.9, $m=3$ elemanlı ULA anten dizisi için % 27.9, $m=5$ elemanlı ULA anten dizisi için % 20.1 değerindedir. Sayısal ifadeler karşılaştırıldığında, $m=3$ elemanlı ULA anten dizisinin yönsüz antene göre % 24.4,

$m=5$ elemanlı ULA anten dizisinin ise % 45.5 değerinde performans artışı sergilediği görülmektedir. Burada açık olarak ifade etmekte fayda vardır ki, $m=3$ ve $m=5$ elemanlı ULA anten dizilerinde anten elemanları arasındaki mesafe $\Delta=0.5\lambda$ ile belirlenmiştir.

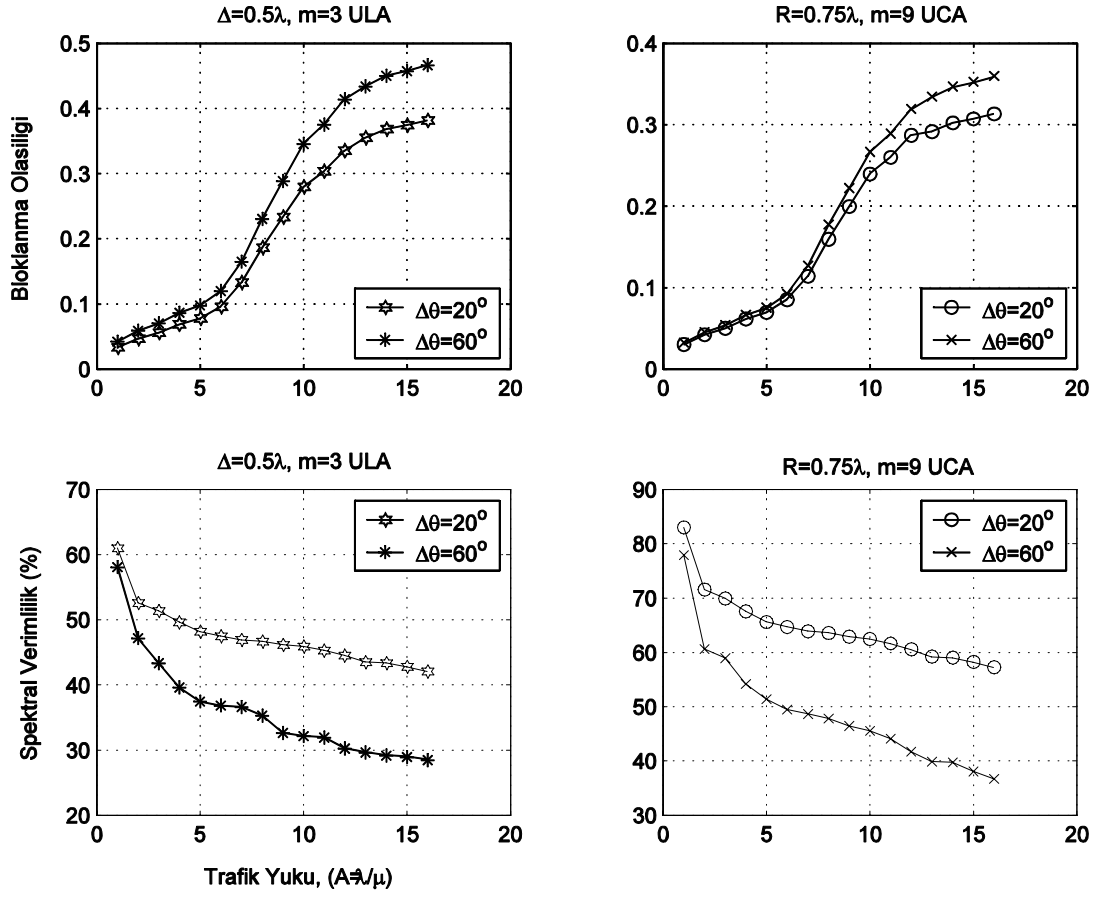
Şekil 5.7 (2)'de ise aynı durum UCA topolojisi için ele alınmıştır. Burada UCA topolojisinde yarıçap (R), geometrik olarak anten elemanları arasındaki mesafe $\Delta=0.5\lambda$ olacak şekilde ayarlanmıştır. Sistem performansının yönsüz anten elemanına göre artışı, ULA topolojisindeki artış gibi oldukça fazladır. Bununla birlikte anten elemanı sayılarındaki artış, sistem performansı ile doğru, bloklanma olasılığı ile ters orantılıdır. Aynı trafik yükü için sayısal değerleri ele aldığımızda, yönsüz anten elemanının bloklanma olasılığı % 36.9, $m=9$ elemanlı UCA anten dizisi için % 23.9, $m=15$ elemanlı UCA anten dizisi için % 16.5 değerindedir. $m=9$ elemanlı UCA anten dizisi için yarıçap $R=0.75\lambda$, $\Delta=0.5\lambda$ olarak, $m=15$ elemanlı UCA anten dizisi için yarıçap $R=\lambda$, $\Delta=0.5\lambda$ olarak belirlenmiştir. Bu durumda, $m=9$ elemanlı UCA anten dizisinin yönsüz antene göre % 35.2, $m=15$ elemanlı UCA anten dizisinin ise % 55.2 değerinde performans artışı sergilediği görülmektedir.

Şekil 5.7 (3)'te ise farklı trafik yüklerinde hem ULA hem de UCA topolojisi için kod tahsis edilen kullanıcı sayısı verilmiştir. Yarıçap ve anten elemanları arasındaki mesafe Şekil 5.7 (1) ve Şekil 5.7 (2) ile belirlenen hali ile ele alınmıştır. Şekilden açıkça görülmektedir ki trafik yükündeki artış, kod tahsis edilen kullanıcı sayısında bir azalmaya neden olmaktadır. Burada da bloklanma olasılıkları ile orantılı bir şekilde kod tahsis edilen kullanıcı sayısında değişim görülmektedir. Örneğin trafik yükü 10 iken, gelen 1000 adet kullanıcı çağrı talebi için kod tahsis edilen kullanıcı sayısı $m=15$ elemanlı UCA, $m=5$ elemanlı ULA, $m=9$ elemanlı UCA, $m=3$ elemanlı ULA anten dizileri ve $m=1$ elemanlı yönsüz anten elemanı için sırasıyla 847.1, 817.3, 782.4, 738.8, 582.9 ortalama değerleri civarındadır.

5.4.8. Açısal yayılımın etkisi

Açısal yayılımdaki değişimin sistem performansına etkisinin incelenmesindeki amaç, kullanılan algoritmanın çoklu yol sinyal etkileşimlerine karşı sağlamış olduğu faydanın test edilmesi yönündedir. Burada incelenen değişik açısal yayılımlara sahip sinyaller, 4.27 no'lu denklemde ifade edilen bit enerjisinin parazit ve gürültü toplamına oranında, farklı sonuçların elde edilmesine neden olacaktır. Sistem kapasitesi üzerinde çoklu yol açısal yayılım etkilerini incelemek için simülasyonlar $\Delta=0.5\lambda$ elemanlar arası mesafeye sahip 3–elemanlı 3–sektörlü ULA ve $R=0.75\lambda$ yarıçaplı ve elemanlar arası mesafesi $\Delta=0.5\lambda$ olan 9–elemanlı UCA topolojileri altında ve $\Delta\theta=20^\circ$ ile $\Delta\theta=60^\circ$ değişimleri temel alınarak incelenmiştir. ULA ve UCA için sonuçlar Şekil 5.8’de verilmektedir.

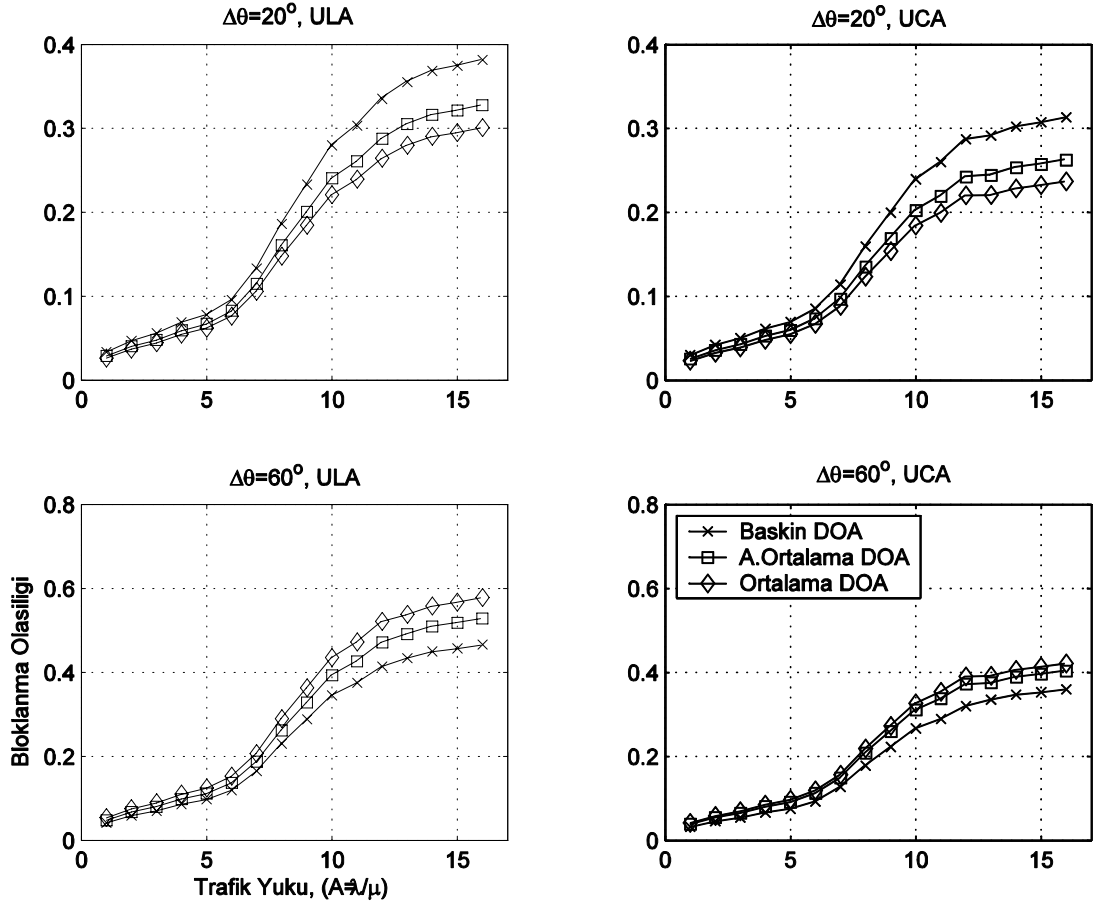
Şekil 5.8’de bulunan dört altşekilde sütunlar farklı topolojileri, satırlar ise bloklanma olasılığı ve spektral verimlilik cinsinden açısal yayılımın etkilerini göstermektedir. Bu amaçla, Şekil 5.8’in birinci sütunu $\Delta=0.5\lambda$ elemanlar arası mesafeye sahip 3–elemanlı 3–sektörlü ULA anten topolojisini, ikinci sütunu $R=0.75\lambda$ yarıçaplı ve elemanlar arası mesafesi $\Delta=0.5\lambda$ olan 9–elemanlı UCA anten topolojisini ele almaktadır. Yine aynı şeklin birinci satırı değişen açısal yayılımlar altında ULA ve UCA anten topolojileri için bloklanma olasılığını incelemektedir. İkinci satırda ise inceleme ölçütü spektral verimlilik olarak düzenlenmiştir. Yukarıdaki şekillerden açık olarak görülmektedir ki, tüm anten topolojilerinde açısal yayılımdaki artış, sistem performansını olumsuz yönde etkilemektedir. Bu ters orantılı ilişki, hem bloklanma olasılıklarında hem de spektral verimliliklerde kendini göstermektedir. Örneğin trafik yükü 10 iken, açısal yayılımdaki artış, bloklanma olasılığı ölçütü altında ULA topolojisi için % 19.07, UCA topolojisi için % 10.15 oranında performans düşüşüne neden olmuştur. Aynı trafik yükü altında spektral verimlilikteki değişim temel alındığında, performans düşüşü ULA topolojisi için % 29.9, UCA topolojisi için % 27.1 ile verilmektedir. Ayrıca ULA anten topolojisindeki performans değişiminin UCA topolojisine göre daha fazla olması, açısal yayılımın etkisinin ULA anten topolojisinde kendini daha fazla gösterdiğinin bir işaretidir.



Şekil 5.8: Farklı trafik yükleri ve açisal yayılımının etkisi

5.4.9. DL ışıma örüntüsü algoritmalarının etkisi

Bölüm 4.5'te DL ışıma örüntüsü algoritmaları hakkında detaylı bilgi verilmiştir. Burada temel amaç, DL boyunca oluşacak zayıflamanın etkisinin UL zayıflamalarına göre elde edilmesi ve bu yönde sinyalin güçlendirilmesidir. Bu doğrultuda Bölüm 4.5'te ele alınan çeşitli DL ışıma örüntüsü algoritmaları farklı trafik yükleri ve farklı açisal yayılımlar altında gerçekleştirilmiş ve Şekil 5.9'da sonuçlar ele alınmıştır.



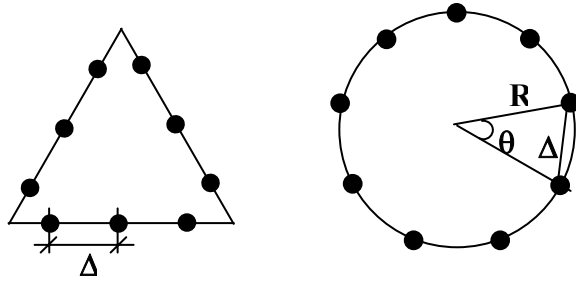
Şekil 5.9: Farklı trafik yükleri ve DL ışıma örüntüsü algoritmalarının etkisi

DL sırasında, UL boyunca elde edilen sinyal yayılım bilgilerine dayanarak her kullanıcı için, anten dizisinde ışıma örüntüsü ağırlık vektörünün tahminine ihtiyaç vardır. FDD esnasında UL ve DL frekansları farklı olduğundan zayıflamalar da farklıdır. Bu nedenle DL iletimi için UL boyunca elde edilen ışıma örüntüsü ağırlıklarının kullanımı doğru değildir (Kavak ve diğ., 2001). Bölüm 4.5'te ifade edilen algoritmaların sistem performansına olan etkisi bloklanma olasılığı cinsinden ele alınmıştır. Burada, açısız yayılımın seçilen algoritmanın performansına olan etkisi açıkça görülmektedir. Düşük açısız yayılım olan yerlerde yani trafik yoğunluğunun çok olmadığı kırsal alanlarda çoklu yollardan kaynaklanan açısız yayılım küçük olacağından, burada ortalama DOA algoritması ile daha iyi sistem performansı elde edilmektedir. Diğer durumda, yani trafik yoğunluğu fazla olan şehir merkezi gibi yerlerde, çoklu yollardan kaynaklanan açısız yayılım büyük olacağından burada baskın DOA algoritmasının kullanılması sistem performansı cinsinden olumlu bir adım olarak değerlendirilebilir. Ayrıca, daha önceki simülasyon

adımlarında ele alınan trafik yükü ile bloklanma olasılığının orantılı olarak artışı, aynı koşullar altında UCA topolojisinin ULA topolojisinden daha yüksek sistem performansı sergilemesi ve açılal yayılım ile sistem performansı arasındaki ters orantı burada da kendisini göstermektedir.

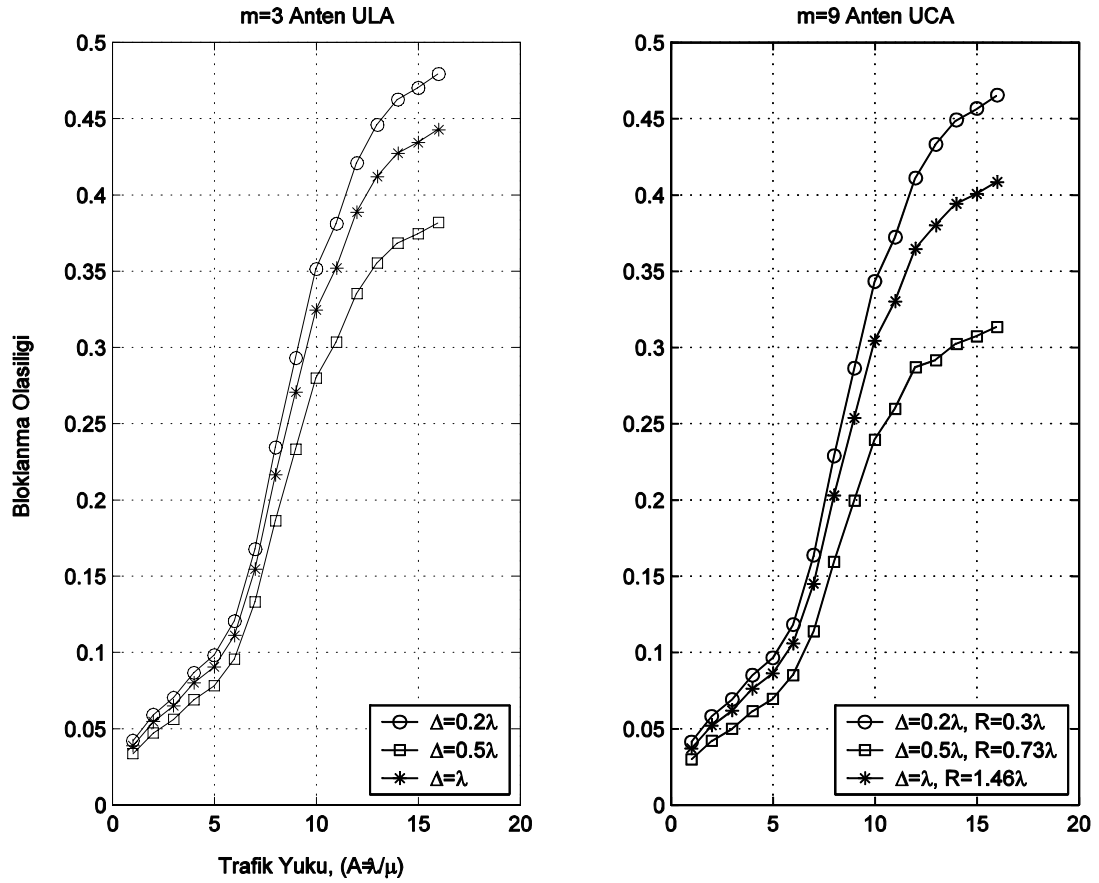
5.4.10. Anten elemanları arasındaki mesafenin etkisi

Şekil 5.10’da gösterildiği gibi anten elemanları arasındaki mesafe, anten topolojileri için önemli bir parametredir. Bu mesafenin optimizasyonu sistem performansını etkiler (Chalize ve diğ., 2007). Bu amaçla, farklı elemanlar arası uzaklıklar (Δ) ve yarıçap (R) değerleri Şekil 5.11’de karşılaştırılmıştır.



Şekil 5.10: ULA ve UCA topolojileri için anten elemanları arası mesafe

Şekil 5.11, 3–elemanlı ULA ve 9–elemanlı UCA anten topolojileri için farklı Δ bilgileri ışığında sistem performansını sergilemektedir. Görülmektedir ki istenilir bir sistem performansının sağlanması için, Δ civarında $\lambda/2=7.93$ cm. olmalıdır. Δ değerinin artması, sistem performansını olumsuz yönde etkilemektedir. UCA anten topolojisinin elemanlar arası mesafesi, ULA’ya eşit olacak şekilde UCA’nın yarıçapı ayarlanmıştır. UCA için optimum sistem performansı $R=11.6$ cm.’de elde edilmiştir ki bu değer de $\Delta=\lambda/2$ değerine eşittir. Şekillerin ve yukarıdaki bilgilerin ışığında, optimum Δ seviyesinin $\lambda/2$ civarında olduğu gözlemlenmiştir.



Şekil 5.11: Farklı trafik yükleri ve anten elemanları arasındaki mesafenin etkisi

5.5. Maliyet Analizi

Maliyet analizinde, kod sınırlı yapıdaki algoritmalar ele alınmıştır. Çünkü kod sınırlı yapıdaki algoritmalar için temel sınırlayıcı etken OVFS kod ağacının kendisidir. SIR sınırlayıcı yapı için bunu söylemek mümkün değildir. Ayrıca kod sınırlı algoritmalar iteratif yapıdadır. Dolayısıyla maliyet analizi bölümünde SIR sınırlı yapı incelenmemiştir. İncelenen CCA, DCA, SGA, D&D-GA, SA ve ASAGA algoritmaları için, her iterasyondaki toplama ve çarpma sayıları hesaplanmıştır. Her ne kadar maliyet analizi zamana bağlı sistemlerin hızı için bize fikir verse de tüm sistemler için CPU yükü ve hızı da önemli bir bileşendir. Her bir algoritma için elde edilen maliyet analizi; yayılım faktörü (SF), kullanıcı sayısı (U), yeni gelen kullanıcı veri transfer hızı ($S(n)$), popülasyon büyüklüğü (n) ve çarpazlamaya giren kullanıcı sayısı (j) cinsinden ele alınmıştır.

$$CCA = \sigma(SF, S(n))/2 + S(n) + U + 2 \quad (5.4)$$

$$DCA = \sigma(SF, S(n)) + S(i) + U + 2 \quad (5.5)$$

$$SGA = \sigma(SF, S(n)) + 2U + (nU + 2)^2 + j(U + 1) \quad (5.6)$$

$$D \& D - GA = \sigma(SF, S(n)) + 2U + (nU + 2)^2 + n(U + 7) + 2j(U + 1) + 2 \quad (5.7)$$

$$SA = \sigma(SF, S(n)) + 6SF(\log_2 SF + 1) + 6SF/S(n) + 2U + 10 \quad (5.8)$$

$$ASAGA = \sigma(SF, S(n)) + 5SF(\log_2 SF + 1) + 5SF/S(n) + 7U + (nU + 2)^2 + 4n + j(U + 1) + 10 \quad (5.9)$$

Tüm bu algoritmalar göz önünde bulundurulduğunda, ASAGA algoritmasının hesaplama yükünün diğerlerine göre ne kadar fazla olduğu görülmektedir. SA algoritması için hesaplama yükü D&D-GA algoritması ile yaklaşık aynı sayısal değeri vermektedir. SA ile D&D-GA karşılaştırmasında yaptığımız gibi sayısal değerler vererek algoritmaların birbirlerine göre durumları hakkında daha net yargıya varılabilir. Ancak mevcut trafiğin değişken olması, dolayısıyla kullanıcı sayısı ve bu kullanıcıların veri transfer hızlarının da değişken olması, verimli bir şekilde sayısal olarak hesaplama yoğunluğunun irdelenmesini engellemektedir. $\sigma(SF, U)$ ile ifade edilen basitleştirme ile

$$\sigma(SF, S(n)) = 2S(n) + 2SF(\log_2 SF + 1) + 2SF/S(n) \quad (5.10)$$

basit (primitive) değeri elde edilir. Hesaplama karmaşıklığına ilişkin detaylı çalışma Ek-C’de mevcuttur. ASAGA, SA, D&D-GA ve SGA algoritmaları tekrarlamalı yapıda olduklarından ve optimizasyon koşulu sağlanana kadar algoritma çalışacağından CCA ve DCA algoritmalarına göre çok daha fazla karmaşıklık getireceği açıktır. Tekrarlamalı yapıdaki algoritmalar yapı gereği en iyi sonucu vermeyi garanti etmeseler de algoritmayı bu yönde çalıştırırlar.

6. SONUÇLAR VE ÖNERİLER

Kablosuz haberleşme sistemlerine tahsis edilen bant genişliği çok değerli bir kaynaktır. Bu nedenle, onun en verimli bir şekilde kullanılması önemli bir sorundur. 3G gibi yüksek veri transfer hızını ihtiyaç duyulan servis kalitesinde sunmak için kaynak yönetiminin optimum şekilde yapılması gerekmektedir. OVSF kod ağacının yönetimi ve optimizasyonu bahsedilen sorunlara çözüm olma yolunda önemli bir aşamadır ve bu tez çalışmasının odağını teşkil eder. Yapılan bu çalışmanın çıktıları sistem performansına etkisi, algoritmik parametrelerin etkisi ve maliyet analizi etkisi gibi tanımlamalar ışığında aşağıda özetlenmiştir:

Sistem performansının etkisini ele aldığımızda;

- Genel olarak tüm algoritmalarda sistem performansı artan trafik yüklerine karşın ters orantılı olarak değişim göstermektedir.
- Kod sınırlı ve SIR sınırlı algoritmaları karşılaştırdığımızda, SIR sınırlı yapının daha az bloklanma olasılığına ve daha fazla spektral verimliliğe sahip olduğu görülmüştür.

Algoritmik parametrelerin etkisini ele aldığımızda;

- Kod sınırlı yapının parametreleri karşılaştırıldığında sistem performansı çaprazlama oranındaki artış ile doğru orantılı, mutasyon oranındaki artış ile ters orantılıdır. Ancak bu iki değişken de bir noktada doyuma ulaşmaktadır. Dolayısıyla en iyi performansı sergileyen kod sınırlı OVSF kod tahsis algoritmasının $p_c=1$ ve $p_m=0$ için elde edildiğini söylemek doğru olmaz.
- Genetik algoritma tabanlı seçim tekniklerinden turnuva metodu diğer seçim tekniklerine göre, aritmetik çaprazlama tekniği de diğer çaprazlama tekniklerine göre ve öteleme mutasyon tekniği yer değiştirme mutasyon tekniğine göre daha iyi performans sergilemektedir.

- Benzetilmiş tavlama için öteleme komşuluk hareketi daha iyi performans sergilemektedir.
- Soğutma çizelgesi olarak Lundy & Mees (L&M), orantısız azalımlı soğutma çizelgesine göre daha iyi performans sergilemektedir. Trafik yükü arttıkça soğutma çizelgesinde yakınsama da o oranda azalmaktadır. İç döngü sayısındaki artış olası daha fazla çözümün ele alınmasına olanak tanıyacağından iç döngü sayısındaki artış sistem performansını olumlu yönde etkiler. Başlangıç ve son sıcaklıklar arasındaki bant azaldıkça arama adımları azalacak ve dolayısıyla sistem performansı artacaktır.
- SIR sınırlı algoritmalar için çoklu yollar önemli bir bileşendir. İdeal, yani çoklu yolların etkisinin ihmal edildiği durumda sistem performansı, çoklu yol etkisinin göz önünde bulundurulduğu duruma göre daha fazla kullanıcıya aynı OVFS kanallama kodunu kullanma imkanı tanır.
- UCA topolojisi, ULA topolojisine göre daha iyi performans artışı sergiler. Ayrıca anten sayısındaki artış da sistem performansını olumlu olarak etkiler. Açısal yayılımdaki artış çoklu yolların çözünürlüğünü zorlaştıracığından sistem performansı ile açısal yayılım ters orantılıdır.
- DL ışıma örüntüsü algoritmaları da çoklu yol yayılımına ve AAA topolojisine göre değişkenlik gösterir. Düşük açısal yayılımda baskın DOA, büyük açısal yayılımda ortalama DOA DL ışıma örüntüsü algoritması daha iyi algoritmik performans sergiler.
- Antenler arasındaki mesafe ULA için $\Delta=0.2\lambda$, UCA için $R=0.3\lambda$ en optimum değerlerdir.

Maliyet analizi etkisini ele aldığımızda;

- Kod sınırlı algoritmaları karşılaştırdığımızda ASAGA algoritmasının hesaplama yoğunluğunun diğerlerine göre daha fazla olduğu görülmektedir. ASAGA algoritmasını sırasıyla SA, D&D-GA ve SGA algoritmaları izlemektedir

Çalışmanın devamında MIMO sistemlerinde OVFS kod tahsis yönteminin uyarlanması yapılabilir. Çoklu anten elemanlarından her bir dala adanan farklı OVFS kodlarının sisteme sağladığı performans değişimi incelenebilir.

Ayrıca hesaplama yoğunluğunu daha aza indiren optimum OVSF kod tahsis yöntemleri incelenebilir. Sistemde çağrı gelişi Poisson, çağrının hatta kalma süresi Exponential olarak modellendiği için gerçeğe yakın bir kablosuz haberleşme ortamı oluşturulmuştur. Dolayısıyla hem kod sınırlı algoritmaların hem de SIR sınırlı çoklu yol etkisinin göz önünde bulundurulduğu algoritmanın performansı gerçek değerler içermektedir. Ayrıca sunulan kod sınırlı algoritma literatürde performans artırıcı çalışmalar olarak yerini alırken, SIR sınırlı algoritma literatüre yeni bir kavram olarak ele alınmıştır

KAYNAKLAR

3GPP2 C.S0024-B, "High Rate Packet Data Air Interface Specifications", **3rd Generation Partnership Project 2**, 779-782, (2006)

Alouini, M., S., and Simon, M., K., "Performance analysis of coherent equal gain combining over Nakagami-m fading channels", **IEEE Transactions on Vehicular Technologies**, 50(6), 1449–1463, (2001).

Assarut, R., Kawanishi, K., Yamamoto, U., Onozato, Y., and Matsushita, M., "Region Division Assignment of Orthogonal Variable Spreading Factor Codes in WCDMA", **IEEE Vehicular Technology Conference, IEEE VTC'01**, 3, 1885-1888, (2001).

Assarut, R., Kawanishi, K., Yamamoto, U., and Onozato, Y., "Performance Evaluation of Orthogonal Variable Spreading Factor Code Assignment Schemes Based on UMTS/IMT-2000 in WCDMA", **IEEE 56th Vehicular Technology Conference, VTC'02**, 3, 1585-1589, (2002).

Back, T., "Evolutionary Algorithms in Theory and Practice", **Oxford University Press, New York**, 19-24, (1996).

Baker, J., E., "Adaptive Selection Methods for Genetic Algorithms," **1st Int. Conference on Genetic Algorithms. J. J. Grefenstette, Ed., Lawrence Erlbaum Associates**, 101–110, (1985).

Bingül, Z., Sekmen, A., S., and Zein, S., "Genetic Algorithms Applied to Real Time Multi-objective Optimization Problems", **IEEE SoutheastCon 2000 Conference**, Nashville, (2000).

Chalise, B., K., Shahram, S., Czylik, A., Gershman, A., B., "Robust Downlink Beamforming Based on Outage Probability Specifications", **IEEE Transactions on Wireless Communications**, 6(10), 3498-3503, (2007).

Chao, C., Tseng, Y., and Wang, L., "Reducing Internal and External Fragmentations of OVSF Codes in WCDMA Systems with Multiple Codes", **IEEE Transactions on Wireless Communications**, 4 (4), 1516-1526, (2005)

Chaudry, P., Mohr, M., and Onoe, S., "The 3GPP Proposal for IMT-2000," **IEEE Communications Magazine**, 39 (12), 72-81, (1999).

Chen, M., and Hwang, R., “Efficient OVSF Code Assignment Strategy for UMTS”, **IEEE 60th Vehicular Technology Conference, IEEE VTC’04**, 7, 5185-5189, (2004).

Chen, M., and Hwang, R., “Efficient OVSF Code Assignment and Reassignment Strategies in UMTS”, **IEEE Transactions on Mobile Computing**, 5(7), 769-783, (2006).

Chen, C., and Chang, Y., “Multi-code Placement and Replacement Schemes for WCDMA Rotated OVSF Code Tree”, **IEEE 6th CAS Symp. on Emerging Technologies: Mobile and Wireless Comm.**, 345-348, May 31-June 2, (2004).

Chen, Y., and Lin, T., “Code Placement and Replacement Strategies for WCDMA Rotated OVSF Code Tree Management”, **IEEE Transactions on Mobile Computing**, 5 (3), 224-239, (2006).

Dahlman, E., Gudmundson, B., Nilsson, M., and Skold, J., “UMTS/IMT-2000 Based on Wideband CDMA”, **IEEE Communications Magazine**, 70-80, (1998).

Dell’Amico, M., Maffioli, F., and Merani, M., “A Tree Partitioning Dynamic Policy for OVSF Code Assignment in Wideband CDMA”, **IEEE Transactions on Wireless Communications**, 3(4), 1013-1017, (2004).

Eberspacher, J., Vogel, H., and Bettstetter, C., “GSM, Switching, Services and Protocols”, **John Wiley & Sons Ltd.**, 181-201, (2001).

Eshelman, L., J., Caruna, R., and Schaffer, J., Lin, D., “Biases in the crossover landscape”, **3rd International Conference on Genetic Algorithms**, 10–19, (1989).

ETSI TS 125 201, Universal Mobile Telecommunications System (UMTS); Physical layer - general description, **3GPP TS 25.201 version 7.5.0 Release 7**, (2008).

ETSI TS 125 211, Universal Mobile Telecommunications System (UMTS); Physical channels and mapping of transport channels onto physical channels (FDD), **3GPP TS 25.211 version 7.5.0 Release 7**, (2008).

ETSI TS 125 212, Universal Mobile Telecommunications System (UMTS); Multiplexing and channel coding (FDD), **3GPP TS 25.212 version 7.7.0 Release 7**, (2008).

ETSI TS 125 213, Universal Mobile Telecommunications System (UMTS); Spreading and modulation (FDD), **3GPP TS 25.213 version 7.4.0 Release 7**, (2008).

ETSI TS 125 214, Universal Mobile Telecommunications System (UMTS); Physical layer procedures (FDD), **3GPP TS 25.214 version 7.8.0 Release 7**, (2008).

ETSI TS 125 215, Universal Mobile Telecommunications System (UMTS); Physical layer; Measurements (FDD), **3GPP TS 25.215 version 7.4.0 Release 7**, (2008).

Evans, J., S., and Everitt, D., "Effective Bandwidth-Based Admission Control for Multiservice CDMA Cellular Networks", **IEEE Transactions on Vehicular Technologies**, 48(1), 36–46, (1999).

Ferng, H., Chin, H., Shiung, D., and Chen, Y., "An OVSF Code Tree Partition Policy for WCDMA Systems Based on the Multi-Code Approach", **IEEE 62nd Vehicular Technology Conference, IEEE VTC'05**, 2, 1212-1216, (2005).

Global Mobile Market Q4 Report, 1-2, (2007) web adresi: (<http://www.gsmtwenty.com/20facts.pdf>) 07.05.2008

Godara, L., C., "Handbook of Antennas in Wireless Communications", **CRC Press**, 31-35, (2002).

Goldberg, D., E., "Genetic Algorithms in Search, Optimization and Machine Learning", **Addison-Wesley Publishing Company Inc. ISBN:0-201-15767-5**, (1989).

Gong, D., Yan, Y., and Lu, J., "Dynamic Code Assignment for OVSF Code System", **IEEE Global Telecommunications Conference, IEEE Globecom'05**, 5, 2865-2869, (2005)

Holma, H., and Toskala, A., WCDMA for UMTS, **John Wiley & Sons, LTD.**, 73-119, (2000).

Holma, H., Toksala, A., "HSDPA/HSUPA for UMTS", **John Wiley & Sons, LTD.**, 95-120, (2006).

Hondroudakis, A., Malard, J., and Wilson G. V., "An Introduction to Genetic Algorithms Using RPL2 [online]", http://www.epcc.ed.ac.uk/computing/training/document_archive/GAS-course/main.html, (**Ziyaret Tarihi: 10 Ekim 2006**).

Hochwald, B., M., and Marzetta, T., L., "Adapting a downlink array from uplink measurements", **IEEE Transaction Signal Processing**, 49(3), 642-653, (2001).

Hou, J., and Lee, M., "Adaptive OVSF Design in WCDMA System", **IEEE Eighth International Symposium on Spread Spectrum Techniques and Applications, IEEE ISSSTA**, 835-838, (2005).

Jungnickel, V., Chan, Y., and Pohl, V., "Performance of MIMO Rake Receivers in WCDMA Systems", **IEEE Wireless Communications and Networking Conference, IEEE WCNC'04**, 4, 2075-2080, (2004).

Kaizhi, H., Jing, W., and Youzheng, W., "Performance analysis of equal gain combining 2D-RAKE receiver in Nakagami-fading channels", **The 57th IEEE Semiannual on Vehicular Technology Conference, VTC'03**, 4, 2579-2583, (2003).

Karakoç, M., Söke, A., and Kavak, A., "Dynamic OVFS Code Allocation in WCDMA Networks Using Dominance and Diploidy Genetic Algorithm," **IEEE Radio and Wireless Symposium, RAWCON'07**, Long Beach USA, (2007).

Karakoç, M., and Kavak, A., "A New Dynamic OVFS Code Allocation Method Based on Adaptive Simulated Annealing Genetic Algorithm (ASAGA)," **The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC'07**, (2007).

Karakoç, M., and Kavak, A., "Performance of Adaptive Antenna Array Based Dynamic OVFS Code Assignment Under Multipath Propagation and Multicell Environment," **Journal of Frequenz**, 11-12, 254-257, (2007).

Kavak, A., "Adaptive antenna arrays for downlink capacity increase in third generation wireless CDMA," **IEEE Radio and Wireless Symposium, RAWCON'01**, 77-80, (2001).

Kavak, A., Yang, W., Xu, G., Vogel, W., J., "Characteristics of vector propagation channels in dynamic mobile scenarios", **IEEE Transaction Antennas Propagation**, 49(12), 1695-1703, (2001).

Kirkpatrick, S., Gelatt Jr., C., D., and Vecchi, M., P., "Optimization by Simulated Annealing", **Science**, 220, 671-680, (1983).

Lai, K., K., Chan, J., W., M., "Developing A Simulated Annealing Algorithm for The Cutting Stock Problem", **Computers and Industrial Engineering**, 32, 115-127, (1997).

Leung, T. W., Yung, C., H., Troutt, M., D., "Applications of Genetic search and simulated annealing to the two-dimensional non-guillotine cutting stock problem", **Computers and Industrial Engineering**, 40, 201-214, (2001).

Leung, T., W., Chan, C., K., and Marvin, D., T., "Application of a mixed simulated annealing-genetic algorithm heuristic for the two-dimensional orthogonal packing problem", **European Journal of Operational Research**, 145, 530-542, (2003).

Li, Y., Tan, K., C., Chwee, K., and Murray-Smith, D. J., "Performance Based Linear Control System Design by Genetic Evolution with Simulated annealing", **34th IEEE conference on Decision and Control**, 731-736, (1995).

Liang, Y., C., and Chin, F., P., S., "Downlink Channel Covariance Matrix (DCCM) Estimation and its Application in Wireless DS-CDMA Systems", **IEEE Journal on Selected Areas in Communications**, 19, 222-232, (2001).

Lutfiyya, H., Mcmillin, B., Poshyanonda, P., and Daglı, C., "Composite Stock Cutting Through Simulated Annealing", **Mathematical Computing Modelling**, 16(1), 57-74, (1992).

Masri, S., F., Smith, A., W., Chassiakkos, A., G., Nakamura, M., and Cauhghey, T., K., "Training Neural Networks by Adaptive Random Search Techniques", ***Journal of Engineering Mechanics***, 125(2), 123-132, (1999).

Minn, T., and Siu, K., "Dynamic Assignment of Orthogonal Variable Spreading Codes in WCDMA", ***IEEE Journal on Selected Areas in Communications***, 18 (8), 1429-1440, (2000).

Molisch, A., F., "Wireless Communications", ***John Wiley & Sons, LTD.***, 4-5, (2005).

Naguib, A., F., Paulraj, A., and Kailath, T., "Capacity improvement with base station antenna arrays in cellular CDMA", ***IEEE Transactions on Vehicular Technologies***, 43(3), 691–698, (1994).

Nathan, J., M., "Wireless A to Z", ***McGraw-Hill Companies***, 48-55, (2003).

Okawa, K., and Adachi, F., "Orthogonal forward link using orthogonal multisreading factor codes for coherent DS-CDMA mobile radio," ***IEICE Transactions on Communications***, E81-B, 777-784, (1998).

Peng, W., Yu, Z., and Dazhong, C., "Efficient Algorithm of OVSF Code Allocation for WCDMA", ***IEEE Canadian Conference on Electrical and Computer Engineering, IEEE CCECE/CCGEI'05***, 2109-2112, (2005).

Perez, F., Vazquez-Avila, J., Jimenez, A., and Guerrero, L. " Call Admission and Code Allocation Strategies for WCDMA Systems with Multirate Traffic", ***IEEE Journal on Selected Areas in Communications***, 24 (1), 26-35, (2006).

Pham, D. T. Karaboğa, D., "Intelligent Optimisation Techniques", ***Springer - Verlag London***, ISBN:1852330287 (2000).

Prasad, R., and Ruggieri, M., "Technology Trends in Wireless Communications", ***Artech House***, 4-7, (2003).

Rappaport, T., S., and Liberti, J., C., "Smart Antennas for Wireless Communications", ***Prentice Hall: New-Jersey***, (1999).

Rouskas, A., N., and Skoutas, D., N., "OVSF Code Assignment and Reassignment at the Forward Link of WCDMA 3G Systems", ***IEEE Ann. Int'l Symp. Personal Indoor and Mobile Radio Comm., IEEE PIMRC'02***, 2024-2048, (2002).

Rouskas, A., and Skoutas, D., "Management of Channelization codes at the Forward Link of WCDMA", ***IEEE Communications Letters***, 9 (8), 679-681, (2005).

Sekine, Y., Kawanishi, K., Yamamoto, U., and Onozato, Y., "Hybrid OVSF Code Assignment Scheme for WCDMA", ***IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, IEEE, PACRIM'03***, 1, 384-387, (2003).

Shahookar, K. Mazumder, P., “VLSI Cell Placement Techniques”, **ACM Computing Surveys**, 23(2), 144-172, (1991).

Shueh, F., and Chen, W., “Minimizing WCDMA Channel Reallocation”, **14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications, IEEE PIMRC’03**, 41-45, (2003).

Steele, R., Lee, C., C., and Gould, P., “GSM, cdmaOne and 3G systems”, **John Wiley & Sons, LTD.**, 65-148, (2001).

Tan, K. C., Li, Y., Murray-Smith, D. C. Sharman, K. C., “System Identification and Linearisation Using Genetic Algorithms with Simulated Annealing”, **1st IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications**, 164-169, (1995).

Torrieri, D., J., “Performance of direct sequence systems with long pseduonoise sequences”, **IEEE Journal on Selected Areas in Communications**, 10(4), 770–781, (1992).

Tsai, Y. And Lin, L., “Quality Based OVSF Code Assignment and Reassignment Strategies for WCDMA Systems”, **IEEE International Conference on Wireless Networks, Communications and Mobile Computing**, 238-243, (2005).

Tseng, Y., and Chao. C., Yu. S., “Code Placement and Replacement strategies for Wideband CDMA OVSF Code Tree Management”, **IEEE Global Telecommunications Conference, IEEE Globecom’01**, 1, 562-566 (2001)

Tseng, Y., and Chao. C., “Code Placement and Replacement Strategies for Wideband CDMA OVSF Code Tree Management”, **IEEE Transactions on Mobile Computing**, 1 (4), 293-302, (2002).

Yu, H., Cho, H., Kang, C., and Hong, D., “A New Multipath Interference Mitigation Technique for High Speed Packet Transmission in WCDMA Downlink”, **IEEE Signal Processing Letters**, 12 (9), 601-604, (2005).

Walke, B., Seidenberg, P., and Althoff, M., P., “UMTS, The fundamentals”, **John Wiley & Sons, LTD.**, 28-33, (2003).

Weile, D., S., and Michielssen, E., “The Control of Adaptive Antenna Arrays with Genetic Algorithms using Dominance and Diploidy”, **IEEE Transactions on Antennas and Propagation**, 49(10), 1424–1433, (2001).

Zigangirov, K., “Theory of Code Division Multiple Access Communication”, **IEEE Series on Digital & Mobile Communication**, 377-380, (2004).

EKLER

EK-A

KOD SINIRLI ALGORİTMALARIN SÖZDE KODU

Bu ekte kod sınırlı yapıda bulunan GA, SA ve ASAGA algoritmalarına ait sözde kodlar verilmektedir. Sözde kod, algoritmanın programa dönüşmesinde temel analizi teşkil etmektedir.

1. Genetik Algoritma

```
1: procedure GMain(agac, mutation_rate, crossover_rate, requested_data_bit_rate)
2: new_population[ ]  $\leftarrow$  population(agac) /*Initializing population*/
3: cost  $\leftarrow$  fitness_function(agac, new_population)
4: (current_best_fitness_value, current_best_chromosome )  $\leftarrow$  min(cost)
5: probable_solution  $\leftarrow$  current_best_chromosome
6: solution  $\leftarrow$  probable_solution
7: previous_fitness_value  $\leftarrow$  current_best_fitness_value
8: new_agac [ ]  $\leftarrow$  matrix_to_agac(solution)
9: (success, fail)  $\leftarrow$  code_assigner(new_agac, requested_data_bit_rate)
10: if success
11: /*Assign requested data rate using new code tree*/
12: else
13: /*Start to GA Loop*/
14: while (fail) do
15: iteration+1
16: selected_population [ ]  $\leftarrow$  selection(new_population,cost)
17: recombined_population [ ]  $\leftarrow$  crossover(selected_population, crossover_rate)
18: mutated_population [ ]  $\leftarrow$  mutation(recombined_population, mutation_rate)
19: cost  $\leftarrow$  fitness_function(new_population, mutated_population)
20: (current_best_fitness_value, current_best_chromosome )  $\leftarrow$  min(cost)
21: probable_solution  $\leftarrow$  current_best_chromosome
22: if current_best_fitness_value  $\geq$  previous_fitness_value
23: previous_fitness_value  $\leftarrow$  current_best_fitness_value
24: solution  $\leftarrow$  probable_solution
25: new_agac [ ]  $\leftarrow$  matrix_to_agac(solution)
26: (success, fail)  $\leftarrow$  code_assigner(new_agac, requested_data_bit_rate)
27: if success
28: /*Assign requested data rate using new code tree*/
29: end /*if success*/
30: end /*if current_best_fitness_value  $\geq$  previous_fitness_value*/
31: if iteration == 100 && fail
32: break
33: /*Requested data bit rate is not assigned*/
34: end /*if iteration == 100 && fail*/
35: end /*while (fail) do*/
36: end /*if success*/
37: end /*procedure GMain(agac, mutation_rate, crossover_rate, requested_data_bit_rate)*/
```

Şekil A.1: GA algoritmasının sözde kodu

2. Benzetilmiş Tavlama

```
1: Procedure SA main (agac, requested_data_bit_rate)
2: [agac_indis, agac_hiz, cozum_indis, cozum_hiz]  $\leftarrow$  population(agac)
3: maliyet  $\leftarrow$  fitness_function(cozum_indis, konusma_indis, cozum_hiz,)
4: mevcut_uygunluk_degeri  $\leftarrow$  maliyet
5: onceki_uygunluk_degeri  $\leftarrow$  mevcut_uygunluk_degeri
6: olasi_cozum  $\leftarrow$  cozum_indis
7: cozum  $\leftarrow$  olasi_cozum
8: /* SA döngüsüne başla*/
9: beta  $\leftarrow$  (itemp - ftemp) / (max_iteration * itemp * ftemp)
10: temp  $\leftarrow$  itemp /*Baslangic Sicakligi */
11: while ( disdongu < 1000 ) do
12: iteration+1
13: while ( denge == 0 ) do
14: dengekosulu = 5
15: [yeni_cozum_indis, yeni_cozum_hiz]  $\leftarrow$  swap_neighbour(cozum_indis, cozum_hiz)
16: yeni_maliyet  $\leftarrow$  fitness_function (yeni_cozum_indis, yeni_cozum_hiz)
17: delta  $\leftarrow$  yeni_maliyet – maliyet
18: if (delta < 0)
19: cozum_indis  $\leftarrow$  yeni_cozum_indis
20: maliyet  $\leftarrow$  yeni_maliyet
21: else
22: olasilik  $\leftarrow$  exp(-delta / temp)
23: if (rand < olasilik)
24: cozum_indis  $\leftarrow$  yeni_cozum_indis
25: maliyet  $\leftarrow$  yeni_maliyet
26: end /* if (rand < olasilik) */
27: end /*if (delta < 0)*/
28: komsusayi+1
29: if ( komsusayi == dengekosulu )
30: denge $\leftarrow$ 1
31: end /* if ( komsusayi == dengekosulu) */
32: end /* while ( denge == 0)*/
33: olasi_cozum $\leftarrow$ cozum_indis
34: temp  $\leftarrow$  temp / (1+beta * temp) /* Lundy & Mees ile Sicakligin Guncellenmesi */
35: guncel_uygunluk_degeri ( iterasyon )  $\leftarrow$  maliyet
36: if ( guncel_uygunluk_degeri ( iterasyon ) < onceki_uygunluk_degeri )
37: onceki_uygunluk_degeri  $\leftarrow$  guncel_uygunluk_degeri ( iterasyon )
38: uygunluk ( iteration )  $\leftarrow$  onceki_uygunluk_degeri
39: cozum  $\leftarrow$  olasi_cozum
40: [ yeni_agac ]  $\leftarrow$  matrix_to_agac( cozum )
41: else
42: uygunluk(iterasyon)  $\leftarrow$ previous_fitness_value
43: [ yeni_agac ]  $\leftarrow$  matrix_to_agac( cozum )
44: end /*if (guncel_uygunluk_degeri ( iterasyon ) < onceki_uygunluk_degeri)*/
45: [ yeni_agac ]  $\leftarrow$  matrix_to_agac( olasi_cozum )
46: (success, fail)  $\leftarrow$  code_assigner( yeni_agac, gelen_yeni_veri_iletim_hızı )
47: if success
48: /*Gelen yeni data transfer hızını yeni ağaçta belirtilen uygun yere ata*/
49: disdongu = 1000
50: end /* if succewss*/
51: if iteration == disdongu
52: disdongu = 1000
53: end /* iteration == disdongu */
54: end /*while ( disdongu < 1000 ) do*/
```

Şekil A.2: SA algoritmasının sözde kodu

3. Uyarlamalı Benzetilmiş Tavlamalı Genetik Algoritma

```
1: Procedure ASAGA main(agac, mutation_rate, crossover_rate, requested_data_bit_rate)
2: /* Başlangıç Popülasyonunun Üretilmesi */
3: [agac_indis, agac_hız, çözüm_indis, çözüm_hız]  $\leftarrow$  population(agac)
4: maliyet  $\leftarrow$  fitness_function(cozum_indis, konuşma_indis, çözüm_hız,)
5: [güncel_en_ iyi_uygunluk, güncel_en_ iyi_kromozom]  $\leftarrow$  min(maliyet)
6: olası_çözüm  $\leftarrow$  güncel_en_ iyi_kromozom
7: önceki_uygunluk  $\leftarrow$  güncel_en_ iyi_uygunluk
8: [yeni_agac]  $\leftarrow$  matris_agac(olası_çözüm)
9: (success, fail)  $\leftarrow$  code_assigner( yeni_agac, yeni_veri_iletim_hızı )
10: if ( success )
11: /* Gelen yeni data transfer hızını yeni ağaçta belirtilen uygun yere ata */
12: else
13: /* GA döngüsüne başla */
14: while ( dışdöngü < 1000 ) do
15: iterasyon+1
16: [seçilen_popülasyon]  $\leftarrow$  seçim (yeni_agac, maliyet)
17: [çaprazlanan_popülasyon, çaprazlanan_hız]  $\leftarrow$  çaprazlama (seçilen_popülasyon, çaprazlama_oranı)
18: [çaprazlama_maliyeti]  $\leftarrow$  fitness_function (çaprazlanan_popülasyon, agac_indis, çaprazlanan_hız)
19: temp  $\leftarrow$  itemp /*Başlangıç Sicaklığı */
20: beta  $\leftarrow$  (itemp - ftemp) / (max_iterasyon * itemp * ftemp)
21: while ( denge==0 ) do
22: [mutasyon_popülasyon, mutasyon_hız]  $\leftarrow$  mutasyon (çaprazlanan_popülasyon, mutasyon_oranı)
23: [mutasyon_maliyeti]  $\leftarrow$  fitness_function (mutasyon_popülasyon, agac_indis, mutasyon_hız)
24: delta  $\leftarrow$  mutasyon_maliyet-çaprazlama_maliyet
25: if ( delta < 0 )
26: cozum_indis  $\leftarrow$  mutasyon_popülasyon
27: maliyet  $\leftarrow$  mutasyon_maliyeti
28: else
29: olasılık  $\leftarrow$  exp(-delta/temp)
30: if ( rand < olasılık )
31: cozum_indis  $\leftarrow$  mutasyon_popülasyon
32: maliyet  $\leftarrow$  mutasyon_maliyeti
33: else
34: cozum_indis  $\leftarrow$  çaprazlanan_popülasyon
35: maliyet  $\leftarrow$  çaprazlama_maliyeti
36: end /* if ( rand < olasılık ) */
37: end /* if ( delta < 0 ) */
38: komsusayı+1
39: if ( komsusayı == dengekoşulu )
40: denge  $\leftarrow$  1
41: end /* if ( komsusayı == dengekoşulu ) */
42: end /* while fail do */
43: olası_çözüm  $\leftarrow$  çözüm_indis
44: /* Sıcaklığı Güncelle */
45: temp  $\leftarrow$  temp/(1+beta*temp)
46: güncel_uygunluk(iterasyon)  $\leftarrow$  maliyet
47: if ( güncel_uygunluk(iterasyon) < önceki_uygunluk )
48: önceki_uygunluk  $\leftarrow$  güncel_uygunluk(iterasyon)
49: uygunluk(iterasyon)  $\leftarrow$  önceki_uygunluk
50: çözüm  $\leftarrow$  olası_çözüm
51: [yeni_agac]  $\leftarrow$  matris_agac(çözüm)
52: else
53: uygunluk(iterasyon)  $\leftarrow$  önceki_uygunluk
54: [yeni_agac]  $\leftarrow$  matris_agac(çözüm)
55: end /* if ( güncel_uygunluk(iterasyon) < önceki_uygunluk ) */
```

```

56: [yeni_agac] ← matris_agac(olası_çözüm)
57: (success, fail) ← code_assigner( yeni_agac, yeni_veri_iletim_hızı )
58: if ( success )
59: /*Gelen yeni data transfer hızını yeni ağaçta belirtilen uygun yere ata*/
60: dışdöngü ← 1000
61: end /* if ( success ) */
62: if ( iterasyon == dışdöngü )
63: dışdöngü ← 1000
64: end /* if ( iterasyon == dışdöngü ) */
65: end /* if ( success ) */

```

Şekil A.3: ASAGA algoritmasının sözde kodu

EK-B

PARAZİT ETKİLERİNİN TESPİTİ

1. UCA Topolojisi

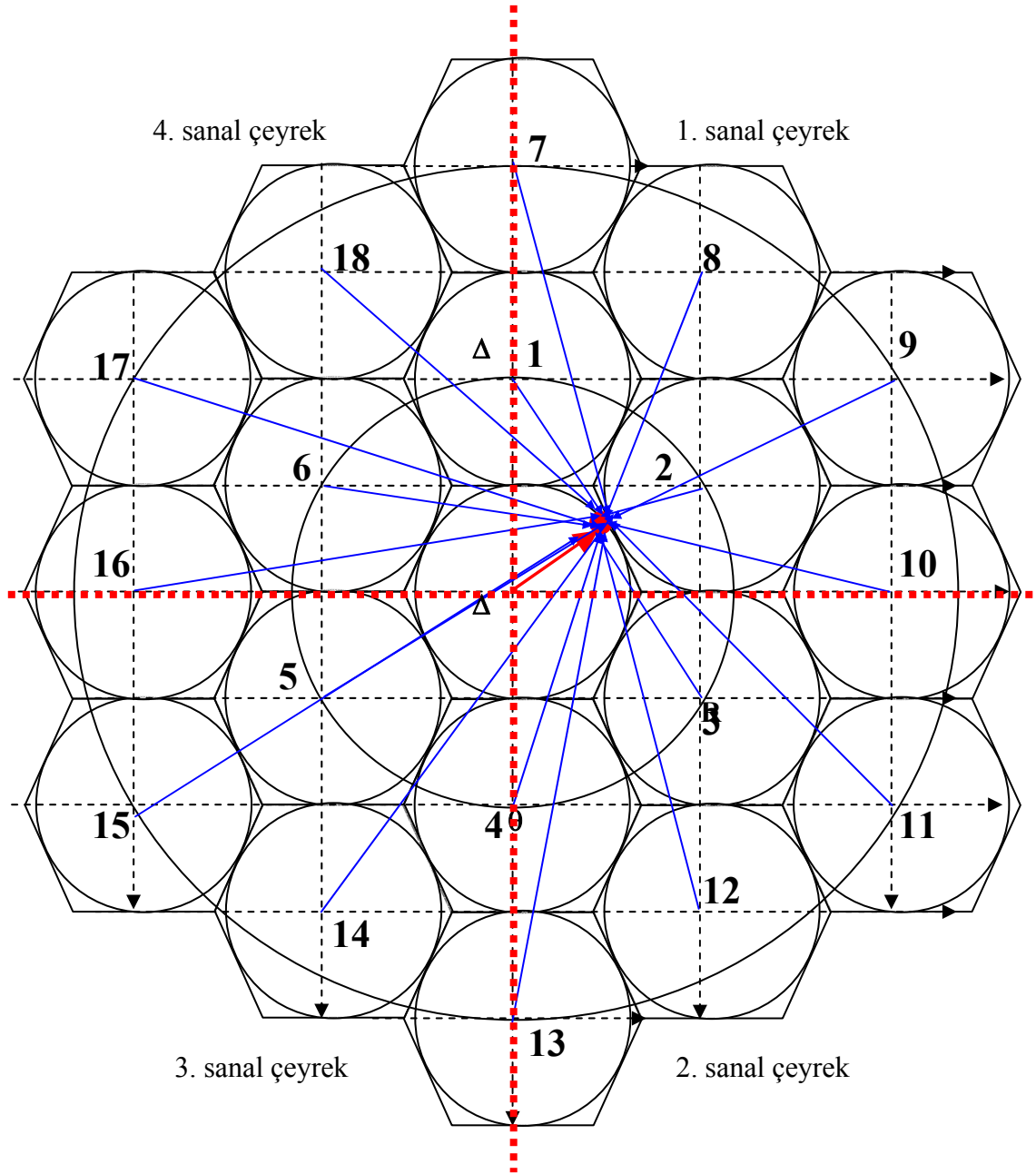
Şekil B.1'deki gibi verilen iki katmanlı bir hücresel ağ topolojisini ele alalım. UCA konfigürasyonunda her bir hücreden gelen ortak kanal parazit (co-channel interference, CCI) bilgisini belirlemek için ağ yapısının sanal olarak 4 çeyrek daireye bölündüğünü varsayalım. Her bir sanal parça 90°'lik açısal spektruma sahiptir. İstenilir kullanıcının DOA açısı bilgisine dayanarak diğer hücrelerden gelen parazit açısı tespit edilir. Burada,

$$D = \frac{3R}{\sqrt{3}} \quad (B.1)$$

D , komşu hücreler arasındaki dik uzaklık, R hücrenin yarıçapıdır. 1. çeyrek daire için diğer hücrelerden gelen parazit bilgileri aşağıda belirtilmiştir.

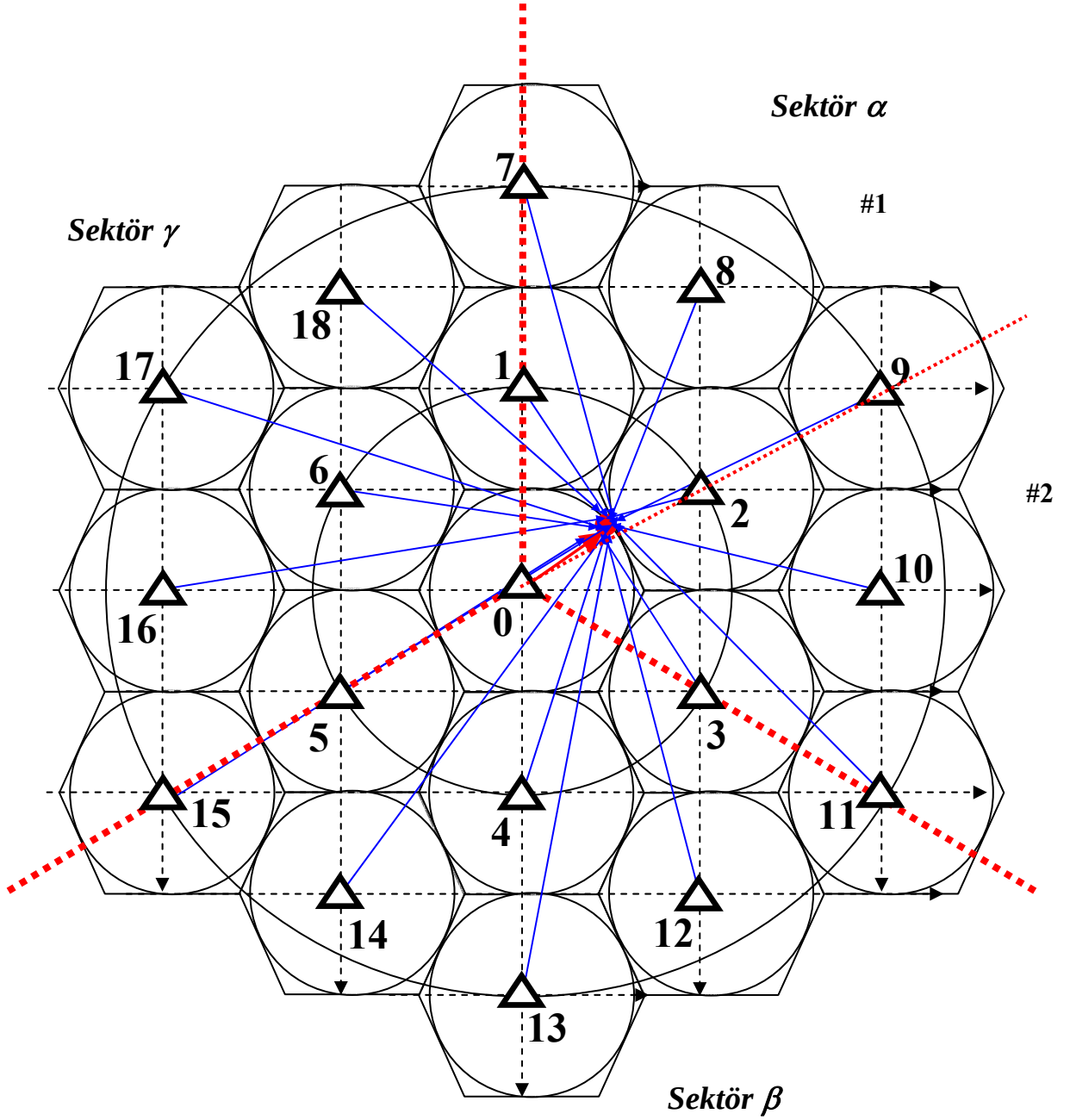
Tablo B.1: İstenilir kullanıcının 1. çeyrekte olduğu durum için parazit sinyallerin geliş açıları.

1. çeyrekten gelen parazit	3. çeyrekten gelen parazit
$\theta_1 = \pi - T^{-1} \frac{\Delta x}{2R - \Delta y}$	$\theta_4 = T^{-1} \frac{\Delta x}{2R + \Delta y}$
$\theta_2 = \pi + T^{-1} \frac{D - \Delta x}{R - \Delta y}$	$\theta_5 = T^{-1} \frac{D + \Delta x}{R + \Delta y}$
$\theta_7 = \pi - T^{-1} \frac{\Delta x}{4R - \Delta y}$	$\theta_{13} = T^{-1} \frac{\Delta x}{4R + \Delta y}$
$\theta_8 = \pi + T^{-1} \frac{D - \Delta x}{3R - \Delta y}$	$\theta_{14} = T^{-1} \frac{D + \Delta x}{3R + \Delta y}$
$\theta_9 = \pi + T^{-1} \frac{2D - \Delta x}{2R - \Delta y}$	$\theta_{15} = T^{-1} \frac{2D + \Delta x}{2R + \Delta y}$
2. çeyrekten gelen parazit	4. çeyrekten gelen parazit
$\theta_3 = 2\pi - T^{-1} \frac{D - \Delta x}{R + \Delta y}$	$\theta_6 = \pi - T^{-1} \frac{D + \Delta x}{R - \Delta y}$
$\theta_{10} = 2\pi - T^{-1} \frac{2D - \Delta x}{\Delta y}$	$\theta_{16} = T^{-1} \frac{2D + \Delta x}{\Delta y}$
$\theta_{11} = 2\pi - T^{-1} \frac{2D - \Delta x}{2R + \Delta y}$	$\theta_{17} = \pi - T^{-1} \frac{2D + \Delta x}{2R - \Delta y}$
$\theta_{12} = 2\pi - T^{-1} \frac{D - \Delta x}{3R + \Delta y}$	$\theta_{18} = \pi - T^{-1} \frac{D + \Delta x}{3R - \Delta y}$



Şekil B.1: İki katmanlı UCA topolojisi, istenilir gezgin kullanıcının 1. çeyrekte olduğu durum.

2. ULA Topoloji



Şekil B.2: İki katmanlı ULA topolojisi, istenilir gezgin kullanıcının sektör α 'da olduğu durum.

Doğrusal anten dizisi konfigürasyonunda hücresel ağ 3 sektöre bölünmüştür. Her sektörün sırasıyla ilgili oldukları açılar, $[30^\circ 150^\circ]$, $[150^\circ 270^\circ]$, $[270^\circ 30^\circ]$ arasında yer almaktadır. Sektör α için ULA topoloji konfigürasyonu ve parazit sinyallerin geliş açıları aşağıda verilmiştir.

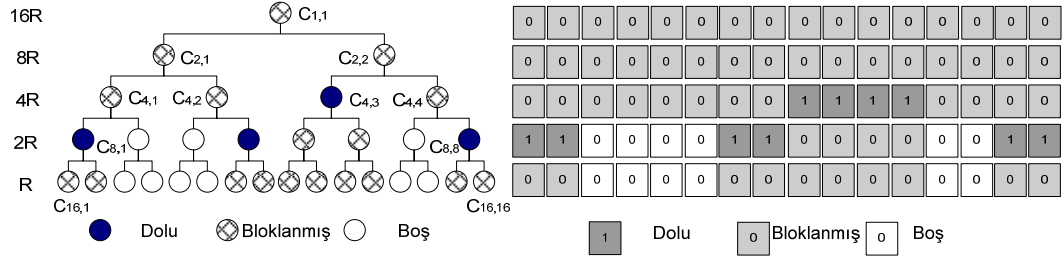
Tablo B.2: İstenilir kullanıcının sektör α 'da olduğu durum için parazit sinyallerin geliş açıları.

Sektör α 'da oluşan parazit	
$\theta_1 = \frac{\pi}{3} - T^{-1} \frac{\Delta x}{2R - \Delta y}$	$\theta_7 = \frac{\pi}{3} - T^{-1} \frac{\Delta x}{4R - \Delta y}$
$\theta_8 = \frac{\pi}{3} + T^{-1} \frac{D - \Delta x}{3R - \Delta y}$	$\theta_{10} = \frac{\pi}{6} + T^{-1} \frac{\Delta y}{2D - \Delta x}$
$30 \leq \theta_0 < 90$ #1 için	
$\theta_2 = \frac{\pi}{6} - T^{-1} \frac{R - \Delta y}{D - \Delta x}$	$\theta_9 = \frac{\pi}{6} - T^{-1} \frac{2R - \Delta y}{2D - \Delta x}$
$90 \leq \theta_0 < 150$ #2 için	
$\theta_2 = \frac{\pi}{3} + T^{-1} \frac{D - \Delta x}{R - \Delta y}$	$\theta_9 = \frac{\pi}{3} + T^{-1} \frac{2D - \Delta x}{2R - \Delta y}$
Sektör β 'dan gelen parazit	Sektör γ 'dan gelen parazit
$\theta_3 = 2\pi - T^{-1} \frac{D - \Delta x}{R + \Delta y}$	$\theta_5 = T^{-1} \frac{D + \Delta x}{R + \Delta y}$
$\theta_4 = T^{-1} \frac{\Delta x}{2R + \Delta y}$	$\theta_6 = \pi - T^{-1} \frac{D + \Delta x}{R - \Delta y}$
$\theta_{11} = 2\pi - T^{-1} \frac{2D - \Delta x}{2R + \Delta y}$	$\theta_{15} = T^{-1} \frac{2D + \Delta x}{2R + \Delta y}$
$\theta_{12} = 2\pi - T^{-1} \frac{D - \Delta x}{3R + \Delta y}$	$\theta_{16} = T^{-1} \frac{2D + \Delta x}{\Delta y}$
$\theta_{13} = T^{-1} \frac{\Delta x}{4R + \Delta y}$	$\theta_{17} = \pi - T^{-1} \frac{2D + \Delta x}{2R - \Delta y}$
$\theta_{14} = T^{-1} \frac{D + \Delta x}{3R + \Delta y}$	$\theta_{18} = \pi - T^{-1} \frac{D + \Delta x}{3R - \Delta y}$

EK-C

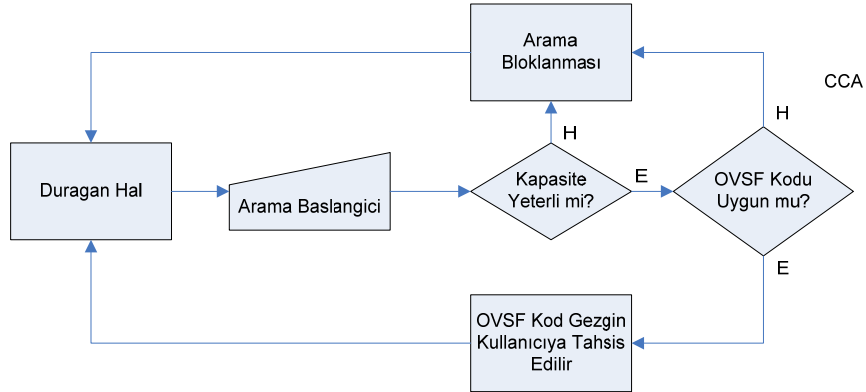
HESAPLAMA YÜKÜ MALİYET ANALİZİ

Bu ekte CCA, DCA, SGA, D&D-GA, SA ve ASAGA algoritmalarının hesaplama yükleri üzerinde durulacaktır. Hesaplama yükü analizinde OVFS kod ağacını yine matris formda tuttuğumuzu varsayarak yapacağız. Matris form aşağıdaki şekilde verilmektedir.

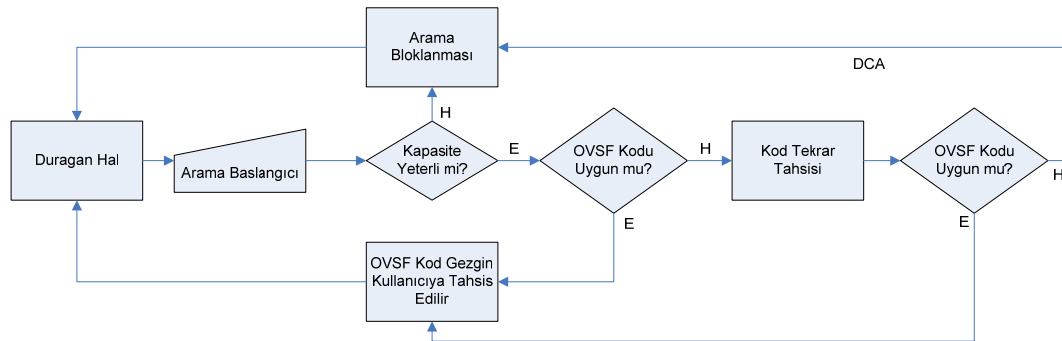


Şekil C.1: OVFS kod ağacı ve onun matrissel ifadesi

Aşağıda ele alınan her bir alt başlık algoritmaların akış diyagramlarına göre belirlenmiştir. CCA ve DCA algoritmaları için akış diyagramı aşağıdaki şekillerde verilmektedir.



Şekil C.2: Geleneksel kod tahsis (CCA) algoritması akış diyagramı



Şekil C.3: Dinamik kod tahsis (DCA) algoritması akış diyagramı

GA (SGA ve D&D-GA), SA ve ASAGA için akış diyagramları sırasıyla Şekil 3.1, Şekil 3.9, Şekil 3.12’de verilmiştir.

Durağan Hal

OVSF kod ağacında bir kanallama kodu isteği yani herhangi bir çağrı talebi yoktur.

Arama Başlangıcı

$S(n)$ veri transfer hızı ile iletim yapmak için uygun OVSF kanallama kodu gereksiniminde bulunan kullanıcının bu isteği kaynak yöneticisine iletilir.

Kapasite Kontrolü

Kapasite kontrolü için

$$n = SF - \sum_{i=1}^U S(i) \quad (C.1)$$

formülü kullanılır. OVSF kod ağacındaki kullanıcıların veri transfer hızları toplanır. Uyarlamalı yapı için $U+2$ adet hesaplama yükü sisteme getirilmiş olur. En kötü durumdaki senaryo için ağacın tamamen dolu ve R veri transfer hızlı kullanıcılardan oluştuğunu varsayalım. Sonuçta sıfır ile toplamayı da katarsak SF adet toplama işlemi yapmış oluruz. Elde ettiğimiz toplam değeri SF ’den çıkartırsak ağacın boş kapasitesini öğreniriz. Sonuçta bu işlem için sisteme $SF+2$ adet toplama yükü getirilmiş olunur.

Uygun OVSF Kod Kontrolü

Gelen veri transfer hızı isteğine göre OVSF kod ağacında $SF/S(n)$ adet arama yapıp ilgili dalın, onun torun ve ata dallarının boş ya da dolu kod olup olmadığı kontrol edilir. Bu işlem $S(n)$ defa yapılır. Sonuçta sıfır ile toplamayı da işin içine kattığımızda $SF/S(n)$ kadar daha işlem yükü gerekecektir. Yani uygun OVSF kod kontrolünün sisteme getirdiği hesaplama yükü $S(n)(\log_2 SF + 1)(SF/S(n)) + (SF/S(n))$ ile verilir.

Kod Tahsis

Eğer veri transfer ihtiyacında bulunan kullanıcı için uygun boş OVSF kanallama kodu varsa bu durumda $S(n)$ hızı yeni kullanıcıya tahsis edilmiş olur.

Arama Bloklanması

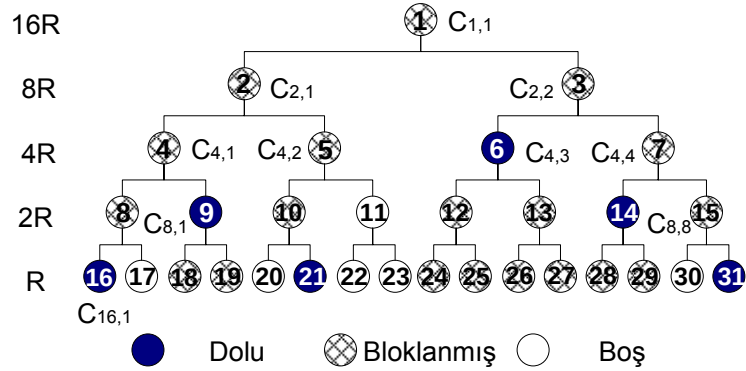
Çağrı talebi reddedilir. Herhangi bir yük yoktur.

Mevcut OVSF Kod Ağacı

OVSF kanallama kodlarının tekrar tahsisi amacıyla yer değiştirme işlemini gerçekleştirmek için ASAGA bloğuna giren kısımdır. OVSF kod ağacındaki dolu kodların yani kod matrisindeki 1’lerin indis numaralarının alınması durumudur. U adet kullanıcıya ait indis bilgisi alınacaktır. İndis bilgilerini içeren OVSF kod ağacı aşağıdaki şekildedir. Yapılan işlem U adet toplamadır.

Başlangıç Popülasyonu

Başlangıç popülasyonu, GA bloğunda işlem görecektir. OVSF kod ağaçlarının indislerini içeren ve büyüklüğü mevcut trafik yoğunluğuna bağlı olarak değişen yapıdır. Genetik algoritmanın en kritik noktası da popülasyon büyüklüğünün kullanıcı sayısına bağlı olarak adaptif olarak belirlenmesidir. Başlangıç popülasyonunun büyüklüğü Denklem B.1. ile verilmektedir. Burada, başlangıç popülasyonu için $n \times U$ boyutlu bir matris oluşturulacaktır ve her bir satırında U tane mevcut OVSF kod ağacının veri transfer hızı ile aynı hızda ve farklı indiste varyasyonları randperm komutu ile üretilir.



Şekil C.4: İndis bilgileri ile oluşturulmuş OVSF kod ağacı

Büyüklüğü belirlemek için yapılan işlem $U+2$ adet toplamadır (kapasite kontrolü). n adet U uzunluğunda başlangıç kromozomunu oluşturmak için yapılan işlem de nU adet toplamadır (Mevcut OVSF Kod Ağacı), yani $nU(U+1)$ adet toplamadır.

Uygunluk Fonksiyonu

Uygunluk değeri OVSF kod tahsis/tekrar–tahsis problemi için özel olarak tanımlanan uygunluk fonksiyonuna göre popülasyondaki her bir kromozom için elde edilir. j . Kromozomun uygunluk değeri $f(j)$ ile gösterilir ve Ch_{ini} 'ye göre j . popülasyonun her bir kromozomun yer değiştirme miktarları toplamını verir.

$$f(j) = \frac{1}{\sum_{i=1}^U \left[(Ch_{ini}(j) - P(j, i)) \times S(i) \right]} \quad (C.2)$$

burada, j kromozom sayısıdır (ağacın kapasitesi), $j=1, \dots, n$. Uygunluk değeri OVSF kod ağacının GA kullanılarak optimizasyonu probleminde 0-1 arasındadır ve 1'e yaklaştıkça kod değişim maliyeti azalır.

Burada, her bir kromozom için uygunluk değerinin hesaplanmasında $2U+1$ adet toplama, $2U+1$ adet çarpma işlemi gerçekleşmektedir. Elde edilen sonucu popülasyon büyüklüğünde genişlettiğimizde $(2U+1)n$ adet çarpma $(2U+1)n$ adet toplama işleminin elde edildiğini görmekteyiz.

Optimizasyon Kriteri

Optimizasyon kriteri SA ya da ASAGA bloğundan çıkmak için probleme özgü tanımlanan özel bir koşuldur ve OVFS kanallama kodu tekrar tahsisi için tanımlanan koşul “en az maliyetle kodların yerlerini değiştirerek veri transfer isteğinde bulunan kullanıcıya uygun boş OVFS kanallama kodu tahsis etmek” tir.

Burada yapılan işlem ihtiyaç duyulan veri transfer hızına uygun boş kanallama kodunun olup olmadığının kontrol edilmesidir. Uygun OVFS kod kontrolü bölümünde olduğu gibi gelen veri transfer hızına göre OVFS kod ağacında $SF/S(n)$ adet arama yapıp ilgili dalın, onun torun ve ata dallarının boş ya da dolu kod olup olmadığı kontrol edilir. Optimizasyon kriterinin sisteme getirdiği hesaplama yükü, uygun OVFS kod kontrolünde olduğu gibi $S(n)(\log_2 SF + 1)(SF/S(n)) + (SF/S(n))$ ile verilir.

Seçim Tekniği

Popülasyondaki kromozomların uygunluk değerleri temel alınarak seçim operatörü n kromozomlu yeni bir popülasyon üretir. Bu yeni popülasyon orijinal popülasyondaki uygunluk değerleri iyi olan kromozomların çeşitli tekniklere göre elde edilmesi ile oluşturulur. Bu bağlamda çalışmada rulet tekerleği, turnuva, en iyi yüzde ve rasgele teknikleri gibi farklı seçim operatörleri incelenmiştir.

Rulet Tekerleği; yine temel olarak $n \times U$ boyutlu bir matris indis rulet tekerleğinden gelen kromozomların indis bilgileri ile doldurulacaktır. Uygunluk fonksiyonu bölümü tarafından popülasyonu oluşturan tüm kromozomlara ait uygunluk değerleri alınır. Dolayısıyla yeni kromozomların rulet tekerleğinden nasıl seçildiğinin ya da seçilen kromozomun uygunluk değerinin yüksek olup olmadığının önemi yoktur. Önemli olan sadece seçim işlemidir. Yapılan işlem rulet tekerleğine yerleştirmeden dolayı n ve seçimden dolayı $n \times U$ adet olmak üzere $n + n \times U$ adet toplamadır.

Turnuva; bu metotta başlangıç popülasyonundan rasgele alınan iki kromozom karşılaştırılarak uygunluk değeri yüksek olan alınır, bu işlem tüm popülasyon tamamlanana kadar devam eder. Bu metodu oluşturan bileşenlerden seçim için 2, karşılaştırma için 1, popülasyonu tamamlama için nU olmak üzere toplam $3nU$ hesaplama yükü sisteme getirilmiş olunur.

En iyi yüzde; bu metotta kromozomlar popülasyonun uygunluk değerleri cinsinden en iyi % ξ lik kısımdan rasgele olarak seçilir. Burada ξ , 0–100 arasında seçilebilir. Bu çalışmada 0–30 arasında rasgele bir değerde seçim işlemini ele aldık. Seçilen kromozomun belirtilen kritere uygun olup olmadığının testi için bir karşılaştırma durumu söz konusudur. Dolayısıyla seçim için 1, karşılaştırma için 1 ve popülasyonu tamamlama için nU olmak üzere toplam hesaplama yükü $2nU$ ile verilir.

Rasgele; bu teknikte seçilen kromozomlar uygunluk değerine bakılmaksızın rasgele olarak elde edilirler. Yapılan işlem nU adet toplamadır.

Çaprazlama Tekniği

Kromozomlar arasındaki bilgi değişimini gerçekleştirerek daha iyi kromozomlar üretmek için çaprazlama tekniği kullanılır. Çaprazlama işlemi önceden belirlenen çaprazlama oranına (p_c) göre popülasyonun tamamına ya da bir kısmına uygulanır. Çaprazlama işlemine tabi tutulacak kromozom sayısı (j)

$$j = n \times p_c \quad (C.3)$$

ile tespit edilir. Çaprazlama tekniği kullanıcıya bağlı olarak girilen çaprazlama oranına bağlı kalınarak gerçekleştirilen bir tekniktir. ASAGA için çaprazlama oranı (p_c) 0.8 değerindedir. Bu bağlamda, çeşitli çaprazlama operatörleri incelenmiştir.

Tek Noktalı; bu çaprazlama operatöründe rasgele seçilen noktadan itibaren anne ve baba kromozomun genleri kendi arasında yer değiştirerek çocuk kromozomları oluşturur. Her bir kromozom için 1 tane kesim noktası alınır. Dolayısıyla tek noktalı çaprazlama tekniği için çaprazlama oranını belirlemek için 1, kesim noktasının belirlenmesi için 1 ve j adet kromozomu kendi arasında yer değiştirmek için jU olmak üzere, tek noktalı çaprazlama tekniğinin sisteme getirdiği hesaplama yükü $j(U+1)+1$ ile verilir.

İki Noktalı; bu teknikte ise rasgele seçilen nokta sayısı ikidir. O zaman sadece kesim noktalarının belirlenmesi için gereken hesaplama yükü 2 olur ve sonuçta sisteme getirilen hesaplama yükü $j(U+2)+1$ ile verilir.

Düzenli; belli bir olasılıkla rasgele belirlenmiş genleri anneden kalan genleri de babadan alınarak çocuk kromozomu oluşturur. Bu teknik ile elde edilen kromozomlar anne ve babaya ait genleri parça parça değil (tek veya iki noktalı çaprazlama tekniğindeki gibi) gen bazında alarak kendilerine ait kromozomu oluştururlar. Bu teknikte olasılığın miktarını belirlemek için 1, hangi genin anneden hangi genin babadan alınacağını belirlemek için U , bir çocuk kromozomu oluşturmak için anne ve babadan alınan genler için toplam U ve çaprazlama oranı miktarınca hesaplamak için j olmak üzere, toplam hesaplama yükü $2jU+1$ ile verilir.

Aritmetik; bu çaprazlama tekniğinde anne ve babanın doğrusal kombinasyonu ile yeni kromozomlar oluşturulur. Bu doğrusal kombinasyon

$$cocuk1 = a * anne + (1 - a) * baba$$

$$cocuk2 = (1 - a) * anne + a * baba \quad (C.4)$$

şeklinde dir. Burada, a rasgele seçilen bir ağırlık faktörüdür. Burada çaprazlama oranını belirlemek için 1, ağırlık faktörünü belirlemek için 1 ve yapılan matematiksel işlemler ile beraber $6jU+2$ toplama ve $4jU$ çarpma hesaplama yükü sisteme getirilmiş olur.

Mutasyon Tekniği

Çaprazlama tekniği sonrasında genetik çeşitliliği arttırmak ve kromozomların yerel minimuma takılmasını önlemek amacıyla mutasyon operatörü kullanılır. Bu işlem için 0-1 arasında rasgele bir sayı seçilir. Eğer bu sayı mutasyon oranından (p_m) küçükse, o zaman mutasyon işlemi başlatılır; aksi takdirde bir sonraki kromozom için başka bir rasgele sayı alınır.

$$\begin{aligned} \text{if } \text{rand}(1) < p_m \quad & p(j, l_{\text{yeni}}) = p(j, k_{\text{eski}}) \text{ and } p(j, k_{\text{yeni}}) = p(j, l_{\text{eski}}) \\ \text{else} \quad & p(j, l_{\text{yeni}}) = p(j, l_{\text{eski}}) \text{ and } p(j, k_{\text{yeni}}) = p(j, k_{\text{eski}}) \end{aligned} \quad (\text{C.5})$$

Mutasyon tekniği, genetik araştırma kısmının önemli bir parçasıdır ve sistemin yerel minimumda takılmasını önler. Çaprazlama tekniğinde olduğu gibi mutasyon tekniği de kullanıcıya bağlı bir parametre değerine göre çalışır. Çalışmamızda çeşitli mutasyon operatörleri incelenmiştir.

Yerdeğiştirme; kromozomda seçilen rasgele iki gen arasında yer değiştirme işlemi gerçekleştirilir. Yapılan işlem yükü mutasyon için seçilen kromozomun uygun olup olmadığını test için 1, her bir kromozom için yer değiştirme amaçlı olarak 2 olmak üzere n adet kromozom için hesaplama yükü $2n+1$ adet toplamadır.

Öteleme; bu mutasyon tekniğinde de seçim koşuluna göre sağa ya da sola öteleme işlemi gerçekleştirilir. Yapılan işlem yükü mutasyon için seçilen kromozomun uygun olup olmadığını test için 1, her bir kromozom için öteleme amacıyla yerdeğiştiren gen sayısı U ve ötelemeli mutasyon işleminin uygulandığı kromozom sayısı n adet toplama olmak üzere sistem için verilen hesaplama yükü $nU+1$ ile ifade edilir.

Fc ve Fm Uygunluğu

ASAGA için GA bloğu ile SA bloğunun birleşim noktasıdır. Burada hem çaprazlama hem de mutasyon bloğundan elde edilen kromozomların uygunlukları hesaplanır. Aslında kromozomun boyutu hem çaprazlama bloğu çıkışı hem de mutasyon bloğu çıkışı için aynı olacağından bir tanesini hesaplayıp 2 ile çarpmak yeterli olacaktır. Bu amaçla bir kromozom için uygunluk fonksiyonu hesabı işlem yoğunluğu ele alınır. Her bir kromozom için uygunluk değerinin hesaplanmasında $(2U+1)n$ adet toplama, $2U$ adet çarpma işlemi gerçekleşmektedir. Bu değerleri 2 ile çarptığımızda $4U+2$ adet toplama, $4U$ adet çarpma işlemi yapılmış olur.

Varolan Çözüme Yakın Yeni Bir Çözüm Üretimi

SA'da var olan çözüme yakın ve OVFS kod ağacının diklik prensibine uygun ikinci olası çözüm üretilir. Bunun için U adet toplama işlemine ihtiyaç vardır. Ancak her bir aktif kullanıcı indisi için kod ağacı kontrol edilmelidir. Mesela birinci kullanıcının veri transfer hızı R olsun. Bu kullanıcı için OVFS kod ağacı $\frac{SF}{R}$ defa taranarak diklik prensibi test edilecektir. Yani toplam iş yükü burada $\frac{SF \cdot U}{R}$ 'dir.

Bunun için $S(n)(\log_2 SF + 1)(SF/S(n)) + (SF/S(n))$ adet toplama ile uygun OVFS kod kontrolü ve U adet toplama ile onların yerleştirilmesi yapılır. Yani sistemdeki toplam hesaplama yükü $S(n)(\log_2 SF + 1)(SF/S(n)) + (SF/S(n)) + U$ ile verilir.

Δc -Çözümlerin Maliyetleri Arasındaki Değişim

SA bloğu için rasgele başlangıç çözümü ve var olan çözüme yakın yeni bir çözüm arasındaki uygunluk değerleri farkı, aynı şekilde ASAGA için çaprazlamadan çıkan kromozom ile mutasyondan çıkan kromozomların uygunluk değerleri arasındaki fark çözümlerin maliyetleri arasındaki değişim (Δc) ile ifade edilir. Bu değişimin sisteme getirmiş olduğu matematiksel yük ise 1 adet toplamadır.

$\Delta c < 0$ ve $\exp(\Delta c/T > R)$ Kriteri

$\Delta c < 0$, maliyet farkının durumunu kontrol eder, burada 1 adet karşılaştırma işlemi yapılacağından değişimin sisteme getirdiği yük 1 adet toplamadır. SA için $\Delta c < 0$ koşulu sağlanıyorsa var olan çözüm yeni çözüm, ASAGA için $\Delta c < 0$ koşulu sağlanıyorsa da var olan çözüm mutasyondan dönen çözümdür. İlgili koşulun sağlanmadığı durumlarda $\exp(\Delta c/T > R)$ koşulu incelenir. Burada R, rasgele seçilmiş bir deðişkendir ($\text{rand}(1)$). Burada da 1 adet karşılaştırma işlemi yapılacağından değişimin sisteme getirdiği yük 1 adet toplamadır. Eğer $\exp(\Delta c/T > R)$ ile verilen koşul sağlanırsa SA için bir deðişiklik olmaz, yine var olan çözüm yeni çözüm olarak alınır. Ancak ASAGA için $\exp(\Delta c/T > R)$ ile verilen koşul sağlanırsa, bu durumda var olan çözüm çaprazlamadan dönen çözüm olur. Yani bu iki koşulun sisteme getirmiş olduğu yük 2 adet toplamadır.

Var olan Çözümün Belirlenmesi

Burada var olan çözüm olarak hangi olası çözümün alınacağına karar verilmesi söz konusudur. ASAGA için mutasyondan dönen kromozomun mu yoksa çaprazlama operatöründen çıkan kromozomun mu alınacağına, SA için ise var olan çözümün mü yoksa var olan çözüme yakın üretilen yeni çözümün mü alınacağına karar veren blok yapısıdır. Burada önce karşılaştırma daha sonra da yeni çözüm olarak alınacak ifadenin etiketlenmesi söz konusudur. Dolayısıyla karşılaştırma için 1, etiketleme için U olmak üzere toplam hesaplama yükü $U+1$ ile ifade edilir.

İç Döngü

İç döngüde ilk olarak her sıcaklık değerinde kaç kez olası yeni çözüm üretileceğine karar verilir. Bu değer simülasyonlarda beş alınmıştır. Yani döngünün sisteme getirdiği hesaplama yükü beş ile çarpılacaktır. Bilindiği üzere iç döngü koşulu yeteri kadar yeni çözüm üretiminin gerçekleştirilmesidir. Her sıcaklık değeri için durumlardan beş tane daha yakın komşulukta OVFS kod ağaçları üretilerek bunların içlerinden uygunluk değeri en küçük olan iç döngüden çıkar. Yani iç döngü ile yerelde yakın komşuluklarda yapılan aramalar sonucunda en iyi uygunluk değerini veren OVFS kod ağacı döngüden çıkar. Bunun için ilk olarak yeni bir çözümün üretilmesi daha sonra da uygunluk değerlerinin karşılaştırılması ve en son olarak da uygunluk değeri en küçük olan çözümün olası çözüm olarak alınması söz konusudur. Yeni çözümün üretilmesi için var olan çözüme yakın bir çözümün üretilmesinde olduğu gibi sisteme $S(n)(\log_2 SF + 1)(SF/S(n)) + (SF/S(n)) + U$ adet toplama yükü getirir. Bu yeni çözümün uygunluk değerini hesaplamak için $(2U+1)n$ adet toplama, karşılaştırmak için 1 adet toplama ve son olarak da küçük olanın alınması ve olası çözüm olarak atanması için $U+1$ adet toplama işlemine gereksinim vardır.

Yani sisteme getirilen toplam hesaplama yükü $S(n)(\log_2 SF + 1)(SF/S(n)) + (SF/S(n)) + U + (2U+1)n + U + 2$ ile verilir. İç döngü 5 defa tekrarlanacağından bu değer 5 ile çarpılır.

Sıcaklığın Güncellenmesi

Sıcaklığı güncellemek amacı ile oluşturulan soğutma çizelgesinin seçiminde orantısal azalımlı ve Lundy & Mees soğutma çizelgeleri kullanılmıştır. Bu çizelgeler 1000 defa tekrar eden dış döngü boyunca kendi değerini yenilemektedir. Yani en kötü durum senaryosu için yapılacak işlem yükü 1000 defa tekrarlanacaktır. Orantısal azalımlı soğutma çizelgesi

$$T_{k+1} = \alpha T_k \quad (C.6)$$

$$\alpha = M \sqrt{\frac{T_s}{T_b}} \quad (C.7)$$

ve Lundy & Mees soğutma çizelge

$$T_{k+1} = \frac{T_k}{1 + \beta T_k} \quad (C.8)$$

$$\beta = \frac{T_b - T_s}{MT_b T_s} \quad (C.9)$$

ile sıcaklık değerlerini güncellemektedirler. [(C-6)–(C-9)] arasında verilen formüllerdeki tüm değerler skaler büyüklüklerdir. Yani, orantısal azalımlı soğutma çizelgesi için 4 adet çarpma 1 adet toplama, Lundy & Mees soğutma çizelgesi için 5 adet çarpma 4 adet toplama işlem yükü sistemde oluşmaktadır.

Sonuçta CCA, DCA, SGA, D&D-GA, SA ve ASAGA için hesaplama yükleri ayrı ayrı hesaplanmıştır. Hesaplama yükleri sırasıyla

$$2S(n) + SF(\log_2 SF + 1) + SF/S(n) + U + 2 \quad (C.10)$$

$$2S(n) + 2SF(\log_2 SF + 1) + 2SF/S(n) + S(i) + U + 2 \quad (C.11)$$

$$2S(n) + 2SF(\log_2 SF + 1) + 2SF/S(n) + 2U + nU(nU + 4) + 4n + j(U + 1) + 4 \quad (C.12)$$

$$2S(n) + 2SF(\log_2 SF + 1) + 2SF/S(n) + 2U + nU(nU + 5) + 7n + 2j(U + 1) + 6 \quad (C.13)$$

$$2S(n) + 8SF(\log_2 SF + 1) + 8SF/S(n) + 2U + 10 \quad (C.14)$$

$$2S(n) + 7SF(\log_2 SF + 1) + 7SF/S(n) + 7U + nU(U + 4) + 4n + j(U + 1) + 14 \quad (C.15)$$

Sonuçları karşılaştırdığımızda, ASAGA algoritmasının sisteme en fazla işlem yükü getirdiği görülmektedir. Bunu SA algoritması, D&D-GA, SGA, DCA ve CCA izlemektedir. Basite indirgenmiş (primitive) sonuçlarda ise

$$CCA = \sigma(SF, S(n))/2 + S(n) + U + 2 \quad (C.16)$$

$$DCA = \sigma(SF, S(n)) + S(i) + U + 2 \quad (C.17)$$

$$SGA = \sigma(SF, S(n)) + 2U + (nU + 2)^2 + j(U + 1) \quad (C.18)$$

$$D \ \& \ D - GA = \sigma(SF, S(n)) + 2U + (nU + 2)^2 + n(U + 7) + 2j(U + 1) + 2 \quad (C.19)$$

$$SA = \sigma(SF, S(n)) + 6SF(\log_2 SF + 1) + 6SF/S(n) + 2U + 10 \quad (C.20)$$

$$ASAGA = \sigma(SF, S(n)) + 5SF(\log_2 SF + 1) + 5SF/S(n) + 7U + (nU + 2)^2 + 4n + j(U + 1) + 10 \quad (C.21)$$

ile verilir. Burada basite indirgeme fonksiyon ise

$$\sigma(SF, S(n)) = 2S(n) + 2SF(\log_2 SF + 1) + 2SF/S(n) \quad (C.22)$$

ile verilir.

KİŞİSEL YAYINLAR ve ESERLER

A. Uluslararası Hakemli Dergilerde (SCI) Yayınlanan Makaleler:

31. 8. **Mustafa Karakoç**, Adnan Kavak, “Genetic Approach for Dynamic OVFS Code Allocation in 3G Wireless Networks,” Applied Soft Computing, Elsevier (Baskıda)
30. 7. **Mustafa Karakoç**, Adnan Kavak, “Performance of Adaptive Antenna Array Based Dynamic OVFS Code Assignment Under Multipath Propagation and Multicell Environment,” Journal of Frequenz, vol. 11-12, pp. 254-257 2007
29. 6. Adnan Kavak, **Mustafa Karakoç**, “Smart Antenna Based OVFS Code Assignment for WCDMA Networks,” Wireless Personal Communications, Springer Verlag. (Baskıda)
28. 5. **Mustafa Karakoç**, Adnan Kavak, “Stochastic Optimization Algorithm Based Dynamic Resource Assignment for 3G Systems,” Springer Verlag Lecture Notes in Computer Science LNCS, vol. 4712, 2007.
27. 4. **Mustafa Karakoç**, Adnan Kavak, “Stochastic Methods for Dynamic OVFS Code Assignment in 3G Networks,” Springer Verlag Lecture Notes in Computer Science LNCS, vol. 4665, 2007.
26. 3. Adnan Kavak, **Mustafa Karakoç**, Murat Torlak, and Caner Özdemir, “Interference Analysis and Capacity for Forward Link 3G CDMA Network with Adaptive Antennas,” Wireless Communications and Mobile Computing, Wiley. (Baskıda)
25. 2. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, “Software radio implementation of smart antenna system for digital signal processor,” Springer Verlag Lecture Notes in Computer Science LNCS, vol. 3356, pp. 96–105, Dec. 2004.
24. 1. **Mustafa Karakoç**, Adnan Kavak, “Evaluation of smart antenna algorithms for cdma2000 reverse link,” Springer Verlag Lecture Notes in Computer Science LNCS, vol. 3042, pp. 1360-1365, May 2004.

B. Uluslararası bilimsel toplantılarda sunulan ve bildiri kitabında (Proceedings) basılan bildiriler:

23. 13. **Mustafa Karakoç**, Adnan Kavak, “A New Dynamic OVFS Code Allocation Method Based on Adaptive Simulated Annealing Genetic

Algorithm (ASAGA),” The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC'07, Sept 2007, Athens Greece.

22. 12. **Mustafa Karakoç**, Alev Söke, Adnan Kavak, “Dynamic OVSF Code Allocation in WCDMA Networks Using Dominance and Diploidy Genetic Algorithm,” 2007 IEEE Radio and Wireless Symposium, RAWCON’07, Jan. 2007, Long Beach USA.
21. 11. **Mustafa Karakoç**, Alev Söke, Adnan Kavak, “Dominance and Diploidy Based Genetic Algorithm Application: Dynamic OVSF Code Assignment for WCDMA Network,” IEEE 2006 International Symposium on Intelligent Signal Processing and Communication Systems, ISPACS’06, Dec. 2006, Tottori Japan.
20. 10. **Mustafa Karakoç**, Adnan Kavak, “Dynamic Resource Allocation for 3G Systems: OVSF Code Assignment Strategies,” 17th Wireless World Research Forum Meeting, WWRF 2006, Nov. 2006, Heidelberg, Germany.
19. 9. **Mustafa Karakoç**, Adnan Kavak, Joseph R. Cleveland, “Dynamic Allocation of OVSF Codes to Access Terminals with an Adaptive Antenna Array,” 16th Annual IEEE Symposium on Personal Indoor and Mobile Radio Communications, PIMRC 2005, Sept 2005, Berlin Germany.
18. 8. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, “Design and Hardware Implementation of a Novel Smart Antenna Algorithm with Using TI DSP’s,” IEEE International Workshop on Satellite and Space Communications, ISWCS 2005, Sept., 2005 .
17. 7. **Mustafa Karakoç**, Adnan Kavak, Caner Özdemir, “Optimization of Array Topology and Capacity for 3G CDMA Network with Adaptive Antennas,” IEEE 14th IST Mobile and Wireless Communications, Submit, June 2005, Dresden Germany.
16. 6. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, “Implementation of Several Smart Antenna Algorithms for cdma2000 Reverse Link on TMS320C6000 DSP Family,” The European DSP Education and Research Symposium EDERS 04, July 2004, Birmingham England.
15. 5. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, “C versus assembly implementation of smart antenna beamformers on floating point DSPs,” International Conference on Signals and Electronic Systems, ICSES 04, May 2004, Ponzan Poland.
14. 4. **Mustafa Karakoç**, Adnan Kavak, “Intelligent base stations for 3G CDMA: Downlink capacity analysis and optimization,” International Conference on Knowledge System Conference, IKS 04, May 2004,

Canakkale Turkey.

13. 3. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, "Implementation and performance comparison of DSP enabled LMS and CM beamformers for cdma2000 reverse link," International Conference on Knowledge System Conference, IKS 04, May 2004, Canakkale Turkey.
12. 2. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, "Real time implementation of LMS beamformer for cdma2000 3G system using TI tms320c6701," ELECO'03, Dec. 2003, Bursa TURKEY.
11. 1. **Mustafa Karakoç**, Adnan Kavak, Halil Yiğit, "Simulation and performance comparison of smart antenna algorithms for 3G wireless communications," 9th International Conference on Information Systems Analysis and Synthesis, July 31 to Aug. 2003, Orlando USA.

C. Ulusal hakemli dergilerde yayınlanan makaleler:

10. 3. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, Halil Yiğit, "Implementation and Performance Analysis of a Novel Smart Antenna Algorithm for 3G CDMA Systems", Journal of The Faculty of Engineering and Architecture of Gazi University, vol. 21, pp. 801-808, 2006.
9. 2. **Mustafa Karakoç**, Halil Yiğit, Kerem Küçük, Adnan Kavak, "Overview of smart antenna methodologies for mobile communications," Journal of Naval Science and Engineering, vol.2, no.1, pp.51-66, 2004.
8. 1. **Mustafa Karakoç**, Adnan Kavak, "Kablosuz haberleşme sistemleri için akıllı antenler," Gazi Üniversitesi Fen Bilimleri Enstitüsü Dergisi, vol.17, no.3, pp.133-144, 2004.

D. Ulusal bilimsel toplantılarda sunulan ve bildiri kitaplarında basılan bildiriler:

7. 7. **Mustafa Karakoç**, Adnan Kavak, "WCDMA Ağlarda Dinamik OVFSF Kod Tahsisi için Genetik Algoritma ve Benzetilmiş Tavlama," 15. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, IEEE SİU 2007, 11-13 July 2007, Eskişehir Turkey.
6. 6. **Mustafa Karakoç**, Adnan Kavak, "Uyarlanabilir Anten Dizisi Kullanan Yeni Dinamik OVFSF Kod Tahsis Algoritması," Union Radio Science International, URSI 06, Sept. 2006, Ankara Turkey.
5. 5. Kerem Küçük, Adnan Kavak, **Mustafa Karakoç**, Halil Yiğit, "C or Assembly Coding on Floating Point Digital Signal Processors for

Software Radio Implementation of a Beamformer?,” Bilişim Teknolojileri Kongresi, Bilkent 2005, June,2005, Eskişehir Turkey.

4. 4. **Mustafa Karakoç**, Adnan Kavak, “Üçüncü nesil kablosuz CDMA için uyarlamalı anten dizilerinin aşağı bağlantı parazit analizi ve sistem kapasitesi,” Union Radio Science International, URSI 04, May 2004, Ankara Turkey.
3. 3. **Mustafa Karakoç**, Adnan Kavak, “Cdma2000 yukarı bağlantı için akıllı anten algoritmalarının gelişimi,” Union Radio Science International, URSI 04, May 2004, Ankara Turkey.
2. 2. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, “Uyarlanabilir akıllı anten algoritmalarının yuvarlanır nokta DSP üzerinde gerçekleştirilmesi ve karşılaştırılması,” Union Radio Science International, URSI 04, May 2004, Ankara Turkey.
1. 1. Kerem Küçük, **Mustafa Karakoç**, Adnan Kavak, “Üçüncü nesil haberleşme sistemlerinde kullanılan akıllı anten algoritmalarının yazılım radyo kullanılarak gerçekleştirilmesi,” Union Radio Science International, URSI 04, May 2004, Ankara Turkey.

ÖZGEÇMİŞ

1978 yılında Kütahya’da doğdu. İlk, orta ve lise öğrenimini Kütahya’da tamamladı. 1997 yılında Kocaeli Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümüne girmiş, 2002 yılında bu bölümden mezun olmuştur. Daha sonra Abdi İbrahim İlaç Fabrikasında Validasyon Mühendisi olarak görev almış, 2003–2007 yılları arasında Kocaeli Üniversitesi Elektronik ve Bilgisayar Eğitimi Bölümünde araştırma görevlisi olarak görev yapmıştır. Ayrıca, 2004 yılında Kocaeli Üniversitesi Elektronik ve Bilgisayar Eğitimi Ana Bilim Dalında yüksek lisansını tamamlamıştır. Şu an Turkcell İletişim Hizmetleri A.Ş.’de Radyo Network bölümünde RF Mühendisi kıdemli uzman olarak görev yapmaktadır.