



EGE ÜNİVERSİTESİ

YÜKSEK LİSANS TEZİ

**GÖREVDEŞ AĞLARDA IZGARA TABANLI
VİDEO AKIŞLANDIRMA UYGULAMALARI İÇİN
İŞBİRLİKÇİ SEÇİMİNİN ETMEN TABANLI
GERÇEKLENMESİ**

Kemal Deniz TEKET

Tez Danışmanı : Yrd. Doç. Dr. Geylani Kardaş

Uluslararası Bilgisayar Anabilim Dalı

Bilim Dalı Kodu : 619.02.04

Sunuş Tarihi : 27.11.2013

Bornova-İZMİR

2013

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

(YÜKSEK LİSANS TEZİ)

**GÖREVDEŞ AĞLARDA IZGARA TABANLI
VİDEO AKIŞLANDIRMA UYGULAMALARI
İÇİN
İŞBİRLİKÇİ SEÇİMİNİN ETMEN TABANLI
GERÇEKLENMESİ**

Kemal Deniz TEKET

Tez Danışmanı: Yrd. Doç. Dr. Geylani Kardaş

Uluslararası Bilgisayar Anabilim Dalı

Bilim Dalı Kodu: 619.02.04

Sunuş Tarihi: 27.11.2013

Bornova-İZMİR

2013

Kemal Deniz TEKET tarafından YÜKSEK LİSANS tezi olarak sunulan “Görevdeş Ağlarda Izgara Tabanlı Video Akışlandırma Uygulamaları İçin İşbirlikçi Seçiminin Etmen Tabanlı Gerçeklenmesi” 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 27.11.2013 tarihinde yapılan tez savunma sınavında aday oybirliği/ ile başarılı bulunmuştur.

Jüri Üyeleri:

İmza

Jüri Başkanı : Yrd. Doç. Dr. Geylani Kardaş

Raportör Üye : Yrd. Doç. Dr. Müge Fesci Sayıt

Üye : Doç Dr. Rıza Cenk Erdur

ÖZET**GÖREVDEŞ AĞLARDA IZGARA TABANLI VIDEO AKIŞLANDIRMA UYGULAMALARI İÇİN İŞBİRLİKÇİ SEÇİMİNİN ETMEN TABANLI GERÇEKLENMESİ**

TEKET, Kemal Deniz

Yüksek Lisans Tezi, Uluslararası Bilgisayar Anabilim Dalı

Tez Yöneticisi: Yrd. Doç. Dr. Geylani Kardaş

Kasım 2013, 58 sayfa

Görevdeş ağlarda ızgara tabanlı video akışlandırma uygulamalarının zaman gereksinimlerinin yerine getirilmesi ve ağ kaynaklarının etkin kullanılması için yazılım etmenlerinin uygulamaları üzerine çalışmalar bulunmaktadır. Bununla beraber, bu çalışmalarda eş seçimi ve ağ farklılıkları konularına değinilmemiştir. Bu boşluğu doldurmak için, bu tezde görevdeş ağlarda ızgara tabanlı video akışlandırma uygulamaları için bir çok etmenli sistem tasarımı ve uygulaması sunulmuştur. Çok etmenli sistemin tasarımı için Prometheus metodolojisi kullanılmış, etmenlerin iç yapıları genel kabul gören Kanı-İstek-Hedef mimarisine göre hazırlanmıştır. Sistemin uygulaması Network Simulator-3 platformunda gerçekleştirilmiştir. Test sonuçları eş seçimi ve ağ farklılık algoritmaları sayesinde önerilen çok etmenli sistemin performansta iyileştirmeler sağladığı göstermiştir. Bu tezde verilen çok etmenli sistem tasarımının, görevdeş ağlarda yapılacak bundan sonraki çalışmalara yol göstereceğine inanılmaktadır.

Anahtar sözcükler: etmen; çok etmenli sistemler; kanı-istek-hedef; görevdeş ağlar; video akışlandırma.

ABSTRACT**AGENT BASED PARENT SELECTION IN PEER-TO-PEER MESH-BASED VIDEO STREAMING SYSTEMS**

TEKET, Kemal Deniz

MSc. in International Computer

Supervisor: Asst. Prof. Dr. Geylani Kardaş

November 2013, 58 pages

Research on the use of software agents in P2P video streaming systems has recently emerged in order to meet the requirements such as time constraints, minimizing delay and efficient utilization of network resources. However parent selection and network awareness are not addressed properly in these leading studies. In order to fill this gap, we introduce the design and implementation of a Multi-agent System (MAS) for Peer-to-Peer (P2P) video streaming in this thesis. The construction of the MAS by applying a well-defined agent-oriented software engineering methodology is discussed. The internal structure of each agent in the proposed MAS is based on the well-known Belief-Desire-Intention (BDI) architecture. Experimental results show that the proposed MAS succeeds the related parent selection and promotes the system execution. BDI-based MAS design given in this work can guide those researchers who aim to implement autonomous mesh-based P2P video streaming systems.

Keywords: agent; multi-agent system (MAS); belief-desire-intention (BDI); peer-to-peer (P2P); video streaming.

TEŞEKKÜR

Öncelikle bu tez konusu üzerinde bana çalışma imkanı sunan ve ayrıca danışmanım olan Yrd. Doç. Dr. Geylani Kardaş' a, deneyimleri, bilgileri ve önerileriyle araştırma ve geliştirmeye yönlendirmesi ve sağladığı kaynaklar ile destek olmasından dolayı teşekkürü bir borç bilirim. Ayrıca tez süresince her türlü fikir ve desteklerinden dolayı Yrd. Doç. Dr. Müge Fesci Sayıt'a, çalışma arkadaşlarım Yağız Kaymak ve Erdem Karayer'e, ve son olarak tez çalışmalarım boyunca yanımda olan, sorunlarımı paylaşan ve yol gösteren, kuzenime, annem ve ablama teşekkür ederim.

“111E022” numaralı “Görevdeş Ağlarda Video Akışlandırma Uygulamaları için Etmen tabanlı bir Alternatif Yol Seçim Yönteminin Geliştirilmesi” isimli proje kapsamında yüksek lisans çalışmamı maddi olarak destekleyen Türkiye Bilimsel ve Teknolojik Araştırma Kurumu' na (TÜBİTAK) teşekkürlerimi sunarım.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	v
ABSTRACT	vii
TEŞEKKÜR	ix
ŞEKİLLER DİZİNİ	xv
ÇİZELGELER DİZİNİ	xvii
1. GİRİŞ	1
2. LİTERATÜR ÖZETİ	3
3. ALTYAPI	6
3.1 Yazılım Etmenleri	6
3.1.1 Kanı-İstek-Hedef mimarisi	7
3.2 Görevdeş Ağlarda Video Akışlandırma	7
3.2.1 Video akışlandırma çözümleri	8
3.2.2 Görevdeş ağlarda video akışlandırma altyapısı	10
3.2.3 Görevdeş ağlarda video akışlandırmanın genel yapısı	14

İÇİNDEKİLER (devam)

Sayfa

4. ÇOK ETMENLİ SİSTEMİN PROMETHEUS İLE KANI-HEDEF-İSTEK MODELİNE UYGUN ANALİZ VE TASARIMI.....	16
4.1 Sistem Tanımlamaları	17
4.2 Mimari Tasarım	19
4.3 Detaylandırılmış Tasarım.....	22
5. NS-3'TE UYGULAMA	32
5.1 Ağ Topolojisinin Oluşturulması	32
5.2 Sistemdeki Dğümler ve Özellikleri.....	33
5.3 Dğümler Arası İletişim	35
5.3.1 Eşin sisteme bağlanması	35
5.3.2 Eşin işbirlikçilerini belirlemesi	36
5.3.3 İşbirlikçiler arası haberleşme	40
5.4 Izgara Tabanlı Akışlandırma Sisteminde Video Gönderimi.....	41
6. SİMULASYON SONUÇLARI VE YORUMLAR	46
6.1 Simulasyon Ortamı	46
6.2 Simulasyon Sonuçları	47

İÇİNDEKİLER (devam)

	<u>Sayfa</u>
7. SONUÇ VE ÖNERİLER	50
KAYNAKLAR DİZİNİ.....	51
ÖZGEÇMİŞ.....	57
EKLER.....	

ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
3.1 İstemci-Sunucu Modeli	9
3.2 Görevdeş Ağ Modeli	10
3.3 Ağaç Yapısı	11
3.4 Çoklu Ağaç Yapısı	11
3.5 Izgara Tabanlı Yapı	12
3.6 Melez Veri Dağıtımı.....	13
3.7 Görevdeş Ağlarda Video Akışlandırmanın Genel Yapısı	15
4.1 Prometheus Metolojisinin Aşamaları	16
4.2 Sistem Hedef Diyagramı	18
4.3 System Join, Partnership, Video Request Etkileşim Diyagramları	21
4.4 Sistem Genel Bakış Diyagramı	22
4.5 Peer Etmen Genel Bakış Diyagramı.....	23
4.6 Video Server Etmen Genel Bakış Diyagramı.....	26
4.7 Tracker Server Etmen Genel Bakış Diyagramı	28

ŞEKİLLER DİZİNİ (devam)

<u>Şekil</u>	<u>Sayfa</u>
5.1 BRITE ile Üretilmiş Örnek Topoloji	33
5.2 Eşin Sisteme Bağlanması	35
5.3 Eşin İşbirlikçilerini Belirlemesi	37
5.4 İşbirlikçi Listesi Dolu Olan Bir Eşin İsteği Yönlendirmesi.	38
5.5 Bir Eşin İşbirlikçi Olarak Kabul Edilmesi	39
5.6 PARTNERSHIP_NACK Mesajı	40
5.7 İşbirlikçiler Arası Tampon Harita Değişimi.....	41
5.8 Video İsteği Gönderilmesi	44
6.1 Network Trafikindeki Paket Oranı	48

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
5.1 Ebeveyn Seçimi için Açıklamalar	42
6.1 Kümülatif Yükleme Bant Genişlikleri Dağılımı	45
6.2 Ortalama Devamlılık Endeksi	46
6.3 Ortalama Video İzlemeye Başlama Süresi	47
6.4 Ortalama Ebeveyn Değişirme Sayısı	47

1. GİRİŞ

Günümüzde video akışlandırma uygulamaları ağ kaynaklarının büyük bir bölümünü kullanmaktadır. Görevdeş ağlarda video akışlandırma sistemleri, video verilerinin eşler (“peer”) arasında olmasını sağlar; böylece video sunucularındaki yükü azaltır. Görevdeş ağlarda video akışlandırma uygulamaları (PPLive, 2013; PPStream, 2013; Xie et al., 2007) ile son kullanıcılar internet üzerinden video izlerken aynı zamanda içerik sağlayıcı görevini üstlenir. Bu uygulamalarda kullanıcı deneyim kalitesinin (“quality of experience”) yeterli seviyede olması için zaman kısıtlarına dikkat edilmeli ve sağlam bir üst katman (“overlay”) oluşturulmalıdır. Ayrıca ağ kaynaklarının optimum kullanımı da göz önünde bulundurulmalıdır. Bu sistemlerde her eş sisteme katkı sağlayabilir, fakat öncelikli görevi kendisinin video izleyebilmesidir.

Görevdeş ağlarda video akışlandırma sistemlerinin bu özellikleri dikkate alındığında, yazılım etmenlerinin (“software agent”) sistem tasarım ve uygulamalarında kullanımı uygun bir altyapı oluşturmayı kolaylaştırabilir. Wooldridge’in (2009) tanımına göre, bir yazılım etmeni belirli bir çevre içinde konumlanan, bu çevre içinde tasarım amaçları doğrultusunda özerk eylemlerde bulunma yeteneğine sahip olan bir bilgisayar sistemidir. Bu bağlamda, yazılım etmenlerinin video akışlandırma sistemlerinde kullanımı üzerine çalışmalar yapılmaya başlamıştır (Molina et al., 2009; Pournaras et al., 2009; Chen et al., 2010; Carrera and Iglesias, 2011; Orynczak and Kotulski, 2011; Menkovski and Liotta, 2013). Bununla birlikte, bu çalışmalarda hata tespiti ve işbirlikçi sistemler ile ilgili çözümler verilmiş ancak video akışlandırma sistemlerinin eş seçimi ve ağ farklılıkları gibi genel ihtiyaçları göz önünde bulundurulmamıştır.

Tez kapsamında video akışlandırma sistemlerinde Kani-İstek-Hedef (“KİH”) (Rao and Georgeff, 1995) modeline göre yeni bir Çok-etmenli Sistem önerilmiştir. Sistemdeki etmenlerin iç yapısı verilmiş ve network simulator 3’te (ns-3) (ns-3, 2013) yapılan simulasyon sonuçları aktarılmıştır. Simulasyonlarda önerilen çok-etmenli sistem ile CoolStreaming (Xie et al., 2007) benzeri sistem testleri yapılmıştır. Etmenlerde uygulanan Ebeveyn Puanlama, Çocuk Yönetme ve Yeni Eş Karşılama davranışları sayesinde CoolStreaming benzeri sisteme göre iyileştirmeler elde edilmiştir. Bu iyileştirmeler devamlılık endeksi, video izlemeye başlama süresi, ebeveyn değiştirme sayısı ve ağ trafik oranı parametreleri ile gösterilmiştir. Bu parametreler iki sebeple seçilmiştir: İlki devamlılık endeksi, video izlemeye başlama süresi ve ebeveyn değiştirme sayısının kullanıcının

deneyim kalitesinde önemli göstergeler olmasıdır. İkinci sebep ise görevdeş ağ uygulamalarının ağ altyapısına fazla yük getirmektedir ve bu fazla yükün azaltılması önem arz etmektedir. Bunun için ağ trafik oranı verilmiştir.

Tez çalışması, TÜBİTAK Araştırma Destek Programları Başkanlığı (ARDEB) tarafından desteklenen 111E022 no' lu "Görevdeş Ağlarda Video Akışlandırma Uygulamaları için Etmen tabanlı bir Alternatif Yol Seçim Yönteminin Geliştirilmesi" başlıklı Ar-Ge projesi kapsamında yürütülmüştür. Tezde önerilen Çok-etmenli Sistem için bu projede geliştirilen akışlandırma ağının altyapısı kullanılmıştır.

Tezin sonraki bölümleri aşağıdaki gibi düzenlenmiştir: İkinci bölümde, konuyla ilgili literatür araştırması aktarılmıştır. Üçüncü bölümde öncelikle yazılım etmenleri anlatılmıştır. Sonrasında görevdeş ağlarda video akışlandırma ile ilgili bilgi verilmiştir. Dördüncü bölümde ns-3'te yapılan simulasyon platformu açıklanmaktadır. Beşinci bölümde, Prometheus (Padgham ve Winikoff, 2003) metodolojisi ile geliştirilen etmen sistemi ve bu sistemin uygulamadaki detayları anlatılmıştır. Altıncı bölümde önerilen çok etmenli sistem ile CoolStreaming benzeri sistemin simulasyon sonuçları verilmiş ve değerlendirilmiştir. Son bölümde tezde elde edilen sonuçlara ve ileriye dönük yapılabilecek çalışmalara yer verilmiştir.

2. LİTERATÜR ÖZETİ

Son dönemde ilgili literatürde görevdeş ağlarda video akışlandırma sistemlerinde çok etmenli sistemlerin kullanılmasını öneren çalışmalar yer almaktadır (Molina et al., 2009; Pournaras et al., 2009; Chen et al., 2010; Carrera and Iglesias, 2011; Orynczak and Kotulski, 2011). Bu sistemler daha çok hata teşhisi ve işbirliğini göz önüne almaktadır. Hata teşhisi üzerine yapılan çalışmalar video akışlandırmasında ortaya çıkabilecek sorunların nasıl giderilebileceğine yoğunlaşırken, işbirlikçi sistemler video akışlandırma performansını ve/veya servis kalitesini arttırmak ve içerik göndermek için etmenler arasındaki anlaşma mekanizması üzerinde durmuşlardır. Bu bölümde literatürde bulunan güncel çalışmalara yer verilmiştir.

(Carrera and Iglesias, 2011), video akışlandırma sorunlarının özerk olarak kontrol edilmesi amacıyla çok etmenli bir mimari önermiştir. Önerilen sistem 2 şekilde kullanılmıştır: video servis sağlayıcıları ve tüketicilerinin yer aldığı video akışlandırma. Ağ kontrolünü gerçekleştirmek için hipotez üretimi ve hipotez onaylanması olmak üzere iki bileşen ortaya konmuştur. Hipotez üretimi ağdaki sorunları analiz ederek sorunun kaynağını çıkarsamak için Bayesian çıkarsama yöntemini kullanır. Bu aşama ile elde edilen çıktı, hangi hipotezin onaylanacağına karar verilen ikinci aşamaya girer.

(Chen et al., 2010), evrimsel oyun teorisi (“evolutionary game theory”) yöntemini kullanarak görevdeş ağlarda video akışlandırma uygulamalarının performansını arttırmayı hedeflemiştir. Sistem performansını arttırmak için akışlandırma sistemi içinde yazılım etmenlerini gruplandırarak gerçek zamanlı bir video akışlandırma sistemi önermişlerdir. Bu çalışmada, video akışlandırma oturumuna katılmak isteyen eşler, akışlandırma performanslarını arttırmak için birbirleriyle iletişim kurarak gruplar oluşturmaktadır. Oluşturulan gruplar da birbirleriyle evrimsel oyun teorisine göre iletişim kurarak birbirlerine video aktarımı yapmaktadır. Bu yaklaşıma göre, sistemde yer alan her grubun video yükleme ve indirme kapasitesi bulunmaktadır. Her grupta normal eşler ve etmenler olmak üzere iki tür eş bulunmaktadır. Diğer gruplardan akışlandırma verilerini almak için normal eşler aralarında pazarlık yaparak k adet eş etmen olarak tayin ederler. Seçilen etmenler diğer gruplardan video verilerini aldıktan sonra verileri grup içinde paylaşırlar. Buradaki problem k sayısının ne olacağı, bir başka deyişle, grup üyeleri içinden kaç tane eşin etmen olarak seçileceğidir. Yapılan çalışma işbirlikçi olmayan yöntemlerle kıyaslanmış ve önerilen

algoritmanın video akışlandırma performansını daha yüksek oranda başardığı iddia edilmiştir.

(Pournaras et al., 2009), dayanıklı ağaç topolojileri oluşturabilmek için uyarlanabilir sanal organizasyon modeli (“adaptive virtual organization model”) önermiştir. Bu çalışmada, her yazılım etmeni bir düğüm ile ifade edilir. Bu düğümler bir sorun olduğu zaman etmenler tarafından yönetilir. Ağaç topolojilerinde en önemli sorunlardan biri bir düğümün sistemden ayrılmasıdır. Bu nedenle bu çalışmada ağaç topolojisindeki düğümlerin yaratabileceği sorunların etkileri üzerine yoğunlaşarak etmenlerin olası hataları nasıl iyileştirebileceği araştırılmıştır.

(Molina et al., 2009) içinde birçok gezici cihazın birbirlerine çoklu ortam içeriği ilettiği gezici ve geçici süreli ağlar (“mobile transient networks”) için bir mimari önerilmiştir. Gezici ağlar içinde çoklu ortam içeriğinin etmenler arasında dağıtılması amacıyla pazarlık protokolü (“negotiation protocol”) modeli kullanılmıştır. Çoklu ortam içeriğinin gönderilip gönderilmemesi kararı etmenler arasında yapılan pazarlık protokolü ile belirlenmektedir. Gezici ağdaki her cihaz, içerik ağına (“content network”) bağlanan bir etmeni temsil etmektedir. Ek olarak, gezici ağda çoklu ortam içeriğini indirmek için öneride bulunan bir koordinatör etmen bulunmaktadır. Koordinatör etmeni, indirilmek istenen içeriği diğer etmenlere önerdikten sonra diğer etmenler bu içeriğin indirilip indirilmeyeceğine dair aralarında pazarlık yaparak ortak bir karara varırlar. İçeriği indirmek isteyen etmenler koordinatör etmeninden içerik verilerini almaya başlar.

(Orynczak and Kotulski, 2011) servis kalitesi ve güvenlik konularında gerçek zamanlı video uygulamaları için etmen tabanlı bir mimari sunmuştur. Sistem alt katmandaki düğümleri temsil eden ve daha etkili yönlendirmeyi sağlayan dinamik yönlendirme tablosu (“dynamic routing table”) oluşturmak için birbirleriyle iletişim kuran etmenlerden oluşur. Her etmen diğer etmenlerden aldığı bant genişliği, kayıp paketler ve paketler arasındaki zaman farkı gibi değerleri düzenli olarak analiz ederek akışlandırma kalitesini kontrol etmektedir. Etmenler verilerle ilgili bir değişiklik fark ettiklerinde güncellenen dinamik yönlendirme tablosunu oluşturmak için bu verileri kullanmaktadır.

(Menkovski and Liotta, 2013), kullanıcı deneyim kalitesini arttırmak için akıllı bir kontrol sistemi önermiştir. Bu sistemde etmen, internet trafiğini

pekiştirilmiş öğrenme (“reinforcement learning”) kullanarak inceler ve en uygun kararları alır.

Yukarıdaki çalışmalar görevdeş ağlardaki video akışlandırma sistemlerindeki bazı problemlere çözüm getirmeyi hedeflemiştir. Fakat, video sistemlerine bütüncül bir bakış açısı ile çözüm getirilmemiştir. Buna ek olarak, görevdeş ağlarda video akışlandırmada eş seçiminin önemi ve trafik lokalizasyonu gibi önemli konulara değinilmemiştir

3. ALTYAPI

Bu bölümde ilk olarak yazılım etmenleri hakkında genel bilgi verilmekte daha sonra görevdeş ağlarda video akışlandırma konusu anlatılmaktadır.

3.1 Yazılım Etmenleri

Yazılım etmenleri için en kabul gören tanım Wooldridge tarafından ortaya atılmıştır. Wooldridge' e (2009) göre, etmen şöyle tanımlanmıştır:

“Bir etmen belirli bir **çevre** içinde bulunan, tasarım amaçları doğrultusunda bu çevrede **özerk** eylemlerde bulunan bir bilgisayar sistemidir.”

Wooldridge (2009), ayrıca etmenlerin 4 özelliğinden bahsetmiştir: Özerk (“Autonomous”), Karşıt-eylemli (“Reactive”), Proaktif (“Proactive”), Sosyal.

- Özerk: Etmenler yapacakları işleri kendileri belirlemelidir. Bir kullanıcı veya başka etmenlerden etkilenmezler.

- Karşıt-eylemli: Akıllı etmenler, çevresinde neler olup bittiğinin farkında olmalıdır. Etmen, çevresinde bir değişiklik olduğunda, tasarım amaçları doğrultusunda bu değişikliğe cevap verir.

- Proaktif: Akıllı etmenler, hedef yönelimli hareket edebilmelidir. Tasarım amaçlarını gerçekleştirebilmek için gerektiğinde girişkenlik (“initiative”) göstermelidir.

- Sosyal: Akıllı etmenler, tasarım amaçlarını gerçekleştirebilmek için diğer etmenlerle ve insanlarla etkileşime girme yeteneğine sahip olmalıdır.

Etmen sistemlerinin mimarilerinin oluşturulması sırasında en çok karşıt-eylemli veya kanı-istek-hedef modelleri kullanılmaktadır. Bu tez kapsamında karşıt-eylemli mimariye göre daha gelişmiş olan kanı-istek-hedef modeli uygulanmıştır. Aşağıdaki altbölümde bu mimari hakkında genel bilgi verilmektedir.

3.1.1 Kanı-İstek-Hedef mimarisi

KİH mimarisinin temeli psikolojiye dayanmaktadır. Bratman (1987) insanın pratik düşünme biçimini (“practical reasoning”) modellemek için ortaya atmıştır. Daha sonra bu modelden esinlenilerek, mantık ve psikoloji bilim dallarına dayanan bir yazılım mimarisi geliştirilmiştir (Rao and Georgeff, 1995).

KİH mimarisi, Çok-Etmenli Sistemler oluşturmak için temel yaklaşımlardan biridir. Adından da anlaşılacağı gibi bu mimaride önemli nokta, sembolik gösterimler kullanarak etmenlerin *Kanı*, *İstek*, *Hedef*lerinin inşa edilmesidir.

Modelin bileşenleri ve yazılım mimarisindeki yerleri aşağıda tanımlanmıştır:

- Kanı (“belief”): Kanılar, etmenin çevre hakkındaki bilgi durumunu temsil eder. Çevreye kendisi ve diğer etmenler de dahildir. Bilgi yerine kanı kavramının kullanılmasının sebebi, etmenin kanılarının doğru olma zorunluluğu olmaması veya gelecekte değişebilecek olmasıdır.

- İstek (“desire”): İstekler, etmenin motivasyon durumunu gösterir. Etmenin gerçekleştirmek istediği amaçları veya durumları temsil eder. Bu Çok-Etmenli Sistemler’de amaçlara ("goal") denk gelir.

- Hedef (“intention”): Hedefler, etmenin bilinç durumunu gösterir (etmenin hangi işi gerçekleştireceğini seçtiğini). Hedefler, etmenin kendini adadığı isteklerdir. Bu Çok-Etmenli Sistemler’de planlara denk gelir.

3.2 Görevdeş Ağlarda Video Akışlandırma

Video akışlandırma, istemciye ağ üzerinden video parçalarının gönderilip, istemcinin video parçalarını birleştirerek oynatmasıdır. İstemcilere video verileri dağıtılırken, istemcilerin videoyu izleyebilmesi için videonun tamamını indirmeyi beklemelerine gerek yoktur.

Video akışlandırma sistemleri verdikleri hizmetlere göre ikiye ayrılır: canlı akışlandırma ve isteğe bağlı video (VoD).

Canlı akışlandırma sistemlerinde -isminden de belli olduğu gibi- video akışı canlı olarak izlenir. Video parçaları sadece belirli zaman aralıklarında ağda yayınlanır.

VoD sistemlerde, video akışları medya sunucularına kaydedilir ve kullanıcının istediği zaman izleyebilmesi hedeflenir. Bu sistem VCR (“Video Cassette Recording”) işlevselliklerini de kullanıcıya sağlar.

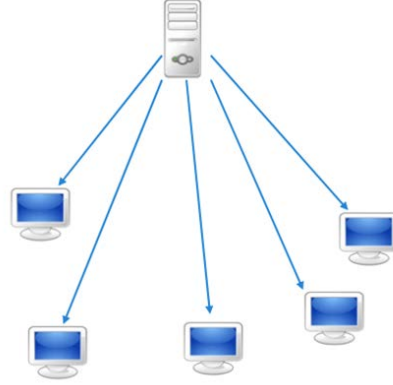
3.2.1 Video akışlandırma çözümleri

Video akışlandırma sistemlerinin uygulanmasında iki farklı çözümden bahsedilebilir: istemci – sunucu ve görevdeş ağlar.

3.2.1.1 İstemci-sunucu

Bu mimaride sunucu video kaynağı, istemci ise izleyici rolündedir. Şekil 3.1’de istemci-sunucu için bir örneğe yer verilmiştir.

Video akışlandırma sistemlerinde istemci – sunucu çözümünün en büyük dezavantajı ölçeklendirilebilme sorunudur. Sunucunun, kaliteli bir videoyu kendisine bağlanacak yüzlerce kullanıcıya göndermesi bantgenişliği yetersizliğinden dolayı mümkün olmayabilir. Ölçeklendirilebilir bir servis sağlamak için sistemde birden çok sunucu bulundurulabilir fakat bu durumda maliyet yükselecektir.



Şekil 3.1. İstemci – Sunucu Modeli

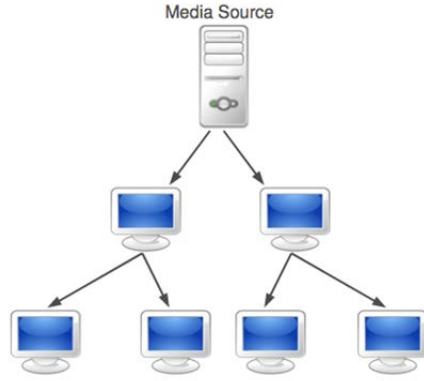
3.2.1.2 Görevdeş Ağlar

Görevdeş ağlarda, düğümler video verisi paylaşımında birbirlerine yardım eder. İhtiyaç duyulan veriye sahip olan düğümler, talepte bulunan düğümlere bu veriyi gönderirler. Şekil 3.2’de bir görevdeş ağ örneği gösterilmiştir.

BitTorrent (Cohen, 2008) benzeri sistemlerden farklı olarak video akışlandırma sistemlerinde dikkat edilmesi gereken durumlar vardır (Hei et al., 2007):

- Video akışlandırma sistemleri gerçek zamanlı çalıştıkları için, zaman kısıtları vardır. Videonun tüm parçalarının oynatımdan önce düğümlerde olması gerekmektedir; zamanında gelmeyen paketler daha sonra istemcilere ulaşsa bile kayıp paket olarak kabul edilir. Buna ek olarak, video izleme başlangıç zamanı (“startup delay”) ve videoyu izleyen düğümler arasındaki oynatma zaman farkları (“playback latency”) çok olmamalıdır.

BitTorrent benzeri sistemler genellikle orta ölçekli gruplarda (<1000 düğüm) çalışır. Bu sebeple bu tür sistemlerde düğüm bilgilerini takipçi sunucudan almak yeterlidir. Fakat büyük ölçekli video akışlandırma sistemleri, binlerce kullanıcının aynı anda sistemde olmasını ve haberleşmesini gerektirir. Bu sebeple BitTorrent benzeri sistemlerden farklı olarak haberleşmede "gossip" protokolleri kullanılır.



Şekil 3.2. Görevdeş Ağ Modeli

3.2.2 Görevdeş ağlarda video akışlandırma altyapısı

Bu bölümde görevdeş ağlarda video akışlandırma için gerekli altyapılar anlatılmıştır.

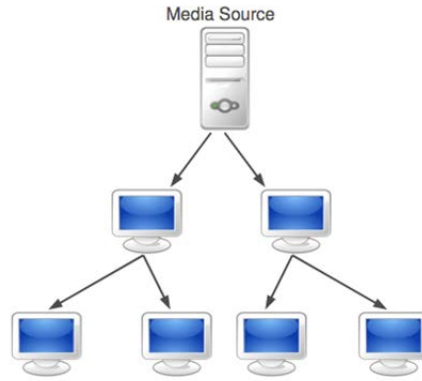
3.2.2.1 Üst-Katman topolojileri

Video akışlandırma sistemlerinde kullanılan üst-katman topolojilerinde iki yöntemden bahsedilebilir. Bu yapılar ağaç yapısı ve ızgara tabanlı yapılar şeklinde kategorize edilir (Liu et al., 2008).

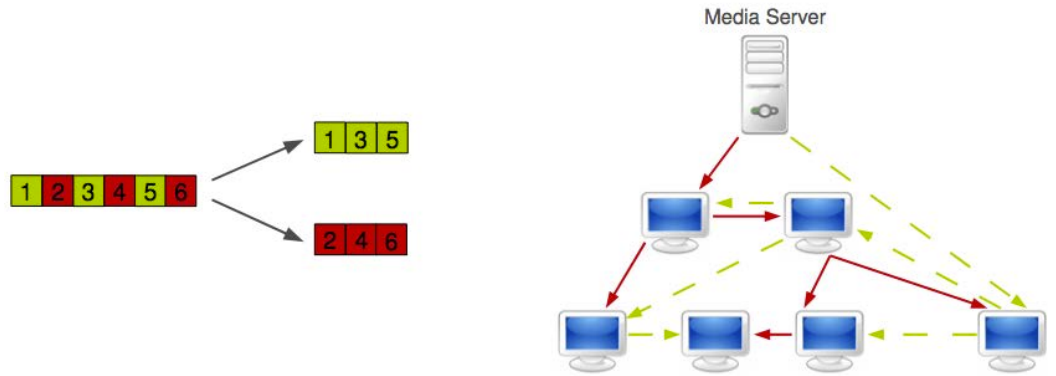
Ağaç Yapısı: Bu yapı da kendi içerisinde ikiye ayrılabilir: tekil ve çoklu gönderim ağaçları. Tekil ağaçlarda, bir adet çoklu gönderim (“multicast”) ağacı oluşturulur. Kök düğümde medya kaynağı bulunur. Düğümler, ara düğümleri ve yaprak düğümleri oluşturur. Veri dağıtımı hızlı ve uygulanması kolaydır. Fakat ağaç yapısı gereği kırılgandır ve sistemdeki yükün büyük kısmını ara düğümler üstlenir. Şekil 3.3’te bir tekil ağaç yapısı gösterilmiştir.

Çoklu ağaç yapısında ise medya kaynağı, akışı alt akışlara böler ve her bir alt akış için ayrı ağaç oluşturulur. Sisteme yeni katılan düğümler bütün ağaçlara katılır. Bu tip bir üst-katman mimarisi, yük dengelemesi açısından orantılıdır ve düğüm kayıplarına dayanıklıdır. Bu mimarinin uygulanması için video verisinin

her ağaçtan ayrı bir parçasının gönderilmesi ve bu parçalardan alındıkça kalitenin yükselmesini sağlayacak şekilde kodlanması gerekir, MDC (Goyal, 2001) bu tip sistemlerde kullanılması önerilen bir video kodlama yöntemidir. Ancak MDC kullanılmayan akışlandırma sistemleri için çoklu ağaç yapısı kullanmak dezavantajlı olabilir. Buna ek olarak bu tip bir mimarinin uygulanması zordur. Şekil 3.4'te bir çoklu ağaç yapısı gösterilmiştir.



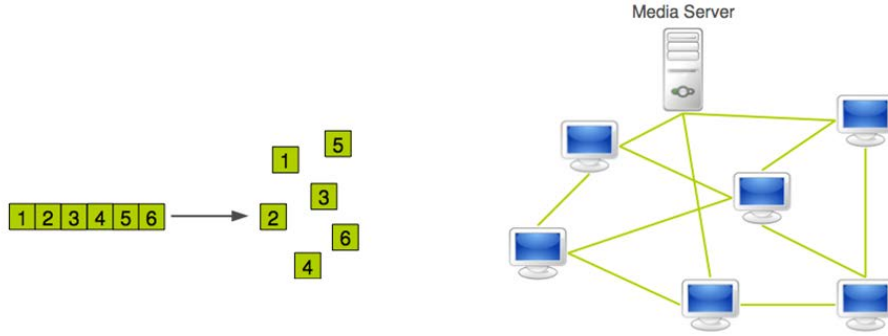
Şekil 3.3. Ağaç Yapısı



Şekil 3.4. Çoklu Ağaç Yapısı

Izgara Tabanlı Yapı: Medya kaynağı, akışı bloklara böler ve düğümlere gönderir. Izgara tabanlı ağa katılan düğümler, periyodik olarak tampon haritalarını değiştirerek, ellerinde bulunan bloklardan haberdar olur ve veri dağıtımını sağlarlar. Bu tip üst-katman mimarisi, yük dengelemesi açısından orantılı, düğüm kayıplarına dayanıklı ve uygulama bakımından kolaydır. Fakat gossip protokolü

ve tampon harita deęişimlerinin periyotlarından dolayı öngörülemeyen gecikmeler yaşanabilir. Şekil 3.5'te örnek bir ızgara tabanlı yapı ve videonun parçalara ayrılması gösterilmiştir.



Şekil 3.5. Izgara Tabanlı Yapı

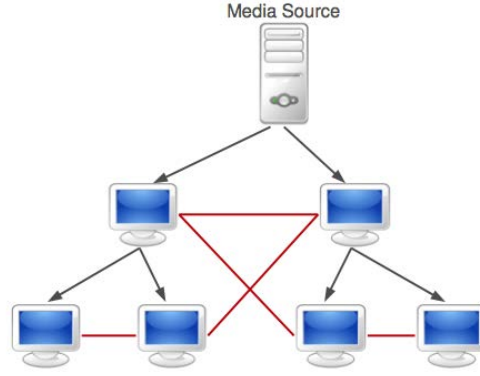
3.2.2.2 Veri dağıtımı

Video akışlandırma sistemlerinde veri dağıtımı için üç yöntem kullanılmaktadır: İtme, Çekme, Melez.

İtme: Bir düğüm aldığı blokları aktif olarak komşularına (çocuklarına) dağıtır. Ağaç topolojilerinde kullanılır. Izgara tabanlı topolojilerde tercih edilmemesinin sebebi tekrar eden mesajların oluşmasıdır.

Çekme: Bu veri dağıtım yönteminde, düğümler kendi aralarında periyodik olarak ellerindeki veriyi tampon haritaları aracılığıyla paylaşırlar. Bir düğüm başka bir düğümden tampon haritasını aldıktan sonra hangi blokları bu düğümden alacağına çeşitli algoritmalarla karar verir. Genellikle ızgara tabanlı ağlarda kullanılır.

Melez: Farklı şekillerde "push" ve "pull" veri dağıtım yöntemleri birleştirilerek melez bir yöntem oluşturulabilir. Genellikle bir ağaç yapısı üzerinden video paketleri düğümlere aktarılır ("push" mekanizması ile) ve aktarılmasında sorun yaşanan paketler komşulardan istenir ("pull" mekanizması ile). Şekil 3.6'da melez bir veri dağıtım yöntemi gösterilmiştir.



Şekil 3.6. Melez Veri Dağıtımı

3.2.2.3 Üst-Katman mimarisi

Video akışlandırma uygulamalarında üst-katman mimarisini geliştirmek için beş yöntemden bahsedilmiştir: merkezi, hiyerarşik, DHT tabanlı, kontrollü Flooding, "gossip" tabanlı (Yiu et al., 2007). Literatürde bulunan çalışmaların birçoğunda, "gossip" protokolü kullanılmakta ve açıklanmaktadır. Bu sebeple bu çalışma kapsamında üst-katman mimarisini oluşturma yöntemi olarak "gossip" protokolü seçilmiştir. Uygulamada "gossip" protokolü kullanılacağı için diğer yöntemlerin detayları verilmemiştir. Diğer yöntemlerle ilgili detaylı bilgi (Yiu et al., 2007) makalesinde bulunabilir.

Gossip Protokolü: Gossip protokolleri bilginin yayılması için kullanılan iletişim protokolleridir. Biyolojik popülasyonlarda viral enfeksiyonların yayılmasına benzer bir şekilde bilginin yayılmasına sebep oldukları için salgın ("epidemic") algoritmalar olarak da bilinirler (Kostadinova, 2008). Gossip protokolleri özellikle büyük ölçekli ve dinamik sistemlerde bilginin yayılması için kullanılırlar. Uygulaması kolay, güçlü ve hatalara karşı dayanıklıdır.

Video akışlandırma uygulamalarında gossip protokolleri, genel kullanımlarından farklı olarak kullanılırlar. Görevdeş ağlarda video akışlandırma sisteminde düğümler, bütün düğümlerin bilgisine değil kısmi bir bölümüne sahiptir. Genel kullanımdan farklı olarak bir düğüm, diğer düğümler arasından rastgele seçim yapmak yerine, elinde bulundurduğu kısmi görünüm ("partial view") içerisinden seçim yapar. Her düğüm sisteme girişinde belirlediği işbirlikçilerine gossip mesajları gönderir. Video akışlandırma sistemlerinde gossip

protokolleri üyelik yönetim bilgilerinin dağıtımı için kullanılır. Düğümler, ellerinde bulundurdıkları video parçalarını paylaşmak için gossip mesajlarını kullanırlar. Ayrıca gossip mesajları sayesinde düğümler sistemde aktif olduklarını işbirlikçilerine bildirirler ve ellerinde bulunan kısmi görünümüleri güncellerler. Görevdeş ağlarda en çok kullanılan gossip protokolleri şunlardır: SCAMP (Ganesh et al., 2002), Cyclon (Voulgaris et al., 2005) ve NeEm (Pereira et al., 2003).

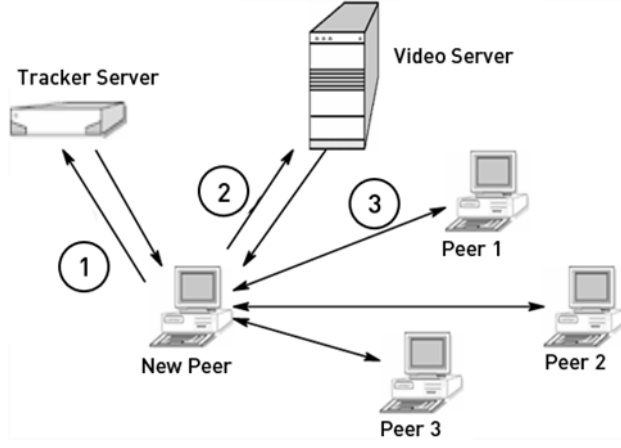
Gossip protokollerinin gerçekleşmesi için SCAMP'in video akışlandırma sistemlerinde kullanılan ve genişletilmiş bir versiyonu olan Ex-Scamp (Wei et al., 2011) harici başka bir protokolle karşılaşmadığı için bu çalışmada Ex-Scamp kullanılmıştır.

3.2.3 Görevdeş ağlarda video akışlandırmanın genel yapısı

Video akışlandırma uygulamalarındaki temel bileşenler şunlardır:

- Takipçi Sunucu (“ tracker server”): Sisteme yeni katılan düğümün (“peer”) ağa dahil olması için sistemde bulunan düğümlerin bir kısmının adresini gönderir.
- Video Sunucusu (“video server”): Video içeriğinin görevdeş ağa dağılmasını sağlayan sunucudur.
- Düğüm (“peer”): Sisteme bağlı olan ve video içeriğini görüntüleyip, birbirleriyle paylaşan düğümlerdir.

Bunun dışında farklı sistemlerde grafik sunucusu, versiyon sunucusu, web sunucusu, log sunucusu, yetkilendirme sunucusu gibi farklı bileşenler bulunmaktadır. Bu bileşenlerin sistemin işleyişinde yardımları olmakla birlikte, sistemin çalışması için zorunlu değildir. Görevdeş ağlarda video akışlandırma sisteminin genel yapısı Şekil 3.7’de gösterilmiştir.

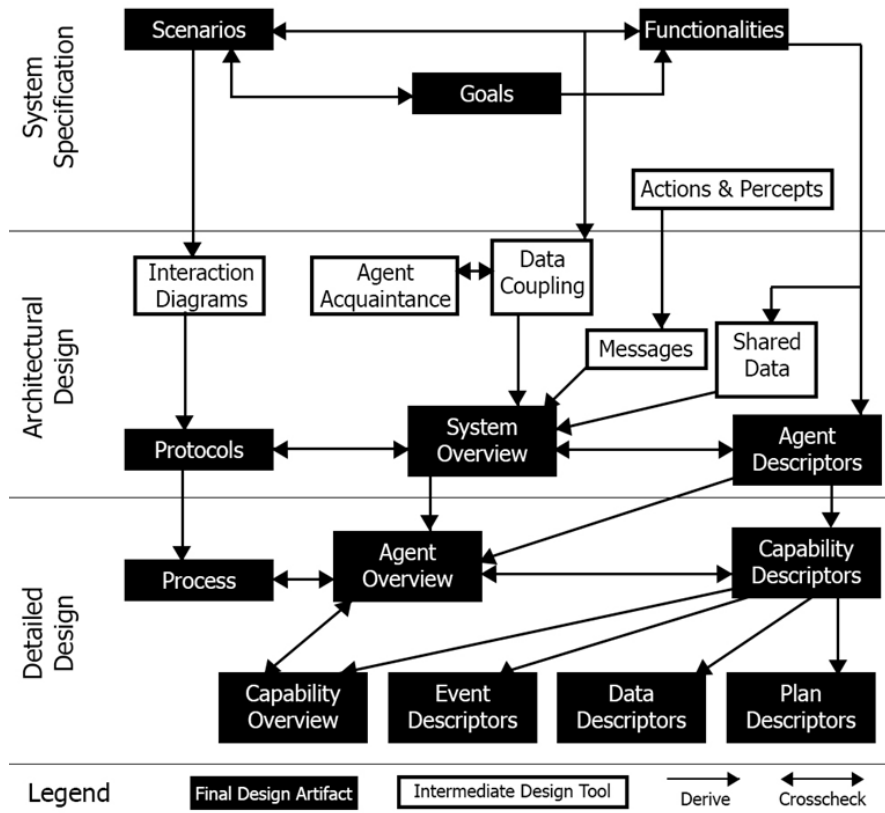


Şekil 3.7. Görevdeş Ağlarda Video Akışlandırmanın Genel Yapısı

4. ÇOK ETMENLİ SİSTEMİN PROMETHEUS İLE KANI-HEDEF-İSTEK MODELİNE UYGUN ANALİZ VE TASARIMI

Izgara tabanlı video akışlandırma sisteminin etmen-tabanlı tasarımı için Prometheus metodolojisini kullanılmıştır (Padgham ve Winikoff, 2003). Prometheus, çok etmenleri sistemleri tanımlama, tasarlama ve gerçekleştirme için geliştirilmiş bir metodolojidir.

Prometheus metodolojisinin 3 aşaması Şekil 4.1’de görülebilir.



Şekil 4.1. Prometheus Metodolojisinin Aşamaları (Padgham ve Winikoff, 2003)

Sistem tanımlamaları (“System Specification”): Amaçlar (“Goal”), sistemin temel işlevsellikleri (“Functionality”), senaryolar (“Scenario”), algılar (“Percept”) ve eylemler (“Action”) tanımlanır.

Mimari tasarım (“Architectural Design”): Sistem tanımlama aşamasında elde edilen verilerden yararlanarak etmen tipleri (“Agent Descriptors”) ve

etmenler arası etkileşimler (“Protocol”) ortaya konulur. Sistemin genel görünümü (“System Overview”) çıkarılır.

Detaylandırılmış tasarım (“Detailed Design”): Etmenlerin içyapısı tasarlanır (“Agent Overview”) ve her etmenin görevini nasıl tamamlayacağı belirlenir.

Prometheus metodolojisi, sistem tasarımını kolaylaştıran Prometheus Design Tool (PDT) (Padgham et al., 2008) adında bir araç desteğine sahiptir. PDT, Prometheus metodolojisi ile etmen tabanlı sistem geliştirmeyi sağlayan bir araçtır. İzgara tabanlı video akışlandırma sisteminin tasarımı PDT ile gerçekleştirilmiştir.

Bu bölümde ızgara tabanlı akışlandırma sisteminin (Teket et al., 2013) tasarımı tartışılmıştır. Önce sistem tanımlamaları aşaması verilmiş, sonra mimari tasarım aktarılmış ve son olarak detaylandırılmış tasarım anlatılmıştır.

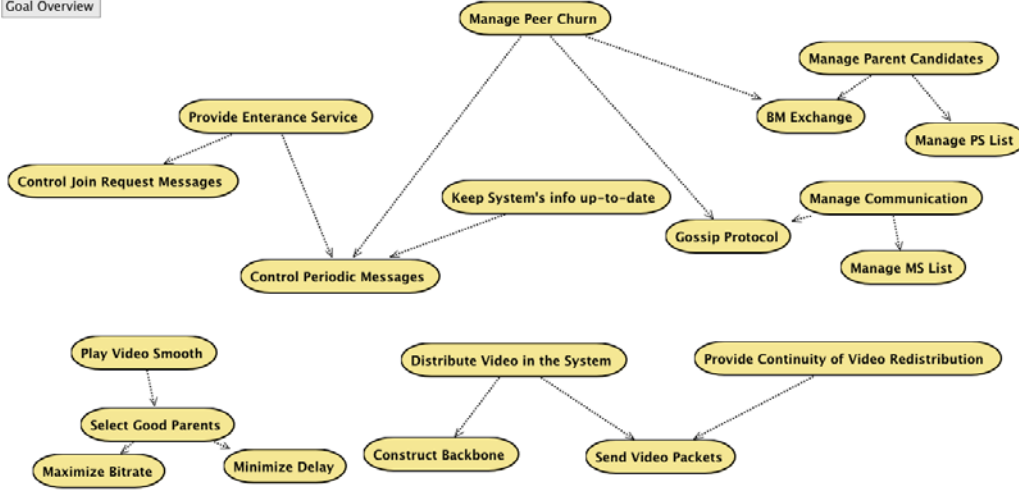
4.1 Sistem Tanımlamaları

Bu aşamada geliştirilmesi hedeflenen sistem için aşağıdaki adımlar uygulanır:

- Gollerin Tanımlanması
- Senaryoların Belirlenmesi
- İşlevselliklerin Belirlenmesi
- Algılar ve Eylemlerin Belirlenmesi

Şekil 4.2’de sistemin gol diyagramı verilmiştir. Hedefler, ilgili işlevsellikler ve senaryolar aşağıda açıklanmıştır. Tezin takip eden bölümlerinde anlam kayıplarının önlenmesi amacıyla tüm hedef, işlevsellik, senaryo, vb. isimlerinin orjinal İngilizce versiyonları kullanılmıştır.

Goal Overview



Şekil 4.2. Sistem Gol Diyagramı

Provide Entrance Service: Sisteme yeni katılan eşlerin video izleyebilmesi için sistemdeki diğer eşlerle iletişime geçmesi gereklidir. Bu sebeple sisteme yeni katılan eşler takipçi sunucuya (“tracker server”) katılım isteği (“join request”) gönderir ve takipçi sunucudan bir eş listesi elde eder.

Keep System’s Information Up-To-Date: Sistemi güncel tutabilmek için takipçi sunucu kalp atışı mesajları (“heartbeat message”) aracılığıyla eşlerin bilgilerini güncel tutar.

Distribute Video in the System: Canlı video akışlandırmanın doğası gereği video verisinin bir kaynak tarafından dağıtılması gerekmektedir. Bu sebeple, sistemde bir video sunucusu (“video server”) bulunur. Bu sunucu sağlam bir altyapı oluşturabilmek için güçlü eşleri kendisine çocuk olarak seçer.

Manage Communication: Üyelik listesi (“membership list”) sistemin kısmi görünümüdür ve her eşin bir üyelik listesi vardır. Eşler, işbirlikçi listesini oluşturabilmek için üyelik listesindeki eşlerle iletişime geçerler.

Manage Parent Candidates: Bir eşin işbirlikçi listesi (“partnership list”) vardır ve ebeveynlerini (“parent”) bu işbirlikçi listesinden seçer. Bunun için işbirlikçilerinin tampon harita (“buffermap”) bilgilerini kullanır.

Play Video Smooth: Bir eşin videoyu düzgün izleyebilmesi için video verisini gerekli zaman aralığında indirmesi gereklidir. Bunun için düzgün

ebeveynler seçmesi gerekir. Ebeveyn seçimi dışında video kalitesinin istenen düzeyde olmaması durumunda yeniden gönderim sistemi devreye girer ve ikincil ebeveynler seçilerek eksik video paketleri alınır.

Provide Continuity of Video Redistribution: Sistemdeki bütün eşler video verisini tamponlarında tutar ve gerektiğinde çocuklarına gönderir.

Manage Peer Churn: Bir eş, üyelik ve işbirlikçi listesini güncel tutmalıdır. Bunun için eşlerin sisteme giriş/çıkışlarını takip eder ve listelerini günceller.

4.2 Mimari Tasarım

Metodolojinin bu aşamasında planlar ile veri eşleşmeleri gerçekleştirilir. Ayrıca etmen tipleri belirlenerek planlar uygun etmenlere atanır. Sistem tanımlamalarında elde edilen veriler ile beraber bu aşamanın sonunda sistem genel bakış diyagramı oluşturulur. Mimari tasarım aşamasında takip edilecek adımlar aşağıdaki gibidir:

- Etmen Tiplerinin Belirlenmesi
- Etmenler Arası Etkileşimin, Etkileşim Diyagramı ve Protokolleri ile Gösterilmesi
- Sistem Yapısının Tasarlanması

Tez çalışmaları kapsamında sistemin tasarımında 3 tip etmen belirlenmiştir: Tracker, Video Server, Peer.

Etmenlerle işlevsellikleri aşağıdaki gibi eşleştirilmiştir:

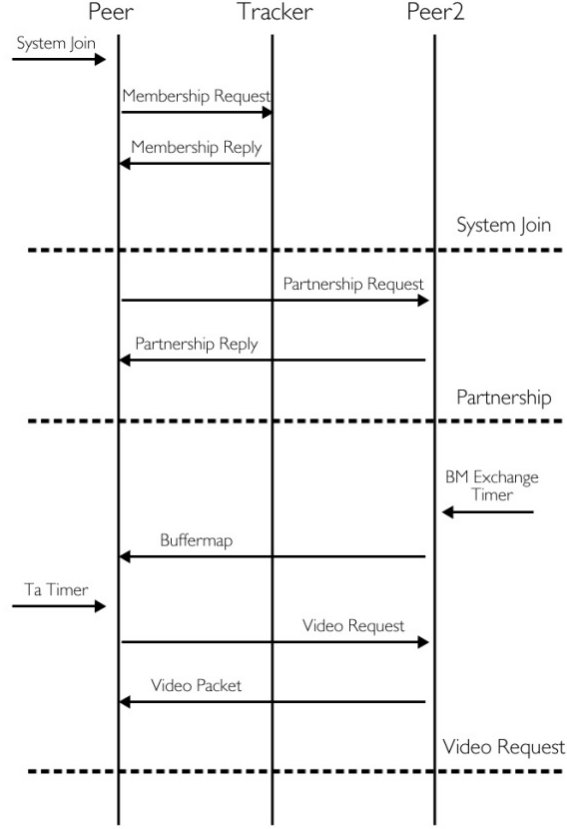
Tracker: Peer Maintenance Service, Peer Information Service

Video Server: Video Distribution Management

Peer: Communication Management, Partnership Management, Video Download Management, Video Redistribution Management, Peer Churn Management

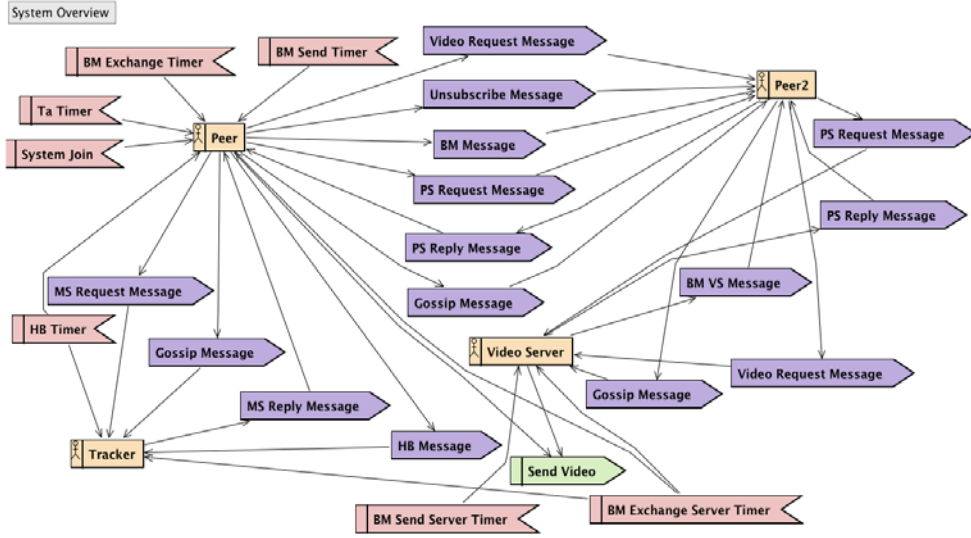
Ayrıca etmenler arası etkileşimler şu şekilde adlandırılmıştır: “System Join”, “Partnership”, “Buffermap Exchange”, “Video Request”, “Receive Video”, “Control Parent”, “Change Parent”, “Update Peer Lists”

“System Join”, “Partnership” ve “Video Request” etkileşim diyagramları Şekil 4.3’te görülebilir. Peer etmeni, “System Join” algısı ile sisteme giriş yapar. Tracker etmenine “Membership Request” gönderir. Tracker etmeni cevap olarak “Membership Reply” mesajını gönderir. “Membership Reply” mesajının içerisinde, üyelik listesi bulunur. Sisteme yeni katılan eş, üyelik listesini aldıktan sonra işbirlikçi listesini oluşturabilmek için üyelik listesindeki eşlere “Partnership Request” gönderir. İşbirlikçi seçimi karşılıklıdır, bu sebeple eğer “Partnership Request” alan eşin işbirlikçi listesinde boş yer varsa, bu isteği kabul eder ve yeni katılan eşe “Partnership Reply” mesajı gönderir. Bu sayede, yeni katılan eş işbirlikçi listesini oluşturur. Sistemdeki Peer etmenleri, periyodik olarak birbirlerine tampon harita bilgilerini gönderirler. Bu bilgileri kullanarak, Peer etmeni ebeveynlerini seçer ve video izlemek için video verisini indirmeye başlar. Ayrıca ebeveynlerinin performansını periyodik olarak kontrol eder, gerekli durumlarda ebeveynini değiştirir.



Şekil 4.3. “System Join”, “Partnership” “Video Request” Etkileşim Diyagramları

Şekil 4.4’te sistem genel bakış diyagramı verilmiştir. Bu diyagram, çok etmenli sistem tasarımının en önemli çıktılarından birisidir. Sistem genel bakış diyagramında, etmen tipleri, algılar, eylemler, mesajlar ve etmenler arası etkileşimler görülebilir. Daha önce bahsedilmiş olan “Buffermap Exchange”, “Receive Video”, “Control Parent”, “Change Parent”, “Update Peer Lists” etkileşimleri bu diyagram üzerinden takip edilebilir.



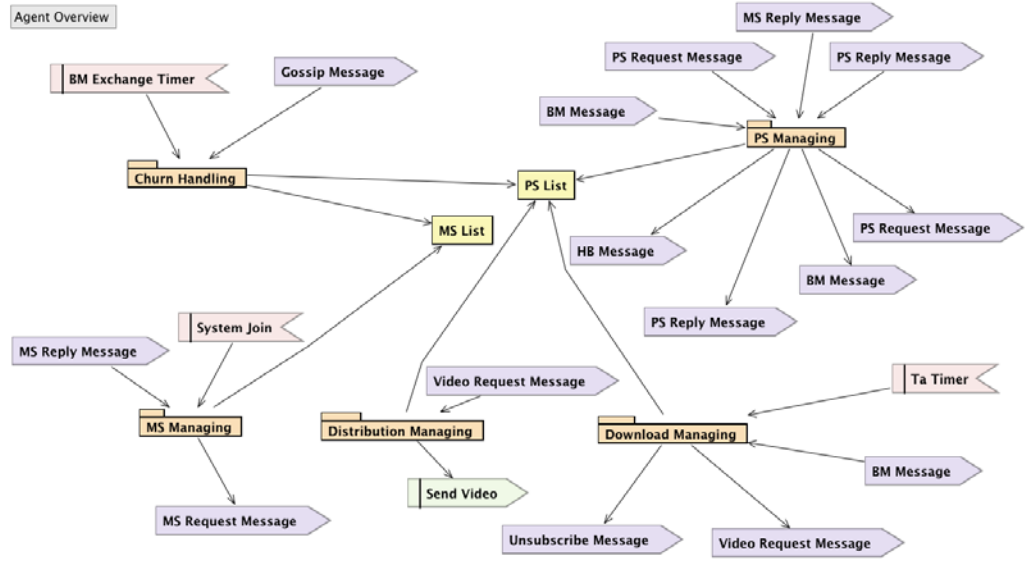
Şekil 4.4. Sistem Genel Bakış Diyagramı

4.3 Detaylandırılmış Tasarım

Detaylandırılmış tasarımdaki adımlar şu şekilde özetlenebilir:

- Etmen yetenekleri genel bakış diyagramlarının hazırlanması
- Olayların ve mesajların oluşturulması
- Algı ve eylem detaylarının belirlenmesi

Şekil 4.5'te Peer etmeninin genel bakış diyagramı verilmiştir. Peer etmeninin 5 adet yeteneği ("capability") bulunmaktadır:



Şekil 4.5. Peer Etmen Genel Bakış Diyagramı

MS Managing: Bu yetenekte, Peer etmeni üyelik listesini yönetir. Eğer sisteme yeni katılan bir düğümse, üyelik listesini oluşturur. Gerekli durumlarda üyelik listesini günceller.

PS Managing: PS Managing yeteneğinde, Peer etmeni işbirlikçi listesini düzenler. İşbirlikçi listesini oluşturur; yer ve yükleme bant genişliği bilgilerine göre işbirlikçi listesini günceller. Ayrıca tampon harita değişimi ve kalp atışı mesajlarından sorumludur.

Churn Handling: Bu yetenekte, Peer etmeni periyodik olarak diğer eşlerin sistemde olma durumunu kontrol eder. Duruma göre, üyelik ve işbirlikçi listelerini günceller. Listelerini Gossip mesaj içeriğine göre de düzenler.

Download Managing: Bu yetenekte, Peer etmeni ebeveynlerini düzenler. Tampon harita değişimi ile işbirlikçilerinin bilgilerini periyodik olarak günceller. Ayrıca ebeveynlerinin bit hızı, gecikme gibi bilgilerini de değerlendirme için saklar ve günceller. Aşağıda Ebeveyn Puanlama davranışına ait zahiri kod ("pseudocode") verilmiştir:

```

01 A := IP Address of Peer
02 tempParent := parent of A
03 tempBitrate := stream bitrate of A
04 tempDelay := end-to-end delay from tempParent to A
05 tempScore := score of parent
06 if(tempBitrate > bitrateThreshold)
07   if(tempDelay < delayThreshold)
08     tempscore = 4
09   else
10     tempscore = 3
11 else
12   if(tempDelay < delayThreshold)
13     tempscore = 2
14   else
15     tempscore = 1

```

Ebeveyn Puanlama Davranışı

Görüldüğü gibi Ebeveyn Puanlama davranışı ile Peer etmeni bit hızı ve gecikme zamanlarına göre ebeveynlerine puan verir. Yüksek bit hızı ve düşük gecikme zamanı daha çok puan demektir. Ebeveyn puanları toplanarak artar. Örneğin, ebeveynin bit hızı bitrateThreshold'un üzerinde ise ve gecikme hızı delayThreshold'dan düşük ise 4 puan alır (satır 6-8). Eğer ebeveynin bit hızı bitrateThreshold'un üzerinde ise fakat gecikme hızı delayThreshold'dan yüksek ise 3 puan alır (satır 9-10).

“MS List” üyelerin listesidir. Üyeler potansiyel ortaklardır, ama Peer etmeni üyelerle iletişime geçmek zorunda değildir. MS List sistemin kısmi görünümü olarak da adlandırılabilir. Aşağıda MS List'in yapısı verilmiştir:

PeerID	IPAddr
--------	--------

PeerID: Bu bölüm Tracker Server etmeni tarafından verilen özel bir sayıdır.

IPAddr: Bu alan listedeki üyelerin IP adreslerini tutar.

“PS List” ortakların listesidir. Ortaklık karşılıklıdır. Peer etmeni ortaklarıyla periyodik olarak haberleşir. Bu sayede ortaklarının elinde bulundurduğu video paketlerini öğrenir ve sistemde olma durumlarını günceller Ayrıca PS List’te, ortaklarının puanlarını tutar. Aşağıda PS List’in yapısı verilmiştir:

PeerID	IPAddr	SameISP	Buffermap	ParentScore	ParentInfo	ChildInfo	ChildBM
--------	--------	---------	-----------	-------------	------------	-----------	---------

PeerID: Bu bölüm Tracker Server etmeni tarafından verilen özel bir sayıdır.

IPAddr: Bu alan listedeki üyelerin IP adreslerini tutar.

SameISP: Bu alan ISP bilgisi için ayrılmıştır. Eğer ortağıyla Peer etmeni aynı ISP de ise bu alanın değeri “true” olarak tanımlanır.

Buffermap: Bu alan 4 tane tam sayı değeri içeren bir diziden oluşur. Ortakların tampon harita bilgileri burada tutulur.

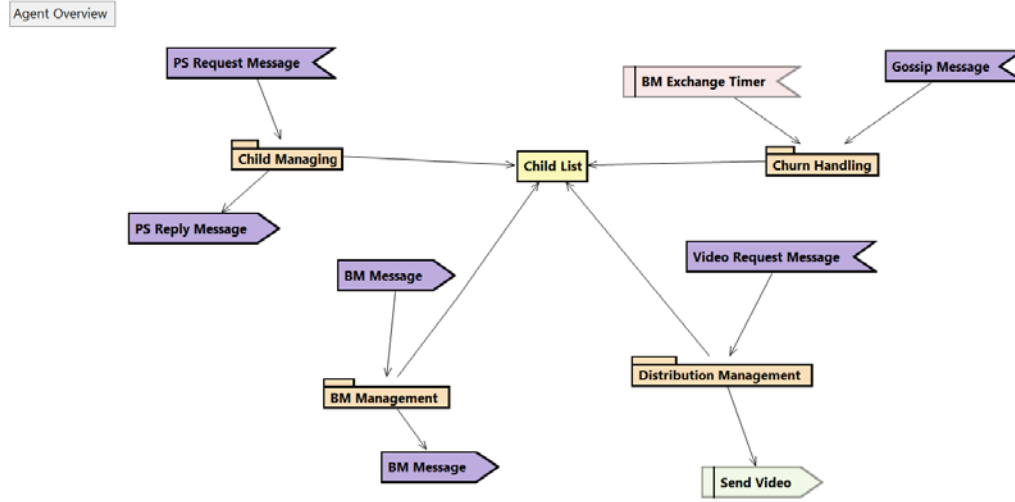
ParentScore: Bu alanda ortakların Ebeveyn Puanlama algoritması ile aldıkları puanlar tutulur. İlk değer olarak 0 tanımlanmıştır. Puanlamalar eklemeli olarak artar.

ParentInfo: Bu alan 4 tane mantıksal değer içerir. Altakışlar hakkında ebeveyn bilgilerini tutar. Eğer “true” olarak tanımlıysa, Peer etmeni ilgili altakışı ortağından alıyor demektir.

ChildInfo: Bu alan 4 tane mantıksal değer içerir. Altakışlar hakkında çocuk bilgilerini tutar. Eğer “true” olarak tanımlıysa, Peer etmeni ilgili altakışı bu ortağına gönderiyor demektir.

ChildBuffermap: Bu alan 4 tane tam sayı değeri içeren bir diziden oluşur. Çocuklara gönderilecek paket numaraları burada saklanır.

Şekil 4.6’da Video Server etmeninin genel bakış diyagramı verilmiştir. Video Server etmeninin 4 adet yeteneği (“capability”) bulunmaktadır:



Şekil 4.6. Video Server Etmeni Genel Bakış Diyagramı

Child Managing: Child Managing yeteneğinde, Video Server etmeni işbirlikçi listesini düzenler ve dağıtım yapacağı eşleri seçer. Çocuk listesini oluşturur; yer ve yükleme bant genişliği bilgilerine göre çocuk listesini günceller. Aşağıda Çocuk Yönetme davranışı verilmiştir:

```

01 A := IP Address of Requester
02 tempISP := peer_ISP
03 if((ISP_Slot[tempISP]<Max_ISP_Slot))
04   if(Upload_Capacity[A]>Min_Upload_Capacity)
05     add A to ISP_Slot[tempISP]
  
```

Çocuk Yönetme Davranışı

Çocuk Yönetme davranışında, Video Server etmeni çocuklarını ISP'ler için ayırdığı tampona (sıra 3) ve başvuran düğümlerin bant genişliğine göre belirler (sıra 4). Her ISP için sınırlı sayıda tamponda yeri bulunur (Max_ISP_Slot). Video Server etmeni ayrıca çocuklarının bant genişliklerini kontrol eder (Min_Upload_Capacity). Eğer bir düğüm bu şartları yerine getirirse Video Server etmeni o düğümü çocuğu olarak belirler ve video gönderime başlar.

Churn Handling: Bu yetenekte, Video Server etmeni periyodik olarak diğer eşlerin sistemde olma durumunu kontrol eder. Duruma göre, çocuk listesini günceller. Listelerini Gossip mesaj içeriğine göre de düzenler.

BM Managing: Bu yetenekte, Video Server etmeni tampon harita değişimlerini düzenler. Video dağıttığı eşlerin geri kalmamasını sağlar ve elinde bulundurduğu video veri bilgisini çocuklarıyla paylaşır.

Distribution Managing: Bu yetenekte, Video Server etmeni video verisinin dağıtımını yönetir.

“Child List” Video Server etmeninin çocuk listesidir. Video Server etmeni çocuklarıyla periyodik olarak haberleşir ve video akışı ile ilgili bilgiler gönderir. Child List yapısı aşağıda verilmiştir:

PeerID	IPAddr	UploadCapacity	ChildInfo	ChildBM
--------	--------	----------------	-----------	---------

PeerID: Bu bölüm Tracker Server etmeni tarafından verilen özel bir sayıdır.

IPAddr: Bu alan listedeki üyelerin IP adreslerini tutar.

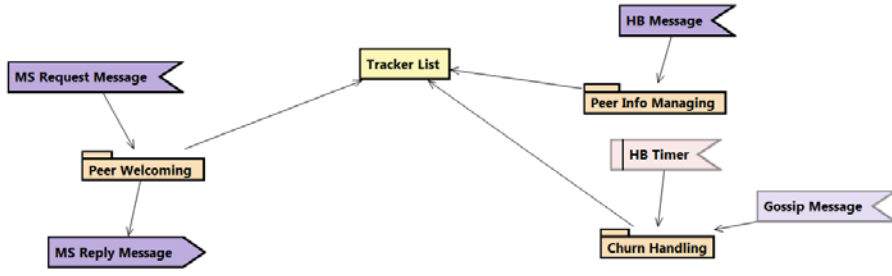
UploadCapacity: Bu alan çocukların gönderme kapasitelerini tutar.

ChildInfo: Bu alan 4 tane mantıksal değer içerir. Altakışlar hakkında çocuk bilgilerini tutar. Eğer “true” olarak tanımlıysa, Peer etmeni ilgili altakışı bu ortağına gönderiyor demektir.

ChildBuffermap: Bu alan 4 tane tam sayı değeri içeren bir diziden oluşur. Çocuklara gönderilecek paket numaraları burada saklanır.

Şekil 4.7’de Tracker etmeninin genel bakış diyagramı verilmiştir. Tracker etmeninin 3 adet yeteneği (“capability”) bulunmaktadır:

Agent Overview



Şekil 4.7. Tracker Server Etmeni Genel Bakış Diyagramı

Peer Welcoming: Bu yetenekte, Tracker etmeni sisteme yeni giriş yapan eşleri kontrol eder. Sisteme yeni giren eşlere uygun liste döndürülmesinden sorumludur. Aşağıda Yeni Eş Karşılama davranışı verilmiştir:

```

01 A := IP Address of Requester
02 tempISP := ISP of Requester
03 TrackerList[] := all online peers
04 sameISPHighUCList:= List for peers with same ISP and high upload capacity
05 sameISPLowUCList:= List for peers with same ISP and low upload capacity
06 differentISPList:= List for peers with different ISP
07 membershipList := Membership List to be sent to A
08 for (i:= 0; i<TrackerList.size();i++)
09   peerISP := ISP of TrackerList[i]
10   peerUploadCapacity := Upload Capacity of TrackerList[i]
11   if(peerISP == tempISP)
12     if (peerUploadCapacity >= Upload_Capacity_Threshold)
13       add TrackerList[i] to sameISPHighUCList
14     else
15       add TrackerList[i] to sameISPLowUCList
16   else
17     add TrackerList[i] to differentISPList
18 if(sameISPHighUCList.size() >= membershipList.size())
19   select peers randomly from sameISPHighUCList and add to membershipList
20 else
21   add all peers in the sameISPHighUCList to membershipList
22 if(sameISPLowUCList.size()>=(membershipList.size()- sameISPHighUCList.size()))
23   select peers randomly from sameISPLowUCList and add to membershipList
24 else
25   add all peers in the sameISPHighUCList to membershipList
26   if(differentISPList.size()>=(membershipList.size()- (sameISPHighUCList.size() +
sameISPLowUCList)))
27     select peers randomly from differentISPList and add to membershipList

```

Yeni Eş Karşılama Davranışı

Satır 8 ve satır 17 arasında Tracker Server etmeni sistemdeki düğümleri gruplar. Satır 8’de Tracker Server etmeninin sistemdeki bütün düğümleri kontrol edeceği belirtilmektedir. Satır 11’de Tracker Server etmeni sisteme yeni katılan düğümle seçili düğümün aynı ISP’de olup olmadığını kontrol eder. Eğer aynı ISP’de iseler ve bant genişliği Upload_Capacity_Threshold’dan büyük ise Tracker Server seçili düğümü sameISPHighUCList’e ekler (satır 13). Eğer bant genişliği Upload_Capacity_Threshold’dan küçük ise sameISPLowUCList’e ekler (satır 15). Eğer sisteme yeni giren düğüm ile seçili düğümün ISP’leri farklı ise, seçili düğüm differentISPList’e eklenir (satır17). Seçim aşaması satır 18’de başlar. Eğer

sameISPHighUCList'teki düğüm sayısı membership listesinin boyutundan büyük ise, Tracker Server etmeni sameISPHighUCList'ten rastgele düğüm seçerek sisteme yeni katılan düğüme membership listesini gönderir (satır 18-19). Eğer sameISPHighUCList'teki düğüm sayısı membership listesinin boyutundan küçük ise, Tracker Server etmeni sameISPHighUCList'teki bütün düğümleri membership listesine ekler (satır 21) ve sameISPLowUCList'i kontrol etmeye başlar. Eğer sameISPLowUCList'teki düğüm sayısı membership listesinde kalan yerden fazla ise, Tracker Server etmeni sameISPLowUCList'ten rastgele düğüm seçerek sisteme yeni katılan düğüme membership listesini gönderir (satır 22-23). Eğer bu süreçten sonra eğer membership listesi hala dolu değil ise, Tracker Server etmeni sameISPLowUCList'teki bütün düğümleri membership listesine ekler (satır 25) ve differentISPList'ten rastgele düğümler seçerek sisteme yeni katılan düğüme gönderir.

Peer Info Managing: Peer Info Managing yeteneğinde, Tracker etmeni sistemdeki eşlerin konum, paylaşım durumu gibi bilgilerini yönetir. Eşler bu bilgileri kalp atışı mesajının içerisinde gönderirler.

Churn Handling: Bu yetenekte, Tracker etmeni periyodik olarak diğer eşlerin sistemde olma durumunu kontrol eder. Bunu kalp atışı sayacına göre yapar. Duruma göre, tracker listesini günceller. Listelerini Gossip mesaj içeriğine göre de düzenler.

“Tracker List” sistemde bulunan düğümlerin listesidir. Peer etmeni periyodik olarak Tracker Server etmeni ile haberleşir. Bu sayede sistemde olduğu bilgisini göndermiş olur. Aşağıda Child List yapısı verilmiştir:

PeerID	IPAddr	UploadCapacity	ISPNo
--------	--------	----------------	-------

PeerID: Bu bölüm Tracker Server etmeni tarafından verilen özel bir sayıdır.

IPAddr: Bu alan listedeki üyelerin IP adreslerini tutar.

UploadCapacity: Bu alan çocukların gönderme kapasitelerini tutar.

ISPNo: Bu alanda düğümün hangi ISP'de bulunduğu bilgisi tutulur.

Ayrıca, yazılım etmenleri tasarımında ve uygulamasında Kanı-İstek-Hedef (KİH) mimarisi göz önünde bulundurulmuştur. Bölüm 3.1.1.'de anlatıldığı gibi KİH mimarisinde, bir etmen hangi golleri seçeceğini ve bu gollere nasıl ulaşacağını belirler. Kanı, etmenin çevresi hakkında sahip olduğu bilgileri, istek, etmenin gerçekleştirmek istediği golleri, Hedef ise etmenin çıkarsama ve plan uygulama mekanizmalarını gösterir. Aşağıda bazı goller ve bu gollerin KİH mimarisine göre nasıl gerçekleştirilebileceği anlatılmıştır.

“Play Video Smooth” golünün gerçekleştirilmesi için Peer etmeni “PS Managing” ve “Download Managing” yeteneklerini kullanır. Peer etmeni Buffermap mesajları ile periyodik olarak PS List’i (ortak listesi) günceller ve her 10 saniyede bir bu bilgileri kullanarak ebeveynlerini kontrol eder. Eğer ebeveyni ile bir sorun olursa (Ebeveynin sistemden çıkması, video akışlandırmanın yavaş olması vs.), Peer etmeni PS List’te bulunan “Buffermap”, “ParentScore” ve “SameISP” alanlarını kullanarak yeni bir ebeveyn seçer.

“Keep System’s Information Up-To-Date” golünde, Tracker Server etmeni “Peer Info Managing”, “Churn Handling” ve “Peer Welcoming” yeteneklerini kullanır. Sistemde video izlemekte olan Peer etmenleri her 5 saniyede bir Tracker Server etmenine Heartbeat mesajı göndermelidir. Eğer Tracker Server etmeni 5 saniyelik periyotta Peer etmeninden (örneğin 34 numaralı düğüm) Heartbeat mesajı almazsa, “Heartbeat Timer” “Churn Handling yeteneğini çalıştırır” ve Tracker Server etmeni 34 numaralı düğümü listesinde siler. Bu 34 numaralı düğümün sistemden çıktığını gösterir.

Video Server etmeni “Distribute Video in the System” golünü gerçekleştirmeye çalışırken, “Child Managing” ve “Distribution Managing” yeteneklerini kullanır. Çocuklarını Çocuk Yönetme davranışına göre seçer ve video paketlerini sisteme dağıtır.

5. NS-3’TE UYGULAMA

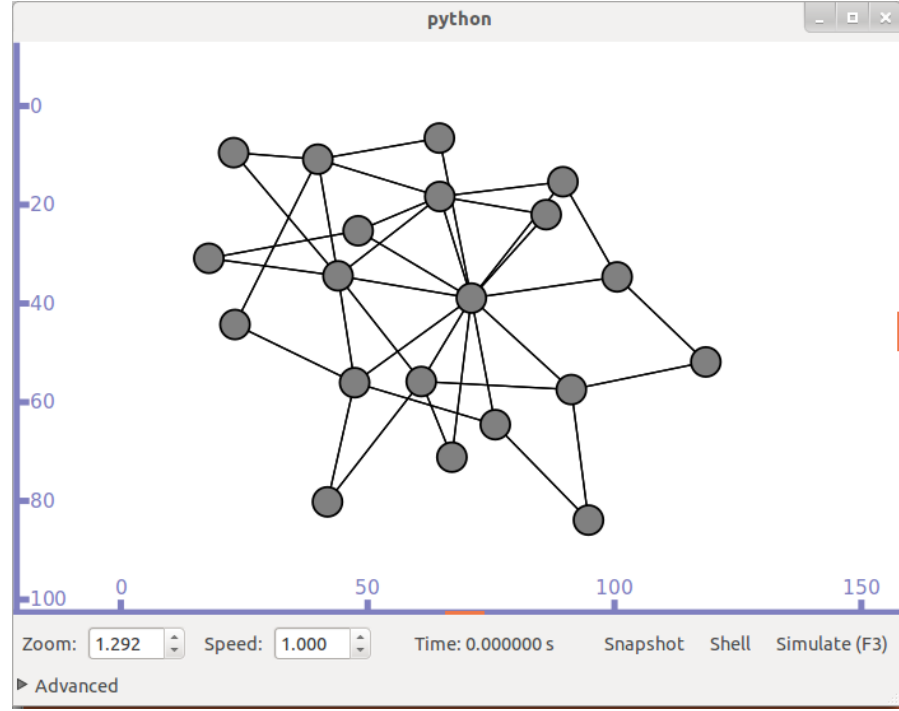
Bu bölümde, CoolStreaming benzeri sistemin ve önerilen Çok-Etmenli Sistemin ns-3 ortamında geliştirilmesi anlatılmıştır. ns-3 kesikli-olay (“discrete event”) network simülatörüdür. Yaygın olarak araştırmalar ve eğitim için kullanılmaktadır. ns-3 altyapısı C++’da kodlanmıştır. Bu çalışmada bahsedilen sistemler de ns-3 ve C++ ile geliştirilmiştir. Kullanılan ns-3 altyapısı 111E022 no’lu TÜBİTAK destekli proje kapsamında bu tezin yazarının da dahil olduğu bir çalışma grubu tarafından geliştirilmiştir. Uygulanan ızgara tabanlı akışlandırma sisteminde, *CoolStreaming* temel alınmıştır. Aşağıdaki konular sırasıyla anlatılmıştır:

- Ağ topolojisinin oluşturulması
- Sistemdeki düğümler ve özellikleri
- Düğümler arası iletişim
- Izgara tabanlı akışlandırma sisteminde video gönderimi

5.1 Ağ Topolojisinin Oluşturulması

Sistem tasarımı için öncelikle ns-3 simülatör ortamında her düğümün aynı zamanda bir yönlendirici görevi üstlendiği 300 düğümlük bir fiziksel topoloji oluşturulmuştur.

Topoloji, NS-3’ye entegre edilen *BRITE* topoloji üreticisi ve *RouterBarabasiAlbert* rastgele topoloji üretim modeli (Barabasi ve Albert., 1999) kullanılarak oluşturulmuştur. Üretilen tüm topolojiler bağlı (“connected”) olup her iki düğüm arasında mutlaka bir yol mevcuttur. Topolojideki her düğüm, komşuları ile uçtan-uca bağlantılar (“point-to-point links”) üzerinden bağlıdır. Bu bağlantılardaki bantgenişliği değerleri ise 1 Gbps olarak belirlenmiştir. Örnek olarak oluşturulan 20 düğümlük fiziksel topoloji Şekil 5.1’de görülebilir.



Şekil 5.1. BRITE ile Üretilmiş Örnek Topoloji

5.2 Sistemdeki Düzgümler ve Özellikleri

Oluşturulan video akışlandırma sisteminde 3 tip düğüm bulunmaktadır: takipçi sunucu (“tracker server”), kaynak sunucu (“video server”) ve eş (“peer”).

Fiziksel topoloji oluşturulduktan sonra bu topoloji üzerinde video akışlandırmanın gerçekleştirileceği bir üst katman ağı oluşturulması gerekmektedir. Bu amaç doğrultusunda üst-ağda bulunan düğümlerin ve her düğümde bulunan video parçalarının listesini barındıran; yeni düğümlerin üst ağa katılımını sağlamak amacıyla onlara üyelik (“*membership*”) listesi gönderen bir takipçi sunucu bulunmalıdır. Tezde uygulanan takipçi sunucu mekanizması, *BitTorrent* (Cohen, 2008) sisteminden yola çıkılarak geliştirilmiştir.

Takipçi sunucu, topolojideki her düğüm tarafından bilinen bir düğümdür (Bittorrent'de olduğu gibi takipçi sunucular, genellikle belirli web siteleri aracılığıyla (torrent siteleri) adresleri kolaylıkla edinilebilecek düğümlerdir. Bu çalışma kapsamında gerçekleştirilen uygulamada takipçi sunucunun, sistemdeki her düğüm tarafından bilindiği kabul edilmiştir).

Takipçi sunucunun, uygulanan sistemdeki görevleri şu şekilde sıralanabilir:

- Sisteme yeni katılacak düğümlere üyelik listesi göndermek,
- Sisteme yeni katılan düğümleri üyelik listesine eklemek,
- Sistemden ayrılan düğümleri tespit ederek ayrılan düğümleri listesinden silmek.

Kaynak sunucu, video yayını gerçekleştiren sunucudur. Kendisini ebeveyn (“parent”) olarak seçen düğümlere istekleri doğrultusunda video paketleri gönderir. Oluşturulan video akışlandırma sisteminde, kaynak sunucuya bağlanabilecek düğüm sayısı sistemdeki düğüm sayısının %20’si ile kısıtlanmıştır. Kaynak sunucunun karşıya yükleme (“upload”) bant genişliği 10 Mbps olarak belirlenmiştir. Akışlandırma sistemi canlı yayın (“live streaming”) yaptığı için her düğüme video gönderim hızı sabit ve 292 Kbps’dır. Kaynak sunucuya bağlanabilecek düğüm sayısı, kaynak sunucunun bant genişliği ile video gönderim hızı değerleri parametrik olup bu değerler istenildiği takdirde değiştirilebilmektedir.

Eşler sisteme video izleme amacıyla bağlanan düğümlerdir. Video paketi aldıkları gibi, kendilerini ebeveyn seçen diğer eşlere bu video paketlerini gönderirler. Eşler, oluşturulan video akışlandırma sistemine 5 milisaniye aralıklarla (istenildiği gibi değiştirilebilir) giriş yapmaktadırlar. Her bir eş, üstel dağılıma göre belirlenen sürelerle sisteme bağlanır ve sistemden ayrılırlar (“peer churn”). Bu süreler MATLAB programında yazılmış üstel dağılım fonksiyonu ile oluşturulmuştur ve eşler bir metin dosyasından kendileriyle ilgili sisteme giriş çıkış verilerini okuyarak sisteme bağlanır ve sistemden ayrılırlar. Sisteme bağlı olan her eşin bir üyelik (“membership”) listesi ve üyelik listesi arasından seçilen bir işbirlikçi (“partner”) listesi bulunmaktadır. Üyelik listesi, sisteme bağlı tüm düğümlerin bir alt kümesidir ve sistemdeki her düğüm, sisteme giriş anında takipçi sunucudan bu listenin ilk halini edinir. Herhangi bir eşin, üyelik listesindeki bütün eşlerle haberleşmesi zorunlu değildir. Öte yandan sistemdeki eşler, işbirlikçi listesinde bulunan eşlerle periyodik olarak haberleşerek sahip oldukları bilgileri (tampon harita bilgisi (“buffermap”) ve çıkış yapan eş listesi gibi) kontrol mesajları kullanarak paylaşır.

5.3 D ğ mler Arası İletişim

Bu b l mde d ğ mlerin birbirleriyle haberleşmek i in kullandıkları y ntemler anlatılmıştır.

5.3.1 Eşin sisteme bağlanması

Bir eşin sisteme girmek i in ger ekleştirdiğı ilk mesajlaşma takip i sunucu ile yaşanmaktadır. Sisteme dahil olacak eşin sistemdeki diğ r eşlerden haberdar olabilmesi i in takip i sunucuyla iletişime ge mesi gereklidir. Aşağıda a ıklanan MEMBERSHIP_REQUEST ve MEMBERSHIP_REPLY mesajları aracılığıyla sisteme bağlanan eş, sistemdeki diğ r b t n eşlerin bilgilerini değıl ama t m eşlerin bir alt k mesini ( yelik listesi – “membership list”) elde eder.

MEMBERSHIP_REQUEST: Sisteme yeni katılan bir eşin ilk g nderdiği mesaj tipidir. Bu mesaj, takip i sunucuya bir eşin sisteme katıldığını bildirmek ve takip i sunucudan bir  yelik listesi almak i in kullanılır.

MEMBERSHIP_REPLY: Bu mesaj, MEMBERSHIP_REQUEST mesajına cevap olarak takip i sunucu tarafından g nderilir. Takip i sunucu bu mesajın i eriğinde  yelik listesi bilgisini (takip i sunucu tarafından, aktif olan t m eşlerin bulunduğı liste i erisinden rastgele se im ile belirlenir) ve sisteme giriş yapmak isteyen eşin bu oturum i in kullanacağı zaman damgasını sisteme yeni katılan eşe g nderir.

Şekil 5.2’de 23 numaralı eşin sisteme bağlanması g sterilmektedir.



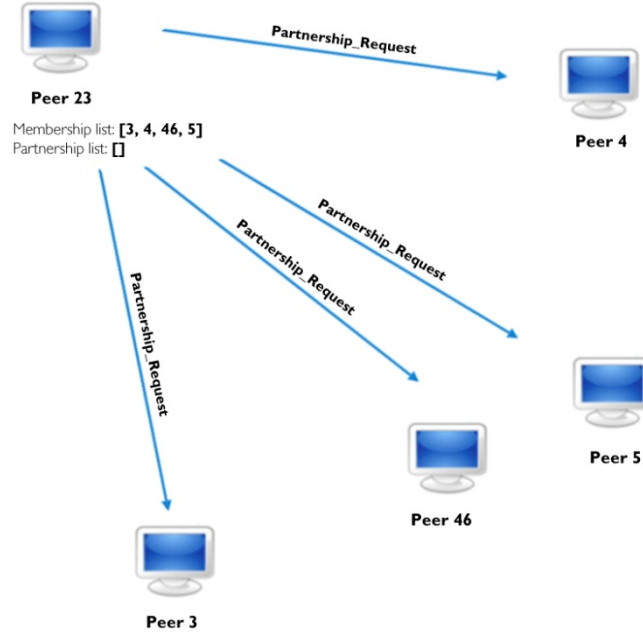
Şekil 5.2. Eşin Sisteme Bağlanması

5.3.2 Eşin işbirlikçilerini belirlemesi

Bir eş sistemle ilgili kısmi görünümü (üyelik listesini) elde ettikten sonra işbirlikçi arayışına girer. İşbirlikçiler, üyelik listesi içerisinde rastgele seçim ile belirlenir. İşbirlikçilerini belirleyen eş, dedikodu (“gossip”) protokolü aracılığıyla işbirlikçilerinin sahip olduğu video paket durumlarını öğrenerek belirlediği işbirlikçilerden video isteğinde bulunur. Eğer işbirlikçi sayısı belirli bir sayının altındaysa veya işbirlikçilerinden birisi sistemden ayrılırsa, eş yeni işbirlikçi arayışına girer. Aşağıda açıklanan PARTNERSHIP_REQUEST, PARTNERSHIP_REPLY, PARTNERSHIP_ACK VE PARTNERSHIP_NACK mesajları ile işbirlikçi seçimi gerçekleştirilir.

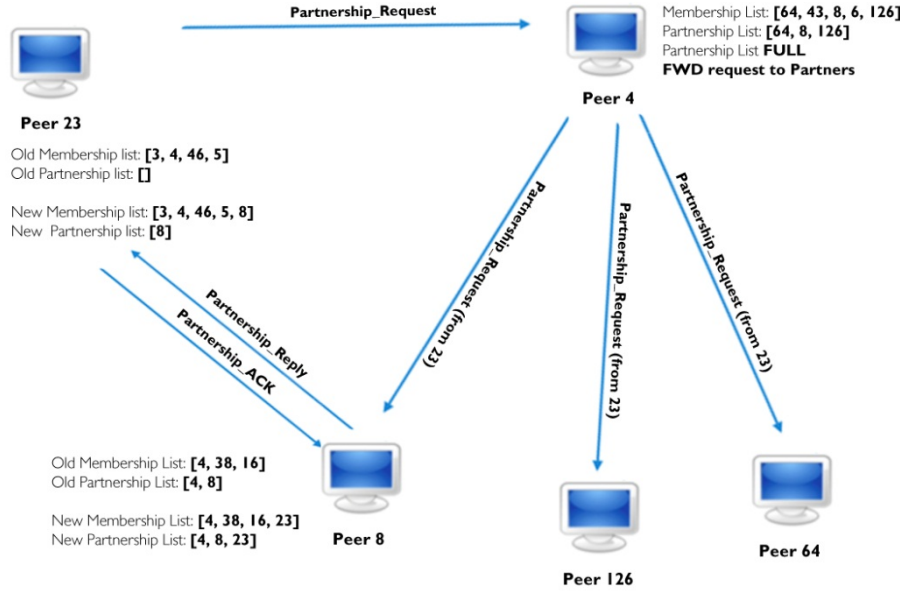
PARTNERSHIP_REQUEST: Sisteme yeni katılan eş, takipçi sunucudan üyelik listesi ve zaman damgası aldıktan sonra video alabilmek için işbirlikçi aramaya başlar. Bu amaçla üyelik listesinde bulunan eşlere PARTNERSHIP_REQUEST mesajı gönderir (Şekil 5.3). Bu mesajın içeriğinde dedikodu protokolü için kaç atlama (“hop count”) yapılacağı ve kendi zaman damgası bulunur. Eğer PARTNERSHIP_REQUEST mesajı alan düğümün üyelik listesinde yer yoksa, mesajı alan düğüm atlama sayısını bir azaltarak bu mesajı kendi işbirlikçilerine yönlendirir (Şekil 5.4). Mesaj içerisindeki göndericiye ait zaman damgasının, mesajı alan düğümde, göndericiden daha önceden alınmış olan zaman damgasından eski olması durumunda mesaj göz ardı edilir.

Şekil 5.3 ve Şekil 5.4’te maksimum üyelik listesinin boyutu 5, işbirlikçi listesinin boyutu 3 olarak belirlenmiştir (bu değerler şekiller için keyfi olarak belirlenmiş olup, sistemde kullanılan liste boyutları farklılık göstermektedir).



Şekil 5.3. Eşin İşbirlikçilerini Belirlemesi

Şekil 5.4'te 23 numaralı eş, üyelik listesindeki 4 numaralı eşe PARTNERSHIP_REQUEST mesajı gönderir. 4 numaralı eşin işbirlikçi listesinde yer olmadığı için mesajın içeriğindeki atlama sayısını bir azaltarak bu mesajı kendi işbirlikçilerine gönderir. Atlama sayısı "0" olana kadar bu yönlendirmeler devam eder. Oluşturulan sistemde, atlama sayısı 2 olarak belirlenmiştir.



Şekil 5.4. İşbirlikçi Listesi Dolu Olan Bir Eşin İsteği Yönlendirmesi

PARTNERSHIP_REPLY: Bu mesaj, PARTNERSHIP_REQUEST mesajına cevap olarak bir eş tarafından ya da kaynak sunucu tarafından gönderilir. Eğer isteği alan düğümün işbirlikçi listesinde boş yer varsa mesajı gönderen eş, alıcının işbirlikçi listesine zaman damgası ile birlikte eklenir. Bunun üzerine alıcı, yeni işbirlikçisine kendi zaman damgasını içeren PARTNERSHIP_REPLY mesajı gönderir. Alıcının işbirlikçi listesinde yer olmaması durumunda ise alıcı, göndericiye yanıt göndermez ve gönderici bu durumu olumsuz yanıt (“negative acknowledgement” - NACK) olarak algılar ve işbirlikçi listesine alıcı düğümü eklemeyiz. Eğer alıcı tarafında işbirlikçi olma isteği olumlu karşılanırsa (gönderici, alıcıdan PARTNERSHIP_ACK mesajı alırsa), gönderici düğüm, tampon haritaların değişimini (“buffermap exchange”) düzenli aralıklarla gerçekleştirmek için zamanlayıcısını başlatır. Bu zamanlayıcı sayesinde 2 saniyede bir BUFFERMAP mesajları işbirlikçiler arasında değiştirilir. Burada 2 saniye olarak belirlenmiş BUFFERMAP mesajı gönderim sıklığı parametrik olup istenildiği gibi değiştirilebilmektedir.

PARTNERSHIP_ACK: PARTNERSHIP_REPLY mesajı alan düğümün işbirlikçi listesi henüz dolmamışsa alıcı düğüm, gönderici düğümü zaman damgası ile beraber işbirlikçi listesine ekler ve cevap olarak PARTNERSHIP_ACK

(“acknowledgement”) mesajı gönderir (Şekil 5.5). Bu mesajı alan düğüm tampon haritaların değişimi için zamanlayıcısını başlatır.

PARTNERSHIP_NACK: PARTNERSHIP_REPLY mesajı alan düğümün işbirlikçi listesi dolu ise bu mesajı alan düğüm, göndericiye PARTNERSHIP_NACK mesajı gönderir. PARTNERSHIP_NACK mesajını alan düğüm ilgili düğümü işbirlikçi listesinden çıkarır (Şekil 5.6).

Şekil 5.5’te 23 numaralı eş, üyelik listesindeki 3 numaralı eşe PARTNERSHIP_REQUEST mesajı gönderir. 3 numaralı eşin işbirlikçi listesinde yer olduğu için istek kabul edilir.



Şekil 5.5. Bir Eşin İşbirlikçi Olarak Kabul Edilmesi

Şekil 5.6’da 23 numaralı eş, üyelik listesindeki 4 numaralı eş dışındaki eşleri işbirlikçi listesine eklemiş, sonrasında 4 numaralı eşe PARTNERSHIP_REPLY mesajı göndermiştir. Ancak göndericinin işbirlikçi listesi 4 numaralı eşden PARTNERSHIP_REPLY mesajı alana kadar dolduğundan 23 numaralı eş, 4 numaralı eşe PARTNERSHIP_NACK mesajı göndermiştir.



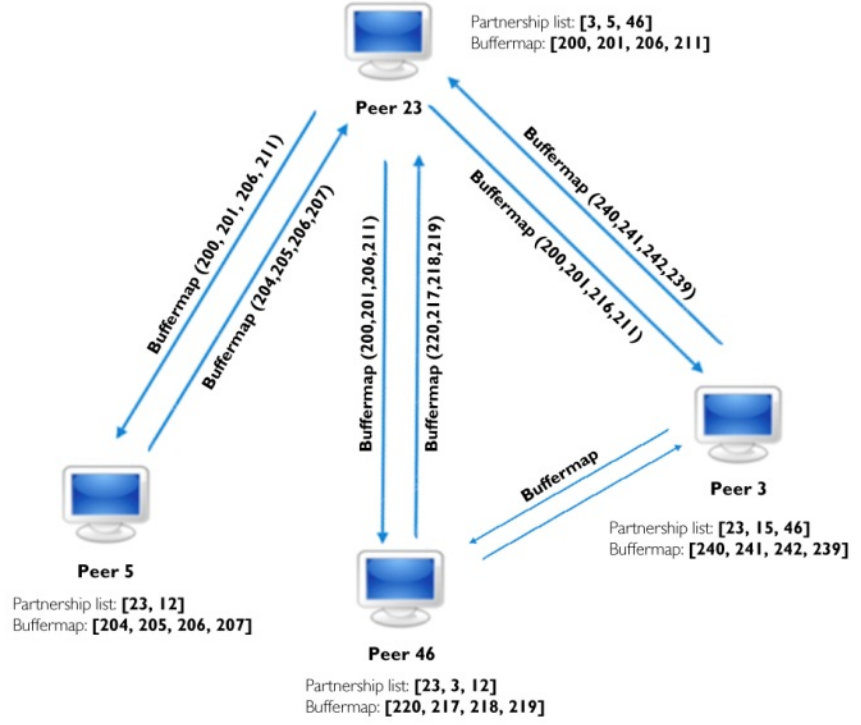
Şekil 5.6. PARTNERSHIP_NACK Mesajı

5.3.3 İşbirlikçiler arası haberleşme

İşbirlikçiler periyodik aralıklarla (hazırlanan sistemde bu süre 2 saniye olarak belirlenmiştir) durum bilgilerini birbirlerine gönderirler.

BUFFERMAP: Bu mesaj işbirlikçiler arasında düzenli aralıklarla gönderilir. Mesajın içeriğinde eşin tampon haritası, alt akışlara (“substream”) ait kayıtlar ve çevrim dışı eşlerin bilgileri yer alır.

Şekil 5.7’de eşlerin birbirlerine gönderdikleri tampon harita bilgileri gösterilmiştir. Tampon harita bilgisini alan eş, kendi tarafında ilgili eşin bilgilerini günceller ve bu bilgiler daha sonra ebeveyn seçimi için kullanılır.



Şekil 5.7. İşbirlikçiler Arası Tampon Harita Değişimi

5.4 Izgara Tabanlı Akışlandırma Sisteminde Video Gönderimi

Bu bölümde, sistemdeki video gönderim mekanizmasının nasıl çalıştığı anlatılacaktır. Önce ebeveyn seçimi (“parent selection”) aktarılacak, sonra düğümlerdeki video gönderim işleyişi ve mesaj tipleri açıklanacaktır.

Ebeveyn Seçimi: Sisteme yeni bağlanan bir eş, işbirlikçilerini belirledikten sonra onların tampon harita bilgilerini almaya başlar. Video akışının başlayabilmesi için bu tampon bilgileri kullanılarak ebeveyn seçilmesi gereklidir. Öte yandan sisteme bağlı olan ve video izleyen bir eşin de çeşitli sebeplerle (ebeveyninin çevrim dışı olması, video akışının yavaş olması veya durması gibi) ebeveynlerini kontrol etmesi ve gerekli durumlarda işbirlikçileri arasından tekrar ebeveyn seçimi yapması gerekmektedir. Bu sebeple periyodik aralıklarla eşler, kendilerinin, ebeveynlerinin ve işbirlikçilerinin tampon harita bilgilerini kıyaslayarak yeni ebeveyn seçimleri gerçekleştirebilirler.

Coolstreaming-benzeri Ebeveyn Seçimi: Coolstreaming-benzeri ebeveyn seçimi aday ebeveynlerin belirlenmesi ile başlar. Ebeveyn adaylarının

belirlenmesi için aşağıdaki eşitsizliklerden faydalanılır. İlgili notasyona ait açıklamalar Tablo 5.1'de verilmiştir.

$$\max\{|B_{P_j} - B_i|, i \leq K\} < T_s, j \leq K \quad (1)$$

$$\begin{aligned} A &= \max\{B_{Q_i}, i \leq K, Q \in \text{partners}\} \\ A - B_{P_j} &< T_p, j \leq K \end{aligned} \quad (2)$$

Tablo 5.1. Ebeveyn Seçimi için Açıklamalar

Gösterim	Açıklama
P_j	j. alt akışın ebeveyni
B_i	Eşin kendi tampon haritasındaki i. alt akış değeri
B_{P_j}	j. alt akışın ebeveyninin tampon harita bilgisi
T_s	Eşin kendi tampon haritasındaki bir alt akış değeri ile ebeveyninin tampon haritasındaki değer arasındaki farkın eşik değeri
T_p	Eşin işbirlikçisinin tampon haritasındaki bir alt akış değeri ile ebeveyninin tampon haritasındaki değer arasındaki farkın eşik değeri
K	Toplam alt akış sayısı

(1) numaralı eşitsizlik; i. eşin, video paketi aldığı ebeveyninin ilgili tampon harita değeri ile kendi tampon haritasındaki bütün alt akış değerleri farkının maksimumu arasındaki farkın T_s eşik değerinden küçük olup olmadığını kontrol eder. Eğer bu fark, eşik değerinden büyük bir değer çıkarsa i. eş, ebeveyninden video akışı için yeterli veri alamadığını anlar ve ebeveyn değiştirme arayışı içerisine girer.

(2) numaralı eşitsizlikte; herhangi bir eş, işbirlikçilerinin maksimum tampon haritası değerini ebeveynlerinin tampon haritası değerleri ile karşılaştırır. Yapılan kıyaslama sonucu aradaki farkın T_p eşik değerinden büyük çıkması, ilgili eşin ebeveyninden video akışı için yeterli veri alamadığını göstermektedir. Böyle bir durumda ilgili eş, ebeveyn değiştirme arayışı içerisine girer.

Herhangi bir eş, (1) veya (2) numaralı eşitsizliklere uygun olmadığını tespit ettiği alt akış için yeni ebeveynini seçerken (2) ve (3) numaralı eşitsizlikleri göz önüne alır.

$$B_{Q_j} - \max\{B_i, i \leq K\} < T_s \quad (3)$$

Coolstreaming-benzeri ebeveyn seçimi (2) ve (3) numaralı eşitsizlikleri sağlayan aday ebeveynler arasından (her alt akış için ayrı) rastgele bir ebeveyn seçilmesi şeklinde gerçekleştirilir.

Sistemde T_s ve T_p kontrolleri T_a zamanlayıcısı ile yapılmaktadır. T_a zamanlayıcısı hazırlanan sistemde parametrik olarak 20 saniyede bir çalışmaktadır ve gerekli durumlarda herhangi bir düğümün yeni ebeveyn(ler) seçmesini sağlar.

Video Akışlandırma: Sistemde video kare hızı (“frame rate”) 25 fps, her kare (“frame”) 1 video paketi (1460 bytes) olarak tanımlanmıştır. Alt akış sayısı 4 olarak belirlenmiştir. Belirtilen tüm bu değerler parametrik olup, istenildiği takdirde değiştirilebilmektedir.

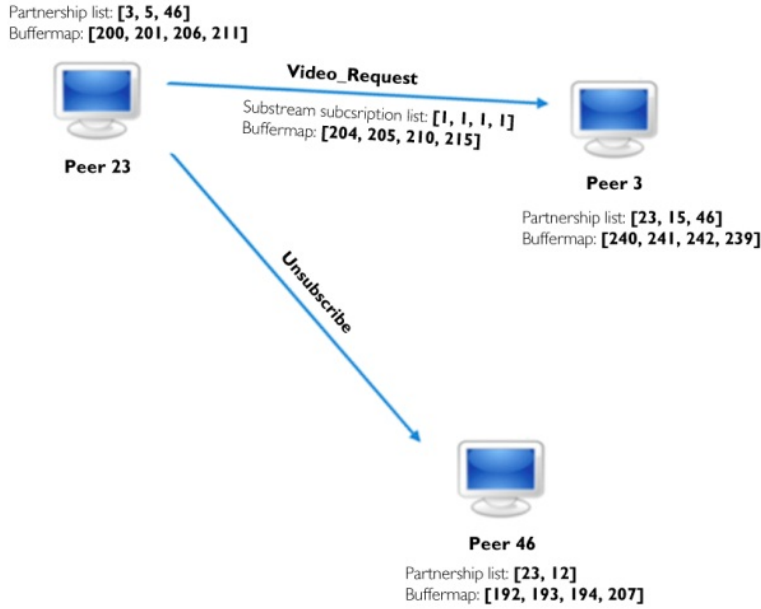
Video İsteği Gönderilmesi: İşbirlikçiler birbirleriyle periyodik olarak tampon haritalarını paylaşarak sahip oldukları video paket bilgilerini kaydederler. Eşler, işbirlikçilerinin tampon harita bilgilerini T_s ve T_p değerleri ile kıyaslayarak hangi işbirlikçi(ler)den video paketi alabileceklerini yani potansiyel ebeveyn(ler)ini belirlerler. Sonrasında bu işbirlikçilerden birini rastgele seçerek video isteğinde (VIDEO_REQUEST) bulunurlar. Sistemdeki bir eş, ebeveyn değiştireceği zaman eski ebeveynine UNSUBSCRIBE mesajı göndererek ondan video paket alımını durdurur.

VIDEO_REQUEST: Bu mesaj ebeveyn olarak seçilen eşe veya kaynak sunucusuna gönderilir. Mesajı gönderen eş, hangi alt akış için paket istediğini ve hangi paket numarasından itibaren video paketlerini istediğini mesajın içeriğinde belirtir. VIDEO_REQUEST mesajını alan düğüm, mesajı gönderen eş, ilgili alt akıştaki çocukları (“child”) arasına ekler ve istediği paketten itibaren video paketi gönderimine başlar.

UNSUBSCRIBE: Bu mesaj eski ebeveyne gönderilerek video paket alımının durdurulması sağlanır. Bu mesajı alan eş, ilgili eşin alt akıştaki çocuk listesinden çıkarır ve video paket istek bilgilerini siler.

Şekil 5.8’de 23 numaralı eş, ebeveyni olan 46 numaralı eşin video gönderiminde aksaklık olduğunu fark ettiğinde (T_s ve T_p kontrolleri ile) 46 numaralı eşin üyelik ve işbirlikçi listelerinden UNSUBSCRIBE mesajı göndererek ayrılmaktadır. Aynı zamanda 3 numaralı eşe VIDEO_REQUEST mesajı göndererek bütün alt akışlar için (1, 2, 3, 4) kayıt olmaktadır. Video paket

isteğini ise her alt akış için kaldığı son video paket numarasına göre ayarlamaktadır.



Şekil 5.8. Video İsteği Gönderilmesi

Video Gönderimi: Sistemdeki her düğümün bir karşıya yükleme kapasitesi bulunmaktadır. Bu bilgiyi sisteme girerken bir metin dosyasından okuyarak elde ederler. Sistemdeki eşlerin karşıya yükleme ortalaması 800 Kbps; kaynak sunucunun video gönderim hızı ise 3 Mbps olarak belirlenmiştir. Her düğüm alt akışlarına kayıtlı çocuk sayısı ile ters orantılı ve karşıya yükleme bant genişliğiyle orantılı olarak video gönderimlerini gerçekleştirir. Video gönderim aralıkları şu şekilde gösterilebilir:

$$\text{Video paket gönderim aralığı} = 1 / [(\text{karşı yükleme bant genişliği}) / \text{video paket boyutu} * \text{toplam çocuk sayısı}] \quad (4)$$

(4) numaralı eşitliğin kullanım örneği şu şekilde verilebilir: Örneğin A eşinin karşı yükleme bit hızı 600 kbps, 1. ve 2. alt akışa 2; 3. ve 4. alt akışlara ise 1 çocuk bağlı olsun. Bu durumda A eşinin video gönderim zaman aralığı,

A eşinin video paket gönderim aralığı = $1 / (((600 * 1000) / ((1460 * 8) * (2 + 2 + 1 + 1))) * 1000)$ olacaktır (hesabın sonunda elde edilen değerin 1000 ile çarpılmasının nedeni, sonucun milisaniye cinsinden olmasını sağlamak içindir).

6. SİMULASYON SONUÇLARI VE YORUMLAR

Bu bölümde simülasyon ortamı ve simülasyon sonucunda elde edilen veriler ortaya konmuştur. Ayrıca CoolStreaming benzeri bir sistem ile önerilen Çok-Etmenli Sistem karşılaştırılmıştır. Son olarak Çok-Etmenli Sistemin CoolStreaming benzeri sisteme göre avantajları anlatılmıştır.

6.1 Simulasyon Ortamı

Testler 300 düğüm için yapılmıştır. Gerçekleştirilen tüm deneylerde bir adet video sunucu, bir adet de takipçi sunucu mevcuttur. Video sunucusu, sistemdeki düğümlerin toplam %10'una video gönderimi yapabilecek yükleme bant genişliğine sahiptir. Sistemdeki tüm eşlerin kümülatif yükleme bant genişlikleri dağılımı Tablo 6.1'de görülebilir. Deneylerde kullanılan ağ topolojileri BRITE topoloji üreticisi ile Barabasi modeline (Barabasi and Albert, 1999) göre üretilmiştir. Kullanılan ağ topolojisi üzerindeki her fiziksel bağlantının gecikme süresi sabit 50 ms olarak belirlenmiştir. Tüm testlerde 15 dakikalık ve 300 Kbps bit hızında video akışlandırması gerçekleştirilmiş; tüm video ve kontrol paketleri günümüzde kullanılan uygulamalarda olduğu gibi TCP üzerinden gönderilmiştir. Video paketlerinin iletimi için TCP tercih edilmesinin sebebi güvenlik duvarı arkasındaki kullanıcıların da video izleyebilmesini sağlamaktır. Ayrıca, düğüm giriş çıkışları için üstel dağılım kullanılmış, giriş süresi ortalaması 500, çıkış süresi ortalaması ise 200 saniye olarak belirlenmiştir. Alt akış sayısı olan K değeri ise 4 olarak tanımlanmıştır. Testlerde seçilen parametreler, (Kaymak et al., 2013) çalışmasındaki parametre analizi sonucunda belirlenmiştir.

Tablo 6.1. Kümülatif Yükleme Bant Genişlikleri Dağılımı

Yüzde	Yükleme Bant genişliği
%10	<50 Kbps
%50	<300 Kbps
%90	<1000 Kbps
%100	<3000 Kbps

6.2 Simulasyon Sonuçları

Bu bölümde simülasyon sonuçları verilmiştir. Sonuçlar ağ trafiğindeki değişim ve deneyim kalitesine göre incelenmiştir. Bunun için;

- Devamlılık Endeksi
- Video İzlemeye Başlama Süresi
- Ebeveyn Değiştirme Sayısı
- Ağ Trafik Oranı

parametreleri incelenmiştir.

CoolStreaming benzeri sistemin ve önerilen Çok-Etmenli Sistemin ortalama devamlılık endeksi değerleri Tablo 6.2’de verilmiştir. Çok-Etmenli Sistem’de uygulanan Ebeveyn Puanlama algoritmasından dolayı Çok-Etmenli Sistem’in performansı CoolStreaming benzeri sisteme göre daha yüksektir.

Tablo 6.2. Ortalama Devamlılık Endeksi

	Devamlılık Endeksi
CoolStreaming benzeri	%87.7
Çok-Etmenli Sistem	%92.2

Tablo 6.3’te CoolStreaming benzeri sistemin ve önerilen Çok-Etmenli Sistemin ortalama video izlemeye başlama süresi verilmiştir. Video izlemeye başlama süresi, düğümün sisteme katıldıktan sonra video izlemeye başlayana kadar geçen süreye denir. CoolStreaming benzeri sistem ağda yakınlığı göz önünde bulundurmadığı için video izlemeye başlama süresi 21 saniyedir. Diğer taraftan, Çok-Etmenli Sistem’de Yeni Eş Karşılama algoritmasında, Tracker Server etmeni düğümlerin yakınlığını da göz önünde bulundurduğu için video izlemeye başlama süresi 16 saniyeye düşmüştür.

Tablo 6.3. Ortalama Video İzlemeye Başlama Süresi

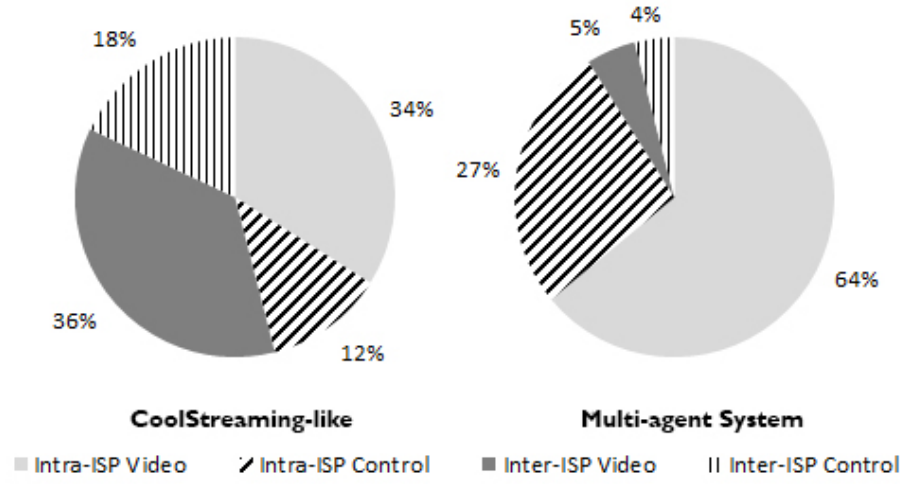
	Video İzlemeye Başlama Süresi
CoolStreaming benzeri	21.2 saniye
Çok-Etmenli Sistem	16.1 saniye

Tablo 6.4'te CoolStreaming benzeri sistemin ve önerilen Çok-Etmenli Sistemin ortalama ebeveyn değiştirme sayısı verilmiştir. Ebeveyn Puanlama algoritmasından dolayı Çok-Etmenli Sistem daha az ebeveyn değiştirirken, CoolStreaming benzeri sistem, rastgele ebeveyn seçtiği için kötü sonuç vermiştir.

Tablo 6.4. Ortalama Ebeveyn Değiştirme Sayısı

	Ebeveyn Değiştirme Sayısı
CoolStreaming benzeri	48.7
Çok-Etmenli Sistem	25.2

Toplam ağ trafiğindeki oranlar Şekil 6.1'de verilmiştir. CoolStreaming benzeri sistemin ağ hakkında hiç bir bilgisi yoktur. Yani ortaklar rastgele seçilir ve bu sebeple ağ omurgasındaki yük artar. Diğer taraftan, Tracker Server etmeninde ağ topolojisi ve düğümlerin yükleme bant genişliği bilgileri bulunur. Bu bilgileri kullanarak Yeni Eş Karşılama algoritması ile düğümlere aynı ISP'den ve yüksek yükleme bant genişliğine sahip düğümleri üye listesinde döndürür. Böylece internet trafiğinin çoğu ISP içerisinde kalır ve ağ omurgasındaki yük düşer. CoolStreaming benzeri sistemde ağ omurgasından geçen paket oranı %54 iken, önerilen Çok-Etmenli Sistem'de bu oran %9'dur.



Şekil 6.1. Ağ Trafikindeki Paket Oranı

7. SONUÇ VE ÖNERİLER

Bu tez çalışmasında video akışlandırma sistemlerinde Kanı-İstek-Hedef ("KİH") modeline göre çalışan yeni bir çok-etmenli sistem hazırlanmıştır. Sistemdeki etmenlere ait iç yapılar verilmiş ve ns-3'te yapılan simulasyon sonuçları aktarılmıştır. Simulasyonlarda, önerilen çok-etmenli sistem ile CoolStreaming benzeri sistem testleri yapılmıştır. Etmenlerde uygulanan davranışlar sayesinde CoolStreaming benzeri sisteme göre iyileştirmeler elde edilmiştir.

Bu çalışmada iyi geliştirilen bir eş seçimi davranışının kullanıcının deneyim kalitesini devamlılık endeksi, video izlemeye başlama süresi ve ebeveyn değiştirme sayısı yönünden iyileştireceği gözlemlenmiştir. Diğer bir deyişle, bu metot sayesinde kullanıcılar daha iyi kaliteyle ve daha az gecikme ile video izleyebilirler.

Ayrıca sistem etmenlerine ait ağ trafik yönetimi davranışının ISP arası trafiği önemli ölçüde azalttığı görülmüştür. Görevdeş ağlarda video akışlandırma uygulamalarının ağ farkındalığı yeterli olmadığı için, bu uygulamalar ağ altyapısında fazladan yüke sebep olurlar. Ek olarak bu davranış ile uçtan uca gecikmeler azalır ve bu sayede kullanıcının deneyim kalitesi artar.

Son olarak uygulanan Çocuk Yönetim davranışı ile önerilen sistemin hatalara ve düğüm giriş-çıkışlarına karşı daha dayanıklı olduğu görülmüştür. Düğüm giriş-çıkışı görevdeş ağlarda video akışlandırma uygulamalarında önemli bir sorun olduğu için bu yöntemin de kullanıcı deneyim kalitesine katkısı olmuştur.

Gelecekte yapılacak çalışmalarda etmenlere pazarlık mekanizmaları eklenebilir. Bu mekanizmalar bant genişliği yönetimi ve video kalite düzenlemesi için kullanılabilir. Buna ek olarak, yeniden gönderim ve yol seçimi gibi mekanizmalar kullanılarak video akışlandırmanın performansı artırılabilir.

KAYNAKLAR DİZİNİ

- Barabasi, A. L. and Albert, R.**, 1999, Emergence of scaling in random networks, Science, 509-512pp.
- Blond, S., Fessant, F. and Merrer, E.**, 2009, Finding good partners in availability-aware P2P networks, ACM Transactions on Autonomous and Adaptive Systems, 472-484 pp.
- Bratman, M.E.**, 1987, Intention, Plans, and Practical Reason, Harvard University Press, Cambridge, 200p.
- BitTorrent**, “BitTorrent”, <http://www.bittorrent.com> (Erişim tarihi: Ağustos 2013).
- Carrera, A. and Iglesias, C. A.**, 2011, Multi-agent architecture for heterogeneous reasoning under uncertainty combining MSBN and ontologies in distributed network diagnosis, Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), 2:159-162pp.
- Chen, Y., Wang, B., Lin, W. S., Wu, Y. and Liu, K. J. R.**, 2010, Evolutionary games for cooperative P2P video streaming, Proceedings of the 17th IEEE International Conference on Image Processing (ICIP), 4453-4456 pp.
- Cohen, B.**, 2008, The BitTorrent Protocol Specification
- Dance, S., Gorman, M., Padgham, L. and Winikoff, M.**, 2003, A deployed multi-agent system for meteorological alerts, Deployed Applications of Autonomous Agents and Multiagent Systems Workshop (AAMAS03), 8p.
- Engmo, L. and Hallen, L.** 2007, Software Agents Applied in Oil Production, MSc Thesis, Norwegian University of Science and Technology, 134p (unpublished).

KAYNAKLAR DİZİNİ (devam)

- Fletcher, M., Lucas, A., Jarvis, D., Jarvis, J., Ronnquist, R. R., Mcfarlane, D. C., Brusey, J. and Thorne, A.,** 2003, JACK-based holonic control of a gift box packing cell, Technical Report, Distributed Information and Automation Laboratory, 35p.
- Ganesh, A. J., Kermarrec, A. M. and Massoulié, L.,** 2003, Peer-to-Peer membership management for gossip-based protocols, IEEE Trans. Computers, 52(2), 139-149 pp.
- Georgeff, M. P. and Ingrand, F. F.,** 1989, Monitoring and control of spacecraft systems using procedural reasoning, Proceedings of the Space Operations Automation and Robotics Workshop, 19p.
- Goyal, V.K.,** 2001, Multiple description coding: compression meets the network, IEEE Signal Processing Magazine, 18(5), 74-94 pp.
- Hei, X., Liang, C., Liang, J., Liu, Y. and Ross, K. W.,** 2006, Insights into PPLive: A measurement study of a large-scale P2P IPTV system, in IPTV workshop in conjunction with International World Wide Web Conference.
- Hei, X., Liu, Y. and Ross, K.W.,** 2007, Inferring network-wide quality in P2P live streaming systems, IEEE Journal on Selected Areas in Communications, 25(9),1640-1654pp.
- Hei., X., Liang, C., Liang, J., Liu, Y. and Ross, K. W.,** 2007, A measurement study of a large-scale P2P IPTV system, IEEE Transactions on Multimedia, 9(8), 1672-1687 pp.
- Kardas, G., Challenger, M., Yildirim, S. and Yamuc, A.,** 2012, Design and implementation of a multi-agent stock trading system, Software: Practice and Experience, Wiley-Blackwell Publishing, Vol. 42, Issue 10, 1247-1273pp.
- Kaymak, Y., Teket, K.D. and Sayit, M.F.,** 2013, Parameter analysis for a push/pull-based P2P live video streaming application, Signal Processing and Communications Applications Conference (SIU), 2013, 1-4 pp.

KAYNAKLAR DİZİNİ (devam)

- Kostadinova, R.**, 2008, Peer-to-Peer Video Streaming, Masters' Degree Project, Sweden.
- Li, B., Xie, S., Qu, Y., Keung, G. Y., Lin, C., Liu, J. and Zhang, X.**, 2008, Inside the new coolstreaming: principles, measurements and performance implications, Proceedings of IEEE International Conference on Information communications (INFOCOM), 1031-1039 pp.
- Li, C. and Chen, C.**, 2009, Measurement based PPStream client behavior analysis, In Proceedings of 2009 International Colloquium on Computing, Communication, Control, and Management (CCCM'09), Sanya, China, 341-345 pp.
- Li, Y.-T.H., Ren, D., Chan, S.-H.G. and Begen, A.C.**, 2009, Low-delay mesh with peer churns for peer-to-peer streaming, IEEE International Conference on Multimedia and Expo (ICME).
- Liu, Y., Guo, Y. and Liang, C.**, 2008, A survey on peer-to-peer video streaming systems, Peer-to-Peer Networking and Applications, 1(1), 18-28 pp.
- Liu, Z., Wu, C., Li, B. and Zhao, S.**, 2010, UUSee: large-scale operational on-demand streaming with random network coding, Proceedings of IEEE International Conference on Information communications (INFOCOM), 2070-2078pp.
- Lucas, A., Rönnquist, R., Ljungberg, M., Howden, N., Corke, P. and Sikka, P.**, 2003, Teamed UAVs - a new approach with intelligent agents, 2nd conference on AIAA "Unmanned Unlimited" Systems Technologies, and Operations – Aerospace, Land, and Sea Conference.
- Mathieson, I., Dance, S., Padgham, L., Gorman, M. and Winikoff, M.**, 2004, An open meteorological alerting system: issues and solutions, Proceedings of the 27th Australasian Computer Science Conference, 351-358pp.

KAYNAKLAR DİZİNİ (devam)

- Menkovski, V., and Liotta, A.,** 2013, Intelligent control for adaptive video streaming, IEEE International Conference on Consumer Electronics (ICCE), 127-128pp.
- Molina, B., Pileggi, S. F., Esteve, M. and Palau, C. E.,** 2009, A negotiation framework for content distribution in mobile transient networks, Journal of Network and Computer Applications, 32:1000-1011pp.
- Murray, G., Steuart, S., Appl, D., Mcilroy, D., Heinze, C., Cross, M., Chandran, A., Raszka, R., Rao, A.S., Pegler, A., Morley, D. and Busetta, P.,** 1995, The challenge of whole air mission modeling, Proceedings of the Australian Joint Conference on Artificial Intelligence AI '95.
- ns-3,** “Network Simulator-3”, <http://www.nsnam.org> (Erişim tarihi: Ağustos, 2013).
- Nordbo, E.,** 2011, Inter-Agent Communication in Multi-Agent Systems, MSc Thesis, Norwegian University of Science and Technology, 83p (unpublished).
- Orynczak, G. and Kotulski, Z.,** 2011, Agent-based infrastructure for real-time applications, Annales UMCS Informatica, 11:33-47pp.
- Padgham, L. and Winikoff, M.,** 2003, Prometheus: A methodology for developing intelligent agents, Lecture Notes in Computer Science, 2585, 174-185pp.
- Padgham, L., Thangarajah, J. and Winikoff, M.,** 2008, Prometheus Design Tool, Proc. AAAI Conference on Artificial Intelligence, 1882-1884pp.
- Pereira, J., Rodrigues, L., Monteiro, M. J., Oliveira, R. and Kermarrec, A.-M.,** 2003, NEEM: Network-friendly epidemic multicast, 22nd International Symposium on Reliable Distributed Systems (SRDS'03), 15pp.

KAYNAKLAR DİZİNİ (devam)

- Pournaras, E., Warnier, M. and Brazier, F.,** 2009, Adaptive agent-based self-organization for robust hierarchical topologies, Proceedings of the 2009 International Conference on Adaptive and Intelligent Systems (ICAIS '09), 69-76pp.
- PPLive**, “PPLive”, <http://www.pplive.com> (Erişim tarihi: Ağustos, 2013).
- PPstream**, “PPStream”, <http://www.ppstream.com> (Erişim tarihi: Ağustos, 2013).
- Rao, A. and Georgeff, M.,** 1995, BDI agents: from theory to practice, Proceedings of the First International Conference on Multi-Agent Systems (ICMAS'95), San Francisco, CA, USA, 312-319pp.
- Sayit, M. F., Tunali, E. T. and Tekalp, A. M.,** 2012, Resilient peer-to-peer streaming of scalable video over hierarchical multicast trees with backup parent pools, Signal Processing: Image Communication, 27:113-125pp.
- Teket, K.D., Kaymak, Y., Kardas G., and Sayit, M.F.,** 2013, Design and implementation of a multiagent P2P video streaming system, 2013 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA 2013), Albena, Bulgaria, 1-7pp.
- Voulgaris, S., Gavidia, D. and van Steen, M.,** 2005, CYCLON: Inexpensive membership management for unstructured P2P overlays, Journal of Network and Systems Management, 13(2), 197-217pp.
- Wei, W., Li, Y. and Hu, X.,** 2011, Coolstream live broadcast system based on Ex-SCAMP, JCIT: Journal of Convergence Information Technology, 6(7), 203-209pp.
- Wooldridge, M.,** 2009, An Introduction to MultiAgent Systems, 2nd Edition, John Wiley & Sons Ltd., 484p.
- Xie, S., Li, B., Keung, G.Y. and Zhang, X.,** 2007, Coolstreaming: design, theory and practice, IEEE Transactions on Multimedia, 9:1661-1671pp.

KAYNAKLAR DİZİNİ (devam)

Yiu, W. P. K., Jin, X. and Chan, S. H. G., 2007, Challenges and approaches in large-scale P2P media streaming, IEEE MultiMedia, 14(2), 50-59pp.

Zhang, X., Liu, J., Li, B. and Yum, Y.-S.P., 2005, CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming, Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies, 2102-2111pp.

ÖZGEÇMİŞ

Kemal Deniz TEKET

Adres: Kazım Dirik Mahallesi 180/1 Sokak No:3 B blok Kat: 5 Daire:15 35100 Bornova/İZMİR

e-mail: denizkemal@gmail.com

Eğitim Durumu

Yüksek Lisans : 2010- , EGE Üniversitesi, Uluslararası Bilgisayar Enstitüsü
 Lisans : 2001-2008, Galatasaray Üniversitesi, Bilgisayar Mühendisliği
 Lise : 1994-2001, İzmir Anadolu Lisesi

Yabancı Dil

Türkçe : anadil
 İngilizce : iyi derecede
 Fransızca : iyi derecede
 Almanca : başlangıç seviyesi

Yayınlar

- Muge Fesci Sayit, Yagiz Kaymak, *Kemal Deniz Teket*, Cihat Cetinkaya, Sercan Demirci and Geylani Kardas, “Parent Selection via Reinforcement Learning in Mesh Based P2P Video Streaming”, International Conference on Information Technology: New Generations (ITNG), 2013, pp. 546-551
- *Kemal Deniz Teket* ve Muge Fesci Sayit, “P2P Video Streaming with ALTO Protocol: A Simulation Study”, IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), 2013
- *Kemal Deniz Teket*, Yagiz Kaymak, Geylani Kardas and Muge Fesci Sayit, “Design and Implementation of a Multiagent P2P Video Streaming System”, International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), 2013, pp. 1-7
- Yagiz Kaymak, *Kemal Deniz Teket* and Muge Fesci Sayit, “A Study on ALTO-assisted P2P Live Video Streaming Systems”, International Workshop on Modeling and Simulation of Peer-to-Peer and Autonomic Systems (MOSPAS), 2013

- Muge Fesci Sayit, Erdem Karayer, *Kemal Deniz Teket*, Yagiz Kaymak, Cihat Cetinkaya, Sercan Demirci and Geylani Kardas, "A Score-Based Packet Retransmission Approach for Push-Pull P2P Video Streaming Systems", International Symposium on Multimedia Applications and Processing (MMAAP), 2013
- Yagiz Kaymak, *Kemal Deniz Teket*, Sercan Demirci, Sinem Getir, Riza Cenk Erdur, Geylani Kardas, " JACK Kanı-İstek-Hedef Modeline dayalı Çok-etmenli bir Kütüphane Yönetim Sisteminin Prometheus Metodolojisi ile Geliştirilmesi", Ulusal Yazılım Mühendisliği Sempozyumu (UYMS), 2012, pp. 121-128
- Yagiz Kaymak, *Kemal Deniz Teket*, Müge Fesci Sayit, "Parameter Analysis for a Push/Pull-based P2P Live Video Streaming Application", Signal Processing and Communications Applications Conference (SIU), 2013, pp. 1-4

İş Deneyimi

- Araştırmacı, Tübitak ödenekli 1001 Projesi, Proje No: 111E022 (2011-2013)
- Yazılım Geliştirici, TT Atölye, (2009-2011)
- Yazılım Geliştirici, Espada, (2008-2009)
- Stajyer, IBM, (2006)
- Stajyer, Çizgi Elektronik (2005)

Programlama Dilleri Bilgisi

- Java, C/C++, C#

EKLER

Ek 1 İngilizce - Türkçe Sözlük

EK-1

İngilizce - Türkçe Sözlük

Action	Eylem
Adaptive Virtual Organization Model	Uyarlanabilir Sanal Organizasyon Modeli
Agent	Etmen
Autonomous	Özerk
Behavior	Davranış
Belief	Kanı
Belief-Desire-Intention	Kanı-İstek-Hedef
Buffermap	Tampon Harita
Capability	Yetenek
Client-Server	İstemci-sunucu
Connected	Bağış
Content Network	İçerik Ağı
Continuity Index	Devamlılık Endeksi
Data Dissemination	Veri Dağıtımı
Desire	İstek
Discrete Event	Kesikli Olay
Dynamic Routing Table	Dinamik Yönlendirme Tablosu
Epidemic	Salgın
Event	Olay
Evolutionary Game Theory	Evrimsel Oyun Teorisi
Frame	Kare
Frame Rate	Kare Hızı
Functionality	İşlevsellik
Hop Count	Atlama Sayısı
Hybrid Data Dissemination	Melez Veri Dağıtımı
Intention	Hedef
Interaction Diagram	Etkileşim Diyagramı
Member	Üye
Mobile Transient Networks	Geçici Süreli Ağlar
Multi-Agent Systems	Çok Etmenli Sistemler

EK-1 (devam)

İngilizce - Türkçe Sözlük

Multiple Tree	Çoklu Ağaç
Negotiation Protocol	Pazarlık Protokolü
Network	Ağ
Node	Düğüm
Overlay	Üst Katman
Partial View	Kısmi Görünüm
Partner	İşbirlikçi (Ortak)
Parent	Ebeveyn
Peer	Eş
Peer-to-Peer Network	Görevdeş Ağ
Proactive	Proaktif
Percept	Algı
Playback Latency	Oynatma Zaman Farkı
Point to Point	Uçtan uca
Practical Reasoning	Pratik Düşünme
Pseudocode	Zahiti Kod
Reactive	Karşıt Eylemli
Reinforcement Learning	Pekiştirilmiş Öğrenme
Software Agent	Yazılım Etmeni
Startup Delay	Video İzlemeye Başlama Süresi
Sub-Stream	Alt Akışlandırma
Tree	Ağaç
Upload Bandwidth	Yükleme Bant Genişliği
Video Streaming	Video Akışlandırma
Video-on-Demand	İsteğe Bağlı Video