T.C
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GERÇEK ZAMANLI MEDYA AKIŞI İŞLEMİNİN YÜRÜTÜLMESİ İÇİN WEB ORTAMI GELİŞTİRİLMESİ

YÜKSEK LİSANS TEZİ

Mehmet GÜL

(06229104)

Anabilim Dalı: Bilgisayar Mühendisliği

Programı:

Tez Danışmanı: Yrd. Doç. Dr. Ahmet ÇINAR

Tezin Enstitüye Verildiği Tarih:

MART-2010

T.C
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GERÇEK ZAMANLI MEDYA AKIŞI İŞLEMİNİN YÜRÜTÜLMESİ İÇİN WEB ORTAMI GELİŞTİRİLMESİ

YÜKSEK LİSANS TEZİ

Mehmet GÜL

(06229104)

Tezin Enstitüye Verildiği Tarih:

Tezin Savunulduğu Tarih:

Tez Danışmanı: **Yrd. Doç. Dr. Ahmet ÇINAR**
Diğer Jüri Üyeleri: **Yrd. Doç. Dr. A. Bedri ÖZER**
Doç. Dr. İbrahim TÜRKOĞLU

MART -2010

İÇİNDEKİLER

Sayfa No

İÇİNDEKİLER II

ÖNSÖZ VI

ÖZET. VII

SUMMARY VIII

ŞEKİLLERİN LİSTESİ ..... …IX

TABLOLARIN LİSTESİ XI

SEMBOLLERİN LİSTESİ XII

1. GİRİŞ 1

1.1. Tezin Amacı **6**

2. ORTAM SIKIŞTIRMALARINDA KULLANILAN METOTLAR 8

2.1. Geçici Referans Numarası Uygulaması **9**

2.2. Sahne Geçişi Belirleme Uygulaması **12**

2.3. Ortam Akışlarında Renk Transferi Metodu **13**

2.4. Fazlalık Bilgilerin Atılması **15**

2.4.1. Uzaysal fazlalık **16**

2.4.2. Spektral fazlalık **16**

2.4.3. Anlık fazlalık **16**

2.4.4. Algılanabilen fazlalık **16**

2.5. Tıkanıklık Kontrol Mekanizması **18**

2.6. Ortam Akışına Yardımcı Yaklaşımlar **19**

2.7. Kaliteye Duyarlı Çoklu-Ortam Dosyaları İçin Öncelikli Akış İşlemi **20**

3. ORTAM AKIŞINDAKİ GÜVENLİK DUVARININ ETKİSİ 24

3.1. Güvenlik Duvarları ve Protokoller **25**

Sayfa No

3.2. Güvenlik Duvarı ve Ortam Akışı **25**

3.3. Güvenlik Duvarı ve Çoklu-Ortamlar **26**

3.4. Güvenlik Duvarı ve Paket Koklanması **27**

4. TCP ÜZERİNDEN ORTAM AKIŞLARI 29

4.1. TCP ve Bant-Genişliği **30**

4.2. TCP ve UDP Protokollerinin Beraber Kullanımı **32**

4.3. Çoklu TCP Bağlantısı Kullanımı **34**

4.4. TCP Protokolü Üzerinden Çoklu-Ortamların Akışının Yapılması **37**

4.4.1. TCP ile SCP protokolleri arasındaki benzerlikler **40**

4.4.2. TCP ile SCP protokolleri arasındaki faklılıklar **41**

4.5. TCP'deki Çoklu-Ortam Tıkanıklık Kontrol Protokolü **41**

5. KABLOSUZ AĞ SİSTEMLERİNDE GERÇEKLEŞEN ORTAM AKIŞLARI 44

5.1. Kablosuz Ağlarda Gerçek Zamanlı Ortam Akışları **45**

5.1.1. Sistem mimarisi **45**

5.2. MAC Standardı **48**

5.3. Kablosuz Ağ Üzerinden Güvenli Ölçeklenebilir Ortam Akışı **51**

5.4. Kablosuz Ev Alan Ağında Çoklu-Ortamların Kaliteli Akışı **55**

5.4.1. Kalite odaklı uyum şeması (QOAS) **55**

5.4.2. QOAS konumlaması **57**

5.5. Kablosuz Çok Kullanıcılı Çapraz-Katman Optimizasyonu **57**

5.5.1. Sistem mimarisi **58**

5.5.2. Parametre soyutlaması **58**

5.5.3. Radyo-Bağlantı katmanı **59**

Sayfa No

5.5.4. Uygulama katmanı **60**

5.6. Çapraz-Katman Optimizasyonunun Performans ve Maliyeti **61**

5.6.1. CLD optimizasyonu **62**

5.6.2. Hesaplama maliyeti **64**

5.6.3. İletişim maliyeti **66**

6. ORTAM AKIŞLARINDAKİ VEKİL SUNUCU UYGULAMALARI. 68

6.1. Çoklu-Ortam Vekil Sunuculara Genel Bir Bakış **69**

6.2. Geri Oynatımda Vekil Sunucuların İşlevleri **70**

6.3. Etkileşimli Ortam Uygulamaları için Vekil Seçimi **72**

6.4. Ortam Akışlarının Dağıtımı için Kod-Çevirimi Yakalama Stratejisi **74**

6.4.1. Kod-Çevirimi Yakalama (TeC) Sistemi Mimarisi **75**

6.4.2. Kod-Çevirimi Yakalama (TeC) Sistemi Kod-Çevirimi Teknikleri **76**

6.5. Popüler Ortam Dosyaları için Yapılan Periyodik Yayınlardaki Ön-Ek Yakalaması **77**

7. UYGULAMA 81

7.1. Kullanılan Yazılımlar **81**

7.1.1. Red5 ortam sunucusu **83**

7.1.2. Adobe Flash Ortam Sunucusu **84**

7.1.3. Red5 Ortam Sunucusu ile Adobe Flash Ortam Sunucusu arasındaki benzerlikler **84**

7.1.4. Red5 Ortam Sunucusu ile Adobe Flash Ortam Sunucusu arasındaki fark . **84**

7.1.5. Adobe Flash CS4 ..... **85**

7.1.6. Hazırlanan yazılım **85**

8. SONUÇ 91

Sayfa No

9. KAYNAKLAR 93

10. ÖZGEÇMİŞ 100

ÖNSÖZ

Bu tezin hazırlanmasındaki katkılarından ötürü danışman hocam Yrd. Doç. Dr. Ahmet ÇINAR hocama teşekkürü bir borç bilirim.

Ayrıca bu tezi her zaman benimle beraber olan ve hiçbir zaman maddi ya da manevi desteklerini esirgemeyen canım aileme itafen sunuyorum. Canımdan birer parça olan yeğenlerime; hiçbir zaman tatlı gülücüklerinin eksik olmaması dileğiyle…

Canım Annem…

ÖZET

Gerçek Zamanlı Medya Akışı İşleminin Yürütülmesi için Web Ortamı Geliştirilmesi

Web akışı internet üzerinde aşamalı bir şekilde ortam verilerinin transferinin yapılarak ortam dosyasını bilgisayarına indirmeden izlenmesi olarak tanımlanır. Ortam türleri tek bir formatta olmadığından, farklı dosya türleri için farklı yaklaşımların geliştirilmesi gerekmektedir. Ortam verilerinin farklı formatlarda olmalarının nedeni farklı sıkıştırma metotları sıkıştırılmış olmalarıdır. Bu sebeple ötürü ki ortam akışları için farklı metotların kullanılması zorunludur.

Ortam türlerinin İnternet ortamında farklı kullanım alanları bulunmaktadır. Örneğin canlı yayın, uyarlanmış canlı yayın ve istenildiğinde izlenebilen yayınlardır. Ortam türlerinin İnternet ortamında hızlı transferi için kullanılan en yaygın araç kod-çözücülerdir. Bu yazılımlar kullanıcı bilgisayarını İnternete bağlayan modeme benzetilebilir. Ortam akışlarının daha güvenli akışını sağlamak amacıyla genellikle açık kaynak yazılımlar tercih edilmektedir. Bazı ortam türleri için ise özel geliştirilen protokoller bulunmaktadır; örneğin MMS gibi. Fakat daha hızlı ortam akışının sağlanması için bu özel protokoller ya da UDP protokolü yeterli gelmediğinden yeni geliştirilen protokol çalışmalarına gerek duymaktadır. Örneğin veri transferinde kullanılan TCP protokolü üzerindeki çalışmalar v.b... gibi gelişen teknolojiyle beraber çeşitlenen kullanıcı istekleri ortam türlerinin de farklı yöntemler kullanılarak kullanıcılara gönderilmesini zorunlu hale getirmiştir.

Bu tezin amacı İnternet ortamında ortam dosya türlerinin akışı için kullanılan farklı metot ve yöntemlerinden en güncel olan çalışmaların tanıtılması ve yapılan farklı çalışmaların ortam türleri akışı üzerindeki etkilerinin incelenmesidir.

Anahtar Kelimeler: Ortam Türleri, Kod-Çözücüler, Ortam Sıkıştırma Metotları, TCP Protokolü, Çoklu Ortamlar, Kablosuz Ağ Yapısında Ortam Akışı, Vekil Sunucu, Çok-Kullanıcılı Ortam Akışı.

SUMMARY

Web Application Development for Real Time Media Flow Process Execution

Web streaming media on the internet in a phased manner to transfer data to the computer downloading the media file is defined as monitoring. Media types are not on a single format, the development of different approaches for different file types are required. Media data in different formats that are caused by different compression methods are being compressed. Because of this reason, the use of different methods for streaming media is required.

Media type fields are available for different uses in the Internet environment. For example, live broadcasts, live and on-demand streaming and broadcast. On the internet for fast transfer of media types most commonly used tools are codec. It can be compared with modem that hardware connects to the user's computer to the internet. Media stream in order to provide a more secure flow of open source software generally are preferred. For some media types are specially developed protocols such as MMS. But to ensure faster flow of this specific protocol environment or UDP protocol does not come just need to work a new protocol is developed. For example, data transfer over TCP protocol used in the studies, etc. emerging technologies such as the user requests vary with different types of media by using the methods to be sent to users has been made compulsory.

The purpose of this thesis on the internet streaming media file types, and different methods used to study the latest methods of promotion and media types of the different studies is to examine the impact on the flow.

Key Words: Media Types, Codec, Different Compression Methods, Multi-User Platforms, Streaming Media in Wireless Internet, Proxy Servers, Streaming Media in Peer-to-Peer Platforms.

ŞEKİLLERİN LİSTESİ

Sayfa No

Şekil 1.1. XML dokümanının filtreleme işlemi **5**
Şekil 2.1. Bozuk ortam görüntüsü ve kod çözücülerle işlenen görüntü **10**
Şekil 2.2. Bit kaydırma işlemi ile şifrelenmiş akım **11**
Şekil 2.3. Sahne değişiminin kare tipi yoluyla belirlenmesi **12**
Şekil 2.4. Şifreleyicinin ortam kareleri için blok diyagramı **14**
Şekil 2.5. Çözümleyicinin blok diyagramı **15**
Şekil 2.6. Dalgacık tabanlı sıkıştırmanın yapısı **17**
Şekil 2.7. Dalgacık ve DCT algoritmaları **18**
Şekil 2.8. Öncelik-işlem kontrolü **20**
Şekil 2.9. Öncelik-işlem zaman çizgisi **21**
Şekil 4.1. ÇokluTCP sistem diyagramı **31**
Şekil 4.2. OMTP iletişim modeli **32**
Şekil 4.3. OMTP **33**
Şekil 4.4. İşgücü indiriminin gösterimi **37**
Şekil 4.5. SCP transfer durum diagramı **39**
Şekil 5.1. Temin sunucusu ve bileşenleri **46**
Şekil 5.2. Ortam sunucusu ve alt sistemleri **47**
Şekil 5.3. Kart sürücüsü yapısı **49**
Şekil 5.4. VOIP trafik akışı **51**
Şekil 5.5. Geleneksel ortam akış yaklaşımları **52**
Şekil 5.6. Kod-çevirimlerinde geleneksel yaklaşımlar **53**
Şekil 5.7. SSS kolama işlemi **54**
Şekil 5.8. SSS kod-çevirimi işlemi **54**
Şekil 5.9. QOAS adaptasyon prensibi şematik gösterimi **56**
Şekil 5.10. Çapraz-katman optimizasyon mimarisi **58**
Şekil 5.11. Üç farklı ortam dosyasının GOP için kayıp kareler bozulma profili **60**

Sayfa No

Şekil 5.12. GOP boyutu bir fonksiyonu olarak RD yan bilgileri gönderen trafik yükü **66**
Şekil 6.1. Veri paketlerinin İnternet üzerinden vekil sunucular aracılığıyla gönderimi **68**
Şekil 6.2. İşbirlikçi yaklaşımla çalışan B ve C vekil sunucularının hem S sunucusuna hem de A vekil sunucusuna yaptıkları yardım **70**
Şekil 6.3. İnternet üzerinden gerçek zamanlı geri oynatım uygulamalarında sonlu yapı **71**
Şekil 6.4. Vekil yakalama yöntemi ile gerçekleştirilen ortam akışı **72**
Şekil 6.5. I-Vekil uygulaması **73**
Şekil 6.6. Kod-çevirimi yakalama vekilleri sistemi ve bileşenleri **76**
Şekil 6.7. İnternette içerik dağıtım ağı **78**
Şekil 6.8. Sunucu kanal sayısının ön-ek ölçüsü ile karşılaştırılması **80**
Şekil 6.9. Periyodik yayına yardımcı olan ön-ek yakalaması **80**
Şekil 7.1. Adobe Flash Player yazılımı indirme ana sayfası **82**
Şekil 7.2. Localhost'ta çalışan sayfanın ilk durumu **87**
Şekil 7.3. İnternette yayın yapılmadan önceki aşama **88**
Şekil 7.4. Yerel bilgisayardaki görüntü ile internet üzerindeki görüntü arasındaki zaman farkı **89**
Şekil 7.5. Bağlantı hızının tespiti **89**
Şekil 7.6. İnternet üzerinden yayının yapılması **90**

TABLOLARIN LİSTELENMESİ

Sayfa No

Tablo 4.1. SCP durumları **40**

Tablo 5.1. Test edilen ortam dizilerinin ana karakterleri..... .. **62**

Tablo 5.2. Üç kullanıcı arasında olası oran tahsisi durumları..... .. **63**

Tablo 5.3. CLO durum sayıları ve farklı kullanıcılar için normalize edilmiş işletim zamanı..... .. **65**

SEMBOLLERİN LİSTESİ

RMF : Rich Music Format
MIDI : Musical Instrument Digital Interface
MP3 : MPEG, audio layer 3
MPEG : Moving Picture Experts Group
JPEG : Joint Photographic Experts Group
GIF : Graphics Interchange Format
TIFF : Tagged Image File Format
HTTP : HyperText Transfer Protocol
RTMP : Real Time Messaging Protocol
PNM : People Near Me Protocol
MMS : Multimedia Message Service
RTSPU : RTSP using UDP is called RTSPU
RTSP : Real Time Streaming Protocol
RTSPT : RTSP using TCP is called RTSPT
MMSU : MMS using UDP is called MMSU
MMST : MMS using TCP is called MMST
IETF : Internet Engineering Task Force
VCR : VideoCassette Recorder
XML : Extensible Markup Language
SGML : Standard Generalized Markup Language
W3C : The World Wide Web Consortium
SDI : Serial Digital Interface
DTD : Document Type Definition
VB : Visual Basic
DWT : Discrete Wavelet Transform
DCT : Discrete Cosine Transform
DCCP : Datagram Congestion Control Protocol
ECN : Explicit Congestion Notifcation
FEC : Forward Error Correction

ADUs : Automatic Dialing Unit

UDP : User Datagram Protocol

TCP : Transmission Control Protocol

TFRC : TCP-Friendly Rate Control

RTT : Round-Trip Time

OMTP : Overlay Media Transport Protocol

MTU : Maximum Transmission Unit

SCP : Streaming Control Protocol

ACK : ACKnowledge

TFWC : TCP-Friendly Window-based Control

VoIP : Voice Over Internet Protocol

PDA : Personal Digital Assistant

LAN : Local Area Network

WLAN : Wireless Local Area Network

WMP : Windows Media Player

DCF : Distributed Coordination Function

QoS : Quality Of Service

DIFS : Distributed Interframe Space

FIFO : First-In, First-Out

MAC : Media Access Control

IEEE : Institute of Electrical and Electronics Engineers

AP : Access Point

MUX : Multi-User eXperience

CEBus : Consumer Electronics BUS

DTMS : Data Transformation Management System

LCD : Liquid Crystal Display

GPRS : General Packet Radio Service

SSS : Secure Scalable Streaming

QOAS : Quality-Oriented Adaptation Scheme

QoE : Quality of Experience

QoDGS : Quality of Delivery Grading Scheme

SAS : Server Arbitration Scheme

GOP : Group Of Pictures

MSE : Mean Squared reconstruction Error

CLD : Cross-Layer Design

KBPS : Kilobytes Per Second

ENDEF : Expected Number of Decodable Frames

PSNR : Peak Signal to Noise Ratio

CLO : Cross-Layer Optimizer

CDN : Content Distribution Network

IMS : IP Multimedia Subsytem

HoA : Home Address

CoA : Care-of-Address

MIP : Mobil IP

SIP : Session Initiation Protocol

VBR : Variable Bit Rate

ISP : Internet Service Provider

WAN : Wide-area Networks

ITU : International Telecommunication Union

ISDN : Integrated Services Digital Network

NBAR : Network Based Application Recognition

ToS : Type of Service

P2P : Peer-to-Peer

CEF : Cisco Express Fowarding

VOD : Video On Demand

PPM : Prediction by Partial Match

RAP : Rate Adaptation Protocol

EC : Error Control

QA : Quality Adaptation

RTO : Retransmission Timeout

ARQ : Automatic Repeat reQuest

TeC : Transcoding-enabled Caching

SCDN : Streaming Content Distribution Network

CBR : Constant-Bit-Rate

1. GİRİŞ

Web Akışı internet üzerinde aşamalı ortam verilerinin transferi için kullanılan bir tanımdır. Kullanıcının herhangi bir ortam dosyasını kendi bilgisayarına internet ortamından indirmeden izlemesi, dinlemesi ya da görmesi olarak ta tanımlanır [1]. Ortam dosya türleri olarak klip, video, ses ve resim gibi dosya türlerini tanımlayabiliriz. İnternet ortamında ortam akışlarının en büyük avantajı transfer edilmiş dosyanın tekrar oynatılabilmesidir. Çünkü istenilen veri artık kullanıcı bilgisayarının önbelleğinde saklı bulunur. Akış işlemleri için kimi zaman bir donanım parçası kullanılırken kimi zamanda bir yazılım kullanılır. Akış işlemlerinde kullanılan yazılımların bazıları lisanlı olabileceği gibi bazıları da açık kaynaklı yazılımlardır. Açık kaynak yazılımlarının en büyük avantajı geliştirilebilir olmalarıdır.

Açık kaynak yazılımlardan ortam akışları için en fazla kullanılanı kod-çözücülerdir. Kod-çözücüler sayesinde sunucu ortam dosyasını şifreleyip-kodlayıp, çözümleyebilir. Kod-çözücüler ilk defa bir donanım olarak kullanılmış ve analog sinyalleri kodlayıp çözümleme işleminde kullanılmıştır. Günümüzde ise donanımsal bir cihaz olarak kullanılmaktan çok bir yazılım olarak kullanılmaktadır. Bu yazılım ortam verilerini şifreleyerek bu verilerin hem internet hem de bilgisayar ortamındaki akışını düzenler. Kod-çözücüleri yaptığı işleve göre bilgisayarları İnternet ortamına bağlayan modemlere benzetebiliriz [2, 3]. Birçok kod-çözücü ortam verilerinin formatını değiştirmekle kalmaz İnternet üzerinden akışını hızlandırmak amacıyla boyutunu da düşürür. Böylece veri ana sunucu belleğinde daha az yer kaplar ve yer sorunu da ortadan kaldırılır.

İnternet üzerinden yapılan ortam akışlarına birkaç örnek vermek gerekirse; uzaktan eğitim uygulamaları, İnternet radyo, İnternet TV, video konferans uygulamaları, vb. Ortam türlerinin İnternet üzerinden izlenmesi için farklı protokoller, programlar, algoritmalar veya cihazlar kullanılır. Bu yazılım ya da donanımların en temel amacı ortam dosyasının kesintisiz ve sorunsuz bir şekilde akışının yapılmasıdır.

Ortam türleri İnternet ortamında yapıldıkları firmaların ya da uydukları standartların getirmiş olduğu çeşitliliğe göre farklılık arz ederler. İnternette en yaygın kullanılan video formatları [3, 4]: RealNetworks için RealMedia; Microsoft için Windows Media; Apple için QuickTime; Adobe Flash ve Director Shockwave; Java uygulamaları iken ayrıca; Beatnik's Rich Music Format (RMF); Musical Instrument Digital Interface (MIDI); MP3; JPEG; GIF; BMP; TIFF gibi ses ve resim formatları da bulunmaktadır.

İnternet ortamında akışı yapılacak olan ortam türünün HTTP protokolüne uydurulması için kullanılan bazı özel akış protokolleri bulunur. Ortam türlerini farklı şekillerde yayınlama imkânımızın yanısıra bu ortam türlerinin hepsi için farklı programlar kullanılır. Bazı ortam türlerini birçok oynatıcı programları açarken, bazılarını ise sadece o tür için üretilen ortam oynatıcıları açmaktadır. HTTP'ye yardımcı olan akış protokolleri: RTMP, PNM, MMS, RTSPU, RTSPT, MMSU, MMST'dir. Bu protokollerden bazıları iki farklı protokolün birleşiminden yani ortak kullanımından oluşan türev protokoller olabilecekken diğerleri ise tek başına işlevsel olan protokollerdir [5]. Bu akış protokollerine kısaca değinecek olursak:

- RTSP protokolü Windows Media'nın otomatik olarak tanımladığı bir protokoldür. Bu protokol ayrıca RealMedia / RealVideo / RealAudio ve QuickTime Video (*.mov, *.mp4, *.sdp) ortam akışları tarafından da kullanılabilir [68].

- MMS protokolü sadece Windows Media ortam türlerinin akışı için kullanılır [69].

- PNM protokolü sadece RealMedia / RealVideo / RealAudio veri akışları için kullanılır.

- IETF tarafından 1998 yılında geliştirilen RTSP protokolü, kullanıcıya kaynaktaki ortamı herhangi bir yerden sunucuda yayını yapılan ortam türünün uzaktan kontrol edilmesine yardımcı olur. RTSP (Gerçek Zamanlı AKIŞ Protokolü) protokolü bu işlemi VCR komutlarıyla gerçekleştirir; oynat, durdur gibi. Bunların yanı sıra kullanıcıya zamana bağlı kontrol imkânını da tanımış olur.

- Gerçek Zamanlı Mesaj Gönderme Protokolü (RTMP) ilk olarak Macromedia firması tarafından geliştirilen ve Adobe Sistemi tarafından kullanılan bir protokoldür. Macromedia Flash Ortam Sunucusunun ses ve video verilerinin akışının internet üzerinden Adobe Flash kullanıcılarına göndermek için kullanılır. RTMP 1935 girişini otomatik olarak kullanır.

- Şimdide türev olarak tanımlanan iki farklı akış protokollünün bir arada çalıştığı protokollere bir göz atalım:

- Eğer RTSP protokolü, UDP protokolünü kullanırsa bu durumda oluşan yeni protokole RTSPU,

- Eğer RTSP protokolü, TCP protokolünü kullanırsa bu durumda oluşan yeni protokole RTSPT,

- Eğer MMS protokolü, UDP protokolünü kullanırsa bu durumda oluşan yeni protokole MMSU,

- Eğer MMS protokolü, TCP protokolünü kullanırsa bu durumda oluşan yeni protokole MMST, denir.

Ortam dosyalarının İnternet üzerinden akışı için kullanılan protokoller ne kadar hızlı olursa olsun ham verinin sıkıştırılmamış hali İnternet üzerinden akışı sırasında ciddi sorunlar ortaya çıkar. Sıkıştırma metotları genel olarak ikiye ayrılır; kayıplı sıkıştırma ve kayıpsız sıkıştırma. Kayıplı sıkıştırma algoritması internet üzerinde gönderilmek istenilen ortam dosyasını alıp içinde bulundurduğu fazlalıkları atıp daha düşük boyuta indirilmesini sağlar. Örneğin bir resim formatı olan JPEG formatındaki herhangi bir resmin aktarımında bu algoritma kullanılırsa algoritma resim üzerindeki gereksiz renk bilgilerini ya da detaylarını atıp boyutunu düşürür. Kayıpsız sıkıştırma algoritması bir önceki algoritmadan farklı olarak orijinal dosya içerisinde bulunan bilgileri atmadan onları daha küçük boyuta sıkıştırarak bu işlemini gerçekleştirir. Bu iki algoritma arasında en önemli fark Kayıpsız Sıkıştırma algoritmasının lisanslı satışının olması ve bu algoritmayı gerçekleştirmek için yeterli özelliklere sahip bir bilgisayarın temin edilmesidir.

Sunucudan herhangi bir ortam dosyası istenildiği zaman kullanıcıya canlı yayın, istenildiği zaman izlenme ve uyarlanmış canlı yayın şeklinde olmak üzere üç farklı şekilde iletilir. Canlı yayın uygulamalarında kullanıcılar yayını eş zamanlı olarak bulundukları yerden izleyebilme imkânıdır. İstenildiği zaman izleme kullanıcılara ortam dosyalarının kaydedilmesi ve kodlanması ve farklı zamanlarda kullanıcılara istedikleri veriyi izlerken ileri geri alıp tekrar ettirilebilir, bu uygulama canlı yayın uygulamasında bulunmamaktadır. Uyarlanmış canlı yayın her iki uygulamanın da beraber kullanılmasıyla gerçekleşen bir uygulamadır, yani uygulama önceden kaydedilmiş ve kodlanmıştır fakat tüm kullanıcılar için eş zamanlı yayın yapılmaktadır.

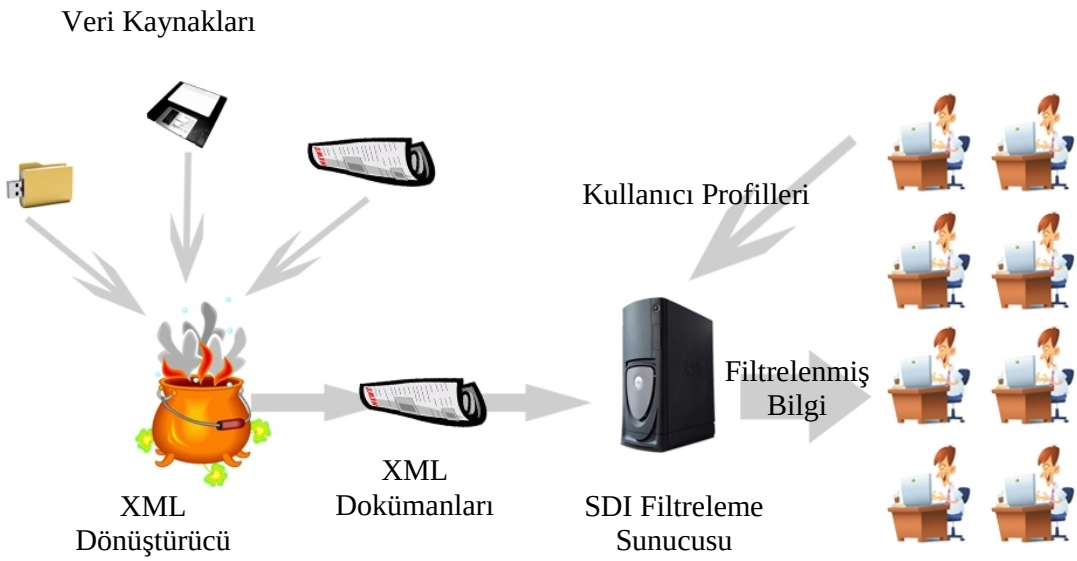
Ortam akışları için genelde açık kaynak uygulamalarının tercih edilmesinin temel nedeni güvenlik açıkları açısından diğer birçok lisanslı yazılımlara göre daha fazla güvenli olmasıdır. En çok kullanılan açık kaynak yazılımlardan birine örnek verecek olursak XML programlama dilidir.

Açık kaynak uygulamalarından XML programlama dili metin tabanlı olması, kullanıcılar tarafından biçimlendirilebilir ve geliştirilebilir bir dil olması bu dilin popülaritesini artırmıştır. XML – Genişletilebilir İşaretleme Dili – dilinin temel amacı internet ortamında gönderilecek olan herhangi bir bilginin kodlama yapılarak sıkıştırıp gönderilmesini sağlamaktır. XML temelde Yaygınlaştırılmış Biçimlendirilebilir Standartları olan Dilin (SGML) bir alt kümesi olarak karşımıza çıkar. XML dilinin standartları Dünya Çapında Ağ Konsorsiyumu (W3C) tarafından belirlenir [6].

XML programlama dili ile herhangi bir doküman gönderilmek istenilirse gönderilecek olan doküman belli filtrelerden geçirilir; bu işlemlerin tümüne Seçilebilir Yayılmış Bilgi (SDI) denir. Kullanılan bu dilde birden fazla farklı filtreleme uygulanır bunlardan bazıları X-Filtresi ve Y-Filtresidir.

Şekil 1.1’de XML dokümanının filtreleme işleminden geçirilmesi işlenmiştir. XML için temelde iki farklı bilgi bulunmaktadır; bunlardan biri kullanıcı adı bir diğeri ise gönderilecek olan veridir. Birçok sistemde bu bilgiler kullanıcılar tarafından oluşturulurken bazı sistemlerde de bu bilgiler sistem tarafından otomatik olarak algılanır. Tanımlanan ya

da tanımlanacak olan bu elementler Tanımlanmış Dosya Türleri (DTD) tarafından belirtilmiştir.



Şekil 1.1. XML dokümanının filtreleme işlemi [7, 8]

SDI sistemi XML dokümanlarına yeni bir düzenleme getirerek XML dokümanlarını filtreleme ünitelerinde işler. Verilerin filtreleme işlemleri esnasında SDI sistemi her bir kullanıcının profilini hazırlar ve bu hazırlama işlemi bir anda milyonları bulabilir. Aranan herhangi bir veri için birçok verinin incelenmesi sisteme bu işlemlerden ötürü aşırı yük getirir. Bu yükü hafiflettirmek için XML programlama dilinde veri akışına yardımcı olacak filtreler geliştirilmiştir [7].

Ortam akışları için kimi zaman bir kod-çözücüler İnternet ya da kullanıcı bilgisayarlarında yardımcı olmak için kullanılırken kimi zaman da akışı hızlandırmak için geliştirilen protokoller, açık kaynak yazılımlar kullanılır. Tüm bu çalışmaların temel hedefi daha güvenli, daha hızlı, daha az kesintili – neredeyse kesintisiz – bir veri akışına imkân sağlamaktır. Ham ortam verisinin İnternet üzerinden akışı İnternet kullanıcıları ve ortam uygulamaları için en temel sorun olduğu için bu alanda çalışmalar her geçen gün hız kazanmıştır. Çalışmalardan bazıları ortam verilerinin içeriğini akış esnasında değiştirmeyi

hedefleyen sıkıştırma algoritmaları üzerine yoğunlaşırken, bazıları da verinin İnternet üzerindeki akışında karşılaşacağı sorunların üstesinden gelmeyi amaçlamıştır.

Hazırlanan bu tez çalışması aşağıdaki gibi organize edilmiştir. Birinci bölümde ortam sıkıştırmalarında kullanılan yöntem ve metotlar üzerinde durulmuştur. İkinci bölümde ise ortam akışlarının İnternet üzerindeki akışları esnasında güvenlik duvarlarının yapılan bu akışa göstermiş oldukları etkileri incelenmiştir. Bir sonraki bölümde kullanıcılara daha hızlı bir akış imkânı tanımak amacıyla TCP protokolü üzerinden gerçekleştirilen alternatif yöntemler üzerinde durulmuştur. Beşinci bölümde gelişen teknolojiler ışığında günümüzde kullanımı artan kablosuz ağ sistemleri üzerinden gerçekleştirilen ortam akışlarına ve geliştirilen metotları incelenmiştir.

1.1. Tezi Amacı

Hazırladığımız bu çalışmada ortam dosyalarının İnternet ortamındaki akışları detaylı bir şekilde incelenip ortam akışları için kullanılan farklı metot ve yöntemlerden bir kaçını sıraladık. Giriş kısmında ortam türleri hakkında kısa bilgi verilirken ortam akışlarının farklı kullanımları hakkında da detaylı bilgi verilmektedir. Ortam akış uygulamalarına canlı yayın uygulaması veya uzaktan eğitim uygulamaları gibi uygulamalar örnek olarak verilebilir. Akışı İnternet ortamında hızlandırmak amacıyla geliştirilen yazılımlar ya da donanımlara birinci bölümde kısaca değinirken bunun yanı sıra geliştirilen yeni protokolleri de ileriki bölümlerde yeri geldikçe değinmeye çalıştık.

Ortam akışları için geliştirilen yazılımları ve önemli algoritmaları bu bölümde değinirken, her ne kadar farklı algoritmalarla ortam akışları hızlandırılmaya çalışılırsa çalışılsın ya İnternet ortamında protokoller arasında veri trafiğini kontrol eden güvenlik duvarları ya da kişisel bilgisayarlardaki güvenlik duvarlarının akış üzerindeki etkilerini ve bu etkileri minimize etmek için neler yapılması gerektiği hakkında bilgi verdik.

Farklı protokoller üzerinde yapılan çalışmaları incelerken gelişen teknolojilerin farklı ihtiyaçları da gün yüzüne çıkardığı ve bu konuda yürütülen çalışmaları inceledik. Yapılan çalışmalar arasında kablosuz ağ yapısı üzerinden ortam akışlarının ihtiyaçları karşılamak amacıyla akıllı ev uygulamalarında sesli ya da görüntülü komut vermek gibi farklı amaçlar içinde kullanıldığını gösterdik. Teknolojinin gelişimiyle beraber birçok

kişinin aynı anda belli platformlar üzerinden bilgi paylaşması ağ yapıları üzerinde çalışmalar yürütenler için yeni bir imkânın kapısını aralamıştır. Çoklu-yollu platformlar üzerinden ortam akışlarının yolları geliştirilmiştir. İnternet üzerinden veri transferi esnasında etkin rol oynayan vekil sunucuların daha aktif kullanılmasıyla beraber geliştirilen çalışmalar ışığında ortam akışları için vekil sunucular daha aktif bir şekilde kullanılmasının yolları araştırılmıştır.

Yapılan her yeni çalışmada hedeflenen tek bir yapı vardır ki o da daha hızlı ve efektif bir ortam akışını heterojen yapıdaki İnternet kullanıcılarına sağlamaktır. Bu çalışmalar ışığında kimi zaman kullanılan algoritma ya da yazılımlar güncellenmekte kimi zaman da kullanılan geleneksel algoritmalara alternatif yeni yol ve yöntemler geliştirilmektedir. Bu çalışmada ortam akışlarının daha hızlı ve daha efektif olmasını amaçlayan metotları inceleyip bu alanda yapılan çalışmalardan bir derleme ortaya sunmayı amaçladık.

2. ORTAM SIKIŞTIRMALARINDA KULLANILAN METOTLAR

Ortam verileri için kullanılan standart bir sıkıştırma metodu olmadığından ortam dosyaları, İnternet üzerinde farklı türlerde kullanıma sunulur. Sıkıştırma algoritmalarının çeşitliliği farklı ihtiyaçları karşılamak için kullanılır. Kimi algoritmalar bellekte daha az yer kaplamayı amaçlarken kimi algoritmalarda daha yüksek bir çözünülürlük imkânını kullanıcılara sunar. Bu nedenle kullanılan her farklı algoritma sonucu oluşan ortam çeşitlerinin İnternet üzerinden akışı yapılması için farklı metotlar kullanılır. Kullanılan her metodun temel amacı daha güvenli ve daha hızlı veri akışını sağlamaktır [9].

Ortam akışları için genellikle ağır şifreleme algoritmaları kullanılır. Bu algoritmalar hem iş yükünü ağırlaştırır hem de ileri düzeyde sistem gereksinimine ihtiyaç duyar. Geliştirilen yaklaşımların temel amacı hem daha düşük sistem gereksinimine hem de daha düşük bir işgücüne ihtiyaç duyan algoritmaları hazırlamaktır.

Ortam verileri incelendiği zaman aslında birbiri peşi sıra gelen fotoğraf karelerinin hızlı bir şekilde gösterilmesidir. Film, önceden hazırlanmış bir kurgu üzerine kurulu bir yapıdır ve bu yapı içerisinde kurgulanan sahneler ayrı ayrı çekilir ve sonunda bütün bir yapı olarak birleştirilir. Kesitler halinde birleştirilen sahneler arasında oluşan geçişler genellikle izleyicilerin pek dikkat etmedikleri bir hususken, bu geçişlerin ortam akışları üzerindeki etkisi göz önüne alındığında bu etkininde minimize edilmesinin yolları üzerine çalışmalar gerçekleştirilmiştir. Renkli fotoğraflarda renkler fotoğraftaki ayrıntıların göz önüne serilmesinde en etkili yollardan biridir. Herhangi bir fotoğrafta belirli bir rengin olmayışı fotoğraf karesinin ya bozuk ya da eksik iletilmesine yol açar. İnternet üzerinden gönderilmek istenilen bir fotoğraf karesinin akışına en fazla fotoğraf karesindeki renklerin etki ettiği bilinmektedir. Bu nedenle geliştirilen bir başka yöntemle ortam verileri İnternet üzerinden siyah-beyaz ve ara renkleri ile iletilirken alıcıya ulaştığında tekrar ters dönüşüm ile renklendirilir ve bu çalışmayla ortam verileri daha hızlı gönderilmiş olunur. Ortam verileri içerisinde fazlalık olarak tanımlanan ve dosya boyutunun artmasına neden olan bilgiler içermektedir.

Fazlalık bilgiler ortam dosyasının sıkıştırılmasında ve akışında ciddi sorunlara yol açarlar. Geliştirilen yöntemlerden biri ortam dosyasında bulunan fazlalık bilgilerin atılmasını hedefler [16, 55]. Kimi algoritma ya da metotlar ortam dosyası üzerinde çalışmalar yaparken diğer bazı algoritmalar ise protokollerin daha efektif çalışması için geliştirilmiştir. Örnek verecek olursak; bazı geliştirilen yazılımlar ortam akışları için kullanılan protokollerde olması muhtemel tıkanıklıkları minimize etmek için kullanılır. Diğer taraftan veri paketleri İnternet ortamında akışı yapılırken bazı paketler kaybolabilir, bu tip bir sorunla yüz yüze gelen veri akışlarında kesinti ya da anlık duraksama meydana gelebilir. Böyle bir sorunu gidermek anlamında hata düzeltme algoritması geliştirilir ki tek amaç kesintinin önüne geçmek ve veri paketlerinin akışını düzenlemektir.

2.1. Geçici Referans Numarası Uygulaması

Ortam akışları için kullanılan ağır şifreleme algoritmalarına alternatif geliştirilen ve geçici referans numaraları kullanarak hazırlanan yöntem ile daha düşük sistem gereksimi duyan bir şifreleme algoritması kullanılarak ortam akışına taşınabilirlik ve hız kazandırılmıştır. Bu yöntemle ortam için kullanılan veri miktarının azaltılması, şifrelenmiş akımın gerçek zamanlı olarak izlenebilmesine olanak tanınmıştır.

Gerçek zamanlı bir ortam, kullanıcı tarafından izlenmek istendiğinde ortam boyutunun büyük olmasından ötürü bu işlem gereğinden fazla zaman almaktadır. Ortam dosyaları genellikle görsel bozulmalara dayalı şifreleme metotlarıyla şifrelenir. Şekil 2.1.a'da gösterilen bu şifreleme ortam dosyasını kullanıcıya bozuk gönderip kullanıcı kod çözücüler sayesinde ortamı tekrar işleyip izlenilebilir hale getirilir (şekil 2.1.b).

Görsel bozulmalara dayalı bir şifreleme sistemiyle şifrelenen herhangi bir ortam dosyası MPEG sıkıştırma yöntemiyle tam olarak sıkıştırılamamaktadır; çünkü görsel öğeleri bozulan herhangi bir ortam dosyası sıkıştırıldıktan sonra kullanıcı tarafından çözülmesi durumunda ortam dosyasında görüntü kaybı olmaktadır. Herhangi bir ortam dosyası düz bir metin gibi işlenip şifrelenmesi durumunda şifreyi çözmek isteyen saldırgana açık noktalar verilmiş olur.

Şekil 2.1.a. Bozuk olan ortam görüntüsü

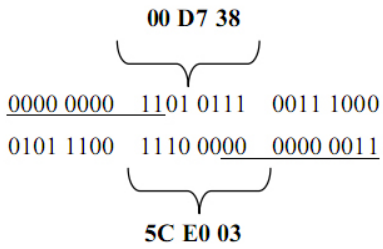
Şekil 2.1.b. Kod çözücülerle işlenen ortam görüntüsü [9]

Başlık bilgisi yerine referans numaralarının kullanımı yöntemi ile MPEG akımında hayati önem taşıyan başlık bilgileri akımdan çıkartılır. Çıkartılan başlık bilgilerinin yerine ortam akımında kullanılması planlanan geçici referans numaraları ile ortam bilinçli bir biçimde bozulur. Böylelikle yeni oluşan şifreleme yöntemi ile ortam dosyasının kaynak ihtiyacı da düşmektedir. Bu sayede akım için gerekli olan zaman ihtiyacı düşerek daha hızlı bir akım gerçekleşmektedir. Bit bazında yapılan referans bozma işlemi 10 bitlik verinin bir sonraki 10 bitlik veri ile yer değiştirilmesiyle olur. Şekil 2.2'de bu işlem gösterilmektedir.

Referans numaraları kullanılarak geliştirilen yöntem ile yapılan denemelerde 309 MB'lık bir ortam dosyası tüm metotlarla karşılaştırılmış ve karşılaştırma sonucunda ortam dosyası 26.40 saniyede sıkıştırılmış ve 93 Mbit/sn ile gönderilmiştir [9].

```
C7 00 F7 D5 82 D8 00 00  01 00 00 D7 38 03 80 00
00 01 B5 81 1F F3 51 80  00 00 01 01 4A 27 0A 0D
7F F6 71 04 7A 49 23 DC  78 AA 91 01 74 78 E9 85
B6 17 6A 0A 0C 60 03 05  E8 00 7F 1C 2E A8 27 13
C5 41 60 3E 89 E1 78 07  93 80 6F 1C 2E D5 00 2E
8F 60 39 00 BC 44 B0 1F  F1 D1 C3 BD E8 3C 8E 64
```

a. Resim başlangıç kodu



b. Byte akımının şifrelenmesi

```
C7 00 F7 D5 82 D8 00 00  01 00 5C E0 03 03 80 00
00 01 B5 81 1F F3 51 80  00 00 01 01 4A 27 0A 0D
7F F6 71 04 7A 49 23 DC  78 AA 91 01 74 78 E9 85
B6 17 6A 0A 0C 60 03 05  E8 00 7F 1C 2E A8 27 13
C5 41 60 3E 89 E1 78 07  93 80 6F 1C 2E D5 00 2E
8F 60 39 00 BC 44 B0 1F  F1 D1 C3 BD E8 3C 8E 64
```

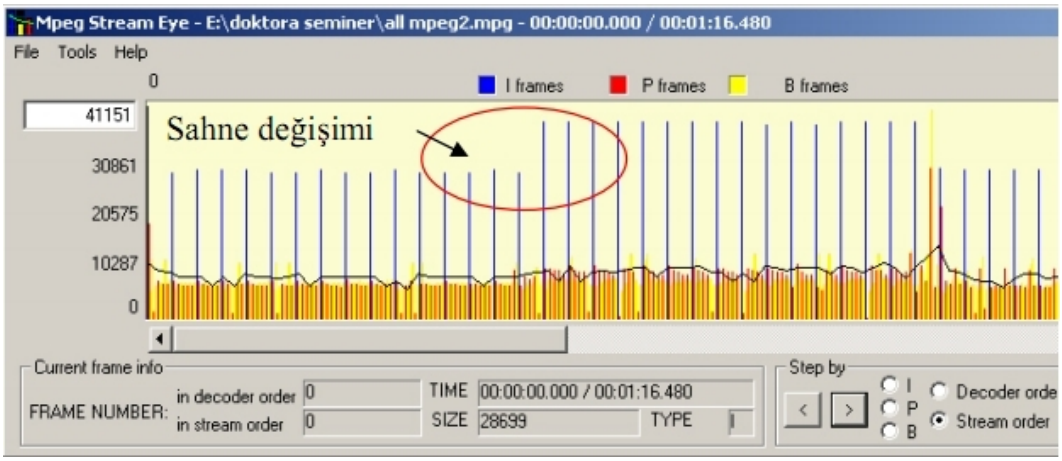
c. Şifrelenmiş akım

Şekil 2.2. Bit kaydırma işlemi ile şifrelenmiş akım [9]

2.2. Sahne Geçişi Belirleme Uygulaması

Ortam dosyaları hazırlanma aşamasında aynı kurguya ait birden fazla ortam dosya parçalarının belirli ortam hazırlama-düzenleme programı ile bir bütünlük halinde yayın taslağına göre hazırlanır. Yayın taslağına göre oluşturulan en son ortam dosyasında hazırlama aşamasında kullanılmış olan ortam parçaları arasında geçişler söz konusudur. Bu sahne geçişleri belirli bir düzen halinde ilerlediği için izleyiciyi rahatsız etmemekle beraber çoğu durumda izleyici bu gelişmelerden haberdar olmaz. Eğer ortam dosyalarının İnternet üzerinden daha hızlı aktarılması istenilirse ortam dosyalarında bulunan sahne geçişleri bir veri tabanında tutulur ve ortaya çıkan ortam dosyasının istenilmesi durumunda kullanılan bu veri tabanı ile akış daha hızlı gerçekleştirilir. Ortam dosyalarında bulunan sahne geçişlerinin tespiti için kullanılan en yaygın yöntem histogram kullanımıdır. Sahne ayrımını tespit etmede kullanılan histogram yönteminin çok büyük bir maliyeti vardır [51].

MPEG yöntemiyle sıkıştırılmış ortam dosyalarında standart ortam dosyalarının aksine üç farklı kare bulunmaktadır. Bunlar; I karesi (tam bir ortam resmidir): gösterilmesi için başka bir resme ihtiyaç duymayan ve böylelikle en çok veriyi kapsayan ortam karesidir. P karesi: bir önceki karede bulunan farklılıkların şifrelenip tutulduğu yerdir ve gösterilmesi için bir önceki kareye ihtiyaç duyar. B karesi: bir önceki ya da bir sonraki karedeki farklılıklar şifrelenmektedir. I karesindeki verinin en fazla % 25'ini içerir. I karesi Gösterilmesi için ya bir önceki ya da bir sonraki kareye ihtiyaç duyar [10].



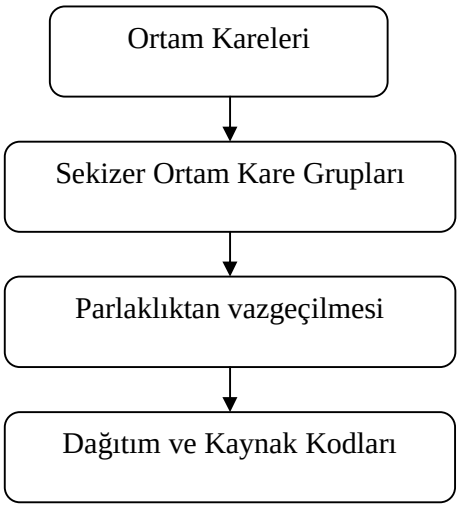
Şekil 2.3. Sahne değişiminin kare tipi yoluyla belirlenmesi [10]

Şekil 2.3’te histogram yöntemi için geliştirilen program ile sahne geçişlerinin tespiti gösterilmektedir. Bu yöntem sayesinde ortam işlemci açısından normal yöntemlerin aksine daha büyük bir zaman kazancı sağlamaktadır [10].

Ortam dosyaları, birbirini takip eden karelerin peşi sıra hızlı bir biçimde oynatılması sayesinde izlenmekte ve hareketliliği tespit edilmektedir. Ortam dosyaları belirli bir sürede arka arkaya hareket eden ilişkili kareler zincirinden oluşmaktadır. Renkli fotoğraflarda herhangi bir ana rengin bulunmaması o fotoğrafın eksik ya da hatalı gösterilmesine yol açar. İnternet ortamında herhangi bir renkli fotoğrafın aktarılması esnasında fotoğraftaki renkler esas rol oynar. Diğer bir değişle fotoğrafın hızlı ya da yavaş aktarılması fotoğrafın çözünürlük kalitesi ve renk kapasitesiyle ters orantılıdır. Yüksek çözünürlüklü bir fotoğraf düşük çözünürlüklü bir diğer fotoğrafa göre daha geç iletilir.

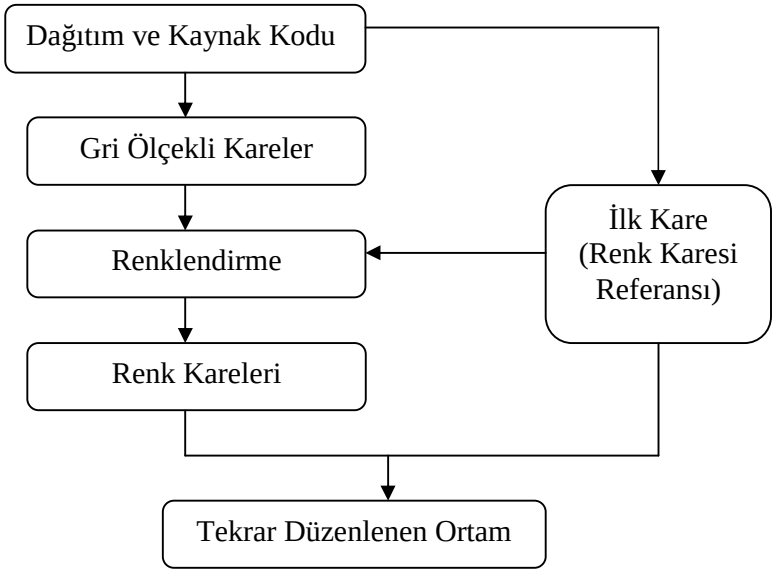
2.3. Ortam Akışlarında Renk Transferi Metodu

İlk olarak; renkler renk uzayındaki lαβ karşılıklarına dönüştürülür, ardından grilik ölçeğindeki her piksel için en iyi örnek renk aralığı belirlenir. Bu belirleme için sıralanan komşu piksellerin istatistiğinden elde edilen veriler kullanılır. Gri ölçeğinde sistemin dikkate alması gereken sadece siyah-beyaz aralığı olmasından ötürü İnternet üzerinden renk transferi daha hızlı gerçekleşmektedir. İnternet üzerinde gri ölçeğindeki renklerle iletilen bir ortam kullanıcıya iletildiğinde kullanıcı tarafından tekrar işlenip renkli olması sağlanmaktadır. Renk transferi algoritması şifreleyici ve çözümleyicide tanımlanması gerekmektedir. Şifreleme cihazında gerçekleşen işlem çok basittir ve herhangi ek bir programlaya ihtiyaç duyulmamaktadır. Çözümleyici tarafından daha önce sekizer kareler şeklinde gruplara ayrılan kareleri şekil 2.4’te gösterilmektedir. Şekil 2.4’te gösterilen diyagramın üçüncü parçası hem kaynağı içerir hem de dağıtım kodlarını içerir. Bu şekil ortam sıkıştırması işlemi için uygulanan standart bir uygulamadır. Üçüncü bloğun içeriğinde ayrıca hareket tahmini, geçiş, geçiş anındaki işlenen miktar kod bilgileri bulunur [11, 52, 53, 54].



Şekil 2.4. Şifreleyicinin ortam kareleri için blok diyagramı [11]

Renk transferi işlemi çözümleyici ve şifreleyicide şu şekilde gerçekleşmektedir. İkinci kare renk transferi için birinci kareyi referans olarak alır ve bu işlem sırasıyla gelen tüm kareler için tekrarlanmakta ve tüm kareler bir önceki kareyi referans alarak ilerler. Parlaklık değerlerinin kaynaktan hedefe transferi için, gri ölçeğindeki her piksel değerinin renkli karelerdeki her piksel değerine karşılık gelmesi gerekmektedir. Şekil 2.5'te ise çözümleyici için blok diyagramını gösterilmiştir; şekilden de anlaşıldığı üzere şifreleyicinin yaptığı işlemlerin neredeyse tam tersi söz konusudur.



Şekil 2.5. Çözümleyicinin blok diyagramı [11]

Bu işlemlerin en temel amacı şifreleyicideki parlaklık değerlerinin göz ardı edilmesi ve gönderilen gri renkli karelerin çözümleyicide tekrardan renklendirilmesi mantığına dayanmaktadır. Bu işlem sayesinde şifreleyici; karelerdeki parlaklık değerleri için ayrı bir işlem yapmak zorunda kalmayacağı için toplam işlem yükü hafiftir. Böylelikle toplam sıkıştırılması gerekli olan piksellerin sayısı düşürülmüş olur [11].

2.4. Fazlalık Bilgilerin Atılması

Ortam dosyaları, içeriğinde şifreleme aşamasında çok sorun çıkartan ayrıca ortam dosyaları için fazlalık olarak tanımlayabileceğimiz bilgileri içerir. Bu bilgiler boyut olarak fazla yer kaplamasından ötürü İnternet üzerinden ortam dosyasının aktarımda veya bilgisayar veya İnternet sunucularında saklama aşamasında problemlere yol açabilir. Sıkıştırma algoritmaları kullanıldığı zaman fazlalık olarak tanımlanan bu bilgiler atılmaya çalışılır. Atılan bilgilere çok fazla dikkat edilir ve orijinal dosyadan fazla uzaklaşmamaya önem gösterilir. Fazlalık olarak tanımlanan bilgiler ise aşağıdaki gibidir.

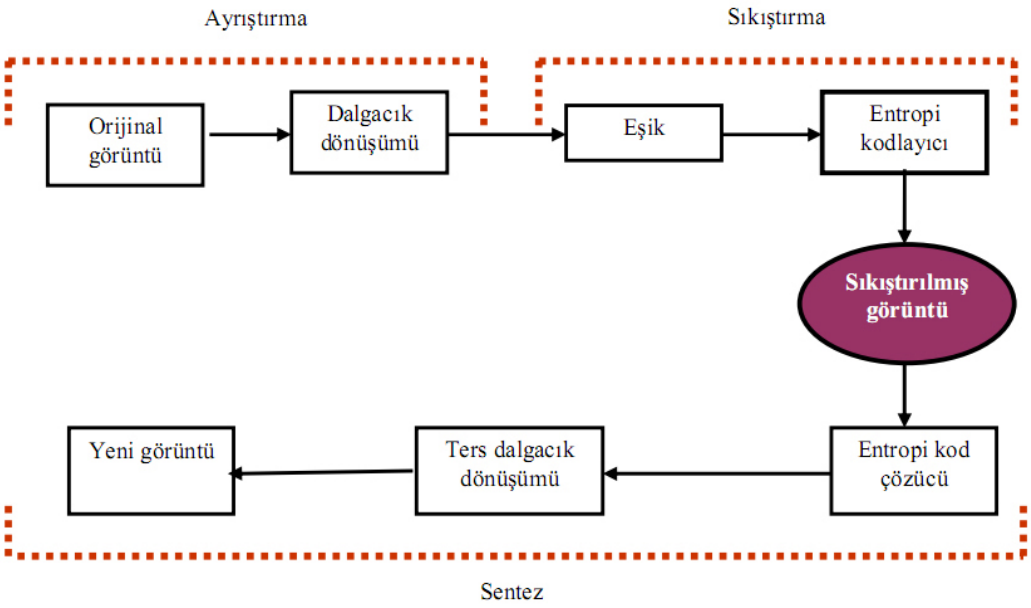
2.4.1. Uzaysal fazlalık: Komşu pikseller arasındaki ilişkidir. Çoğu görüntülerde yan yana pikseller arasında keskin geçiş yoktur. Sıkıştırma yapmadan evvel, birbiriyle ilişkili pikseller (özellikle yan yana olanlar) tespit edilir.

2.4.2. Spektral fazlalık: Farklı renk bileşenlerinin birbirleriyle olan ilişkileridir. Renk bileşenleri farklı bile olsa çoğunlukla birbirleriyle ilişkilidir.

2.4.3. Anlık fazlalık: Video videolarında birbirini takip eden çerçeveler arasındaki ilişkidir. Birbirini takip eden çerçeveler arasında çok büyük fark yoktur. Sadece hareket eden bölüm kadar fark vardır ve diğer bölümler hemen hemen aynıdır. Sıkıştırma işlemi içerisinde bu aynı bölümler fazlalık olarak değerlendirilir.

2.4.4. Algılanabilen fazlalık: İnsanın görme sistemi her şeyi eşit olarak algılamaz. Uzaysal, spektral ve anlık kısıtlamalara bağlı kalarak görüntüde bazı gereksiz bilgiler atılabilir. İnsan gözü bu küçük değişimi algılayacak kadar hassa değildir.

Fazlalıklar görüntü sıkıştırılmasında karşılaşılan en büyük sorunlardan biridir. Dalgacık tabanlı kodlama ile ortam dosyaları Ayrık Dalgacık Dönüşümü (DWT) alınarak kodlanır. Böylelikle herhangi bir ortam dosyası diğer birçok sıkıştırma algoritmasına (DCT – Ayrık Kosinüs Transformu – tabanlı sıkıştırma gibi) göre daha fazla sıkıştırılma imkânı tanınır. Bant genişliği sorunu dâhil saklama alanı gibi sorunlarında önüne geçilmiş olur. Dalgacık tabanlı sıkıştırmanın temel yapısı şekil 2.6'de gösterilmektedir.



Şekil 2.6. Dalgacık tabanlı sıkıştırmanın yapısı [16].

Dalgacık tabanlı sıkıştırmanın dayandığı esas algoritma ortam dosyasındaki karelerin alınıp o karelerin değişik çözünürlüklere ayrılarak belirlenmesidir. Kodlamanın bu bölümü ayrıştırma olarak tanımlanır. Dalgacık tabanlı sıkıştırma metodu ortam görüntülerine yaklaşık 1:300 oranından bir iyileştirme imkanı sunar. Dalgacık tabanlı algoritmaların en büyük dezavantajı yüksek kapasiteli cihazlara ihtiyaç duymasıdır. Aşağıdaki şekil 2.7'de en çok kullanılan Ayrık Kosinüs Dönüşümü algoritması ve dalgacık tabanlı algoritma ile sıkıştırılmış aynı resmin iki farklı görüntüsü gösterilmektedir [16, 55].

Şekil 2.7.a. Dalgacık Algoritması ile yapılan sıkıştırma [16]

Şekil 2.7.b. DCT Algoritması ile yapılan sıkıştırma [16]

2.5. Tıkanıklık Kontrol Mekanizması

Son zamanlarda geliştirilen ve UDP protokolü üzerine kurulu olan birkaç mekanizma ile geliştirilen DCCP (Datagram Congestion Control Protocol) adındaki protokol, tıkanıklığı minimize ederken veri paketlerinin aktarımında güvenli bir ortamı tam olarak sağlamaktadır. Bu bölümde DCCP protokolünün işlevi incelenecektir [17, 56].

DCCP protokolü IETF topluluğu tarafından ortam trafiğini desteklemek amacıyla önerilmiş bir iletim protokolüdür [18]. Bu protokolde iletilen verilerin güvenliği tam anlamıyla sağlanmaktadır. Veri tıkanıklılığını gidermek amacıyla herhangi bir paket kaybı olması durumunda ECN olarak tanımlanan bilgilendirme mekanizması kullanılmakta, bu mekanizma ile de göndericiye, alıcının hangi paketleri alabildiğine ve hangi paketlerin kaybolduğuna dair bilgiler gönderilmektedir [19]. Bu mekanizmada tıkanıklık durumunda veri paketleri atılmamakta yönlendiriciler tarafından işaretleme işlemi gerçekleşmektedir [57].

DCCP protokolündeki paket yapıları incelendiğinde bu protokoldeki fonksiyonlar on dört adet paket türü ile gerçekleştirilir. Bu paketlerden sekizi bağlantı için kullanılırken diğer sekizi senkronizasyon için kullanılır. Her bir DCCP paketi birer sıra numarası taşır böylece paket kayıpları kolayca belirlenip rapor edilebilir. TCP protokolünde sıra

numaraları bayt tabanlı olarak artış gösterirken, DCCP protokolünde sıra numaraları her bir paket için bir artırılır. Bir DCCP paketi, başlık ve içerilmesi zorunlu olmayan uygulama verisinden oluşur [20, 56].

DCCP protokolünde bağlantının kurulması için önce kullanıcı ve sunucu arasında hangi tıkanıklık mekanizmasının ve parametrelerinin kullanılacağına dair bir anlaşmanın olması gerekir. Her uç nokta için farklı mekanizmalar ve farklı gereksinimler bulunmaktadır, dolayısıyla bağlantı kurulmadan önce sunucu ve kullanıcı arasında bir takım ön görüşmeler yapılır. DCCP, güvenilir olmayan bir veri iletimi gerçekleştirse de, onay bilgilerinin gönderiminin güvenilir olması beklenir ve alıcının gönderilen herhangi bir onay bilgisinden birisinin onaylanması gerekmektedir. Her paket için farklı onay bilgisi bulunmaktadır, onay bilgisi gelen paketin gönderimi gerçekleşir [17, 58].

2.6. Ortam Akışına Yardımcı Yaklaşımlar

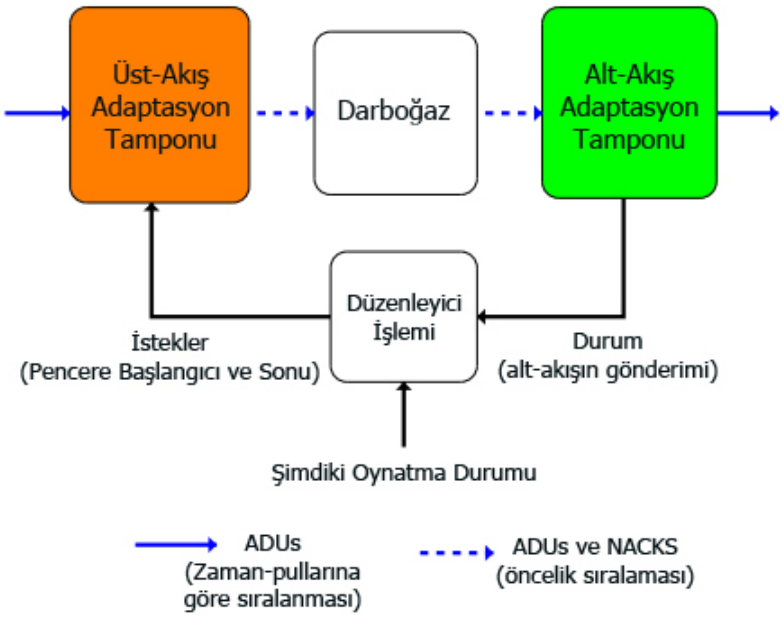
Ortam dosyalarının İnternet üzerinde aktarımı esnasında gönderilen paketlerden bazıları aktarım esnasında kaybolabilir. İleri Hata Düzeltme (FEC) sistemi ile kayıp olan veri paketlerinin tekrar elde edilmesi için ortam akışına ek bilgiler yerleştirilir. Bu bilgiler kısaca; kaynak kodlama, kanal kodlama ya da kaynak/kanal kodlama olarak üçe ayrılır.

İnternet üzerinde akışı yapılacak olan ortam dosyasının gönderimi esnasında bölümlere ayrılır, ayrılan her bir bölüm ise *k* paketine çoğaltılır. Özel bir işlemden geçirilen her bölümün paket sayısı n sayısına çıkartılır ($n > k$). Alıcı tarafından istekte bulunan herhangi bir ortamın gönderimi olduğu zaman alıcıya n paketten K tanesi iletilmelidir ($K >= k$). Fazladan üretilen $n - k$ tane paketten dolayı ve alıcının ayrıca K adet paketin iletilmesinin beklemesinden ötürü gerçekleşecek sıkışıklıklar bu yaklaşımın en büyük dezavantajıdır. Ek paketlerinin üretimi esnasında kayıp oranlarının karakteristikleri dikkate alınmaz. Kayıp oranının düşük olduğu durumda gereğinden fazla ek bilgi kullanılırken, kayıp oranlarının yüksek olduğu durumda ise tam aksine ek bilgi miktarı yeterli değildir ve dolayısıyla eksik paket kurtarımı da mümkün değildir. Bu sorunun önüne geçilmesinin bir tek yolu vardır; kodlayıcının alıcıdan düzenli olarak kayıp istatistiklerini almasıdır [21].

Kaynak kodlama tabanlı FEC yaklaşımında ise *n*. pakete (*n - 1*). adet paket kurtarmaya yönelik bilgiler eklenir. (*n - 1*). paketin kaybolması durumunda n. paket kullanılarak kayıp olan paket kurtarılır. n. pakete eklenen bilgiler (*n - 1*). pakete yerleştirilen GOB'a (Group of Blocks) ait değerlerin daha büyük katsayılar ile nicemlenmiş hallerindedir. Bu nedenle kurtarılan veriler daha düşük kaliteye sahiptirler [21].

2.7. Kaliteye Duyarlı Çoklu-Ortam Dosyaları İçin Öncelikli Akış İşlemi

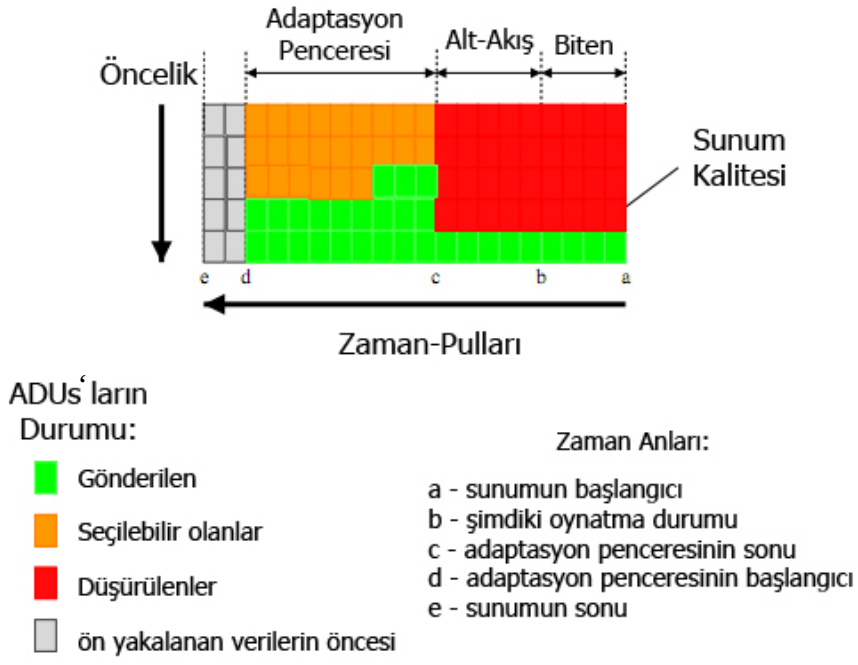
Öncelik sırasına göre akış işlemi, uygulama verilerini uygulama veri üniteleri (ADUs) olarak parçalar ve her bir parçaya bir zaman-damgası yani öncelik etiketi tanımlar. Sonlu bir işlem için ADUs ların kullanılması durumunda zaman damgası ve öncelik etiketi gerçek zamanlı bir işlem için önemli bilgiler barındırır. Öncelik etiketi ortam dosyasının içeriğini açar ve gerekli bilgileri önceden tanımlar çünkü İnternet üzerinden veri akışı olurken bazı kaynak kısıtlamaları olabilir, bunlar: network, işlem ya da kaplanılan yer. Aşağıdaki şekil 2.8'de öncelik işlem kontrolü gösterilmektedir.



Şekil 2.8. Öncelik-İşlem kontrolü [46].

Şekil 2.9'da ardışık dizilen ortam dosyaları için tanımlanan Öncelik-İşlem kontrol mekanizması tanımlanmaktadır. Eşleşen yakalanmış verilerin tekrar sıralanması ardışık olarak dizilen ortam parçalarının dar bir boğazda karşılıkları bulunur ve dizilir. Tekrar yakalama sıralamasının kapasitesinde zaman değerleri kıstas alınır yani zaman damgası vurulan ADUs ların etiketi dikkate alınır. Öncelik-İşlemi algoritması akış için üç alt bileşen barındırmaktadır: üst akış tamponu, alt akış tamponu ve işlem düzenleyici.

Üst akış tamponu ve alt akış tampon algoritmaları şu şekilde işletilmektedir. Üst akış tampon işlem düzenleyici aracılığıyla tüm ADUs'ların hepsi zamana göre sıraya dizer. Düzenleyici her seferinde bunları daha önceden sıraya dizip pencere yapısına bırakmış olduğu verilere göre iletir. Gönderilmemiş eski ADUs'lar veri penceresinde silinir ve böylece yeni veriler için yer açılmış olur. Tüm ADUs'lar alt adaptasyon tampon akışlarına belirlenen öncelik sırasına göre dar boğazlardan geçerek elenir. Alt adaptasyon tampon akışları tüm ADUs'ları toplar ve tekrardan kendi zaman pullarına göre sıralar.



Şekil 2.9. Öncelik-İşlem zaman çizgisi [46].

ADUs'ların akışı yukarıdaki şekil 2.9'da daha net bir şekilde gösterilmektedir. Zaman çizelgesi normal oynatma zamanına göre belirlenir, yani sunum, zaman çizelgesinde "a" anında başlar ve "e" anında sonlanır. Adaptasyon penceresine göre sıralanan ADUs üst-akış ve alt-akışları tekrar dizilir; adaptasyon penceresindeki aralık zamanları kullanılarak kaliteli bir duyarlılık ölçüsü hazırlanır. Geniş zaman aralığına sahip olanlar daha az duyarlı olmasına karşın daha kararlı bir kaliteli adaptasyon örneği göstermektedir. Düzenleyici penceredeki ölçümlere göre bir düzenleme yaparak kaliteli bir akışı düzenler [46].

Sonuç olarak ortam türlerinin İnternet üzerinde yayınlanmasında karşılaşılan en önemli sorun hız sorunudur. Daha hızlı veri transferi için ortam dosyalarının sıkıştırılmasında farklı algoritmalar kullanılabileceği gibi İnternet üzerinden akışının daha hızlı olması içinde farklı algoritmalar hazırlanıp tasarlanmıştır. Ortam türleri için kullanılan farklı algoritmalar ortam türlerinin farklı uzantılara sahip olmasına neden olur. Oluşan farklı türdeki ortam verileri için ortak bir algoritmanın kullanılması neredeyse imkânsızdır. Kullanılacak ortak algoritma ile her farklı ortam türü aynı hız oranına sahip bir şekilde İnternet üzerinden gönderimi yapılamaz. Ortam türleri için geliştirilen algoritmalar ortam dosyası üzerinde çeşitli etkilerde bulunarak hızlı bir veri gönderimi için gerekli değişiklikleri yapar. Bu algoritmalardan kimi şifreleme algoritması üzerinde yapılabilecek değişiklikler olabilecekken kimi de ortam dosyası kareleri üzerinde değişik etkilerde bulunur.

Ağır şifreleme algoritmalarına alternatif olarak geliştirilen ve sıkıştırma metodu için farklı bir yöntem deneyen algoritmalardan biri geçici referans numaralarının kullanımıdır. Böylelikle daha düşük sistem gereksinimi duyan şifreleme algoritması ile veri akışı hem hızlı bir şekilde sıkıştırılır hem de daha hızlı bir şekilde transfer edilir. Sıkıştırma şeklini değiştiren bir başka algoritma ile ortam dosyalarının resim yapıları üzerinde gerekli değişiklikler yaparak veri aktarımı hızlandırılır. Diğer bir değişle ortam dosyalarının sıralı resim yapılarından olduğu için resim karelerinin değişiklikleri üzerine yapılan çalışmalarla sahne değişimi farklı bir metotla tespit edilir, toplam iş yükü düşürülmüş olur ve zaman kazancı gerçekleşmiş olur.

Ortam dosyalarının resim/fotoğraf karelerinden meydana geldiği bilindiğine göre; resim karelerini oluşturan birden fazla farklı rengin İnternet üzerinden aktarımı yerine sadece siyah/beyaz ve gri tonları kullanılarak yapılan yeniden yapılandırmada resim karelerinin daha hızlı bir şekilde aktarılır. Gönderimin yapıldığı tarafta da tekrar derlenip renklendirme işlemleri yapılarak renklendirilir. Bu gerçekleşen basamaklar diğer renkli transferlere alternatif olarak daha hızlı gerçekleştiği tespit edilmiştir.

Ortam kareleri üzerinde yapılan birçok farklı sıkıştırma algoritması bulunmaktadır. Bu algoritmalardan bazıları 1:300 gibi belirgin bir oranda iyileştirme imkanı sunar, ama diğer taraftan bu kadar ciddi oranda yapılan değişiklikler için yüksek kapasiteli cihazlara – donanımlara ihtiyaç duymaktadır.

Şu ana kadar yapılan değişiklikler genellikle ortam verilerinin türlerini değiştirmekle ilgilenirken diğer taraftan geliştirilen bazı metotlarla veri akış protokollerinin daha efektif çalışması sağlanır. Bu gelişmeler kimi zaman protokolün işleyişinde değişiklik yaparken kimi zaman da gönderim esnasında oluşabilecek tıkanıklık gibi sorunların giderilmesi için de kullanılır.

Ortam verileri için geliştirilen algoritmaların türü nasıl olursa olsun veri aktarımının yavaş işlemesindeki en büyük engellerden biri olarak bilinen ortam dosyasının boyutu ve içeriği konusunda yapılacak değişikliklerin en temel amacı daha hızlı bir gönderimi sağlamaktır.

3. ORTAM AKIŞINDAKİ GÜVENLİK DUVARININ ETKİSİ

İnternet ortamında akış için hazırlanan herhangi bir ortam dosyası sunucudan kullanıcıya gönderildiğinde yolda birkaç kontrolden geçer, bu kontrol güvenlik duvarları aracılığıyla gerçekleşir. Ortam verisi sadece İnternet ortamında faal durumda bulunan güvenlik duvarları tarafından kontrol edilmez ayrıca kullanıcı bilgisayarlarında bulunan güvenlik duvarlarının da altında bulunur. Güvenlik duvarları kullanıcı sistemlerinde yetkili olmayan girişleri engellemek amacıyla çalışan yazılım ya da donanımlardır. Sistem üreticileri her ne kadar güvenlik açıkları için güncellemeleri öneriyorsa da çoğu durumda güncellemelerinde yetersiz kaldığı durumlar olabilir, bu durumda lisanslı güvenlik duvarları kullanıcı bilgisayarlarını özel olarak korumaya alır.

Güvenlik duvarları kullanıcının veri alış-verişlerinde kullandığı kapıları kontrol eder. Gerekli gördüğü kapılar ya da yazılımcı firma tarafından önceden yüklenip belirtilen kapılar kapalı tutup veri alış-verişi sınırlanır. Böyle bir durumda kullanıcı güvenlik duvarları üzerinde gerekli değişiklikleri yapıp gerekli izinleri verdirebilir. Temel görevi veri güvenliği ve kontrolü olan güvenlik duvarları İnternet ortamında belirlenen herhangi bir sistemde sistem yöneticileri tarafından da kontrol edilebilirler. Sistem yöneticileri ağ yapısının düzenlemek için tüm yetkisini kullanabilir.

Güvenlik duvarlarına herhangi bir müdahale olmaması durumunda ortam verilerine iki şekilde müdahale eder: ortam sunucusuna bağlanmak isteyen herhangi bir kullanıcının erişimini sınırlar ya da ortam akışlarının sık kullandığı UDP protokolüne erişim bağlantılarını kapamasıdır.

İnternet üzerinden güvenlik açıklarının önemini değerlendirilirken veri hırsızlığı için kullanılan Paket Koklama uygulaması genellikle veri trafiğinde oluşabilecek herhangi bir açıktan kaynaklanır. Ortam akışları için veri alış-verişinde güvenlik duvarlarında yapılabilecek herhangi bir değişiklikle sistemde istenmeden ciddi sorunlar gün yüzüne çıkabilir.

3.1. Güvenlik Duvarları ve Protokoller

Ortam dosyaları İnternet üzerinden kullanıcıya gönderilmek istendiğinde kullanıcıya ulaşması durumunda ya kullanıcının bilgisayarındaki ya da İnternet üzerindeki güvenlik duvarlarına takılmadan iletilmesi söz konusu bile olamaz.

Güvenlik duvarı kimi zaman herhangi bir yazılım kimi zamanda herhangi bir donanımsal parça olarak karşımıza çıkar. Güvenlik duvarının temel işlevi ise network üzerindeki bilgisayarlar arasında veri alışverişlerini izlemek ve kontrol etmektir. Kullanıcılar bir sistem üzerinde kurulu olan güvenlik duvarlarında gerekli değişiklikleri yaparak ya kendi sistemleriyle ilgili ya da bulundukları ağ üzerindeki diğer kullanıcıların İnternete bağlanmalarıyla ilgili gerekli değişiklikleri yapabilir. Bu değişiklikler içerisinde sistemlerin erişimlerini sınırlamak ya da kullanıcılara daha fazla yetki tanımak gerekebilir.

Birçok güvenlik duvarı paket filtreleme sistemine sahiptir. Paket filtreleme sisteminde gönderilecek ya da iletilecek olan herhangi bir veri paketler halinde incelenir. Daha gelişmiş güvenlik duvarları ise TCP ya da UDP den gönderilen veri paketlerinin içeriğini incelemekle kalmaz ayrıca veri paketleri tarafından kullanılan TCP/UDP deki bağlantıların durumlarını da izler [12].

3.2. Güvenlik Duvarı ve Ortam Akışı

Güvenlik duvarı ortam akışlarına iki şekilde müdahale eder. İlk olarak güvenlik duvarı ortam sunucusuna bağlanmak isteyen herhangi bir kullanıcıyı daha bağlanmadan engelleyebilir. Çünkü birçok ortam sunucusu kullanıcı tarafından kendisine istekte bulunmadığı müddetçe ortam dosyalarını göndermez. Ortam sunucusuna istekte bulunmak için kullanılan bağlantılar güvenlik duvarı tarafından kapatılmış ya da engellenmiştir (örneğin 80 bağlantısı gibi). Bu sorunu aşmak, güvenlik duvarlarından sakınmak için ortam sunucularını güvenlik duvarlarının dışına bırakmamız gerekmektedir. Bu sorundan sakınmanın bir diğer yolu da İnternet ortamında oluşturulacak bir alt-networkun oluşturulması sayesinde olur. Bu alt-network için özel olarak kullanılan güvenlik duvarları bulunur.

Ortam akışlarına güvenlik duvarlarının müdahalesinin bir diğer yolu da ortam trafiği için tüm UDP bağlantılarını kapamasıdır. Çünkü UDP, bağlantılarını TCP bağlantılarından daha zor kontrol eder. UDP bağlantılarının kapatılmasının önemini bir örnekle açıklayalım; UDP üzerinden herhangi bir yetkisiz kullanıcı hiç istenmedik bir ortam dosyasını UDP protokolü üzerinden akışını gerçekleştirebilir. Böyle bir durumun engellenmesi için bu yol alınmaktadır [13, 62].

3.3. Güvenlik Duvarı ve Çoklu-Ortamlar

İnternet ortamında çoklu-ortamların popülerliklerinin artması çoklu-ortamların işlenmesinde ciddi güvenlik açıklarını oluşturur. Gelen veri akışlarını yerleştirmek için standart bir paket filtreleme güvenlik duvarı bir veya daha fazla bağlantı noktasının açılması gerekmektedir. Açılan her bağlantı noktası için bu defa güvenlik kaygıları gün yüzüne çıkar. Çünkü kullanıcının bulunduğu yerin dışındaki kırıcılar dâhili kullanıcıların adresini belirlemek için bu bağlantı noktaları kullanabilir.

Çoklu-ortam uygulamaları isteğinde bulunan kullanıcı sunucu dışında düz bir TCP ve UDP bağlantısını kullanarak bağlanır. Sunucu normal bir şekilde bağlantıyı yanıtlar. Bu iletişim normal TCP ve UDP modelinin takibini kurar ve güvenlik duvarı normalde bu bağlantının üstesinden gelir. TCP ya da UDP gelen akışları toplamak amacıyla, standart bir paket filtreleme sistemi gelen bağlantılar için bir veya daha fazla yüksek bağlantıyı açık tutmalıdır (TCP ve UDP bağlantı noktaları 1023 daha yüksek bağlantı noktaları gibi). Böyle bir uygulama güvenlik açığının oluşmasına yol açar, çünkü yüksek bağlantı noktası dâhili ağ kullanımı için açık olabilir (örneğin, bir veritabanı sunucusu veya diğer servis tarafından). Dışarıdaki kırıcılar yüksek bağlantıların açık dâhili sunucularındaki adreslerini bulmak için networku araştırır. Diğer taraftan birçok çoklu-ortam üreticileri UDP yerine TCP'yi daha düşük giderleri olduğu için kullanımını teşvik eder.

Paket filtreleme yönlendiricileri akışı yapılan veri paketlerini kontrol edemez. Çünkü sunucu kullanıcı tarafından istekte bulunan UDP veri paketlerini tekrar aynı bağlantı üzerinden göndermez. Paket filtrelemelerinin veri akışını kontrol etmesinin bir tek yolu UDP'nin birçok yüksek bağlantı noktalarını yönlendirici üzerinde açık tutmaktır [14].

3.4. Güvenlik Duvarı ve Paket Koklanması

Ağ üzerinden herhangi akışı yapılan bir verinin çalınması işlemine Paket Koklaması uygulaması denilmektedir. Bu işlem ağ içerisinde bulunan ve daima çalışır durumda bulunan, çalışırken de tüm trafiği kontrol eden yazılımsal ya da donanımsal elemanlardır [25]. Bu uygulamalardan bazıları, Ethernet kartı modülü vasıtasıyla, incelendiği zaman görülmektedir ki “promiscuous mode” olarak tanımlanan ve ayrımsız olarak her türlü paketi izinsizce ele geçirip işleyebilme özelliğine sahip olan uygulamalardır [26, 63].

Paket koklayıcıları iki farklı şekilde çalışır. Paylaşımlı Ortamlarda (Shared Ethernet) tüm kullanıcılar hub sayesinde paket alımı ve gönderiminde ortak bir ağı kullanarak haberleşirler. Böyle bir yapıya gizlenen ayrımsız tür özelliğine sahip Ethernet kartları kullanılarak iletilen herhangi bir veriye gizlice ulaşılır. Tüm bu yapılandan ne alıcının ne de kullanıcının haberi olur. Diğer bir yol ise Anahtarlı Ortamlarda (Swithced Ethernet) tüm bilgisayarlar hub yerine anahtar vasıtasıyla birbirine bağlanır ve haberleşir. Böyle bir ortamda ise uygulamaya geçecek olan yazılım ya da donanım anahtara çok fazla istekte bulunur. Bu istekler karşısında anahtarın kendi özelliğinden sıyrılıp hub gibi davranması zorlanır. Hub modunda çalışmaya başladığı an ilk yol takip edilerek akışı yapılan veri koklanır [27, 63].

Sonuç olarak güvenlik duvarı sadece kişinin bilgisayarına gelebilecek saldırılara karşı korumakla kalmaz İnternet üzerinden yapılan bilgi alışverişlerini de kontrol eder. İnternet üzerinden zararlı yazılımların gönderimi konusunda bilgisayar kullanıcıları için birçok tuzak bulunmaktadır. Bu yazılımların yayılması amacıyla kullanılan ilgi çekici sloganlar için magazin medyasında ünlü olan kişilerin isimleri kullanılmıştır. Virüslere karşı alınabilecek en önemli önlem güvenli yazılımlarını kullanmaktır.

İnternet üzerinden akışı yapılacak olan veri türünün ne olduğu önemsenmeden veri alışverişinde kullanılan protokollerde bulunan güvenlik amaçlı filtreleme programları herhangi bir gönderim esnasında verileri paketler halinde inceler ve kontrol eder. Bu paket filtreleme yazılımları ayrıca protokollerdeki veri bağlantı durumlarını da izler.

Protokollerdeki güvenlik duvarları ortam akışlarına iki farklı yolla müdahale eder. Bunlardan biri ya kullanıcıyı bağlanmadan engeller ya da akış protokolleri için kullanılan UDP gibi protokollerin tüm bağlantılarını kapamasıyla olur. Güvenlik duvarlarının bu önlemleri almasına karşın ağ dışındaki kırıcılar açık hataları kontrol edip sisteme bağlanabilir ve sistem için zararlı faaliyetlerde ya da bilgi hırsızlığında bulunabilir. Güvenli bir sistemin devamı açık hatların kontrol altına alınmasıyla olur.

Bu önlemlerin alınmasına karşın ağ dışındaki kırıcılar sistem içerisindeki veri trafiğini incelemek amacıyla paket koklaması yöntemiyle verileri tespit edip bilgi hırsızlığında bulunur. Ortam akışlarında yapılan herhangi bir güvenlik zafiyeti ağ dışındaki sistem kırıcıları tarafından kötü amaçlı kullanılır. Ortam verileri diğer veri türleri gibi İnternet ortamında bazen protokoller arasındaki güvenliği sağlayan ve trafiği yönlendiren yazılımlar ya da donanımlar aracılığıyla kontrol altında tutulması, yapılan bu işlemin sonucunda ortam verilerinin akışında ciddi anlamda gönderim esnasında düşüşlerin veya aksaklıkların oluşmasına sebebiyet vermektedir [27]

4. TCP ÜZERİNDEN ORTAM AKIŞLARI

Ortam akışları genellikle UDP protokolünü kullanır fakat ilk çıktığında sadece metin belgelerinin iletimi için kullanılan TCP protokolü üzerinde yapılan değişiklikler sayesinde sadece metin verileri değil diğer veri türlerinin de akışı yapılır hale gelmiştir. Ortam akışlarının İnternet üzerinden akışı esnasında gerçekleşen zorluklardan ötürü değişik yöntem ve çalışmaların amacı daha hızlı, daha güvenli ve daha efektif veri transferini sağlamaktır. Yapılan çalışmalar göstermiştir ki ortam akışlarının İnternet üzerinden UDP protokolüne alternatif diğer protokoller üzerinden de gönderilebilir. Bu protokollerden biri de TCP protokolüdür. TCP protokolü üzerinden ortam akışlarında dikkat edilmesi gereken en önemli nokta veri kesintisi oluşma ihtimalidir. Bu ihtimal dâhilinde yapılan kimi çalışmalarda TCP protokolü bant-genişliği paylaşımı prensibi üzerine çalışmalar yapılırken kimi çalışmalarda da tek yönlü TCP bağlantısının yerine çok yönlü veri bağlantısı kurarak ortam akışları şekillendirilir. Bant-genişliği yaklaşımında ortam dosyaları farklı bit-oranlarına göre şifrelenir [15]. Böylelikle farklı bit-oranlarıyla sıkıştırılan veri dosyaları sunucuda saklanıp kullanıcıya gönderilmek için hazır bekletilir. En yapılan çalışmalardan biri de TCP ve UDP protokollerinin beraber kullanımının yolları üzerinde yoğunlaşmaktadır.

TCP protokolünde oluşan tıkanıklık sorunu veri transferi için ciddi sorunlar oluşturduğu için ya protokol üzerinden yapılandırmaya gidilmeli ya da farklı metotlarla bu ciddi sorun aşılmalıdır. TCP protokolünde yapılacak herhangi bir yapılandırılma aşamasında tüm TCP bağlantıları etkilenir ve böyle bir işlemin gönderici çekirdeğinde tekrar derlenmesi gerekir. Bu sebepten ötürü geliştirilen yöntemlerden biri de ÇokluTCP bağlantı metodudur. Bu metotta birden fazla bağlantı kullanılarak tıkanıklık riski minimuma indirilmiş olur. Tıkanıklık sonucunda oluşabilecek kesintilerin önüne geçebilmek amacıyla Akış Kontrol Protokolü adında yeni bir protokol oluşturulup veri gönderiminin devamlılığı sağlanabilir [37]. Ortam verilerinin TCP üzerinden gönderim metotları çeşitlilik arz ettiği için yapılan her çalışmada farklı metotlar denenerek daha hızlı veri transferini TCP ile gerçekleştirmenin yolları amaçlanmıştır. Aşağıda değinilen yöntemlerden biri de oran kontrol mekanizması prensibine dayalı geliştirilen TFRC kontrol

protokolüdür. Bu yeni protokol sayesinde veri yolarında gerçekleşen tıkanıklık sorunu farklı bir metotla çözümlenmektedir.

Yapılan her çalışma temelde UDP protokolü üzerinden gönderilen ortam verileri için farklı protokol kullanımları ile daha hızlı bir yapılandırma amaçlanmaktadır. Kullanılan aracı protokollerin hepsinde veri transferi TCP protokolü üzerinden gerçekleşir ve böylelikle TCP kontrol protokolünün eksikliklerinin giderilmesi amaçlanır.

4.1. TCP ve Bant-Genişliği

Bant-genişliği paylaşımı için aynı RTT üzerinden akışı yapılan iki bağımsız ortam akışı olsun, bunlar için aynı sayıda bağlantı noktası önerilsin. Fakat bu bağımsız ortam akışlarından biri yüksek oranla şifrelenirse kullanıcı alacağı bu yeni servisi orantısız ve kötü olarak algılar. Büyük paket oranı ayarı için ise akışı yapılan ortam dosyaları hem düşük hem de yüksek bant genişlikleri için oran uyuşması sağlamak amacıyla birçok bit oranına göre şifrelenir. Bu uygulamaya örnek verecek olursak; Real Network akıllı sistem teknolojisi birçok bant genişliği için şifreleme yaparak kullanıcılara geniş bir imkân sunar. Her ne kadar bazı akışlar oran-değiştirilebilme özelliğine sahipken, değiştirilebilir oranlara sahip olan oranların sayısı ise çok azdır.

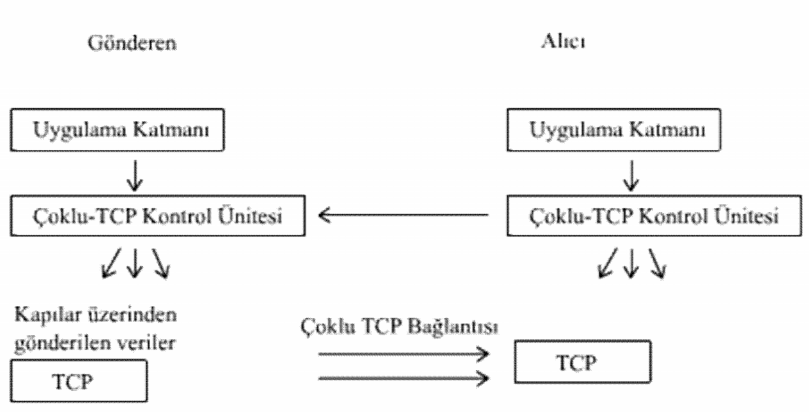
TCP üzerinden yayını yapılacak olan ve transfer edilmek için saklanan ortam verileri adreslenir. TCP üzerinden tekil akışın çoklu bağlantılar üzerinden gönderilmesinin avantajı çoklu bağlantılar ortam akışları için bant-genişliği paylaşımında bulunması ve bununla beraber büyük tıkanıklıkları uyarılmış pencereler sayesinde gidermesidir [15].

Çoklu-oran imkânına sahip olan ortam dosyaları çoklu TCP bağlantılarında transferleri en rahat olan dosyalardır. Düşük oranlarla şifrelenmiş olan ortam dosyalarının akışını rahatlatmak anlamında hızlandırma işlemi yapılır. Bu hızlandırma işlemi çoklu bant genişliği imkânı olduğu zaman geçerli olur.

Yeterli olmayan bant-genişliği oran artırımı mümkün olduğu zaman, düşük oranlardaki devamlı transfer için kaynak oluşturma hizmet kesintileri algılanması azaltılabilir [15]. Klasik ortam akışı algoritmaları zorunlu olmadıkça TCP protokolünü

kullanmaz verilerini UDP protokolü üzerinden gönderir. TCP üzerinden çoklu-ortam akışı yapılmak istenildiği zaman bazı ciddi sorunlar ortaya çıkar. Sonlu güvenilir bir bağlantının kurulabilmesi ve sürdürülebilmesi için TCP kullanıcı bilgisinden kayıp paketleri bulup alıcıdan tekrar kullanıcıya iletilmesini sağlamalıdır. Yapılan çalışmada elde edilen sonuçlar ışığında bir akış içerisinde kayıp olan herhangi bir paket akışının birden kesilmesine yol açabilir [32]. Eğer TCP protokolü modifiye edilmeden veri aktarımındaki bu düşüş giderilebilseydi daha aktif ve az sıkışıklık ile veri transferi gerçekleşebilir.

Bu bölümde klasik tekyönlü TCP bağlantıları yerine ortam akışları için TCP üzerinden çoklu bağlantı metodu şekil 4.1'de gösterilmektedir. ÇokluTCP kontrol ünitesi uygulama katmanının altında ve transfer katmanının da – hem alıcı hem de verici transfer katmanı kullanılıyor – üstünde uygulanacak bir şekilde tasarlanmıştır [36].



Şekil 4.1. ÇokluTCP sistem diyagramı [44].

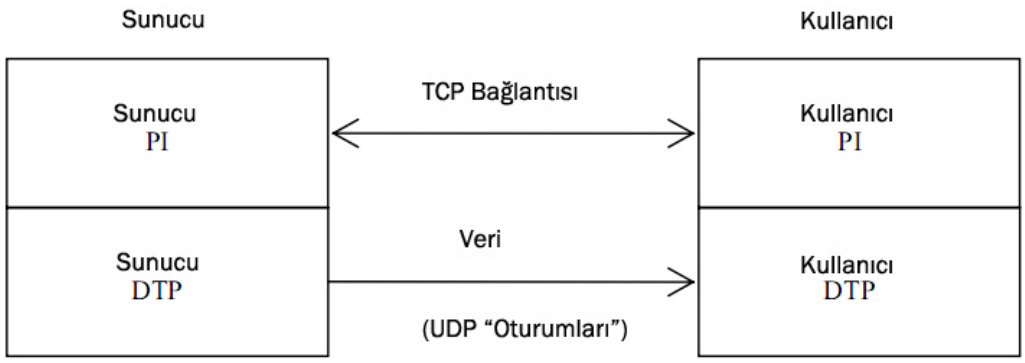
Alıcıdaki ÇokluTCP kontrol ünitesi akış uygulamasından giriş özelliklerini alır, akış uygulamasında akış oranı ve işgücü direnci gibi bilgiler bulunmaktadır. İşgücü direnci uygulaması anlık trafik akışı toleransı olarak tanımlanır. Alıcıdaki ÇokluTCP kontrol ünitesi anlık işgücü ölçer ve bu bilgiyi oranları kontrol etmek, çoklu TCP bağlantıları tarafından oluşan işgücü direnci ve alıcının pencerelerini her bağlantıda dinamik olarak değiştirmek için kullanılır [36].

4.2. TCP ve UDP Protokollerinin Beraber Kullanımı

İnternette her durumda transfer protokolü olarak TCP ve UDP'nin kullanıldığına daha önceden değinmiştik. TCP protokolü tekrar gönderme prensibine göre çalışırken oluşabilecek tıkanıkları dikkate alır ve veriyi tekrardan gönderir; diğer taraftan UDP ise tıkanıklık kontrolü yapmaksızın veriyi gönderir.

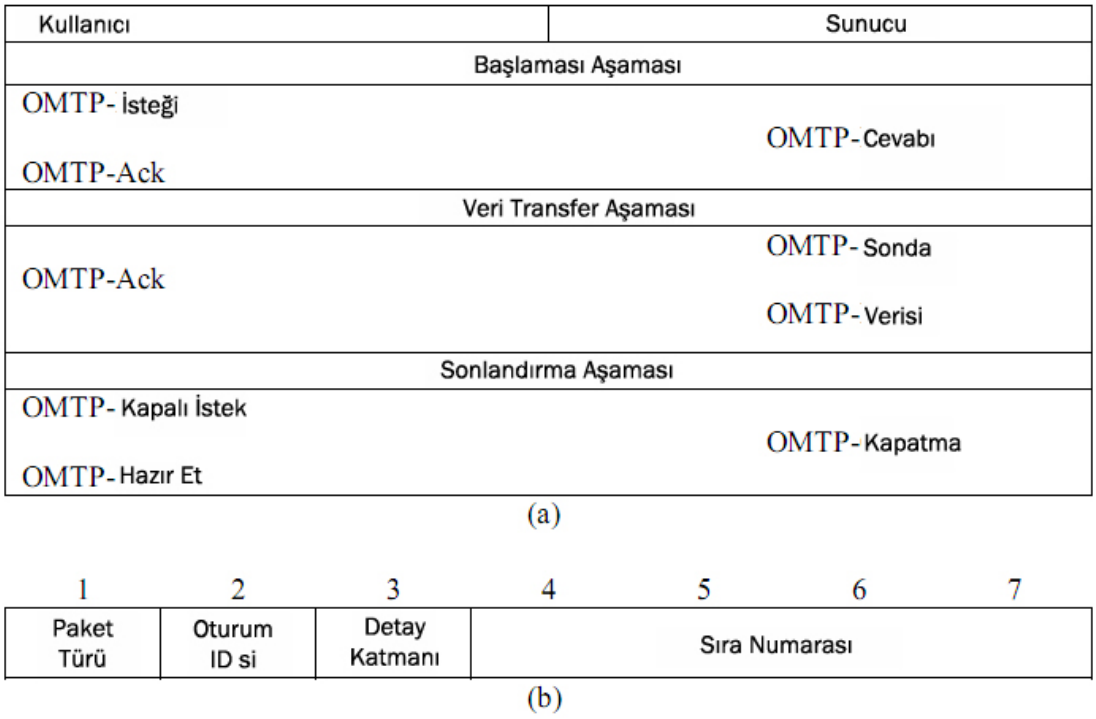
Bu işlemler esnasında UDP, TCP trafiğindeki networkta bulunan tüm uygun bant-genişliklerini sıkboğaz eder. Fakat RTCP (RTP kontrol protokolü) tüm tıkanıklık bilgilerini TCP'den alır ve böylece çoklu-ortamlar için kaliteli bir servis imkânı tanınmış olur. Kullanımda olan birçok Veri-bloğu Tıkanıklık Kontrol Protokolü (DCCP), UDP protokolünün görevini yapmaya çabalar yani kullanıcıya daha az tıkanıklık ile daha akıcı bir veri akışı imkânı tanımaya çalışılır.

TCP ile UDP protokollerinin eş zamanlı kullanılması ile çoklu-ortamların daha akıcı bir şekilde iletilmesi için önerilen yönteme incelenerek, TCP ile UDP protokollerinden belli özellikler alınarak yeni bir protokol oluşturulur. Bu protokol Ortam Transferi Katman Protokolü (OMTP) dür. OMTP daha az işgücü gerektiren UDP protokolü ve güvenilir bir TCP tıkanıklık kontrolünün birleştirilmesi ile oluşur. Aşağıdaki şekil 4.2'de OMTP protokolünün çalışma mantığı gösterilmektedir. OMTP iki ana bölümden oluşur, bu bölümler birbirlerinden ayrı olmasına karşın birbirleriyle ilişkili olarak çalışır. Bu bölümler: tipik UDP veri transferi oturumuna benzerken; bir diğeri ise TCP bağlantı kontrol bilgisiyle eşdeğer çalışan birimdir.



Şekil 4.2. OMTP iletişim modeli [35]

OMTP protokolündeki veri türleri olarak OMTP protokolünde temelde dokuz farklı türden veri paketi bulunmaktadır, bu paketlerin her biri farklı fonksiyonlara sahiptir. Her paket türü tipik TCP başlığına sahiptir ve ayrıca üç aşama vardır: başlangıç, veri transferi ve sonlandırma. Bu basamaklar aşağıdaki şekil 4.3'te gösterilmektedir.



Şekil 4.3. OMTP (a) Paket Türleri (b) Bayt türünden Formatları [35]

Başlangıç aşamasında kullanıcıdaki uygulama OMTP'yi sunucudaki çoklu-ortam oturumu başlar duruma getirmek için kullanır. Bu durum TCP bağlantısının kurulmasıyla ve üç OMTP paketinin değiş-tokuşu esnasında gerçekleşir; değiştirilen üç paket: OMTP-İstek, OMTP-Cevap ve OMTP-Onay paketleridir. Her OMTP-Veri paketi UDP 2 baytlık başlık bilgisine – oturum ID'si ve katman detay alanı – parçasına sahiptir. Bunların dışında herhangi bir bilgiye ihtiyaç yoktur; çünkü UDP tek bir paket türü kullanır. Her OMTP paketi TCP bağlantısında değiştirilir.

OMTP-istek paketindeki ilk bayt paket türü koduna sahiptir, ikinci bayt ise oturum ID'sine sahiptir, üçüncü bayt istekte bulunan detaylı katman numarasına sahiptir ve dördüncü bayt başlangıç sıra numarasına sahiptir. OMTP-istek paketi ayrıca ekstra bilgileri

barındırma seçeneğine de sahiptir örneğin maksimum Katman Detayı değeri gibi alan miktarı ve güncel transfer oranı uyumu gibi. Geri kalan paket türleri ise aşağıdaki bilgilere sahiptir:

- OMTP-İstek paketi üç baytlık bilgiye maksimum Detay Katmanı bilgisi numarası – bu numara bir önceki istek detay numarasına ya eşit ya da küçüktür – ve kullanıcı tarafından istekte bulunan istek numarasına sahiptir.
- OMTP-Onay paketi üç baytlık ilk anlaşılan Detay Katmanı ve ilk başlangıç numarasına sahiptir. Bu paket türü hem sunucu hem de kullanıcı tarafından kullanılır.
- OMTP-Soruşturma paketi üç baytlık güncel Detay Katmanı numarası ve güncel birer birer artan sıra numarasına sahiptir. Sunucu OMTP-Soruşturma paketini periyodik olarak gönderir.
- OMTP-Kapalı-İstek paketi üç baytlık güncel Detay Katmanı ve güncel sıra numarasının bir fazlasına sahiptir; OMTP-Kapalı paketiyle birebir aynı formattadır, aradaki tek fark Detay Katman alanı sonlandırıcı kod için kullanılır.

Son paketimiz, OMTP-Reset paketidir ve reset paketiyle aynı formattadır, fakat OMTP oturumunu aniden sonlandırma özelliğine sahiptir. Bu paket hem sunucu hem de kullanıcı tarafından kullanılır [35].

4.3. Çoklu TCP Bağlantısı Kullanımı

TCP protokolünün ortam dosyalarının transferinde kullanıma uygun olmadığına yani yapısı gereği veri transferi esnasında anlık kesintilerin olması ortam akışları için kabul edilemez bir durumdur. Eğer TCP protokolü üzerindeki anlık veri kesintilerinin önüne protokolü tekrardan yapılandırmaya gitmeden bu soruna bir çözüm bulunduğu takdirde daha hızlı daha akıcı bir veri transferini elde etmiş olur. Çoklu-ortamların TCP üzerinden akışının daha kolay yapılabilmesi için ortam dosyalarının transferinde çoklu TCP bağlantısının bu soruna getireceği farklı bir yaklaşım incelenir.

Herhangi bir ortam dosyasında karşılaşılabilecek fazla işgücünü hafifletmek amacıyla kullanıcı göndereceği verinin transfer oranında küçük değişiklikler yaparak

düşürmesi gerekir ya da tıkanıklıktan kaçınma işlemi hızlandırılmalıdır ya da ilk iki durumu birleştirmelidir. Bu durumlar için söz konusu olan birkaç dezavantaj bulunmaktadır. Bu dezavantajlardan ilki yapılacak böyle bir değişiklikte tüm TCP bağlantıları etkilenir ve böyle bir işlemin gönderici çekirdeğinde tekrar derlenmesi gerekmektedir. Bir diğeri ise bu sorunların düşürülmesi izole edilmiş makinelerde potansiyel TCP istikrarsızlıkları networkta oluşabilir. Sonuncu dezavantaj ise gönderim hızına bu değişikliklerin dinamik olarak nasıl etkileyeceği kesin değildir.

Bu sebepten ötürü geliştirilen çokluTCP bağlantısı ile klasik TCP bağlantılarından vazgeçilmiş ve birden fazla TCP bağlantısı kullanılarak çoklu-ortamların akış uygulamaları daha bir efektif yapılmıştır. Bu yaklaşımda TCP'nin tekrardan modifiye edilmesi gerekmemektedir. Aşağıdaki şekil 4.4'de çokluTCP algoritması resmedilmektedir. ÇokluTCP kontrol ünitesi aşağıdaki uygulama katmanı ve üstteki transfer katmanına hemen uygulamaya geçer. Alıcıdaki çokluTCP kontrol ünitesi akış uygulamasındaki giriş özelliklerini alır. Alıcıdaki çokluTCP üniteleri esas işgücünü ölçer ve bu bilgiyi oranları kontrol edilmek için kullanır.

ÇokluTCP algoritmasının düşük performanslı networklarda işgücünü nasıl hafiflettiği incelenirse; eğer networkta herhangi bir tıkanıklık yoksa alıcı veri akışını tekil TCP bağlantısıyla kontrol eder. Bu zaman içinde gerçekleşen işgücünü formulize etmek gerekirse:

$$T = \frac{W_{max}\, MTU}{RTT} \tag{4.1}$$

şeklinde olur. Bu denklemde RTT döngüsel tur zamanını, MTU ise TCP deki maksimum transfer ünitesi olarak tanımlanır. Eğer paket kaybı olması durumunda, TCP aşağıdaki şekil 4.4.a'da görüldüğü gibi transfer hızını yarıya indirerek akışa devam eder. Böylelikle veri transferi düşmesiyle beraber formül (D):

$$D = \left(\frac{1}{2}\right)\left(\frac{W_{max}\, MTU\, RTT}{2}\right)\left(\frac{W_{max}}{2\, RTT}\right) = \frac{W_{max}^2\, MTU}{0} \tag{4.2}$$

olur. Denklemde çıkarılacak sonuçlardan biride TCP'deki pencere ölçüsünün her turda Wmax/2 dan Wmax kadar arttığı ve eşitliğin WmaxRTT/2 çıktığı görülmektedir. Eğer,

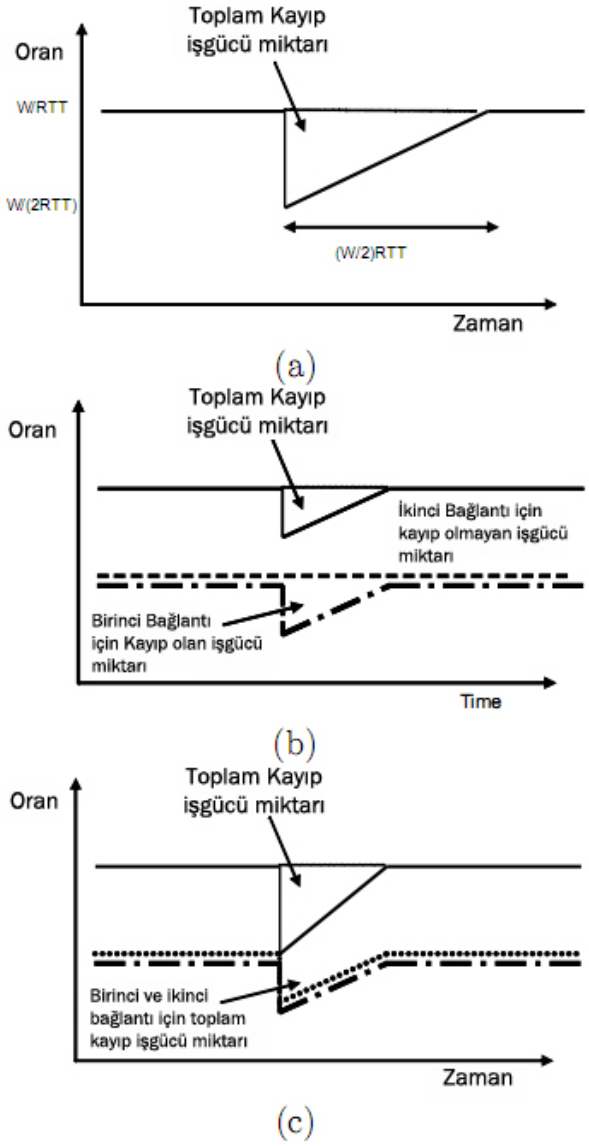
açıkça değinmek gerekirse, kayıp olan paketlerin sayısı akış oturumu esnasında artarsa, toplam işgücü azalması başlangıç tamponu tüketirse, yeniden oynatmalarda durmalara neden olur. İki TCP bağlantısının aynı uygulama için kullanılması incelenirken durum biraz daha açıklığa kavuşur. Uygulama akışı sabit tutulmaya çalışılırsa yani W_{max}/RTT durumunu bir TCP bağlantısı için sağlamak isteniliyorsa, her iki bağlantı için $W'_{max} = W_{max}/2$ durumunun sağlanması lazım; aşağıdaki şekil 4.4.b'de gösterildiği gibi. Şimdide kesintinin sadece bağlantıların birinde olduğunu varsayılsın ve toplam işgücü azalması:

$$D' = \left(\frac{W'_{max}\, MTU\, RTT}{8}\right) = \left(\frac{W^2_{max}\, MTU}{32}\right) = \frac{D}{4} \tag{4.3}$$

şeklinde gerçekleşir. 4.3'teki eşitlikten çıkan sonuca göre, ikili TCP bağlantının herhangi birisinde gerçekleşen bir kayıp oranı tekil TCP bağlantısıyla gönderilen veride oluşabilecek kayıp oranından dört kat daha düşüktür. Eğer ikili bağlantının her ikisinde de paket kaybı oluşursa bu sefer oluşan toplam kayıp oranı tekil TCP bağlantısında oluşabilecek paket kayıp oranının yarısına eşittir; şekil 4.4.c'de gösterilmiştir. "N" gibi bir değişkeni herhangi bir uygulama için TCP deki bağlantı sayısı olarak tanımlansın ve "n" değişkeni ise aynı anda oluşabilecek tıkanıklık esnasındaki kayıp oranı olarak tanımlansın, işgücü azalma miktarı:

$$D_N = \frac{nW^2_{max}\, MTU}{N^2} \tag{4.4}$$

olur. 4.4 ncü eşitlikten görüldüğü gibi işgücü azalma miktarı uygulama için kullanılan TCP bağlantı sayısının karesiyle ters orantılıdır. Ne kadar az TCP bağlantısı kurulursa TCP işgücünü azalması da hafifletir [36].



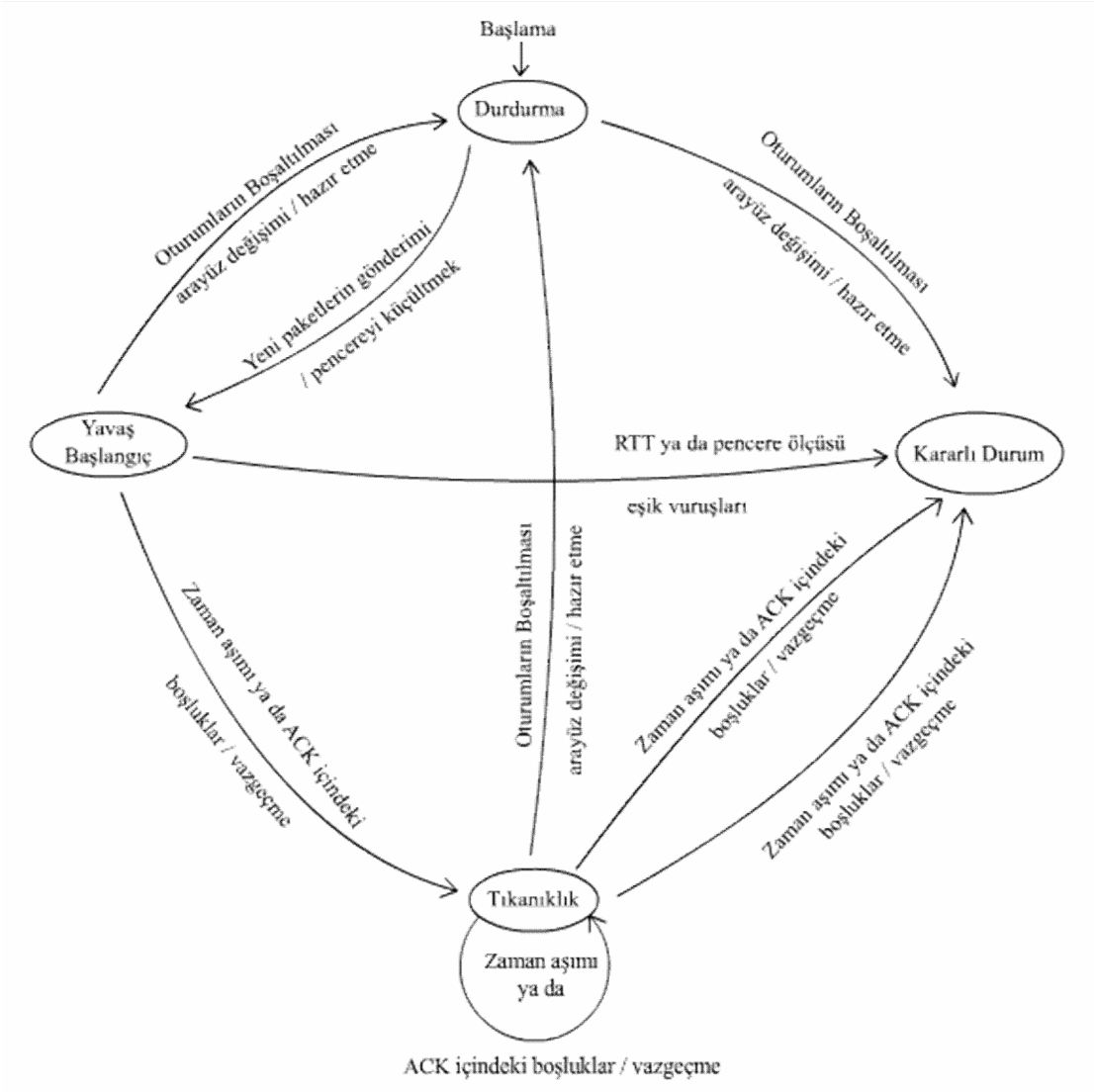
Şekil 4.4. işgücü indiriminin gösterimi (a) tekil kayıplı bir TCP bağlantısı; (b) tekil kayıplı iki TCP bağlantısı ve (c) çift kayıplı iki TCP bağlantısı [36].

4.4. TCP Protokolü Üzerinden Çoklu-Ortamların Akışının Yapılması

Çoklu-ortamların İnternet üzerinden akışının genellikle UDP protokolü üzerinden yapıldığı ve daha hızlı veri transferi için alternatif yollar arandığına değinildi ve bununla birlikte Transfer Kontrol Protokolü (TCP) üzerinden gönderim için metotları geliştirilmektedir. TCP protokolü üzerinden çoklu-ortamların gönderimine değinmeden

önce TCP hakkında kısa bir bilgi verilirse; TCP İnternet üzerindeki en baskın transfer protokolüdür. Bu protokol birçok verinin uygulamalar arasındaki bağlantıyı sağlamakla kalmaz ayrıca veri transferi yolunda herhangi bir tıkanıklığın olması durumunda veri transferini keser veya transfer esnasında herhangi bir darboğazın olmaması durumunda gönderim oranını artırır. Kullanmakta olduğumuz İnternet TCP protokolü üzerine kurulu olduğu için diğer tüm protokollerinde bu kontrol protokolüne uyumlu hareket etmesi gerekmektedir.

TCP üzerinden gönderilen herhangi bir veri paketler halinde iletilirken paketlerden herhangi birisi kaybolursa kontrol protokolü tekrar istekte bulunur ve ardından kayıp paket kurtarılmış olur. Fakat diğer taraftan bu işlem çoklu-ortam dosyalarının gerçek zamanlı akışı için kullanılabilecek uygun bir metot değildir. Bu sorunun aşılması için birçok yöntem geliştirilmektedir. TCP üzerinden ortam akışlarında karşılaşılabilecek sorunlardan biride kullanıcıların güvenlik duvarlarının etkisi altında olmalarıdır; bu güvenlik duvarları HTTP trafiğini düzenler. Bu sorunların aşılması için geliştirilen yöntemlerden biri Akış Kontrol Protokolü (SCP) olarak geliştirilen yeni protokoldür. Bu akış protokolü sıkışıklıkları kontrol ederek hem TCP hem de SCP üzerindeki bant-genişliğini kontrol eder ve akışı daha rahat olacak bir şekilde düzenler.



Şekil 4.5. SCP Transfer Durum Diagramı [37]

SCP iki yapıdan oluşur bunlar: ortam gönderici ve gönderici tarafından verinin alındığına dair network bağlantı alıcısı. Her paket bir sıra numarasına sahiptir ve her paket SCP'de zamanlarına göre sıralanır ve gönderildikleri zamanlar kayıt altına alınır. Alıcıdan gelen her onay bilgisi paketlerle ilişkili olarak ACK paket sıra numarasına göre numaralandırılır. SCP işlemini yukarıdaki şekil 4.7'de gösterilmektedir ve her aşama aşağıdaki tablo 4.1'de kısaca anlatılmaktadır [37];

Tablo 4.1. SCP Durumları, network ve oturum durumları ve tıkanıklık politikaları.

Durum	Network ve Oturum Durumu	Tıkanıklık Penceresi Ayarlama Politikası
Düşük Başlangıç	Uygun bant-genişliği daha bulunmadı	Her ulaşan ACK onayı pencere ölçüsünü genişletir ve böylelikle SCP yeni tıkanıklık pencerelerini açar.
Sabit	Uygun bant-genişliklerinin hepsi kullanımda	Maksimum işgücünün elde edilmesi için SCP network içinde uygun olan tamponların hepsini kullanıma sunar; bu işlemi tamponun aşırı zorlanmasının önüne geçebilmek ve uygun bant-genişliklerini takip etmek için yapar.
Sıkışıklık	Network tıkanıklığı	SCP pencereyi ikiye bölerek çarpımsal bir zorlamaktan vazgeçer. Denemeler sonucu oluşan ısrarcı tıkanıklıklardan vazgeçilir.
Durdurma	Network içinde çok iyi durumda olmayan paketler	Yeni paket gönderildiği zaman, SCP pencereleri küçültür ve yavaş başlangıç aşamasını tekrardan başlatır.

4.4.1. TCP ile SCP protokolleri arasındaki benzerlikler:

- Her ikisi de kullanıcıya tıkanıklık tespitini olumlu onay bilgisi geldikten sonra başlatır ve tıkanıklık kaçınma politikasına göre tıkanıklık penceresini kullanır;

- Başlangıç aşamasında; SCP, TCP gibi davranır ve hemencecik uygun network bant-genişlikleri tespit eder.

4.4.2. TCP ile SCP protokolleri arasındaki faklılıklar:

- TCP protokolünün tersine, ne zaman ki network ta herhangi bir tıkanıklık yoksa SCP oranları birleştirme yoluna girişir.

- SCP networkta kayıp olan paketlerin tekrar gönderimini üstlenmez, böylece beklenilmeyen herhangi bir gelişme olmaz ve bant-genişliği boşuna kullanılmamış olur.

- TCP protokolünde tıkanıklık pencere ölçüsü lineer olarak artarsa, paket kayıpları tespit edilir ve oran yarıya indirilir diğer taraftan SCP elde ettiği bant-genişliğini akışın sonuna kadar koruyarak belli bir orana sahip akış imkânı sağlamış olur.

- SCP mobil işlemlerin dâhil olmasıyla şiddetli bant-genişliği zorlayıcılarına maruz kalır.

SCP protokolünde tekrar gönderim olayı olmadığı için gecikmelerinde önüne geçilmiş olur. SCP çoklu akışlar arasında sabit network bant-genişliği kullanımını ilke edinir. SCP kısıtlı veri oranlarının üstesinden gelebilecek özelliğe sahiptir. Network içi yeniden yerleştirme kabiliyetine sahip olduğundan ötürü mobil hizmetlerin de sisteme bağlanıp bağlanmadığının farkına varılır [37].

4.5. TCP'deki Çoklu-Ortam Tıkanıklık Kontrol Protokolü

TCP üzerinden ortam akışlarının daha hızlı olması için UDP üzerinden ortam akışlarına alternatif olarak geliştirilmeye çalışılan yöntemler vardır. TCP üzerinden geliştirilen yöntemlerden biri de TCP-Dostça Oran-tabanlı Kontrol (TFRC) mekanizması daha duyarlı bir yapı olarak akışı kontrol eder. Günümüzde Veri-Yolu Tıkanıklık Kontrol Protokolü (DCCP) TFRC kontrol mekanizmasına adapte edilerek ani değişikliklerin önüne geçer.

TFWC, TCP-Dostça Pencere-tabanlı Kontrol Mekanizması TCP işgücünün kullandığı tüm denklem hesaplamalarının aynısını kullanırken, diğer taraftan TFRC, oran-tabanlı kontrol mekanizmasından gelen hesaplamaları kullanır. TFWC göndericileri tıkanıklık penceresini (*cwnd*) gönderdiği veri oranlarında kullandığı denklemlerle hesaplar ayrıca TFRC gibi TFWC kontrol mekanizması da ortam verilerinde kaybolan herhangi bir veri varsa onları tekrardan göndermez. Basitleştirilmiş TCP denklemlerini görmek için [46] bakabilirsiniz, bu denklem kullanılarak veri gönderimi için kullanılan paket pencerelerini türetebiliriz ve böylece eşitliğin her iki tarafını $t_{RTT}/_{s}$ eşitliği ile çarparsak yeni bir *cwnd* hesaplaması ortaya çıkartılır. Böylece;

$$W = \frac{1}{\sqrt{\frac{2p}{3}} + \left(12\sqrt{\frac{3p}{8}}\right) p\,(1+32p^2)} \tag{4.5}$$

eşitliği ortaya çıkar. Denklemde W pencere ölçüsü ve p ise paket sayısını simgelemektedir. TFWC göndericileri tıpkı TCP göndericileri gibi davranır, ilk paket kaybolmayıncaya kadar herhangi bir raporu ilgili birime göndermez. Bu aşamadan ack (onay mesajı) gelene kadar TFWC ortalama kayıp aralığını hesaplar. Kayıp oranlar (4.5) denklemden hesaplanır ve yeni bir *cwnd* değeri ortaya çıkar; ardından eğer sırada bekleyen yeni verilerin değeri (4.6. Denklemden çıkan sonuç) tatmin eder bir değer olarak ortaya çıkarsa, yeni verilerin gönderimi başlanır.

$$Sıradaki\ yeni\ veri\ numarası \leq cwnd + onaylanmamış\ mesaj\ sıra\ numarası \tag{4.6}$$

ACKmesajları her defasında TFWC göndericisine gelir ve mesajlar güncellenir; böylelikle tekrar gönderim zaman aşımı değerleri de güncellenmiş olur. Yenilenen zaman aşımı değerleri ile tekrar gönderim zamanı da yeni veriler için her defasında güncellenir ve yeni paketler böylelikle bu zamanlamaya göre gönderilmiş olur. Eğer zaman değeri şimdiki veri için sona ererse, sıradaki yeni veri paketi ardından gönderilir [47].

Sonuç olarak ortam dosyalarının İnternet üzerinden aktarımı sağlanırken genellikle UDP protokolü kullanır, ortam verilerinin bu protokolü kullanmasına alternatif olarak gerçekleştirilen çalışmalar göstermiştir ki farklı veri transfer protokolleri kullanılarak ta ortam verilerinin akışı sağlanabilir. Bu akış protokollerinin en başında İnternetteki veri

transferinin bel kemiğini oluşturan TCP protokolü gelmektedir. TCP protokolü ortam verisi gibi sürekliliğin sağlanması gereken veri akışları için uygun bir protokol değildir. Bu nedenle TCP protokolünün dezavantajlarını minimize edecek ve ortam akışları için uygun bir hale getirecek farklı çalışmalar bulunmaktadır.

TCP üzerinden veri transferi gerçekleştiği zaman sabit bir hızda veri transferi gerçekleşmez, bu nedenle TCP protokolünün yapısı iyi incelenerek geliştirilen yöntemlerden biri ortam dosyalarını farklı oranlarda sıkıştırılma tekniğidir. Farklı oranlarda sıkıştırılan ortam dosyaları bant-genişliğinin düşmesi esnasında da rahatlıkla iletilebilir.

Daha önce UDP protokolünün ortam dosyalarının transferinde kullanılan en yaygın protokol olduğuna değinmiştik diğer taraftan TCP protokolünün bazı dezavantajları göz ardı edilirse veri transferinde UDP protokolüne göre daha hızlı olduğundan da bahsetmiştik. Çalışmalar sonucunda UDP ve TCP protokollerinin beraber kullanımı ile geliştirilen yeni protokoller sayesinde ortam gönderimleri daha hızlı ve efektif bir şekilde gerçekleşir.

TCP üzerinden ortam verilerinin akışı esnasında oluşabilecek herhangi bir tıkanıklık sonucunda veri aksaması olmaması için birden fazla TCP bağlantısı prensibiyle gerçekleştirilen yeni metot ile düşük performanslı çalışan ağlarda bile etkin sonuçlar alınmaktadır. Diğer taraftan oluşabilecek kesintilerin tespiti için tıkanıklık kontrol algoritmaları ile kayıp olan veri paketleri hızlı bir biçimde bulunup veri transferi kaldığı yerden devam eder.

5. KABLOSUZ AĞ SİSTEMLERİNDE GERÇEKLEŞEN ORTAM AKIŞLARI

Kablosuz ağ yapıları incelendiğinde ilk olarak gerçek zamanlı veri gönderimine ihtiyaç duymayan uygulamalar için örneğin e-mail, ftp ve bazı web uygulamalarına yönelik hazırlanmıştır. Her geçen gün gelişen ihtiyaçlar bu yeni teknolojinin farklı alanlarda da kullanılmasını zorunlu hale getirmiştir. Kablosuz ağ yapısını tasarlayanlar farklı kullanıcıların farklı özellikte sisteme erişim imkânlarından ötürü birçok zorlukla karşılaşırlar. Bu zorlukların temelinde heterojen yapıdaki kullanıcılar bulunmaktadır, örneğin görüntü çözünürlük ihtiyaçları, güç durumları, iletişim imkânları ve kullanıcı bilgisayarların farklı hesaplama ölçütleri gibi. Başarılı bir sistemin oluşturulması için tasarımcıların heterojen yapıdaki kullanıcıların ihtiyaçlarını karşılayacak bir sistemi oluşturması gerekmektedir.

İnternet üzerinden gelişen en popüler uygulamalardan biri de İnternet üzerinden görüntülü ya da sesli iletişim uygulamalarıdır. Bu uygulamanın kablosuz ağlar üzerinden uygulamalarına da VoIP uygulamaları denir. Kablosuz ağ yapısının MAC çalışma prensibiyle çalıştığı için bazı gecikmeler oluşur. Bu gecikmeler gerçek zamanlı olan ya da olmayan uygulamalarda kuyruk stratejisinden kaynaklanmaktadır. Kablosuz ağlar cep telefon hizmetlerinde gelişen teknolojilerin örneğin 3G ve 4G teknolojileri gibi ihtiyacını da karşılamak zorundadır. Kullanıcıların cep telefonu ya da cep bilgisayarı üzerinden internet sitelerine girip ortam dosyalarını izlemeleri ya da canlı TV izlemeleri gibi uygulamalardır.

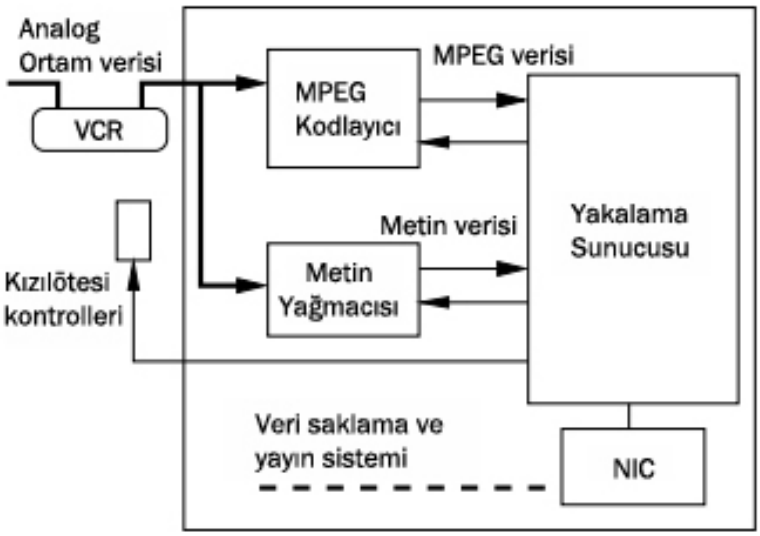
Kablosuz ağ kapsamındaki kullanıcılara ortam dosyası gibi uygulamaların verilmesinde bazı zorlukların aşılması gerekmektedir. Bu zorluklar arasında bilginin sunucuda saklanması, istenilen bilginin kullanıcıya gönderimi veya istenilen bilginin kullanıcıya iletilmesi için temini gibi zorluklardır. Kablosuz ağlarda ortam akışının daha hızlı gerçekleştirilmesi için geliştirilen ölçeklenebilir akış stratejisi yapısına sahip sistem ile hazırlanan veriler daha güvenli ve efektif bir şekilde iletilir.

5.1. Kablosuz Ağlarda Gerçek Zamanlı Ortam Akışları

Günümüz de İnternet üzerindeki en popüler uygulamalardan biride İnternet üzerinden yayın yapan TV uygulamalarıdır. Bu uygulamaların izlenmesi için kullanıcılar bilgisayarlarına Paket Derleyicisi/Ters-Derleyici (PDA) cihazları kullanarak İnternet yardımıyla kendi bilgisayarlarında canlı yayın izleme imkânını elde ederler. Çoğu zaman iş amaçlı kullanılan bu uygulamaları en fazla kullanan kitle ise canlı yayında gelişmeleri takip eden borsacılar 24 saat haber yayını yapan CNN ya da CNBC gibi kanallarını takip ederler. Kullanıcılar WiVision sistemi yardımıyla bu yayınları mobil olarak izleme imkânını elde ederler [40, 70]. Bu zorluklara örnek verecek olursak böyle bir sistemin oluşturulması için bazı zorlukların aşılması gerekmektedir; Örneğin, çoklu-ortamların elde edilmesi, bilginin saklanması ve sonra kullanıcılara gönderilmesi gibi.

5.1.1. Sistem Mimarisi

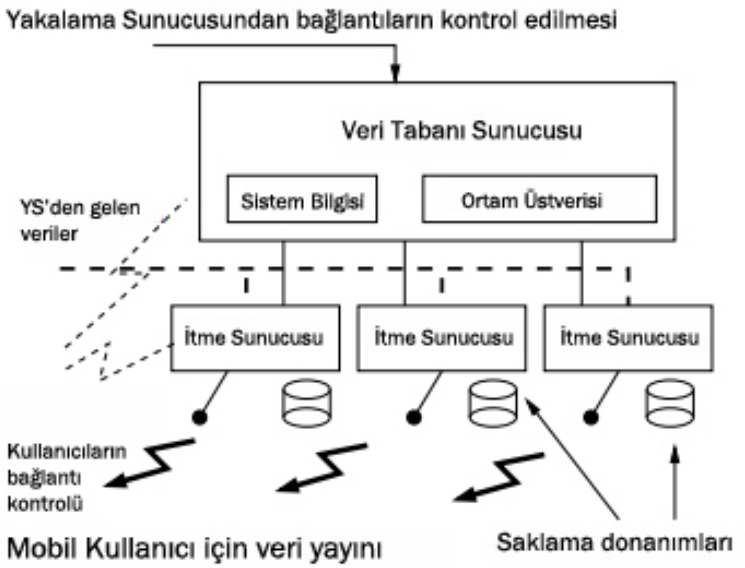
Ortam dosyası Temin Sunucusu ortam verilerini analog sinyalden dijital gerçek zamanlı sinyale çevirir ve gönderimini yapar. Analog sinyallerin temini için kablolu TV gibi kaynaklar kullanılır. Gerçek-zamanlı sayısallaştırma ve sıkıştırma işlemi için özel donanımlara ihtiyaç duyulmaktadır. Ortam sinyalleri görüntüye eklenmiş olan metin yazılarıyla da ilgilenmek zorundadır, senkronize edilmiş yazı metni içeren ortam dosyası akışı esnasında yazı metinleri zaman damgasıyla beraber işlenir. Bu zaman pulları kullanıcılar tarafından daha sonra senkronize edilmiş yazı metinlerinin gösterilmesi için kullanılır. Ortam Temin Sunucusu veriyi gelecekte geri-oynatım olması olasılığına karşın saklama sunucusuna gönderir. Bu veri güvenilir bir şekilde gönderilmesi gerekmektedir ki böylece kullanıcı Temin Sunucusu ile Saklama Sunucusu arasında transfer mekanizması olan TCP'yi seçebilsin. Temin Sunucusu ve bileşenleri aşağıdaki şekil 5.1'de gösterilmektedir.



Şekil 5.1. Temin sunucusu ve bileşenleri [40]

WiVision sisteminin Ortam Sunucusu alt-sistemi ortam akışların saklanması, yayını ve yönetilmesinden sorumludur. Ortam sunucusuna şifrelenmiş veri ve Temin Sunucusundan gelen görüntüye işlenmiş metin verileri gelir, gelen verilerin tekrardan işlenmesi ve veri merkezinde saklanıp gerçek zamanlı dağıtılması işlemleri yönetilir. Sunucu ayrıca mobil kullanıcıları için isteğe saklı verilerin tekrar oynatım ihtiyaçlarını da karşılar [70].

Ortam sunucuları genellikle iki bileşenden oluşmaktadır; Üst-veri Sunucusu ve İtme Sunucusu. Üst-veri sunucusu genellikle üst verilerin yönetilmesi işlemlerinden sorumludur, örneğin ortam klip bilgileri, sunucu bilgileri vb. İtme sunucuları ise verilerin saklanması, yayınlama aşamasında sunucuya akışı işlemlerini yönetir. Sunucu aşağıdaki şekil 5.2'de gösterilmektedir.



Şekil 5.2. Ortam sunucusu ve alt sistemleri [40]

Mobil kullanıcılara gerçek-zamanlı veri transferi akışının kablosuz bağlantılarla yapılması işlemi WiVision uygulamalarının en çok üzerinde durdukları konudur. İtme sunucuları genellikle verilerin yayınlanmasından sorumludur. Ayrıca network ağı üzerindeki bağlantı noktalarını da birbiriyle ilişkilendirmek zorunlulukları bulunmaktadır. Diğer taraftan da gerçek zamanlı veri akışını sağlaması gerekmektedir. Üç farklı veri yayını yapma seçeneğine sahiptir; bunlar, tek-yönlü transfer, çok-yönlü transfer ve tüm kullanıcılara birden yayın yapmak. Veri transferi işlemini WLAN altyapısı gerçekleştirmektedir. Maksimum Wi-Fi network bant-genişliği 11 Mbps ve gözlenen bant genişliği sadece 6 Mbps'dir. MPEG-1 dosyası için ihtiyaç duyulan bant-genişliği ise sadece 1.5 Mbps tir. Bu durumda çok-yönlü bir veri transferi olması durumunda istatistiksel olarak maksimum dört kullanıcı sisteme bağlanıp ortam dosyasını izleyebilir. Bununla birlikte kullanıcılara Servis Kalitesi garantisi verilmemesi böyle bir senaryonun kabul edilmesini mümkün kılmaz. Veri transferinin doğru seçimi çok-yönlü adreslemeyi gerektirir; bu adresleme sisteminde çok-yönlü gruplarda her bir kanal için ilgili oluşturulur ve mobil kullanıcılar sisteme bağlanıp, sistemin gereksinimleri dâhilinde ayrılabilirler. Bu maksimum kullanıcı sayısında herhangi bir sınırlama olmaksızın herhangi bir kablosuz noktasında dört farklı kanal ile birlikte bağlanmasına olanak sağlanır.

Gerçek zamanlı bir ortam akışının sağlanması için kablosuz network ağ sisteminde en son noktada bulunan ortam kullanıcılarına TV izleyicileri gibi kesintisiz hizmetin verilmesi gerekmektedir. Gerçekleşecek canlı ortam akışı için seçilecek olan belirli bir kanal İtme Sunucusundaki kapıların ortam yayını için dinlenmesi gerekir. Kullanıcı ortam bilgisini metin olarak okuyup ya da ekran görüntüsünü görüp tıklayabilir. Kullanıcıların bilgisayarlarında kullandıkları işletim sistemine göre farklı platformlar ortam dosyalarının açılması için kullanılır; örneğin Windows işletim sistemi için Windows Media Player (WMP) gibi. İnternet üzerinden WMP platformunun kullanımı bazı sıkıntıları da beraberinde getirir. WMP Microsoft Media Server (MMS) protokolünü kullanması ya da HTTP web sunucusu üzerinden akışının yapılması gerekmektedir. Diğer taraftan İtme Sunucusu yayın için UDP protokolünün kullanımını ve tek-yönlü geri-oynatımı için tasarlanmıştır. UDP protokolü WMP kullanımı için uygun değildir. Bunun dışında WMP, MPEG akışları için özel başlıkları içermesi gerçek-zamanlı akış için uygun olan bir tasarım değildir [40, 70].

5.2. MAC Standardı

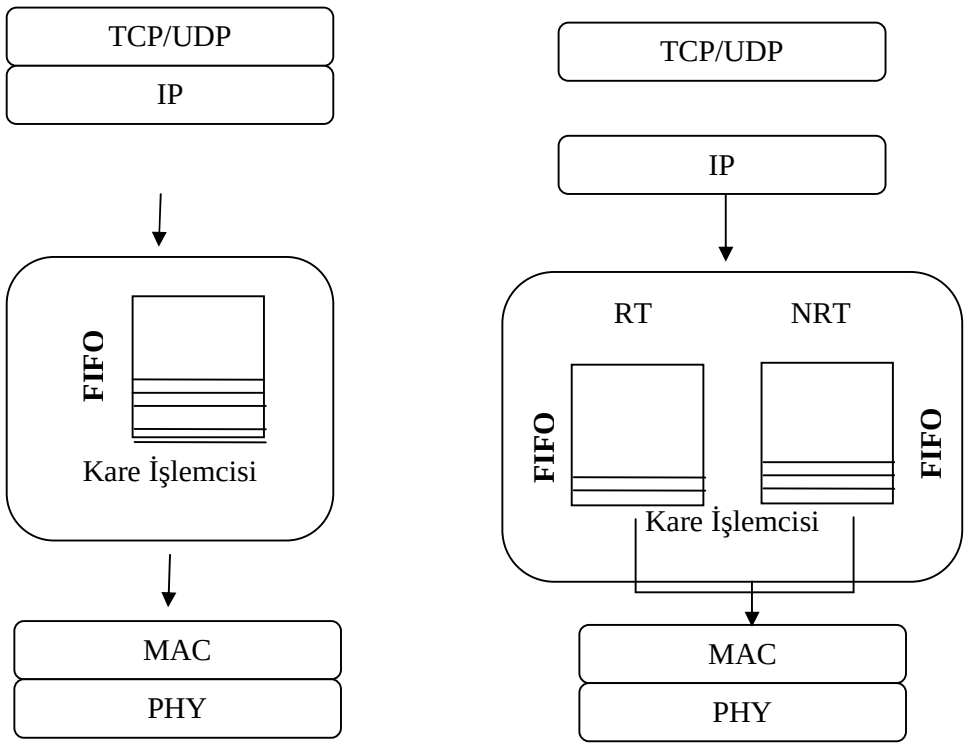
Kablosuz ağların ilk çıkışı gerçek zamanlı bir veri akışına ihtiyaç duymayan e-mail, ftp ve bazı web uygulamaları gibi uygulamalar içindir. Bu durum yerel ağlarda QoS sağlanmamasına yol açar. 802.11 MAC standardı DCF (Dağıtımlı Koordinasyon Fonksiyonu) tabanlı çalışır.

Kablosuz ağlarda kullanılan MAC standardında iletim kuyruğuna bir paket ulaştığına kanal dinlenmekte, eğer kanal boş değilse kanal boş olana kadar beklenmektedir. Kanal boşaldığında ise DCF kareler arasında boşluk olarak tanımlanan DIFS zaman aralığı kadar bekler bu bekleme sırasında hala kanal boş ise bu kanaldan vazgeçilir. Bu işlem için bir vazgeçme zaman aralığı belirlenir ve kanal her boşaldığında vazgeçme zaman aralığı sayacı bir azaltılır. Sayaç sıfır olduğunda ise istenilen veri gönderilir.

Kablosuz ağlar MAC çalışma prensibiyle çalıştığı için VoIP uygulamalarında gecikmelere neden olurlar; kısaca açıklamak gerekirse gerçek zamanlı uygulamalar ve gerçek zamanlı olmayan uygulamalar kuyruk stratejisi nedeniyle (FIFO) aynı öncelikte

kullanılmaktadır; bundan ötürü gecikmeler meydana gelmektedir. Kablosuz ağlarda gecikmelerin önüne geçmek için çift kuyruk yaklaşımı geliştirilmiştir.

Çift kuyruk algoritmasında kablosuz ağ kartı sürücüsünde iki ayrı kuyruk yapısı oluşturulur; oluşturulan iki ayrı kuyruk öncelik faktörünü dikkate alır ve kuyrukların birinde sadece gerçek zamanlı uygulamalar barındırılırken bir diğerinde ise gerçek zamanlı olmayan uygulamalar barındırılır. Gerçek zamanlı paketler kuyrukta tamamen gönderilmeden önce bir diğerine izin verilmez, böylelikle gerçek zamanlı uygulamalarda VoIP uygulamaları gibi zaman gecikmesinin önüne geçilir. Aşağıdaki şekil 5.3'de kart sürücüsünün bu yaklaşım uygulanmadan önceki ve yaklaşım uygulandıktan sonraki yapısı görülmektedir.



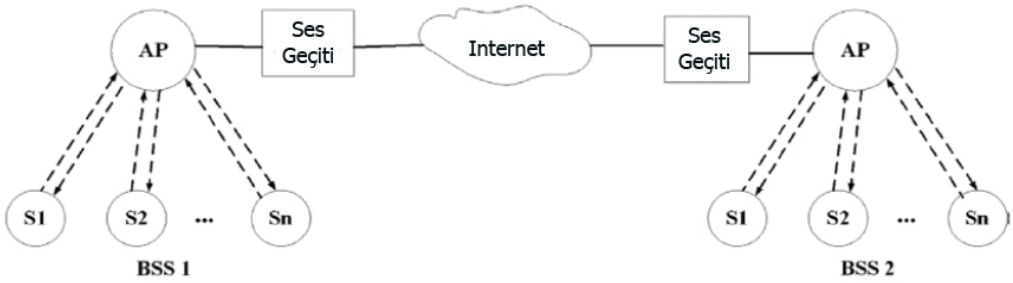
a) Orijinal iletim yöntemi **b)** Değiştirilmiş iletim modeli

Şekil 5.3. Kart sürücüsü yapısı [24]

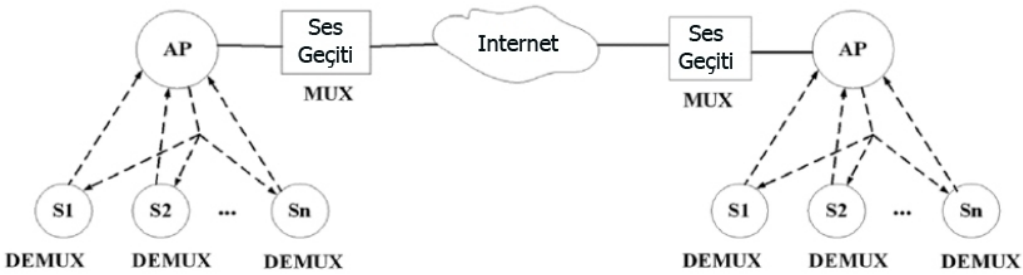
Çift kuyruk algoritmasının işe yaramadığı zamanlarda Adil MAC dağıtık yönteminin kullanımı tercih edilir. MAC tekniğinde kullanılan ve küçük paketler için boşa

geçen zaman dilimleri çok büyük sorun oluşturur. Çünkü IEEE 802.11 MAC protokolü paket boyutlarını dikkate almadan gecikme sayacını belirler. Bu yaklaşımda önerilen bir terminal paket gönderim hakkı kazandığında sadece tek bir paket yollayıp tekrar DCF mekanizmasını çalıştırmamakta bir paket kümesi gönderme şansı yakalamaktadır. Fakat terminaller arası adaletin sağlanabilmesi için yani bir terminalin kanal kullanım hakkını alınca bunu sürekli kullanmaması için her terminale maksimum iletim zamanı verilmekte ve bu da iletim zamanı sonunda terminal paket gönderme işlemine ara vermektedir.

VoIP uygulamaları kablosuz ağlarda iletildiğinde iki önemli sorunla karşılaşır. Bu sorunlardan ilki bir VoIP oturumu yaklaşık 10kb/s'lık bir kanal kapasitesi kullanan ve ayrıca yerel ağların yaklaşık 11Mb/s lık bir kapasiteye sahip olması protokol yükleri sebebiyle sadece 12 (Gsm 6.10 için) oturumunu desteklemesidir. Diğer bir sorun ise VoIP performansının ağda var olan diğer paketlerin trafiğinden olumsuz şekilde etkilenmesidir. Bu sorunların çözülmesi amacıyla bağlantı noktaları (AP) MAC protokolünde değişiklik ve kanalın VoIP kapasitesini artıran yeni bir yöntem önerilmiştir. Sadece AP'de değişimin önerilmesinin sebebi son kullanıcılardaki 802.11 protokolüne uyumlu donanımların değiştirilmesi maliyet açısından daha düşüktür. Kablosuz bağlantılarda kullanılan ve VoIP uygulamaları için çok-yollu iletim yöntemi VoIP uygulamalarının iletimi esnasında MUX tarafından Voice GateWay'de geçerken birleştirilmesi yöntemine dayanmaktadır. İletim aşamasında her pakete ayrı bir ID numarası verilir, iletilmek istenilen paketler tüm terminallere gönderilirken alıcı durumdaki terminaller gelen paketlerin kendilerine ait olup olmadığını paketlerin ID'lerine bakarak çıkartır. Şekil 5.4'de kullanılan klasik kablosuz ağ bağlantı yöntemi ile önerilen çok-yollu kablosuz ağ yönteminin karşılaştırılmasını gösterilmektedir [24].



(a)



(b)

Şekil 5.4. (a) kullanılan VoIP için trafik akışı **(b)** çok-yollu VoIP trafik akışı [24]

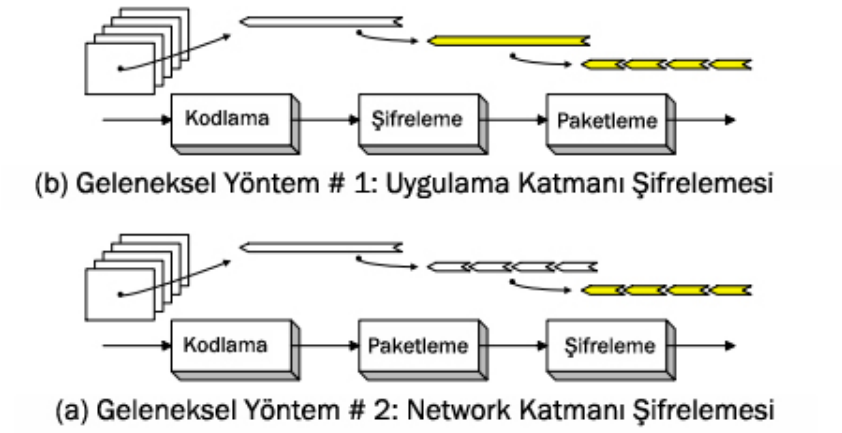
5.3. Kablosuz Ağ Üzerinden Güvenli Ölçeklenebilir Ortam Akışı

Kablosuz veri akışları sistem tasarımcıları için birçok zorluk barındırmaktadır. Örneğin kullanıcı farklı görüntü, güç, iletişim ve hesaplama yeteneklerine sahip olabilir. Bununla beraber kablosuz iletişim bağlantıları farklı maksimum bant-genişlikleri ve kalite katmanlarına sahip olabilir. Başarılı bir kablosuz ortam akış sistemi heterojen kullanıcılar için zaman değişkenli kablosuz iletişim bağlantıları ile yapılmalıdır. Ölçeklenebilirlik sistemi ortam akışını farklı cihaz kapasitesine sahip kullanıcılara akış olanağı sağlamalıdır. Efektiflik kullanıcılara networkun ve cihaz kaynaklarının maksimum kullanımını demektir. Güvenlik ise kablosuz network ağ sisteminin veri içeriğinin herhangi bir çalınma yöntemine karşı korumadır; örneğin Paket Koklaması gibi.

Bu bölümde Geliştirilen Güvenli Ölçeklenebilir Akış (SSS) sistemi ile kablosuz ağ sistemlerinde veri transferinde ölçekleme, efektiflik ve güvenli veri transferi imkânları sağlanarak veri akışının yapılması üzerinde duruldu. SSS ortam dosyasını güvenli ölçeklenebilir paketler şeklinde şifreler ayrıca farklı kodlama dillerine de çevirerek ara network düğümlerinde düşük karmaşıklık işgücüyle hiçbir çözümlemeye ihtiyaç duymadan veri paketlerinin işlenmesine imkân verilmiş olur.

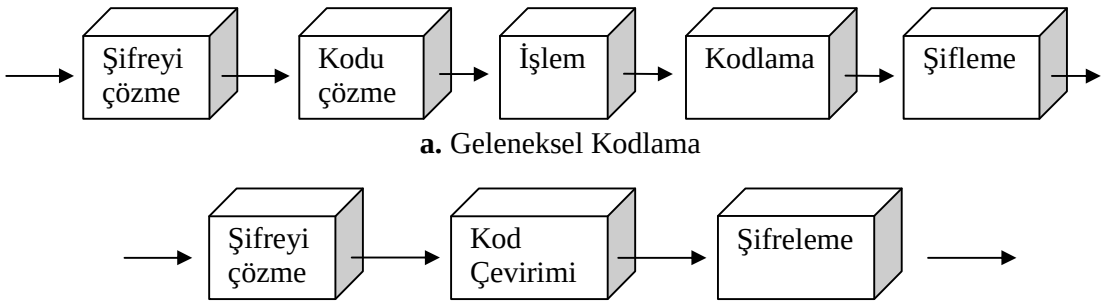
Şekil 5.5.a'da uygulama-katmanında şifrelenmiş güvenli ortam akışı gösterilmektedir. Ortam ilk olarak *interframe* sıkıştırma algoritmaları örneğin MPEG ya da H.263 ya da *intraframe* sıkıştırma algoritmaları kullanılarak bit-akışı için şifrelenir. Bit-akışı sonucunda şifrelenmiş ve şifrelenmiş akış sonucunda da paketlenmiş ve network transfer protokolü örneğin UDP gibi, kullanılarak transfer edilir. Bu işlemin güçlüğü veri paketlerinin kaybında görülmektedir. Özellikle, hata kurtarma zordur çünkü kayıp paketten gelen verilerin olmamasıyla, şifreleme çözme ve / veya kod çözme imkânsız olmasa bile zordur.

Şekil 5.5.b'de network-katmanı sıkıştırma ile güvenli ortam akış sistemini gösterilmekte. Bu sistem bir önceki sistemde kullanılan ortam sıkıştırma algoritmalarını kullanırken, diğer taraftan bu sistemde paketleme kodlu video içeriği, gördüğü bir şekilde yapılabilir ve böylece hata kurtarma daha iyi bir şekilde gerçekleştirilir. Örneğin ortak yaklaşım UDP tarafından yapılan RTP transfer protokol ile MPEG sıkıştırmayı kullanır. RTP zaman damgaları gibi akış parametrelerini sağlarken diğer taraftan paketleme için MPEG veri yüklemesi metotlarını da önerir.



Şekil 5.5. Geleneksel ortam akış yaklaşımları [39]

Her iki yaklaşımda ortam verilerinin sıkıştırılmış bir biçimde güvenli veri transferi yapar; fakat eğer network kod-çevirimine ihtiyaç duyarsa, şekil 5.6.a'da gösterilen yöntemin kullanılması gerekmektedir. Kod-çevirimi işlemi bir çözümleme, şifre-çözme, işleme, tekrar kodlama ve tekrar şifreleme işlemlerinden oluşur. İşlemin hesaplama ihtiyacını düşürmek amacıyla kod-çevirimi işlemindeki şifre-çözme, işleme ve tekrar kodlama modüllerinin birleştirilmesi gerekmektedir. Diğer taraftan kod-çevirimi algoritmaları birçok network düğümündeki kod-çevirimi işlemlerinin sağlıklı işlememesinden ötürü hesaplama ihtiyaçlarını geliştirmeleri gerekmektedir, öyle ki birçok kritik eksiklik her kod-çevirimi işlemindeki temel şifre-çözme ihtiyaçlarının karşılanmasını engeller. Her akışın şifresi çözümleme aşamasında başka bir muhtemel saldırı noktası açılması bu sistemde güvenlik zafiyeti oluşmasına sebep olur. Böylece, her kod-çözümleme işlemi sistemde daha genel bir güvenlik tehdidinin oluşmasına yol açar.

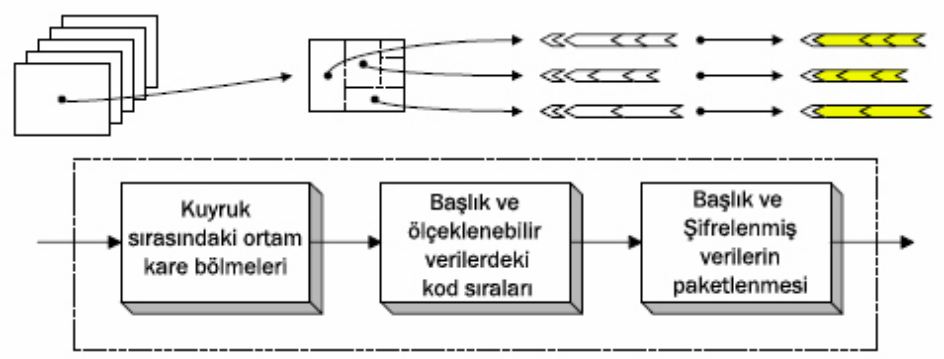


a. Geleneksel Kodlama

b. Geliştirilen Yöntemlerle Geleneksel Kod-Çevirimleri

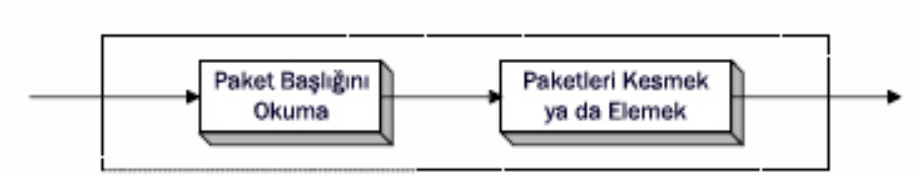
Şekil 5.6. Kod-çevirimlerinde geleneksel yaklaşımlar [39]

Güvenli Ölçeklenebilir akışta veri kodlaması şu şekilde oluşur. SSS kodlaması giriş ortam karelerini güvenli ölçekleme ile kodlanır, bu kodlama sayesinde kablosuz network heterojen kullanıcılara daha rahat bir akış yapılır. SSS kodlaması ortak sıkıştırma, paketleme ve kodlama şifreleme modül tasarımları ile geliştirildi. Özellikle ölçeklenebilir kodlama ve paketleme modülleri ile aşamalı şifreleme tekniklerinin birleşimi yapılır. SSS kodlaması ile ortam akışı kod-çevirimi işlemlerinde sonradan gereksinim duyulan bit-oran düşürümü ve özel resim boyutu küçültme taktikleri aşamalarının herhangi bir ortam çözümlemesi olmadan yapılabilmesi özelliği kazandırılır.



Şekil 5.7. SSS kolama işlemi [39]

Standart Güvenli ortam kodlaması teknikleri her ölçeklenebilir ortam verisi için ayrı bir kod kullanılırken geliştirilen yeni SSS kodlama metoduyla ilk olarak ortam kareleri bölümlere ayrılır her bölümde iki parça halinde kodlanır; bunlar başlık verisi ve ölçeklenebilir ortam verileridir. Bu işlemin ardından ölçeklenebilir ortam verisi şifreleme tekniklerine göre şifrelenir ve son olarak aşamalı şifreleme ölçeklenebilir ortam verisinden gelen şifrelenmemiş başlık verisi eklenerek sonlandırılır. Güvenli ölçeklenebilir paketleme işleminin sonucunda veri akışı network üzerinden gerçekleştirilir.



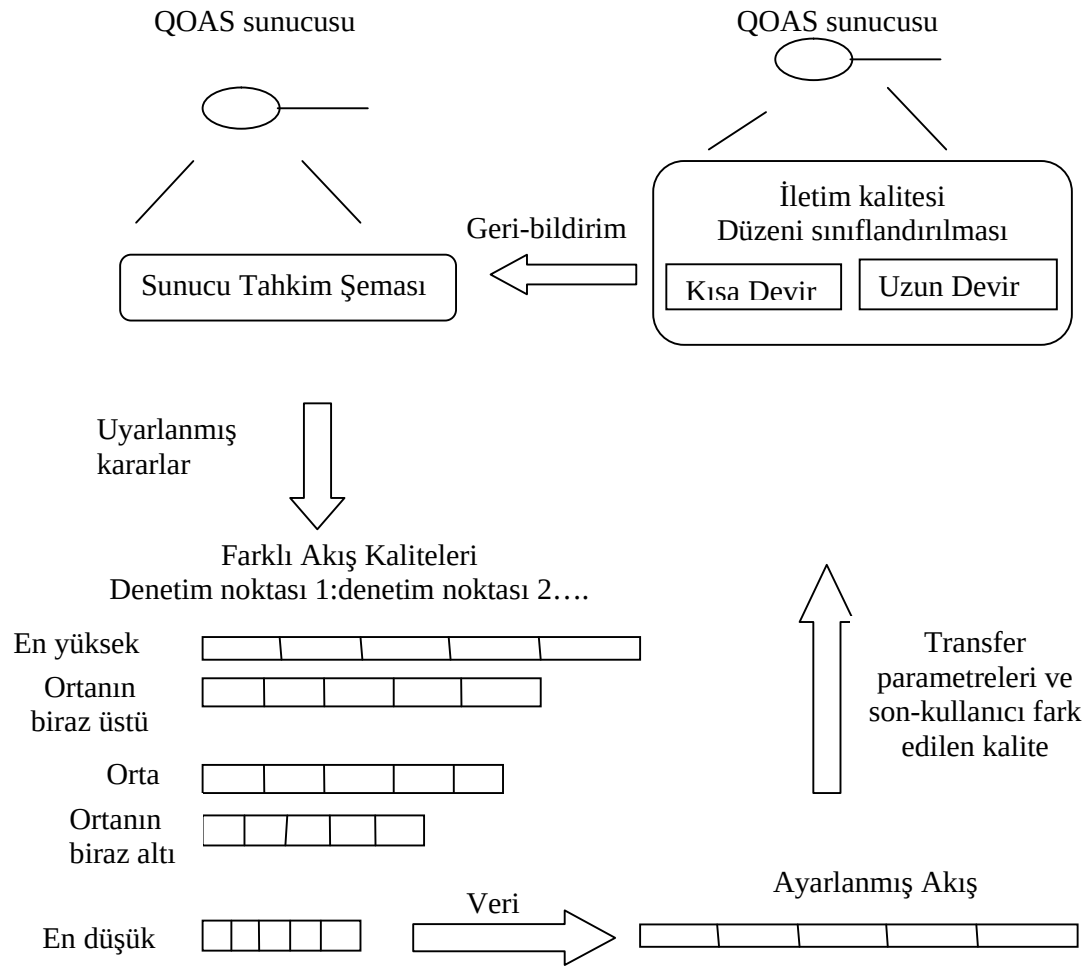
Şekil 5.8. SSS kod-çevirimi işlemi [39]

SSS kodlamada, ölçeklenebilir kodlama ve paketleme modülleri şifreleme modüllerinin kombinasyonundan oluşur. Bu kombinasyonda sonradan gelen SSS kod-çevirimleri işlemlerine izin verir. SSS kod-çevirimi işleminin sonucu yukarıdaki şekil 5.8'de gösterilmektedir. SSS kod-çevirimleri her paketin başındaki şifrelenmemiş başlık verilerine göre ardından şifrelenmemiş başlıklar tarafından belirlenen uygun bölümlerdeki ucu kesilmiş paketlere göre okuyup kod-çevirimlerini yapabilir. Klasik kod-çevirimi yöntemlerine göre yapılan kod-çevirimleri ile SSS yöntemiyle yapılan kod-çevirimi işlemlerinin gösterimi yukarıdaki şekil 5.7'tedir [39, 72].

5.4. Kablosuz Ev Alan Ağında Çoklu-Ortamların Kaliteli Akışı

5.4.1. Kalite odaklı uyum şeması (QOAS): Çoklu-ortam akışlarında kullanıcıların Kalite Deneyimini (QoE) artırmak için kullanılan tek yönlü oran tabanlı şemadır. Bu şemada kullanıcı ve sunucu merkezli bileşenler bulunmaktadır; bu bileşenler hem çift yönlü ortam verilerini içerir hem de network teslimatı için paket kontrolünü içerir. Kullanıcı bir taraftan transferi kontrol ederken diğer taraftan Kaliteli Teslimat Programı Değerlendirmesinde (QoDGS) kullanılan QoE-ilişkili parametreleri kullanır. QoDGS genellikle kaliteli teslimat değerinin hesaplamasını yapar. Sunucu Tahkim Şeması (SAS) bu değerleri analiz eder ve var olan teslim durumlarındaki artan kullanıcı QoE değerlerini sağlamak için uygun kararları önerir [73].

QOAS farklı sunucu durumları için birer numara tanır ve bu numara işlemi her akış oturumu için farklı akış kalitesi belirlemek için kullanır. Akış kalitesi sürümleri sıkıştırma ilişkili parametreler açısından farklılık gösterir örneğin çözünürlük, kare oranı, renk derinliği gibi ve böylece farklı bant-genişliklerine de ihtiyaç duyulur. Transfer esnasında sunucu dinamik olarak kendi durumunu kullanıcının QoDGS geribildirimine göre değiştirir. Örneğin, son-kullanıcı kalite değerinin düştüğünü rapor ederse, sunucu hemen kalite durumunu düşürür diğer bir değişle gönderilen verilerin miktarını azaltır. Gelişmiş durumlarda ise sunucu ulaşılan akış kalitelerini kademeli olarak artırır. Şekil 5.9’da QOAS adaptasyon prensibinin çalışmasını gösterilmektedir [41].



Şekil 5.9. QOAS adaptasyon prensibi şematik gösterimi [41]

Kullanıcı yerindeki QoDGS sonlu-kullanıcı algılama kalitesindeki teslimat durumunun etkilerini izler ve değerlendirir. Sınıflandırma işlemi kayıp paket oranı ve gecikme gibi hem kısa-terimli hem de uzun-terimli değişkenlerin hesaplanmasına dayanır. Kısa-terimli değerlendirmeler geçici etkilerin hızlı öğrenilmesi için önemlidir; örneğin ani trafik değişiklikleri gibi. Uzun-terimli değişkenler ulaştırılan tüm çevrenin yavaş değişken takibinin hesaplanması için kullanılır; örneğin herhangi yeni bir kullanıcının sisteme dâhil olup olmadığını kontrol etmesi gibi. Sunucu yerindeki Sunucu Tahkim Şemasını (SAS) kullanıcıdaki ardışık QoDGS değerleri ve bu değerlerin ortalaması ile alınan asimetrik ayar kararlarıyla ilgilenir [41, 73].

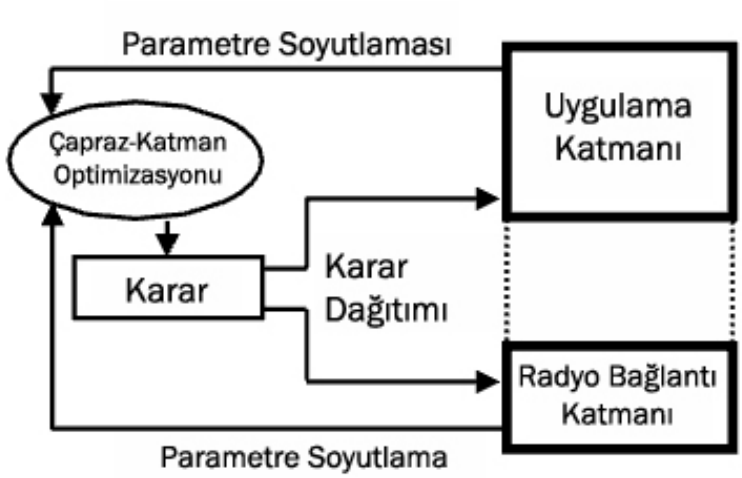
5.4.2. QOAS konumlaması: Çoklu ortamların kablosuz ev ağları üzerindeki akışı esnasındaki QOAS performansının test edilmesi için sunucu tarafındaki QOAS bileşenleri akıllı evlerin erişim noktalarına yerleştirilir. Teslimat koşullarının son derece dinamik varyasyonlarına yeterince hızlı uyarlamalı ve tepki vermek amacıyla kullanıcıdan her defasında doğru bilgi alınması gerekmemektedir. Böylece QOAS düşük geri-dönüşüm rapor paketleri ile çok yüksek geri dönüşüm frekansını sağlayabilir; bu düşük geri-dönüşüm rapor paketleri her 100 milisaniyede bir gönderilir. Bu değer düşük masraf ihtiyacı ile ihtiyaç duyulan en çok güncel bilgilerini dengeler. Her iki QoDGS kısa-terimli ve uzun-terimli değerler periyodik olarak izlenir ve bir dizi halinde sıralanır. Uyarlamalı sonuçların hemen alınması gerekmektedir ve böylece SAS iyileştirici periyotlar her 6 saniyelik periyotlar halinde güncellenebilsin [41, 73].

5.5. Kablosuz Çok Kullanıcılı Çapraz-Katman Optimizasyonu

Mobil iletişim sistemleri için günümüzde geliştirilen en son yenilik Çapraz-Katman uygulamasıdır. Bu uygulamanın tasarımıyla çoklu-ortamların içeriğiyle ilgilenen servislerin daha rahat konfigürasyonu yapılır. Çapraz-Katman tasarımının konsepti, ara katman bilgilerini iki ya da daha fazla katman arasındaki iletişim optimizasyonunu protokol demetleriyle değiş-tokuş etmekle yükümlüdür. Bu uygulama ayrıca diğer tüm iletişim ağlarında da sağlanırken, esas odak noktası kablosuz ortamın benzersiz yapısından dolayı ağ sistemleriyle ilgili çalışmalarda kullanılır.

Bu bölümde birçok kullanıcı için kablosuz ortam akışlarında Çapraz-Katman optimizasyon yaklaşımı incelendi. Çapraz-Katman yaklaşımında uygulama katmanı ve radyo bağlantı katmanının ortak kullanımı üzerinde duruldu. Radyo bağlantı katmanı olarak tanımlanan fiziksel katman ve protokol demetindeki veri bağlantı katmanlarının birleşmesinden oluşur. Sıra kablosuz ağ bağlantısının daha iyi bir şekilde işlemesi için yapılan iyileştirmeler üzerinde yapılan çalışma ve sırasıyla tasarlanan sistemin mimari yapısını incelendikten sonra sırasıyla parametre soyutlaması, radyo-bağlantı katmanı ve uygulama katmanında olan değişiklikler incelendi. Son olarak çapraz-katman uygulaması için yapılan simülasyon değerleri incelendi.

5.5.1. Sistem mimarisi: Oluşturulan sistemde baz istasyonunda ortam akışını gerçekleştirecek bir sunucu ve bu sunucudan birçok çoklu-ortam akış isteğinde bulunan kullanıcıların olduğu kabul edilsin. K sunucudan istekte bulunan kullanıcı sayısını temsil etsin; bu kullanıcılar aynı kablosuz ağ sisteminde ve aynı ağ kaynaklarını ortak olarak kullanıyor olsun, sadece kullanıcılar sunucudan farklı ortam dosyaları isteğinde bulunsun. Aşağıdaki şekil 5.10'da gösterilen altyapıda sonlu servis kalitesi optimizasyonunu sağlayan yapısını göstermektedir.



Şekil 5.10. Çapraz-Katman optimizasyon mimarisi [43]

Yukarıdaki şekilde görevler ve veri önerilen optimizasyon konsept bağlantı noktalarında akışı gerçekleştirir. Gerekli durum bilgileri ilk olarak uygulama katmanından ve parametre soyutlama süreci ile radyo bağlantı katmanında toplanır. Parametre soyutlama süreci parametreler içindeki özel parametre katmanı dönüşümlerinin sonucunda oluşur. Optimizasyon belirli görev fonksiyonları tarafından gerçekleştirilir.

5.5.2. Parametre soyutlaması: Bağlama optimizasyonu işleminin gerçekleştirilmesi için durum bilgisi seçilen katmanlardan soyutlanmalıdır ve Çapraz-Katman optimize işleminden elde edilmelidir. Bu işlemin yapılması gerekmektedir çünkü özel katman parametresi ya da özel teknoloji parametreleri anlaşılmaz olabilir ya da diğer katmanlar tarafından artık kullanılmıyor olabilir.

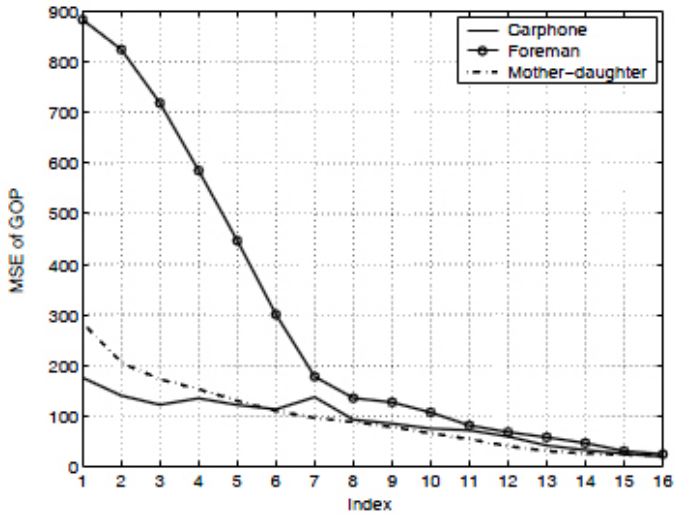
5.5.3. Radyo-Bağlantı katmanı: Fiziksel katman güç kontrolü transferi, kanal tahmini, senkronizasyon, sinyal şekli, modülasyon ve sinyal tespiti gibi konularla ilgilenirken veri bağlantı katmanı radyo kaynaklarıyla ve hata kontrolüyle ilgilenir ve her iki katmanda kablosuz kanallardaki eşsiz kanallara sıkı sıkıya bağlıdırlar. Radyo-Bağlantı katmanında birçok özel teknolojik parametreler bulunduğundan ötürü parametre soyutlama işlemi gerekli hale gelmiştir. Veri demeti $\mathcal{R} = \{r_1, r_2, r_3, \dots\}$ ve radyo bağlantı özel parametrelerini de $r_i = \{r_i^1, r_i^2, r_i^3, \dots\}$ (örneğin modül harfleri, kod oranları, hava zamanı, güç transferi, uyumluluk zamanı gibi) olarak tanımlanır. Bu özel radyo bağlantı parametreleri değişken olabilme ihtimaline karşılık, R seti her türlü muhtemel değer kombinasyonlarını içerir ve her bir demet bir muhtemel kombinasyonu simgelemektedir. Soyutlama parametre işlemini formülize etmek için soyutlama parametreler demeti $\widetilde{r_i} = \{\widetilde{r_i^1}, \widetilde{r_i^2}, \widetilde{r_i^3}, \dots\}$ bölümündeki $\widetilde{\mathcal{R}} = \{\widetilde{r_1}, \widetilde{r_2}, \widetilde{r_3}, \dots\}$ seti tanımlanır. $\mathcal{R}$ ve $\widetilde{\mathcal{R}}$ değerlerine radyo bağlantı katmanı soyutlama parametrelerinin haritalandırılması denir.

Örnek tekil kullanıcı için bir senaryo oluşturacak olursak, dört anahtar parametresi soyutlanabilir. Bu değerler transfer edilen veri '*d*', transfer edilen paket hata oranı "e", veri paketi ölçüsü '*s*' ve kanal uyumluluk zamanı '*t*' dir. Bu soyutlanmış parametreler $\widetilde{r_i} = \{d_i, e_i, s_i, t_i\}$ değerlerinin oluşmasına yol açar. K kullanıcısı için yapılacak senaryoda ise her kullanıcı için soyutlama parametresi birer birer genişletilebilir. Parametre demeti $\widetilde{r_i}$ değeri 4K parametresi içerir bunlar, $\widetilde{r_i} = \left\{d_i^{(1)}, e_i^{(1)}, s_i^{(1)}, t_i^{(1)}, \dots\dots, d_i^{(K)}, e_i^{(K)}, s_i^{(K)}, t_i^{(K)}\right\}$ dir. Bu parametreler grup olarak bir kullanıcıya aittir. Transfer veri oranı '*d*' modül planlaması, kanal kodlaması ve çoklu kullanıcı planlamasından çıkarılır. Transfer hata paket oranı '*e*' güç transferinden, kanal tahmininden, sinyal tespitinden, modül şemasından, kanal kodlamasından ve güncel kullanıcı pozisyonu gibi değerlerden türetilir. Kanal uyumluluk zamanı '*t*' kullanıcı hızı ve kullanıcıyı çevreleyen ortam değerinden türetilirken veri paket ölçüsü '*s*' kablosuz sistem standardı tarafından belirlenir. Alternatif olarak, paket hata oranı transferinin '*e*' ve kanal uyumluluk zamanı '*t*' değerlerini iki boyutlu Gilbert-Elliott Modeline göre dönüştürülür; bu dönüşümdeki karşılıkları sırasıyla p ve q değerleridir. Dönüşüm [43] verildiği şekliyle;

$$p = \frac{es}{td} \quad \text{ve} \quad q = \frac{(1-e)s}{td} \tag{5.1}$$

olarak tanımlanırken 'p' iyi durumdan kötü duruma geçiş olasılığı değerini ve 'q' ise kötü durumdan iyi duruma geçiş olasılığı değerini belirtmektedir. Parametre soyutlama demeti; $\tilde{r_i} = \left\{d_i^{(1)}, p_i^{(1)}, s_i^{(1)}, q_i^{(1)}, \ldots\ldots, d_i^{(K)}, p_i^{(K)}, s_i^{(K)}, q_i^{(K)}\right\}$ olur. Parametre soyutlama işleminin basamağının en önemli avantajı $\tilde{r_i}$ demeti artık özel teknoloji olmaması ve sadece radyo bağlantı katmanındaki anahtar karakteristiklerinden yakalanmasıdır.

5.5.4. Uygulama katmanı: Akışı yapılacak olan ortam verileri sıkıştırılmış, paketlenmiş ve bir plan dâhilinde transfer için hazırlanmıştır. Çapraz-Katman optimizasyonu için soyutlanmış anahtar parametreler sıkıştırılmış kaynak verilerinin karakteristikleriyle birebir ilgilidir. Bu tanımı formülize etmek gerekirse; uygulama katmanı özel parametrelerinin demetler kümesi $\mathcal{A} = \{a_1, a_2, a_3, \ldots.\}$ soyutlanmış parametreleri $a = \{a_i^1, a_i^2, a_i^3, \ldots\}$, a_i^j dir. Uygulama katmanı özel parametreleri değişken olduğundan ötürü A setindeki tüm olası değer kombinasyonlarını içerir ve her 'i' veri demeti değeri setteki olası bir kombinasyonuna karşılık gelmektedir. Tanımın bir sonraki aşaması; demetler kümesi $\widetilde{\mathcal{A}} = \{\widetilde{a_1}, \widetilde{a_2}, \ldots\}$ soyutlama parametreleri $\widetilde{a_i} = \{\widetilde{a_i^1}, \widetilde{a_i^2}, \widetilde{a_i^3}, \ldots.\}$, $\widetilde{a_i^j}$ dir. $\mathcal{A}$ ve $\widetilde{\mathcal{A}}$ arasında oluşan değer aralığına uygulama katmanı soyutlama parametreleri olarak tanımlanır.



Şekil 5.11. Üç farklı ortam dosyasının GOP için kayıp kareler bozulma profili [43].

Bu çalışmada elde edilen optimize Çapraz-Katman soyutlama bilgileri kodlama bozulması ve kayıp kareler için bozulma profilidir. Yukarıdaki şekilde üç farklı ortam dosyası için kodlama bozulması ve kayıp kareler için bozulma profili gösterilmektedir ve her ortam dosyası 15 kare/saniye olarak düzenlenmiştir. Her kare gurubu (GOP) şifresi çözülebilen iç-karelerden bağımsız olarak çalışır. Geri kalan 14 kare ise gömülü karelerdir, bu karelerin hepsi kare gurubundaki bir önceki karenin hatasız çözümlemesiyle beraber çözümlenecek olan karelerdir. Bozulma değerleri yeniden gözden geçirilen esas kare hataları (MSE) tarafından belirlenir. MSE gösterimdeki ve orijinal ortam dosyası arasındaki sıralamadan ölçülür. Yukarıdaki şekil 5.11'de gösterilen dizinde kayıp olan özel kareler belirtilmiştir. Bu hata gizleme stratejisinin bir parçası olarak GOP tüm aşağıdaki çerçeveler için deşifre edilmemiş olmadığı kabul edilir ve deşifre-edilmemiş kareler yerine en son doğru deşifre kareler gösterilir. 16. dizin diğer tüm kareler doğru bir şekilde ulaşırsa MSE değerini verir, bu değeri kodlama bozulması olarak kabul edilir. Kayıp ilk kare (I-karesi) yeniden düzenleme kalitesi üzerinde dramatik etkisi vardır. GOP'un kayıp olan son P-karesi (15. dizin) sıkıştırma içinde çok az miktarda bozulmaya sebep olur, bu bozulma oranı da sıkıştırma esnasında göz ardı edilen bir değerdir [43].

5.6. Çapraz-Katman Optimizasyonunun Performans ve Maliyeti

Gelecekte kablosuz ağ sistemleri artan karmaşık işlemleri destekleyen bir yapıya bürünecek ve çok az bir iş-gücü gerektiren performans gereksinimleri ile gerçek zamanlı – canlı – ya da gerçek zamanlı olmayan ortam akışları, 3-Boyutlu enteraktif görsel uygulamaları – örneğin yön bulma sistemleri gibi – çok kısa hesaplamalar sayesinde hemen kullanıcının hizmetine sunulabilecek. Bu işlemlerin yapılabilmesi için etkin ve ucuz bir ağ yapısının tasarlanması gerekmektedir. Günümüzde bir kısım ağ uzmanının üzerinde yoğunlaştığı sistem kablosuz ağ üzerinden efektif ve ucuz maliyetli olabilecek örnek çalışma Çapraz-Katman Tasarımıdır (CLD). Bu çalışma temelde bilgi alışverişi ve çoklu protokol katmanları arasında ortak optimizasyona dayanmaktadır.

Mobil iletişimindeki önceki CLD uygulamaları genellikle bireysel katmanların en iyi şekilde kullanımı noktasında odaklanırdı. Bir önceki konuda anlatılan kablosuz çoklu-kullanıcı uygulamalarında Çapraz-katman optimizasyonundaki çalışmada kablosuz ağ sistemindeki ortam akışları için Çapraz-Katman mimarisi detaylı bir şekilde

anlatılmaktadır. Farklı katmanlardan gelen parametreler soyutlanır ve çapraz-katman optimizeleri elde edilmek için kullanılır. Klasik kablosuz ağ kullanımına alternatif olarak geliştirilen CLD kullanımının performans değerlerini ve çok kullanıcılı kablosuz ortam akışı platformlarındaki maliyeti incelendi.

5.6.1. CLD optimizasyonu: Üç farklı kullanıcı üç ayrı ortam dosyası için istekte bulunsun; bunlar Mother & Daughter (MD), Carphone (CP) ve Foreman (FM) dir. Her üç ortam dosyası 176 x 144 piksel çarpanında çözünürlüğe ve her üçü 300 kare sırasından oluşurken kare oranı da 30 kare olarak kabul edilsin. Ortam dosyaları iki farklı sıkıştırma algoritmasıyla kodlansın ve bunlardan ilki 100 Kbps oranında diğeri ise 200 Kbps oranında Xvid kodeks aracılığıyla hazırlansın. Her kare grubu (GOP) 15 kare olarak belirlensin, bir tane I-karesi ve 14 tane de P-karesi. Aşağıdaki tablo 5.1'de kullanılan ortam dosyalarının ana karakterleri verilmiştir. Sıkıştırılmış ve gösterilen ortam sırası arasındaki ortalama PSNR değeri performans değerinin ölçümünden elde edilir.

Tablo 5.1. Test edilen ortam dizilerinin ana karakterleri

Ortam dizileri	Uzunluk (s)	Kareler	PSNR (dB) 100 Kbps	PSNR (dB) 200 Kbps
FM	10	300	32.45	34.67
CP	10	300	33.32	36.78
MD	10	300	36.31	39.80

Radyo-bağlantı katmanında, sistemde transfer edilen toplam sembol oranı 300k sembol/s olarak kabul edilsin. Veri paketi ölçüsü 54 baytta eşit olsun, bu değer IEEE802.11a ya da HiperLAN2 standardının kabulüdür. Kanal uyumluluk zamanı ise 50 ms saniye olarak her kullanıcı için kabul edilsin. Artan paket hata oranları ortalama değeri SNR fonksiyonundan hesaplansın [43]. Toplam transfer bit oranı modülasyon şeması m ve K kullanıcıları için $\mathcal{R}_{max}^{m,\mathcal{K}}$ olarak kısıtlansın ve $\mathcal{R}_{max}^{m,\mathcal{K}} = n \cdot \mathcal{K} \cdot \mathcal{R}$ formülünde '*n*', '*m*' modülasyon şeması için sembol başına bit oranını simgelerken '*K*' kullanıcı sayısını simgelemektedir ve '*R*' değeri ise bir kullanıcı için ortalama sembol oranını simgelemektedir. Her üç kullanıcı ve R = 100k sembol/s oranı için sırasıyla

$$\mathcal{R}_{max}^{BPSK,3} = 300\frac{kbit}{s} \text{ ve } R_{max}^{QPSK} = 600\frac{kbit}{s} \tag{5.2}$$

olarak BPSK ve QPSK değerleri tanımlanır. Bu tanımlamanın ardından herhangi bir k kullanıcısı için muhtemel transfer edilen bit-oranı set değeri C_k olarak belirlenir ve

$$C_k = \{0, r, 1.5r, 2r, 3r\} \tag{5.3}$$

formülündeki 'r' değeri 100 Kbps dir. Transfer oranı seti ile beraber toplam oran kısıtlaması bize 26 muhtemel oran vermiş olur. Eğer mümkün transfer oranı kaynak oranını aşarsa, GOP'un en önemli kareleri defalarca gönderilir.

Tablo 5.2. Üç kullanıcı arasında olası oran tahsisi durumları

Durum	Modülasyon Şeması	Transfer edilen veri oranı (Kbps)		
		Kullanıcılar		
		1	2	3
1	BPSK	100	100	100
2	BPSK	100	200	0
3	BPSK	100	0	200
…	…	…	…	…
13	BPSK	0	0	300
14	QPSK	200	200	200
15	QPSK	0	300	300
16	QPSK	300	0	300
…	…	…	…	…
20	QPSK	300	150	150

Tablo 5.2'teki her durum kullanıcıların mevcut iletim hızına bağlı olarak, operasyonel modları bir dizi neden olabilir. Örneğin tablodaki ikinci durum için, ikinci kullanıcı 200 Kbps ile uygun orana sahiptir. Toplamda her üç kullanıcı arasında 72 farklı operasyonel mod elde edilmiştir. Çapraz-Katman optimizesi fark edilen kullanıcı ortam kalitesini maksimum yapan her GOP parametre değerini seçer. Bu her kullanıcı için bir hesaplama ve alıcı tarafından beklenen kalite seçiminin hesaplanması gerekmektedir. Bu değerler sırasıyla aşağıdaki gibi elde edilir:

1. PSNR içinde beklenilen tekrar yenileme kalitesi hesaplaması;

$$PSNR_{exp} = \sum_{i=1}^{l} p_i D_i \tag{5.4}$$

tarafından verilirken '*l*' değeri farklı kayıp kalıpları numarasını, 'p_i' muhtemel kayıp kalıplarını, 'D_i' değeri ise bozulma matriksinden gelen '*i*' ninci kayıp kalıp için tekrar yenileme kalitesi sonucunu simgelemektedir. Belirli bir kayıp olasılığı 'p_i' kalıbı Gilbert-Elliot model transfer olasılıklarından hesaplanır [45].

2. Bir GOP içindeki Şifresi-Çözülebilen Karelerin Beklenen Sayısını (ENDEF) hesaplanması

$$ENDEF = \sum_{i=1}^{l} p_i d_i \tag{5.5}$$

tarafından verilir ve 'd_i' değeri belirli kayıp kalıpları için şifresi-çözülebilen kare sayısını simgelemektedir. 'D_i' değerinin verilmemesi durumunda ENDEF tahmini PSNR değerinin elde edilmesinde kullanılır.

5.6.2. Hesaplama maliyeti: Çapraz-katman optimizasyonunun ek hesaplama maliyeti özellikle tüm olası durum parametre ayarları için objektif hesaplama fonksiyonundan (ortalama beklenen PSNR ya da ENDEF gibi) elde edilir. İlk olarak normalize edilmiş işletim zamanını tanımlayalım;

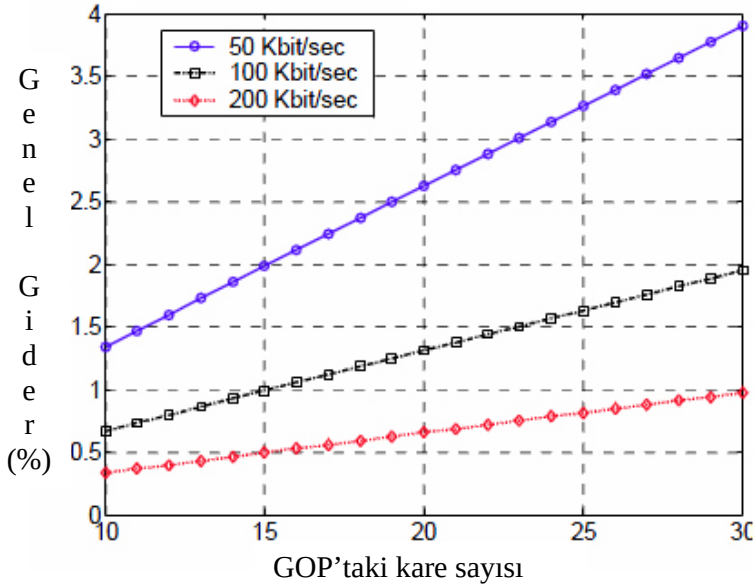
$$T_k = K \,.\, m \,.\, s \,.\, r \tag{5.6}$$

ve ardından formüldeki '*K*' değerini kullanıcı sayısı, '*m*' farklı modül şema sayısı, '*s*' muhtemel kaynak bit-oranı durumu sayısı ve '*r*' muhtemel kanal bit-oranı sayısı olarak belirtilir. Aşağıdaki tablo 5.3'te farklı kaynak yerleşimleri sayısı ve farklı kullanıcılar için mormalize edilmiş çapraz-katman işletim zamanını göstermektedir.

Tablo 5.3. CLO durum sayıları ve farklı kullanıcılar için normalize edilmiş işletim zamanı

Kullanıcı sayısı	Çalışma modları sayısı	Normalize edilmiş işletim zamanı
2	13	2*2*2*5
3	72	3*2*2*5
4	345	4*2*2*5
5	1610	5*2*2*5
6	7811	6*2*2*5
7	36372	7*2*2*5
8	169135	8*2*2*5
9	787554	9*2*2*5
10	3507183	10*2*2*5

Çalışma modlarının sayısının kullanıcı sayısıyla beraber çok hızlı artmasına rağmen, farklı durumlar için hesaplanan zamanın lineer olarak arttığı görülür. Bunun nedeni normalize edilmiş zaman parametre seti için verilmiş (5.2) ve (5.3) hesaplamadan çıkan yaklaşık değerlerden ortaya çıkar. Kalan görevin hesaplama maliyeti kullanıcı sayısının az olmasından ötürü ihmal edilir, $K \leq 10$. Kullanıcı sayısının çok fazla olması durumu önemli bir nota olarak ortaya çıkar ve kullanıcı sayısına bağlı olarak işlem modlarıda artış gösterir [44].



Şekil 5.12. GOP boyutu bir fonksiyonu olarak RD yan bilgileri gönderen Trafik yükü [44]

5.6.3. İletişim maliyeti: CLO üzerinden iletişim çoğunlukla ağ üzerinden parametre soyutlamasıdır. Özellikle çapraz-katman optimizesi amacıyla ortam sunucusu tarafından bozulma-oranındaki bilgisi transfer olur. Yukarıdaki şekil 5.12'de farklı kaynak oranları gösterilmektedir. Bu şekilden her yarım saniye ve her bir I ve bir P karesini içeren GOP değerleri tahmin edilebilir.

Sonuç olarak gelişen teknolojiler insanoğluna farklı imkânlar sunduğunda kullanılan imkânların da gelişen teknolojilere adapte edilmesi gerekmektedir. Gelişen teknolojinin sunduğu olanaklardan biri de kablosuz kullanım imkânıdır. Kablosuz ağ üzerinden veri alışverişi olabileceği gibi ortam alışverişi içinde kullanılan bu yeni teknoloji kullanıcılarına çok geniş seçenekler sunmaktadır.

Kablosuz ağ birçok alanda kullanılır; bunlardan bazıları görüntülü konuşma, akıllı ev sistemleridir. Kablosuz veri transferinde ortam akışları bu çeşitlilikte farklı kullanımlarda karşımıza çıkar. Örneğin akıllı ev sistemlerin görüntülü ve sesli ya da sadece sesli konuşma ile komut göndererek ev sistemindeki herhangi bir olayı gerçekleştirebiliriz.

Kablolu İnternetteki veri transferinde karşılaşılan sorunların bir kısmı kablosuz ağ yapısında da gerçekleşir. Daha hızlı veri gönderimi için geliştirilen algoritmalardan birinde gerçek zamanlılık ihtiyacı için uygulamalar farklı iki sıraya dizilir ve gerçek zamanlı VoIP gibi görüntülü konuşma uygulamaları gibi uygulamalar öncelikli bir şekilde gönderilmesi sağlanır.

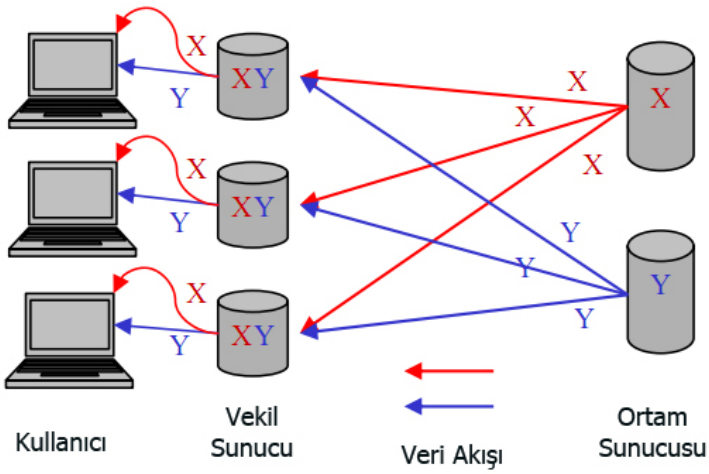
Kablosuz ağ üzerindeki kullanıcıların farklı yapıdaki sistemleri kullanıcıların heterojen bir yapı oluşturmasına yol açarken bulundukları sistemin kullanıcılarına daha iyi hizmet vermek anlamında bu heterojen yapıya uygun hizmet sunması gerekmektedir. SSS sistemi ile kablosuz ağ sistemindeki kullanıcılara daha efektif, güvenli ve kesintisiz bir akış sağlanmaktadır.

Daha hızlı veri transferi için kablosuz ağ üzerinde bulunan veri transfer katmanlarının çapraz kullanımı sayesinde iş gücü ihtiyacı düşürülmüş ve daha hızlı bir akış sağlanmış olur. Yapılan tüm bu yenilikler gelişen teknolojinin imkânlarının en iyi şekilde kullanımını sağlamaktır.

6. ORTAM AKIŞLARINDAKİ VEKİL SUNUCU UYGULAMALARI

Çoklu-ortam dosyaları İnternet üzerinde ortam sunucuları üzerinde saklanır ve herhangi bir kullanıcı tarafından istekte bulunması durumunda kullanıcının sunucuya uzaklığından kaynaklı sunucunun giriş/çıkış (G/Ç) bant-genişliğinin yeterli ölçüde geniş olması gerekmektedir; aksi takdirde sunucu kullanıcıyı reddetmek zorunda kalır.

Ortam akışlarında vekil sunucusunun kullanımına değinmeden önce vekil sunucusunu kısaca tanıyalım. Vekil sunucular genellikle ortam sunucuları ve kullanıcılar arasında bulunur ve arada oluşabilecek herhangi bir gecikmeyi düşürür ve güvenlik duvarlarının kullanıcılarına olan etkilerini minimize eder. Aşağıdaki şekil 6.1'de vekil sunucuların ortam sunucuları ile kullanıcılar arasındaki trafiği nasıl düzenlediği gösterilmektedir [32]. Protokol bazlı iletişimlerde yapılan herhangi bir saldırı vekil sunucuya yönelir; örneğin http (web) vekili gibi. Böyle bir saldırıda sadece etkilenen vekil makine olur; iç ağda bulunan web erişiminde bulunan her makine değil [16].



Şekil 6.1. Veri paketlerinin İnternet üzerinden Vekil sunucular aracılığıyla gönderimi [32]

6.1. Çoklu-Ortam Vekil Sunuculara Genel Bir Bakış

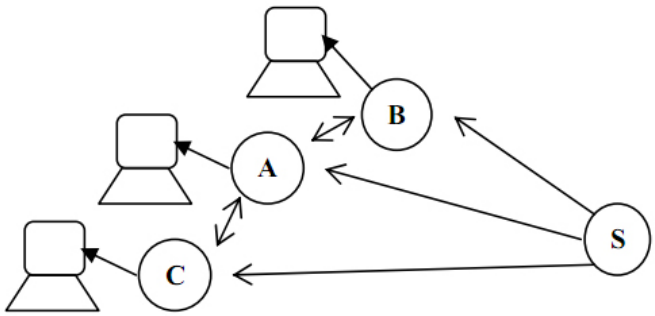
Geleneksel parça yakalama algoritmaları sunucu ile kullanıcı arasındaki yakınlığı ve yakalanan parçaların İnternet akışında frekansını dikkate alır. Ortam akışında gerçekleşen yakalanan parça fonksiyonu denklemi;

$YakalananParça = \frac{(d_i^{r1} * n_i^{r2})}{(t_i^{r3} * s_i^{r4})}$ dir ve d_i değeri i parçasının yakalanması anında gerçekleşen gecikme, n_i i parçasının akışı esnasında kullanılacak olan referans numarası, t_i akış esnasında gerçekleşen gecikme zamanı, s_i ise sadece yakalanan i parçasının ölçüsünü simgelemektedir. $r1 = 0.1$ değeri sabit iken $r2 = r3 = r4 = 1$ değerini alır [32].

Kısmi parça tahmini (PPM) sistemi bir karar mekanizması geliştirir. Bu karar mekanizması akış için hazırlanacak herhangi bir dosya için daha önce kullanıcının aynı istekte bulunup bulunmadığına bakar. Eğer kullanıcı daha önceden aynı istekte bulunmuşsa bu sistem URL adreslerinin kaydını tutar ve bu sayede bir karar mekanizması geliştirir. Bu kayıt listesine Markov Tahmin Ağacı denir. Mekanizma her parça için belirli bir eşik değeri limiti getirir ve bu tahminleri gelecekte yapılabilecek herhangi bir isteği düzenlemek için kullanır. Bu uygulamalar sayesinde kullanıcı tarafından İnternette gerçekleşebilecek herhangi bir istek için ön-yakalama hazırlanmış olur. Ön-yakalama uygulaması sayesinde akışı yapılacak ortam dosyasının başlangıç beklemesi minimize edilmiş olur. Diğer taraftan akış için hazırlanacak dosyalar veri trafiğinin aşırı yüklenmesine iki sebepten yol açarlar; bunlar: vekil sunucun yanlış tahminde bulunması, diğeri ise dinamik içerikler ön-yakalamadan sonra kolayca zaman aşımına uğraması.

Vekil sunucular için geliştirilen metotlardan biri işbirlikçi yakalama mekanizmasıdır. Eğer kullanıcının istemiş olduğu herhangi bir ortam dosyasının çoklu-vekil sunucularından akışa hazırlanması amaçlanıyorsa ilk olarak vekil sunucuların kendi yakalama kapasitelerini artırmaları gerekmektedir. Böyle bir durumda vekil sunucular aşırı yüklenir. Aşırı yüklemenin önüne geçebilmek amacıyla Servis Kalitesini (QoS) daha iyi tasarlamak amacıyla vekil sunucuların işgücünü minimize edecek bir karar vermesi gerekmektedir. Aşağıdaki şekil 6.2'de işbirlikçi yaklaşıma dayalı vekil sunucuların çalışma

mantığı gösterilmektedir. Aşağıda gösterilen yapıda öngörülen üç yaklaşım gösterilmektedir; bunlar Hiyerarşik yaklaşım, klasör tabanlı yaklaşım ve hash-based yaklaşımıdır.



Şekil 6.2. İşbirlikçi yaklaşımla çalışan B ve C Vekil sunucularının hem S sunucusuna hem de A Vekil sunucusuna yaptıkları yardım.

Hiyerarşik yapı vekil sunucularda bir ağaç yapısı oluşturur ve her katmandaki vekil sunucuları kendi alt katmanlarındaki çocuk vekil sunucularının içerikleriyle ilgili bilgileri toplar. Vekil sunucular klasör tabanlı yaklaşımlarda kendi klasörlerindeki bilgileri diğer vekil sunucularla değiştirir. Bu uygulamanın yapılabilmesi için vekil sunucuların hem fazla bir yere ihtiyaçları hem de fazladan bant-genişliğine ihtiyaçları olmaktadır. Hash-tabanlı yaklaşımda ise işbirlikçi yaklaşımdaki her vekil sunucuda klasörler tanımlanır ve eğer herhangi bir klasör tekrar geri alınırsa, bu klasör ya belirlenen vekil sunucuda yakalanmıştır ya da klasörün hepsi yakalanmıştır. Böylece bu son mekanizma ile hem işgücü düşürülmüş olur hem de yakalanan alan minimize edilmiş olur [32].

6.2. Geri Oynatımda Vekil Sunucuların İşlevleri

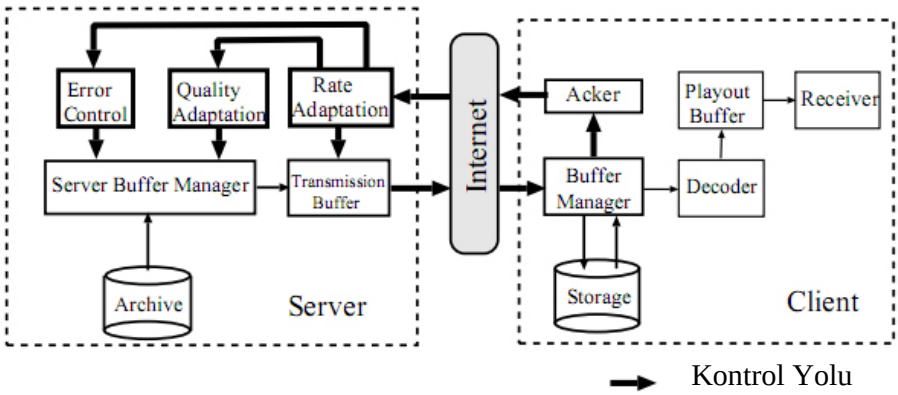
P2P geleneksel kullanıcı tabanlı sunucu yapısına meydan okur. Fakat birçok güncel P2P network uygulaması dosya paylaşımı ve transferi konusunda belli limitlere sahiptir, BitTorent gibi. Çoklu ortam paylaşımları azdır. Eğer kullanıcı diğer kullanıcılarla gerçek zamanlı ortam ya da ses programlarını paylaşmak isterse çoklu ortam paketlerinin gönderim fonksiyonelliğine sahip olması gerekmektedir.

RTP paketlerinin bağlanmasında çeşitli zorluklar gerçekleşir. Bu zorluklar her yeni istek oturumu için bazı alanlar örneğin zaman-damgası ve RTP paketlerinin sıra numaraları

rastgele bir numara seçilerek verilir. Bu demektir ki diğer istek oturumu örneğin önceden kaydedilmiş bir verinin kalıntısı (20 sn ile sona kadar) sunucudaki, farklı sıradan başlangıç zaman damgası ve istek numarasına sahiptir.

Günümüzde bilişim teknolojilerindeki hızlı gelişmeler ticari amaçlı İnternet üzeri ortam akışlarının popülaritesini de artırmıştır. Her geçen gün bu ihtiyaçların daha da artması beklenmekte ve gelişmeler ışığında daha hızlı bir hizmet verilmesi zorunlu hale gelmiştir.

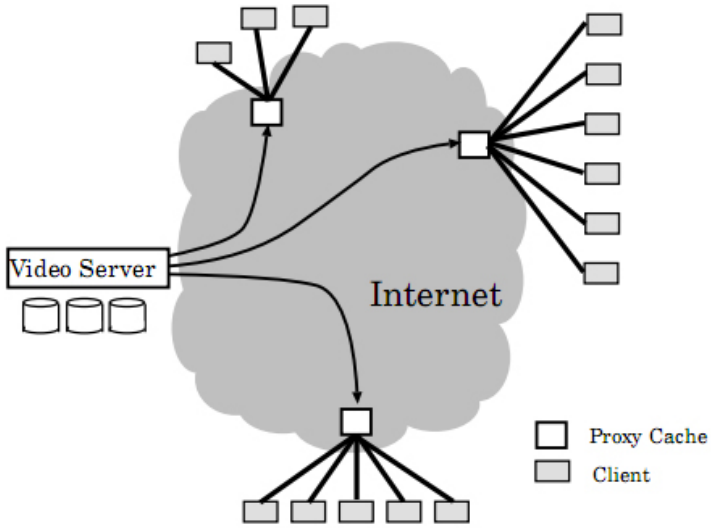
Akışı yapılan herhangi bir ortam dosyası için geri-oynatım çok önemli bir unsurdur. Şekil 6.3'te İnternet ortamında ortam akışları için kaliteli geri-oynatım altyapısı gösterilmektedir. Şekilde gösterilen yapı (sonlu algoritma) Oran Adaptasyon Protokolü (RAP) tarafından test edilmiştir [37]. RAP algoritması kısaca oran tabanlı sıkışıklık kontrolünü gerçekleştiren bir mekanizmadır; devamlı olarak bağlantıları kontrol eder. Hata Kontrol (EC) modülü ile sunucuda oluşabilecek hatalar kontrol edilir. Sunucudaki Kalite Adaptasyon (QA) modülü ile de transfer edilen akışın kalitesi düzenlenir.



Şekil 6.3. İnternet üzerinden gerçek zamanlı geri oynatım uygulamalarında sonlu yapı [30].

Şekil 6.4'te vekil yakalama yoluyla gerçekleştirilen gerçek zamanlı ortam ya da ses akışları için sonlu yapının altyapısı gösterilmiştir. Sunucular genellikle birden fazla isteği eşzamanlı yerine getirmek zorundadır. Akışı yapılacak olan ortam dosyası genellikle büyük boyutlarda ve yerel disk üzerinde akışın geri-oynatımı için gerekli alanın yeteri kadar olmaması ayrıca yerel diskler üzerinde geri-oynatım yapılması durumunda başlangıç

beklemesinin büyümesinden ötürü ortam akışlarında büyük sorunlar ortaya çıkmaktadır. Her vekilin akışları daha rahat yakalayabilmesi için akışlar eşit boyutlarda parçalanır. Orijinal kaynak ile kullanıcı arasındaki ya da vekil sunucu ile kullanıcı arasındaki her akış TCP tabanlı sıkışıklık kontrol mekanizmalarına uygun bir biçimde olması gerekmektedir. Diğer bir değişle sıkışıklık kontrol mekanizması ve kalite adaptasyonu hem orijinal sunucu hem de vekil sunucu tarafından yapılabilir.



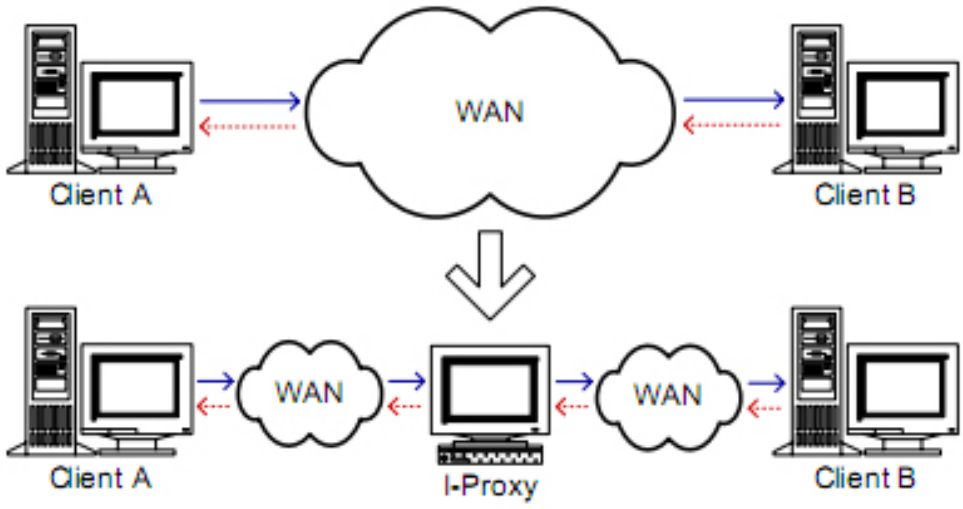
Şekil 6.4. Vekil Yakalama yöntemi ile gerçekleştirilen ortam akışı [30].

6.3. Etkileşimli Ortam Uygulamaları için Vekil Seçimi

İnternet ortamında gelişen teknolojilerin uğraş alanlarından biride veri dağıtımını kolaylaştıracak herhangi bir yolun bulunması üzerinedir. Yapılan her yeni metodun temel amacı genellikle metin tabanlı veri transferinde kullanılan İnternet üzerinde daha hızlı sesli-görsel verilerin iletimini sağlamaktır; fakat İnternet üzerinde bulunan gizli tuzaklardan ötürü veri transferinde bazen veri paketleri kaybolurken bazen de veri paketlerinde gecikme gerçekleşir. Bunların oluşması Kalite Servisinde (QoS) istenmeyen sonuçlar ortaya çıkartır.

İnternette kullanılan uygulamalardan biride enteraktif uygulamalardır ve bu uygulamalarda oluşabilecek herhangi bir gecikme uygulama için çok büyük bir sorun teşkil etmektedir. İnternet üzerinde etkileşim halinde bulunan birbirinden uzak iki kullanıcı arasında oluşabilecek herhangi bir gecikme IP yönlendirici mekanizması tarafından belirlenir. IP yönlendirici mekanizmasına herhangi bir kullanıcı herhangi bir değişiklikte bulunamaz.

Eğer herhangi iki kullanıcı arasında kurulan bağlantıda atlayış noktaları var ise bu kullanıcılar arasında gecikmelerin oluşması beklenilen bir durumdur. Bu sorunun önüne geçebilmek için önerilen pratik yollardan biri İnternet üzerinde kurulabilecek herhangi iki sonlu uzak bağlantıyı daha küçük alt networklar halinde parçalamaktır; bu uygulamayı aşağıdaki şekil 6.5'te görmektesiniz. Bölme uygulamasının iki önemli avantajı olmaktadır; ilki her alt networkta kullanıcılar arasındaki mesafe daha az olduğundan ötürü iki kullanıcı arasında daha az atlayış noktası bulunmaktadır, böylelikle kullanıcılar daha doğru ve kesin bilgiye ulaşır; diğer bir avantaj ise I-Vekiller sayesinde hata kontrollerinin yapılmasıdır.



Şekil 6.5. I-Vekil uygulaması [41]

Vekil sunucular İnternetin ilk kullanımından beri kullanımda olan bir uygulamadır ve temel amacı aracı görevini üstlenmektir. Diğer taraftan bazı mesaj servisleri de I-Vekil sunucusuna benzer görevleri yerine getiren uygulamalara sahiptirler, örneğin MSN mesaj servisi gibi [34].

Eğer iki kullanıcı gerçek zamanlı iletişim için kamera ile ortam dosyasının yakalanarak şifrelenip ardından gönderilmesi için bir uğraş içerisinde olsun, bu işlevde I-Vekiller devreye girer. Bununla beraber, I-Vekil sunucuları herhangi bir veriyi çok kısa zamanda yakalar. Paket değişimi IP networklarında, veri paketleri sıraya girer, yönlendirilme işlemleri yapılır, paketler tekrar sıraya dizilir vb. gibi işlemleri yapılır. Bu işlemler esnasında, kullanıcı istekte bulunduğu verinin paketleri iletilirken kayboldu mu yoksa iletilirken bir gecikme mi oluştuğu hakkında herhangi bir bilgiye sahip değildir. Diğer taraftan Tekrar Gönderim Zamanaşımı (RTO) tahmin metodunun veri paketlerinin ulaşması için geçecek süreyi hesaplaması beklenir. Eğer paketler hedefe beklenilen sürede ulaşmazsa, veri paketi kaybolmuş kabul edilir. Bunun sonucunda kullanıcı sunucuya tekrardan gönderim için istekte bulunur; aynı işlem sunucu tarafından da yapılabilir. Tekrar gönderim isteği uygulamasına alternatif olarak I-Vekil sunucuları isteği hemen kabul eder ve veri gönderimini hemen yapar. Bu uygulamaya Erken Tekrar Gönderim uygulaması denir. Genellikle, Erken Tekrar Gönderim sonlu tekrar gönderim uygulamasına göre daha kısa zamanda gerçekleştiği istatistikî olarak ta belirlenmiştir.

Bazı veri paketleri I-Vekil sunucuları arasında da kaybolabilir, böyle bir olayın gerçekleşmesi durumunda kullanıcı kayıp paketleri belirleyebilir ama kullanıcı kayıp paketleri belirlemeden önce I-Vekil sunucuları kayıp paketleri hemen belirler ve ardından ana sunucuya istek gönderir. Bu istek o kadar çabuk gerçekleşir ki kullanıcı daha I-Vekil sunucularına istekte bulunmadan olay gerçekleşmiş ve tekrar gönderim kayıp paketler için başlamış olur [34].

6.4. Ortam Akışlarının Dağıtımı için Kod-Çevirimi Yakalama Stratejisi

Her geçen gün gelişen İnternet teknolojisinde hala çözülmeyi bekleyen en önemli sorun şifreleme/sıkıştırma, yakalama, gönderme ve ortam dağıtımıdır. Veri türlerinin aktarımında en önemli yardımcı unsur vekil sunucularının kullanımı olduğundan ötürü bu sunucularını daha aktif kullanımı için birçok algoritma geliştirilmektedir. Yapılan her algoritmada son kullanıcıların eşit ağ sistemine sahip oldukları kabul edilir ve kabul üzerine algoritmanın temelleri geliştirilir. Diğer taraftan İnternet kullanıcılarının çoğu birbirinden farklı ağ ve sistem yapılarını kullandıkları için heterojen bir yapı oluşur. Örnek

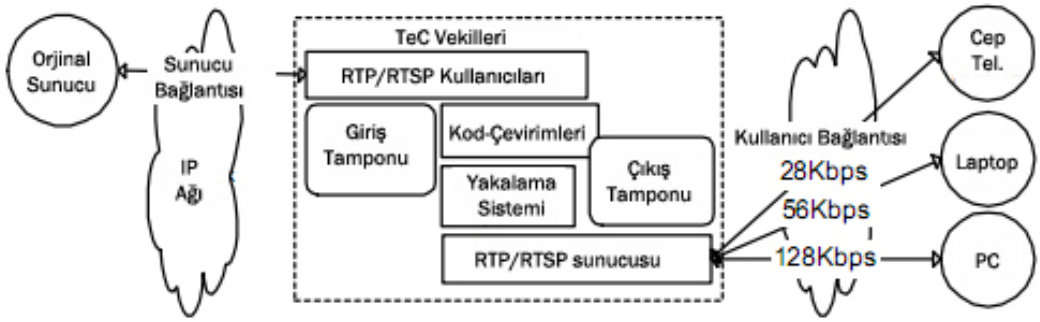
olarak çevirmeli İnternet bağlantıları genellikle 28-56 Kbps bağlantısı kullanırken kablosuz bağlantılar ya da 1Mbit bağlantılı olanlar aynı hızda veri alışverişinde bulunmazlar.

Böyle heterojen yapılara uygun olarak geliştirilen Kod-Çevirmeli Yakalama (TeC) vekil sunucularının kullanımı ile ortam dağıtımları daha rahat yapılmaktadır. Bu sistemin amacı efektif zengin ortam alışverişini heterojen yapılarda gerçekleştirmektir. Bağlantı hızının farklı olması durumunda TeC vekil sunucuları kullanıcıların isteklerine göre ortam dosyalarını farklı formatlarda kodlayarak gönderimini gerçekleştirir. TeC vekil sunucularının en büyük avantajı kaynak ortam sunucularının kullanıcı isteğine göre ortam dosyalarını farklı formatlarda göndermek zorunda kalmamalarıdır. Birçok kodlama tekniği heterojen yapıdaki kullanıcıların daha rahat bilgi alışverişinde bulunması için farklı metotlar sunmaktadır, örneğin ölçeklenebilir ve katmanlı kodlama teknikleri gibi. Bu yakalama teknikleri genellikle MPEG-2 ve MPEG-4 kodlama standartlarına göre hareket ederken öte yandan İnternet üzerindeki birçok ortam dosyası aynı kodlama yapısına göre kodlanıp akış için ortam sunucularına konulmamıştır. Farklı uzantılarda ve farklı sıkıştırma yapılarında olan bu ortam türleri için aynı algoritmayı kullanmak genellikle efektif bir sonuç vermez.

Üç farklı TeC yakalama stratejisi ile kullanıcılara farklı uzantılara sahip ortam dosyalarının akışını aynı sistem içerisinde daha rahat bir şekilde mümkün kılar. Önerilen algoritmalardan ilk ikisi genellikle ortak bir ortam türü için yakalama algoritması sunmakta iken diğer üçüncü algoritma aynı ortam dosyasının birçok farklı sürümü için yakalama imkânı sunar ki böylece kod-çevirimi işlemi için toplam işgücü hafifletilmiş olur.

6.4.1 Kod-Çevirimi Yakalama (TeC) Sistemi Mimarisi: TeC vekil sunucularının bileşenleri aşağıdaki şekil 6.6'da gösterilmektedir. Sunucu kaynak ortam sunucusuna bir kullanıcı gibi davranır. Alıcının akışı giriş tamponunda gerçekleşir ve kod-çevirim devamlı bir şekilde bit akışını giriş tamponundan çekerken aynı zamanda kod-çevirimli bitleri de çıkış tamponuna iter. Vekil sunucu ortam içeriğini hem giriş hem de çıkış tamponundan temin edebilirken vekil sunucu son kullanıcıya bir sunucu gibi davranır. Gerçek zamanlı bir akış sağlanması amacıyla TeC vekilleri ortam kod-çevirimlerini dinamik olarak farklı uzantılar için heterojen yapıdaki son kullanıcılara ulaştırılmak üzere yaparlar. Farklı türdeki ortam dosyaları birbirlerine dönüştürülürse bu tip ortam dosyaları

için kodu değiştirilebilir ortam dosyaları denir örneğin 128 Kbps bit-oranıyla kodlanan ortam dosyası 64 Kbps bit-oranındaki ortam dosyasına ve 64 Kbps bit-oranındaki ise 128 bit-oranına dönüştürülebilir. TeC vekil sunucuları herhangi bir ortam dosyasından (n-1) farklı sürüm elde edebilir; '*n*' muhtemel sayıdaki sürüm sayısıdır.



Şekil 6.6. Kod-Çevirimi Yakalama vekilleri sistemi ve bileşenleri [48]

6.4.2 Kod-Çevirimi Yakalama (TeC) Sistemi Kod-Çevirimi Teknikleri: Ortam kod-çevirimi işlemi bolca hesaplama gerektiren bir işlemdir, bu nedenle birçok araştırmacı efektif bir hesaplama yapabilmek amacıyla farklı çalışmalar yürütmektedirler. Bu çalışmalar içerisinde bölge-tabanlı sıkıştırma yaklaşımı en iyi performansı vermektedir. Bu yaklaşımda giriş ortam verileri kısmen sıkıştırılmaz. Oran adaptasyonu da bu yaklaşımda kullanılır. Bu yaklaşım sayesinde klasik şifreleme-çözümleme yaklaşımlarındaki işlem hızı artmaktadır. TeC vekil sunucuları bölge-tabanlı sıkıştırma tekniğini kullanmaktadırlar. TeC vekil sunucularının kullanmış olduğu iki farklı kod-çözücü – bit-oranı düşürümü ve özel çözünürlük düşürümü kod-çözücü – kullanır, aşağıdaki şekilde bit-oran düşümü kod-çözücü işlemi gösterilmektedir. DCT katsayıları çözümleme dağınımından elde edilir. Bit-akışındaki diğer tüm bilgiler ise şifreleme dağınımından ya da çıkış bit-akışından direkt olarak elde edilir eğer herhangi bir düzenlemeye ihtiyaç yoksa. Özel çözünürlük düşürümü kod-çevirimleri sıkıştırılmış bölgelere özel olarak uygulanabilir. Kod-çözücü ortam dosyası için hareket bilgilerini orijinal ortamdan direkt olarak temin eder ve böylece pahalı hareket tahmin işlemlerinden sakınmış olur.

Her iki kullanılan metot sıkıştırılmış bölge bilgilerini elden geldiğince kullanmaya çalışır ki kod-çevirimi işlemi CPU'yu elden geldiğince daha az zorlamış olur. Bu

karakteristik yapı kod-çevirimi sunucuları için çok önemlidir, böylelikle yoğun kod-çevirimleri rahatlıkla yapılmış olur.

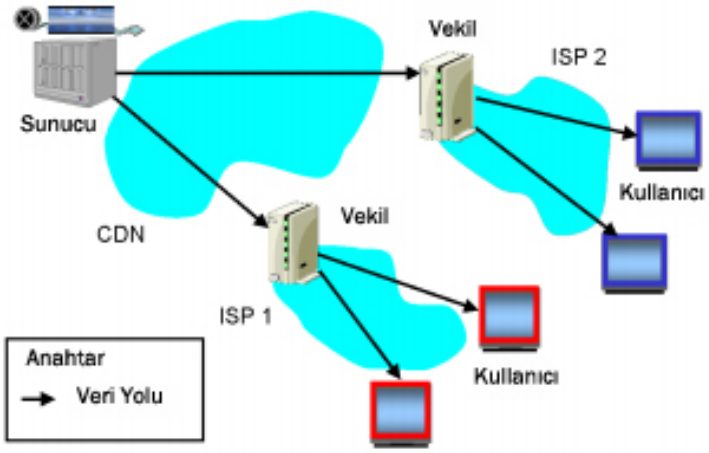
Kod-Çevirimi Yakalama (TeC) sistemi vekil sunuculardaki kod-çevirimi sistemlerine alternatif olarak kullanıp daha iyi performansın verildiği görülmüştür. TeC vekilleri heterojen yapıdaki ağ yapılarında yakalama işlemini diğer yaklaşımlardan daha iyi yapmaktadır. Günümüzde kullanılan İnternetteki birçok ortam içeriği tek-katmanlı şifreleme yöntemiyle hazırlanmıştır. Hazırlanmış olan tek-katmanlı şifreleme yöntemiyle hazırlanan ortam dosyaları kod-çevirimine eklenirse yakalama vekilleri efektif bir yakalama gerçekleştirilebilir [48].

6.5. Popüler Ortam Dosyaları için Yapılan Periyodik Yayınlardaki Ön-Ek Yakalaması

Bu bölümde vekil ön-eklerin yakalama sisteminin periyodik yayınına ve birden fazla eşzamanlı bağlanmayan kullanıcıların örneğin network içerisindeki İçerik Akış Dağıtımı (SCDN) yapısındaki popüler ortamların efektif bant-genişliğine birleştirilmesine bakacağız. Esnek dağıtım şeması özellikle farklı bölgelerde olan kullanıcıların ihtiyaçlarını karşılayabilir. Anahtar yapısı uygun transfer şemasının dâhil olduğu dağıtım şeması için kullanılan son-ek (sunucudan yayın yapan) ve ön-eklerin (vekil sunucularından yayın yapan) fark edilmesini zorlar. Bu durumun aşılması için geliştirilen iki yol bulunmaktadır; bunlardan ilki uygun son-ekin transfer şemasında belirlenmesi ve sunucu vekil yolunun toplam kullanımını minimize eden vekil tamponundaki karşılığının belirlenmesidir. Bir diğer yol ise her vekil için periyodik yayın ve yamanın birleşiminden oluşan uygun ön-ek transfer şeması belirlenir.

Geliştirilen algoritma ile vekil tampon alanı belirlenerek ağ içerisindeki sunucu ile vekil sunucu arasındaki yolda toplam bant-genişliği kullanımı minimize edilir. Son-ek ve ön-ek çiftlerinin bozularak transfer edilmesi farklı vekillerin farklı ön-ek transfer şemalarının kullanılmasına olanak sağlamış olur. Böyle bir uygulama vekillerin kendi ön-eklerinin transfer şemasını küçük değerdeki değişkenlerle evrensel son-eklerin transfer şemasının değişmesine hiçbir ihtiyaç duymayan herhangi bir ortam için yerel ihtiyaçları

dinamik olarak uyarlar. Aşağıdaki şekil 6.7'de Ağ içerisindeki İçerik Akış Dağıtımı (SCDN) yapısı gösterilmektedir.



Şekil 6.7. İnternette içerik dağıtım ağı [49]

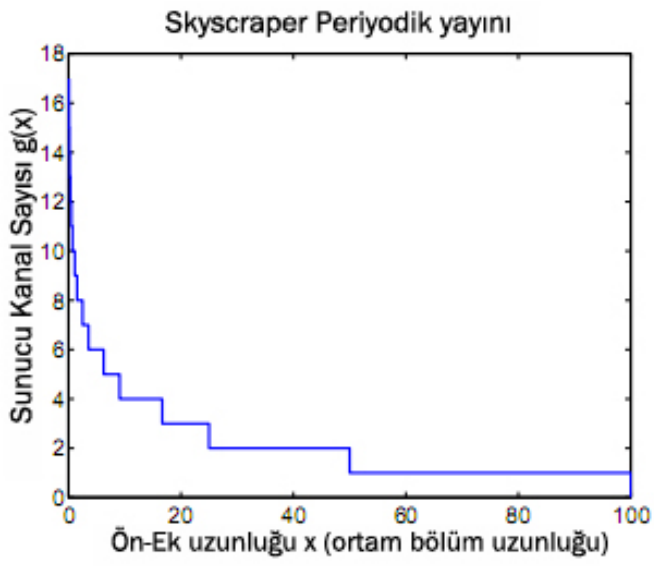
Bir tek vekil sunucunun ağ yapısı içerisinde olan birçok kullanıcıya yayın yapıyor olsun, hem sunucu ile vekil sunucu arasındaki ağ yolları hem de vekil sunucu ile kullanıcı arasındaki ağ yollarının çoklu yayın yapma kapasitesi olsun. Ağ içerisinde sisteme bağlanan kullanıcının her zaman için ortam dosyasını başlangıçtan itibaren geri-oynatma isteğinde bulunuyor olsun. Ön-eklerin kullanıcıya ön-ek transfer şeması kullanılarak akışı yapılsın. Kalan ortam son-ekleri ise son-ek transfer şeması kullanılarak yapılsın. Ön-ek ve son-ek transfer şemaları beraber özel bir vekil sunucusu için ortam transfer şemasını oluşturur.

Sabit-Bit-Oranlı (CBR) ortam dosyalarının eşit geri-oynatım oranlarının olduğu kabul edilir. İlk olarak *i* ortamı l_i saniye uzunluğuna sahip olsun ve vekil sunucu tarafından $x_i l_i$ saniyelik kısmı yakalansın; $0 < x_i \leq 1$. Bununla birlikte x_i değeri *i* ortamı için ön-ek ölçüsü olsun. Vekil sunucu tarafından yakalanan ortam dosyaları bellek miktarını aşamaz; yani $\sum_{i=1}^{K} x_i l_i \leq B$. Böyle bir şemayı ortaya çıkarmaktaki amaç sunucu-vekil sunucu arasındaki transfer için gerekli olan bant-genişliğini minimize etmektir, $C = \sum_{i=1}^{K} c_i$ denkleminde c_i, *i* ortam dosyası için kullanılan sunucu sayısıdır.

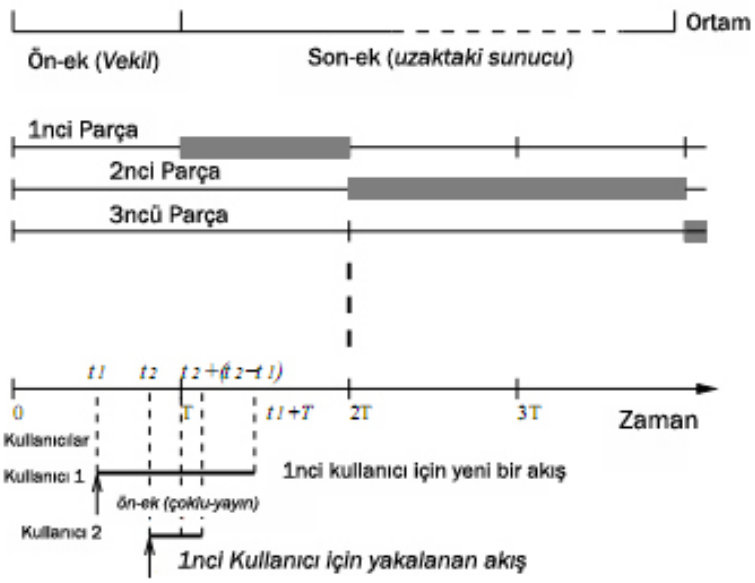
Ön-ek ve son-eklerin transfer şemaları için iki basamaklı bir yaklaşım belirlendi. Birinci adım: her ortam için uygun bir son-ek transfer şeması ve ilgili vekil tamponu ataması yapıldı. İkinci adım: her vekil sunucusu için uygun bir ön-ek transfer şeması belirlendi. Bu ölçeklenebilir yaklaşımdan vekil ön-ek yakalaması ile beraber periyodik yayın kombinasyonları türetilir.

Tekil bir ortam verisinin dağıtımı: vekil sunucu ve kaynak sunucular ön-ek ve son-eklerin yayınlanmasında beraber sorumludurlar. Kaynak sunucu birçok vekil sunucu üzerinden son-eklerin isteğinde bulunan kullanıcı isteklerini alırken son-ek transfer tekniği için uygun periyodik yayın belirlenir. Verilen ön-ekler için ilk son-ekin parça ölçüsünün artışı son-ek parçalarının sayısını ve toplam son-ek transfer bant-genişliği azalır. Eğer ön-ek yakalaması eksik olursa, ilk parçanın uzunluğu herhangi bir kullanıcı için kötü bir başlangıç anlamına gelir. Hazırlanan şemada, ayrık ön-ek transfer şeması ortamın ilk parçası tarafından kullanılır ki böylece başlangıç beklemesi için büyük parçalar oluşmamış olur. Herhangi bir sorunsuz transferin kullanıcının ön-ekler ile son-ekleri yer-değiştirir şekilde oynatması esnasında oluşması için ön-eklerin sonuna ulaşılmadan son-eklerin başlamasıyla olur. Bu eylemi garantilemek amacıyla son-eklerin ilk parçası ön-eklerden büyük olmaması gerekmektedir. Bu iki eylemin kombinasyonundan oluşan sistem aşağıdaki şekil 6.8'de gösterilmektedir.

Aşağıdaki şekil 6.9'da son-ekin birkaç parçaya bölünmesi gösterilmektedir. 1nci kullanıcı t_1 zamanında ulaşırken, vekil sunucu ise ön-ek transferi için yeni bir akışı başlatır. $\mathcal{T}$ zamanında 1nci kullanıcı 1nci bölümden itibaren son-eklerin akışına başlarken aynı ortam dosyası için 2nci kullanıcı t_2 zamanındaki erişimi gerçekleşir. Eş zamanlı olarak vekil sunucusu parçaları 2nci kullanıcıya gönderir, $[0,\ t_2 - t_1]$. Dikkat edilmesi gereken önemli husus, 2nci kullanıcı $\mathcal{T}$ zamanında son-eklere erişir ve 1nci kullanıcı gibi gölgeli parçalara eş zamanlı erişir. Hangi kullanıcının hem ön-eki hem de son-eki hangi zamanda aldığına dair tutulan periyodik bir zaman bulunmaktadır. Böylelikle kullanıcıların ihtiyaç duydukları kanal sayısı, en kötü durum için, ön-ek ve son-eklerin temin edilmesi için gerekli olan toplam kanal sayısı olarak belirlenir.



Şekil 6.8. Sunucu kanal sayısının ön-ek ölçüsü ile karşılaştırılması [49]



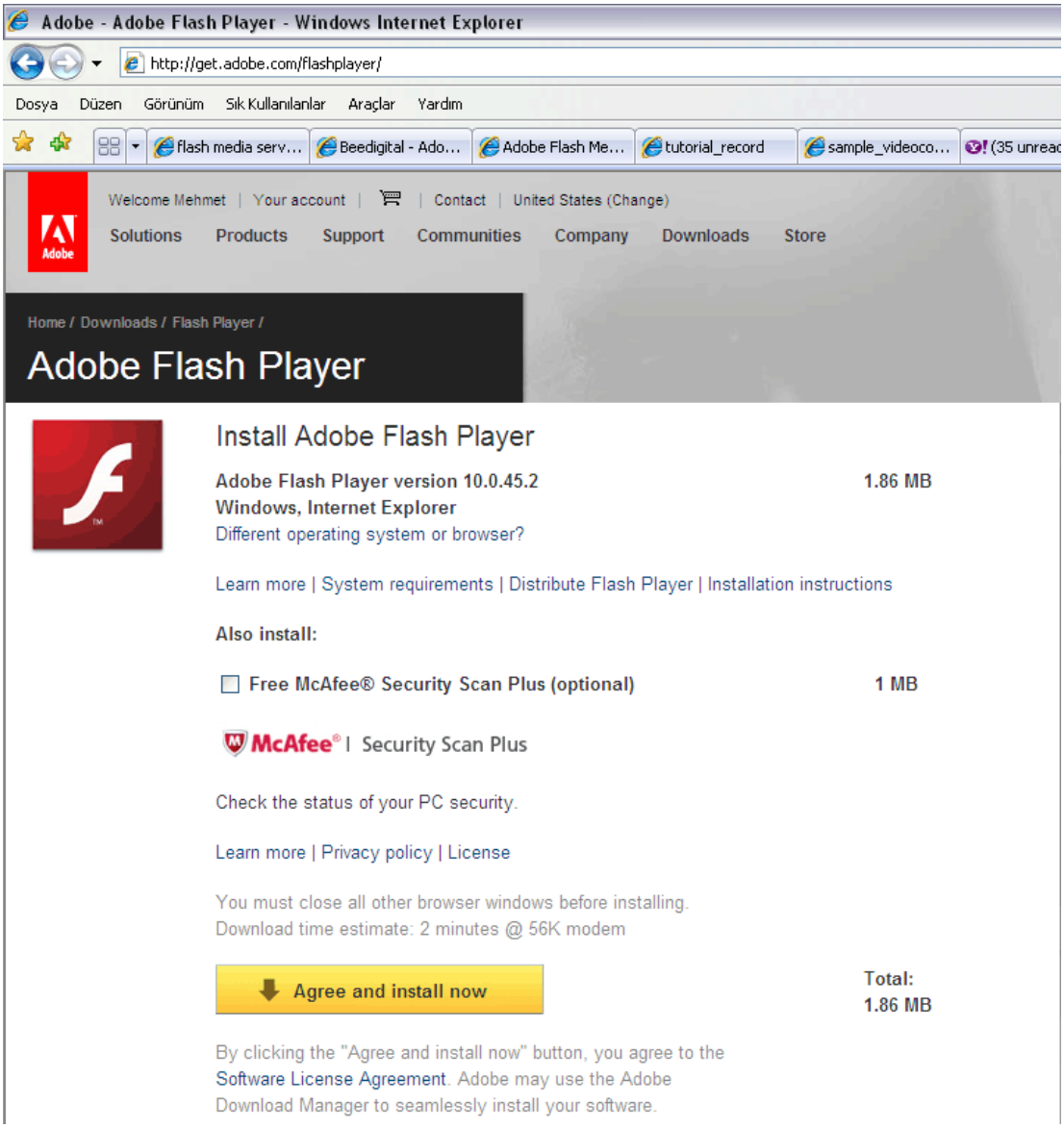
Şekil 6.9. Periyodik yayına yardımcı olan ön-ek yakalaması [49]

7. UYGULAMA

7.1. Kullanılan Yazılımlar

Hazırlanan yazılımın hazırlık aşamasında yazılımın daha fazla bilgisayar kullanıcısına daha hızlı ve rahat bir şekilde iletilmesi ilke edinilmiştir. Bu amaçla internet üzerinde popülaritesi en fazla olan uygulamalardan biri olan Adobe Flash yazılımı tercih edilmiştir. Adobe Flash yazılımı kullanılarak hazırlanan uygulamalar eğer İnternet üzerinde yayınlanmak isteniliyorsa bilgisayar kullanıcısının Adobe firmasının kendi web sayfasına gidip 1.86 MB lık ufak yazılımın indirilip kurulması gerekmektedir. Aşağıdaki şekil 7.1'de gösterilen web adresi üzerinden yazılımı indiren kullanıcı, hızı ne kadar düşük olursa olsun indirme ve kurma işlemi için geçen toplam sürenin minimum olması amaçlanmıştır. Araştırmalar İnternet kullanıcılarının ortak kaygısının yapılan herhangi bir işlem için geçen sürenin minimum seviyede olmasını tercih ettiklerini görtermiştir.

Uzaktan eğitim yazılımı için Adobe Flash yazılımının tercih edilmesinin bir sebebi de dünyadaki tüm bilgisayar kullanıcılarının 99% Adobe Flash Player uygulamasını kendi kişisel bilgisayarlarına kurup kullanmalarıdır [74]. Hazırlanacak olan herhangi bir yazılımın işlevsel olması amaçlanıyorsa birçok kullanıcı tarafından uygulama için gerekli ön eklenti yazılımlarının kurulu olması tercih edilmelidir. Örneğin Microsoft firması tarafından hazırlanan ve Adobe Flash uygulamalarına alternatif olarak geliştirilen Silverlight yazılımıyla hazırlanan uygulamaların İnternet kullanıcıları tarafından kullanılması durumda Microsoft Framework 2.0 sürümü ya da daha üst sürümlerinin kullanılması gerekmektedir. Framework 2.0 yazılımı ise 23.84 MB tır; böyle bir yazılımın İnternet üzerinden indirilmesi boyutu sadece 1.86 MB olan bir oynatıcı yazılımının indirilmesinden daha fazla zaman alır. Diğer taraftan Adobe Flash Player hem Microsoft hem de Linux gibi diğer işletim sistemlerinde kullanılabiliyorken, Silverlight uygulaması sadece Microsoft işletim sistemleri üzerinde çalışır.



Şekil 7.1. Adobe Flash Player yazılımı indirme ana sayfası

İnternet kullanıcıları heterojen bir yapıdadır. Kullanıcılar arasında uzman yazılımcılar olabilecekken, sadece adres satırına birkaç adres yazıp belirli web sayfalarına girebilen bilgisayar kullanıcıları da bulunmaktadır. Bu heterojen yapı göz önüne alındığında hazırlanacak yazılımın daha efektif çalışması için her kullanıcının ihtiyacına cevap verebilecek yazılımlar geliştirilmelidir.

Daha önce Macromedia firması tarafından oluşturulan ve günümüzde ise Adobe firması tarafından geliştirilen Adobe Flash Media Server (Adobe Flash Ortam Sunucusu) ile Adobe Flash yazılımıyla hazırlanan uygulamalar çalıştırılmaktadır. Adobe Flash Ortam Sunucusu yazılımına alternatif olarak geliştirilen hem Adobe Flash uygulamalarını hem de Java yazılımıyla hazırlanan enteraktif uygulamaların çalışmasına yardımcı olan Red5 Media Server (Red5 Ortam Sunucusu) yazılımının en büyük avantajı ücretsiz olmasıdır.

Aşağıda uzaktan eğitim yazılımı için kullanılan ve projenin hazırlık aşamasında incelenen Red5 Ortam Sunucusu gibi yazılımlar incelenmiştir.

7.1.1. Red5 Ortam Sunucusu: Adobe Flash Ortam Sunucusu yazılımına alternatif olarak geliştirilen açık kaynak bir yazılımdır. Java programlama dili ile hazırlanmıştır. Açık kaynak bir yazılımdır. Flash Oynatıcı ile Ücretsiz Bağlantı Odaklı Sunucu RTMP protokolü arasında etkileşim sağlar. Açık kaynak yazılımlara olan ilgi ve bu yazılımlar üzerinde geliştirilen projelere olan ilgi lisanlı yazılımlara kıyasla sağladıkları avantajlardan ötürüdür.

RTMP protokolüne daha önce kısaca değinmiştik; Flash uygulamaları için geliştirilmiş olan bu protokol kişiye özel kapalı kaynak bir protokoldür. Red5 Ortam Sunucusu Flash uygulamaları için RTMP protokolünü kullanır. Red5 Ortam Sunucusu RTMP protokolünün yanı sıra Aktif Mesaj Formatı olan (AMF) protokolü de kullanılır. Kullanılan bu protokol Actionscript nesne modeli için tasarlanmış ikili ileti gönderim protokolüdür. Red5 Ortam Sunucusu Adobe Flash uygulamalarının yanı sıra Java programında hazırlanan uygulamaları da çalıştırılır.

Red5 Ortam Sunucusunun yazılımcının bilgisayarına kurulması için Java Eclipse yazılımı ile Apache Ant ve MySQL yazılımlarının kurulması gerekmektedir. Apache Ant ve MySQL yazılımları veri tabanı ile iletişimi sağlamak amacıyla kullanılır. Red5 Ortam Sunucusu birçok işletim sisteminde de sorunsuz çalışmaktadır. Red5 Ortam Sunucusunun çalıştığı işletim sistemlerinden bazıları Windows, Debian/Ubuntu, MacOSX ve Linux tır.

Red5 Ortam Sunucusunda herhangi bir uygulamanın çalıştırılması için Flash CS3 veya Flash CS4 yazılımlarında hazırlanan karşılıklı yazışma, görüntülü konuşma,

enteraktif oyun, vb… gibi uygulamalar hazırlanır daha sonra Red5 Ortam Sunucu yardımıyla çalıştırılır.

7.1.2. Adobe Flash Ortam Sunucusu: Flash Ortam Sunucusu daha öncede değindiğimiz üzere ilk olarak Macromedia firması tarafından Flash uygulamalarının yayınlanması için kullanılması amaçlanan bir yazılım olarak oluşturulurken daha sonra Adobe firması tarafından satın alınıp günümüzde de geliştirilmeye devam edilen bir yazılımdır. Bu sunucu ile sadece Flash uygulamaları çalıştırılmaktadır.

Bu yazılımın çalıştırılması için daha önceden kurulması gerekli herhangi bir ek program bulunmamaktadır. Adobe Ortam Sunucusu 3.5 sürümü ile hem Red5 Ortam Sunucusu için hem de Adobe Flash Ortam Sunucusu için gerekli olan Apache Ant yazılımı varsayılan olarak kurulmaktadır. Bu varsayılan programın dışında herhangi bir programın kurulmasına gerek duyulmamakla beraber kendiliğinden kurulması da kullanıcı için fazladan herhangi bir iş yükünü gerektirmemektedir.

Kullanılan Adobe Ortam Sunucusu daha önceden Adobe Flash CS3 veya Adobe Flash CS4 yazılımlarından hazırlanan karşılıklı konuşma veya görüntü verilerinin aktarımı gibi uygulamalarının Flash Player yardımı ile çok kullanıcılı zengin içerikli Internet uygulamaları arasında – RIAs (Rich Internet Applications) – bağlantı kurmak için kullanılır. Bu yazılım lisanslı bir yazılım olması dikkat edilmesi gereken başka bir hususdur.

7.1.3. Red5 Ortam Sunucusu ile Adobe Flash Ortam Sunucusu arasındaki benzerlikler:

- Her iki sunucuda RTMP ve AMF protokollerini kullanır.
- Her iki sunucuda da Adobe Flash uygulamalarını çalıştırır.

7.1.4. Red5 Ortam Sunucusu ile Adobe Flash Ortam Sunucusu arasındaki farklar:

- Red5 Ortam Sunucusu açık kaynak bir yazılımken; Adobe Flash Ortam Sunucusu lisanslı bir yazılımdır.

- Red5 Ortam Sunucusu açık kaynak olmasından ötürü geliştirilebilir özelliğine sahipken; Adobe Flash Ortam Sunucusu lisanslı yazılım olduğu için sadece Adobe firması tarafından geliştirilir.

- Red5 Ortam Sunucusu hem Adobe Flash uygulamalarını hem de Java uygulamalarını çalıştırır; Adobe Flash Ortam Sunucusu ise sadece Adobe Flash uygulamalarını çalıştırır.

7.1.5. Adobe Flash CS4: Adobe yazılım firmasının çıkarmış olduğu en son yazılım sürümüdür. Eğitsel materyaller ve enteraktif uygulamalar yapılmasında kullanılan iki boyutlu animasyon programıdır. Enteraktif uygulamaların yapılması için Flash yazılımının kullandığı en son kod yazma dili ise Actionscript 3 programlama dilidir. Actionscript 3 programlama dilinden önce kullanımda olan Actionscript 2 ve 1 dilleriyle de her ne kadar enteraktif uygulamalar hazırlanıyor olsa da Actionscript 3 programlama dili ile hazırlanan uygulamalar öncekilere kıyasla daha efektiftir bir sonuç verir.

7.1.6. Hazırlanan yazılım: hazırlanan uygulama için yukarıdaki incelemeler ışığında uygulama için karar kılınan yazılım Flash tabanlı bir uygulama ve bu uygulama ile beraber çalışan RTP (RTTP – Real-time Transport Protocol) prokolünün kullanımına karar verilmiştir. Kısaca RTTP protokolünün çalışma prensibini inceleyelim.

RTTP protokolü gerçek zamanlı ses ve görüntü vb. uygulamalarının iletilmesi için tasarlanan bir protokoldür. Bu protokol kaynak ayrımını (*resource reservation*) yapmaz; diğer taraftan hizmet kalitesinin devamlılığına da güvence vermez (QoS). Bunların yanı sıra herhangi bir tıkanıklığın gerçekleşmesi durumunda ise veri aktarım hızını kontrollü bir şekilde düşürür. Veri akışı esnasında RTCP (RTP Control Protocol) ile beraber çalışır ve böylelikle oturumda yer alan tüm kullanıcılara periyodik olarak kontrol paketleri göndererek alma kalitesi (*reception quality*) ile ilgili bilgi düzenli sağlar.

RTP yoğun iletişim ve özellikleri taşıyan bas-konuş telefon, video telekonferans uygulamaları ve web gibi zamanlı medya, tabanlı dâhil eğlence sistemlerinde kullanılır. Bu medya akışlarını genellikle VoIP uygulamaları için çok fazla tercih edilir. RTP protokolü UDP protokolünün üstünde çalışır ve UDP protokolünün çoğaltma ve sağlama servislerini

kullanır. Her iki protokolde veri transfer protokolünün bir parçası olarak görev yapar. RTP sıklıkla sinyal fonksiyonunu sağlayan SIP ve H.323 ile birlikte kullanılır.

RTP protokolünün ayırt edici bir özelliği de kendi altında bulunan uygun networklarda ya da diğer transfer protokollerinde de çalışmasıdır. RTP bir verinin birden fazla noktaya ulaşmasında yardımcı olur. Yukarıda da değindiğimiz üzere telekonferans, VoIP ve görüntülü konuşma vb. gibi uygulamalarda birden fazla kişiye aynı veri aktarımı gerçekleşir.

RTP protokolü temelde iki ana bölümden oluşmaktadır:

- Gerçek zamanlı veri özelliğini taşıyan verilerin taşınmasında sorumludur.

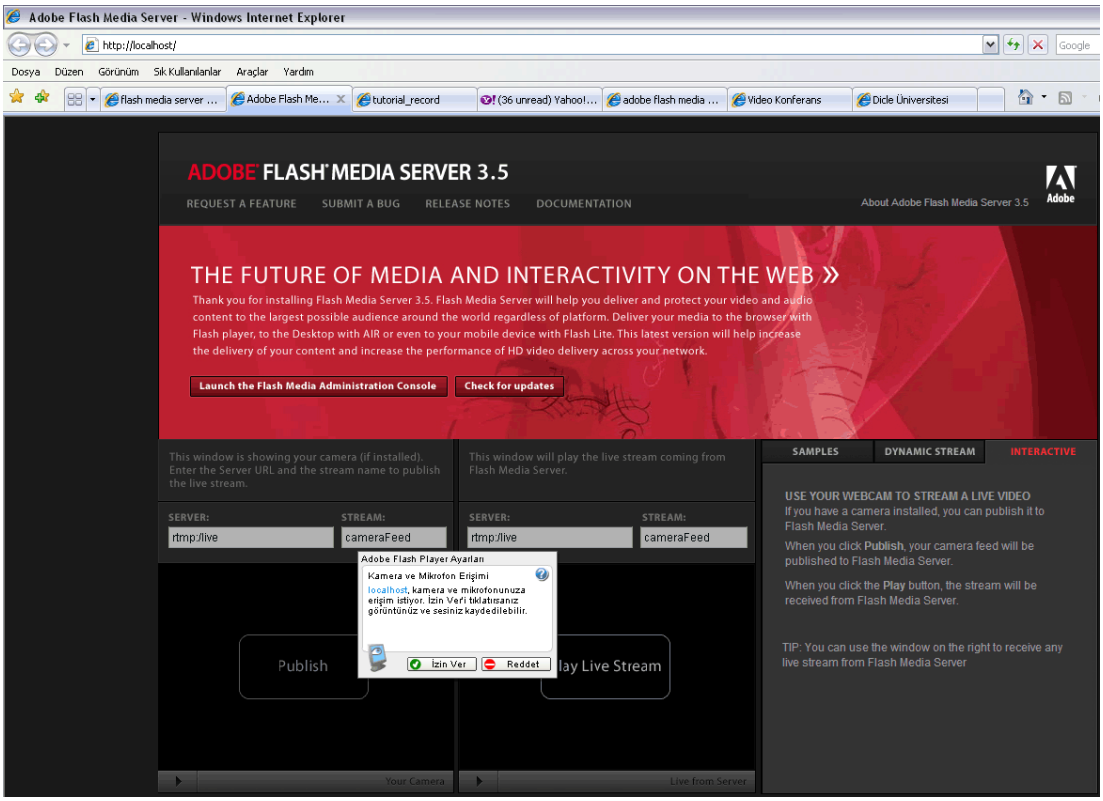
- RTCP kontrol protokolü ile hizmet kalitesini (QoS) ve katılımcıların oturum gidiş hakkındaki bilgilerinin taşındığını kontrol eder [75,76].

Hazırlanabilecek uzaktan eğitim uygulaması için RTTP protokolünün işlevselliği ve Adobe Flash uygulamalarının yaygınlığı göz önüne alınarak bu iki önemli husus ışığında yapılan uygulama aşağıdaki gelişmektedir.

İlk olarak hazırlanan uygulamanın hazırlık aşamasının bir ekran görüntüsü şekil 7.4'de gösterilmektedir. Yazılım programını İnternet üzerinde çalışır vaziyete getirmeden önce lokal bilgisayarda Adobe Flash Ortam Sunucusu çalıştırılır ve sonra kişisel bilgisayarda herhangi boş bir internet sayfası açılarak, açılan sayfanın adres satırına ise "localhost" yazılır. Bu işlemi yapmaktaki amaç bilgisayara bağlı olan web kamerasının doğru çalışıp çalışmadığını kontrol etmektir. Bu işlem için şekil 7.2'te gösterilen sağ tarafta bulunan "INTERACTIVE" butonu tıklanır ve şekil 7.2'te çıkan uyarı ekranında evet butonu tıklanır. Programla beraber daha önceden kurulu olan "SWF" dosyasının çalıştırılmasına izin verildikten sonra "Publish " ve hemen ardından "Play Live Stream" butonlarına tıklanır.

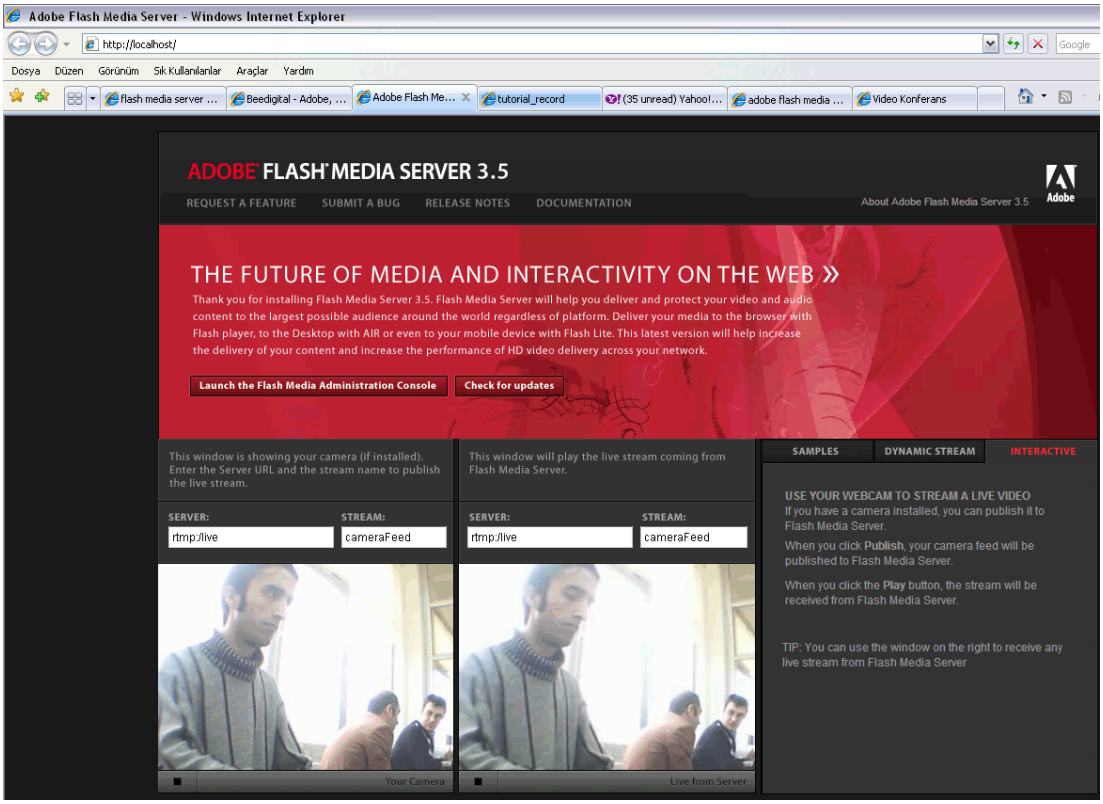
Bu aşamaya kadar çalıştırılan işlem sadece Adobe Flash Ortam Sunucusunun kişisel bilgisayara kurulması ve çalıştırılmasıyla beraber sanal bir sunucu oluşturulmuştur. Hazırlanan sanal sunucu ile kullanıcının bilgisayarında sanal bir İnternet çalıştırılmakta ve

muhtemel bir uygulamanın çalıştırılması durumunda kullanıcıya görüntü transferinin karşı tarafa nasıl iletileceği hakkında bilgi verilmektedir.



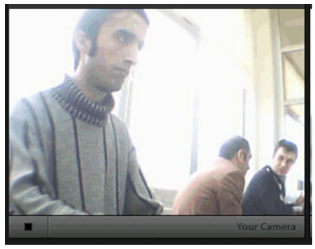
Şekil 7.2. Localhost'ta çalışan sayfanın ilk durumu

"Localhost" sayfasının çalıştırılmasıyla beraber "rttp://live" bölümünde muhtemel herhangi bir görüntünün RTTP protokolü kullanılarak İnternet üzerinden karşı tarafa nasıl iletileceği gösterilmiştir. Flash Ortam Sunucusu yukarıda değindiğimiz üzere RTTP protokolü kullanır. Onay tuşuna basıldığında oluşan görüntü şekil 7.3'te gösterilmiştir. Şekil 7.3'teki resim dikkatle incelendiğinde kameradan gelen görüntü her iki bölümde farklı şekilde yansıtılmaktadır.

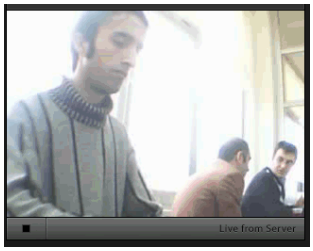


Şekil 7.3. İnternette yayın yapılmadan önceki aşama

Bir önceki paragrafta anlatılmak istenileni biraz daha açıklamak gerekirse her ne kadar İnternet üzerindeki ortam verilerinin aktarımı hızlandırılmaya çalışılsa da günümüzde hiçbir zaman birebir anlık görüntü aktarımı gerçekleştirilmemektedir. Aşağıdaki şekil 7.4.a. ve 7.4.b'deki görüntüler arasındaki zaman farkı birebir görülmektedir.



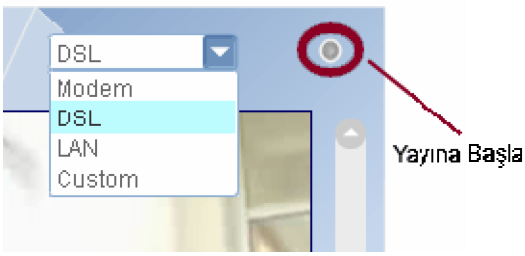
a. Yerel bilgisayardaki anlık görüntü



b. İnternet üzerinden gönderilen görüntü

Şekil 7.4. Yerel bilgisayardaki görüntü ile İnternet üzerindeki görüntü arasındaki zaman farkı

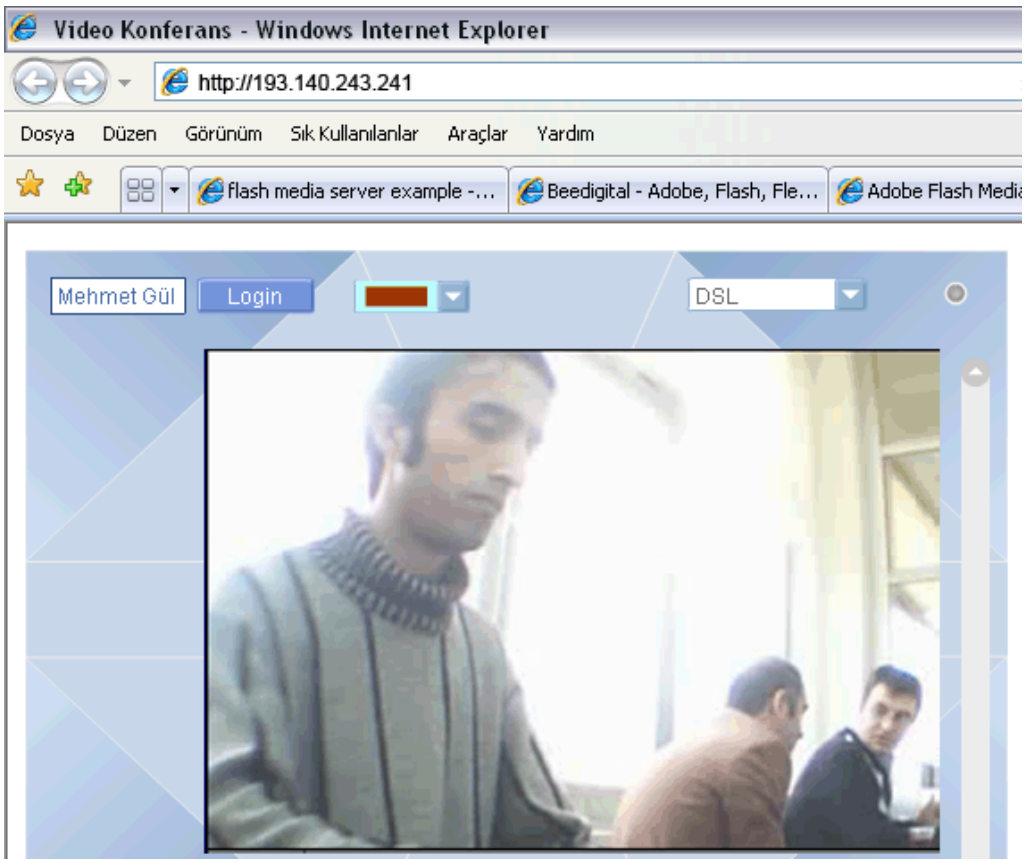
Kullanıcının İnternet hızı hazırlanan uygulamadaki bağlantı hızı kombo kutusundaki değerlerden herhangi biri seçilerek tespit edilir. Ardından yayına başlamak için aşağıdaki şekil 7.5'teki butona "Yayına Başla" basılması gerekmektedir. İnternet üzerinde program çalıştırılmadan önce karşı tarafın görmesi amacıyla bir kullanıcı adı yazılır; kullanıcı adı kişinin gerçek adı olabileceği gibi takma bir adda olabilir. Kişi daha sonra bağlantı hızını belirler. Bağlantı hızının belirtildiği yer aşağıdaki şekil 7.5'te gösterilmektedir.



Şekil 7.5. Bağlantı hızının tespiti

Bu işlemlerin hemen ardından hazırlanan uygulama ya kişisel bilgisayarın sunucu olarak tanıtılması ve sabit IP ile diğer kullanıcılara kendi kişisel bilgisayarı üzerinden yayın yapması gerekmekte ya da Adobe Flash Ortam Sunucusu kurulu olan herhangi bir sunucu üzerinden sabit IP ile yayın yapılması gerekmektedir.

İnternet üzerinde Adobe Flash Ortam Sunucusunun kurulu olduğu bazı özel web sunucuları aracılığıyla da yayın yaptırılabilir. Yayın esnasında veri transferi RTTP protokolünü kullanır. Aşağıdaki şekilde gösterilmekte olan "http://193.140.243.241" IP adresi yerine "rttp://193.140.243.241" adresi de yazılsa sonuç değişmeyecektir, yayın tekrar gösterilecektir. IP adresinin herhangi bir Domain adresine yönlendirilmesiyle de istenilen her yerde IP adresi yerine belli bir domain ismide girilebilir. Örneğin "www.yaziliadresim.com" yazıldığında yönlendirilmiş olan sunucuya IP adresine yönelir ve bağlantı RTTP protokolü aracılığıyla gerçekleşir. Yapılan uygulama İnternete bağlı bir bilgisayarın sabit IP'si üzerinden aşağıdaki şekil 7.6'da gösterilen IP üzerinden gerçekleştirildi. Aşağıdaki şekilde gösterilen uygulama Dicle Üniversitesi Kampüsünde gerçekleştirildi. Kullanılan IP Dicle Üniversitesi networkünde DHCP sunucusu tarafından atanan bir IP'dir.



Şekil 7.6. İnternet üzerinden yayının yapılması

8. SONUÇ:

Şu ana kadar anlatılanlar ışığında ister uzaktan eğitim uygulamaları olsun, ister karşılıklı görüntü transferinin gerçekleştirilmesi amacı taşısın ortam transferini amaçlayan herhangi bir uygulamada hız ve veri güvenliği önemli bir unsurdur.

Hazırlanacak olan uygulamalarda ise dikkat edilmesi gereken bazı önemli unsurlar da bulunmaktadır. Bunlar zaman içerisinde kullanıcıların değişen tercih ve istekleri ışığında şekillenen yazılım uygulamalarıdır. Böyle bir uygulamaya verilebilecek en güzel örnek Flash oynatıcıdır. Flash yazılımının tercih edilmesindeki en önemli faktör iki boyutlu animasyonlu enteraktif uygulamaların yapılıyor olmasıdır. Yapılan bu uygulamaların boyutunun küçük olması düşük İnternet hızlarına sahip kullanıcılar için bile pek fazla sorun teşkil etmemesine yol açar.

Daha önce Macromedia firması tarafından kullanılan ve geliştirilen Flash yazılımı için ayrıca özel bir akış protokolü geliştirilmiş ve halen kullanılıyor olması, Flash uygulamalarının daha hızlı aktarılmasını sağlamaktadır.

Web tabanlı geliştirilecek herhangi bir uygulamada kullanıcının genel tercihlerinin yanı sıra hazırlama aşamasında kullanılacak olan yazılımlarda önemli bir unsurdur. Hazırlık aşamasında genel bir proje için ilk olarak kullanıcı tercihleri, ardından kullanılacak program ya da programlar ve son olarak projenin tüm aşamalardaki maliyet önemli bir yer kaplamaktadır.

Yukarıda da değinilen Adobe Flash uygulamalarına alternatif olarak Microsoft firması tarafından geliştirilmiş Silverlight yazılımı ve uygulamaları karşılaştırılır. En fazla tercih edilen uygulama, birçok heterojen kullanıcısının teknik donanımsal yapıları dikkate alınmadan verilebilecek bir uygulama ile sadece daha kısıtlı bir kullanıcı kitlesine verilebilecek bir uygulama arasında tercih yapılması gerektiğinde daha fazla kişiye hitap eden uygulamalar daha çok tercih edilir. Daha önceden değinildiği gibi Silverlight yazılımı sadece Microsoft işletim sistemleri tarafından kullanılırken Flash uygulamaları Linux,

Unix gibi işletim sistemlerinde bile çalışıyor olması; Silverlight için 23.84 MB lık Framework 2.0 veya üstü versiyonlarının kurulması gerekirken Flash uygulamaları için sadece 1.86 MB lık bir oynatıcının İnternetten indirilip kuruluyor olması vb. gibi özelliklerin Flash uygulamasının İnternet kullanıcıları tarafından oluşturulabilecek herhangi bir uzaktan eğitim uygulaması için daha fazla tercih edilmesine yol açar.

Son olarak hazırlanacak projenin maliyetinin düşürülmesi amaçlanırsa Red5 Ortam Sunucusunun kullanılması gerekir. Red5 Ortam Sunucusu Adobe Flash Ortam Sunucusunun yapmış olduğu tüm görevleri eksiksiz yapmakla kalmaz ayrıca ücretsiz ve açık kaynak olmasından ötürü daha fazla tercih edilir.

Sonuç olarak hazırlanan tezin amacı ilk olarak ortam akışlarının İnternet üzerindeki hareketleri esnasında karşılaştıkları etkiler sıralanmıştır. Bunlar sırayla ortam dosyalarının farklı sıkıştırma metotları ile sıkıştırılmaları ve bu sıkıştırma algoritmalarının ortam akışları üzerindeki etkileri incelendi. Genellikle kullanıcı bilgisayarlarında kullanılmakta olan güvenlik duvarlarının aslında İnternette protokollerini de koruduğu ve bu güvenlik duvarlarının İnternet üzerinde yayın yapan her türlü dosyayı kontrol ettiğine, ortam dosyası dâhil, değinildi. Ortam dosyaları genellikle UDP protokolü üzerinden akışı gerçekleştirirken UDP protokolüne alternatif olarak geliştirilen yöntemlerden biride ortam dosyasının TCP protokolü üzerinden gönderimidir. Gelişen teknolojiler ışığında kullanımı artan kablosuz uygulamalar üzerinden ortam akışı yapılması için geliştirilen yöntemler üzerinde durulduktan sonra son olarak vekil sunucuların ortam akışları üzerindeki etkileri üzerinde durulmuştu.

Yayınlanan ortam dosyalarının akışları esnasında kullandıkları yollar ve metotlar incelendikten sonra hazırlanan yazılım için en uygun seçenekler yukarıda sözü geçen önemli hususlarda dikkate alınarak kullanışlı ve düşük maliyetli bir yazılım ortaya çıkarmaya çalışıldı.

8. KAYNAKLAR

[1] **R. Baecker, P. Wolf, K. Rankin**, "The ePresence Interactive Webcasting and Archiving System: Technology Overview and Current Research Issues", Proc. Elearn, 2004, pages 2532-2537

[2] **A. SAID, W. A. Pearlman**, "A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees", IEEE Transactions on Circuits and System for Video Technology, Vol. 6, No. 3, June 1996, pages 243-250

[3] Codec, Wikipedia, http://en.wikipedia.org/wiki/Main_Page, 2008

[4] **J. Lorance**, "Webcasting and Web Streaming - Get your videos out to your online audience", http://www.techsoup.org/learningcenter/software/archives/page9965.cfm, October 30, 2003

[5] Streaming media protocols, http://all-streaming-media.com/faq/streaming-media/faq-streaming-media-protocols.htm, 2008

[6] **T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, F. Yergeau** "Extensible Markup Language(XML) 1.0", W3C, http://www.w3.org/TR/2008/REC-xml-20081126/, 26 November 2008

[7] **M. Altınel, M. J. Franklin**, "Efficient Filtering of XML Documents for Selective Dissemination of Information", Proceedings of the 26th VLDB Conference, Cairo, Egypt, 2000, pages 53-64

[8] **A. Singh**, "XFilter / YFilter", 2007, pages 1-35

[9] **D. TAŞKIN, N. SUÇSUZ, C. TAŞKIN**, "MPEG Akımını Geçici Referans Numaralarını Kullanarak Şifreleme", 2007

[10] **D. TAŞKIN, N. SUÇSUZ**, "Sıkıştırılmış Ortamda Çerçeve Tipine Dayalı Gerçek Zamanlı Sahne Değişimi Belirleme", 2006

[11] **K. M. RAO, S. K. MITRA**, "A Study on The Application of Color Transfer Technique for Video Compression", 2007

[12] **A. J. MARCELLA, D. MENENDEZ**, "Cyber forensics: a field manual for collecting, examining, and preserving evidence of computer crimes", 2008, pages 310-313

[13] **W. SIMPSON**, "Video over IP: a practical guide to technology and applications", 2006

[14] CISCO SYSTEM, "Cisco's PIX Firewall Series and Stateful Firewall Security", 1997, pages 1-12

[15] **J. BRASSIL, P. SHARMA**, Hewlett-Packard Laboratories, "Streaming Media via TCP: Approaches to Bandwidth Sharing and Rate Adjustment", 2002

[16] **E. AKSAN, S. DOĞAN**, "Dalgacık Tabanlı Görüntü Sıkıştırma Tekniği", Temmuz 2004, Havacılık ve Uzay Teknolojileri Dergisi, 47-53

[17] **A. ALAYBEYOĞLU, Y. BAYRAKDAR, A. KANTARCI**, "Tıkanıklık Kontrolü için Yeni Bir İletim Protokolü: DCCP", 2008, Akademik Bilişim, 621-626

[18] **E. KOHLER, M. HANDLEY, S. FLOYD**, "Datagram Congestion Control Protocol", ftp://ftp.rfc-editor.org/in-notes/rfc4340.txt, 2009

[19] **K. K. RAMAKRISHNAN, S. FLOYD, D. BLACK**, Sept., "The addition of explicit congestion notifcation (ECN) to IP. RFC 3168"; Internet Engineering Task Force, 2001

[20] **E. KOHLER, M. HANDLEY, S. FLOYD**, "Designing DCCP: Congestion Control Without Reliability", 2006

[21] **A. KANTARCI**, "İnternet Üzerinde Video Akışlandırma Teknolojilerinde Son Yaklaşımlar Üzerine Bir İnceleme", 2002, pages 75-84

[22] **Ö. ZEYDAN**, "Kişisel Bilgisayarlarda İnternet Güvenliği", 2008

[23] **D. AKKUŞ**, "Internet'e Bağlanırken Gerekenler: Firewall ve Proxy", http://www.belgeler.org/howto/proxy-fw.html, Ocak 2003

[24] **İ. TELLİOĞLU**, "WLAN'da VoIP Performansını Artırıcı Yaklaşımlar", Ocak 2008

[25] **S. DAHAR**, "Sniffers Basics and Detection", http://www.linuxsecurity.com/content/view/123570/49/, 2006

[26] **D. SANAI**, "Detection of Promiscuous Nodes Using ARP Packets", August 2001, pages 1-13

[27] **F. AKGÜN, E. BULUŞ, Ş. ŞEN**, "Bilgisayar ağları üzerinde iletilen verilere zarar vermek için kullanılan önemli teknikler ve korunma yollarının incelenmesi", 2005

[28] **C. PERKINS**, "IP Mobility Support for IPv4", August 2002

[29] **M. YUMURTACI, A. KEÇEBAŞ**, "Akıllı Ev Teknolojileri ve Otomasyon Sistemleri", Mayıs 2009,

[30] **R. REJAIE, M. HANDLEY, H. Y., D. ESTRIN**, "Proxy Caching Mechanism for Multimedia Playback Streams in the Internet", January 1999, pages 1-10

[31] **R. REJAIE, M. HANDLEY, D. ESTRIN**, "RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the internet", 2005, pages 1337-1345

[32] **P. K. CHUNG**, “An Overview Of Multimedia Proxy Servers”, 2004, pages 34-39

[33] **A. C. BEGEN, Y. ALTUNBAŞAK**, “Timely inference of late/lost packets in real-time streaming applications, in Picture Coding Symp. (PCS)”, 2004

[34] **A. C. BEGEN, M. A. BEGEN, Y. ALTUNBAŞAK, M. R. CIVANLAR**, “Proxy Selection for Interactive Video”, 2006

[35] **P. FOULIRAS, A. MANITSARIS, P. FOTARIS**, “OMTP: Combining TCP And UDP For Multimedia Streaming”, 2004, pages 9-13

[36] **S. TULLIMAS, T. NGUYEN, R. EDGECOMBE, S. C. CHEUNG**, “Multimedia Streaming Using Multiple TCP Connections”, 2005

[37] **S. JOSHI, D. A. DON**, “Multimedia Streaming; Can TCP Handle It?”, 2008

[38] **S. FLOYD, M. HANDLEY, J. PADHYE, J. WIDMER**, “Equation-Based Congestion Control for Unicast Applications”, 2000

[39] **S. J. WEE, J. G. APOSTOLOPOULOS**, “Secure Scalable Video Streaming For Wireless Networks”, May 2001

[40] **P. DE, S. SHARMA, A. SHUVALOV, T. C. CHIUEH**, “WiVision: A Wireless Video System for Real-Time Distribution and On-Demand Playback”, 2004

[41] **G. M. MUNTEAN, N. CRANLEY**, “User Quality of Experience-aware Multimedia Streaming over Wireless Home Area Network”, 2007

[42] **L. U. CHOI, E. STEINBACH, W. KELLERER**, “Cross Layer Optimization For Wireless Multi-User Video Streaming”, 2005

[43] **M. T. IVRLAC**, “Parameter selection for the Gilbert-Elliott model”, May 2003

[44] **S. KHANTT, E. STEINBACHT, M. SGROI, W. KELLERER**, “Cross-Layer Optimization for Wireless Video Streaming – Performance and Cost”, 2005

[45] **Y. PENG, S. KHAN, E. STEINBACH, M. SGROI, W. KELLERER**, “Adaptive resource allocation and frame scheduling for adaptive multi-user video streaming”, 2005

[46] **C. KRASIC, J. WALPOLE**, “Priority-Progress Streaming for Quality-Adaptive Multimedia”, 2001

[47] **S. H. CHOI, M. HANDLEY**, “Fairer TCP-Friendly Congestion Control Protocol for Multimedia Streaming Applications”, 2007

[48] **B. SHEN, S. J. LEE, S. BASU**, “Streaming Media Caching with Transcoding-Enabled Proxies”, 2003

[49] **Y. GUO, S. SEN, D**. , “Prefix Caching assisted Periodic Broadcast for Streaming Popular Videos”, 2002

[50] ThreatExpert, http://www.threatexpert.com/files/video.exe.html, 2009

[51] **V. PODLOZHNYUK**, “Histogram calculation in CUDA”, November 2007

[52] **B. A. ATTEA, S. M. HAMEED, A. D. ABOUD**, “Content-based Grayscale Image Colorization”, 2006

[53] **T. WELSH, M. ASHIKHMIN, K. MUELLER**, “Transferring Color to Greyscale Images”, 2006

[54] **T. CHEN, Y. WANG, V. SCHILLINGS**, “GRAYSCALE IMAGE MATTING AND COLORIZATION”, 2004, pages 1164-1169

[55] **N. B. CHOPADE, A. A. GHATOL**, “Analysis of discrete wavelet based image compression technique: A review”, November 2009, 915-919

[56] **E. KOHLER**, “Datagram Congestion Control Protocol (DCCP) Congestion Control ID 3 Dropped Packets Option”, October 2006,

[57] **E. KOHLER, M. HANDLEY, S. FLOYD**, "Designing DCCP: Congestion Control Without Reliability"; 2006

[58] **N. CARDOT, M. TRAMPUŠ, K. WOLFF**; "Datagram Congestion Control Protocol (DCCP)"; 2007

[59] Real Time Streaming Protocol, RTSP; http://en.wikipedia.org/wiki/Real_Time_Streaming_Protocol; 2009

[60] Microsoft Media Server, MMS; http://en.wikipedia.org/wiki/Microsoft_Media_Server; 2007

[61] Standard Generalized Markup Language, SGML; http://en.wikipedia.org/wiki/Standard_Generalized_Markup_Language; 2009

[62] Voice Over Internet Protocol (VoIP), 2009

[63] **S. MANWANI**, "ARP Cache Poisoning Detection and Prevention", 2003

[64] **L. MASSOULI, J. ROBERTS**, "Bandwidth sharing: objectives and algorithms" October 2009

[65] **B. K. KANCHERLA, G. M. NARAYAN, K. GOPINATH**, "Performance Evaluation of Multiple TCP connections in iSCSI", 2008

[66] **P. O'HANLON**, "A brief look at using multiple TCP connections for real-time flows", 2004

[67] **K. A. BEGAIN, C. BALAKRISHNA, L. A. GALINDO, D. MORO**, "The Way Forward – Paths to Follow", 2009

[68] **E. Kohler**,; "Faster Restart for TCP Friendly Rate Control (TFRC)", June 2006

[69] **S. FLOYD, M HANDLEY, J. PADHYE, J. WIDMER**, "Equation-Based Congestion Control for Unicast Applications"; 2000

[70] **L. B. D. PINHO, C. L. D. AMORIM**, “Investigating the Performance of Video-on-Demand Systems over WLANs Using Generic Association Control”, 2005, pages 187-198

[71] **M. SIKANDAR, H. KHIYAL, A. KHAN, E. SHEHZADI**, “SMS Based Wireless Home Appliance Control System (HACS) for Automating Appliances and Security”, 2009, pages 887-894

[72] **A. M. ESKICIOGLU, S. DEXTER, E. J. DELP**, “Protection of multicast scalable video by secret sharing: simulation results”, 2004

[73] **G. M. MUNTEAN, P. PERRY, L. MURPHY**, “Quality-Oriented Adaptation Scheme (QOAS) for High Bit-rate Multimedia Streaming”; 2004

[74] http://www.adobe.com/products/flashplayer/faq/

[75] http://www.javvin.com/protocolRTP.html

[76] http://en.wikipedia.org/wiki/Real-time_Transport_Protocol

ÖZGEÇMİŞ

Adı Soyadı : Mehmet GÜL

Doğum Yeri: Diyarbakır

Doğum Tarihi: 24/03/1982

Medeni Hali: Bekar

Yabancı Dili: İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise : Anadolu Teknik Lisesi

Lisans : Ortadoğu Teknik Üniversitesi – Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü

Yüksek Lisans :

Çalıştığı Kurum/Kurumlar ve Yıl: Dicle Üniversitesi Silvan Meslek Yüksekokulu.