



EGE ÜNİVERSİTESİ

YÜKSEK LİSANS TEZİ

**OPENFLOW AĞLARI ÜZERİNDE ÇALIŞAN
ÇIKARSAMAYA DAYALI UÇTAN UCA
ÇOKLUORTAM AKIŞLANDIRMA UYGULAMASI**

Tuba UZAKGİDER

Tez Danışmanı: Yrd. Doç. Dr. Müge SAYIT

Uluslararası Bilgisayar Anabilim Dalı

**Bilim Dalı Kodu: 619.02.04.
Sunuş Tarihi: 16.03.2015**

E. Ü. FEN BİLİMLERİ ENSTİTÜSÜ

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

(YÜKSEK LİSANS TEZİ)

**OPENFLOW AĞLARI ÜZERİNDE ÇALIŞAN
ÇIKARSAMAYA DAYALI UÇTAN UCA
ÇOKLUORTAM AKIŞLANDIRMA UYGULAMASI**

Tuba UZAKGİDER

Tez Danışmanı : Yrd. Doç. Dr. Müge SAYIT

Uluslararası Bilgisayar Anabilim Dalı

Bilim Dalı Kodu : 619.02.04

Sunuş Tarihi : 16.03.2015

**Bornova-İZMİR
2015**

Tuba UZAKGİDER tarafından YÜKSEK LİSANS tezi olarak sunulan “OpenFlow Ağları Üzerinde Çalışan Çıkarsamaya Dayalı Uçtan Uca Çokluortam Akışlandırma Uygulaması” başlıklı bu çalışma E.Ü. Lisansüstü Eğitim ve Öğretim Yönetmeliği ile E.Ü. Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi’nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve 16/03/2015 tarihinde yapılan tez savunma sınavında aday oybirliği/oyçokluğu ile başarılı bulunmuştur.

Jüri Üyeleri:

İmza

Jüri Başkanı

: Yrd. Doç Dr. Müge SAYIT

.....

Raportör Üye

: Yrd. Doç. Dr. Deniz ÇOKUSLU

.....

Üye

: Doç. Dr. Muhammed CİNSDİKİCİ

.....

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

ETİK KURALLARA UYGUNLUK BEYANI

E.Ü. Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili hükümleri uyarınca Yüksek Lisans Tezi / Doktora Tezi olarak sunduğum “OpenFlow Ağları Üzerinde Çalışan Çıkarsamaya Dayalı Uçtan Uca Çokluortam Akışlandırma Uygulaması” başlıklı bu tezin kendi çalışmam olduğunu, sunduğum tüm sonuç, doküman, bilgi ve belgeleri bizzat ve bu tez çalışması kapsamında elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara atıf yaptığımı ve bunları kaynaklar listesinde usulüne uygun olarak verdiğimi, tez çalışması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını, bu tezin herhangi bir bölümünü bu üniversite veya diğer bir üniversitede başka bir tez çalışması içinde sunmadığımı, bu tezin planlanmasından yazımına kadar bütün safhalarda bilimsel etik kurallarına uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul edeceğimi beyan ederim.

16 / 03 / 2015

Tuba Uzakgider

ÖZET**OPENFLOW AĞLARI ÜZERİNDE ÇALIŞAN ÇIKARSAMAYA
DAYALI UÇTAN UCA ÇOKLUORTAM AKIŞLANDIRMA
UYGULAMASI**

UZAKGİDER, Tuba

Yüksek Lisans Tezi, Uluslararası Bilgisayar Anabilim Dalı

Tez Danışmanı: Yrd. Doç. Dr. Müge SAYIT

Mart 2015, 38 sayfa

Klasik ağ mimarisinden farklı bir yaklaşım olarak, kontrol düzlemini ve veri düzlemini birbirinden ayıran "Yazılım Tanımlı Ağlar (ing. Software Defined Networking-SDN)", günümüzde popülerlik kazanmış konular arasında gelmektedir. Bu tez çalışmasında, YTA üzerinde çalışan video akışlandırma uygulamaları için pekiştirmeli öğrenme tabanlı bir yaklaşım geliştirilmiştir. Önerilen sistemde, trafik akışlarının yeniden yönlendirilmesi ve video gönderim kalitesinin değiştirilmesi için en iyi zamanı belirleyen bir öğrenme modeli tasarlanmıştır. Öğrenme modeli, dinamik olarak yönlendirme ve video kalitesi uyarlaması yaparken, paket kayıp oranını, kalite değişimlerini ve kontrol birimi maliyetini minimum seviyeye çekmeye amaçlamaktadır. Öğrenme tabanlı bu yaklaşım, en kısa yol yönlendirmesi ve aç-gözlü yaklaşım ile karşılaştırılarak test edilmiştir. Sonuçlar, önerilen sistemin farklı ağ senaryolarında, Deneyim Kalitesi (Quality of Experience- QoE) ve ağ maliyeti bakımından önemli ölçüde daha iyi başarıma sahip olduğunu göstermektedir.

Anahtar Sözcükler: Yazılım Tanımlı Ağlar (SDN), Pekiştirmeli öğrenme, UDP Üzerinden Video Akışlandırma

ABSTRACT**AN INFERENCE BASED END-TO-END VIDEO STREAMING
APPLICATION RUNNING OVER OPENFLOW NETWORKS**

UZAKGİDER, Tuba

Msc in Department of International Computer.

Supervisor: Asst. Prof. Dr. Müge SAYIT

March 2015, 38 pages

As a different approach from the conventional network architecture, Software Defined Networks (SDN), which decouples the control and data plane, is one of the topics that have gain popularity nowadays. In this thesis, Reinforcement Learning based approach for adaptive video streaming systems running over SDN is proposed. In the proposed system, a learning model is developed in order to determine the optimal time to re-route traffic flows and to change the bitrate of the video. The learning model aims to minimize packet loss rate, quality changes and controller cost while dynamically adapts the flow routes and video quality. the performance of this learning based approach is tested by comparing it to the shortest path routing and a greedy approach. The results show that the proposed system significantly outperforms these approaches in terms of Quality of Experience (QoE) and network cost under different network scenarios.

Keywords: Software Defined Networks (SDN), Reinforcement Learning, Video Streaming Over UDP

TEŞEKKÜR

Yüksek lisans çalışmam süresince kıymetli fikirlerini ve bilgisini benimle paylaşan, her konuda bana sabırla katlanan, her daim desteğini yürekten hissettiğim ve bana kendi deneyimleri ile çalışmam için imkan sağlayan, beni araştırmaya ve geliştirmeye yönlendiren sayın danışmanım Yrd. Doç. Dr. Müge SAYIT'a teşekkürlerimi bir borç bilirim. Yüksek lisansım boyunca yardımlarını benden esirgemeyen ve benimle birlikte çözüm yolları arayarak yanımda olan sayın hocam Prof. Dr. Mehmet Emin DALKILIÇ'a teşekkürlerimi sunarım.

Tez çalışmam boyunca yolumu aydınlatan, fikirlerini ve bilgisini benden esirgemeyen Cihat CETİNKAYA'ya; yüksek lisansım süresince beni araştırmaya yönlendiren, hedeflere ulaşmam için yol gösteren tüm kıymetli hocalarıma teşekkür ederim. Ayrıca Bornova Altay Mesleki ve Teknik Anadolu Lisesi Bilişim Teknolojileri Alanı öğretmen arkadaşlarıma destekleri için teşekkürü bir borç bilirim.

Yüksek lisans öğrenimimde beni maddi olarak destekleyen Türkiye Bilimsel ve Teknolojik Araştırma Kurumu'na (TÜBİTAK) teşekkürlerimi sunarım.

Yüksek lisans eğitimim boyunca bana her zaman destek olan, bana güvenip inanan, her anımda benimle birlikte heyecanlanıp, benimle birlikte mutlu olan canım anneme, canım babama ve hep yanımda olan canım kardeşime; bu süreçte beni hiç yalnız bırakmayan sevgili nişanlıma çok teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET	VII
ABSTRACT	IX
TEŞEKKÜR	XI
İÇİNDEKİLER	XIII
ŞEKİLLER DİZİNİ	XV
ÇİZELGELER DİZİNİ	XVII
SİMGELER VE KISALTMALAR DİZİNİ	XIX
SİMGELER VE KISALTMALAR DİZİNİ (DEVAMI)	XX
1. GİRİŞ	1
2. TEMEL KAVRAMLAR	3
2.1 Çokluortam Akışlandırma (Multimedia Streaming).....	3
2.2 Uyarlamalı Bit Hızı Video Akışlandırma (Adaptive Bitrate Video Streaming)5	
2.3 Yazılım Tanımlı Ağ (YTA, ing. Software Defined Network- SDN)	6
2.3.1 OpenFlow protokolü	7
3. LİTERATÜR ÖZETİ.....	11
4. SİSTEM ALTYAPISI	15

İÇİNDEKİLER (Devamı)

Sayfa

4.1 Video Akışlandırma Uygulaması.....	15
4.2 YTA Mimarisi.....	15
4.3 YTA Üzerinden Uyarlamalı Akışlandırma İçin Pekiştirmeli Öğrenme.....	17
4.3.1 Pekiştirmeli öğrenme ve Q-öğrenme.....	17
4.3.2 YTA kontrol birimi için pekiştirmeli öğrenme modeli tasarımı	19
5. YTA AĞLAR ÜZERİNDE UYARLAMALI VE ÖĞRENME TABANLI ÇOKLUORTAM AKIŞLANDIRMA UYGULAMASI.....	25
5.1 Ağ Topolojisinin Oluşturulması.....	25
5.2 Öğrenme Parametrelerinin Ayarlanması.....	27
5.3 Karşılaştırmalı Test Sonuçları.....	29
6. SONUÇ VE ÖNERİLER	35
KAYNAKLAR DİZİNİ	37
ÖZGEÇMİŞ	41

ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
2.1 Yazılım Tanımlı Ağ Yapısı (ONF White Paper, 2012).....	8
2.2 OpenFlow Komut Seti Örneği (ONF White Paper, 2012)	8
2.3 Anahtar cihaz kayıtlanması (ONF White Paper, 2012).....	9
2.4 Kontrol birimi-anahtar cihaz hayatta mısın mesajları (ONF White Paper, 2012).....	10
2.5 Zaman aşımı mesajı (ONF White Paper, 2012).....	10
4.1 Tez çalışmasında kullanılan YTA mimarisi	16
5.1 Benzetimde kullanılan ağ topolojisi	26
5.2 Ortalama kare kayıp yüzdeleri.....	30
5.3 Zamanla gözlemlenen kalite değişiklikleri	31
5.4 Ortalama iletim hızı	32
5.5 Ortalama kalite değişimi.....	33
5.6 Ortalama yol değişimi	33

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
4.1 Eylem Tanımlamaları	21
4.2 s_{ij} durumunda i ve j değerlerine göre uygun eylem listesi.....	22
5.1 Video Katmalarının Bit Hızı.....	27
5.2 Farklı sıcaklık değerleri için benzetim sonuçları.....	28
5.3 Farklı azaltma faktörü değerleri için benzetim sonuçları	28

SİMGELER VE KISALTMALAR DİZİNİ

<u>Simgeler</u>	<u>Açıklama</u>
a_t	t anındaki eylem
s_t	t anındaki durum
r_{t+1}	t+1 anındaki ödül
kg	kullanılabilir bant genişliği (ing. available bandwidth)
α	Kontrol birimi maliyeti
β	Kalite Değişim Maliyeti
 <u>Kısaltmalar</u>	
YTA	Yazılım Tanımlı Ağ (ing. Software Defined Network - SDN)
BT	Bilişim Teknolojileri
QoS	Hizmet Kalitesi (ing. Quality of Service)
QoE	Kullanıcı Deneyim Kalitesi (ing. Quality of Experience)
TCP	İletim Denetimi Protokolü (ing. Transmission Control Protocol)
HTTP	Hiper Metin Transfer Protokolü (ing. Hyper-Text Transfer Protocol)
DASH	HTTP Üzerinden Dinamik Uyarlamalı Akışlandırma (ing. Dynamic Adaptive Streaming over HTTP)

SİMGELER VE KISALTMALAR DİZİNİ (Devamı)**Kısaltmalar**

SVC	Ölçeklenebilir Video Kodlama (ing. Scalable Video Coding)
UPA	Uygulama Programlama Arayüzü (ing. Application Programming Interface- API)
UDP	Kullanıcı Veri Bloğu İletişim Kuralları (ing. User Datagram Protocol)
RTP	Gerçek Zamanlı Transfer Protokolü (ing. Real Time Transport Protocol)
PSNR	Tepe Sinyal-Gürültü Oranı (ing. Peak Signal-to-Noise Ratio)
Kbps	Kbit Per Second
Mbps	Mbit Per Second

1. GİRİŞ

Günümüz internet mimarisinde mevcut ağın, olası değişimlere hazırlanması, ya da yapılacak bir değişikliğin ağa entegre edilmesi işlemi çok uzun zamanlara ve çok fazla iş gücünün harcanmasına neden olmaktadır. Ağa yeni bir cihaz ekleme işleminde bile BT (Bilişim Teknolojileri) uzmanları çok sayıda anahtara (ing. switch), yönlendiricilere (ing. router), Web doğrulama portallarına dokunmak zorunda kalmaktadır.

Klasik ağ mimarilerinde yaşanan problemlere çözüm olabilmesi için kontrol düzleminin ve veri düzleminin ayrılması günümüzde üzerinde çalışılan popüler konular arasında gelmektedir. Bu kapsamda, yapılan bu tez çalışmasında Yazılım Tanımlı Ağ (ing. Software Defined Network-SDN) kullanılmıştır. Yazılım Tanımlı Ağ'larda kontrol düzlemini "kontrol birimi" adı verilen cihazlar yönetirken, yönlendiriciler sadece gelen komutlara göre paketleri yönlendirme işlemi yapmaktadır (ONF White Paper, 2012).

Yapılan çalışma kapsamında, günümüz İnternet ortamında kullanılan "en kısa yoldan gönderme" eyleminin her zaman iyi sonuçlar doğurmadığı ve hizmet niteliğinin (ing. Quality of Service – QoS)'nın sağlanabilmesi için iletim yollarında oluşabilecek tıkanıklıklara karşı yeniden yönlendirme ile çokluortam akışlarını minimum düzeyde gecikme ile iletilmesi hedeflenmiştir.

YTA ağı üzerinde kontrol birimi ile yönlendiriciler arasındaki mesajlaşmalar için OpenFlow protokolü kullanılmaktadır (ONF White Paper, 2012). Bu çalışmada, kontrol birimi ağdan gerekli istatistikleri alarak, çoklu ortam uygulamalarının hizmet niteliğini sağlayabilecek yolları belirlemektedir. Belirleme işlemi yapılırken, kontrol birimi öğrenme algoritması kullanılmaktadır. Öğrenme algoritması olarak "Pekiştirmeli Öğrenme (ing. Reinforcement Learning)" kullanılmaktadır. Bu öğrenme algoritmasında, yapılan her eylem sonucunda ödül (ing. reward) olarak bulunduğu durumda seçmesi gereken en uygun eylemi öğrenmiş olacaktır. Başlangıçta durum ile ilgili bir bilgi olmadığından, kötü eylemlerin sonucunda alacağı düşük veya eksi ödüller için, tekrar aynı duruma geldiğinde bu eylemin seçilme olasılığı azaltılmış olacaktır.

Belli bir süre sonra içinde bulunduđu durum için seçilebilecek en iyi eylemi seçerek en uygun sonuç bulunmuş olacaktır.

Bu çalışmada hizmet niteliğinin sağlanabilmesi için yeniden yönlendirme işleminin yanı sıra, video verisini barındıran sunucu cihazın, video gönderim bit hızı da gerekli görüldüğü takdirde değiştirilmektedir. Böylece YTA ağ üzerinde uyarlamalı bir çalışma oluşturulmaktadır. Yolun tıkanık olması durumunda, kontrol biriminin gerek gördüğü durumlarda, sunucu cihazın video gönderim hızını azaltarak, yol değiştirme maliyeti olmadan hizmet niteliğinin sağlanabilmesi söz konusudur. Tez kapsamında yapılan çalışmanın literatürden en temel farkı, YTA ağları üzerinde UDP ile gönderim yapan video akışlandırma sisteminde istemcilerin izlediği video kalitesini hem akış yollarının değişimi ile hem de hız uyarlaması ile sağlamasıdır.

Tezin sonraki bölümleri sırası ile aşağıdaki gibi düzenlenmiştir; İkinci bölümde tez çalışması ile ilgili temel kavramlar üzerinde durulmaktadır. Üçüncü bölümde konu ile ilgili literatür taraması sunulmuştur. Dördüncü bölümde sistem altyapısının oluşturulması anlatılmıştır. Beşinci bölümde YTA ağlar üzerinde uyarlamalı ve öğrenme tabanlı çokluortam akışlandırma uygulaması detayları anlatılmış ve diğer sistemler ile karşılaştırmalı sonuçları verilmiştir. Altıncı ve son bölümde ise tez çalışmasında elde edilen sonuçlar ve ileride yapılabilecek çalışmalara yer verilmiştir.

2. TEMEL KAVRAMLAR

2.1 Çokluortam Akışlandırma (ing. Multimedia Streaming)

Çokluortam akışlandırma uygulamaları paket kayıplarına karşı toleranslı olsa da gecikmelere karşı yüksek düzeyde duyarlı uygulamalardır. Bu nedenle, hizmet gereksinimleri açısından, çokluortam uygulamaları, statik içerik gönderim uygulamalarına taban tabana bir zıtlık oluşturmaktadır. Çokluortam akışlandırma uygulamalarının günümüz Internet ortamının sağladığı en iyi gönderimde (best effort) garanti edilemeyen bir katılıkla düşük gecikme ve yüksek bit hızında iletilme gereksinimleri vardır. Çokluortam akışlandırma sistemlerinin başarımı kullanıcı deneyim kalitesi (ing. Quality of Experience - QoE) ile ölçülmektedir. Kullanıcı deneyim kalitesi parametreleri genel olarak PSNR, video duraklama süresi, kayıp paket oranı, başlangıç bekleme zamanı, kalite değişim sıklığı olarak listelenebilir. İstemcilere birim zamanda gelen paket sayısı (ortalama bit hızı), PSNR, video duraklama süresi gibi parametrelerle ilişkili olduğu için bazı sistemlerin başarımı ortalama bithızı değerlendirilerek de ölçülebilir.

Video akışlandırma uygulamaları, iletim katmanı protokolü olarak iletim denetimi protokolü (ing. Transmission Control Protocol-TCP) üzerinden veri gönderimini sağladığı güvenilir veri gönderim altyapısı sebebiyle tercih edebilir. Son dönemde HTTP üzerinden video akışlandırma verisinin gönderimi HTTP'nin TCP üzerinde çalışan bir protokol olması ve güvenlik duvarlarından kolayca geçebilmesi sebebiyle yaygın olarak tercih edilmektedir. TCP protokolü kullanımının bir başka avantajı, aynı hatları kullanan kullanıcılar arasında hat kapasitesinin adaletli bir şekilde paylaşımını yapabilmesidir (Kurose and Ross, 2013).

Diğer yandan, TCP tarafından uygulanan sıkışıklık kontrolü video akışlandırma uygulamasının veri gönderim hızını kısıtladığı için UDP üzerinden gönderim de birçok uygulama tarafından tercih edilmektedir. Örneğin Skype, UDP üzerinden video verisi göndermekte, güvenlik duvarına takıldığı noktalarda TCP ile gönderime geçmektedir (Kurose and Ross, 2013). TCP üzerinden akışlandırma yapıldığı takdirde, akışlandırma başlamadan önce TCP bağlantısının

kurulumu yapılmalıdır. Video akışlandırma uygulamalarında başlangıç bekleme zamanı kullanıcı deneyim kalitesini etkileyen önemli bir parametredir ve TCP kurulum süreci başlangıç bekleme zamanının uzamasına sebep olmaktadır. UDP'de iletme başlamadan önce gönderici ve alıcı tarafın anlaşmaya yani el sıkışma mesajlarını göndermeye ihtiyaç yoktur. Akış kontrolü ve tıkanıklık kontrolü olmadığından güvenilir değildir. UDP gibi veri bloğu protokolleri; medya akışlarını küçük paket serileri şeklinde gönderir. Bu basit ve etkilidir, fakat burada paketin hedefe ulaşmasını garanti eden bir mekanizma yoktur. UDP'de, TCP gibi sıralı bir gönderim söz konusu değildir. Dolayısıyla, uygulama katmanında paketlerin sıralanması ve paket kayıplarını minimuma indirmek için gerektiğinde yeniden gönderimi ya da hata kurtarma yöntemleri gibi yaklaşımların kullanılması gereklidir.

Video akışlandırma uygulamaları için altyapı oluşturan RTP (ing. Real Time Transport Protokol) protokolü ise UDP üzerinde çalışmaktadır (Kurose and Ross, 2013). Gönderici tarafında, video verisi RTP paketi içine konulur ve RTP paketleri de UDP segmentinin içinde bir alt katmana gönderilir. Alıcı tarafında ise sırasıyla UDP segmentinden çıkarılan RTP paketinin içindeki video verisi alınarak çözülmesi ve işlenmesi için medya oynatıcısına gönderilir. RTP, video verilerinin zamanında ulaşmasını veya QoS'ı garantilemez. Bu protokolde sırasız iletilecek paketlerin önlenmesini sağlayacak bir mekanizma da bulunmamaktadır. Veri iletimi sırasında iki uç arasında bir RTP oturumu kurulur. Her farklı medya türü için cihazlar arası farklı bir oturum oluşturulur. RTCP (Real Time Transport Control Protocol) ise RTP ile birlikte kullanılan ve akışlandırma oturumuyla ilgili istatistiklerin istemci ve sunucuya iletilmesini sağlayan bir protokoldür. RTP/RTCP protokolü kullanılan bir akışlandırma sisteminde, sunucu ve istemciler belirli aralıklarla alınan paket miktarı, paket kayıp oranı, paketler arası varış zamanı gibi bilgilerin bulunduğu RTCP paketlerini birbirlerine gönderir. Bu paketlerde yer alan bilgiler yardımı ile akışlandırma sırasında kalitenin değişimi için kararlar alınabilir.

2.2 Uyarlamalı Bit Hızı Video Akışlandırma (ing. Adaptive Bitrate Video Streaming)

Uyarlamalı bithızı çokluortam akışlandırmada amaç, değişken trafik yoğunluğuna sahip bir ağ üzerinde çalışan çokluortam ağlarının ihtiyaçlarını desteklemektir. Oynatım süresince videonun bithızını dinamik olarak mevcut duruma adapte etmek, videoda oluşabilecek kesilmeleri ve depolama süresini minimuma indirebilecektir.

Kullanılabilir bant genişliği dikkate alınarak, bir istemci, daha iyi kalitede video isteğinde bulunabilir. Bu durum kullanıcının daha iyi deneyime sahip olması olarak düşünülebilir. Kullanıcı deneyimi kalitesini arttırmak için kullanılan etkin yöntemlerden biri olarak uyarlamalı bit hızı değişimi gösterilebilir (Georgopoulos et al., 2013). Uyarlamalı video akışlandırma sisteminin kurulabilmesi için sunucu tarafında farklı kalitelere kodlanmış video verisinin bulunmasına ihtiyaç duyulmaktadır. Bu amaçla, video verisi oluşturulduktan sonra farklı kalitelere kodlanarak sunucuda saklanabilir. Bu durumda, istemciye gönderilen videonun kalitesi değiştirilmek istendiği zaman, gönderilen kodlanmış video dosyası değiştirilmelidir. H.264 AVC gibi katmansız bir kodlayıcı kullanıldığı zaman kalite değişim noktası olarak I tipi video karesi kullanılır (Wiegand et al., 2003). I tipi video karelerinin Group of Pictures (GOP) deseni içerisinde kullanım periyoduna bağlı olarak kalite değişimi değişimin talep edildiği anda değil, daha gecikmeli olarak gerçekleştirilebilir. Bu gecikmenin kısıtlanması için I tipi video karelerinin periyodu arttırılabilir ancak bu durum bithızının artmasına neden olur. H.264 AVC ile kodlanmış veride kalite değişiminin yapılabilmesi için tasarlanan SP ve SI tipi anahtar video kareleri (Setton and Girod, 2006), video bithızının artmasına sebep olmadan değişimin kısa sürede yapılabilmesi sağlamaktadır. Literatürde SP/SI tipi video kareleri ile uyarlamalı video akışlandırma sağlayan çeşitli çalışmalar bulunmaktadır (Setton et al., 2005) (Sayıt ve Tunalı, 2012).

Son dönemde kullanımı yaygınlaşmaya başlayan DASH (ing. Dynamic Adaptive Streaming over HTTP-) standardı, HTTP üzerinden uyarlamalı video gönderimini, sunucu tarafında aynı videonun birden fazla kalitede kodlanmasıyla

gerçekleştirmektedir. İstemciler tarafından çalıştırılan hız uyarlama algoritmasına göre, her bir video parçası için, istemcinin uygun gördüğü kalitede talebe bağlı olarak gönderim gerçekleştirilmektedir (Sodagar, 2011).

Aynı video verisinin farklı bit hızlarında kodlanarak farklı kalite varyasyonlarının elde edilmesi yaklaşımının dezavantajı, sunucu tarafında her bir farklı kalitedeki video için ayrı bir kodlama işlemi yapması ve saklama alanı kullanmasıdır. Video kalite uyarlaması aynı zamanda, H.264/SVC gibi katmanlı bir video kodlayıcı bileşeni kullanılarak da sağlanabilir. (Schwarz et al., 2007), SVC tarafından sunulan kalite uyarlama yaklaşımı, yüksek iletim ve depolama verimliliği sağlar, her kalite temsili için videonun bir kez kodlanmış olduğu katmanlı olmayan bir kodlayıcı ile karşılaştırıldığında maliyet ve gereksiz fazlalıkları azalttığı görülmektedir. Video dosyası, bir temel katman ve bir veya daha fazla artış katmanından oluşan SVC kullanılarak kodlanmıştır. İstemci sadece temel katmanı alırsa, en düşük kalitede video oynatılırken, her artış katmanı kalitede artışı sağlamaktadır. Bithızı uyarlama ise, kullanılan hız uyarlama algoritmasına göre hız değişimine karar verildiği anda genişleme katmanlarının eklenmesi ya da çıkartılmasıyla yapılır. (Görkemli, 2007), (Renzi, 2008), (Shao, 2010) çalışmaları, SVC verisi kullanarak uyarlamalı akışlandırma yapan sistemlere örnek olarak verilebilir.

2.3 Yazılım Tanımlı Ağ (YTA, ing. Software Defined Network- SDN)

Yazılım Tanımlı Ağ, son dönemde ortaya atılmış ve geliştirilmekte olan bir teknolojidir. Patenti 2003 yılında Bob Burke ve Zac Carman tarafından "Service Preference Architecture (SPA)" patent uygulamasıyla (Software Defined Networking, 2015) alınan YTA, bilgisayar ağları alanında üzerinde son 4-5 yıldır yoğunluklu olarak çalışılan bir konudur.

YTA, ağlarda, kontrol düzlemi ile veri düzleminin ayrılmasını esas almaktadır. Ağdaki paketin nasıl yönlendirileceğinin kararını veren mekanizma ile paketi yönlendiren cihaz fiziksel olarak birbirinden ayrılmaktadır. Böylece ağın kontrolü merkezi bir cihaza verilerek, ağ ekipmanları basitleştirilip, ucuzlatılabilmekte ve ağ yönetimi kolaylaştırılıp, üreticiden bağımsız bir hale

gelebilmektedir(ONF White Paper, 2012). YTA ile ağın değişimine de her zaman hazır halde bulunulabilecektir. Bugünün statik ağı; iş, son kullanıcı ve değişen piyasa ihtiyaçlarına hızla cevap verebilen bir hizmet platformuna YTA ile dönüştürecektir (ONF White Paper, 2012).

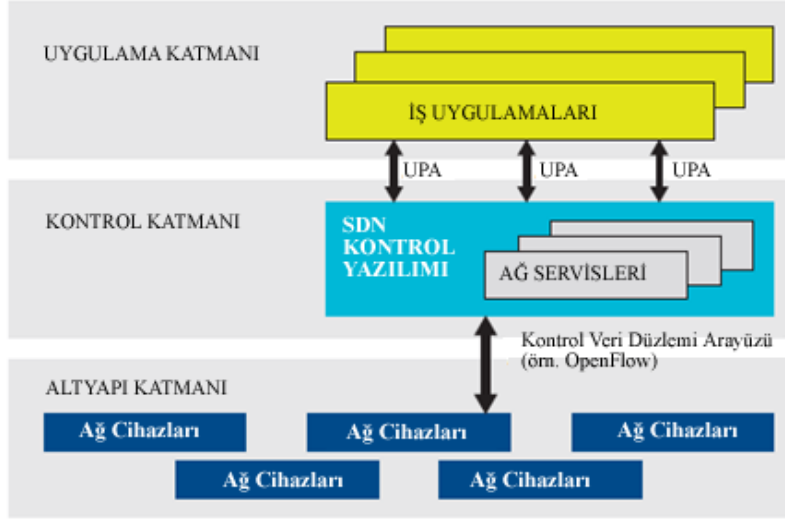
2.3.1 OpenFlow protokolü

OpenFlow, anahtar cihazlar ile kontrol birimi cihazın haberleşmesine izin veren ilk protokoldür(ONF White Paper, 2012). Anahtar cihazlar ağ topolojisinin ve trafik yoğunluğu bilgilerini OpenFlow kullanarak kontrol birimi cihaza gönderebilmekte, kontrol birimi de trafik istatistiklerini sorgulayıp, anahtar cihazlara yönlendirme kurallarını gönderebilmektedir. Bundan dolayı YTA, gerçek zamanlı trafik istatistiklerini ve temeldeki ağ topolojisinin dikkate alarak uygulamaya özel yönlendirme kuralları geliştirmeyi destekler (McKeown et al., 2008).

OpenFlow tabanlı YTA, hem işletmelere hem de bireysel kullanıcılara önemli avantajlar sunmaktadır. Avantajları şu şekilde sıralanabilir (ONF White Paper, 2012):

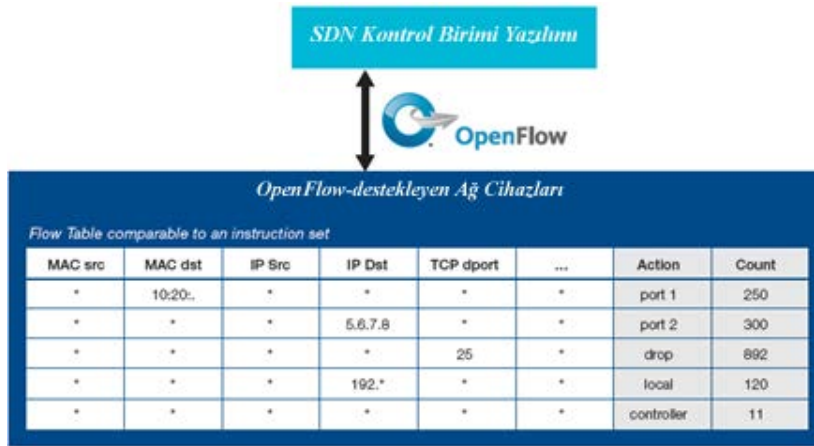
- Ağ cihazlarının kontrolü ve yönetilmesini bir çok üretici firmadan alarak, merkezileştirmiştir.
- Ortak UPA (Uygulama Programlama Arayüzü, ing. Application Programming Interface- API) kullanarak temelde yatan düzenleme, tedarik sistemleri ve uygulamalar gibi ağ detaylarını yönetir ve otomasyon geliştirir.
- Bireysel aygıtları yapılandırmaya ve ya üretici firmanın yeni sürümlerini beklemeye gerek kalmadan, ağın yeni özelliklerini ve hizmetlerini hızlı bir şekilde sunan yeniliktir.
- Ağ davranışlarının kullanıcı ihtiyaçlarına göre sorunsuz bir şekilde uyarlanabilmesi için, uygulamaların merkezi ağ durum bilgisinden

yararlanması gibi son kullanıcı deneyimleri daha iyidir.



Şekil 2.1 Yazılım Tanımlı Ağ Yapısı (ONF White Paper, 2012)

OpenFlow bir CPU talimat seti ile karşılaştırılabilir. Şekil 2.1'de gösterildiği gibi, OpenFlow protokolü, bilgisayar sistemini programlayan bir CPU komut seti gibi, ağ cihazlarının yönlendirme planlarını programlayan bir harici yazılım uygulamasıdır.



Şekil 2.2 OpenFlow Komut Seti Örneği (ONF White Paper, 2012)

OpenFlow protokolleri iki taraflı olarak hem ağ altyapı cihazlarına hem de YTA kontrol yazılımına (kontrol birimi) uygulanmalıdır. OpenFlow, ağ trafiğini temel alan daha önce YTA kontrol yazılımı tarafından durağan veya dinamik

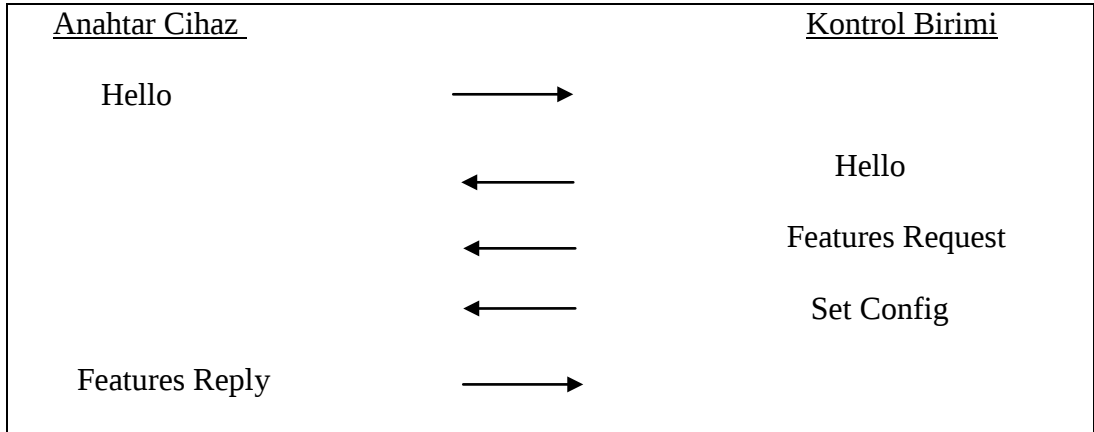
olarak tanımlanmış eşleşme kurallarını kullanır (ONF White Paper, 2012).

OpenFlow protokolünde, ağ kontrolünün taşındığı merkezi ünitelere "*kontrol birimi*" adı verilmektedir. Yönlendiriciler, kontrol birimlerinden gelen komutlara göre yönlendirme işlemi yapmaktadırlar.

Kontrol birimlerinden olan Beacon, Java tabanlı bir kontrol platformudur. Herhangi bir işletim sistemi üzerinde çalışabilen Eclipse Tümlşik Geliştirme Ortamı kullanılarak kolay bir şekilde geliştirilebilir. Floodlight kontrol birimi de Java tabanlı bir kontrol birimidir. Floodlight, OpenFlow standardını destekleyen, artan sayıda anahtar cihazları, yönlendiriciler, sanal anahtar cihazları ve erişim noktaları ile çalışmak üzere tasarlanmıştır(OpenFlow Tutorial, 2015). OpenDayLight; Beacon ve Floodlight gibi Java tabanlı bir kontrol birimidir. Temelde modüler ve esnek bir kontrol platformudur(OpenDayLight, 2015).

OpenFlow mesajlarından bazıları aşağıda verilmektedir;

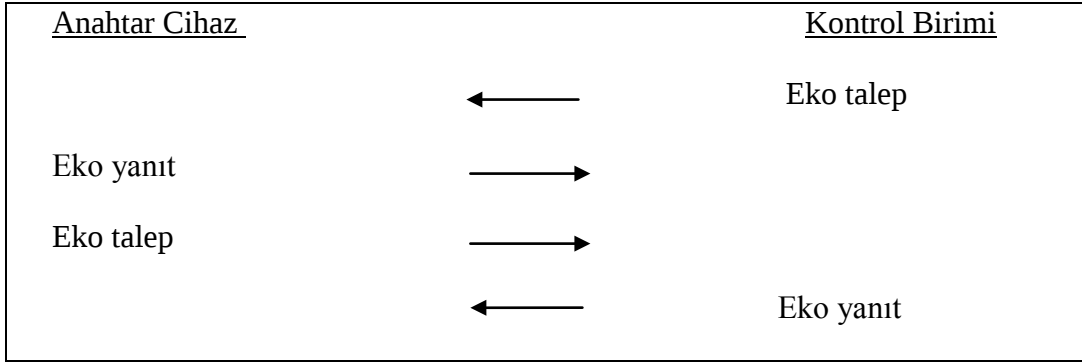
1) Anahtar cihazın kayıtlanması:



Şekil 2.3 Anahtar cihaz kayıtlanması (ONF White Paper, 2012)

Hello mesajları ile Openflow Protocol versiyonuna karar verilmektedir. Features Reply mesajı ile anahtar cihazlara ait olan yol tanımlayıcısı, tek seferde maksimum kaç paketi tamponladığı, kaç adet tabloyu desteklediği, desteklenen openflow_action tipleri, port tanımlama bilgileri gönderilmektedir.

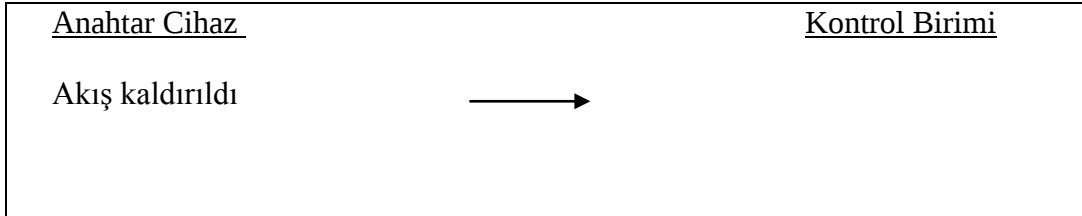
2) Hayatta mısın mesajları :



Şekil 2.4 Kontrol birimi-anahtar cihaz hayatta mısın mesajları (ONF White Paper, 2012)

Bu mesajlar her 15 saniyede bir tekrar etmektedir.

3) Zaman aşımı mesajı :



Şekil 2.5 Zaman aşımı mesajı (ONF White Paper, 2012)

Belli bir süre akış kullanılmazsa zaman aşımından dolayı akış silinmektedir(OpenFlow Switch Specification Version 1.0.0, 2009).

3. LİTERATÜR ÖZETİ

Bu bölümde literatürde sunulan ve YTA üzerinde video akışlandırma uygulamaları üzerine yapılan çalışmaların özetlerine yer verilmiştir.

(Kassler et al., 2012), kullanıcıların hizmet anlaşmalarına göre uygun yolların atanmasını önermiştir. Önerilen sistemde bu tip bir yaklaşımın kablolu ve kablosuz ağ bileşenlerine sahip karma ağlar üzerinde kullanım avantajları tartışılmıştır.

(McDonagh et al., 2013) çalışmasında, YTA ağları üzerinde çalışan ve hizmet kalitesini belirli bir düzeyde tutmayı amaçlayan IPTV uygulaması önerilmiştir. Önerilen hizmette, sunucu ve istemci arasında iki yol bulunmaktadır. Akışın gönderildiği varsayılan yolda herhangi bir sorunun algılanması durumunda, video paketlerini göndermek için alternatif yol seçilmektedir. Bu çalışmada bir alternatif yol olduğundan, yol seçim metotları tartışılmamıştır.

(Kurose and Ross, 2013), DASH standardında, orijinal videodan, farklı kaliteler için farklı bit hızlarında kodlanmış birkaç çeşit video bulunmaktadır. İstemci dinamik olarak bir kaç saniyede bir video segmenti talebinde bulunur. Kullanılabilir bant genişliği miktarı arttığında, istemci yüksek kalitede video verisi isteğinde bulunmakta; kullanılabilir bant genişliği miktarı düştüğünde ise düşük kaliteli video verisi isteğinde bulunmaktadır. DASH standardında, 3G bağlantısı gibi düşük hızla bağlanan kullanıcılar daha düşük bit hızına (ve daha düşük kaliteye) sahip video versiyonunu alırken, fiber bağlantılı kullanıcılar daha yüksek kalitede video versiyonunu almaktadır. Ayrıca DASH ile istemcilerin oturum boyunca kullanılabilir bant genişliğindeki değişimlere de uyum sağlanması gerçekleştirilir.

(Georgopoulos et al., 2013) çalışmasında çokluortam ağlarında kullanıcı deneyim kalitesini arttırmak ve adaletli kullanım için uyarlamalı video akışlandırmanın öneminden bahsedilmektedir. Bu çalışmada DASH standardı kullanılmıştır. Burada üzerinde durulan konu, bant genişliği tahsis algoritması değil, her video uygulamasında en verimli Kullanıcı deneyim kalitesi fonksiyonu

için en uygun bithızı temsillerini araştırmaktır. Videonun gönderildiği cihazların çözünürlüklerinin farklı olması sonucu, her cihaza farklı bithızında gönderim gerçekleştirilmiştir, böylece k'ın en uygun düzeye ulaşması hedeflenmiştir. Sadece farklı cihazlara gönderilen video akışlarının bithızı değiştirildiği için, bu çalışmada da bir önceki çalışma (McDonagh et al., 2013)'de olduğu gibi yol seçme yöntemleri tanımlanmamıştır.

(Georgopoulos et al., 2014), VoD (Video on Demand) akışlandırma uygulamaları için, ön bellekte depolama servislerini temel alan YTA ağları önerilmiştir. Ön bellekte depolamaya ek olarak bu hizmetler yük dengelemesini de sağlamaktadır.

(Seddiki et al., 2014), farklı uygulama akışlarına göre, iletim hızını ayarlayan, YTA tabanlı akış hızı şekillendirici önerilmektedir. Sistem, uyarlamalı video akışlandırma sunucusu ve istemcileri ile test edilmiştir. Bu sistemde dinamik akış yönlendirmesi uygulanmamıştır.

(Cetinkaya vd. 2014), DASH (Dynamic adaptive HTTP streaming- Dinamik uyarlamalı HTTP akışlandırması) istemcileri için video parçalarının bithızına göre dinamik bir şekilde yeniden yönlendirme önermiştir. Video dosyaları, kalite uyarlama sağlamak amacıyla, DASH sunucusunda, farklı kalite temsilleri ile yer almaktadır. Bu çalışmada yol seçim stratejisi TCP davranışı dikkate alınarak tasarlanmış, ve video kalitesi DASH istemci yazılımı tarafından seçilmektedir.

(Egilmez vd., 2012), bir sunucu ve bir istemci ile oluşturulan ağda YTA yönlendirme kabiliyeti kullanılarak, geleneksel kısa yol yönlendirmesi ile karşılaştırmış ve daha iyi hizmet niteliği sağladıklarını deneyimleyerek göstermişlerdir. Bu makalede iletim protokolü olarak UDP ve TCP karşılaştırılmıştır. Ayrıca bu makalede, mevcut yol üzerinde tıkanıklık olması durumunda yeniden yönlendirme işlemi yapılarak bulunacak iki yol kullanılmıştır. Yolda tıkanıklık olup olmadığının kontrolü ise, yolun bant genişliğinin kullanım oranının %70'i aşıp aşmadığına bakılarak karar verilmiştir. Bant genişliği kullanımı %70'in altında ise, tıkanıklık ölçüm değeri 0'a eşitlenmektedir. Eğer

bant genişliği kullanımı %70'i aşıyorsa yolun tıkanık olduğu sonucuna verilip, yeniden yönlendirme işlemleri başlatılmaktadır. Yeniden yönlendirme işlemlerinde kullanılacak olan yollardan biri geleneksel olan kısa yol, diğeri ise hizmet niteliğini sağlayan yoldur. Hizmet niteliğini sağlayan yolu bulurken, LARAC algoritması kullanılmıştır.

(Egilmez vd., 2013), (Egilmez vd., 2012) çalışmasında önerilen yönlendirme metodunu katmanlı video kodlama ile birleştirmiştir. Burada önerilen ilk yaklaşıma göre sadece temel katmanda yeniden yönlendirme işlemi yapılmış, diğerkatmanlarda yeniden yönlendirme yapılmamış, en kısa yoldan gönderilmeye devam edilmiştir. Önerilen bir başka yaklaşıma göre ise, temel katman daha yüksek bir öneme sahip olmakla beraber, hiç paket kaybı olmadan veya en az paket kaybıyla iletilebileceği yoldan gönderilirken, daha az öneme sahip olan geliştirme katmanları kalan diğerkatmanlardan gönderilmektedir.

(Laga et al., 2014), makalesinde katman tabanlı yönlendirme yapıldığından, temel katmana önem verilmiştir. Temel katman kesin gönderilecek şekilde yol seçimi yaptırılmaktadır. En büyük öneme sahip olan temel katman gönderilirken, sadece en kısa yol seçilmemektedir. Çünkü en kısa yolu kullanma her durumda en hızlı iletimi sağlayamayabilir. Bu sebeple temel katman için, paketlerin istemciye zamanında ulaştırılacak yol seçilmektedir. Temel katman için seçilen en etkin yolun dışında kalan diğerkatmanlar da, genişletme (enhancement) katmanları için kullanılmaktadır. Önerilen sistem sonuçları ile varsayılan en kısa yol çözüm sonuçları karşılaştırılmıştır. Bu çalışmada yoğunluklu olarak temel katman üzerinde durulması ve temel katmanı gönderirken seçilen yol ile diğerkatmanları gönderirken seçilen yolun farklı olması bu çalışmayı, tez kapsamında kullanılan yöntemlerden ayırmaktadır. Ayrıca bu çalışmada tıkanıklık gibi yolun sıkışık olduğu durumlarda sadece yol değiştirme işlemi yapılmıştır. Gönderim hızı ayarlamaya yönelik bir çalışma yapılmamıştır.

(Egilmez ve Tekalp, 2014), büyük ölçekli YTA'lerin, birden çok kontrol birimi ile, her kontrol biriminin kendi etki alanı içerisinde uygun hizmet niteliğini gerçekleştirerek ve diğerkatmanları kontrol birimlerine sağladıkları yönlendirme bilgilerini paylaşarak alanlar arası hizmet niteliği yönlendirme işlemini, dağıtık

kontrol düzlemi tarafından yönetilebileceğini öngörmektedir. Bu makalede topoloji birleştirme ve ağ topolojisini etkin bir şekilde kazanmak için bağlantı özetleme yöntemleri kullanılmıştır. Hizmet niteliği sağlayarak ağ üzerindeki durum bilgilerini, birden çok etki alanında bulunan kontrol birimleri arasında yönlendirme, ölçekleme ve güvenli bir şekilde iletmek için iki dağıtık kontrol düzlemi tasarlanmıştır.

Tez kapsamında yapılan çalışmanın literatürden en temel farkı, YTA ağları üzerinde UDP ile gönderim yapan video akışlandırma sisteminde istemcilerin izlediği video kalitesini hem akış yollarının değişimi ile hem de hız uyarlaması ile sağlamasıdır. Bu amaçla kontrol birimi ağın durumuna bağlı olarak hem istemcilerin kullanıcı deneyim kalitesini arttırmak, hem de yeniden yönlendirme maliyetini düşük tutmak için aldığı karar sürecini Markov Karar Süreci olarak modelleyerek zamanla belirli koşullarda alınacak en iyi kararları öğrenmektedir.

4. SİSTEM ALTYAPISI

Bu bölümde tez kapsamında geliştirilen sistemin ayrıntıları verilmektedir. Yapılan çalışmada, kontrol birimi belirli aralıklarla sunucu ile haberleşerek trafik istatistiklerini almakta, durum belirleyip, öğrenme modeline göre duruma uygun eylemi seçerek, eylem sonucunda yapılması gerekenlere göre yol veya video gönderim kalitesini değiştirmekte, gerekli görüldüğünde de herhangi bir eylem gerçekleştirmeden bir sonraki periyodu beklemektedir.

4.1 Video Akışlandırma Uygulaması

Yapılan tez çalışmasında çok katmanlı kodlayıcı ile RTP (Real Time Transport Protocol- Gerçek Zamanlı İletim Protokolü) üzerinden video akışları gönderilmektedir. RTP ile gerçek zamanlı ses, görüntü ya da benzetim verilerinin uçtan uca taşınması sağlanmaktadır (Schulzrinne, 2003).

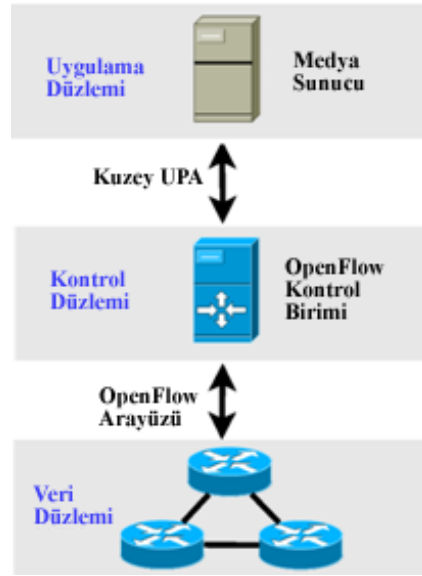
Çok katmanlı kodlayıcıda temel katmana ek olarak bir veya birden fazla genişleme katmanı bulunmaktadır. Çok katmanlı kodlayıcı ile istemci- sunucu arasındaki yolun kullanılabilir bantgenişliğinin izin verdiği ölçüde kalite değişimleri yapılmaktadır. Kalite değişimleri katman ekleme/çıkarma ile yapılmaktadır.

4.2 YTA Mimarisi

Yapılan çalışmada Şekil 4.1'de gösterilen YTA mimarisi kullanılmıştır. Bu mimaride kontrol birimine kuzey UPA'den video sunucu bağlanırken, güney UPA'den OpenFlow destekleyen ağ cihazları bağlanmaktadır.

Sisteme giren istemci, başlangıçta sunucuya video verisi için istek gönderdiğinde, istemcinin bağlı bulunduğu anahtar cihazda bulunan akış tabloları boş olduğundan anahtar cihaz, kontrol birimine yönlendirme işlemi için mesaj göndermektedir. Kontrol birimi, anahtar cihaza, istemciden sunucuya olan yolu en kısa yol olacak şekilde ataması için geri mesaj gönderir. İstemcinin bağlı bulunduğu anahtar cihaz, istemcinin video isteği mesajını bir sonraki anahtar

cihaza göndermekte ve mesajı alan anahtar cihazda da başlangıçta akış tabloları boş olduğundan aynı süreç burada da gerçekleşmektedir. İsteği alan sunucu, video paketi gönderim mesajını bağlı bulunduğu anahtar cihaza göndermektedir. Mesajı alan anahtar cihazda sunucudan istemciye olan akış bilgisi olmadığından tekrar kontrol birimine başvurmaktadır. Kontrol birimi başlangıçta sunucudan istemciye olan yolu da en kısa yol olacak şekilde ayarlamakta, sonraki adımlarda seçilecek eyleme göre yolu değiştirebilmektedir.



Şekil 4.1 Tez çalışmasında kullanılan YTA mimarisi

Anahtar cihazların başlangıçta akış tabloları boş olduğundan kontrol biriminden gelen akış mesajlarına göre tablosunu doldurmaya başlamaktadır. İstemciden sunucuya olan yolda sonradan değişiklik yapılmazken, sunucudan istemciye olan yolda, öğrenme zamanlayıcısının ilk defa dolması ile seçilecek olan eyleme göre, tek yönlü olacak şekilde akış tablolarında güncelleme yapılmaktadır. Her seferinde zamanlayıcının dolmasıyla, kontrol biriminde bulunan öğrenme modeli, yolun ne zaman değişeceğini, değişim sonucunda hangi yolun seçileceğini tanımlamaktadır. Kontrol birimi mevcut yolun değiştirilmesine karar verdiğinde, yeni yol seçimi için istemci ile sunucu arasındaki yolların kullanılabilir bant genişliğini dikkate almaktadır. Kullanılabilir bant genişliğini hesaplayabilmek için periyodik olarak, mevcut trafik bilgisine ihtiyaç duymaktadır. Bu sebeple, kontrol birimi periyodik olarak anahtar cihazlardan sistemdeki trafik istatistik bilgilerini istemektedir. Bu bilgiler, yolda bulunan

anahtar cihazların birbirine bağlandığı portlardan alınan, gönderilen byte sayısı istatistiklerine göre hesaplanmaktadır. Böylece kontrol birimi, gerçek zamanlı trafik bilgilerini alır ve yolun toplam kapasitesinden çıkararak kullanılabilir bant genişliği kapasitesini hesaplar. Burada dikkat edilmesi gereken, yolda bulunan en tıkanık bağlantının kullanılabilir bant genişliğinin, yolun toplam kullanılabilir bant genişliğine eşit olması durumudur.

Sunucudan istemciye olan yoldaki kullanılabilir bant genişliği hesaplamasından sonra video gönderim kalitesinin değişmesi gerekiyorsa, kontrol birimi kaliteyi değiştirmesi için sunucuya sinyal göndermektedir. Mevcut istemci için, kontrol biriminin sunucuya gönderdiği mesaja göre genişleme katmanları eklenir veya çıkarılır.

4.3 YTA Üzerinden Uyarlamalı Akışlandırma İçin Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme yaklaşımında, gerçekleşen bir olgunun ödüllendirilmesi veya cezalandırılmasını bir mantık içerisinde düzenleyerek istenen bir sonucun ortaya çıkmasını sağlamaktır. Durum vektörünün olası değerleri ve kontrol sinyalinin olası değerleri için oluşturulan tabloda her bir kombinasyon için belirlenen ödül değerleri tablolandır ve istenen davranışı ortaya çıkaracak kontrol sinyalinin üretilmesi en yüksek ödül değerine sahip durumlar içerisinde seçilir (İmamoğlu vd., 2009).

4.3.1 Pekiştirmeli öğrenme ve Q-öğrenme

Pekiştirmeli öğrenme ile, keşif fazında, ortam veya sistem ile ilgili bazı bilgiler elde edilebilir ve bu bilgiler ileri ki adımlarda kullanılabilir. Karar mekanizmasının eylemleri (actions) Markov Karar Süreci olarak modellenmiştir. Karar mekanizması, Markov sürecine dayanarak tüm olası durumlar arasında t zamanında bir s_t durumunda bulunmaktadır. Sonrasında karar mekanizması s_t durumunda tanımlı eylemler kümesi olan $A(s_t)$ 'den a_t eylemini seçerek ($a_t \in A(s_t)$), bir sonraki durum olan s_{t+1} durumuna geçer ve r_{t+1} ödülünü alır. Seçilen eylemin ödülü önceden bilinmiyorsa, Q-öğrenme tekniği kullanılır. Q-öğrenme

teknikinde, s durumunda t zamanında seçilen eylem a 'nın ödülü tahmin edilmektedir. Bu tahmin için, karar mekanizması seçimin gösterildiği $Q(s, a)$ değerini günceller ve Q -tablosu olarak bilinen tabloya değerleri kaydeder. Bu değer, s durumunda seçilen a eylemin birikimli ödülüne karşılık gelmektedir. Q değerleri aşağıda verilen (1) fonksiyonuna göre güncellenmektedir. Formüldeki γ ve η değerleri sırası ile azaltma faktörü ve öğrenme faktörü olarak gösterilmiştir.

$$\hat{Q}(s_t, a_t) = \hat{Q}(s_t, a_t) + \eta(r_{t+1} + \gamma \max_{a_{t+1}} \hat{Q}(s_{t+1}, a_{t+1}) - \hat{Q}(s_t, a_t)) \quad (1)$$

$$P(a | s) = \frac{\exp[Q(s, a)/T]}{\sum_{b \in A} \exp[Q(s, b)/T]} \quad (2)$$

t zamanında s_t durumunda seçilecek eylemin olasılığına (2) de verilen softmax fonksiyonu ile karar verilir(Alpaydın, 2004). T değeri sıcaklık değişkenini göstermektedir ve başlangıçta yüksek bir değer atanmaktadır. Sıcaklık değişkeni, benzetimli tavlama metodunda kullanılan bir parametredir. Benzetimli tavlama yöntemi, minimum maliyet bulma problemlerinde olası yerel çözümlerden birinin seçilmeyerek global çözüme doğru yönlendirilme yapılmasını sağlayan olasılıksal bir metottur(Bertsimas and Tsitsiklis, 1993). Bu metodta sıcaklık yüksek bir değerden başlayarak zaman içerisinde azaltılır. Rastgelelik, yüksek sıcaklıklarda daha fazlayken, düşük sıcaklıklarda azalarak açgözlü bir seçime yönlendirmektedir.

Q -değerleri, her eylemin seçilme olasılığının aynı olduğunu gösteren 0 ile başlatılmaktadır. Q -değerleri (1) de verilen fonksiyona göre güncellenir(Bertsekas, 2007). T parametresi her güncelleme sırasında aşamalı olarak azaltılmaktadır. Bu azalma sistemin yavaş yavaş keşif fazından kullanma fazına geçmesini sağlamaktadır. Keşif fazında tüm eylemlerin seçilme olasılığı neredeyse aynı iken, kullanım fazında karar mekanizması Q -değeri daha yüksek olan eylemi seçme eğilimini göstermektedir. Öğrenme oranı da sıcaklık değeri gibi zamanla azalmaktadır.

4.3.2 YTA kontrol birimi için pekiştirmeli öğrenme modeli tasarımı

Önerilen öğrenme modelinde, asıl amaç, kontrol biriminin ne zaman yeni bir yol seçeceği ve sunucunun video kalitesini ne zaman ayarlayacağını tanımlamaktır. Burada kontrol birimi karar mekanizmasıdır ve mevcut duruma karar verip, alınan ödüle göre Q-değerlerini güncellemektedir. Bu yaklaşım için, kontrol birimi periyodik olarak istemciye gönderilen videonun bit hızını ölçümler. Kontrol birimi aynı zamanda, sunucu ile istemci arasında bulunan diğer yollarında kullanılabilir bant genişliğini ölçümler ve aday yollardan maksimum kullanılabilir bant genişliğine sahip olan yolu seçer. Mevcut duruma karar verilirken, bu üç ölçüm dikkate alınmaktadır. Katmanlı video verisinde kalite değişimleri video katmanları eklenerek veya çıkarılarak sağlanmıştır. Sadece temel katmanı alan bir istemci, videoyu en düşük kalitede oynatmaktadır. Her eklenen genişleme katmanı video kalitesini artırır ve i . genişleme katmanı sadece temel katman ve i 'nin altında bulunan diğer katmanlar alınmış ise çözülebilir. Dikkat edilmesi gereken nokta; br_r , r genişleme katmanını içeren bit hızı olmak üzere, her zaman $br_{i+1} > br_i > br_0$ olmasıdır. Durumlar, aday yolların kullanılabilir bant genişliği ve mevcut yol arasındaki ilişkiye göre; sunucudan gönderilen genişleme katman sayısı ve mevcut yolun kullanılabilir bant genişliği arasındaki ilişkiye göre belirlenmektedir. s_{ij} , istemcinin mevcut durumunu göstermektedir. i ve j indeksleri, sırasıyla aşağıda verilen (3) ve (4) fonksiyonlarının çıkışlarına göre atanmıştır. fonksiyonlardaki r , r_{aday} aday yol üzerinden gönderilebilecek maksimum genişleme katman sayısını gösterirken, $r_{güncel}$ mevcut yoldan gönderilebilecek genişleme katman sayısını göstermektedir, r ise sunucudan gönderilen mevcut genişleme katman sayısını temsil etmektedir. Örneğin, sunucu istemciye sadece temel katmanı gönderiyorsa, r değeri 0'a eşit olur. Eğer mevcut yolun kullanılabilir bant genişliği temel katmanı göndermek için yeterli kapasiteye sahip değil ise $r_{güncel}$ değeri -1'e eşitlenir. Benzer şekilde aday yolun kullanılabilir bant genişliği temel katmanı göndermek için yeterli kapasiteye sahip değil ise r_{aday} değeri -1'e eşitlenir. Kullanılan modelde tek seferde ikiden fazla genişleme katmanı ekleme veya çıkarmaya izin verilmemiştir. Bu yaklaşım kaliteyi büyük ölçüde değiştirmeye neden olduğundan, bu tür değişiklikler kullanıcıların bakış açısından rahatsız edici olmaktadır. Eğer ikiden fazla genişleme katmanının eklenmesi veya çıkarılması gerekiyorsa, bu işlem iki

periyod içinde yapılabilir. Bu nedenle (3) ve (4) fonksiyonunda $r+3$ veya $r-3$ durumları tanımlanmamıştır.

$$i = \begin{cases} 0, & \text{eğer } r_{aday} > r_{güncel} \text{ ve } r_{aday} \geq r+2 \\ 1, & \text{aksi halde, eğer } r_{aday} > r_{güncel} \text{ ve } r_{aday} = r+1 \\ 2, & \text{aksi halde, eğer } r_{aday} > r_{güncel} \text{ ve } r_{aday} \leq r+1 \\ 2, & \text{aksi halde, eğer } r_{aday} = r_{güncel} = -1 \text{ ve } kbg_{aday} > kbg_{güncel} \\ 3, & \text{aksi halde, eğer } r_{aday} \leq r_{güncel} \end{cases} \quad (3)$$

$$j = \begin{cases} 0, & \text{eğer } r_{güncel} \geq r+2 \\ 1, & \text{aksi halde, eğer } r_{güncel} = r+1 \\ 2, & \text{aksi halde, eğer } r_{güncel} = r \\ 3, & \text{aksi halde, eğer } r_{güncel} \leq r-2 \text{ ve } r \geq 2 \\ 4, & \text{aksi halde, eğer } r_{güncel} \leq r-1 \text{ ve } r \geq 1 \\ 5, & \text{aksi halde, eğer } r_{güncel} \leq r-1 \text{ ve } r < 1 \end{cases} \quad (4)$$

(3) fonksiyonunda görüldüğü gibi, i indeksi tanımlanırken iki farklı koşul 2 değerini sağlamaktadır. Burada kg_{aday} ve $kg_{güncel}$ sırasıyla, aday yolun ve mevcut yolun kullanılabilir bant genişliğini göstermektedir. Mevcut yolun ve aday yolun her ikisinin de temel katmanı gönderebilecek bant genişliğine sahip olmaması durumu burada özel durum olmaktadır, fakat aday yolun kullanılabilir bant genişliği mevcut yoldan daha büyüktür. Kapasite yeterli olmadığında, en azından kayıp paket sayısını düşürmek için bu özel durum tanımlanmıştır.

4 farklı i değeri ve 6 farklı j değeri olmak üzere toplam durum sayısı s_{ij} 24'e eşittir. Bununla birlikte, i ve j tanımlamalarındaki matematiksel denklemlerin çakışmasından dolayı bazı durumlar geçerli değildir. Örneğin; s_{10} tanımlamasında $r_{aday} > r_{güncel}$, $r_{aday} = r+1$ ($i=1$ için) ve $r_{güncel} \geq r+2$ ($j=0$ için) aynı anda geçerli değildir. Benzer durumlar s_{11} , s_{20} ve s_{21} için de geçerlidir ve bu durumlar uygulanabilir değildir. Böylece modelin toplamdaki durum sayısı 20'ye düşmektedir. Durumlar ağdaki her istemci için ayrı ayrı tanımlanırken, Q -değerleri aynı tabloda güncellenebilir.

Çalışmada kullanılan öğrenme modelinde $A = \{a_0, a_1, \dots, a_7\}$ olmak üzere, A , eylemler kümesini göstermektedir. Eğer ilk eylem seçilirse, kontrol birimi, mevcut yol, mevcut hızda video akışını göndermeye izin vermiyor olsa bile, bir

sonraki periyoda kadar bekler. Bazı ağ altyapılarında, yüksek trafik hacim değişiklikleri oluştuğunda ortamın dengeye kavuşabilmesi için beklemek en iyi seçenek olabilir. Bu gibi durumlar için bu eylem tanımlanmıştır. Kontrol birimi a_1 veya a_2 eylemlerinden birini seçerse, sırasıyla, akış yolunu aday yollardan biri olacak şekilde ayarlar ve bir veya iki genişleme katmanı ekler. Kontrol birimi a_3 eylemini seçerse, sadece mevcut yolu değiştirir, fakat kaliteyi değiştirmez. Öte yandan a_4 , a_5 , a_6 , a_7 eylemlerinden biri seçilerek mevcut yol değiştirilmeden sadece genişleme katmanı eklenip çıkarılabilir. Bu eylemlerin seçiminde, yolun kullanılabilir bant genişliği dikkate alınmaktadır. Tablo 4.1'de eylem tanımlamalarının listesi yer almaktadır. Modelin izin vermediği 2'den fazla katman ekleme veya çıkarma işlemi için herhangi bir eylem tanımlanmamıştır.

Tablo 4.1 Eylem Tanımlamaları

Eylem	Eylem Tanımlamaları
a_0	Bekle
a_1	Mevcut yolu değiştir ve bir katman ekle
a_2	Mevcut yolu değiştir ve iki katman ekle
a_3	Mevcut yolu değiştir
a_4	Bir katman ekle
a_5	Bir katman çıkar
a_6	İki katman ekle
a_7	İki katman çıkar

Bazı durumlarda bazı eylemlerin seçilmesi anlamsız olmaktadır. Örneğin mevcut yol en iyi kalitede video gönderimi için yeterli kapasiteye sahipse ve sunucu tüm video katmanlarını gönderiyorsa, kontrol birimi, yolu veya kaliteyi değiştirmemelidir. Bu özel durumda seçilebilecek en uygun eylem a_0 eylemidir. Bu şekilde oluşabilecek bir kaç durum için hangi eylemlerin seçilmesi gerektiği bir kurallar kümesi ile tanımlanmıştır. Her s_{ij} durumu için uygun eylemler, i ve j değerlerine göre sırasıyla Tablo 4.2a ve 4.2b'de gösterilmiştir. Örneğin, $i=0$ ve $j=0$ için s_{00} durumunda uygun eylemler a_0 , a_4 , a_6 'dır.

Tablo 4.2 s_{ij} durumunda i ve j değerlerine göre uygun eylem listesi

i değeri	Uygun Eylemler	j değeri	Uygun Eylemler
$i = 0$ (eğer $j > 1$)	a_0, a_1, a_2	$j = 0$	a_0, a_4, a_6
$i = 0$ (eğer $j \leq 1$)	a_0	$j = 1$	a_0, a_4
$i = 1$	a_0, a_1	$j = 2$	a_0
$i = 2$	a_0, a_3	$j = 3$	a_0, a_5, a_7
$i = 3$	a_0	$j = 4$	a_0, a_5
		$j = 5$	a_0

a) i değerine göre uygun eylemler b) j değerine göre uygun eylemler

Kontrol birimi, bir önce ayarlanan zamanlayıcıya göre istemcilerin durumunu tanımlar. Durumlar sistemdeki her istemci için farklı şekilde tanımlanır. Sonrasında kontrol birimi softmax fonksiyonu(Alpaydin, 2004) tarafından belirlenen olasılığa göre mevcut durumda uygun olan bir eylemi seçer. Bir sonraki adımda, zamanlayıcının süresi dolduğunda, kontrol birimi ödülü hesaplar, Q-tablosunu günceller, yeni bir eylem seçer ve seçilen eylemi gerçekleştirir.

Kullanıcı deneyim kalitesi göz önüne alındığında, paket kayıp oranı ve kalite değişim sayısı önemlidir. Yetersiz bağlantı kapasitesi paket kayıplarına neden olduğundan, kontrol birimi seçilen yolun kapasitesi ile videonun bit hızı arasındaki farkı hesaplar. Bu fark, olası kayıp paket oranı hakkında bilgi verdiği için, kontrol biriminin kayıp paket sayısını elde etmek için istemci ile iletişim kurmasına gerek yoktur. Ağ tarafından bakıldığında, yolun her değişiminde, kontrol biriminin anahtar cihazlara akış mesajlarını göndermesi sisteme fazladan karmaşıklık ve yük getirmektedir. Bu sebeple, öğrenme modelinde amaç, istemcilere mümkün olduğunca, kalite ve yol değiştirme sıklığını azaltarak, en yüksek kalitede video göndermektir. (5) de ödül fonksiyonu gösterilmektedir. Formüle göre, $bit\ hızı$ sunucu tarafından gönderilen mevcut katmanın bit hızını; $hız_{max}$ mevcut yol üzerinden gönderilebilecek maksimum katmanın bit hızını; α kontrol birimi maliyetini ve β kalite değişim yükünü göstermektedir. Hesaplamalardan sonra ödül negatif değer çıkarsa, hatalı karar vermenin seçilen

eyleme büyük etkisi olacağından, sıfıra eşitlenmektedir. Video bit hızının mevcut yolun kullanılabilir bant genişliğine yakın olduğu durumlarda, yüksek ödül değeri veren *bit hızı* ve $|hız_{max} - bit\ hızı|$ terimleri arasındaki fark artmaktadır. Diğer bir deyişle, eğer bit hızı, alt yapı kapasitesi ile uyumlu ise ödül yüksek olmaktadır.

$$\text{ödül} = bit\ hızı - (|hız_{max} - bit\ hızı|) - \alpha - (\text{eklenen veya çıkarılan katman sayısı} * \beta) \quad (5)$$

Kontrol biriminin öğrenme algoritmasının adımları aşağıdaki sözde kodda verilmiştir.

Prosedür Öğrenme

Başlangıçta,

$T = \text{Yüksek bir değer}$

Sıcaklık

$d_T = 0.99$

Sıcaklık düşürücü

$\eta = 1$

Öğrenme Hızı

$d_\eta = 0.99$

Öğrenme Yavaşlatıcı

$\gamma = 0..1$ arasında 1'e yakın bir değer (0.95)

Azaltma faktörü

Her adımda,

Durum belirle.

$s(t) = \text{Şu an hangi durumdayım.}$

Karar verme

Softmax ile her a eylemi için orantılı bir seçim

Eylemlerin seçme şansları ile orantılı rastgele bir $a(t)$ belirle.

Uygulama

$a(t)$ eylemini gerçekleştir

$s(t+1)$ durumunu belirle

Ödül

Eylemin etkilerini ölç

$R(s(t), a(t)) = \text{ödül}(s(t+1))$

Öğrenme

$Q(s(t), a(t))$ hesapla

Sonlandırma,

$T = T \times d_T$

$\eta = \eta \times d_\eta$

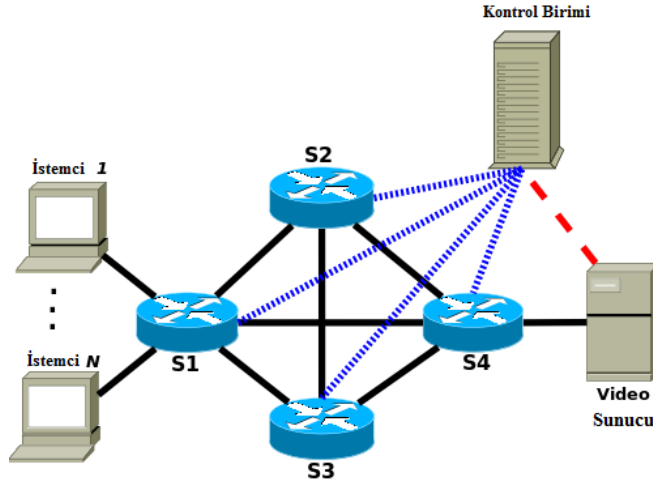
Tezin bir sonraki bölümünde, bu bölümde ayrıntıları verilen öğrenme modelinin ve bu modele bağlı olarak YTA üzerinde çalışan uyarlamalı video akışlandırma sisteminin Mininet ortamında geliştirilmesi anlatılmıştır.

5. YTA AĞLAR ÜZERİNDE UYARLAMALI VE ÖĞRENME TABANLI ÇOKLUORTAM AKIŞLANDIRMA UYGULAMASI

5.1 Ağ Topolojisinin Oluşturulması

Benzetimde Şekil 5.1 de gösterilen topoloji kullanılmıştır. Sistemde 6 istemci, 1 sunucu ve bir kontrol birimi bulunmaktadır. Noktalı çizgiler OpenFlow protokolünü ve kesikli çizgi kontrol birimi ile sunucu iletişimini göstermektedir. Topolojinin oluşturulması ve benzetimin yürütülmesi için Mininet (Lantz et al., 2010) ortamı kullanılmıştır. Mininet, deneyimler ve YTA ağ ile OpenFlow protokolünün kullanıldığı uygulamaların test edilmesi için tasarlanmış bir ağ emülatörüdür. Bu çalışmada kontrol birimi yazılımı olarak, piyasadaki bir çok OpenFlow anahtar cihazları ile uyumlu olan Floodlight (Floodlight, 2015) kullanılmıştır.

Öğrenme Temelli yaklaşıma ek olarak, uyarlamalı akış yolu ve kalite seçimi yapan açgözlü yaklaşım da uygulanmıştır. Bu yaklaşımda da öğrenme tabanlı yaklaşıma benzer şekilde, istemci sisteme girdiğinde, kontrol birimi bu istemci için bir zamanlayıcı kurmaktadır. Açgözlü yaklaşımda, her zamanlayıcı, bir istemci için çalıştırılmaktadır ve kontrol birimi bu istemciye video paketlerini göndermek için maksimum kullanılabilir bant genişliğine sahip yolu atamaktadır. Bu nedenle en iyi başarımı elde etmek için her devrimde dinamik olarak akış yolları değiştirilir. Yol seçiminden sonra, kontrol birimi sunucuya istemci ile arasında atanan yolunun kullanılabilir bant genişliğine göre, maksimum sayıda katman gönderilmesi için sinyal üretir.



Şekil 5.1 Benzetimde kullanılan ağ topolojisi

Başarımın ölçülmesi için, kullanıcı deneyim kalitesi ve ağa bağlı olarak 5 farklı parametre gözlemlenmiştir. Bunlardan ilki, kayıp video karelerinin yüzdesidir. Bir video karesi birden fazla pakete bölünmüştür ve bu paketlerden herhangi biri kaybolursa, video karesi istemci tarafında oluşturulamaz. Bu yüzden kayıp paketlerin yerine kayıp karelerin sayısını ölçmek alınan kullanıcı deneyim kalitesini daha doğru gösterir. Kayıp karelerin yüzdesine ek olarak, ortalama kalite değişim sayısı, ortalama yol değişim sayısı ve video sunucusunun ortalama iletim hızı da ölçülmektedir. İstemci sisteme katıldığında, videoyu oynatmadan önce gecikme etkisini azaltmak için tamponunu doldurmaktadır. Tüm deneylerde, tüm yaklaşımlar için bu ilk tamponlama süresi 5 saniyeye eşitlenmiştir. Her istemci için zamanlayıcı, kontrol birimi tarafından 10 saniyeye ayarlanmıştır. En kısa-yol yönlendirmesi kullanan uyarlamalı olmayan sistemlerde video dosyası bir temel ve bir genişleme katmanından oluşmaktadır; diğer bütün yaklaşımlarda video dosyası bir temel ve iki genişleme katmanından oluşmaktadır. Her bir katmanın bit hızı Tablo 5.1'de verilmektedir. Video paketleri, UDP üzerinden RTP kullanılarak gönderilmektedir. Dolayısıyla, her benzetimde paket kayıpları gözlemlenmiştir.

Tablo 5.1 Video Katmalarının Bit Hızı

<i>Katman</i>	<i>Bit Hızı</i>
<i>Temel Katman</i>	670 Kbps
<i>Genişleme Katmanı 1</i>	800 Kbps
<i>Genişleme Katmanı 2</i>	980 Kbps

5.2 Öğrenme Parametrelerinin Ayarlanması

Öğrenme parametrelerinin ayarlanması için gerçekleştirilen benzetim setinde, eş zamanlı olarak altı kullanıcı sisteme katılmış ve sunucudan tüm kareler gönderilene kadar sistemden ayrılmamışlardır. İç bağlantıların (S2 ve S3 arasındaki direkt bağlantı, ve S1 ve S4 arasındaki direkt bağlantı) kapasitesi 2 Mbps ve diğer bağlantıların kapasitesi 1 Mbps olarak ayarlanmıştır.

Deneylerin başlangıcında öğrenme hızı 1 olarak ayarlanmış ve stokastik ortamlar (Sutton and Barto, 1998) için önerilen en uygun yaklaşım olarak zamanla azaltılmıştır. Öğrenme hızı değeri ve sıcaklık değişkeni her devinimde 1% oranında azalmıştır. (5) de verilen ödül fonksiyonunda tanımlanan kontrol birimi maliyeti (α) ve kalite değişim maliyeti (β) 100'e ayarlanmıştır. En iyi başarıyı sağlayan diğer öğrenme parametrelerinin belirlenmesi amacıyla, bir dizi deney uygulanmıştır. Bu deneylerde, farklı sıcaklık değişkenleri ve azaltma faktörü değerleri için deney başarımları gözlemlenmiştir. Öğrenme parametrelerinin en uygun değerlerini belirlemek için, beş farklı değer, her biri için otuz benzetim yapılarak değerlendirilmiştir. Benzetim sonuçları tüm kullanıcılardan elde edilen sonuçların ortalaması alınarak hesaplanmıştır. Tablo 5.2'de farklı sıcaklık değerlerine göre, ortalama kayıp kare yüzdesi, ortalama yol değişim sayısı ve ortalama kalite değişim sayısı gösterilmektedir. Deneylerin ilk kümesinde azaltma faktörü (γ) 1 olarak ayarlanmıştır. Tabloda görüldüğü gibi en iyi başarımlar T=1500 ve T=2000 değerlerinde elde edilmiştir. Öncelikli odak noktası kullanıcı deneyim kalitesi olduğundan ve diğer gözlenen parametreler bu iki değer için çok yakın bulunduğundan, kayıp sayısı yüzdesi minimum olan 2000 değeri sıcaklık değeri olarak seçilmiştir.

Tablo 5.2 Farklı sıcaklık değerleri için benzetim sonuçları

<i>T (Sıcaklık)</i>	<i>Kayıp Kare Yüzdesi</i>	<i>Kalite Değişim Ortalaması</i>	<i>Yol Değişim Ortalaması</i>
1000	24.68	13.30	6.70
1500	18.61	11.52	4.72
2000	17.76	12.00	5.50
2500	18.91	13.00	6.00
3000	20.00	13.80	6.00

Azaltma faktörünün belirlenmesi için, sıcaklık değeri 2000 olarak ayarlandıktan sonra deneyler tekrarlanmış, ve sonuçta azaltma faktörünün beş farklı değeri için ortalaması alınmıştır. Sonuçlar Tablo 5.3'te verilmektedir. Tüm parametreler için en iyi başarımla, azaltma faktörünün 0.95'e eşit olduğunda elde edilmektedir, bu sebeple azaltma faktörü 0.95 olarak seçilmiştir.

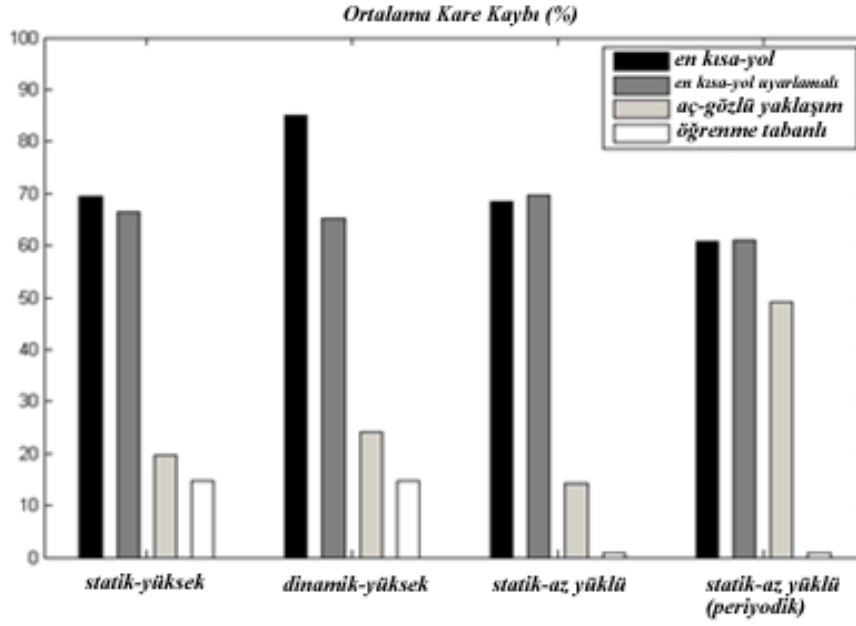
Tablo 5.3 Farklı azaltma faktörü değerleri için benzetim sonuçları

<i>(Azaltma Faktörü)</i>	<i>Kayıp Kare Yüzdesi</i>	<i>Kalite Değişim Ortalaması</i>	<i>Yol Değişim Ortalaması</i>
0.80	17.57	11.32	4.62
0.85	17.39	12.34	6.02
0.90	15.85	12.24	5.28
0.95	14.64	11.32	4.04
1.00	17.76	12.00	5.50

Öğrenme parametrelerine karar verildikten sonra, önerilen yaklaşımla, aç-gözlü yaklaşım, uyarlamalı en kısa-yol ve uyarlamalı olmayan en kısa-yol yönlendirmesi yaklaşımları başarımlarını göstermek için test edilmiştir. Bir sonraki alt bölümde sonuçlar karşılaştırmalı olarak verilmektedir.

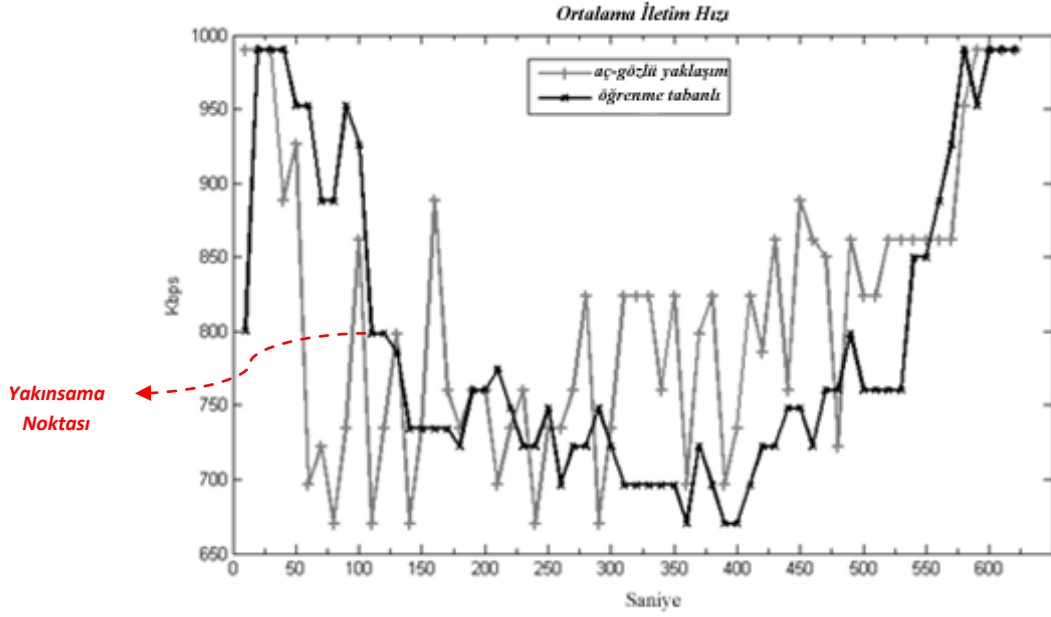
5.3 Karşılaştırmalı Test Sonuçları

Önerilen sistemin başarımını gözlemleyebilmek için, Şekil 5.1'de verilen topoloji üzerinde farklı ağ senaryoları durumlarında benzetimler yapılmıştır. Tüm senaryolar için, benzetimler 600 saniye sürmektedir. Statik senaryolarda, tüm istemciler eş zamanda sisteme katılmakta ve benzetim bitene kadar sisteme bağlı kalmaktadırlar. Dinamik senaryoda, altı istemciden üçü on/off modele göre (Taqqu et al., 1997) sisteme dahil olmakta ve sistemden çıkmaktadır: 500 kare alındıktan sonra sistemden ayrılmakta, sistem dışı olduğu 20 saniyelik devinimden sonra tekrar sisteme katılmakta, 500 kare aldıktan sonra tekrar sistemden ayrılmakta ve bu şekilde devam etmektedir. İlk senaryoda, önceki alt bölümde tanımlandığı şekilde, altı istemci aynı anda sisteme dahil olmakta ve benzetim bitene kadar ayrılmamaktadırlar. Tüm iç bağlantıların kapasiteleri 2 Mbps ve diğer bağlantıların kapasiteleri 1 Mbps olarak ayarlanmıştır. Ağ kapasitesi video bit hızına göre sınırlı olduğundan, bu senaryo "*statik yüksek yüklü ağ senaryosu*" olarak adlandırılmıştır. Aynı bağlantı kapasitesine sahip ağ üzerinden, üç istemcinin benzetim sonuna kadar bağlı kaldığı, diğer üç istemcinin ise aralıklı olarak sisteme bağlı kaldıkları senaryo "*dinamik yüksek yüklü ağ senaryosu*" olarak adlandırılmıştır. "*Statik az yüklü ağ senaryosu*"nda ise, tüm bağlantıların kapasitesi, ağ alt yapısının en yüksek kalitede video gönderilmesine izin verdiği 2 Mbps olarak ayarlanmıştır. Son olarak "*statik az yüklü ağ senaryosu- periyodik giriş ile*", istemciler 30 sn aralıkla ve sırayla sisteme katılmakta, sunucu tüm kareleri gönderene kadar da sistemden ayrılmamaktadırlar.



Şekil 5.2 Ortalama kare kayıp yüzdeleri

Şekil 5.2'de tüm yaklaşımlar ve tüm ağ senaryoları için ortalama kare kaybı oranları verilmiştir. Beklendiği gibi, hem uyarlamalı hem de uyarlamalı olmayan en kısa-yol yönlendirmesinde kullanıcılar aynı yol üzerinden video paketlerini aldığından, bu durum kayıp video karesi sayısının artmasına yol açmıştır. Tüm senaryolar için önerilen öğrenme tabanlı yaklaşıma göre, aç-gözlü yaklaşımda daha fazla kare kaybı olmuştur. Aç-gözlü yaklaşımda kontrol birimi, maksimum kullanılabilir bant genişliğine sahip olan yolu atar ve bu bant genişliğine göre mümkün olan maksimum kaliteyi ayarlaması için sunucuya sinyal gönderir. Ancak sunucu, istemciye video paketlerini göndermeye başlamadan önce bile, bant genişliği değişebilir. Bant genişliğindeki bu değişiklikler kullanıcıların çevrimiçi ve çevrimdışı olmasından kaynaklanabilir. Ayrıca, her kullanıcı için akış yollarının her devrimde yeniden belirlenmesi, sisteme ek dinamizm getirmektedir. Diğer taraftan, öğrenme tabanlı yaklaşımda, kontrol birimi, ortamı gözlemlemek için beklemeyi bir eylem olarak seçebilmekte ve bir müddet beklemektedir. Önerilen öğrenme modelinde, kontrol biriminin tüm senaryolar için, kararlar hakkında en uygun noktada birleştiği görülmektedir.

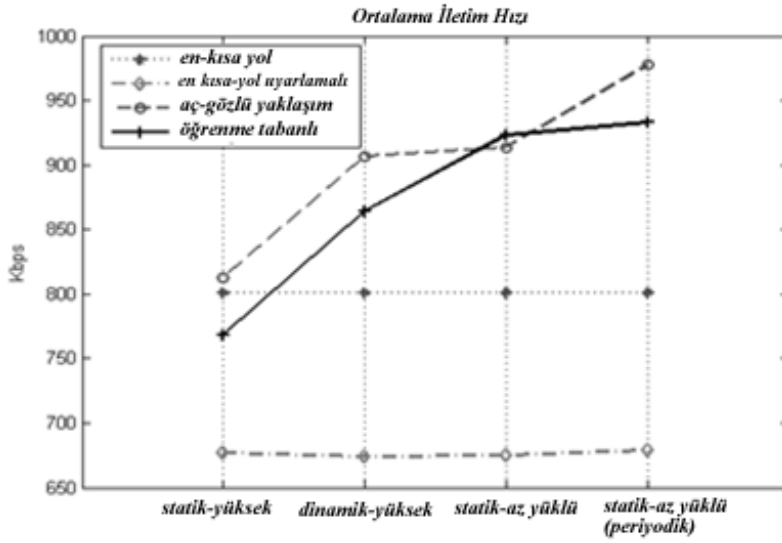


Şekil 5.3 Zamanla gözlemlenen kalite değişiklikleri

Modelin yakınsama zamanını görmek için, bir kullanıcıda yaşanan kalite değişiklikleri gözlemlenmiş ve statik yüksek yüklü ağ senaryosunda ortalaması alınmıştır. Bu gözlem Şekil 5.3'te verilmiştir. Grafikte gösterilen kalite değişim salınımlarının 100. saniyeden sonra daha küçük hale gelmesi nedeniyle, bu kullanıcı için yakınsama zamanı 100. saniye civarındadır. Grafikte, aynı senaryo için aç-gözlü yaklaşımda gözlemlenen kalite değişimleri de görülmektedir.

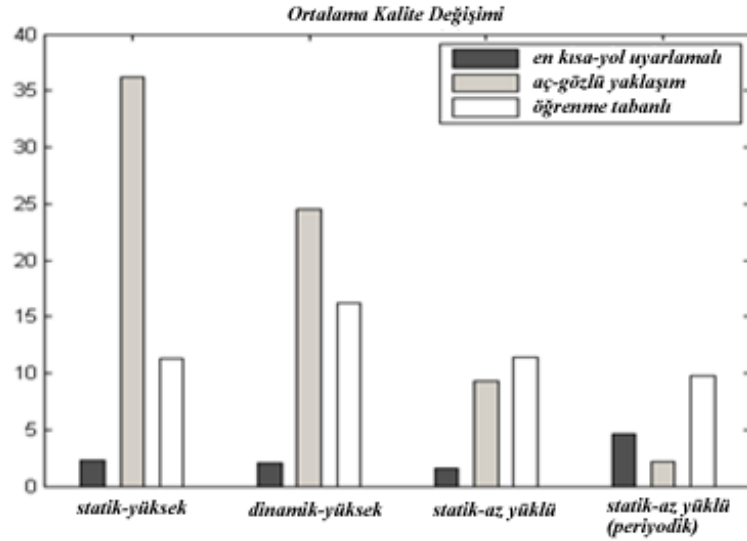
Sunucu tarafından gönderilen hızların, ortalama iletim hızı Şekil 5.4'te verilmiştir. Uyarlamalı olmayan en kısa-yol yönlendirmesinde beklendiği gibi, iletim hızı her zaman orta hızda olmaktadır. Diğer taraftan, en kısa yol üzerinde yeterli kapasite olmadığından genişleme katmanlarının gönderilemediği uyarlamalı en kısa-yol yönlendirmesinde bit hızı, kararlı bir şekilde en düşük seviyededir. En yüksek ortalama iletim hızı aç-gözlü yaklaşımda gözlemlenmektedir. Sunucu, mümkün olduğunca çok sayıda video katmanını göndermeye çalışsa bile, bu durum gereksiz şekilde ağ kaynaklarını tüketmenin yanında kare kaybı oranını da arttırmaktadır. Öğrenme tabanlı yaklaşımın ortalama iletim hızının, aç-gözlü yaklaşımın ortalama iletim hızını geçtiği tek senaryo statik az yüklü ağ senaryosudur. Ayrıca bu senaryo için öğrenme tabanlı yaklaşımda, diğer yaklaşımlar arasındaki en düşük ortalama kare kaybı oranı

oluştugu görülmektedir. Bu durum, öğrenme tabanlı yaklaşım ile eğer makul bir eylem ise kontrol mekanizması kaliteyi arttırır ifadesini doğrulamaktadır.



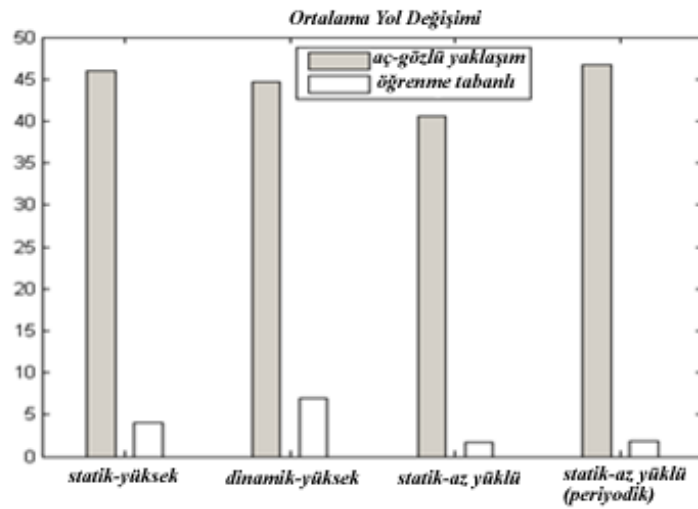
Şekil 5.4 Ortalama iletim hızı

Uyarlamalı olmayan en kısa-yol yaklaşımında oturum boyunca video kalitesi ve gönderilen yol değişmediğinden, uyarlamalı en kısa-yol, aç-gözlü yaklaşım ve öğrenme tabanlı yaklaşımların ortalama kalite değişimleri ve yol değişim istatistikleri sırasıyla Şekil 5.5 ve Şekil 5.6'da verilmektedir. Aç-gözlü yaklaşımda, kalite değişim sayısında kısıtlayıcı olmadığından, yüksek yüklü senaryoda en fazla sayıda kalite değişimi gözlemlenmektedir. Şekilde görüldüğü gibi, kapasite değişiklikleri, kalitede keskin salınımlara neden olmakta; dolayısı ile kalite değişim sayısını artmaktadır. Diğer taraftan, öğrenme tabanlı yaklaşım ile benzetimlerde ortalama kalite değişim sayısı her zaman 20'nin altındadır. Aralıklı olarak bağlanan kullanıcılar ağ kalitesini yüklü ve yüksek yüklü durumlar arasında değiştirdiği için, kalite değişikliklerinin çoğu dinamik senaryoda gözlenir. Kapasitenin kilitlenmesinden dolayı, genellikle, sunucunun videoyu en düşük kalitede gönderdiği uyarlamalı en kısa-yol yönlendirmesinde en düşük sayıda kalite değişim sayısı gözlemlenmektedir.



Şekil 5.5 Ortalama kalite değişimi

Şekil 5.6'da görüldüğü gibi, tüm senaryolarda, öğrenme tabanlı yaklaşıma göre aç-gözlü yaklaşımda çok daha sık yol değişikliği olmaktadır. Öğrenme tabanlı yaklaşımda, oturumda akışları idare etmek için bağlantıların bant genişliğinin yeterli olmasından, "statik az yüklü ağ senaryosunda" en az sayıda yol değişikliği oluşmaktadır. Diğer taraftan, yol değiştirildiğinde, kontrol mekanizması yolda bulunan ilgili anahtar cihazlara bununla ilgili mesajlar gönderdiği için, kontrol birimi ve anahtar cihaz arasındaki mesaj karmaşıklığı ve dolayısıyla ağ maliyeti artar.



Şekil 5.6 Ortalama yol değişimi

Son olarak, istemcinin ilk istek gönderdiği zaman ile ilk video karesinin oynatım zamanı arasında geçen süre olan başlangıç gecikmesi ölçülmüştür. Tüm senaryolarda, benzer başlangıç gecikmesi ölçülmüş ve sisteme eş zamanlı katılan kullanıcılar ile ağ senaryoları için tüm yaklaşımlar arasında anlamlı bir fark gözlemlenmemiştir. Ortalama olarak, ön-tamponlama süresi de dahil olmak üzere başlangıç gecikmesi 6 saniyeye eşittir. Ancak, periyodik girişler ile statik hafif yüklü ağ senaryosunda diğer yaklaşımlarda başlangıç gecikmesi tekrar altı saniyeye eşit iken, uyarlamalı olmayan en kısa-yol yönlendirmesi benzetimlerinde başlangıç gecikmesi 25 saniyeye kadar çıkmaktadır. Periyodik girişli senaryoda uyarlamalı olmayan en kısa-yol yönlendirmesinde, en son kullanıcının geciken paketleri ağda yüksek derece sıkışıklığa neden olmaktadır. Bu sebeple, belirtilen durumda yüksek başlangıç gecikmeleri gözlemlenmektedir.

6. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, OpenFlow ağları üzerinde çalışan çıkarsamaya dayalı çokluortam akışlandırma uygulaması gerçekleştirilmiştir. Bu çalışmada kontrol biriminin çevreyi gözlemleyerek verdiği kararların sonuçlarını değerlendirmesi ve istemci durumu ile uyumlu olan en uygun kararı öğrenmesi hedeflenmiştir. Bu sebeple, kontrol birimi tarafından, yol ve kalite değişimlerini en alt seviyeye indirerek dinamik yönlendirme ve dinamik kalite değişimleri ile ilgili en uygun işlemlerin yapılması için pekiştirmeli öğrenme modeli tasarlanmıştır. Yapılan öğrenme tabanlı yaklaşım çalışması ile geleneksel en kısa-yol yönlendirmesi başarımları karşılaştırılmış ve tüm senaryolar için ortalama kayıp kare oranının %89 azaldığı görülmüştür. Ayrıca, OpenFlow ağları üzerinde çalışan uyarlamalı çokluortam akışlandırma uygulaması için aç-gözlü yaklaşım da uygulanmıştır. Aç-gözlü yaklaşımda kontrol birimi, ağ kapasitesindeki değişimlere ani tepki verme suretiyle istemci kalitesini en iyilemeye çalışmaktadır. Aç-gözlü yaklaşım ile tez kapsamında önerilen öğrenme tabanlı yaklaşım karşılaştırıldığında, öğrenme tabanlı yaklaşımın kayıp kare oranının %70 azaldığı, yol değişikliğinin %91 azaldığı ve %32 oranında kalite değişimlerinin azaldığı görülmüştür. Bu sonuçlar kullanıcı deneyim kalitesi ile ilgili parametreler ve ağ maliyeti bakımından, aç-gözlü yaklaşıma göre önerilen öğrenme tabanlı yaklaşımın başarımlarının daha yüksek olduğunu göstermektedir.

Bu gibi öğrenme modeli için ana tasarım motivasyonu, paket kayıp oranını, kalite değişikliğini ve kontrol birimi maliyetini en aza indirebilmek için sistemin ne derecede uyarlanabilir olduğunu belirlemektir. Bu parametrelerin en aza indirgenebilmesinin yanında, benzetim sonuçları, büyük oranda uyarlanabilir sistemlerin sadece maliyetli değil, aynı zamanda kullanıcı deneyim kalitesinin düşmesine de neden olduğunu göstermektedir. Bunun başlıca nedeni, OpenFlow protokollerinin gerçek zamanlı ağ kapasitesini sorgulamasına rağmen, özellikle dinamik ağlarda akış yolu değişikliğine karar verilirken doğru trafik bilgisine ulaşmanın genellikle çok zor olmasıdır.

Gelecekte yapılacak çalışmalarda, birden çok etki alanına sahip ağlar üzerinde öğrenme tabanlı bu yaklaşım uygulanarak, kontrol birimleri arasındaki

mesaj karmaşıklığı en aza indirgenebilmesi geliştirilebilir.

KAYNAKLAR DİZİNİ

- Alpaydin, E.**, 2004, Introduction to Machine Learning, Machine Learning, The MIT Press.
- Bertsekas, D.P.**, 2007, Dynamic Programming and Optimal Control, Vol. II (3rd ed.), Athena Scientific.
- Bertsimas D. and Tsitsiklis J. ,** 1993, Simulated Annealing, Statistical Science, Vol. 8, Number 1, 10-15pp.
- Cetinkaya, C., Karayer, E., Sayit, M. and Hellge, C.**, 2014, SDN for Segment Based Flow Routing of DASH, In Proceedings of the IEEE 4th International Conference on Consumer Electronics - Berlin (ICCE-Berlin), 74-77 pp.
- Egilmez, H.E., Dane, S.T., Bagci, K.T. and Tekalp, A.M.**, 2012, OpenQoS: An OpenFlow Controller Design for Multimedia Delivery with End-to-End Quality of Service over Software-Defined Networks, In Proceedings of Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), 1-8 pp.
- Egilmez H.E., Civanlar S. ve Tekalp A.M.**, 2013, An Optimization Framework for QoS-Enabled Adaptive Video Streaming over OpenFlow Networks, IEEE Transactions On Multimedia, vol. 15, no. 3, 710-715 pp.
- Egilmez, H.E. ve Tekalp, A.M.**, 2014, Distributed QoS Architectures for Multimedia Streaming over Software Defined Networks, IEEE Transactions On Multimedia, vol. 16, no. 6, 1597-1609 pp.
- Floodlight**, <http://www.projectfloodlight.org/floodlight/> (Erişim tarihi: 09 Şubat 2015)
- Georgopoulos, P., Broadbent, M., Plattner, B. and Race, N.**, 2014, Cache as a Service : Leveraging SDN to Efficiently and Transparently Support Video-on-Demand on the Last Mile, In Proceedings of 23rd International Conference on Computer Communications and Networks (ICCCN).

KAYNAKLAR DİZİNİ(devam)

- Georgopoulos, P., Elkhatib, Y., Broadbent, M., Mu, M. and Race, N.,** 2013, Towards Network-Wide QoE Fairness Using OpenFlow-Assisted Adaptive Video Streaming”, In Proceedings of ACM SIGCOMM Workshop on Future Human-Centric Multimedia Networking, 15-20 pp.
- Gorkemli B., Tekalp A.M.,** 2007, Streaming SVC Coded Video over DCCP, Signal Processing and Communications Applications.
- İmamoğlu, N., Eresen, A. ve Efe, M.Ö.,** 2009, Döner Sarkaç Sisteminin Pekiştirmeli Öğrenme Algoritmaları ile Kontrolü, TOBB Ekonomi ve Teknoloji Üniversitesi, Ankara.
- Kassler, A., Skorin-Kapov, L., Dobrijevic, O., Matijasevic, M. and Dely, P.,** 2012 Towards QoE-Driven Multimedia Service Negotiation and Path Optimization with Software Defined Networking, In Proceedings of 20th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 1-5 pp.
- Kurose, J. and Ross, K.,** 2013, Computer Networking A Top-Down Approach, Pearson, Boston, 889 p.
- Laga, S., Van Cleemput, T., Van Raemdonck, F., Vanhoutte, F., Bouten, N., Claeys, M. Amd De Turck, F.,** 2014, Optimizing Scalable Video Delivery Through OpenFlow Layer-Based Routing, In Proceedings of the IEEE Network Operations and Management Symposium (NOMS), 1-4 pp.
- Lantz, B., Heller, B. and McKeown, N.,** 2010, A Network In a Laptop: Rapid Prototyping for Software-Defined Networks, In Proceedings of ACM SIGCOMM Workshop on Hot Topics in Networks, 1-6 pp.
- McDonagh, P., Olariu, C., Hava, A. and Thorpe, C.,** 2013, Enabling IPTV Service Assurance Using OpenFlow, In Proceedings of 27th International Conference on Advanced Information Networking and Applications Workshops (WAINA), 1456-1460 pp.

KAYNAKLAR DİZİNİ(devam)

- McKeown, N., Anderson, T., Balakrishnan H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S. and Turner, J.,** 2008, OpenFlow: Enabling Innovation In Campus Networks, SIGCOMM Computer Communications Review., vol. 38, no. 2, 69-74 pp.
- ONF(Open Networking Foundation) White Paper,** 2012, Software Defined Networking: The New Norm for Networks.
- OpenFlow Switch Specification Version 1.0.0,** 2009, <http://archive.openflow.org/documents/openflow-spec-v1.0.0.pdf>, (Erişim tarihi: 9 Şubat 2015).
- OpenFlow Tutorial,** http://archive.openflow.org/wk/index.php/OpenFlow_Tutorial, (Erişim tarihi: 9 Şubat 2015).
- OpenDayLight,** <http://www.opendaylight.org/project/technical-overview>, (Erişim tarihi: 9 Şubat 2015).
- Renzi D., Amon P., Battista S.,** 2008, Video Content Adaptation Based on SVC and Associated RTP Packet Loss Detection and Signaling, WIAMIS, 97-100 pp.
- Sayıt M., Tunah E.T.,** 2012, Adaptive Video Streaming with H.264 SP Frames, *Pamukkale University Journal of Engineering Sciences*, Vol. 18, Issue 2, 73-83 pp.
- Schulzrinne, H., Casner, S., Frederick, R. and Jacobson, V.,** 2003, Packet Design.
- Schwarz, H., Marpe, D. and Wiegand, T.,** 2007, Overview of the Scalable Video Coding Extension of the H.264/AVC Standard, IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no.9, 1103-1120 pp.

KAYNAKLAR DİZİNİ(devam)

- Seddiki, M.S., Shahbaz, M., Donovan, S., Grover, S., Park, M., Feamster, N. and Song, Y.**, 2014, FlowQoS: QoS for The Rest of Us, In Proceedings of the third workshop on ACM Hot topics in software defined networking (HotSDN), 207-208 pp.
- Setton E. and Girod B.**, 2006, Rate-Distortion Analysis and Streaming of SP and SI frames, *IEEE Transactions on Circuits and Systems for Video Technology (CSVT)*, Vol. 16(No. 6): 733-743 pp.
- Setton E., Noh J., Girod B.**, 2005, Rate-Distortion Optimized Video Peer-to-Peer Multicast Streaming, P2PMMS, 39-48 pp.
- Shao B., Renzi D., Amon P., Xilouris G., Zotos N., Battista S., Kourtis A., Mattavelli M.**, 2010, An Adaptive System for Real-Time Scalable Video Streaming with End-to-End QoS Control, 11th Int. Work. on Image Analysis for Multimedia Interactive Services (WIAMIS).
- Sodagar, I.**, 2011, The MPEG-DASH Standard for Multimedia Streaming over the Internet, IEEE Multimedia.
- Software Defined Networking**, http://en.wikipedia.org/wiki/Software-defined_networking, (Erişim tarihi: 9 Şubat 2015).
- Sutton, R. and Barto, A.**, 1998, Reinforcement Learning: An Introduction, MIT Press.
- Taqqu M. S., Willinger W. and Sherman R.**, Apr. 1997, Proof of a Fundamental Result in Self-Similar Traffic Modeling, *ACM SIGCOMM Computer Communication Review*, Volume 27 Issue 2, 5-23 pp.
- Wiegand T., Sullivan G., Bjontegaard G., and Luthra A.**, 2003, Overview of the H.264/AVC video coding standard, *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 13, No. 7, 560 –576 pp.

ÖZGEÇMİŞ

Tuba UZAKGİDER

Adres: Erzene Mahallesi Türkeli Caddesi Bağdiken Apt. No:10 Kat: 6 Daire: 6
Bornova/İZMİR

Eğitim Durumu

Yüksek Lisans : 2010- , Ege Üniversitesi, Uluslararası Bilgisayar
Enstitüsü
Lisans : 2005-2009, Gazi Üniversitesi, Teknik Eğitim Fakültesi
Lise : 2001-2005, Denizli Türk Eğitim Vakfı (TEV) Anadolu
Lisesi

Yabancı Dil

Türkçe : Anadil

İngilizce : İyi derecede

İş Deneyimi

2009-2010, T.C. Milli Eğitim Bakanlığı Gündoğan Fahriye Ilıcak Mesleki ve
Teknik Anadolu Lisesi Bilişim Teknolojileri Alanı Öğretmeni

2010- ,T.C. Milli Eğitim Bakanlığı Bornova Altay Mesleki ve Teknik Anadolu
Lisesi Bilişim Teknolojileri Alanı Öğretmeni

Programlama Dilleri

Java, C/C++, C#, ASP.NET