

**T.C.  
ONDOKUZ MAYIS ÜNİVERSİTESİ  
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI**



**BİLİŞSEL RADYO TABANLI IOT AĞLARINDA ENERJİ  
VERİMLİ RPL ALGORİTMASI TASARIMI**

Yüksek Lisans Tezi

**Ferhat ARAT**

Danışman

**Dr. Öğr. Üyesi Sercan DEMİRCİ**

SAMSUN  
2021

## TEZ KABUL VE ONAYI

Ferhat ARAT tarafından, Dr. Öğr. Üyesi Sercan DEMİRCİ danışmanlığında hazırlanan “Bilişsel Radyo Tabanlı IoT Ağlarında Enerji Verimli RPL Algoritması Tasarımı” başlıklı bu çalışma, jürimiz tarafından 14.6.2021 tarihinde yapılan sınav sonucunda oy birliği ile başarılı bulunarak Yüksek Lisans Tezi olarak kabul edilmiştir.

	Unvanı Adı Soyadı Üniversitesi Ana Bilim/Ana Sanat Dalı	İmza	Sonuç
Başkan	Doç. Dr. Mustafa Alper AKKAŞ Bolu Abant İzzet Baysal Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı		<input checked="" type="checkbox"/>
			Kabul
			<input type="checkbox"/>
			Ret
Üye (Danışman)	Dr. Öğr. Üyesi Sercan DEMİRCİ Ondokuz Mayıs Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı		<input checked="" type="checkbox"/>
			Kabul
			<input type="checkbox"/>
			Ret
Üye	Doç. Dr. Sedat AKLEYLEK Ondokuz Mayıs Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı		<input checked="" type="checkbox"/>
			Kabul
			<input type="checkbox"/>
			Ret

Bu tez, Enstitü Yönetim Kurulunca belirlenen ve yukarıda adları yazılı jüri üyeleri tarafından uygun görülmüştür.

ONAY

... / ... / ...

Prof. Dr. Ali BOLAT  
Enstitü Müdürü

## BİLİMSEL ETİĞE UYGUNLUK BEYANI

Hazırladığım Yüksek Lisans tezinin bütün aşamalarında bilimsel etiğe ve akademik kurallara riayet ettiğimi, çalışmada doğrudan veya dolaylı olarak kullandığım her alıntıya kaynak gösterdiğimi ve yararlandığım eserlerin Kaynaklar'da gösterilenlerden oluştuğunu, her unsurun enstitü yazım kılavuzuna uygun yazıldığını ve TÜBİTAK Araştırma ve Yayın Etiği Kurulu Yönetmeliği'nin 3. bölüm 9. maddesinde belirtilen durumlara aykırı davranılmadığını taahhüt ve beyan ederim.

12/07/2021  
Ferhat ARAT

## TEZ ÇALIŞMASI ÖZGÜNLÜK RAPORU BEYANI

**Tez Başlığı:** Bilişsel Radyo Tabanlı IoT Ağlarında Enerji Verimli RPL Algoritması Tasarımı

Yukarıda başlığı belirtilen tez çalışması için şahsım tarafından 14/04/2021 tarihinde intihal tespit programından alınmış olan özgünlük raporu sonucunda;

Benzerlik oranı : % 10

Tek kaynak oranı : % 4 çıkmıştır.

14/04/2021  
Dr. Öğr. Üyesi Sercan DEMİRCİ

## ÖZET

### BİLİŞSEL RADYO TABANLI IOT AĞLARINDA ENERJİ VERİMLİ RPL ALGORİTMASI TASARIMI

Ferhat ARAT

Ondokuz Mayıs Üniversitesi

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Ana Bilim Dalı

Yüksek Lisans, Haziran/2021

Danışman: Dr. Öğr. Üyesi Sercan DEMİRCİ

Nesnelerin İnterneti (IoT) kavramı, internete bağlanan cihaz sayısındaki artışa bağlı olarak çokça anılmaya başlamıştır. Algılayıcılar yardımıyla toplanan verinin işlenmesi ve ardından ortak bir kanal olan internet üzerinden paylaşılması prensibi IoT ağ paradigmasının temelini oluşturmaktadır. Bilişsel Radyo Ağları (CRN), günümüz ağ yapılarındaki spektrum yetersizliği probleminin çözümüne odaklanan bir ağ yapısıdır. Mevcut spektrum atanma ve kullanım şekline bakıldığında, ücret karşılığı kanal kullanım hakkına sahip lisanslı kullanıcıların, bu spektrum kanallarını verimsiz kullandığı ortaya çıkmıştır. Artan cihaz sayısı ve buna bağlı olarak yine artan kullanıcı sayısı, atıl spektrum bandı gibi problemler CRN kavramını ortaya çıkarmıştır. Lisanssız kullanıcıların, birincil kullanıcılar ağda aktif olmadığında iletişim yapması ilkesi ile atıl spektrum bantlarının verimli ve aktif bir şekilde kullanımı amaçlanmaktadır.

Tez kapsamında, CRN tabanlı IoT ağlarda çalışan yönlendirme algoritması tasarlanmıştır. CR düğümler kullanılabilir durumdaki frekans bantlarını algılar, bu frekanslara geçiş yaparlar ve kendilerine ayrılan zaman diliminde bant üzerinde iletişim kurarlar. Frekans geçiş maliyeti, en uygun durumdaki frekansa geçişe doğrudan bağlıdır. IoT ağlarında mevcut yönlendirme protokolleri, yönlendirmede frekans geçişini dikkate almamaktadır. Yönlendirme yolu, frekans geçiş maliyetine bakılmaksızın oluşturulur. Düşük Güç ve Kayıplı Ağlar için Yönlendirme Protokolü (RPL) de IoT ağlarda çalışan standart bir yönlendirme protokolüdür. CR cihazlarının karakteristik özellikleri ile, mevcut RPL algoritması geliştirilerek frekans geçişi farkındalıklı RPL (RPL-Sw) için yönlendirme amaç fonksiyonu oluşturulmuştur. Ayrıca, CR tabanlı Aç Gözlü RPL (CR-GreedyRPL) algoritması tasarlanmıştır. Tasarlanan bu model, yönlendirme yolu seçiminde frekans geçişini dikkate almakta, farklı frekans bantları kullanarak iletişimi desteklemektedir. Önerilen ve geliştirilen modellerin kapasite, enerji verimliliği ve enerji tüketimi açısından en iyi yolu seçmesi ve bu yönüyle yönlendirmede belirleyici olması amaçlanmaktadır. Tasarlanan bu modeller ile Dar Boğaz Tabanlı Yönlendirme (DTY) ve RPL algoritmaları Python programlama dili kullanılarak yapılan benzetimler sonucu karşılaştırılmış, sonuçlar ve bulgular değerlendirilmiştir. Yapılan benzetimler sonucu, önerilen algoritmaların performans açısından DTY ve RPL algoritmasına göre daha başarılı sonuçlar verdiği görülmüştür.

**Anahtar Sözcükler:** IoT, yönlendirme, CRN, spektrum, RPL, frekans geçişi, enerji.

## ABSTRACT

### DESIGN OF ENERGY EFFICIENT RPL ALGORITHM IN COGNITIVE RADIO BASED IOT NETWORKS

Ferhat ARAT

Ondokuz Mayıs University

Institute of Graduate Studies

Department of Computer Engineering

Master, June/2021

Supervisor: Asst. Prof. Dr. Sercan DEMİRÇİ

The Internet of Things (IoT) concept has started to be mentioned a lot due to the increase in the number of devices connected to the Internet. The principle of processing the data collected with the help of sensors and then sharing it over the internet, which is a common channel, forms the basis of IoT network paradigm. Cognitive Radio Networks (CRNs) is a network structure that focuses on solving the spectrum deficiency problem in today's network structures. Considering the current spectrum assignment and usage samples, it has been revealed that licensed users with channel usage rights for a fee use these spectrum channels inefficiently. Problems such as the increasing number of devices and the increasing number of users, and the idle spectrum band have revealed the CRN concept. It is aimed to use idle spectrum bands efficiently and actively with the principle that unlicensed users communicate when the primary users are not active on the network.

Within the scope of the thesis, a routing algorithm working on CRN-based IoT networks is designed. CR nodes perceive available frequency bands, switch to these frequencies and communicate on the band in their allocated time slot. The frequency switching cost is directly depends on the optimal frequency switching. Current routing protocols in IoT networks do not consider frequency switching in routing. The routing path is created regardless of the frequency switching cost. Routing Protocol for Low Power and Loss Networks (RPL) is also a standard routing protocol that works in IoT networks. With the characteristic features of CR devices, the existing RPL algorithm is developed and the routing objective function is created for the frequency switching aware RPL (RPL-Sw). In addition, the CR-based GreedyRPL (CR-GreedyRPL) algorithm is designed. This designed model considers the frequency switching in the routing path selection and supports communication by using different frequency bands. The proposed and developed models are aimed to choose the best way in terms of capacity, energy efficiency and energy consumption and to be decisive in this aspect. These designed models are compared with the results of the simulations made using Widest Path Routing (DTY) and RPL algorithms using Python programming language, and the results and findings are evaluated. As a result of the simulations, it has been seen that the proposed algorithms give more successful results than DTY and RPL algorithms in terms of performance.

**Keywords:** IoT, routing, CRN, spectrum, RPL, frequency switching, energy.

## ÖNSÖZ VE TEŞEKKÜR

Bu günlere gelmemde en büyük pay sahibi olan, şimdilik 19 sene süren eğitim hayatımın maddi manevi destekçisi, her zaman yanımda olan canım annem, babam ve abime,

Kıymetli bilgileri ile bu sürecin temellenmesinin önünü açan, çalışmaya teşvik eden, yol gösteren danışmanım, sayın, Dr. Öğretim Üyesi Sercan DEMİRCİ hocama,

Bu süreçte, öncesinde ve sonrasında benden desteğini esirgemeyen, Mustafa Reşit TAVUS, Aykut KARAKAYA ve Durmuş Özkan ŞAHİN hocalarıma,

Son olarak da üzerimde emeği olan, beni destekleyen arkadaşlarım ve diğer herkese,

Sonsuz teşekkürlerimi sunarım.

Ferhat ARAT

## İÇİNDEKİLER

<b>KISALTMALAR .....</b>	<b>Vii</b>
<b>ŞEKİLLER DİZİNİ.....</b>	<b>X</b>
<b>TABLolar DİZİNİ.....</b>	<b>Xii</b>
<b>1. GİRİŞ.....</b>	<b>1</b>
1.1. Önceki Çalışmalar .....	9
1.2. Motivasyon ve Katkı .....	13
1.3. Organizasyon.....	16
<b>2. DÜŞÜK GÜÇ VE KAYIPLI AĞLAR İÇİN YÖNLENDİRME PROTOKOLÜ (RPL).....</b>	<b>17</b>
2.1. RPL Algoritması Genel Yapısı.....	17
2.2. RPL Amaç Fonksiyonları ve Yönlendirme Metrikleri.....	22
2.3. RPL Benzetim Sonuçları .....	25
<b>3. BİLİŞSEL RADYO ÖZELLİKLİ IOT (CR-IOT) DÜĞÜMLERDE ENERJİ MODELİ.....</b>	<b>30</b>
3.1. CR-IoT Düğümlerde Tüketilen Enerjinin Modellenmesi.....	30
3.2. CR-IoT Düğümlerde Enerji Verimliliğinin Modellenmesi .....	34
3.3. Bilişsel Radyo Tabanlı Aç Gözlü RPL Algoritması (CR-GreedyRPL) .....	35
3.4. Frekans Değiştirme Farkındalıklı RPL Algoritması (RPL-Sw) .....	39
3.4.1. RPL-Sw Benzetim Sonuçları .....	40
3.5. Dar Boğaz Tabanlı Yol (DTY) Problemi .....	43
<b>4. BENZETİM.....</b>	<b>46</b>
4.1. Tasarlanan Modelin Örnek Test Çizgesinde Gösterimi .....	46
4.2. Benzetim Sonuçları .....	47
<b>5. SONUÇ VE DEĞERLENDİRME.....</b>	<b>55</b>
<b>KAYNAKLAR .....</b>	<b>58</b>
<b>EKLER .....</b>	<b>61</b>
<b>EK 1. ÖZ GEÇMİŞ.....</b>	<b>61</b>
<b>EK 2. BENZETİMLERE AİT PROGRAM PARÇALARI .....</b>	<b>62</b>

## KISALTMALAR

FCC	Federal İletişim Komisyonu (Federal Communications Commission)
CRN	Bilişsel Radyo Ağları (Cognitive Radio Networks)
PU	Birincil Kullanıcı (Primary User)
SU	İkincil Kullanıcı (Secondary User)
CR	Bilişsel Radyo (Cognitive Radio)
IoT	Nesnelerin İnterneti (Internet of Things)
PDA	Kişisel Dijital Asistan (Personal Digital Assistant)
TCP/IP	İletim Kontrol Protokolü/İnternet Protokolü (Transmission Control Protocol/Internet Protocol)
WSN	Kablosuz Sensör Ağları (Wireless Sensor Networks)
IPv4	Versiyon 4 İnternet Protokolü (Internet Protocol Version 4)
IPv6	Versiyon 6 İnternet Protokolü (Internet Protocol Version 6)
RPL	Düşük Güç ve Kayıplı Ağlar için Yönlendirme Protokolü (Routing Protocol for Low-Power and Lossy Networks)
DTY	Darboğaz Tabanlı Yönlendirme Algoritması (Widest Path Algorithm)
RPL-Sw	Frekans Geçiş Farkındalıklı RPL Algoritması (Switching Aware RPL Algorithm)
CR-GreedyRPL	CR Tabanlı Frekans Değişirme Farkındalıklı Açgözlü RPL Algoritması (CR-based Greedy RPL Algorithm)
OSI	Açık Sistemler Ara Bağlantısı (Open Systems Interconnection)
ISO	Uluslararası Standardizasyon Örgütü (International Organization for Standardization)
DSA	Dinamik Spektrum Erişimi (Dynamic Spectrum Access)
QoS	Servis Kalitesi (Quality of Service)
UDP	Kullanıcı Datagram Protokolü (User Datagram Protocol)
LEACH	Düşük Enerjili Uyarlanabilir Kümeleme Hiyerarşisi (Low Energy Adaptive Clustering Hierarchy)
SAR	Kendi Kendini Uyarlayabilen Yönlendirme (Self Adaptive Routing)

CSRP	Ortak Kanal Seçimi ve Yönlendirme Protokolü (Joint Channel Selection and Routing Protocol)
TDRP	Gecikme Tabanlı Yönlendirme Protokolü (Routing Protocol Based on Delay)
JRCA	Ortak Yönlendirme ve Kanal Tahsisi (Joint Routing and Channel Allocation)
C-TRP	Bilişsel Radyo Ağları için Ağaç Yönlendirme Protokolü (Tree Routing Protocol for Cognitive Radio Network)
CTBR	Bilişsel Ağaç Tabanlı Yönlendirme (Cognitive Tree-Based Routing)
STOD-RP	İstek Tabanlı Spektrum Ağacı Yönlendirme Protokolü (Spectrum Tree Based On Demand Routing Protocol)
IETF	İnternet Mühendisliği Görev Gücü (Internet Engineering Task Force)
ROLL	Düşük Güç ve Kayıplı Ağlar Üzerinden Yönlendirme (Routing Over Low Power and Lossy Networks)
6LoWPAN	Düşük Güçlü Kablosuz Kişisel Alan Ağları üzerinden IPv6 (IPv6 over Low -Power Wireless Personal Area Networks)
IoMT	Multimedya Nesnelerinin İnterneti (Internet of Multimedia Things)
ETX	Beklenen İletim Sayısı (Expected Transmission Count)
QU-RPL	Kuyruk Verimi Tabanlı RPL (Queue Utilization based RPL)
ROEE	Kaynak Odaklı ve Enerji Verimli Protokol (Resource Oriented and Energy Efficient Protocol)
ELT	Beklenen Yaşam Süresi (Expected Life Time)
LLN	Düşük Güç ve Kayıplı Ağlar (Low-Power and Lossy Networks)
DTN	Gecikme/Kesilme Dayanıklı Ağ (Delay/Disruption Tolerant Networking)
IEEE	Elektrik ve Elektronik Mühendisleri Enstitüsü (Institute of Electrical and Electronics Engineers)
P2P	Noktadan Noktaya (Point-to-Point)
P2MP	Noktadan Çok Noktaya (Point-to-Multi Point)
MP2P	Çok Noktadan Noktaya (Multi Point-to-Point)

DODAG	Hedefe Yönelik Yönlü Çevrimsiz Çizge (Destination Oriented Directed Acyclic Graph)
DAG	Yönlü Çevrimsiz Graf (Directed Acyclic Graph)
OF	Amaç Fonksiyonu (Objective Function)
DIO	DODAG Bilgi Nesnesi (DODAG Information Object)
DAO	Hedef Reklam Nesnesi (Destination Advertisement Object)
DIS	DODAG Bilgi Talebi (DODAG Information Solicitation)
ICMPv6	Versiyon 6 İnternet Kontrol Mesaj Protokolü (Internet Control Message Protocol Version 6)
SNR	Sinyal/Gürültü Oranı (Signal to Noise Ratio)
SISO	Tek Girişli Tek Çıkışlı (Single Input Single Output)
MIMO	Çok Girişli Çok Çıkışlı (Multiple Input Multiple Output)

## ŞEKİLLER DİZİNİ

Şekil 1.1. Spektrum bandı kullanımı (Akyıldız vd., 2006).....	1
Şekil 1.2. Nesnelerin interneti uygulama alanları (Ala Al-Fuqaha vd., 2015) .....	6
Şekil 1.3. Nesnelerin interneti için önerilen mimariler (Ala Al-Fuqaha vd., 2015) .....	6
Şekil 1.4. Örnek bir yönlendirme yolu .....	8
Şekil 1.5. Tez kapsamında incelenen çalışmalar .....	15
Şekil 2.1. a) MP2P iletişim b) P2P iletişim c) P2MP iletişim .....	17
Şekil 2.2. Aşağı yönlü mesaj gönderimi ve DIO mesajı dağılımı .....	19
Şekil 2.3. Yukarı yönlü mesaj gönderimi ve DAO mesajı dağılımı .....	20
Şekil 2.4. DIS mesajı dağılımı .....	21
Şekil 2.5. a) Depolamayan (non-storing) mod b) Depolayan (storing) mod .....	22
Şekil 2.6. Farklı amaç fonksiyonları ve yönlendirme metriklerine göre kapasite .....	27
Şekil 2.7. Farklı paket boyutlarında yönlendirme yolunun kapasite değişimi.....	28
Şekil 2.8. Farklı amaç fonksiyonlarına göre enerji tüketimi .....	28
Şekil 2.9. Farklı amaç fonksiyonlarına göre enerji verimliliği.....	29
Şekil 2.10. Yönlendirme yolunun uçtan uca gecikmesi .....	29
Şekil 3.1. Çerçeve, düğüm ve bağlantı temsili .....	30
Şekil 3.2. Veri gönderimi yapacak düğüme ait çerçeve yapısı.....	30
Şekil 3.3. Veri alımı yapacak düğüme ait çerçeve yapısı.....	31
Şekil 3.4. Uyku modunda olan düğüme ait çerçeve yapısı.....	31
Şekil 3.5. CR-GreedyRPL için tasarlanan örnek çizge .....	36
Şekil 3.6. Farklı SNR değerleri altında enerji tüketimi .....	41
Şekil 3.7. Farklı SNR değerleri altında enerji verimliliği.....	42
Şekil 3.8. Farklı frekans sayıları altında enerji tüketimi.....	42
Şekil 3.9. Farklı frekans sayıları altında enerji verimliliği .....	43
Şekil 3.10. Akış modeli için örnek topoloji (Trevisan, 2011) .....	43
Şekil 4.1. SNR değerlerinin değişiminin yol kapasitesine etkisi .....	49
Şekil 4.2. Frekans sayısı değişiminin kapasiteye etkisi .....	49
Şekil 4.3. Frekans sayısı değişiminin enerji tüketimine etkisi.....	50
Şekil 4.4. Frekans sayısı değişiminin enerji verimliliğine etkisi .....	51

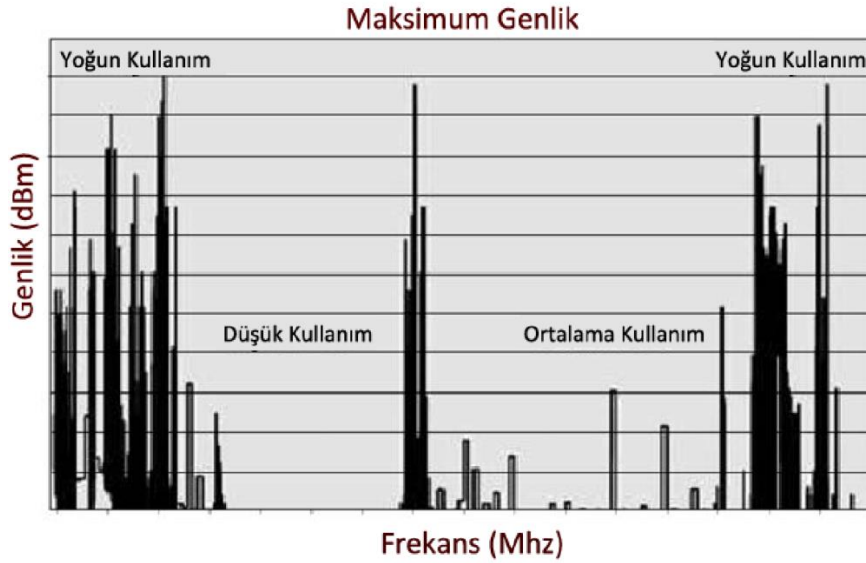
Şekil 4.5. Birim frekans geçiş gecikmesinin yol kapasitesine etkisi .....	51
Şekil 4.6. Birim frekans geçiş gecikmesinin enerji tüketimine etkisi.....	52
Şekil 4.7. Birim frekans geçiş gecikmesinin enerji verimliliğine etkisi .....	53
Şekil 4.8. Zaman dilimi değerlerinin yol kapasitesine etkisi.....	53
Şekil 4.9. Zaman dilimi değerlerinin enerji tüketimine etkisi .....	54
Şekil 4.10. Zaman dilimi değerlerinin enerji verimliliğine etkisi.....	54

## TABLULAR DİZİNİ

Tablo 2.1. Benzetim parametreleri .....	26
Tablo 2.2. Benzetimin temel özellikleri .....	26
Tablo 3.1. Benzetim parametreleri .....	40
Tablo 4.1. Frekans matrisi çizelgesi .....	46
Tablo 4.2. Kapasite matrisi çizelgesi.....	47
Tablo 4.3. Olasılık matrisi çizelgesi .....	47
Tablo 4.4. Benzetim parametreleri .....	48

# 1. GİRİŞ

Son yıllarda kablosuz ağ teknolojilerinin ve bu teknolojiyi kullanan cihaz sayısının artması ile, kablosuz ağ kullanımında da büyük artış meydana gelmiştir (Akyıldız, et al., 2006). Sosyal paylaşım ağları, çoklu veri ve dosya paylaşım ihtiyacı, yüksek miktarda veri transferi gibi ağ uygulamalarının artışı, kablosuz ağlarda yayın yapılan spektruma daha fazla ihtiyaç duyulmasına sebep olmaktadır (Shiang and Van Der Schaar, 2008). Kablosuz ağ sistemleri için frekans bandı yani spektrum, bu bantların verimsiz kullanılması sebebiyle önemli bir iletişim kaynağıdır. Fakat bu kaynak, lisanslı kullanıcılar tarafından etkin ve verimli bir şekilde kullanılmamaktadır. Federal İletişim Komisyonu (FCC) ölçümlerine göre, mevcut spektrumlarının yaklaşık %15 ile %90'ı arası aktif olarak kullanılmaktadır (Akyıldız, vd., 2006). Bir başka deyişle, tahsis edilen spektrum bandının büyük bir bölümü atıl durumda kalmakta yani kullanılmamaktadır. Birçok ülkede spektrum bantlarının lisanslı kullanıcılara ücretli olarak tahsisiyle birlikte, spektrum yetersizliği sorunu ön plana çıkmıştır. Şekil 1.1, atanan spektrum bantlarının verimsiz kullanımını göstermektedir.



Şekil 1.1. Spektrum bandı kullanımı (Akyıldız, et al., 2006)

Spektrum bantlarındaki verimsiz kullanım, kablosuz iletişim ağlarında yeni bir paradigmayı doğurmuştur. Bu yeni paradigma, atıl durumdaki frekans bantlarını fırsatçı bir şekilde kullanılması ve gerektiğinde diğer kullanıcılar ile paylaşılmasına olanak sağlayan Bilişsel Radyo Ağları (Cognitive Radio Networks-CRN) teknolojisidir.

CRN tanımsal olarak, birincil kullanıcıların yani spektrum bandını kullanma yetkisine sahip lisanslı kullanıcıların, bandı kullanmadıkları zaman aralıklarında ağı, ikincil yani lisanslı olmayan kullanıcıların kullanmasına olanak sağlayan yapıyı ifade etmektedir. CRN, spektrum bandındaki boşlukların, dinamik olarak algılanıp erişilmesine imkan vermektedir. Aynı zamanda CRN, çalıştığı ortam parametreleri ve diğer parametreleri değiştirebilme yeteneğine sahip ağ teknolojisidir.

Lisanslı ve lisanssız kullanıcılardan oluşan Bilişsel Radyo Ağları yapısında spektrum bandının kullanım hakkı önceliğine göre kullanıcılar sınıflandırılmıştır. Lisanslı spektrum bantlarında, belirlenmiş lisansa sahip olan ve kendisine tahsis edilen bant üzerinden iletişim yapma hakkına sahip lisanslı kullanıcılar birincil kullanıcı (Primary User-PU) olarak adlandırılmaktadır. Bu tür kullanıcılar, önceliği en yüksek kullanıcı sınıfı olarak da nitelendirilebilmektedir. Ayrıca ağ üzerinde, lisanslı bandı kullanarak iletişim yapma hakkına sahiptir. Herhangi bir kullanıcı, bu bant üzerinde birincil kullanıcının iletişim ve kullanım önceliğini değiştirmemektedir. Bunun yanı sıra, PU'lar tarafından kanal veya bir başka deyişle spektrum bandının kullanılmadığı zaman aralıklarında kanala erişerek kullanım hakkı elde eden kullanıcılar lisanssız ikincil kullanıcı (Secondary User-SU) olarak adlandırılmaktadır. İkincil kullanıcılar, birincil kullanıcılar spektrum bandını kullanmaya başladığı zaman aralığında yeni bir iletişim kanalı bularak bandı terk etmeli ve lisanslı, birincil kullanıcının iletişimini bozmamalıdır.

Bilişsel Radyo Ağları'nda asıl hedef, farklı ağ mimarisi ve spektrum bantlarına erişim noktasında gelenekselden daha farklı bir ağ yapısı meydana getirerek veya var olan ağ yapısını verimli bir şekilde kullanarak yüksek bant genişliğine ulaşmaktır. Bu yüzden CRN yapısı, bilişsel radyo (CR) kullanıcılarına bazı sorumluluklar yüklemektedir. Bu sorumluluklar bilişsel radyo kullanıcılarının karakteristik özellikleri olarak tanımlanmaktadır. Bunlar aşağıdaki gibi özetlenmektedir (Akyıldız, et al., 2006):

- Spektrum sezme (*Spectrum sensing*): Atıl durumda olan, birincil kullanıcı tarafından kullanılmayan spektrum bandının fark edilmesi.
- Spektrum yönetimi (*Spectrum management*): İletişimi başlatmak için fark edilen ve kullanıma en uygun olan spektrumun belirlenmesi ve seçilmesi.

- Spektrum hareketliliği (*Spectrum mobility*): Gerektiği durumlarda, kullanımına karar verilen spektrumun değiştirilmesi.
- Spektrum paylaşımı (*Spectrum sharing*): Bir arada bulunan ikincil kullanıcılar arasında spektrumun paylaşılması.

CRN, dinamik spektrum erişimli (DSA) ağlar için tasarlanmış ağ teknolojisidir (Nejatian, et al., 2013). CRN’de ikincil kullanıcılar, lisans satın alarak kullanım önceliği almış kullanıcılar ile çalışmaktadır. Çevresi ile sürekli bir etkileşim içerisinde olan CR kullanıcıları, anlık olarak değiştirdikleri parametreler ile birlikte spektrum boşluklarını sezer ve sezgi sonucu elde ettiği verileri değerlendirerek spektrumlara erişim sağlarlar. Bu yönüyle bakıldığında CR kullanıcılarının karakteristik özellikleri göze çarpmaktadır (Park, et al. vd., 2013).

CRN’de, birincil kullanıcılar ağ üzerinde iletişim hakkına satın aldıkları lisansın dolayısıyla sahiptir. Lisans çerçevesindeki spektrum aralıklarını iletişim için kullanırlar. Bu lisanslı kullanıcılardan oluşan birincil ağ yapısı, merkezi bir dağılıma sahip olmak zorunda olmadığı için dağıtılmış ve tasarsız şekilde de olabilmektedir. İkincil kullanıcı olan lisanssız CR kullanıcılar, birincil kullanıcılar tarafından spektrumun kullanılmadığı durumlarda fırsatçı bir şekilde spektruma erişim sağlayarak haberleşirler. Fakat bu spektrum erişimi için literatürde doğrudan belirlenmiş bir yol bulunmamaktadır. Farklı erişim teknikleri kullanılarak erişim sağlanır, bunlar fırsatçı spektrum erişimi ve eş zamanlı erişim tekniğidir (Chen, et al., 2011). Fırsatçı spektrum erişimi, CR kullanıcılarının, spektrum boşluklarını sezmesi, algılaması ve buna bağlı olarak parametrelerini yapılandırması ile gerçekleşir. İletişim bu yolla başlar. Lisanslı kullanıcı ağda tekrar aktif olduğunda, CR kullanıcıları ağı terk eder. Eş zamanlı erişim tekniğinde ise CR kullanıcısı lisanslı kullanıcı ile bir arada bulunur ve sinyali bozmayacak tolerans eşik değeri aşılmayana kadar iletişim yapar (Mitola, 2002).

Geleneksel ağ modelleri için önerilen yönlendirme algoritmaları ve modelleri pek çok bakımdan ele alınarak reaktif, proaktif, hibrit gibi sınıflara ayrılmıştır (Bhatia, et al., 2016). Bu ağlardaki yönlendirme gerçeklemeleri, teknolojik gelişmelerin ağ üzerindeki etkisi ve yükü dolayısıyla yerini farklı ağ teknolojileri kullanılarak yönlendirmeye bırakmıştır. CR teknolojisi de bu yeni araştırma alanlarından birisidir. CRN üzerinde yönlendirme, ağdaki kullanıcıların ve CR ağının karakteristik

özellikleri düşünülerek tasarlanmaktadır. Tasarlanan yönlendirme modelleri ağ parametrelerinin ve spektrum mevcudiyetinin sürekli değişmesinden dolayı yönlendirme zorlukları ile karşı karşıya kalmaktadır. Genel çerçeveden bakıldığında, boştaki spektrumun kullanımı ağ üzerindeki yönlendirme performansını artırabilmektedir. Fakat yönlendirme yolunun seçilmesinde atlama sayısı, tıkanıklık, gecikme gibi geleneksel ölçütler kullanılması, CRN için tam anlamıyla verimli olmamaktadır. Ali ve diğerleri yaptıkları çalışmada, CRN üzerinde bu yönlendirme zorluklarını dört ana başlık altında sınıflandırmışlardır (Ali, et al., 2011):

- Link kullanılabilirliği (*Link Availability*), CR kullanıcılarının spektrum boşluklarını algılaması ve ağı kullanacak durumda olması nedeniyle zamana ve spektruma göre müsaitlik durumuna bağlıdır.
- Spektrum sağrlık sorunu (*Deafness Problem*), CR kullanıcıların mevcut spektrumu kısıtlı süre zarfında kullanacak oluşu nedeniyle, gerektiğinde kullanılacak mevcut kanallar arasında geçiş yapması gerekmektedir. Bu da özellikle frekans değiştirme gecikmesi nedeniyle, sağrlık sorununa ve bu sorun da yönlendirmede gecikmeye sebep olmaktadır.
- Tek yönlülük (*Unidirectional Links*), alıcı ile aynı kanalı kullanmak zorunda olan CR kullanıcıları için, aynı kanalda kalabilme garantisi olmadığından dolayı yaşanan zorluktur. Yani gönderim anına kadar aynı kanalda kalınacağına garanti bulunmamaktadır.
- Heterojenlik (*Heterogeneous*), CR kullanıcıları, sezip karar verdiği spektrumu kullanırken farklı tür kablosuz ağlar arasında geçiş yapmak durumundadır. Kanalları kısa süre kullanabilen CR kullanıcısı, bu ağlar arasında geçiş yapabilmelidir.

Cesana ve diğerleri (2011) yaptıkları çalışmada, yönlendirme zorluklarını spektrum farkındalığı, kaliteli yönlendirme oluşturulması ve bu rotanın bakım/onarımı şeklinde tanımlanmıştır. Yine aynı çalışmada CRN için yönlendirme algoritmalarını spektrum mevcudiyetinin durumuna, tam spektrum bilgisi veya yerel spektrum bilgisi, göre iki ana kategoride sınıflandırılmıştır.

CRN yapısının kendi özellikleri gereği, yönlendirme algoritmaları gelenekselden daha farklı olarak tasarlanmıştır. CR kullanıcılarının spektrum sezme,

karar verme ve hareketlilik gibi karakteristik özellikleri, CRN'deki yönlendirme davranışında belirleyici rol oynamaktadır.

Nesnelerin İnterneti (IoT), internete bağlanabilen, birbirleri ile veri alışverişinde bulunan nesnelerin oluşturduğu ağ yapısıdır. Bu kavram ilk olarak 1999 yılında Kevin Ashton tarafından kullanılmıştır (Palattella, et al., 2012). Gündelik yaşamın bir parçası haline gelen internet ile beraber, internete bağlanan cihaz sayısı da artmıştır. USNIC verilerine göre, internete bağlanan bu cihazların 2025 yılına gelinceye kadar günlük hayatın her alanında yer alacağı öngörülmektedir (Yick, et al., 2008). Nesnelerin, insan müdahalesi olmadan sistemi algıladığı, hedefe yönlendirdiği, veriyi işlediği ve depoladığı işlevler, IoT yapısının temelini oluşturmaktadır. Bu kavram yalnızca internete bağlanan cihazlardan ve bu cihazların haberleşmesinden oluşmamaktadır. Algılayıcı ve tanımlayıcı cihazlardan oluşan yapıların bilgi üretmesi de bu kavramın içerisindedir. Bütün bu tanımlamalar ile IoT, cihazların mevcut ve alışlagelmiş şekliinden ve işleyişinden daha akıllı hale getirilmesi şeklinde tanımlanabilir.

Teknolojik gelişmeler, her şeyin ve herkesin iletişim halinde olacağı gerçeğini ortaya koymaktadır. Birbirlerine bağlı bir şekilde, her yerde ve her zaman iletişim kurma olanağı sağlayan internet yapısı, nesneler ve insanlar arasında sonsuz bir iletişimi mümkün kılar. Cihazlar verinin algılanması, veri transferi, verinin işlenmesi gibi işlevler ile günlük hayatta rol alırlar. Bu cihazlar, günlük ev aletlerinden, algılayıcılara kadar birçok alanda yer almaktadır. Akıllı sensör cihazları, fiziksel nesnelere ve hareketleri algılar. Algılanan hareket ve nesnelere, internet ortamında aktarılır ve işlenir. Akıllı cihazların birçoğu algılayabilen, hesaplama yapabilen ve internet ortamına aktarabilen cihazlardır. Geniş kapsamlı olanaklara sahip akıllı cihazlar, sağlık, endüstri, tarım gibi çeşitli sektörlerde aktif olarak kullanılmakta ve işleri kolaylaştırıcı rol üstlenmektedir. Farklı sektörlerde göre IoT uygulama alanları Şekil 1.2'de gösterilmektedir.



Şekil 1.2. Nesnelerin interneti uygulama alanları (Ala Al-Fuqaha, et al., 2015)

Standart bilgisayar ağları içerisinde yer alan protokol ve standartlar gibi, IoT yapısı da kendine özgü mimariye sahiptir. TCP/IP mimarisine benzer şekilde birçok mimari önerilse de günümüzde temel olarak varsayılan 5 katmanlı mimari kullanılmaktadır (Ala Al-Fuqaha, et al., 2015). İlk olarak uygulama, ağ ve algılama olmak üzere üç temel katmandan oluşan mimari önerilse de farklı soyutlamalar ile katman sayısı artırılmaya ve katmanlar arası görev ayrımı netleştirilmeye çalışılmıştır. IoT için başlangıçtan bugüne önerilen katmanlı mimari yapıları Şekil 1.3'te gösterilmiştir.



Şekil 1.3. Nesnelerin interneti için önerilen mimariler (Ala Al-Fuqaha, et al., 2015)

Kablosuz Sensör Ağları (WSN), kapsam ve işlev bakımından IoT ağ paradigmasına bağlıdır. WSN, IoT ile aynı şekilde, kablosuz bir ağ ortamında birbirlerine bağlanmış birçok algılayıcı düğümden oluşan ağ yapısıdır. RFID cihazları, sıcaklık, nem gibi fiziksel ölçüm algılayıcıları gibi cihazlar WSN yapısını oluşturmaktadır. Bu fiziksel cihazlar, ortamdan topladıkları verileri birbirlerine veya ortak bir merkeze göndererek haberleşmektedir. IoT gibi günlük hayatın pek çok alanında kendine yer bulmaktadır. Bütün bu tanımlamalar ile birlikte WSN, IoT ağların sunduğu imkan ve özellikler bakımından eşdeğer olarak nitelendirilebilir.

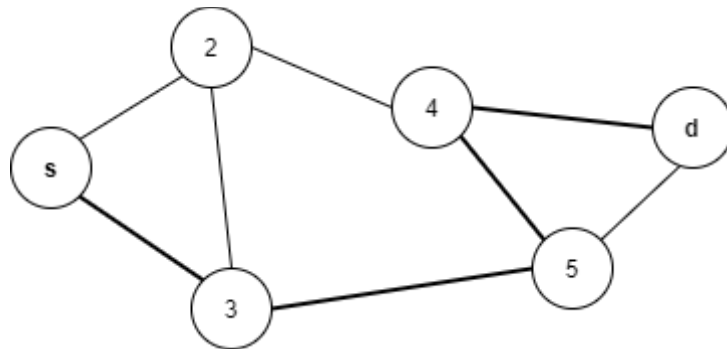
Mikro boyuttaki sensör düğümlerinin üretilmeye başlaması ve teknolojik devrimlerin kapsamının genişlemesi ile birlikte, WSN teknolojisi daha uygulanabilir hale gelmiştir (Parvin and Rahim, 2008). Düşük güç kapasiteli, düşük maliyetli cihazlardan oluşan bu ağlarda haberleşmenin mümkün olması, sensör cihazlarının karakteristik özellikleri ve ağ yapısına göre iletişim standartlarının belirlenmesine bağlıdır. Çok sayıda sensör düğümünden oluşan WSN, çevresel değişimlerin algılanması, işlenmesi ve iletilmesi ile haberleşme ekosistemini oluşturur. Ağı oluşturan cihazların yapısı ve standart iletişim cihazlarından farklı birtakım özellikler, WSN ve IoT alanlarında iletişimin daha farklı ele alınmasına neden olmuştur. İletişimdeki önemli aşamalardan olan yönlendirme problemi de bu ağlar için önemli bir araştırma alanı olmuştur. Cihazların ve çalıştıkları ağların düşük güç ve kayıplı oluşu, yönlendirme alanındaki zorlukların başında gelmektedir.

Yönlendirme protokolü, paketlerin veya mesaj nesnelерinin bir düğümden diğer bir düğüme iletilmesinden sorumludur. Bu tanımdaki düğüm kavramı, sensör ağlarını veya iletişim ağlarını oluşturan bağımlı/bağımsız cihazlar olarak tanımlanabilir. Bütün iletişim ağlarında olduğu gibi, önerilen ve geliştirilen yönlendirme protokolü, istenen ölçütler bakımından en iyi yönlendirme yolunu elde etmek ile sorumludur. Bhattacharyya ve diğerleri, yaptıkları sınıflandırma çalışmasında WSN'deki yönlendirme protokollerini reaktif ve proaktif olarak iki gruba ayırmıştır. Proaktif protokollerde yönlendirme rotası belirli periyotlar ile farklı kontrol mesajları kullanılarak güncellenmektedir. Reaktif protokoller ise yönlendirme rotası oluşturmanın gerekli olduğu durumlarda iletim yolunu kuracak şekilde çalışmaktadır (Bhattacharyya, et al., 2018).

Düşük güç ve kayıplı olan sensör ağlarında, verinin iletimi için pek çok yönlendirme protokolü önerilmiştir. Önerilen protokoller, yönlendirme zorluklarını

dikkate alarak belirlenen başarı ölçütlerinde bir noktadan diğerine iletişim sağlamaktadır. Bunun bir örneği, IETF ROLL çalışma grubu tarafından, düşük güç ve kayıplı ağlar (LLN) için tasarlanmış 6LoWPAN protokolü olan RPL'dir (Winter, et al., 2012). RPL, iletim ağı oluşturduğunda en iyi rotayı bulmaya odaklı yönlendirme protokolüdür (Bhattacharyya, et al., 2018).

Yönlendirme, tanımsal olarak, bir ağıdaki veya birden fazla ağ içerisinde iletilmek istenen veri ya da bilgiler için iletim trafiğinin belirlenmesidir. Bu tanımdaki iletim trafiği belirlenme süreci, yol seçimi baz alınarak gerçekleşir. Standart iletişim ağları ve haberleşmede kullanılan cihazlar belirli bir referans model içerisinde sınıflandırılmaktadır. OSI (Open Systems Interconnection) modeli olarak adlandırılan referans yapı, ISO (International Organization for Standardization) çalışma grubu tarafından önerilmiştir. Önerilen modele göre yönlendirme, üçüncü katman cihazları tarafından gerçekleştirilen bir işlemdir. Bütün bu tanımlamalar eşliğinde yönlendirme kavramı, belirlenen kaynak veya başlangıç noktasından yine belirlenen bitiş veya hedef noktaya gitmek için yapılan yol seçimi olarak tanımlanabilir. Bu seçim yapılırken istenen performans ve başarı ölçütleri dikkate alınır. Bu başarı ölçütleri yüksek yol kapasitesi, düşük enerji tüketimi, yüksek enerji verimliliği gibi kısıtlar olabilir. Yönlendirmede amaç ise kaynaktan hedefe bu kısıtlar çerçevesinde bir yol oluşturmaktır. Şekil 1.4'te herhangi bir performans ölçütüne bağlı olmaksızın örnek bir yönlendirme yolu verilmiştir. *s* ile isimlendirilmiş düğüm başlangıç noktası; *d* ile isimlendirilmiş düğüm hedef düğüm olacak şekilde belirlenmiş, buna bağlı olarak yönlendirme yolu  $s - 3 - 5 - 4 - d$  olarak oluşturulmuştur.



Şekil 1.4. Örnek bir yönlendirme yolu

CRN ve IoT için literatürde birbirleriyle bağlantılı veya bağlantısız birçok yönlendirme algoritması bulunmaktadır. Bu algoritmalar, yönlendirme yolunun seçiminde belirlenen performans ve başarı ölçütlerine göre sınıflandırılmaktadır.

Geleneksel bilgisayar ve iletişim ağlarında yönlendirme için çeşitli protokol ve algoritmalar geliştirilmiştir. Günümüz standart bilgisayar ağları, benzersiz olarak tanımlanan ve atanan Versiyon 4 İnternet Protokolü (IPv4) ile adreslenmektedir. Gelişen ve büyüyen internet ağ yapısı, internete bağlanan cihaz sayısındaki artış gibi pek çok faktör, IPv4 protokolünün yetersizliği sorununu da beraberinde getirmiştir. IoT ve WSN kavramlarının fiziksel olarak kullanımı ile birlikte IP yetersizliği sorununun önüne geçebilmek için Versiyon 6 İnternet Protokolü (IPv6) doğmuştur. IPv4'ten daha büyük bir adresleme yapısına sahip olan IPv6 kullanan ağlar için de yönlendirme kavramı, önemli bir araştırma konusu haline gelmiştir. Bununla birlikte, IPv6 ağları ve bu ağları kullanan cihazlar için de yönlendirme protokolleri geliştirilmiştir.

### **1.1. Önceki Çalışmalar**

Bu alt bölümde, CRN ve IoT için literatürde yer alan yönlendirme çalışmaları aktarılacaktır.

Cai ve diğerleri (2015), kurgulanan statik bilişsel radyo ağı topolojisinde,  $\sigma$  boyuta sahip paketi, belirlenen kaynak düğümden, varış noktasındaki düğüme iletim gecikmesini minimize ederek iletecek şekilde var olan algoritmalar üzerinde çalışma yapmıştır. Yapılan benzetimde D-SP-R ve A-Opp-R algoritmalarının yanı sıra, TS-Opp-R ve TS-Opp-RP algoritmalarına ait performans çıktıları dikkate alınmıştır. Bu dört algoritma sonucunda, ortalama gecikme performansı için, örneğin kanal 2'den kanal 10'a geçme maliyeti, A-Opp-R algoritması kullanıldığında diğerlerine göre nispeten kötü sonuçlar vermiştir. Yaptıkları çalışmada elde ettikleri verileri grafiksel verilere dökerek, kanal sayısındaki artışla birlikte ortalama gecikme maliyetinin baştaki küçük düşüğe rağmen, belirli bir noktadan sonra arttığını göstermişlerdir. Talay ve Altılar (2013) ise NS-2 simülatörü kullanarak benzetimini yaptıkları çalışmada, SAR üzerinde yönlendirme yolu seçimine iyileştirmeyi amaçlamışlardır. Simülasyonda sabit sayıda kanal kullanılmış, birincil kullanıcı için ON/OFF olmak üzere iki durum varsayılmıştır. Farklı parametreler eşliğinde, düğüm sayısının uçtan uca gecikme, iş verimi/kapasite üzerinde etkili olduğu sonucu ortaya çıkmıştır. Bu denemeler, SAR'ın uçtan uca gecikmeyi aza indirdiği ve kapasiteyi maksimize ettiğini göstermiştir. Cesana ve diğerleri (2011) yaptıkları çalışmada, yönlendirmeyi tam ve yerel frekans bilgisine sahip olup olmama durumuna göre sınıflandırmışlardır. Frekans değiştirme gecikmesinin, yani bir kanaldan başka bir kanala geçme maliyetini sabit olacağını

varsaymışlardır. Çalışmada yakın kanaldan uzak kanala geçmenin frekans değiştirme gecikmesinde farka neden olmayacağını gösterilmektedir. Yapılan çalışma sonucunda, frekans değiştirme maliyeti bazlı sınıflandırmada kullandıkları ve önerdikleri fonksiyonlara bağlı olarak, frekans değiştirme maliyetini dikkate almamışlardır. Wang ve diğerleri (2018) CSRP olarak adlandırdıkları protokol ile yol seçim kararlılığını sağlamak, veri iletim oranını arttırmak ve gecikme süresini azaltmak amacıyla çalışma yapmışlardır. Simülasyon sonucu, önerilen CSRP'nin uçtan uca gecikme ve veri iletim oranı baz alındığında, TDRP ve PUC-JRCA protokollerinden daha iyi olduğunu göstermişlerdir. CSRP'nin veri iletim oranında TDRP'den; uçtan uca gecikmede PUC-JRCA protokolünden daha iyi sonuç verdiği grafiksel gösterimle sunulmuştur. Yosra ve diğerleri (2016), CO-QORP isimli protokol önermişlerdir. Önerilen protokol yapısı, birincil kullanıcıların girişimini azaltarak ve maliyeti minimize ederek mevcut kanalın niteliklerini iyileştirmektedir. Bütün bunlar neticesinde önerilen yapı, uçtan uca gecikmeyi indirgeyerek bu metrik özelinde daha yüksek performans ile çalışmaktadır. Tasarlanan algoritma, birincil kullanıcı tarafından kullanılmayan kanalın seçilmesi, servis kalitesi (QoS) garanti ederek bir sonraki düğümün seçilmesi ve kanal değiştirme maliyetini azaltarak, verimlilik ve paket iletim oranının maksimize edilmesi optimizasyonları ile çalışmaktadır. Mohammed Hashem ve diğerleri (2017) bilişsel radyo ağlarında yönlendirme sürecinin, merkezi altyapının yokluğu, birincil kullanıcının ani olarak spektrumu kullanmasından kaynaklanan bağlantı hataları gibi zorluğundan dolayı TRP protokolünü geliştirmişlerdir. C-TRP olarak önerdikleri protokol ile birlikte uçtan uca gecikme, paket iletim oranı gibi performans ölçütlerinde başarıyı sağlayarak iletim performansını artırmayı hedeflemişlerdir. Farklı simülasyon parametrelerini kullanarak benzetim yapmışlardır. Bunun sonucunda, önerilen C-TRP protokolünün, diğer iki yönlendirme protokolü olan CTBR ve STOD-RP'ye oranla, performans ölçütleri dikkate alındığında daha başarılı yönlendirme yaptığı ortaya konulmuştur.

İletim yolunun belirlenmesi çeşitli parametreler ile doğrudan veya dolaylı olarak ilişkilidir. IoT/WSN cihazlarının daha önce bahsedilen kısıtlı yeteneklere sahip oluşu, yönlendirmede de farklı yöntemlerin önerilmesine neden olmuştur.

Alvi ve diğerleri (2015), IoMT için çevreci bir yönlendirme protokolü önermişlerdir. Bu algoritma, RPL'nin geliştirilmiş halidir ve gecikme, batarya tüketimi gibi metrikler kullanılarak ebeveyn düğümün seçilmesi şeklinde

çalışmaktadır. Önerilen bu protokol ağ karbon ayak izinin ve enerji tüketiminin minimize edilmesini dikkate almaktadır.

Chang ve diğerleri (2013), var olan RPL algoritması üzerinde geliştirmeler yaparak enerji yönelimli yönlendirme algoritması önermişlerdir. ETX ve diğer enerji metriklerini birleştirerek amaç fonksiyonunu oluşturmuşlardır. Oluşturulan amaç fonksiyonuna göre yönlendirme yolu seçilmektedir. Böylece, yalnızca düğümler üzerinde enerji tüketimi hesabının aksine, bütün ağ genelinde enerji tüketim dengesi oluşturmuşlardır. COOJA network simülatörü kullanarak önerilen şemayı standart RPL algoritması ile karşılaştırmışlardır. Kim ve diğerleri (2016), RPL'deki yük dengeleme ve tıkanıklık problemini ele almışlardır. Yoğun ağ trafiğindeki paket kayıplarının çoğunun tıkanıklıktan kaynaklandığını ve bu paket kayıplarına RPL'de ebeveyn düğüm seçme işleminin neden olduğunu belirtmişlerdir. Bu problemin önüne geçmek için, QU-RPL adı verilen iyileştirilmiş RPL algoritması önerilmiştir. Önerilen bu model ile yönlendirme ağacında trafik yükünü dengeleyerek uçtan uca gecikme ve paket iletim performanslarını artırmayı amaçlamışlardır. Modelin performans artışı, Tiny RPL adı verilen, atlama sayısını esas alan OF0 amaç fonksiyonunu kullanan RPL ile karşılaştırarak göstermişlerdir. Barbato ve diğerleri (2013) kablosuz sensör ağlarında cihazların kısıtlı güce sahip olması nedeniyle enerji verimli bir yönlendirme protokolü önermişlerdir. RPL tabanlı ROEE olarak adlandırılan bu protokolde, birden fazla metrik içeren amaç fonksiyonu kullanmışlardır. Yalnızca atlama sayısı yerine enerji miktarı ve pil tüketimi gibi metrikleri içeren OF oluşturmuşlardır. Önerilen ROEE ile RPL protokolü ağ yaşam süresi, güç tüketimi ve aktif canlı düğüm sayısı bakımından karşılaştırılmıştır. Iova ve diğerleri (2015) toplam enerji tüketimini minimize etmek için enerji farkındalıklı yönlendirme protokolü tasarlamışlardır. Protokol, ağ ömrünü uzatmak için her düğümün aynı miktarda enerji tüketmesi temelinde çalışmaktadır. Enerji darboğaz olayını belirlemeyi ve ağ genelinde enerji tüketimini verimli hale getirmeyi amaçlamışlardır. Enerji dengeli topoloji üretmek için beklenen yaşam süresi (ELT) metriği kullanarak amaç fonksiyonu önermişlerdir. Algoritma, ebeveyn düğüm seçimi esnasında düğümlerin anlık yaşam süresi ve darboğaz yaşam süresi durumlarını dikkate alma prensibi ile çalışmaktadır. Lin ve diğerleri (2015) tarafından basit güç kontrol mekanizması önerilmiştir. Önerilen bu model, her düğümün iletim gücünün, kendi link ve kuyruk kayıplarına göre ayarlaması şeklinde çalışmaktadır. Böylece yük dengeleme problemini çözmeyi amaçlamışlardır.

Oluşturulan amaç fonksiyonu kuyruktaki ve link üzerindeki enerji kayıplarını dikkate almaktadır. Böylece yönlendirme ağacını ağdaki trafik yüküne göre oluşturarak paket iletim performansını artırmışlardır. Boubekur ve diğerleri (2015), RPL algoritmasının rota kararsızlığı gibi verimsizliğinin önemli paket kayıplarına sebep olmasından dolayı, yeni bir yaklaşım geliştirilmiştir. RPL’de var olan sorunların çözümü, genellikle kullanılan fonksiyon ve metriklerin geliştirilmesine dayanmaktadır. RPL’deki rota kararsızlığına çözüm bulmak amacıyla bir düğümün, ağaç kurulumu sırasında maksimum çocuk sayısının kısıtlanması temelinde çözüm geliştirilmiştir. RPL algoritmasındaki kontrol mesajları kullanılmıştır ancak yeni model, bağlantı kalitesi metriğinden bağımsız olarak çalışmaktadır. Yapılan benzetimler sonucunda enerji tüketiminde, gecikme ve iletim oranında iyileştirmeler gösterilmiştir. Zhao ve diğerleri (2016), güvenilirlikten ödün vermeden enerji verimliliği ve veri iletimi sağlayan model önermiştir. LLN’lerde hem güvenilirlik hem enerji tasarrufu gibi önemli parametrelerde başarı elde etmenin zorluğu çalışmanın motivasyonudur. RPL algoritmasındaki rota keşfi olarak adlandırılan aşağı yönlü yönlendirme aşaması için geliştirme yapılmıştır. Rota keşfinde bütün düğümlerin değil de yalnızca bir düğümün alt kümesinin kullanılması önerilmiştir. Bu yaklaşım, enerji verimliliği noktasında anahtar olmuştur. Mobil düğümün, mevcut ebeveyn ile bağlantısını kesmeden önce yeni bir düğüm seçmesi temelinde geliştirilmiş RPL algoritması, Hoghooghi ve Javidan (2020) tarafından önerilmiştir. ETX, alınan sinyal gücü, kalan enerji gibi metriklerle yeni ebeveyn seçilmektedir. Önerilen model ve bu metrikler ile hareketlilik tespiti ve güvenilir ebeveyn seçimi yapmayı amaçlamışlardır. Algoritma, ebeveyn düğümlerin kontrol paketleri ile ağdaki bağlantıyı kontrol etmesi prensibi ile çalışmaktadır. Bağlantı zayıfladığında, hareketli düğüm ebeveyn düğüm tarafından bilgilendirilmekte ve enerji tüketimleri ve ağ yükü buna bağlı olarak azaltılmaktadır. Her düğümün ebeveyn düğümleri saklamak için bir bellek alanı bulunmaktadır. Ebeveyn düğümler, atanan öncelik durumlarına göre belirlenmekte ve kalıcı bağlantı seçilecek şekilde işlem devam etmektedir. Wei ve diğerleri (2013) yaptıkları çalışmada, DTN isimli, dinamik ve topoloji değişimlerini göz önünde bulunduran bir algoritma önermişlerdir. Bu algoritma, hareketliliğe sahip, bağlantı kesilmeleri ve bağlantı hatalarının daha sık yaşandığı ağlar için geliştirilmiştir. Aynı zamanda gecikme de optimize edilmektedir. Yönlendirme hesaplaması yaparak her bir adımda optimal yolu seçmeyi amaçlamaktadır. İletişim devamlılığı yerine alternatif yol da bulmaktadır.

Shah ve Rabaey (2002), yayınladıkları çalışmada, yönlendirme/yol seçimi yapılırken düşük enerji kullanımının her zaman ağ ömrü ve bağlantı sürekliliğinin sağlanmasında ölçüt olamayabileceğini belirtmişlerdir. Bu fikirle birlikte, zaman zaman optimal olmayan yolun seçimine de bağlı olarak enerji kullanımı ve ağ sürekliliği sağlayacak çalışma gerçekleştirmişlerdir. Bir başka deyişle, düğümlerin enerjisiyi optimal bir şekilde kullanması, ağ bağlantısında kopmalar yaşanmaması amaçlanmıştır.

## **1.2. Motivasyon ve Katkı**

Günümüzde, internet teknolojilerinin gelişen ve değişen ihtiyaçlara cevap vermek üzere her geçen gün farklılaşması ve yaygınlaşmasıyla pek çok alanda bu kavram somut olarak kullanılmaya başlamıştır. Özellikle internete bağlanan cihaz sayısındaki artış ile, geleneksel internet ve ağ yapıları değişime uğramıştır. Yeni kavramlar, yeni protokoller ortaya çıkmıştır. Bunlarla birlikte yaşanan bu hızlı değişim beraberinde bu alanlarda yeni problemler ve buna bağlı olarak da yeni iyileştirmeler getirmiştir.

Geleneksel bilgisayar ağlarında iletişim, ağı kullanmak için belirli bir ücret karşılığı lisans satın alan kullanıcıların birbirleri ile haberleşmesine dayanmaktadır. Lisanslı kullanıcılar ağ üzerinde belirlenen spektrum bantlarını kullanmaktadır. Artan kullanıcı sayısına bağlı olarak lisanslı kullanıcıların da sayısının artışı bilgisayar ağlarında, CRN isimli paradigmanın ortaya atılmasına sebep olmuştur. Fakat kullanıma sunulmak istenen bu CRN paradigması, ağ kullanıcılarının karakteristik özelliklerine bağlı olarak zorluklar barındırmaktadır. Bilişsel radyo kullanıcılarının ve bilişsel radyo ağlarının iletişim kalitesi için haberleşmenin birtakım zorluk ve problemlerden arındırılması gerekmektedir. Bu yüzden literatürde, CRN’de iletişim ve haberleşmenin en önemli aşamalarından birisi olan yönlendirme işlevi için birçok farklı çalışma önerilmiş ve incelenmiştir.

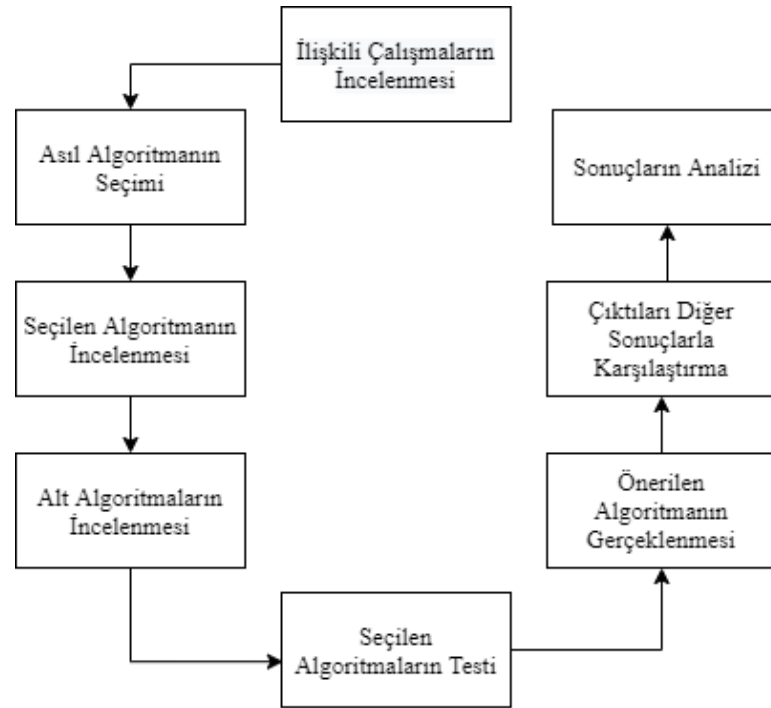
IoT, internet ortamının yaygınlaşması sonucu haberleşen cihaz sayısının artışına bağlı olarak birçok alanda etkisini gösteren yapıdır. Hemen hemen her sektör işleyişini internet ve internet kullanarak birbirleriyle haberleşen cihazlar üzerinden gerçekleştirmektedir. Artan talep, talebe bağlı olarak standart iletişim ağ ve protokollerinin ve yapılarının yetersiz kalışına sebep olmaktadır. Bunun en yaygın örneklerinden birisi internet adresleme protokolü olan IP yetersizliği problemidir. Bu

yüzden adresleme yapısı deęişmiş, daha büyük ve fazla talebe cevap vermek üzere internet adresleme protokolü kullanıma alınmıştır. Bilgisayar ağlarının temeli olarak gösterilebilecek haberleşme, deęişen ve gelişen birçok parametre ile doğrudan ilişkilidir. Bilişsel radyo ağlarında ve dięer ağlarda olduęu gibi, IoT ağlarında da iletişimin en önemli aşaması olan yönlendirme üzerine literatürde birçok çalışma ve model bulunmaktadır.

Mevcut cihazlar tek bir antene yani tek bir alıcı verici duruma sahip olacak şekilde tasarlanmıştır. Her cihaz eş zamanlı olarak tek bir frekans bandını kullanabilmekte, yeni bir frekansa geçememektedir. Farklı frekans bantlarına geçiş yapamayan bu cihazlar düşünöldüğünde, var olan IoT yönlendirme çalışmalarında frekans deęiştirme dikkate alınmamıştır. Örneęin  $f1$  frekansından  $f5$  frekansına geçiş için senaryolar düşünölmemiştir. Buna baęlı olarak frekans geçişleri sabit olarak kabul edilmiştir. Ancak, CRN yapısı gereęi, geniş frekans aralıklarında çalışması düşünölmek üzere tasarlanmıştır. Askeriye gibi alanlarda frekans deęiştirmenin ihmal edilemeyecek düzeyde oluşu, frekans geçiş senaryolarının dikkate alınmasını zorunlu hale getirmiştir. Bütün bunlarla birlikte, frekans deęiştirme kavramı bir cihazın bulunduęu frekans bandından başka bir frekans bandına geçiş yapabilmesini ifade etmektedir. Frekans geçişini ensasında donanımsal bir anahtarlama gecikmesi ve buna baęlı olarak da enerji tüketimi meydana gelmektedir (Bayhan ve Alagöz, 2012).

Bu tez kapsamında, IoT ağları için CRN entegrasyonu yapılması amaçlanmaktadır. IoT ağlarında CR cihazların var olması temeliyle yönlendirme çalışması sunulması, ana motivasyonu oluşturmaktadır. CR düęümler kullanılabilir durumdaki frekans bantlarını algılar, bu frekanslara geçiş yapar ve kendilerine ayrılan zaman diliminde bant üzerinde iletişim yapar. Frekans geçiş maliyeti, en uygun durumdaki frekansa geçişe doğrudan baęlıdır. IoT ağlarında mevcut yönlendirme protokolleri, yönlendirmede frekans geçişini dikkate almamaktadır. Yönlendirme yolu, frekans geçiş maliyetine bakılmaksızın oluşturulur. Bu bağlamda ilk olarak var olan ve kullanılan IPv6 yönlendirme algoritması olan Düşük Güç ve Kayıplı Ağlar için Yönlendirme Protokolü (RPL), belirlenen geliştirme ortamında farklı parametre ve senaryolar ile gerçekleştirilmiştir. Bu gerçekleştirme sonucunda herhangi bir ekleme ve deęiştirme yapılmaksızın, benzetimi yapılan RPL algoritmasının sonuçları enerji kullanımı, enerji verimlilięi, gecikme ve kapasite bakımından deęerlendirilmiştir. Bu benzetimin temel amacı, üretilen ve algoritmanın çalıştırıldıęı ağ topolojisinde

algoritmanın davranışının incelenmesidir. Daha sonra RPL algoritması revize edilerek, bilişsel radyo özelliği olarak frekans değiştirme maliyeti (*switching cost*) farkındalığı olan, enerji verimli kanal değiştirmeyi dikkate alarak yol seçimi yapılması için algoritma (RPL-Sw) geliştirilmiştir. Ayrıca, bir ağ üzerindeki en düşük kapasiteli yolun en büyük kapasitede olacak şekilde seçilmesi temeline dayanan Darboğaz Tabanlı Yol problemi (Widest Path Algorithm-DTY) yine benzetim ortamında gerçekleştirilmiş, bu algoritmanın da ilgili senaryodaki davranışı gözlemlenmiştir. Bir başka deyişle, üretilen ağ topolojisindeki seçilen yönlendirme yolu incelenmiştir. Daha sonra, yol seçimi aşamasında sıklıkla kullanılan Aç Gözlü Algoritma, CR düğümlerde çalışacak şekilde revize edilmiş, var olan RPL algoritması ile birlikte çalışacak şekilde yeniden tasarlanmıştır. Önerilen model ile, IoT ağlarda yönlendirmenin enerji verimliliği, enerji tüketimi, kapasite, kanal değiştirme maliyeti gibi birçok performans ölçütü bakımından iyileştirilmesi, RPL algoritmasının geliştirilmesi ve CR özelliği katılarak CRN ağlarında da kullanılması amaçlanmaktadır. Tasarlanan bu CR Tabanlı Frekans Değiştirme Farkındalıklı Açgözlü RPL Algoritması (CR-GreedyRPL) belirlenen ölçüt ve kısıtlamalar ile birlikte uygulamaya dökülmüştür. Tez çalışmasında benzetimi yapılan, geliştirilen ve önerilen tüm algoritmalar performans metriklerine bakılarak karşılaştırılmıştır. Şekil 1.5, tez kapsamında yapılan çalışmalar için izlenen adımları gösteren akış şemasıdır.



Şekil 1.5. Tez kapsamında incelenen çalışmalar

Tezin içeriği ve çalışma akış sırası özetlenmiştir. Birinci adımda, tez ile ilişkili genel bir literatür taraması yer almaktadır. İkinci adım, belirlenen ve üzerinde daha sonra revizyon yapılacak olan asıl algoritmanın seçimidir. Tez kapsamında RPL algoritması, asıl algoritma olarak belirlenmiştir. Üçüncü adım, asıl algoritma olan RPL'nin incelenmesini içerir. Dördüncü adım, karşılaştırma yapmak üzere belirlenen diğer algoritmaların belirlenmesi ve açıklanmasını içerir. Beşinci adım, belirlenen tüm algoritmaların benzetiminin yapılması ve istenen performans ölçütlerinde sınanmasını içermektedir. Altıncı adımda, motivasyon katkı bölümünü oluşturan algoritmanın tasarlanması yer almaktadır. Yedinci ve sekizinci adımlar ise tüm algoritma ve önerilen modellerin karşılaştırma ve sonuçların yorumlanması aşamasını göstermektedir.

### **1.3. Organizasyon**

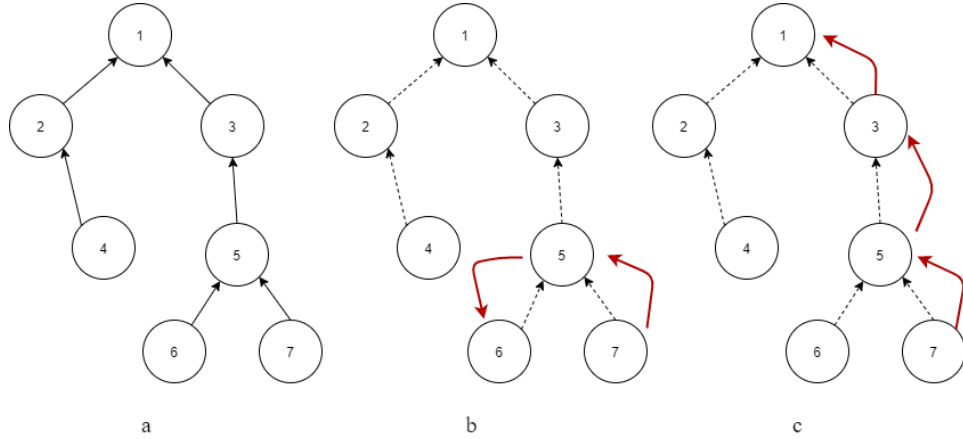
Bölüm 2'de, tez kapsamında üzerinde çalışılan, IPv6 yönlendirme protokollerinden RPL'ye genel bir bakış sunulmaktadır. Ayrıca, RPL algoritmasındaki mesaj yapıları ve amaç fonksiyonları ile birlikte, RPL algoritmasının benzetimine ait parametreler ve sonuçlar aktarılmaktadır. Bölüm 3'te, CR-IoT düğümlerde yönlendirme esnasında gerçekleşen enerji tüketimi ve enerji verimliliği önerilmekte ve modellenmektedir. Ayrıca, RPL algoritması ve tasarlanan diğer algoritmalarda kullanılması amacıyla amaç fonksiyonları sunulmaktadır. Daha sonra, RPL algoritmasının geliştirilmesi ve CR özelliği katılarak CRN ağlarında da kullanılması amacıyla tasarlanan CR özellikli aç gözlü algoritma önerilmektedir. Ayrıca, RPL algoritmasında, frekans değiştirme enerji tüketimini de dikkate alacak şekilde yeni bir amaç fonksiyonu önerilmiş ve amaç fonksiyonu doğrultusunda benzetimi yapılan, RPL-Sw olarak temsil edilen algoritmanın benzetim parametreleri ile benzetim sonuçları sunulmaktadır. Son olarak, Dar Boğaz Tabanlı Yol algoritması açıklanmaktadır. Bölüm 4'te, CR-GreedyRPL algoritması örnek senaryo üzerinden modellenmekte ve diğer bütün algoritmalara ait benzetim parametreleri ve benzetim sonuçları aktarılmaktadır. Ayrıca, algoritmalara ait benzetim sonuçları değerlendirilmektedir. Bölüm 5'te, sonuç ve gelecek çalışmalar için öneriler sunulmaktadır.

## 2. DÜŞÜK GÜÇ VE KAYIPLI AĞLAR İÇİN YÖNLENDİRME PROTOKOLÜ (RPL)

Bu bölümde, tez çalışması kapsamında referans olarak kabul edilen RPL algoritması anlatılacaktır. Tezin sonraki bölümlerindeki çalışmalar, RPL üzerinde yapılan değişikliklerin CRN ile birleştirilmesi, RPL düğümlerine CR özelliği kazandırılması temelinde devam etmektedir.

### 2.1. RPL Algoritması Genel Yapısı

IoT ağlarında kaynak kısıtlı cihazlar için yönlendirme protokolü olan RPL, çalışma şekli ve kullanımı bakımından uzaklık vektör ve kaynak yönlendirme protokolüdür (Abid vd., 2015). IEEE 802.15.4 Fiziksel ve MAC katmanlarında çalışmaktadır. Ağ topolojisindeki değişikliklerin belirlenmesi için düğümlerin çok noktalı mesaj yayını (*broadcast*) yapması nedeniyle, yönlendirme yolu ve yönlendirme tabloları periyodik olarak güncellenmektedir. Bu nedenle proaktif yönlendirme algoritması olarak sınıflandırılmaktadır. RPL algoritması, çok noktadan-noktaya (*MP2P*), noktadan-çok noktaya (*P2MP*) ve noktadan-noktaya (*P2P*) veri iletim akışlarını desteklemektedir (Lim, 2019). Şekil 2.1, RPL algoritmasının farklı iletişim trafiğindeki davranışını, örnek bir ağ yapısı üzerinden göstermektedir.



Şekil 2.1. a) MP2P iletişim b) P2P iletişim c) P2MP iletişim

RPL, ağ üzerinde yer alan düğümlerden oluşan topolojiyi hedefe yönelik yönlü çevrimsiz çizge (DODAG) biçiminde oluşturmaktadır. Yönlü ve çevrim içermeyen ağaç topolojisi olarak temsil edilen DAG ağacının özelleştirilmiş hali olan DODAG yapısında bulunan kök düğümler, ağaç hiyerarşisinin en üstünde yer almaktadır. Ebeveyn-çocuk ilişkisi içeren DODAG yapısında, ebeveyn düğümler ağ cihazlarından

yönlendiricileri temsil etmektedir. Ebeveyn düğümler, bir sonraki atlama yapılacak yönlendirme cihazı olarak görev yapmaktadır ve bunlar kullanılarak yönlendirme yolu oluşturulmaktadır. Kök, ebeveyn ve çocuk düğümlerden oluşan bu ağ yapısındaki haberleşme trafiği, yukarı (*upward*) ve aşağı (*downward*) olmak üzere iki yönlü olarak sağlanmaktadır. Ağ geneline bakıldığında da bütün düğümler düşük güç ve kayıplı cihazları temsil etmektedir.

RPL ağacındaki düğümlerin her biri kelime karşılığı sıra olarak tanımlanabilecek *rank* değerine sahiptir. Bu değer, düğümlerin kök düğüme göre pozisyonunu belirlemek amacıyla kullanılmaktadır. Bu değer kök düğümden yaprak düğümlere doğru ilerledikçe artmakta, çocuk düğümlerden kök düğüme doğru ilerledikçe azalmaktadır. Genel olarak bu değer, kök düğümden 0 olacak şekilde tanımlanmakta ve sayısal olarak artarak çocuk düğümlerin sıra değeri belirlenmektedir. Fakat sıra değerinin hesaplanması, RPL'nin önemli temel adımlarından olan amaç fonksiyonuna (OF) göre yapılmaktadır. Bir başka deyişle sıra değeri amaç fonksiyonuna bağlıdır. Amaç fonksiyonu, yönlendirme metriklerini ve kısıtları içermektedir. Düğümlerin pozisyonu, ebeveyn düğümün seçimi gibi farklı noktalarda karar verici ve tanımlayıcı olmaktadır. RPL ağacı, bahsedilen tanımlayıcı ve belirleyiciler kullanılmaktadır.

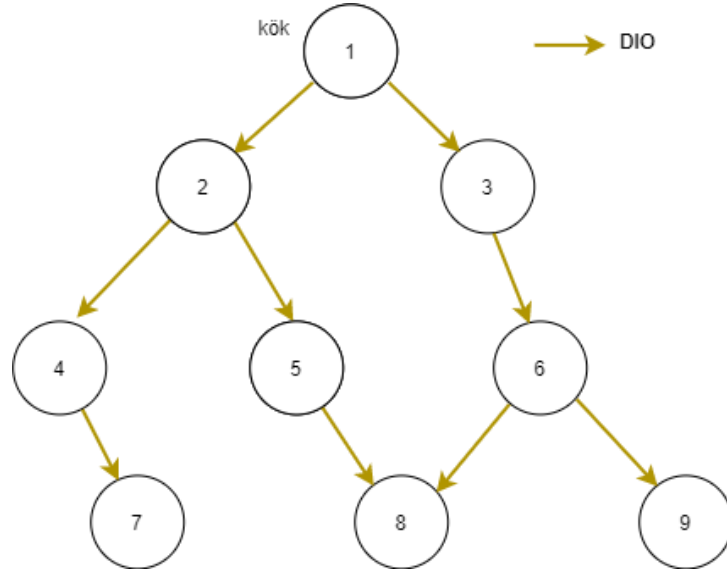
- RPL Instance Kimliği; DODAG bilgi nesnesi (DIO) mesajının gönderildiği RPL'ye ait kimlik,
- DODAG Kimliği; RPL'nin gönderildiği topoloji bölümüne ait kimlik,
- DODAG Versiyon Numarası,
- Düğümlere ait sıra değerleri

İletişim kuralları ve iletişimdeki kısıtların belli olmasıyla birlikte, RPL'de mesaj akışı meydana gelmektedir. Aşağı ve yukarı yönlü mesajlardan oluşan bu iletişim periyodu, kök düğümlerden başlayarak ebeveyn düğümlere doğru iletilen aşağı yönlü iletişim ile başlamaktadır. Aşağı yönlü yönlendirme P2MP ve P2P sağlar. Bu yönlendirme aşamasında çocuk ve ebeveyn düğümlerden oluşan ağaç oluşturulmaktadır. DIO mesajları, aşağı yönlü mesaj iletiminde kullanılan mesaj türüdür. Yukarı yönlü iletişimde ise, DIO mesajları gönderilen çocuk düğümlerin, amaç fonksiyonuna göre hesapladıkları sıra değerlerine göre ebeveyn düğüm seçme işlemi yapılmaktadır. Her çocuk düğüm, ebeveyn düğüm seçerek kök düğüme kadar

mesajlar iletilmektedir. Bu işlem hedef reklam nesnesi (DAO) mesajları kullanılarak yapılır. Bütün bu tanımlamalar eşliğinde RPL, üç temel ICMPv6 paket tipini kullanmaktadır (Idrees and Witwit, 2018):

- DIO: Aşağı yönlü iletişim (downward) aşamasında kök düğümden başlayarak çocuk düğümlere doğru gönderilen mesajdır. Yeni bir DAG ağacı oluşturulmaktadır. DODAG kökü tarafından yapılmaktadır. Çok noktalı (multicast) olarak gönderilmektedir. Ebeveyn seçimi için gereken bilgileri taşımaktadır.
- DAO: Yukarı yönlü iletim (upward) aşamasında, çocuk düğüm içermeyen düğümlerden başlayarak gönderilen yukarı yönlü mesajdır.
- DIS: DODAG bilgi talebi. Bir düğüm başka bir RPL düğümünden DIO talep etmek istediğinde kullanılmaktadır. Bir düğüm ağa katıldığında bu mesaj türünü kullanmaktadır. Bu mesaj türü ile çok noktalı (multicast) yayın yaparak ağın durumuna göre müsait durumdaki ebeveyn seçimi yapmaktadır.

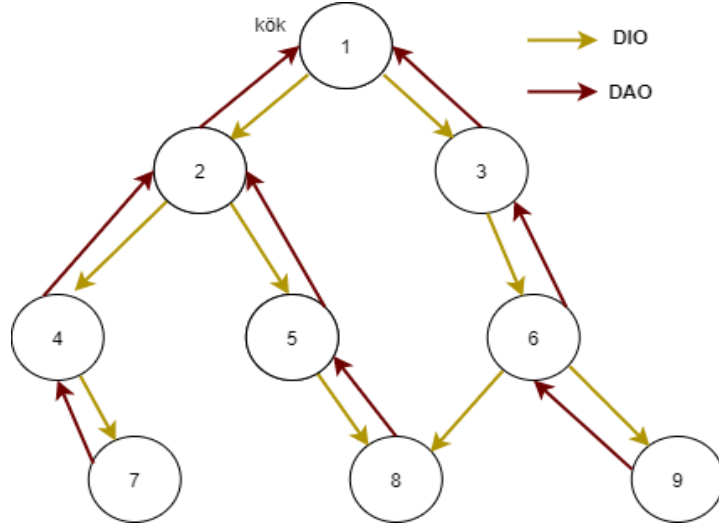
Şekil 2.2, RPL’de kullanılan DIO mesajlarının akışına bağlı olarak gerçekleşen aşağı yönlü iletişim (downward) aşamasını göstermektedir.



Şekil 2.2. Aşağı yönlü mesaj gönderimi ve DIO mesajı dağılımı

Kök düğümden başlayarak, alt düğüm olan çocuk düğümlere gönderilen DIO mesajları, ağaç oluşturulması ve keşiften sorumludur. Bu aşama tamamlandıktan sonra, yani DIO mesajı alacak çocuk düğüm kalmadığı andan itibaren yukarı yönlü

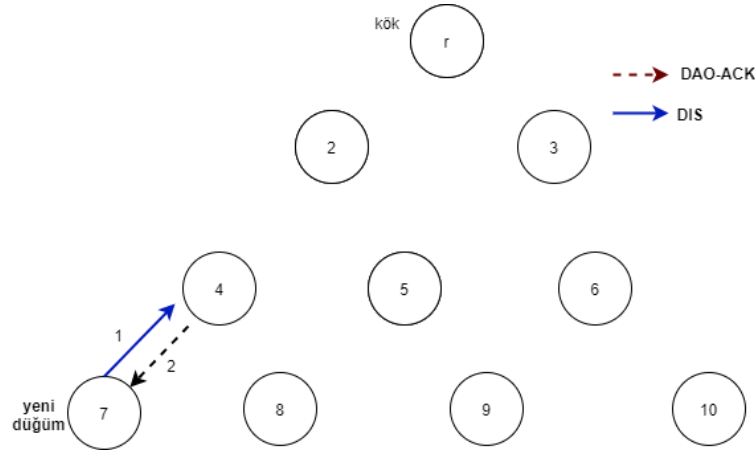
yönlendirme başlamaktadır. Şekil 2.3, RPL’de kullanılan DAO mesajlarının akışına bağlı olarak gerçekleşen yukarı yönlü iletişim (upward) aşamasını göstermektedir.



Şekil 2.3. Yukarı yönlü mesaj gönderimi ve DAO mesajı dağılımı

Yukarı yönlü yönlendirmede, amaç fonksiyonuna göre hesaplanan sıra değerleri üzerinden ebeveyn seçimi yapılır. Amaç fonksiyonu, yönlendirme ölçütlerini belirlemektedir. Sonraki bölümlerde ele alınacak amaç fonksiyonları, kapasitenin maksimize edilmesi, enerji verimliliğinin maksimize edilmesi, enerji tüketiminin ve gecikmenin indirgenmesi şeklinde tanımlanabilir. Bu tanımlamalara uygun matematiksel ifadeler, mesaj akışı esnasında hesaplanarak ebeveyn seçiminde belirleyici olmaktadır. En alt basamaktaki çocuk düğümünden kök düğümüne kadar ebeveyn seçimi, amaç fonksiyonuna göre gerçekleşmektedir.

Şekil 2.4, RPL ağacına yeni eklenen bir düğümünden yayılan DIS mesaj akışını göstermektedir.

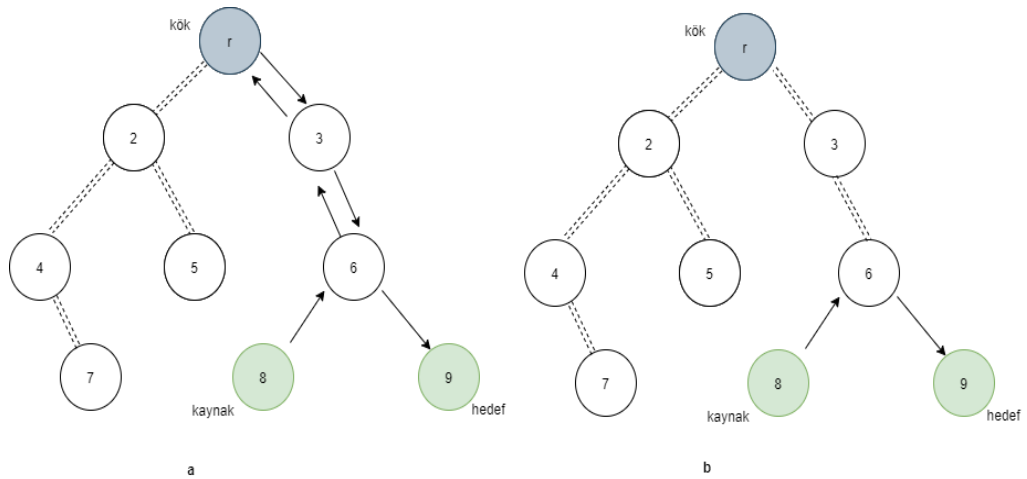


Şekil 2.4. DIS mesajı dağılımı

RPL, yönlendirme yolu oluşturulması ve rota kararı noktasında, iki temel amaç fonksiyonu kullanılmaktadır (Kim, et al., 2018). Bunlar *OF0* ve *ETX* olarak temsil edilen fonksiyonlardır. *OF0*, bağlantı durumuna bakmaksızın, yalnızca kök düğüme olan mesafesi en kısa olan düğümlerin seçilerek yönlendirme rotasının oluşturulmasını amaçlamaktadır. *ETX* ise, ebeveyn seçimi ve buna bağlı olarak yönlendirme rotası oluşturma işlemini enerji, iletim gücü, yük dengesi gibi önemli ölçütlere bağlı kalarak gerçekleştirmektedir. Bu durumda sıra değerleri, kök düğümden uzaklığı değil, istenen veya beklenen ölçütler bakımından fonksiyon sonucu hesaplanan değeri göstermektedir. Ebeveyn düğüm seçiminde, en küçük ya da en büyük sıra değerlerine göre atama yapılmaktadır. Örneğin, *OF0* amaç fonksiyonuna göre yapılan rota keşfi ve yönlendirme yolu oluşturulması, gecikme, paket kaybı, enerji kullanımı bakımından en uygun yolu vermeyebilir. Bu durumda amaç fonksiyonu, belirlenen kısıtlara ve başarı ölçütlerine göre belirlenerek yönlendirme yolu oluşturulmaktadır. Ayrıca, kök düğüme yakın olan düğümlerin yüklenme oranının diğer düğüme göre daha fazla olduğu ve bu durumun da bataryalarının diğerlerine oranla daha hızlı tükendiği belirlenmiştir (Vasseur, 2011).

RPL’de mesaj akışı, ebeveyn seçimi gibi iş süreçleri devam ederken, bu süreç zamanlayıcı algoritma ile yönetilmektedir. IETF tarafından standartlaştırılan Trickle Timer Algoritması, RPL üzerinde kontrol mesajlarının gönderilme sıklığını yönetmek için kullanılmaktadır. Bu algoritma yardımıyla iletim dinamik olarak kontrol edilmektedir. DIO mesajları, Trickle Timer (RFC6206) kullanılarak iletilmektedir. Trickle Timer işlevi, düğümler arasında mesaj iletiminden dolayı oluşabilecek tutarsızlıklar için kullanılmaktadır.

RPL protokolü, yönlendirme yolu oluşturulurken düğümlerin depolayan (*storing*) ve depolamayan (*non-storing*) modlarda çalışmaları prensibi ile çalışmaktadır (Ionescu, et al., 2013). Aşağı ve yukarı yönlü mesaj iletimleri ve ebeveyn seçimlerinin ardından oluşan yönlendirme yol bilgisi, depolandığı yere göre RPL algoritmasını sınıflandırmaktadır. Depolama yapılmadan çalışan modda, ara düğümler yönlendirme yol bilgisini tutmamaktadır. Yönlendirme yolunun tamamı DODAG kök düğümünde tutulmaktadır. Kaynaktan hedefe olan yol, kök düğüm üzerinden belirlenmektedir. Paketler her zaman kök üzerinden hedefe ulaşmaktadır. Böylece her düğüm ayrıca bellek alanı tüketimi yapmamaktadır. Fakat ağ boyutu ve yol bilgisi arttıkça, kök düğüm üzerinde çok fazla yük oluşabilir. Şekil 2.5, RPL algoritmasındaki depolayan ve depolamayan çalışma şeklini göstermektedir.



Şekil 2.5. a) Depolamayan (non-storing) mod b) Depolayan (storing) mod

Şekil 2.5'teki yönlendirme yolu, düğümlerin yol bilgisini tutmadığı depolamayan modda (a) iken 8-6-3-r-3-6-9 olacak şekilde belirlenmektedir. Kaynak düğüm olan 8, hedef düğüm olan 9 arasında düğümler diğer düğümlere ait bilgiyi saklamadığı için kök düğüm üzerinden iletişim gerçekleşmektedir. Diğer durum olan, her düğümün yol bilgisini tuttuğu modda (b) ise aynı kaynak düğümden aynı hedef düğüme olan yol, 8-6-9 şeklinde belirlenmektedir.

## 2.2. RPL Amaç Fonksiyonları ve Yönlendirme Metrikleri

RPL yönlendirme protokolü, yönlendirme metriklerini OF üzerinden sağlamaktadır. Minimum uzunlukta yol seçimi, minimum enerji kullanımı ve gecikme, maksimum paket iletim oranı gibi istenen yönlendirme davranışları amaç fonksiyonunu belirlemektedir. RPL üzerindeki geliştirmeler de bu fonksiyonun

işlevine göre farklılıklar göstermektedir. Temel olarak link ve düğüm metrikleri olarak kategorize edilen amaç fonksiyonu (Benamar and Lamaazi, 2020), ağ performansında belirleyici olmaktadır. RPL algoritması, çeşitli amaç fonksiyonlarına göre yönlendirme yolunu seçmektedir. Amaç fonksiyonu bağlantının kalitesini belirlemektedir. Bu bölüm özelinde yapılan benzetimlerde çeşitli amaç fonksiyonları kullanılmıştır. Amaç fonksiyonuna bağlı olarak da yönlendirme metrikleri belirlenmiştir. Ebeveyn düğüm seçimi için kullanılan metrikler aşağıdaki gibidir:

- 1) Kapasite (P1): İki düğüm arasındaki bağlantı ve iletişim parametresidir. Kapasite, birim zamanda gönderilen veri miktarı olarak tanımlanmaktadır. Bu amaç fonksiyonu Denklem 2.1 ve 2.2’de verilmiştir ve en yüksek kapasiteye sahip düğümleri bulmak için kullanılır. Kapasite, ağ performansı için en yüksek değerde olmalıdır ve en yüksek kapasite bulma fonksiyonu, P1 olarak temsil edilmiştir.  $B$  olarak sembolize edilen metrik Denklem 2.1’de gösterildiği gibi hesaplanmaktadır.

$$B = (max) \sum_{\forall l_{i,j} \in P} \delta_{ij} W \log_2(1 + SNR_{i,j}) \text{ Mbps} \quad (2.1)$$

Denklem 2.1’de yer alan  $P$ , ağ topolojisindeki bütün düğümlerin kümesini temsil etmek için kullanılmakta ve  $l_{i,j}$ ,  $P$  kümesindeki herhangi bir linki temsil etmektedir.

$$\delta_{ij} = \begin{cases} 1, & \text{Eğer } l_{i,j} \text{ linki üzerinde pozitif veri akışı varsa} \\ 0, & \text{diğer durumda} \end{cases} \quad (2.2)$$

Denklem 2.2’de yer alan  $\delta_{ij}$ , link üzerindeki veri akışını temsil eder.  $\delta_{ij} = 1$  durumu, link üzerinde pozitif veri akışı olduğu durumu temsil etmekte ve hesaplama bu değer üzerinden yapılmaktadır.  $\delta_{ij} = 0$  durumu da diğer durumları temsil etmektedir.  $W$  temsili Hertz biriminden bant genişliğine işaret etmektedir. Bu bölümdeki benzetimde  $W$  değeri sabit olarak varsayılmıştır.  $SNR_{i,j}$ , sinyal/gürültü oranını (signal-to-noise-ratio) temsil etmektedir ve her bir  $l_{i,j}$  linki için bu değerler, ortalama değerler üzerinden rastgele seçim ile belirlenmiştir.

- 2) Enerji tüketimi (P2): Enerji tüketimi, ağdaki düğümlerin ağdaki işlemler için harcadığı toplam enerji miktarıdır. Bu amaç fonksiyonu P2 ile temsil edilmiştir

ve Denklem 2.3'te verilmiştir. En düşük seviyede enerji tüketen düğümleri bulmak için kullanılır. Tüm düğümler için enerji tüketimi en aza indirilmelidir.  $E_C$  olarak temsil edilmekte ve Denklem 2.3'te gösterildiği gibi hesaplanmaktadır.

$$E_C = (\min) \sum_{\forall i,j \in P} \delta_{ij} (ETX + E_{Circuitry} + E_{Idle}) \quad (2.3)$$

$$+ \sum_{\forall i,j \notin P} \delta_{ij} E_{Idle} \quad \text{joule}$$

ETX, RFC6551 tarafından Beklenen İletim Sayısı olarak tanımlanır. LLN'lerde sıklıkla kullanılmaktadır. ETX, paketin kaynak düğümden hedefe başarılı bir şekilde iletilmesi olarak tanımlanmaktadır. Denklemler 2.4 ve 2.5'te gösterildiği gibi hesaplanmaktadır.

$$ETX = P t_{t_x} \quad \text{Joule} \quad (2.4)$$

$$t_{t_x} = \frac{\text{packet size}}{B} \quad \text{saniye} \quad (2.5)$$

Denklem 2.4'te yer alan  $P$  değeri, sabit olarak değerlendirilmiş olan güç değerini temsil etmektedir.  $E_{Circuitry}$  değeri, cihazların kendine özgü özelliklerine göre enerji tüketimidir. Denklem 2.6'da gösterildiği gibi hesaplanmaktadır.

$$E_{Circuitry} = P_C t_{t_x} \quad \text{Joule} \quad (2.6)$$

$P_C$  cihazın güç değeri olarak temsil edilir ve sabit olarak kabul edilmiştir.

$T_{remain}$ , cihazların iletişim için kalan sürelerini temsil etmektedir ve Denklem 2.7'de gösterildiği gibi hesaplanmaktadır.

$$T_{remain} = T - t_{t_x} \quad \text{saniye} \quad (2.7)$$

$E_{Idle}$  cihazların ağ üzerinde herhangi bir işlem yapmadıkları durumda, bir başka deyişle boş kaldıkları zaman periyodlarında harcadıkları enerjidir. Denklem 2.8'de gösterildiği gibi hesaplanmaktadır.

$$E_{Idle} = P_{Idle} T_{remain} \quad \text{Joule} \quad (2.8)$$

3) Enerji verimliliği (P3): Enerji verimliliği, ele alınan bir süre boyunca düğüm başına tüketilen enerjiye karşı alınan çıktı olarak tanımlanabilir. Ağdaki enerji tüketimini belirleyen ölçüdür. Bu amaç fonksiyonu P3, Denklem 4.9'da verilmiştir ve ağda maksimum enerji verimli düğümleri bulmak için kullanılmaktadır. Düğümlerin enerji verimliliğini en üst düzeye çıkarmak, yönlendirmedeki enerji tüketiminin en aza indirilmesini sağlamaktadır.  $E_E$  olarak temsil edilir ve Denklem 2.9'da gösterildiği gibi hesaplanmaktadır.

$$E_E = (max) \quad R/E_C \quad \text{bits/joule} \quad (2.9)$$

$R$  değeri, link üzerinde taşınabilecek maksimum bit sayısını temsil etmektedir ve Denklem 2.10'da gösterildiği gibi hesaplanmaktadır.

$$R = B * T_s \quad \text{bits} \quad (2.10)$$

4) Uçtan uca gecikme: Uçtan uca gecikme, bir paketin ağ üzerinden kaynaktan hedefe iletilmesi için geçen süreyi ifade eder. Ayrıca yayılma gecikmesi, iletim gecikmesi, kuyruklama gecikmesi ve işlem gecikmesinin toplamı olarak da hesaplanmaktadır.

### 2.3. RPL Benzetim Sonuçları

Bu bölümde, Python programlama dilini kullanarak RPL algoritmasının performansını değerlendiren benzetim parametreleri ve sonuçları sunulmuştur. Farklı koşullar altında RPL'nin genel davranışını analiz etmek için çeşitli simülasyon ayarları ve parametreleri dikkate alınmıştır. Semboller ve simülasyon parametreleri Tablo 2.1'de gösterilmektedir. Benzetimler tek bir kök düğüme sahip, statik bir DODAG ağ topolojisi üzerinde yapılmıştır.

Tablo 2.1. Benzetim parametreleri

<b>Sembol</b>	<b>Tanım</b>	<b>Değer</b>
$t$	Simülasyon süresi	20 iterasyon
$L$	Paket boyutu	[100,1000] byte
$I$	Paket gönderim aralığı	0.5 sn
$n$	Düğüm sayısı	[5, 20]
$snr$	SNR değer ortalamaları	[2.5, 20] dB
$P$	Güç	1.98 W
$W$	Kanal bant genişliği	5 MHz
$P_c$	Devreleme gücü	0.21 W
$P_{idle}$	Boş anda harcanan güç	0.99 W
$T_s$	Zaman periyodu	100 sn

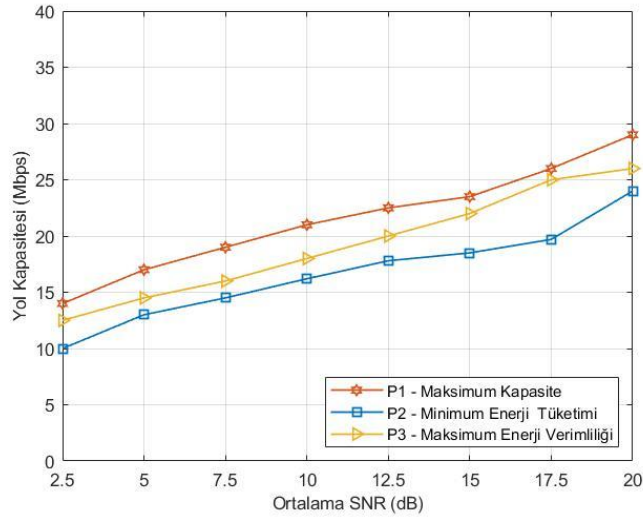
Algoritmada, farklı senaryolar için benzetim yapılmıştır. Sabit düğüm sayısı senaryosunda, ağ topolojisi için rastgele dağıtılmış 20 düğüm üretilmiştir. Modelin tek kök düğümü olduğundan DODAG Id değeri kullanılmamıştır. Mesaj sınıfı, DIO, DAO ve DIS mesaj sınıfları dikkate alınarak oluşturulmuştur; DIO, DAO ve DIS. Bölüm 2.4'te belirtildiği gibi, her düğümün ve RPL ağacının kimlik bilgileri dikkate alınır. Diğer senaryolarda düğüm sayısı 5, 7, 9, 10, 13 ve 20 olarak belirlenmiş ve düğümler statik olarak kabul edilmiştir. Her farklı parametre için yineleme limiti 20 olarak belirlenmiştir ve paket gönderim zaman aralığı 0.5 saniyedir. Benzetimde kullanılan SNR değerleri farklı ortalama değerlere göre üssel olarak üretilmiş, atanmıştır. Bir linke ait SNR değeri farklı ortalama değerler baz alınarak üstel bir dağılım gösterdiği varsayılmış ve bu değerler çeşitli ortalamalar ile rastgele oluşturulmuştur. Her düğüm aynı zaman periyoduna sahiptir. Protokol değerlendirmesi, enerji tüketimi, ortalama ağ kapasitesi, yönlendirme yolu kapasitesi ve ortalama uçtan uca gecikme gibi performans ölçütleri kullanılarak gerçekleştirilir. Tablo 2.2, RPL benzetiminin temel özelliklerini özetlemektedir.

Tablo 2.2. Benzetimin temel özellikleri

<b>Özellik</b>	<b>Tanım</b>
Algoritma	RPL
Benzetim aracı	Python
Ağ topolojisi	Rastgele dağıtılmış, sabit
Mesaj akışı	MP2P, P2P, P2MP
Kontrol mesajları	DIO, DAO, DIS

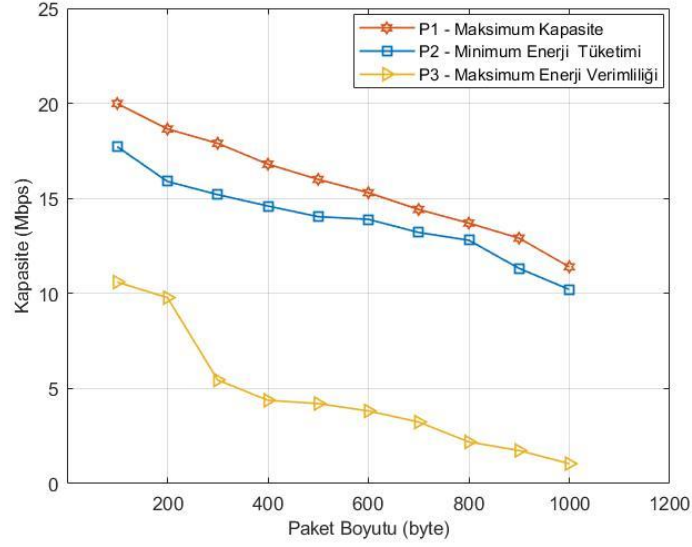
Benzetim sonuçları, belirlenen tekrar sayısının ardından ortalama değerler şeklinde verilmiştir. Tablo 2.1’de gösterildiği üzere benzetim süresi 20 tekrar sayısı, bant genişliği 5 MHz olarak belirlenmiştir. Diğer parametreler de sabit olacak şekilde varsayılmıştır. Benzetim başlamadan önce kaynak ve hedef düğüm bilgileri algoritmaya argüman olarak gönderilmiştir. Yineleme sayısı belirlenen değere ulaştığında bir sonraki yineleme, farklı bir amaç fonksiyonu ve performans ölçütüne göre gerçekleştirilmiştir.

Şekil 2.6, 2.5 ile 20 arasında değişen farklı SNR değerlerinde, RPL algoritmasının yol kapasitesi karşılaştırmasını göstermektedir. Algoritma, kapasite maksimizasyonu hedefliyorsa, enerji tüketimi ve enerji verimliliği ölçütlerinde optimal olmayan şekilde davranmaktadır. Bunun nedeni, algoritmanın ebeveyn seçiminde yalnızca en yüksek kapasiteli yola göre seçim yapması ve diğer ölçütleri ebeveyn seçimi esnasında dikkate almamasıdır. Ayrıca SNR değerindeki artış sinyal kalitesini de artırdığından, kapasite değeri SNR artışından olumlu yönde etkilenmektedir.



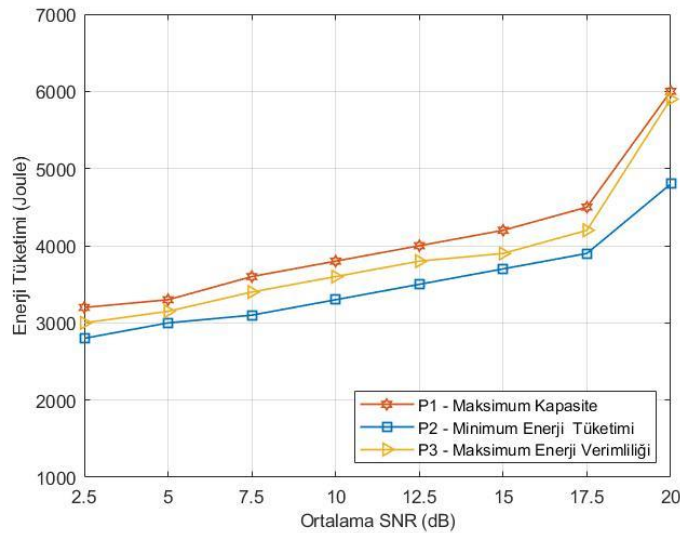
Şekil 2.6. Farklı amaç fonksiyonları ve yönlendirme metriklerine göre kapasite

Şekil 2.7, sırasıyla 200, 300, 400, 600, 800 ve 1000 bayt’lık paket boyutları için seçilen yönlendirme yolunun kapasitesini göstermektedir. Benzetimin SNR değeri bu simülasyon çalışması için 10 dB olarak sabit kabul edilmiştir. Algoritma, yönlendirme yolunu üç ölçütün tümünü değerlendirerek seçmektedir. Ek olarak, paket boyutu arttığında, kapasite değeri azalmaktadır. Başka bir deyişle, daha büyük paketler için kapasitedeki artış, küçük paketler için olan artıştan daha hızlıdır.



Şekil 2.7. Farklı paket boyutlarında yönlendirme yolunun kapasite değişimi

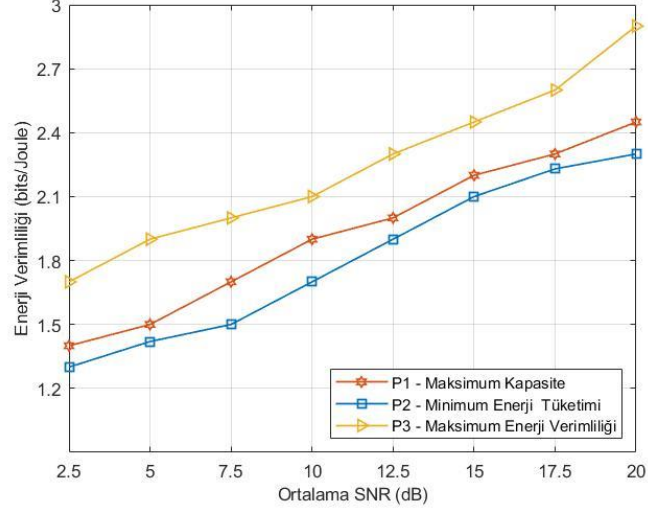
Şekil 2.8, yönlendirme yolunun enerji tüketiminin diğer yönlendirme metrikleri altında değişimini göstermektedir. Görüldüğü üzere, SNR değeri arttıkça kapasite, enerji kullanım verimliliği ve enerji tüketimi de artmaktadır. Algoritma, en düşük enerji tüketimini baz alır ve ebeveyn düğümlere bağlı oluşan yönlendirme yolu bu düğümlerin seçimi üzerinden oluşturulmaktadır. Böylece, enerji tüketimi baz alınarak oluşturulan yönlendirme yolu, diğer ölçütlere kıyasla daha az enerji tüketmektedir.



Şekil 2.8. Farklı amaç fonksiyonlarına göre enerji tüketimi

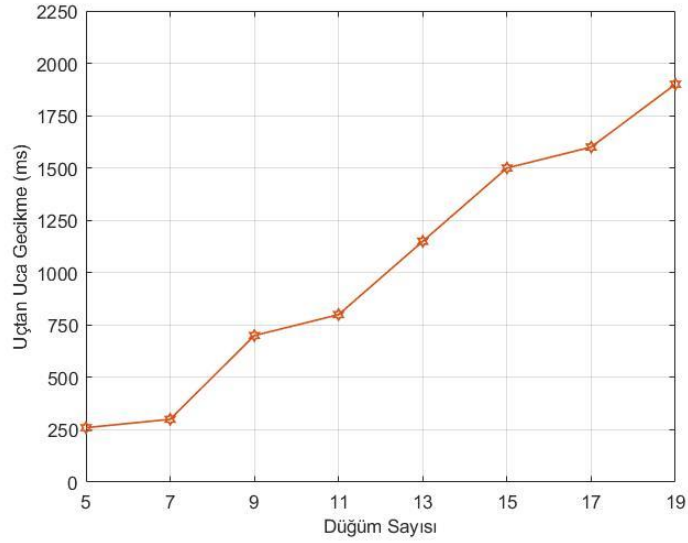
Şekil 2.9, düğümlerin, algoritma tarafından oluşturulan yönlendirme yolu boyunca enerji verimliliğini göstermektedir. Yönlendirme yolu oluşturulurken, enerji verimliliği en yüksek düğüm ebeveyn düğüm olarak seçilmektedir. SNR değeri ne

kadar yüksek ise iletişim kalitesi ve iletim performansı da bu ölçüde yüksek olmaktadır. Böylece birim enerji başına gönderilen bit sayısı artmaktadır.



Şekil 2.9. Farklı amaç fonksiyonlarına göre enerji verimliliği

Şekil 2.10, sırasıyla 5, 7, 9, 10, 13, 15, 17, 19 düğüm sayıları için RPL ağacında seçilen yönlendirme yolunun uçtan uca gecikmesine ait karşılaştırmadır. Kaynaktan hedefe uçtan uca gecikme, düğüm sayısı arttıkça artmaktadır. Düğüm sayısının artmasının, ağdaki birim işlem sayısını artırması ve buna bağlı olarak kuyruk, işlem, işletim ve yayılma gecikmeleri artacaktır.



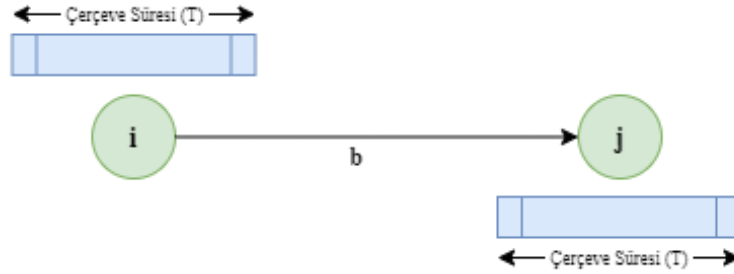
Şekil 2.10. Yönlendirme yolunun uçtan uca gecikmesi

### 3. BİLİŞSEL RADYO ÖZELLİKLİ IOT (CR-IOT) DÜĞÜMLERDE ENERJİ MODELİ

Bu bölümde, CR özellikli IoT ağlar için tasarlanan enerji tüketim modeli, enerji verimliliği modeli ve tanımlamaları aktarılacaktır.

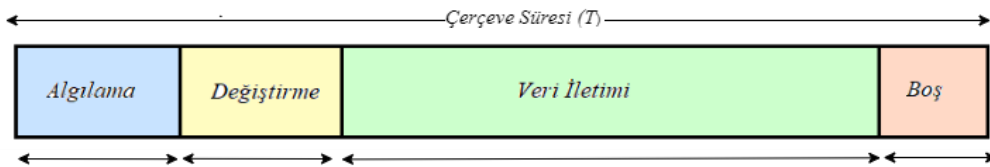
#### 3.1. CR-IoT Düğümlerde Tüketilen Enerjinin Modellenmesi

Ağ cihazları, belirlenen kısıtlar dahilinde iletişim yapmak için zaman çerçevesi (time slot) yapısına sahiptir. Bu çerçeve süresi, her bir düğüm için ağ üzerinde buldukları süre boyunca yapılan iş parçacıklarına göre azalmaktadır. Her bir düğüm aynı zaman çerçevesine sahip olacak şekilde temsil edilen topolojiye düğüm çerçeve ve bağlantı temsili Şekil 3.1’de gösterilmiştir.



Şekil 3.1. Çerçeve, düğüm ve bağlantı temsili

CR özellikli IoT düğümleri (CR-IoT), buldukları ağ üzerinde iletişimi başlatmak için boşta bulunan frekans bantlarını algılama işlevini yerine getirmelidir. Şekil 3.2’de gösterilen çerçeve yapısında,  $l_{ij}^b$  linki üzerindeki harcanan enerjinin, veri iletim modunda olan düğüm için tasarlanan modeli sunulmaktadır. CR-IoT düğümlerinden  $i$  ve  $j$ , birbirleri ile iletişim kurmak için  $b$  linki üzerinde ortak olan  $b$  frekans bandına antenlerini ayarlamaktadır. Hem gönderim yapan hem de veriyi alan CR düğümleri ortak frekans bantlarına geçmelidir.



Şekil 3.2. Veri gönderimi yapacak düğüme ait çerçeve yapısı

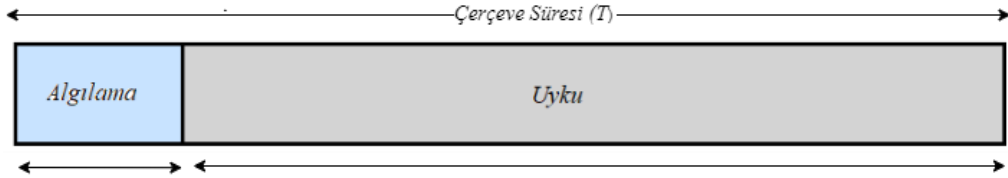
Gönderici düğümde olduğu gibi veri alım modunda olan düğüm de antenini gönderici düğüm ile aynı frekans bandına ayarlamalıdır. Şekil 3.3’te gösterilen çerçeve

yapısında  $l_{ij}^b$  linki üzerindeki harcanan enerjinin, veri alım modunda olan düğüm için tasarlanan modeli sunulmaktadır.



Şekil 3.3. Veri alımı yapacak düğüme ait çerçeve yapısı

CR-IoT düğümleri veri iletim ve veri alım modu dışında ayrıca uyku modunda da enerji harcamaktadır. Bu durumda düğüm, iletişim için kullanıma uygun frekans bandı bulamamış olarak değerlendirilebilir. Şekil 3.4, uyku modunda olan düğüme ait çerçeve yapısını temsil etmektedir.



Şekil 3.4. Uyku modunda olan düğüme ait çerçeve yapısı

Düğüm, yukarıdaki şekiller üzerinde temsil edildiği şekliyle, iletişim esnasında ve durağan halde çerçevelerinden tüketim yaparlar. Bu esnada da enerji tüketimi gerçekleşir.

CR-IoT düğümlerin, kullanılabilir durumdaki frekans bandını algılama işlemi esnasında harcadığı enerji  $E_{sense}$ , Denklem 3.1'de gösterildiği gibi hesaplanmaktadır.

$$E_{sense} = P_{sense} t_{sense} \text{ joule} \quad (3.1)$$

Denklem 3.1'de yer alan  $P_{sense}$  değeri, algılama işlemi için harcanan gücü temsil etmektedir.  $t_{sense}$  ifadesi bu işlem için harcanan süreyi temsil etmektedir ve Denklem 3.2'de gösterildiği gibi hesaplanmaktadır.

$$t_{sense} = T\alpha \text{ saniye} \quad (3.2)$$

Denklem 3.2’de yer alan  $\alpha$  parametresi, algılama işlemi esnasındaki zaman oranını temsil etmektedir ve değeri sabittir.  $T$  değeri, düğüme ait çerçeve süresini ifade etmektedir.

CR-IoT düğümler, kullanılabilir durumdaki frekans bandını algıladıktan sonra antenlerini o banda ayarlamak için frekans geçişi yapmalıdır. Frekans geçişi için harcanan enerji  $E_{switch}$ , Denklem 3.3’te gösterildiği gibi hesaplanmaktadır.

$$E_{switch} = P_{switch} t_{switch}^{ij,b} \quad \text{saniye} \quad (3.3)$$

Denklem 3.3’te yer alan  $P_{switch}$  değeri, frekans değiştirmek için harcanan frekans değiştirme gücünü,  $t_{switch}^{ij,b}$  değeri de  $b$  bandını kullanan  $l_{ij}$  linki üzerindeki frekans geçiş süresini temsil etmektedir. Ağda aktif durumda bulunan düğümler frekans algılama ve frekans geçişi işlemlerinden sonra veri iletimi için kalan çerçeve süresi  $T - t_{sense} - t_{switch}^{ij,b}$  olarak hesaplanmaktadır.

Düğümler algılama ve frekans değiştirme işlemlerini gerçekleştirdikten sonra iletim yapacak durumdadır. Çerçeve yapısında temsil edildiği şekliyle, veri iletimi için de CR-IoT düğümleri enerji tüketimi yapmaktadır. Veri iletimi için harcanan enerji  $E_{tx}$ , en temel haliyle Denklem 3.4’te gösterildiği gibi hesaplanmaktadır.

$$E_{tx} = P_{tx} t_{tx}^{ij,b} \quad \text{joule} \quad (3.4)$$

$P_{tx}$  değeri veri iletimi için harcanan gücü temsil etmektedir. Veri iletimi için geçen süre  $t_{tx}^{ij,b}$ , paket boyutu  $L$  ve  $b_{ij}$  linki üzerindeki kapasite  $C_{ij}^b$  olarak temsil edildiği durumda Denklem 3.5’te gösterildiği gibi hesaplanmaktadır.

$$t_{tx}^{ij,b} = \frac{L}{C_{ij}^b} \quad \text{saniye} \quad (3.5)$$

Düğümler, veri iletimi esnasında, sahip oldukları karakteristik özellikler dolayısıyla donanımsal olarak devre gücü harcamaktadır. Bu güç  $P_{circuit}$ , veri iletimi

ve alımı gibi etkenlerden bağımsızdır ve Denklem 3.6'da gösterildiği gibi hesaplanmaktadır.

$$P_{circuit} = \beta + \left(\frac{1}{\eta} - 1\right) P_{tx} \quad \text{Watt} \quad (3.6)$$

Denklem 3.6'da yer alan  $\beta$  değeri, düğümün donanımsal özelliğinden dolayı bir bileşeni,  $\eta$  değeri de yükselteç (amplifikatör) verimliliğini,  $P_{tx}$  ise devre gücünü göstermektedir. Bunlarla birlikte veri iletimi esnasında tüketilen enerji Denklem 3.7'de gösterildiği gibi hesaplanmaktadır. (Shu, et al., 2006):

$$E_{tx} = (P_{tx} + P_{circuit}) t_{tx}^{ij,b} \quad \text{joule} \quad (3.7)$$

Kullanılabilir durumdaki frekans bandını algılayıp ortak bir frekansa geçiş yapan düğümler, veri iletimi yaptıktan sonra kendilerine gelen veriyi almak için de enerji tüketmektedir. Veri alımı için tüketilen enerji  $E_{rx}$ , Denklem 3.8'de gösterildiği gibi hesaplanmaktadır.

$$E_{rx} = E_{rx/bit} L \quad \text{joule} \quad (3.8)$$

Denklem 3.8'de yer alan  $E_{rx/bit}$  değeri, düğümlerin bir bit veriyi almak için harcadığı enerjidir.

Boş durumda olan alıcı, veri almaya hazır durumda olmasına rağmen veri alımında bulunmaz. Cihaz aktif durumdadır ve enerji tüketmektedir. Veri iletimi yaptıktan sonra bir sonraki işleme kadar boş duruma geçen bu CR düğümün harcadığı enerji,  $E_{idle}$  Denklem 3.9'da gösterildiği gibi hesaplanmaktadır (Bayhan ve Alagöz., 2012).

$$E_{idle} = P_{idle} (T - t_{sense} - t_{switch}^{ij,b} - t_{tx}^{ij,b}) \quad \text{joule} \quad (3.9)$$

Şekil 3.4'te temsil edildiği gibi uyku durumunda olan düğümler, kullanılabilir durumda olan frekans bandına geçiş yaptıktan sonra tümüyle uyku durumuna

geçmektedir. Düğümün uyku halinde harcadığı süre  $T - T_{sense}$  kadar olmaktadır. Uyku durumunda harcanan güç  $P_{sleep}$  olmak üzere, harcanan enerji  $E_{sleep}$ , Denklem 3.10’da gösterildiği gibi hesaplanmaktadır.

$$E_{sleep} = P_{sleep}(T - T_{sense}) \quad \text{joule} \quad (3.10)$$

Tüm işlemler dikkate alındığında B bandını kullanan link üzerinde harcanan toplam enerji  $E_{ij}^b$  Denklem 3.11’de gösterildiği gibi detaylı bir şekilde tanımlanabilir (Demirci, 2018).

$$\begin{aligned} E_{ij}^b = & 2P_{sense}t_{sense} + (P_{switch}t_{switch}^i + P_{switch}t_{switch}^j) \quad (3.11) \\ & + P_{idle}(\max(t_{switch}^i, t_{switch}^j) - \min(t_{switch}^i, t_{switch}^j)) \\ & + (P_{tx} + P_{circuit})t_{tx}^{ij,b} + \frac{E_{rx}}{\text{bit}} L + 2P_{idle}(T - T_{sense} \\ & - t_{switch}^{ij,b} - t_{tx,rx}^{ij,b}) \quad \text{joule} \end{aligned}$$

Enerji tüketim adımları göz önüne alındığında, harcanan toplam enerji  $E_{CRN}$ ,

$$E_{CRN} = \sum_{\forall l_{i,j} \in P} E_{ij}^b \delta_{ij} + \sum_{\forall l_{i,j} \notin P} P_{sleep}(T - T_{sense}) \quad \text{joule} \quad (3.12)$$

şeklinde temsil edilmektedir. Denklem 3.12’de yer alan  $\delta_{ij}$  değişkeni link üzerinde veri akışı olduğu durumda 1 değerini almakta, diğer durumlarda ise 0 olarak hesaplanmaktadır.

### 3.2. CR-IoT Düğümlerde Enerji Verimliliğinin Modellenmesi

Enerji verimliliği kavramı, birim enerji başına iletilen veri olarak tanımlanabilir. Bir başka deyişle birim enerji başına kapasite değeridir. Bir yönlendirme yolundaki en yüksek kapasite değeri, o yol üzerindeki darboğaz değerine eşit olmaktadır. Enerji verimliliği değeri  $E_E$ , Denklem 3.13’te gösterildiği gibi hesaplanmaktadır.

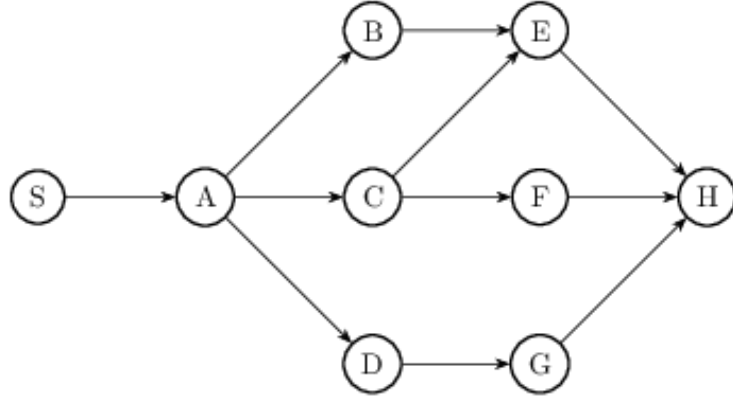
$$E_E = \frac{L}{E_{CRN}} \quad \text{bits/joule} \quad (3.13)$$

Denklem 3.13'te yer alan  $L$  değeri paket boyutunu,  $E_{CRN}$  ise Denklem 3.12'de hesaplanan enerji tüketimini temsil etmektedir.

### 3.3. Bilişsel Radyo Tabanlı Aç Gözlü RPL Algoritması (CR-GreedyRPL)

Bu alt bölümde, RPL algoritmasının geliştirilmesi ve CR özelliği katılarak CRN ağlarında da kullanılması amaçlanarak tasarlanan aç gözlü RPL algoritması (CR-GreedyRPL) ve buna bağlı olarak amaç fonksiyonları sunulacaktır.

Yönlendirme yolunun seçilmesi işlemi, var olan algoritmaların istenilen düzeye ve performans çıktıklarına yakınsanması amacıyla geliştirilmesi sonucu gerçekleşmektedir. Kaynaktan hedefe olan iletimin en düşük düzeyde kaynak kullanımı, en yüksek kapasite ile veri taşınması, en düşük düzeyde gecikme ile yapılması gibi sonuçlar, önerilen ve geliştirilen algoritmaların tercih sebebidir. Ağ üzerindeki yollar göz önüne alındığında, başlangıç noktasından hedef noktaya varıncaya kadar akışın en yüksek olduğu yolun bulunmasına ilişkin DTY algoritmasındaki akış ilkesi örnek çizge üzerinden Bölüm 3.5'te anlatılacaktır. Benzer şekilde, iletme bağlı yönlendirme yolunun kapasitesine bağlı seçimde kullanılan CR-GreedyRPL tasarlanmıştır. Tasarlanan bu algoritma, mevcut düğümden sonra gelen bir sonraki düğümü seçerken, kanal değiştirme maliyetini dikkate almaktadır. Algoritma, kaynak düğüm için çalıştığında, kapasite değeri en yüksek komşu düğüm bir sonraki düğüm olarak seçilmektedir. Çünkü kaynak düğüm, bir sonraki düğüme geçişte frekans geçişi yapmaz. Ara düğümler için çalışma şekli ise kaynak düğüme göre farklılık göstermektedir. Ara düğüm, bir sonraki düğümü seçerken, frekans geçişi yapacağı için kapasitede geçiş maliyetine bağlı olarak azalma olmaktadır. Geçiş maliyetleri hesaplanarak, net kapasite değeri en yüksek olan düğüm seçilmektedir. Hedef düğüm de kaynak düğüm gibi tek bir tane olduğundan, bir önceki adımda seçilen ara düğüm frekans değeri ile hedef düğüm frekans değeri arasında hesaplama yapılarak kapasite değeri bulunmaktadır. Şekil 3.5'te, CR-GreedyRPL için tasarlanan örnek çizge yapısı gösterilmiştir.



Şekil 3.5. CR-GreedyRPL için tasarlanan örnek çizge

Örnek gerçekleştirme için 9 adet düğüm ve 11 adet bağlantı bulunmaktadır. Düğümler arası bağlantılara atanan ve değer aralığı  $[0, 50]$  olan 50 adet frekans kullanılmıştır. Frekanslar arasındaki her birim geçiş 20 MHz, frekans değiştirmeden kaynaklanan frekans değiştirme gecikmesi de 0.1ms/MHz olacak şekilde belirlenmiştir. Düğümler arası bağlantılara ait SNR değerleri, 10 dB ortalamaya sahip olacak şekilde, üssel dağılım kullanılarak üretilmiştir. Her bir düğüm aynı zaman dilimine sahiptir ve bu değer 100 ms olacak şekilde belirlenmiştir. Linkler üzerinde CR düğümünün iletişim yapma olasılığı yani linkin CR tarafından kullanılma olasılığı  $(0.5, 1]$  aralığından olacak şekilde rastgele atanmıştır.

Düğümler arası iletişimde CR düğümleri, birbirleri ile veri alışverişinde bulunabilmek için aynı frekans bandı üzerinde bulunmak durumundadır. Mevcut frekanslar üzerinden, ortak frekans bandına geçiş için harcanan gecikme, frekans değiştirme gecikmesidir ve Denklem 3.14 ve Denklem 3.15'te gösterildiği gibi hesaplanmaktadır.

$$t_{sw}^i = t_f |b - b'| \text{ saniye} \quad (3.14)$$

$$t_{sw}^j = t_f |b - b''| \text{ saniye} \quad (3.15)$$

Bu durumda, iki düğümün ortak bir frekansa geçmesi ve o bant üzerinden haberleşmesi için en yüksek gecikmeye sahip olan frekans değiştirme gecikmesi baz alınır. Çünkü ortak frekansa geçiş için düğümlerin bekleme süresi, daha düşük sürede frekans değiştirme yapan düğüm olsa da en yüksek süreye bağlıdır. Denklem 3.16'da gösterildiği gibi hesaplanmaktadır:

$$t_{sw}^{ij,b} = \max(t_{sw}^i, t_{sw}^j) \text{ saniye} \quad (3.16)$$

$t_f$  temsili, birim bant genişliği için gereken frekans değiştirme süresidir.

Denklem 3.16 ile hesaplanan frekans geçiş süresi dikkate alınarak,  $b$  bandı üzerinde oluşan kapasite Shannon formülü kullanılarak Denklem 3.17'de gösterildiği gibi hesaplanmaktadır.

$$C_{ij}^b = W_{ij} \log_2(1 + SNR_{ij}) p_{ij}^b \text{ bits/saniye} \quad (3.17)$$

Denklem 3.17'de yer alan  $W_{ij}$  değeri,  $i, j$  linkine atanan bant genişliğini,  $SNR_{ij}$  değeri,  $i, j$  linkini kullanan  $j$  alıcısı için sinyal-gürültü oranını,  $p_{ij}^l$  değeri de linkin kullanım olasılığını temsil etmektedir.

Algoritma 1, CR-GreedyRPL için tasarlanan algoritmanın DAO mesaj akışı aşamasına ait sözde kodunu göstermektedir.

---

**Algoritma 1.** DAO Mesaj Akışı

---

```

1  upon tüm düğümler için sendDAO(v, G) ile DAO mesajı
2      foreach düğüm v ∈ VG do
3          foreach parent p ∈ ParentList(v) do
4              if p.rank = p'.rank then
5                  OF'ye göre rastgele ebeveyn düğüm seç;
6                  v.parent ← rastgele (OF (max/min(p.rank)))
7                  düğümünü ata;
8                  sendDAO (p, G);
9              else
10                 OF'ye göre ebeveyn düğüm seç;
11                 v.parent ← OF (max/min(p.rank)) düğümünü ata;
12                 sendDAO (p, G);
13         end upon

```

---

Algoritma 2, CR-GreedyRPL için tasarlanan algoritmanın DIO mesaj akışı aşamasına ait sözde kodunu göstermektedir.

---

**Algoritma 2. DIO Mesaj Akışı**

---

```
1  foreach düğüm  $v \in V_G$  ve kenar  $e \in V_E$ 
2       $v.freq \leftarrow -\infty, e.freq \leftarrow [0, 50]$ 
3       $e.snr \leftarrow random[2.5, 20]$ 
4       $v.cap, e.cap \leftarrow None$ 
5       $v.parentList \leftarrow \emptyset$ 
6       $v.isSent = True$ 
7  upon tüm düğümler için sendDIO( $v, G$ ) ile DIO mesajı
8      foreach  $k \in N(v)$  do
9          if  $k = v$  then
10              $C_{kv}$  hesapla;
11              $k.freq \leftarrow (k, v)$  frekansını ata;
12              $e_{kv}.cap \leftarrow C_{kv}$ ;
13              $k.cap \leftarrow C_{kv}$ ;
14              $k.isSent = True$ ;
15             sendDIO ( $k, G$ )
16         else if  $k \neq v$  ve  $k \neq t$  then
17             foreach  $k \in N(v)$  do
18                  $C_{vk} (1 - \frac{t_{switch}^{vk}}{T})$  hesapla,
19                 foreach  $n \in N(k)$  do
20                     En düşük frekans gecikmesini hesapla;
21                      $t_{switch}^{vk} = \max (|v.freq - e_{vk}.freq|,$ 
22                          $|e_{vk}.freq - e_{kn}.freq|)$ ;
23                      $e_{vk}.cap \leftarrow C_{vk} (1 - \frac{t_{switch}^{vk}}{T})$ ;
24                      $k.cap \leftarrow C_{kv}$ ;
25                      $k.freq \leftarrow e_{vk}$  frekansını ata;
26                      $t_{switch}^{kn} = |e_{vk}.freq - e_{kn}.freq|$ ;
27                      $n.cap \leftarrow C_{kn} (1 - \frac{t_{switch}^{kn}}{T})$ ;
28                      $n.freq \leftarrow e_{kn}$  frekansını ata;
29                      $x.isSent = True$ ;
30                     sendDIO ( $n, G$ )
31                 else if  $k = t$  then
32                      $t_{switch}^{vk} = |e_{vk}.freq - v.freq|$ ;
33                      $v.cap \leftarrow C_{vk} (1 - \frac{t_{switch}^{vk}}{T})$ ;
34                      $v.freq \leftarrow e_{vk}$ 
35             end upon
```

---

### 3.4. Frekans Deęiřtirme Farkındalıklı RPL Algoritması (RPL-Sw)

Bu bölümde, RPL algoritması için frekans deęiřtirme enerjisini dikkate alan bir amaç fonksiyonu tanımlanacaktır. Yeni amaç fonksiyonu tanımlanan RPL-Sw algoritması Bölüm 2’de aktarılan standart RPL (Pure RPL) algoritması ile karşılaştırılacak, Pure RPL ve RPL-Sw algoritmalarına benzetim sonuçları aktarılacaktır.

Veri iletişimi, bilgisayar ağlarında önemli bir araştırma alanıdır. Özellikle algılanan ve toplanan verinin büyük miktarda olduęu ve paylaşımının gerektięi IoT ağlarında iletişim, aktif ve geliřtirmeye açık bir alandır. Heterojen cihazlardan oluřan ağ yapısında, verinin istenen zamanda ve düzeyde paylaşımı, bu iletişimin kalitesinde belirleyici olmaktadır. Düşük güç ve kayıplı yapıda olan IoT cihazlarında iletişimin sağlanması, cihazların karakteristik özellikleri düşünöldüğünde farklı zorluklar içermektedir. Çok sayıda cihazın birbirine bağlanarak oluřturduęu bu ağda, iletişim esnasında yüksek miktarda enerji tüketimi, gecikme ve düşük iletim oranı gibi sonuçlar QoS ile birlikte performansı da etkilemektedir. RPL, veri iletimi ve haberleşmenin temelini oluřturan yönlendirme işlemi için, IoT ağlarında tanımlı protokollerden birisidir. Düşük güç ve kayıplı ağlar için yönlendirme protokolü olan RPL, çeřitli yönlendirme metrikleri kullanarak verinin kaynaktan hedefe iletilmesini sağlamaktadır. Sabit spektrum bandı kullanarak yapılabilen haberleşme, CRN paradigması ile birlikte farklı iletişim yöntemlerine dönüşmüřtür. Ağ ve cihaz karakteristikleri düşünöldüğünde, mevcut IoT cihazların var olan spektrumları kullanmak dışında bir alternatifi bulunmamaktadır. Sabit frekans üzerinden, tek bir anten ile yayın yapan bu cihazlar, CR özellięi ile, frekans bantları üzerinde gezinebilir ve ortak bir frekansa geçiř yaparak haberleşebilir. Mevcut IoT yönlendirme ve ağ protokolleri frekans deęiřtirmesi işlevine izin vermemekte ve bu noktada hesaplama yapmamaktadır.

RPL yönlendirme yolu ve ağ geneline ait toplam enerji tüketimi 2.2 numaralı alt başlık kapsamında verilmiřtir. Düşümler, yönlendirme esnasında ortak frekans bandı bularak antenlerini bu banda göre ayarlayacak řekilde frekans geçiři yapmaktadırlar. Frekans geçiři işlemi için harcanan enerji tüketimi Denklem 3.18’de gösterildięi gibi hesaplanmaktadır. Frekans geçiřinde tüketilen enerji de dikkate alındığında, toplam enerji tüketimi de Denklem 3. 19’da gösterildięi gibi hesaplanmaktadır:

$$E_{Sw} = P_{Sw} t_{Sw} \text{ joule} \quad (3.18)$$

$$E_C = \sum_{\forall i,j \in P} \delta_{ij} (ETX + E_{Circuitry} + E_{Idle} + E_{Sw}) + \sum_{\forall i,j \in P} \delta_{ij} E_{Idle} \text{ joule} \quad (3.19)$$

$P_{Sw}$ , CR özellikli IoT cihazların frekans değiştirme güç değerini,  $t_{Sw}$  değeri frekans değiştirme gecikmesini temsil etmektedir.  $t_{Sw}$  değeri, Denklem 3.16'daki şekliyle hesaplanmaktadır.

### 3.4.1. RPL-Sw Benzetim Sonuçları

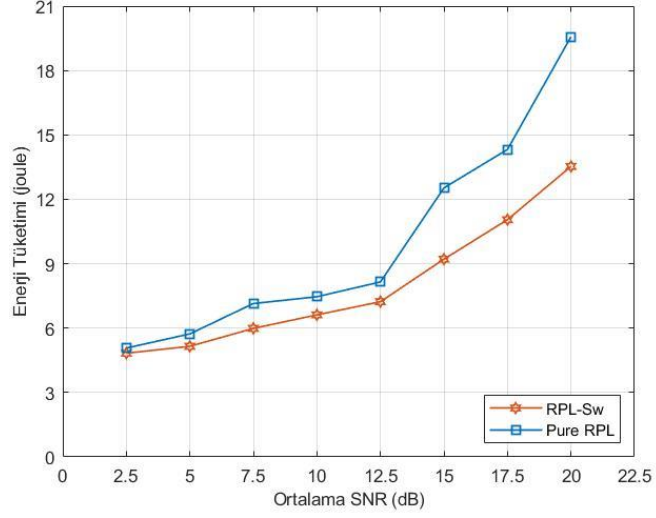
Bu bölümde, Pure RPL ve RPL-Sw algoritmalarına ait benzetim sonuçları sunulmaktadır. Python programlama dili kullanılarak yapılan benzetimin, farklı amaç fonksiyonları ve parametreler eşliğinde davranışları gözlenmiştir. Semboller ve simülasyon parametreleri Tablo 3.1'de gösterilmektedir. Benzetimler, tek bir kök düğüme sahip, statik bir DODAG ağ topolojisi üzerinde yapılmıştır.

Tablo 3.1. Benzetim parametreleri

Sembol	Tanım	Değer
$t$	Benzetim süresi	20 iterasyon
$L$	Paket boyutu	100 B
$I$	Paket gönderim aralığı	0.5 sn
$N$	Düğüm sayısı	20
$snr$	SNR değeri	[2.5, 20] dB
$P$	Cihaz gücü	1.98 W
$W$	Kanal bant genişliği	$5 \times 10^6$ Hz
$P_C$	Cihaz devre gücü	0.21 W
$P_{idle}$	Cihaz durağan hal gücü	0.99 W
$T_s$	Zaman dilimi süresi	100 ms
$\beta$	Frekans geçiş gecikmesi	0.1 ms/MHz
$P_{Sw}$	Frekans değiştirme gücü	0.75 W
$F$	Frekans sayısı	[10,50]
$p_{i,j}$	Kanal kullanım sayısı	[0.5, 1]

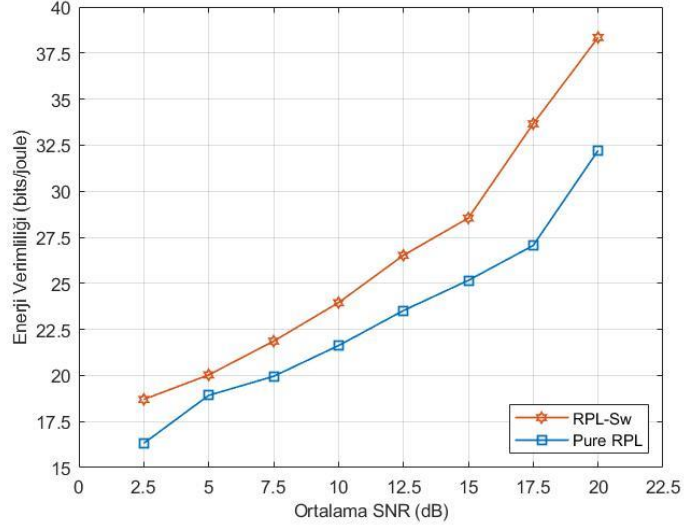
20 adet statik düğümden oluşan ve bir adet kök düğüm içerecek şekilde üretilen topoloji üzerinde benzetim yapılmıştır. Farklı ortalama değerlerde rastgele üretilen 100 adet SNR değeri düğümler arası bağlantılara atanmış ve 20 iterasyona ait ortalama değerler ile benzetim sonuçlandırılmıştır. Protokolün benzetiminde minimum enerji tüketimi, maksimum kapasite ve enerji verimliliği gibi performans metrikleri, amaç fonksiyonu olarak kullanılmıştır.

Şekil 3.6'da farklı *SNR* değerlerinde RPL algoritmalarının enerji tüketimi metriği altındaki davranışı verilmiştir. Kaynaktan hedefe olan yola ait enerji tüketim değerleri, *SNR* değeri arttıkça artmaktadır. Frekans değiştirme farkındalıklı önerilen yola ait enerji tüketimi, bu ölçütü dikkate almayan Pure RPL algoritmasına göre daha düşüktür. Pure RPL, yol seçimi esnasında frekans değiştirmeyi dikkate almadığından, daha yüksek enerji tüketimi meydana gelmektedir.



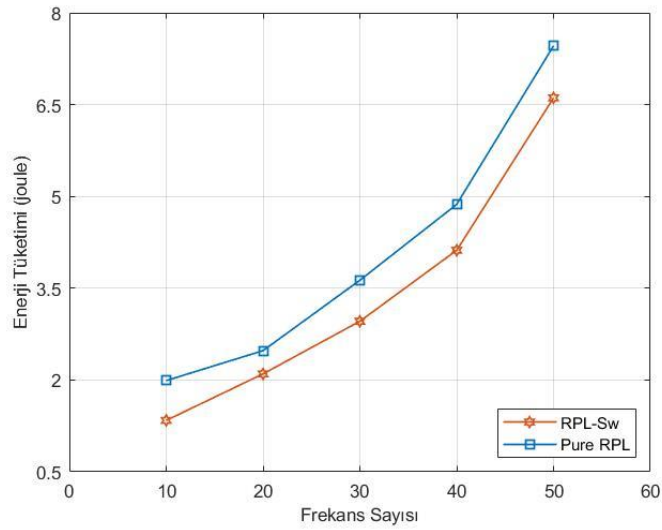
Şekil 3.6. Farklı *SNR* değerleri altında enerji tüketimi

Şekil 3.7, farklı *SNR* değerlerinde, RPL algoritmalarının enerji verimlilik metriği altındaki davranışlarını göstermektedir. Yüksek *SNR* değerinin, iletişimin kalitesini ve iletim performansını artırmasına bağlı olarak birim enerji başına gönderilen veri miktarı da artmaktadır. Frekans değiştirme gecikmesi ve buna bağlı olarak frekans değiştirmeye harcanan enerji, Pure RPL algoritmasında daha düşük enerji verimliliğine sebep olmaktadır.



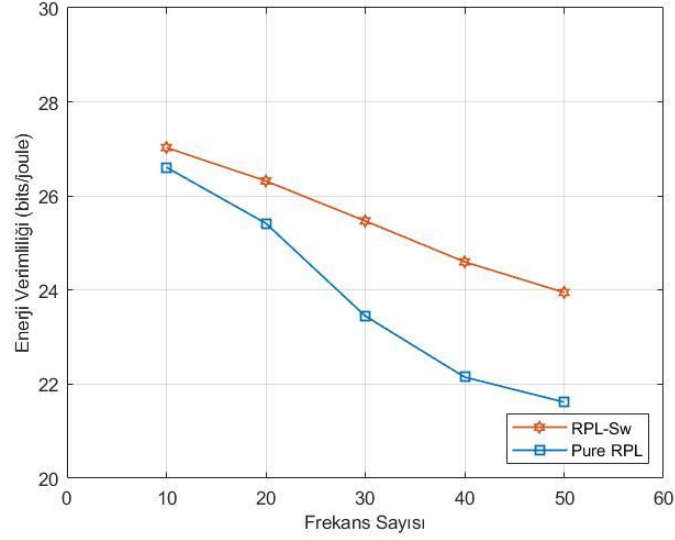
Şekil 3.7. Farklı SNR değerleri altında enerji verimliliği

Şekil 3.8, farklı frekans sayıları altında, RPL-Sw ve Pure RPL algoritmalarının enerji tüketim değerlerini göstermektedir. Frekans sayısı arttıkça, cihazların geçebileceği frekans aralığı ve buna bağlı olarak enerji tüketimi de artmaktadır.



Şekil 3.8. Farklı frekans sayıları altında enerji tüketimi

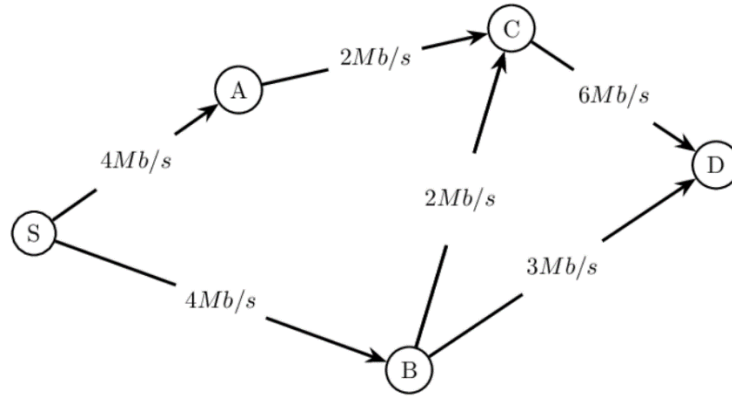
Şekil 3.9, farklı frekans sayıları altında, RPL-Sw ve Pure RPL algoritmalarının enerji verimliliğini göstermektedir. Frekans sayısı arttıkça, cihazların geçebileceği frekans aralığı ve buna bağlı olarak enerji tüketimi de artmaktadır. Enerji tüketimi arttıkça birim enerji ile taşınan veri miktarı azalmaktadır.



Şekil 3.9. Farklı frekans sayıları altında enerji verimliliği

### 3.5. Dar Boğaz Tabanlı Yol (DTY) Problemi

Yönlendirme problemi, seçilen yolun belirlenen kısıtlar dahilinde istenen performans ölçütüne uygun olarak çalışması ile doğrudan ilişkilidir. Bunun için de ağ üzerindeki akış (*flow*) trafiğinin belirlenmesi ve gerektiğinde en yüksek ya da en düşük olacak şekilde davranması gerekmektedir. Akış modeli de bunlardan birisidir. Bu modelde, belirli bir ağ üzerinde kaynak düğümden hedef düğüme ulaşıncaya kadar, akışın en yüksek değerde olması amaçlanmaktadır. Akış modeli de bu problem üzerine geliştirilmiştir. Modele göre, bir ağ üzerinde kaynağa giriş yapan akış ile hedefe ulaşan akışın, akış korunum ilkesi dolayısıyla eşit olması gerekmektedir (Trevisan, 2011). Şekil 3.10'da düğümler ve bağlantılar içeren örnek ağ topolojisinde, akış ilkesi gösterilmektedir.



Şekil 3.10. Akış modeli için örnek topoloji (Trevisan, 2011)

Şekildeki örnek ağ topolojisinde,  $S$  kaynak düğümünden  $6 \text{ Mb/s}$  'lik veri göndermek istediğimizi düşünelim.  $S$  kaynak düğümü gelen bu  $6 \text{ Mb/s}$  'lik veriyi, gönderebileceği en yüksek kapasiteyi bir linke, kalanı da diğer linke verecek şekilde bölünmektedir. Bunun sonucunda  $s$  düğümü  $4 \text{ Mb/s}$  'lik veriyi  $B$  düğümüne, kalan  $2 \text{ Mb/s}$  'lik veriyi de  $A$  düğümüne link üzerinden gönderdiğini varsayalım. Bir sonraki adımda  $B$  düğümü kendisine gelen  $4 \text{ Mb/s}$  'lik veriyi  $3 \text{ Mb/s}$ 'lik kısmını  $D$  düğümüne kalan  $1 \text{ Mb/s}$  'lik bölümünü de  $C$  düğümüne gönderir. Diğer tarafta ise,  $A$  düğümü kendisine gelen  $2 \text{ Mb/s}$  'lik veriyi herhangi bir değişiklik yapmadan  $C$  düğümüne gönderir. Bu durumda  $C$  düğümüne gelen toplamda  $3 \text{ Mb/s}$  'lik veri bulunur.  $C$  düğümü de bulundurduğu  $3 \text{ Mb/s}$  'lik veriyi herhangi bir bölmeye uğratmadan  $6 \text{ Mb/s}$  'lik link üzerinden  $D$  düğümüne gönderir. Son durumda  $D$  düğümü,  $C$  düğümünden  $3 \text{ Mb/s}$ ,  $B$  düğümü üzerinden de  $3 \text{ Mb/s}$  olacak şekilde toplamda  $6 \text{ Mb/s}$  'lik verir alır. Baştaki akış korunum ilkesi dikkate alındığında, giren  $6 \text{ Mb/s}$  ile ulaşan  $6 \text{ Mb/s}$ 'lik verinin korunduğu ve bu ilkenin sağlandığı gözlenmektedir.

Ağ üzerinde belirlenen bir kaynak düğüm ile varış noktası olarak belirlenen hedef düğüm arasındaki linklerin maksimize edilmesi, maksimum yol kapasitesi (maximum path capacity), DTY olarak literatürde tanımlanmaktadır. Dijkstra algoritmasının modifiye edilmesi ile elde edilen bu yaklaşım, öncelik kuyruğu veri yapısı kullanılarak algoritmik olarak tasarlanmaktadır. Ağ üzerindeki düğümler dikkate alınarak oluşturulan iletim yolunaki en düşük kapasiteli link, o yolun darboğazı (bottleneck) olarak nitelendirilmekte ve kapasite bu bağlantı linkine göre değer almaktadır. Bu yüzden, ağdaki bağlantı linkinin kapasitesinin en büyük olacak şekilde belirlenmesi, yönlendirmede önemli bir problemdir. Algoritma 3'te, DTY algoritmasına ait sözde kod gösterilmektedir.

---

**Algoritma 3. DTY Algoritması**

---

```
foreach düğüm  $v \in V_G$  ve kenar  $e \in V_E$ 
┌
│  $v.freq \leftarrow -\infty, e.freq \leftarrow [0, 50]$ 
│  $e.snr \leftarrow random[2.5, 20]$ 
│  $v.cap, e.cap \leftarrow None$ 
│  $v.parent \leftarrow \emptyset$ 
│  $v.isSent = True$ 
└
queue = []
queue  $\leftarrow src$ 
upon tüm düğümler için DTY( $v, G$ )
  while queue do:
    src  $\leftarrow$  queue.pop
    foreach kenar  $e \in V_E$  and düğüm  $v \in V_G$  do
      queue.append( $e$ )
      dist  $\leftarrow$  max( $e.cap(\min((src-v).cap, e.cap))$ )
      if dist >  $e.cost$  then
         $e.cost \leftarrow$  dist
         $v.parent \leftarrow$  src
upon tüm düğümler için path(src, current)
  while src  $\neq$  current do:
    print out src, current.parent düğümlerini yazdır
    current  $\leftarrow$  current.parent düğümünü ata
```

---

## 4. BENZETİM

Bu bölümde, tasarlanan enerji tüketim ve enerji verimliliği modelinin örnek çizge üzerindeki çalışması aktarılacak, Pure RPL, RPL-Sw, DTY ve CR-GreedyRPL algoritmalarına ait benzetim parametreleri sunulacak ve bu algoritmalara ait performans karşılaştırmalarına ait sonuçlar değerlendirilecektir.

### 4.1. Tasarlanan Modelin Örnek Test Çizgesinde Gösterimi

Bu bölümde Şekil 3.5'te yer alan örnek çizge kullanılarak tasarlanan enerji tüketim modelinin nasıl çalıştığı aktarılacaktır. Şekil 3.5'te verilen örnek çizge, 9 adet CR-IoT düğümünden ve 11 adet bağlantı linkinden oluşmaktadır. Toplam frekans sayısı 50 olacak şekilde belirlenmiştir. Düğüm ve düğümler arasındaki bağlantılara atanan frekans değer aralığı  $[0, 50]$ , frekanslar arasındaki mesafe 5 MHz olarak belirlenmiştir. Düğümler arasındaki bağlantı linklerinin CR-IoT düğümler tarafından kullanım olasılığı  $(0.5, 1]$  değer aralığıdır. Düğümler arasındaki linklerde yer alan kapasiteyi belirlemek için kullanılan parametre olan  $SNR$  değerleri, test çizgesi için ortalaması 10 dB olacak şekilde rastgele üssel olarak üretilmiştir. Frekanslar arasındaki geçişten kaynaklanan gecikme 0.1ms/MHz olacak şekilde, paket boyutu 1000 bayt yani 8000 bit olarak sabit değer seçilmiştir. Düğümlerin iletişim için kullanabileceği zaman dilimi süresi (time slot) 100 ms olarak belirlenmiştir. Tablo 4.1'de düğümler arasındaki bağlantılara atanan frekanslar gösterilmektedir. Tablo 4.2'de düğümler arası bağlantılara ait kapasite değerleri Mbps biriminden gösterilmektedir. Tablo 4.3'te kanal kullanım olasılıklarını gösteren olasılık matrisi gösterilmektedir.

Tablo 4.1. Frekans matrisi çizelgesi

	<i>S</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>
<i>S</i>	-	$f_{10}$	-	-	-	-	-	-	-
<i>A</i>	-	-	$f_9$	$f_{15}$	$f_{23}$	-	-	-	-
<i>B</i>	-	-	-	-	-	$f_{17}$	-	-	-
<i>C</i>	-	-	-	-	-	$f_{25}$	$f_7$	-	-
<i>D</i>	-	-	-	-	-	-	-	$f_1$	-
<i>E</i>	-	-	-	-	-	-	-	-	$f_{25}$
<i>F</i>	-	-	-	-	-	-	-	-	$f_{19}$
<i>G</i>	-	-	-	-	-	-	-	-	$f_{37}$
<i>H</i>	-	-	-	-	-	-	-	-	-

Tablo 4.2. Kapasite matrisi çizelgesi

	<i>S</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>
<i>S</i>	-	27.96	-	-	-	-	-	-	-
<i>A</i>	-	-	37.8	82.68	47.98	-	-	-	-
<i>B</i>	-	-	-	-	-	42.6	-	-	-
<i>C</i>	-	-	-	-	-	21.33	54.47	-	-
<i>D</i>	-	-	-	-	-	-	-	67.7	-
<i>E</i>	-	-	-	-	-	-	-	-	20.73
<i>F</i>	-	-	-	-	-	-	-	-	50.13
<i>G</i>	-	-	-	-	-	-	-	-	22.16
<i>H</i>	-	-	-	-	-	-	-	-	-

Tablo 4.3. Olasılık matrisi çizelgesi

	<i>S</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>
<i>S</i>	-	0.94	-	-	-	-	-	-	-
<i>A</i>	-	-	0.96	0.68	0.53	-	-	-	-
<i>B</i>	-	-	-	-	-	0.53	-	-	-
<i>C</i>	-	-	-	-	-	0.67	0.58	-	-
<i>D</i>	-	-	-	-	-	-	-	0.99	-
<i>E</i>	-	-	-	-	-	-	-	-	0.76
<i>F</i>	-	-	-	-	-	-	-	-	0.82
<i>G</i>	-	-	-	-	-	-	-	-	0.76
<i>H</i>	-	-	-	-	-	-	-	-	-

Aç gözlü algoritma sonucu oluşan yönlendirme yolu  $S \rightarrow A \rightarrow C \rightarrow F \rightarrow H$  şeklindedir. Yönlendirme yoluna ait oluşan kapasite aşağıdaki gibi hesaplanmaktadır.

$S \rightarrow A$  bağlantı linki için oluşan verimli kapasite  $27.96 \text{ Mbps} \times 0.94 = 26.28 \text{ Mbps}$  olur.  $A \rightarrow C$  bağlantı linki için oluşan verimli kapasite  $\min(82.68 \text{ Mbps} \times 0.68 \times (1 - (20 \times |f_{10} - f_{15}| \times 0.1 \text{ ms/MHz}) / T), 82.68 \text{ Mbps} \times 0.68) \times (1 - (20 \times |f_7 - f_{15}| \times 0.1 \text{ ms/MHz}) / T)) = 47.22 \text{ Mbps}$  olur.  $C \rightarrow F$  bağlantı linki için oluşan verimli kapasite  $\min(54.47 \text{ Mbps} \times 0.58) \times (1 - (20 \times |f_7 - f_{15}| \times 0.1 \text{ ms/MHz}) / T), 54.47 \text{ Mbps} \times 0.58) \times (1 - (20 \times |f_7 - f_{19}| \times 0.1 \text{ ms/MHz}) / T))$  olur.  $F \rightarrow H$  bağlantı linki için oluşan verimli kapasite  $\min(54.47 \text{ Mbps} \times 0.58) \times (1 - (20 \times |f_7 - f_{15}| \times 0.1 \text{ ms/MHz}) / T), 54.47 \text{ Mbps} \times 0.58) \times (1 - (20 \times |f_7 - f_{19}| \times 0.1 \text{ ms/MHz}) / T)) = 24.01 \text{ Mbps}$  olur.  $S \rightarrow H$  bağlantı yolunun verimli kapasite değeri  $31.24 \text{ Mbps}$  olur. Aç gözlü algoritma sonucu harcanan toplam enerji  $85,65 \text{ Joules}$  ve ortalama enerji verimliliği  $93,4 \text{ bits/Joule}$  olarak hesaplanır.

#### 4.2. Benzetim Sonuçları

Bu bölümde tez kapsamında referans olarak seçilen Pure RPL, tasarlanan RPL-Sw, DTY algoritması ve tasarlanan CR-GreedyRPL için benzetim parametreleri ve

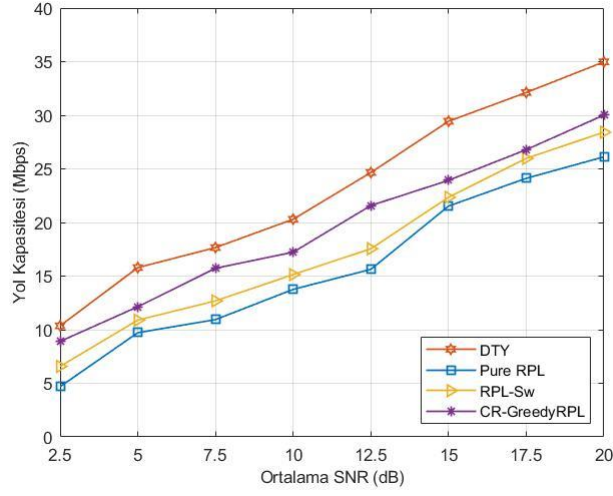
benzetim sonuçları sunulmaktadır. Python programlama dili kullanılarak yapılan benzetimin, farklı amaç fonksiyonları ve parametreler eşliğinde davranışları gözlenmiştir. 25 adet düğümden oluşan topoloji üzerinde benzetim yapılmıştır. Farklı ortalama değerlerde rastgele üretilen SNR değeri düğümler arası bağlantılara atanmış ve 30 iterasyon için ortalama değerler ile benzetim sonuçlandırılmıştır. Algoritmaların benzetimi için enerji tüketimi, kapasite ve enerji verimliliği gibi performans metrikleri kullanılmıştır. Benzetim parametreleri Tablo 4.4'te gösterilmiştir.

Tablo 4.4. Benzetim parametreleri

<b>Sembol</b>	<b>Tanım</b>	<b>Değer</b>
$t$	Benzetim süresi	30 iterasyon
$L$	Paket boyutu	1000 B
$I$	Paket gönderim aralığı	0.5 s
$N$	Düğüm sayısı	25
$snr$	SNR değeri	[2.5, 20] dB
$P$	Cihaz gücü	1.98 W
$W$	Kanal bant genişliği	$5 \times 10^6$ Hz
$P_C$	Cihaz devre gücü	0.21 W
$\alpha$	Kanal algılama zaman oranı	0.025
$\eta$	Güç verimliliği	0.9
$P_{idle}$	Cihaz durağan hal gücü	0.99 W
$T_s$	Zaman dilimi süresi	[100, 500] ms
$\beta$	Frekans geçiş gecikmesi	0.1 ms/MHz
$P_{Sw}$	Frekans değiştirme gücü	0.75 W
$F$	Frekans sayısı	[10,50]
$p_{i,j}$	Kanal kullanım oranı	[0.7, 1]
$E_{rx/bit}$	Veri alımı için enerji tüketimi	5 nJ/bit
$P_{sense}$	Algılama gücü	110 mW
$P_{sleep}$	Uyku gücü	10 mW
$P_{tx}$	İletim gücü	0.75 W

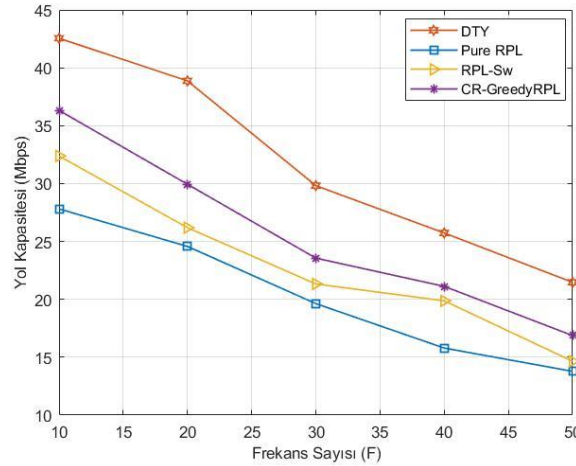
Şekil 4.1, değişen SNR değerlerinin yönlendirme yolu kapasitesine etkisini göstermektedir. SNR değerlerinin artışı sinyal kalitesini de artırdığından, kapasite artışına neden olduğu görülmektedir. DTY algoritması, bağlantılar üzerindeki kapasite değerinin en yüksek olduğu yolu en düşük kapasite değerlerine dikkat ederek seçtiği için, yol kapasitesi en yüksek çıkmaktadır. Tasarlanan CR-GreedyRPL algoritmasının, amaç fonksiyonu olarak bu çalıştırma için en yüksek kapasite değerini gözetmesi ve düğümlerin kendinden sonraki düğümlere ait yol kapasitelerini dikkate alması, yönlendirme yolunun kapasitesinin Pure RPL ve RPL-Sw algoritmalarından daha

yüksek çıkmasına sebep olur. Pure RPL algoritması, yönlendirme esnasında ebeveyn seçiminde frekans geçişini dikkate almadığından, geçiş maliyeti kapasiteyi azaltır.



Şekil 4.1. SNR değerlerinin değişiminin yol kapasitesine etkisi

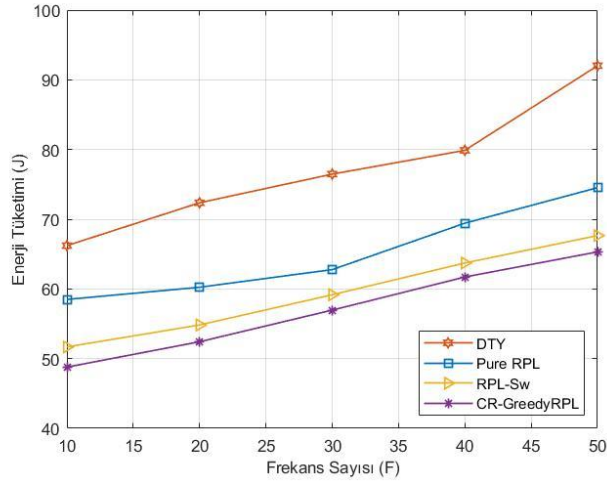
Şekil 4.2’de frekans sayısı değişiminin yol kapasitesine etkisi gösterilmektedir. Frekans sayısı artışı, düğümlerin geçiş yapabileceği frekansların da sayısını artırmaktadır. Düğümler daha uzaktaki frekansa geçiş yaptıkça kapasite değeri azalmaktadır. Yönlendirme yolu oluşturulurken frekans geçiş aralığının artması, tüm algoritmalar için kapasite değerini azaltıcı etki oluşturmaktadır.



Şekil 4.2. Frekans sayısı değişiminin kapasiteye etkisi

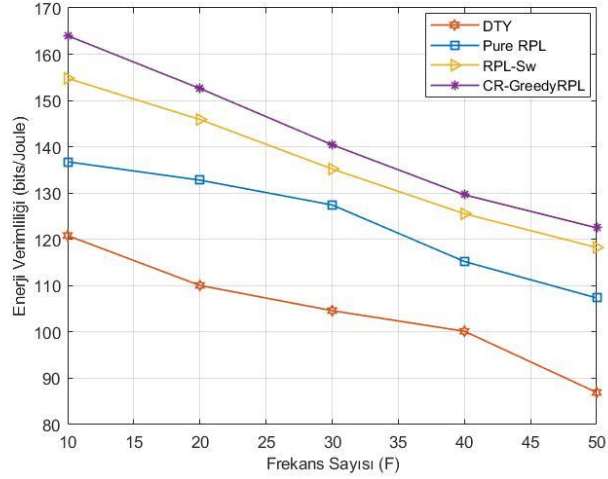
Şekil 4.3’te frekans sayısı değişiminin enerji tüketimine etkisi gösterilmektedir. Şekil incelendiğinde frekans sayısı arttıkça enerji tüketimi de artmaktadır. Frekans sayısının artışı, düğümlerin daha uzak frekanslara geçişine olanak sağlamakta, bu da

frekans geiş gecikmesini artırarak frekans geiş enerjisini artırmaktadır. CR-GreedyRPL algoritmasının frekans geişi yaparken dğümlerin kendisinden sonraki dğümlere baėlı linklere ait frekansları da dikkate alması, frekans geiş süresini ve frekans geişine baėlı enerji tüketimini azaltmaktadır. Dolayısıyla, CR-GreedyRPL algoritması daha düşük enerji tüketmektedir.



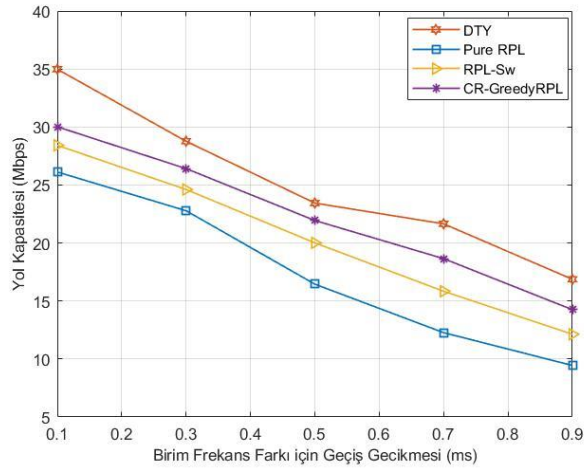
Şekil 4.3. Frekans sayısı deėişiminin enerji tüketimine etkisi

Şekil 4.4'te, frekans sayısı deėişiminin enerji verimliliğine etkisi gösterilmektedir. Frekans sayısı arttıkça, frekans geiş gecikmesine de baėlı olarak enerji tüketimi de artmaktadır. Sabit paket boyutu olduėu alıřtırmada tüketilen enerjinin artışı, birim enerjide gönderilebilecek veri sayısını azaltmaktadır. CR-GreedyRPL algoritmasının düşük enerji tüketimi, yüksek enerji verimliliğine sebep olmaktadır. Yüksek kapasite ile yönlendirme yolu seçen DTY, yüksek enerji tüketimine sahip olduğundan enerji verimliliėi en düşüktür.



Şekil 4.4. Frekans sayısı değişiminin enerji verimliliğine etkisi

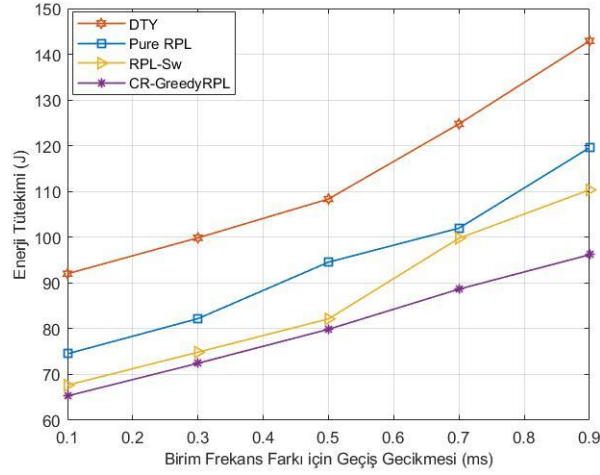
Şekil 4.5'te MHz başına frekans geçiş gecikmesinin yol kapasitesine etkisi gösterilmektedir. Birim frekans geçiş gecikmesi arttıkça yol kapasitesi bütün algoritmalar için azalmaktadır. Birim frekans gecikmesi, frekans değiştirme esnasında ortaya çıkan gecikmeyi arttıracaktır. Bir başka deyişle, birim frekans geçiş gecikmesi, MHz başına her bir düğüm için verimli kapasite değerinde azaltıcı etki yaratacağından, yol kapasitesi azalmaktadır.



Şekil 4.5. Birim frekans geçiş gecikmesinin yol kapasitesine etkisi

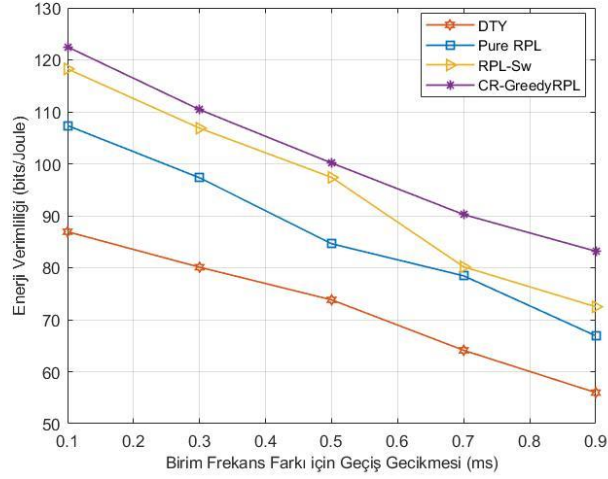
Şekil 4.6'da, MHz başına frekans geçiş gecikmesinin tüketilen toplam enerjiye etkisi gösterilmektedir. Birim frekans geçiş gecikmesi arttıkça, düğümlerin birim frekans geçişi için harcayacakları süre artmaktadır. Frekans geçiş süresinin artışı, frekans geçiş enerjisini artırmaktadır. Tüm algoritmalarda, geçiş gecikmesi artışı enerji tüketimini artırmaktadır. CR-GreedyRPL algoritmasının, frekans geçişi yaparken,

düğümün kendisinden sonraki düğümlere bağlı linklere ait frekansları da dikkate alması, frekans geçiş süresini ve enerjisini azaltmaktadır. Geçiş gecikmesinin artışının toplam enerjiyi artırmasına rağmen CR-GreedyRPL daha düşük enerji harcamaktadır.



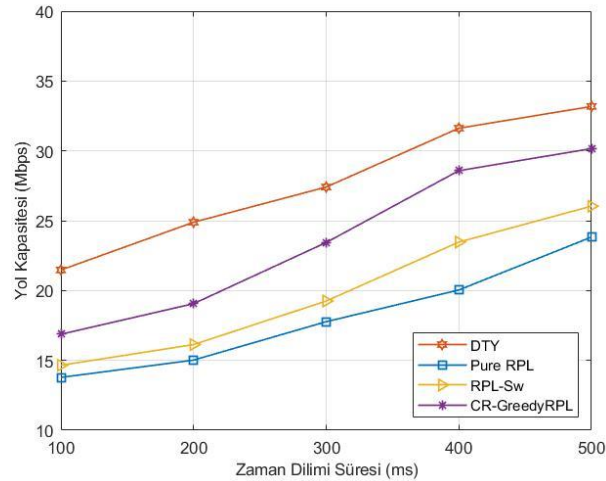
Şekil 4.6. Birim frekans geçiş gecikmesinin enerji tüketimine etkisi

Şekil 4.7’de birim frekans geçiş gecikmesinin enerji verimliliğine etkisi gösterilmektedir. Şekil incelendiğinde MHz başına frekans geçiş gecikmesinin artışı, enerji verimliliğini azaltmaktadır. Frekans geçiş gecikmesi arttıkça enerji tüketimi de artmaktadır. Sabit paket boyutu olduğu çalıştırmada tüketilen enerjinin artışı, birim enerjide gönderilebilecek veri sayısını azaltmaktadır. CR-GreedyRPL algoritmasının düşük enerji tüketimi, yüksek enerji verimliliğine sebep olmaktadır. Yüksek kapasite ile yönlendirme yolu seçen DTY, yüksek enerji tüketimine sahip olduğundan enerji verimliliği en düşüktür.



Şekil 4.7. Birim frekans geçiş gecikmesinin enerji verimliliğine etkisi

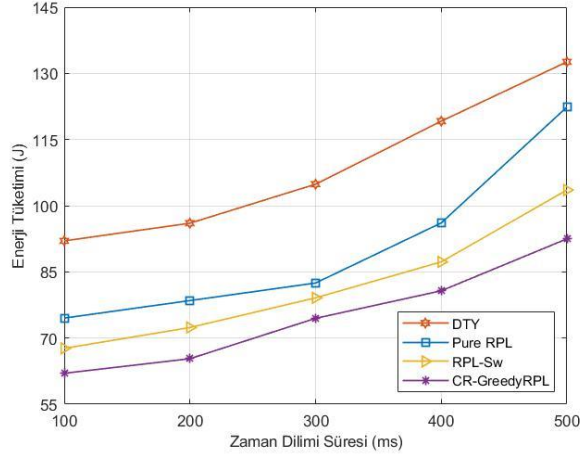
Şekil 4.8’de değişen zaman dilimi değerlerinin yol kapasitesine etkisi gösterilmektedir. Zaman dilimi değerinin artışı, yönlendirme yol kapasitesini artırdığı görülmektedir. Düğümlerin veri transferi için kullanabileceği zaman diliminin artışı, veri iletimi için olumlu etki yapmaktadır. CR düğümler, zaman dilimi sürelerinin bir kısmını iletim öncesindeki ve sonrasındaki işlemler için harcamaktadır. Geriye kalan süre, iletişim için kullanılabilir durumdadır. Bu süre, düğümlerin zaman dilimi süreleri arttıkça artacağından, verimli kapasite değeri artmaktadır.



Şekil 4.8. Zaman dilimi değerlerinin yol kapasitesine etkisi

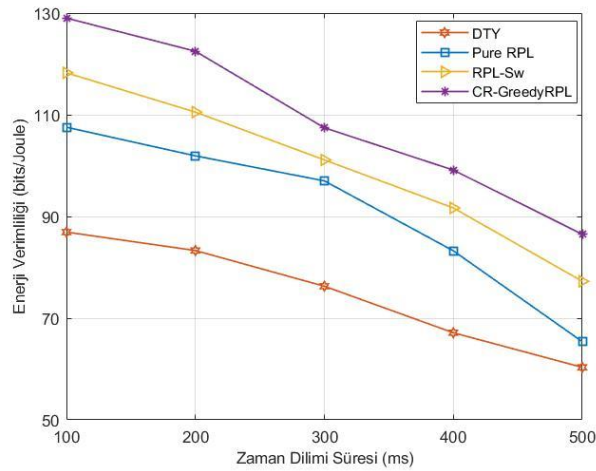
Şekil 4.9’da değişen zaman dilimi sürelerinin, yönlendirme boyunca tüketilen toplam enerjiye etkisi gösterilmektedir. İletişim halinde olan düğümler daha fazla

zaman dilimi süresinde daha fazla veri transferi yapacağından tüketilen enerji de artacaktır.



Şekil 4.9. Zaman dilimi değerlerinin enerji tüketimine etkisi

Şekil 4.10'da değişen zaman dilimi sürelerinin, enerji verimliliğine etkisi gösterilmektedir. İletişim halinde olan düğümler daha fazla zaman dilimi süresinde daha fazla veri transferi yapacağından tüketilen enerji de artacaktır. Sabit paket boyutu olduğu çalıştırmada tüketilen enerjinin artışı, birim enerjide gönderilebilecek veri sayısını azaltmaktadır. CR-GreedyRPL algoritmasının düşük enerji tüketimi, yüksek enerji verimliliğine sebep olmaktadır. Yüksek kapasite ile yönlendirme yolu seçen DTY, yüksek enerji tüketimine sahip olduğundan enerji verimliliği en düşüktür.



Şekil 4.10. Zaman dilimi değerlerinin enerji verimliliğine etkisi

## 5. SONUÇ VE DEĞERLENDİRME

Bu bölümde, tezin geneline ait bir değerlendirme sunulacak, katkılar ile gelecek çalışmalar için öneriler verilecektir.

IoT kavramı ile, internete bağlanan cihaz sayısının son yıllarda artması ve buna bağlı olarak ortaya çıkan adresleme sorununa, IPv4 adres yapısının geliştirilerek IPv6 adres yapısının kullanılmaya başlanmasıyla çözüm bulunmaya çalışılmıştır. Fiziksel cihazların gelenekselden daha akıllı hale gelmesi ve her an her yerde internete bağlanarak iletişim ağı kurması da internet adres protokolü olan IP yetersizliği problemi gibi spektrum yetersizliği sorununu ortaya çıkarmıştır. Sınırlı sayıda spektrum bulunması ve var olan bu spektrum bantlarının da cihazlar birincil kullanıcılar verimli kullanılmaması, frekans bantlarının atıl durumda kalmasına sebep olmaktadır. Bu sebeple ortaya atılan CR teknolojisi ve DSA ile spektrum yetersizliği problemine çözüm bulmaya çalışılmaktadır. Lisanslı olmayan kullanıcıların, lisanslı kullanıcılara atanan frekans bantlarını kullanmaya çalışması, ağ üzerindeki trafik yükünü artırmakta ve kullanıcıların çakışmalar ile düşük kalitede iletişim yapmasına neden olmaktadır.

Tez kapsamında, IoT ağlar için ve CRN için literatürde var olan yönlendirme algoritmaları derlenmiştir. Ayrıca, IoT ağlarda çalışan yönlendirme protokolü olan RPL incelenmiş ve benzetimi yapılmıştır. Düşük güç ve kayıplı cihazlar için tasarlanan RPL algoritmasının, farklı frekanslara geçiş yapabilme yeteneğinin olmamasından hareketle, RPL-Sw önerilmiştir. Önerilen RPL-Sw ile düğümlerin, yönlendirme yolu oluştuktan sonra frekans geçiş maliyetleri ve bu maliyetlerin performans ölçütlerine etkileri incelenmiştir. CR düğümler kullanılabilir durumdaki frekans bantlarını algılar, bu frekanslara geçiş yapar ve kendilerine ayrılan zaman diliminde bant üzerinde iletişim yapar. Frekans geçiş maliyeti, en uygun durumdaki frekansa geçişe doğrudan bağlıdır. IoT ağlarında mevcut yönlendirme protokolleri, yönlendirmede frekans geçişini dikkate almamaktadır. Yönlendirme yolu, frekans geçiş maliyetine bakılmaksızın oluşturulmaktadır. Bu bağlamda, CR tabanlı IoT ağlarda çalışması amaçlanan ve frekans geçişini dikkate alan CR-GreedyRPL algoritması tasarlanmıştır. Tasarlanan bu modeller, standart RPL algoritması ve DTY algoritması ile benzer hesaplama metrikleri ile performans bakımından karşılaştırılmıştır. DTY algoritması, kapasite bakımından her durumda en yüksek değere sahiptir ve buna bağlı olarak frekans geçişini dikkate almadığından enerji tüketimi yüksek, enerji verimliliği

düşüktür. CR-GreedyRPL algoritmasına ait enerji tüketimi ve buna bağlı olarak enerji verimliliği değerleri daha başarılı sonuçlar vermiştir.

Mevcut kablosuz sistemler ve ağ cihazları düşünüldüğünde, cihazların farklı frekanslara geçiş yaparak haberleşmesi ve ağ trafiği oluşturması gereksinimleri, geniş frekans aralıklarında çalışan CRN teknolojisi için ön plana çıkmaktadır. Düşük güç ve kayıplı IoT cihazlar da farklı frekans bantlarına geçiş yapamayacak özelliktedir. Bu tez kapsamında yapılan CR-IoT kompozisyonunun, frekans değiştirmenin gerekli olduğu ve değiştirmeye bağlı olarak ortaya çıkan gecikme ve enerji tüketiminin ihmal edilemeyecek düzeyde olduğu askeriye, uydu ağları gibi alanlar dışında da IoT cihazların farklı frekanslarda çalışmasına katkıda bulunması hedeflenmektedir. Bu sayede ağ trafik yoğunluğunun azalması, paket iletim oranı ve iletişim kalitesinin artması amaçlanmaktadır.

Tezin katkıları aşağıdaki gibi özetlenebilir:

- Literatürde yer alan yönlendirme çalışmaları, hem IoT hem de CRN yapıları için özetlenmiştir.
- Düşük güç ve kayıplı ağlarda çalışan RPL algoritması, CR özellikli düğümler için de çalışacak şekilde yeniden düzenlenmiş (RPL-Sw) ve frekans değiştirme özelliğini dikkate alacak şekilde yeni bir amaç fonksiyonu tasarlanmıştır. Bu sayede IoT düğümlerin CR düğümler gibi davranarak yönlendirme yolunu frekans geçişi farkındalıklı olarak seçmesi sağlanmıştır.
- Var olan DTY problemi tarafından kaynaktan hedef düğüme kadar olan yönlendirme yoluna ait kapasite, enerji tüketimi ve enerji verimliliği hesaplamaları yapılmıştır.
- Ağ üzerinde bulunan düğümlerin haberleşme esnasındaki tükettiği enerji ve sahip olduğu enerji verimliliği hesaplamaları için yöntemler formüle edilmiştir. Ayrıca kapasite hesabı için standart hesaplama yöntemi, CR düğümleri dikkate alacak şekilde düzenlenmiştir.
- Aç gözlü yönlendirme algoritması, her düğümün tek bir antene sahip olması ve farklı frekanslara geçiş yapabilmesi senaryosu dikkate alınarak CR özellikli ve RPL algoritması kısıtları ile çalışacak şekilde iyileştirilmiştir.

Tasarlanan CR-GreedyRPL algoritmasına ait kapasite, enerji tüketimi ve enerji verimliliği sonuçları alınmıştır.

- Var olan algoritmalar ve önerilen algoritmalar, belirlenen parametreler eşliğinde performans metrikleri bakımından karşılaştırılmıştır.

Günümüz kablosuz sistemlerinde cihazların haberleşmesi ve veri iletiminde daha kaliteli hizmet ihtiyacı, farklı sistemlerin kullanımını beraberinde getirmiştir. Alıcı ve verici cihazlarda tek bir anten var olması sebebiyle, iletişim bu antenler üzerinden gerçekleşir. Tek Girişli Tek Çıkış (Single Input Single Output-SISO) olarak adlandırılan bu sistemler, artan kapasite ve kaliteli iletişim ihtiyacına cevap vermek adına Çoklu Girişli Çok Çıkışlı (Multiple Input Multiple Output-MIMO) sistemlere evrilmiştir. MIMO sisteminde cihazlar, birden fazla sayıda verici ve alıcı antene sahiptir. Verici ve alıcıda birden çok antenin kullanılması ile yüksek veri hızları ve kapasite artışları sağlanmıştır (Turan, 2015). MIMO sistemler, farklı teknikler kullanarak link üzerindeki hata oranını azaltmak, yüksek veri iletim hızlarına çıkmak ve link kapasitesini artırmak gibi nitelikleri amaçlamaktadır.

CRN ağları ve IoT ağlarında çalışan cihazlar, tek bir antene sahiptir. Tez kapsamında tasarlanan ve CRN tabanlı IoT ağlarında çalışması presibine dayanan CR-GreedyRPL algoritmasında her cihaz, veri iletim ve alım aşamalarında, diğer cihaz ile iletişim kurmak için ortak bir frekans bandına antenlerini ayarlamalıdır.

Gelecek çalışmalarda, daha yüksek kanal kapasitesine çıkmak ve veri iletim performansını artırmak amacıyla MIMO sistemlerin karakteristik özelliklerinin, CRN tabanlı IoT ağlarda kullanılması amaçlanmaktadır. Dolayısıyla, MIMO sistemlerinde yer alan, verinin her bir antenden eş zamanlı olarak gönderilmesi ilkesi baz alınarak ilave güç tüketimi ve ilave bant genişliği ihtiyacının ortaya çıkmaması sağlanabilir.

## KAYNAKLAR

- Akyildiz, I. F., Lee, W. Y., Vuran, M. C., & Mohanty, S. (2006). NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer networks*, 50(13), 2127-2159.
- Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE communications surveys & tutorials*, 17(4), 2347-2376.
- Ali, A., Iqbal, M., Baig, A., & Wang, X. (2011). Routing techniques in cognitive radio networks: A survey. *International Journal of Wireless & Mobile Networks*, 3(3), 96-110.
- Alvi, S. A., Shah, G. A., & Mahmood, W. (2015). Energy efficient green routing protocol for internet of multimedia things. *IEEE tenth international conference on intelligent sensors, sensor networks and information processing (ISSNIP)* (pp. 1-6). IEEE.
- Barbato, A., Barrano, M., Capone, A., & Figiani, N. (2013). Resource oriented and energy efficient routing protocol for IPv6 wireless sensor networks. *IEEE Online Conference on Green Communications (OnlineGreenComm)* (pp. 163-168). IEEE.
- Bayhan, S., & Alagoz, F. (2012). Scheduling in centralized cognitive radio networks for energy efficiency. *IEEE Transactions on Vehicular Technology*, 62(2), 582-595.
- Bhatia, D., & Sharma, D. P. (2016). A comparative analysis of proactive, reactive and hybrid routing protocols over open source network simulator in mobile ad hoc network. *International Journal of Applied Engineering Research*, 11(6), 3885-3896.
- Bhattacharyya, T. R., & Pushpalatha, M. (2018). Routing protocols for internet of things: a survey. *International Journal of Engineering & Technology*, 7(2.4), 196-199.
- Boubekour, F., Blin, L., Leone, R., & Medagliani, P. (2015). Bounding degrees on RPL. *11th ACM Symposium on QoS and Security for Wireless and Mobile Networks* (pp. 123-130).
- Cai, Z., Duan, Y., & Bourgeois, A. G. (2015). Delay efficient opportunistic routing in asynchronous multi-channel cognitive radio networks. *Journal of Combinatorial Optimization*, 29(4), 815-835.
- Cesana, M., Cuomo, F., & Ekici, E. (2011). Routing in cognitive radio networks: Challenges and solutions. *Ad Hoc Networks*, 9(3), 228-248.
- Chang, L. H., Lee, T. H., Chen, S. J., & Liao, C. Y. (2013). Energy-efficient oriented routing algorithm in wireless sensor networks. *IEEE International Conference on Systems, Man, and Cybernetics* (pp. 3813-3818). IEEE.
- Demirci, S. (2018). Bilişsel radyo ağlarında frekans atama, kanal zamanlama ve yönlendirme. Basılmamış Doktora Tezi. Ege Üniversitesi Uluslararası Bilgisayar Ana Bilim Dalı, 123, İzmir.
- Gaddour, O., Koubâa, A., & Abid, M. (2015). Quality-of-service aware routing for static and mobile IPv6-based low-power and lossy sensor networks using RPL. *Ad Hoc Networks*, 33, 233-256.
- Gan, W., Shi, Z., Zhang, C., Sun, L., & Ionescu, D. (2013). MERPL: A more memory-efficient storing mode in RPL. *19th IEEE International Conference on Networks (ICON)* (pp. 1-5). IEEE.

- Hashem, M., Barakat, S. I., & AttaAlla, M. A. (2017). Enhanced tree routing protocols for multi-hop and multi-channel cognitive radio network (EMM-TRP). *Journal of Network and computer applications*, 100, 69-79.
- Hoghooghi, S., & Javidan, R. (2020). Proposing a new method for improving RPL to support mobility in the Internet of things. *IET Networks*, 9(2), 48-55.
- Iova, O., Theoleyre, F., & Noel, T. (2015). Using multiparent routing in RPL to increase the stability and the lifetime of the network. *Ad Hoc Networks*, 29, 45-62.
- Kim, H. S., Kim, H., Paek, J., & Bahk, S. (2016). Load balancing under heavy traffic in RPL routing protocol for low power and lossy networks. *IEEE Transactions on Mobile Computing*, 16(4), 964-979.
- Lamaazi, H., & Benamar, N. (2020). A comprehensive survey on enhancements and limitations of the RPL protocol: A focus on the objective function. *Ad Hoc Networks*, 96, 102001.
- Levis, P., Clausen, T., Hui, J., Gnawali, O., & Ko, J. (2011). The trickle algorithm (rfc 6206). Internet Engineering Task Force (IETF), 1-13.
- Liang, Y. C., Chen, K. C., Li, G. Y., & Mahonen, P. (2011). Cognitive radio networking and communications: An overview. *IEEE transactions on vehicular technology*, 60(7), 3386-3407.
- Lim, C. (2019). A survey on congestion control for RPL-based wireless sensor networks. *Sensors*, 19(11), 2567.
- Lin, M., Kim, H. S., & Bahk, S. (2015). Transmission power control for large scale industrial applications in low power and lossy networks. *International Conference on Information and Communication Technology Convergence (ICTC)* (pp. 380-382). IEEE.
- Mitola, J. I. (2002). Cognitive radio. An integrated agent architecture for software defined radio.
- Musaddiq, A., Zikria, Y. B., & Kim, S. W. (2018). Energy-aware adaptive trickle timer algorithm for RPL-based routing in the internet of things. *28th International Telecommunication Networks and Applications Conference (ITNAC)* (pp. 1-6). IEEE.
- Nejatian, S., Syed-Yusof, S. K., Latiff, N. A., & Asadpour, V. (2013). Integrated handoff management in cognitive radio mobile ad hoc networks. *IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)* (pp. 2887-2892). IEEE.
- Palattella, M. R., Accettura, N., Vilajosana, X., Watteyne, T., Grieco, L. A., Boggia, G., & Dohler, M. (2012). Standardized protocol stack for the internet of (important) things. *IEEE communications surveys & tutorials*, 15(3), 1389-1406.
- Park, J. H., Nam, Y., & Chung, J. M. (2013). Analysis of channel access with spectrum handoff in cluster based cognitive radio sensor networks. *International Conference on ICT Convergence (ICTC)* (pp. 232-233). IEEE.
- Parvin, S., & Rahim, M. S. (2008). Routing protocols for wireless sensor networks: a comparative study. *International Conference on Electronics, Computer and Communication* (pp. 891-894).
- Shah, R. C., & Rabaey, J. M. (2002). Energy aware routing for low energy ad hoc sensor networks. *IEEE Wireless Communications and Networking Conference Record. WCNC 2002* (Cat. No. 02TH8609) (Vol. 1, pp. 350-355). IEEE.

- Shiang, H. P., & Van Der Schaar, M. (2008). Queuing-based dynamic channel selection for heterogeneous multimedia applications over cognitive radio networks. *IEEE Transactions on multimedia*, 10(5), 896-909.
- Shu, T., Krunz, M., & Vruthula, S. (2006). Joint optimization of transmit power-time and bit energy efficiency in CDMA wireless sensor networks. *IEEE Transactions on Wireless Communications*, 5(11), 3109-3118.
- Talay, A. C., & Altılar, D. T. (2013). Self adaptive routing for dynamic spectrum access in cognitive radio networks. *Journal of Network and Computer Applications*, 36(4), 1140-1151.
- Trevisan, L., (2011). *The Maximum Flow - Minimum Cut Theorem (Bölüm 9.)* Erişim: 19.11.2020, <https://people.eecs.berkeley.edu/~luca/cs261/lecture09.pdf>.
- Turan, M. (2015). MIMO sistemler için sinyal tanıma algoritmaları. Basılmamış Yüksek Lisans Tezi. İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Mühendisliği Anabilim Dalı, 79, İstanbul.
- T. Winter, et al. ,RPL: IPv6 routing protocol for low - power and lossy networks, RFC 6550 (2012) 1 – 157. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc6550>. Erişim: 20.12.2020.
- Vasseur, J. (2011). Terminology in low power and lossy networks. Work in Progress.
- Wang, Y., Zheng, G., Ma, H., Li, Y., & Li, J. (2018). A joint channel selection and routing protocol for cognitive radio network. *Wireless Communications and Mobile Computing*, 2018.
- Wei, Y., Wang, J., & Wang, J. (2013). A delay/disruption tolerant routing algorithm for iot in harsh environment. *6th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)* (pp. 143-146). IEEE.
- Witwit, A. J., & Idrees, A. K. (2018). A comprehensive review for RPL routing protocol in low power and lossy networks. *International Conference on New Trends in Information and Communications Technology Applications* (pp. 50-66). Springer, Cham.
- Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer networks*, 52(12), 2292-2330.
- Yosra, M., Mohamed, A., & Sami, T. (2016). Cognitive QoS-on demand routing protocol (CO-QORP) in cognitive radio ad-hoc network. *International Symposium on Networks, Computers and Communications (ISNCC)* (pp. 1-4). IEEE.
- Zhao, M., Ho, I. W. H., & Chong, P. H. J. (2016). An energy-efficient region-based RPL routing protocol for low-power and lossy networks. *IEEE Internet of Things Journal*, 3(6), 1319-1333.

## **EKLER**

### **Ek 1. ÖZ GEÇMİŞ**

Ferhat ARAT, Sinop'ta doğdu. Ondokuz Mayıs Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü'nden 2018 yılında mezun oldu. 2018 yılında OMÜ LEE Bilgisayar Mühendisliği Tezli Yüksek Lisans programına girdi. Aralık, 2019 yılından bu yan Samsun Üniversitesi Yazılım Mühendisliği Bölümü'nde Araştırma Görevlisi olarak görev yapan Ferhat ARAT iyi derecede İngilizce bilmektedir. Temel ilgi alanları, Bilgisayar ve İletişim Ağları, Bilgisayar Yazılımıdır. (25/05/2021).

#### **İletişim Bilgileri**

E mail: ferhat.arat@bil.omu.edu.tr/ ferhat.arat@samsun.edu.tr

ORCID ID: <https://orcid.org/0000-0002-4347-0016>

#### **Yayınlanmış Çalışmalar:**

1. Arat, F., & Demirci, S. (2019, September). Analysis of Spectrum Aware Routing Algorithms in CR Based IoT Devices. In 2019 4th International Conference on Computer Science and Engineering (UBMK) (pp. 751-756). IEEE.
2. Arat, F., & Demirci, S. (2020, April). Energy and QoS Aware Analysis and Classification of Routing Protocols for IoT and WSN. In 2020 7th International Conference on Electrical and Electronics Engineering (ICEEE) (pp. 221-225). IEEE.
3. Arat, F., & Demirci, S.(2021), Experimental Analysis of Energy Efficient and QoS Aware Objective Functions for RPL Algorithm in IoT Networks. Sakarya University Journal of Computer and Information Sciences (SAUCIS).
4. Arat, F., & Demirci, S.(2021), A Novel Objective Function for Frequency Switching Cost Aware RPL Algorithm, değerlendirme aşamasında

## Ek 2. BENZETİMLERE AİT PROGRAM PARÇALARI

Ek 2 bölümü, tez kapsamında tasarlanan ve uygulanan algoritmalara ait benzetim sınıflarından oluşmaktadır.

### *RPL mesaj sınıfları*

```
class Message:
    def __init__(self, msgId, rank):
        self.msgId = msgId
        self.rank = rank

class DIOMsg(Message):
    def __init__(self, msgId, rank, msgType):
        Message.__init__(self, msgId, rank)
        self.msgType = msgType

    def getMessage(self):
        print("Message Details: ", "ID: ", self.msgId, "RANK: ", self.rank, "TYPE: ", self.msgType)

class DAOMsg(Message):
    def __init__(self, msgId, rank, msgType, delayDAO):
        Message.__init__(self, msgId, rank)
        self.msgType = msgType
        self.delayDAO = 1

    def getMessage(self):
        print("Message Details: ", "ID: ", self.msgId, "RANK: ", self.rank, "TYPE: ", self.msgType)
```

### *Topolojiye ait Graph sınıfı*

```
class Graph:
    def __init__(self):
        self.nodes = {}

    def add_nodes(self, nodeList):
        for node in nodeList:
            self.nodes[node.nodeName] = node.neighbors

    def get_node_list(self):
        print(self.nodes)

    def get_node_ids(self):
        l = []
        for n in self.nodes:
            l.append(n.nodeName)
        return l

    def add_edge(self, node_from, node_to):
        self.nodes[node_from.nodeName] = node_from.neighbors
        self.nodes[node_to.nodeName] = node_to.neighbors

    def add_edges(self, edgeList):
        for edge in edgeList:
            self.add_edge(edge[0], edge[1])

    def edge_list(self):
        if len(self.nodes) >= 1:
            for key in self.nodes.keys():
                print(str(key) + " : " + str(self.nodes[key]))
        else:
            return dict()
```

### *Topolojide yer alan düğümlere ait Node sınıfı*

```
class Node:
    def __init__(self, nodeName, nodeId, nodeRank, parentList, choosenParent):
        self.nodeName = nodeName
        self.nodeId = nodeId
        self.nodeRank = nodeRank
        self.choosenParent = choosenParent
        self.parentList = []
        self.neighbors = []
        self.neighborObjs = []
        self.parentRank = []
        self.isSent = False
        self.isReceived = False
        self.current_frequency = 0
        self.neighbor_with_snr = []
        self.fr_set = []
        self.parent_list_with_of = {}

    def add_neighbor(self, neighbor):
        self.neighbors.append(neighbor)
        #neighbor.neighbors.append(self.nodeName)

    def add_neighbors(self, neighborList):
        for neighbor in neighborList:
            self.neighbors.append(neighbor)
            #neighbor.neighbors.append(self.nodeName)

    def add_neighborObj(self, neighbor):
        self.neighborObjs.append(neighbor)
        #neighbor.neighbors.append(self.nodeName)

    def add_neighborsObj(self, neighborObjList):
        for neighbor in neighborObjList:
            self.neighborObjs.append(neighbor)
            #neighbor.neighbors.append(self.nodeName)

    def get_neighbor(self):
        for n in self.neighbors:
            print(self.neighbors)
```

### *Topolojide yer alan düğümlere başlangıç değeri atanmasına ait modül*

```
for node in g:
    for n in node.get_connections():
        n.cost = -math.inf
        n.node_snr = snr_list[random.randrange(11)]
        n.freq = random.randrange(1,50)
queue.append(g.get_node('s'))

for node in g:
    for n in node.get_connections():
        print(n.get_id(), n.freq)
```

## RPL yukarı yönlü iletim modülüne ait program parçası

```
def runUpward(self, daoMsg, last_nodeList, of):
    if(of != "thr"):
        start = time.time()
        for n in last_nodeList:
            print(daoMsg.msgType + " is sending..")
            if len(n.parentList) > 0:
                minPar = n.parentRank.pop(0)
                realP = n.parentList.pop(0)
                for index,p in enumerate(n.parentList):
                    if n.parentRank[index] < minPar:
                        minPar = n.parentRank[index]
                        realP = n.parentList[index]
                n.parentRank.clear()
                n.parentList.clear()
                n.parentRank.append(minPar)
                n.parentList.append(realP)
                n.chosenParent = n.parentList[0].nodeName
                print("For "+str(n.nodeName) + ", Chosen parent : "+ str(n.parentList[0].nodeName) +
                    ", link throughput : "+str(round(n.parentRank[0]/pow(10,6),3))+ " Mbps")
                self.write_file((n.parentRank[0]/pow(10,6),3))
                pathStr.append(n.nodeName)
                done= time.time()
                self.timeDif(start, done, daoMsg)
                self.runUpward(daoMsg,n.parentList, of)
            else:
                print(str(n.nodeName) + " is root node. No parent.")
                pathStr.append(self.nodeName)
                print("Possible RPL Path is: ")
                print(pathStr)
        else:
            start = time.time()
            for n in last_nodeList:
                print(daoMsg.msgType + " is sending..")
                if len(n.parentList) > 0:
                    minPar = n.parentRank.pop(0)
                    realP = n.parentList.pop(0)
                    for index,p in enumerate(n.parentList):
                        if n.parentRank[index] > minPar:
                            minPar = n.parentRank[index]
                            realP = n.parentList[index]
                    n.parentRank.clear()
                    n.parentList.clear()
                    n.parentRank.append(minPar)
                    n.parentList.append(realP)
                    n.chosenParent = n.parentList[0].nodeName
                    print("For "+str(n.nodeName) + ", Chosen parent : "+ str(n.parentList[0].nodeName) +
                        ", link throughput : "+str(round(n.parentRank[0]/pow(10,6),3))+ " Mbps")
                    self.write_file((n.parentRank[0]/pow(10,6),3))
                    pathStr.append(n.nodeName)
                    done= time.time()
                    self.timeDif(start, done, daoMsg)
                    self.runUpward(daoMsg,n.parentList, of)
                else:
                    print(str(n.nodeName) + " is root node. No parent.")
                    pathStr.append(self.nodeName)
                    print("Possible RPL Path is: ")
                    print(pathStr)
```

### *Düğümlere ait frekans değiştirme ve ortak frekansa geçiş işlevlerine ait modül*

```
def find_common_freq(self,n):
    print(self.nodeName, self.current_frequency, n.nodeName, n.current_frequency)
    lst3 = [value for value in self.fr_set if value in n.fr_set]
    if(lst3 != []):
        print("Common frequencies")
        print(lst3)
        selected = lst3[random.randint(0, len(lst3))-1]
        print("selected frequency = " + str(selected))
        return selected
    else:
        print("No common frequency")
        return None
```

### *Frekans değiştirme işlemi esnasındaki gecikmeyi hesaplayan modül*

```
def sw_delay(self, n, sel):
    d1 = 20 * abs(self.current_frequency - sel) * delay_per_fr
    d2 = 20 * abs(n.current_frequency - sel) * delay_per_fr
    max_delay = max(d1, d2)
    print("Channel switching delay: " + str(max_delay))
    self.write_sw(max_delay)
    return max_delay
```