

IPTV İÇİN ULAŞTIRMA PROTOKOLÜ GELİŞTİRİLMESİ

Alisettar HÜSEYNLİ

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

TEMMUZ 2013

ANKARA

Alisettar HÜSEYNLİ tarafından hazırlanan “IPTV İÇİN ULAŞTIRMA PROTOKOLÜ GELİŞTİRİLMESİ” adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Prof. Dr. M. Ali AKCAYOL

.....

Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Prof. Dr. O. Ayhan ERDEM

.....

Bilgisayar Mühendisliği Anabilim Dalı, G.Ü.

Prof. Dr. M. Ali AKCAYOL

.....

Bilgisayar Mühendisliği Anabilim Dalı, G.Ü.

Doç. Dr. Erdoğan DOĞDU

.....

Bilgisayar Mühendisliği Anabilim Dalı, TOBB ETÜ

Tez Savunma Tarihi: 17/07/2013

Bu tez ile G.Ü. Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onamıştır.

Prof. Dr. Şeref SAĞIROĞLU

.....

Fen Bilimleri Enstitüsü Müdürü

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Alisettar HÜSEYNLİ

IPTV İÇİN ULAŞTIRMA PROTOKOLÜ GELİŞTİRİLMESİ

(Yüksek Lisans Tezi)

Alisettar HÜSEYNLİ

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Temmuz 2013

ÖZET

Bu tezde IPTV uygulamalarında kullanılmak üzere kuyruk önceliklendirmeye dayalı yeni bir ulaşım katmanı protokolü (Priority Queue - Stream Control Transmission Protocol - PQ-SCTP) geliştirilmiştir. Geliştirilen protokol veri paketlerinde önceliklendirme yapmakta ve bu önceliğe bağlı olarak kuyrukta sıralama yapmaktadır. Geliştirilen protokol Network Simulator 2 (ns-2) benzetim aracı ile test edilmiş ve elde edilen sonuçlar SCTP protokolü ile karşılaştırılmıştır. Benzetim sonuçları PQ-SCTP protokolünün farklı kuyruk boyutları için kuyruk gecikmesi, gecikme değişimi, jitter gibi performans ölçütlerinde daha başarılı olduğunu göstermiştir.

Bilim Kodu : 902.1.063
Anahtar Kelimeler : IPTV, PQ-SCTP, SCTP, kuyruk önceliklendirme
Sayfa Adedi : 92
Tez Yöneticisi : Prof. Dr. M. Ali AKCAYOL

**DEVELOPMENT OF THE TRANSPORT LAYER PROTOCOL FOR IPTV
(M.Sc. Thesis)**

Alisettar HUSEYNLI

**GAZİ UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
July 2013**

ABSTRACT

In this thesis, a priority queue based new transport layer protocol (Priority Queue - Stream Control Transmission Protocol - PQ-SCTP) has been developed for IPTV applications. The protocol defines the priority to data packets and depending on the priority receiver queue is sorted. Experiments on protocol was performed using the simulation tool Network Simulator (ns-2). The simulation results are compared with the SCTP protocol for different queue sizes and in performance criteria like queuing delay, delay variation and jitter. For the developed protocol better results have been achieved.

**Science Code : 902.1.063
Key Words : IPTV, PQ-SCTP, SCTP, priority queue
Page Number : 92
Supervisor : Prof. Dr. M. Ali AKCAYOL**

TEŐEKKÜR

Her konuda desteęini ve bilgisini benden esirgemeyen deęerli tez danıőmanım Prof. Dr. M. Ali AKCAYOL'a, yardımları ile beni yönlendiren Uzm. Dr. Mehmet ŐİMŐEK'e, eęitim hayatımda her zaman bana destek olan babama ve anneme teőekkürü kendime borç bilirim.

İÇİNDEKİLER

	Sayfa
ÖZET.....	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	x
ŞEKİLLERİN LİSTESİ	xi
SİMGELER VE KISALTMALAR.....	xiv
1. GİRİŞ.....	1
2. IPTV	5
2.1. IPTV Hizmetleri	7
2.1.1. Çoklu televizyon yayını	7
2.1.2. İsteğe bağlı video (VoD).....	8
2.1.3. Triple play hizmeti	8
2.1.4. Web/mail hizmeti	9
2.1.5. VoIP hizmeti	9
2.1.6. Bireysel reklamcılık	9
2.2. IPTV ve Internet TV	10
2.3. IPTV Uygulamaları	11
2.4. IPTV Mimarisi	11
2.5. Erişim Ağları	16
2.5.1. DSL ağlar	16
2.5.2. Carrier-grade ethernet	17
2.5.3. Kablosuz ağlar.....	17
2.5.4. Fiber-optik ağlar.....	18

	Sayfa
2.6. Omurga Ağları.....	19
2.6.1. ATM.....	20
2.6.2. MPLS	20
2.6.3. SONET.....	21
2.6.4. Metro ethernet	21
2.6.5. Multicast ve IGMP	21
2.7. İçerik Dosya Formatları	21
2.8. STB (Set Top Box).....	22
3. IPTV VE SCTP	23
3.1. SCTP ve Diğer Ulaşım Katmanı Protokollerinin Karşılaştırılması	23
3.1.1. SCTP protokolünün çalışma şekli.....	23
3.1.2. SCTP protokolünün paket yapısı	26
3.1.3. Diğer ulaşım katmanı protokolleri ile karşılaştırılması.....	28
3.2. Video İletişiminde SCTP Protokolünün Kullanımı	29
3.3. IPTV’de SCTP Protokolü.....	30
4. KUYRUK ÖNCELİKLENDİRMELİ ULAŞTIRMA KATMANI PROTOKOLÜ	34
4.1. PQ-SCTP Protokolünün Mimarisi	34
4.2. PQ-SCTP Protokolünde Paketlere İçerik Tabanlı Öncelik Verilmesi	35
4.3. PQ-SCTP Protokolünde Rtx Paketlere Öncelik Verilmesi	36
4.4. PQ-SCTP Protokolünde Alıcı Kuyruğunun Sıralanması	37
4.5. PQ-SCTP Protokolünde Üst Katman İçin İşlem Gecikmesini Tanımlanması	42
5. DENEYSEL SONUÇLAR.....	44
5.1. Benzetim Ortamı	47
5.1.1. ns-2 için hazırlanmış SCTP protokolü	49

	Sayfa
5.1.2. SCTP uygulaması üzerinde yapılan deęişiklikler	50
5.2. Dumbbell Topolojisi Simülasyonu.....	52
5.3. Parking Lot Topoloji Simülasyonu	66
6. SONUÇLAR	80
KAYNAKLAR	81
EKLER	84
EK-1. Kayıt dosyası örneęi	85
EK-2. Dumbbell.tcl OTCL script kodu	86
EK-3. Parkinglot.tcl OTCL script kodu	89
ÖZGEÇMİŞ	92

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Servisler ve gerekli bant genişlikleri.....	22
Çizelge 3.1. SCTP protokolünde veri parçası türleri	27
Çizelge 4.1. PQ-SCTP için bit değerleri ve öncelik sırası	36
Çizelge 4.2. SCTP ve PQ-SCTP için kuyruk boyutuna bağlı throughput değişimi...	41
Çizelge 5.1. Dumbbell topolojisinde SCTP protokolü için benzetim sonuçları	58
Çizelge 5.2. Dumbbell topolojisinde PQ-SCTP protokolü için benzetim sonuçları..	59
Çizelge 5.3. Dumbbell topolojisinde toplam kuyruk gecikmeleri	60
Çizelge 5.4. Parking lot topolojisinde SCTP protokolü için benzetim sonuçları	71
Çizelge 5.5. Parking lot topolojisinde PQ-SCTP protokolü için benzetim sonuçları	72
Çizelge 5.6. Parking lot topolojisinde toplam kuyruk gecikmeleri.....	73

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. IP multicast	13
Şekil 2.2. IPTV ağ katmanları.....	14
Şekil 2.3. Genel IPTV Mimarisi	15
Şekil 2.4. DSL ağın yapısı	17
Şekil 2.5. Tipik WiMAX ağ yapısı	18
Şekil 3.1. 4-way handshaking	24
Şekil 3.2. SCTP ortak başlık yapısı.....	26
Şekil 3.3. SCTP veri parçası başlığı.....	28
Şekil 3.4. IMS tabanlı IPTV ve PQoS mimarisi	32
Şekil 4.1. SCTP Payload Data türündeki veri parçasının yapısı	35
Şekil 4.2. SCTP ve PQ-SCTP alıcı kuyruğu karşılaştırması.....	38
Şekil 4.3. PQ-SCTP kuyruğa ekleme algoritması için akış diyagramı	39
Şekil 4.4. PQ-SCTP kuyruk boyutuna bağlı olarak throughput değişimi	40
Şekil 4.5. PQ-SCTP protokolü için kuyruktan atma algoritması için akış diyagramı	43
Şekil 5.1. Nam uygulaması ekran görüntüsü	49
Şekil 5.2. Dumbbell topolojisi	52
Şekil 5.3. ns-2 üzerinde düğüm ve protokol yapısı	54
Şekil 5.4. Dumbbell topolojisinde 1. öncelik grubu için ortalama kuyruk gecikmesi	55

Şekil	Sayfa
Şekil 5.5. Dumbbell topolojisinde 2.öncelik grubu için ortalama kuyruk gecikmesi.....	55
Şekil 5.6. Dumbbell topolojisinde 3.öncelik grubu için ortalama kuyruk gecikmesi.....	56
Şekil 5.7. Dumbbell topolojisinde kuyruk boyutuna bağlı alıcıya ulaşan veri parçası adedi	57
Şekil 5.8. Dumbbell topolojisinde kuyruk boyutuna bağlı toplam kuyruk gecikmesi.....	57
Şekil 5.9. Dumbbell topolojisinde 1.öncelik grubu için kuyruk gecikmesi değişimi	61
Şekil 5.10. Dumbbell topolojisinde 2.öncelik grubu için kuyruk gecikmesi değişimi	61
Şekil 5.11. Dumbbell topolojisinde 3.öncelik grubu için kuyruk gecikmesi değişimi	62
Şekil 5.12. Dumbbell topolojisinde 1.öncelik grubu için jitter dağılımı.....	63
Şekil 5.13. Dumbbell topolojisinde 2.öncelik grubu için jitter dağılımı.....	64
Şekil 5.14. Dumbbell topolojisinde 3.öncelik grubu için jitter dağılımı.....	65
Şekil 5.15. Parking lot topolojisi.....	66
Şekil 5.16. Parking lot topolojisinde kuyruk boyutuna göre throughput değişimi	67
Şekil 5.17. Parking lot topolojisinde 1.öncelik grubu için ortalama kuyruk gecikmesi.....	68
Şekil 5.18. Parking lot topolojisinde 2.öncelik grubu için ortalama kuyruk gecikmesi.....	68
Şekil 5.19. Parking lot topolojisinde 3.öncelik grubu için ortalama kuyruk gecikmesi.....	69
Şekil 5.20. Parking lot topolojisinde kuyruk boyutuna bağlı alıcıya ulaşan veri parçası adedi	70

Şekil	Sayfa
Şekil 5.21. Parking lot topolojisinde kuyruk boyutuna bağlı toplam kuyruk gecikmesi.....	73
Şekil 5.22. Parking lot topolojisinde 1.öncelik grubu için kuyruk gecikmesi değişimi	74
Şekil 5.23. Parking lot topolojisinde 2.öncelik grubu için kuyruk gecikmesi değişimi	74
Şekil 5.24. Parking lot topolojisinde 3.öncelik grubu için kuyruk gecikmesi değişimi	75
Şekil 5.25. Parking lot topolojisinde 1.öncelik grubu için jitter dağılımı	76
Şekil 5.26. Parking lot topolojisinde 2.öncelik grubu için jitter dağılımı	77
Şekil 5.27. Parking lot topolojisinde 3.öncelik grubu için jitter dağılımı	78

SİMGELER VE KISALTMALAR

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

Simgeler	Açıklama
ms	Milisaniye
s	Saniye
Mb/s	Mega bit/saniye
Gb/s	Giga bit/saniye
Kb/s	Kilo bit/saniye
Rtx	Tekrar gönderim
Kısaltmalar	Açıklama
ASCONF	Address Configuration - Adres Konfigürasyonu
ADSL	Asymmetric Digital Subscriber Line - Bakımsız Sayısal Abone Hattı
ASCONF-ACK	Address Configuration Acknowledgment - Adres Konfigürasyon Onayı
ATM	Asynchronous Transfer Mode - Eş Zamansız Aktarım Modu
CBR	Object Tool Command Language - Nesne Araçlı Komut Dili
CoD	Content on Demand - İsteğe Bağlı İçerik
CRC	Cyclic Redundancy Check - Döngüsel Artıklık Denetimi
DAR	Dynamic Address Reconfiguration - Dinamik Adres Düzenlenmesi

Kısaltmalar	Açıklama
DVR	Digital Video Recorder - Sayısal Video Kaydedici
FTTC	Fiber to The Cabinet - Kabinete Kadar Fiber
FTTH	Fiber to The Home - Eve Kadar Fiber
FTTN	Fiber to The Node - Noktaya Kadar Fiber
FTTRO	Fiber to The Regional Office - Bölgesel Ofise Kadar Fiber
GPON	Gigabit Passive Optical Network - Gigabit Pasif Optik Ağ
HDTV	High Definition Television - Yüksek Çözünürlüklü Televizyon
HOL	Head Of Line - Hat Başı
ICC	Instant Channel Change - Anlık Kanal Değişimi
IETF	Internet Engineering Task Force - İnternet Mühendisliği Görev Gücü
IGMP	Internet Group Managment Protocol - İnternet Grup Yönetim Protokolü
IP	Internet Protocol - İnternet Protokolü
IPDV	IP Packet Delay Variation - IP Paket Gecikme Değişimi
IPTV	Internet Protocol Television - İnternet Protokolü Üzerinden Televizyon Yayını
KIT	Kingstone Interactive Television - Kingstone Etkileşimli Televizyonu
LAN	Local Area Network - Yerel Alan Ağı
MPEG	Moving Picture Experts Group - Hareketli Resim Uzmanları Grubu
MPLS	Multi Protocol Label Switching - Çoklu Protokol Etiket Anahtarlama
NKE	Network Kernel Extension - Ağ Çekirdek Eklentisi

Kısaltmalar	Açıklama
OTCL	Object Tool Command Language - Nesne Araçlı Komut Dili
PQoS	Perceived QoS - Algılanan QoS
PQ-SCTP	Priority Queued SCTP - Kuyruk Önceliklendirmeli SCTP
PR-SCTP	Partial-Reliable SCTP - Kısmi-Güvenilir SCTP
PSTN	Public Switch Telephone Network - Genel Aktarmalı Telefon Şebekesi
QoS	Quality of Service - Hizmet Kalitesi
RFC	Request For Comments - Yorumlar İçin Talep
RTP	Real-time Transfer Protocol - Gerçek Zamanlı Taşıma Protokolü
RTT	Round-Trip Time - Gidiş Geliş Gecikmesi
SACK	Selective Acknowledgement – Seçici Onay
SCTP	Stream Control Transmission Protocol - Akış Denetimli İletişim Protokolü
SSN	Stream Sequence Number – Akış Sıra Numarası
STB	Set Top Box - Set Üstü Kutusu
TCP	Transmission Control Protocol - İletim Kontrol Protokolü
TDM	Time Division Multiplexing - Zaman Bölmeli Çoğullama
UDP	User Datagram Protocol - Kullanıcı Veri Bloğu Protokolü
VLAN	Virtual Local Area Network - Sanal Yerel Alan Ağı
VoD	Video on Demand - İsteğe Bağlı İzleme
VoIP	Voice over Internet Protokol - İnternet Protokolü Üzerinden Ses Yayını

Kısaltmalar**Açıklama****WLAN**

Wireless Local Area Network - Kablosuz Yerel Alan Ağı

WMN

Wireless Mesh Networks - Kablosuz Örgü Ağları

1. GİRİŞ

Bilgisayar teknolojisinin her alanda kullanıldığı çağımızda televizyon yayınları da bu teknolojiler arasında yerini almıştır. IPTV (Internet Protocol Television - İnternet Protokolü Üzerinden Televizyon) uygulamaları bilgisayar teknolojisi ile klasik televizyon yayınının yakınsamasıdır. Bu anlamda klasik televizyon yayını ile hem ortak hem de farklı özellikleri bulunmaktadır. IPTV klasik televizyon yayınının özelliklerini kendi içinde barındırmakla beraber bu teknolojiye yeni özellikler de eklemektedir. IPTV teknolojisinin tanımı ve mimarisi tezin ilerleyen bölümlerinde detaylı olarak açıklanacaktır.

IPTV bilgisayar ağları üzerinden yayın yapmaktadır. Bu, hem de internet iletişim katmanlarını ve farklı ağ protokollerini kullandığı anlamına gelir. Örneğin; Ağ katmanında IP (Internet Protocol - İnternet Protokolü), Ulaşım katmanında UDP (User Datagram Protocol - Kullanıcı Veri Bloğu İletişim Kuralları) gibi protokoller kullanılmaktadır. IPTV hizmet kalitesini etkileyen en önemli etkenler sırasında jitter, sırası bozulmuş paketler, yayına bağlanma gecikmesi, kanal değişim gecikmesi vb. problemler gelmektedir [1]. Yapılan bilimsel çalışmalarda farklı ağ katmanları için farklı protokoller kullanılarak bu problemlerin önlenmesi ve hizmet kalitesinin artırılması amaçlanmıştır.

Literatürde IPTV için farklı ulaşım katmanı protokolleri önerilmektedir. Önerilen ulaşım katmanı protokolleri içerisinde SCTP (Stream Control Transmission Protocol - Akış Denetimli İletişim Protokolü) protokolü gerçek zamanlı uygulamalarda başarılı olmuştur. SCTP protokolü 2000 yılında “Signaling Transport” çalışma grubu tarafından oluşturulmuştur. Bu protokol yapısı ve güvenliği itibarıyla diğer ulaşım katmanı protokollerinden farklılık gösterir [2, 3]. SCTP protokolü daha çok gerçek zamanlı veri iletiminde kullanılmaktadır [4]. SCTP protokolünün yapısı ve işleyişi ile ilgili detaylı bilgi tezin ilerleyen bölümlerinde verilmiştir.

IPTV konusunda literatürde çok sayıda çalışma mevcuttur.

IPTV için geniş içeriğe sahip kitap 2007 yılında G. Held tarafından “Understanding IPTV” adı altında hazırlanmıştır. Bu kitapta IPTV uygulamaları hem donanım hem de yazılım tabanlı olarak ele alınmış, internet üzerinden TV yayını ile IPTV’nin farkları ortaya koyulmuş ve kullanılan protokoller incelenmiştir [5].

IPTV uygulamaları konusunda yayınlanan makaleler arasında en geniş araştırma makalelerinden biri Y. Xiao ve ark. tarafından 2007 yılında yapılmıştır. Bu makalede IPTV hizmetleri ve ağ mimarisine değinilmiş, farklı ağ mimarileri için hizmet kalitesi, tıkanıklık denetimi vb. konular ele alınmıştır [1].

IPTV’nin ağ mimarisini açıklayan başka bir makale de 2007 yılında P. Arberg ve ark. tarafından hazırlanmıştır. Bu makalede IPTV mimarisi konusunda Ericsson şirketinin kullandığı yönteme yer verilmiştir [6].

S. T. Kim ve ark. 2007 yılında yaptıkları çalışmada IPTV uygulamaları için SCTP protokolünün genişletilmiş şekli olan PR-SCTP (Partial-Reliable SCTP – Kısmi-Güvenilir SCTP) protokolünün daha başarılı olduğunu göstermişlerdir [2].

S. Park ve ark. tarafından 2008 yılında yayınlanan makalede ise IPTV uygulamalarının sadece statik ağlarda değil hem de mobil kullanımından bahsedilmiştir. Yazarlar IPTV uygulamalarının IP bazlı iletişimin olduğu her yerde gerçekleştirileceğini göstermişlerdir. Ayrıca makalede mobil IPTV’nin kısıtlamaları da açıklanmıştır [7].

2009 yılında M. Baba ve arkadaşlarının yayınladığı “Buffer-based Low-Delay Playout Control Methods for IPTV Terminals” makalesinde yüksek jitter ve yüksek oynatma tamponuna sahip ağlarda başlangıç oynatma gecikmesi minimize edilmiştir. Bu çalışmada hem de benzer yöntemle jitterin yumuşatıldığı gösterilmiştir [8].

IPTV’de jitterin iyileştirilmesine yönelik çalışmalara örnek olarak Ted H. Szymanski ve D. Gilbert’in 2009 yılında beraber yaptıkları çalışma gösterilebilir [9]. Bu çalışmada daha önce birinci yazar tarafından yayınlanmış düşük jitter planlama

algoritması yardımı ile paket anahtarlama IP yönlendiricilerde jitterin minimize edildiği gösterilmiştir. Çalışmanın hem de “Telerobotic” ve “Telemedicine” gibi IP tabanlı uygulamalarda başarılı olabileceği öne sürülmüştür.

2011 yılında S. Boussen ve N. Tabbane “Using PR-SCTP for IPTV QoS Adaptation over IMS Network” isimli çalışmalarını yayınlamışlardır [3]. Bu çalışmada SCTP protokolünün gerçek zamanlı veri iletişimde TCP (Transmission Control Protocol – İletim Kontrol Protokolü) ve UDP protokollerinden daha iyi sonuçlar ortaya koyduğu gösterilmiştir. Makalede ayrıca mobil IPTV uygulamaları için de SCTP protokolü önerilmiştir.

A. Balk ve ark. tarafından 2002 yılında yayınlanmış makalede ns-2 (Network Simulator 2 – Ağ Benzetim Aracı) benzetim aracı üzerinde MPEG-4 (Moving Pictures Experts Group - Hareketli Görüntü Uzmanları Birliği) formatında video iletişimi sağlanmış ve SCTP protokolünün video uygulamalarında önemli derecede iyileştirme sağladığı gösterilmiştir [11].

Benzer şekilde M. Derini tarafından 2005 yılında yayınlanan “MPEG-4 Video Transfer with SCTP-Friendly Rate Control” başlıklı makalede ağdaki tıkanıklık durumuna bağlı olarak gönderim oranını düzenleyen protokol eklentisi ile SCTP protokolü ns-2 benzetim aracı üzerinde test edilmiş ve SCTP protokolünün UDP protokolünden daha başarılı olduğu gösterilmiştir [12].

K. H. Kim ve ark. tarafından 2009 yılında gerçek zamanlı veri iletişimi için iletişim kontrollü SCTP protokolünden bahseden makale yayınlanmıştır. Yazarlar bu makalede SCTP protokolüne ara katman ekleyerek alıcı kuyruğunun durumuna bağlı paket gönderimi gerçekleştirmişlerdir. Bu çalışmada yeniden gönderim oranını düşürerek daha iyi hizmet kalitesinin elde edilmesi amaçlanmıştır [13].

Bu tez çalışmasında IPTV teknolojisi için SCTP protokolü temel alınarak PQ-SCTP (Priority Queue SCTP – Kuyruk Önceliklendirmeli SCTP) protokolü geliştirilmiştir. PQ-SCTP protokolü gönderilen verinin içeriğine göre işlem yapmaktadır. Gönderici

tarafında verilere öncelik atanmakta, alıcı tarafında ise öncelik durumuna bağlı olarak sıralama yapılmaktadır. Sonuç olarak daha öncelikli paketler ön sıralara çıkmakta ve alıcı tarafında daha önce işleme alınmaktadır. Ayrıca protokol öncelikli paketlerin alıcı kuyruğundan atılma oranını da azaltmaktadır. Önceliklendirme işlemi gönderici tarafından veri içeriğine göre yapılabilir. Protokolde dört farklı öncelik durumu bulunuyor. Tekrar gönderilen paketler en yüksek önceliğe sahipken diğer öncelik grupları da 1.grup, 2.grup, 3.grup olarak sıralanmaktadır. En düşük öncelik 3.grup paketlere verilir ve bu paketler alıcı kuyruğunda en sonda yer alır. Protokolün işleyişi ve elde edilen deneysel sonuçlar tezin ilerleyen bölümlerinde detaylı olarak açıklanacaktır.

Bu protokol ile özellikle IPTV gibi gerçek zamanlı ve/veya hassas tekrar gönderim işlemine sahip uygulamalarda veri içeriği olarak önemli paketlerin, özellikle de tekrar gönderilen paketlerin alıcı kuyruğunda bekleme süresini, dolayısıyla jitteri ve IPDV'yi (IP Packet Delay Variation – IP Paket Gecikme Değişimi) düşürerek daha iyi hizmet kalitesi elde edilmiştir.

Bu tez altı bölümden oluşturulmuştur. Birinci bölüm çalışma özeti ve genel literatür taramasını içermektedir. İkinci bölümde IPTV teknolojisinin tanımı ve mimarisinden bahsedilmiştir. Üçüncü bölümde SCTP protokolü ve IPTV uygulamalarında bu protokolün kullanımı anlatılmıştır. Dördüncü bölüm geliştirilen PQ-SCTP protokolü ile ilgili çalışmayı içermektedir. Beşinci bölümde yapılan deneysel çalışmalar verilmiştir. Altıncı bölümde ise tez çalışması sonucunda elde edilen sonuçlar bulunmaktadır.

2. IPTV

Bu bölümde IPTV teknolojisinin tanımı ve tarihçesi, IPTV teknolojisinde kullanıcılara sunulan hizmetler, IPTV uygulamaları ve mimarisinden bahsedilecektir. Ayrıca içerik dosya formatları ve IPTV ile Internet TV arasındaki farklara değinilecektir.

IPTV 2000’li yılların başından gündemde olan bir konu olmasına rağmen bazı nedenlerden dolayı (geniş bant ağ altyapısının sağlanamaması, konu ile ilgili standartların belirlenmemesi, internet kullanımının yaygınlaşmaması vb.) yaygınlaşmamıştır. Gün geçtikçe bu sorunların ortadan kalkması ile beraber IPTV teknolojisi de hızlı bir şekilde yaygınlaşmaya başlamıştır. Bu teknoloji ile ilgili teknik bilgilerden bahsetmeden önce farklı kaynaklarda IPTV’nin nasıl tanımlandığına bakabiliriz. Örneğin; IPTV için Uluslararası Telekomünikasyon Birliği (ITU) şu tanımı yapmıştır [14].

“IPTV; televizyon, video, ses, metin, grafik, veri gibi çoklu ortam servislerinin güvenli, güvenilir, etkileşimli ve gerekli servis kalitesini sağlayacak şekilde IP tabanlı ağlar üzerinden iletilmesidir.”

Benzer şekilde Türkiye Telekomünikasyon Kurumu IPTV teknolojisini aşağıdaki şekilde tanımlamaktadır [15].

“IPTV, televizyon yayınlarının, geleneksel şekilde kablo TV, uydu veya havadan seyirciye iletimi yerine, internet teknolojileri kullanılarak geniş bant altyapısı üzerinden gerçekleştirilen yayın sistemidir.”

“SVS Telekom & Satellite Systems” tarafından verilen tanım ise şöyledir [16].

“IPTV, diğer multimedya hizmetleriyle beraber TV ve video sinyallerini, internet bağlantınızın ucundan alabilmenize olanak tanıyan çok kullanışlı bir sistemdir.”

Yukarıda verilen tüm tanımlardan görüldüğü gibi IPTV teknolojisi kendi bünyesinde birden fazla bilişim ve haberleşme teknolojisini barındıran karmaşık yapıya sahip bir sistemdir.

IPTV'nin kısa tarihçesi

Klasik televizyon yayınlarında istemci ile sunucu taraf arasında sadece tek taraflı iletişim sağlanmaktaydı. IPTV yayınlarında ise istemci ile sunucu arasında çift taraflı iletişim sağlanmaktadır. Örneğin; bir IPTV uygulamasında yayın durdurulabilir veya kullanıcının isteğine bağlı olarak ileri veya geri sarılabilir.

IPTV teknolojisinin gelişmesi ve uygulanması aşağıdaki gibi bir kronoloji izlemektedir.

- 1994 yılında ABC televizyon kanalı “CU-SeeMe videoconferencing” yazılımı kullanarak ilk kez internet üzerinden yayın yaptı.
- İngiltere’de bölgesel telekomünikasyon operatörü olan Kingstone Communications KIT (Kingstone Interactive Television – Kingstone Etkileşimli Televizyonu) projesi ile ilk kez 1999 yılında geniş bant üzerinden çeşitli TV ve VoD (Video on Demand – İsteğe bağlı Video) yayını yapan ilk şirket oldu. Bu yayın ADSL (Asymmetric Digital Subscriber Line - Bakımsız Sayısal Abone Hattı) altyapısı üzerinden son kullanıcıların hizmetine sunulmaktaydı.
- 1999 yılında NBTel (Bell Aliant olarak bilinmektedir) Kanada’da DSL üzerinden iMagic TV projesi ile IPTV dağıtımına başladı. Daha sonra bu proje Alcatel şirketi tarafından satın alındı.
- 2002 yılında Kanada’da SaskTel Lucent Stinger de DSL hattı üzerinden IPTV dağıtımına başladı. 2006 yılında bu proje ile Kuzey Amerika’da ilk kez HDTV (High Definition Television - Yüksek Çözünürlüklü Televizyon) yayını başlamış oldu.
- İsveç’te ilk IPTV hizmeti Bredbandsbolaget şirketi tarafından başlatıldı. 2009 yılında TeliaSonera şirketi bu projeyi devraldı.

- ABD’de IPTV hizmeti AT&T tarafından 2006 yılında U-Verse adı altında başlatıldı. 2009 yılından beri AT&T 100’den fazla yüksek çözünürlüklü televizyon kanalı IPTV projesini genişletmiş durumdadır. Diğer projelerden farklı olarak AT&T proje için özel bir IP ağı kurmuştur.
- TPG şirketi Avustralya’da ilk kez 2007 yılında ADSL2 + altyapısı üzerinden IPTV servis sağlayıcılığını üstlendi. Şirket 45’e yakın yerel ve uluslararası kanal yayını yapmaktadır.

2.1. IPTV Hizmetleri

IPTV servisleri kendi bünyesinde birden fazla farklı hizmeti barındırmaktadır. Temel olarak bu hizmetler çoklu televizyon yayını, VoD, triple play, VoIP (Voice over IP - IP üzerinden ses yayını), web/email erişimi, adreslenebilir reklamcılık olarak gruplandırılabilir. Bu hizmetler klasik televizyon yayınından çok farklıdır. IPTV hizmeti bir anlamda iletişimin, yayının ve erişimin yakınsamasıdır [15, 17].

2.1.1. Çoklu televizyon yayını

IPTV’den bahsedildiğinde ilk akla gelen kavram internet altyapısı üzerinden televizyon yayının yapılmasıdır. Bu teknolojiyi kullanan aboneler en temel hizmet olarak dijital televizyon yayınlarından yararlanabilmektedirler.

Sayısal işaret işleme teknolojisinin kullanıldığı ilk televizyon yayını 1983 yılında başlamıştır. Daha sonra görüntülerin sayısal ortamda tutulması ve aktarılmasının yaygınlaşması ile beraber bu konuda standartlaştırma yönüne gidilmiştir. İlerleyen zamanlarda MPEG tarafından MPEG-2 vb. görüntü ve ses sıkıştırma yöntemleri dijital televizyon yayıncılığında kullanılmaya başlanmıştır. Günümüzde her ne kadar analog televizyon yayınına devam eden ülkeler bulunsa da bu sayı gittikçe azalmaktadır ve önümüzdeki yıllar içinde analog yayının yerini tamamen dijital yayına devredeceği beklenmektedir. Bu hizmete servis kalitesi açısından baktığımızda da dijital yayının analog yayından daha verimli ve daha kaliteli yayın yaptığı görülmektedir [18].

IPTV teknolojisi kendi içinde en temel hizmet olarak bulundurduğu dijital yayın hizmetini abonelere bilindik yayın hizmetinden farklı olarak daha esnek yapıda sunmaktadır. IPTV kullanıcıları izlemek istedikleri yerel veya uluslararası kanalları kendileri belirlemektedirler. Klasik televizyon yayıncılığında tüm kanallar kullanıcının evine kadar gelmektedir. IPTV’de ise sadece kullanıcının izlemek istediği kanalın yayını yapılmaktadır [16].

2.1.2. İsteğe bağlı video (VoD)

İsteğe bağlı video hizmeti IPTV teknolojisinin en meşhur hizmetlerinden biri olarak bilinmektedir. VoD hizmetinde kullanıcılara sadece kanalların değil hem de televizyon programlarını tercih ettikleri zaman ve tercih ettikleri sıraya göre izlenme olanağı sunulmaktadır. Aslında isteğe bağlı görüntü hizmeti CoD (Content on Demand – İsteğe Bağlı İçerik) hizmetinin bir alt hizmetidir. IPTV hizmetleri arasında çok tutulması beklenen bu hizmet hem de ağ altyapısında yüksek bant genişliği ve servis sağlayıcı tarafında büyük veri depolama birimlerinin bulunmasını gerektirmektedir.

VoD hizmeti altında hem de yayına göre ödeme vb. hizmetler sunulmaktadır. Bu hizmetin amacı kullanıcı herhangi bir programı tercih ettiğinde tüm kanal için değil sadece o program yayını için ödeme yapmasıdır.

2.1.3. Triple play hizmeti

Bu hizmet IPTV üzerinden eşzamanlı olarak ses, veri ve görüntü bilgisinin aktarılmasına dayanmaktadır. Bu hizmet sayesinde kullanıcılar gerçek zamanlı görüntü izlemekle beraber anlık mesajlaşma, çevrimiçi oyun gibi hizmetlerden de yararlanabilmektedirler.

2.1.4. Web/mail hizmeti

Web/mail hizmeti sayesinde kullanıcılar IPTV üzerinden web ve mail sunucularına da erişebilirler. Bu hizmet bilgisayarın açık olmadığı veya tamamen olmadığı durumlarda abonelerin televizyon üzerinden web üzerinde gezinme ve mail alışı – verişi yapma fırsatı sunmaktadır. Bu hizmet için oluşturulmuş altyapı kullanılarak sosyal ağlarda yapılan güncellemeler de televizyon üzerinden kullanıcıya sunulabilir [1, 19].

2.1.5. VoIP hizmeti

Günümüzde VoIP hizmeti bazı anlık mesajlaşma uygulamalarında kendi başına kullanılmaktadır. Bu hizmeti sunan servis sağlayıcıları bilgisayar kullanıcılarının kullandığı uygulamaları sadece kendi aralarında haberleştirmekle kalmayıp hem de PSTN (Public Switched Telephone Network - Genel Aktarmalı Telefon Şebekesi) şebekesi üzerinden ev ve cep telefonlarıyla da bağlantı kurmaktadır. IPTV teknolojisi kapsamında VoIP hizmeti de aynı mantıkla yürütülmektedir. Ancak bunun yanı sıra geniş bant IPTV hattı üzerinden sadece ses değil hem de görüntü iletilebilir. Sonuç olarak IPTV televizyon üzerine monte edilmiş bir web kamerası aracılığı ile görüntü konuşma olanağı sunabilir [16].

Uludağ Üniversitesinde yapılmış başka bir yüksek lisans çalışmasında da SIP tabanlı telefon servislerinin IPTV üzerinde uyarlanabilir olduğu gösterilmiştir [19].

2.1.6. Bireysel reklamcılık

IPTV'nin kullanıcılara sunduğu diğer hizmetler göz önünde bulundurulduğunda her abonenin kendine özel portalının oluşturulması ihtiyacı ortaya çıkmaktadır. Kullanıcı portallarının var olması aboneye özel reklam ve duyuruların iletilebilmesi anlamına gelmektedir. Bu şekilde reklam pazarının başarısı ölçeklenebilir hale gelir. Ayrıca portal üzerinden abonenin isteğine bağlı olarak daha önce izlediği programlar, tercih

ettiği kanallar, beğendiği mesajlar ve reklamlar göz önünde bulundurularak bireysel reklamcılık yapılabilir [16].

IPTV servis sağlayıcıları hem de yerel tabanlı çalışacakları için abonelere yayın yapılan reklam ve duyurular da yerel olacaktır.

2.2. IPTV ve Internet TV

Internet protokolü üzerinden TV yayını yapılması genelde Internet üzerinden Televizyon yayını yapılması anlamına gelmektedir. Ancak, bu iki kavram bir birinden çok uzaktır ve kesinlikle karıştırılmaması gerekmektedir. IPTV ve Internet TV üzerindeki en temel fark Internet TV'nin bilinen internet ağı üzerinden gerçek zamanlı veya gerçek zamanlı olmayan görüntü yayını yapmasına rağmen IPTV kendine özel ağ altyapısı üzerinden yayın yapmaktadır. IPTV ve Internet TV arasındaki diğer önemli farklar bu şekilde sıralanabilir [19-21];

- IPTV servis sağlayıcısına bağlı olarak sadece belirli bir alanda yayın (dağıtım) yapmaktadır. Internet TV için ise bu konuda herhangi kısıtlama söz konusu değildir. İnternetin erişilebildiği her yerde Internet TV yayını izlenebilir.
- IPTV abonelerine belirli bir hizmet kalitesi QoS (Quality of Service – Hizmet Kalitesi) garantisini vermektedir. Internet TV'de ise böyle bir garanti söz konusu değildir.
- IPTV yayının abone tarafından izlenebilmesi için alıcı cihazın (STB - Set Top Box) ilerde detaylı olarak anlatılacaktır) ve TV alıcısının olması gerekmektedir. Internet TV ise herhangi kişisel bilgisayar üzerinden izlenebilir.
- IPTV abonelerine güvenilir yayın sağlamaktadır. Internet TV'de ise böyle bir garanti verilemez.
- IPTV yayını kendi içinde kapalı olduğu için kullanıcıların hesapları güvenli ortamda tutulmaktadır. Internet TV'de ise kullanıcı açık internet ortamında olduğundan böyle bir güvenlik söz konusu değildir.

- IPTV hizmeti belirli şirketler tarafından sağlandığından kurulum sonrası teknik destek verilmektedir. İnternet TV’de ise böyle bir destek mevcut değildir.
- IPTV üzerinden yayınlanan içerik telif hakları ile korunmaktadır. İnternet TV’de ise genel olarak böyle bir koruma yoktur.
- IPTV yayınında genel olarak MPEG-2, MPEG-4, H.264 gibi formatlar kullanılmaktadır. İnternet TV yayınlarında ise Windows Media Player, Quick Time, Flash ve benzer teknolojilere başvurulmaktadır.
- IPTV kayıtlı kullanıcı ve IP adreslerine yayın yapmaktadır. İnternet TV’de ise internet ağı üzerinde bulunan herhangi bir kullanıcı verilen yayını izleyebilmektedir.
- Kaliteli ve sürekli IPTV yayını için en az 4 Mb/s hız gerekmektedir. İnternet TV’de ise hız internet hizmetini sağlayan servis sağlayıcısına göre farklılık göstermektedir.

2.3. IPTV Uygulamaları

Günümüzde yaygın IPTV uygulamalarına örnek olarak Microsoft TV gösterilebilir. Microsoft TV uygulaması bilinen IPTV hizmetlerine ek olarak DVR (Digital Video Recorder - Sayısal Video Kaydedici), ICC (Instant Channel Change - Anlık Kanal Değişimi) gibi hizmetler de sağlamaktadır. Bu uygulama telekomünikasyon hizmet sağlayıcısı olan BellSouth firması tarafından hizmete sunulmaktadır. Başka bir örnek İngiltere’de Video Networks Ltd. (VNL) tarafından Cisco Internet Protocol kullanılarak hizmete sunulan HomeChoice uygulamasıdır. HomeChoice uygulamasında triple play hizmeti DSL ağ altyapısı üzerinden sunulmaktadır. Bu teknoloji ile ilgili AT&T ve Verizon firmaları son kullanıcılara erişim için fiber optik kablo altyapısı kullanmayı teklif etmiştir [1].

2.4. IPTV Mimarisi

IPTV uygulamalarının geniş çapta yaygınlaşması için fiziksel altyapıda değişiklikler gerekmektedir. Birden fazla hizmeti aynı anda ve kaliteli şekilde (QoS garantili)

sunabilmek için geniş bant genişliği IPTV teknolojisi için olmazsa olmazlardandır. Tezin bu bölümünde IPTV teknolojisinin temel mimarisinden ve geniş bant ağ altyapısı için kullanılabilecek günümüz ağ teknolojilerinden bahsedilecektir.

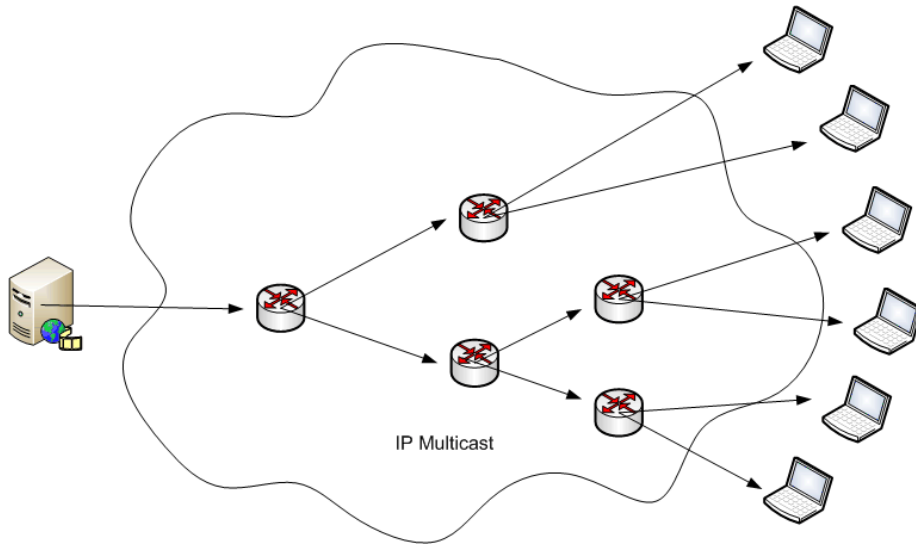
IPTV hizmetinde omurga yapı olarak ADSL2, ADSL2+, VDSL gibi teknolojiler kullanılabileceği gibi yüksek hızlı taşıyıcı sınıf Ethernet (high speed carrier grade Ethernet) veya yüksek hızlı IEEE 802.11n kablosuz yerel ağ bağlantısı da kullanılabilir. IPTV uygulaması veri, ses ve videonun birleşimi olduğu için altyapı olarak da farklı gereksinimler doğurmaktadır. Örneğin basit anlamıyla bir IPTV hizmetinin sağlanabilmesi için kaynaklar (VoD veri tabanı, programlar), yüksek hızlı omurga yapı, DSL, Carrier Ethernet, WLAN (Wireless Local Area Network – Kablosuz Yerel Alan Ağı) veya fiber optik altyapılı dağıtım ağı ve kullanıcılar için IPTV cihazları (HDTV, STB vb.) gerekmektedir. IPTV hizmetinin temel kısıtları ise çok fazla verinin depolanması, QoS'in sağlamanın zorluğu ve maliyetin yüksek olmasıdır. QoS hem video hem de ses yayınlarında dikkate alınmalıdır. QoS için temel engeller jitter, paketlerin kaybolma olasılığı, bağlantı kopma olasılığı, boş kanal durumu, kanal seçme süresi vb. şekilde sıralanabilir [20, 22].

IPTV topolojisi ile ilgili Yang Xiao ve ark. "Internet Protocol Television (IPTV): The Killer Application for the Next-Generation Internet" başlıklı makalesinde şunları söylemektedir [1].

"IPTV ağı kendi içinde Erişim ağı (Access Network) ve Omurga ağı (Core Network) olarak iki ağdan oluşmaktadır."

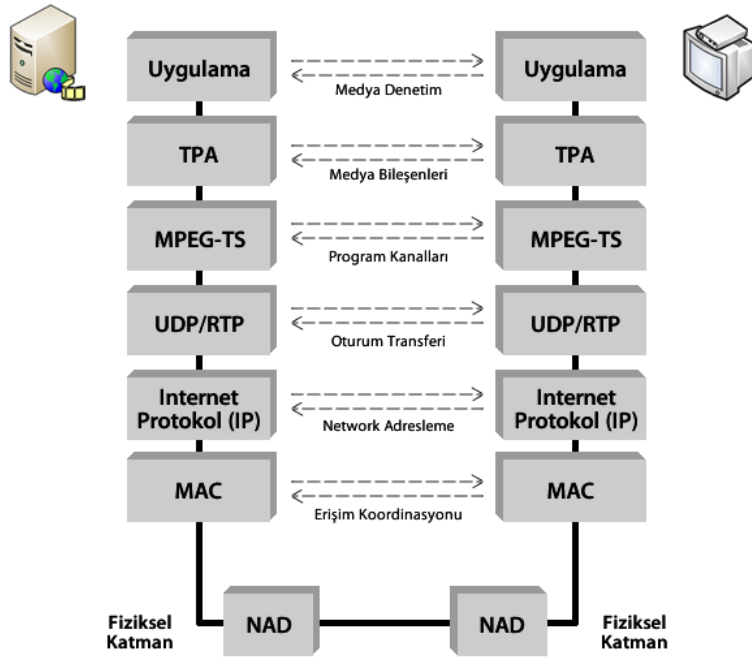
Omurga ağı IPTV yayın hizmetini sunan, uydu ve karasal yayınlara bağlı ana merkezden dağıtım noktalarına yapılan bağlantı için kullanılmaktadır. Erişim ağı ise dağıtım noktaları ile kullanıcılar arasındaki bağlantı için kullanılmıştır. Erişim ağı için DSL, Carrier-Grade Ethernet, FTTH (Fiber To The Home) gibi teknolojiler kullanılabilir. Gerçek uygulamalarda en çok DSL teknolojisi tercih edilmektedir. IPTV servislerinin literatürde iki farklı şekilde dağıtıldığından bahsedilmektedir.

1. Peer-to-peer dağıtım: peer-to-peer dağıtımda ağda bulunan her bir düğüm, hem istemci hem de sunucu görevini yapmaktadır. Ağa yeni bir düğüm eklendiğinde bu düğüm diğer düğümlerden veri aldığı gibi kendisi de başka düğümlere bilgi akışı sağlamaktadır. Peer-to-peer yapısı dağıtım için mantıklı gözükse de, yeterli seviyede QoS sağlayamadığı için gerçek uygulamalarda tercih edilmemektedir.
2. Multicast dağıtım: multicast dağıtımda ana merkezden gönderilen paketler, ağ üzerindeki yönlendiricilerde kopyalanarak çoğaltılmakta ve istemcilere gönderilmektedir (Şekil 2.1).



Şekil 2.1. IP multicast [1]

Multicast yayın MPLS (Multi Protocol Label Switching – Çoklu Protokol Etiket Anahtarlama) anahtarlama kullanılarak gönderilmektedir. İstemci ve Sunucu arasında bağlantı ise IGMP (Internet Group Management Protocol – İnternet Grup Yönetim Protokolü) protokolü yardımı ile kontrol edilmektedir. Transport katmanında UDP ve RTP (Real-time Transport Protocol – Gerçek Zamanlı İletişim Protokolü) protokolleri kullanılmaktadır. IPTV servislerinin katmanlı mimarisi Şekil 2.2’de görülebilir.



Şekil 2.2. IPTV ağ katmanları [20]

Yang Xiao ve ark. yaptığı çalışmada MobileTV, IPTV servislerinde QoS garantisi, erişim ve omurga ağlarında kullanılan farklı dağıtım teknolojileri için kabul kontrollerinden (admission control), IPTV tıkanıklık kontrolünden bahsedilmiştir.

Peter Arbeg ve ark. yayınladığı “Network infrastructure for IPTV” makalesinde de IPTV servisleri için Ericsson’un geliştirdiği ve kullandığı topolojiden bahsedilmiştir [6]. Ericsson topolojisinde de multicast yapı kullanılmıştır. Ericsson topolojisinin temel noktaları şunlardır:

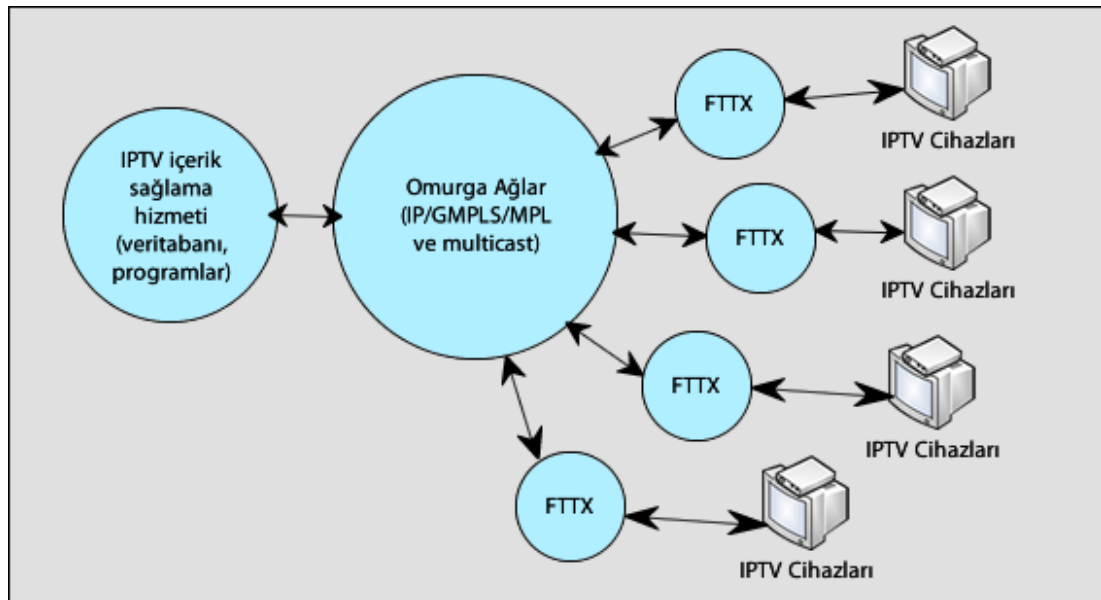
1. Access node – DSL, fiber üzerinden Ethernet veya GPON (Gigabit Passive Optical Network – Gigabit Pasif Optik Ağ) kullanan istemci ve metro toplama düğümleri (metro aggregation nodes) arasındaki bağlantılar.
2. Metro toplama düğüm – IP edge düğümler ve Access düğümler arasında bağlantıyı gerçekleştiren düğümler.
3. IP edge düğüm – IPTV head end’den gelen paketleri dağıtan ve policy-control yardımı ile QoS denetleyen birim.

Bu çalışmada erişim ağlarının (Access Networks) hem multicast hem de unicast servislerin her ikisinin desteklemesi gerektiği gösterilmiştir.

Multicast dağıtım QoS'in sağlanması açısından daha uygun olduğu için gerçek uygulamalarda kullanılmaktadır. Ancak, bu yapının da bazı problemleri bulunmaktadır. Kaybolan veri paketleri, tıkanıklık, sırasız gelen paketler, paket kayıpları, kanal değişim gecikmesi, paketlerin gecikmesi ve jitter bu problemlerden bazılarıdır.

IPTV genel mimari yapısı Şekil 2.3'de görülmektedir. Bu yapıdan da görüldüğü gibi IPTV mimarisi üç temel bileşenden oluşmaktadır.

- Servis sağlayıcı tarafı
- Abone, yani son kullanıcı tarafı
- Servis sağlayıcı ve abone arasında bağlantıyı sağlayan ağ altyapısı



Şekil 2.3. Genel IPTV Mimarisi [1]

Servis sağlayıcının temel işlevi farklı kanallardan (uydu, karasal vb.) TV sinyallerini alıp aboneye iletmektir. Bunun yanı sıra servis sağlayıcılar abone portallarının yönetiminden de sorumludurlar.

Aboneler sistemin diğer ucunda bulunmaktadır. Aboneler ev veya ofislerinde bulunana televizyon cihazlarını STB ile beraber çalıştırarak servis sağlayıcılarının ilettiği sinyali almaktadırlar.

Ağ altyapısı servis sağlayıcısına bağlı olarak farklı şekillerde düzenlenebilir. En temel şekliyle ağ altyapısı kablolu veya kablosuz olarak gruplandırılabilir. IPTV ağ altyapısı kendi içinde de 2 farklı kısma ayrılmaktadır [1];

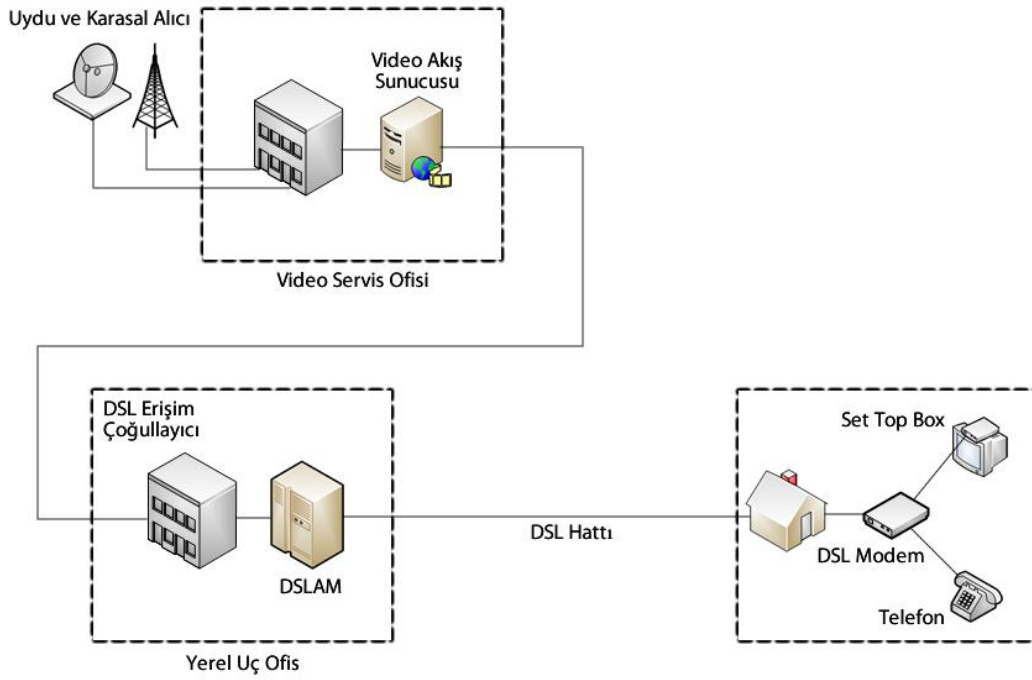
- Erişim Ağı (Access Network)
- Omurga Ağı (Core Network)

2.5. Erişim Ağları

IPTV’de erişim ağları abonelerle omurga ağ arasında bağlantı sağlamaktadır. Erişim ağları farklı ağ teknolojileri kullanılarak tasarlanabilir. Bu teknolojilere örnek olarak DSL, Carrier-grade Ethernet, Fiber-optik, WiMAX vb. gösterilebilir. Kullanılan her bir teknolojinin kendine göre avantaj ve dezavantajları bulunmaktadır [23].

2.5.1. DSL ağlar

Bilindiği gibi DSL teknolojileri servis sağlayıcılarına bağlı olarak 128Kb/s ~ 24Mb/s farklı hızlar sunabilmektedir. Örneğin, ADSL standartları 2km mesafede 8Mb/s hızla gönderim yapabilirken ADSL2 + 24Mb/s hız sağlayabilmektedir. DSL teknolojisinin başka bir avantajı da modemlerin birden fazla bilgisayarı Ethernet, HomePlug veya IEEE 802.11 kablosuz LAN (Local Area Network – Yerel Alan Ağı) ile birleştirebilir olmasıdır. DSL teknolojisi IPTV uygulamalarının erişim ağlarında ilk seçim olarak gözükmemektedir [24]. Şu an Türkiye’de IPTV yayını sağlayan Türk Telekom şirketine bağlı “Tivibu” hizmeti ADSL ağ altyapısını kullanarak abonelerine hizmet sunmaktadır. Şekil 2.4’de DSL ağ yapısı verilmiştir.



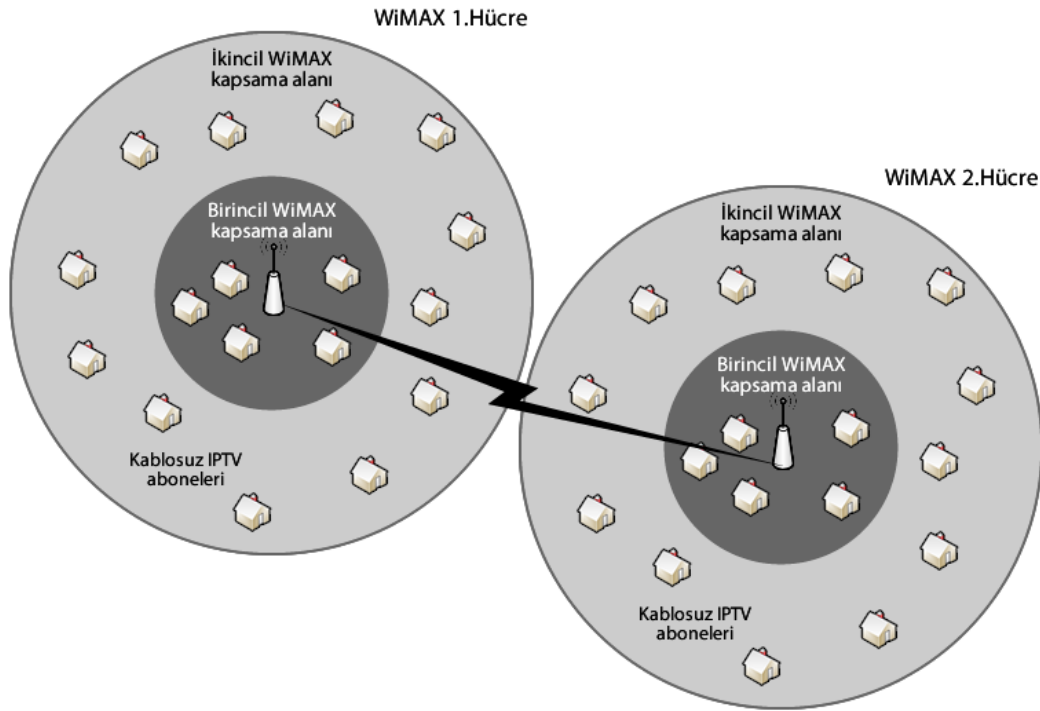
Şekil 2.4. DSL ağın yapısı [19]

2.5.2. Carrier-grade ethernet

Carrier-grade Ethernet 10Gb/s kadar hız sunabilmektedir. Ayrıca VLAN (Virtual Local Area Network – Sanal Yerel Alan Ağı) teknolojisi ile unicast/multicast/broadcast yayın yapılabilir. IPTV uygulamaları için geniş bant genişliği ihtiyacı ve QoS garantisi dikkate alındığında Carrier-grade Ethernet ağların kullanımı uygun gözükmektedir.

2.5.3. Kablosuz ağlar

Mobil cihazlarda IPTV uygulamalarının desteklenmesi için ise yüksek hızlı IEEE 802.11n kablosuz LAN kullanılabilir [7, 10]. Gazi Üniversitesinde E. Mahmoudian tarafından yapılan başka bir yüksek lisans tezinde ise IPTV yayınlarının gerekli QoS standartlarının gözeterek WiMAX üzerinden yapılabileceği gösterilmiştir [26]. Tipik bir WiMAX ağ yapısı Şekil 2.5’de sunulmuştur.



Şekil 2.5. Tipik WiMAX ağ yapısı [25]

2.5.4. Fiber-optik ağlar

Fiber ağlar düşük maliyete daha yüksek bant genişliği sunduğundan IPTV için kullanışlı ağ teknolojisi olarak gözükmektedir. Ayrıca, elektromanyetik etkenlerden etkilenmemesi de fiber ağların tercih sebeplerindedir.

Fiber optik ağların yüksek hız olanağından yararlanabilmek için Fiber To The Cable veya Fiber To The Carrier grade Ethernet (FTTC) teknolojisi kullanılabilir. FTTC teknolojisinde temel mantık kullanıcılara yakın mesafelerde fiber ağın ethernet ağı ile değiştirilmesidir. Fiber-optik ağlar birkaç farklı kategoride ele alınabilir;

- FTTR (Fiber To The Regional Office – Bölgesel Ofise Kadar Fiber). Bu şekilde fiber optik kablolar ana yayın merkezinden bölgesel merkeze kadar gelmektedir. Daha sonra bölgesel merkez diğer dağıtım teknolojilerini kullanarak sinyali kullanıcılara iletmektedir.

- FTTC (Fiber To The Cabinet/Curb – Kabinete Kadar Fiber). Bu şekilde fiber ađlar ana yayın merkezinden abonelere yakın noktalarda sokaklarda bulunan güvenli kabinetlere kadar gelmektedir.
- FTTN (Fiber To The Node – Noktaya Kadar Fiber). FTTN ađlarda fiber kablo ile abone arasındaki mesafe 300m'den daha fazladır.
- FTTH (Fiber To The Home – Eve Kadar Fiber). Bu şekilde fiber kablolar ana yayın merkezinden abonenin evine kadar gelmektedir.
- FTTA (Fiber To The Antenna – Antene Kadar Fiber). FTTA ađlarda fiber kablo abonelerin bulunduğu apartmana kadar gelmekte, apartman için dağıtım ise bakır kablolarla yapılmaktadır.

2.6. Omurga Ađları

IPTV teknolojisi adresleme için IP kullanmaktadır. Taşıma katmanında ise üç farklı protokol tercih edilebilir;

- TCP: bu protokol gönderilen paketlerin doğruluđunu, hedefe ulaşıp ulaşmadığını kontrol etmektedir. Bu nedenle diđer protokollere kıyasla daha yavaş çalışmaktadır. IPTV için TCP protokolünün kullanılması iletim hızında düşmelere neden olabilir.
- UDP: UDP protokolü TCP'den farklı olarak veri paketleri üzerinde herhangi kontrol işlemi yapmamaktadır. UDP protokolü veri paketinin sadece karşı tarafa gönderilmesinden sorumludur. Paketin hedefe iletilip iletilmemesi ile ilgilenmez. Bu özelliklerinden dolayı UDP protokolü TCP protokolünden daha hızlı çalışmaktadır.
- RTP: bu protokol ses ve video gibi gerçek zamanlı verinin internet ortamında taşınması ihtiyacı nedeniyle ortaya çıkmış bir protokoldür. Ses ve video gibi verilerin iletilmesinde alıcı için paketlerin doğrulanmasından veya kayıp paketlerin tekrar gönderilmesinden daha önemli paketlerin belirli aralıklarla sabit hızla gönderilmesidir. Bir anda hızlanan veya yavaşlayan veri iletişimi sesin veya görüntünün anlamsız şekilde kullanıcıya sunulması anlamına

gelmektedir. RTP protokolü yukarıda bahsedilen sorunları ortadan kaldırmaktadır. Genel olarak RTP protokolü video ve ses gibi çoklu ortam öğelerinin iletilmesinde TCP protokolünün istenmeyen yönlerini kaldırmakta ve UDP protokolü üzerine gerekli işlevleri eklemektedir.

Omurga ağlar için kullanılacak teknolojiler de birkaç başlık altında toplanabilir [25];

- ATM (Asynchronous Transfer Mode – Eş zamansız Aktarım Modu)
- MPLS
- SONET
- Metro Ethernet
- Multicast yayın ve IGMP

2.6.1. ATM

ATM yüksek bant genişliğine sahip ağlarda kullanılmaktadır. Daha çok bakır koaksiyel ve fiber-optik ağlarda tercih edilmektedir. Fiber-optik ağlarda en yüksek verimlilik elde edildiği gözlemlenmiştir. ATM’de veri paketi 5 baytlık başlık ve 48 baytlık veri bölümlerinden oluşmaktadır.

2.6.2. MPLS

MPLS ağ üzerinde etiketlemeğe dayalı yönlendirme yapmaktadır. Böyle bir ağda yönlendiriciler (routerler) kendi aralarında sanal yollar oluşturmakta ve bu yollar üzerinden veri iletişimi yapılmaktadır. Gönderilen paketlerin başına gideceği yönlendiricinin adresi yazılmaktadır. MPLS’in en büyük avantajı paket kontrollerinin her düğümde yapılmamasıdır. Bu da ağ üzerinden paketlerin daha hızlı hareket etmesine neden olmaktadır.

2.6.3. SONET

SONET birden fazla veriyi beraber göndererek hızlı iletişim sağlamaktadır. Bu amaçla kendi içinde TDM (Time Division Multiplexing - Zaman Bölmeli Çoğullama) mantığını kullanmaktadır. SONET protokolü de ATM gibi fiber optik ağlarda yüksek verimlilik göstermektedir [25].

2.6.4. Metro ethernet

Metro Ethernet teknolojisi 100Gb/s hıza kadar destek sağlamaktadır. Bunun yanı sıra çok hızlı veri iletimi ve düşük veri kaybına neden olmaktadır. Metro Ethernet yapısı IPTV yayınlarında daha çok merkez yayın ofisi ile bölgesel yayın ofisleri arasında veri iletişimi sağlamak amacıyla kullanılabilir.

2.6.5. Multicast ve IGMP

Multicast yapıda aynı anda birden fazla kullanıcıya eşzamanlı olarak dağıtım yapılmaktadır. Multicast protokoller iletişim için IP/TCP/RTP protokollerini kullanmaktadır. Birden fazla alıcıya ses, veri ve video gönderimi yapılabilir.

Multicast yayınlar genelde IGMP protokolü üzerinden yapılmaktadır. IGMP protokolü ses, video görüntü ve çevrimiçi oyunlar gibi sürekli veri iletiminde (streaming) kullanılmaktadır.

2.7. İçerik Dosya Formatları

IPTV uygulamalarında görüntü içeriğinin formatlanması için genel olarak MPEG-2, MPEG-4, H.264 gibi görüntü formatları kullanılmaktadır. Bu formatların tercih edilmesindeki en büyük neden ister sabit, isterse de değişken bit hızlarında (bit rate) yüksek sıkıştırma oranı sağlamaları ve kalite ile bant genişliği arasında doğru bir şekilde dengeleme yapmalarıdır. Ancak STB'lerin ve IPTV için kullanılacak diğer aygıtların tasarımlarının basitleştirilmesi ve farklı cihazların beraber çalışabilmesi açısından tek formatın belirlenmesi çalışmaları sürdürülmektedir.

IPTV hizmetinde farklı servisler ve farklı formatlar için deęişik bant genişlięi deęerleri hesaplanmaktadır. Bu deęerler izelge 2.1’de grlebilir.

izelge 2.1. Servisler ve gerekli bant genişlikleri [19]

Servis adı	Kodlama tipi	Bant genişlięi (Mb/s)
HDTV	MPEG2	16
SDTV	MPEG2	5
VoD	MPEG2	5
HDTV	MPEG4	6
SDTV	MPEG4	2
VoD	MPEG4	2

2.8. STB (Set Top Box)

Set Top Box IPTV hizmetinin en nemli bileşenlerindendir. STB son kullanıcının evinde/ofisinde servis saęlayıcısından gelen sinyali dijital grntye dnştrerek televizyon veya bilgisayara aktarmaktadır.

STB cihazlarının grevi modem işlevi ile sınırlı deęildir. STB’ler hem de uzaktan kumanda ile alıřmakta ve yayını kaydetme, ileri/geri sarma, durdurma oynatma gibi grevleri de yerine getirmektedir.

İnternette gezinen, web/mail hizmetleri gibi IPTV teknolojisini saęladıęı hizmetler de STB aracılıęı ile yapılmaktadır. Airties, Nokia vb. teknoloji firmaları bu amalara hizmet eden STB cihazları retmektedir.

3. IPTV VE SCTP

Tezin önceki bölümlerinde de bahsedildiği gibi IPTV uygulamalarında farklı ulaştırma katmanı protokolleri kullanılmaktadır. UDP ve TCP protokolleri bunlara örnek olarak gösterilebilir. Bilindiği gibi UDP ve TCP protokollerinin tercih edilmesi bazı nedenlere dayanmaktadır. Örneğin; gerçek zamanlı uygulamalarda bağlantının güvenilirliğini dikkate almayan UDP protokolü tercih edilirken, gönderilen verinin hedefe eksiksiz olarak ulaşmasının önemli olduğu uygulamalarda TCP protokolü kullanılmaktadır. Literatürde SCTP protokolünün IPTV’de kullanılmasını öneren birkaç çalışma bulunmaktadır. Bu bölümde özellikle SCTP protokolü üzerinde yapılan çalışmalar, IPTV uygulamalarında SCTP protokolünün kullanımı ve SCTP protokolü ile diğer ulaşım katmanı protokolleri arasındaki ortak ve farklı özelliklerden bahsedilecektir.

3.1. SCTP ve Diğer Ulaşım Katmanı Protokollerinin Karşılaştırılması

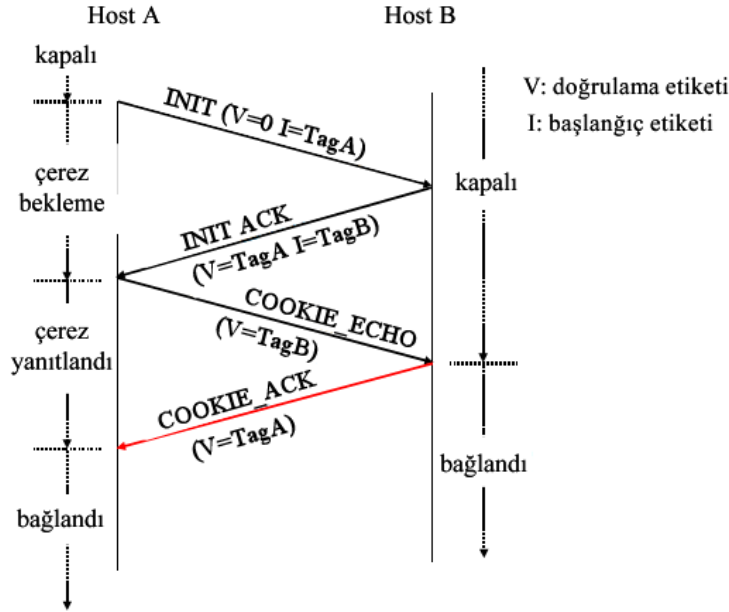
SCTP Protokolü özellikle veri akış (streaming) uygulamaları göz önünde bulundurularak geliştirilmiş bir protokoldür. SCTP Protokolü mesaj tabanlı (message-oriented) çalışmaktadır. H.248, H.323 ve SIP gibi üst katman protokolleri daha karmaşık hizmet talep ettiğinden UDP ve TCP protokolleri yetersiz kalmaktadır [2]. SCTP protokolü mesajların sıralı ve güvenli şekilde iletilmesini sağlar. IETF (Internet Engineering Task Force - İnternet Mühendisliği Görev Gücü) tarafından son hali ile RFC (Request For Comments - Yorumlar İçin Talep) 4960 belgesinde yer almaktadır [27].

3.1.1. SCTP protokolünün çalışma şekli

SCTP veri iletiminde doğrulama ve onay mekanizması kullanılmaktadır. Bu sayede Flood ataklarından korur, veri parçalarının kaybolmasını ve kopyalanmasını haber verir. SCTP protokolü veri transferi için özel yol seçmekte ve yolun bağlantısını kontrol etmektedir.

Diğer transport katmanı protokollerinden farklı olarak çoklu veri akışı (multistream) yapısını desteklemektedir.

SCTP protokolü TCP protokolünden farklı olarak 4-way handshaking (dört aşamalı bağlantı kontrolü) kullanmaktadır. Bu sayede SYN saldırıları engellenmektedir [4]. 4-way handshaking işlemi Şekil 3.1’de görülmektedir.



Şekil 3.1. 4-way handshaking

TCP protokolünde 3-way handshaking işlemi kullanılmaktadır. Bu işlemlerde sırasıyla SYN, SYN_ACK ve ACK mesajları istemci ve sunucu arasında gidip gelmektedir. SYN saldırısı TCP protokolünde son ACK mesajının gönderilmemesi ve peş peşe SYN mesajının gönderilmesi sonucunda oluşur. SCTP protokolünde son COOKIE_ACK mesajının sunucu tarafından gönderilmesi SYN saldırısını önlemektedir.

HOL blocking önleme

HOL (Head of Line - Hat Başı) Blocking problemi TCP protokolünde bir paketin kaybolma veya bozulmadan dolayı sürekli tekrarlanması sonucunda ortaya çıkar. HOL Blocking iletişim sırasında tıkanıklığa neden olur. Bu problemi önlemek için

TCP protokolünde birden fazla TCP bağlantısı kullanılır. Bu da daha fazla handshaking işlemi ve daha fazla SYN kayıplarına neden olmaktadır. SCTP protokolünde ise birden fazla kanalda veri gönderimi yapıldığından dolayı HOL Blocking problemi ortaya çıkmaz [4].

Multihoming ve DAR

SCTP protokolünün diğer önemli özelliği multihoming'dir. Diğer ulaşım katmanı protokollerinden farklı olarak SCTP protokolünde alıcı ve/veya gönderici tarafında birden fazla IP adresi ve port numarası tanımlanabilmektedir. Bu adresler kendi aralarında haberleşebilir. Bu özellik sayesinde ağ üzerinde paket kayıplarının oluşması durumunda alternatif adresler kullanılabilir. IETF tarafından tanımlanmış RFC 5061 belgesinde SCTP protokolü genişletilerek multihoming ile beraber hem de DAR (Dynamic Address Reconfiguration – Dinamik Adres Düzenlenmesi) özelliği eklenmiştir [28]. DAR eklentisi bağlantıyı (association) yeniden başlatmadan adresde düzenleme yapmaktadır. DAR özelliği sadece SCTP protokolünde bulunmaktadır ve uygulamaların bağlantı zamanlarında iyileştirme yapmaktadır. Bu eklenti özellikle mobil ağlarda etkin olarak kullanılabilir.

Kısmi güvenilirlik

IETF tarafından yayınlanmış RFC 3758 belgesinde SCTP protokolü için geliştirilmiş kısmi güvenilirlik eklentisi anlatılmaktadır [29]. Bu eklenti sayesinde ağ üzerinde paketlerin yeniden gönderim oranları düşürülür. Kısmi güvenilirlik eklentisi R. Stewart ve ark. tarafından şöyle açıklanmıştır [4].

“Video oyununda bir oyun karakteri koordinatlarının 400ms aralıkla güncellendiğini varsayalım. Verinin kaybolması ve gerekli zaman aralığında (400ms) gönderilmemesi durumunda bu verinin tekrar gönderilmesi herhangi bir anlam ifade etmemektedir. Çünkü bir sonraki pozisyon bilgisi artık kuyruğa alınmış bulunur.”

Kısmi güvenilirlik aslında veri için yaşam ömrü tanımlamaktadır. Kısmi güvenilirlik üç farklı şekilde tanımlanır:

- Zaman tabanlı güvenilirlik: tanımlanan zaman aralığında istenilen kadar yeniden gönderim yapılabilir.
- Tampon tabanlı güvenilirlik: belirlenen kuyruk boyutuna ulaşıldığında en eski paketler atılır bunun yerine yeni paketler alınır
- Yeniden gönderim sayısı: yeniden gönderim için sayı belirlenir ve bu sayı kadar yeniden gönderim işlemi yapılır.

3.1.2. SCTP protokolünün paket yapısı

SCTP paketleri ortak başlık ve veri parçalarından (chunk) oluşmaktadır. Her bir veri parçası ayrıca kendi başlık formatını içermektedir. SCTP ortak başlık yapısı Şekil 3.2’de verilmiştir.

Bit	0 - 7	8 - 15	16 - 23	24 - 31
+0	Kaynak Port		Hedef Port	
32	Doğrulama Etiketi			
64	Sağlama (checksum) Etiketi			
96	1.veri parçasının türü	1.veri parçası etiketleri	1.veri parçasının uzunluğu	
128	1.veri parçasının taşıdığı veri			
...	...			
...	N.veri parçasının türü	N.veri parçası etiketleri	N.veri parçasının uzunluğu	
...	N.veri parçasının taşıdığı veri			

Şekil 3.2. SCTP ortak başlık yapısı

Ortak başlık alanları ve açıklamaları aşağıda verilmiştir.

- Kaynak port – göndericinin port numarası
- Hedef port – alıcının port numarası
- Doğrulama etiketi – 32 bitlik rastgele sayı (önceki bağlantıdan ayırabilmek için)

- Sağlama (checksum) etiketi – CRC32c algoritması ile doğrulama için üretilmiş sayı [30].

SCTP veri parçaları

IETF RFC-4960 belgesine göre SCTP protokolü için 255 farklı veri parçası türü belirlenebilir. 0-14 arası veri parçası türleri protokol tarafından kullanılmaktadır. Diğer numaralar ise boşta bırakılmıştır. Veri parçası numaraları ve türleri Çizelge 3.1’de görülmektedir.

Çizelge 3.1 SCTP protokolünde veri parçası türleri

ID	Veri parçası türü
0	Payload Data (DATA)
1	Initiation (INIT)
2	Initiation Acknowledgement (INIT ACK)
3	Selective Acknowledgement (SACK)
4	Heartbeat Request (HEARTBEAT)
5	Heartbeat Acknowledgement (HEARTBEAT ACK)
6	Abort (ABORT)
7	Shutdown (SHUTDOWN)
8	Shutdown Acknowledgement (SHUTDOWN ACK)
9	Operation Error (ERROR)
10	State Cookie (COOKIE ECHO)
11	Cookie Acknowledgement (COOKIE ACK)
12	Reserved for Explicit Congestion Notification Echo (ECNE)
13	Reserved for Congestion Window Reduced (CWR)
14	Shutdown Complete (SHUTDOWN COMPLETE)

INIT, INIT ACK ve SHUTDOWN COMPLETE veri parçaları dışında diğer veri parçaları (farklı olsalar dahi) aynı paket içerisinde birleştirilebilir. SCTP paket boyutunu aşan veri parçaları parçalanarak diğer paketin içerisine aktarılabilir.

Parçalama işleminin yapıldığı veri parçalarında sıra numarası ve parçalanmış (fragmented) olduğunu gösteren etiket (flag) belirlenir.

Veri parçası başlığı ve payload data veri parçası

Veri parçası başlığı tüm veri parçaları için ortaktır ve Şekil 3.3'de verilmiştir.

Bit	0 - 7	8 - 15	16 - 23	24 - 31
+0	Veri parçasının türü	Veri parçası etiketleri	Veri parçasının uzunluğu	
...	Veri parçasının taşıdığı veri			

Şekil 3.3. SCTP veri parçası başlığı

Veri parçası başlığında kullanılan alanlar ve açıklamaları aşağıda gösterilmiştir;

- Veri parçasının türü (8 bit) - gönderilen veri parçasının türünü belirler. ID değerleri kullanılır (Bkz. Çizelge 3.1).
- Veri parçasının etiketleri (8 bit) - gönderilen veri parçası için belirlenen etiketleri içerir. Bu alan veri parçasının türüne bağlı olarak değişmektedir.
- Veri parçasının uzunluğu (16 bit) - veri parçasının türü, veri parçasının etiketi, veri parçasının uzunluğu ve veri parçasının değeri ile beraber tüm alanların bayt olarak uzunluğunu içerir. Veri parçasının değerinin boş olması durumunda bu değer dörde eşittir.
- Veri parçasının değeri – veri parçasının taşıdığı ana bilgiyi içermektedir. Bu alan veri parçasının türüne bağlı olarak farklı şekilde tanımlanabilir [27].

3.1.3. Diğer ulaşım katmanı protokolleri ile karşılaştırılması

SCTP Protokolünün en yaygın kullanılan ulaşım katmanı protokollerinden hem TCP hem de UDP protokolleri ile birçok ortak özellikleri bulunmaktadır. SCTP Protokolü UDP protokolü gibi mesaj tabanlı, TCP protokolü gibi bağlantı yönelimlidir. TCP protokolünde olduğu gibi güvenilir veri transferi, akış kontrolü ve tıkanıklık kontrolü sağlamaktadır. Hem sıralı hem de sırasız veri dağıtımını yapabilmektedir. Bunun

dışında SCTP protokolü sadece kendine has bazı özellikler de barındırmaktadır. CRC (Cyclic Redundancy Check - Döngüsel Artıklık Denetimi) tabanlı doğrulama, SYN flood saldırılarını önleme, multihoming, multistreaming bunlara örnek olarak gösterilebilir [4].

3.2. Video İletişiminde SCTP Protokolünün Kullanımı

SCTP Protokolünün gerçek zamanlı veya gerçek zamanlı olmayan uygulamalarda kullanımı ile ilgili birçok bilimsel çalışma bulunmaktadır. Bu çalışmalardan bazılarında aşağıda değinilmiştir.

K. H. Kim ve ark. tarafından 2009 yılında yapılan çalışmada gerçek zamanlı akış uygulamaları için ek katman ile SCTP protokolü uygulanmıştır [13]. Bu katman gönderim kontrolü işlemini gerçekleştirmektedir. Bu çalışmadaki ana fikir gönderim yapmadan gönderilecek olan paketin alıcı tarafında oynatılıp oynatılmayacağını belirlemektir. Bunun sayesinde gereksiz yeniden gönderimlerin önüne geçilerek hizmet kalitesinin artırılması amaçlanmıştır. Gönderim kontrol (transmission control) alt katmanı paketleri önemine göre kontrol etmekte ve onlar için belirli yeniden gönderim katsayısı belirlemektedir. Örneğin; yapılan çalışmada MPEG-4 formatı üzerinde çalışıldığından ses verisi ve I çerçeveler 4 defa, P çerçeveler 2 defa, B çerçeveler 1 defa, kontrol paketleri ise alıcıya ulaşmaya kadar yeniden gönderilmektedir. Yeniden gönderim için beklenen süre TC-SCTP protokolünde olduğu gibi RTT (Round-Trip Time - Gidiş Geliş Gecikmesi) dikkate alınarak tanımlanmıştır. Çalışma ns-2 benzetim aracı yardımı ile test edilmiş ve gönderim oranı, paket hata oranı, gecikme gibi parametrelerde SCTP ve PR-SCTP protokollerinden daha iyi sonuçlar elde edilmiştir.

A. Balk ve ark. tarafından yayınlanan makalede ise ns-2 benzetim aracı için kendi geliştirmiş oldukları uygulamadan bahsetmektedirler. Bu uygulama yardımı ile MPEG-4 formatındaki verinin iletilmesi test edilmiş ve alıcı tarafa daha fazla verinin ulaştığı gösterilmiştir [11].

3.3. IPTV’de SCTP Protokolü

S. T. Kim ve ark. tarafından yapılan “Performance of SCTP for IPTV Applications” başlıklı çalışmada IPTV uygulamalarında gerçek zamanlı veri aktarımında kullanılan UDP ve TCP gibi taşıma katmanı protokolleri yerine SCTP protokolünün kullanılması tercih edilmiştir. Makalede öncelikle IPTV uygulamalarından bahsedilmiş, daha sonra farklı gerçek zamanlı ve multistreaming uygulamalarda SCTP kullanımı ile ilgili kaynaklara değinilmiştir. Özellikle SCTP kullanımının IPTV’de uygulama katmanında farklı SID (stream identifier) ile farklı veri akışlarının mantıksal olarak ayrıştırılabileceği özelliği göz önünde tutulmuştur [2].

Makalede PR-SCTP (Partial Reliable) SCTP protokolünden bahsedilmiştir. PR-SCTP IETF tarafından RFC 3758 belgesinde yayınlanmaktadır. SCTP protokolünün PR-SCTP uzantısı uygulamaya zaman bazında güvenilirlik kazandırmaktadır. Bu yaklaşım literatürde lifetime yaklaşımı olarak geçmektedir. Lifetime yaklaşımı özellikle gerçek zamanlı uygulamalarda kullanılmakta ve paketlerin gönderimi için istemci tarafından önceden zaman belirlenmektedir. Belirlenen zaman aralığında gönderilemeyen paketlerin tekrar gönderimi göz ardı edilmektedir. SCTP istemcisi PR-SCTP uzantısını kullanabilmek için INIT mesajında sunucuya etiket bilgisi göndermektedir. FORWARD-TSN mesajında gerçek zamanlı uygulamanın onay bilgisi istemci tarafa dönmektedir. Makalede ayrıca PR-SCTP’de paket bazında hesaplama yapabilmek için Eş. 3.1 gösterilmiştir.

$$Lifetime_N = R_{APP} + \frac{L_N}{B_{MEDIA} \times 1024} (ms) \quad (3.1)$$

Bu eşitlikte;

$Lifetime_N$: Uygulama mesajının ömrü (ms),

R_{APP} : Rate control’a dayalı gecikme (ms),

L_N : Uygulama mesajının boyutu (bayt),

B_{MEDIA} : Ortamın ortalama bit rate’i (b/s)

Bu eşitlikte R_{APP} değişkeni uygulama bufferinin durumuna, network gecikmesine ve jittere bağlı olarak değişmektedir.

Makalede SCTP'nin multi-streamingi destekleyen yapısı kullanılarak sunucu ve istemci arasında tek bağlantı (association) üzerinden birden fazla kanalın yayın yapılabileceğinden bahsedilmiştir. Bu özellik hem HOL Blocking problemini ortadan kaldırmakta, hem de IPTV'de önemli problemlerden biri olan kanallar arası geçiş süresini azaltmaktadır.

Yazarlar yaptıkları çalışmada Linux işletim sistemi için hazırlanmış lksctp-tools ve sctplib araçlarını, ayrıca uygulama katmanı ile iletişim için socketapi kütüphanesini kullanmışlar.

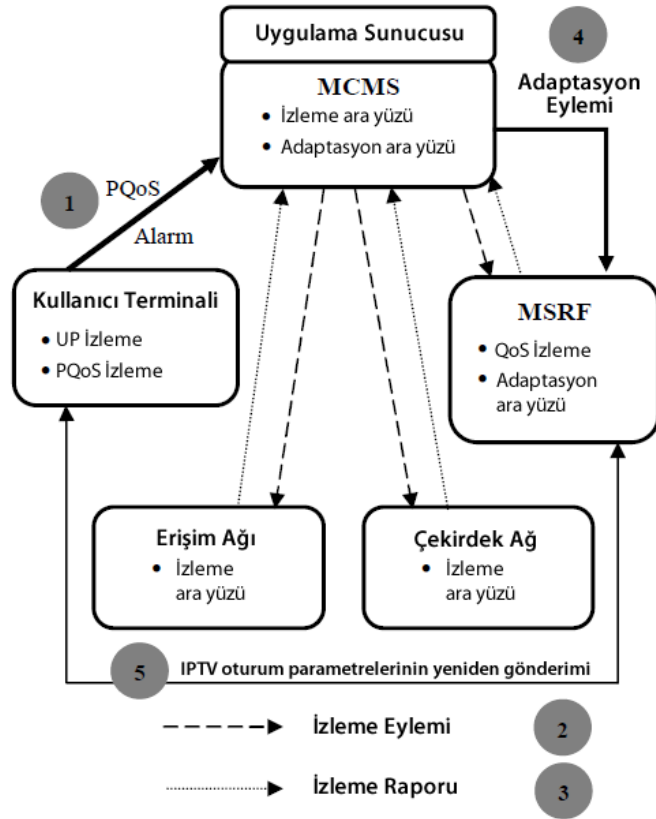
Yapılan araştırmada 5 farklı test sonucu elde edilmiştir [2].

1. Farklı kayıp oranlarında (loss rate) TCP, UDP, SCTP ve PR-SCTP taşıma tamamlama zamanı karşılaştırılmış UDP protokolü için en iyi performans elde edilmiş, PR-SCTP için ise UDP'ye yakın sonuçlar elde edilmiştir. Yazarlar ayrıca UDP protokolünde kayıp oranının olmamasını da vurgulamışlar.
2. Farklı kayıp oranlarında PR-SCTP ve UDP gönderme-alma oranları karşılaştırılmış ve daha iyi QoS için PR-SCTP en iyi sonucu vermiştir.
3. Farklı akış sayıları ve farklı kayıp oranlarında PR-SCTP taşıma tamamlama zamanı karşılaştırılmış ve 3 akış kadar PR-SCTP'de iyi performans elde edilmiştir.
4. Farklı kayıp oranlarında tek bağlantıda 3 akış TCP ve SCTP için taşıma tamamlama zamanında karşılaştırılmış ve SCTP'de daha düşük taşıma zamanı bulunmuştur.
5. TCP ve SCTP arasında kanal değişim gecikmesi karşılaştırılmış ve SCTP için daha iyi sonuç elde edilmiştir.

S. Boussen ve ark. tarafından hazırlanmış diğer makalede ise IPTV uygulamalarında SCTP protokol kullanımı ile beraber QoS artırmak için farklı bir mimari teklif

edilmiştir [3]. Yazarlar öncelikle PR-SCTP protokolünden bahsetmiş ve bu protokolün paket yaşam ömrü (lifetime) yaklaşımının özellikle video akışı ve VoIP gibi gerçek zamanlı uygulamalarda çok kullanışlı olduğunu vurgulamışlar. SIP üzerinden SCTP kullanımı, çoklu ortam akışı, özellikle MPEG-4 çoklu ortam taşınmasında kısmi güvenilirlik (partial reliability) uygulamaları literatür taraması olarak makalede yer almaktadır. Özellikle, MPEG-4 gibi I, B ve P çerçevelerden (frame) oluşan akışlarda B ve P çerçevelerin I çerçevesine bağlı olduğu göz önünde bulundurulursa I çerçevelerin daha güvenilir olarak gönderilmesi önem arz etmektedir.

Makalede IPTV uygulamalarında PQoS (Perceived QoS – Algılanan QoS) için IMS (Internet Multimedia Subsystem) tabanlı mimari geliştirilmiştir. Bu mimari genel hatları ile Şekil 3.4’de verilmiştir.



Şekil 3.4 IMS tabanlı IPTV ve PQoS mimarisi [3]

Yazarlar SCTP'nin multi-homed, DAR ve multi streaming özelliklerini kullanarak istemci taraflı PQoS yönetimi yapmayı amaçlamışlar. Bilindiği gibi SCTP protokolü bir bağlantı (association) üzerinde multi-homed servis sağlayabilmektedir. Multihomed hizmeti aynı anda istemci tarafında birden fazla IP adresinin tanımlanması ile gerçekleştirilmektedir. Bu IP adreslerinden sadece bir tanesi (primary address) veri akışı kabul etmekte, diğer adresler ise QoS için kullanılmaktadır. Makalede IMS istemcisi tarafında ikincil yolları kontrol etmek için özel bir birim tasarlanmıştır. HEARTBEAT veri parçaları kullanılarak PQoS durumu kontrol edilmektedir. İstemci tarafında herhangi sıkıntının oluşması durumunda alarm mesajı oluşturulmaktadır.

Multistreaming özelliği SCTP'de tek association üzerinde birden fazla akış yapabileceği sağlamaktadır. Akışlar bağlantı içinde SID ile bir birinden ayrılmaktadır. Bu özellik IPTV uygulamasında özellikle kanal değişim gecikmesi için uygun görülmüştür.

SCTP DAR uzantısı istemci ve sunucu arasında bağlantıyı koparmadan adres değişikliği sağlamaktadır. Bu uzantı IETF tarafından RFC 5061 belgesinde yayınlanmaktadır [28].

DAR uzantısı özellikle mobil IPTV uygulamasında çok kullanışlı bir protokol olarak ele alınmıştır. DAR uzantısı aşağıdaki gibi çalışmaktadır. Öncelikle yeni IP adresi ASCONF (Address Configuration – Adres Konfigürasyonu) mesajı ile adres değişikliği isteği gönderilmektedir. Sunucu tarafından ASCONF-ACK (Address Configuration Acknowledgment - Adres Konfigürasyon Onayı) mesajı geri döndürülmektedir. Handover işlemi tamamlandıktan sonra yeni eklenen adres ana adres olarak tanımlanmakta, daha sonra ise eski adres IP adres listesinden silinmektedir.

Mobil IPTV uygulamasında yazarlar SCTP DAR uzantısı ile beraber hem de SIP protokolünü kullanmayı teklif etmişler. Özellikle handover sırasında bi-casting (aynı anda hem eski hem de yeni IP adresine veri transferi) yayın uygun görülmüştür [3].

4. KUYRUK ÖNCELİKLENDİRMELİ ULAŞTIRMA KATMANI PROTOKOLÜ

Bu bölümde tez kapsamında geliştirilmiş olan PQ-SCTP protokolünün çalışma şekli, önceliklendirme mantığı ve işlem gecikmesi anlatılacaktır.

4.1. PQ-SCTP Protokolünün Mimarisi

Ağ üzerinde gönderilen paketlerin hepsi aynı önem derecesine sahip değildir. Örneğin; MPEG formatında kodlanmış video dosyasını ele alacak olursak I çerçeveler daha fazla bilgi taşımakta, B çerçeveler ise sadece değişim bilgisini içermektedir. Başka bir deyişle I çerçeveler alıcı taraf için daha fazla önem taşımaktadır [11]. Aynı şekilde gerçek zamanlı iletişimin veya veri akışının bulunduğu uygulamalarda yeniden gönderilen paketlerin işleme sırası diğer paketlere göre daha önceliklidir. Çünkü belli bir zaman sonra bu paketler önemini kaybedecektir. SCTP protokolünde alıcı tamponu her akış için sadece SSN (Stream Sequence Number – Akış Sıra Numarası) numarasına göre sıralama yapmaktadır. Bu sıralamada önemli ve az önemli paketlerin kuyruktaki dağılımı yaklaşık olarak eşit olmaktadır. Sonuç olarak belli bir zaman aralığında daha önemsiz bir veri parçası üst katmanda işleme alınırken SSN numarası büyük olan veri parçası kuyrukta beklemeye devam edecektir. Bu da önemli ve daha az önemli veri parçalarının kuyrukta ortalama bekleme süresi, IPDV, jitter gibi ölçümlerin de bir birine yakın değerlere sahip olacağı anlamına gelir. PQ-SCTP protokolü bu sorunu ortadan kaldırmak için geliştirilmiştir.

PQ-SCTP, SCTP protokolü temel alınarak geliştirilmiştir. Protokolde hem alıcı hem de gönderici tarafında değişiklikler yapılmıştır. PQ-SCTP protokolü gönderilen veride önceliklendirme yaparak hizmet kalitesini arttırmayı amaçlamıştır. PQ-SCTP protokolü sadece Payload Data türündeki veri parçaları üzerinde işlem yapmaktadır. Diğer veri parçası türlerinde herhangi bir değişiklik yapılmamıştır.

4.2. PQ-SCTP Protokolünde Paketlere İçerik Tabanlı Öncelik Verilmesi

PQ-SCTP protokolünde önceliklendirme işlemi gönderici tarafında yapılmaktadır. Önceliklendirme Payload Data türündeki veri parçaları oluşturulurken başlık kısmında bulunan veri parçası etiketleri kullanılarak tanımlanır. Bölüm 3.1.2.'de anlatıldığı gibi veri parçası başlığında sekiz bit uzunluğunda veri parçası etiketi alanı bulunmaktadır. Payload Data türündeki veri parçaları için SCTP protokolünde bu alanlardan son dört biti kullanılmaktadır. Diğer dört bit ise ayrılmıştır (Şekil 4.1).

Bit	0 - 7	8 - 15	16 - 23	24 - 31
+0	Veri parçası türü = 0	Ayrılmış	I U B E	Veri parçasının uzunluğu
32	TSN			
64	Stream Identifier		Stream Sequence Number (SSN)	
96	Payload Protocol Identifier			
128	Veri parçasının taşıdığı veri			
...				

Şekil 4.1. SCTP Payload Data türündeki veri parçasının yapısı

Kullanılan bit etiketleri ve açıklamaları aşağıda verilmiştir:

- I – SACK veri parçasının gecikme olmadan hemen gönderileceğini gösterir.
- U – Unordered (Sırasız) Bu etiket gönderilen veride herhangi bir sırlamanın olmadığı anlamına gelmektedir. Bu durumda alıcı taraf SSN numarasını kuyruk sıralamasında dikkate almayacaktır.
- B – Beginning (Başlangıç) Bu etiket Payload Data türündeki veri parçalarının parçalanması (fragmenting) durumunda kullanılır ve parçalanmış verinin ilk veri parçasını gösterir.
- E – Ending (Son) “Başlangıç” etiketine benzer şekilde parçalanmış veriler için kullanılır ve son veri parçasını gösterir.

Diğer dört bitlik etiket alanının ilk iki biti ise sıralama için kullanılmıştır. Sıralama amacıyla PQ-SCTP için dört farklı öncelik sırası belirlenmiştir. Bit değerleri ve öncelik sırası Çizelge 4.1’de sunulmuştur.

Çizelge 4.1. PQ-SCTP için bit değerleri ve öncelik sırası

Bit değeri	Öncelik sırası
00	1. öncelikli veri parçası
01	2. öncelikli veri parçası
10	3. öncelikli veri parçası
11	Tekrar gönderilen veri parçası (rtx)

Sıralama mantığı dikkate alınarak PQ-SCTP üzerinde yapılan tüm deneylerde verilerin MPEG-2 dosya formatındaki iletildiği kabul edilmiştir. MPEG formatında I, P ve B paketleri sırasına göre birinci, ikinci ve üçüncü öncelikli paket olarak tanımlanmıştır.

4.3. PQ-SCTP Protokolünde Rtx Paketlere Öncelik Verilmesi

Yeniden gönderilen paketlerin işlenmesi özellikle gerçek zamanlı uygulamalarda zaman açısından ciddi önem kazanmaktadır. Gerçek zamanlı iletimde işlem gecikmesi süresi kadar işlenmeyen rtx paketleri veri olarak önemini kaybetmektedir. Veri akışında bir sonraki çerçeveye geçildiğini kabul edersek bir önceki çerçeve bilgisini taşıyan rtx paketinin yeniden gönderilmesi anlamsız hale gelmektedir [13]. Aynı şekilde alıcı tarafına ulaşan ancak işlem süresinden daha fazla alıcı kuyruğunda bekleyen rtx paketi de veri akışı açısından anlamsızdır. Bu paket işlem sırası geldiğinde atılacağına göre kuyrukta bekletilmesi kuyruk alıcı penceresi (kuyruğu) boyutunun küçüleceği nedeniyle dezavantaj olarak değerlendirilebilir.

Rtx paketlerin bu açıdan en önemli paket olduğunu kabul edersek kuyruk sıralamasında da bu paketlere en yüksek öncelik tanımlanmalıdır. PQ-SCTP protokolünde MPEG-2 çerçevelerine paket önceliği verildiği gibi yeniden gönderilen

pakete de bu şekilde öncelik verilmektedir. Yeniden gönderilen pakette çerçeve türü dikkate alınmadan tanımlanmaktadır. Yapılan çalışmada yeniden gönderilen paketler öncelik sırasına göre en başta yer alır.

4.4. PQ-SCTP Protokolünde Alıcı Kuyruğunun Sıralanması

PQ-SCTP gönderici tarafında paketlere tanımladığı önem sırasına göre alıcı tarafının tamponunda (kuyruğunda) bu önceliğe göre sıralama yapmaktadır. Sıralama algoritması için hazırlanmış sözde kod aşağıda verilmiştir.

```
void function onceliklendir (gelenvp, kuyruk)
set gnum to gelenvp-nin oncelik degeri;
set vp to null;
if gnum = 11 //tekrar gönderim
    kuyruk.head = gelenvp;
else
    for (i = kuyruk.head; i <= kuyruk.length; i = kuyruk.head.sonraki) {
        set vp to i;
        set nnum to vp oncelik degeri;
        if gnum = 00 //birinci oncelikli
            if nnum != 00 and nnum != 11
                break;
        else if gnum = 01 //ikinci oncelik
            if nnum != 00 and nnum != 01 and nnum != 11
                break;
        else if gnum = 10 //ucuncu oncelik
            if nnum != 00 and nnum != 01 and nnum != 10 and nnum != 11
                break;
    }
    if (vp == null) || (gnum != nnum)
        vp.onceki = gelenvp;
```

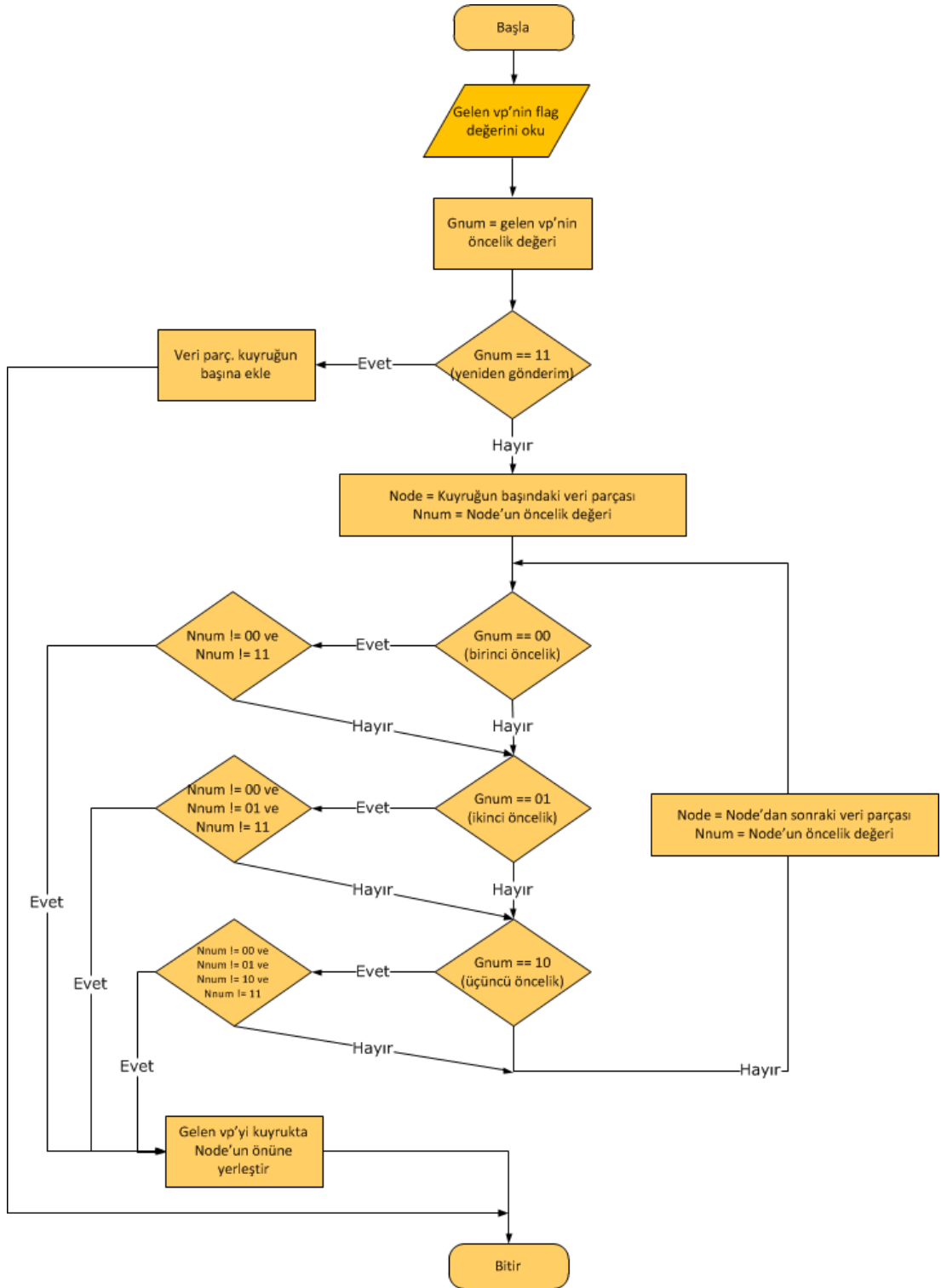
Algoritma bu şekilde çalışmaktadır: Alıcı tarafında kullanılan kuyrukta bağlı liste (linked list) yapısı kullanılır. Öncelikle, gelen veri parçasının önceliklendirme etiket değerleri okunur. Eğer gelen değer 11 ise (tekrar gönderilen veri parçası) gelen veri parçası kuyruğun başına eklenir ve bağlı listedeki bağlantılar düzenlenir. Gelen veri

parçasının öncelik etiketinin 11'den farklı olması durumunda kuyruk üzerinde başlangıç düğümünden son düğüme doğru hareket edilir. Döngünün her adımında öncelik değerleri kontrol edilir. Gelen veri parçası birinci öncelikli ve kuyrukta döngünün bulunduğu veri parçası ikinci veya üçüncü öncelikli ise döngü durdurulur. Gelen veri parçası mevcut döngüdeki veri parçasından bir önceki düğüme yerleştirilir. Gelen veri parçasının sonraki döngülerde de kuyruğa eklenmesini önlemek için döngüden “break” komutu ile çıkılması gerekmektedir. Benzer şekilde ikinci önceliğe sahip veri parçalarında da kuyrukta üçüncü öncelikli düğüm bulununcaya kadar ilerlenir. Üçüncü önceli veri parçalarında ise kuyruğun sonuncu düğümü mevcut düğüm olarak işaretlenir. Sonuç olarak alıcı kuyruğunda Payload Data türündeki veri parçalarının sıralaması tekrar gönderilen veri parçaları (rtx), birinci, ikinci ve üçüncü öncelikli veri parçaları şeklinde olacaktır. Belirli sabit sıralamada gönderilen Payload Data veri parçaları SCTP ve PQ-SCTP protokolleri için alıcı kuyruğunda Şekil 4.2’de verilen dizilime sahip olacaktır. Algoritmanın akış diyagramı Şekil 4.3’de sunulmuştur.



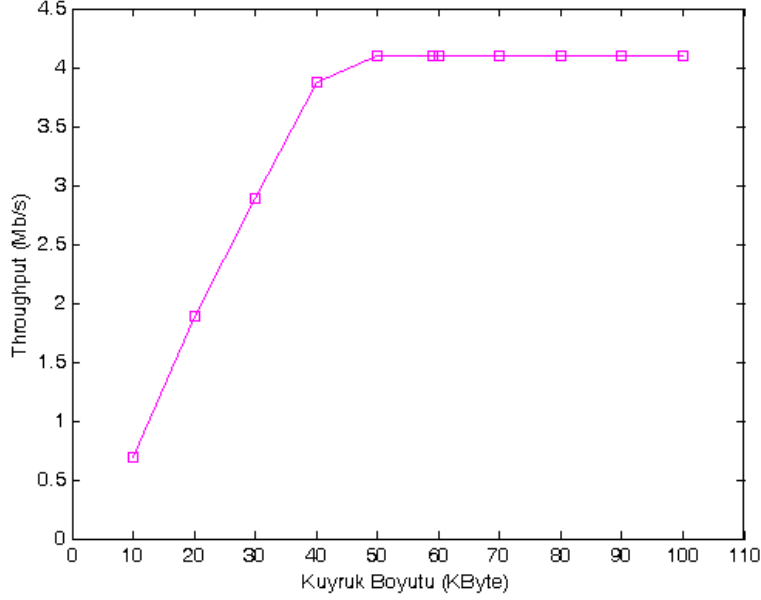
Şekil 4.2. SCTP ve PQ-SCTP alıcı kuyruğu karşılaştırması

SCTP protokolü gönderilen SACK veri parçalarında Advertised Receiver Window Credit (a_rwnd) alanı kullanılmaktadır. Bu alanda alıcı kuyruk boyutu bayt olarak gönderici tarafına gönderilmekte, gönderici taraf ise alıcının kuyruk boyutuna bağlı olarak veri gönderim oranını düzenlemektedir. A_rwnd alıcı kuyruğunda taşma ve atılma durumlarının oluşmasını engellemektedir. Buna rağmen bazı uç durumlarda atılmaların ortaya çıkması kaçınılmazdır. Örneğin; alıcı tarafında çok yüksek kuyruk işlem gecikmesi (~600ms), yüksek gönderim oranına rağmen çok düşük kuyruk boyutu ve düşük bağlantı gecikmesi durumunda kuyrukta atılmalar oluşmaktadır.



Şekil 4.3. PQ-SCTP kuyruğa ekleme algoritması için akış diyagramı

Sabit işlem gecikmesi (2ms) ve gönderim oranı (4Mb/s) ile kuyruk boyutunun değişmesine bağlı olarak throughput değerinin değişimi Şekil 4.4’de görülmektedir.



Şekil 4.4. PQ-SCTP kuyruk boyutuna bağlı olarak throughput değişimi

Bu deney bölüm 5.2’de detaylı anlatılmış olan dumbbell topolojisi üzerinde gerçekleştirilmiştir. Topolojideki düğümler arasındaki bağlantıların bant genişliği 20 Mb/s olarak alınmıştır. Hat gecikmesi ise 10ms’dir. Arka plan trafiği oluşturan düğümlerin 6 adet olduğu ve her birinin 0,5 Mb/s gönderim oranı ile veri ilettiği dikkate alınır ise bağlantının en tıkanık noktasında 7 Mb/s trafik oluşacaktır. Bu durumda ağ üzerinde tıkanıklığın oluşması söz konusu değildir. Şekilden görüldüğü gibi 10-100KB kuyruk boyutları ile deney tekrarlanmıştır. Throughput değerinin gönderim oranına eşit olduğu aralık 40-50KB kuyruk boyutlarına denk gelmektedir. 50KB kuyruk boyutu üzerindeki değerlerde ise throughput değeri değişmemektedir. [5]’de MPEG-2 kodlaması için 4Mb/s gönderim oranında kuyruk boyutu 59392 B (58KB) olarak tanımlanmıştır. Bu deney hem SCTP, hem de PQ-SCTP protokolleri için çalıştırılmış ve her iki protokolde de aynı sonuçlar elde edilmiştir. Çizelge 4.2’de elde edilen sonuçlar verilmiştir.

Çizelge 4.2. SCTP ve PQ-SCTP için kuyruk boyutuna bağlı throughput değişimi

Kuyruk Boyutu	SCTP		PQ-SCTP	
	Throughput (MB/s)	Alınan Veri (B)	Throughput (MB/s)	Alınan Veri (B)
10KB	686,60	10296624	686,60	10296624
20KB	1887,47	28311600	1887,47	28311600
30KB	2893,75	43393968	2893,75	43393968
40KB	3871,99	58078512	3871,99	58078512
50KB	4102,71	61540656	4102,71	61540656
58KB	4102,76	61531248	4102,76	61531248
60KB	4102,77	61540656	4102,77	61540656
70KB	4102,80	61540656	4102,80	61540656
80KB	4102,00	61529904	4102,00	61529904
90KB	4102,76	61537968	4102,76	61537968
100KB	4102,09	61531248	4102,09	61531248

Kuyruğun dolması zamanı ortaya çıkan atılmalar için SCTP protokolünde herhangi bir işlem yapılmamaktadır. Atılmalar sadece SSN numarası temel alınarak yapılır. Eğer gelen veri parçasının SSN numarası kuyruktaki en büyük SSN numarasından daha büyük ise gelen veri parçası atılır. Gelen veri parçasının SSN numarasının kuyruktaki en büyük SSN numarasından daha küçük olması durumunda ise büyük SSN'e sahip veri parçası atılır, gelen veri parçası ise kuyruğa eklenir [27].

PQ-SCTP kuyruktaki veri parçalarını önem sırasına göre sıraladığı gibi kuyruktan atılma işlemini de bu sıralamaya uygun şekilde yapmaktadır.

```

set anum to atilanvp oncelik deęeri;
if anum = 00 //birinci oncelik
    set sonvp to kuyruk.tail;
    set snum to sonvp-nin oncelik deęeri;
    if sum != 00 and snum != 11
        delete sonvp
        kuyruk.tail = atilanvp
else if anum = 01 //ikinci oncelik
    set sonvp to kuyruk.tail;

```

```

set snum to sonvp-nin oncelik deęeri;
if snum = 10
    delete sonvp
    kuyruk.tail = atilanvp
else
    delete atilanvp

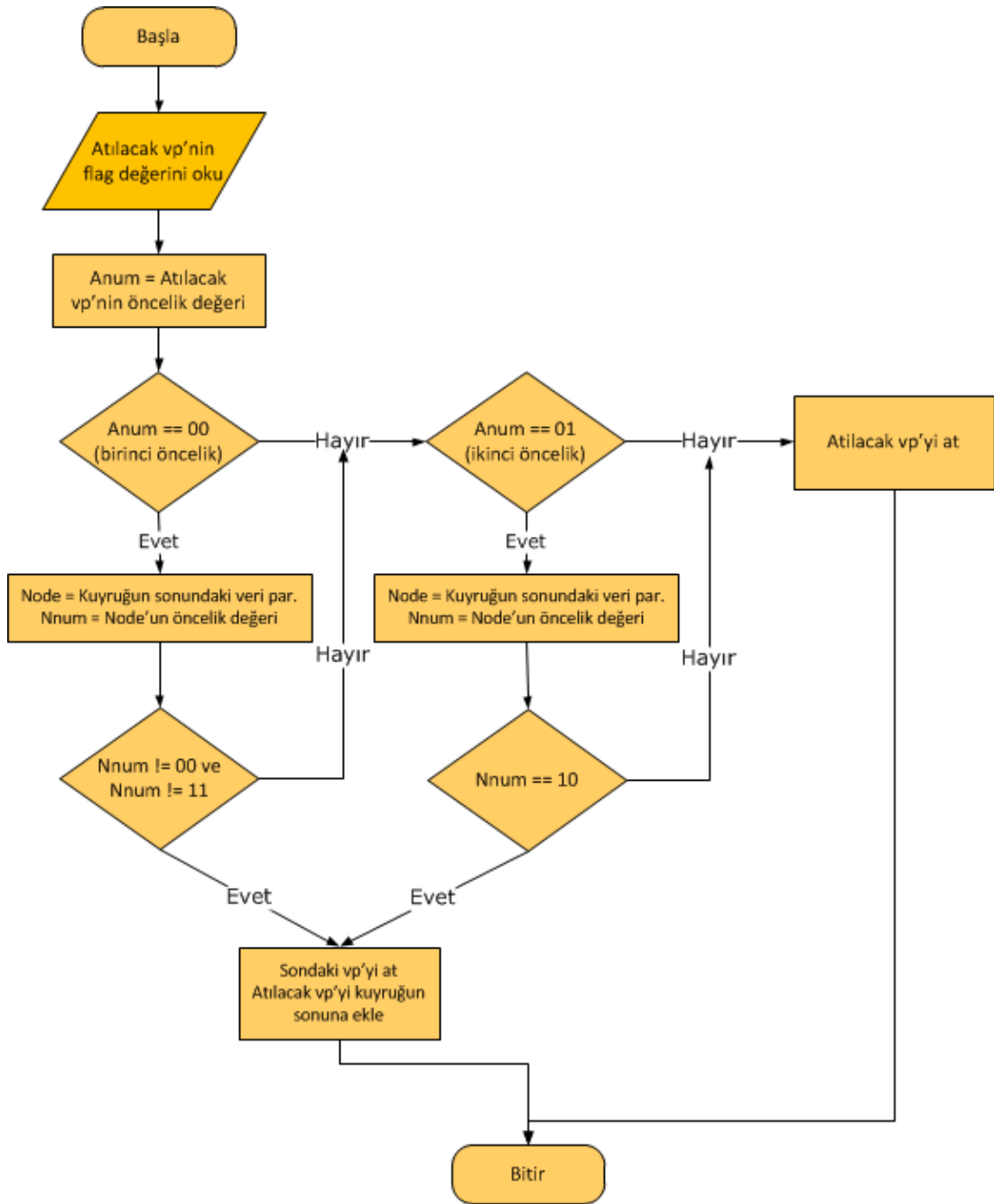
```

Koddan da görüldüğü gibi öncelikle kuyruğun dolmasından dolayı atılması planlanan veri parçasının öncelik değeri alınır. Öncelik birinci derece ise kuyruktaki son düğüm kontrol edilir. Son düğümün ikinci veya üçüncü önceliğe sahip olması durumunda son düğüm silinir, atılması planlanan birinci öncelikli veri parçası kuyruğa eklenir. Veri parçasının kuyruğa eklenmesi sırasında PQ-SCTP'nin kuyruk sıralaması mantığı bölüm 4.4'de anlatıldığı şekliyle uygulanır. Atılması planlanan veri parçası ikinci önceliğe sahip ise kuyruktaki son düğümün üçüncü öncelikle olması durumu kontrol edilir. Son düğüm üçüncü önceliğe sahip ise bu düğüm kuyruktan silinir, atılması planlanan veri parçası kuyruğa eklenir. Atılması planlanan veri parçasının önceliği birinci ve ikinci dereceden farklı ise veri parçası atılır. Atılma algoritmasının akış şeması Şekil 4.5'de verilmiştir.

4.5. PQ-SCTP Protokolünde Üst Katman İçin İşlem Gecikmesinin Tanımlanması

Alıcı tarafın kuyruğuna alınan paketler üst katman tarafından işleme tabi tutulmaktadır. İşlem makinaya bağlı olarak farklı gecikme sürelerinde yapılır. Bu süre işlem gecikmesi (processing delay) olarak tanımlanmaktadır. Örneğin; video formatındaki veri için paketlerin kuyrukta bekletilmesi, açılması, decode edilmesi, işlenmesi vb. süreler işlem gecikmesini ortaya çıkaran nedenlerdir. İşlem gecikmesi bu işlemin yapıldığı makinaya bağlı olarak farklılıklar gösterebilmektedir.

Yapılan çalışmada kuyruktan veri parçalarının silinmesi tanımlanan bekleme süresine bağlı olarak gerçekleşmektedir. Kuyruğa eklenen her bir paket kuyruktan silinerek işleme alınmakta, işlem gecikmesi tamamlanınca bir sonraki paket kuyruktan silinmektedir.



Şekil 4.5. PQ-SCTP protokolü için kuyruktan atma algoritması için akış diyagramı

5. DENEYSEL SONUÇLAR

PQ-SCTP protokolü ile SCTP protokolü karşılaştırılarak farklı sonuçlar elde edilmiştir. Denemeler dumbbell ve parking lot olmak üzere iki farklı topoloji üzerinde yapılmıştır. Tezin bu bölümünde kullanılan benzetim ortamı, benzetim ortamı üzerinde yapılan değişiklikler, elden edilen sonuçlar ve onların değerlendirilmesi ele alınacaktır. Deneysel sonuçların incelenmesine geçmeden önce sonuçlar için hangi performans ölçütlerinin ve benzetim parametrelerinin dikkate alındığı aşağıda gösterilmiştir. Bu ölçüt ve parametrelerin incelenmesi, sonuçların daha iyi bir şekilde anlaşılması açısından önem arz etmektedir.

Benzetim parametreleri:

Benzetim süresi (s) – benzetim programının çalıştığı süreyi ifade etmektedir. Hem dumbbell, hem de parking lot topolojileri üzerinde yapılan denemelerde bu süre sabit olarak 120 saniye alınmıştır.

İşlem gecikmesi (s) – bir önceki bölümde anlatıldığı gibi, işlem gecikmesi veri parçasının kuyrukta işleme alınması için geçen süreyi (bekleme süresi) ifade etmektedir. Bekleme süresi sabit değer olmayıp, yazılım ve donanıma bağlı olarak değişmektedir. Ayrıca işlem süresi tanımlanırken cihazın ayrılmış (dedicated) olup olmadığı, aynı anda farklı işlemler yaptığı da göz önünde bulundurulmalıdır.

Kuyruk boyutu (B) – Alıcı tarafındaki kuyruğun boyutunu bayt cinsinden ifade etmektedir. Kuyruk boyutu da işlem süresi gibi cihaza bağlı olarak değişen parametredir. Deneyler farklı kuyruk boyutları üzerinde gerçekleştirilmiştir.

Veri gönderim oranı (Mb/s) – gönderici taraf için tanımlanmaktadır. Farklı kalitede video yayını için farklı gönderim oranları tanımlanabilir. Yapılan deneylerde MPEG formatı için [13]'de belirtilen gönderim oranları dikkate alınmıştır.

Performans kriterleri:

Throughput (Mb/s) – bağlantı üzerinde başarılı gönderilen verinin ortalama gönderim oranını ifade etmektedir.

Kuyrukta ortalama gecikme süresi (ms) – Payload Data türündeki veri parçalarının alıcı kuyruğunda ortalama bekleme süresini ifade eder. Bu ölçüt öncelik grupları bazında ele alınmıştır. Her bir öncelik grubu için veri parçasının kuyruğa eklenme ve kuyruktan silinme zamanları arasındaki farkın ortalama değeridir. Ortalama bekleme süresi hangi grubun daha sık işleme alındığını göstermek açısından önemli ölçüttür.

Kuyrukta toplam gecikme süresi (s) – Payload Data türündeki veri parçalarının benzetim süresince kuyrukta toplam gecikme süresini ifade eder. Bu ölçüt protokolün doğruluğunu değerlendirmek için kullanılmıştır. Eşit benzetim ortamı şartları altında SCTP ve PQ-SCTP protokolleri için kuyrukta toplam gecikme süresinin yakın çıkması gerekmektedir. Kuyruktaki toplam gecikme değerlerinin bulunması için aşağıdaki eşitlik kullanılmıştır.

$$\sum_{k=1}^n T_k * E_k \quad (5.1)$$

Bu eşitlikte n öncelik gruplarının sayısını, T_k k . öncelik grubu için ortalama kuyruk gecikmesini, E_k ise k . öncelik grubu için kuyruğa eklenen veri parçası sayısını ifade etmektedir.

IPDV (ms) – IETF RFC 3393 belgesinde paket gecikmeleri arasındaki değişim olarak tanımlanmaktadır [31]. Bu çalışmada da kuyruktaki gecikme sürelerinin değişimi olarak ele alınacaktır. IPDV ölçüt öncelik grupları bazında ele alınmıştır. Ortalama, maksimum ve minimum değerleri hesaplanmaktadır. Kuyruktaki IPDV için ortalama değer aşağıdaki eşitlik ile hesaplanır.

$$\frac{\sum_{k=2}^n |t_k - t_{k-1}|}{n-1} \quad (5.2)$$

Bu eşitlikte n öncelik grubu bazında kuyruktaki veri parçası sayısını, t_k ise k . veri parçası için kuyruktan silinme zamanını ifade etmektedir.

Jitter – bir ölçüm kriteri olarak değişim farkını ifade eder. Jitterin hesaplanması için ortalama değer belirlememiz gerekmektedir [31, 32]. Bu değer deneylerde kuyruktaki ortalama gecikme süresi olarak alınmıştır. Her bir veri parçası için ortalama bekleme süresi ile grup bazındaki ortalama bekleme süresi farklı jitter değerini ortaya çıkaracaktır. Jitter aşağıdaki eşitlik yardımı ile hesaplanmaktadır.

$$\mu = \frac{\sum_{k=1}^n d_k}{n} \quad (5.3)$$

$$J_m = d_m - \mu \quad (5.4)$$

Eş. 5.4’de n , farklı öncelik grubuna göre kuyruktaki veri parçası sayısını, d_k k . gecikmeyi, μ ortalama gecikmeyi, J_m ise m . gecikmenin (d_m) ortalama gecikmeden farkını ifade etmektedir.

Kuyruğa eklenen veri parçası sayısı (adet) – bu ölçüt de öncelik grupları bazında ele alınmış olup alıcı taraf ulaşan ve atılmayan veri parçası sayısını ifade etmektedir.

Kuyruktan atılan veri parçası sayısı (adet) – alıcı tarafa ulaşan ama kuyruğun dolu olmasından dolayı atılan veri parçası sayısını ifade eder. Bu ölçüt öncelik grupları bazında ele alınmıştır. Önceki bölümde anlatıldığı gibi normal şartlar altında protokolda atılmalar ortaya çıkmadığından bu değer genelde sıfıra eşit olmaktadır. Bazı özel şartlar altında kuyruktaki atılmalar ortaya çıkmaktadır.

Alıcıya ulaşan veri parçası sayısı (adet) – alıcıya ulaşan toplam veri parçası sayısını ifade etmektedir. Atılan ve kuyruğa eklenen veri parçalarının toplamı olarak hesaplanır. Bu değer de kuyruktaki toplam bekleme süresine benzer şekilde protokolün doğruluğunu denemek için ele alınmıştır. Eşit benzetim ortamı şartları altında SCTP ve PQ-SCTP protokolleri için yakın değerlerin elde edilmesi beklenmektedir.

Yazılım ve donanım altyapısı

Bu tez kapsamında yapılan tüm deneyler Linux tabanlı Ubuntu 12.04 işletim sistemi üzerinde ns-2 benzetim ortamı yardımı ile yapılmıştır. Kullanılan cihaz Intel Core i5-2400 3.1GHz işlemciye ve 8GB belleğe sahiptir.

5.1. Benzetim Ortamı

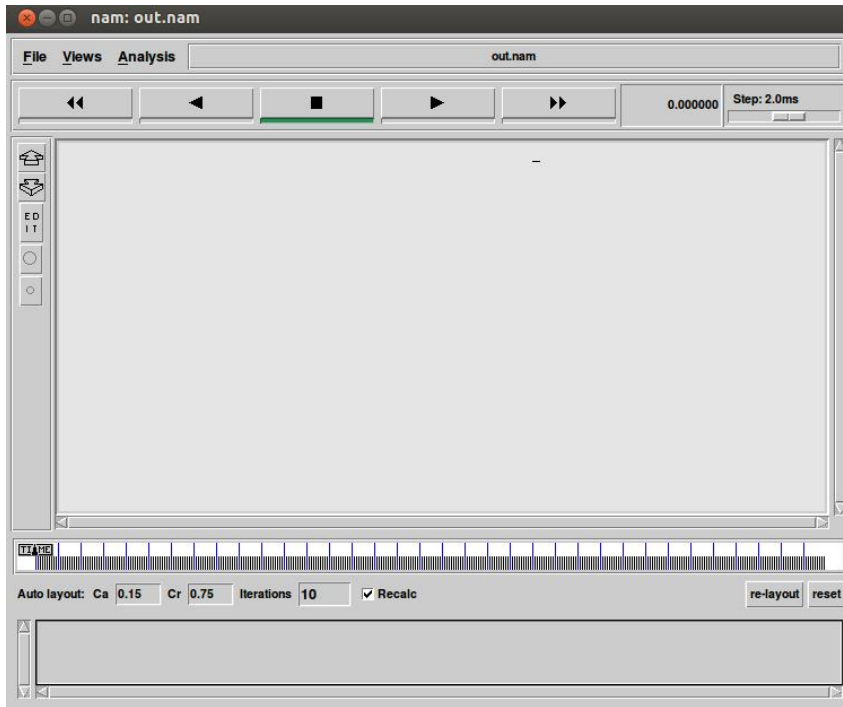
SCTP protokolü için birçok farklı uygulama ve benzetim ortamları bulunmaktadır. FreeBSD, Linux, Mac OS X ve Solaris 10 gibi farklı işletim sistemleri için SCTP çekirdek uygulamaları hazırlanmıştır. FreeBSD 7.0 sürümü ile SCTP desteğini tam olarak sağlamaktadır. Mac OS X işletim sistemi de FreeBSD çekirdek uygulaması tabanlı NKE (Network Kernel Extension – Ağ Çekirdek Eklentisi) uygulamasını desteklemektedir. Linux işletim sisteminin hemen hemen tüm sürümlerinde SCTP uygulaması mevcuttur. Solaris işletim sistemi ise 10 sürümü ile SCTP desteğini sağlamıştır.

Çekirdek uygulamaları ile beraber SCTP protokolü için farklı benzetim ortamları da oluşturulmuştur. Open SS7, Delaware üniversitesinde P.E.L. tarafından ns-2 eklentisi, yine aynı üniversitede İlknur Aydın tarafından hazırlanmış QualNet eklentisi vb. çalışmalar benzetim ortamlarına örnek olarak gösterilebilir [33].

Bu tez çalışmasında ns-2 benzetim aracı üzerinde Delaware Üniversitesi P.E.L. tarafından geliştirilen SCTP uygulaması kullanılmıştır. ns benzetim ortamı 1989 yılında REAL ağ simülatörü üzerinde geliştirilerek hazırlanmıştır. 1995 yılından sonra ns projesi DARPA tarafından desteklenmeye başlamıştır. Şu an ns 2.35 sürümü ile geliştirilmesi devam etmektedir. ns-2 benzetim ortamının tercih edilme nedeni literatürde birçok çalışmada kullanılmış olması ve açık kaynak kod ile değişiklik yapılmasına olanak sağlamasıdır.

ns-2 benzetim aracı Linux işletim sisteminde çalışmaktadır. Cygwin programı yardımı ile Windows işletim sistemlerine de uyarlanması mümkündür. ns-2 C++ programla dili ile kodlanmıştır. OTCL (Object Tool Command Language – Nesne Araçlı Komut Dili) script dili ile ns-2 için script komutları ve senaryolar oluşturulabilir. Bu senaryolar derlenmiş ns-2 programı yardımı ile çalıştırılır ve trace (iz) dosyaları oluşturulur. İz dosyaları benzetim süresinde belirlenen topoloji üzerinde yapılan işlemleri kayıt altına almaktadır. Senaryo ile ilgili değerlendirmeler trace dosyaları yardımı ile yapılır. ns-2’da benzetim senaryolarını oluşturmak için düğümler, bağlantılar, trafik oluşturucular, farklı katmanlarda işlem yapan protokoller vb. bileşenler bulunmaktadır. Bu bileşenler C++ programlama dili ile tanımlanmasına rağmen OTCL scriptleri tarafından erişilebilir ve senaryolar oluşturulur. Benzetim ortamı kablolu ve kablosuz ağlarla beraber birçok farklı ağ çeşidini desteklemektedir [34-36].

ns-2 benzetim ortamından elde edilen veriler trace dosyası ile beraber “.nam” uzantılı başka bir kayıt dosyasında da tutulmaktadır. “.nam” dosyası nam (network animator) ara yüz uygulamasının kullanımı amacıyla üretilmektedir. Nam ara yüzü yardımı ile OTCL scripti oluşturulmuş senaryo görsel olarak izlenebilir. Nam programı senaryoda kullanılan düğümleri, bağlantı yollarını, paketlerin gönderimi vb. bileşenleri göstermektedir. Ayrıca nam programı yardımı ile benzetim senaryosunda ileri ve geri gidilebilir. Animasyonun hızı ayarlanabilir. Nam programının ekran görüntüsü Şekil 5.1’de görülmektedir.



Şekil 5.1. Nam uygulaması ekran görüntüsü

5.1.1. ns-2 için hazırlanmış SCTP protokolü

Bu uygulama ns-2.35/sctp yolunda bulunmaktadır. Protokolün temel işlemleri sctp.h ve sctp.cc C++ dosyaları tarafından yapılmaktadır. SCTP uygulaması, normal bağlantının başlatılması, Payload Data türündeki veri parçalarının gönderilmesi, alınan Payload Data veri parçaları için geri dönüşümün yapılması, yeniden gönderimlerin yönetimi, multihoming, multistreaming, sıralı ve sırasız gönderimler, alınan veri parçalarındaki boşlukların geri bildirim, bağlantı hatalarının tespit edilmesi, heartbeat veri parçalarının gönderilmesi vb. SCTP RFC belgesinde tanımlanmış özellikleri kapsamaktadır. Bu uygulama kısmi güvenilirlik eklentisini de desteklemektedir. Bağlantılar SCTP protokolünde tanımlandığı gibi başlama veri parçaları yardımı ile 4-way handshake şeklinde gerçekleştirilir. SCTP uygulamasına otuz farklı parametre OTCL scripti yardımı ile gönderilebilir. Alıcı kuyruk boyutunun başlangıç değeri (initialRwnd), gönderici kuyruk boyutunun başlangıç değeri (initialCwnd), kısmi güvenilirlik eklentisinin kullanılması (reliability), Payload Data veri parçası boyutu (dataChunkSize), maksimum yeniden gönderim sayısı (pathMaxRetrans), yönlendirme hesaplama gecikmesi (routeCalcDelay), sıralı

ve sırasız paket gönderim bildirimini (unordered) vb. değişkenler örnek olarak gösterilebilir. SCTP uygulaması oluşturulurken klasik ns düğüm mantığında bazı değişiklikler yapılmıştır. Bu değişiklikler ns düğümlerinin SCTP protokolünde tanımlanmış olan multihoming ve multistreaming özelliklerini desteklemek amacıyla gerçekleştirilmiştir [35].

SCTP uygulamasında bazı eksikler de bulunmaktadır. Örneğin; alıcı tarafına ulaşan veri parçaları üzerinde herhangi işlem yapılmamakta ve veri parçaları bekletilmeden kuyruktan silinmektedir. Ayrıca bağlantıların kapatılması için gereken işlemler uygulamada tanımlanmamıştır. Bu nedenle benzetim süresinin bitmesi durumunda bağlantı açık şekilde uygulama sonlandırılmaktadır.

5.1.2. SCTP uygulaması üzerinde yapılan değişiklikler

Tezin önceki bölümünde anlatılan PQ-SCTP protokolünün önceliklendirme, sıralama ve atılma kontrol algoritmalarını benzetim aracında düzgün bir şekilde çalıştırmak için mevcut SCTP kodları üzerinde bazı değişiklikler yapılmıştır. Bu değişikliklerden bazıları PQ-SCTP protokolünün tanımlanması için, bazıları ise SCTP uygulamasında bulunan eksikliklerin giderilmesi amacıyla yapılmıştır. Öncelikle, alıcı tarafında kuyruğa eklenen paketlerin üst katmana gönderilmesini simule etmek amacıyla kuyruk için işlem gecikmesi tanımlanmıştır. İşlem gecikmesi kuyruğa eklenen veri parçalarının tanımlanan gecikme süresi aralığında silinmesini sağlamaktadır. Gecikme süresinin tanımlandığı değişken “recvAppLayerDelay” olarak isimlendirilmiştir. Bu değişken için OTCL bağlantısı oluşturulmuş ve script dosyalarında tanımlanabilmesi sağlanmıştır. “recvAppLayerDelay” değişkeninin varsayılan değeri sıfırdır. Bu da hiçbir değer tanımlanmaması durumunda kuyruğa eklenecek veri parçalarının anında silme işlemine tabi tutulması anlamına gelmektedir. PQ-SCTP protokolü için yapılan bu değişiklik deneylerde eşit şartların sağlanması amacıyla SCTP protokolüne de uygulanmıştır.

SCTP uygulaması üzerinde yapılan başka bir değişiklik ise alıcı tarafın kuyruğunda yapılan işlemlerin kayıt altına alınmasıdır. İşlemler “.txt” uzantılı dosya altında

kaydedilmektedir. Kuyruk üzerinde yapılan işlemler eklenme, silinme ve atılma olarak üç farklı komut altında tutulur. Kuyruk işlemlerinin incelenmesi için oluşturulan kayıt dosyasının örneği Ek-2’de gösterilmiştir. Kayıt satırı örneği ve alanların açıklamaları ise aşağıda verilmiştir.

```
e -t 0.084808 -p 0 -src 8 -dst 14 -tsn 1
s -t 0.090808 -p 128 -src 8 -dst 14 -tsn 2
a -t 0.160808 -p 64 -src 8 -dst 14 -tsn 19
```

e – eklenme, s- silinme, a-atılma işlemlerini ifade etmektedir.

-t – işlemin gerçekleştiği zaman, saniye olarak

-p – işlem yapılan veri parçasının öncelik değeri (0 - 1.öncelik, 64 - 2.öncelik, 128 - 3.öncelik, 192 - yeniden gönderim grubunu göstermektedir)

-src – veri parçasını gönderen kaynak düğüm numarası

-dst – veri parçasının gönderildiği hedef düğüm numarası

-tsn – veri parçasının gönderim sıra numarası

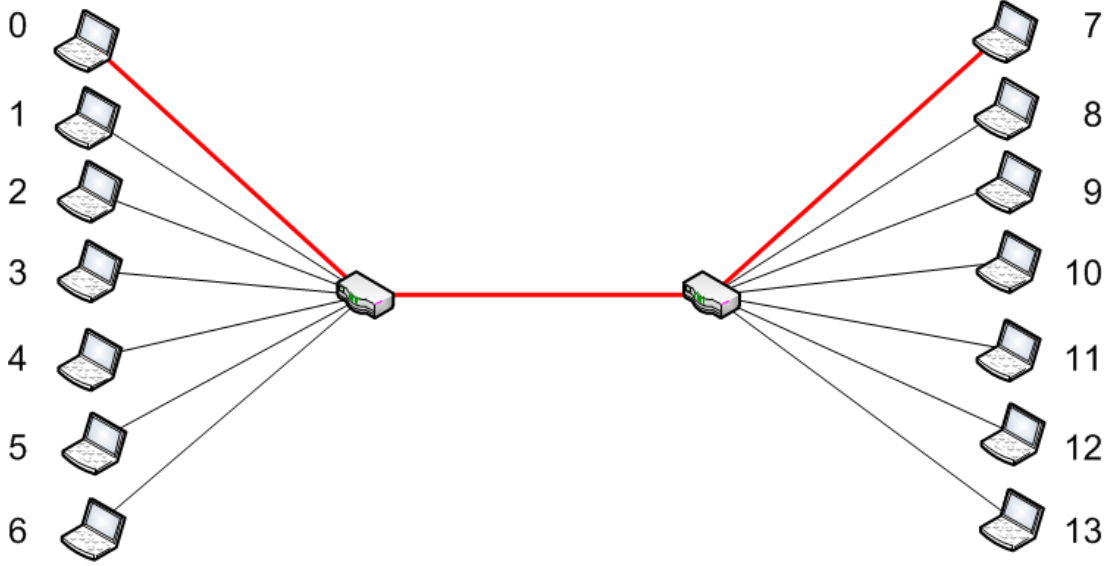
Kuyruk üzerinde yapılan işlemler hem SCTP hem de PQ-SCTP uygulamalarında kaydedilmektedir.

Gönderici tarafında veri parçaların öncelik etiketi tanımlanması için hem SCTP hem de PQ-SCTP protokollerinde “GenChunk” metodunda değişiklik yapılmıştır. Bu değişiklik önceden belirlenen sıralamaya göre üretilen veri parçalarına öncelik değeri tanımlamaktadır. Öncelik sıralaması için “IBBPBBPBBP” çerçeve sıralaması kullanılmıştır. Bu dizi [13]’te kullanılan veri parçası sıralaması referans olarak üretilmiştir. Yeniden gönderilen paketlerin önceliği ise “RtxMarkedChunks” metodunda tanımlanmaktadır.

Bu değişiklikler dışında PQ-SCTP sıralama ve kuyruktan atılma işlemlerini gerçekleştirebilmek için sadece PQ-SCTP uygulamasına önceki bölümde anlatılan algoritmalar eklenmiştir. Kuyruksal sıralama işlemi “InsertInBufferStream”, kuyruktan atılma işlemi ise “ProcessDataChunk” metotlarında çalışmaktadır.

5.2. Dumbbell Topolojisi Simülasyonu

Dumbbell topolojisi 16 düğümden oluşmaktadır. Bu düğümlerden yedisi gönderici diğer yedisi ise alıcı düğümlerdir. Gönderici ve alıcı düğümler arasında iki adet yönlendirici düğüm bulunmaktadır. Topoloji Şekil 5.2’de görülmektedir.



Şekil 5.2. Dumbbell topolojisi

Dumbbell topolojisinde alıcı ve gönderici düğümler arasında çiftler halinde bağlantı bulunmaktadır. 0 numaralı düğüm 7 numaralı düğümlle, ... 6 numaralı düğüm ise 13 numaralı düğümlle veri alışverişi yapmaktadır. Gönderici ve alıcı düğümlerin hepsinde ulaşım katmanı protokolü olarak SCTP veya PQ-SCTP protokolü kullanılmıştır. Gönderici düğümler CBR (Constant Bit Rate – Sabit Bit Oranı) yardımı ile sabit oranda veri üretmektedir. Bu topolojide 0 numaralı düğüm ile 7 numaralı düğüm arasındaki veri alışverişi dikkate alınmıştır. Diğer düğümler arka plan trafiği oluşturmaktadır. Topoloji için hazırlanmış OTCL scripti Ek-3’de verilmiştir. Bazı kod parçacıkları ise aşağıda açıklanmıştır.

```
set sctp($i) [new Agent/SCTP]
sctp($i) set initialRwnd_ 10240
sctp($i) set recvAppLayerDelay_ 0.002
```

Yukarıdaki kodda öncelikle yeni SCTP nesnesi (agent) oluşturulmaktadır. Daha sonra bu nesne önceden oluşturulmuş düğümlere atanacaktır. Daha sonra SCTP nesnesi için bazı başlangıç parametreleri tanımlanmaktadır. Bu örnekte başlangıç alıcı kuyruk uzunluğu 10240 B (10KB) olarak tanımlanmıştır. Alıcı taraf işlem gecikmesi süresi ise 0.002 s (2ms) olarak belirtilmiştir.

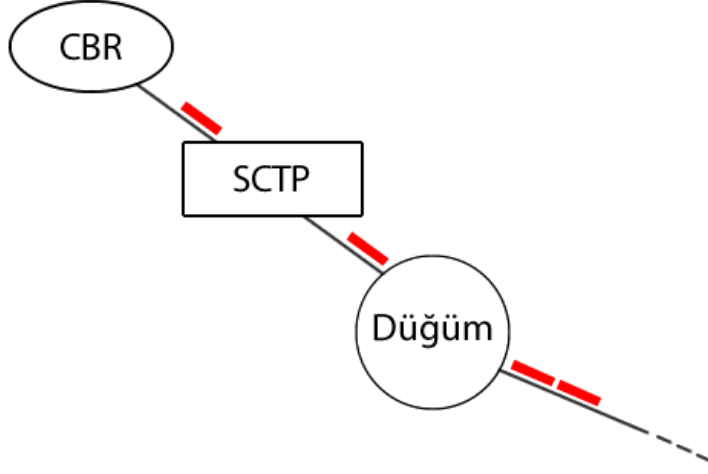
```
set cbr(1) [new Application/Traffic/CBR]
cbr(1) set packetSize_ 1310
cbr(1) set rate_ 4Mb
cbr(1) attach-agent $sctp(1)
```

Bu kodda ise yeni CBR trafik nesnesi tanımlanmaktadır. Görüldüğü gibi CBR nesnesi 1310 B paket boyutuna ve 4Mb/s gönderim oranına ayarlanmıştır. Son satır ise oluşturulan CBR nesnesinin daha önce oluşturulmuş SCTP nesnesi ile ilişkilendirme işlemini göstermektedir.

```
$ns connect $sctp(1) $sctp(2)
```

İki farklı SCTP nesnesi yukarıdaki satır ile birbirine bağlanmaktadır. Bu \$sctp(1) ve \$sctp(2) nesnesine sahip olan düğümlerin kendi aralarında SCTP protokolü yardımı ile haberleşeceği anlamına gelmektedir.

Genel olarak bu akış şöyle ifade edilebilir. Gönderici ve alıcı taraftaki her düğüm için SCTP nesnesi oluşturulmuş ve düğüm ile ilişkilendirilmiştir. Benzer şekilde her bir SCTP nesnesi bir CBR trafik oluşturucu ile bağlantı içerisindedir. CBR tarafından üretilen veri SCTP nesnesi üzerinde SCTP protokolü kuralları uygulanarak paket oluşturmakta, oluşturulan paket ise düğümler arasındaki bağlantı üzerinden gönderilmektedir. Bu durum Şekil 5.3.'de görülmektedir.



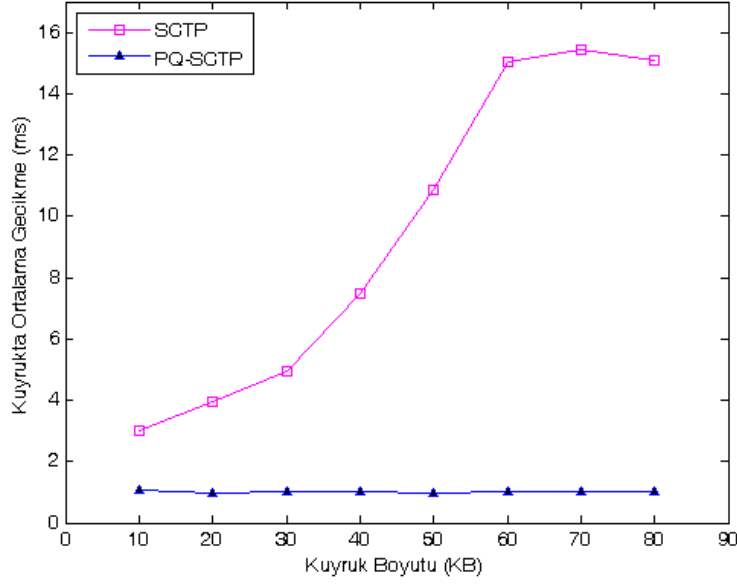
Şekil 5.3. ns-2 üzerinde düğüm ve protokol yapısı

Dumbbell topolojisi üzerinde yapılan deneyler ve elde edilen sonuçlar aşağıda verilmiştir.

Dumbelle topolojisi üzerinde yapılan tüm deneylerde bant genişlikleri ağ üzerinde tıkanıklık oluşmaması şeklinde alınmıştır. Gönderici düğümler ve yönlendirici düğüm arasında 20Mb/s bant genişliği ve 10ms hat gecikmesi tanımlanmıştır. Benzer şekilde alıcı düğümler ve yönlendirici düğüm arasında bulunan bağlantı da 20Mb/s bant genişliği ve 10ms hat gecikmesine sahiptir. İki yönlendirici arasındaki bağlantı ise 15Mb/s bant genişliği ve 10ms hat gecikmesi olarak tanımlanmıştır.

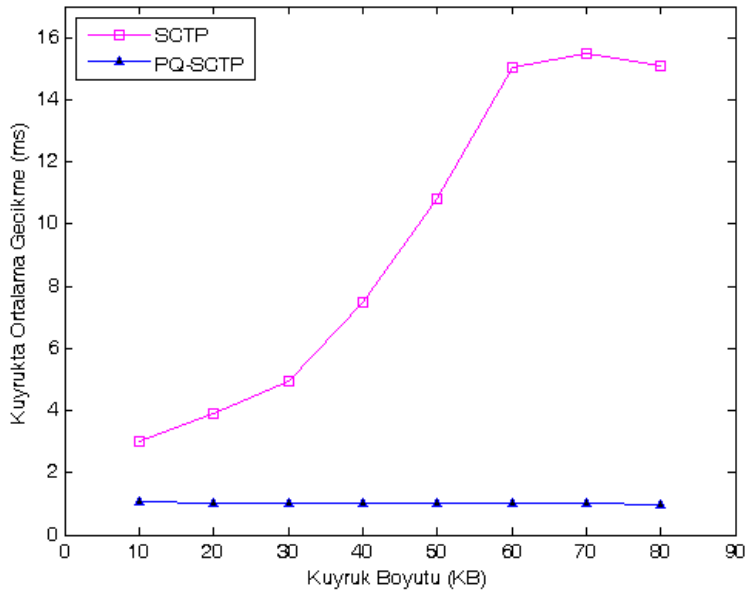
Dumbbell topolojisi için kuyruk boyutuna bağlı olarak throughput değişimi bölüm 4.4'de anlatılmıştır.

İlk deneyde sabit gönderim oranı (4Mb/s) ve işlem gecikmesi (2ms) için kuyruk boyutuna bağlı olarak ortalama kuyruk gecikmesi değerleri elde edilmiştir. Şekil 5.4'de 1.öncelik grubu için ortalama kuyruk gecikmesi değerleri verilmiştir.



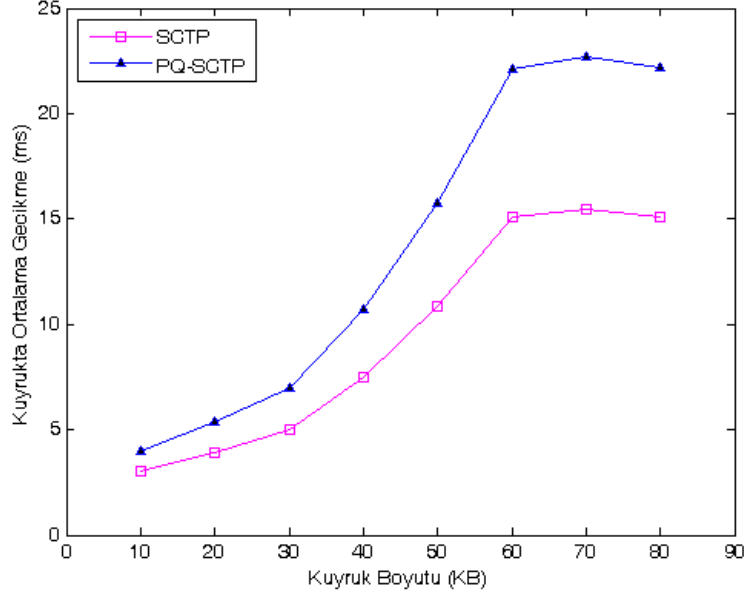
Şekil 5.4. Dumbbell topolojisinde 1.öncelik grubu için ortalama kuyruk gecikmesi

Görüldüğü gibi SCTP protokolünde değerler 60KB kuyruk boyutuna kadar yükselmiş, daha sonra ise sabitlenmiştir. PQ-SCTP protokolünde ise kuyruk gecikmesi ~1ms değerine sahiptir. Benzer şekilde 2.öncelik grubu için ortalama kuyruk gecikmesi değerleri Şekil 5.5’de sunulmuştur.



Şekil 5.5. Dumbbell topolojisinde 2.öncelik grubu için ortalama kuyruk gecikmesi

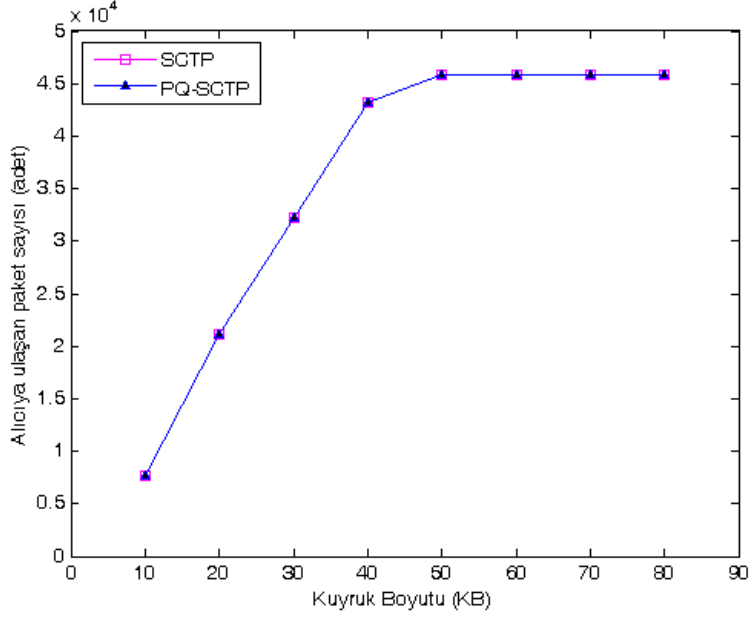
3.öncelik grubu için ortalama kuyruk gecikmesi değerleri ise Şekil 5.6'da görülmektedir.



Şekil 5.6. Dumbbell topolojisinde 3.öncelik grubu için ortalama kuyruk gecikmesi

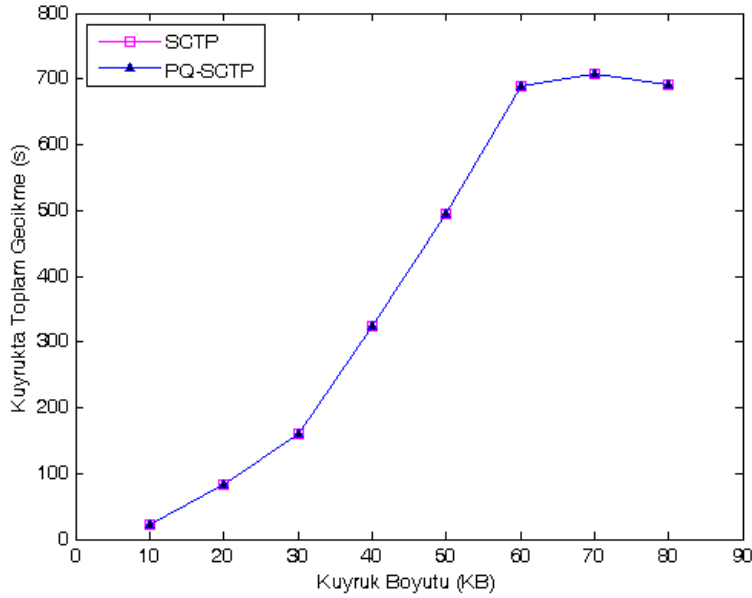
Görüldüğü gibi 3.öncelik grubunda SCTP protokolü daha düşük ortalama gecikme değerlerine sahiptir. İki protokol arasındaki gecikme farkı ise maksimum throughput değerine yaklaştıkça (60KB) daha da artmaktadır. 3.önceliğe sahip veri parçaları için PQ-SCTP protokolünde böyle değerlerin elde edilmesi deneylerin doğruluğunu göstermektedir.

Deneylerin doğruluğunu test etmek için başka bir yöntem olarak iki protokol arasında toplam kuyruk boyutu farkı hesaplanmıştır. Şekil 5.7'de görüldüğü gibi hem SCTP, hem de PQ-SCTP protokolleri için alıcı taraf aynı kuyruk boyutlarında eşit miktarda veri parçası kabul etmektedir.



Şekil 5.7. Dumbbell topolojisinde kuyruk boyutuna bağlı alıcıya ulaşan veri parçası adedi

Eş. 5.1'e göre hesaplanan toplam kuyruk bekleme değeri Şekil 5.8'de verilmiştir.



Şekil 5.8. Dumbbell topolojisinde kuyruk boyutuna bağlı toplam kuyruk gecikmesi

Değerler Çizelge 5.1 ve Çizelge 5.2'de sunulmuştur. Sarı renkle gösterilen alan 4Mb/s gönderim oranı için önerilen kuyruk boyutudur [5].

Çizelge 5.1. Dumbbell topolojisinde SCTP protokolü için benzetim sonuçları

Protokol (120ms end-to-end delay)	SCTP protokolü için sonuçlar (Dumbbell.tc)									
	120s					2ms				
Simülasyon süresi	120s									
İşlem Gecikmesi	2ms									
Kuyruk Boyutu (Bayt)	10240	20480	30720	40960	51200	59392	61440	71680	81920	
Transmission rate (Mb/s)	4									
1.grup Ort. Kuyrukta Gecikme (ms)	3,011	3,942	4,941	7,473	10,840	11,176	15,060	15,451	15,111	
2.grup Ort. Kuyrukta Gecikme (ms)	3,016	3,899	4,970	7,477	10,825	11,195	15,060	15,473	15,102	
3.grup Ort. Kuyrukta Gecikme (ms)	3,001	3,923	5,007	7,476	10,826	11,192	15,062	15,477	15,127	
Rtx Ort. Kuyrukta Gecikme (ms)	0	0	0	0	0	0	0	0	0	
Toplam Kuyrukta Gecikme (s)	23,024	82,566	161,144	323,058	495,774	512,332	689,635	708,506	692,176	
1.grup Ort. IPDV	1,8712045	3,080465	2,6447405	4,4931375	7,5033867	9,1855783	13,6232614	13,9522018	13,3509514	
2.grup Ort. IPDV	2,2084797	2,5024227	3,6587983	6,2082559	6,5863281	6,8235821	8,8084125	8,9063019	8,6070467	
3.grup Ort. IPDV	2,2010433	2,6162933	2,8662461	3,1823014	3,2327972	3,256483	3,4944245	3,5052238	3,4613402	
Rtx Ort. IPDV	0	0	0	0	0	0	0	0	0	
1.grup Max. IPDV	6,264	7,273	11,338	14,590	20,811	20,811	20,811	20,811	20,811	
2.grup Max. IPDV	3,865	8,214	9,390	13,582	24,590	28,369	28,370	28,369	28,370	
3.grup Max. IPDV	5,694	8,447	11,640	17,486	25,044	30,888	30,889	30,889	30,889	
Rtx Max. IPDV	0	0	0	0	0	0	0	0	0	
1.grup Min. IPDV	0,050	0,013	0,004	0,005	0,005	0,012	0,168	0,239	0,083	
2.grup Min. IPDV	0,085	0,065	0,005	0,078	0,059	0,012	0,317	0,317	0,065	
3.grup Min. IPDV	0,377	0,072	0,221	0,221	0,174	0,173	0,047	0,174	0,174	
Rtx Min. IPDV	0	0	0	0	0	0	0	0	0	
1.grup Max. Jitter	3,390	4,846	7,555	9,087	15,568	19,784	15,911	25,272	42,109	
2.grup Max. Jitter	3,386	4,664	7,103	9,780	15,297	19,719	15,900	25,606	42,274	
3.grup Max. Jitter	3,401	7,384	8,168	11,041	16,509	19,780	16,073	25,815	42,620	
Rtx Max. Jitter	0	0	0	0	0	0	0	0	0	
1.grup Min. Jitter	-3,007	-3,941	-4,938	-7,469	-10,838	-11,171	-15,014	-15,449	-15,104	
2.grup Min. Jitter	-3,013	-3,898	-4,968	-7,476	-10,824	-11,187	-15,060	-15,470	-15,094	
3.grup Min. Jitter	-2,998	-3,922	-5,006	-7,475	-10,825	-11,190	-15,061	-15,476	-15,127	
Rtx Min. Jitter	0	0	0	0	0	0	0	0	0	
Alıcıya ulaşan toplam paket sayısı	7661	21065	32287	43213	45789	45782	45789	45789	45781	

Çizelge 5.2. Dumbbell topolojisinde PQ-SCTP protokolü için benzetim sonuçları

Protokol (120ms end-to-end delay)	PQ-SCTP protokolü için sonuçlar (Dumbbell.tcl)									
	120s									
	2ms									
Simülasyon süresi	10240	20480	30720	40960	51200	61440	71680	81920		
İşlem Gecikmesi	4									
Kuyruk Boyutu (Bayt)	10240	20480	30720	40960	51200	61440	71680	81920		
Transmission rate (Mb/s)	4									
1.grup Ort. Kuyrukta Gecikme (ms)	1,091	0,986	1,004	0,999	0,986	1,004	0,997	1,001		
2.grup Ort. Kuyrukta Gecikme (ms)	1,093	0,997	1,000	0,997	0,995	1,005	0,999	0,991		
3.grup Ort. Kuyrukta Gecikme (ms)	3,962	5,383	6,986	10,716	15,746	22,092	22,713	22,183		
Rtx Ort. Kuyrukta Gecikme (ms)	0	0	0	0	0	0	0	0		
Toplam Kuyrukta Gecikme (s)	23,024	82,569	161,144	323,068	495,792	689,704	708,579	692,205		
1.grup Ort. IPDV	0,2702362	0,7142128	0,6732724	0,7306657	0,7323867	0,772117	0,8246466	0,8386011		
2.grup Ort. IPDV	0,2809359	0,5982002	0,610461	0,5828824	0,5769698	0,5812241	0,5794112	0,5804917		
3.grup Ort. IPDV	1,8973578	2,0844574	2,3180042	2,5925858	2,8708787	2,8265534	3,0467075	3,0390084		
Rtx Ort. IPDV	0	0	0	0	0	0	0	0		
1.grup Max. IPDV	1,830	1,915	1,987	1,977	1,983	1,935	1,870	1,830		
2.grup Max. IPDV	1,915	1,928	1,957	1,987	1,974	1,987	1,930	1,935		
3.grup Max. IPDV	6,435	8,584	12,381	16,842	25,455	31,629	31,629	31,629		
Rtx Max. IPDV	0	0	0	0	0	0	0	0		
1.grup Min. IPDV	0,050	0,013	0,004	0,005	0,005	0,005	0,013	0,013		
2.grup Min. IPDV	0,085	0,007	0,004	0,005	0,005	0,005	0,013	0,012		
3.grup Min. IPDV	0,220	0,005	0,012	0,046	0,018	0,012	0,047	0,220		
Rtx Min. IPDV	0	0	0	0	0	0	0	0		
1.grup Max. Jitter	0,909	1,014	0,996	1,001	1,014	1,007	0,996	1,003		
2.grup Max. Jitter	0,906	1,003	1,000	1,003	1,005	1,014	0,995	1,001		
3.grup Max. Jitter	3,180	5,924	12,189	13,885	15,697	20,330	19,876	38,579		
Rtx Max. Jitter	0	0	0	0	0	0	0	0		
1.grup Min. Jitter	-1,091	-0,985	-1,004	-0,999	-0,986	-0,993	-1,004	-0,996		
2.grup Min. Jitter	-1,092	-0,997	-1,000	-0,997	-0,995	-0,986	-1,005	-0,999		
3.grup Min. Jitter	-3,959	-5,382	-6,985	-10,715	-15,745	-16,291	-22,091	-22,712		
Rtx Min. Jitter	0	0	0	0	0	0	0	0		
Aliciya ulaşan toplam paket sayısı	7661	21065	32287	43213	45789	45782	45789	45781		

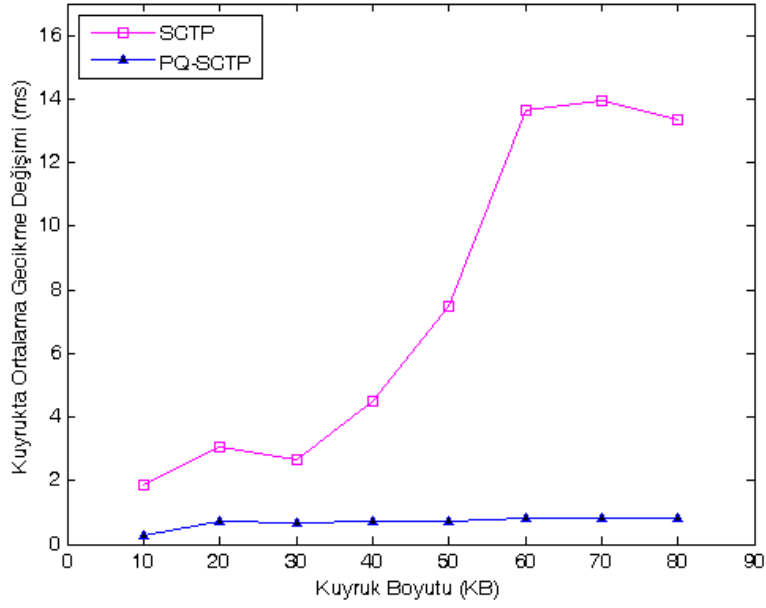
Şekil 5.8’de görüldüğü gibi eşit şartlar altında hem SCTP, hem de PQ-SCTP protokolleri için kuyrukta toplam bekleme süreleri aynı değerlere sahiptir.

Çizelge 5.3’de SCTP ve PQ-SCTP protokolleri için farklı kuyruk boyutlarında toplam kuyruk gecikmesi sonuçları sunulmuştur.

Çizelge 5.3. Dumbbell topolojisinde toplam kuyruk gecikmeleri

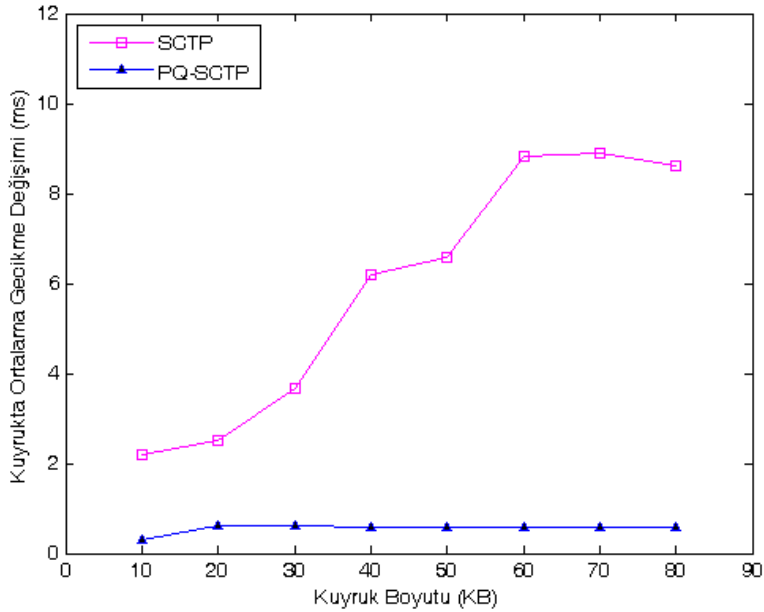
Kuyruk Boyutu	SCTP	PQ-SCTP
	Kuyrukta toplam gecikme (s)	Kuyrukta toplam gecikme (s)
10KB	23,024	23,024
20KB	82,566	82,569
30KB	161,144	161,144
40KB	323,058	323,068
50KB	495,774	495,792
60KB	689,635	689,704
70KB	708,506	708,579
80KB	692,176	692,205

Dumbbell topolojisi için gecikme süreleri ile beraber gecikme değişimi değerleri de elde edilmiştir. Gecikme değişimi değerleri Eş. 5.2 kullanılarak hesaplanmıştır. 1.öncelik grubu için gecikme değişimi Şekil 5.9’da görülmektedir.



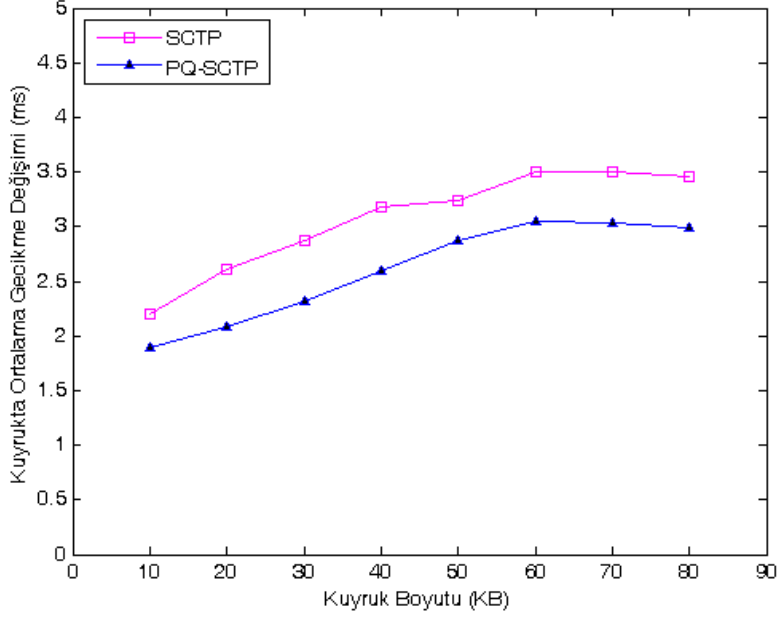
Şekil 5.9. Dumbbell topolojisinde 1.öncelik grubu için kuyruk gecikmesi değişimi

Benzer şekilde 2.öncelik grubu için de gecikme değişimi değerleri elde edilmiştir. Sonuçlar Şekil 5.10’da verilmiştir.



Şekil 5.10. Dumbbell topolojisinde 2.öncelik grubu için kuyruk gecikmesi değişimi

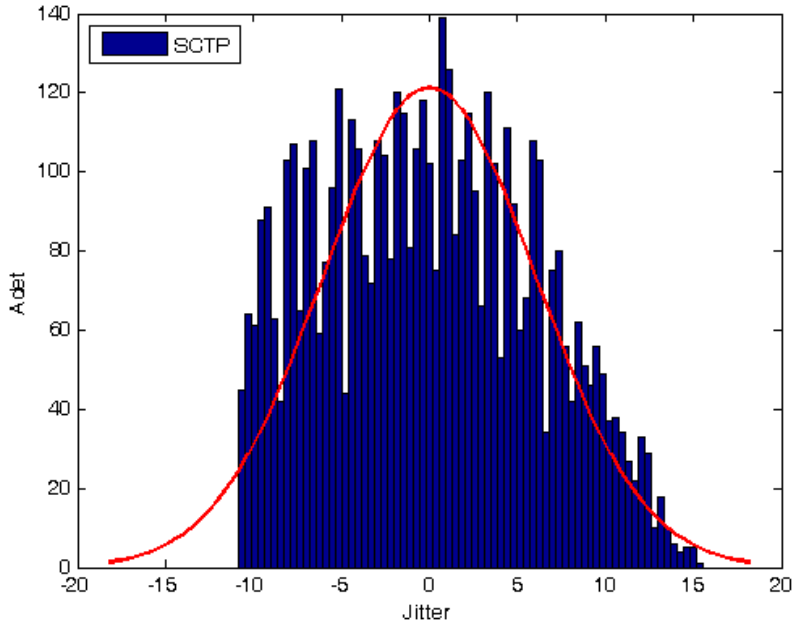
3. öncelik grubu için elde edilmiş gecikme değişimi değerleri ise Şekil 5.11’de verilmiştir.



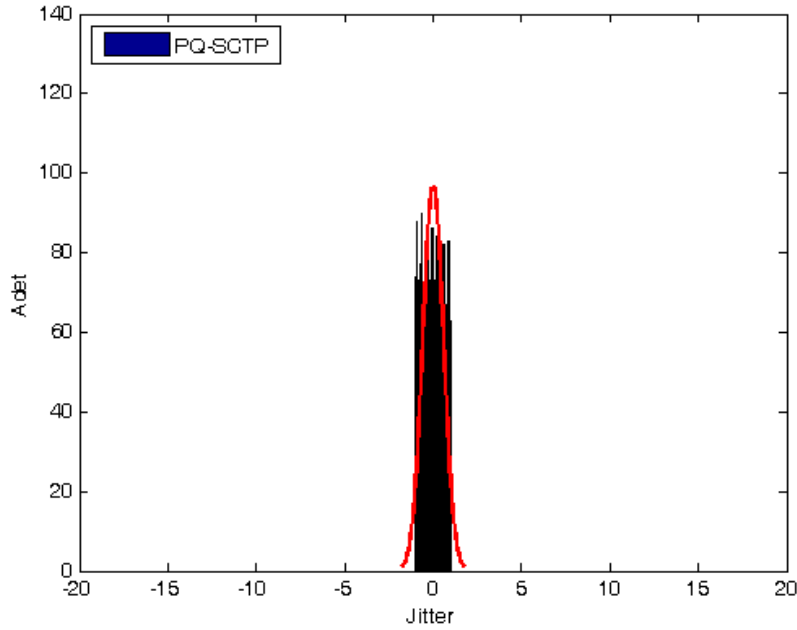
Şekil 5.11. Dumbbell topolojisinde 3. öncelik grubu için kuyruk gecikmesi değişimi

1. ve 2. öncelik grupları için elde edilen ortalama gecikme değişimi değerleri SCTP protokolü için sürekli artış gösterirken PQ-SCTP protokolünde çok fazla değişim göstermemektedir. 3. öncelik grubunda ise SCTP protokolü PQ-SCTP protokolünden daha iyi sonuç üretmiştir. Bu durum 3. öncelikli veri parçalarının kuyruk sonunda birikmesinden kaynaklanmaktadır.

Son olarak dumbbell topolojisi için farklı öncelik gruplarında jitterin dağılımı incelenmiştir. Jitter Eş. 5.4’de verilen yöntemle hesaplanmıştır. 1. öncelik grubu için SCTP ve PQ-SCTP protokollerindeki jitter dağılımı Şekil 5.12’de görülebilir.



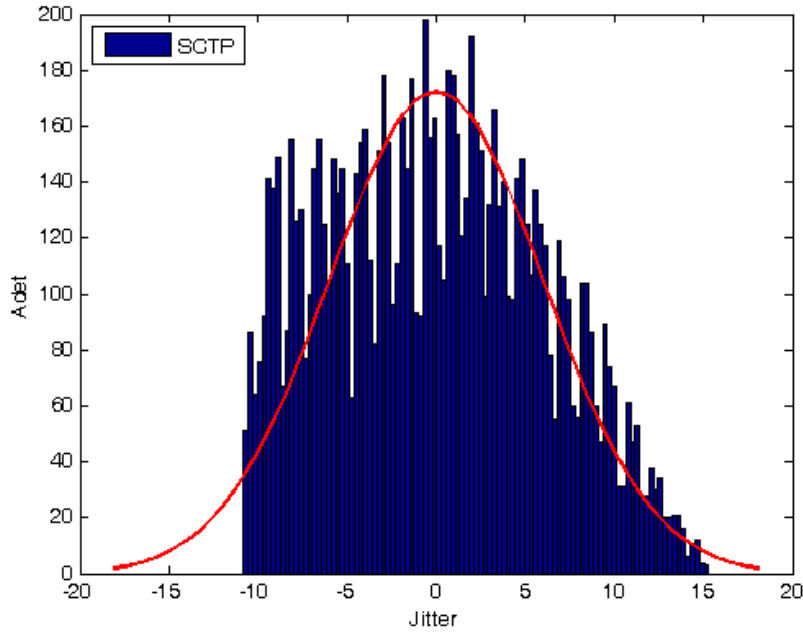
a



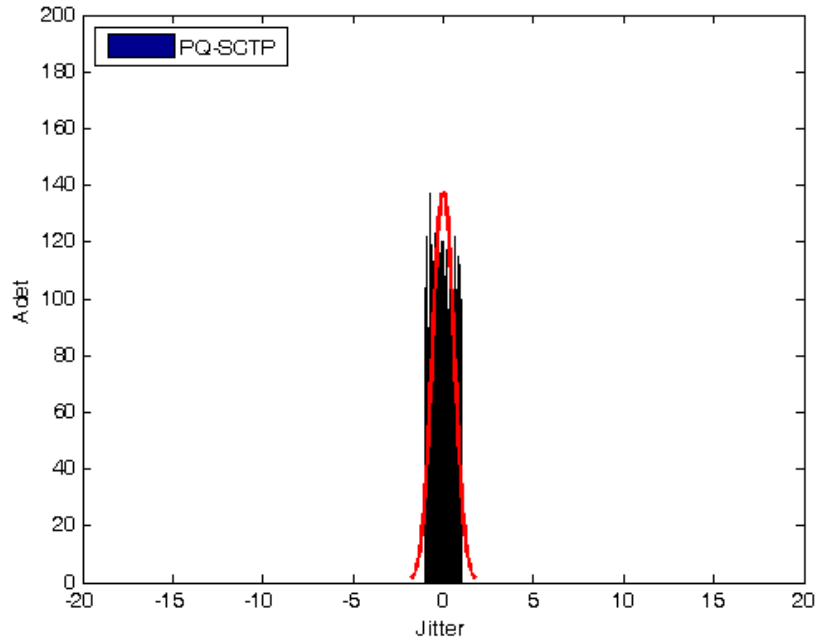
b

Şekil 5.12. Dumbbell topolojisinde 1.öncelik grubu için jitter dağılımı
 a. SCTP protokolü için jitter dağılımı b. PQ-SCTP protokolü için jitter dağılımı

2.öncelik grubuna sahip veri parçaları için jitter dağılımları ise Şekil 5.13'de verilmiştir.



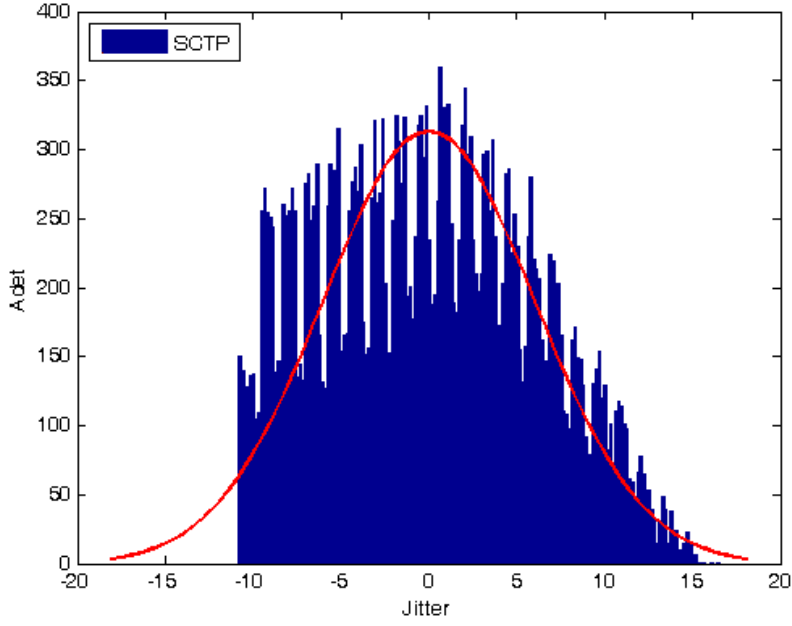
a



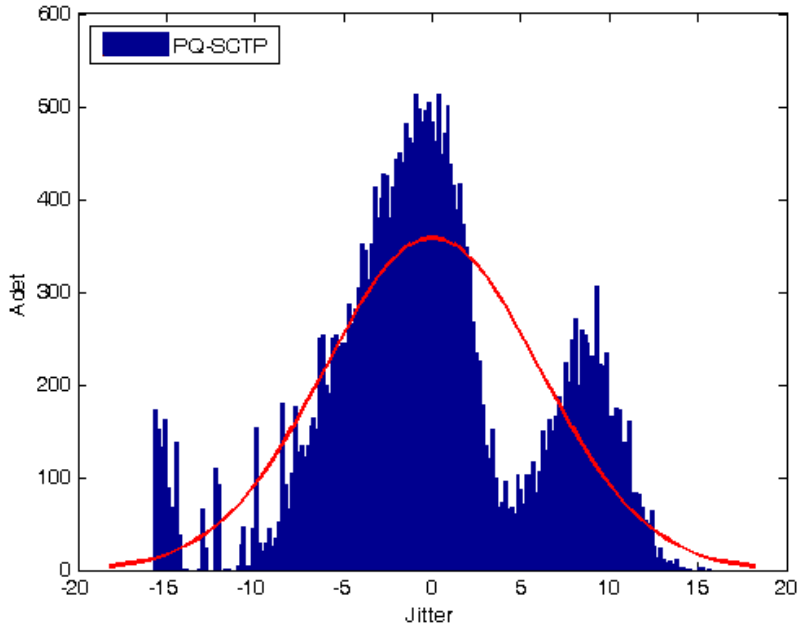
b

Şekil 5.13. Dumbbell topolojisinde 2.öncelik grubu için jitter dağılımı
 a. SCTP protokolü için jitter dağılımı b. PQ-SCTP protokolü için jitter dağılımı

3.öncelik grubu için jitter dağılımları Şekil 5.14'de görülebilir.



a



b

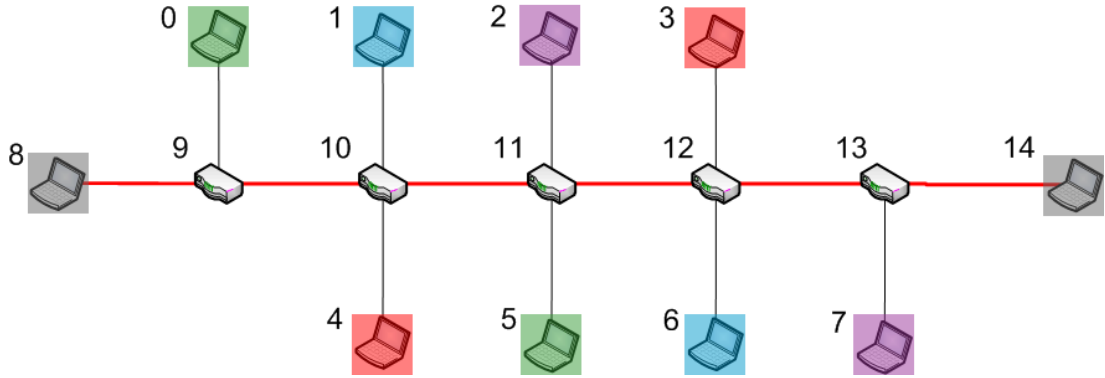
Şekil 5.14. Dumbbell topolojisinde 3.öncelik grubu için jitter dağılımı
 a. SCTP protokolü için jitter dağılımı b. PQ-SCTP protokolü için jitter dağılımı

Dağılım grafiklerinden görüldüğü gibi 1.öncelik grubuna sahip veri parçalarının jitter dağılımı SCTP protokolü için -10ms ile 10ms arasında değişirken PQ-SCTP

protokolünde bu değerler -1ms ile 1ms arasına yayılmıştır. Ayrıca jitterin dağılımı SCTP ile kıyaslandığında eşit değerlere sahiptir. 2. öncelik grubu için de dağılım aynı şekildedir. 3. öncelik grubun dağılımı ise hem PQ-SCTP, hem de SCTP protokolünde aynı aralığa sahiptir. 1. ve 2. grublardaki iyileşmeye rağmen 3. gruptaki düşüş benzetim değerlerinin doğruluğunu kanıtlamaktadır.

5.3. Parking Lot Topoloji Simülasyonu

Parking lot topolojisi on beş düğümden oluşmaktadır. Bu düğümlerden beşi gönderici, beşi alıcı geri kalan düğümler ise yönlendiricidir. Topoloji Şekil 5.15’de verilmiştir.

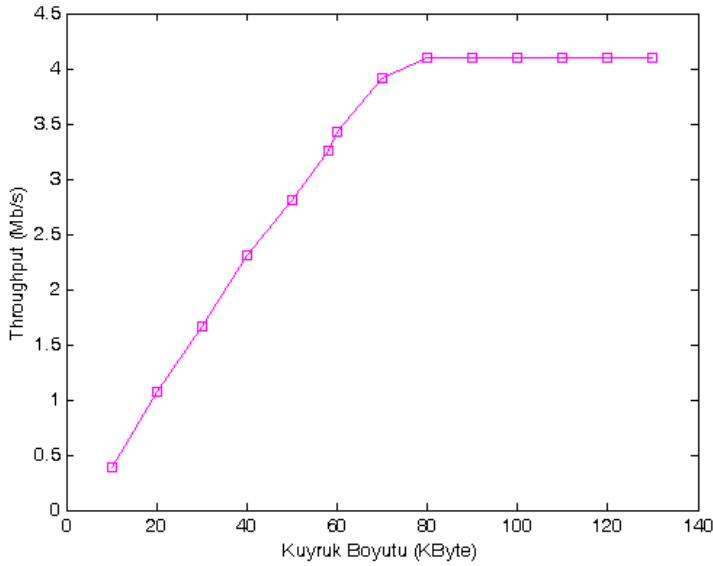


Şekil 5.15. Parking lot topolojisi

Şekilde gösterildiği gibi aynı renge sahip gönderici ve alıcı düğümler kendi aralarında veri alış verişi yapmaktadır. Bu topolojide 8 numaralı düğüm ile 14 numaralı düğüm arasındaki iletişim dikkate alınmıştır. Diğer düğümler sadece arka plan trafiği oluşturmaktadır. Arka plan trafiği oluşturan düğümlerin tümünde üç adet yönlendirici düğüm üzerinden bağlantı yapmalarına dikkat edilmiştir. Dumbbell topolojisinde olduğu gibi parking lot topolojisinde de her bir düğüme simülasyonda kullanılan protokole bağlı olarak SCTP veya PQ-SCTP nesnelere bağlanmıştır. Gönderici düğümlerde bulunan protokol nesnelere CBR trafik üreticisi ile ilişkilendirilmiştir. Topoloji için hazırlanmış OTCL scripti Ek-4’de verilmiştir. Topolojide bant genişliği ve hat gecikme parametreleri sabit olarak alınmıştır. Omurga ağ üzerinde bulunan tüm bağlantılar (8-14 numaralı düğümler arası) 10Mb/s

bant genişliği ve 10ms hat gecikmesine sahiptir. Kenar bağlantılar için 20Mb/s bant genişliği, 10ms hat gecikmesi tanımlanmıştır. Düğümler arasındaki veri alışverişi Şekil 5.15’de renk çiftleri olarak tanımlanmıştır. Bu durumda trafiğin en yoğun olduğu bağlantılar 10-11 ve 11-12 düğümleri arasında olacaktır. 8 – 14 düğümleri arasında 4Mb/s gönderim oranı, diğer düğümlerde ise 0,5 Mb/s gönderim oranı bulunduğu dikkate alınır en yoğun trafik 5,5Mb/s olacağı beklenmektedir. Bu durumda ağ üzerinde hiçbir şekilde tıkanıklık oluşmayacaktır.

Tezin daha önceki bölümlerinde bahsedildiği gibi SCTP protokolü veri parçalarındaki a_rwnd alanı yardımı ile bant genişliğini dolayısıyla paket atılmalarını kontrol etmektedir. Kuyruk (pencere) boyutuna bağlı olarak topolojideki throughput değişimi Şekil 5.16’de görülmektedir.

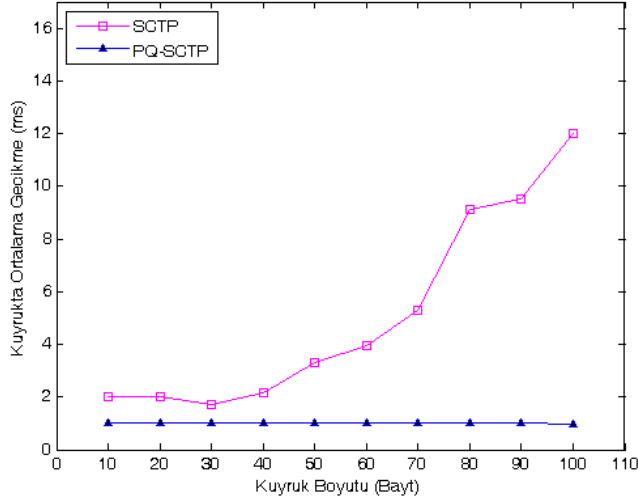


Şekil 5.16. Parking lot topolojisinde kuyruk boyutuna göre throughput değişimi

Görüldüğü gibi 80KB kuyruk boyutuna kadar througput değeri sürekli artmaktadır. 80KB ve üzeri kuyruk boyutlarında ise maksimum throughput değerine ulaşılmıştır.

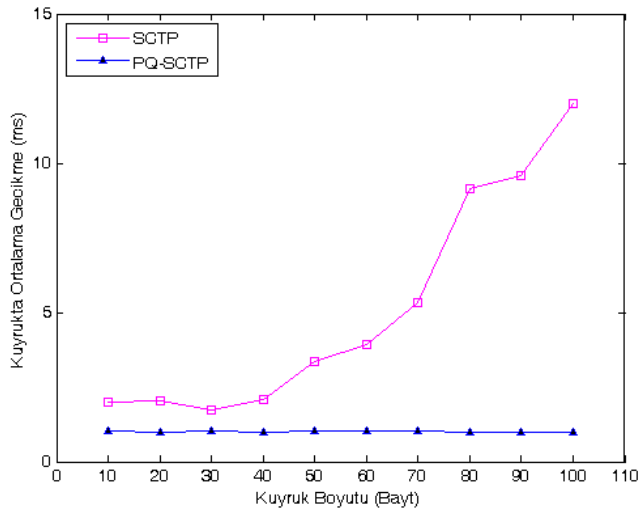
Parking lot topolojisi üzerinde yapılan diğer deneyler ve elde edilen sonuçlar aşağıda verilmiştir.

İlk başta sabit gönderim oranı (4Mb/s) ve işlem gecikmesinde (2ms) farklı kuyruk boyutları için öncelik gruplarının kuyrukta ortalama gecikme süreleri kontrol edilmiştir. Şekil 5.17’de 1.öncelik grubu için SCTP ve PQ-SCTP protokollerinde ortalama kuyruk gecikmesi sunulmuştur.



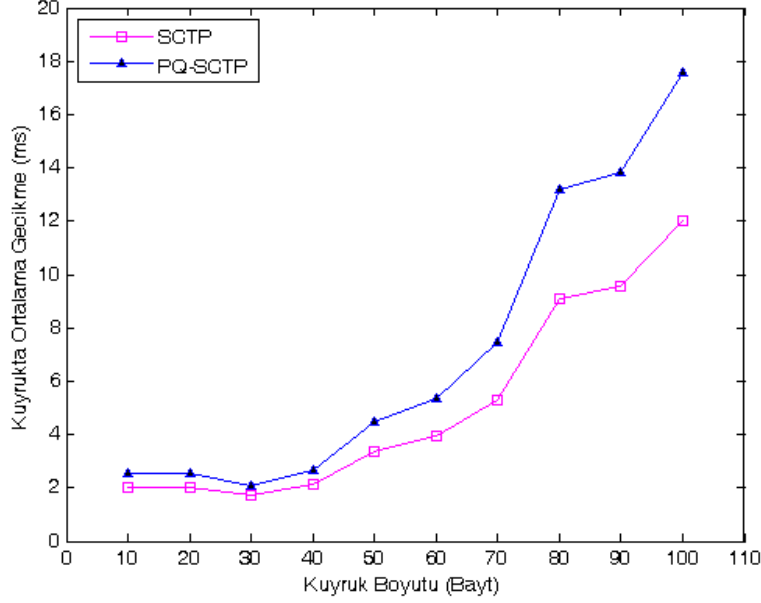
Şekil 5.17. Parking lot topolojisinde 1.öncelik grubu için ortalama kuyruk gecikmesi

Grafikten görüldüğü gibi SCTP protokolü için 10-100KB kuyruk boyutu değişiminde I çerçevelerin kuyrukta gecikme süresi 2ms ile 12ms arasında yükselirken PQ-SCTP protokolü için 1ms sabit olarak kalmaktadır. Şekil 5.18’de 2.öncelik grubu (P çerçeve) için benzer sonuçlar görülmektedir.



Şekil 5.18. Parking lot topolojisinde 2.öncelik grubu için ortalama kuyruk gecikmesi

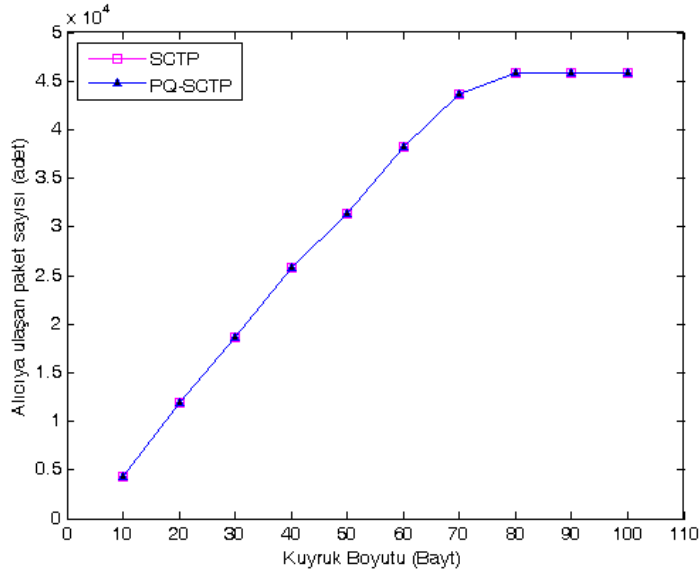
Bu grafikten de görüldüğü gibi SCTP protokolü için ortalama kuyruk gecikmesinde sürekli bir yükseliş olmasına rağmen PQ-SCTP protokolünde bu değer ~1ms olarak sabit kalmaktadır. 3.öncelik grubu (B çerçeve) için ortalama kuyruk gecikmesi ise Şekil 5.19’de verilmiştir.



Şekil 5.19. Parking lot topolojisinde 3.öncelik grubu için ortalama kuyruk gecikmesi

3.öncelik grubu için hem SCTP, hem de PQ-SCTP protokolleri için benzer artış söz konusudur. Ayrıca PQ-SCTP protokolünde ortalama gecikme sürelerinin SCTP protokolüne kıyasla daha fazla olduğu gözlenmiştir. 3.öncelik grubunda bulunan veri parçaları her zaman kuyruğun sonlarında yer alacağı için gecikmede böyle bir artışın oluşması olağan durumdur. Her üç grafikte 80KB kuyruk boyutuna erişildiğinde SCTP protokolü için ortalama gecikme sürelerinde ani artış gözükmektedir. Bu artış ağın maksimum throughput değerine erişmesinden kaynaklanmaktadır.

SCTP protokolünde gönderici tarafı için paket önceliklendirilmesi dışında herhangi bir değişiklik yapılmadığından alıcıya ulaşan paket sayısında herhangi bir değişiklik beklenmemektedir. Şekil 5.20’de hem SCTP, hem de PQ-SCTP protokolleri için aynı kuyruk boyutlarında alıcıya eşit sayıda veri parçası ulaştığı görülmektedir.



Şekil 5.20. Parking lot topolojisinde kuyruk boyutuna bağlı alıcıya ulaşan veri parçası adedi

Toplam veri parçası sayısı gibi farklı öncelik grupları için kuyruğa eklenen veri parçası sayısı da eşit değerlere sahiptir. Değerler Çizelge 5.4 ve Çizelge 5.5’de verilmiştir. Sarı renkle gösterilen alan 4Mb/s gönderim oranı için önerilen kuyruk boyutudur [5].

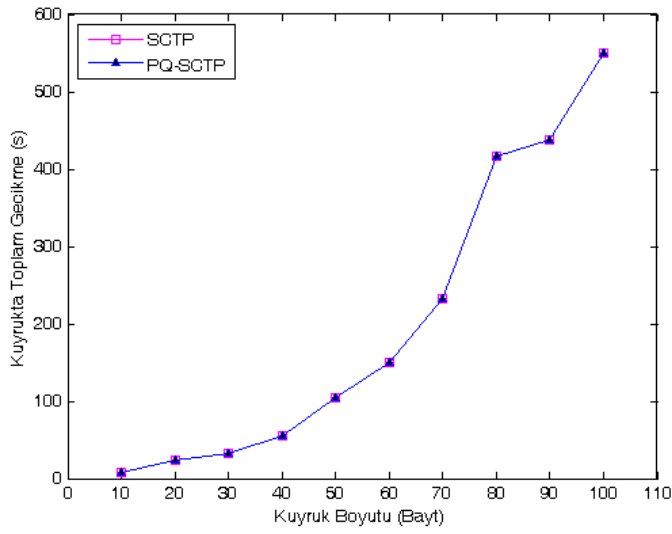
Her iki protokol için eşit gönderim oranı ve eşit işlem gecikmesi dikkate alınırsa aynı kuyruk boyutlarında kuyruktaki tüm veri parçalarının toplam gecikme sürelerinin bir birine çok yakın değerlere sahip olması gerekir. Eş. 5.1’e göre yapılan hesaplamalarda bu değerlerin eşit olduğu gözlemlenmiştir. SCTP ve PQ-SCTP protokolleri için toplam gecikme değerlerinin değişimi Şekil 5.21’de görülmektedir.

Çizelge 5.4. Parking lot topolojisinde SCTP protokolü için benzetim sonuçları

Protokol (120ms end-to-end delay)	SCTP protokolü için sonuçlar (parking lot.tcl)											
	120s											
	2ms											
İşlem Gecikmesi	10240	20480	30720	40960	51200	61440	71680	81920	92160	102400		
Kuyruk Boyutu (Bayt)						59392	61440	71680	81920	92160	102400	
Transmission rate (Mb/s)						4						
1.grup Ort. Kuyrukta Gecikme (ms)	2,005	2,008	1,714	2,162	3,319	4,096	3,951	5,296	9,102	9,530	12,0232	
2.grup Ort. Kuyrukta Gecikme (ms)	2,015	2,059	1,729	2,088	3,335	4,094	3,923	5,315	9,125	9,568	12,0155	
3.grup Ort. Kuyrukta Gecikme (ms)	2,018	2,040	1,727	2,112	3,338	4,079	3,920	5,306	9,102	9,548	12,0314	
Rtx Ort. Kuyrukta Gecikme (ms)	0	0	0	0	0	0	0	0	0	0	0	
Toplam Kuyrukta Gecikme (s)	8,758	24,317	32,080	54,463	104,607	148,733	150,207	231,870	416,831	437,113	550,4244204	
1.grup Ort. IPDV	1,2524025	1,3585566	1,1601109	1,3861428	2,0935241	2,485533	2,5492278	3,2059769	5,7230633	6,036812	6,9034375	
2.grup Ort. IPDV	1,4023109	1,3334941	1,0708036	1,329746	2,0466922	2,3290791	2,1966306	2,8437686	4,0147332	4,1921226	4,4559489	
3.grup Ort. IPDV	1,2743872	1,1983072	0,9891503	1,1370503	1,4670914	1,5807916	1,5420051	1,7105402	1,9571317	1,9847367	2,014404	
Rtx Ort. IPDV	0	0	0	0	0	0	0	0	0	0	0	
1.grup Max. IPDV	5,334	8,896	8,896	15,624	21,093	25,498	26,464	28,393	31,969	34,192	35,367	
2.grup Max. IPDV	4,893	9,169	8,734	17,719	24,507	28,394	28,784	29,511	34,115	38,224	38,263	
3.grup Max. IPDV	4,892	8,949	8,678	18,482	26,691	28,250	29,808	30,400	36,563	40,343	41,044	
Rtx Max. IPDV	0	0	0	0	0	0	0	0	0	0	0	
1.grup Min. IPDV	0,006	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,003	0,003	0	
2.grup Min. IPDV	0,000	0,003	0,000	0,000	0,000	0,000	0,002	0,000	0,000	0,000	0	
3.grup Min. IPDV	0,000	0,000	0,000	0,000	0,000	0,004	0,003	0,003	0,004	0,003	0,004	
Rtx Min. IPDV	0	0	0	0	0	0	0	0	0	0	0	
1.grup Max. Jitter	3,987	6,994	8,610	17,526	20,505	23,442	26,689	26,117	27,222	31,520	28,6398	
2.grup Max. Jitter	4,257	7,157	9,105	19,111	24,293	25,230	27,273	26,188	30,050	31,005	28,0294729	
3.grup Max. Jitter	4,298	7,852	9,054	18,197	25,179	26,134	27,213	25,936	29,183	32,392	29,5206492	
Rtx Max. Jitter	0	0	0	0	0	0	0	0	0	0	0	
1.grup Min. Jitter	-2,003	-2,008	-1,707	-2,160	-3,317	-4,092	-3,949	-5,292	-9,087	-9,501	-12,0182	
2.grup Min. Jitter	-2,008	-2,057	-1,727	-2,088	-3,333	-4,092	-3,918	-5,311	-9,125	-9,566	-12,0105271	
3.grup Min. Jitter	-2,018	-2,040	-1,727	-2,112	-3,338	-4,079	-3,920	-5,306	-9,102	-9,546	-12,0313508	
Rtx Min. Jitter	0	0	0	0	0	0	0	0	0	0	0	
Alıcıya ulaşan toplam paket sayısı	4345	11918	18587	25784	31361	36415	38280	43691	45770	45769	45766	

Çizelge 5.5. Parking lot topolojisinde PQ-SCTP protokolü için benzetim sonuçları

Protokol (120ms end-to-end delay)	PQ-SCTP protokolü için sonuçlar (parking lot.tcl)										
	10240	20480	30720	40960	51200	59392	61440	71680	81920	92160	102400
Simülasyon süresi	120s										
İşlem Gecikmesi	2ms										
Kuyruk Boyutu (Bayt)	10240	20480	30720	40960	51200	59392	61440	71680	81920	92160	102400
Transmission rate (Mb/s)	4										
1.grup Ort. Kuyrukta Gecikme (ms)	1,020	1,018	1,002	1,036	1,020	1,027	1,014	1,003	1,003	1,012	0,9935
2.grup Ort. Kuyrukta Gecikme (ms)	1,011	1,008	1,009	1,008	1,026	1,010	1,017	1,010	1,008	1,004	0,9939
3.grup Ort. Kuyrukta Gecikme (ms)	2,517	2,555	2,086	2,660	4,492	5,619	5,378	7,457	13,158	13,822	17,5445
Rtx Ort. Kuyrukta Gecikme (ms)	0	0	0	0	0	0	0	0	0	0	0
Toplam Kuyrukta Gecikme (s)	8,758	24,317	32,080	54,463	104,607	148,729	150,209	231,870	416,831	437,109	550,4440877
1.grup Ort. IPDV	0,6944523	0,6492923	0,6763056	0,6586931	0,6746607	0,6523998	0,6555509	0,6241537	0,538342	0,5337261	0,4961739
2.grup Ort. IPDV	0,6760124	0,6913506	0,6660252	0,6812356	0,7015484	0,7141235	0,7034275	0,7289973	0,7746919	0,7668429	0,7891746
3.grup Ort. IPDV	1,6429399	1,5932581	1,3338939	1,5143895	1,8493067	1,9504414	1,9110575	2,0522332	2,2811133	2,2967277	2,339139
Rtx Ort. IPDV	0	0	0	0	0	0	0	0	0	0	0
1.grup Max. IPDV	1,891	1,994	1,947	1,989	1,950	1,962	1,994	1,993	1,994	1,993	1,994
2.grup Max. IPDV	1,891	1,951	1,968	1,993	1,974	1,994	1,952	1,994	1,968	1,994	1,994
3.grup Max. IPDV	5,668	8,949	9,852	20,482	26,615	28,224	31,674	32,400	38,563	42,343	41,044
Rtx Max. IPDV	0	0	0	0	0	0	0	0	0	0	0
1.grup Min. IPDV	0,006	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0
2.grup Min. IPDV	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0
3.grup Min. IPDV	0,003	0,000	0,003	0,000	0,000	0,002	0,004	0,003	0,005	0,003	0,018
Rtx Min. IPDV	0	0	0	0	0	0	0	0	0	0	0
1.grup Max. Jitter	0,980	0,982	0,998	0,964	0,980	0,974	0,987	0,997	0,997	0,988	1,0064952
2.grup Max. Jitter	0,988	0,992	0,991	0,992	0,974	0,990	0,983	0,990	0,992	0,996	1,0061096
3.grup Max. Jitter	4,866	8,599	11,858	21,649	24,246	24,858	29,050	27,775	32,035	29,382	28,2894816
Rtx Max. Jitter	0	0	0	0	0	0	0	0	0	0	0
1.grup Min. Jitter	-1,018	-1,018	-1,000	-1,036	-1,018	-1,026	-1,013	-1,003	-1,001	-1,010	-0,9935048
2.grup Min. Jitter	-1,004	-1,008	-1,009	-1,008	-1,026	-1,010	-1,017	-1,010	-1,008	-1,004	-0,9938904
3.grup Min. Jitter	-2,517	-2,555	-2,086	-2,660	-4,492	-5,619	-5,378	-7,457	-13,157	-13,820	-17,5445184
Rtx Min. Jitter	0	0	0	0	0	0	0	0	0	0	0
Alıcıya ulaşan toplam paket sayısı	4345	11918	18587	25784	31361	36415	38280	43691	45770	45769	45766



Şekil 5.21. Parking lot topolojisinde kuyruk boyutuna bağlı toplam kuyruk gecikmesi

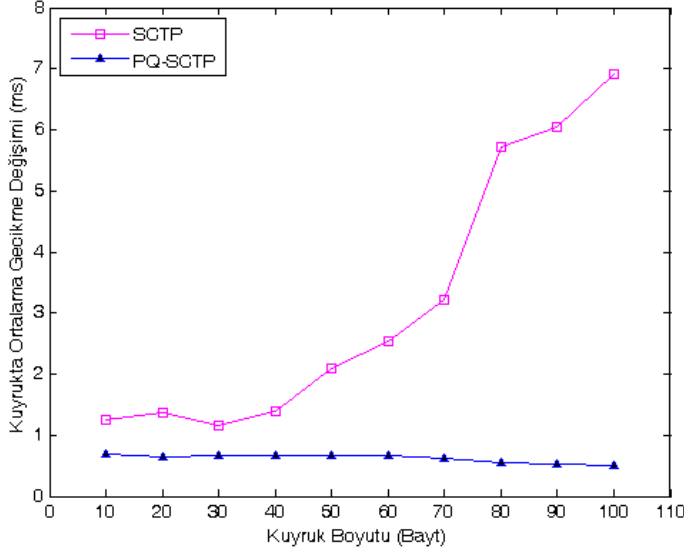
Protokoller için toplam gecikme süreleri Çizelge 5.6.'de sunulmuştur.

Çizelge 5.6. Parking lot topolojisinde toplam kuyruk gecikmeleri

Kuyruk Boyutu	SCTP	PQ-SCTP
	Kuyrukta toplam gecikme (s)	Kuyrukta toplam gecikme (s)
10KB	8,758	8,758
20KB	24,317	24,317
30KB	32,080	32,080
40KB	54,463	54,463
50KB	104,607	104,607
60KB	150,207	150,209
70KB	231,870	231,870
80KB	416,831	416,831
90KB	437,113	437,109
100KB	550,424	550,444

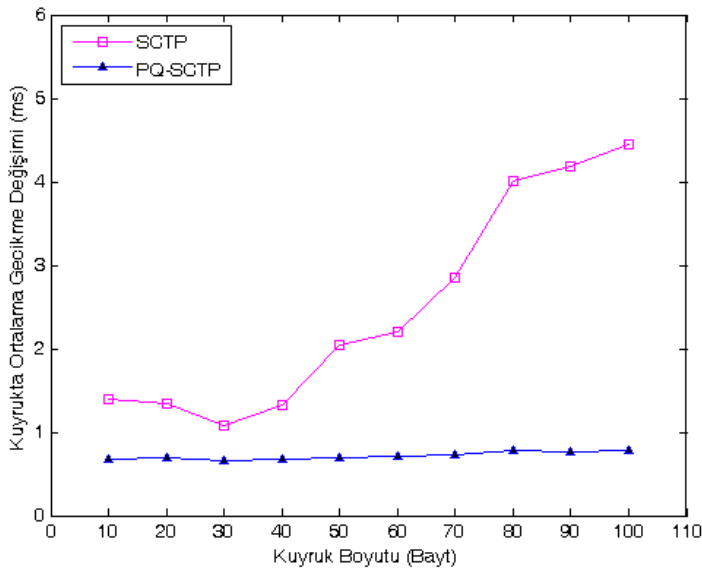
Yukarıda gösterilen çizelge ve grafikten de görüldüğü gibi PQ-SCTP ve SCTP protokolleri aynı kuyruk boyutlarında toplam kuyruk gecikmesi için aynı değerlere sahip olmaktadır.

Kuyrukta bekleme dışında topoloji için veri parçaları arası ortalama gecikme değişimi (IPDV) değerleri de elde edilmiştir. Eş. 5.2 ile kuyruktaki ortalama gecikme değişimi hesaplanmaktadır. 1.öncelik grubu için ortalama gecikme değişimi Şekil 5.22’de verilmiştir.



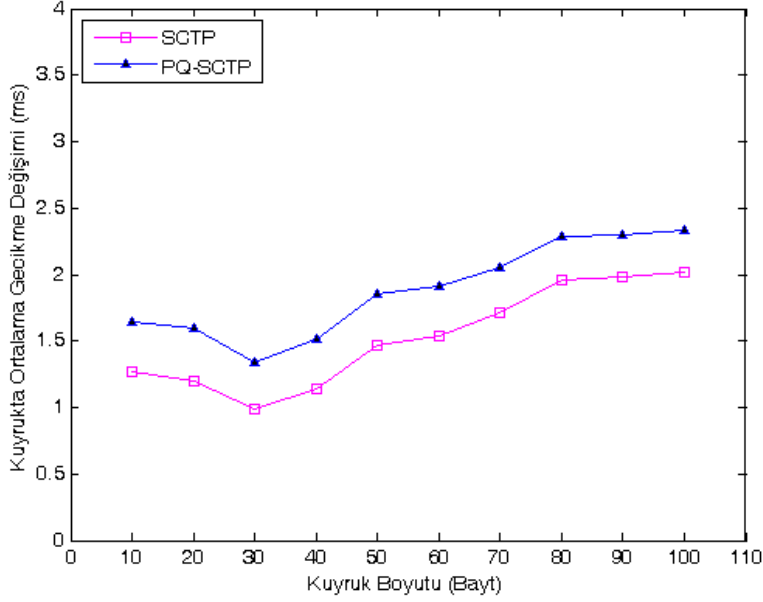
Şekil 5.22. Parking lot topolojisinde 1.öncelik grubu için kuyruk gecikmesi değişimi

Benzer şekilde 2.öncelik grubu için ortalama gecikme değişimi Şekil 5.23’de verilmiştir.



Şekil 5.23. Parking lot topolojisinde 2.öncelik grubu için kuyruk gecikmesi değişimi

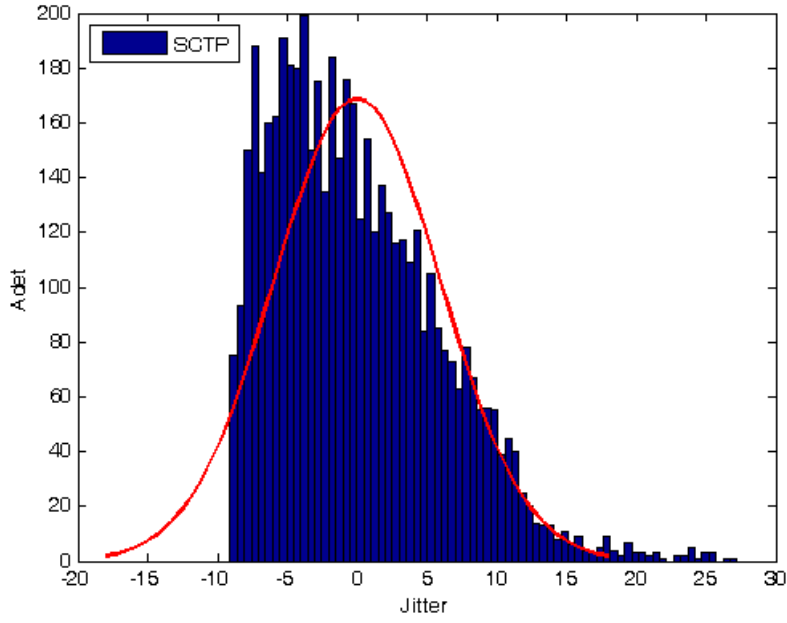
3.öncelik grubu için hesaplanmış ortalama gecikme değişimi ise Şekil 5.24’de görülmektedir.



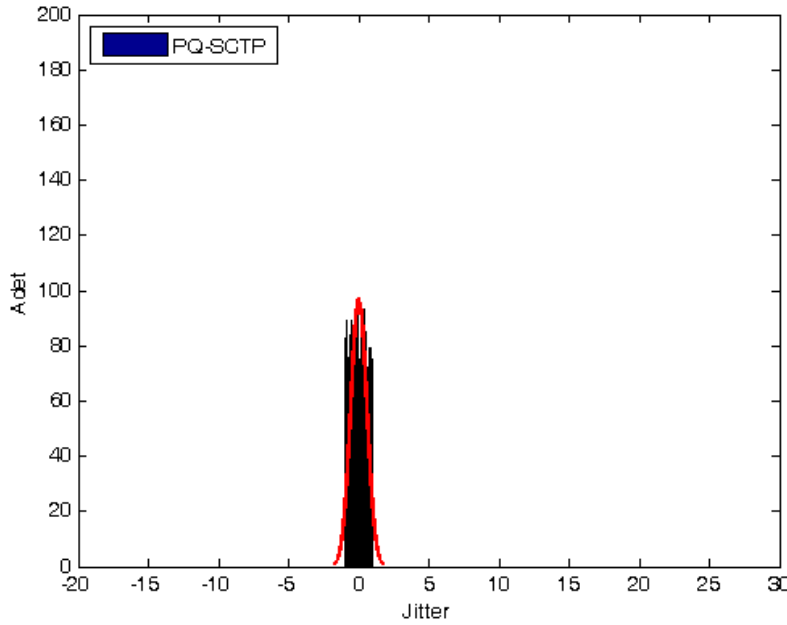
Şekil 5.24. Parking lot topolojisinde 3.öncelik grubu için kuyruk gecikmesi değişimi

Yukarıda gösterilen grafiklerden görüldüğü gibi 1. ve 2. öncelik gruplarında SCTP protokolünde sürekli yükselme olmasına rağmen PQ-SCTP protokolünde değerlerde önemli bir değişiklik söz konusu değildir. 3.öncelik grubu için ise hem SCTP, hem de PQ_SCTP protokolleri benzer şekilde artış sağlamasına rağmen PQ-SCTP protokolü daha yüksek değerlere çıkmaktadır. Özellikle 3.öncelik grubu için değişimin bu şekilde olması ortalama kuyruk gecikmesi değerlerinde olduğu gibi değerlerin doğruluğunu göstermektedir.

Deneyle sırasında dikkate alınan üçüncü önemli ölçüt ise jitterin dağılımı olmuştur. Jitter değerleri Eş. 5.4 ile hesaplanmıştır. SCTP ve PQ-SCTP protokolleri için 1.öncelik grubunda jitterin dağılımı Şekil 5.25’de sunulmuştur.



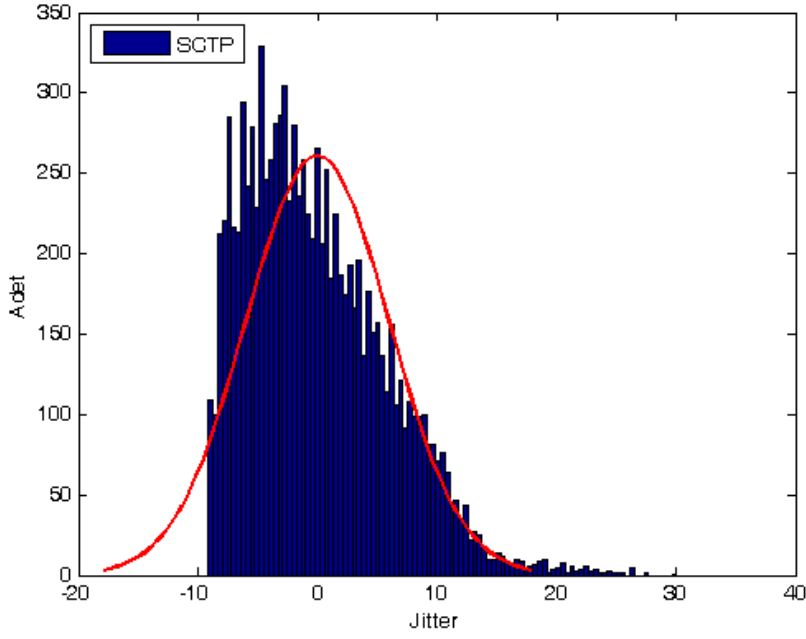
a



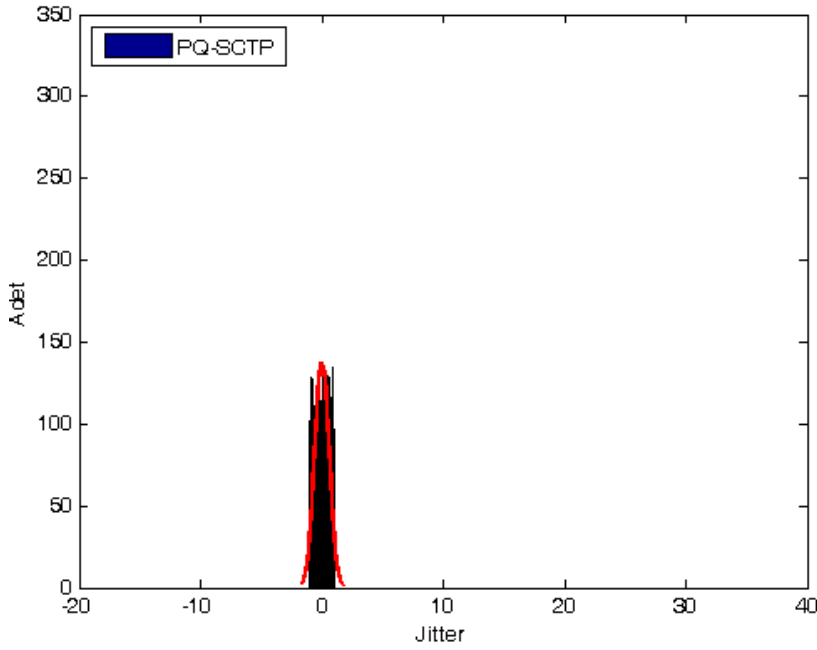
b

Şekil 5.25. Parking lot topolojisinde 1.öncelik grubu için jitter dağılımı
 a. Sctp protokolü için jitter dağılımı b. PQ-Sctp protokolü için jitter dağılımı

Benzer şekilde 2.öncelik grubu için jitter dağılımı da Şekil 5.26’de verilmiştir.



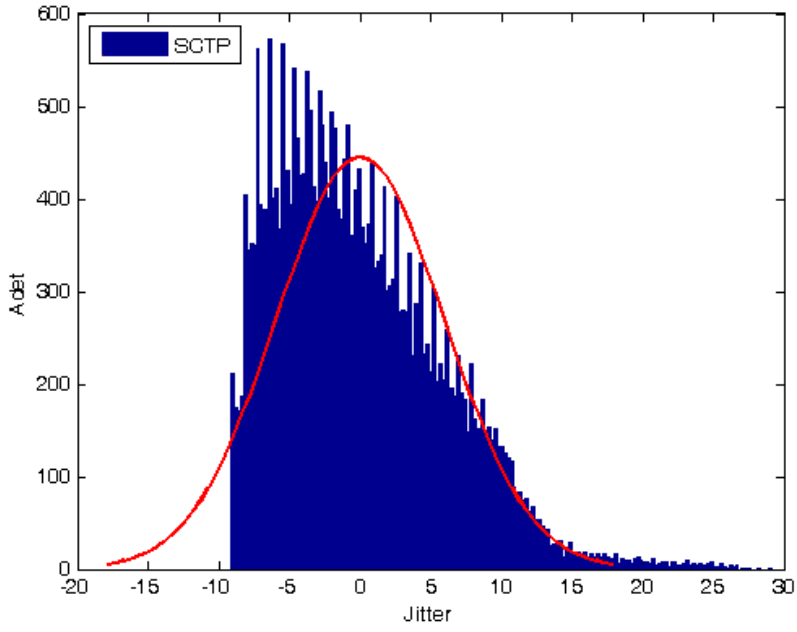
a



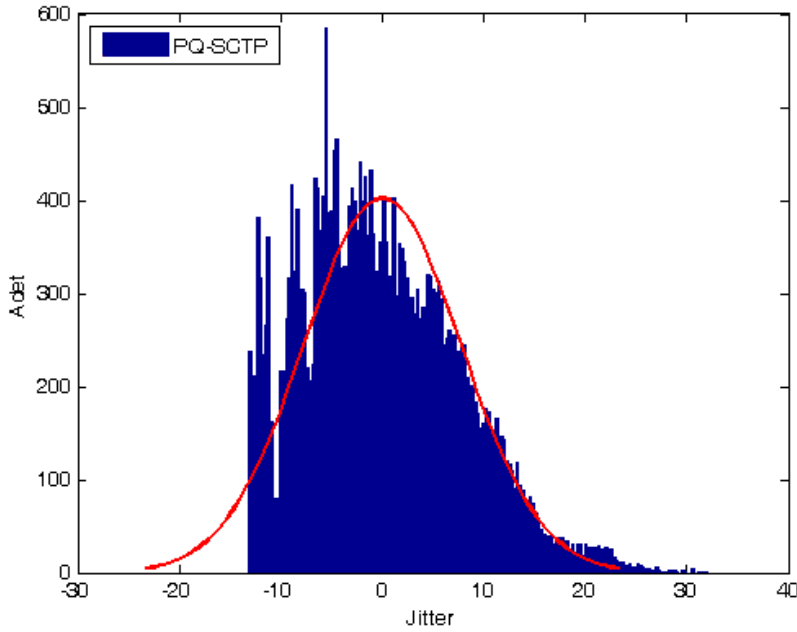
b

Şekil 5.26. Parking lot topolojisinde 2.öncelik grubu için jitter dağılımı
 a. SCTP protokolü için jitter dağılımı b. PQ-SCTP protokolü için jitter dağılımı

3.öncelik grubu için elde edilen jitter dağılımları ise Şekil 5.27’de görülmektedir.



a



b

Şekil 5.27. Parking lot topolojisinde 3.öncelik grubu için jitter dağılımı
 a. SCTP protokolü için jitter dağılımı b. PQ-SCTP protokolü için jitter dağılımı

Yukarıdaki dağılımlardan görüldüğü gibi 1.öncelik grubu için SCTP protokolünde jitter -10ms ile 15ms arası değerler sahipken PQ-SCTP protokolünde bu değerler -

1ms ile 1ms arasındadır. 2.öncelik grubu için de aynı değerler söz konusudur. Ancak 1.ve 2.grublardan farklı olarak 3.öncelik grubunda SCTP protokolündeki dağılım diğer gruptaki dağılımla benzerlik göstermesine rağmen PQ_SCTP protokolündeki dağılımda üst değerler 20ms'e kadar çıkmaktadır.

6. SONUÇLAR

Bu tezde IPTV uygulamalarında kullanılmak üzere ulaşım katmanında çalışan kuyruk önceliklendirmeli yeni bir protokol (Priority Queue – Stream Control Transmission Protocol - PQ-SCTP) geliştirilmiştir. Bu protokol ns-2 benzetim ortamında C++ programlama dili ile geliştirilmiş ve iki farklı topoloji üzerinde kuyruk gecikmesi, gecikme değişimi, throughput, jitter gibi performans ölçütleri göz önünde bulundurularak test edilmiştir. Protokolde gönderilen verilere öncelik tanımlanarak daha önemli bilgilerin alıcıda daha hızlı bir şekilde işleme alınmasıyla hizmet kalitesinde iyileştirme yapılmıştır.

Protokol gönderici tarafında Payload Data veri parçalarına öncelik eklemekte, alıcı tarafında ise bu önceliğe göre kuyruk sıralaması yapmaktadır. Yapılan çalışmada 4 farklı öncelik grubu belirlenmiş ve sonuçlar grup bazında değerlendirilmiştir.

Yapılan deneysel çalışmalarda hem SCTP, hem de PQ-SCTP protokolleri için sonuçlar elde edilmiş ve bu sonuçlar karşılaştırılmıştır. Elde edilen sonuçlar 1. ve 2. öncelik grubuna ait Payload Data türündeki veri parçalarında ortalama kuyruk gecikmesi, kuyruk gecikmesi değişimi, jitter dağılımı gibi performans ölçütlerinde PQ-SCTP protokolünün daha başarılı olduğunu göstermiştir.

KAYNAKLAR

1. Xiao, Y., Du, X., Zhang, J., Hu, F. And Guizani, S., "Internet Protocol Television (IPTV): The Killer Application for the Next-Generation Internet", *IEEE Communications Magazine*, 126-134 (2007).
2. Kim, S. T., Koh, S. J. And Kim, Y. J., "Performance of SCTP for IPTV Applications", *Advanced Communication Technology, The 9th International Conference*, Gangwon-Do, 2176 - 2180 (2007).
3. Boussen, S., Tabbane, N., Arnaud, J. And Krief, F., "Using PR-SCTP for IPTV QoS Adaptation over IMS Network", *Wireless and Mobile Networking Conference*, Toulouse, 1-7 (2011).
4. Stewart, R., Tuxen, M. And Lei, P., "SCTP: What is it, and how to use it?", *Cisco Systems*, 1-10 (2008).
5. Held, G., "Understanding Iptv", *Auerbach Publications*, ABD, 76-80 (2007).
6. Arberg, P., Cagenius, T., Tidblad, O. V., Ullerstig, M. And Winterbottom, P., "Network infrastructure for IPTV", *Ericsson Review*, 79-83 (2007).
7. Jeong S. And Hwang, C., "Mobile IPTV Expanding The Value Of IPTV", Samsung Electronics, Digital Media R&D Center, *Seventh International Conference On Networking*, 296-301 (2008).
8. Baba, M., Kurokawa, H. And Kato, Y., "Buffer-based Low-Delay Playout Control Methods for IPTV Terminals", *Information Technology R&D Center, Mitsubishi Electric Corporation, IEEE*, (2009).
9. Szymanski, T. H. And Gilbert, D., "Internet Multicasting of IPTV With Essentially-Zero Delay Jitter", *IEEE Transactions On Broadcasting*, 20-30 (2009).
10. Rong, B., Qian, Y., Guiagoussou, M. H. And Kadoch, M., "Improving Delay and Jitter Performance in Wireless Mesh Networks for Mobile IPTV Services", *IEEE Transactions On Broadcasting*, 642-651 (2009).
11. Balk, A., Sigler, M., Gerla M. And Sanadidi, M.Y., "Investigation of MPEG-4 Video Streaming over SCTP", *Proc. World Multiconference on Systemics, Cybernetics, and Informatics*, (2002).
12. El Derini, M. N. And Elshikh, A. A., "MPEG-4 Video Transfer with SCTP-Friendly Rate Control", *The Second International Conference on Innovations in Information Technology (IIT'05)*, Dubai, 1-11 (2005).

13. Kim, K. H., Jeong, K. M., Kang, C. H. And Seok, S. J., “A transmission control SCTP for real-time multimedia streaming”, *Computer Networks*, 54: 1418–1425 (2010).
14. Chae-Sub, L., “IPTV over Next Generation Networks, Broadband Convergence Networks”, *2nd IEEE/IFIB International Workshop on, Genava*, 1-18 (2007).
15. Cantekinler, M. K., Bolat, A., Çetin, T., Güçlü, T., Çaycı, A. D. And Yılmaz, R., “IPTV Tabanlı Hizmetler: VoIP ve IPTV”, *Telekomünikasyon Kurumu, Sektörel Araştırma Ve Stratejiler Dairesi Başkanlığı*, Ankara, (2008).
16. SVS Telekom & Satellite Systems, “IPTV”, 1-7 (2007).
17. Song, J., Jang, T. And Sohn, S. Y., “Conjoint Analysis For IPTV Service”, Yonsei University, *Expert Systems With Applications*, 7860-7864 (2009).
18. Ahmad, K. And Begen, A. C., “IPTV and Video Networks in the 2015 Timeframe: The Evolution to medianets”, Cisco, *IEEE Communications Magazine*, 68-74 (2009).
19. Yükselten, H., “SIP Tabanlı Telefon Servislerinin IPTV Sistemine Uyarlanabilirliği”, Yüksek Lisans Tezi, *Uludağ Üniversitesi Fen Bilimleri Enstitüsü*, Bursa, (2010).
20. Rahmanian, S., “IPTV Network Infrastructure”, *Huawei Technologies Co. Ltd.*, 1-32 (2008).
21. Çiftçi, G. T., “Uzaktan Eğitimde IPTV’nin Kullanılabilirliğine İlişkin Bir Delphi Çalışması”, Yüksek Lisans Tezi, *Anadolu Üniversitesi Sosyal Bilimler Enstitüsü*, Eskişehir, (2011).
22. Shihab, E., Cai, L., Wan, F. And Gulliver, A., “Wireless Mesh Networks for In-Home IPTV Distribution”, University of Victoria, *IEEE Network*, 52-57 (2008).
23. Han, S., Lisle, S. And Nehib, G., “IPTV Transport Architecture Alternatives And Economic Considerations”, Fujitsu Network Communications, *IEEE Communications Magazine*, 70-77 (2008).
24. Shihab, E. And Cai, L., “IPTV Distribution Technologies in Broadband Home Networks”, *University of Victoria IEEE*, 765-768 (2007).
25. Pınarbaşı, M., “IPTV Ağ Dağıtım Teknolojileri, İçerik Taşıma Mimarisi Ve Çok Noktaya Yayın”, Yüksek Lisans Tezi, *İstanbul Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, (2009).

26. Mahmoudian, E., “Wimax Üzerinden İnternet Protokol Televizyon Ve Servis Kalitesi – Qos”, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, (2010).
27. Stewart, R., “Stream Control Transmission Protocol”, *RFC 4960 Internet Engineering Task Force*, 1-153 (2007).
28. Stewart, R., Xie, Q., Tuexen, M., Maruyama, S. And Kozuka, M., “Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration”, *RFC 5061 Internet Engineering Task Force*, 3-5 (2007).
29. Stewart, R., Ramalho, M., Xie, Q., Tuexen, M. And Conrad, P., “Stream Control Transmission Protocol (SCTP) Partial Reliability Extension”, *RFC 3758 Internet Engineering Task Force*, 1-22 (2004).
30. Forouzan, B. A., “Data Communications and Networking 4th ed.”, *Mc Graw Hill*, Singapore, 913-915 (2007).
31. Demichelis, C. And Chimento, P., “IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)”, *RFC 3393 Internet Engineering Task Force*, 1-5 (2002).
32. Şimşek, M., “Paket Anahtarlama Ağılarda Hizmet Kalitesi Tabanlı Ulaştırma Protokolü Ve Kuyruk Yapısı Geliştirilmesi Ve Uygulaması”, Doktora Tezi, *Gazi Üniversitesi Bilişim Enstitüsü*, Ankara, 1-16 (2012).
33. Aydın, I., “SCTP-Based Concurrent Multipath Transfer in The Contexts Of Multihop wireless Networks And Tcp-Friendliness”, Doktora Tezi, *University of Delaware*, Delaware, 91-96 (2010).
34. İnternet: The Network Simulator – ns-2. <http://www.isi.edu/nsnam/ns>
35. The VINT Project, “The ns Manual”, K. Fall and K. Varadhan, *UC Berkeley*, 302-311 (2011).
36. Issariyakul, T., Hossain, E., “Introduction to Network Simulator NS2”, *Springer*, New York, 405-411 (2009).

EKLER

EK-1. Kayıt dosyası örneği

e -t 0.212245 -p 0 -src 0 -dst 7 -tsn 1
e -t 0.212986 -p 0 -src 1 -dst 8 -tsn 1
e -t 0.213726 -p 0 -src 2 -dst 9 -tsn 1
s -t 0.214245 -p 0 -src 0 -dst 7 -tsn 1
e -t 0.214466 -p 0 -src 3 -dst 10 -tsn 1
s -t 0.214986 -p 0 -src 1 -dst 8 -tsn 1
e -t 0.215206 -p 0 -src 4 -dst 11 -tsn 1
s -t 0.215726 -p 0 -src 2 -dst 9 -tsn 1
e -t 0.215947 -p 0 -src 5 -dst 12 -tsn 1
s -t 0.216466 -p 0 -src 3 -dst 10 -tsn 1
e -t 0.216687 -p 0 -src 6 -dst 13 -tsn 1
s -t 0.217206 -p 0 -src 4 -dst 11 -tsn 1
e -t 0.217427 -p 128 -src 0 -dst 7 -tsn 2
s -t 0.217947 -p 0 -src 5 -dst 12 -tsn 1
e -t 0.218167 -p 128 -src 1 -dst 8 -tsn 2
s -t 0.218245 -p 128 -src 0 -dst 7 -tsn 2
s -t 0.218687 -p 0 -src 6 -dst 13 -tsn 1
e -t 0.218908 -p 128 -src 2 -dst 9 -tsn 2
s -t 0.218986 -p 128 -src 1 -dst 8 -tsn 2
e -t 0.219648 -p 128 -src 3 -dst 10 -tsn 2
s -t 0.219726 -p 128 -src 2 -dst 9 -tsn 2
e -t 0.220388 -p 128 -src 4 -dst 11 -tsn 2
s -t 0.220466 -p 128 -src 3 -dst 10 -tsn 2
e -t 0.221129 -p 128 -src 5 -dst 12 -tsn 2
s -t 0.221206 -p 128 -src 4 -dst 11 -tsn 2
e -t 0.221869 -p 128 -src 6 -dst 13 -tsn 2
s -t 0.221947 -p 128 -src 5 -dst 12 -tsn 2
e -t 0.222609 -p 128 -src 0 -dst 7 -tsn 3

EK-2. Dumbbell.tcl OTCL script kodu

```

#-----Dumbbell Toplojisi-----#

set ns [new Simulator]

set nf [open out.nam w]

$ns namtrace-all $nf

    set allchan [open all.tr w]

    $ns trace-all $allchan

$ns color 1 Blue
$ns color 2 Red

proc finish {} {
    global ns nf

    $ns flush-trace

    close $nf

    #exec nam out.nam &

    exit 0
}

for {set i 1} {$i < 8} {incr i} {
    set s($i) [$ns node]
}

for {set i 1} {$i < 8} {incr i} {
    set r($i) [$ns node]
}

for {set i 1} {$i < 3} {incr i} {
    set n($i) [$ns node]
}

#link tanımlama

$ns duplex-link $s(1) $n(1) 20Mb 10ms DropTail
$ns duplex-link $n(2) $r(1) 20Mb 10ms DropTail

for {set i 2} {$i < 8} {incr i} {
    $ns duplex-link $s($i) $n(1) 20Mb 10ms DropTail
}

```

EK-2. (Devam) Dumbbell.tcl OTCL script kodu

```

for {set i 2} {$i < 8} {incr i} {
    $ns duplex-link $r($i) $n(2) 20Mb 10ms DropTail
}

$ns duplex-link $n(1) $n(2) 15Mb 10ms DropTail
#$ns queue-limit $s(1) $n(1) 5000
#link tanımlama son

for {set i 1} {$i < 15} {incr i} {
    set sctp($i) [new Agent/SCTP]
    $sctp($i) set debugMask_ 0x00303000
    $sctp($i) set debugFileIndex_ 0
    $sctp($i) set class_ 1
}

for {set i 7} {$i < 15} {incr i} {
    $sctp($i) set initialRwnd_ 51200
    $sctp($i) set recvAppLayerDelay_ 0.002
}

$sctp(1) set class_ 2
for {set i 1} {$i < 8} {incr i} {
    $ns attach-agent $s($i) $sctp($i)
    $ns attach-agent $r($i) $sctp([expr ($i + 7)])
}

set cbr(1) [new Application/Traffic/CBR]
$cbr(1) set packetSize_ 1310
$cbr(1) set rate_ 4Mb
$cbr(1) attach-agent $sctp(1)

for {set i 2} {$i < 8} {incr i} {
    set cbr($i) [new Application/Traffic/CBR]
    $cbr($i) set packetSize_ 1310
    $cbr($i) set rate_ 0.5Mb
}

```


EK-2. (Devam) Dumbbell.tcl OTCL script kodu

```
    $cbr($i) attach-agent $sctp($i)
}
for {set i 1} {$i < 8} {incr i} {
    $ns connect $sctp($i) $sctp([expr ($i + 7)])
}
for {set i 1} {$i < 8} {incr i} {
    $ns at 0.0 "$cbr($i) start"
    $ns at 120.0 "$cbr($i) stop"
}
$ns at 120.0 "finish"
$ns run
```

EK-3. Parkinglot.tcl OTCL script kodu

```

#-----Parking Lot Toplojisi-----#

set ns [new Simulator]

set nf [open out.nam w]

$ns namtrace-all $nf

    set allchan [open all.tr w]
    $ns trace-all $allchan

$ns color 1 Blue
$ns color 2 Red

proc finish {} {
    global ns nf

    $ns flush-trace

    close $nf

    #exec nam out.nam &

    exit 0
}

for {set i 1} {$i < 16} {incr i} {
    set n($i) [$ns node]
}

#link tanımlama

#önce backbone linki tanımlayalım

$ns duplex-link $n(9) $n(10) 10Mb 10ms DropTail
$ns duplex-link $n(10) $n(11) 10Mb 10ms DropTail
$ns duplex-link $n(11) $n(12) 10Mb 10ms DropTail
$ns duplex-link $n(12) $n(13) 10Mb 10ms DropTail
$ns duplex-link $n(13) $n(14) 10Mb 10ms DropTail
$ns duplex-link $n(14) $n(15) 10Mb 10ms DropTail

```

EK-3. (Devam) Parkinglot.tcl OTCL script kodu

```

#kenar linkler

$ns duplex-link $n(1) $n(10) 20Mb 10ms DropTail
$ns duplex-link $n(2) $n(11) 20Mb 10ms DropTail
$ns duplex-link $n(5) $n(11) 20Mb 10ms DropTail
$ns duplex-link $n(3) $n(12) 20Mb 10ms DropTail
$ns duplex-link $n(6) $n(12) 20Mb 10ms DropTail
$ns duplex-link $n(4) $n(13) 20Mb 10ms DropTail
$ns duplex-link $n(7) $n(13) 20Mb 10ms DropTail
$ns duplex-link $n(8) $n(14) 20Mb 10ms DropTail

#link tanımlama son

for {set i 1} {$i < 11} {incr i} {
    set sctp($i) [new Agent/SCTP]
    $sctp($i) set debugMask_ 0x00303000
    $sctp($i) set debugFileIndex_ 0
    $sctp($i) set class_ 1
    $sctp($i) set initialRwnd_ 133120
    $sctp($i) set recvAppLayerDelay_ 0.002
}

$sctp(9) set class_ 2
for {set i 1} {$i < 10} {incr i} {
    $ns attach-agent $n($i) $sctp($i)
}

$ns attach-agent $n(15) $sctp(10)
    set cbr(9) [new Application/Traffic/CBR]
    $cbr(9) set packetSize_ 1310
    $cbr(9) set rate_ 4Mb
    $cbr(9) attach-agent $sctp(9)

```

EK-3. (Devam) Parkinglot.tcl OTCL script kodu

```

for {set i 1} {$i < 5} {incr i} {
    set cbr($i) [new Application/Traffic/CBR]
    $cbr($i) set packetSize_ 1310
    $cbr($i) set rate_ .5Mb
    $cbr($i) attach-agent $sctp($i)
}

$ns connect $sctp(9) $sctp(10)
$ns connect $sctp(1) $sctp(6)
$ns connect $sctp(2) $sctp(7)
$ns connect $sctp(3) $sctp(8)
$ns connect $sctp(4) $sctp(5)
#for {set i 1} {$i < 5} {incr i} {
#    $ns connect $sctp($i) $sctp([expr ($i + 4)])
#}

    $ns at 0.0 "$cbr(9) start"
    $ns at 120.0 "$cbr(9) stop"
for {set i 1} {$i < 5} {incr i} {
    $ns at 0.0 "$cbr($i) start"
    $ns at 120.0 "$cbr($i) stop"
}
$ns at 120.0 "finish"
$ns run

```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : HÜSEYNLİ, Alisettar
Uyruğu : Azerbaycan
Doğum tarihi ve yeri : 14.02.1989 Bakü
Medeni hali : Bekâr
Telefon : 0 (507) 854 44 28
e-mail : alisettar1@hotmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Lisans	Gazi Üniversitesi/ Bilgisayar Mühendisliği	2010
Lise	Bakü Türk Lisesi	2006

Yabancı Dil

Türkçe, İngilizce, Rusça

Hobiler

Kitap okumak, Bisiklet kullanmak