



**KABLOSUZ YEREL ALAN AĞLARINDA MAC KATMANINDA
ANALİTİK YÖNTEM KULLANARAK PERFORMANS GELİŞTİRME**

Hacı Bayram KARAKURT

**DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

MART 2021

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Hacı Bayram Karakurt

08/03/2021

KABLOSUZ YEREL ALAN AĞLARINDA MAC KATMANINDA ANALİTİK YÖNTEM KULLANARAK PERFORMANS GELİŞTİRME

(Doktora Tezi)

Hacı Bayram KARAKURT

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Mart 2021

ÖZET

Kablosuz yerel alan ağlarında (Wireless Local Area Network-WLAN) hizmet kalitesini (Quality of Service-QoS) arttırmak için genellikle 802.11e Ortam Erişim Kontrolü (Medium Access Control-MAC) protokolü kullanılmaktadır. MAC protokolünde hizmet kalitesini etkileyen parametrelerden RTS (Request to Send) Eşik Değeri (RTSED), Parçalama Eşik Değeri (PED) ve Arabellek Boyutu (AB) ağ performansını etkileyen en önemli parametreler arasında yer almaktadır. Bu tez çalışmasında RTSED, PED ve AB giriş parametreleri bulanık mantık, Yapay Sinir Ağları (YSA) ve geri besleme kontrol tabanlı modellerle optimize edilmiştir. Böylece IEEE 802.11e Çarpışma Önleme ile Taşıyıcı Algılama Çoklu Erişim (CSMA/CA) protokolünde kanal verimliliği ve birim zamanda işlenen paket miktarı artırılmış, paket kayıpları ve gecikme süresi azaltılmıştır. WLAN'larda ilk olarak veri setleri ile yapılan çalışmada bulanık mantık yöntemi kullanılarak gecikme süresinde %36 ile %38 aralığında, yükleme süresinde %2 ile %10 aralığında ve birim zamanda işlenen paket miktarında %25 ile %44 aralığında iyileştirmeler elde edilmiştir. İkinci çalışmada YSA algoritması ile alıcı ve verici için kanal kullanım durumu (Channel Utilization Receiver, Channel Utilization Transmitter), alınan veri trafiği ve gönderilen veri trafiğinin tahminlenmesi sağlanmıştır. Üçüncü çalışmada WLAN'larda dış faktörlerin hizmet kalitesine etkisini incelemek amacıyla düğümlerin konumuna göre iç ve dış ortamlara kablosuz sunucu, telekom vericisi ve jammerlar (sinyal bozucular) eklenerek RTSED, PED ve AB giriş parametrelerinin kanal verimliliğine olan etkisi incelenmiştir. Ayrıca döngü tablosu kullanılarak kaba kuvvet algoritması ile kanal verimliliği artırılmıştır. Bu yapıda yeni bir model olan ajan yapısı ile ağ katmanı, düğüm katmanı ve proses katmanları zamanlama fonksiyonu ile güncellenmiş ve kanal kullanım durumu %17 oranında artırılmıştır. Son olarak tez çalışmasının temelini de oluşturan çalışmada ise geri besleme kontrollü yöntem ile Riverbed Modeller üzerinden gömülü bulanık mantık algoritması kullanılarak benzetim anında birim zamanda işlenen paket miktarı %26,48, kanal kullanım durumu %2,30, alınan veri trafiği %14,59 ve gönderilen veri trafiği ise %17,06 oranında artırılmıştır. Bu çalışmalarda kullanılan geri besleme kontrollü algoritma ile RTSED, PED ve AB giriş parametreleri ideal hale getirilirken; düğüm sayısı, varışlar arası süre, iletim gücü v.b dış parametrelerinde yeni modelde performans iyileştirmelerine olan etkileri grafiklerle gösterilmiştir. Bütün bu test sonuçları yeni modelin WLAN'lar için yüksek oranda performans iyileştirmesi sağladığını göstermiştir.

Bilim Kodu : 92432

Anahtar Kelimeler : Kablosuz yerel alan ağları, RTS eşik değeri, parçalama eşik değeri, bulanık mantık, yapay sinir ağları, geri besleme kontrolü

Sayfa Adedi : 114

Danışman : Dr. Öğr. Üyesi Cemal KOÇAK

PERFORMANCE EVALUATION WITH USING ANALYTIC METHOD IN MAC LAYER FOR WIRELESS LOCAL AREA NETWORKS

(Ph. D. Thesis)

Hacı Bayram KARAKURT

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

March 2021

ABSTRACT

In order to increase the Quality of Service (QoS) in wireless local area networks (WLAN), the 802.11e Medium Access Control (MAC) protocol is generally used. Among the parameters affecting the service quality in the MAC protocol, RTS (Request to Send), Threshold Value (RTSTV), Fragmentation Threshold Value (FTV) and Buffer Size (BS) are the most important parameters that affect network performance. In this thesis, RTSTV, FTV and BS input parameters have been optimized with fuzzy logic, Artificial Neural Networks (ANN) and feedback-based control models. In this way, with the IEEE 802.11e Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol, the channel utilization and throughput increased, packet losses and delay are reduced. With the first study with data sets, in WLANs, by using fuzzy logic method, between 36% and 38% improvement in delay, between 2% and 10% improvement in load and between 25% and 44% improvement in throughput were achieved. In the second study, estimation of channel utilization receiver, channel utilization transmitter, data traffic received and data traffic sent were provided with ANN algorithm. In the third study, in order to examine the effect of external factors on QoS in WLANs, the effect of RTSTV, FTV and BS input parameters on channel utilization was examined by adding wireless servers, telecom transmitters and jammers to indoor and outdoor locations according to the position of the nodes. Besides, by using the look-up table with brute force algorithm, the channel utilization has been increased. With the agent structure, which is a new model, the network layer, node layer and process layers have been updated with the timing function and the channel utilization has been increased by 17%. Finally, in the study that also forms the basis of the thesis, by using the feedback controlled method and the embedded fuzzy logic algorithm over Riverbed Models, at the time of simulation the throughput has been increased by 26,48%, the channel utilization has been increased by 2,30%, the data traffic received has been increased by 14,59% and the data traffic sent has been increased by 17,06%. While RTSTV, FTV and BS input parameters are optimized with the feedback controlled algorithm used in these studies; the effects of external parameters such as number of nodes, interarrival time, transmit power, etc. on performance improvements in the new model has been demonstrated with the graphics. All these test results have shown that the new model provides a high rate of performance improvement for WLANs.

Science Code : 92432

Key Words : Wireless local area network, RTS threshold, fragmentation threshold, fuzzy logic, artificial neural networks, feedback control

Page Number : 114

Supervisor : Assist. Prof. Dr. Cemal KOÇAK

TEŞEKKÜR

Bu tez çalışmasının planlanmasında ve uygulanmasında ilgi, yardım ve desteğini esirgemeyen değerli danışman hocam Dr. Öğr. Üyesi Cemal KOÇAK'a yapmış olduğu yardımlar, tavsiyeler, bana vermiş olduğu cesaret ve sürekli desteğinden dolayı ve ayrıca güveni ile sabrı için kendisine şükranlarımı sunuyorum. Tez izleme komitesinde yer alan Prof. Dr. Recep DEMİRCİ hocama ve Prof. Dr. Ertan ONUR hocama tezimin ilerleme aşamasındaki görüşleri, önerileri ve araştırmamı çeşitli açılardan genişletmemizi teşvik eden yorumlarından dolayı çok teşekkür ediyorum. Bugünlere gelmemde emekleri yadsınamaz olan sevgili anneme, babama ve abime manevi desteklerinden dolayı en kalbi duygularıyla teşekkür ederim. Çalışmalarım süresince fedakârlık göstererek beni destekleyen eşime ve oğluma sabırları ve destekleri için çok teşekkür ederim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	x
ŞEKİLLERİN LİSTESİ	xi
SİMGELER VE KISALTMALAR.....	xv
1. GİRİŞ.....	1
2. WLAN’LARDA CSMA/CA, GİRİŞ/ÇIKIŞ DEĞİŞKENLERİ VE BENZETİM ORTAMI.....	13
2.1. Kablosuz Yerel Alan Ağlarında 802.11e MAC Protokolünde CSMA/CA Modeli.....	13
2.2. RTSED.....	16
2.3. PED.....	18
2.4. AB.....	19
2.5. Benzetim Ortamı ve Özellikleri.....	19
2.5.1. Riverbed Modeller.....	21
2.5.2. Riverbed Modeller ve diğer benzetim araçları arasındaki farklar.....	22
2.5.3. Riverbed Modeller yazılımının hiyerarşik yapısı.....	23
3. GERÇEKLEŞTİRİLEN BULANIK MANTIK, YSA ve GERİ BESLEME KONTROL TABANLI PERFORMANS ARTTIRMA.....	25
3.1. Bulanık Mantık ve Giriş Değişkenleri ile Performans Geliştirme.....	25
3.1.1. Üyelik fonksiyonları ve bulanık kural tabanının oluşturulması.....	26
3.1.2. RTSED, PED ve AB’nin bulanık mantıkla belirlenmesi.....	29

	Sayfa
3.1.3. Geliştirilen modelin arayüz kodlama denklemlerinin oluşturulması	30
3.2. YSA Tabanlı Giriş Değişkenleri ile Performans Geliştirme	33
3.2.1. Alınan ve gönderilen veri trafiği için YSA çalışmaları	34
3.2.2. Alıcı ve verici kanal kullanımı için YSA model çalışmaları.....	37
3.3. Geri Besleme Kontrol Tabanlı RTSED, PED ve AB Yöntem Analizleri ile Performans Geliştirme	39
3.3.1. Geri besleme kontrolü ile kanal kullanımının benzetim anında döngü tablosu ve kaba kuvvet algoritması ile optimize edilmesi.....	39
3.3.2. Geri besleme kontrolünün zaman ayarlı Op_sim_time fonksiyonu ile uygulanması.....	47
3.4. Bulanık Mantık Tabanlı Geri Besleme Kontrolü ile Düğüm Bazlı Birim Zamanda İşlenen Paket Miktarının Optimize Edilmesi.....	48
3.4.1. Geri besleme kontrolünde bulanık kural tabanının gösterimi	51
3.4.2. RTSED, PED ve AB'nin geri besleme kontrollü bulanık mantıkla belirlenmesi.....	57
4. DENEYSEL SONUÇLAR YORUMLAR.....	55
4.1. Klasik Yöntem Test Sonuçları.....	55
4.1.1. RTSED ile yükleme, gecikme ve birim zamanda işlenen paket miktarı test sonuçları	61
4.1.2. PED ile yükleme, gecikme ve birim zamanda işlenen paket miktarı test sonuçları.....	62
4.1.3. AB ile yükleme, gecikme ve birim zamanda işlenen paket miktarı test sonuçları	62
4.1.4. RTSED, PED ve AB ile alınan ve gönderilen veri trafiği, gönderici ve alıcı kanal kullanım durumu test sonuçları.....	63
4.1.5. RTSED, PED ve AB ile düğüm bazlı birim zamanda işlenen paket miktarı test sonuçları	67
4.2. Bulanık Mantık Tabanlı Yöntem Analizi ile Çıkış Parametreleri Performans Karşılaştırmaları.....	69
4.3. YSA Tabanlı Yöntem Analizi ile Performans Arttırımı.....	78

Sayfa

4.4. Farklı Ortam Değişkenleri ile Performans Karşılaştırmaları	81
4.5. Geri Besleme Kontrolü ile Kanal Verimliliğinin Arttırılması	85
4.5.1. Döngü tablolu kaba kuvvet algoritması ile performans arttırımı.....	86
4.5.2. Bulanık mantık tabanlı geri besleme kontrolü ile düğüm bazlı birim zamanda işlenen paket miktarının benzetim anında optimize edilmesi ...	87
4.6. Farklı Düğüm Sayıları, Varışlar Arası Süre ve İletim Gücü ile Önerilen Model Performans Karşılaştırmaları ve Test Sonuçları	90
5. SONUÇLAR VE ÖNERİLER	101
KAYNAKLAR.....	103
ÖZGEÇMİŞ	113

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Çekişme penceresi kısıtlama hesaplamaları	14
Çizelge 2.2. AC'ler için varsayılan EDCA parametreleri.....	14
Çizelge 2.3. Benzetim araçları	21
Çizelge 3.1. 27 konfigürasyonlu parametre evrensel küme ve test aralıkları	26
Çizelge 3.2. Bulanık kural tablosu	27
Çizelge 3.3. 125 konfigürasyonlu parametre evrensel küme ve test aralıkları	34
Çizelge 3.4. 27 farklı giriş değerleri test sonuçları	35
Çizelge 3.5. YSA regresyon analizi ve veri karşılaştırması.....	37
Çizelge 3.6. Bulanık mantık kural tablosu	52
Çizelge 4.1. Giriş ve çıkış parametreleri ortalama test sonuçları	55
Çizelge 4.2. 125 giriş ve çıkış parametreleri ortalama test sonuçları	56
Çizelge 4.3. Giriş ve çıkış üyelik ağırlıklarının farklı bölümlmelerine göre iyileştirme karşılaştırmaları.	77
Çizelge 4.4. Farklı ortam değişkenlerinin çıkış parametresine etkisi	84
Çizelge 4.5. Kablosuz yerel alan ağları performans karşılaştırma ölçütleri	91

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. IEEE 802.11e referans modeli	13
Şekil 2.2. Ağda gönderici ve alıcı ilişkisi (CSMA/CA modeli)	15
Şekil 2.3. RTS/CTS mekanizmasında çarpışma	17
Şekil 2.4. CSMA/CA çerçevelerinin genel iç yapıları	17
Şekil 2.5. RTS/CTS mekanizmasında ağ tahsis vektörü.....	19
Şekil 2.6. Editörler arası geçişler	24
Şekil 3.1. Bulanık mantık tabanlı bilgi işlemenin genel çerçevesi	25
Şekil 3.2. Bulanık mantık giriş ve çıkış parametreleri genel çerçevesi	26
Şekil 3.3. MATLAB bulanık mantık tasarım şeması.....	27
Şekil 3.4. Giriş üyelik fonksiyonları	29
Şekil 3.5. Çıkış üyelik fonksiyonları.....	30
Şekil 3.6. YSA için hedef düğümlerin yerleşimi	34
Şekil 3.7. Geliştirilmiş 4 katmanlı MLP (Multilayer Perceptron) ile YSA model mimarisi.....	36
Şekil 3.8. RTS/CTS temel erişim metotları	38
Şekil 3.9. Giriş ve çıkış parametrelerinin şematik gösterimi	38
Şekil 3.10. Riverbed Modeler model yapısı.....	40
Şekil 3.11. Riverbed Modeler proses model kodlama sınıfları.....	41
Şekil 3.12. Riverbed Modeler sınıf tanımlamaları.....	41
Şekil 3.13. Durum değişkeni (SV).....	42
Şekil 3.14. Geçici değişken (TV).....	42
Şekil 3.15. Başlık bloğu (HB).....	43
Şekil 3.16. Fonksiyonel blok (FB).....	43
Şekil 3.17. Riverbed Modeler hata ayıklama ekranı.....	44

Şekil	Sayfa
Şekil 3.18. Riverbed Modeler döngü tablosu ajan yapısı	45
Şekil 3.19. Wlan_accepted_frame_stats_update () fonksiyon yapısı	46
Şekil 3.20. Op_sim_time fonksiyon kullanımı	48
Şekil 3.21. Bulanık mantık sisteminin şematik gösterimi.....	49
Şekil 3.22. Bulanık mantık tabanlı geri besleme kontrolü şematik gösterimi	50
Şekil 3.23. Bulanık mantık tabanlı geri besleme kontrolü genel akış diyagramı	51
Şekil 3.24. Giriş üyelik fonksiyonları	53
Şekil 3.25. Çıkış üyelik fonksiyonları.....	54
Şekil 4.1. RTSED performans karşılaştırmaları	61
Şekil 4.2. PED performans karşılaştırmaları.....	62
Şekil 4.3. AB performans karşılaştırmaları.....	63
Şekil 4.4. RTSED örnek alıcı kanal kullanım durumu performans karşılaştırması.....	64
Şekil 4.5. PED örnek alıcı kanal kullanım durumu performans karşılaştırması.....	64
Şekil 4.6. PED için ortalama gönderilen veri trafiği performans karşılaştırmaları.....	65
Şekil 4.7. AB alıcı kanal durumu performans karşılaştırmaları.....	66
Şekil 4.8. AB için ortalama gönderilen veri trafiği performans karşılaştırmaları	66
Şekil 4.9. RTSED performans karşılaştırmaları	67
Şekil 4.10. PED performans karşılaştırmaları.....	68
Şekil 4.11. AB performans karşılaştırmaları.....	68
Şekil 4.12. Bulanık mantık tabanlı MATLAB test sonuçları-1	70
Şekil 4.13. Bulanık mantık tabanlı MATLAB test sonuçları-2	71
Şekil 4.14. Bulanık mantık sonuçlarının Riverbed Modeler’de uygulanması	71
Şekil 4.15. Performansın üç boyutlu genel gösterimi	72
Şekil 4.16. Giriş üyelik fonksiyonları	74

Şekil	Sayfa
Şekil 4.17. Çıkış üyelik fonksiyonları.....	75
Şekil 4.18. 5 Giriş ve 7 çıkış bölmeli parametreleri ile bulanık mantık tabanlı test sonuçları	76
Şekil 4.19. Gecikme süresi performansının üç boyutlu genel gösterimi	78
Şekil 4.20. YSA eğitim sonucu.....	78
Şekil 4.21. YSA eğitim performansı.....	79
Şekil 4.22. YSA modelinde eğitim, onaylama ve test performans karşılaştırması.....	79
Şekil 4.23. YSA sonuçları: eğitim, onaylama ve test	80
Şekil 4.24. YSA hata histogramı-30 bins.....	80
Şekil 4.25. Ortam değişkenliği testi 10 adet kablosuz sunucu ve 10 adet telekom vericisi dış ortam konumlandırması.....	82
Şekil 4.26. Ortam değişkenliği testi 10 adet kablosuz sunucu ve 10 adet telekom vericisi iç ortam konumlandırması.....	82
Şekil 4.27. Ortam değişkenliği testi 10 adet kablosuz sunucu ve 10 adet telekom vericisi iç ortam konumlandırması ve 4 adet jammer eklenmesi.....	83
Şekil 4.28. Döngü tablosu tabanlı geri besleme kontrolü performans sonuçları.....	87
Şekil 4.29. Geri besleme kontrollü bulanık mantık sonuçlarının Riverbed Modeller’de uygulanması	89
Şekil 4.30. Düğüm bazlı birim zamanda işlenen paket miktarı sonuçları üç boyutlu gösterimi.....	90
Şekil 4.31. Yeni modelde farklı düğüm sayıları ile düğüm bazlı birim zamanda işlenen paket miktarı performans karşılaştırması	92
Şekil 4.32. Yeni modelde farklı düğüm sayıları ile alınan veri trafiği performans karşılaştırması	93
Şekil 4.33. Yeni modelde farklı düğüm sayıları ile gönderilen veri trafiği performans karşılaştırması	93
Şekil 4.34. Yeni modelde farklı varışlar arası süreler ile bazlı birim zamanda işlenen paket miktarı performans karşılaştırması.....	94
Şekil 4.35. Yeni modelde farklı varışlar arası süreler ile alınan veri trafiği performans karşılaştırması	95

Şekil	Sayfa
Şekil 4.36. Yeni modelde farklı varışlar arası süreler ile gönderilen veri trafiği performans karşılaştırması	95
Şekil 4.37. Yeni modelde farklı iletim güçleri ile bazlı birim zamanda işlenen paket miktarı performans karşılaştırması	96
Şekil 4.38. Yeni modelde farklı iletim güçleri ile alınan veri trafiği performans karşılaştırması	97
Şekil 4.39. Yeni modelde farklı iletim güçleri ile gönderilen veri trafiği performans karşılaştırması	97
Şekil 4.40. Yeni modelde farklı düğüm sayıları ile kanal kullanım durumu performans karşılaştırması	98
Şekil 4.41. Yeni modelde farklı varışlar arası süreler ile kanal kullanım durumu performans karşılaştırması	99
Şekil 4.42. Yeni modelde farklı iletim güçleri ile kanal kullanım durumu performans karşılaştırması	99

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler

Açıklamalar

bit/sn	Bit/saniye
GHz	Gigahertz
kbit/sn	Kilobit/saniye
Mbp	Megabit Per Second
ms	Milisecond
ns	Nanosecond
ps	Picosecond
R²	Varyansın Mutlak Fraksiyonu
sn	Saniye
us	Microsecond

Kısaltmalar

Açıklamalar

AB	Arabellek Boyutu
AC	Access Categories
ACK	Acknowledgement
AIFSN	Arbitration Inter-Frame Space Number
ANN	Artificial Neural Networks
AP	Access Point
BDP	Bandwidth Delay Product
CAP	Controlled Access Phase
CCNA	Cisco Certified Network Associate
CCNP	Cisco Certified Network Professional
CFP	Contention Free Period
CP	Contention Period
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance

Kısaltmalar	Açıklamalar
CTS	Clear to Send
CW	Contention Window
DB	Diagnostic Block
DCF	Distributed Coordination Function
DES	Data Encryption System
DIFS	Distributed Inter Frame Space
DLMA	Deep Reinforcement Learning Multiple Access
DSSS	Direct Sequence Spread Spectrum
EDCA	Enhanced Distributed Channel Access
EDCF	Enhanced Distributed Coordination Function
FB	Functional Block
FCS	Frame Check Sequence
FHSS	Frequency Hopping Spread Spectrum
FLCMAC	Fuzzy Logic Based Cooperative Media Access Control
GTNetS	Georgia Tech Network Simulator
HB	Header Block
HCCA	HCF Controlled Channel Access
HCF	Hybrid Coordination Function
IEEE	Institute of Electrical and Electronics Engineers
LAN	Local Area Network
LM	Levenberg Marquardt
MAC	Media Access Control
MAN	Metropolitan Area Network
MATLAB	Matrix Laboratory
MEP	Mean Percentage Error
MLP	Multilayer Perceptron
NAM	Network Animator
NAV	Network Allocation Vector
NS	Network Simulator
OPNET	Optimized Network Engineering Tool
OSI	Open Systems Interconnection
PCF	Point Coordination Function

Kısaltmalar**Açıklamalar**

PDF	Portable Document Format
PED	Parçalama Eşik Değeri
PHYMAC	Physical Layer Media Access Control
QoS	Quality of Service
RMS	Root Mean Square
RMSE	Root Mean Square Error
RTSED	RTS Eşik Değeri
RTS	Request to Send
SIFS	Short Inter Frame Space
SV	State Variable
TB	Terminates Block
TC	Traffic Categories
TCP	Transport Control Protocol
TV	Temporary Variable
VoIP	Voice Over Internet Protocol
WAN	Wide Area Network
WLAN	Wireless Local Area Network
WMNs	Wireless Mesh Networks
TXOP	Transmission Oppurtunity
YSA	Yapay Sinir Ağları.

1. GİRİŞ

Kablosuz yerel alan ağları (Wireless Local Area Network-WLAN); birbirlerine yakın mesafede bulunan kablosuz cihazlar arasında veya kablolu cihazlar ile kablosuz cihazlar arasında iletişimi sağlamak amacı ile kullanılan ağlardır. Verimli, üretken ve taşınabilir olan kablosuz ağlar günümüzde kablolu ağların kullanıldığı bütün ortamlarda kullanılmaktadır. WLAN'larda hizmet kalitesini (Quality of Service-QoS) arttırmak için birçok çalışma yapılmaktadır. WLAN'larda verimli QoS için genellikle 802.11e MAC (Media Access Control) protokolü önerilmektedir. WLAN'larda yapılan çalışmalarda QoS'ı arttırmak amacı ile birçok parametre incelenmiştir [1-2]. IEEE (Institute of Electrical and Electronics Engineers) 802.11e'nin MAC katmanında kullanılan dağıtık koordinasyon fonksiyonu (Distributed Coordination Function-DCF); fonksiyon yapısındaki belirli parametrelerin performansı için önemli işleve sahiptir [3-4].

WLAN, IEEE tarafından yayınlanan bir dizi standardıdır ve günümüzde çok sık kullanılan uygulama alanlarına sahiptir. İlk olarak IEEE 802 LAN/MAN (Local Area Network/Metropolitan Area Network) standart komitesi 1997 yılında IEEE 802.11 standartını yayınlamıştır. Bu standarda göre Doğrudan Sıralı Yayılma Spektrumu (Direct Sequence Spread Spectrum-DSSS) veya Frekans Atlamalı Yayılma Spektrumu (Frequency Hopping Spread Spectrum-FHSS) teknikleri kullanılarak 2 Mbp'e kadar veri akışı sağlanmaktadır. Daha sonra WLAN için 802.11x adlarıyla günümüze kadar birçok standart yayınlanmıştır. Dünyanın birçok yerinde kullanılan 802.11b, 11 Mbps veri iletişim hızına sahiptir ve 2.4 GHz bandında çalışmaktadır. Günümüzde ise aynı frekans bandında yer alan ve veri iletişimini 54 Mbps'e kadar çıkaran 802.11g standardı cihazlar genel olarak kullanılmaktadır [5-6].

LAN, hastane, okul, bina gibi sınırlı alanlarda kurulan ve birçok kişisel bilgisayar, yazıcı, tarayıcı, gibi cihazların yer aldığı ağlardır. Sunucu üzerindeki uygulamalar (ulaşım, elektronik posta hizmeti, dosya paylaşımı gibi) için daha önceden IEEE 802.11x (Amerika'da) ve HiperLAN (Avrupa'da) teknolojileri kullanılmaktaydı [7]. Temel olarak HiperLAN2 ve 802.11a, frekans bandı, frekans seçimi ve ortama erişim metodu dışında birbirine yakın özellikleri taşımaktadır. HiperLAN2 ve 802.11a, frekans bandı, frekans seçimi ve ortama erişim metodu dışında birbirine yakın özellikleri taşımaktadır. WLAN'lar, RTSED, PED ve AB gibi birçok parametreye sahiptirler. Gizli düğüm

problemlerinde uygun RTSED'ler seçilmediğinde ağda gizli düğüm problemi ortaya çıkmakta ve sistem boşa düşmektedir [8-10]. PED ile büyük veriler parçalara bölünerek iletilmektedir. AB ile dinamik bir bellek sağlanmaktadır. Li ve arkadaşları [11] yapmış oldukları çalışmada AB değeri ile Bant Genişliği Gecikme (Bandwidth Delay Product–BDP) kuralında uygun değerleri ayarlayarak gecikmeyi de önlemişlerdir. Bütün bu giriş değerleri test edilirken gecikme süresi, yükleme, birim zamanda gönderilen paket miktarı, alınan ve gönderilen veri trafiği gibi çıkış parametre değerleri ile konfigürasyonlar oluşturularak QoS ölçümü de sağlanılmaktadır.

DCF kontrol fonksiyonu Çarpışma Önleme ile Taşıyıcı Algılama Çoklu Erişim (Carrier Sense Multiple Access with Collision Avoidance-CSMA/CA) tabanlı olması nedeniyle kablosuz yerel alan ağlarında çarpışmaları azaltmıştır. CSMA/CA'da bulunan sanal RTS (Request to Send) ve CTS (Clear to Send) paketleri ağda belirli şekilde paket çarpışmalarını azaltmasına rağmen belirli bir enerji ve zaman kaybı ortaya çıkartmaktadır [12-14]. Bu nedenle CSMA/CA protokolünün aktif edip edilmemesini sağlayan RTS Eşik Değeri (RTSED), Parçalama Eşik Değeri (PED) ve Arabellek Boyutu (AB) verileri uygulamalarda optimize edilmektedir. Kullanılan parametrelerden RTSED değeri CSMA/CA protokolünün başlatılmasına karar verici gibi davranırken, PED ise protokol başladıktan sonra verilerin büyüklüğünü kontrol etmekte ve bölünüp bölünmeyeceğine karar vermektedir [15-16]. CSMA/CA protokolünde önemli bir yere sahip olan diğer bir parametre ise AB'dir. Maliyet açısından önemli olan bu parametre verinin AB'de ne kadar süre beklemesi gerektiği bilgisini saklamaktadır. AB'nin küçük olması ağda kuyruklanmaya ve dolayısıyla paket kayıplarına neden olmaktadır [17-19]. Bu nedenle paket kaybını engellemek için ideal AB değerinin seçilmesi de son derece önemlidir [20-23].

Sanal RTS paketlerinin optimizasyonu için Liu ve arkadaşları [24] yapmış oldukları çalışmada uyarlanabilir RTSED algoritması ile RTSED değerini olasılık hesaplamalarıyla güncellemiştir. Çalışmada kanal verimliliği kullanılan yöntem ve standart yöntemle karşılaştırılmış ve performans iyileştirmeleri analiz edilmiştir. Abusubaih [25] yapmış olduğu çalışmada WLAN'larda çarpışmaları azaltmak için RTS/CTS'de zaman bölümlmesi yaparak kombine bir yaklaşım ortaya çıkartmıştır. Yaklaşımında tüm kullanıcılar tarafından deneyimlenen performans iyileştirmesi gerçekleştirilmiştir. Yapılan başka bir çalışmada, Ahsan ve arkadaşları [26] RTSED'nin sabit olması durumunda

karşılaşılan problemleri göz önünde bulundurarak dinamik ve uyarlanabilir bir model sunmuştur. NS-2 benzetim aracı kullanılarak yapılan test çalışmalarında sabit CSMA/CA kullanımı ile dinamik RTS/CTS mekanizması kullanımı performans karşılaştırması yapılmıştır. Test sonuçları farklı düğüm sayıları ile analiz edilmiş ve birim zamanda işlenen paket miktarının dinamik yöntemle arttırıldığı gözlemlenmiştir. Yin ve arkadaşları [27] yapmış oldukları çalışmada IEEE 802.11 protokolünde sabit RTS/CTS ile optimize kullanılan RTSED mekanizmasının performanslarını karşılaştırmıştır. Gerçekleştirilen çalışmada, IEEE 802.11 standart ayarlarında RTS/CTS'nin ayarlanabilir olarak kullanılmadığı ve bu nedenle paket kayıplarının olduğu ortaya çıkartılmıştır. NS-3 kullanılarak dinamik RTSED'nin farklı düğüm sayılarıyla farklı oranlarda performans artışları sağladığı gösterilmiştir. Bütün bu çalışmalar dinamik RTSED'nin kullanılması gerekliliğini ortaya çıkarmaktadır.

Gecikme süresi (Delay), paketlerin uçtan uca gecikmesini saniye (sn) olarak temsil etmektedir. Bu paketler, ağdaki tüm WLAN düğümlerinin MAC'ları tarafından alınan ve daha yüksek katmana iletilen paketlerdir. Bu gecikme uygulamalarda MAC katmanında verilerin erişim noktası (Access Point-AP) üzerinden parçalanarak bağımsız olarak iletilmesi aşamasındaki gecikmeyi de ifade etmektedir. Yükleme süresi (Load), ağın tüm WLAN düğümlerindeki üst katmanlarına aktarılan verinin saniyedeki toplam bit sayısını (bit/sn) temsil etmektedir. Yükleme süresi, WLAN katmanları için veri iletiminde harcanan toplam süredir. Birim zamanda işlenen paket miktarı (Throughput), WLAN katmanları tarafından geri dönüş olarak iletilen ve ağın tüm WLAN düğümlerindeki üst katmanlarına gönderilen verilerinin birim zamandaki toplam bit sayısını (bit/sn) ifade etmektedir.

Veri trafiği, WLAN'larda genel veri akışını ifade etmektedir. Ağ bandında düğümler arasında veri trafik akışı sağlanırken iki düğüm arasında hedef düğümlere aktarımı yapılan trafik, gönderilen veri trafiği olarak adlandırılmaktadır. Hedef düğümden gelen veri trafiği ise alınan veri trafiği çıkış parametresi olarak kabul edilmektedir. Gönderilen ve alınan veri trafiği ne kadar arttırılırsa performans da o kadar artmaktadır. Genel olarak WLAN'larda paket kayıpları ve paket iletim hızları gönderilen ve alınan veri trafikleri ile doğrudan ilişkilidir [28].

Kanal kullanım verimliliği bant genişliğinin kullanım etkinliği olarak bilinmektedir. Örneğin, çıkışı 100 Mbit/s ethernet bağlantısında veri akış hızımız 85 Mbit/s ise kanal

verimliliği %85'dir. Kablosuz ağlardaki hizmet kalitesinin artırılması için kanal verimliliğinin mümkün olduğunca yüksek değerlere çıkarılması gerekmektedir [29-31]. WLAN'larda kanal kullanım oranını arttırmak için birçok giriş parametresi kullanılmış ve farklı algoritmalarla iyileştirmeler sağlanmıştır. Literatürde yapılan çalışmalarda giriş ve çıkış parametreleri ile farklı algoritmalar ve yöntemler kullanılarak kablosuz ağların performansı geliştirilmiş ve hizmet kalitesi artırılmıştır.

Kanal kullanım verimliliği kanal kullanım durumu olarak da bilinmektedir. Bu parametre bant genişliğinin kullanım etkinliği olarak bilinmektedir. Örneğin Purohit ve Joshi'nin [32] yapmış oldukları çalışmada kablosuz iletişimde serpiştirme tekniği ile konuşma iletişimini %36'dan % 83,32'ye yükseltmişlerdir. Başka bir örnekte ise Lin ve arkadaşları [33] yapmış oldukları çalışmada kullanıcıların kanal kullanım durumuna göre kanalın farklı kullanıcılara senkronizasyonu tam serbestlik dereceli Fiziksel Katmanlı Ortam Erişim Kontrolü (Physical Layer Media Access Control-PHYMAC) tasarımı ile sağlayarak kanal kullanım durumunu arttırmış ve IEEE 802.11ac'lerde %75'e varan kanal kullanımı sağlamışlardır. Yapılan başka bir çalışmada Saketh ve arkadaşları [34] Markov zincirine dayalı model ile hem kullanıcılar hem de veriler önceliklendirilmiştir ve birincil ve ikincil veri iletim kanalı oluşturulmuştur. Önerilen yöntemde, dinamik bant genişliği erişimi ile ikincil kullanıcıların kanalın boşta kalma süresini azaltarak kanal verimini iyileştirdiği görülmüştür. Başka bir çalışmada da aynı şekilde dinamik arabellek boyutu ve grafik teorisi ile algoritmada en kötü durum zaman karmaşıklığı kullanılıp kanal kullanım oranı arttırılmıştır [35]. Benzeri çalışmalar çarpışma sayısını azaltarak ve farklı kural tabanlı iletişim methotları uygulanarak da gerçekleştirilmiştir [36-37]. Yapılan başka bir çalışma da ise Shih ve arkadaşları [38] kanal erişimini sağlamak için Rhythm metodunu önermişlerdir. Rhythm metoduna göre MAC protokolünde WiFi senkronize edilmektedir ve böylece kablosuz ağlarda kanal kullanımı arttırılmaktadır.

IEEE 802.11'de üç temel kontrol fonksiyonu bulunmaktadır. Bunlar DCF, noktasal koordinasyon fonksiyonu (Point Coordination Function-PCF) ve gelişmiş dağıtık koordinasyon fonksiyonu (Enhanced Distributed Coordination Function-EDCF)'dur. Bu kontrol fonksiyonları kanal erişim mekanizlarını tanımlamaktadır [39-42]. DCF'de çekişme tabanlı çerçevede rastgele geri besleme kullanıldığı için DCF altında kanal erişimi kontrolü ve tahminleme zordur. DCF'de kullanılan CSMA/CA protokolü güvenilir veri aktarımını sağlamakta ve paket kayıplarını azaltmaktadır [43].

Tez çalışmasında kullanılacak olan Riverbed Modeler, genel olarak bilişim sektöründe ticari amaçla kullanılmakla beraber akademik çalışmalarda ve üniversitelerde eğitim amaçlı olarak laboratuvar ortamlarında da kullanılmaktadır [44]. Zubairi ve Zuber [45] yaptıkları çalışmada Riverbed Modeler benzetim aracıyla üniversite laboratuvar ağını modellemiş ve ağda performans analizi yapmışlardır. Ağdaki performans analizinde; trafik akışları, topolojiler ve protokollerle farklı istatistiksel bilgiler toplanmış ve analizleri yapılmıştır. Riverbed Modeler’de ağırlıklı olarak LAN ve WAN (Wide Area Network)’ların standartlarında modeller tasarlanmıştır. Örneğin Lawal ve arkadaşları [46] merkezi ve dağıtık ağ modellerinin hizmet kalitesini sabit ağlarda Riverbed Modeler kullanarak ölçmüşlerdir. Konfigürasyonlarda gecikme ve birim zamanda işlenen paket miktarı çıkış parametreleri ile performans analizi gerçekleştirilmiştir. Zhao ve Xie [47] Kablosuz Örgü Ağlar (Wireless Mesh Networks-WMNs) ile ağlarda farklı protokollerle performans analizi yapmışlardır. Erdem ve Tuğral [48] yaptıkları çalışmada Riverbed Modeler benzetim aracı kullanarak farklı kablosuz ağ standartlarının sınıflandırmasını yaparak IEEE’nin geliştirmiş olduğu 802.11a/b/g ve HiperLAN/2 kablosuz ağ standartlarının ağ performans analizlerini yapmışlardır. Çakır ve Kaptan [49] yaptıkları çalışmada İnternet Protokolü üzerinden Ses (Voice Over Internet Protocol-VoIP) aktarımı mimarisine yönelik gerçekleşen tehdit saldırılarını incelemiş, bu saldırılara karşı gerçekleştirilen önlemleri Riverbed Modeler üzerinden test etmiş ve benzetim araçları ile güvenlik üzerine de çalışmaların yapılacağını göstermişlerdir.

Riverbed Modeler nesne tabanlı yazılım ve geliştirmeye açık kaynak kodları nedeniyle uygulamalarda tercih edilmekte ve editörler yardımı ile istatistiksel verilerin aynı grafik arayüzü üzerinde gösterilerek performans karşılaştırmasına da olanak sağlamaktadır [50]. Mekkanen ve arkadaşları [51] yapmış oldukları çalışmada akıllı elektronik cihazların verimliliklerini farklı konfigürasyonlar üzerinden karşılaştırmışlardır. Ayrıca Riverbed Modeler Taşıma Kontrol Protokolü (Transport Control Protocol-TCP), iletişim protokolü ve tıkanıklık testinde de kullanılmaktadır.

Kablosuz ağlarda ve farklı mühendislik çözümlerinde hizmet kalitesini arttırmak için bulanık mantık tabanlı birçok çalışma gerçekleştirilmiştir [52-54]. Örneğin Zhaoxiang ve arkadaşları [55] tarafından 802.11 protokolü için EDCA (Enhanced Distributed Channel Access)’da yer alan veri iletimi önceliği (priority) için bulanık mantık kullanan bir yöntem geliştirilmiştir. İlgili yöntem Bulanık Mantık Tabanlı İşbirlikli Ortam Erişim Kontrolü

(Fuzzy Logic Based Cooperative Media Access Control-FLCMAC) olarak adlandırılmıştır. Önerilen yöntemde tekli ve çok atlamalı kablosuz iletimde gecikme süresi ve birim zamanda işlenen paket miktarının daha verimli hale geldiği ortaya konulmuştur. Kablosuz ağlarda yapılan bir diğer bulanık mantık çalışmasında, WLAN ağlarda hassas trafik için VoIP protokolünün kullanıcı sayısını arttırmak amacıyla veri iletiminde ideal paket boyutu seçimi yapılmıştır. Sonuçlar paket boyutu seçimi için en ideal değerlerin bulanık mantıkla ortaya çıktığını göstermiştir. Ayrıca bu çalışmada minimum gecikme ve kabul edilebilir paket kaybı ile iyileştirme gerçekleştirilmiş ve VoIP kullanıcı sayısı arttırılmıştır [56]. IEEE 802.11e ağlar için yapılan başka bir çalışmada Collotta [57], denge modeli üzerine kurulan bulanık mantık tabanlı bir yöntem ile AP güçlendirmesi sağlamış ve hizmet kalitesi arttırmıştır. Yapılan çalışmada bulanık mantıkla AP'lerde yük dengeleme modeliyle hizmet kalitesinin yaklaşık %10 oranında arttığı ortaya konulmuştur. Kablosuz yerel alan ağlarında bulanık mantık tabanlı yapılan bu yeni çalışmalar ve sunulan yeni protokoller hizmet kalitesini arttırmıştır.

YSA (Yapay Sinir Ağları), insan beynini taklit eden ve hafızada yer alan bilgileri hatırlatma veya genelleme yapma yolları ile yeni bilgiler türetebilmek amacı ile kullanan mantıksal yazılım algoritmalarıdır. İlk sinir ağ modeli McCulloch ve Walter Pitts tarafından 1943 yılında mantıksal bir hesaplama ile ortaya çıkartılmıştır [58]. YSA, güçlü öğrenme ve uygulanabilirliğinin yüksek olması nedeniyle kablosuz ağlarda ve iletişim tekniklerinde sıkça kullanılmıştır [59-63]. Temel YSA yapısında girişler X , ağırlıklar W ve toplama fonksiyonu NET olarak kullanılmaktadır. Aktivasyon fonksiyonu f , aktivasyon fonksiyonu eşik değeri b ve çıkış değeri o olarak ifade edilmiştir. Aktivasyon fonksiyonu bütün ağırlıklar ile girişlerin çarpımlarının toplamı olarak ifade edilmektedir [64].

YSA modellerinde veri setini öğrenme ve öğrenilen süreçte ortaya çıkan hataların yayılması genel işlev açısından önemlidir. YSA modelleri elde bulunan veri setlerini kara kutu içerisinde nöronlar arasında çeşitli varyasyonlar uygulayarak bağlantı ağırlıklandırmalarını güncelleyerek öğrenirler. Öğrenme süresinde gerçek performans sonuçlarıyla, eğitim süresince farklı girdilerden elde edilen sonuçların karşılaştırılması ve geriye dönük ağırlıkların güncellenmesi ile sonuçlar elde edilmektedir. Gerçek performans ve vektörler tarafından YSA ile elde edilen çıktılar arasındaki hata değeri daha önceden limit koyularak test edilir ve YSA'nın öğrenme kabiliyetleri ölçülür. Optimizasyon

probleminde YSA'daki çalışma parametreleri ile gizli katmanda yer alan nöron sayısı kullanılmaktadır [65-67]. Yapılan başka bir çalışmada Yiğit ve Ersoy [68] kablosuz ağlarda AP'lerin ideal konumlarını belirlemek amacıyla YSA kullanmış ve sinyal gücünü arttırmıştır. Sinyal gücünü arttırmak amacıyla YSA'da giriş değişkeni olarak frekans bandı, AP sayısı ve alıcı eşik değeri giriş parametreleri kullanılmıştır. Sinyal boyutu ise çıkış parametresi olarak kullanılmıştır.

Geri besleme (Feedback) kontrolü neden-sonuç ilişkisi içerisinde döngü ile bir basamağın önceki basamakları etkilemesi ile ilerleyen sistem mekanizmasıdır. İlk olarak mekanik sistemlerde kullanılan geri besleme kontrolü makine öğrenmesinde de sıkça kullanılmaktadır. Bu yapıda ajan (agent) adı verilen ve devamlı gerçek zamanlı öğrenime dayanan bir çalışma prensibi bulunmaktadır. Amaç problemlerin çözümünde en ideal çözüm yollarını gerçek zamanlı sistemde tespit edebilmektir. Geri beslemede pekiştirmeli öğrenme yapısına benzer bir yapı bulunmaktadır. Bu yapıda çevre (environment) veri setlerinin toplandığı ve ajandan gelen sonuca göre güncellendiği ortamdır. Genel olarak geri besleme mekanizması gerçek zamanlı çalışmaktadır.

Geri besleme kontrollü çalışmalarda dinamik programlama ve Monte Carlo gibi farklı kategoriler bulunmaktadır. Geri besleme birçok algoritmanın temel yapısının oluşturmaktadır. Bu algoritmalarından bir tanesi pekiştirmeli öğrenmeye dayalı metodlardan Q-öğrenme algoritmasıdır. Q algoritmasında modele ihtiyaç bulunmamakta ve ajan çevreden bağımsız çalışmaktadır. Burada ajan deneme-yanılma yöntemi ile en ideal yöntemi bulmaya çalışmaktadır. Algoritma yapısında bir Q öğrenme tablosu vardır ve her seferinde tablo güncellenmektedir. Her durum s ve her eylem a olarak düşünüldüğünde bütün durumların araştırılıp en ideal değerlerin bulunması gerekmektedir. Bu yapıda durumlar ve eylemler $s+1$ ve $a+1$ olarak ilerleme kaydetmekte ve her seferinde daha ideal değerler bulunduğu sonuca güncellenmektedir [69-70].

Geri besleme kontrolü ile kablolu ve kablosuz ağlarda birçok problemin çözümü gerçekleştirilmiştir. Örneğin yapılan bir çalışmada Derin Pekiştirmeli Öğrenme Çoklu Erişim (Deep Reinforcement Learning Multiple Access-DLMA) adı verilen yöntemde, aynı kablosuz ortamda ve spesifik olarak farklı MAC protokollerini senkronize olarak çalıştırmak için geri besleme kontrollü pekiştirmeli öğrenme kullanılmıştır. DLMA'nın hedefi ideal bir kanal belirlemektir. DLMA'da amaç toplam verimi arttırmaktır ve bu

nedenle farklı protokollerin ne zaman çalışacağı deneme-yanılma yöntemi ile belirlenmiştir [71].

Yapılan başka bir çalışmada ise Bhattacharyya ve arkadaşları [72], kablosuz ağlarda video uygulamalarında servis kalitesini arttırmak için önceliklendirme yapılması adına geri besleme kontrolü kullanmıştır. Geliştirilen yeni yaklaşımda kullanıcılara kablosuz bağlantılarda video uygulamalarının daha hızlı yüklenmesi için mevcut algoritmalar yerine pekiştirmeli öğrenme yöntemi kullanılıp daha yüksek servis kalitesi sunulmuştur.

Başka bir çalışmada ise Oliveira ve arkadaşları [73], bitki köklerini stabilize etmek amacıyla kablosuz ağlar üzerinden bir kontrol sistemi sağlamışlardır. Mekanizmada kablosuz kanal üzerinden stokastik maksimum iletim aralıkları kullanılmıştır. Çalışmada stabilize edici durum, geri besleme denetleyicisi ve stokastik maksimum iletim aralık parametreleri aynı anda sentezlenmiştir.

Geri besleme kontrolü ve denetleyicileri ile pekiştirmeli öğrenmeye dayalı yukarıdaki problemler ve benzeri birçok problemde amaç gerçek zamanlı ve döngüsel olarak çözüm geliştirmektir. Yapay sinir ağlarında, trafikte yakıt harcamalarını azaltmada, trafiği azaltmada [74-75] ve sanal ağlarda [76] da geri beslemeye dayalı pekiştirmeli öğrenme ile birçok çözüm yöntemi geliştirilmiştir.

Riverbed Modeler’de kullanılan *Op_sim_time* fonksiyonu açık kaynak kodlu değildir ve benzetimin koşum süresini belirlemektedir. *Op_sim_time* zamanlama fonksiyonu olarak tanımlanmaktadır. Birim olarak *double* değişkeni ile implemente edilmektedir. Benzetimin koşum süresi ekranında girilen süre toplam süreyi ifade etmektedir. Koşum süresinin altında belli bir müddet kısıtlama getirmek için veya belli şartlar altında belirli süreden sonra benzetimi koşturmak için *Op_sim_time* kullanmak gerekmektedir [77]. Örneğin giriş değerlerinin ne kadar süre tutulacağı veya durum (state) transferlerinin beklemesi işlemleri *Op_sim_time* fonksiyonuyla yapılabilmektedir.

Kablosuz ağlarda yapılan çalışmalarda, döngüsel algoritmalarda genelde geri beslemeli yöntemler tercih edilmiştir. Örneğin kablosuz algılayıcı ağlarda güvenliği arttırmak için yapılan bir çalışmada şifreleme amaçlı geri beslemeli bir yöntem kullanılmıştır [78]. Ağ tabanlı kontrol sistemlerinde geri beslemeli çalışmalarda genel olarak arka planda bir

döngü tablosu koşulmaktadır. Döngü tablosunda verilerin limitleri doğrultusunda belirli aralıklarla döngü içerisindeki değerler test edilmektedir. VigilNet gibi büyük ölçekli gözetim ağlarında sistemin ömrü hiyerarşik bir yapıda çalışma evreleri ile döngüsel olarak kontrol edilmektedir [79]. Kablosuz ağlarda kaba kuvvet algoritması kullanılarak kanal kullanım oranının artırılması da yine döngü tablosu ile gerçekleştirilmiştir [80].

Kaba kuvvet algoritması, doğrusal basit bir arama ile ideal sonuçları bulan ve köklü geçmişi bulunan bir algoritmadır. Kaba kuvvet algoritması maliyet açısından hesaplı olduğu için donanım ağırlıklı sistemlerde tercih edilmeye devam etmektedir. Bu algoritma ile basitçe birçok problem çözülebilmektedir. Kaba kuvvet algoritması birçok uygulamada kötü amaçlı siber saldırılar ve şifre kırmalar için de kullanılmıştır. Alt yapı ve güvenlik çözümlü çalışmalarda genel olarak kaba kuvvet algoritması test amaçlı tersine çözüm için kullanılmıştır. Örneğin yapılan bir çalışmada Şanlı ve arkadaşları [81], akıllı kartlar için Veri Şifreleme Sistemi (Data Encryption System-DES) ile birlikte kaba kuvvet algoritması kullanarak üst seviye güvenlik sağlamıştır.

Yapılan çalışmalardan da anlaşılacağı üzere WLAN'larda giriş değişkenleri ile farklı algoritmalar ve yöntemler kullanılarak bazen maliyetten bazen de tıkanıklıktan kaynaklanan problemler azaltılmış ve performans artırılmıştır.

WLAN'larda CSMA/CA protokolünde giriş ve çıkış parametreleri değerlendirildiğinde henüz çözümü olmayan birçok problem bulunmaktadır. Örneğin veri iletiminde verinin belli değer üzerinde olması durumunda paket kayıpları yaşanmakta ve verinin iletiminde gecikmeler meydana gelmektedir. CSMA/CA modelinde sistemin devamlı boş (idle) kalması nedeniyle arabellek boyutunun dinamik olması ve arabellek boyutunun sistem tabanlı dinamik olarak seçilmesi gerekliliği bulunmaktadır. Ayrıca alıcı ve verici için kanal verimliliği, alınan ve gönderilen veri trafiği gibi performans çıktılarında iyileştirme yapılma ihtiyacı bulunmaktadır. RTS/CTS, PED ve AB'nin optimize edilmesi için farklı algoritma ve yöntemlere ihtiyacı vardır. MAC protokolünde DCF kontrol fonksiyonunda CSMA/CA için iç mekanizmada bir takım çalışmalar gerçekleşmesine rağmen mekanizmanın aktif olmasını sağlayan gerekli koşullar detaylı olarak irdelenmemiştir. Bütün bu çalışmalar gerçekleştirilirken daha verimli ve faydalı modeller için benzetim anında QoS'in artırılmasına ihtiyaç vardır.

Tez çalışmasında WLAN IEEE 802.11e CSMA/CA protokolünde RTS/CTS mekanizmasındaki eşik değerler ve arabellek boyutunu benzetim anında ideal hale getirmek için yeni algoritmalar ve yöntemler geliştirilmiştir. Standart çalışmalardan farklı olarak dinamik RTSED ile birlikte dinamik PED ve AB değerleri birlikte değerlendirilmiştir. Yeni yöntemde RTSED, PED ve AB'nin belirlenmesi aşamasında bulanık mantık, YSA ve geri besleme kontrolü kullanılmıştır. Bu amaçla WLAN'larda CSMA/CA modelinde birim zamanda işlenen paket miktarı, yükleme süresi, gecikme süresi, alıcı kanal kullanım durumu, verici kanal kullanım durumu, alınan veri trafiği ve gönderilen veri trafiği çıkış parametreleri ile performans iyileştirilmesi yapılmıştır. Gerçekleştirilen çalışmalardan bulanık mantık tabanlı yöntem ile giriş değerleri optimize edilerek birim zamanda işlenen paket miktarı ve yükleme süresi arttırılmış, gecikme süresi ise azaltılmıştır. Giriş değişkenlerine YSA uygulanarak alıcı ve verici için kanal kullanım durumu, alınan ve gönderilen veri trafikleri tahminlemesi yapılmıştır. Gerçekleştirilen çalışmalarda ayrıca dış faktörlerin WLAN'lara etkisini incelemek amacıyla kablosuz sunucu, telekom verici ve jammer (sinyal bozucu) cihazları eklenmiştir. Eklenen cihazların alıcı kanal kullanım durumu, verici kanal kullanım durumu, alınan ve gönderilen veri trafiklerine olan etkileri incelenmiştir. Benzetim anında Riverbed Modeler üzerindeki ajan yapısı ile bulanık mantık tabanlı geri besleme kontrolü uygulanmış ve düğüm bazlı birim zamanda işlenen paket miktarı arttırılmıştır. RTSED, PED ve AB giriş değerleri proses katmanında prosesin içerisinde Wlan_mac.pr.m kod bloğunda ajan yapısı ile dinamik olarak güncellenmiştir. Ayrıca uygulanan yöntemde düğüm sayısı, varışlar arası süre ve iletim gücü dış parametreleri kullanılarak benzetim anında geri besleme kontrollü modelin farklı konfigürasyonlarında kanal verimliliğini, alınan ve gönderilen veri trafiklerini arttırdığı ortaya konulmuştur. Bütün bunlara ek olarak çalışma zamanı *Op_sim_time* fonksiyonu ile kontrol edilmiş ve gelecek çalışmalar için uygun bir model aracı oluşturulmuştur.

Tezin devam eden bölümlerinde WLAN'larda CSMA/CA, giriş/çıkış değişkenleri ve benzetim ortamı ile ilgili genel kavramlar bölüm 2'de anlatılmıştır. Tezin uygulamasında kullanılan yöntem ve araçlarda ayrıca 2. bölümde sunulmuştur. Bölüm 3'te tez çalışmasında gerçekleştirilen model yapısı ile uygulanan algoritmalar ve yöntemler sunulmuştur. Bu bölümde bulanık mantık, YSA ve geri besleme kontrollü modeller giriş değişkenlerine uygulanmış ve çıkış değişkenleri analiz edilmiştir. Çalışma sonucunda uygulanan modelin ve algoritmaların sonuçları bölüm 4'te verilmiştir. Bu bölümde klasik

yöntem test sonuçları ile uygulanan model ve koşulan algoritma sonuçları karşılaştırılarak performans arttırılışları analiz edilmiştir. Çalışmadan çıkan sonuçlar ve öneriler 5. bölümde sunulmuştur.

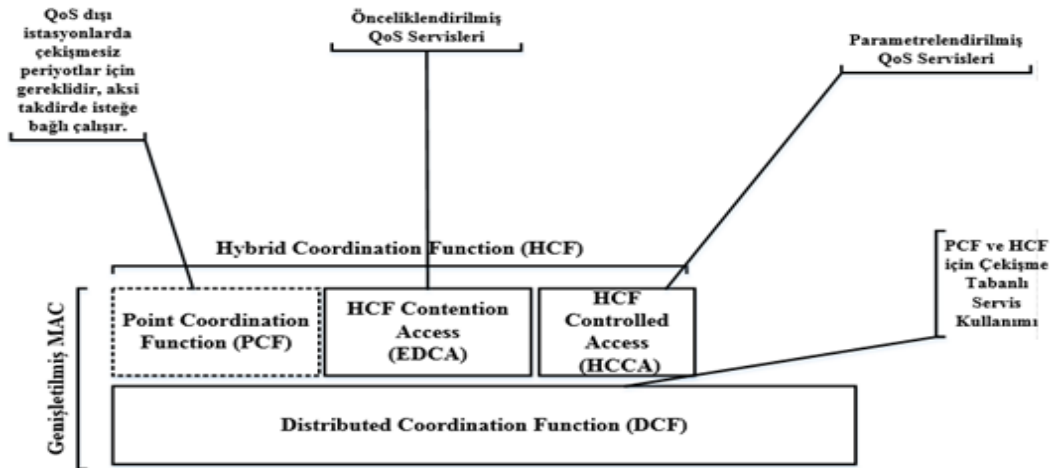
2. WLAN'LARDA CSMA/CA, GİRİŞ/ÇIKIŞ DEĞİŞKENLERİ VE BENZETİM ORTAMI

Bu bölümde WLAN'larda kullanılan CSMA/CA protokolü, RTSED, PED ve AB giriş değişkenlikleri incelenmiştir. Ayrıca benzetim ortamı ve özellikleri, farklı benzetim araç karşılaştırmaları ve benzetim aracı hiyerarşik yapısı sunulmuştur.

2.1. Kablosuz Yerel Alan Ağlarında 802.11e MAC Protokolünde CSMA/CA Modeli

Kablosuz yerel alan ağlarında son zamanlarda kablosuz ses ve video konferans uygulamaları hızla artmaktadır. Bu uygulamaların kullanımının hızla yayılması, performansında arttırılmasını zorunlu hale gelmiştir. WLAN'larda performansı arttırmak için QoS desteği, en önemli kilit işleve sahiptir [82-85]. QoS desteği için IEEE 802.11e'de MAC katmanında DCF kontrol fonksiyonu kullanılmaktadır. DCF'de en önemli performans kriterleri arasında gecikme süresi, yükleme ve birim zamanda işlenen paket miktarı bulunmaktadır. Bu performans kriterlerini ölçmek amacıyla performansı etkileyen RTSED, PED ve AB giriş değişkenleri farklı değerler ile test edilmektedir.

Şekil 2.1'de görülen IEEE 802.11e referans modeli, MAC katmanında yapılan değişiklikler yoluyla WLAN uygulamaları için QoS'i arttırmayı amaçlamıştır. Bu standart, ses ve multimedya gibi gecikmeye duyarlı uygulamalar için kritik öneme sahiptir. Orjinal 802.11 MAC protokolü, kablosuz istasyonlar için DCF ve PCF modları ile tasarlanmıştır [86].



Şekil 2.1. IEEE 802.11e referans modeli

802.11e referans modelinde ortam erişim denetiminde değişiklik yapılarak ses ve video akışları için hizmet kalitesi iyileştirmesi gerçekleştirilmiştir. 802.11e standardı, DCF ve PCF fonksiyonlarını hibrit kordinasyon fonksiyonu (Hybrid Coordination Function-HCF) ile geliştirmiştir. HCF’de EDCA ve HCF kontrollü kanal erişimi (HCF Controlled Channel Access-HCCA) olmak üzere iki kanal yöntemi bulunmaktadır [87]. Hem EDCA hem de HCCA, trafik kategorileri (Traffic Categories-TC) tanımlamaktadır. Örneğin, e-postalar düşük öncelikli bir sınıfa atanabilmekte ve kablosuz LAN üzerinden ses, yüksek öncelikli bir sınıfa atanabilmektedir.

EDCA’da yüksek öncelikli trafiğin gönderilme şansı düşük öncelikli trafiğe göre daha yüksektir. Yüksek öncelikli trafiğe sahip bir istasyon, paketini göndermeden önce ortalama olarak düşük öncelikli trafiğe sahip bir istasyondan daha az beklemektedir. EDCA’daki öncelik seviyeleri erişim kategorileri (Access Categories-AC) olarak tanımlanmaktadır. Çekişme penceresi (Contention Window-CW), her erişim kategorisinde beklenen trafiğe göre ayarlanabilir ve daha yoğun trafiğe sahip kategoriler için daha geniş bir pencere gerektirir. CW_{min} ve CW_{max} değerleri, sırasıyla 802.11e tarafından desteklenen her fiziksel katman için Çizelge 2.1’deki gibi tanımlanmaktadır.

Çizelge 2.1. Çekişme penceresi kısıtlama hesaplamaları

AC	CW_{min}	CW_{max}
Arka Plan	CW_{min}	aCW_{max}
En İyi Performans	CW_{min}	aCW_{max}
Video	$(CW_{min}+1)/2-1$	aCW_{max}
Ses	$(CW_{min}+1)/4-1$	$(aCW_{max}+1)/2-1$

AC’ler için varsayılan EDCA parametreleri Çizelge 2.2’deki gibidir.

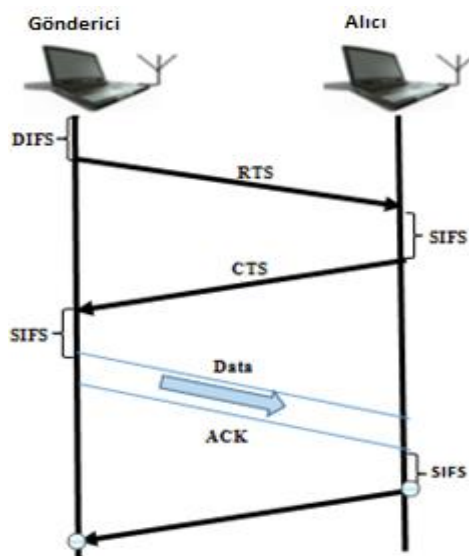
Çizelge 2.2. AC’ler için varsayılan EDCA parametreleri

AC	CW_{min}	CW_{max}	AIFSN	Max TXOP
Arka Plan	15	1023	7	0
En İyi Performans	15	1023	3	0
Video	7	15	2	3.008ms
Ses	3	7	2	1.504ms
Legacy DCF	15	1023	2	0

Burada tahkim çerçeveler arası alan numarası (Arbitration Inter-Frame Space Number-AIFSN) arka plan ve en iyi performans dışında 2 olarak tanımlanmaktadır. Maximum iletim olanağı (Transmission Opportunity-TXOP) ise video ve ses dışında 0'dır.

HCCA, PCF'ye benzer şekilde çalışmaktadır. HCCA, çekişme periyotunda (Contention Period-CP) PCF'den farklı olarak çekişmesiz periyodu (Contention Free Period-CFP) kullanmaktadır. HCCA CP'ye bölünmeden CFP'lerin başlatılmasına izin vermektedir. Bu tür bir CFP, 802.11e'de kontrollü erişim aşaması (Controlled Access Phase-CAP) olarak adlandırılmaktadır. Bir CAP, bir istasyona bir çerçeve göndermek veya bir istasyondan çekişmesiz bir şekilde bir çerçeve almak istediğinde AP tarafından başlatılmaktadır.

DCF; CSMA/CA tabanlıdır. IEEE 802.11e'nin MAC katmanında kullanılan CSMA/CA protokolü, çarpışmaları azaltmak için tasarlanmıştır. RTSED CSMA/CA için kritik öneme sahiptir. CSMA/CA'da yer alan RTS ve CTS çerçeveleri Şekil 2.2'de gösterildiği gibi karşılıklı istek paketleri göndermektedir. Bayt cinsinde olan RTS eşik değerinden büyük veri gelmesi halinde tıkanıklığı önlemek için CSMA/CA aktif olmaktadır [88-90]. RTSED'nin uygun değerlerde seçilmesi, gizli düğüm problemlerini çözmek ve veri trafiğinde gecikmeleri önlemek için önemlidir. CSMA/CA aktif edilirken kullanılan RTSED, yükleme ve birim zamanda işlenen paket miktarı içinde kritik öneme sahip olduğu için bütün çıkış parametre performanslarına bakılarak en uygun şekilde seçilmelidir.



Şekil 2.2. Ağda gönderici ve alıcı ilişkisi (CSMA/CA modeli)

WLAN'larda DCF'de CTS sinyalinin gelmesi ile CSMA/CA protokolü aktif hale gelmektedir ve veri akışı başlamaktadır. Veriler belli büyüklüğün (PED) üzerinde olduğu zaman birçok paket kaybı ve gecikmeler meydana gelmektedir. Büyük veriler hedef düğüme iletilirken yükleme ve birim zamanda işlenen paket miktarını ve QoS'i düşürmektedir. Bu nedenle büyük veriler küçük parçalara bölünerek iletilmektedir [91-94]. Bölünecek parçaların hangi büyüklükte bölüneceği PED olarak adlandırılmakta ve bayt cinsinden ifade edilmektedir. Verilerin iletilmesinde en ideal PED değerinin belirlenmesi performans için kritik öneme sahiptir [95].

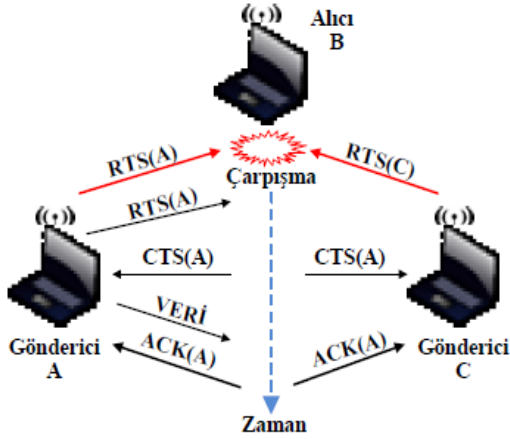
Kablosuz ağlarda veri iletiminde ideal performans için bir diğer önemli parametre arabellek boyutudur. Bit cinsinden ifade edilen AB, kablosuz ağlarda DCF modelindeki CSMA/CA için maliyet açısından çok önemlidir. Arabelleğin büyük olması daha fazla hafızaya ihtiyaç duyurmaktadır. Arabellek düşüklüğü ağda sıkışıklığa ve paket kayıplarına sebep olmaktadır. Paket kayıplarını azaltmak ve uygun maliyetli fiziksel kapasiteye sahip arabellek kullanmak ağdaki sıkışıklığı azaltacak ve paket kayıplarını minimuma indirecektir [96].

2.2. RTSED

WLAN 802.11e protokolünde kullanılan CSMA/CA yöntemi, paket çarpışmalarını ve kayıplarını önlemektedir. CSMA/CA yönteminde veri akışı başlamadan önce kanal, dağıtık çerçeveler arası alan (Distributed Inter Frame Space-DIFS) süresi kadar beklemektedir. Sonrasında RTS sinyali hedef düğüme iletilmektedir. RTS ve CTS sinyalleri arasında kanal Kısa Çerçeveler Arası Alan (Short Inter Frame Space-SIFS) süresi kadar beklemektedir. CTS sinyali geldikten ve SIFS süresi beklenildikten sonra veri akışı sağlanmaktadır. Daha sonra hedef düğümünden CTS mesajı geldikten sonra verilerin iletimi sağlanmaktadır. CSMA/CA yönteminin aktif olarak kullanılabilmesi için gelen verinin boyutunun RTSED'den büyük olması gerekmektedir. RTS/CTS sanal paketlerinin kullanılması esnasında harcanan fazla enerji nedeniyle CSMA/CA protokolünün aktif edip edilmemesi kararını belirleyen parametre RTSED'dir.

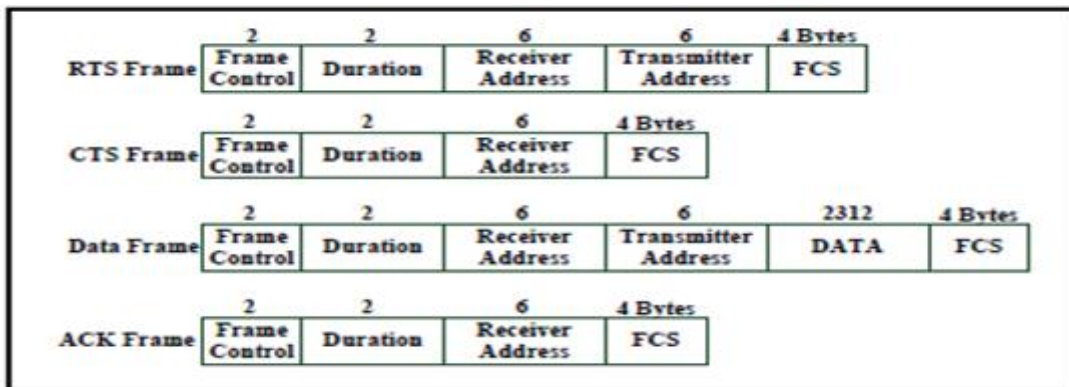
CSMA/CA modelinde bazen bir alıcıya birden fazla düğüm tarafından gönderilen verilerin çakışmasının önlenmesi Şekil 2.3'te gösterildiği gibi RTS mekanizması ile sağlanmaktadır. Burada A ve C düğümündeki RTS çerçevelerinden giden sinyaller B düğümünde

çarpışmaktadır ve bu durumda A ve C düğümleri belli bir rastgele süre (random time) bekleyip RTS çerçevelerini yeniden göndermektedirler.



Şekil 2.3. RTS/CTS mekanizmasında çarpışma

IEEE 802.11 DCF yapısında genel olarak RTS çerçeve yapısı, CTS çerçeve yapısı, veri (data) çerçeve yapısı ve Onay (Acknowledgement-ACK) çerçeve yapıları bulunmaktadır. Çerçeve yapıları yönetim, veri ve kontrol olarak 3'e ayrılmaktadır. Şekil 2.4'te gösterildiği gibi CSMA/CA iç yapısında; kontrol çerçeveleri RTS ve CTS 20'şer bayt, ACK ise 14 bayt olarak yer almaktadır. Çerçevelerde yer alan çerçeve kontrolü (Frame Control) alanındaki tip başlığı alt alanı çerçevenin hangi tip olduğu bilgisini tutmaktadır. Ağda veri iletimi esnasında veri çerçevesinin boyutu RTS'den büyük olduğu durumda RTS çerçevesi gönderilmekte ve CTS çerçevesinin dönüşü sağlanmaktadır. CTS çerçevesi geldikten sonra veri çerçevesi gönderilmektedir. ACK çerçevesi alındıktan sonra veri akışının doğru şekilde sağlandığı teyit edilmektedir. Mekanizmanın kullanılması ile ağlarda çarpışmalar önlenmekte ve gizli düğüm problemi de ortadan kalkmaktadır [97].



Şekil 2.4. CSMA/CA çerçevelerinin genel iç yapıları

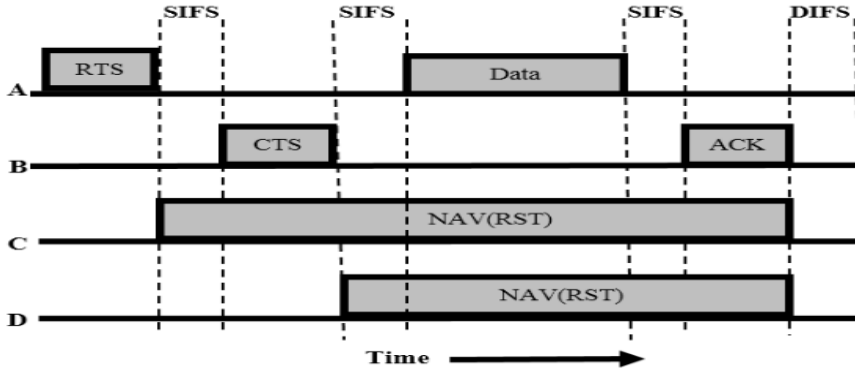
RTS ve CTS sinyalleri 20'şer bayt olduğu için veri çerçevelerine göre çok daha küçük boyutta olup, çarpışmalara sebep olmamaktadır. Bu yapıda çerçeve kontrol sırası (Frame Check Sequence-FCS) 4 bayt ile etiketlenmektedir. Çerçeve kontrolü 2 bayt, alıcı ve gönderici adresleri 6 bayt ve çerçeve süresi 2 bayt olarak etiketlenmektedir [98].

Sanal RTS ve CTS sanal paketleri ağda çok fazla gidiş ve gelişlere sebebiyet verdiği için bir enerji kaybına yol açmaktadır. Enerji kaybını önlemek için verinin RTS eşik değeri belirlenmektedir. RTS/CTS mekanizmasının kullanılıp kullanılmayacağına karar veren veri boyutu değeri RTS eşik değeridir. Bazı durumlarda ise RTS ve CTS sanal paketlerinden daha düşük verilerin gelmesi halinde RTS/CTS mekanizması aktif olmamaktadır. Bu nedenle de RTS eşik değerinin tanımlanması gerekmektedir. Bütün konfigürasyon çıkış parametrelerinde farklı performansların oluşması nedeniyle RTSED'nin ideal değerinin bulunması gerekir.

2.3. PED

WLAN DCF kontrol fonksiyonlarında CSMA/CA modelinde standarttan daha büyük verilerin taşınması esnasında ağda gecikmeler meydana gelmekte ve çoğu zamanda paket kayıpları olmaktadır. Yaşanan paket kayıplarını minimum seviyeye indirip ağdaki gecikmeyi azaltmak için büyük veriler parçalanarak daha küçük veriler haline getirilmektedir. Paketlerin hangi boyutlarda bölüneceğini belirleyen parametre PED'dir [99].

Şekil 2.5'te DCF'de RTS ve CTS sinyallerinden sonra gönderilen ve verinin taşınması sırasında kullanılan Ağ Tahsis Vektörü (Network Allocation Vector-NAV), bekleme zamanını göstermektedir. NAV, ACK sinyali gelene kadar veri alışverişini önlemektedir. Uzun veri bloklarının hatalı taşınma olasılıkları fazladır. NAV'ın kullanılması sadece ilk çerçeveyi koruyacaktır. Bu nedenle WLAN ortamında verilerin dinamik uygun çerçeveler ile iletilmesi hataları azaltacaktır [100-101]. Veri bloğunun hangi büyüklükteki değerler için parçalara bölüneceği PED ile belirlenir.



Şekil 2.5. RTS/CTS mekanizmasında ağ tahsis vektörü

CSMA/CA evrensel kümesinde PED değerleri 0 ile 1024 bayt arasında kullanılmaktadır.

2.4. AB

WLAN CSMA/CA protokolünde bekleme süreleri net olmadığı için AB'nin dinamik olarak seçilmesi ağlarda performans artışı sağlamaktadır. AB belirlenirken RTSED ve PED değerleri de her zaman göz önünde bulundurulmalıdır [80]. AB farklı algoritmalar ve yöntemlerle kullanıldığında kanal verimliliğini büyük oranda arttırmaktadır.

AB farklı algoritmalar ve yöntemlerle kullanıldığında kanal verimliliğini büyük oranda arttırmaktadır. Bu nedenle AB'nin kullanılacağı yöntem ve algoritma seçimleri probleme uygun olarak derinlemesine analiz ile gerçekleştirilmelidir.

Tez çalışmasında, 802.11e ile genişletilen modeldeki DCF üzerinden, CSMA/CA tetikleyici parametreleri güncellenerek ses ve multimedya uygulamalarında performans artışı sağlanmıştır. Çalışmada doğrulama işlemi uzman sezgileri ile benzetim aracı üzerinden elde edilen sonuçlar ile oluşturulmuştur. Yapılan çalışmada olay bazlı ağ tasarımları kullanılmıştır.

2.5. Benzetim Ortamı ve Özellikleri

Benzetim, gerçek dünyayı taklit eden ve bilişim sistemlerinde mevcut veya uygulanacak işlemler hakkında nesnel bilgi alabileceğimiz, eğitim amaçlı kullanılan uygulamalar olarak bilinmektedir. Ayrıca benzetim farklı mühendislik alanlarında kullanılan çok önemli bir teknolojidir. Genel olarak kullanım alanları ekonomi, finans, kimya, fizik, biyoloji olmakla

beraber bilgisayar mühendisliği ve makine mühendisliği gibi alanlarda da sıkça kullanılmaktadır [102].

Benzetimi yapılacak haberleşme sistemlerinin karmaşık yapıları nedeniyle, sisteme ait tüm detayların hassas olarak ele alınması mümkün değildir. Bunun yerine ağ, genellikle matematiksel ya da mantıksal ilişkilerle ifade edilen model şeklinde tanımlanır. Daha sonra bilgisayar yardımıyla oluşturulan modelin benzetimi yapılır. Benzetim sonuçlarının doğruluğu modelin gerçek sisteme ne kadar yakın olduğuna bağlıdır. Gerçeğine daha yakın model daha fazla detay gerektirir. Buna paralel olarak modelin tasarlanması ve benzetimin gerçekleştirilmesi de daha uzun zaman alır.

Bilgisayar ağlarında, ağların büyüklüğü ve yönetiminin gücü nedeniyle yüksek maliyetler ortaya çıkmaktadır. Bu nedenle bilgisayar ağ tasarımları gerçek uygulamardan önce benzetim araçları ile modellenmektedir. Özellikle çalışan canlı sistemlerde mevcut yapıyı bozmamak için benzetim araçları daha sık kullanılmaktadır. Ayrıca ağ protokollerinin performans karşılaştırmaları ve yeni protokollerin testleri aşamasında benzetim araçları sıkça kullanılmaktadır [103].

Ağ benzetimi test aşamalarında OSI (Open Systems Interconnection) ve TCP referans modelleri detaylı olarak bütün parametreleri ile problemsiz test edilebilmektedir. Ağ benzetimi, ağ ürünlerini geliştiren firmalar tarafından eğitim amaçlı sıkça kullanılmaktadır. CISCO firması CCNA (Cisco Certified Network Associate), CCNP (Cisco Certified Network Professional) v.b eğitimler için laboratuvar ortamı yerine geliştirdiği benzetim programlarını kullanmaktadır [104].

Benzetim araçlarının bazıları açık kaynak kodlu olmasına rağmen bazıları ise ticari amaçlı olup, bilişim sektöründe yapılan çalışmalarda kullanılmaktadır.

Ticari ve açık kaynak kodlu ağ benzetim araçlarından bazıları Çizelge 2.3'te yer almaktadır.

Çizelge 2.3. Benzetim araçları

Ticari olanlar (Kapalı Kaynak) Yazılımlar	Ticari olmayanlar (Açık Kaynak) Yazılımlar
Riverbed Modeler	NS-2
Cisco Packet Tracer	NS-3
QualNet	Omnet++
Router.Sim	SSFNet
	J-Sim
	GTNetS (Georgia Tech Network Simulator)

Her benzetim aracının kendine özel çok çeşitli özellikleri bulunmaktadır. Genel olarak benzetim araçları; birden fazla düğüm ile birkaç konfigürasyon üzerinde trafik akışını kontrol etmek, analizini gerçekleştirmek ve performansını arttırmak amacıyla kullanılmaktadır.

2.5.1. Riverbed Modeler

Riverbed Modeler ilk olarak “OPNET (Optimized Network Engineering Tool) Technologies” tarafından ticari marka olarak üretilen popüler bir ağ simülasyon aracıdır. 2008 yılının sonlarına doğru popülerliğini arttırmış ve 2012’de performans şirketi “Riverbed Technology” tarafından satın alınmıştır. Riverbed Modeler benzetim ortamında ağ performans araştırmaları ve modellemeleri yapılmaktadır. Yazılım ortamında esnek ağ haberleşme cihazları, protokoller ve uygulamalar test edilebilmektedir. Riverbed Modeler benzetim aracı kullanıcı dostu arayüzü ile görselliği ön plana almakta ve grafik editör arayüzü ile kullanıcıların benzetim ortamını kullanması için çok uygun bir ortam sağlamaktadır. Grafik arayüzü ile bütün ağ topolojilerinin ve OSI’deki katmanlarda birçok uygulama ve modellemenin test edilmesini sağlamaktadır. Riverbed Modeler’de gerçek sistemlerin uygulanması ve grafik tasarım eşleşmesini sağlamak amacıyla nesne tabanlı programlama tekniği kullanılmaktadır.

Riverbed Modeler üç temel fonksiyona sahiptir. Bunlar genel olarak modelleme, benzetim ve analizdir. Modelleme fonksiyonu çevresel grafik uygulamaları ile bütün modellerin ve protokollerin denenebileceği bir ortam sunar. İstatistik sonuçları grafiklerle, animasyon yöntemleri ile veya hata ayıklama (debug) ekranı ile kullanıcılar tarafından kolaylıkla görüntülenebilmektedir [105].

2.5.2. Riverbed Modeler ve diğer benzetim araçları arasındaki farklar

Benzetim aracı kıyaslamalarında uygulama içeriği ve veri analiz yöntemleri öne çıkmaktadır. Riverbed Modeler, diğer benzetim araçları ile kıyaslandığı zaman çeşitli avantaj ve dezavantajları ortaya çıkmaktadır. NS (Network Simulator)-2, NS-3, Omnet++ ve Cisco Packet Tracer en yaygın olarak kullanılan benzetim araçlarıdır.

NS-2 ve Riverbed Modeler

NS nesne tabanlı olup C++ programa diliyle yazılmıştır. İlk olarak 1989 yılında geliştirilmeye başlanılmıştır [106]. NS-2’de her koşum sonrasında NAM (Network Animator) adlı bir dosya üretilmekte ve ağ yapısı ile trafik akış analizi burada yapılmaktadır. Veri akış süresi, veri iletim süresi, yüklenen paket miktarı v.b bilgiler NAM tarafından okunarak görsel arayüz üzerinden sunulmaktadır [8]. NS-2 açık kaynak kodlu olduğundan ücretsizdir ve Riverbed Modeler’e göre fiyat konusunda daha avantajlıdır. Ancak görsel arayüz açısından Riverbed Modeler çok daha performanslıdır. NS-2, dokümantasyon açısından Riverbed Modeler’e benzer şekilde performans göstermektedir [107].

Omnet++ ve Riverbed Modeler

Omnet++; açık kaynak kodlu, nesne tabanlı ve C++ programlama dili ile yazılmış ücretsiz bir benzetim aracıdır. Omnet++; ileri düzey alt yapıya sahip olup geniş bir dokümantasyon veri tabanına sahiptir. Ethernet, IPv6, mobil ip gibi birçok protokol için hazır protokolü bulunmaktadır. Yeni bir protokol veya model geldiğinde ilgili algoritmalar Omnet++ üzerinde kodlanarak test edilebilmektedir [108]. Omnet++ benzetim aracının grafik akış arayüzü Riverbed Modeler kadar gelişmiş seviyede değildir.

NS-3 ve Riverbed Modeler

NS-3; dokümantasyonu zengin, nesne yönelimli programlama dilleri olan Python ve C++ ile yazılmış esnek ve geliştirilebilir bir benzetim aracıdır. NS-3, topolojiler ve zamanlayıcılar gibi farklı konfigürasyonları destekleyen bir alt yapıya sahiptir. Ayrıca linux tabanlı işletim sistemlerinde de kullanılmaktadır [109]. Genel olarak grafik arayüzü

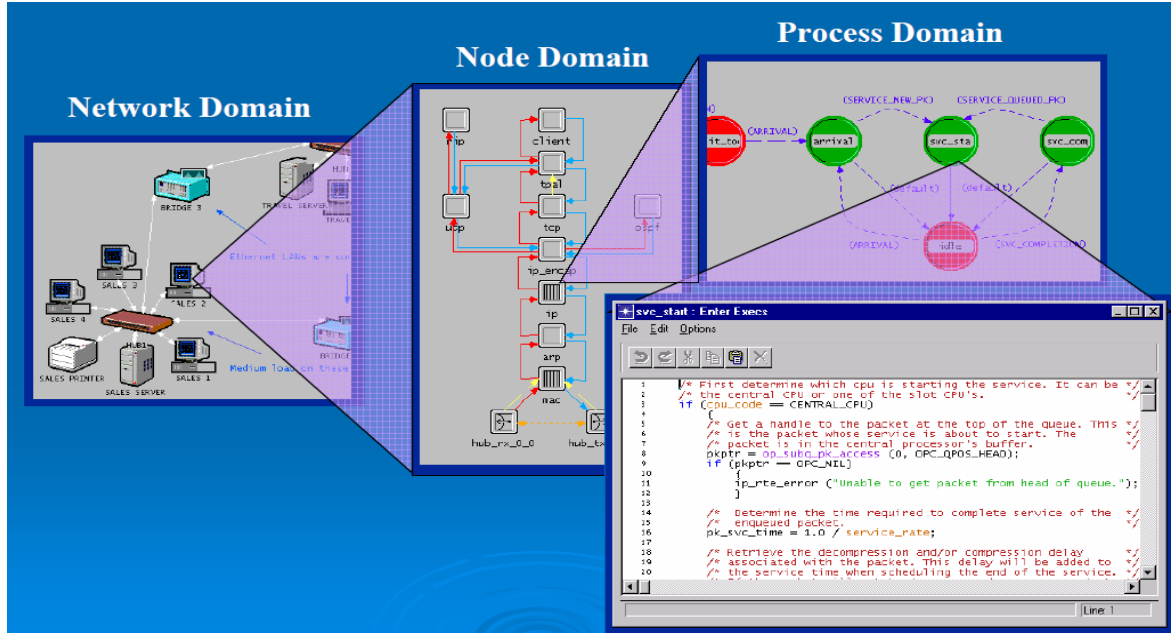
Riverbed Modeler kadar zengin değildir. Son yıllarda devamlı gelişmekte olan NS-3, kod satır sayısı ve kullanıcı sayısını arttırmaya devam etmektedir.

Sonuç olarak Riverbed Modeler, sağladığı altyapı ve performans analizi açısından diğer benzetim araçlarına göre daha fazla ön plana çıkmaktadır. Ayrıca Riverbed Modeler ile sonuçların grafiklerle ve animasyon yöntemleri ile görsel arayüzlerle sunulması kullanıcılara kolaylıklar sağlamaktadır. Bütün bu avantajları nedeniyle kablosuz ağlarda performans analizi için tez çalışması uygulamalarında Riverbed Modeler kullanılmıştır.

2.5.3. Riverbed Modeler yazılımının hiyerarşik yapısı

Riverbed Modeler; proje, düğüm ve proses editörleri olmak üzere 3 temel editör kullanmaktadır. Editörler genelde hiyerarşik bir yapıya sahiptir. Proje editörü coğrafik konumlandırmaların yapıldığı, bağlantıların ve ağ modellerinin oluşturulduğu editördür. Üst editör olduğu için düğüm modellerini güncelleme yetkisine sahiptir ve sonuçlar grafik arayüz üzerinden burada gösterilmektedir [110]. Düğüm editörü ise nesnelerin geliştirildiği ve modellendiği editördür. Verilerin iletilmesi, nesnelerin işlevlerinin kontrol edilmesi ve proses editörlerinin değiştirilmesi düğüm editöründe yapılmaktadır. Riverbed Modeler hiyerarşik modellemesinde en altta yer alan editör olan proses editöründe ise durum geçiş diyagramları bulunmakta ve bu editörde proto-C kodları ile yazılım geliştirilmektedir [111].

Şekil 2.6'da proses editörü, düğüm editörü ve proje editörünün hiyerarşik yapısı ve ilişkisi gösterilmiştir. Proje editöründe yer olan nesneler düğüm editöründe yer alan modüllerden meydana gelmiştir. Modüllerin işlevi ve fonksiyonları proto-C kodu içeren geçiş diyagramları tarafından sağlanmaktadır [110].



Şekil 2.6. Editörler arası geçişler

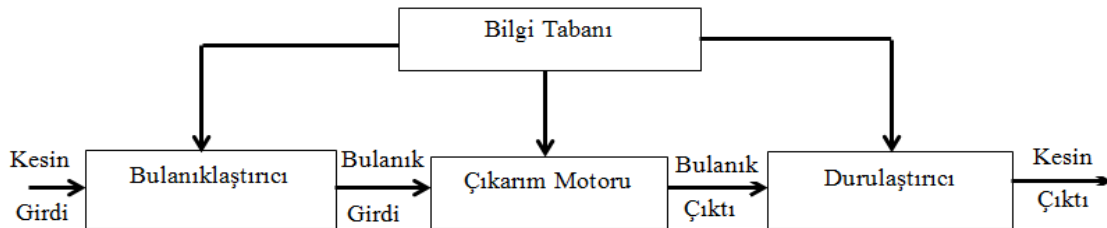
Riverbed Modeler benzetim aracında ayrıca modülasyon editörü, PDF (Portable Document Format) editörü, arayüz kontrol bilgisi editörü, anten model editörü ve bağlantı model editörü de bulunmaktadır. PDF editöründe matematiksel işlemler yapılmakta, arayüz kontrol bilgisi editöründe haberleşme kontrol bilgileri tutulmakta ve anten model editöründe ise kablosuz alıcı ve vericiler arasında anten örnekleri oluşturulmaktadır. Bağlantı modülünde ise kanal sayısı, bağlantı modelleri ve veri iletişim hızı özelliklerinin düzenlemesi sağlanmaktadır. Bütün bu avantajları nedeniyle tez çalışmasında Riverbed Modeler kullanılmıştır.

3. GERÇEKLEŞTİRİLEN BULANIK MANTIK, YSA ve GERİ BESLEME KONTROL TABANLI PERFORMANS ARTTIRMA

Bu bölümde RTSED, PED ve AB giriş değişkenleri ile bulanık mantık, YSA ve geri besleme kontrollü modeller kullanılarak çıkış parametrelerinden gecikme süresi, yükleme süresi, birim zaman da gönderilen paket sayıları, alınan ve gönderilen veri trafiği, alıcı kanal kullanım durumu ve verici kanal kullanım durumu ve düğüm bazlı birim zamanda işlenen paket miktarı çıkış parametreleri optimize edilmiştir. İlk olarak WLAN'larda farklı giriş ve çıkış parametreleri ile elde edilen veri setleri kullanılarak bulanık mantık tabanlı performans arttırımı sağlanmıştır. Sonraki aşamada YSA uygulamaları ile performans tahminlemesi gerçekleştirilmiştir. Daha sonra giriş ve çıkış parametreleri kontrol edilerek benzetim anında bulanık mantık, döngü tablosu ve kaba kuvvet algoritması ile performansın arttırılması sağlanmıştır. Ayrıca dış faktörlerin performans etkileri analiz edilmiş ve önerilen yeni yöntemin farklı çevresel değişkenlerle test edilmesi sağlanarak performans artışları analiz edilmiştir.

3.1. Bulanık Mantık ve Giriş Değişkenleri ile Performans Geliştirme

Bulanık mantıkta Şekil 3.1'deki gibi bilgi tabanı ile desteklenen bulanıklaştırma ünitesi, çıkarım motoru ve durulaştırma ünitesi bulunmaktadır. Bulanıklaştırma ünitesinde giriş değişkenlikleri Riverbed Modeler'de test edilerek uygun bulanık kümelerle dönüştürülmüştür. Giriş değişkenlerine etiket verilerek sözel ifadelerle dilsel bir ölçek değişikliği yapılmış ve kural tabanı oluşturulmuştur. Çıkarım motorunda ise kurallar üzerinde bulanık mantık kuralları uygulanarak bulanık çıkışlar elde edilmiştir. Daha sonra durulaştırma ünitesinde sözel değişkenler sayısal olarak ifade edilerek kesin çıktılar elde edilmiştir.



Şekil 3.1. Bulanık mantık tabanlı bilgi işlemenin genel çerçevesi

Şekil 3.2’de gösterildiği gibi Mamdani bulanık mantık sisteminde giriş parametreleri olarak RTSED, PED ve AB değerleri kullanılmıştır. Çıkış parametreleri olarak ise gecikme süresi, yükleme ve birim zamanda işlenen paket miktarları kullanılmıştır.



Şekil 3.2. Bulanık mantık giriş ve çıkış parametreleri genel çerçevesi

3.1.1. Üyelik fonksiyonları ve bulanık kural tabanının oluşturulması

Veri setleri oluşturmak için RTSED, PED ve AB giriş değerleri Çizelge 3.1’de yer alan 27 farklı konfigürasyon ile test edilmiştir.

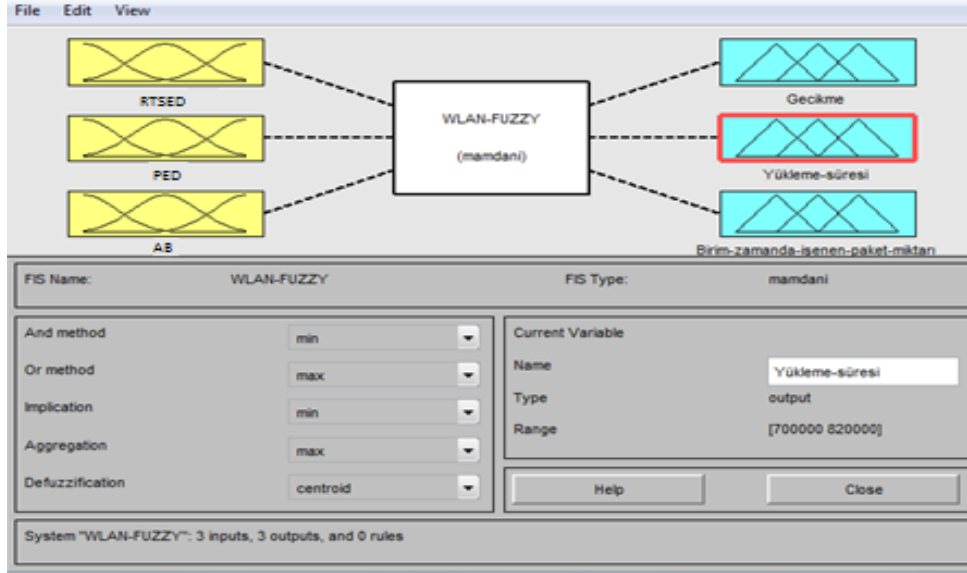
Çizelge 3.1. 27 konfigürasyonlu parametre evrensel küme ve test aralıkları

Giriş değer aralıkları	RTSED (bayt)	PED (bayt)	AB (bit)
1-9	512	512-700-1024	8000-10 000-16 000
10-18	700	512-700-1024	8000-10 000-16 000
19-27	1 024	512-700-1024	8000-10 000-16 000

Çizelge 3.1’de yer alan girdiler Riverbed Modeler’de yapılan test çalışmalarında elde edilen ortalama gecikme süreleri, ortalama yükleme ve ortalama birim zamanda işlenen paket miktarı değerleri bulanık kural tabanı oluşturmak için RTSED, PED ve AB seçimlerinde kullanılmıştır.

Test çalışmaları kapsamında MATLAB (Matrix Laboratory) ile gerçekleştirilen Mamdani bulanık mantık tasarım şemasında çıkış parametreleri olarak kullanılan gecikme, 0,008 sn ile 0,003 sn aralığında, yükleme 700 000 bit/sn ile 820 000 bit/sn aralığında ve birim zamanda işlenen paket miktarı ise 250 000 bit/sn ile 850 000 bit/sn aralığında elde edilmiş

ve çıkış üyelik kümeleri de bu değerler arasında sınıflandırılmıştır. Kullanılan MATLAB bulanık mantık tasarım şeması Şekil 3.3'te verilmiştir.



Şekil 3.3. MATLAB bulanık mantık tasarım şeması

Bulanık mantık sisteminde, üç farklı değişken girdi olarak, üç farklı değişken ise çıktı olarak kullanılmıştır. Gerçekleştirilen test çalışmaları ile değerlerin üyelik fonksiyonları ve kural tabanı Çizelge 3.2'deki gibi belirlenmiş ve kümeler oluşturulmuştur.

Çizelge 3.2. Bulanık kural tablosu

GİRİŞLER			ÇIKIŞLAR		
RTSE D (bayt)	PED (bayt)	AB (bit)	Gecikme (sn)	Yükleme (bit/sn)	Birim zamanda işlenen paket miktarı (bit/sn)
KÜ	KÜ	A	F	F	ÇA
KÜ	KÜ	NO	F	ÇA	NO
KÜ	KÜ	Ç	ÇF	F	ÇF
KÜ	NO	A	NO	NO	ÇA
KÜ	NO	NO	F	NO	A
KÜ	NO	Ç	ÇF	F	ÇF
KÜ	BÜ	A	A	F	ÇA
KÜ	BÜ	NO	NO	A	ÇF

Çizelge 3.2. Bulanık kural tablosu-devam

GİRİŞLER			ÇIKIŞLAR		
KÜ	BÜ	Ç	F	ÇF	ÇF
NO	KÜ	A	NO	NO	ÇA
NO	KÜ	NO	F	F	NO
NO	KÜ	Ç	ÇF	NO	ÇF
NO	NO	A	A	NO	ÇA
NO	NO	NO	NO	NO	ÇF
NO	NO	Ç	ÇF	ÇA	F
NO	BÜ	A	A	A	ÇA
NO	BÜ	NO	NO	NO	NO
NO	BÜ	Ç	F	A	ÇF
BÜ	KÜ	A	A	NO	ÇA
BÜ	KÜ	NO	N	NO	NO
BÜ	KÜ	Ç	ÇF	A	F
BÜ	NO	A	ÇA	ÇA	ÇA
BÜ	NO	NO	NO	A	A
BÜ	NO	Ç	F	A	ÇF
BÜ	BÜ	A	ÇA	A	ÇA
BÜ	BÜ	NO	A	ÇA	A
BÜ	BÜ	Ç	NO	A	ÇF

Kural tabanında kullanılan küçük (KÜ), büyük (BÜ), Normal (NO), çok-az (ÇA), az (A), fazla (F), çok-fazla (ÇF) değerlerinde 27 farklı kural ile sözel sınıflar oluşturulmuştur. Kural tabanı;

[*EĞER RTSED= '...' ve PED= '...' ve AB= '...' İSE gecikme= '...' ve yükleme= '...' ve birim zamanda işlenen paket miktarı= '...'] şeklinde oluşturulmuştur.*

Örneğin *EĞER RTSED= 'KÜÇÜK' ve PED= 'KÜÇÜK' ve AB= 'AZ' İSE gecikme= 'FAZLA' ve yükleme= 'FAZLA' ve birim zamanda işlenen paket miktarı= 'ÇOK-AZ'*dır.

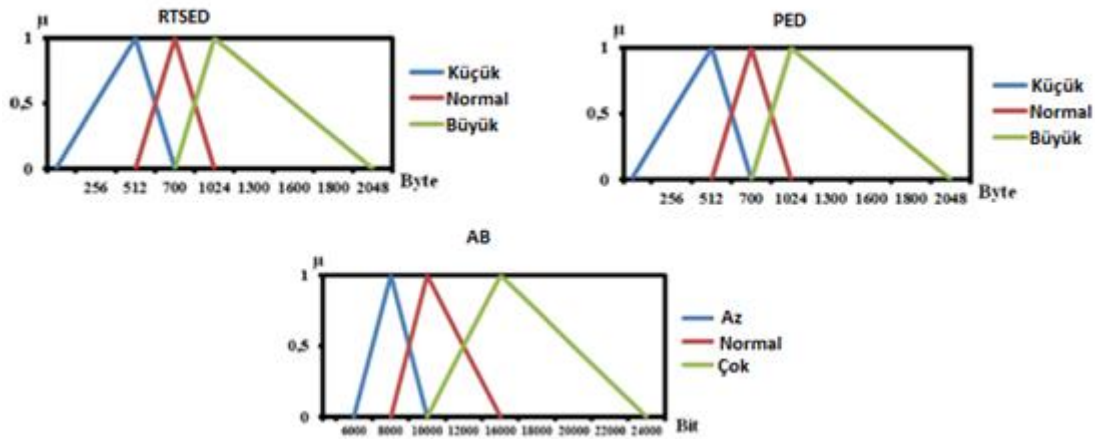
3.1.2. RTSED, PED ve AB'nin bulanık mantıkla belirlenmesi

Bulanık mantık çıktıları için çıkarım yöntemi olarak birleşim-kesişim (max-min) operatörü kullanılmıştır. Durulaştırma yöntemi olarak ise ağırlık merkezi kullanılmıştır. Ağırlık merkezi metodu formül 1'deki gibi hesaplanmıştır.

u RTSED ve PED için üyelik fonksiyonunda x ekseninde yer alan bayt değeri iken; AB için üyelik fonksiyonunda x ekseninde yer alan bit değeri olmak üzere;

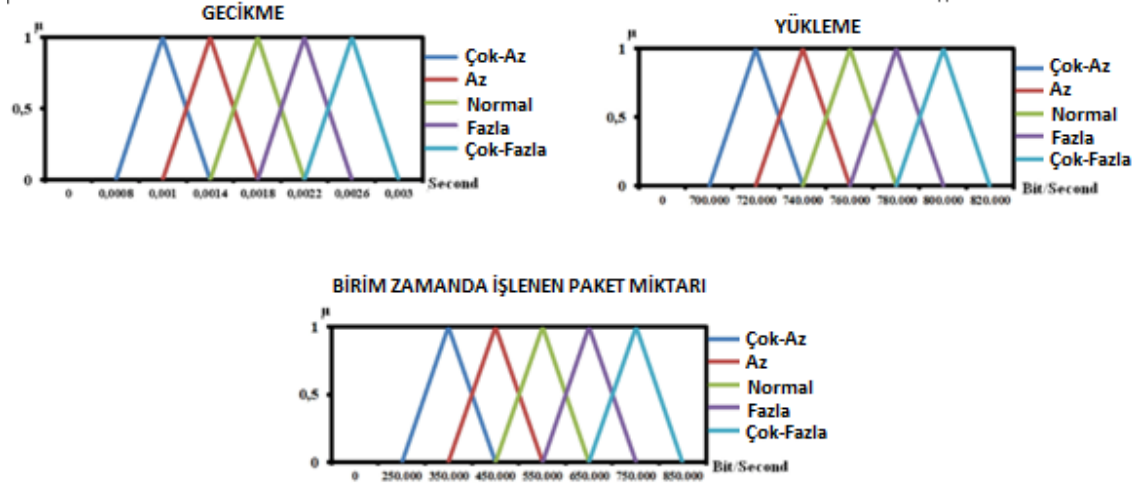
$$U = \frac{\int \mu(u).u.du}{\int \mu(u).du} \quad (1)$$

Şekil 3.4'te yer alan giriş üyelik fonksiyonları için RTSED 0-512 bayt arası 'Küçük', 512-700 bayt aralığı için 'Normal', 700-1024 bayt aralığı için ise 'Büyük' değerleri kullanılmıştır. PED içinde 0-512 bayt aralığı 'Küçük', 512-700 bayt aralığı için 'Normal', 700-1024 bayt aralığı için ise 'Büyük' değerleri kullanılmıştır. AB'de ise 6000-10 000 bit aralığı 'Az', 10 000-16 000 bit aralığı 'Normal' 16 000-24 000 bit aralığı için ise 'Çok' sözel değerleri kullanılmış ve bu aralıklar küme aralıkları olarak belirlenmiştir.



Şekil 3.4. Giriş üyelik fonksiyonları

Şekil 3.5'te yer alan çıkış üyelik fonksiyonları için gecikme süresinde 0,0008-0,0014 sn aralığı için 'çok-az', 0,0010-0,0018 sn aralığı için 'Az', 0,0014-0,0022 sn aralığı için 'Normal', 0,0018-0,0026 sn aralığı için 'Fazla' ve 0,0022-0,003 sn aralığı için 'çok-fazla' sözel değerleri kullanılmıştır.



Şekil 3.5. Çıkış üyelik fonksiyonları

Yükleme için 700 000-740 000 bit/sn aralığı ‘çok-az’, 720 000-760 000 bit/sn aralığı için ‘Az’, 740 000-780 000 bit/sn aralığı için ‘Normal’, 760 000-800 000 bit/sn aralığı için ‘Fazla’ ve 780 000-820 000 bit/sn aralığı için ‘çok-fazla’ sözel değerleri kullanılmıştır. Birim zamanda işlenen paket miktarı için ise 250 000-450 000 bit/sn aralığı için ‘Çok-az’, 350 000-550 000 bit/sn aralığı için ‘Az’, 450 000-650 000 bit/sn aralığı için ‘Normal’, 550 000-750 000 bit/sn aralığı için ‘Fazla’ ve 650 000-850 000 bit/sn aralığı için ‘çok-fazla’ sözel değerleri kullanılmış ve bu aralıklar küme aralığı olarak tanımlanmıştır.

3.1.3. Geliştirilen modelin arayüz kodlama denklemlerinin oluşturulması

Yapılan bulanık mantık çalışmalarında MATLAB’ta elde edilerek oluşturulan mamdani metodu, çıkarım, durulaştırma, giriş ve çıkış üyelik fonksiyonları ile kural tabanı metodları kodlama alanı aşağıda yer almaktadır.

System:

Name='WLAN-FUZZY', Type='mamdani', Version=2.0, Num_Inputs=3,
 Num_Outputs=3, Num_Rules=27, And_Method='min', Or_Method='max',
 Imp_Method='min', Agg_Method='max'
 Defuzz_Method='centroid'

Input1:

Name='RTSED'

Range=[0 1024]

NumMFs=3

MF1='Kısa':trimf,[0 512 700]

MF2='Orta':trimf,[512 700 1024]

MF3='Uzun':trimf,[700 1024 2046]

Input2:

Name='PED'

Range=[0 1024]

NumMFs=3

MF1='Kısa':trimf,[0 512 700]

MF2='Orta':trimf,[512 700 1024]

MF3='Uzun':trimf,[700 1024 2048]

Input3:

Name='AB'

Range=[6000 16 000]

NumMFs=3

MF1='Az':trimf,[6000 8000 10 000]

MF2='Normal':trimf,[8000 10 000 16 000]

MF3='Çok':trimf,[10 000 16 000 24 000]

Output1:

Name='Gecikme'

Range=[0.0008 0.003]

NumMFs=5

MF1='Çok-Az':trimf,[0,0008 0,001 0,0014]

MF2='Az':trimf,[0,001 0,0014 0,0018]

MF3='Normal':trimf,[0,0014 0,0018 0,0022]

MF4='Fazla':trimf,[0,0018 0,0022 0,0026]

MF5='Çok-Fazla':trimf,[0,0022 0,0026 0,003]

Output2:

Name='Yükleme-süresi'

Range=[700 000 820 000]

NumMFs=5

MF1='Çok-Az':trimf',[700 000 720 000 740 000]

MF2='Az':trimf',[720 000 740 000 760 000]

MF3='Normal':trimf',[740 000 760 000 780 000]

MF4='Fazla':trimf',[760 000 780 000 800 000]

MF5='Çok-Fazla':trimf',[780 416.184971098 800 416.184971098 820 416.184971098]

Output3:

Name='Birim-zamanda-işenen-paket-miktarı'

Range=[250 000 850 000]

NumMFs=5

MF1='Çok-Az':trimf',[250 000 350 000 450 000]

MF2='Az':trimf',[350 000 450 000 550 000]

MF3='Normal':trimf',[450 000 550 000 650 000]

MF4='Fazla':trimf',[550 000 650 000 750 000]

MF5='Çok-Fazla':trimf',[650 000 750 000 850 000]

Kural sayısal gösterimi:

1 1 1, 4 5 1 (1) : 1

1 1 2, 4 1 3 (1) : 1

1 1 3, 5 4 5 (1) : 1

1 2 1, 3 3 1 (1) : 1

1 2 2, 4 3 2 (1) : 1

1 2 3, 5 4 5 (1) : 1

1 3 1, 2 3 1 (1) : 1

1 3 2, 3 2 3 (1) : 1

1 3 3, 4 5 5 (1) : 1

2 1 1, 3 3 1 (1) : 1

2 1 2, 4 4 3 (1) : 1

2 1 3, 5 3 5 (1) : 1

2 2 1, 2 3 1 (1) : 1

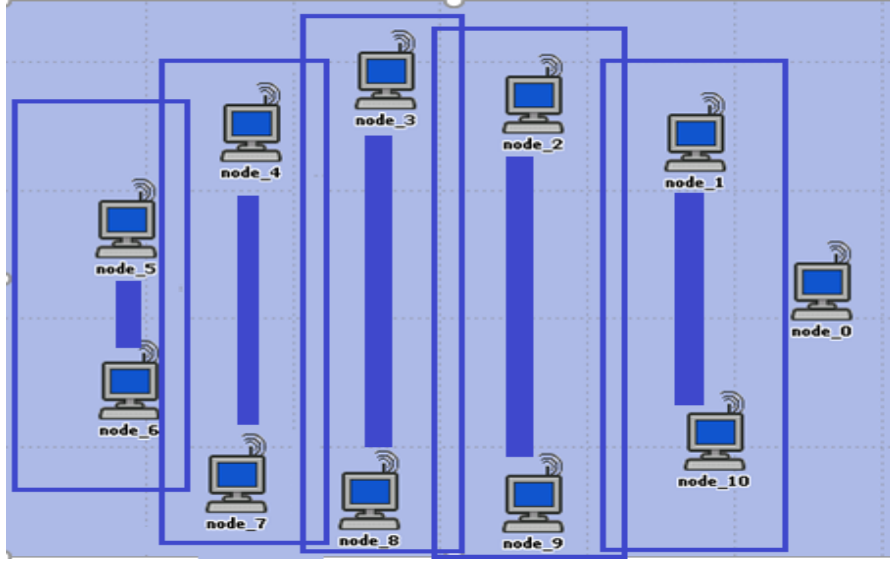
2 2 2, 3 3 3 (1) : 1

2 2 3, 5 1 4 (1) : 1
 2 3 1, 2 2 1 (1) : 1
 2 3 2, 3 3 3 (1) : 1
 2 3 3, 4 2 5 (1) : 1
 3 1 1, 2 3 1 (1) : 1
 3 1 2, 3 3 3 (1) : 1
 3 1 3, 5 2 5 (1) : 1
 3 2 1, 1 1 1 (1) : 1
 3 2 2, 3 2 2 (1) : 1
 3 2 3, 4 2 5 (1) : 1
 3 3 1, 1 2 1 (1) : 1
 3 3 2, 2 1 2 (1) : 1
 3 3 3, 3 2 5 (1) : 1

Geliştirilen model için elde edilen sonuçlara göre gecikme süresi, yükleme süresi ve birim zamanda işlenen paket miktarı optimize edilmiştir. Elde edilen sonuçlar bölüm 4'te klasik yöntem test sonuçları ile karşılaştırılmış ve performans artış oranları çıkartılmıştır.

3.2. YSA Tabanlı Giriş Değişkenleri ile Performans Geliştirme

WLAN'larda alınan veri trafiği ve gönderilen veri trafiği tahminlemesi için YSA algoritması kullanılarak, RTSED, PED ve AB giriş parametreleri optimize edilmiştir. Bu amaçla ilk olarak standart test çalışmalarında kullanılan Riverbed Modeler benzetim aracında gerçekleştirilen ağ modeli Şekil 3.6'da gösterilmiştir. 11 düğüm kullanılarak 1 adet düğüm erişim noktası olarak kalan 10 düğüm ise karşılıklı simetrik olarak konumlandırılmış ve düğüm 4 ve düğüm 7 arasındaki veri trafiği analiz edilmiştir.



Şekil 3.6. YSA için hedef düğümlerin yerleşimi

Çizelge 3.3'te yer alan girdiler YSA ve döngü tablosu çalışmalarında alıcı kanal verimliliği, verici kanal verimliliği, alınan veri trafiği ve gönderilen veri trafiği çıkış parametrelerini optimize etmek için kullanılmıştır.

Çizelge 3.3. 125 konfigürasyonlu parametre evrensel küme ve test aralıkları

Giriş değer aralıkları	RTSED (bayt)	PED (bayt)	AB (bit)
1-25	600	600-700-900-1000-1200	9000-10 000-12 000-13 000-15 000
26-50	700	600-700-900-1000-1200	9000-10 000-12 000-13 000-15 000
51-75	900	600-700-900-1000-1200	9000-10 000-12 000-13 000-15 000
76-100	1000	600-700-900-1000-1200	9000-10 000-12 000-13 000-15 000
101-125	1200	600-700-900-1000-1200	9000-10 000-12 000-13 000-15 000

3.2.1. Alınan ve gönderilen veri trafiği için YSA çalışmaları

İlk olarak Riverbed Modeler'de yapılan test çalışmasında farklı RTSED, PED ve AB değerleri için alıcı kanal kullanım durumu %5,83 ile %19,6 arasında değişkenlik göstermiştir. Verici kanal kullanım durumu ise %0,58 ile %2,08 arasında değişkenlik göstermiştir. Alınan veri trafiği 453 001 bit/sn ile 1 080 477 bit/sn arasında, gönderilen veri trafiği ise 44 665 bit/sn ile 117 564 bit/sn arasında inişli ve çıkışlı değerler ortaya koymuştur. Çizelge 3.4'teki ölçek kullanılarak giriş parametreleri evrensel kümede RTSED

ve PED için 600 bayt, 900 bayt ve 1200 bayt değerleri ile test edilirken, AB ise evrensel kümede 9000 bit, 12 000 bit ve 15 000 bit değerleri ile test edilmiştir.

Çizelge 3.4. 27 farklı giriş değerleri test sonuçları

Konfigürasyon Test No	RTSED (bayt)	PED (bayt)	AB (bit)	Alınan Veri Trafığı (bit/sn)	Gönderilen Veri Trafığı (bit/sn)
1	600	600	9000	547 725	64 826
2	600	600	12 000	1 041 157	110 998
3	600	600	15 000	1 029 788	100 998
4	600	900	9000	503 070	50 902
5	600	900	12 000	922 228	102 865
6	600	900	15 000	930 434	89 055
7	600	1200	9000	464 714	51 496
8	600	1200	12 000	882 342	84 935
9	600	1200	15 000	922 743	102 256
10	900	600	9000	564 168	57 200
11	900	600	12 000	1 079 527	117 564
12	900	600	15 000	1 050 169	114 878
13	900	900	9000	508 493	54 458
14	900	900	12 000	930 463	107 933
15	900	900	15 000	942 382	102 021
16	900	1200	9000	453 001	47 517
17	900	1200	12 000	869 619	100 447
18	900	1200	15000	931 431	97 473
19	1200	600	9000	568 112	65 527
20	1200	600	12 000	1 065 258	95 066
21	1200	600	15000	1 080 477	110 960
22	1200	900	9000	499 593	55 048
23	1200	900	12 000	976 559	88 436
24	1200	900	15 000	973 551	99 012
25	1200	1200	9000	466 978	44 665
26	1200	1200	12 000	943 827	89 538
27	1200	1200	15 000	906 218	99 012

Yapılan çalışmada WLAN’larda kanal kullanım durumunu tespit etmek, alınan veri trafiği ve gönderilen veri trafiği çıkış parametrelerini tahmin etmek için YSA kullanılmıştır. YSA’da RTSED, PED ve AB giriş değerleri tahminlemede girdi olarak kullanılmıştır. YSA iki aşamada çalışmaktadır. İlk aşama da uygun RTSED, PED ve AB değerleri eğitilerek giriş değişkenlerinin ağırlıklarının sözlüğü oluşturulmuştur. Sonraki adım olan test aşamasında ise istatistiksel yöntem ile çoklu regresyon analizi yapılmıştır.

Yapılan çalışmada istatistiksel hata analizi, LM (Levenberg Marquardt) öğrenme algoritması tekniği kullanılarak gerçekleştirilmiştir. LM kullanılmasının sebebi hızlı eğitim sağlamasıdır. YSA'da transfer fonksiyonları; tansig, purelin, logsig v.b olarak adlandırılmaktadır [112-113]. Yapılan çalışmada tansig, logsig ve purelin transfer fonksiyonları uygulanmıştır. Kök Ortalama Kare Hatasının (Root Mean Square Error-RMSE) istatistiksel hataları (Ortalama Karesel Hata), fraksiyonu R^2 (Varyansın Mutlak Fraksiyonu) ve hata yüzdesi ise MEP (Mean Percentage Error-Ortalama Hata Yüzdesi) dir. R^2 , RMSE ve MEP değerleri için kullanılan formüller formül 2, 3 ve 4'te gösterilmiştir.

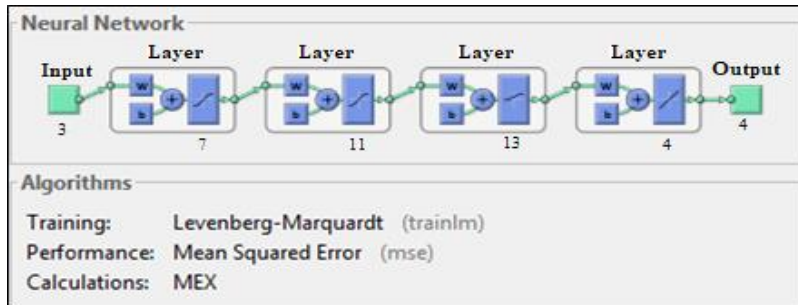
p örnek boyut, t beklenen sonuç, o gözlemlenen sonuç olmak üzere;

$$RMSE = \left((1/p) \sum_j |t_j - o_j|^2 \right)^{1/2} \quad (2)$$

$$R^2 = 1 - \left(\frac{\sum_j (t_j - o_j)^2}{\sum_j (o_j)^2} \right) \quad (3)$$

$$MEP = \frac{\sum_j \left(\frac{t_j - o_j}{t_j} \right) \times 100}{p} \quad (4)$$

3 girişli (RTSED, PED ve AB) ağ modeli ve 4 çıkış (alıcı kanal kullanım durumu, verici kanal kullanım durumu, alınan veri trafiği ve gönderilen veri trafiği) 4 + 11 + 13 + 4 olarak modellenmiştir. MATLAB programı tarafından elde edilen YSA modelinin ağ yapısı Şekil 3.7'de gösterilmiştir.



Şekil 3.7. Geliştirilmiş 4 katmanlı MLP (Multilayer Perceptron) ile YSA model mimarisi

Çizelge 3.5'te verilen çoklu regresyon analizi sonuçlarına göre, $R^2 = 0,999999999$, RMS (Root Mean Square) = 3,02E-05 ve MEP% = 0,003845 olarak hesaplanmıştır.

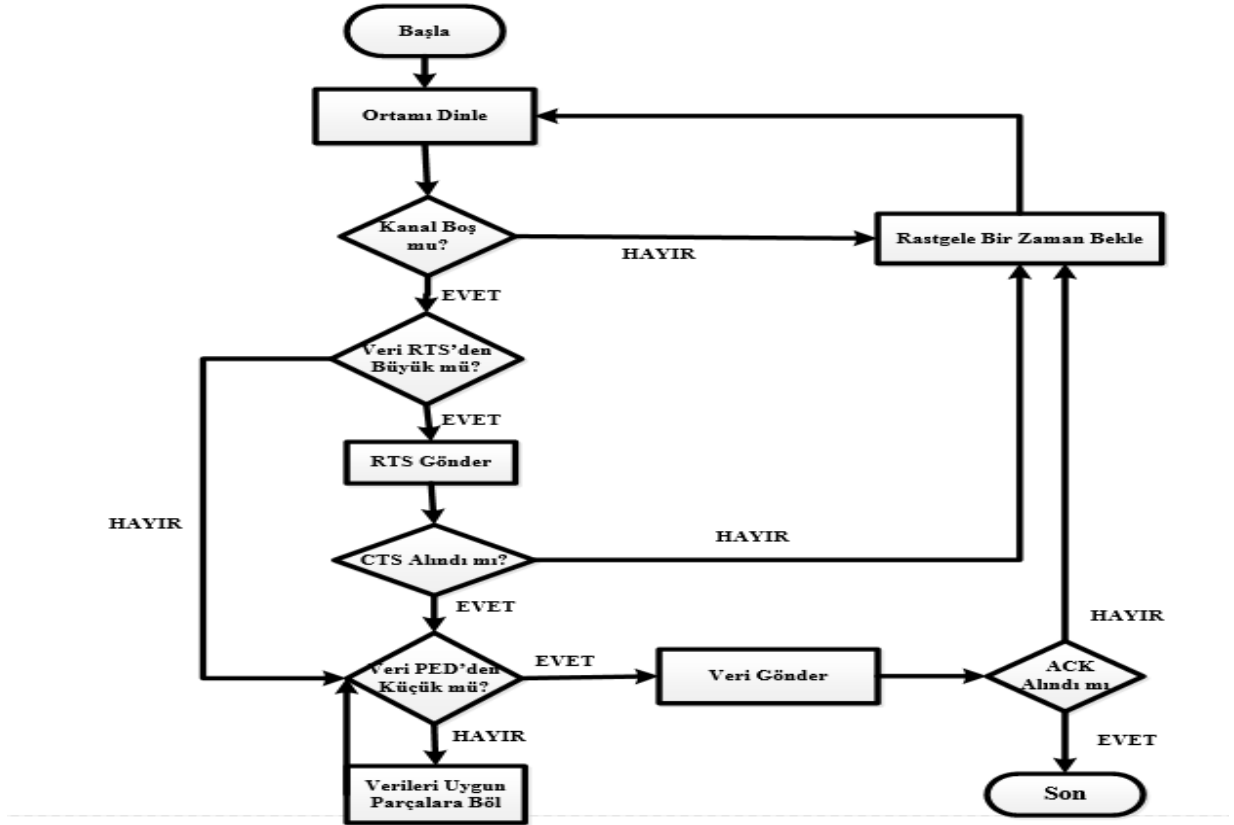
Çizelge 3.5. YSA regresyon analizi ve veri karşılaştırması

Alınan Veri Trafîği				Gönderilen Veri Trafîği			
	Ortalama Hata	RMS	R ²		Ortalama Hata	RMS	R ²
		1,9E-05	1		MEP%	3,02E-05	1
YSA	0,001829	1,15E-06	1830,976	YSA	0,003845	2,92E-06	2114,305
5,477266	0,000289	2,51E-10	30,00044	6,483002	0,006194	1,61E-07	42,02931
0,010412	0,006329	4,34E-13	0,000108	11,1	0,001778	3,9E-08	123,2099
10,29791	0,000267	7,56E-10	106,0469	10,10003	0,002241	5,12E-08	102,0105
5,03088	0,003572	3,23E-08	25,30975	5,090023	0,003471	3,12E-08	25,90834
9,222288	8,76E-05	6,53E-11	85,0506	10,28693	0,004192	1,86E-07	105,821
9,304329	0,000119	1,22E-10	86,57054	8,906003	0,005646	2,53E-07	79,31689
4,647311	0,003682	2,93E-08	21,5975	5,149988	0,007536	1,51E-07	26,52238
8,82345	0,000336	8,76E-10	77,85326	8,493972	0,00556	2,23E-07	72,14756
9,227637	0,002246	4,29E-08	85,14929	10,22596	0,003564	1,33E-07	104,5703
5,641521	0,002811	2,52E-08	31,82676	5,719974	0,000452	6,69E-10	32,7181
10,79521	0,00053	3,27E-09	116,5366	11,75604	0,003086	1,32E-07	138,2044
10,50163	0,000617	4,2E-09	110,2841	11,48803	0,002033	5,46E-08	131,9749
5,08494	0,000205	1,09E-10	25,85662	5,446034	0,0043	5,48E-08	29,65929
9,304661	0,000333	9,61E-10	86,57672	10,7929	0,003748	1,64E-07	116,4866
9,42384	0,00021	3,9E-10	88,80876	10,20193	0,001617	2,72E-08	104,0795
4,529689	0,007076	1,03E-07	20,51809	4,751977	0,005828	7,67E-08	22,58128
8,696149	0,000477	1,72E-09	75,623	10,04511	0,004099	1,7E-07	100,9043
9,313567	0,007979	5,52E-07	86,74253	9,747216	0,000866	7,12E-09	95,00821
5,681064	0,000994	3,19E-09	32,27448	6,553091	0,005963	1,53E-07	42,943
10,65263	0,000504	2,88E-09	113,4786	9,506973	0,003919	1,39E-07	90,38253
10,80481	0,000341	1,36E-09	116,7439	11,09601	0,000134	2,22E-10	123,1215
4,996021	0,00182	8,27E-09	24,96023	5,504967	0,003041	2,8E-08	30,30467
9,765536	0,000555	2,93E-09	95,36569	8,844043	0,005012	1,96E-07	78,2171
9,735482	0,000286	7,76E-10	94,77961	9,900987	0,002152	4,54E-08	98,02954
4,669832	0,001115	2,71E-09	21,80733	4,467008	0,011379	2,58E-07	19,95416
9,438292	0,000236	4,95E-10	89,08136	8,953984	0,002058	3,39E-08	80,17383
9,062757	0,00637	3,33E-07	82,13357	9,900811	0,003933	1,52E-07	98,02605

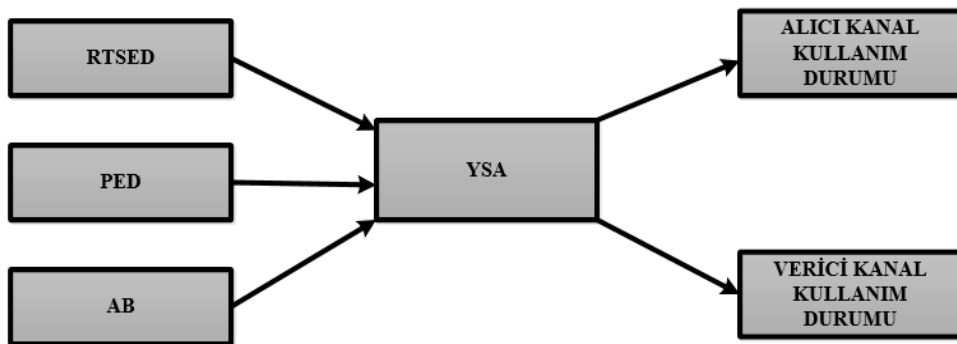
3.2.2. Alıcı ve verici kanal kullanımı için YSA model çalışmaları

CSMA/CA protokolünün aktif olabilmesi için DCF kontrol fonksiyonunda verinin RTSED değerinin üzerinde bir büyüklükte olması gerekmektedir. CSMA/CA’da CTS sinyali alındıktan sonra kontrol edilen PED değerine göre verilerin iletilmesi sağlanmaktadır. PED değerinden büyük değerler gelmesi halinde veriler uygun parçalara bölünmektedir. Veriler parçalara ayrıldıktan sonra veri akışı sağlanmaktadır. Algoritmanın akış şeması Şekil 3.8’de gösterilmiştir.

YSA modelinde CSMA/CA protokolünde veri akışı esnasında RTSED, PED ve AB parametrelerinin alıcı ve gönderici kanal kullanım durumuna etkisinin belirlenmesi için kullanılan model Şekil 3,9’da verilmiştir.



Şekil 3.8. RTS/CTS temel erişim metotları



Şekil 3.9. Giriş ve çıkış parametrelerinin şematik gösterimi

YSA modelinde çıkış parametreleri olarak kullanılacak olan alıcı ve gönderici kanal kullanım durumlarının maximum olması hedeflenmektedir. YSA modelinde giriş parametrelerinde RTSED ve PED değerleri 600 bayt, 700 bayt, 900 bayt, 1000 bayt ve 1200 bayt seçilirken, AB ise 9000 bit, 10 000 bit, 12 000 bit, 13 000 bit ve 15 000 bit seçilmiştir. Alıcı kanal kullanımı; RTSED ve PED 600 bayt ile 1200 bayt arasında azalmakta, AB 9000 bit ile 13 000 bit arasında iken artmakta ve 13 000 bit ile 15 000 bit arasında azalmaktadır. Gönderici kanal kullanımı; RTSED 600 bayt 1000 bayt arasında azalmakta, 1000 bayt ile 1200 bayt arasında artmakta, PED 600 bayt ile 1200 bayt arasında azalmakta ve AB 9000 bit ile 12 000 bit arasında artmakta, 12 000 bit ile 15 000 bit arasında ise azalmaktadır.

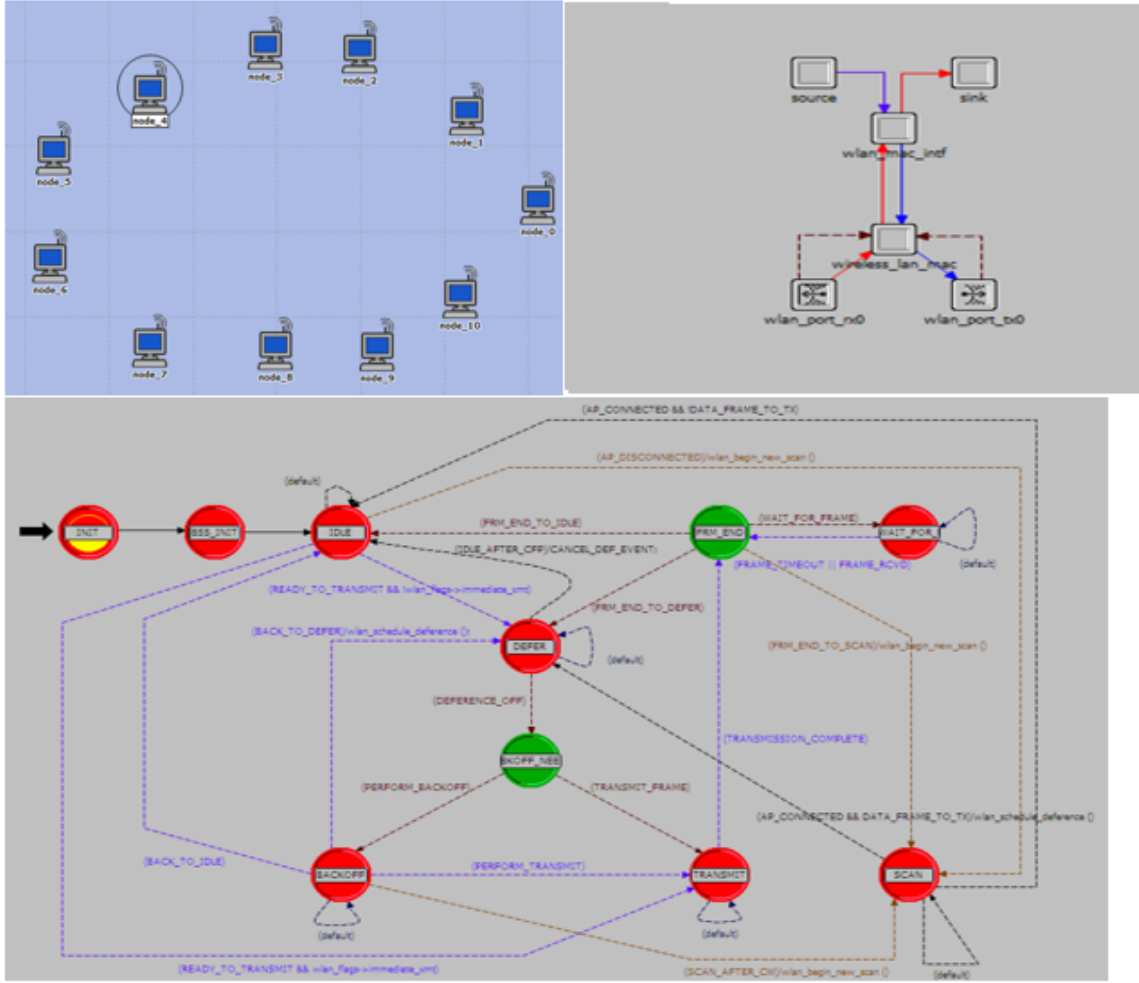
Geliştirilen model için elde edilen sonuçlara göre alınan ve gönderilen veri trafiği tahminlemesi optimize edilmiştir. Elde edilen sonuçlar bölüm 4'te klasik yöntem test sonuçları ile karşılaştırılmış ve performans iyileştirmeleri analiz edilmiştir.

3.3. Geri Besleme Kontrol Tabanlı RTSED, PED ve AB Yöntem Analizleri ile Performans Geliştirme

Yapılan çalışmalarda veri setleri ile bulanık mantık tabanlı ve YSA tabanlı çeşitli algoritmalar kullanılarak RTSED, PED ve AB giriş parametrelerinin modellenmesi gerçekleştirilmiştir. Bu bölümde ise algoritma ve yöntemlerin gerçek zamanlı olarak Riverbed Modeler'e gömülü olarak aktarılması ve dinamik bir yapı oluşturulması sağlanmıştır. Dinamik modelde geri besleme kontrol metodu, döngü tabanlı kaba kuvvet ve bulanık mantık algoritmalarında kullanılmıştır.

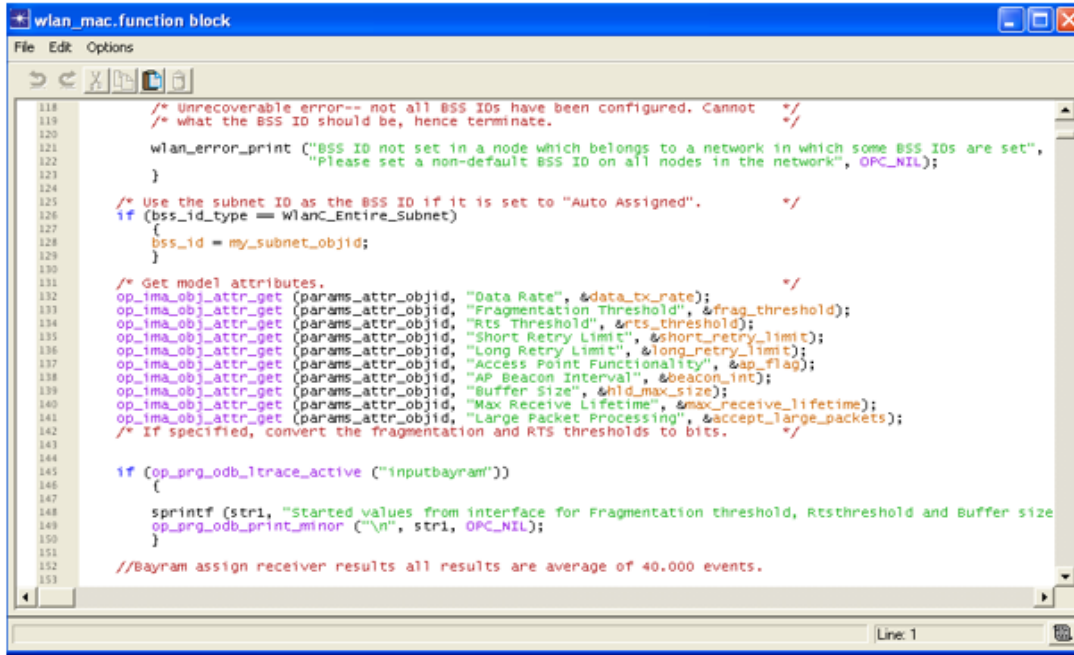
3.3.1. Geri besleme kontrolü ile kanal kullanımının benzetim anında döngü tablosu ve kaba kuvvet algoritması ile optimize edilmesi

Riverbed Modeler'de Şekil 3.10'da gösterildiği gibi üç katman bulunmaktadır. En üstte ağ katmanı, ortada düğüm katmanı ve en altta ise proses katmanı bulunmaktadır.



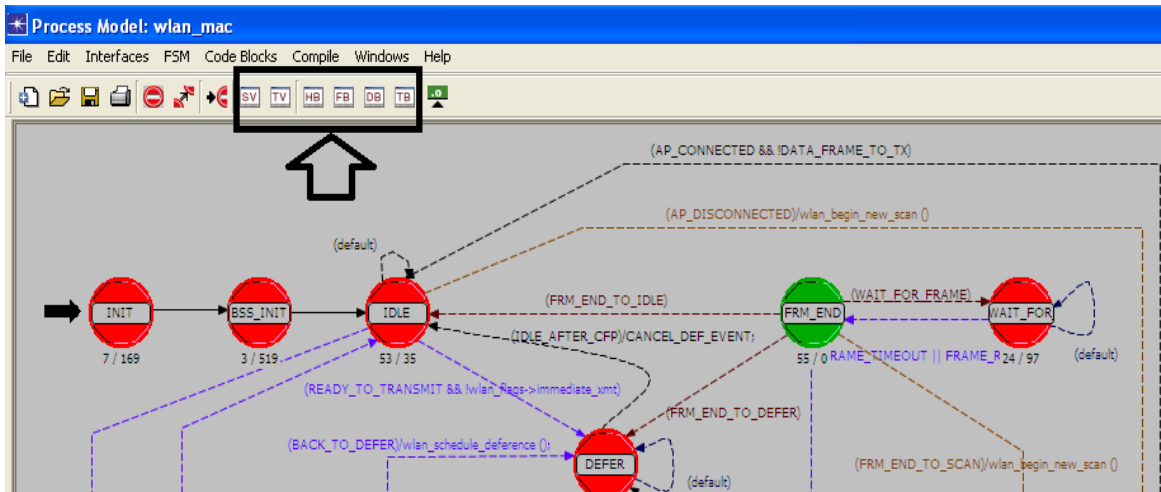
Şekil 3.10. Riverbed Modeler model yapısı

Ağ katmanında düğümler yerleştirilip konfigürasyonlar koşulurken; düğüm modeli katmanında OSI modelinde yer alan 7 ara katman yer almaktadır. Katmanlar yukarıdan aşağıya doğru uygulama, sunum, oturum, taşıma, ağ, veri link ve fiziksel katman olarak adlandırılmıştır. Veri link katmanında alt katman olan MAC katmanından proses model katmanına eriştiğimizde döngüler içerisinde prosesin işleme yöntemi (proses model katmanı) yer almaktadır. Modeller ve katmanlar göz önünde bulundurulduğunda geri besleme için kullanılan kod blok yapısında proses katmanında her prosesin içerisinde döngüler ve fonksiyonlar bulunmaktadır. Riverbed Modeler, Şekil 3.11’de gibi fonksiyonlarda kullanıcılara geliştirme yapma seçeneği sunmaktadır.

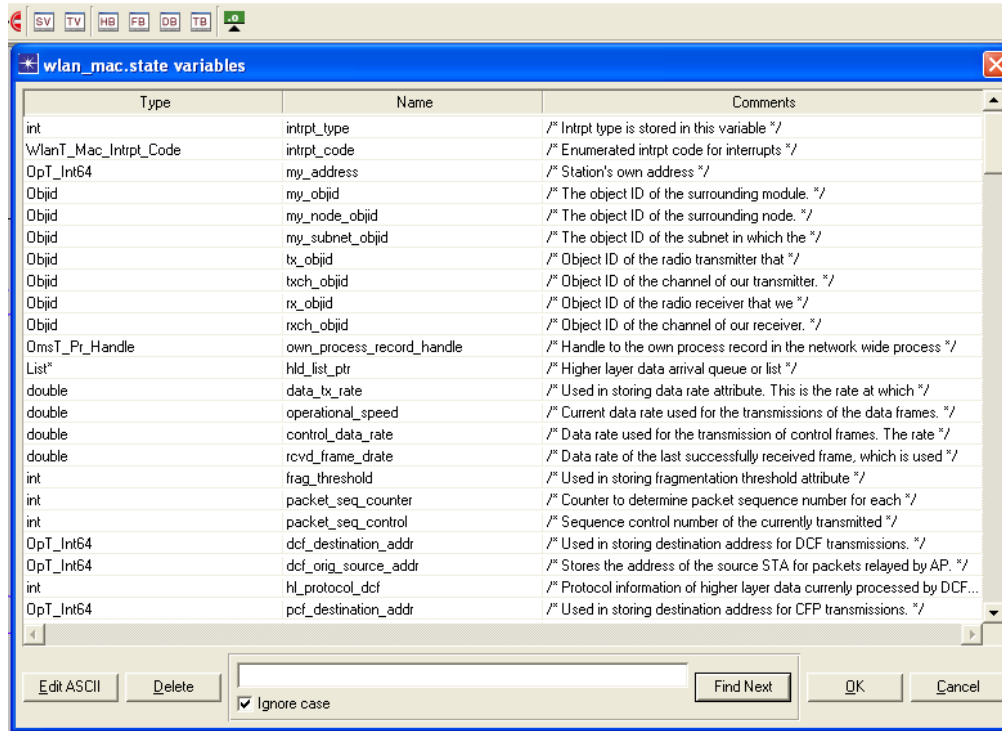


Şekil 3.11. Riverbed Modeler proses model kodlama sınıfları

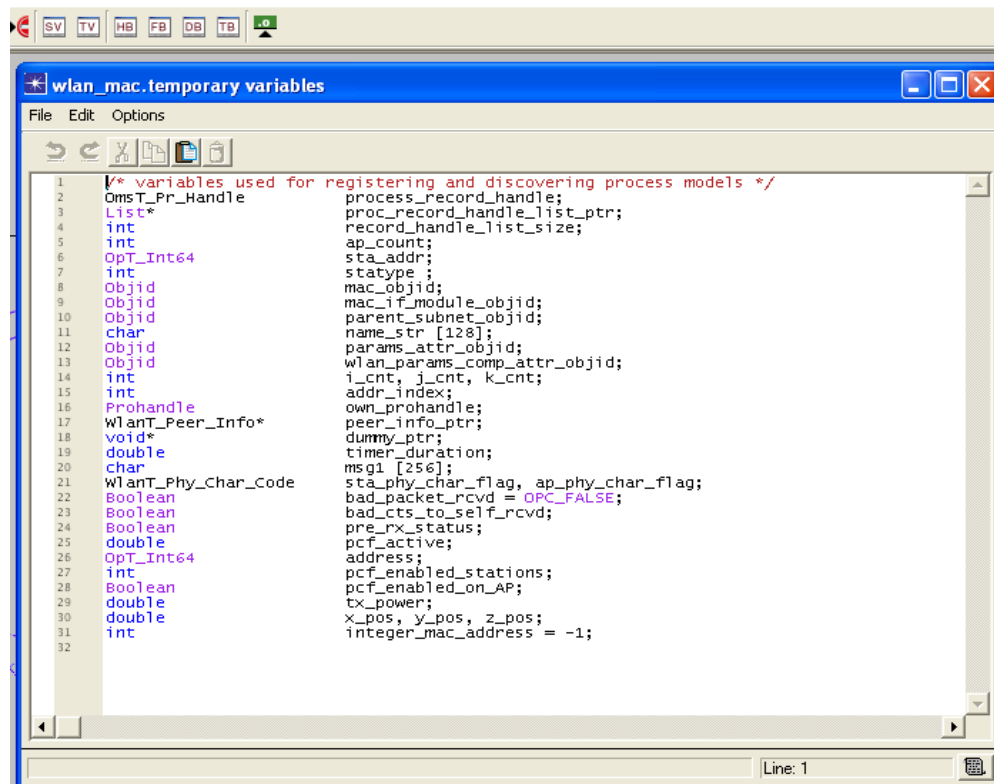
Şekil 3.12’de yer alan hazır fonksiyonlarla birçok protokol farklı sınıf tanımlamaları ile koşulmaktadır. Şekil 3.13’te yer alan durum değişkeni (State Variable-SV) alanında küresel değişkenler tanımlanmaktadır. Şekil 3.14’te yer alan geçici değişken (Temporary Variable-TV) alanında ise geçici değişkenler tanımlanmaktadır.



Şekil 3.12. Riverbed Modeler sınıf tanımlamaları

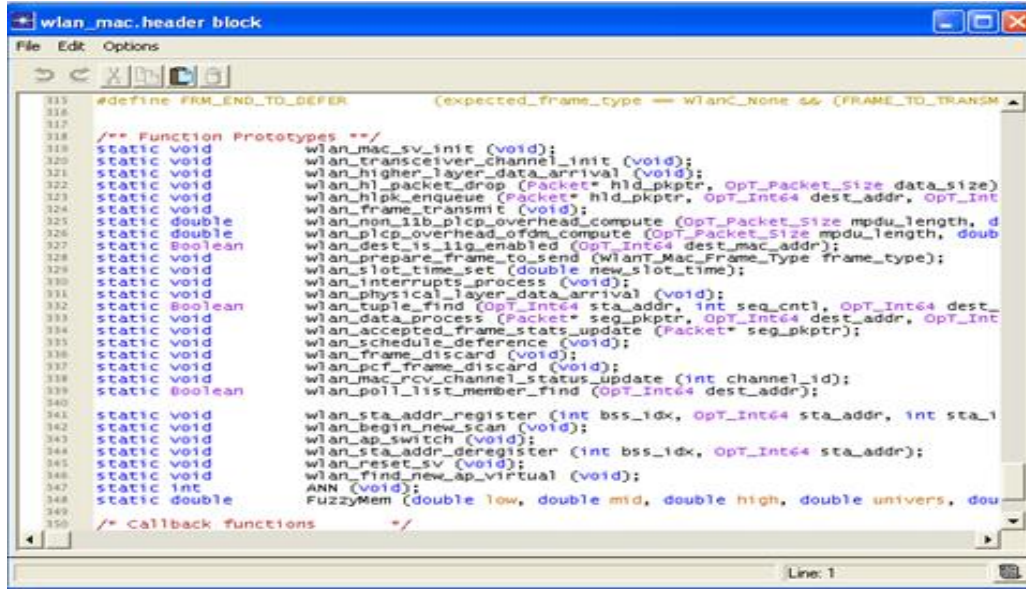


Şekil 3.13. Durum değişkeni (SV)



Şekil 3.14. Geçici değişken (TV)

Kullanılacak olan sınıflar ve fonksiyonlar başlık bloğunda (Header Block-HB) Şekil 3.15'te gösterildiği gibi, fonksiyonların implemantasyonu ise fonksiyonel blokta (Functional Block-FB) Şekil 3.16'da gösterildiği gibi yapılmaktadır. Teşhis bloğunda (Diagnostic Block-DB) hata mesajları kontrol edilmektedir. Sonlandırma bloğunda (Terminates Block-TB) ise program sonlandırılmaktadır. *Wlan_mac.pr.m* bütün blokların yer aldığı kod yapısını içermektedir.

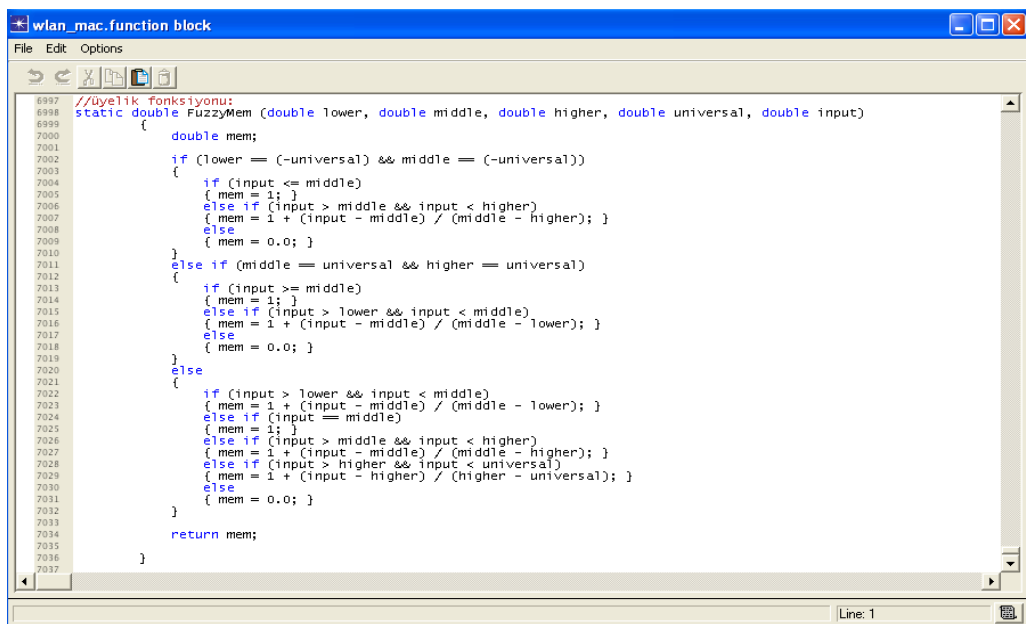


```

wlan_mac.header block
File Edit Options

315 #define FRM_END_TO_DEFER (expected_frame_type == wlanC_None && (FRAME_TO_TRANSM
316
317 /** Function Prototypes */
318 static void wlan_mac_sv_init (void);
319 static void wlan_transceiver_channel_init (void);
320 static void wlan_higher_layer_data_arrival (void);
321 static void wlan_hl_packet_drop (Packet* hld_pkptr, Opt_Packet_Size data_size)
322 static void wlan_hl_pk_enqueue (Packet* hld_pkptr, Opt_Int64 dest_addr, Opt_Int
323 static void wlan_frame_transmit (void);
324 static double wlan_non_11b_plcp_overhead_compute (Opt_Packet_Size mpdu_length, d
325 static double wlan_plcp_overhead_ofdm_compute (Opt_Packet_Size mpdu_length, doub
326 static void wlan_dest_is_11g_enabled (Opt_Int64 dest_mac_addr);
327 static void wlan_prepare_frame_to_send (WlanT_Mac_Frame_Type frame_type);
328 static void wlan_slot_time_set (double new_slot_time);
329 static void wlan_interrupts_process (void);
330 static void wlan_tuple_find (Opt_Int64 sta_addr, int seq_cnt1, Opt_Int64 dest_
331 static Boolean wlan_data_process (Packet* seg_pkptr, Opt_Int64 dest_addr, Opt_Int
332 static void wlan_accepted_frame_stats_update (Packet* seg_pkptr);
333 static void wlan_schedule_deference (void);
334 static void wlan_frame_discard (void);
335 static void wlan_pcf_frame_discard (void);
336 static void wlan_mac_rcv_channel_status_update (int channel_id);
337 static Boolean wlan_poll_list_member_find (Opt_Int64 dest_addr);
338
339 wlan_sta_addr_register (int bss_idx, Opt_Int64 sta_addr, int sta_l
340 static void wlan_begin_new_scan (void);
341 static void wlan_ap_switch (void);
342 static void wlan_sta_addr_deregister (int bss_idx, Opt_Int64 sta_addr);
343 static void wlan_reset_sv (void);
344 static void wlan_find_new_ap_virtual (void);
345 static int ANN (void);
346 static double FuzzyMem (double low, double mid, double high, double univers, dou
347
348 /** Callback functions */
349
350
  
```

Şekil 3.15. Başlık bloğu (HB)



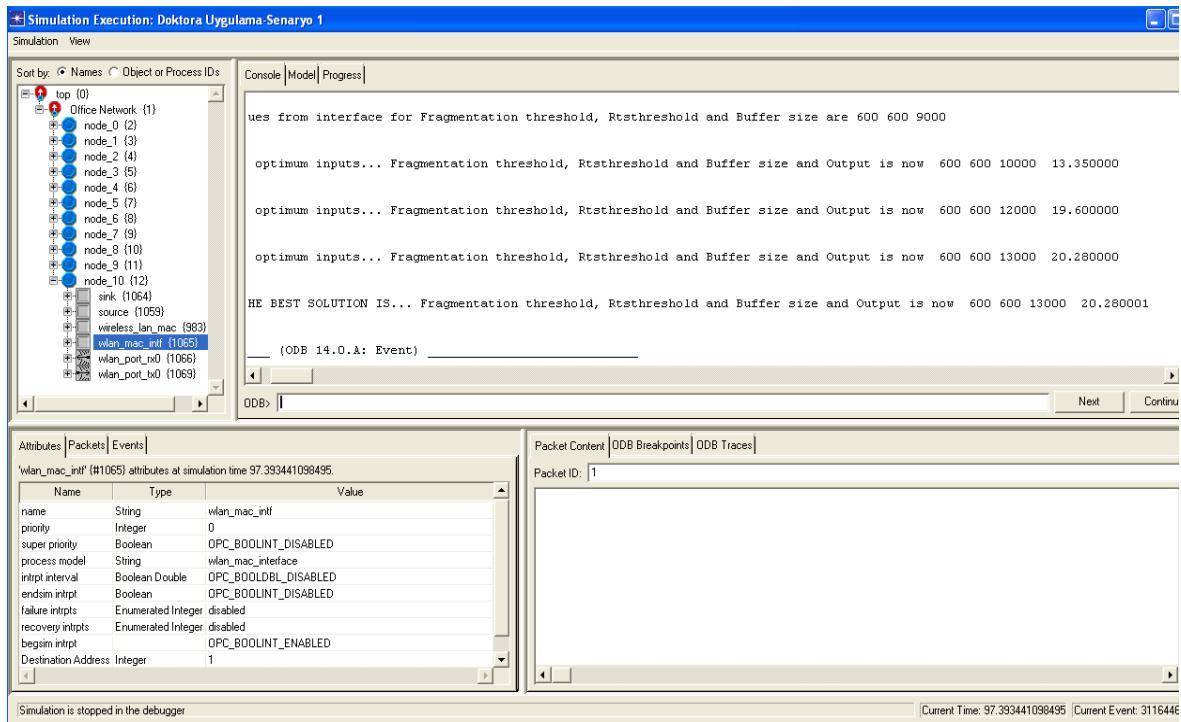
```

wlan_mac.function block
File Edit Options

6997 //Üyelik fonksiyonu:
6998 static double FuzzyMem (double lower, double middle, double higher, double universal, double input)
6999 {
7000     double mem;
7001     if (lower == (-universal) && middle == (-universal))
7002     {
7003         if (input <= middle)
7004         {
7005             mem = 1;
7006         }
7007         else if (input > middle && input < higher)
7008         {
7009             mem = 1 + (input - middle) / (middle - higher);
7010         }
7011         else if (middle == universal && higher == universal)
7012         {
7013             if (input >= middle)
7014             {
7015                 mem = 1;
7016             }
7017             else if (input > lower && input < middle)
7018             {
7019                 mem = 1 + (input - middle) / (middle - lower);
7020             }
7021             else
7022             {
7023                 mem = 0;
7024             }
7025         }
7026         else if (input > lower && input < middle)
7027         {
7028             mem = 1 + (input - middle) / (middle - lower);
7029         }
7030         else if (input == middle)
7031         {
7032             mem = 1;
7033         }
7034         else if (input > middle && input < higher)
7035         {
7036             mem = 1 + (input - middle) / (middle - higher);
7037         }
7038         else if (input > higher && input < universal)
7039         {
7040             mem = 1 + (input - higher) / (higher - universal);
7041         }
7042         else
7043         {
7044             mem = 0;
7045         }
7046     }
7047     return mem;
7048 }
7049
  
```

Şekil 3.16. Fonksiyonel blok (FB)

Riverbed Modeler arayüzlerinde yer alan buton fonksiyonları açık kaynak kodlu değildir. Döngü tablosu kullanılarak gerçekleştirilen uygulamada değişiklikler Şekil 3.17’de verilmiş olan hata ayıklama ekranından takip edilmektedir. Hata ayıklama ekranında giriş değişkenleri basılıp güncellemeler takip edilmektedir. İdeal değer elde edildiğinde ise yine sonuçlar metin tipinde basılmaktadır. Konsol arayüzüne ek olarak model ve uygulama animasyon (animation) ile takip edilebilmektedir. Animasyon ekranında veriler konfigürasyonda yer alan bütün düğümler için aynı şekilde güncellenmekte ve görülmektedir.

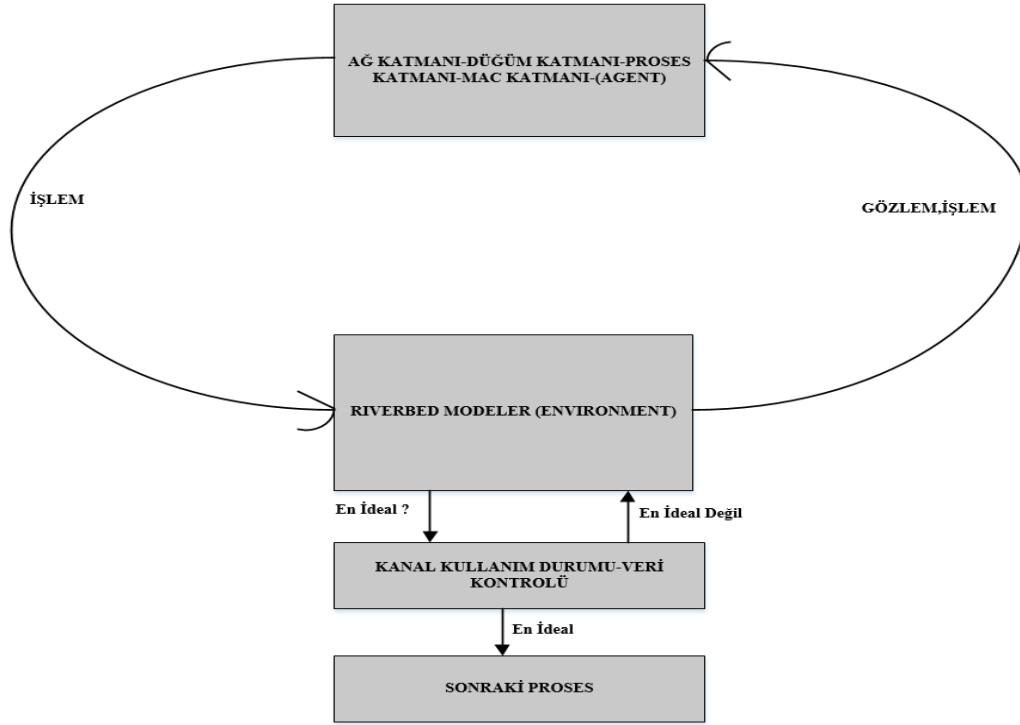


Şekil 3.17. Riverbed Modeler hata ayıklama ekranı

Riverbed Modeler’deki çalışma prensibi göz önünde bulundurularak RTSED, PED ve AB giriş parametrelerinin gerçek zamanlı olarak Riverbed Modeler’de benzetim başlarken kendi kendini güncellemesi amacı ile geri besleme kontrol yöntemi kullanılmıştır. Kanal verimliliğini arttırmak amacı ile dinamik bir yapı kurulmuş ve sistemin ideal giriş parametreleri ile çalışması sağlanmıştır.

Önerilen modelde giriş parametreleri başlangıçta varsayılan (default) değerlerinde iken bu giriş değerleri her seferinde koda gömülü olan çıkış parametresi ile kontrol edilmekte ve ideal değerinde değil ise döngüyü tekrar etmektedir. Örneğin başlangıçta yer alan değerler

önce ağ katmanı sonra düğüm katmanı ve en son da proses model katmanında ilgili kod alanında kontrol çıkış parametreleri ile karşılaştırmalı olarak ajan kullanarak kontrol edilmektedir. Kullanılan değer ideal değil ise Riverbed Modeler prosesinde (environment) döngü tekrar ajan ile sürmektedir. Bütün değerler ideal hale geldiğinde Şekil 3.18’de gösterildiği gibi proses döngüden çıkıp sonraki prosese devam etmektedir.



Şekil 3.18. Riverbed Modeler döngü tablosu ajan yapısı

Çıkış parametresi Riverbed Modeler’e ortalama 18 900 344 event ve 125 farklı giriş değerlerinde toplam ortalama olay sayısı 2 362 543 000 sonuç ile ilgili diziye aktarılmış ve döngüsel geri beslemeye sokulmuştur. Burada yer alan kanal kullanım durumu alıcı kanal kullanım durumunu ifade etmektedir ve veriler Şekil 3.18’de dizine hazır olarak aktarılmıştır. Riverbed Modeler’in yapısı gereği giriş değişkenleri ancak 1. proses (init proses) üzerinden değiştirilebilmektedir. Bu nedenle modelimizde veri kontrollerinden sonra (döngü sonrası) ve ideal giriş değerleri ile sonraki proses işletilmektedir.

Yapılan çalışmamızda giriş değişkenlerinden RTSED, PED ve AB’nin benzetim anında koşumu kontrol edilerek çalışma esnasında güncellenmesi sağlanmıştır. Bu amaçla servis kalitesini direkt etkileyen birim zamanda işlenen paket miktarı üzerinden çıkış değerleri kaba kuvvet algoritması ile kontrol edilerek benzetimde bir geri besleme sistemi

oluşturulmuştur. Kod blok yapısında *wlan_mac.pr.m* üzerinde yapılan değişiklikler ile kod yapısı güncellenmiştir. İlk olarak giriş değerlerinde kullanılacak olan değerler durum değişken (state variables) alanında tanımlanmış ve fonksiyon blok (Functional Blok) alanında bulunan *wlan_mac_sv_init ()* fonksiyonu kullanılarak ara girdilere ilk değerler atanmıştır. Şekil 3.19’da gösterilen FB’de yer alan *wlan_accepted_frame_stats_update ()* fonksiyonu kullanılarak *pk_size* parametre girişi üzerinden birim zamanda işlenen paket miktarının her bir saniyede giriş değerlerini kullanılarak güncellenmesi sağlanmıştır.

```
static void
wlan_accepted_frame_stats_update (Packet* seg_pkptr)
{
    double ete_delay, pk_size;
    char str1[256], str2[256];

    /** This function is called just before a frame received from **/
    /** physical layer being forwarded to the higher layer to **/
    /** update end-to-end delay and throughput statistics. **/
    FIN (wlan_accepted_frame_stats_update (seg_pkptr));

    /* The packet has been accepted, tag the packet for aptrack receive */
    aptrack_rx_record (seg_pkptr);

    /* Total number of bits sent to higher layer is equivalent to a */
    /* throughput. */
    pk_size = (double) op_pk_total_size_get (seg_pkptr);

    //Bayram Coding lines if the throughput is bigger then previous it will change the intial values with in below inputs and will use this.
    if(pk_size>prv_pk_size)
    {
        if(op_prg_odb_ltrace_active ("parameters"))
        {
            sprintf (str1, "Parameters: Frag. Threshold %u, Rts Threshold %u, buffer size %u.", (frag_threshold/8), (rts_threshold/8), hld_max_size); //Divided by 8 is fo
            sprintf (str2, "Searching for throughput: Current %f, Previous %f.", pk_size, prv_pk_size);
            op_prg_odb_print_minor (str1, OPC_NIL);
            op_prg_odb_print_minor (str2, OPC_NIL);
        }
        prv_pk_size = pk_size;
        best_rts_threshold = rts_threshold;
        best_frag_threshold = frag_threshold;
        best_hld_max_size = hld_max_size;
    }
}
```

Şekil 3.19. Wlan_accepted_frame_stats_update () fonksiyon yapısı

Modelde giriş parametreleri başlangıçta varsayılan değerlerinde iken; giriş değerleri her seferinde koda gömülü olan birim zamanda işlenen paket miktarı ve kanal kullanm durumu ile kontrol edilmekte ve ideal değerinde değil ise döngüyü tekrar etmektedir. Döngü her giriş değerinde bir saniye beklemektedir. Bir saniye içerisinde giriş değişkenlerine o aşamada dış etkenden bağımsız bir şekilde sistemi kontrol edebilme mekanizması sağlanmaktadır. Örneğin başlangıçta yer alan değerler önce ağ katmanını sonra düğüm katmanını ve daha sonra da proses model katmanında ilgili kod alanında kontrol çıkış parametreleri ile karşılaştırmalı olarak kontrol edilmekte ve eğer değer ideal değil ise Riverbed Modeler prosesinde tekrar ajan ile döngü sürmektedir. Bütün değerler ideal hale geldiğinde proses döngüden çıkıp ideal değerini alarak benzetime devam etmektedir.

Riverbed Modeler ağ katmanında, kullanıcıya hata ayıklama ekranından takip değişiklikleri sunulmaktadır. Hata ayıklama ekranında ara giriş değişkenleri basılıp, konsol üzerinden takip yapılabilir. İdeal değerler elde edildiğinde ise yine ilgili ekrana basılmaktadır.

Örnek bir konsol yapısında çıktılar şu şekilde olmaktadır:

```
*Time : 1,393990274014985 sec, [23m 13s . 990ms 274us 014ns 985ps]
*Event : execution ID (25 029 225), schedule ID (#25966024), type (stream intrpt)
*Source: execution ID (25 029 223), top.Office Network.node_4.wlan_port_rx0
         [Objid=484] (radio receiver)
*Data : instrm (0), packet ID (5295844)
*Module : top.Office Network.node_4.wireless_lan_mac [Objid=401] (processor)
          *Best Parameters: Frag. Threshold 959, Rts Threshold 959, buffer size 12470.
          * Current tested parameters: Frag. Threshold 1352, Rts Threshold
          1352, buffer size 16400.
```

Burada *time* geçen süreyi ifade ederken, gerçekleşen olay sayısı ise *event* olarak verilmektedir. Düğüm ismi *module* alanına basılmaktadır. En altta bulunan alanlar ise *string* alanına yazmış olduğumuz *text* olup, *so far* ise mevcut durumda daha ideal bir birim zamanda işlenen paket miktarı verisi olduğunu ifade etmektedir. *Search*, yeni ve daha ideal bir değer gelmiş olduğunu ifade etmektedir. Aynı zamanda giriş değişkenleri de saniyelik ve ideal değerleri ile ekrana basılmıştır. Bu çalışma ile elde edilen test sonuçları bölüm 4'te klasik yöntem test sonuçlarıyla karşılaştırılmış ve performans iyileştirmeleri sunulmuştur.

3.3.2. Geri besleme kontrolünün zaman ayarlı Op_sim_time fonksiyonu ile uygulanması

Giriş parametreleri incelendiğinde RTSED ve PED değerleri için evrensel küme 512 bayt ile 2048 bayt arasında seçilmiştir. AB için ise evrensel küme 8000 bit ile 24 000 bit arasında seçilmiştir. AB değerleri 10'ar bit arttırılarak bütün ara değerlerin test edilmesi sağlanmıştır. Veriler test edilirken her değer için 1'er saniye beklenilmesi durumunda daha net sonuçların elde edildiği ortaya çıkmıştır. Değerler 1'er saniye olarak seçilmediği durumda henüz yeterli sürede ve seviyede test yapılmadan bir sonraki giriş değeri gelmekte ve güncelleme yapılmaktadır. Bu durumda 1'er saniye beklenilmesi için Şekil 3.20'de

verilen *Op_sim_time* zamanlama fonksiyonu kullanılmıştır. Zamanlama fonksiyonunda benzetimde gerçekleştirilen herhangi bir uygulama geciktirilebilir veya özel bir tetikleme yöntemi ile zaman ayarlı uygulamalar tanımlanabilmektedir.

```
//Bayram Coding lines if the throughput is bigger then previous it will change the intial values with in below inputs and will use this.
if(pk_size>prv_pk_size)
{
    if(op_prg_odb_trace_active ("parameters"))
    {
        sprintf (str1, "Parameters: Frag. Threshold %u, Rts Threshold %u, buffer size %u.", (frag_threshold/8), (rts_threshold/8), hld_max_size); //Divided by 8 is for
        sprintf (str2, "Searching for throughput: Current %f, Previous %f.", pk_size, prv_pk_size);
        op_prg_odb_print_minor (str1, OPC_NIL);
        op_prg_odb_print_minor (str2, OPC_NIL);
    }
    prv_pk_size = pk_size;
    best_rts_threshold = rts_threshold;
    best_frag_threshold = frag_threshold;
    best_hld_max_size = hld_max_size;

    if ((op_sim_time() - prv_sim_time)*1000 >= 1000) //Divided by 1000 for milliseconds and until 1 seconds waited to work without changing the initial 3 values.
    {
        if (frag_threshold<16389)
        {
            frag_threshold += 8; //frag thresold will increase by 1
        }

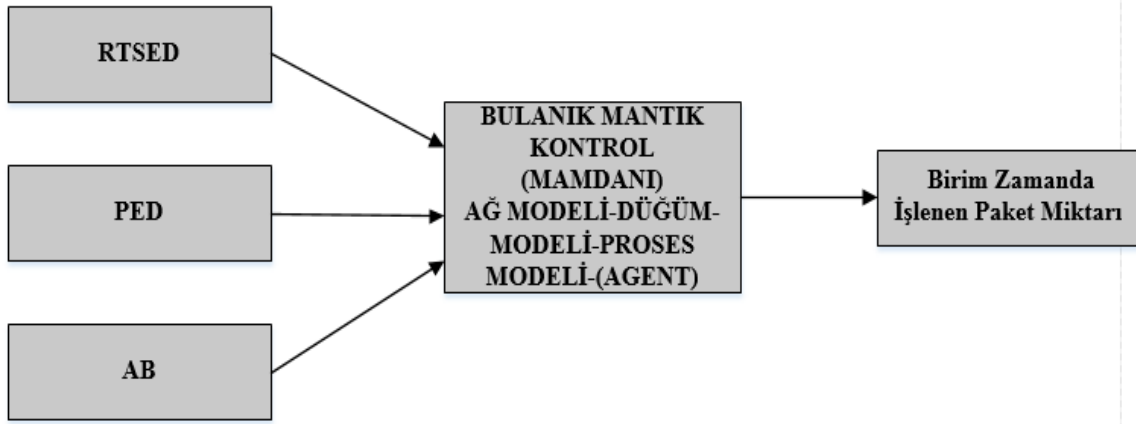
        if (rts_threshold<16389) //rts thresold will increase by 1
        {
            rts_threshold += 8;
        }

        if (hld_max_size<24000) //buffer size will increase by 10
        {
            hld_max_size += 10;
        }
        prv_sim_time = op_sim_time(); //to go 1 seconds to front.
    }
}
```

Şekil 3.20. Op_sim_time fonksiyon kullanımı

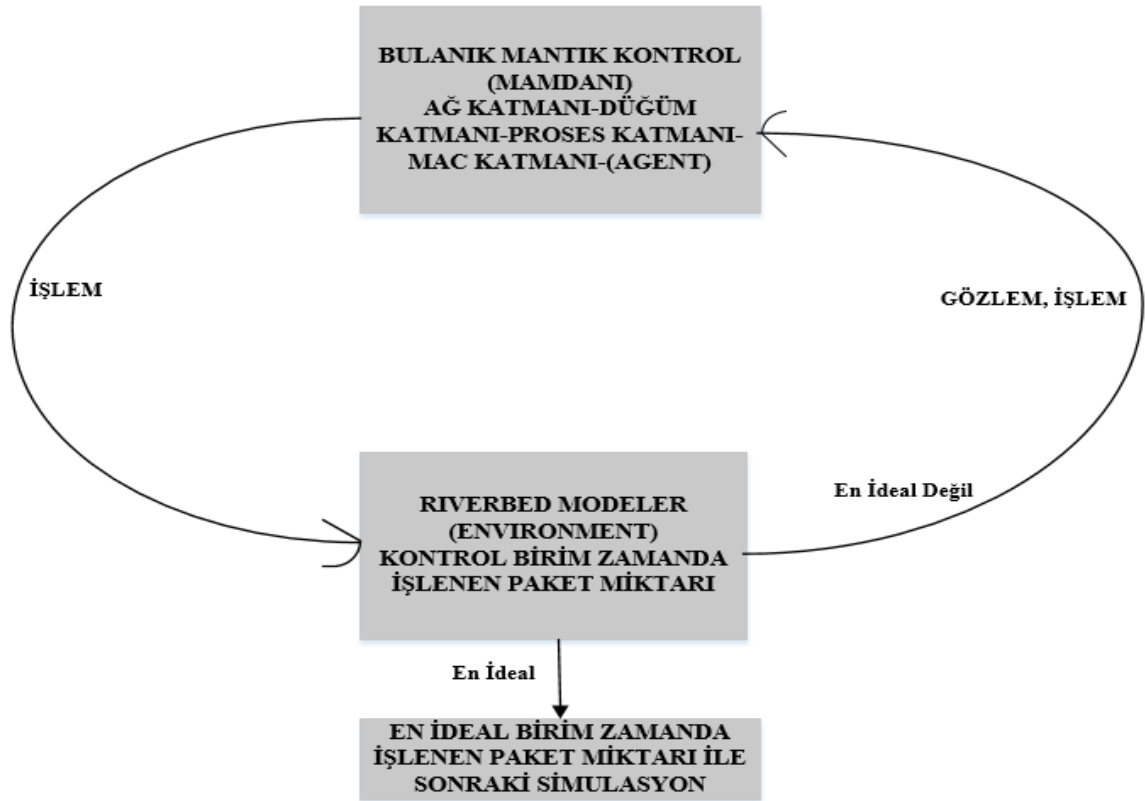
3.4. Bulanık Mantık Tabanlı Geri Besleme Kontrolü ile Düğüm Bazlı Birim Zamanda İşlenen Paket Miktarının Optimize Edilmesi

Bu bölümde benzetim esnasında düğüm bazlı birim zamanda işlenen paket miktarının arttırılması için bulanık mantık tabanlı geri besleme kontrollü uygulama gerçekleştirilmiştir. Kullanılan mamdani bulanık mantık tabanlı sistemde bulanıklaştırma ünitesi, çıkarım motoru ve durulaştırma ünitesi kullanılmıştır. Şekil 3.21’de gösterildiği gibi Mamdani bulanık mantık sisteminde giriş parametreleri olarak RTSED, PED ve AB değerleri kullanılmıştır. Çıkış parametresinde ise düğüm bazlı birim zamanda işlenen paket miktarı kullanılmıştır.



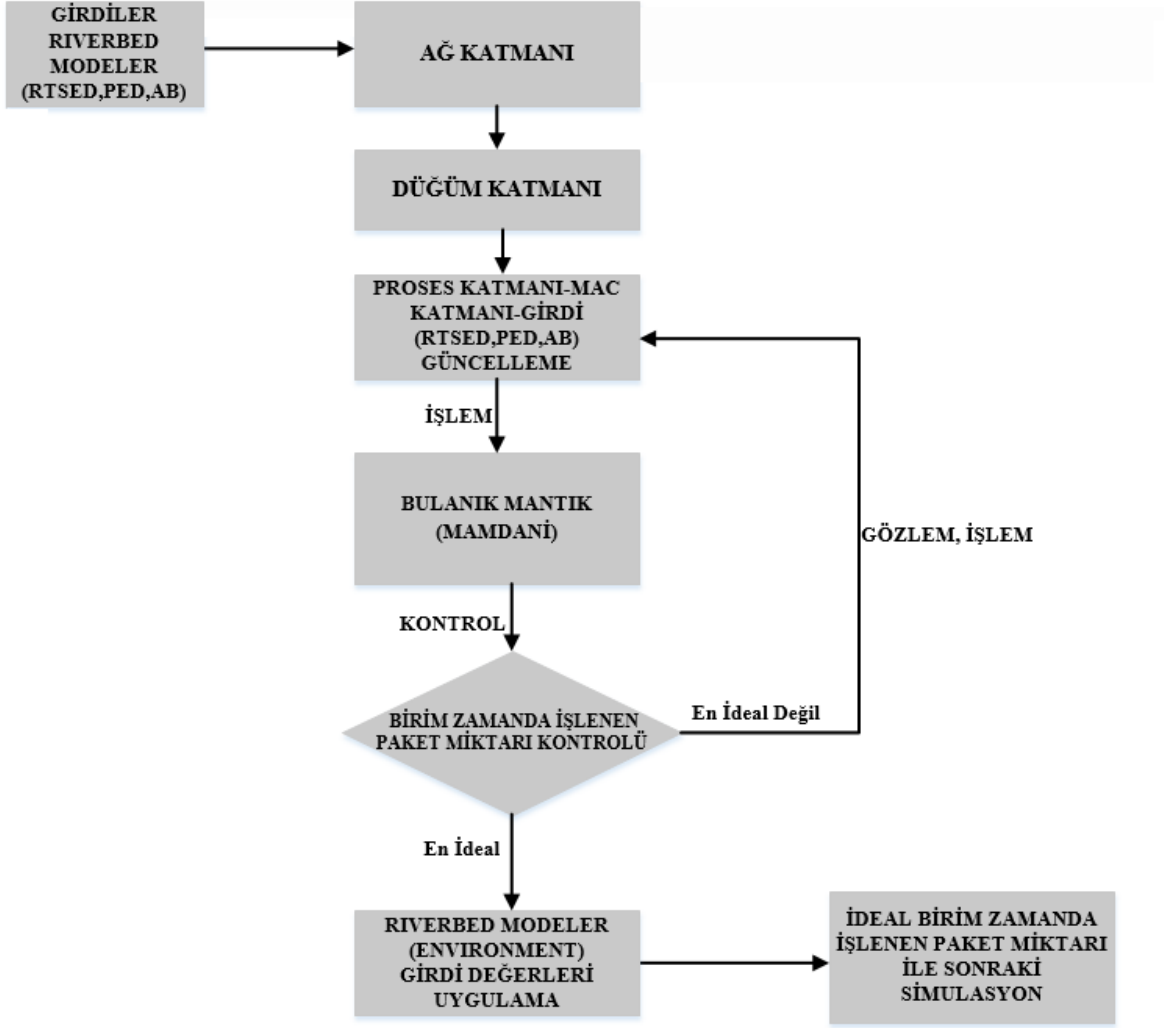
Şekil 3.21. Bulanık mantık sisteminin şematik gösterimi

Yapılan çalışmamızda giriş değişkenlerinden RTSED, PED ve AB için çıkış değerlerinin benzetim anında koşumu kontrol edilerek, benzetimin koşturulduğu esnada güncellenmesi amaçlanmıştır. Çalışmada servis kalitesini direkt etkileyen birim zamanda işlenen paket miktarı üzerinden çıkış değerleri kontrol edilerek, benzetimde bir geri besleme sistemi oluşturulmuştur. Kod blok yapısında *wlan_mac.pr.m* üzerinde yapılan değişiklikler ile kod yapısı güncellenmiştir. İlk olarak giriş değerlerinde kullanılacak olan değerler SV alanında tanımlanmış ve FB alanında bulunan *wlan_mac_sv_init ()* fonksiyonu kullanılarak ara girdilere ilk değerleri atanmıştır. FB’de yer alan *wlan_accepted_frame_stats_update ()* fonksiyonu kullanılarak *pk_size* parametre girişi üzerinden *throughput’un* her bir saniye de giriş değerleri kullanılarak güncellenmesi sağlanmıştır. Sistemde yer alan geri besleme kontrolü Şekil 3.22’de ki gibi modellenmiş ve bulanık mantık algoritması uygulanmıştır. Modelde giriş parametreleri başlangıçta varsayılan değerlerinde iken; giriş değerleri her seferinde koda gömülü olan birim zamanda işlenen paket miktarı ile kontrol edilmekte ve ideal değerinde değil ise döngüyü tekrar etmektedir. Döngü her giriş değerinde bir saniye beklemektedir. Burada bir saniye ile amaç giriş değişkenlerine o aşamada belirli bir süre ve dış etkenden bağımsız bir şekilde kontrol edebilme mekanizmasını sağlamaktır. Riverbed Modeler prosesindeki ajan ile döngü sağlanmaktadır. Bütün değerler bulanık mantık tabanlı olarak ideal hale geldiğinde, proses döngüden çıkıp ideal giriş değerleri ile testlerine devam etmektedir.



Şekil 3.22. Bulanık mantık tabanlı geri besleme kontrolü şematik gösterimi

Tez çalışmasında gerçekleştirilen modelin genel akış diyagramı Şekil 3.23'te yer almaktadır.



Şekil 3.23. Bulanık mantık tabanlı geri besleme kontrolü genel akış diyagramı

Benzetim koşum sürecini test etmek amacıyla gerçekleştirilen çalışmada; tanımlanan sayaç ile benzetim anında geri beslemeli bulanık mantık algoritmasının toplamda 1258 saniye koştuğu görülmüştür.

3.4.1. Geri besleme kontrolünde bulanık kural tabanının gösterimi

Riverbed Modeler’de yapılan benzetim esnasında elde edilen değerler bulanık kural tabanı oluşturmak için RTSED, PED ve AB seçimlerinde kullanılmıştır. Bulanık mantık sisteminde Çizelge 3.6’da kullanılan üç farklı değişkeni girdi olarak, birim zamanda işlenen paket miktarı ise çıktı olarak kullanılıp, test çalışmaları ile değerlerin üyelik fonksiyonları belirlenmiş ve kümeler oluşturulmuştur. Kural tabanında kullanılan küçük (KÜ), büyük

(BÜ), Normal (NO), çok-az (ÇA), az (A), fazla (F), çok-fazla (ÇF) değerlerinde 27 farklı kural ile sözel sınıflar oluşturulmuştur. Kural tabanı;

[EĞER RTSED= '...' ve PED= '...' ve AB= '...' İSE gecikme= '...' ve yükleme= '...' ve birim zamanda işlenen paket miktarı= '...'] şeklinde oluşturulmuştur.

Örneğin EĞER RTSED= 'KÜÇÜK' ve PED= 'KÜÇÜK' ve AB= 'AZ' İSE gecikme= 'FAZLA' ve yükleme= 'FAZLA' ve birim zamanda işlenen paket miktarı= 'ÇOK-AZ'dır.

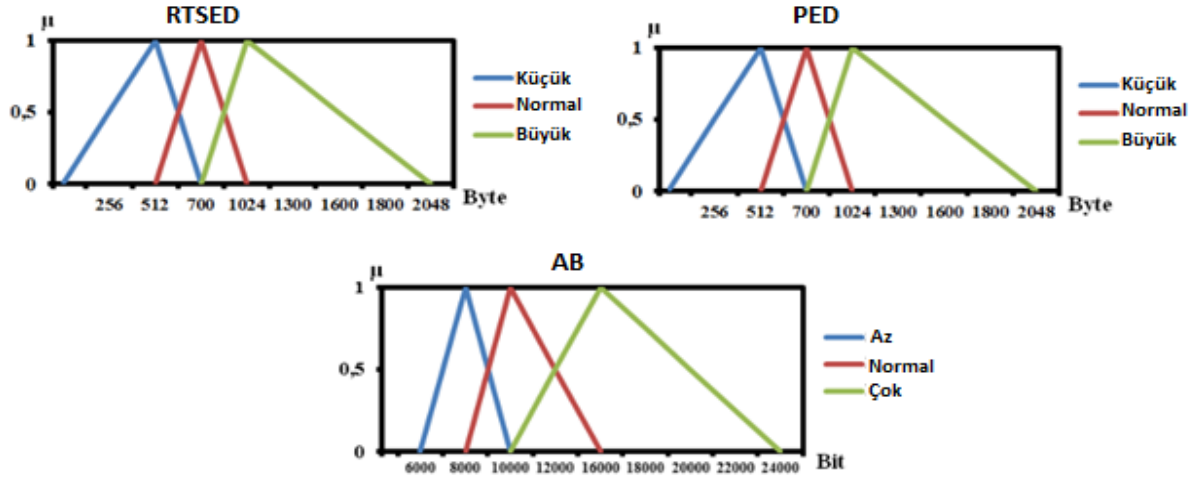
Çizelge 3.6. Bulanık mantık kural tablosu

Girişler			Çıkış
RTSED (bayt)	PED (bayt)	AB (bit)	Birim Zamanda İşlenen Paket Miktarı (bit/sn)
KÜ	KÜ	A	ÇA
KÜ	KÜ	NO	NO
KÜ	KÜ	Ç	ÇF
KÜ	NO	A	ÇA
KÜ	NO	NO	A
KÜ	NO	Ç	ÇF
KÜ	BÜ	A	ÇA
KÜ	BÜ	NO	ÇF
KÜ	BÜ	Ç	ÇF
NO	KÜ	A	ÇA
NO	KÜ	NO	NO
NO	KÜ	Ç	ÇF
NO	NO	A	ÇA
NO	NO	NO	ÇF
NO	NO	Ç	F
NO	BÜ	A	ÇA
NO	BÜ	NO	NO
NO	BÜ	Ç	ÇF
BÜ	KÜ	A	ÇA
BÜ	KÜ	NO	NO
BÜ	KÜ	Ç	F
BÜ	NO	A	ÇA
BÜ	NO	NO	A
BÜ	NO	Ç	ÇF
BÜ	BÜ	A	ÇA
BÜ	BÜ	NO	A
BÜ	BÜ	Ç	ÇF

3.4.2. RTSED, PED ve AB'nin geri besleme kontrollü bulanık mantıkla belirlenmesi

Bulanık mantık çıktıları için birleşim-kesişim operatörü kullanılmıştır. Durulaştırma işlemi için ise ağırlık merkezi yöntemi kullanılmıştır.

Şekil 3.24'te yer alan giriş üyelik fonksiyonları için RTSED 0-700 bayt arası 'Küçük', 512-1024 bayt aralığı için 'Normal', 700-2048 bayt aralığı için ise 'Büyük' değerleri kullanılmıştır. PED içinde 0-700 bayt aralığı 'Küçük', 512-1024 bayt aralığı için 'Normal', 700-2048 bayt aralığı için ise 'Büyük' değerleri kullanılmıştır. AB'de ise 6000-10 000 bit aralığı 'Az', 8000-16 000 bit aralığı 'Normal' 10 000-24 000 bit aralığı için ise 'Çok' sözel değerleri kullanılmış ve bu aralıklar küme aralıkları olarak belirlenmiştir.



Şekil 3.24. Giriş üyelik fonksiyonları

Birim zamanda işlenen paket miktarı için çıkış üyelik fonksiyonları Şekil 3.25'te gösterilmiştir. Burada; 0-20 000 bit/sn aralığı için 'Çok-az', 10 000-30 000 bit/sn aralığı için 'Az', 20 000-60 000 bit/sn aralığı için 'Normal', 30 000-80 000 bit/sn aralığı için 'Fazla' ve 60 000-90 000 bit/sn aralığı için 'çok-fazla' sözel değerleri kullanılmış ve bu aralıklar küme aralığı olarak tanımlanmıştır.



Şekil 3.25. Çıkış üyelik fonksiyonları

Bölüm 4'te tez çalışmasında elde edilen test sonuçları klasik yöntem test sonuçlarıyla karşılaştırılarak WLAN'lardaki performans iyileştirmeleri sunulmuştur.

4. DENEYSEL SONUÇLAR YORUMLAR

Bu bölümde ilk olarak klasik yöntemde kullanılan RTSED, PED ve AB giriş parametrelerinin çıkış parametrelerine olan etkisi incelenmiştir. Sonrasında tez çalışmasında RTSED, PED ve AB giriş parametrelerine uygulanan YSA, geri besleme kontrolü v.b yöntemlerin performansına olan etkileri analiz edilmiştir. Uygulanan modelde yükleme süresi, gecikme süresi, birim zamanda işlenen paket miktarı, alıcı kanal verimliliği, verici kanal verimliliği, alınan ve gönderilen veri trafiği çıkış parametrelerindeki performans iyileştirmeleri klasik yöntemle karşılaştırmalı olarak analiz edilmiştir.

4.1. Klasik Yöntem Test Sonuçları

Test çalışmalarında ilk olarak gerçekleştirilen ve 27 farklı giriş değerleri için elde edilen sonuçlar Çizelge 4.1’de yer almaktadır. Riverbed Modeler’de yapılan test çalışmalarında elde edilen RTSED, PED ve AB giriş değişkenleri ile ortalama gecikme süreleri, ortalama yükleme ve ortalama birim zamanda işlenen paket miktarı çıkış değişkenleri bulanık mantık uygulamalarında kural tabanı oluşturmak için kullanılmıştır.

Çizelge 4.1. Giriş ve çıkış parametreleri ortalama test sonuçları

RTSED (bayt)	PED (bayt)	AB (bit)	Gecikme (sn)	Yükleme (bit/sn)	Birim Zamanda İşlenen Paket Miktarı (bit/sn)
512	512	8000	0,002116	778 023	287 590
512	512	10 000	0,002500	703 351	448 887
512	512	16 000	0,002955	754 858	731 638
512	700	8000	0,001886	734 847	273 626
512	700	10 000	0,002158	732 568	464 315
512	700	16 000	0,002572	774 678	752 662
512	1024	8000	0,001530	777 242	279 715
512	1024	10 000	0,001879	732 225	765 236
512	1024	16 000	0,002263	796 284	776 778
700	512	8000	0,001886	778 023	287 590
700	512	10 000	0,002345	777 242	492 891
700	512	16 000	0,002793	751 164	723 661
700	700	8000	0,001649	754 322	279 465
700	700	10 000	0,001983	748 523	745 790
700	700	16 000	0,002366	726 172	704 615

Çizelge 4.1. Giriş ve çıkış parametreleri ortalama test sonuçları-devam

RTSED (bayt)	PED (bayt)	AB (bit)	Gecikme (sn)	Yükleme (bit/sn)	Birim Zamanda İşlenen Paket Miktarı (bit/sn)
700	1024	8000	0,001280	747 050	274 573
700	1024	10 000	0,001708	774 678	486 888
700	1024	16 000	0,002077	731 524	715 832
1024	512	8000	0,001530	748 599	283 977
1024	512	10 000	0,001984	748 523	490 043
1024	512	16 000	0,002507	740 540	717 304
1024	700	8000	0,001150	698 544	257 925
1024	700	10 000	0,001638	747 716	473 553
1024	700	16 000	0,002097	748 523	724 671
1024	1024	8000	0,000804	752 394	279 484
1024	1024	10 000	0,001363	697 320	458 820
1024	1024	16 000	0,001803	745 736	728 669

RTSED, PED ve AB giriş parametrelerinin çıkış parametrelerine olan etkisini daha az hata toleransı ile test edebilmek için 125 farklı değerlerde Çizelge 4.2’de yer alan temel test çizelgesi oluşturulmuştur. RTSED ve PED değerleri için 600-1200 bayt, AB için ise 9000-15 000 bit evrensel kümeleri belirlenmiş ve temel test çizelgesi oluşturulmuştur. Farklı giriş parametreleri alıcı kanal verimliliği, verici için kanal verimliliği, alınan veri trafiği ve gönderilen veri trafiği çıkış parametreleri için referans veriler olarak kullanılmıştır.

Çizelge 4.2. 125 giriş ve çıkış parametreleri ortalama test sonuçları

Giriş Değer Test No	RTSED (bayt)	PED (bayt)	AB (bit)	Kanal Kullanım Durumu Alıcı (%)	Kanal Kullanım Durumu Verici (%)	Alınan Veri Trafiği (bit/sn)	Gönderilen Veri Trafiği (bit/sn)
1	600	600	9000	10,75	1,25	547 725	64 826
2	600	600	10 000	13,35	1,33	689 853	65 846
3	600	600	12 000	19,6	2,08	1 041 157	110 998
4	600	600	13 000	20,28	2,02	1 083 750	99 395
5	600	600	15 000	19,45	1,94	1 029 788	100 998
6	600	700	9000	10,25	1,18	525 286	60 509
7	600	700	10 000	12,8	1,41	667 763	71 981
8	600	700	12 000	19,029	1,95	1 026 491	98 417

Çizelge 4.2. 125 giriş ve çıkış parametreleri ortalama test sonuçları-devam

Giriş Değer Test No	RTSED (bayt)	PED (bayt)	AB (bit)	Kanal Kullanım Durumu Alıcı (%)	Kanal Kullanım Durumu Verici (%)	Alınan Veri Trafığı (bit/sn)	Gönderilen Veri Trafığı (bit/sn)
9	600	700	13 000	19,02	1,94	1 026 427	100 269
10	600	700	15 000	18,33	2,06	986 157	100 437
11	600	900	9000	9,68	1,02	503 070	50 902
12	600	900	10 000	11,54	1,34	604 389	72 222
13	600	900	12 000	17,25	1,86	922 228	102 865
14	600	900	13 000	18,06	1,754	986 414	88 754
15	600	900	15 000	17,09	1,69	930 434	89 055
16	600	1000	9000	9,26	0,96	482 741	47 928
17	600	1000	10 000	11,92	1,16	633 444	57 776
18	600	1000	12 000	16,66	1,75	908 585	94 619
19	600	1000	13 000	17,16	1,68	943 312	84 610
20	600	1000	15 000	16,65	1,74	942 764	90 933
21	600	1200	9000	8,88	0,95	464 714	51 496
22	600	1200	10 000	11,21	1,19	605 607	62 576
23	600	1200	12 000	16,05	1,59	882 342	84 935
24	600	1200	13 000	16,78	1,83	920 910	94 153
25	600	1200	15 000	16,94	1,76	922 743	102 256
26	700	600	9000	10,26	1,14	549 302	63 179
27	700	600	10 000	12,2	1,42	656 186	79 263
28	700	600	12 000	18,85	1,9	1 034 333	97 458
29	700	600	13 000	19,42	2,07	1 058 595	119 678
30	700	600	15 000	20,07	2,1	1 098 704	112 220
31	700	700	9000	9,92	1,1	536 212	59 391
32	700	700	10 000	12,55	1,29	683 991	66 495
33	700	700	12 000	18,23	1,92	1 017 794	94 853
34	700	700	13 000	17,42	2,15	970 047	118 566
35	700	700	15 000	18,69	1,94	1 033 099	108 865
36	700	900	9000	8,85	0,96	485 645	51 515

Çizelge 4.2. 125 giriş ve çıkış parametreleri ortalama test sonuçları-devam

Giriş Değer Test No	RTSED (bayt)	PED (bayt)	AB (bit)	Kanal Kullanım Durumu Alıcı (%)	Kanal Kullanım Durumu Verici (%)	Alınan Veri Trafığı (bit/sn)	Gönderilen Veri Trafığı (bit/sn)
37	700	900	10 000	11,84	1,18	657 027	60 203
38	700	900	12 000	16,65	1,67	929 027	98 487
39	700	900	13 000	17,45	1,89	984 598	98 364
40	700	900	15 000	16,8	1,78	937 181	100 123
41	700	1000	9000	8,52	0,88	474 921	44 445
42	700	1000	10 000	11,07	1,19	614 034	65 504
43	700	1000	12 000	17,45	1,88	975 541	98 482
44	700	1000	13 000	16,76	1,76	957 964	90 604
45	700	1000	15 000	16,22	1,9	903 867	107 365
46	700	1200	9000	7,96	0,84	445 987	46 620
47	700	1200	10 000	10,29	1,06	583 473	56 107
48	700	1200	12 000	15,31	1,75	874 096	96 981
49	700	1200	13 000	16,07	1,48	903 516	83 570
50	700	1200	15 000	15,61	1,67	886 784	92 058
51	900	600	9000	9,44	0,96	564 168	57 200
52	900	600	10 000	12,08	1,33	707 570	77 560
53	900	600	12 000	18,52	2,03	1 079 527	117 564
54	900	600	13 000	18,06	1,99	1 051 571	111 440
55	900	600	15 000	18,26	1,97	1 050 169	114 878
56	900	700	9000	8,84	0,82	536 975	50 596
57	900	700	10 000	11,4	1,25	678 110	72 374
58	900	700	12 000	16,98	1,76	998 754	103 576
59	900	700	13 000	16,82	1,77	994 630	100 184
60	900	700	15 000	17,35	1,95	1 003 343	111 845
61	900	900	9000	8,2	0,87	508 493	54 458
62	900	900	10 000	10,57	1,11	641 372	64 166
63	900	900	12000	15,68	1,74	930 463	107 933
64	900	900	13 000	15,63	1,61	933 678	89 498

Çizelge 4.2. 125 giriş ve çıkış parametreleri ortalama test sonuçları-devam

Giriş Değer Test No	RTSED (bayt)	PED (bayt)	AB (bit)	Kanal Kullanım Durumu Alıcı (%)	Kanal Kullanım Durumu Verici (%)	Alınan Veri Trafığı (bit/sn)	Gönderilen Veri Trafığı (bit/sn)
65	900	900	15 000	15,13	1,72	942 382	102 021
66	900	1000	9000	7,7	0,78	485 224	48 607
67	900	1000	10 000	9,95	0,92	611 681	53 744
68	900	1000	12 000	15,61	1,72	949 639	102 501
69	900	1000	13 000	14,95	1,41	903 202	82 072
70	900	1000	15 000	15,06	1,58	911 948	91 880
71	900	1200	9000	7,02	0,74	453 001	47 517
72	900	1200	10 000	9,32	0,98	582 117	63 446
73	900	1200	12 000	14,08	1,64	869 619	100 447
74	900	1200	13 000	14,4	1,49	882 245	88 250
75	900	1200	15 000	15	1,61	931 431	97 473
76	1000	600	9000	8,64	0,89	555 101	53 240
77	1000	600	10 000	10,88	1,16	671 694	65 732
78	1000	600	12 000	16,32	1,82	980 179	105 411
79	1000	600	13 000	17,84	1,95	1 062 686	115 397
80	1000	600	15000	18,03	1,94	1 077 571	114 498
81	1000	700	9000	8,41	0,9	537 915	62 642
82	1000	700	10 000	10,82	1,09	651 811	71 734
83	1000	700	12 000	16,28	1,73	998 541	102 442
84	1000	700	13 000	17,02	1,76	1 037 152	104 954
85	1000	700	15 000	16,99	1,75	1 031 292	105 772
86	1000	900	9000	7,56	0,78	507 391	50 740
87	1000	900	10 000	10,03	1,04	651 811	64 369
88	1000	900	12 000	15,61	1,66	962 815	102 155
89	1000	900	13 000	15,59	1,51	973 987	89 899
90	1000	900	15 000	15,58	1,7	964 306	110 104
91	1000	1000	9000	7,42	0,75	504 094	49 502
92	1000	1000	10 000	9,78	0,92	635 378	57 518

Çizelge 4.2. 125 giriş ve çıkış parametreleri ortalama test sonuçları-devam

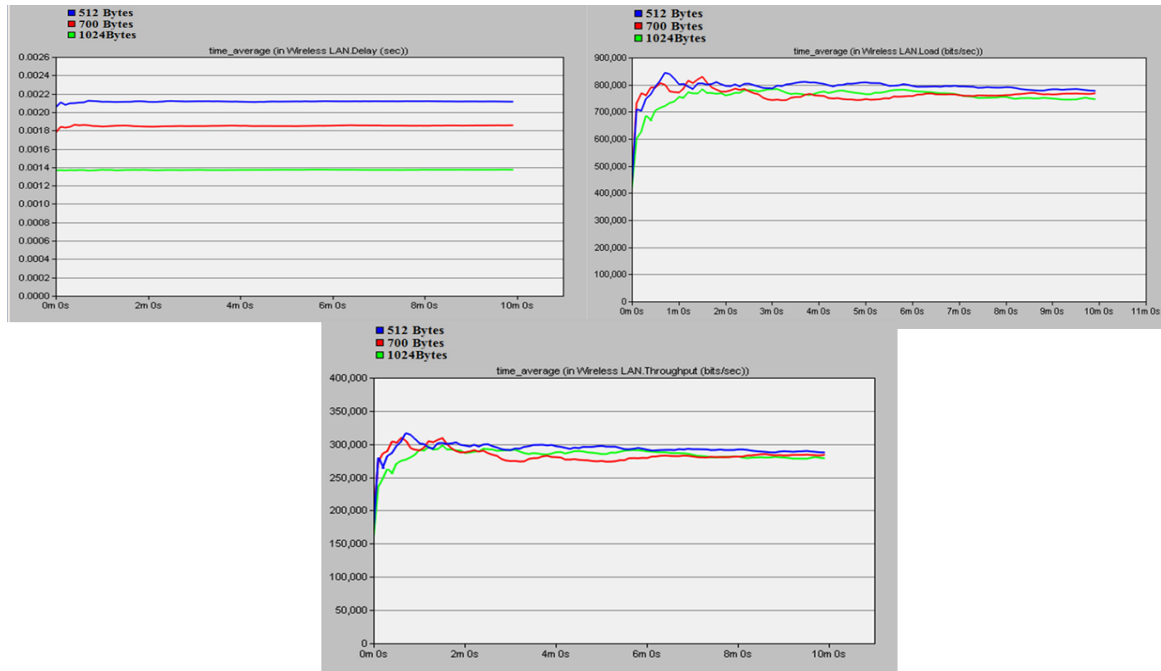
Giriş Değer Test No	RTSED (bayt)	PED (bayt)	AB (bit)	Kanal Kullanım Durumu Alıcı (%)	Kanal Kullanım Durumu Verici (%)	Alınan Veri Trafığı (bit/sn)	Gönderilen Veri Trafığı (bit/sn)
93	1000	1000	12 000	15,83	1,53	931 700	95 343
94	1000	1000	13 000	15,78	1,65	442 032	102 155
95	1000	1000	15 000	14,92	1,37	611 201	82 292
96	1000	1200	9000	6,32	0,6	937 235	42 504
97	1000	1200	10 000	9,13	0,96	960 089	59 451
98	1000	1200	12 000	14,68	1,4	875 046	85 052
99	1000	1200	13 000	15,06	1,44	568 112	87 823
100	1000	1200	15 000	13,9	1,67	695 450	107 325
101	1200	600	9000	8,04	0,92	568 112	65 527
102	1200	600	10 000	10,25	1,02	695 450	68 245
103	1200	600	12 000	16,59	1,67	1 065 258	95 066
104	1200	600	13 000	17,13	1,78	1 086 869	105 053
105	1200	600	15 000	17,04	1,76	1 080 477	110 960
106	1200	700	9000	7,65	0,73	546 653	52 789
107	1200	700	10 000	9,47	0,98	653 426	68 989
108	1200	700	12 000	15,38	1,72	1 008 027	102 540
109	1200	700	13 000	15,52	1,65	1 009 792	112 370
110	1200	700	15 000	15,77	1,7	1 027 580	99 854
111	1200	900	9000	6,67	0,72	499 593	55 048
112	1200	900	10 000	8,78	0,88	631 104	59 379
113	1200	900	12 000	14,5	1,44	976 559	88 436
114	1200	900	13 000	14,12	1,55	928 961	104 620
115	1200	900	15 000	14,52	1,56	973 551	99 012
116	1200	1000	9000	6	0,61	462 639	44 635
117	1200	1000	10 000	8,49	0,92	615 365	68 300
118	1200	1000	12 000	14,73	1,68	979 284	115 486
119	1200	1000	13 000	13,76	1,56	930 124	99 665
120	1200	1000	15000	14,06	1,5	937 828	101 459

Çizelge 4.2. 125 giriş ve çıkış parametreleri ortalama test sonuçları-devam

Giriş Değer Test No	RTSED (bayt)	PED (bayt)	AB (bit)	Kanal Kullanım Durumu Alıcı (%)	Kanal Kullanım Durumu Verici (%)	Alınan Veri Trafiği (bit/sn)	Gönderilen Veri Trafiği (bit/sn)
121	1200	1200	9000	5,83	0,58	466 978	44 665
122	1200	1200	10 000	7,93	0,87	596 308	68 738
123	1200	1200	12 000	13,7	1,43	943 827	89 538
124	1200	1200	13 000	13,71	1,51	925 967	110 064
125	1200	1200	15 000	13,29	1,44	906 218	99 012

4.1.1. RTSED ile yükleme, gecikme ve birim zamanda işlenen paket miktarı test sonuçları

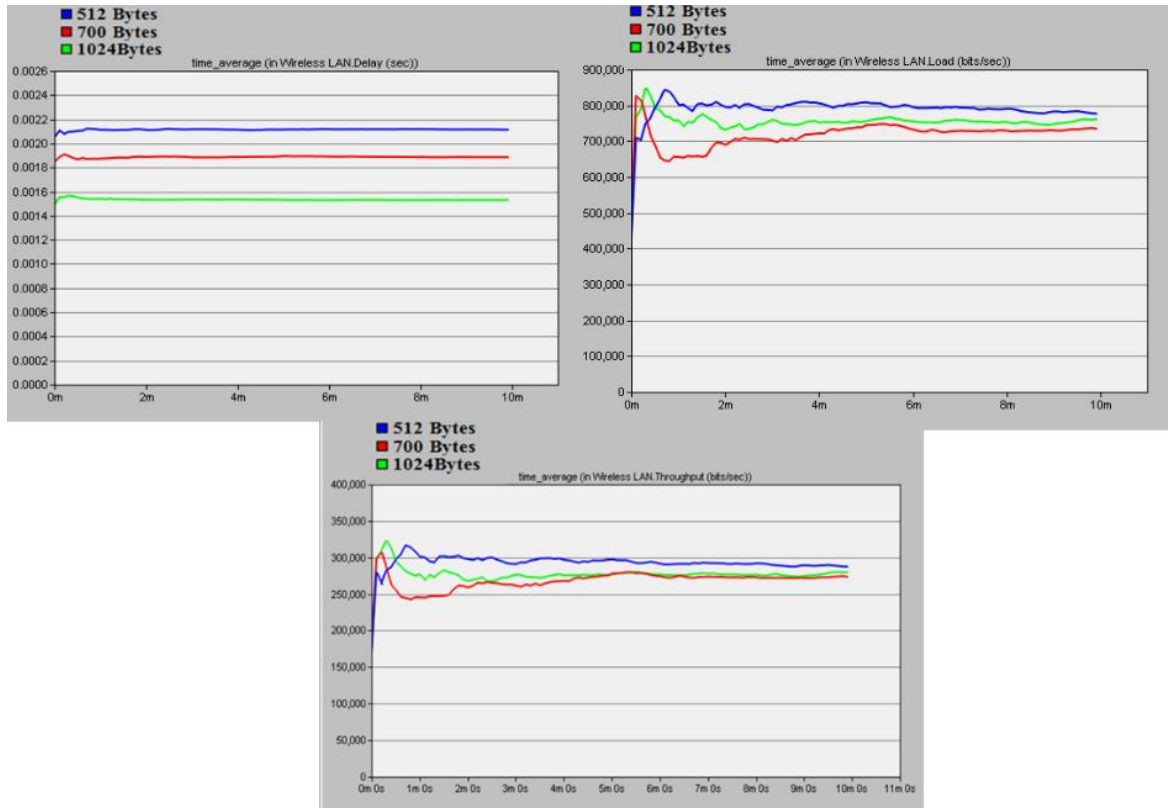
Evrensel kümede 512-700-1024 bayt RTSED değerleri seçilmiştir. Şekil 4.1’de verilen grafiklerde RTSED değeri artarken gecikmenin ve yüklemenin azaldığı görülmektedir. RTSED’nin artmasıyla birim zamanda işlenen paket miktarının çok az miktarda düzensiz olarak azaldığı görülmektedir.



Şekil 4.1. RTSED performans karşılaştırmaları

4.1.2. PED ile yükleme, gecikme ve birim zamanda işlenen paket miktarı test sonuçları

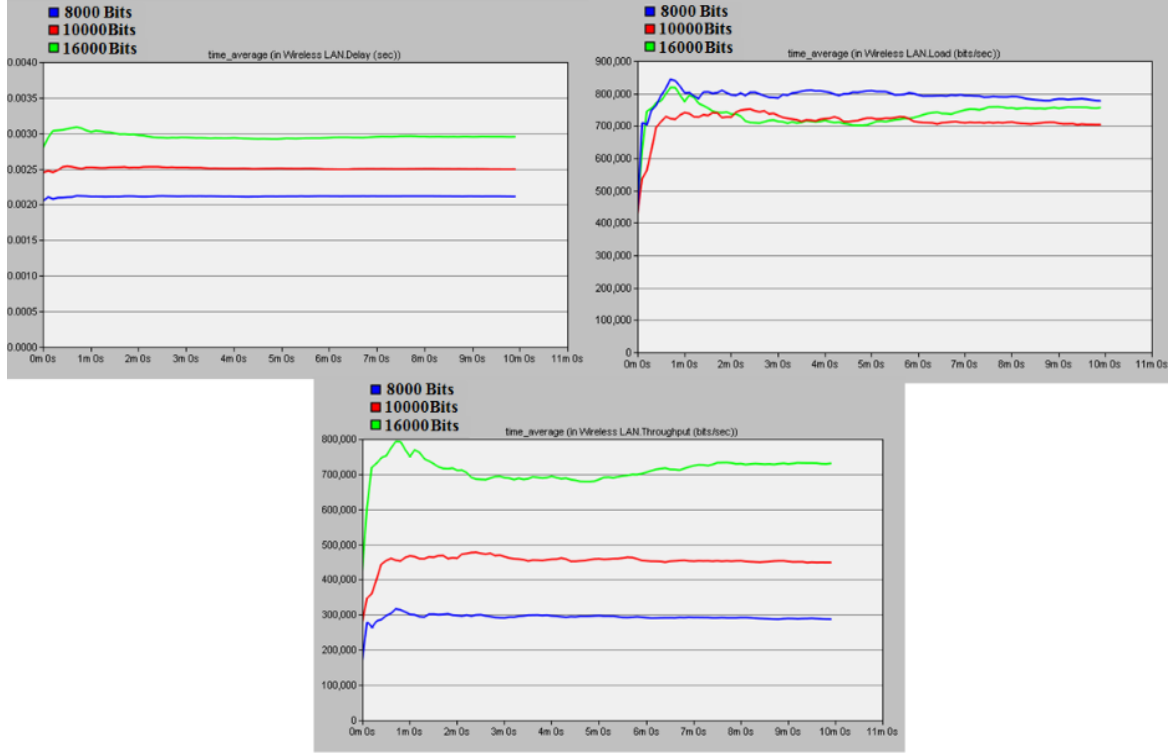
Yapılan çalışmada PED değerleri için 512-700-1024 bayt değerleri kullanılmıştır. Gecikme süresinin bu aralıkta düzensiz azaldığı, yüklemenin; 512 bayt ile 700 bayt arasında azaldığı, 700 bayt ile 1024 bayt arasında ise arttığı görülmektedir. Şekil 4.2’de birim zamanda işlenen paket miktarının ise 512 bayt ve 700 bayt arasında azalıp, 700 bayt ile 1024 bayt arasında düzensiz olarak arttığı görülmektedir.



Şekil 4.2. PED performans karşılaştırmaları

4.1.3. AB ile yükleme, gecikme ve birim zamanda işlenen paket miktarı test sonuçları

CSMA/CA’da CTS sinyalinin gelmemesi durumunda rastgele bir zaman beklenildiği için arabellek boyutu 0 - 16 000 bit arasında farklı değerler ile test edilmiştir. Şekil 4.3’te gösterildiği gibi CSMA/CA modeli için AB değerinin artması (8000-10 000-16 000 bit) gecikme süresini düzenli olarak arttırmaktadır. Yükleme; 8000 bit 10 000 bit arasında azalmakta, 10 000 bit ile 160 000 bit arasında ise artmaktadır. AB değeri arttığında birim zamanda işlenen paket miktarı da düzensiz olarak artmaktadır.

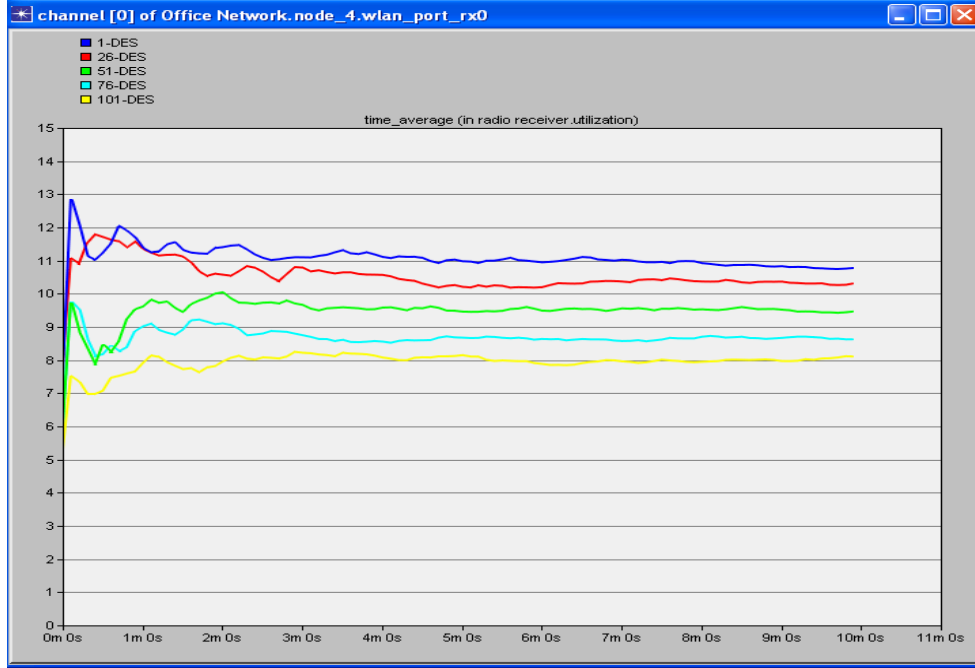


Şekil 4.3. AB performans karşılaştırmaları

4.1.4. RTSED, PED ve AB ile alınan ve gönderilen veri trafiği, gönderici ve alıcı kanal kullanım durumu test sonuçları

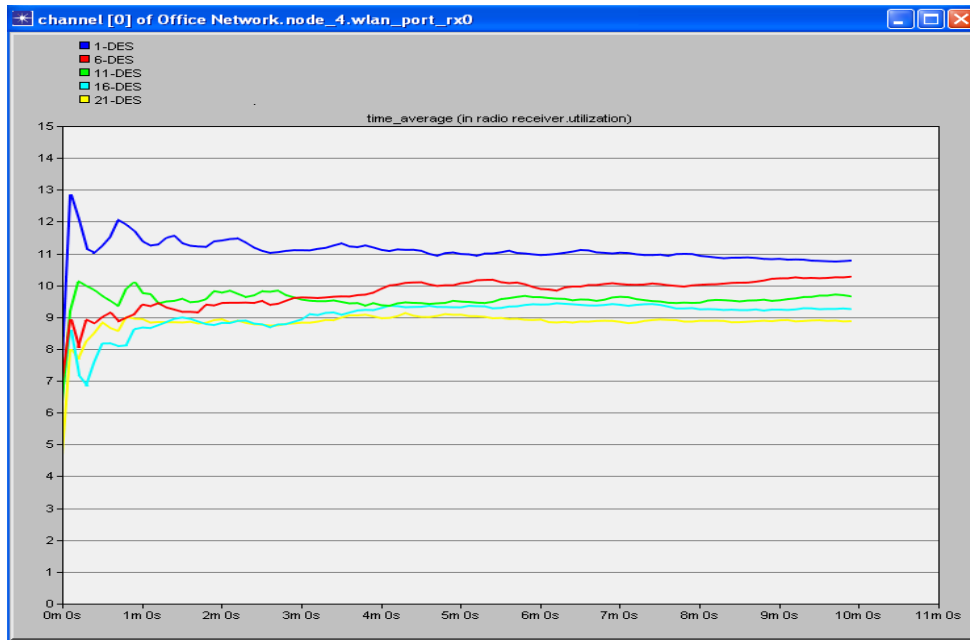
RTSED değerleri evrensel kümede alınan ve gönderilen veri trafiği, alıcı kanal kullanım durumu ve gönderici kanal kullanım durumu için 600-700-900-1000-1200 bayt olarak seçilmiştir.

Yapılan test çalışmaları sonucunda kanal kullanım durumu hem alıcı hem de verici için RTSED ile ters orantılı olarak Şekil 4.4'te gösterildiği gibi değişmiştir. Alınan veri trafiği ise RTSED ile birlikte düzenli olarak artmaktadır. Gönderilen veri trafiği; 900 bayt ve 1200 bayt arasında artmakta ve 600 bayt ile 900 bayt arasında ise azalmaktadır.



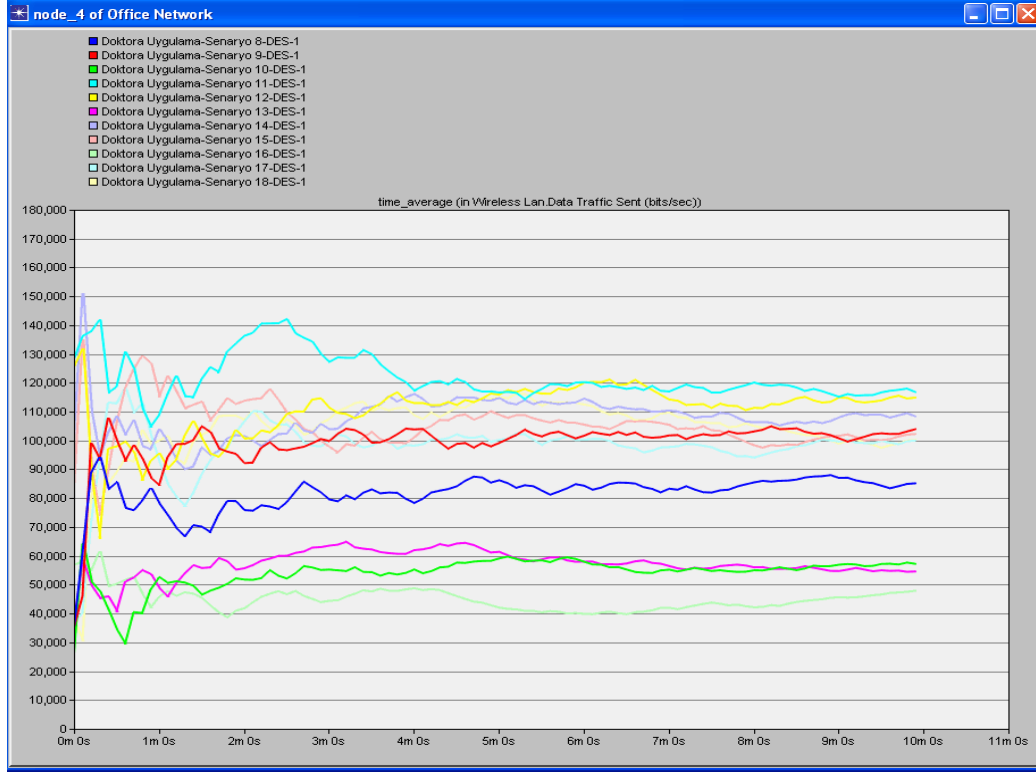
Şekil 4.4. RTSED örnek alıcı kanal kullanım durumu performans karşılaştırması

Evrensel kümede alınan veri trafiği, gönderilen veri trafiği, alıcı kanal kullanım durumu ve gönderici kanal kullanım durumu için ise 600-700-900-1000-1200 bayt PED değerleri seçilmiştir. PED değeri artarken, alıcı ve verici kalan kullanım durumu Şekil 4.5'teki gibi düzenli olarak azalmaktadır.



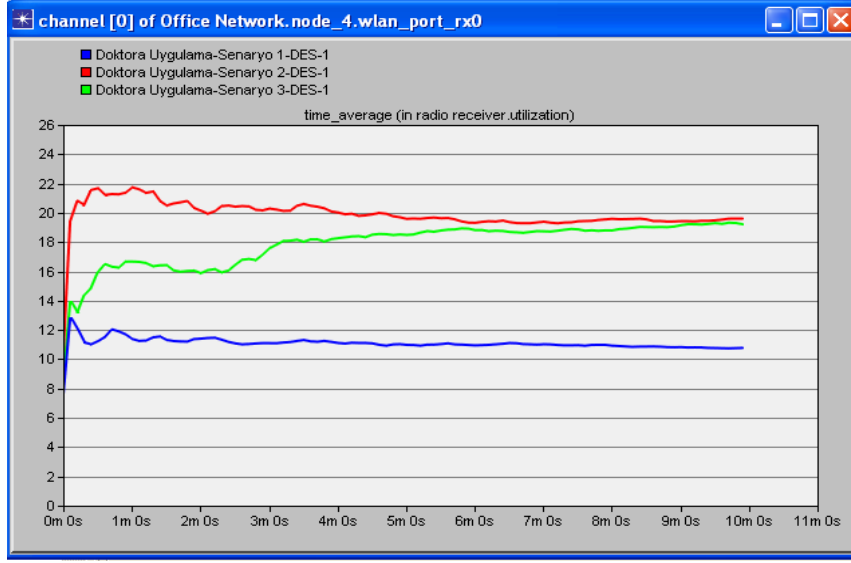
Şekil 4.5. PED örnek alıcı kanal kullanım durumu performans karşılaştırması

Alınan veri trafiği Şekil 4.6’da görüldüğü gibi 600 bayt ile 1200 bayt arasında artarken, gönderilen veri trafiği; 600 900 bayt ve 1200 bayt arasında artmakta, 600 bayt ile 900 bayt arasında ise azalmaktadır.



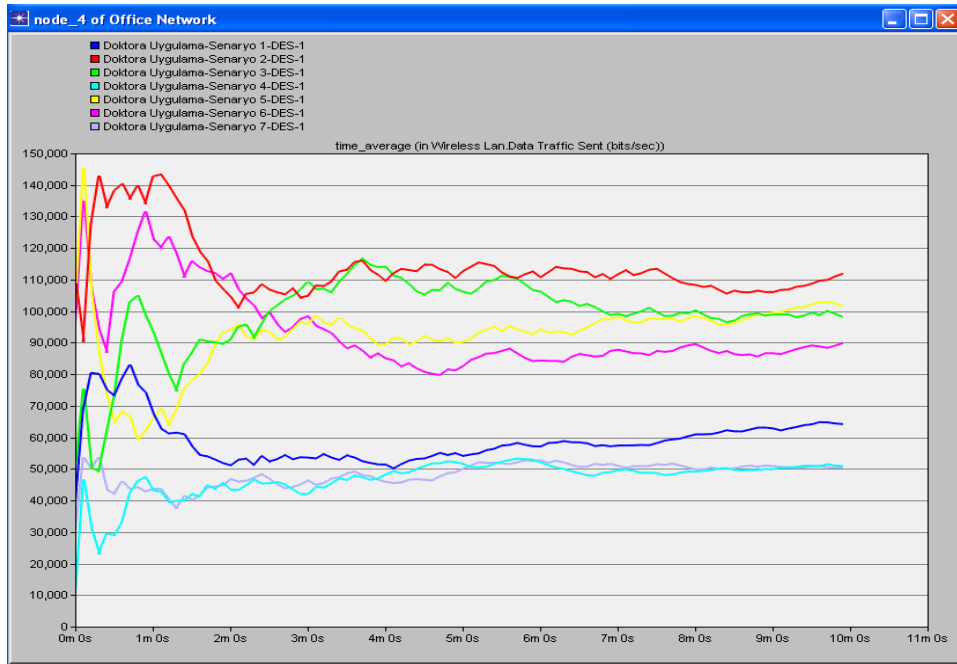
Şekil 4.6. PED için ortalama gönderilen veri trafiği performans karşılaştırmaları

Evrensel kümede AB değerleri 9000-10 000-12 000-13 000-15 000 bit seçilmiştir. Kanal kullanım durumu alıcı için; Şekil 4.7’deki gibi AB 9000 bit ile 13 000 bit arasında artmakta, 13 000 bit ile 15 000 bit arasında ise azalmaktadır. Kanal kullanım durumu verici için; AB 9000 bit ile 12 000 bit arasında artmakta, 12 000 bit ile 15 000 bit arasında ise azalmaktadır.



Şekil 4.7. AB alıcı kanal durumu performans karşılaştırmaları

AB'nin ölçülmesi için Riverbed Modeler benzetim aracı kullanılarak yapılan test çalışmalarında evrensel kümede 9000 bit, 12 000 bit ve 15 000 bit girdileri kullanılmıştır. Yapılan test çalışmalarında kanal kullanım durumu alıcı için 9000 bit ile 15 000 bit arasında artarken, kanal kullanım durumu verici için; 12 000 bit ile 15 000 bit arasında ise azalmakta, 9000 bit ile 12 000 bit arasında ise artmaktadır. Alınan veri trafiği ve gönderilen veri trafiklerinde; Şekil 4.8'deki gibi AB girdileri 9000 bit ile 12 000 bit arasında iken düzenli artış ve 12 000 bit ile 15 000 bit arasında ise azalış görülmektedir.



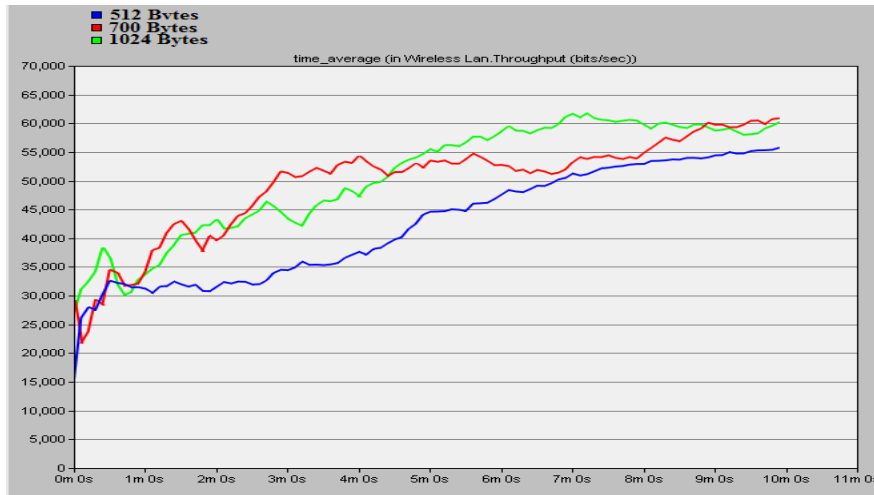
Şekil 4.8. AB için ortalama gönderilen veri trafiği performans karşılaştırmaları

4.1.5. RTSED, PED ve AB ile düğüm bazlı birim zamanda işlenen paket miktarı test sonuçları

Düğüm bazlı birim zamanda işlenen paket miktarı incelendiğinde RTSED, PED ve AB için sonuçlar farklılık göstermektedir.

RTSED

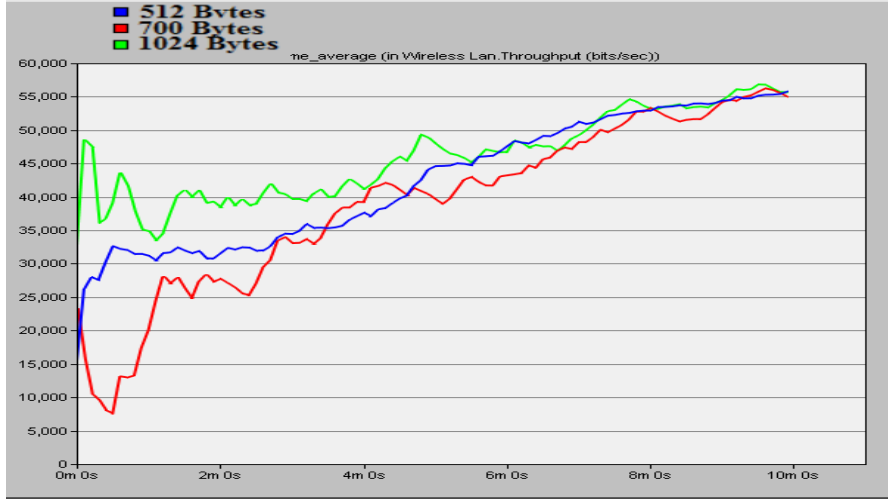
Evrensel kümede 512-700-1024 bayt RTSED değerleri seçilmiştir. Şekil 4.9’da verilen grafiklerden, RTSED değeri 512 bayt ile 700 bayt arasında artarken birim zamanda işlenen paket miktarı da artmıştır. RTSED değeri 700 bayt ve 1024 bayt arasında ise birim zamanda işlenen paket miktarı nispeten eşit çıkmakla beraber benzetim koşumunun sonuna doğru 700 bayt değeri daha fazla düğüm bazlı birim zamanda paket işlemiştir.



Şekil 4.9. RTSED performans karşılaştırmaları

PED

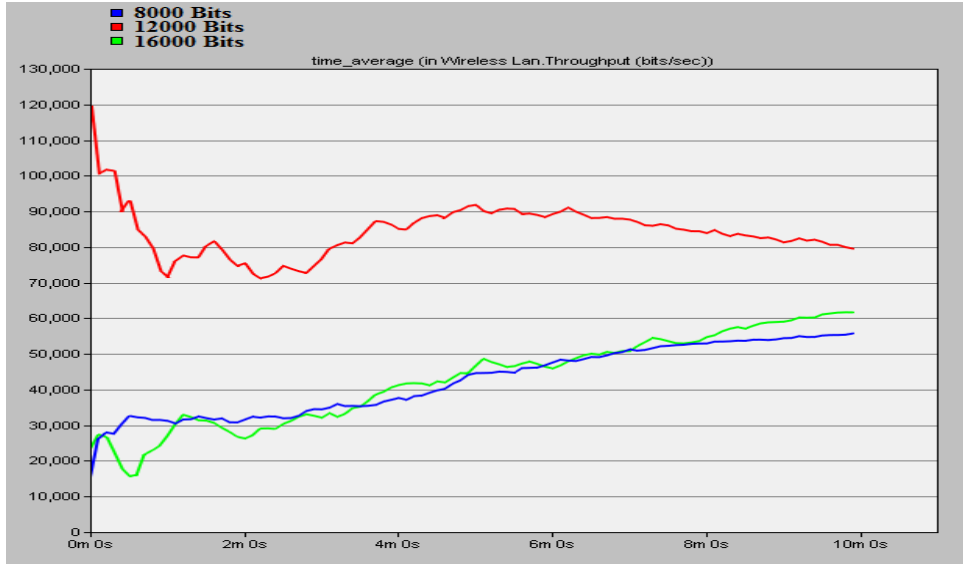
CSMA/CA’da PED değeri için evrensel küme 512-700-1024 bayt değerleri olarak belirlenmiştir. Birim zamanda işlenen paket miktarı 512 bayt, 700 bayt ve 1024 bayt arasında PED değerinin azalması ile Şekil 4.10’daki gibi artmaktadır.



Şekil 4.10. PED performans karşılaştırmaları

AB

Arabellek boyutu evrensel kümesi 0-16 000 bit arasında belirlenmiştir. Düğüm bazlı birim zamanda işlenen paket miktarı 12 000 bite doğru artmaktadır. Dolayısıyla 8000 bit ile 12 000 bit arasında artmakta, 12 000 bit ile 16 000 bit arasında ise Şekil 4.11'deki gibi azalmaktadır.



Şekil 4.11. AB performans karşılaştırmaları

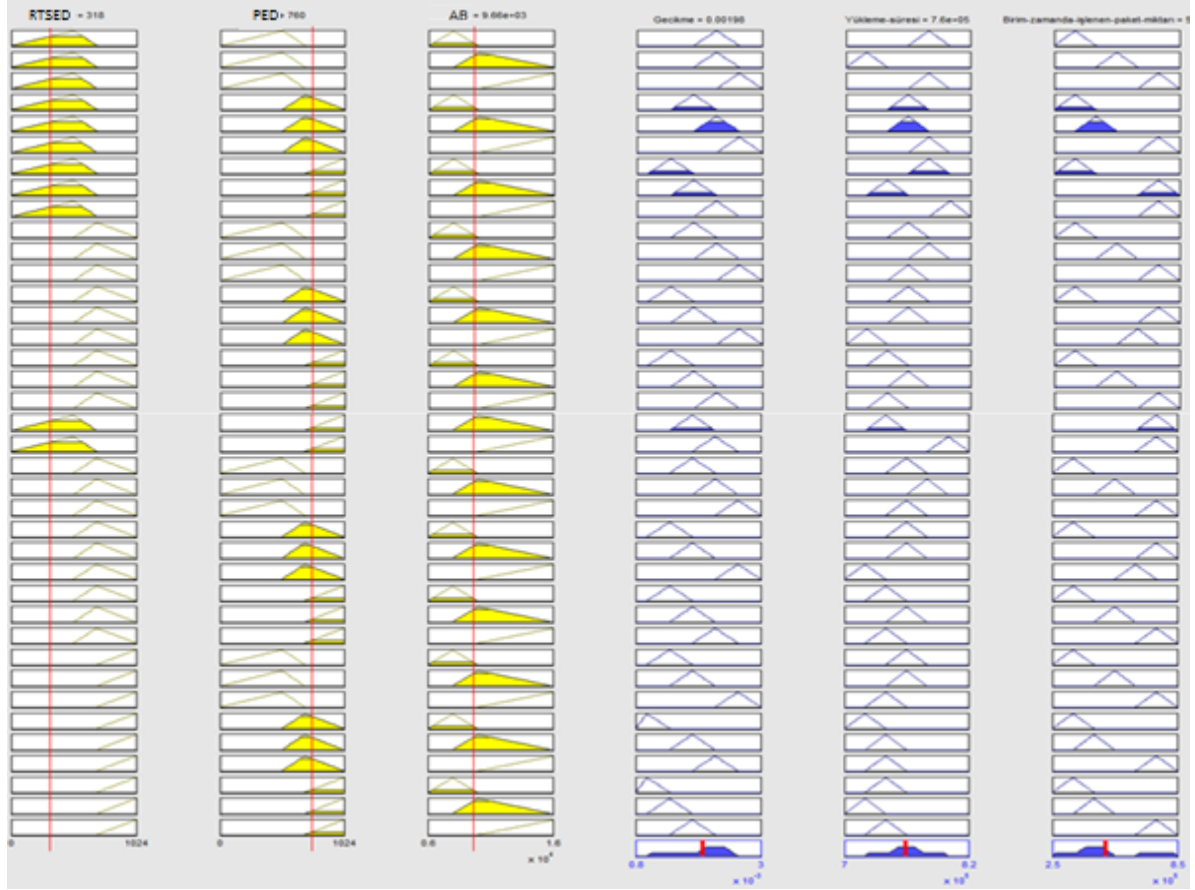
Sonraki bölümlerde klasik test sonuçları ile tez çalışmasında önerilen ve gerçekleştirilen sonuçlar karşılaştırılacaktır.

4.2. Bulanık Mantık Tabanlı Yöntem Analizi ile Çıkış Parametreleri Performans Karşılaştırmaları

Bulanık mantık tabanlı uygulama sonucunda ideal gecikme, yükleme ve birim zamanda işlenen paket miktarı değerlerinin tespiti için 1. örnek girdi değerlerimiz; RTSED için 318 bayt, PED için 760 bayt ve AB için 9660 bit olarak belirlenmiştir. Birleşim-kesişim çıkarım metodu ile öncelikli olarak örnek girdilerin bulunmuş olduğu küme aralığındaki ağırlıklar tespit edilmiştir. Bu değerlerin en düşük olanına göre çıkış üyelik fonksiyonlarında küme aralık değerleri de belirlenmiştir.

1. Örnek kural tabanı; *EĞER* RTSED='(Küçük)', PED='(Normal, Büyük)' ve AB='(Normal, Az)' İSE gecikme='(Normal, Fazla)', Yükleme='(Normal, Fazla)', Birim zamanda işlenen paket miktarı='(Normal, Az)'

Varyasyonları kullanılarak bu aralıklardaki oranların en düşük olanına göre işletilmiştir. Şekil 4.12'de görülen çıkış üyelik fonksiyonlarında yer alan değerlerin çıkarım işleminden sonra ağırlık merkezi yöntemi ile durulaştırılmasıyla gecikme 0,0019 sn, yükleme 760 500 bit/sn ve birim zamanda işlenen paket miktarı 505 000 bit/sn olarak ölçülmüştür.

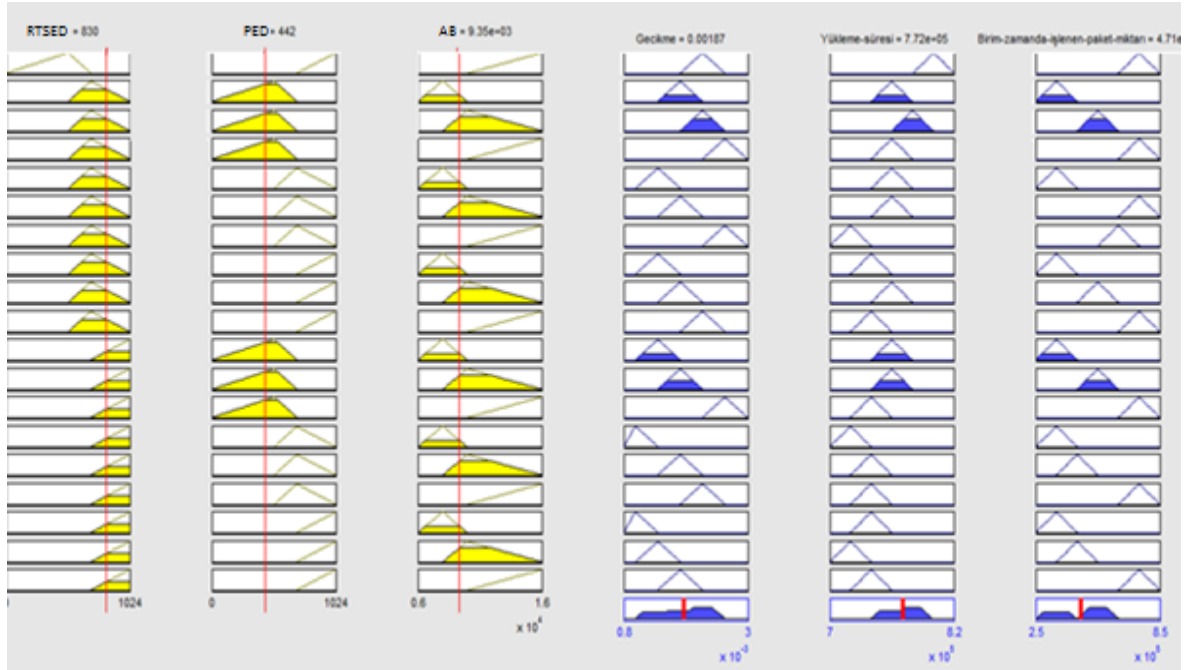


Şekil 4.12. Bulanık mantık tabanlı MATLAB test sonuçları-1

2. Örnek girdi değerlerimiz; RTSED için 830 bayt, PED için 442 bayt ve AB için 9350 bit kullanılmış ve birleşim-kesişim çıkarım metodu ile öncelikli olarak örnek girdilerin bulunmuş olduğu küme aralığındaki ağırlıkları tespit edilmiştir. Giriş verilerinin en düşük olanına göre çıkış üyelik fonksiyonlarındaki küme aralıkları da belirlenmiştir.

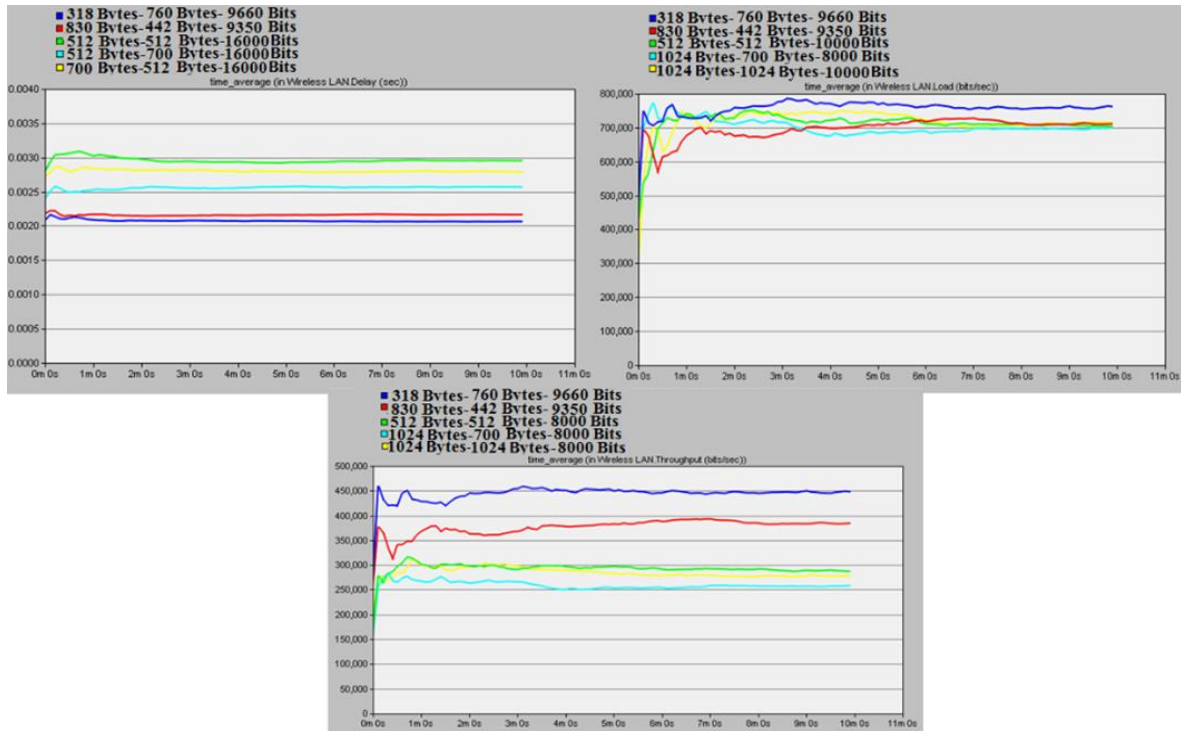
2. Örnek kural tabanı; $EĞER RTSED = (Normal, Büyük)$ ve $PED = (Küçük)$ ve $AB = (Normal, Az)$ İSE $gecikme = (Normal, Fazla)$, $Yükleme = (Çok-az, az)$, $Birim zamanda işlenen paket miktarı = (Normal, Az)$

Varyasyonları kullanılarak bu aralıklardaki oranların en düşük olanına göre işletilmiştir. Daha sonra çıkış üyelik fonksiyonlarında yer alan değerlerin birleşiminden sonra ağırlık merkezi yöntemi ile durulaştırılmasıyla gecikme, yükleme ve birim zamanda işlenen paket miktarları için ideal olan gecikme 0,00187 sn, yükleme 772 000 bit/sn, birim zamanda işlenen paket miktarı 470 000 bit/sn olarak Şekil 4.13'teki gibi elde edilmiştir.



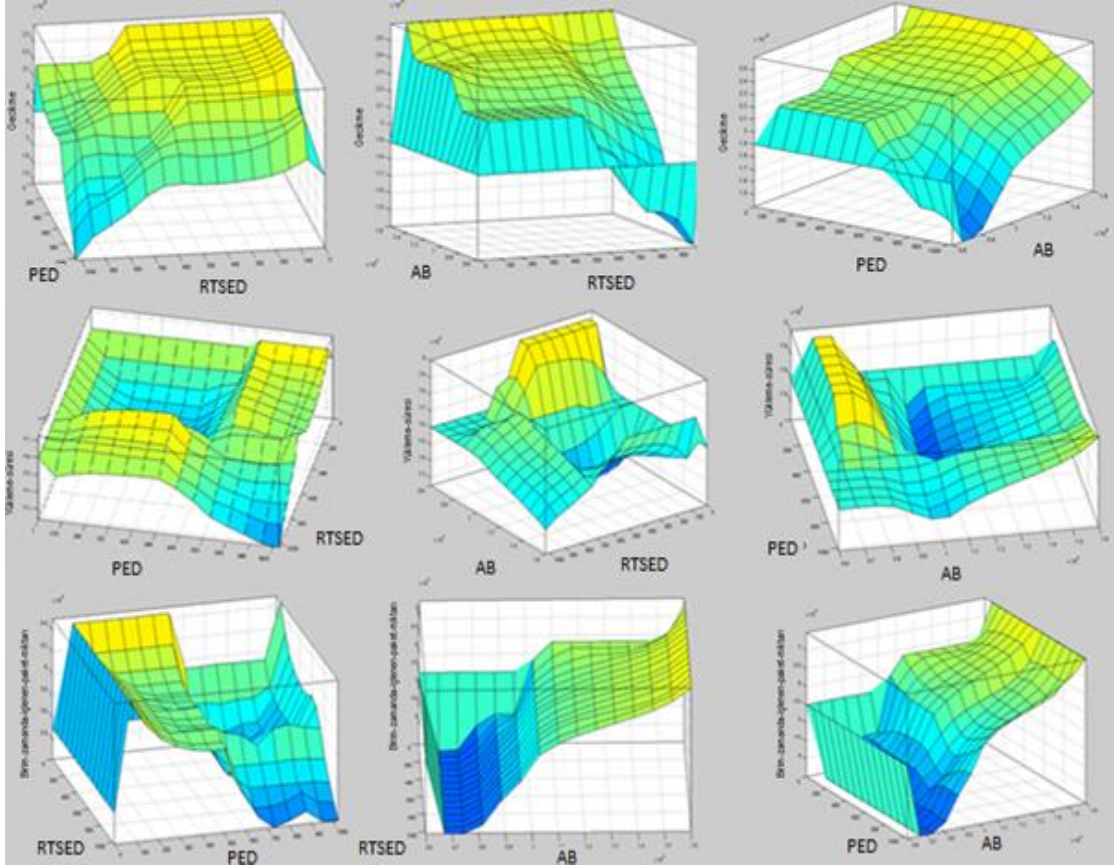
Şekil 4.13. Bulanık mantık tabanlı MATLAB test sonuçları-2

Elde edilen ideal değerler farklı konfigürasyonlarla karşılaştırıldığında Şekil 4.14'te görüldüğü gibi örnek girdilerimizin oluşturmuş olduğu konfigürasyonların performansının çok daha pozitif olduğu görülmektedir.



Şekil 4.14. Bulanık mantık sonuçlarının Riverbed Modeler'de uygulanması

Şekil 4.15'te giriş ve çıkış parametreleri için bulanık mantık çalışması ile elde edilen üç boyutlu performans ölçeği gösterilmektedir.



Şekil 4.15. Performansın üç boyutlu genel gösterimi

Gecikme Süresi

Riverbed Modeler’de gerçekleştirilen WLAN ağ modelinden elde edilen ortalama gecikme süreleri 0,0030 ile 0,0025 sn aralığında Çizelge 4.1’deki gibi iken bulanık mantık uygulanan örnek 1 ve örnek 2’de elde edilen gecikme süreleri yaklaşık olarak 0,0019 sn ve 0,0018 sn olarak ölçülmüştür. Bu sonuçlarla gecikme süresinde %36 ile %38 aralığında bir iyileştirme gerçekleştirilmiştir.

Elde edilen sonuçlar incelendiğinde genel ölçek için birtakım bulgular ortaya çıkmıştır. Şekil 4.15’te performansların üç boyutlu olarak karşılaştırılmasında ortaya çıkan sonuçlar da RTSED ve PED’in artması durumunda genel olarak gecikme süresinin azaldığı, RTSED’nin artışı ile gecikme süresinin daha az oranda azaldığı görülmektedir. AB ile

RTSED'nin karşılaştırılmasında ise AB'nin artışı ile gecikme süresinin arttığı, ancak AB'nin en yüksek olduğu durumda gecikmenin zirvede olmadığı görülmektedir. RTSED'nin artması durumunda ise gecikme süresi azalmaktadır. AB ile PED karşılaştırılmasında ise AB'nin artışı ile gecikmenin arttığı, PED'in artmasıyla da gecikmenin azaldığı görülmektedir.

Yükleme Süresi

WLAN ağ modelinden elde edilen yükleme süreleri ortalama 700 000 bit/sn ile 730 000 bit/sn aralığında Çizelge 4.1'deki gibi iken örnek 1 ve örnek 2'de elde edilen yükleme süreleri yaklaşık olarak 770 000 bit/sn ve 740 000 bit/sn elde edilmiştir. Yüklemeye süresi için %2 ile %10 aralığında bir iyileştirme gerçekleştirilmiştir.

Yüklemeye süresi karşılaştırmalarında PED ve RTSED'nin yaklaşık olarak 200-700 bayt arasında olduğu durumda en az yüklemenin olduğu görülmektedir. PED'in veya RTSED'den sadece birinin azalması ile yüklemenin arttığı görülmüştür. Yüklemeye süresinin en yüksek değeri, AB'nin en düşük ve RTSED'nin 0-512 bayt aralığında olduğu durumda ortaya çıkmıştır. Yüklemeye için PED ve AB karşılaştırılmasında ise; PED'in 0-400 bayt aralığı ve AB'nin en düşük olduğu 8000-9000 bit aralığı en iyi yüklemeye süresi olarak görülmektedir.

Birim Zamanda İşlenen Paket Miktarı

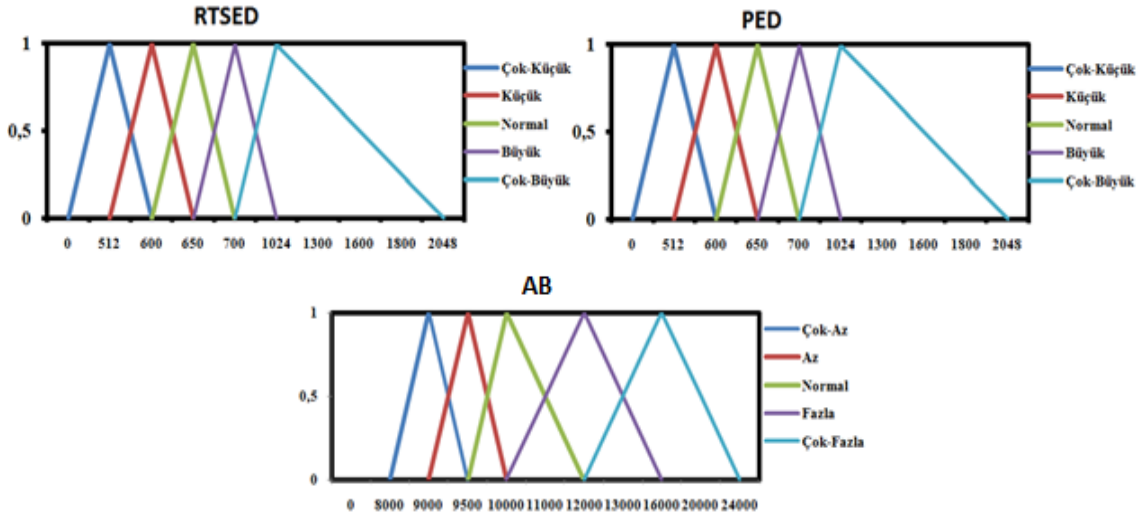
WLAN ağ modelinden elde edilen birim zamanda işlenen paket miktarı süreleri 250 000 bit/sn-300 000 bit/sn olarak elde edilmişti. Örnek 1 ve örnek 2'de elde edilen birim zamanda işlenen paket miktarı süreleri 450 000 bit/sn ve 400 000 bit/sn olarak ölçülmüştür. Burada da %25 ile %44 aralığında bir iyileştirme yapılmıştır.

Birim zamanda işlenen paket miktarında ise PED ve RTSED'nin karşılaştırılmasında; PED'in 700-1024 bayt arasında artmasıyla orantılı olarak azalma, PED ve RTSED'nin beraber artmasıyla da azalma görülmektedir. Birim zamanda işlenen paket miktarı için AB ile RTSED ve PED karşılaştırıldığında her iki durum içinde AB'nin artmasıyla düzenli olarak artış görülmektedir.

5 Giriş ve 7 Çıkışlı Uygulama Test Sonuçları

Veri setleri ile yapılan bulanık mantık tabanlı çalışmada sonuçların daha hassas ve net elde etmek için giriş parametrelerinde yer alan evrensel küme 5, çıkış değerlerinde yer alan evrensel küme ise 7 farklı değer ile tekrar ağırlandırılmıştır. Bulanık mantık kural tabanı yeniden oluşturulmuş ve gecikme süresi, yükleme süresi, ve birim zamanda işlenen paket miktarı değerleri çıkış parametreleri ölçülmüştür.

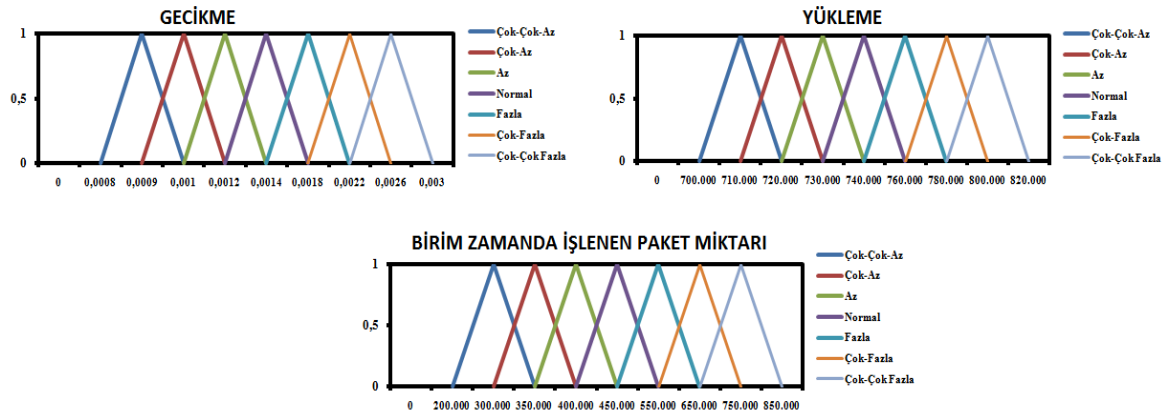
Giriş değişkenleri için üyelik fonksiyonları Şekil 4.16'da gösterildiği gibi tanımlanmıştır. RTSED ve PED için 0 bayt ile 600 bayt arasında “çok küçük”, 512 bayt ile 650 bayt arasında “küçük”, 600 bayt ile 700 bayt arasında “normal”, 650 bayt ile 1024 bayt arasında “büyük ” ve 700 bayt ile 2048 bayt arasında “çok büyük ” kullanılmıştır. AB için 8000 bit ile 9500 bit arasında “çok az”, 9000 bit ile 10 000 bit arasında “az”, 9500 bit ile 12 000 bit arasında “normal”, 10 000 bit ile 16 000 bit arasında “fazla” kullanılmıştır. 12 000 bit ile 24 000 bit arasında ise “çok fazla” sözel değeri kullanılmıştır. Bu aralıklar, bulanık çıktı kümelerinin giriş aralığı olarak tanımlanır.



Şekil 4.16. Giriş üyelik fonksiyonları

Çıkış değişkenleri için üyelik fonksiyonları Şekil 4.17'de gösterildiği gibi tanımlanmıştır. Gecikme için 0,0008 ile 0,0010 sn arasında “çok çok az”, 0,0009 sn ile 0,0012 sn arasında “çok az”, 0,0010 sn ile 0,0014 sn arasında “az” sözel değerleri kullanılmıştır. 0,0012 ve 0,0018 sn arasında “normal”, 0,0014 ile 0,0022 sn arasında “fazla”, 0,0018 ile 0,0026 sn arasında “çok fazla” ve 0,0022 ile 0,003 sn arasında “çok çok fazla” sözel değerleri kullanılmıştır. Yükleme süresi için 700 000 ve 720 000 bit/sn arasında “çok çok az”, 710 000 ve 730 000 bit/sn arasında “çok az”, 720 000 ve 740 000 bit/sn arasında “az” değerleri

kullanılmıştır. 730 000 ve 760 000 bit arasında “normal”, saniyede 740 000 ve 780 000 bit arasında "fazla" kullanılır, 760 000 ile 800 000 bit arasında saniyede "çok çok fazla" kullanılır ve 780 000 ve 820 000 bit/sn arasında ise "çok çok fazla" değerleri kullanılmıştır. Birim zamanda işlenen paket miktarı için ise 200 000 ve 350 000 bit/sn arasında “çok çok az”, 300 000 ve 400 000 bit/sn arasında “çok az”, 350 000 ve 450 000 bit/sn arasında “az” değerleri kullanılmıştır. Saniyede 400 000 ile 550 000 bit arasında, “normal”, 450 000 ile 650 000 bit arasında saniyede, “fazla” 550 000 ile 750 000 bit/sn arasında “çok fazla” ve 650 000 ile 850 000 bit/sn aralığında “çok çok fazla” sözel değerleri kullanılmıştır. Bu aralıklar, bulanık çıktı kümelerinin çıkış aralığı olarak tanımlanır.



Şekil 4.17. Çıkış üyelik fonksiyonları

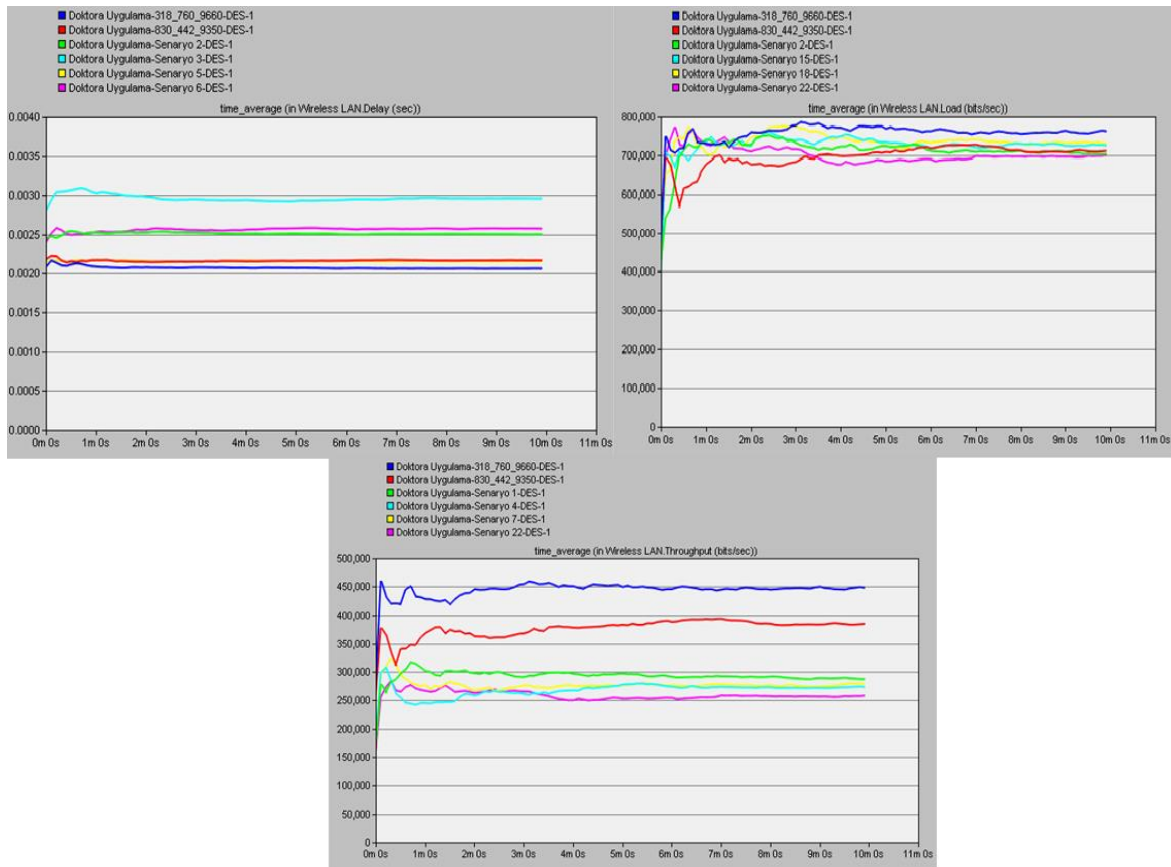
İdeal gecikme süresi, yükleme süresi ve birim zamanda işlenen paket miktarı değerlerini belirlemek için bazı örnek girişler bulanık sistemde uygulanmıştır. İlk örnek girdi değerlerimiz; RTSED için 318 bayt, PED için 760 bayt ve AB için 9660 bit'tir.

1. Örnek kural tabanı; $EĞER\ RTSED = '(Çok-Küçük)'$, $PED = '(Büyük, Çok\ Büyük)'$ ve $AB = '(Normal, Az)'$ İSE $gecikme = '(Çok\ Fazla, Fazla, Normala)'$, $Yükleme = '(Çok\ Fazla, Normal)'$, $Birim\ zamanda\ işlenen\ paket\ miktarı = '(Normal, Çok-Çok\ Az)'$ varyasyonları ideal performansı hesaplamak için kullanılmıştır. Daha sonra, birleşim-kesişim çıkarım yöntemi ve bulanıklaştırma için ağırlık merkezi yöntemi uygulanarak gecikme, yükleme ve birim zamanda işlenen paket miktarı değerleri sırasıyla 0.0019 sn, 760 500 bit/sn ve 355 000 bit/sn sonuçları elde edilmiştir.

İkinci örnek girdi değerlerimiz; RTSED için 830 bayt, PED için 442 bayt ve AB için 9350 bit.

2. Örnek kural tabanı; $EĞER RTSED = '(Büyük, Çok-Büyük)'$ ve $PED = '(Çok-Küçük)'$ ve $AB = '(Az, Çok-Az)'$ İSE $gecikme = '(Çok-Çok-Fazla, Çok Fazla, Normal)'$, $Yükleme = '(Çok-Çok-Az, Çok-Fazla)'$, $Birim\ zamanda\ işlenen\ paket\ miktarı = '(Normal, Çok-Çok-Az)'$ varyasyonları ideal performansı hesaplamak için kullanılmıştır. Daha sonra, birleşim-kesişim çıkarım yöntemi ve bulanıklaştırma için ağırlık merkezi yöntemi uygulanarak gecikme, yükleme ve birim zamanda işlenen paket miktarı için sırasıyla 0,00217 sn, 752 000 bit/sn ve 387 000 bit/sn elde edilmiştir.

Daha önce Riverbed Modeler'de klasik yöntem testlerinden elde edilen örneklerimizi ve bulanık mantık tabanlı konfigürasyonları karşılaştırdığımızda bulanık mantık tabanlı performansın Şekil 4.18'de gösterildiği gibi önceki örneklerde elde edilen ideal değerlerden çok daha olumlu olduğu görülmektedir.



Şekil 4.18. 5 giriş ve 7 çıkış bölmeli parametreleri ile bulanık mantık tabanlı test sonuçları

Riverbed Modeler'de gerçekleştirilen WLAN ağ modelinde elde edilen ortalama gecikme süresi 0,0030 ile 0,0025 sn aralığında iken; örnek 1 ve 2'de uygulanan bulanık mantıkta elde edilen gecikme süreleri yaklaşık olarak 0,0019 sn ve 0,0021 sn aralığında yer

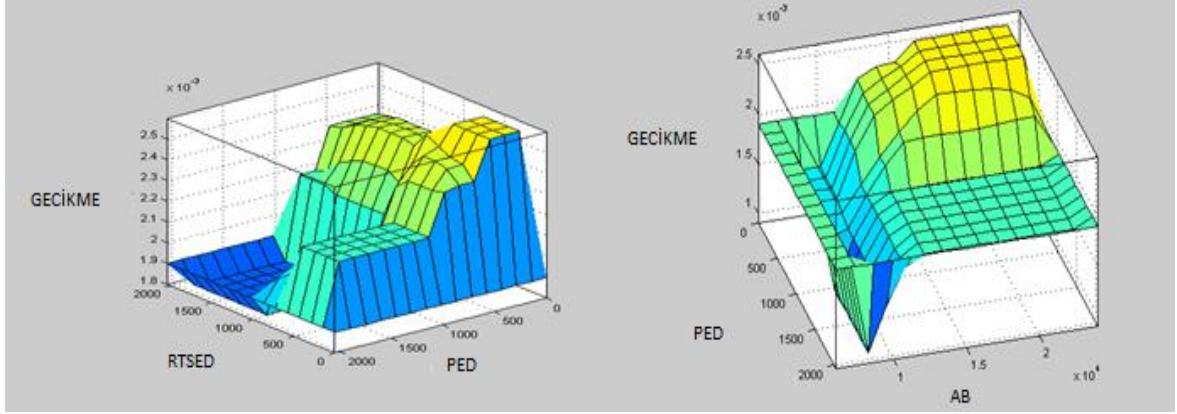
almaktadır. Gecikme süresindeki azalış oranı %16 ile %36 arasındadır. Yükleme süresi klasik test sonuçlarına göre 700 000 bit/sn – 730 000 bit/sn aralığında iken; örnek 1 ve 2'de uygulanan bulanık mantıkta elde edilen yükleme süreleri sırasıyla yaklaşık 760 500 bit/sn ve 752 000 bit/sn olmuştur. Yükleme süresindeki artış oranı %4 ile %8 arasındadır. Birim zamanda işlenen paket miktarı ise klasik test sonuçlarına göre 250 000 bit/sn – 350 000 bit/sn aralığında iken; örnek 1 ve 2'de uygulanan bulanık mantıkta elde edilen birim zamanda işlenen paket miktarı yaklaşık olarak 355 000 bit/sn – 387 000 bit/sn aralığındadır. Birim zamanda işlenen paket miktarı artış oranı %10 ile %40 arasında gerçekleşmiştir.

Giriş için 3, çıkış için 5 bölmeli üyelik ağırlığı ile giriş için 5, çıkış için 7 bölmeli üyelik ağırlığı ile yapılan bulanık mantık test sonuçları karşılaştırmaları Çizelge 4.3'te yer almaktadır.

Çizelge 4.3. Giriş ve çıkış üyelik ağırlıklarının farklı bölümlmelerine göre iyileştirme karşılaştırmaları.

Çıkış	Giriş için 3 Çıkış için 5 Bölmeli Üyelik Ağırlığı	Giriş için 5 Çıkış için 7 Bölmeli Üyelik Ağırlığı
Gecikme Süresi Azalma Oranı	28%-36%	16%-36%
Yükleme Süresi Artış Oranı	2%-10%	4%-8%
Birim Zamanda İşlenen Paket Miktarı Artış Oranı	25%-44%	10%-40%

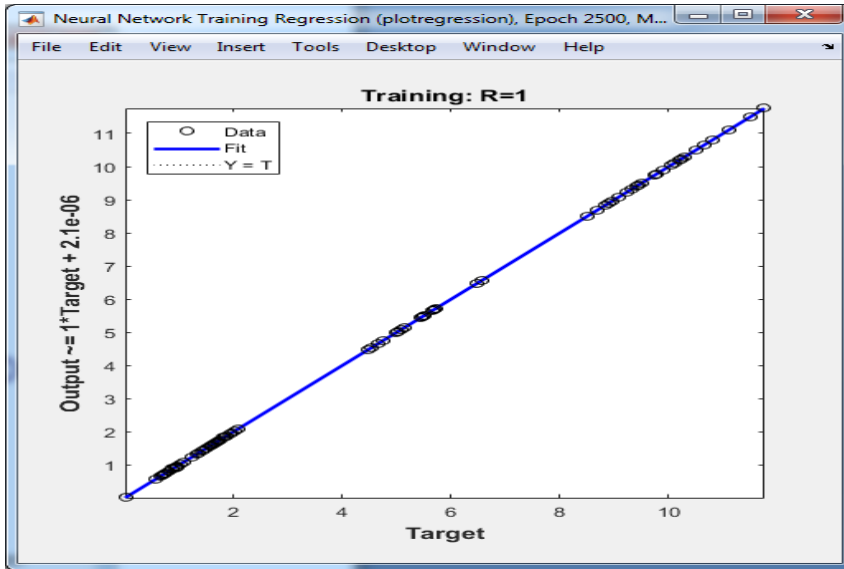
Yapılan test çalışmalarında elde edilen performans iyileştirmelerine ek olarak bulanık mantık tabanlı çalışmada (5 giriş 7 çıkışlı) RTSED, PED ve AB için Şekil 4.19'daki gibi performans ölçeği de çıkartılmıştır.



Şekil 4.19. Gecikme süresi performansının üç boyutlu genel gösterimi

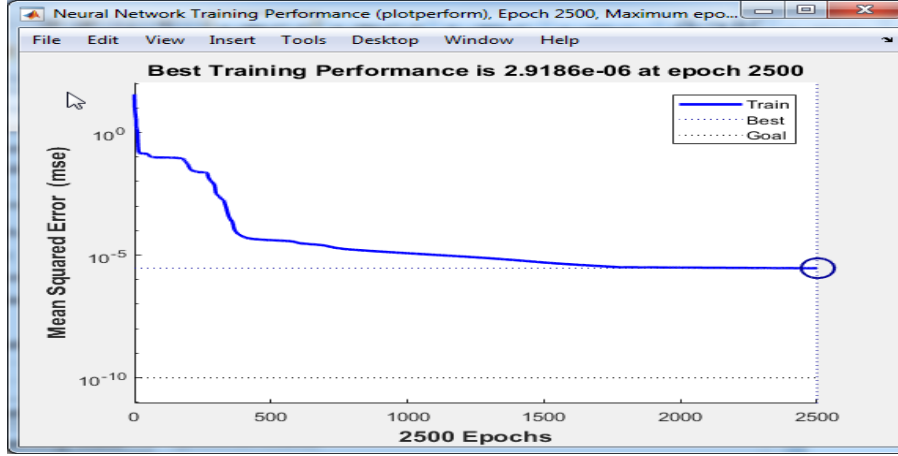
4.3. YSA Tabanlı Yöntem Analizi ile Performans Arttırımı

MATLAB’da gerçekleştirilen uygulama sonucunda YSA modelinin elde edilen eğitim sonucu Şekil 4.20’de gösterilmiştir. Grafiğe göre eğitim yüzdesi maksimumdur. Yapılan eğitim sonucunda 0,9999999 ($R=1$) sonucu; eğitimin kabul edilebilirliğini ve YSA’dan elde edilen değerler ile test verilerinden elde edilen değerlerin de örtüştüğünü göstermektedir.



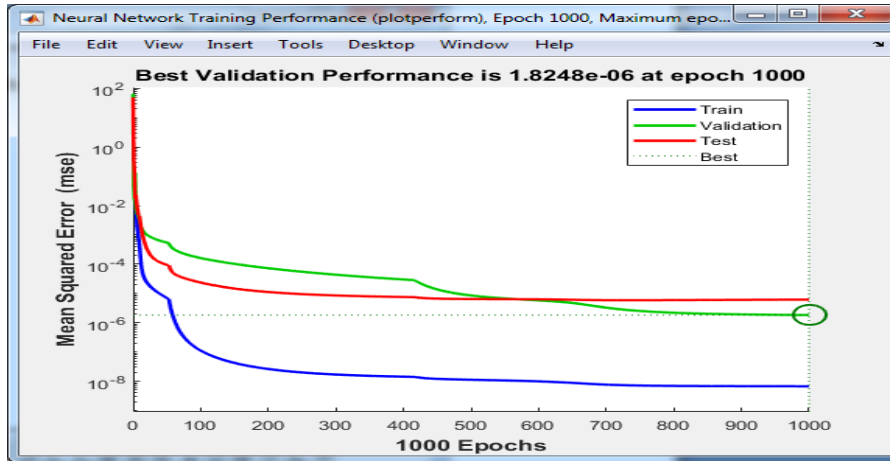
Şekil 4.20. YSA eğitim sonucu

Şekil 4.21, YSA'nın eğitim performansını göstermektedir. Eğitim performansı Riverbed Modeler benzetim aracı ile elde edilen gerçek çıkış değerleri arasındaki sapma ile bağlantılıdır. Eğitim performansının en iyi değerinin 10^{-5} olduğu görülmektedir.



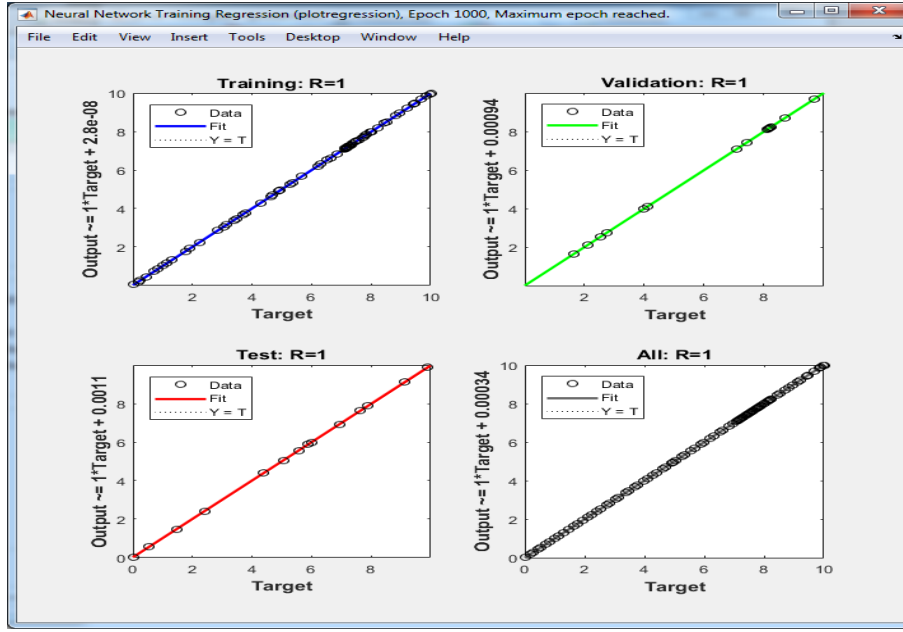
Şekil 4.21. YSA eğitim performansı

Şekil 4.22’de YSA modelinin eğitim, onaylama ve test performans karşılaştırması verilmiştir. Öğrenme performansının 10^{-6} ’dan daha küçük olduğu, ayrıca test ve tahmin yeteneklerinin 10^{-6} ’dan daha fazla olduğu gözlenmiştir. Sonuçlar, YSA modelinin ezber yapmadığını ve giriş ile çıkış verileri arasında bağlantı kurulabileceğini göstermektedir. Elde edilen sonuçlar, geliştirilen YSA modeli sayesinde en iyi öğrenme değerlerinin oluşturulduğu kanıtlanmıştır.



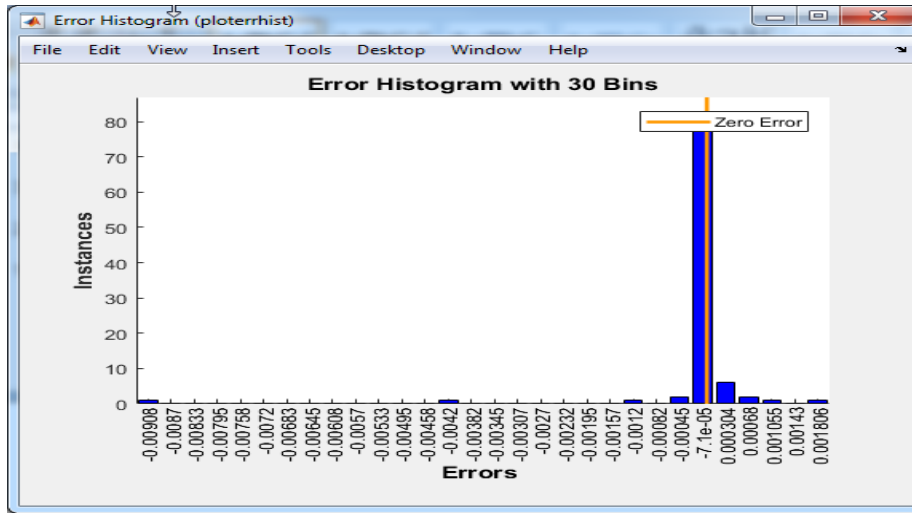
Şekil 4.22. YSA modelinde eğitim, onaylama ve test performans karşılaştırması

Şekil 4.23’te YSA’nın eğitim, test ve onaylama performans tahminlemesi görülmektedir. YSA’nın performansı, tahmini çıkış parametreler ile gerçek çıkış değerleri arasındaki sapma ve sapma oranı ile alakalıdır.



Şekil 4.23. YSA sonuçları: eğitim, onaylama ve test

Şekil 4.24, YSA'nın hata histogramını (30 Bins) göstermektedir. Grafikten $-7.1e^{-0.5}$ hata aralığında bir tahminleme yapıldığı görülmektedir.



Şekil 4.24. YSA hata histogramı-30 bins

İstatistiksel analiz sonuçlarına bakıldığında $RMS = 3,02E-05$, $R^2 = 0,999999999$ ve $MEP \% = 0,003845$ olarak hesaplanmıştır. Riverbed Modeler benzetim aracı kullanılarak yapılan test çalışmalarında elde edilen değerler ile YSA kullanılarak tahmin edilen alınan ve gönderilen veri trafiği karşılaştırıldığında iyi bir uyum ortaya çıkmıştır. Böylece YSA ile WLAN'larda performansın artırılması sağlanmıştır.

4.4. Farklı Ortam Değişkenleri ile Performans Karşılaştırmaları

İncelemiş olduğumuz giriş parametrelerine ek olarak dış faktörlerin performansa olan etkilerini incelemek amacıyla ortama farklı cihazlar (kablosuz sunucu, telecom vericisi ve jammer gibi) eklenmiştir. Eklenen cihazların alınan ve gönderilen veri trafikleri ile alıcı ve verici kanal kullanım durumlarını farklı oranlarda direkt etkilediği görülmüştür.

Kablosuz sunucu eklenmesi

Gerçekleştirilen çalışmamızda ilk olarak mevcut 11 adet düğümle aynı ortamda kalacak şekilde, içine ve dışına sırası ile 2 adet, 4 adet, 6 adet, 8 adet ve 10 adet kablosuz sunucu eklenmiştir. Kablosuz sunucu sayısı arttıkça kanalın kullanım durumu alıcı için artmaktadır. İlk kablosuz sunucu eklendiğinde kanal kullanımı verici için yaklaşık %10 oranında azalmaktadır. Ancak daha sonraki eklemelerde ise kablosuz sunucu sayısı arttıkça kanalın kullanım durumu artmaktadır. Kablosuz sunucu sayısı arttıkça alınan veri trafiği %0,5 ile %4 arasında azalmaktadır. Kablosuz sunucu dışarıda iken gönderilen veri trafiği azalmaktadır.

Telekom vericisi eklenmesi

Sonraki aşamada mevcut 11 adet düğümle ve 10 adet kablosuz sunucu ile aynı ortamda kalacak şekilde iç ve dış ortama sırası ile 2 adet, 4 adet, 6 adet, 8 adet ve 10 adet telecom vericisi eklenmiştir. Telekom vericisi eklendikçe kanalın kullanım durumu alıcı için azalmaktadır. Telekom vericisi eklendikçe kanalın kullanımı verici için %1 ile %12 arasında azalmaktadır. Telekom vericisi eklendikçe alınan veri trafiği, vericinin konumuna göre -%2 ile %2'ye kadar değişmektedir. Telekom vericisi eklendikçe gönderilen veri trafiği, vericinin konumuna göre %1 ile %17 arasında azalmaktadır.

Veri iletimi esnasında Şekil 4.25 ve Şekil 4.26'da gösterildiği gibi 10 adet telecom vericisi iç ve dış otamda konumlandırılarak vericilerin performansa olan etkileri incelenmiştir. Kablosuz sunucu ve telecom vericileri dışarda iken kanal kullanım durumu alıcı için artarken, içerde iken azalmaktadır. Telekom vericisi dışarda iken kanalın kullanım durumu verici için azalırken içeride iken verici için artmaktadır. Gönderilen veri trafiği, kablosuz

sunucu ierde iken kablosuz sunucu sayısı ile doęru orantılı olarak %1 ile %16 arasında artmaktadır.



Şekil 4.25. Ortam deęişkenlięi testi 10 adet kablosuz sunucu ve 10 adet telekom vericisi dıř ortam konumlandırması



Şekil 4.26. Ortam deęişkenlięi testi 10 adet kablosuz sunucu ve 10 adet telekom vericisi iç ortam konumlandırması

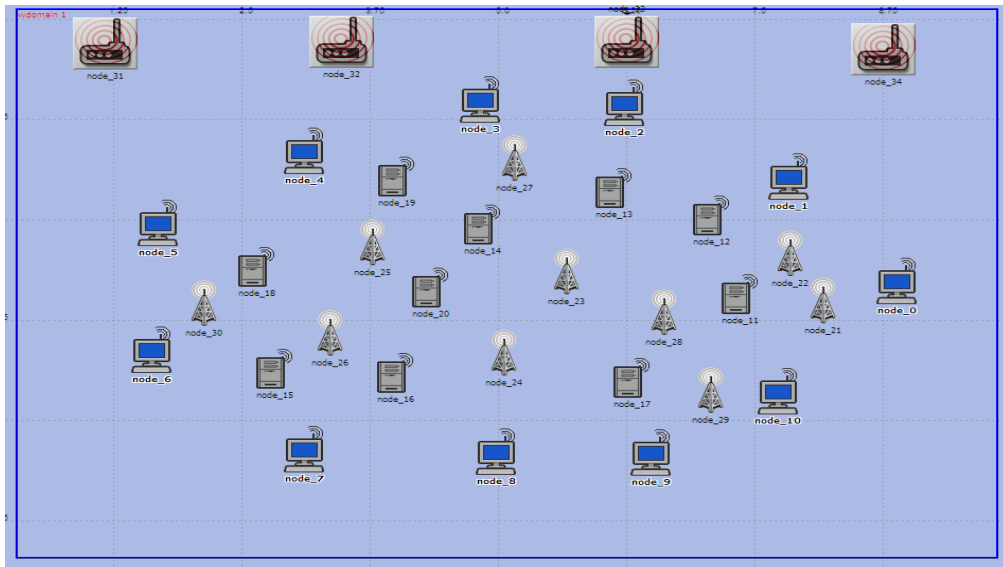
Son olarak da telekom vericileri'nin kablosuz sunucu olmadan performansa olan etkileri incelenmiştir. Telekom vericisi dıřarda iken alınan veri trafięi %3 ile %7 arasında artarken

içerde iken %4 e kadar azalmaktadır. Telekom vericisi dışarda iken gönderilen veri trafiği %2 ile %21 arasında azalırken, içeride iken %8 ile %27 arasında artmaktadır.

Jammer eklenmesi

Jammer, cep telefonu trafiği için çevresinde ölü bir bölge oluşturmaktadır. Telefon Jammer'ları cep telefonu sinyallerini engellemektedir. Sinema, Müze, tiyatro ve ibadethane gibi ortamlarda cep telefonunun kullanımını engellemek için de Jammer kullanılmaktadır. Ayrıca uzaktan kumandalı bombaların sinyalini kesmede de Jammer kullanılmaktadır.

Mevcut 11 adet düğümle ve 10 adet kablosuz sunucu ve 10 adet telekom vericisi ile aynı ortamda kalacak şekilde iç ve dış ortama sırası ile 1 adet, 2 adet, 3 adet ve 4 adet Jammer eklenmiştir. Şekil 4.27'de gösterildiği gibi bütün kablosuz sunucular ve telekom vericileri iç bölgede konumlandırılmıştır. Jammer eklendiğinde alıcı kanal kullanım durumunda %1 ile %3 arasında azalma meydana gelmektedir. Ayrıca verici kanal kullanım durumunda ise %3 ile %11 arasında azalma meydana gelmektedir. Jammer eklendiğinde alınan veri trafiği %1 ile %3 arasında azalmaktadır. Gönderilen veri trafiğinde ise jammer konumuna bağlı olarak orantısız azalma meydana gelmektedir.



Şekil 4.27. Ortam değişkenliği testi 10 adet kablosuz sunucu ve 10 adet telekom vericisi iç ortam konumlandırması ve 4 adet jammer eklenmesi

Sonuç olarak ortam değişkenleri değiştirildiğinde çıkış parametreleri için Çizelge 4.4'te yer alan farklılıklar meydana gelmiştir.

Çizelge 4.4. Farklı ortam değişkenlerinin çıkış parametresine etkisi

Kanal Kullanım Durumu Alıcı
Kablosuz sunucu sayısı arttıkça kanalın kullanım durumu genel olarak artmaktadır.
Telekom vericisi eklendikçe kanalın kullanım durumu genel olarak azalmaktadır.
Kablosuz sunucu ve telekom vericileri dışarda iken kanal kullanım durumu genel olarak artarken, içerde iken genel olarak azalmaktadır.
Jammer eklendiğinde %1 ile %3 arasında azalma meydana gelmektedir.
Kablosuz sunucu sayısı azaldıkça kanalın kullanım durumu genel olarak azalmaktadır.
Kanal Kullanım Durumu Verici
İlk kablosuz sunucu eklemede kanal kullanımı yaklaşık %10 oranında azalmaktadır. Sonraki eklemelerde, kablosuz sunucu sayısı arttıkça kanalın kullanım durumu artmaktadır.
Telekom vericisi eklendikçe kanalın kullanım %1 ile %12 arasında azalmaktadır.
Telekom vericisi dışarda iken kanalın kullanım durumu azalırken, içerde iken artmaktadır.
Jammer eklendiğinde %3 ile %11 arasında azalma meydana gelmektedir.
Kablosuz sunucu sayısı düğümlerin iç ortam konumunda arttıkça kanalın kullanım durumu azalmaktadır.
Alınan Veri Trafikçi
Kablosuz sunucu sayısı arttıkça alınan veri trafikçi %0,5 ile %4 oranında azalmaktadır.
Telekom vericisi eklendikçe alınan veri trafikçi, vericinin konumuna göre %2 kadar artmakta veya %2 ye kadar azalmaktadır.
Telekom vericisi dışarda iken alınan veri trafikçi %3 ile %7 arasında artarken, içerde iken %4 e kadar azalmaktadır.
Jammer eklendiğinde alınan veri trafikçi %1 ile %3 arasında azalmaktadır.
Kablosuz sunucu sayısı azaldıkça alınan veri trafikçi %4'e kadar azalmaktadır.

Çizelge 4.4. Farklı ortam değişkenlerinin çıkış parametresine etkisi-devam

Gönderilen Veri Trafikçi
Kablosuz sunucu dışarıda iken gönderilen veri trafiği azalmaktadır. Kablosuz sunucu içerde iken, kablosuz sunucu sayısı gönderilen veri trafiği ile doğru orantılı olarak %1 ile %16 oranında artmaktadır.
Telekom vericisi eklendikçe gönderilen veri trafiği, vericinin konumuna göre %1 ile %17 arasında azalmaktadır.
Telekom vericisi dışarda iken gönderilen veri trafiği %2 ile %21 arasında azalırken, içerde iken %8 ile %27 arasında artmaktadır.
Jammer eklendiğinde gönderilen veri trafiği Jammer konumuna göre genel olarak azalmaktadır.
Kablosuz sunucu sayısı azaldıkça da gönderilen veri trafiği, konumuna göre %13'e kadar artmaktadır.

4.5. Geri Besleme Kontrolü ile Kanal Verimliliğinin Arttırılması

Geri besleme kontrolü ile kanal verimliliğinin arttırılması için yapılan çalışmada, ilk olarak WLAN'larda CSMA/CA modelinde veri iletimi esnasında kanal kullanım verimliliğinin döngü tablosu ile artışı amaçlanmaktadır. Çıkış parametreleri olarak kullanılacak olan kanal kullanım durumunun maximum olması hedeflenmektedir. İkinci gerçekleştirilen çalışmada ise bulanık mantık tabanlı geri besleme kontrolü ile benzetim anında düğüm bazlı birim zamanda işlenen paket miktarının arttırılması sağlanmıştır.

Uygulanan modelde giriş parametreleri başlangıçta varsayılan değerlerinde iken; her koşum sonrası koda gömülü olan çıkış parametresi ile kontrol edilmekte ve ideal değerinde değil ise artarak döngüyü tekrar etmektedir.

Yapılan test çalışmaları sonucunda konsol (console) ekranında geri besleme kontrol yöntemi ile elde edilen ideal birim zamanda işlenen paket miktarları değerleri sunulmuştur. Evrensel kümede benzetim, 1536 (2048*512) farklı konfigürasyonla birer saniye aralıklarla koşulmuştur. Koşum esnasında *Op_sim_time* fonksiyonu kullanılmıştır. Bu aralığın tamamının test edilebilmesi her bir saniye için değerlerin koşulacağı hesap edildiğinde, 25.6 (1536/60) dakika koşum yapılması gerekmektedir. Koşum tamamlandığında ideal

fonksiyon deęerleri ortaya ıkartılmıřtır. Birim zamanda iřlenen paket miktarı iin ideal deęer dm_4 iin 11992 olarak elde edilmiřtir.

4.5.1. D tablolu kaba kuvvet algoritması ile performans arttırımı

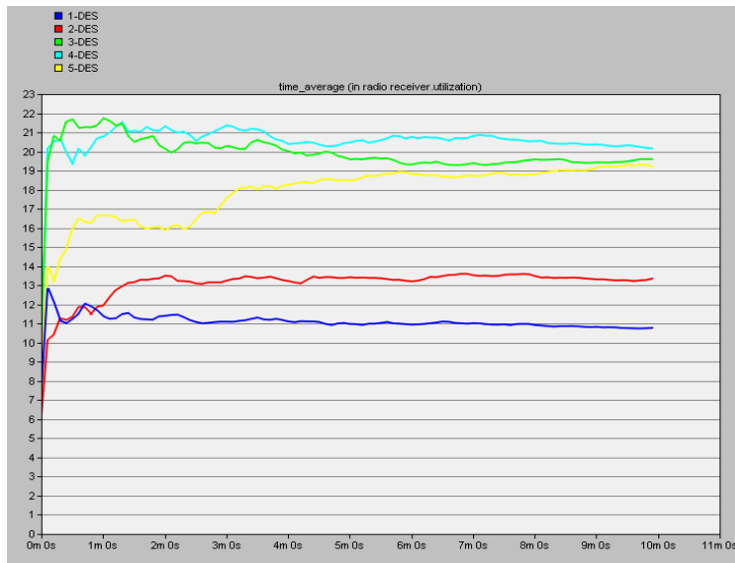
Daha nceden 2 362 543 000 (İki milyar z altmıř  bin beř yz kırk ) test yapılarak elde edilen alıcı kanal kullanım durumu, ajanlar aracılıęıyla proses modeli ierisinde yer alan *Wlan_mac.pr.m* kod blok yapısında i ie dler kullanılarak test edilmiřtir. Her d hata ayıklama ekranından takip edilmiřtir ve sonu olarak kod ierisinde alıcı kanal kullanım durumunu arttırmak amacı ile dinamik bir yapı kurulmuřtur. Modelde sistem artık ideal giriř parametreleri ile alıřmaktadır.

Kanal kullanım durumu iin evrensel kmede %5,83 ile %20,28 arasındaki veriler seilmiřtir. Giriř parametrelerinde; RTSED ve PED 600 bayt ve 1200 bayt aralıęında, AB ise 9000 bit ile 15 000 bit aralıęında seilmiřtir. Kanal kullanım durumunu lmek amacı ile RTSED ve PED deęerleri 600 bayt, 700 bayt, 900 bayt, 1000 bayt ve 1200 bayt deęerleri ile test edilirken, AB 9000 bit, 10 000 bit, 12 000 bit, 13 000 bit ve 15 000 bit deęerleri ile test edilmiřtir. Test ařamasında kanal verimlilięini arttırmak iin evrensel kmede yer alan her deęer test edilmiřtir.

WLAN'larda nceki alıřmalarda performans geliřtirmek amacıyla Riverbed Modeler zerinden elde edilen veri setleri giriř deęiřkenleri iin farklı algoritmalarla optimize edilmiřtir. Bu ařamada yapmıř olduęumuz alıřmada ise RTSED, PED ve AB giriř deęerleri proses katmanında *Wlan_mac.pr.m* kod bloęunda ajan yapısı ile dinamik olarak gncellenmiřtir. Bu dinamik yapıda kaba kuvvet algoritması kullanılmıřtır.

Proses katmanında her prosesin ierisinde dler ve fonksiyonlar bulunmaktadır ve kullanıcılara Riverbed Modeler geliřtirme yapma seeneęi sunmaktadır. Burada Riverbed Modeler kendi ierisinde yer alan hazır fonksiyonlarla birok protokol kořmaktadır. Riverbed Modeler ıkıř fonksiyonlarında yer alan kanal verimlilięini gerek zamanlı grntlemeye izin vermemektedir. Zira kanalın verimlilięi prosesin tamamı bittikten sonra ortalama deęer olarak grntlenmektedir. Bu nedenle sistemimizde yer alan dsel tabloda 125 karaktere sahip kanal kullanım durumu kod blok yapısında test edilmiřtir.

Uygulanan yeni yöntemle birlikte dinamik olarak giriş değişkenleri kendini güncelleyebilmekte ve en ideal kanal kullanım durumunu ortaya çıkarmaktadır. Örnek olarak döngü tablosunda çıkış değeri için en ideal değer %20,28 döngü tablosunda dördüncü aşamada elde edilmiştir. Kaba kuvvet algoritması ilk dört giriş değerlerinde ideal durumunu güncellemiş ve sonrasında ise ideal değeri kazandıktan sonra güncelleme yapmamıştır. Test sonuçları karşılaştırmalarına bakıldığında en ideal kanal kullanım durumunun %20,28 ile dördüncü test giriş değerlerinde olduğu Şekil 4.28’de görülmektedir.



Şekil 4.28. Döngü tablosu tabanlı geri besleme kontrolü performans sonuçları

Gerçekleştirilen çalışma ile WLAN’larda döngü tablosu ile ideal birim zamanda işlenen paket miktarı için giriş değerleri ataması en hızlı yolla yapılarak benzetim esnasında zamandan kazanç sağlanmıştır.

4.5.2. Bulanık mantık tabanlı geri besleme kontrolü ile düğüm bazlı birim zamanda işlenen paket miktarının benzetim anında optimize edilmesi

Yapılan test çalışmalarının sonucunda düğüm bazlı birim zamanda işlenen paket miktarı değerlerinin tespiti için 1. örnek girdi değerlerimiz; RTSED için 689 bayt, PED için 706 bayt ve AB için 10 004 bit kullanılmış ve birleşim-kesişim çıkarım metodu ile öncelikli olarak örnek girdilerin bulunmuş olduğu küme aralığındaki ağırlıkları tespit edilmiştir.

1. Örnek kural tabanı; $E\check{G}ER$ $RTSED = '(Küçük)'$, $PED = '(Normal, Büyük)'$ ve $AB = '(Normal, Çok)'$ İSE Birim zamanda işlenen paket miktarı = $'(Çok Fazla)'$

Varyasyonları kullanılarak bu aralıklardaki oranların en düşük olanına göre işletilmiştir. Daha sonra çıkış üyelik fonksiyonlarında yer alan değerlerin çıkarım işleminden sonra ağırlık merkezi yöntemi ile durulaştırılmasıyla, birim zamanda işlenen paket miktarları için ideal olan paket miktarı 89 965 bit/sn olarak elde edilmiştir.

2. Örnek girdi değerlerimiz; $RTSED$ için 684 bayt, PED için 665 bayt ve AB için 9903 bit kullanılmış ve birleşim-kesişim çıkarım metodu ile öncelikli olarak örnek girdilerin bulunmuş olduğu küme aralığındaki ağırlıkları tespit edilmiştir.

2. Örnek kural tabanı; $E\check{G}ER$ $RTSED = '(Küçük, Normal)'$ ve $PED = '(Küçük, Normal)'$ ve $AB = '(Normal, Az)'$ İSE Birim zamanda işlenen paket miktarı = $'(Fazla, Çok Fazla)'$

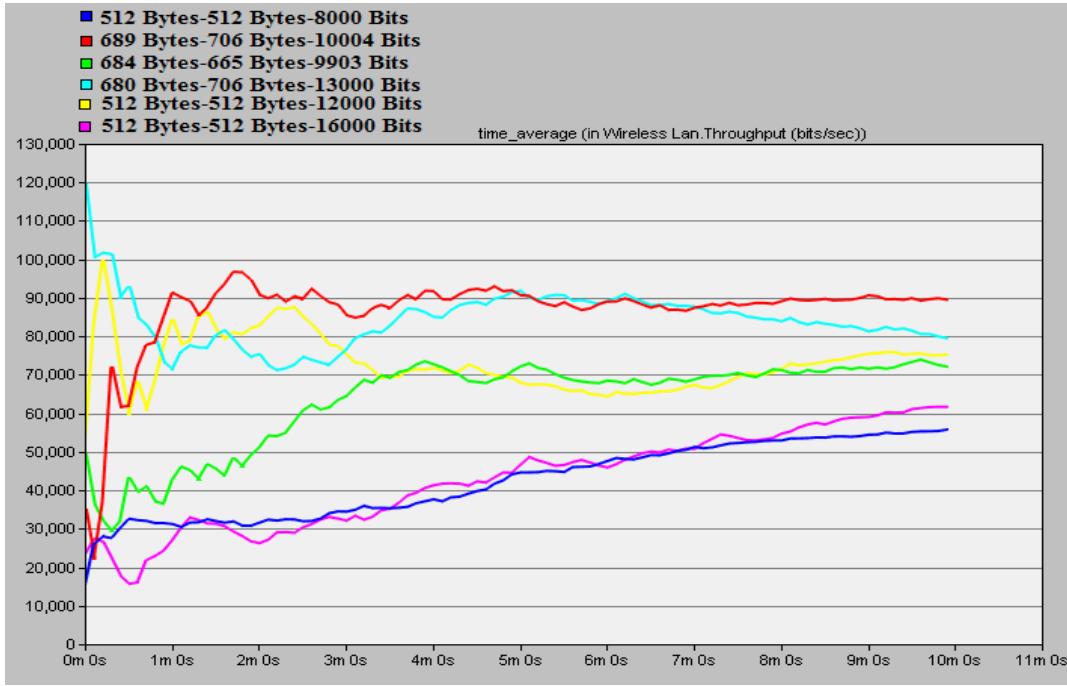
Varyasyonları kullanılarak bu aralıklardaki oranların en düşük olanına göre işletilmiştir. Daha sonra çıkış üyelik fonksiyonlarında yer alan değerlerin çıkarım işleminden sonra ağırlık merkezi yöntemi ile durulaştırılmasıyla, ideal olan düğüm bazlı birim zamanda işlenen paket miktarı 72 615 bit/sn olarak ölçülmüştür.

3. Örnek girdi değerlerimiz; $RTSED$ için 680 bayt, PED için 706 bayt ve AB için 13 000 bit kullanılmış ve birleşim-kesişim çıkarım metodu ile öncelikli olarak örnek girdilerin bulunmuş olduğu küme aralığındaki ağırlıkları tespit edilmiştir.

3. Örnek kural tabanı; $E\check{G}ER$ $RTSED = '(Küçük, Normal)'$ ve $PED = '(Normal, Büyük)'$ ve $AB = '(Normal, Çok)'$ İSE Birim zamanda işlenen paket miktarı = $'(Fazla, Çok Fazla)'$

Varyasyonları kullanılarak bu aralıklardaki oranların en düşük olanına göre işletilmiştir. Daha sonra çıkış üyelik fonksiyonlarında yer alan değerlerin çıkarım işleminden sonra ağırlık merkezi yöntemi ile durulaştırılmasıyla, ideal olan düğüm bazlı birim zamanda işlenen paket miktarı 79 915 bit/sn olarak ölçülmüştür. Elde edilen ideal değerler farklı giriş değerleri ile karşılaştırıldığında Şekil 4.29'da görüldüğü gibi örnek girdilerimizin oluşturmuş olduğu çalışmaların performanslarının çok daha pozitif olduğu görülmektedir.

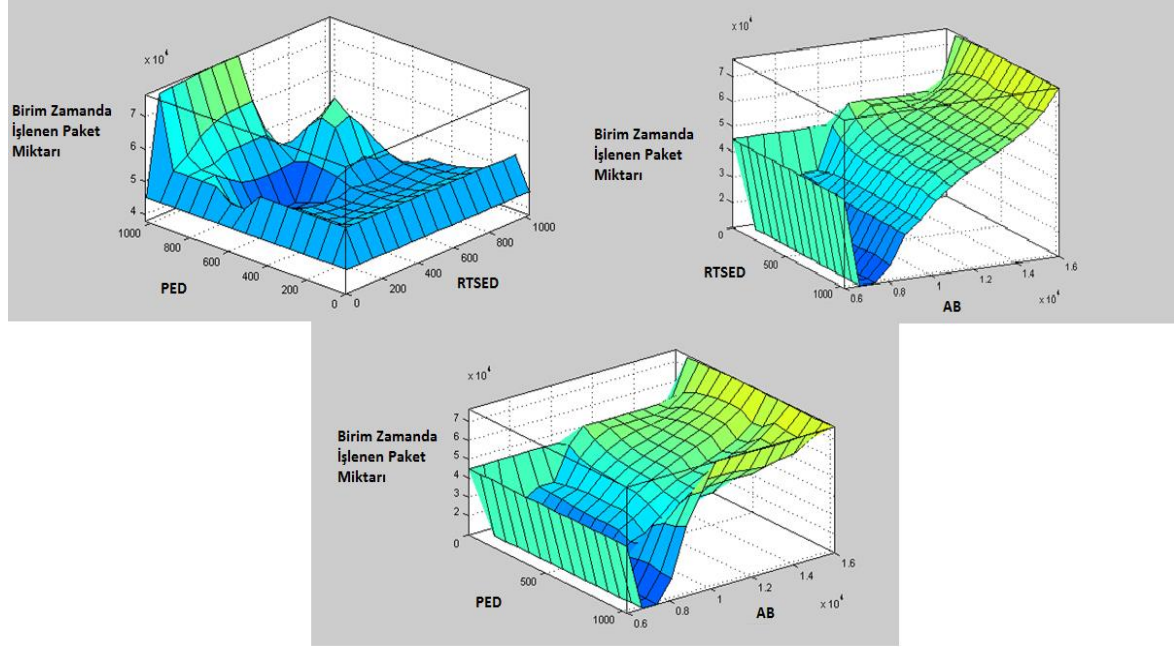
Riverbed Modeller’de gerçekleştirilen standart WLAN ağ modelinde düğüm bazlı birim zamanda işlenen paket miktarı değerleri normalde 0 bit ile 65 000 bit aralığında iken, geri besleme kontrolü ile elde edilen ve sisteme uygulanan düğüm bazlı birim zamanda işlenen paket miktarları, örnek 1, örnek 2 ve örnek 3’te 89 965 bit/sn, 72 615 bit/sn ve 79 915 bit/sn olarak ölçülmüştür. Burada düğüm bazlı birim zamanda işlenen paket miktarı değerlerinde ortalama %20 oranında iyileştirme sağlanmıştır.



Şekil 4.29. Geri besleme kontrollü bulanık mantık sonuçlarının Riverbed Modeller’de uygulanması

Şekil 4.30’da performansların üç boyutlu olarak karşılaştırılmasında ortaya çıkan sonuçlarda RTSED ile PED değerlerini karşılaştırıldığında; RTSED’nin 600 bayta kadar artması ile birim zamanda işlenen paket miktarı sabit kalırken, 600 bayt sonrası dramatik bir düşüş ve sonrasında ise tekrardan 800 baytta yerel bir zirve noktasının olduğu görülmüştür. RTSED’nin artması durumunda ise birim zamanda işlenen paket miktarı düzensiz de olsa artmıştır. AB ile RTSED’nin karşılaştırıldığında; AB’nin artışı ile düğüm bazlı birim zamanda işlenen paket miktarının arttığı, RTSED’ni artışı ile ise birim zamanda işlenen paket miktarının değişmediği görülmüştür. AB ve PED üç boyutlu karşılaştırmasında; benzer şekilde AB’nin artışı ile düğüm bazlı birim zamanda işlenen paket miktarının arttığı, PED’in artışı ile ise birim zamanda işlenen paket miktarının değişmediği görülmüştür. Yapılan çalışma ile WLAN’larda DCF kontrol fonksiyonlarında CSMA/CA yönteminde kullanılan RTSED, PED ve AB’nin performansı orantılı olarak

etkilemediği ortaya konulmuştur. Ancak en ideal performans bulanık mantık tabanlı geri besleme kontrolü ile elde edilmiştir.



Şekil 4.30. Düğüm bazlı birim zamanda işlenen paket miktarı sonuçları üç boyutlu gösterimi

4.6. Farklı Düğüm Sayıları, Varışlar Arası Süre ve İletim Gücü ile Önerilen Model Performans Karşılaştırmaları ve Test Sonuçları

Bu bölümde bulanık mantık tabanlı geri besleme yöntemiyle arttırılan birim zamanda işlenen paket miktarına ek olarak dış parametrelerin performansa etkisi incelenmiştir. Yapılan test çalışmalarında düğüm sayılarının değişkenliğine göre üç farklı test ortamı Çizelge 4.5'teki gibi oluşturulmuştur. Benzetim ortamlarında dış giriş parametreleri olarak düğüm sayısı, iletim gücü (W) ve varışlar arası süre (saniye-üstel) parametreleri kullanılmıştır. Test ortamı olarak ofis ortamı, laboratuvar ortamı ve konferans salonu ortamları seçilmiştir. Ofis ortamı için 413 000 213 test, laboratuvar ortamı için 1 341 108 083 test ve konferans salonu ortamı için 4 511 261 878 test koşulmuştur.

Çizelge 4.5. Kablosuz yerel alan ağları performans karşılaştırma ölçütleri

Uygulama Alanı	Düğüm Sayısı	İletim Gücü (W)	Varışlar Arası Süre (saniye-üstel)	Test Sayısı
Ofis Ortamı	5-7-10	0.005-0.02-0.05-0.1	0.02-0.04-0.06-0.08	413 000 213
Laboratuvar Ortamı	10-15-20	0.005-0.02-0.05-0.1	0.02-0.04-0.06-0.08	1 341 108 083
Konferans Salonu Ortamı	20-30-40	0.005-0.02-0.05-0.1	0.02-0.04-0.06-0.08	4 511 261 878

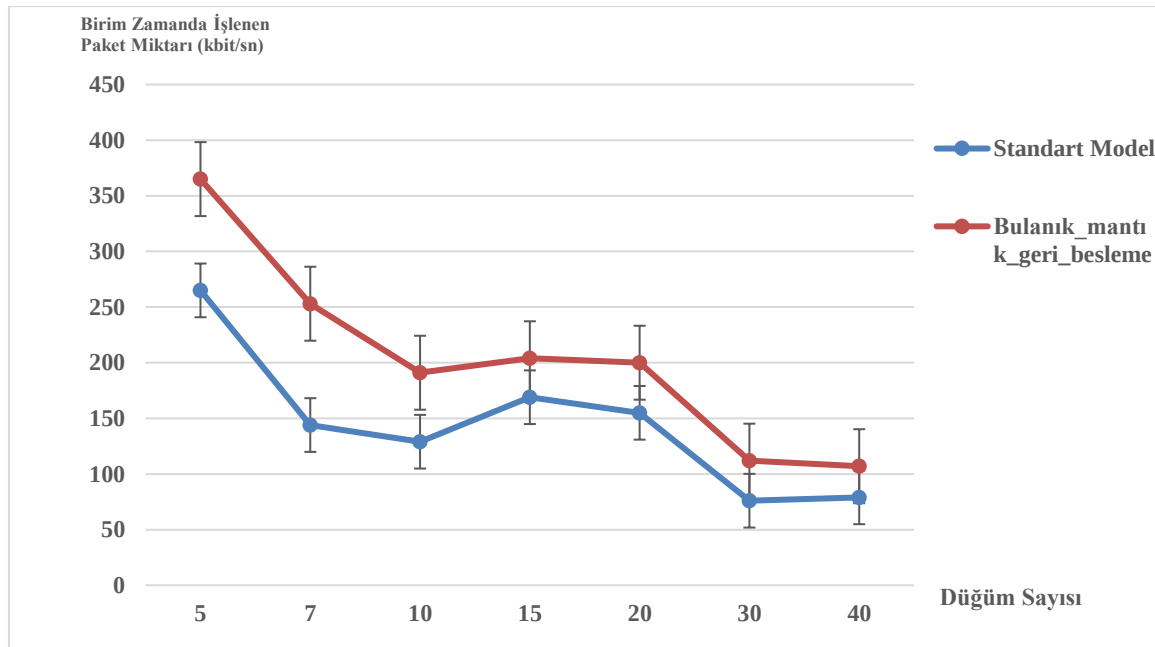
Koşulan testlerde birim zamanda işlenen paket miktarı, kanal kullanım oranı, alınan veri trafiği ve gönderilen veri trafikleri çıkış parametreleri incelenmiştir. Yapılan test çalışmalarında, RTSED, PED ve AB değerleri bulanık mantık algoritması ile tekrar koşulmuştur. Gerçekleştirilen testlerin sonucunda tez çalışmasının çalışma sahası genişletilmiş ve dış parametreler de hesaplanarak daha genel ve güvenli bir performans ölçeği çıkartılmıştır.

Yapılan çalışmalarda genelleme yapıldığında; düğüm sayısı artarken birim zamanda işlenen paket miktarında düzenli bir azalış görülmüştür. İletim gücü artarken birim zamanda işlenen paket miktarı da düzenli olarak artmıştır. Varışlar arası süre artarken birim zamanda işlenen paket miktarı da düzenli olarak azalmıştır. Düğüm sayısı ve iletim gücü artarken, birim zamanda işlenen paket miktarı da düzensiz olarak azalmıştır. Düğüm sayısı ve varışlar arası süre artarken, birim zamanda işlenen paket miktarı da düzenli olarak azalmıştır. İletim gücü ve varışlar arası süre artarken birim zamanda işlenen paket miktarı da düzenli olarak azalmıştır. Düğüm sayısı ve iletim gücü ve varışlar arası süre artarken birim zamanda işlenen paket miktarı da düzenli olarak azalmıştır. Genel olarak varışlar arası süre baskın ters orantılıdır. Düğüm sayısı da baskın olmayacak şekilde ters orantılıdır. İletim gücü baskın olmayacak şekilde doğru orantılıdır.

Yapılan çalışmaların tamamında RSTED, PED ve AB değerleri bulanık mantıkla çalıştırılmış ve çıkış parametreleri için ayrı ayrı grafik çıkartılarak performans analizi gerçekleştirilmiştir.

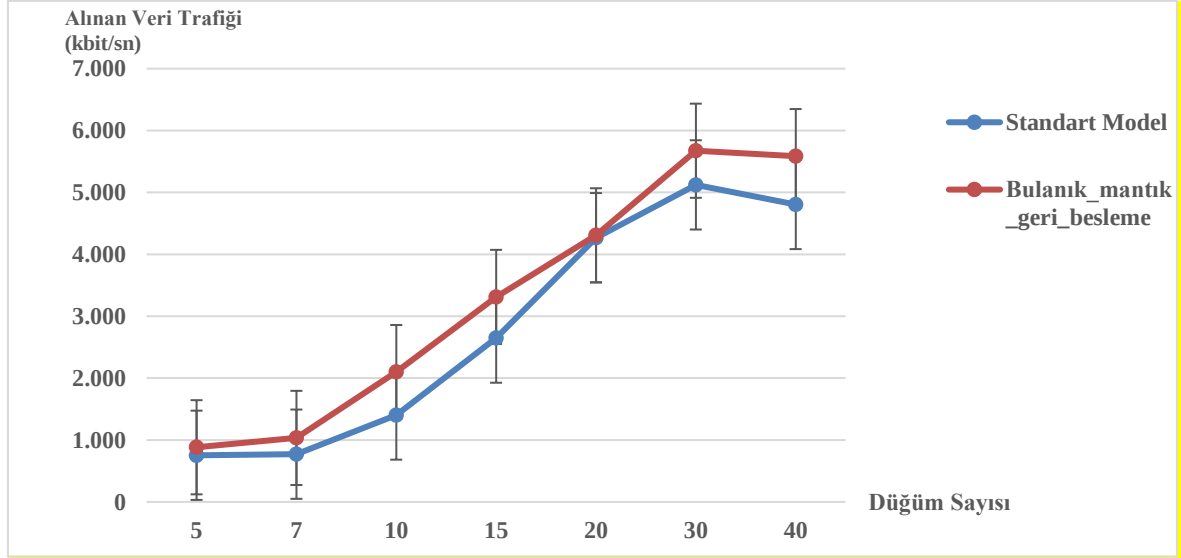
Farklı D ğ m Sayıları ile Performans Analizi

Yapılan test alıřmalarında kurulan   farklı ortamda d ğ m sayıları; ofis ortamı iin 5-7-10, laboratuvar ortamı iin 10-15-20 ve konferans salonu iin ise 20-30-40 deėerleri ile test edilmiřtir. Yeni y ntemde birim zamanda iřlenen paket miktarı, alınan ve g nderilen veri trafiėinde belirgin bir řekilde artıř g zlemlenmiřtir. řekil 4.31'de g r len deėerler; standart y ntemde birim zamanda iřlenen paket miktarları d ğ m sayılarına g re sırasıyla 265, 144, 129, 169, 155, 76 ve 79 kbit/sn iken, bulanık mantık tabanlı geri besleme kontroll  y ntemde kullanılan algoritmayla sırasıyla 365, 253, 191, 204, 200, 112 ve 107 kbit/sn olarak  l lm řtir.



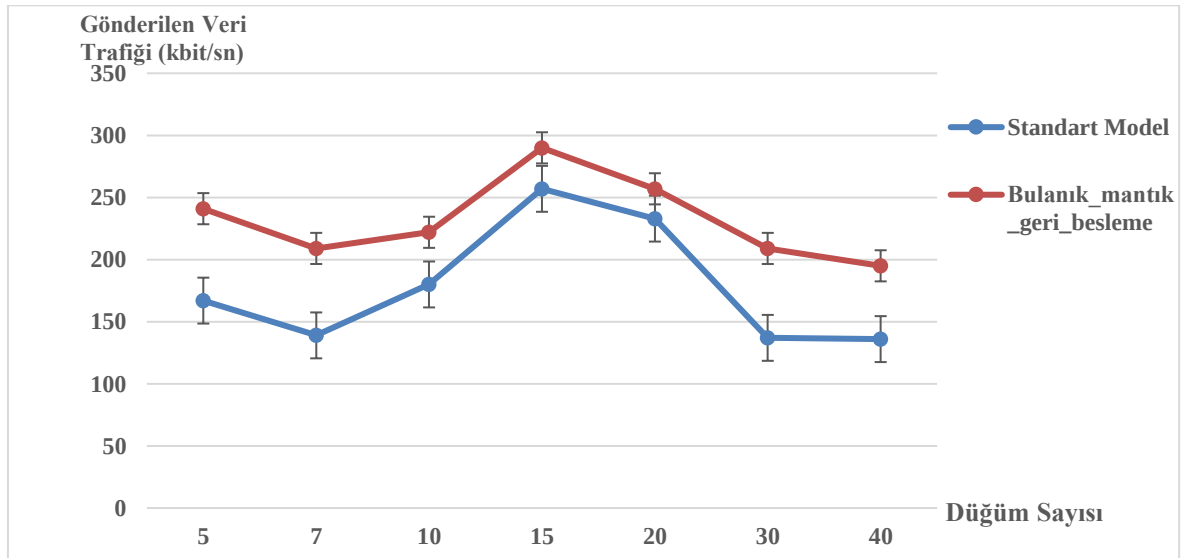
řekil 4.31. Yeni modelde farklı d ğ m sayıları ile d ğ m bazlı birim zamanda iřlenen paket miktarı performans karřılařtırması

řekil 4.32'de g r len deėerler; standart y ntemde alınan veri trafiėi d ğ m sayılarına g re sırasıyla 753, 771, 1404, 2647, 4269, 5122 ve 4806 kbit/sn iken, bulanık mantık tabanlı geri besleme kontroll  y ntemde sırasıyla 883, 1034, 2098, 3312, 4307, 5674 ve 5587 kbit/sn olarak  l lm řtir.



Şekil 4.32. Yeni modelde farklı düğüm sayıları ile alınan veri trafiği performans karşılaştırması

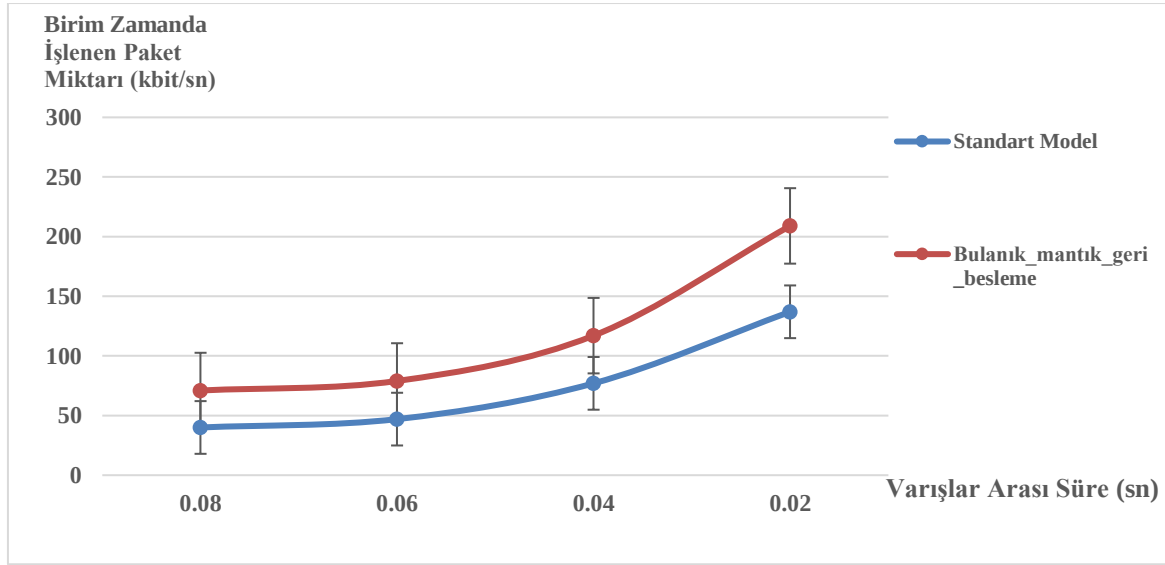
Şekil 4.33'te görülen değerler; standart yöntemde gönderilen veri trafiği düğüm sayılarına göre sırasıyla 167, 139, 180, 257, 233, 137 ve 136 kbit/sn iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla 241, 209, 222, 290, 257, 209 ve 195 kbit/sn olarak ölçülmüştür.



Şekil 4.33. Yeni modelde farklı düğüm sayıları ile gönderilen veri trafiği performans karşılaştırması

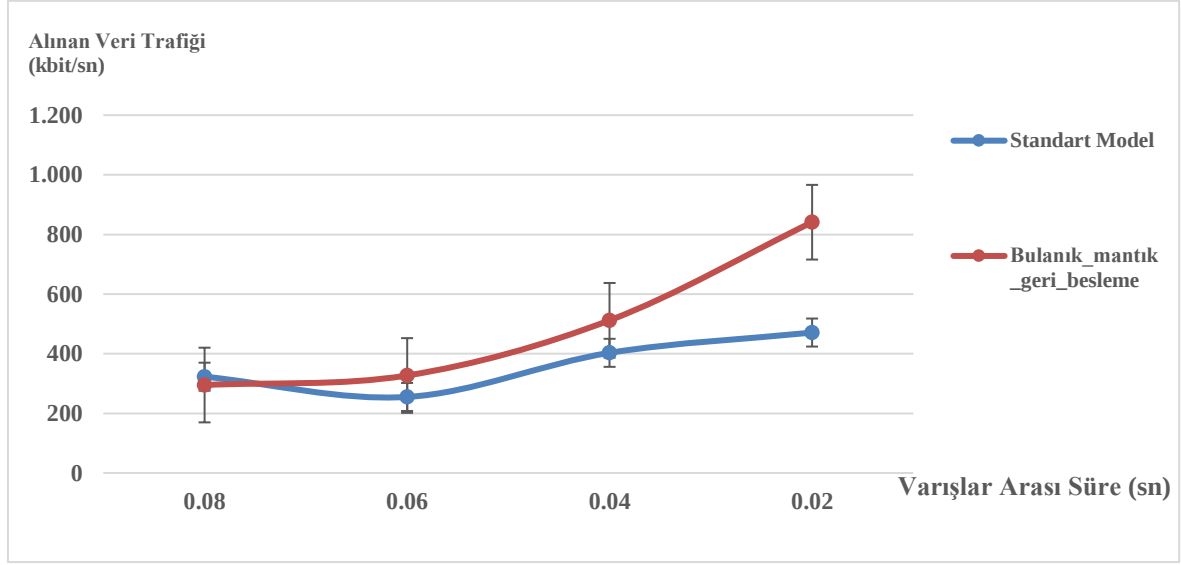
Farklı Varışlar Arası Süreleri ile Performans Analizi

Varışlar arası süre 0.02-0.04-0.06-0.08 değerleri ile test edilmiştir. Şekil 4.34'te görülen değerler; standart yöntemde birim zamanda işlenen paket miktarları varışlar arası sürelerine göre tersten 0.08-0.06-0.04-0.02 için sırasıyla 40, 47, 77 ve 137 kbit/sn iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla 71, 79, 117 ve 209 kbit/sn olarak ölçülmüştür.



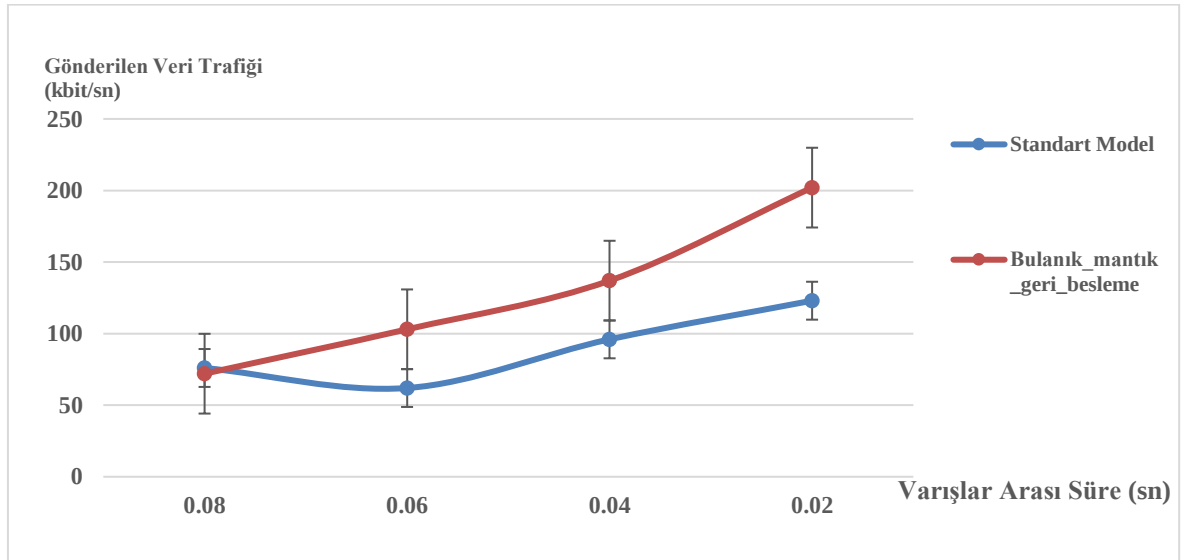
Şekil 4.34. Yeni modelde farklı varışlar arası süreler ile bazlı birim zamanda işlenen paket miktarı performans karşılaştırması

Şekil 4.35'te görülen değerler; standart yöntemde alınan veri trafiği varışlar arası sürelerine göre tersten 0.08-0.06-0.04-0.02 için sırasıyla 323, 255, 403 ve 471 kbit/sn iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla 295, 327, 512 ve 841 kbit/sn olarak ölçülmüştür.



Şekil 4.35. Yeni modelde farklı varışlar arası süreler ile alınan veri trafiği performans karşılaştırması

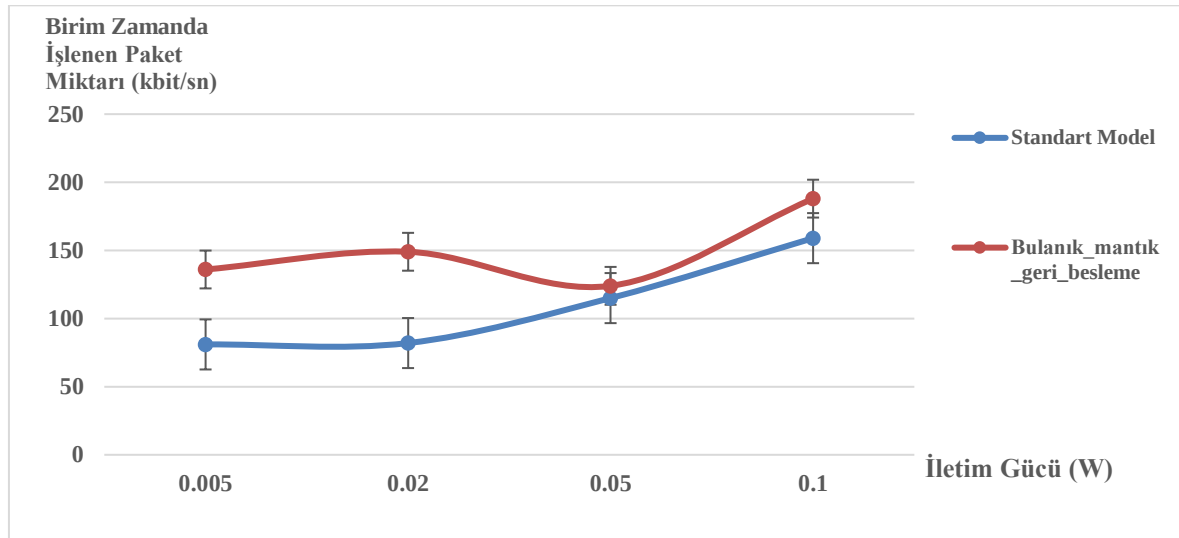
Şekil 4.36'da görülen değerler; standart yöntemde gönderilen veri trafiği varışlar arası süreler göre tersten 0.08-0.06-0.04-0.02 için sırasıyla 76, 62, 96 ve 123 kbit/sn iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla 72, 103, 137 ve 202 kbit/sn olarak ölçülmüştür.



Şekil 4.36. Yeni modelde farklı varışlar arası süreler ile gönderilen veri trafiği performans karşılaştırması

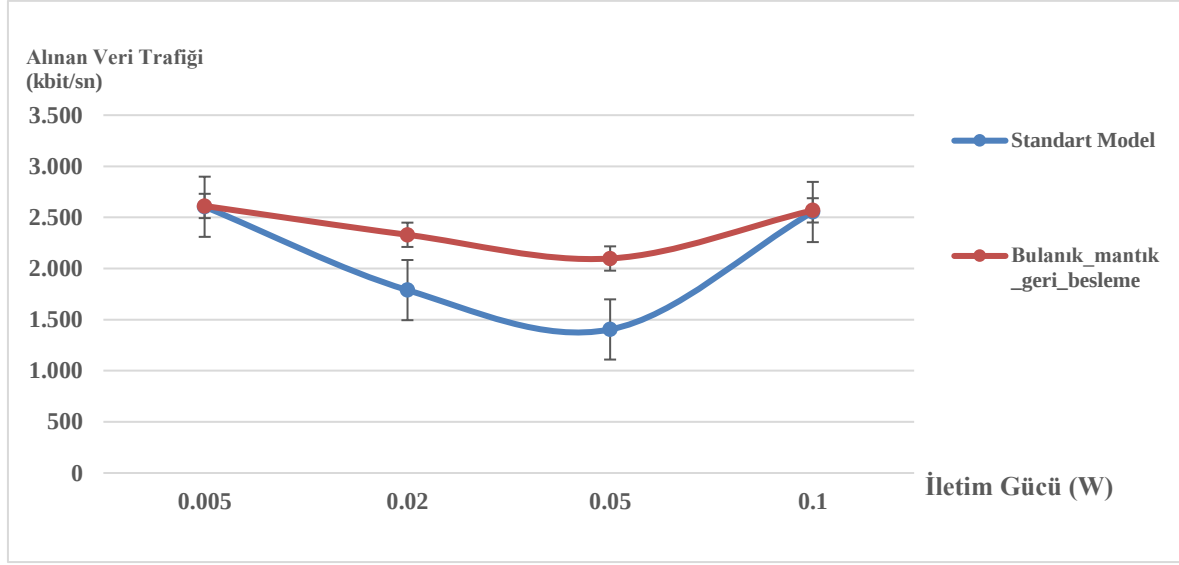
Farklı İletim Gücü ile Performans Analizi

İletim gücü 0.005-0.02-0.05-0.1 değerleri ile test edilmiştir. Şekil 4.37’de görülen değerler; standart yöntemde birim zamanda işlenen paket miktarları iletim gücüne göre sırasıyla 81, 82, 115 ve 159 kbit/sn iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla 136, 149, 124 ve 188 kbit/sn olarak ölçülmüştür.



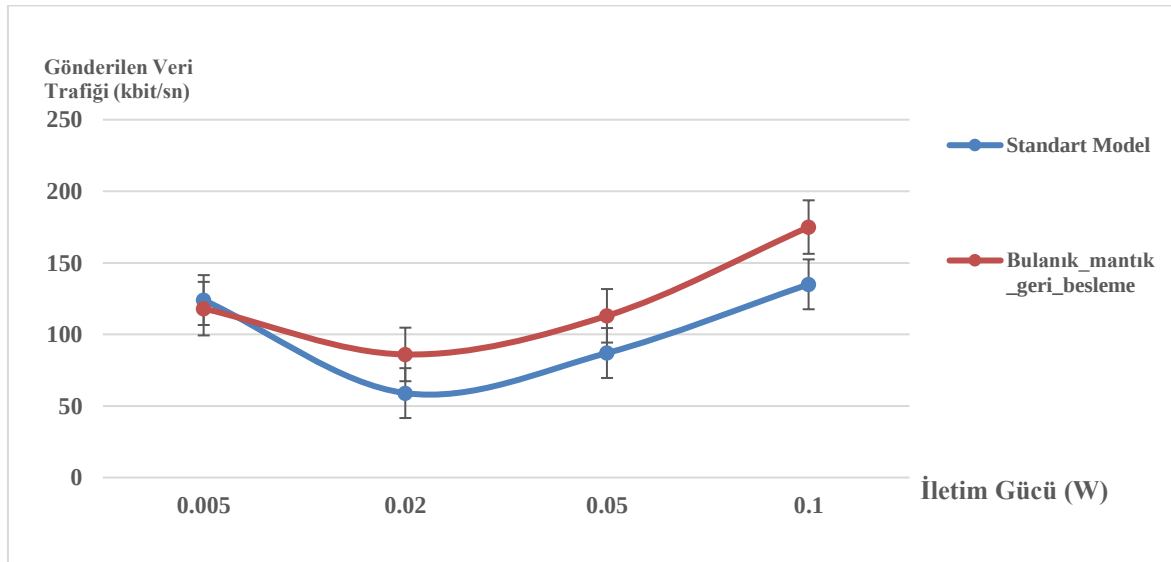
Şekil 4.37. Yeni modelde farklı iletim güçleri ile bazlı birim zamanda işlenen paket miktarı performans karşılaştırması

Şekil 4.38’de görülen değerler; standart yöntemde alınan veri trafiği iletim gücüne göre sırasıyla 2604, 1789, 1404 ve 2553 kbit/sn iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla 2612, 2330, 2098 ve 2569 kbit/sn olarak ölçülmüştür.



Şekil 4.38. Yeni modelde farklı iletim güçleri ile alınan veri trafiği performans karşılaştırması

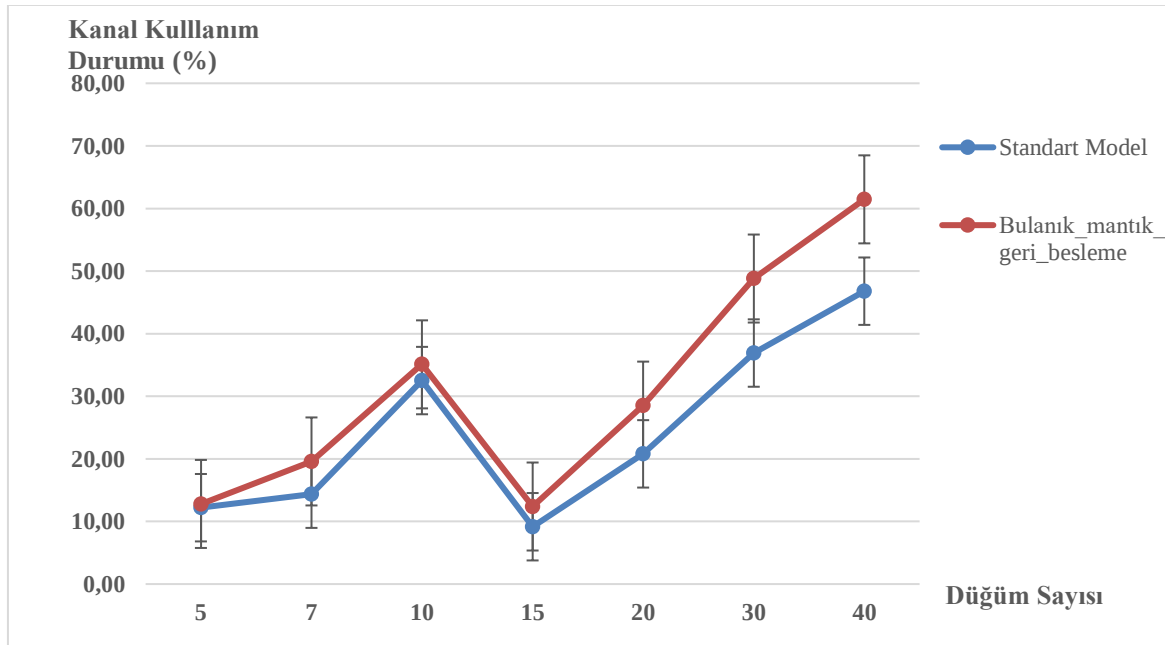
Şekil 4.39’da görülen değerler; standart yöntemde gönderilen veri trafiği iletim gücüne göre sırasıyla 124, 59, 87 ve 135 kbit/sn iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla 118, 86, 113 ve 175 kbit/sn olarak ölçülmüştür.



Şekil 4.39. Yeni modelde farklı iletim güçleri ile gönderilen veri trafiği performans karşılaştırması

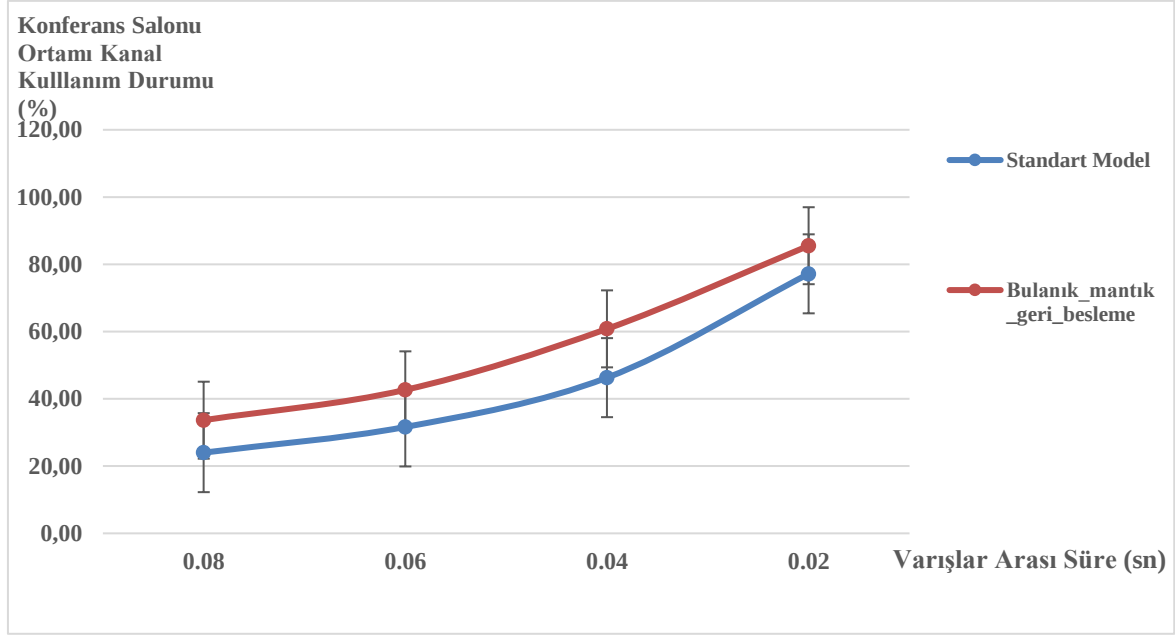
Kanal Kullanım Durumu Performans Analizi

Kanal kullanım oranlarını standart yöntemle karşılaştırmak amacıyla düğüm sayıları sırasıyla 5-7-10-15-20-30-40 değerleriyle test edilirken, varışlar arası süre tersten 0.08-0.06-0.04-0.02 değerleriyle ve iletim gücü sırasıyla 0.005-0.02-0.05-0.1 değerleri ile test edilmiştir. Şekil 4.40'da görülen değerler; standart yöntemde kanal kullanım durumu düğüm sayılarına göre sırasıyla %12,20, %14,36, %32,51, %9,16, %20,80, %36,91 ve %46,79 iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla %12,80, %19,59, %35,10, %12,39, %28,51, %48,81 ve %61,46 olarak ölçülmüştür.



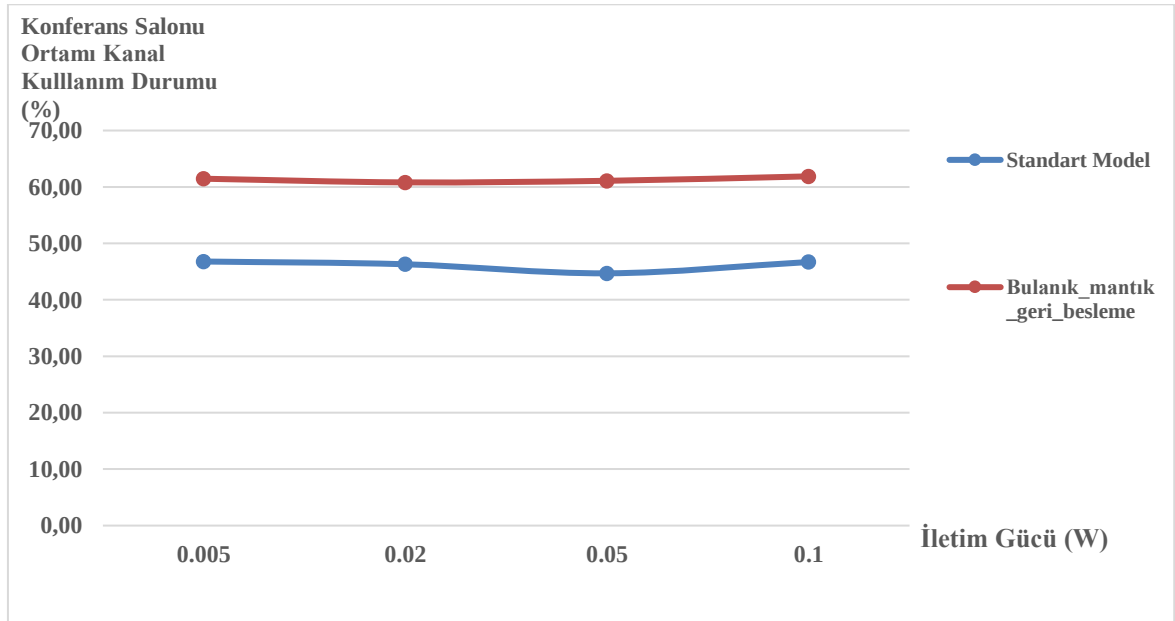
Şekil 4.40. Yeni modelde farklı düğüm sayıları ile kanal kullanım durumu performans karşılaştırması

Konferans salonu seçilerek yapılan test çalışmalarında Şekil 4.41'de görülen değerler; standart yöntemde kanal kullanım durumu varışlar arası süresine göre sırasıyla tersten %24, %31,64, %46,30 ve %77,17 iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla tersten %33,65 %42,68, %60,81 ve %85,53 olarak ölçülmüştür.



Şekil 4.41. Yeni modelde farklı varışlar arası süreler ile kanal kullanım durumu performans karşılaştırması

Konferans salonu seçilerek yapılan test çalışmalarında Şekil 4.42’de görülen değerler; standart yöntemde kanal kullanım durumu iletim gücüne göre sırasıyla %46,79, %46,30, %44,70 ve %46,70 iken, bulanık mantık tabanlı geri besleme kontrollü yöntemde sırasıyla %61,46, %60,81, %61,09 ve %61,88 olarak ölçülmüştür.



Şekil 4.42. Yeni modelde farklı iletim güçleri ile kanal kullanım durumu performans karşılaştırması

Gerçekleştirilen çalışmaların tamamı incelendiğinde benzetim anında çalışan bulanık mantık tabanlı uygulamamızda birim zamanda işlenen paket miktarı %26,48, kanal kullanım durumu %2,30, alınan veri trafiği %14,59 ve gönderilen veri trafiği ise %17,06 oranında arttırılmıştır. Bütün bu testler yeni modelin WLAN'lar için yüksek oranda performans iyileştirmesi sağladığını göstermiştir.

5. SONUÇLAR VE ÖNERİLER

WLAN'larda 802.11e'de DCF kontrol fonksiyonunda CSMA/CA protokolü ağdaki gecikmeleri önlemekte ve gizli düğüm problemlerini ortadan kaldırmaktadır. CSMA/CA'da gizli düğüm probleminin çözümü için RTSED'nin, paket kayıplarını azaltmak için PED optimizasyonunun büyük önemi bulunmaktadır. Fiziksel bellekten kazanç sağlamak ve gecikmeleri önlemek için ise AB'nin ideal değerlerde olması gereklidir. WLAN'larda ağdaki hizmet kalitesini arttırmak için dinamik ve uygun RTSED, PED ve AB giriş değerlerinin belirlenmesi gerekmektedir. Tez çalışmasında, giriş değerlerinin ideal verilerle belirlenmesi için bulanık mantık, YSA ve geri besleme kontrol yöntemleri kullanılmıştır. Gerçekleştirilen çalışmalarda gecikme süresi, yükleme ve birim zamanda işlenen paket miktarı, alıcı kanal kullanım durumu, verici için kanal kullanım durumu, alınan veri trafiği ve gönderilen veri trafiği gibi hizmet kalitesini arttıran çıkış parametrelerinde performans artışı sağlanmıştır. Ayrıca giriş ve çıkış değerlerini kapsayan yeni bir tasarım ölçeği çıkartılmıştır.

Yapılan çalışmalarda ilk olarak WLAN'larda tıkanıklık, gizli düğüm problemi ve paket kayıplarını azaltmak için Riverbed Modeler üzerinde ağ tasarlanarak RTSED, PED ve AB parametrelerinin veri analizleri yapılmıştır. Gerçekleştirilen çalışmada veri setleri ile MATLAB üzerinden bulanık mantık algoritması koşulmuştur. Koşulan algoritma sonrasında elde edilen ideal giriş değerleri Riverbed Modeler'a uygulanarak performans artışı sağlanmıştır. Bu yöntemde kullanılan bulanık mantık tabanlı modelle elde edilen ideal RTSED, PED ve AB değerleri ile gecikme süresi ortalama %37 oranında azaltılmış, yükleme süresi ortalama %6 ve birim zamanda işlenen paket miktarı ise ortalama %34,5 oranında arttırılmıştır.

İkinci olarak gerçekleştirilen çalışmada, optimizasyon yöntemlerinden YSA ile alınan ve gönderilen veri trafiği ideal değerleri elde edilmiştir. RTSED, PED ve AB parametreleri YSA kullanılarak tahminlenmiş ve kanal verimliliği ile alınan ve gönderilen veri trafiği değerleri arttırılmıştır. Yapılan çalışmada veri setleri Riverbed Modeler üzerinden alınarak MATLAB ortamında YSA algoritması koşulmuştur. Elde edilen giriş tahmin değerleri Riverbed Modeler'da yeni giriş değerleri ile koşularak hizmet kalitesi arttırılmıştır.

Üçüncü olarak gerçekleştirilen çalışmada ise kablosuz sunucu, telekom vericisi ve sinyal bozucu gibi çevresel faktörlerin kanal verimliliğine, alınan veri trafiği ve gönderilen veri trafiklerine olan etkileri incelenmiştir. İlgili çalışmada genel olarak bir ölçek tablosu çıkartılmış ve gelecek çalışmalarda örnek olarak kullanılabilecek bir çalışma gerçekleştirilmiştir. Yapılan çalışmada bulanık mantık ve YSA algoritmalarının kablosuz sunucu, telekom vericisi ve sinyal bozucularına olan etkileri ortaya çıkartılmıştır.

Son olarak gerçekleştirilen ve tezin ana omurgasını oluşturan çalışmada ise Riverbed Modeler üzerinde geri besleme kontrol tabanlı algoritma kullanılarak benzetim anında giriş değişkenlerinin güncellenmesi sağlanmıştır. İlk olarak kaba kuvvet algoritması ile döngü tablosu kullanılarak kanal kullanım oranı %17 oranında arttırılmıştır. Yapılan güncellemeler ve ajan yapısı ile ağ katmanı, düğüm katmanı ve işlem katmanı güncellenmiştir. Sistemde zamanlama fonksiyonu kullanılarak tabloda yer alan bütün giriş (RTSED, PED, AB) değerleri belirli süreyle test edilmiştir. Daha sonra bulanık mantık tabanlı geri besleme kontrolü ile düğüm bazlı birim zamanda işlenen paket miktarı %26,48, kanal kullanım durumu %2,30, alınan veri trafiği %14,59 ve gönderilen veri trafiği ise %17,06 oranında arttırılmıştır. Modeldeki ajan yapısında ağ katmanı, düğüm katmanı ve proses katmanı bulanık mantık tabanlı geri besleme kontrolü ile güncellenerek tam bir geri besleme yöntemi elde edilmiştir. Gerçekleştirilen çalışma ile benzetim ortamında ideal çıkış değerleri için giriş değerlerinin bulanık mantık tabanlı geri besleme kontrolü ile otomatik güncellenmesi sağlanmıştır. Gerçekleştirilen performans iyileştirmelerinde yeni modelin farklı düğüm sayısı, varışlar arası süre, iletim gücü v.b dış parametrelerle ne kadar oranda pozitif artış sağladığı ayrı ayrı ortaya konulmuştur.

İlerleyen süreçte yapılacak çalışmalarda, WLAN'larda CSMA/CA protokolünde incelenen giriş değişkenlerine ek olarak, bölümlene boyutu, PCF ve EDCF kontrol fonksiyonlarına da geri besleme kontrollü algoritmalar uygulanarak hizmet kalitesi arttırılabilir. WLAN'larda hizmet kalitesini arttıracak çalışmalarda optimizasyon algoritmaları ve zamanlama fonksiyonları birlikte kullanılarak performans arttırılabilir. Bütün bunlara ek olarak WLAN'larda benzetim anında otomatik giriş değişkenleri güncellemeleri ile farklı protokollerde de iyileştirmeler sağlanabilir.

KAYNAKLAR

- [1] Singh, H., Singh, T., and Kaur, M. (2014). Improving the Quality of Service of EDCF over DCF For Real Time Applications Using Probability Algorithm. *International Journal of Advanced Research in Computer and Communication Engineering*, 3(4), 6330-6333.
- [2] Dalvi, A., Svamy, P., and Meshram, B. B. (2011). DCF Improvement for Satisfactory Throughput of 802.11 WLAN. *International Journal on Computer Science and Engineering*, 3(7), 2862-2868.
- [3] Borsuk, B., ve Koçak, C. (2016). Kablosuz Ağlarda Gizli Düğüm Probleminde IEEE 802.11 MAC Katmanı RTS/CTS Mekanizmasının Çoklu Ortam Uygulamalarında Performansa Etkisi. *Bilişim Teknolojileri Dergisi*, 9(2), 187-195.
- [4] Yun, J. H., and Seo, S. W. (2007). Novel collision detection scheme and its applications for IEEE 802.11 wireless LAN, *Computer Communication*, 30(6), 1350-1366.
- [5] Kurt, O., Tuncay, G., Özkan, S., Tüzün, M., ve Gezgin, D. (2015). Yeni Kablosuz Yerel Alan Ağ Standartları neler vaat ediyor? Bir Karşılaştırma: IEEE 802.11ac, 802.11ad, 802.11ax. *XVII. Akademik Bilişim Konferansı*, 130-135.
- [6] Ayav, T. (2017). A Real-Time Communication Protocol for IEEE 802.11 Wireless Networks. *Dokuz Eylul University-Faculty of Engineering Journal of Science and Engineering*, 19(55), 1-15.
- [7] Yılmaz, E., ve Ozturk, E. (2007, Ekim). Yeni Nesil Kablosuz İletişim Teknolojileri Karşılaştırmalı Analizi. *EMO III. İletişim Teknolojileri Ulusal Sempozyumu, İtusem*.
- [8] Karakurt, H. B. (2015). *Kablosuz Yerel Alan Ağlarında Kontrol Fonksiyonları İçin Parçalama Eşik Değeri İle Performans Geliştirme*. Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Ankara, 14-15.
- [9] Yoon, S. G., Kang, S. G., and Nam, C. (2018). Priority Inversion Prevention Scheme for PLC Vehicle-to-Grid Communications Under the Hidden Station Problem. *IEEE Transactions On Smart Grid*. 9(6), 5887-5896.
- [10] Bian, K., Park, J. M., Chen, L., and Li, X. (2014). Addressing the Hidden Terminal Problem for Heterogeneous Coexistence Between TDM and CSMA Networks in White Space. *IEEE Transactions On Vehicular Technology*. 63(9), 4450-4463.
- [11] Li, T., Leith, D., and Malone, D. (2011). Buffer Sizing for 802.11 Based Networks. *IEEE/ACM Transactions On Networking*, 19(1), 156-169.

- [12] Malik, S., Chaudhary, R., Pathak, A., and Chakraborty, S. P. (2015). Modeling and Analysis of IEEE 802.11 DCF MAC. *Procedia Computer Science*, 57, 473-482.
- [13] Choi, S., Prado, J. D., N. Shankar, S., and Mangold, S. (2003). IEEE 802.11e Contention-Based Channel Access (EDCF) Performance Evaluation. *Communications, ICC '03. IEEE International Conference on*, 1151-1156.
- [14] Kaur, I., Bala, M., and Bajaj, H. (2012). Performance Evaluation of Wlan by Varying PCF, DCF and Enhanced DCF Slots to Improve Quality of Service. *IOSRJCE*, 2(5), 29-33.
- [15] Karakurt, H. B., ve Kocak, C. (2015). Kablosuz Ağlarda PCF, DCF ve EDCF Fonksiyonlarında Parçalama Eşik Değeri. *17. Akademik Bilişim Konferansı*, 795-803.
- [16] Sidelnikov, A., Yu, J., and Choi, S. (2002). Fragmentation/Aggregation Scheme for Throughput Enhancement of IEEE 802.11n WLAN. Network. in: *Proceedings of IEEE APWCS, Daejeon*.
- [17] Zhao, Z., Darbha, S., and Reddy, A. L. N. (2004). A method for estimating the proportion of nonresponsive traffic at a router. *IEEE/ACM Transactions Networking*, 12(4), 708-718.
- [18] Raina, G., and Wischik, D. (2005). Buffer Sizes for Large Multiplexers: TCP Queueing Theory and Instability Analysis. *Next Generation Internet Networks*, 173-180.
- [19] Enachescu, M., Ganjali, Y., Goel, A., McKeown, N., and Roughgarden, T. (2006). Routers with Very Small Buffers. *25th IEEE International Conference on Computer Communications. Proceedings*, 1-11.
- [20] Preveze, B. (2011). *Çoklu Ortam İletişiminde Bilişsel Yöntemler*. Doktora Tezi, Başkent Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Anabilim Dalı, Edirne, 93-96.
- [21] Wischik, D., and McKeown, N. (2005). Part I: buffer sizes for core router. *Acm Sigcomm Computer Communication Review*, 35(3), 75-78.
- [22] Pilosof, S., Ramjee, R., Raz, D., Shavitt, Y., and Sinha, P. (2003). Understanding TCP fairness over Wireless LAN. *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, IEEE Societies*, 2, 863-872.
- [23] Thottan, M., and Weigle, M. C. (2006). Impact of 802.11e EDCA on mixed TCP-based applications. *WICON '06 Proceedings of the 2nd annual international workshop on Wireless internet*.

- [24] Liu, J., Guo, W., Xiao B. L., and Huang, F. (2006). RTS Threshold Adjustment Algorithm for IEEE 802.11 DCF*. *6th International Conference on ITS Telecommunications Proceedings*, 654-658.
- [25] Abusubaih, M. (2012). Joint RTS/CTS and time slotting for interference mitigation in multi-BSS 802.11 wireless LANs. *Computers and Electrical Engineering*, 38, 672-680.
- [26] Ahsan, S. M. R., Islam, M. S., Hassan, N., and Rahman, A. (2010). Packet Distribution Based Tuning of RTS Threshold in IEEE 802.11. *The IEEE symposium on Computers and Communications*, 1-6.
- [27] Yin, Y., Gao, Y., Manzoor, S., and Hei, X. (2019). Optimal RTS Threshold for IEEE 802.11 WLANs: Basic or RTS/CTS?. *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation*, 1620-1625.
- [28] Karakurt, H. B., Koçak, C., and Doğru, İ. A. (2018). Multi-layer security and its impact on performance in wireless networks. *Karaelmas Science and Engineering Journal*, 8(1), 121-125.
- [29] Asadi, A., Sciancalepore, V., and Mancuso, V. (2015, January). On the Efficient Utilization of Radio Resources in Extremely Dense Wireless Networks. *IEEE Communications Magazine*, 126-132.
- [30] Singh, S., Chandra, N., and Tripathi, A. K. (2015). Performance Analysis for Channel Utilization in Wireless LAN. *International Journal of Computer Applications*, 122(20), 40-44.
- [31] Yazid, M., Medjkoune, L. B., Aissani, D., Amrouche, N., and Bakli, K. (2014). Analytical Analysis of Applying Packet Fragmentation Mechanism on Both Basic and RTS/CTS Access Methods of the IEEE 802.11b DCF Network Under Imperfect Channel and Finite Load Conditions. *Wireless Personal Communications*, 77(1), 477-506.
- [32] Purohit, H., and Joshi, K. (2016). A Simulation Based Comparative Study on Channel Utilization Enhancement Using DITMC Technique in Speech and Data Communication of Mobile Network. *6th International Conference on Information Communication and Management*, 201-204.
- [33] Lin, C.H., Chen, Y. C., and Lin, C. J. (2018). FDoF: Enhancing Channel Utilization for 802.11ac. *IEEE/Acm Transactions On Networking*, 26(1), 465-477.
- [34] Saketh, M., Reddy, S. K., Karmakar, R., Chattopadhyay, S., and Chakraborty, S. (2007-December). IEEE 802.11ac DBCA: A Tug of War Between Channel Utilization and Fairness. *Globecom, 2017 IEEE Global Communications Conference*, 1-6.

- [35] Kobayashi, H., Kameda, E., Terashima, Y., and Shinomiya, N. (2018). Towards sustainable heterogeneous wireless networks: A decision strategy for AP selection with dynamic graphs. *Computer Networks*, 132, 99-107.
- [36] Wang, S. Y., Chou, C. L., Liu, K. C., Ho, T. W., Hung, W. J., Huang, C. F., Hsu, M. S., Chen, H. Y., and Lin, C. C. (2009). Improving the Channel Utilization of IEEE 802.11p/1609 Networks. *2009 IEEE Wireless Communications and Networking Conference*, 1-6.
- [37] Laufer, R., and Kleinrock, L. (2016). The Capacity of Wireless CSMA/CA Networks. *IEEE/Acm Transactions On Networking*. 24(3), 1518-1532.
- [38] Shih, C. F., Krishnaswamy, B., Jian, Y., and Sivakumar, R. (2018). Scheduled WiFi using distributed contention in WLANs: algorithms, experiments, and case-studies. *Wireless Netw*, 24, 89-112.
- [39] Isizoh, A. N., Anazia, A. E., Okide, S. O., and Okwaraoka, C. A. P. (2013). Effects Of Different Fragmentation Thresholds On Data Dropped And Retransmission Attempts In A Wireless Local Area Network. *International Journal of Engineering Research and Applications (IJERA)*, 3(2), 76-079.
- [40] Xiao, Y. (2003). Enhanced DCF of IEEE 802.11e to Support QoS. *Proceedings of IEEE WCNC*. 1290-1296.
- [41] Vijay, B. T., and Malarkodi, B. (2016). Improved QoS in WLAN using IEEE 802.11e. *Twelfth International Multi-Conference on Information Processing*, 89, 17-26.
- [42] Xu, K., Gerla M., Bae S. (2003). Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks. *Ad Hoc Networks*, 1, 107-123.
- [43] Doğancı, Y. U. (2008). *802.11 Standartlarını Kullanarak Pozisyon Tespiti*. Yüksek Lisans Tezi, TOBB Ekonomi ve Teknoloji Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Ankara, 15-16.
- [44] Akbaş, D. (2010). *Bir Kurumsal Ağın ve Güvenlik Yapılarının Modellenmesi ve Analizi*. Yüksek Lisans Tezi, Haliç Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 31-64.
- [45] Zubairi, J. A., and Zuber, M. (2000). Suny Fredonia Campus Network Simulation and Performance Analysis Using OPNET. *Opnetwork2000*. Washington DC, 1-3.
- [46] Lawal, İ. A., Said, A. M., Mu'azu, A. A., and Shah, P. A. (2014). Performance Comparison of Centralized and Distributed Network Models to Support QoS in Fixed WiMAX. *Procedia Social and Behavioral Sciences*, 129, 441, 452.
- [47] Zhao, W., and Xie, J. (2011). OPNET-based modeling and simulation study on handoffs in Internet-based infrastructure wireless mesh network. *Computer Networks*, 55, 2675-2688.

- [48] Erdem, O. A., ve Tuğral, N. (2009). Kablosuz Bilgisayar Ağlarının Karşılaştırılmalı İncelenmesi. *Bilişim Teknolojileri Dergisi*, 2(2), 1-10.
- [49] Çakır, C., ve Kaptan, H. (2009). VoIP Teknolojilerinde OPNET Tabanlı Güvenlik Uygulaması. *Bilişim Teknolojileri Dergisi*, 3(2), 1-7.
- [50] Wei, P., Hong, Z., and Shi M. (2016). Performance Analysis of HTTP and FTP Based on OPNET. *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, Okayama, 1-4.
- [51] Mekkanen, M., Antila, E., Virrankoski, R., and Elmusrati, M. (2013). Using OPNET to Model and Evaluate the MU Performance based on IEC 61850-9-2LE. *Procedia Computer Science*, 36, 72-79.
- [52] Biçen, M., Çalhan, A., ve Yücedağ, İ. (2016). Kablosuz Heterojen Algılayıcı Ağlarda Bulanık Mantık Tabanlı Ağ Geçidi Seçimi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 4, 655-660.
- [53] Vadiati, M., Asghari, M. A., Nakhaei, M., Adamowski, J., and Akbarzadeh, A. H. (2016). A fuzzy-logic based decision-making approach for identification of groundwater quality based on groundwater quality indices. *Journal of Environmental Management*, 184, 255-270.
- [54] Kustiawan, I., and Chi, K. H. (2015). Handoff Decision Using a Kalman Filter and Fuzzy Logic in Heterogeneous Wireless Networks. *IEEE Communications Letters*, 19(12), 2258- 2261.
- [55] Zhaoxiang, W., Hai-lun, X., and Wei, D. (2008). A fuzzy logic cooperative MAC for MANET. *The Journal Of China Universities Of Posts And Telecommunications*, 15(1), 55-60.
- [56] Frantti, T., and Koivula, M. (2011). Fuzzy packet size control for delay sensitive traffic in ad hoc networks. *Expert Systems with Applications*, 38, 10188-10198.
- [57] Collotta, M. (2015). FLBA: A fuzzy algorithm for load balancing in IEEE 802.11 networks. *Journal of Network and Computer Applications*, 53, 183-192.
- [58] Mcculloch, W., and Pitts W. (1943). A Logical Calculus Of The Ideas Immanent In Nervous Activity. *Bulletin Of Mathematical Biophysics*, 5, 115-133.
- [59] Sun, H., Chen, X., Shi, Q., Hong, M., Fu, X., and Sidiropoulos, N. D. (2017). Learning to optimize: Training deep neural networks for wireless resource management. *In Signal Processing Advances in Wireless Communications (SPAWC), 2017 IEEE 18th International Workshop on*, 1-6.
- [60] Jiang, C., Zhang, H., Ren, Y., Han, Z., Chen, K. C., and Hanzo, L. (2017). Machine learning paradigms for next-generation wireless Networks. *IEEE Wireless Communications*, 24(2), 98–105.

- [61] Kato, N., Fadlullah, Z. M., Mao, B., Tang, F., Akashi, O., Inoue, T., and Mizutani, K. (2017). The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective. *IEEE wireless communications*, 24 (3), 146-153.
- [62] Bi, S., Zhang, R., Ding Z., and Cui S. (2015). Wireless communications in the era of big data. *IEEE Communications Magazine*, 53(10), 190–199.
- [63] O’Shea, T. J., and Hoydis, J. (2017). An introduction to machine learning communications systems, *CoRR*, 1-10.
- [64] Kaplan, Y., Saray, U., Emeksiz, C., Yeşilnacar, Y.O., Önal, S., ve Karaca, V. (2013). Yapay Sinir Ağı Geri Yayınım Algoritması kullanılarak Rüzgar Hızı Tahmini. IX. *Clean Energy Symposium, UTES’13*, 1-7.
- [65] Erdem, S. (2007). YSA ve GA Temelli Yeni Bir Algoritma İle Doğrusal Olmayan Optimizasyon. *İşletme Fakültesi Dergisi*. 8(2), 219-231.
- [66] Schmidt, M., Block, D., and Meier, U. (2017). Wireless Interference Identification with Convolutional Neural Networks. *IEEE 15th International Conference on Industrial Informatics (INDIN)*, 180-185.
- [67] Chen, M., Challita, U., Saad, W., Yin, C., and Debbah, M. (2017). *Machine Learning for Wireless Networks with Artificial Intelligence*. A Tutorial on Neural Networks, Beijing Laboratory of Advanced Information Network, 11-12.
- [68] Yiğit, T., and Ersoy, M. (2014). Testing and Design of Indoor WLAN Using Artificial Intelligence Techniques. *Elektronika ir Elektrotechnika*, 20(6), 154-157.
- [69] Ozan, C., Başkan, Ö., Haldenbilen, S., ve Ceylan, H. (2012). Pekiştirmeli Öğrenme Yöntemi ile Dinamik Kullanıcı Dengesi Altında Ulaşım Ağ Tasarımı. *Asyu 2012, Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu*, 20-24.
- [70] Ozan, C. (2012). *İyileştirilmiş Pekiştirmeli Öğrenme Yöntemi Ve Dinamik Yükleme İle Kentiçi Ulaşım Ağlarının Tasarlanması*, Doktora Tezi, Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Denizli, 63-69.
- [71] Yu, Y., Wang, T., and Liew, C. (2019). Deep-Reinforcement Learning Multiple Access for Heterogeneous Wireless Networks. *IEEE Journal On Selected Areas In Communications*, 37(6), 1277-1290.
- [72] Bhattacharyya, R., Bura, A., Rengarajan, D., Rumuly, M., Shakkottai, S., Kalathil, D., Mok, R. K. P., and Dhamdhere, A. (2019). QFlow: A Reinforcement Learning Approach to High QoE Video Streaming over Wireless Networks. *Mobihoc ’19, Catania, Italy*, 1-10.
- [73] Oliveira, A. M., Varma V. S., Postoyan R., Morarescu I. C., Daafouz J., Costa O. V. I. (2018). Network-Aware Design of State-Feedback Controllers for Linear Wireless Networked Control Systems. *IFAC PapersOnLine*, 51-16, 205–210.

- [74] Isele, D., Rahimi, R., Cosgun, A., Subramanian, K., and Fujimura, K. (2018). Navigating Occluded Intersections with Autonomous Vehicles using Deep Reinforcement Learning. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2034-2039.
- [75] You, C., Lu, J., Filev, D., and Tsiotras, P. (2019). Advanced planning for autonomous vehicles using reinforcement learning and deep inverse reinforcement learning. *Robotics and Autonomous Systems*, 114, 1-18.
- [76] Yao, H., Chen, X., Li, M., Zhang, P., and Wang, L. (2018). A novel reinforcement learning algorithm for virtual network embedding. *Neurocomputing*, 284, 1-19.
- [77] Lu, Z., and Yang, H. (2012). *Unlocking the Power of OPNET Modeler*. Published in the United States of America by Cambridge University Press, New York, 81-82.
- [78] Bandırmalı, N., Ertürk, İ., Çeken, C., ve Bayılmış, C. (2008-Kasım). Skipjack Şifreleme Algoritması Kullanarak Gecikme Duyarlı ve Enerji Etkin Kablosuz Algılayıcı Ağ Güvenlik Hizmeti. *Elektrik ve Elektronik ve Bilgisayar Mühendisliği Sempozyumu, ELECO'08*, 152-157.
- [79] Soylu, T. (2012). *Kablosuz Algılayıcı Ağların Uygulama Alanları ve Bir Algılayıcı Düğüm Tasarımı*. Yüksek Lisans Tezi, Trakya Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Edirne, 31-32.
- [80] Karakurt, H.B., Koçak C. (2020). Kablosuz Yerel Alan Ağlarında Kanal Kullanımının MAC Katmanında Döngü Tablosu ile Artırılması. *GU J Sci, Part C*, 8(3), 542-560.
- [81] Şanlı, M., Zengin, F., ve Urhan, O. (2005). Akıllı Kart ile Anahtar Güncellemeli 3-DES Algoritması Kullanarak Ön Ödemeli Sistem Uygulaması. *Proceeding IEEE*, 76-79.
- [82] Stallings, W. (2005). *Wireless Communications and Networks*, 2nd ed., Prentice Hall, Englewood Cliffs, NJ, 346-348.
- [83] Sarmah, S., and Sarma, S. K. (2016). Performane Analysis of IEEE 802.11 WLAN by Varying PCF, DCF and EDCF to Enhance Quality of Service. *International Journal of Computer Applications*, 138(3), 22-25.
- [84] Nj, M., Suryana, N., Sahib, S., and Hussin, B. (2017). RTS/CTS Framework Paradigm and WLAN Qos Provisioning Methods. *IJACSA*, 8(2), 176-186.
- [85] Cheng, S. S., Huang, C. M., and Tu, T. H. (2017). A CSMA/CA-Based Media Access Protocol Using the Registered Backoff Time Vector (RBTv) Mechanism for M2M Communications. *Mobile Netw Appl*, 22, 83-97.

- [86] Moraes, R., Portugal, P., Vasques, F., and Fonseca, J. A. (2008). Limitations of the IEEE 802.11e EDCA protocol when supporting real-time communication. *2008 IEEE International Workshop on Factory Communication Systems*, 119-128.
- [87] Sharma, P., and Gurpreet, Singh. (2016). Comparison of Wi-Fi IEEE 802.11 Standards Relating to Media Access Control Protocols. *International Journal of Computer Science and Information Security (IJCSIS)*, 14, 856-862.
- [88] Zarinni, F. (2015). *Understanding and Improving Performance in Next-Generation WLANs and Cellular Networks*. Dissertation, Stony Brook University, 37-38.
- [89] Kocak, C., and Karakurt, H. B. (2019). Fuzzy logic-based performance improvement on MAC layer in wireless local area networks. *Neural Computing and Applications*, 31, 6113–6128.
- [90] Lee, Y., Chung, M. Y., and Lee, T. J. (2008). Performance Analysis of IEEE 802.11 DCF under Nonsaturation Condition. *Mathematical Problems in Engineering*, 17, 1-17.
- [91] Boggia, G., Camarda, P., Grieco, L. A., and Mascolo, S. (2007). Feedback-Based Control for Providing Real-Time Services With the 802.11e MAC. *IEEE/Acm Transactions On Networking*, 15(2), 323-333.
- [92] Jabeen, Q., Khan, F., Khan, S., and Jan, M. A. (2016). Performance Improvement in Multihop Wireless Mobile Adhoc Networks. *J. Appl. Environ. Biol. Sci*, 6(4S), 82-92.
- [93] Giweli, N., Shahrestani, S., and Cheung, H. (2016). QoS-aware Spectrum Sensing in White-Fi Cognitive Radio Networks. *10th International Conference on Signal Processing and Communication Systems (ICSPCS)*, Sydney, Australia, 1-8.
- [94] Ibrahim, I. K. (2006). *Handbook of research on mobile multimedia*. Context Aware Collaborative Working Environments, 2.nd ed. Linz, Austria, 141-143.
- [95] Kaur, J., Mohan, V., and Modi, N. (2014). Analyzing Load and Delay in Wireless LAN using TTL and Fragmentation. *An International Journal of Engineering Sciences*, 1, 86-90.
- [96] Kocak, C., and Karakurt, H. B. (2019). Data traffic optimization in wireless local area networks with artificial neural networks. *Journal of Polytechnic*, 22, 737–747.
- [97] Costa, R., Portugal, P., Vasques, F., Montez, C., and Moraes, R. (2015). Limitations of the IEEE 802.11 DCF, PCF, EDCA and HCCA to handle real-time traffic. *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*, IEEE, Cambridge, UK, 931-936.

- [98] Li, T. (2007). *Improving Performance for CSMA/CA Based Wireless Networks*. Ph. D. Dissertation, Hamilton Institute National University of Ireland, Maynooth Maynooth, Co. Kildare, Ireland, 24-33.
- [99] Mewara, H. S., and Manghnani, D. (2014). Rising up the Performance of Wireless Local Area Network by Varying Fragmentation Threshold using OPNET. *International Journal of Computer Applications*, 97(7), 22-25.
- [100] Wang, Q., and Yuan, D. (2010). An Adaptive Backoff Algorithm for IEEE 802.11 DCF with Cross-layer Optimization. *2010 6th International Conference on Wireless Communications Networking and Mobile Computing (Wicom)*, 1-4.
- [101] Zhai, H., Kwon, Y., and Fang, Y. (2004). Performance analysis of IEEE 802.11 MAC protocols in wireless LANs. *Wireless Communications And Mobile Computing* 4, 917-931.
- [102] Pan, J. (2008). *A Survey of Network Simulation Tools Current: Status and Future Developments*. A project report written under the guidance of Prof. Raj Jain, 1-13.
- [103] Bıyan, M., and Alan, M. A. (2013). Bilgisayar Ağının Modellenmesi ve Uzaktan Eğitim için Performans Ölçümü. *Cumhuriyet Üniversitesi, Sosyal Bilimler Araştırmaları Dergisi*. 2, 99-111.
- [104] Sayar, M. (2013). Simulasyona Dayalı Ağ Temelleri Dersi Eğitimi. *15. Akademik Bilişim Konferansı AB 2013*, 62-68.
- [105] Jalali, S.Y., Wani, S., and Derwesh, M. (2014). Qualitative Analysis and Performance Evaluation of RIP, IGRP, OSPF and EGRP Using OPNET™. *Advance in Electronic and Electric Engineering*, 4(4), 389-396.
- [106] Hanle, C., and Hofmann, M. (1998). Performance Comparison of Reliable Multicast Protocols using the Network Simulator ns-2. *Proceedings 23rd Annual conference on local computer Networks*, 222-237.
- [107] Drajić, M., and Kokić, I. (2012). A comparative analysis of simulators NS2 and OPNET IT Guru Academic Edition, *2012 20th Telecommunications Forum (Telfor)*, 1780-1783.
- [108] Tekin, U., ve Soğukpınar, İ. (2007-Kasım). IPsec Benzetim Yazılımı: Tasarım ve Gerçekleme. *12. Elektrik, Elektronik, Bilgisayar, Biyomedikal Mühendisliği Ulusal Kongresi ve Fuarı*, Eskişehir, 1-5.
- [109] Çavuşoğlu, Ü., ve Zengin, A. (2012). NS-2 ve NS-3 Ağ Simülatörlerinin Ölçeklenebilirlik Analizi ve Karşılaştırma. *Bilişim Teknolojileri Dergisi*, 5(3), 41-49.
- [110] Chang, X. (1999). Network Simulations with OPNET. *1999 Winter Simulation Conference Proceedings*, 307-314.

- [111] Hao, J., Wu, J., and Guo, C. (2011). Modeling and simulation of CAN network based on OPNET. *2011 IEEE 3rd International Conference on Communication Software and Networks*, 577-581.
- [112] Ozkan, M. T. (2013). Experimental and artificial neural network study of heat formation values of drilling and boring operations on Al 7075 T6 workpiece. *Indian Journal of Engineering and Materials Sciences*, 20(4), 259-268.
- [113] Ozkan, M. T., Ulas, H. B., and Bilgin, M. (2014). Experimental design and artificial neural network model for turning the 50crv4 (sae 6150) alloy using coated carbide/cermet cutting tools. *Materials and Technology*, 48(2), 227–236.



GAZİ GELECEKTİR..