

**OPTİK WDM AĞLARINDA SANAL TOPOLOJİ
UYARLAMASINA DAĞITILMIŞ BİR YAKLAŞIM**

**YÜKSEK LİSANS TEZİ
Müh. İsmail Caner BİROL
(504021502)**

**Tezin Enstitüye Verildiği Tarih : 19 Eylül 2005
Tezin Savunulduğu Tarih : 22 Eylül 2005**

Tez Danışmanı : Yrd. Doç. Dr. Feza BUZLUCA
Diğer Jüri Üyeleri Prof. Dr. Bilge GÜNSEL
Yrd. Doç. Dr. Ayşegül GENÇATA

EYLÜL 2005

ÖNSÖZ

Yüksek lisans çalışmalarım sırasında, desteğini, bilgisini ve uzmanlığını benden esirgemeyen Yrd. Doç. Dr. Feza BUZLUCA'ya teşekkürü bir borç biliyorum.

İsmail Caner BİROL

EYLÜL 2005



İÇİNDEKİLER

ÖNSÖZ.....	ii
İÇİNDEKİLER	iii
KISALTMALAR	v
TABLO LİSTESİ.....	vi
ŞEKİL LİSTESİ.....	vii
ÖZET.....	viii
SUMMARY	x
1. GİRİŞ	1
1.1 Problem ve Önerilen Çözüm.....	2
1.2 Bölümlerin Özetleri.....	4
2. OPTİK WDM AĞLARI	5
2.1 WDM - Dalgaboyu bölümlemeli çoğullama.....	5
2.2 Sanal Topoloji	6
2.2 Yönlendirme.....	9
2.3 Sanal Topolojiyi yeniden düzenleme gereksinimi	12
2.4 Konu ile ilgili daha önce yapılan çalışmalar	13
3. DAĞITILMIŞ SANAL TOPOLOJİ UYARLAMASI.....	16
3.1 Yönlendirme.....	16
3.2 Kısıtlar ve Kabüller	17
3.3 Çalışma Prensipleri	19
3.3.1 Uyarılama İçin Gerekli Veriler	19
3.3.2 Işıkyolu ekleme	20
3.3.3 Işıkyolu Silme	23
3.3.4 Kontrol Mesajları	27
3.3.5 Örnek Çalışma	28
3.4 Osilasyon Problemi	31
3.5 Etiket Kaydırma	32
4. SİMÜLASYON.....	34
4.1 Simülasyon Ortamı	34
4.2 Uyarılmalı ve Uyarılamasız Sanal Ağ Değerleri.....	36
4.3 80-15 Uyarılması ile 70-15 Uyarılması Karşılaştırması	42
4.4 Değişik Simülasyon Senaryoları	51

5. SONUÇLAR VE TARTIŞMA	52
5.1 Simülasyon Sonuçları	52
5.2 Gelecekteki Çalışma Önerileri	53
KAYNAKLAR	54
ÖZGEÇMİŞ.....	56



KISALTMALAR

WDM	: Wavelength Division Multiplexing (Dalgaboyu Bölümlemeli Çoğullama)
EAY	: Etiket Anahtarlama Yolu
Mbps	: Megabit per second (Saniyedeki megabit sayısı)
Gbps	: Gigabit per second (Saniyedeki gigabit sayısı)
Tbps	: Terabit per second (Saniyedeki terabit sayısı)
OC	: Optical Channel (Optik Kanal)
OC-n	: Saniyede 51.84 megabit * n'lik veri oranı

TABLO LİSTESİ

	<u>Sayfa Numarası</u>
Tablo 2.1 : Düğüm 3'ün Yönlendirme Tablosu	11
Tablo 2.2 : Düğüm 2'nin Yönlendirme Tablosu	11
Tablo 2.3 : Düğüm 1'in Yönlendirme Tablosu	11
Tablo 2.4 : Düğüm 6'nın Yönlendirme Tablosu	12
Tablo 3.1 : Ağ üzerindeki EAY'lar	28
Tablo 3.2 : D ₄ Düğümünün Başlangıçtaki Yönlendirme Tablosu	33
Tablo 3.3 : D ₄ Düğümünün Güncellenmiş Yönlendirme Tablosu	33
Tablo 4.1 : W _H =70, W _L =15 Uyarlaması Ağ Değişim Raporu	42
Tablo 4.2 : W _H =80, W _L =15 Uyarlaması Ağ Değişim Raporu	42

ŞEKİL LİSTESİ

	<u>Sayfa Numarası</u>
Şekil 2.1 : Örnek Fiziksel Topoloji.....	7
Şekil 2.2 : Örnek Sanal Topoloji	8
Şekil 2.3 : Yeni Sanal Topoloji.....	10
Şekil 3.1 : Düğümlerdeki Arabirim Tablosunun Yapısı.....	19
Şekil 3.2 : Düğümlerdeki Yük Tablosunun Yapısı.....	19
Şekil 3.3 : Işıkyolu Ekleme Algoritması.....	21
Şekil 3.4 : Işıkyolu Ekleme İsteği Algoritması.....	23
Şekil 3.5 : Tablo Güncelleme İsteği Algoritması	23
Şekil 3.6 : Işıkyolu Silme Algoritması.....	24
Şekil 3.7 : Işıkyolu Silme İsteği Algoritması.....	24
Şekil 3.8 : Işıkyolu Ekleme Algoritmasının Akış Diyagramı.....	25
Şekil 3.9 : Işıkyolu Silme Algoritmasının Akış Diyagramı.....	26
Şekil 3.10 : Işıkyolu Ekleme Mesajları.....	27
Şekil 3.11 : Işıkyolu Silme Mesajı.....	27
Şekil 3.12 : Başlangıçtaki Sanal Topoloji.....	28
Şekil 3.13 : 1. Aşamadaki Sanal Topoloji	29
Şekil 3.14 : 2. Aşamadaki Sanal Topoloji	30
Şekil 3.15 : 3. Aşamadaki Sanal Topoloji	31
Şekil 4.1 : Simülasyonda Kullanılan NSFNET Topolojisi.....	34
Şekil 4.2 : Uyarlamasız Simülasyon EAY Sayıları	36
Şekil 4.3 : $W_H=80$, $W_L=20$ Uyarlamalı Simülasyonu EAY Sayıları	37
Şekil 4.4 : Uyarlamasız Simülasyon en Yüksek ve en Düşük Yük Talepleri.....	38
Şekil 4.5 : $W_H=80$, $W_L=20$ Simülasyonu en Yüksek ve en Düşük Yük Talepleri	38
Şekil 4.6 : Uyarlamasız Simülasyon Tıkanık Işıkyolu Sayıları	39
Şekil 4.7 : $W_H=80$, $W_L=20$ Simülasyonu Tıkanık Işıkyolu Sayıları.....	39
Şekil 4.8 : Uyarlamasız Simülasyon TAOAU Değerleri.....	40
Şekil 4.9 : $W_H=80$, $W_L=20$ Simülasyonu TAOAU Değerleri	40
Şekil 4.10 : Uyarlamasız Simülasyon Düşürülen Paket Oranları	41
Şekil 4.11 : $W_H=80$, $W_L=20$ Simülasyonu Düşürülen Paket Oranları.....	41
Şekil 4.12 : $W_H=70$, $W_L=15$ Uyarlaması EAY Sayıları	44
Şekil 4.13 : $W_H=80$, $W_L=15$ Uyarlaması EAY Sayıları	44
Şekil 4.14 : $W_H=70$, $W_L=15$ Uyarlaması en Yüksek ve en Düşük Yük Talepleri.....	45
Şekil 4.15 : $W_H=80$, $W_L=15$ Uyarlaması en Yüksek ve en Düşük Yük Talepleri.....	45
Şekil 4.16 : $W_H=70$, $W_L=15$ Uyarlaması TAOAU Değerleri	47
Şekil 4.17 : $W_H=80$, $W_L=15$ Uyarlaması TAOAU Değerleri.....	47
Şekil 4.18 : $W_H=70$, $W_L=15$ Uyarlaması Işıkyolu Ekleme Sayıları	48
Şekil 4.19 : $W_H=80$, $W_L=15$ Uyarlaması Işıkyolu Ekleme Sayıları	48
Şekil 4.20 : $W_H=70$, $W_L=15$ Uyarlaması Işıkyolu Silme Sayıları	49
Şekil 4.21 : $W_H=80$, $W_L=15$ Uyarlaması Işıkyolu Silme Sayıları	49
Şekil 4.22 : $W_H=70$, $W_L=15$ Uyarlaması Düşürülen Paket Oranları	50
Şekil 4.23 : $W_H=80$, $W_L=15$ Uyarlaması Düşürülen Paket Oranları	50

OPTİK WDM AĞLARINDA SANAL TOPOLOJİ UYARLAMASINA DAĞITILMIŞ BİR YAKLAŞIM

ÖZET

Teknoloji ilerledikçe hem internet ortamının kullanıcı sayısı artmakta hem de bu kullanıcılar internet üzerinden daha sık ve daha hacimli veri iletişimi yapmaktadır. Bu ihtiyacı karşılayabilmek için internet kapasitesinin her geçen gün artırılması gerekmektedir. Kapasite artışını sağlayabilmek için günümüzde kullanılan popüler yöntem, ışıkla veri transferidir. Işıkla veri transferini mümkün kılmak için optik ağlar kullanılmaktadır.

Optik ağlar çok yüksek veri iletim kapasitesine sahiptir. Bu yüksek kapasiteyi mümkün olduğu kadar verimli kullanabilmek için, bu ağlar birçok kaynağın kullanımına açılmaktadır. Farklı ve çok sayıda kaynağın optik ağı kullanabilmeleri için de çeşitli teknikler geliştirilmiştir. Tekniklerden biri, ışığı oluşturan dalgaboylarını kullanmaktır. Optik ağlarda, ışığın farklı dalgaboylarında farklı veri taşıyabilme tekniğinin adı WDM yani dalgaboyu bölümlenmeli çoğullamadır. Bu teknikle etkin ve sağlıklı bir iletişim sağlamak için, ağ üzerindeki trafik akışlarına uygun sanal topolojiler oluşturulmaktadır. Fakat başlangıç aşamasındaki trafik akışına göre oluşturulan sanal topoloji, zamanla trafik akışının değişmesiyle verimliliğini yitirmektedir. Verimliliğin devamını sağlayabilmek için de sanal topolojinin yeni trafik akışına göre yeniden düzenlenmesi ya da uyarlanması gerekmektedir.

Sanal topolojinin uyarlaması ile ilgili birçok çalışma yapılmış, fakat problem genelde merkezi olarak ele alınmıştır. Bu çalışmada, probleme dağıtılmış bir çözüm getirilmiş, uyarlama görevi merkezdeki bir algoritmadan alınarak, sanal topolojiyi oluşturan düğümlere verilmiştir.

Önerilen yeni dağıtılmış sanal topoloji uyarlaması yönteminin çalışabilirliğini ve verimliliğini denemek amacıyla, yazılımsal olarak bir simülasyon ortamı oluşturulmuştur. Bu ortamdan elde edilen sonuçlar, sanal topoloji düzenlemesi için hedef kabul edilen trafik ağırlıklı ortalama adım uzaklığının düşürüldüğünü, ağın

yükünün dengelenerek daha çok trafiğin ağı kullanmasına imkan sağlandığını ve yüklenmeden dolayı oluşan tıkanıklıkların çözülerek, uyarlama işleminin başarıyla gerçekleştirildiğini göstermiştir. Sonuç olarak, dağıtılmış yaklaşımın sanal topoloji uyarlaması probleminin çözümünde kullanılabileceği kanısına varılmıştır.



A DISTRIBUTED APPROACH FOR VIRTUAL TOPOLOGY ADAPTATION IN OPTICAL WDM NETWORKS

SUMMARY

As technology improves both the number of internet users and the amount of data transmitted over internet increases. To satisfy the needs of these users, the capacity of internet must be increased relatively. Nowadays the most popular data transmission media is the light. To transmit data over light, optical networks are established.

Optical networks has very large scale of capacity. To use these huge capacity effectively, optical networks sources are shared among numerous users or systems. To make these capacity available for numerous users, several techniques were developed. One of these techniques is using the wavelengths of light. In optical networks terminology, transmitting different data on different wavelengths of light is called as WDM as wavelenght division multiplexing. To supply an effective and accurate data transmission over light, virtual topologies are established according to the traffic flows in the network. As time passes and the traffic characteristics of the network changes, the virtual topology established according to the initial traffic conditions of the networks loses its effectiveness. To satisfy the continuity of effectiveness the virtual topology must be reconfigured or adapted according to the new traffic characteristics.

Many studies has been made over adaptation of virtual topology so far, but the problem was handled by using central solutions. In this study a distributed approach to this problem was introduced and the adaptation was performed by separate nodes instead of by a adaptation center.

To test the useability and effectiveness of proposed distributed approach in a simulation environment, a software testbed was written. According to the results of simulations, traffic weighted avarage hop distance, which is one of the major targets in reconfiguration process, is reduced, much more traffic is allowed to flow on the network by balancing the load of the network and the number of congestions

originated from the overloaded situation of the network is reduced. Finally it is concluded that a distributed approach can overcome the virtual topology reconfiguration problem.



1. GİRİŞ

Elektronik ortamda veri alışverişine olanak sağlayan internete duyulan ihtiyaç ve bunun neticesinde internet kullanımı gün geçtikçe artmaktadır. Artan ihtiyacı karşılamak için internet ortamının kapasitesi gün geçtikçe geliştirilmeye çalışılmaktadır. Bu çalışmalar neticesinde veri iletimi için yeni materyaller geliştirilmekte ve geliştirilen yeni materyalleri kullanabilen yeni aygıtlar üretilmektedir. Günümüzün bu materyaller arasında yükselen değerlerinden biri ışık, uygulanan yöntem de ışıkla veri iletimidir.

Işık ile veri iletimini gerçekleştirmek için fiber kablolar kullanılmaktadır. Fiber kablolar, 50 Tbps veri iletim kapasitesiyle iletişim için oldukça hacimli bir ortam sağlamaktadırlar. Sağlanan bu hacimli ortamın yanında, ışık verisi düşük güç gereksinimi, düşük bozulma oranı, düşük sinyal gücü azalması ile iletişimi daha sağlıklı ve kullanılabilir kılmaktadır.

Günümüzde yaygın olarak kullanılan metal teller üzerinden elektrik ile veri iletiminde, veriyi oluşturan 1 ve 0'lar, farklı gerilim değerleriyle ifade edilmektedir. Optik ortamda ise veriyi ifade edebilmek için lazer adı verilen ışık kaynakları kullanılmaktadır. Bu ışık kaynakları, ışığı çok kısa aralıklarla açıp kapayarak 1 ve 0'ları ifade edebilmektedirler. Alıcı tarafında ise bu ışık verisi foto algılayıcılar ile elektrik verisine çevrilmektedir.

Fiberler tek bir kaynağın hizmetine verilmek için oldukça fazla kapasiteye sahiptir. Böyle bir durumda fiberin kapasitesinin büyük bir kısmı ziyan olmuş olur. Bunun için fiberi birçok kaynağın kullanabilmesi adına çeşitli yöntemler geliştirilmiştir. Bunlardan birincisi (TDM) zaman ya da frekans bölümlenmeli çoğullama'dır [1]. Bu yöntemde farklı veriler belirli bir zaman aralığına bölünerek aynı ortamda-fiberde gönderilirler. Bu çoğullamada ana unsur veri yoğunluğudur. Düşük yoğunluklu farklı veriler fiber ortamına birleştirilerek çıkarılır. Böylece fiber kapasitesi daha etkin bir şekilde kullanılmış olur. Bu yöntem için bir standart belirlenerek kapasite, birbiriyle kesişmeyen frekans kanallarına bölünmüştür. Bir kanal en düşük OC-1 (54 Mbps) ile ifade edilir ve OC'nin ikili katları şeklinde artan değerler alabilir. Bu değerlere örnek

olarak OC-12 (648 Mbps) ve OC-48 (1944 Mbps) verilebilir. Fiberi farklı kaynakların kullanımına açmanın bir diğer yöntemi de (WDM) dalgaboyu bölümlmeli çoğullamadır. Bu yöntemde ışığın spektrumundan yararlanılarak, ışığın birbiriyle kesişmeyen farklı dalgaboylarında farklı veriler taşınır. TDM ağlarının WDM'e göre servis kalitesi çeşitliliği, esnek band genişliği ayarlaması gibi üstünlükleri olsa da, ticari gelişmelerden ve mevcut durumdaki kapasite ihtiyacını karşılayabildiğinden, günümüzde yaygın olarak WDM yöntemi kullanılmaktadır [2].

1.1 Problem ve Önerilen Çözüm

Optik WDM ağlarında, kaynakları etkin kullanmak için oluşturulan sanal topoloji, zamanla verimliliğini yitirmekte ve yeniden oluşturulması ya da düzenlenmesi gerekmektedir. Daha önce konu ile ilgili birçok çalışma yapılmış ve çeşitli yöntemler geliştirilmiştir. Ne var ki geliştirilen yöntemler genelde merkezi bir yaklaşımla çalışmaktadır. Bu yöntemler yeniden düzenleme ya da uyarılama için gerekli olan ağ verisini tek bir merkezde toplama ve düzenleme ile ilgili hesaplamaları yapıp yeni sanal topoloji oluşturulduğunda, bu değişiklikleri ağa yansıtmak esasına dayanmaktadır. Bu yöntemlerin avantajları olmasına karşın bazı dezavantajları da bulunmaktadır.

Merkezi yöntemlerin avantajları :

- Tüm ağı gözönüne alarak çözüm üretme.
- Elde ağla ilgili tüm veri olduğu için olabildiğince etkin bir çözüm üretme.

Merkezi yöntemlerin dezavantajları :

- Merkezde bir arıza durumunda, sanal topolojiyi yeniden düzenleme işlemlerinin devam ettirilememesi.
- Ağdan veri toplama gerekliliği.
- Fazla verinin getirdiği işlemsel yük.

Merkezi yöntemler, elindeki veriye dayanarak çok etkin çözümler üretebilir. Ancak merkezde bir arıza olması durumunda, ağ üzerindeki kontrol yitirilmiş olur ve bu arıza giderilene kadar, sanal topoloji düzenlenmesi yapılamayacağı için, ağda tıkanma ve EAY reddedilmesi gibi sorunlar yaşanabilir. Bu yöntemler, çözüm üretebilmek için tüm ağ verisini kullanırlar. Her ne kadar tüm veriyi kullanmak etkin çözümler

retilmesini saęlasa da, bu verileri aędan toplamanın bir maliyeti vardır. Toplanan veri eřitlilięi ne kadar artarsa, bu verilerin iletiminin aęa getireceęi yk de o kadar ok olur. Aynı zamanda, dinamik trafięe gre iřlem yapan algoritmalar iin veri toplama sıklıęı da nem tařıtmaktadır. Veriler ne kadar sık toplanırsa, aęda oluřan yk de o oranda artacaktır. Tm veriler merkeze geldikten sonra, bu verileri iřleyip sanal topolojiyi yeniden dzenlemekten sorumlu algoritma, ne kadar ok veri kullanırsa, iřlem sresi de o kadar uzayacaktır. Yoęun veriye sahip byke aęlarda, algoritmanın karmařıklıęı ne kadar yksek olursa, iřlem sresi de o kadar uzar. İřlem sresinin uzaması da dzenleme iřleminin gecikmesine yol aar. Bu gecikme neticesinde aęa zamanında mdahale edilemeyip, aęda yařanacak sorunlar engellenememiř olabilir.

Bu alıřmada sanal toplojinin yeniden dzenlenmesi iin daęıtılmıř bir yaklařım sunulmuřtur. Bu yaklařım, aę zerindeki dinamik trafik kořullarına gre sanal topolojiyi uyarlayarak dzenleme iřlemini gerekleřtirir. Uyarlama iřlemini dęmler tek bařlarına ynetmektedir. Her dęm kendi yerel verilerini kullanarak uyarlama yapılıp yapılmayacaęına karar verir ve yerel zmler reterek iřlemleri gerekleřtirir. İřlemlerde sadece yerel veriler kullanıldıęından, az miktarda veri zerinde iřlem yapılmaktadır. Uyarlama iin kullanılan algoritma da basit olduęundan, zm, merkezi sistemlere gre ok daha kısa srede elde edilir. Ayrıca iřlemlerde yerel verinin kullanılması, aędan veri toplama ihtiyaını da ortadan kaldırmaktadır.

nerilen daęıtılmıř yntemin merkezi ynteme gre bazı avantajlarının olmasının yanında bazı dezavantajları da bulunmaktadır. Bunlardan ilki, bařlangı ařamasında bir temel sanal topoloji oluřturulması ve oluřturulan bu sanal topolojiye ait ıřık yollarının silinememesidir. Bu sanal topolojinin veriminin azalması durumunda, ıřık yolları ekleyip, eklenen ıřık yollarının silinmesi aracılıęıyla uyarlama yapılmaktadır. İkinci dezavantaj ise, sadece yerel verilerle yerel zmler retildięinden, algoritmanın etkinlięi merkezi sistemlere gre dřktr. Bu yntemde, daha fazla miktarda ıřık yolu ekleme ve silme iřlemleri yapılmakta, bunun sonucunda da, bu ıřık yollarını kullanan trafikler daha fazla yeniden ynlendirilmeye tabii tutulmaktadır. Neticede rahatsız edilen trafik miktarı, merkezi sistemlere gre daha fazla olmaktadır.

1.2 Bölümlerin Özetleri

Bu çalışma toplam beş bölümden oluşmaktadır. Bu bölümü takip eden dört bölümün özeti aşağıdaki gibidir.

- Bölüm 2’de, optik WDM ağları hakkında temel bilgiler kısaca verilmiş, bu çalışma ile ilgili kavramlar açıklanmıştır.
- Bölüm 3’te, yapılan çalışma ayrıntılı olarak anlatılmış, örneklerle açıklanmıştır.
- Bölüm 4’te, bu çalışmanın sınıandığı simülasyon ortamı anlatılmış, simülasyonlardan elde edilen sonuçlar grafikler aracılığıya aktarılmıştır.
- Bölüm 5’te, bu çalışmadan çıkarılan sonuçlar belirtilmiş, etkili ve verimsiz tarafları vurgulanmıştır.

2. OPTİK WDM AĞLARI

Optik ağlarda veri ışık ile taşınmaktadır. Bu yüzden düğümler arasında ışık ile haberleşmeyi mümkün kılmak için, taşıyıcı olarak elektrik kabloları yerine fiber kablolardan yararlanılmaktadır. Fiber kablolarda iletim için, veriler bir ışık kaynağı aracılığıyla kodlanarak fiber ortamına aktarılır. Alıcı tarafında ise bu ışık sinyallerini algılayabilen algılayıcılar ile veri ışık kodundan elektrik koduna çevrilir. Günümüzde ışık teknolojisi henüz gelişme aşamasında olduğu için, ışık verisini işleyen aygıtlar kullanılamamaktadır. Bu yüzden veri, ışık ortamı olan fibere aktarılırken elektrikten optik sinyallere çevrilir. Alıcı tarafında ise optik sinyaller tekrar elektrik sinyallerine çevrilerek veri işlenebilir hale getirilir.

Fiberler temelde iki tiptir. Bunlardan birincisi bir anda tek ışık sinyalini taşıyabilen fiberler ve diğeri ise bir anda birden çok ışık sinyalini taşıyabilen fiberlerdir. Aynı anda birden fazla ışık sinyalini taşıyabilen fiberler, ışık sinyalinin gücünü çabuk yitirdiklerinden kısa mesafelerdeki iletişim için kullanılırlar. Bunun yanında, bir anda sadece bir ışık sinyali taşıyabilen fiberler, ışık sinyalini çok uzak mesafelere iletebilirler. Ayrıca bu tip fiberler, ışığın geniş bir spektrumunu kullanmaya olanak sağladıklarından, yüksek veri alışverişi yapılan ağlarda kullanılabilirler.

2.1 WDM - Dalgaboyu bölümlmeli çoğullama

Bir anda tek bir ışık sinyalinin iletimine olanak sağlayan fiberlerin kapasitesi yaklaşık 50 Tbps'dir. Böylesi yüksek bir hızda veri üreten ve işeyebilen aygıtlar henüz geliştirilememiştir. Elektrikle çalışan aygıtların bu kapasiteyi olabildiğince kullanabilmeleri adına, kapasiteyi aynı anda birden çok kullanıcının hizmetine sunmak için birtakım teknolojiler geliştirilmiştir. Bu teknolojilerden biri WDM'dir. Bu teknoloji sayesinde birbiriyle kesişmeyen farklı dalgaboylarında, farklı trafiklerin iletimi gerçekleştirilir [3]. Bir dalgaboyu yaklaşık 40 Gbps kapasiteye sahiptir. Örnek olarak, 160 tane dalgaboyunu destekleyen bir fiberde, toplam 6.4 Tbps veri iletimi sağlanabilir .

Optik WDM ağlarında ışık sinyali kullanarak veriyi bir noktadan diğerine taşımak için bazı optik ağ ekipmanları kullanılır. Bunlar optik güçlendiriciler, dalgaboyu çeviricileri, dalgaboyu ekle/çıkar çoğullayıcılar ve optik bağdaştırıcılardır. Optik güçlendiriciler, fiberde ilerlemesi sırasında gücünü yitiren ışık sinyalinin gücünü arttırmak için kullanılır. Böylece veri daha uzak mesafelere taşınabilmektedir. Dalgaboyu çeviricileri, bir dalgaboyundaki veriyi başka bir dalgaboyuna aktarmak için kullanılır. Yönlendiricilerden, bir dalgaboyundaki verinin başka bir dalgaboyundan yoluna devam etmesi gerektiğinde faydalanılır. Dalgaboyu ekle/çıkar çoğullayıcılar ise, veriyi fiberdeki belirli bir dalgaboyundan göndermek ve belirli bir dalgaboyundan çekmek için kullanılır. Optik bağdaştırıcılar, iki farklı fiberdeki iki dalgaboyunu birbirine bağlamak için kullanılır. İki tip bağdaştırıcı vardır. Bunlardan birincisi, iki fiberdeki sadece aynı dalgaboylarını birbirine bağlayabilirken diğeri farklı dalgaboylarını da birbirine bağlayabilmektedir. Optik bağdaştırıcılar aracılığıyla, veri başlangıçtan bitişe kadar ışıkyolundan çıkmadan hızlı bir şekilde iletebilmektedir [4].

2.2 Sanal Topoloji

Optik WDM ağlarında veri başlangıç düğümünden bitiş düğümüne giderken arada başka düğümlerden de geçmek zorundaysa, bu düğümlerde bir yönlendirme işleminden geçirilir. Yönlendirme işlemini gerçekleştirecek aygıtlar, elektrik verisini işleyebildiklerinden, fiberden ışık halinde gelen verinin elektriğe çevrilmesi gerekmektedir. Verinin yönlendirilmesi tamamlanıp, düğümden çıkması gereken çıkış belirlendiğinde, veriyi tekrar ışık olarak kodlamak gereklidir. Işıktan elektriğe ve elektrikten ışığa olan bu çevrimler, maliyetli işlemler olduklarından, çevrimler sırasında gecikmeye neden olurlar. Yönlendirme sayısı arttıkça da gecikme değerleri artar. Bu gecikmeleri azaltmak için yönlendirme sayılarını azaltmak gereklidir. Bunun için de optik WDM ağlarında ışıkyolu kavramı geliştirilmiştir.

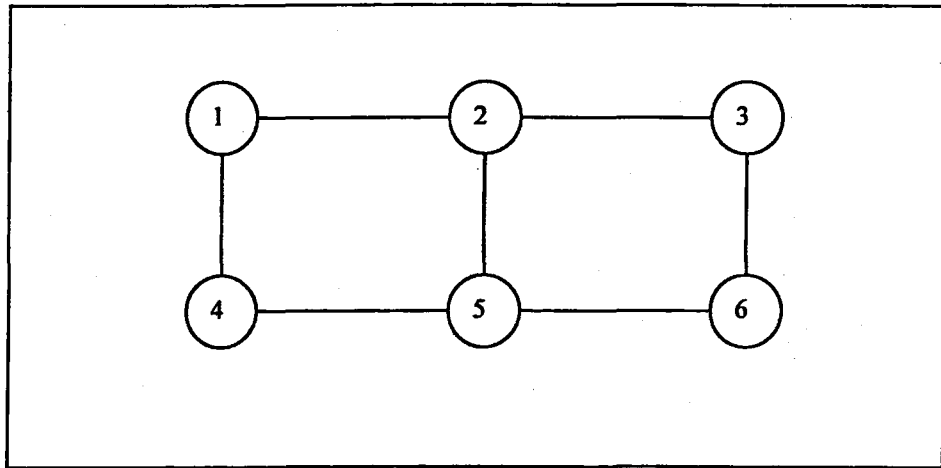
İşıkyolu optik WDM ağındaki trafik akışlarına göre oluşturulur ve trafiğin başlangıç ve bitiş düğümleri arasında tek adımlık bir iletişim sağlar. Ancak düğümlerdeki alıcı ve verici sayılarının ve de fiberlerdeki dalgaboylarının sayılarının sınırlı olmasından ötürü, tüm başlangıç ve bitiş düğümleri için ışıkyolu kurulmaz ve çok adımlı iletim yapılmak zorunda kalınır [5]. Işıkyolu, bir başlangıç düğümünden başlayıp birkaç fiber ve düğüm geçerek bitiş düğümüne ulaşabildiği gibi, trafik akışına göre tek bir

fiber geçerek de bitişe ulaşabilir. Işıkyolu, bir fiberden diğer bir fibere geçerken, bu iki fiberdeki bağlantıyı optik bağdaştırıcılar sağlar. Dalgaboyu çevrimi yapamayan optik bağdaştırıcılı bir ağda, ışıkyolu, başlangıç düğümünden bitiş düğümüne kadar aynı dalgaboyunu kullanır. Dalgaboyu çevrimi yapabilen optik bağdaştırıcılarda bu kısıt ortadan kalkar; Işıkyolu, farklı fiberlerde farklı dalgaboylarını kullanabilir. Dalgaboyu çevrimi yapabilen optik bağdaştırıcılar ile ağ kaynakları daha verimli kullanılır. Diğer bağdaştırıcıda, bir ışıkyolunu kurabilmek için, ışıkyolunun geçeceği tüm fiberlerde aynı dalgaboyunun boş olması kısıtı aranır. Bu da bazen ışıkyolunun kurulamaması sonucunu doğurabilmektedir.

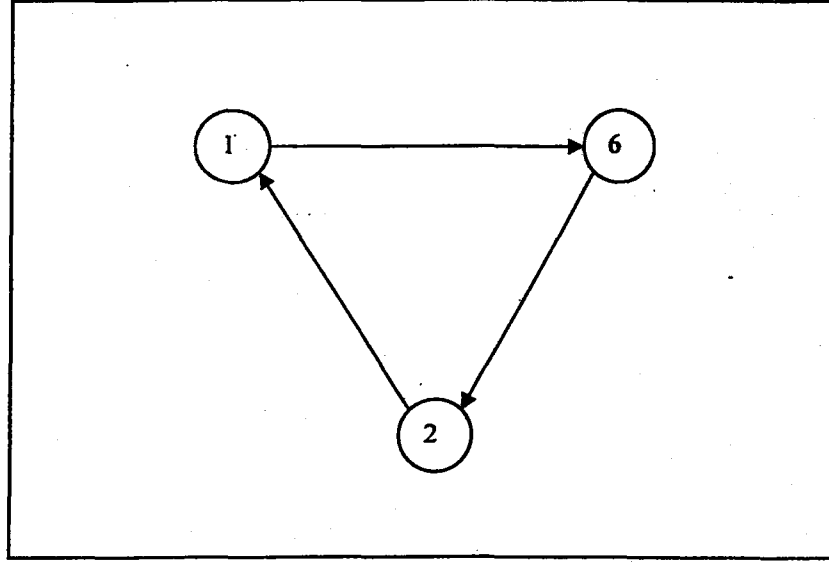
Kontrol düzlemi için geliştirilen GMPLS protokolü ile, optik bağdaştırıcılarda ışıkyolu anahtarlarının yanında, ışıkyolu grubu anahtarlama ve fiber anahtarlama da yapılabilmektedir [6]. Anahtarlama yapılan bu genellemeler ile, optik bağdaştırıcıların yükü azaltılmaktadır.

Optik WDM ağında bir ışıkyolu kurulduğu zaman, ışıkyolunun başlangıç ve bitiş düğümü sanal düğüm adını alır. Sanal düğümlerden ve ışıkyollarından oluşmuş olan yapıya, sanal topoloji adı verilir. Ağ trafiği, kurulan bu sanal topoloji üzerinden iletilir.

Şekil 2.1’de örnek bir fiziksel topoloji görülmektedir. Bu topolojideki trafik akışının 1’den 6. düğüme, 6’dan 2. düğüme, ve 2’den 1. düğüme olduğu varsayılırsa, ışıkyolları bu akışlar doğrultusunda kurulmalıdır. Buna tanımlara göre sanal topoloji Şekil 2.2’deki gibi olmalıdır.



Şekil 2.1 : Örnek Fiziksel Topoloji



Şekil 2.2 : Örnek Sanal Topoloji

Veri yoğunluğunun fazla olmadığı trafikler için ağ kaynaklarını kullanarak bir ışık yolu kurulmayabilir. Böyle trafikler için eğer sanal topolojide trafiğin başlangıç düğümünden bitiş düğümüne bir ışık yolu güzergahı varsa, trafik mevcut topoloji üzerinden iletilir. Bu durumda trafik, varış düğümüne birden fazla adımda gidebilir ve her adımda bir yönlendirmeye tabii tutulur. Trafiğin bu şekilde iletilmesinde dikkat edilmesi gereken nokta, trafiğin yükünün, geçtiği ışık yollarının kapasitesinin dolmasına ve hatta bu kapasiteyi aşarak bir tıkanmaya sebep olmamasıdır. Bu durumda hem bu trafik hem de geçtiği ışık yollarını kullanan diğer trafikler, tıkanmadan dolayı veri kaybına uğrarlar. Örnek olarak Şekil 2.2'deki sanal topoloji gözönüne alınırsa, 2. düğümden 6. düğüme doğrudan bir ışık yolu yoktur ama 1. düğüm üzerinden yönlendirilerek 2. düğümden 6. düğüme olan bir trafik akışının iletimi sağlanabilir.

Optik WDM ağları ilk geliştirildiklerinde bir ışık yolunu sadece bir trafik kullanabiliyordu. Ancak bu trafiğin veri yoğunluğu, ışık yolunun kapasitesinin çok altında kalabiliyordu. Bu boş olan kapasiteyi kullanabilmek için düşük veri yoğunluğuna sahip trafikler, ışık yollarının kapasitesini aşmayacak şekilde TDM ile birleştirilerek ışık yollarına çıkarılması teknolojisi geliştirildi ve bu teknolojiye grooming adı verildi [7].

Grooming'ın uygulanabilmesi için optik ağda WDM ile TDM teknolojilerini birarada kullanabilen optik bağdaştırıcıların kullanılması gerekmektedir. Bu çalışmada farklı

trafiklerin aynı ışıkyolu üzerinden iletilmesi de grooming yetenekli optik bağdaştırıcılar ile sağlanmaktadır.

2.2 Yönlendirme

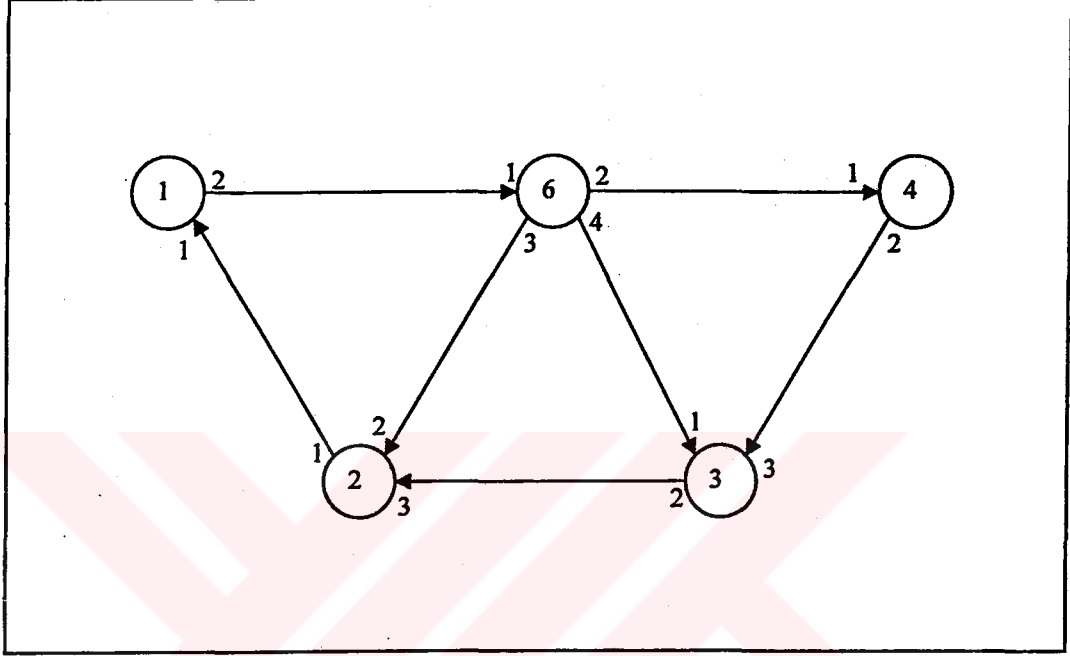
Optik ağdaki mevcut sanal topoloji üzerinde her trafik akışı tek bir adımda varışa ulaşamayabilmektedir. Bu durumda trafik akışının iletileceği bir sonraki ışıkyolu, trafiğin geldiği ışıkyolundan çıktıktan sonra bulunduğu düğümün yönlendiricisinde belirlenmektedir. Optik ağlarda yönlendirme temelde iki şekilde yapılmaktadır. Bunlardan birincisi verinin varış adresi esaslı yönlendirilmesi, ikincisi de etiket anahtarlamalı yollar üzerinden yönlendirilmesidir [8].

Varış adresi esaslı yönlendirmede, ağın kurulumu aşamasında tüm düğümlerin yönlendirme tabloları oluşturulur. Bu tablolarda düğümün her çıkışı için, bu çıkışa yönlendirilecek olan adres uzaylarının maskeleri bulunur. Gönderilecek verinin başlığına gideceği düğümün adres bilgisi yazılarak ağa çıkarılır. Veri yönlendirilmek üzere bir düğüme geldiğinde, varış adresi tablodaki maskelerle karşılaştırılır. Uygun maske bulunduğunda, veri, maskenin ilişkili olduğu çıkışa yönlendirilir. Bu yönlendirme dinamik bir yapıya sahiptir. Bir veri paketi varışa farklı güzergahlardan ulaşabilir. Ağda yeni bir başlangıç-varış düğüm çifti arasında bir trafik akışı başlatmak için ayrı bir bağlantı kurulumu yapılmasına gerek yoktur. IP yönlendirme bu şekilde yapılmaktadır.

Etiket anahtarlamalı yönlendirmede, trafik gönderilmeye başlamadan önce başlangıç ve bitiş düğümleri arasında bir yol kurulur. Yol kurma işlemi için, öncelikle yolun geçeceği ara düğümler belirlenir. Sonra bu düğümler kendi aralarındaki sinyalleşme ile, trafiğin hangi ışıkyolunda hangi etiketle işaretleneceğine karar verirler. Bu işlem sırasında her düğüm, kendine gelen ışıkyolunda verinin taşınması gereken etikete karar verir. Aynı zamanda da kendi yönlendirme tablolarını bu etiket değerlerine göre düzenlerler. Trafiğin geçeceği tüm düğümlerde bu işlemler bitirildikten sonra, başlangıç ve bitiş düğümü arasında etiket anahtarlamalı bir yol kurulmuş olur.

Şekil 2.2'deki sanal topolojiyi, ağda 6. düğümden 3. düğüme, 4. düğümden 3. düğüme ve 3. düğümden 2. düğüme trafiklerin de eklendiğini ve her bir trafik için de bir ışıkyolu kurulduğunu varsayarak güncellersek, Şekil 2.3'deki sanal topolojiyi elde etmiş oluruz. Sanal ağ mevcut haliyle her bir trafik için bir ışıkyolu içerdiğinden, trafiklerin hiçbirini yönlendirmeye tabii tutulmadan varış düğümlerine

ulaşmaktadır. Bu trafiklere 1. düğümünden 3. düğüme ve 6. düğümünden 1. düğüme olan trafikler eklenirse ve bu trafikler için mevcut sanal topolojiyi kullanılırsa, ara düğümlerde yönlendirme gerekli olacaktır. 1-3 trafiği için 6. düğüm üzerinden ve 6-1 trafiği de 2. düğüm üzerinden etiket anahtarlamalı yollar kurulduğu düşünülecek olursa, bu durumda örnek etiketler ve yönlendirme tabloları Tablo 2.1, Tablo 2.2, Tablo 2.3, Tablo 2.4'deki gibi olur.



Şekil 2.3 : Yeni Sanal Topoloji

Tablolarda çıkış arabirim ve etiket çifti boş olan satırlar, trafiğin bu tablonun düğümünde sonlandığı anlamına gelir. Örnek olarak 3. düğümün tablosunun 1,2 ve 3. satırları verilebilir. Giriş arabirim ve etiket çifti boş olan satırlar, ilgili trafiklerin bu düğümden kaynaklandığı anlamına gelir. Buna da örnek olarak, 2. düğümün 1. satırı verilebilir. Hem giriş hem de çıkış değerleri dolu olan satırlar için, trafiğin bu düğümden yönlendirildiği anlamı çıkarılır. 6. düğümün tablosunun 5. satırı da buna örnek verilebilir. Bu tablolara göre 6-1 trafiği, 6. düğümün 3. arabiriminden 7 etiketiyle 2. düğüme gönderilir. 2. Düğüme 7 etiketiyle gelen trafik, 9 etiketiyle 1. arabirimden 1. düğüme yönlendirilir.

Tablo 2.1 : Düzüm 3'ün Yönlendirme Tablosu

Sıra No.	Giriş		Çıkış	
	Arabirim	Etiket	Arabirim	Etiket
1	1	3		
2	3	11		
3	4	6		
4			2	12

Tablo 2.2 : Düzüm 2'nin Yönlendirme Tablosu

Sıra No.	Giriş		Çıkış	
	Arabirim	Etiket	Arabirim	Etiket
1			1	8
2	2	7	1	9
3	2	2		
4	3	10		

Tablo 2.3 : Düzüm 1'in Yönlendirme Tablosu

Sıra No.	Giriş		Çıkış	
	Arabirim	Etiket	Arabirim	Etiket
1			2	4
2			2	5
3	1	8		
4	1	9		

Tablo 2.4 : Düğüm 6'nın Yönlendirme Tablosu

Sıra No.	Giriş		Çıkış	
	Arabirim	Etiket	Arabirim	Etiket
1			2	1
2			3	2
3			4	3
4	1	4		
5	1	5	4	6
6			3	7

2.3 Sanal Topolojiyi yeniden düzenleme gereksinimi

Optik WDM ağında başlangıç trafik koşullarına göre oluşturulmuş sanal topoloji, zamanla trafik karakteristiğinin değişmesiyle verimliliğini yitirmeye başlar. Verimsizlik bir kaç maddede başgösterir.

- Işıkyollarının aşırı yüklenmesi
- Işıkyollarının kapasitesinin çok altında kullanılması
- Trafiklerin adım sayısının artması

Trafik karakteristiğinin değişmesiyle, bir ışıkyolunda kapasitesini zorlayacak kadar çok trafik geçmeye başlayabilir. Ya da bazı yeni yoğun trafikler, varışa ulaşmak için birden fazla ışıkyolu kullanmak zorunda kalabilirler. Bu trafikler, geçtikleri her ışıkyolunda yüklenmeye sebep olurlar ve sayıca çok ve/veya yoğun trafiklerin kullandığı bir ışıkyolundan da geçiyorlarsa, bu ışıkyolunda tıkanmaya sebep olabilirler. Tüm bu durumlar sonucunda, ağda bazı trafikler tıkanmadan dolayı veri kaybına uğrayabilir. Aynı zamanda, ağda yeni kurulacak bağlantılar da yüklenmeden ötürü reddedilebilirler.

Trafik karakteristiğinin değişmesinin bir olumsuz sonucu da, zamanla bazı ışıkyollarını kullanan trafiklerin kesilmesi ya da yoğunluğunun azalması nedeniyle

bu ıřıkıyollarının kapasitesinin olduka az bir kısmının kullanılır hale gelmesidir. Bu durum sistem kaynaklarının bořa kullanılmasına neden olur. Aėda bu tip ıřıkıyolları, zamanla ihtiyaı olan ıřıkıyollarının kurulması iin yeterli kaynak bulunamamasına sebep olur. ıřıkıyolu kurulması gereken zamanda kurulamaması da aėı kullanan trafiklerin isteklerini karřılayamama ve mevcut tıkanıklıkların giderilememesi sonucunu doėurur.

Trafik akıřları gznne alınarak oluřturulan sanal topolojide, trafiklerin mmkn olan en az sayıda ynlendirmeye tabii tutulmalarına alıřılır. nk trafik ne kadar ok ynlendirilirse, ynlendirme iřleminin yapıldıėı dėmlerde o kadar ok optik-elektrik-optik evriminin yapılması gerekecek, bu da gecikmeyi arttıracaktır. Zamanla trafik karakteristiėinin deėiřmesiyle, bu trafiklere uygun olmayan sanal topolojide, ok ynlendirme olacak, bu da hem ortalama trafik gecikmesini arttıracak, hem de trafiklerin ok fazla ıřıkıyolu iřgal etmesi sebebiyle de kaynaklar ařırı tketelecektir.

Aėda zamanla verimsizliėe sebep olan nedenler yznden, sanal aė belirli zamanlarda yeniden dzenlenmeli ve gncel trafik karakteristiklerine uygun hale getirilmedilir. Bu dzenleme temelde iki řekilde yapılmaktadır. Bunlardan birincisi, belirli aralıklarla tm sanal topolojiyi yeni řartlara gre yeniden dzenlemek, ikincisi ise dinamik trafiėe gre sanal topolojiyi uyarlamaktır. Birinci yntemde tm aė gznne alınıp deėiřtirilir, ikincisinde ise sanal topolojide ufak deėiřikliklerle yeni trafiklere uyumlu hale getirilir.

Optik WDM aėlarında optik alıcılar, vericiler, fiberlerdeki dalgaboyları, optik baėdařtırıcılar gibi ekipmanlar pahalı ve pahalı olduėu iin de sınırlı olduėundan, bu ekipmanlar etkin bir řekilde kullanılmalıdır. Aėda hangi trafikler iin ıřıkıyolu kurulacaėı ya da daha genel deyiřle ıřıkıyollarının ne řekilde oluřturulacaėı, bir konfigrasyon ve performans problemidir. Bu problemi zezebilmek iin algoritmalar ve yaklařımlar geliřtirilmiřtir.

2.4 Konu ile ilgili daha nce yapılan alıřmalar

Sanal topolojinin dzenlenmesi ile ilgili literatrde birok alıřma mevcuttur.

[9]'daki alıřmada, aėdaki ıřıkıyolu ykleri ve trafiklerin ykleri belirli aralıklarla bir merkeze iletilir. Belirli aralıklarla merkezde alıřan bir uyarlama algoritması, tm

ışık yollarını tarayarak en yüksek yüklü ışık yolunu bulur. Bu ışık yolunun yükü belirlenen bir yüksek eşik değerini geçtiğinde, bu ışık yolunu kullanan trafiklerden en yüklü olanı için bir ışık yolu kurulur. Böylece yüklü olan ışık yolunun yükü azaltılmış, ağ rahatlatılmış olur. Çalışan algoritma eğer bir ışık yolu ekleme işlemi yapmamışsa, bu sefer ışık yollarını tarayarak en az yüklü olanı bulur. Eğer bu ışık yolunun yükü belirlenmiş bir düşük eşik değerinin altındaysa, bu ışık yolu silinir ve bu ışık yolunu kullanan trafikler varsa, alternatif güzergahlara yönlendirilir. Algoritmanın belirli aralıklarla çalışarak ve her seferinde sanal topolojiyi etkileyecek sadece bir değişiklik yapıp ağdaki bağlantıları en az derecede rahatsız ederek, sanal topolojiyi yeni trafiklere uyumlu hale getirir.

[10]'daki çalışmada, sanal topoloji yeniden düzenlemesine farklı bir açıdan bakılmıştır. Bu çalışmada optik veri taşımasında karşılaşılan tıkanıkları çözmek yerine, düğümlerde meydana gelebilen optik veri işleminde karşılaşılan tıkanıklık ele alınmıştır. Günümüz optik ortamında, veri taşıma sırasında tıkanıklığın meydana gelme olasılığının düşük olduğu belirtilmiştir. Düğümlerde meydana gelebilecek tıkanıklıklardan kaynaklanan veri kayıpları adreslenmiş, düğümlerde gerçekleşen optik-elektrik-optik çevriminden kaynaklanan gecikmeleri azaltacak yeni bir yeniden düzenleme yöntemi sunulmuştur.

[11]'deki çalışmada, açık olan 3 farklı konu aydınlatılmaya çalışılmıştır. Bu konulardan birincisi, sanal topolojiyi ne kadar sıklıkta yeniden düzenlemek gerektiğidir. Bunun için Markov karar verme süreci teorisine dayalı bir formülasyon geliştirilmiştir. Bunun dışında yeniden düzenleme işleminin nasıl planlanması gerektiğine ait parametrik yöntemler geliştirilmiş ve yeniden düzenlemenin sanal topolojiye sağladığı yarar, ölçülebilir değerler ile ifade edilmiştir.

[12]'deki çalışmada geliştirilen yöntemde, zamanla değişen trafik karakteristikleri için mevcut sanal topolojiden yeni sanal topolojiler üretilmiş ve bunların içinden en uygun olanı, eski sanal topoloji ile değiştirilmiştir. Hesaplama sırasında yeniden düzenleme miktarı ve eniyilemeye çalışılan hedef arasındaki karşıtlıklar da hesaba dahil edilmiştir.

[13]'te optik geniş alan ağlarını yeniden düzenleme için çeşitli algoritmalar sunulmuş ve bu algoritmaların karşılaştırmalı değerlendirmeleri yapılmıştır. Öne sürülen algoritmalarda yeniden düzenleme sırasında mümkün olduğu kadar ağın kesintiye

uęramamasına alıřılmıştır. Varılan sonulara gre basit algoritmaların getirdięi iřlemsel yk az olmasına karřın verimsiz sonular retmiř, iyi sonu veren algoritmaların da iřlemsel ykleri dolayısıyla byk aęlara uygulanamayacaęı kanısına varılmıştır.



3. DAĞITILMIŞ SANAL TOPOLOJİ UYARLAMASI

3.1 Yönlendirme

Problem çözümünde ağdaki veri yönlendirmesinin etiket anahtarlama olduğu kabul edilmiştir. Varış adresi esaslı yönlendirmede, yönlendirmenin yapıldığı düğümde, trafiğin hangi girişlerden geldiğine ilişkin bir bilgi tutulmaz. Yönlendirmede esas olan, verinin varış adresidir, kaynak adresine bakılmaz. Gelen verinin varış adresi, düğümdeki maskeleme tablosundaki değerlerle karşılaştırılarak hangi çıkışa yönlendirileceği kararı verilir. Bu sebeplerden, düğüm üzerinden geçen verilerin geliş-gidiş güzergahı belirlenemeyeceğinden, bu tip yönlendirme öne sürülen uyarlama yöntemine uygun değildir.

Etiket anahtarlama yönlendirmede, trafik gönderilmeden önce sanal topolojideki başlangıç ve bitiş düğümleri arasındaki sinyalleşme ile EAY adı verilen bir bağlantı kurulur. Bu kurulum aşamasında, trafiğin izleyeceği ışıkyolu güzergahı belirlenerek, bu güzergah üzerindeki sanal düğümlerin yönlendirme tabloları, sinyalleşme protokolüyle belirlenen etiket değerlerine göre güncellenir. Bu yönlendirme sayesinde düğümler, üzerlerinden geçen trafikler hakkında uyarlama için gerekli temel bilgiye sahip olmaktadır. Bu trafikler hakkında yönlendirme ile ilgili bilinmesi gereken temel kavramlar, trafiklerin düğümün hangi girişlerinden geldikleri ve hangi çıkışlardan yollarına devam ettikleridir. Bu bilgilerin dışında trafiklerin detaylı karakteristikleri de tutularak, optik ağlar üzerinde akan etiket anahtarlama trafikler için trafik mühendisliği yapmak da mümkündür [14]. EAY'ların kurulumu aşamasında trafik mühendisliğinin kullanılması, uyarlamaya olan ihtiyacı da azaltacaktır.

Tablo 2.1'de örnek bir yönlendirme tablosu görülmektedir. Bu tabloda, etiket anahtarlama kullanılan giriş ve çıkış noktaları, arabirim ile ifade edilmiştir. Arabirim, ilgili EAY için düğüme bağlı hangi fiberdeki hangi dalgaboyu bilgisidir. Aynı dalgaboyu, birden fazla fiberde bulunduğu ve bir düğüme birden fazla fiber bağlı olabileceği için, yönlendirmede bu iki değerin de bilinmesi gereklidir.

3.2 Kısıtlar ve Kab ller

Ağda yeni bir ışıkyolunun kurulması ya da etiket anahtarlamalı yolların izlemesi gereken ışıkyollarının belirlenmesi için t m ağın bilgisine ihtiya  duyulmaktadır. Bu bilgilere  rnek olarak t m ağıdaki d ğ mler, bu d ğ mler arasındaki fiberler, bu fiberlerdeki dalgaboyu sayıları, y kleri a ısından kullanılmaya elveriřli dalgaboyları, ışıkyolu oluřturmak i in d ğ mlerdeki alıcı ve verici sayıları verilebilir. Dağıtılmış sanal topoloji uyarlaması ise, yerel verilerle birbirinden bağımsız  alıřan d ğ mlerle ger ekleřtirilmektedir. Bu sebepten, ışıkyolu eklenmesi ve etiket anahtarlamalı bir yolun kurulması i in, bu hizmetleri saėlayan arabirimlerden faydalanıldığı varsayılmıştır. Iřıkyolları kurulumuyla ilgili optik aė kontrol mekanizmasına [15]'teki  alıřma  rnek verilebilir.

Iřıkyolu kurma iřlemine ger ekleřtiren arabirimin dikkat ettiėi ilk veri, ışıkyolunun kurulacaėı bařlangı  d ğ m ndeki boř verici ile bitiř d ğ m ndeki boř alıcı sayılarıdır. Boř alıcı ya da verici bulunamadığı durumda ışıkyolu kurulamaz. Arabirim, boř alıcı ve verici bulduktan sonra, ışıkyolunun bařlangı  ve bitiř d ğ mleri arasındaki en kısa fiziksel yolu kullanarak ışıkyolunu kurmaya  alıřır. İkinci olarak dikkat edilen nokta, ışıkyolunun ge eceėi fiberlerdeki kullanılmayan dalgaboyu sayısıdır. Fiberlerdeki uygun dalgaboyları kullanılarak ışıkyolu ağın  zelliėine g re iki farklı řekilde kurulabilmektedir. Bunlardan birincisi, dalgaboyu  evrimine izin vermeyen optik baėdařtırıcılarla kurulan ışıkyoludur. Iřıkyolunun ge eceėi bařlangı  ve bitiř d ğ m n n arasında kalan d ğ mlerde optik baėdařtırıcılar kullanılır. Bu optik baėdařtırıcılar, d ğ me giren fiberdeki dalgaboyuyla, d ğ mden  ıkan bir fiberdeki dalgaboyunu anahtarlama yoluyla birbirine baėlar. Bu anahtarlama iřlemi sırasında giriř fiberindeki dalgaboyu,  ıkıř fiberinde bařka bir dalgaboyuyla birleřtirilebilir. Bu birleřtirme sırasında iki dalgaboyu arasında bir  evrim iřlemi olmaktadır. Bazı d ğ mler ise bu  zellikten yoksundur. Bu tip optik baėdařtırıcılara sahip d ğ mlerde, giriř ve  ıkıř fiberinde aynı dalgaboyunun kullanılması zorunludur. Bu y zden ışıkyolu, bařlangı  d ğ m nden varıř d ğ m  arasındaki fiberlerde hep aynı dalgaboyunu kullanır. Bařka bir deyiřle ışıkyolu tek bir dalgaboyu  zerine kurulur.

Ağda bulunduėu varsayılan optik baėdařtırıcı, dalgaboyu  evrimi yapabilen cinstendir. Bu y zden ışıkyolunun kurulabilmesi i in ışıkyolunun ge eceėi her fiberde en az bir kullanılmayan dalgaboyunun olması yeterlidir.

Arabirim, ıřık yolu kurulacak iki dđđđm arasındaki en kısa fiziksel yolu esas alarak ıřık yolunu kurmaya alıřır. Bu yol  zerindeki fiberlerden en az birinde uygun dalgalı boyu bulunamazsa, bu sefer bir sonraki en kısa g zergah  zerinden iřlem tekrarlanır. 5 İterasyon sonunda uygun bir g zergah bulunamazsa, ıřık yolunun kurulamadıđı varsayılır. Bu kısıtlama, bir ıřık yolu kurulumu iin ok fazla miktarda fiberi kullanma ihtimalini ortadan kaldırmak iin yapılmıřtır.  nk  her iterasyon sonunda kullanılan g zergah, bir  ncekinden daha uzundur ve daha fazla fiber kullanımına neden olmaktadır. Fazla fiber ve dolayısıyla dalgalı boyu kullanımı, y kl  bir ađda kaynakları artan bir hızla t keteceđinden b yle bir kısıtlama getirilmiřtir.

Uyarılama y nteminde kullanılacak bařka bir arabirim ise etiket anahtarlamaalı bađlantıların izlemesi gereken g zergahı belirlemekten sorumludur. Bu arabirim sadece uyarılamanın alıřmasını g zlemlemek amacıyla oluřturulan EAY'ların kurulmasıyla ilgilenmektedir. Arabirim, sanal topolojiyi esas alarak bařlangı ve bitiř d đ mleri arasındaki en kısa ıřık yolu g zergahını bulur. Bu noktadan sonra kurulacak etiket anahtarlamaalı bađlantının kullanmak istediđi bant geniřliđi bilgisine ihtiya duyulur. Kullanılacak bant geniřliđi, belirlenen g zergahta yeralan ıřık yollarının en az birinin kullanılmayan kapasitesinden fazla ise bu g zergahta EAY kurulamayacađına karar verilerek, bir sonraki en kısa ıřık yolu g zergahı iin iřlem tekrarlanır. İřık yolu kurma iřleminde olduđu gibi 5 deneme sonunda EAY kurulamazsa, EAY reddedilir. Bunun da sebebi, etiket anahtarlamaalı bađlantıların fazla miktarda ıřık yolu kullanarak hem kaynakları ařırı t ketmesine, hem de fazla miktarda y nlendirilerek O-E-O evrimi sırasındaki gecikmenin artmasını sınırlandırmaktır.

Dađıtılmıř sanal topoloji uyarılması, yerel bilgilerle yerel  z mler  retir. D đ mler, sahip oldukları yerel bilgiyle ađın b t n ne iliřkin kararlar veremeyeceđi iin de, uyarılama  nceden kurulmuř olan bir sanal topoloji  zerinde alıřmaktadır. Oluřturulmuř sanal topolojide veri alıřveriři gerekleřtiren her fiziksel d đ m, aynı zamanda sanal d đ m olmalıdır. Bařka bir deyiřle, sanal topoloji bađlı olmalıdır. Aksi halde, sanal topolojiye bađlı olmayan bir d đ me EAY kurulması gerektiđinde, uyarılama y nteminin bu EAY isteđini de karřılaması gerekecek, d đ mler t m ađ bilgisine sahip olmadıkları iin de bu isteđi karřılamak m mk n olmayacaktır.

3.3 Çalışma Prensibi

Sanal topoloji uyarlama algoritması, mevcut sanal topoloji üzerinde, aşırı yüklenmiş olan ışık yollarını rahatlatmak amacıyla ek ışık yolları oluşturma, ve algoritma tarafından kurulmuş ışık yollarının yükleri azaldığında da bu ışık yollarını silme prensibine göre çalışmaktadır. Her bir düğüm, kendisinden çıkan aşırı yüklenmiş ışık yollarında bu yüke katkı sağlayan en yoğun trafikleri bulur. Daha sonra da bu trafiklerin geldiği ve bir sonra gideceği düğümler arasında ışık yolu kurarak, hem kendi giriş ve çıkış ışık yollarını rahatlatmış olur, hem de trafiğin adım sayısını bir azaltarak, daha az adımda varışa gitmesini sağlar. Zamanla yükleri düşen bu yeni ışık yolları da yine düğümler tarafından silinerek uyarlama gerçekleştirilmiş olur.

3.3.1 Uyarlama İçin Gerekli Veriler

Arabirim	Sanal Komşu	Dalgaboyu	Yük	Fiber (Port)

Şekil 3.1 : Düğümlerdeki Arabirim Tablosunun Yapısı

Arabirim	Etiket	Yük

Şekil 3.2 : Düğümlerdeki Yük Tablosunun Yapısı

Uyarlama için yerel veriler kullanılacağından, düğümler sadece kendi üzerlerinden geçen etiket anahtarlama yolları ve sanal topolojiye göre kendisine bağlı ışık yolları ile ilgili bilgilere ihtiyaç duymaktadır. Düğümün üzerinden hangi bağlantıların geçtiği bilgisi, etiket anahtarlama yollarında kullanılan yönlendirme tablosudur. Bu tablo MPLS, ya da GMPLS gibi etiket anahtarlama yönlendirme sistemlerinde kullanılan tablodur; Uyarlama algoritmasına özel bir tablo değildir. Ancak bu tabloya ek olarak bir takım başka bilgilere de ihtiyaç duyulmaktadır. Bu bilgilerin ne olduğu, Şekil 3.1'deki örnek arabirim tablosunda görülmektedir. Bu tabloda, yönlendirme tablosundaki arabirimin açılımını ifade eden, arabirimin fiber numarası ve dalgaboyu

bulunmaktadır. Bunlara ilave olarak arabirimdeki ışık yolunun diğer ucundaki düğüm olan sanal komşunun kimliği ve bu ışık yolunun yük miktarı yer almaktadır. Yük miktarı, arabirimi ve dolayısıyla ışık yolunu kullanan tüm etiket anahtarlama yollarının yük miktarlarının toplamıdır. Yolların yük miktarları periyodik olarak ölçülerek, aynı ışık yolunu kullanan yolların toplam yük miktarı, bu tablodaki ilgili arabirim yük miktarı alanına yazılır. Uyarılama işlemleri ilk olarak bu tablo taranarak yapılmaktadır. Bu tabloya göre yük miktarı W_H değerinin üzerindeki ışık yolları için ışık yolu ekleme, yük miktarı W_L değerinin altındaki ışık yolları için de bu ışık yolunu silme işlemleri tetiklenmektedir.

Uyarılamada kullanılan 3. ve son tablo ise Şekil 3.2’de görülen, düğümün üzerinden geçen etiket anahtarlama yollarına ait yük miktarlarıdır. Bu tablodaki yollar, yönlendirme tablosundaki yollardır. Karar mekanizmasında sadece düğümünden çıkan ışık yollarının yükleri kullanıldığından, bu tabloda sadece çıkış arabirimlerindeki EAY yükleri tutulmaktadır. Bir EAY’ı ifade etmek için, EAY’ın kullandığı çıkış arabirimi ve bu arabirimde kullandığı etiket bilgisi gereklidir. Bir arabirimde aynı etiketi kullanan başka EAY olamayacağı için bu bilgi, EAY belirlemek için yeterlidir.

3.3.2 Işık yolu ekleme

Uyarılama işlemi aracılığıyla sanal topolojiye ışık yolu eklemek için sanal topolojideki ışık yollarının yükünün, belirlenen yüksek eşik değeri olan W_H ’den fazla olması gereklidir. Bir düğüm, sanal topolojiye göre sadece kendisinden çıkan ışık yollarından sorumludur. Algoritmik ifadesi Şekil 3.3’te verilen algoritmayı çalıştıracak olan düğüm, sanal topolojiye göre arabirim tablosunu kendisinden çıkan ışık yollarını bulmak için tarar. Tarama sırasında toplam yük değeri W_H değerinden yüksek olan ilk ışık yolunun yükünü düşürmek için, yeni bir ışık yolu kurulma çalışmaları başlar.

LightpathAddition()

Sort outgoing lightpaths descending according to their loads

For every outgoingLightpath i

 If Load(i) > W_H

 Group LSPs on i according to the nodes they are coming from

 Sort LSP-Groups descending according to their total loads

 establishLightpath $\leftarrow$ False

 For every LSP-Group j

 totalLoad $\leftarrow$ 0

 Clear LSP-List

 For every LSP k in LSP-Group j

 totalLoad $\leftarrow$ totalLoad + Load(k)

 Add k to LSP-List

 If totalLoad > W_L

 establishLightpath $\leftarrow$ True

 sourceNeighbour $\leftarrow$ Neighbour(k)

 break

 End-If

 End-For

 If establishLightpath = True

 targetNeighbour $\leftarrow$ Neighbour(i)

 confirmation $\leftarrow$ MSG(sourceNeighbour):LightpathRequest(this,
 targetNeighbour, LSP-List)

 If confirmation = True

 break

 Else // Işıkyolu kurulamadı, bir sonraki ışıkyolu için işlem yap

 continue

 End-If

 End-If

 End-For

 If confirmation = True

 break

 End-If

Else // W_H 'ı aşan ışıkyolları bitti. Eklenecek ışıkyolu yok

 LightpathDeletion()

 break

End-If

End-For

Şekil 3.3 : Işıkyolu Ekleme Algoritması

Yeni bir ışıkyolunun kurulabilmesi için, algoritmanın çalıştığı düğüme gelip, düğümden bu ışıkyolunu kullanarak geçen EAY'lar gözönüne alınır. Sadece düğüme gelip geçen EAY'lar için işlem yapılması, toplamda EAY'ların yeniden yönlendirme sayılarını düşürmek içindir. Bu tip gelip geçen EAY'ların bulunabilmesi için yönlendirme tablosundan yararlanılır. Bu tabloya göre hem giriş alanları hem de çıkış alanları dolu olan EAY'lar düğüme gelip geçen EAY'lardır. Yüğü W_H değerini aşan ışıkyolunu kullanarak düğümden çıkan ve bu düğüme başka bir düğümden gelen EAY'lar, geldikleri sanal düğüme göre gruplanır. Bu gruplamanın amacı, bu EAY gruplarının geldiği sanal komşuyu belirlemek, dolayısıyla yeni ışıkyolu kurma işleminde bu sanal komşunun kullanılması içindir. Daha sonra oluşturulan bu EAY grupları toplam yüklerine göre azalan sırada sıralanırlar. Sıralanan EAY gruplarının içindeki EAY'lar için toplam yükleri W_L 'den büyük olacak şekilde alt gruplar oluşturulur ve toplam yük değerleri W_L 'yi geçen en az sayılı grup için, bu EAY grubunun geldiği sanal komşuya bir ışıkyolu kurma mesajı gönderilir. Bu mesajın içinde, ışıkyolunun hangi düğüme kurulacağı ve hangi EAY'ların bu ışıkyoluna kaydırılacağı bilgisi bulunmaktadır. Mesajı alan düğüm ışıkyolunu kurmak için arabirim desteğine ihtiyaç duyar. Şekil 3.4'te mesajı alan düğümün çalıştırdığı algoritma gösterilmiştir. Arabirim tarafından istenen başlangıç ve bitiş düğümleri arasında bir ışıkyolu kurulabilirse, bu düğüm tarafından yeni ışıkyolunun bittiği düğüme bir tablo güncelleme mesajı gönderilir. Mesajı alan düğüm, Şekil 3.5'teki algoritmayı çalıştırarak yönlendirme tablolarını günceller. Mesajı gönderen düğüm daha sonra da kendi yönlendirme tablolarını yeni kurulan ışıkyoluna göre günceller ve EAY listesinde belirtilen EAY'lar bu yeni kurulan ışıkyoluna yönlendirilir. Eğer ışıkyolu kurma isteği sistem tarafından karşılanamazsa, bu sefer ışıkyolu kurma isteği gönderen düğüme olumsuz cevap verilir. Bu düğüm de ışıkyollarını tarama işlemine bir sonraki çıkış ışıkyolundan devam eder. Sanal topoloji üzerinde yapılan değişiklikleri sınırlandırmak için, düğümlerde bir defada bir işlem yapılmaktadır. Bu prensibe göre ışıkyolu kurma mesajını gönderen düğüm, eğer olumlu cevap alır ve ışıkyolu kurulursa, uyarılama işlemini sonlandırarak sırayı bir sonraki düğüme geçirir.

LightpathRequest(vtrNeighbor, targetNeighbor, LSP-List)

```
LightpathEstablished ← System:EstablishLightpath(this, targetNeighbor)
If LightpathEstablished = True
  Msg(targetNeighbor):UpdateTableRequest(this, LSP-List)
  For every LSP i in LSP-List
    UpdateTable(i)
  End-For
End-If
Msg(vtrNeighbor): ReplyMessage(LightpathEstablished)
```

Şekil 3.4 : Işıkyolu Ekleme İsteği Algoritması

UpdateTableRequest(sourceNeighbor, LSP-List)

```
For every LSP i in LSP-List
  UpdateTable(sourceNeighbor, i)
End-For
```

Şekil 3.5 : Tablo Güncelleme İsteği Algoritması

3.3.3 Işıkyolu Silme

Sanal topolojideki ışıkyolları, düşük eşik değeri olan W_L 'den az bir yüklenme oranına sahiplerse, kaynakları işgal etmemeleri için silinirler. Işıkyolu silme işlemine, ekleme işleminde olduğu gibi yine düğümler tekbaşlarına karar vermektedirler. Uyarlama algoritmasını çalıştırmaya hak kazanan düğüm, bir ışıkyolu ekleme işlemini tetiklememişse, Şekil 3.6'daki algoritmayı çalıştırır. Düğüm, kendisinden çıkan ışıkyollarını tarayarak, yükü W_L 'den düşük olan ışıkyolları için silme kararı vermektedir. Bu kararda ışıkyolunun temel sanal topolojiye ait olup olmadığı bilgisi önem taşır. Temel sanal topoloji değiştirilmediğinden, eğer ışıkyolu sonradan uyarlama algoritması tarafından eklenmiş bir ışıkyolu ise silinir. Aynı zamanda silinecek ışıkyolunun üzerindeki EAY'ların silinme sonrası yönlendirilebileceği alternatif bir ışıkyolunun olması gerekir. Silme işleminden sonra, bu ışıkyolundaki EAY'lar, bir önceki güzergahları olan ışıkyollarına yönlendirilirler. Silinen ışıkyolunun bitiş düğümüne Şekil 3.7'deki mesaj gönderilerek silinen ışıkyolu ile ilgili tanımları tablosundan çıkarması sağlanır.

LightpathDeletion()

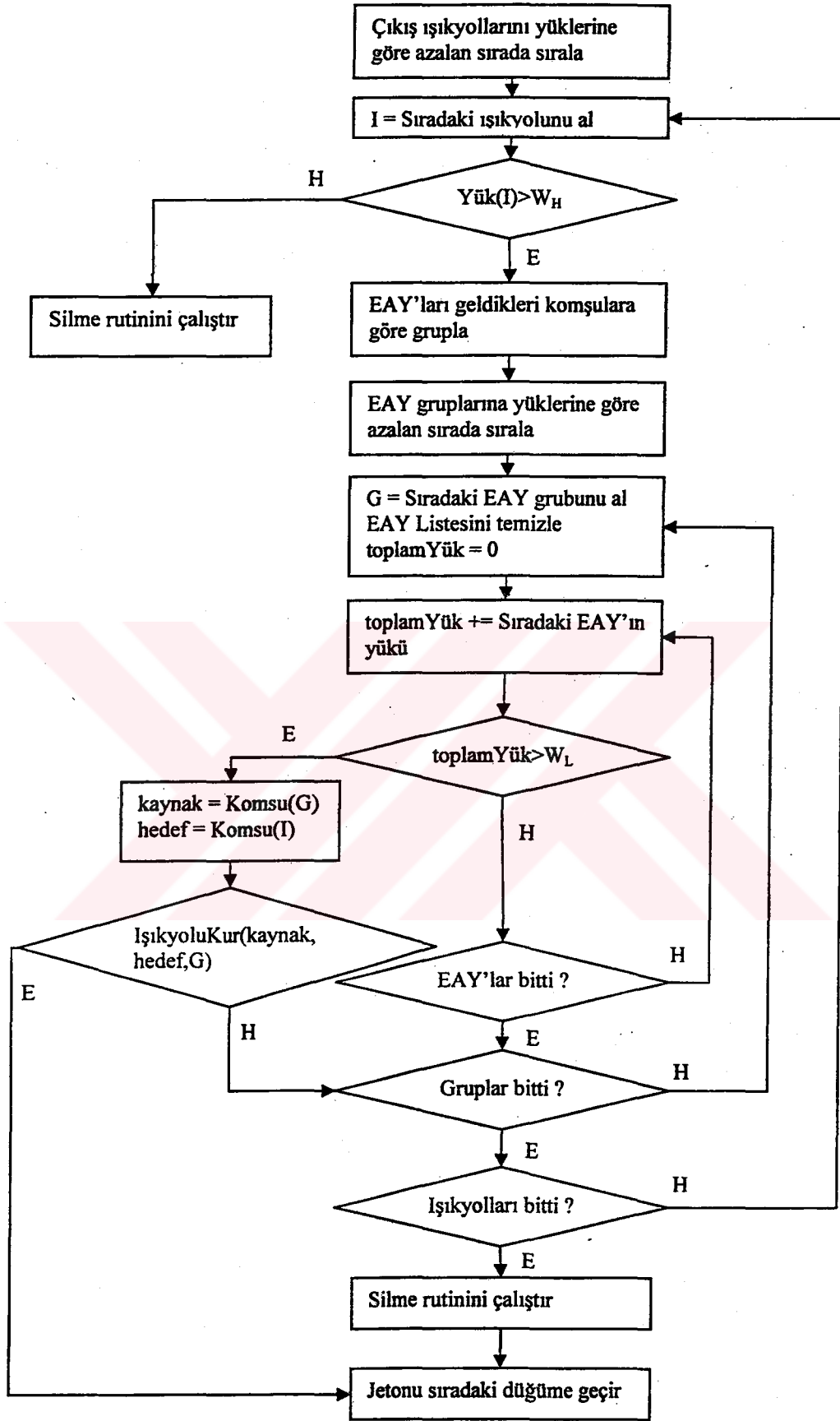
```
Sort outgoingLightpaths ascending according to their loads
For every outgoingLightpath i
  If Load(i) <  $W_L$  AND NOT(isDefaultLightpath(i))
    abortDeletion  $\leftarrow$  True
    For every LSP j on outgoingLightpath i
      abortDeletion  $\leftarrow$  True
      For every outgoingLightpath k
        If k = i
          continue
        End-If
        If DefinedOnLightpath(j, k)
          abortDeletion  $\leftarrow$  False
          break
        End-If
      End-For
    If abortDeletion = True
      break
    End-If
  End-For
  If abortDeletion = True
    continue
  End-If
  For every LSP j on outgoingLightpath i
    RestoreTable(j)
  End-For
  System:DeleteLightpath(i)
  Msg(gidenKomşu):LightpathDeleteRequest(i)
  Break
Else
  Break
End-If
End-For
```

Şekil 3.6 : Işıkyolu Silme Algoritması

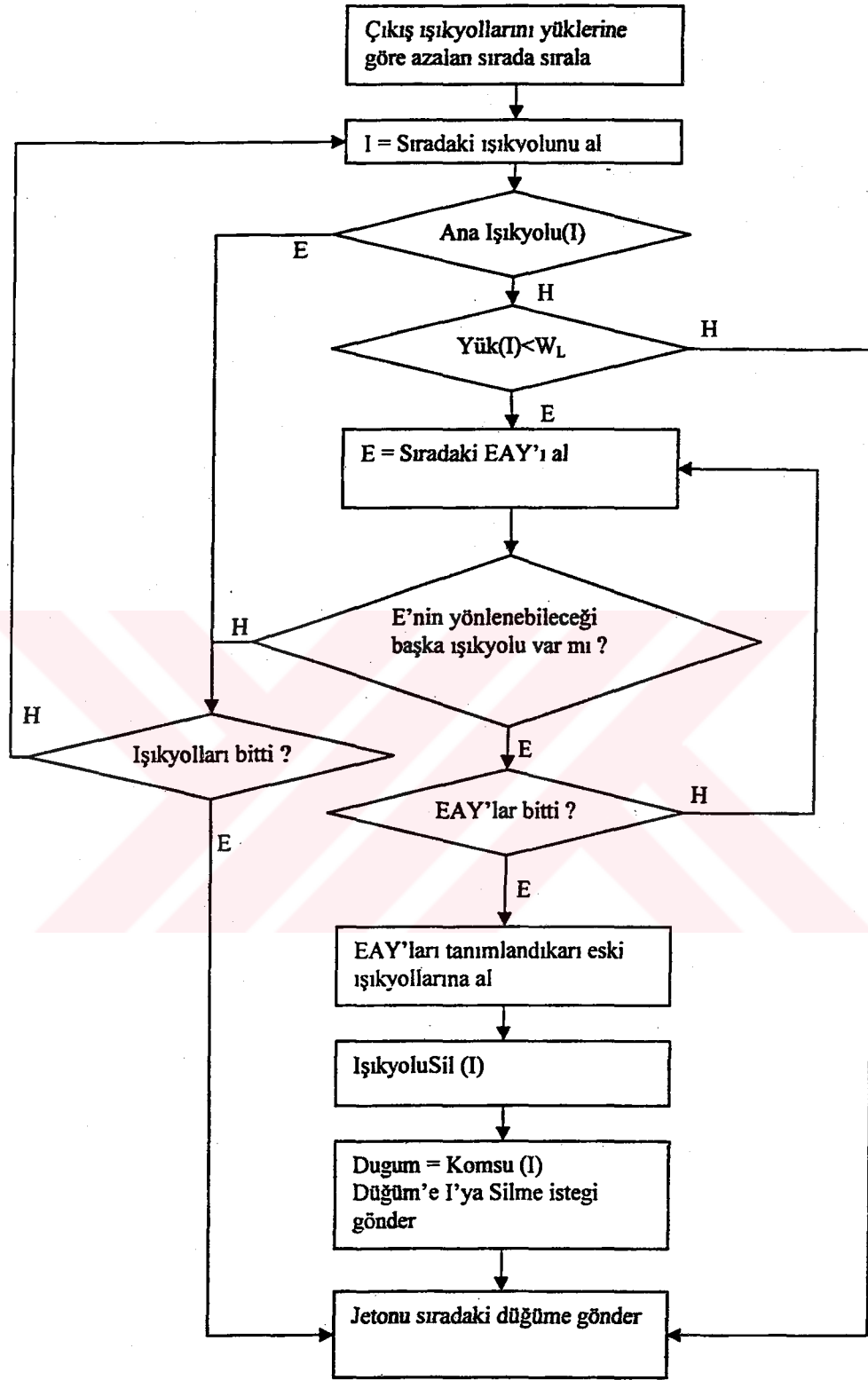
LightpathDeleteRequest (lightpathToDelete)

```
removeFromTables(lightpathToDelete)
```

Şekil 3.7 : Işıkyolu Silme İsteği Algoritması



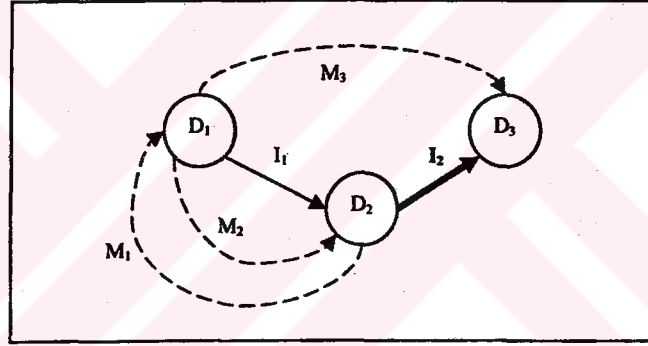
Şekil 3.8 : Işıkyolu Ekleme Algoritmasının Akış Diyagramı



Şekil 3.9 : Işıkyolu Silme Algoritmasının Akış Diyagramı

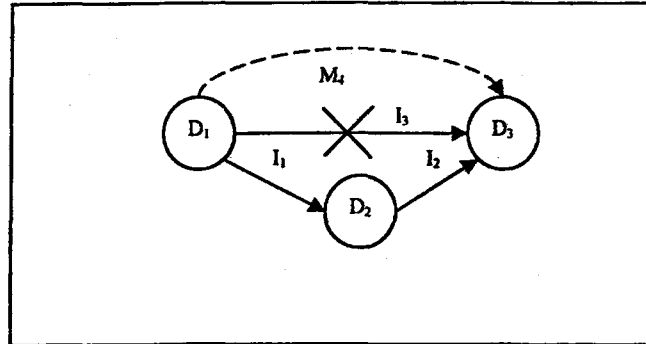
3.3.4 Kontrol Mesajları

Bu bölümde ışıkyolu ekleme ve silme işlemleri sırasında ağda gerçekleşen kontrol mesajları açıklanmıştır. Şekil 3.10'da ışıkyolu ekleme işlemi sırasında mesajlaşmalar gösterilmiştir. D_2 düğümünün uyarılma algoritmasını çalıştırdığı varsayılırsa, I_2 ışıkyolundaki bir yüklenmede, trafik yükünün D_1 düğümünden geldiğinin belirlenmesi sonucunda bu düğüme, ışıkyolu kurma isteği anlamına gelen M_1 mesajını gönderir. D_1 düğümü bu mesaja M_2 mesajıyla karşılık verir. Bu mesaj olumlu ise yeni bir ışıkyolu kurulmuş demektir ve D_2 düğümü algoritmayı çalıştırmayı sonlandırır. D_1 düğümü daha sonra D_3 düğümüne tablo güncelleme mesajı olan M_3 mesajını göndererek, D_3 düğümünün yeni ışıkyolundan gelecek EAY'ları yönlendirmesi sağlanır. M_2 mesajının olumsuz olması durumunda ise, D_2 düğümü algoritmayı çalıştırmaya devam ederek, ışıkyolu kurulabilecek başka bir alternatif arar.



Şekil 3.10 : Işıkyolu Ekleme Mesajları

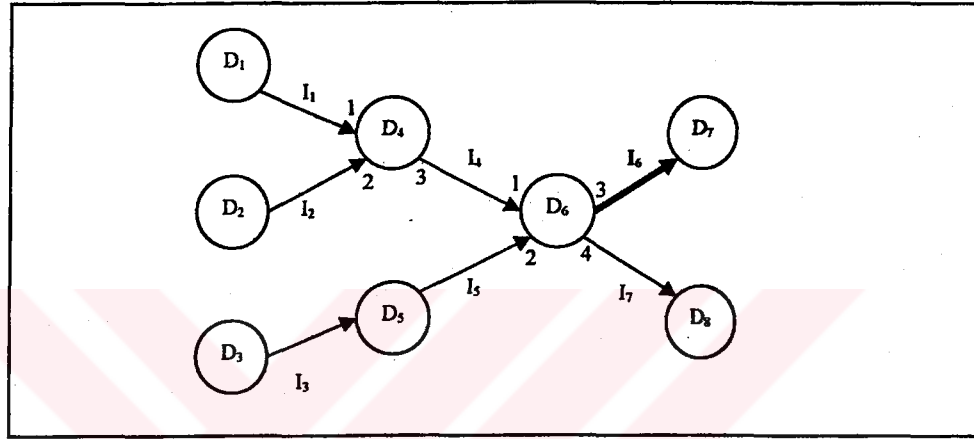
Şekil 3.11'de ışıkyolu silme işlemine ait bir mesaj görülmektedir. Uyarılma algoritmasını çalıştıran D_1 düğümü I_3 ışıkyolunu silme kararı verdiğinde D_3 düğümüne M_4 mesajını göndererek, silinen ışıkyoluna ilişkin bilgilerin tablosundan çıkarmasını sağlar.



Şekil 3.11 : Işıkyolu Silme Mesajı

3.3.5 Örnek Çalışma

Bu bölümde, uyarlama algoritmasının işleyişi, örnek bi senaryo üzerinde şekillerle açıklanmıştır. Senaryoya göre sanal topolojisi Şekil 3.12’de görülen ağ üzerinde, T0 anında EAY’lar Tablo 3.1’de belirtilen yük durumlarına ulaşmışlardır. Bu EAY’lar D6 düğümünde darboğazla karşılaşmakta, ve bu düğümün çıkış ışık yollarında yüklenmeye sebep olmaktadır. Işıkyollarının kapasiteleri 40 Gbps olarak varsayılmıştır. Düğümlerde çalışan uyarlama algoritmaları için yüksek eşik değeri olan $W_H = 35$ Gbps ve düşük eşik değeri olan $W_L = 7$ Gbps olarak kabul edilmiştir.



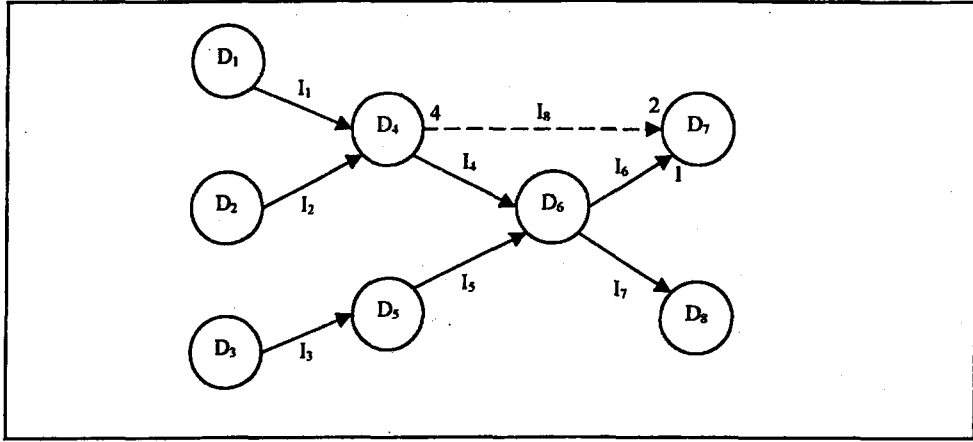
Şekil 3.12 : Başlangıçtaki Sanal Topoloji

Tablo 3.1 : Ağ üzerindeki EAY’lar

	Başlangıç	Bitiş	Yük(Gbps)
EAY1	D1	D7	5
EAY2	D2	D7	4
EAY3	D4	D7	6
EAY4	D3	D7	10
EAY5	D6	D7	12
EAY6	D5	D8	7
EAY7	D6	D8	12

Tablo 3.1’deki EAY güzergahlarına ve bu EAY’ların yük değerlerine göre her ışık yolunun toplam yükleri hesaplandığında I6 ışık yolu en yüksek yüklü ışık yolu olarak gözlemlenmektedir. Bu ışık yolunu, EAY1, EAY2, EAY3, EAY4 ve EAY5 kullanarak, toplamda I6 ışık yolunda 37 Gbps’lik bir yük oluşturmaktadırlar. Bu yük

değeri W_H 'den yüksek olduğu için, uyarılma algoritması tarafından ele alınıp, yükü azaltılmaya çalışılacaktır.

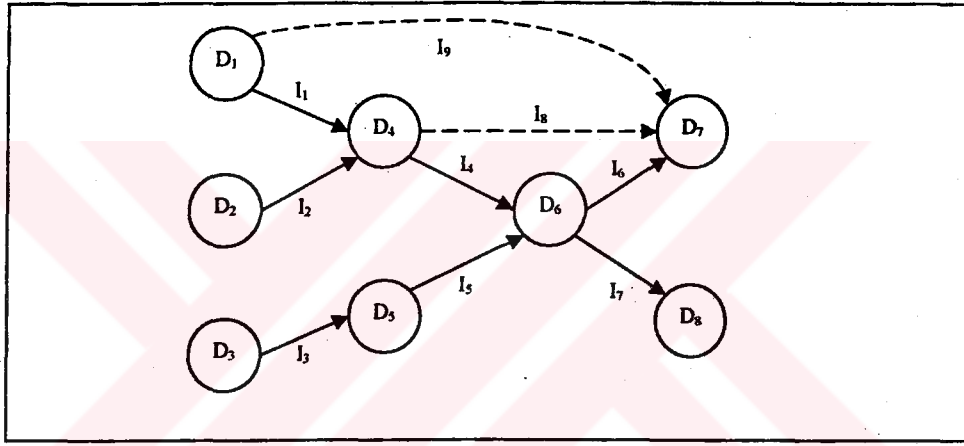


Şekil 3.13 : 1. Aşamadaki Sanal Topoloji

I6 ışık yolunun çıktığı düğüm olan D6, uyarılma algoritmasını çalıştırma sırasını aldığı anda, öncelikle kendisinden çıkan ışık yolları olan I6 ve I7 ışık yollarının yüklerini kontrol eder. I7, EAY6 ve EAY7 tarafından kullanıldığı ve bu EAY'ların toplam yükü olan 19 Gbps toplam yük ile W_H 'nin altında kaldığından bu ışık yolu için işlem yapılmaz. I6'nın yükü W_H 'yi aştığı için, algoritma bu ışık yolunun yükünü düşürmeye çalışır. Bunun için, öncelikle, bu ışık yolunun en çok hangi gelen trafik ile beslendiğinin bulunması gerekir. Bu sebeple, trafikler D6'ya geldikleri komşu düğümlere göre gruplanır. İşlem sonunda, EAY1, EAY2 ve EAY3 D2 düğümünden, EAY4 D5 düğümünden gelir ve EAY5 ise D6 düğümünün kendisinden çıkmaktadır. Gruplama işleminden sonra, bu gruplar toplam yüklerine göre sıralandıklarında EAY1-2-3 grubu 15 Gbps ile en çok yüke sahiptir ve bu gruptaki EAY'lar için bir ışık yolu oluşturulmaya ve I6'nın yükü azaltılmaya çalışılır. Bu noktada amaç, yeni oluşturulacak ışık yolunun yükünün W_L 'nin üzerinde kalmasına dikkat ederek, osilasyonu engellemek ve de olabildiğince az trafiği rahatsız edip uyarlamayı gerçekleştirmektir. Bu sebepten, kendi içlerinde yüklerine göre azalan sırada sıralanan EAY gruplarındaki EAY'lardan, toplamda W_L 'yi geçen en yüksek yüklüleri, yeni oluşturulacak ışık yoluna alınır. Örnekte EAY1-2-3 grubu için, bu grubun geldiği D4 düğümü ile, bu grubun kullandığı çıkış ışık yolu olan I6'nın varış düğümü D7 arasında I8 ışık yolu kurulur. Işık yolu kurulduktan sonra D4 bir mesajla uyarılarak, toplamda 11 Gbps ile 7 Gbps olan W_L 'yi geçen en yüksek yüklü iki EAY, EAY3 ve EAY1'in bu yeni kurulan ışık yoluna yönlendirilmesi sağlanır. Bu

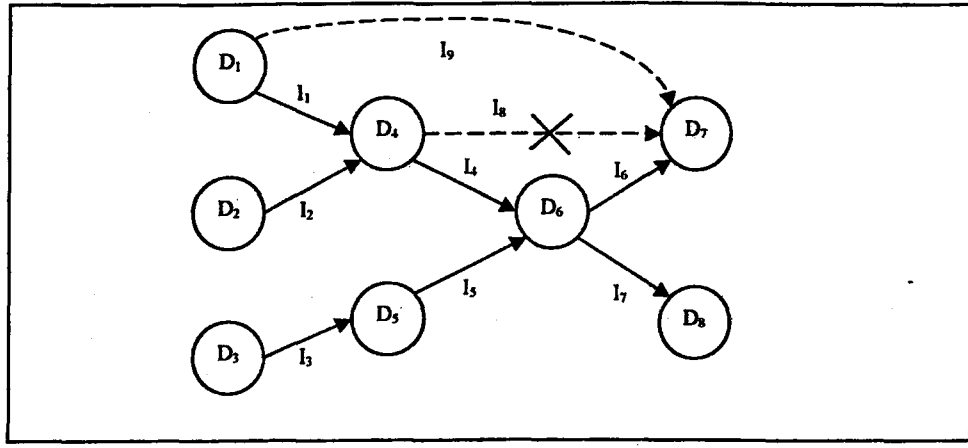
sayede 37 Gbps yüklü olan I6 ışık yolunun yükü 26 Gbps'ye indirilmiş olur. Sanal topolojinin son hali Şekil 3.13'te gösterilmiştir.

İlerleyen zamanda EAY1'in yükünün 27 Gbps ve EAY3'ün yükünün de 9 Gbps olarak arttığı ve diğer EAY'ların yüklerinin sabit kaldıkları varsayıldığında, I8 ışık yolunda toplam 36 Gbps'lik bir yük oluşacaktır. Bu değer de W_H 'nin üzerinde kaldığından, D4 düğümü uyarlama algoritmasını çalıştırdığında ve yukarıda anlatılan işlemleri yaptığında EAY1 için D1 ve D7 düğümü arasında yeni bir ışık yolu kurulmasını sağlayacaktır. Bu noktada D1 düğümü haberdar edilerek, EAY1 trafiğini I1 ışık yolundan çekip, I8 ışık yoluna yönlendirmesi sağlanacaktır. Son durumda ise I8'de 9 Gbps yükü sadece EAY3 kalacak, sanal ağın son hali Şekil 3.14'teki gibi olacaktır.



Şekil 3.14 : 2. Aşamadaki Sanal Topoloji

Zamanla EAY3'ün yükünün 3 Gbps'ye düştüğü ve diğer EAY'ların yüklerinin sabit kaldığı varsayılırsa, üzerinde sadece EAY3'ü taşıyan I8'in yükü, 7 Gbps olan W_L 'nin altında kalacaktır. D4 için uyarlama algoritması çalıştığında, bu düğümünden çıkan I4 ve I8 ışık yollarının yükleri W_H 'nin altında olduğundan bir ışık yolu ekleme işlemi yapılmayacak ve ışık yolu silme işlemi tetiklenecektir. Bu noktada I8 ışık yolunun yükü W_L 'nin altında kaldığından, bu ışık yolu silinecek, EAY3 de bir önceki güzergahına, yani I4 ışık yoluna yönlendirilecektir. Sanal ağın son durumu Şekil 3.15'te gösterilmiştir.



Şekil 3.15 : 3. Aşamadaki Sanal Topoloji

3.4 Osilasyon Problemi

Uyarılma algoritmasının çalıştığı düğüm, ışık yolu kurulması kararı verdiğinde, seçilen EAY'ların bu yeni ışık yoluna yönlendirilmesi gerekmektedir. Yönlendirme, ışık yolu kurma mesajı gönderilen düğüm tarafından yapılmaktadır. Ancak, yönlendirme yapıldıktan sonra eğer yönlendirilen bu EAY'ların toplam yükü, düşük eşik değerinden az ise, bir sonraki sefer yönlendirmeyi yapan düğüm bu ışık yolunu silme yoluna gidecek ve EAY'ları eski ışık yollarına yönlendirecektir. Bu işlem, zaten ilk yönlendirmeden önce W_H değerinin üzerinde bir yüke sahip olan eski ışık yollarında yine bir yüklenmeye sebep olacak, devamında ise ışık yolu kurma işlemleri tekrarlanacaktır. Bu noktadan sonra sürekli ışık yolu kurma ve silme işlemleri peşisıra tekrarlanarak osilasyona sebep olacaktır. Bu durumun engellenebilmesi için, ışık yolu kurma aşamasında, kurulan yeni ışık yoluna yönlendirilecek olan EAY'ların toplam yükünün, düşük eşik değeri olan W_L 'den fazla olmaları koşulu aranmaktadır. Bu koşul sağlanamıyorsa ışık yolu kurulmaz. Koşulun sağlandığı durumda, EAY'ların toplam yükü W_L 'den fazla olduğundan, bir sonraki adımda ışık yolu silme işlemleri yapılmaz. İlerleyen zamanlarda bu ışık yolunun yükü W_L 'nin altına düşebilir ve silinme kararı verilebilir. Böyle bir durumda, ışık yolu kurulmasına karar verecek olan düğüm, yeniden yönlendirilecek EAY'ların toplam yükünün W_L 'den yüksek olması koşulu arayacağından, ya ışık yolu hiç kurulmayacak ya da bu sınırı aşan bir EAY grubu için ışık yolu kurulacaktır. Bu sayede osilasyon engellenmiş olur.

3.5 Etiket Kaydırma

Işıkyolu ekleme işlemi gerçekleştirildikten sonra, belirlenen EAY'lar yeni ışıkyoluna yönlendirilmelidir. Bunun için EAY'ların yeni ışıkyolunda hangi etiketlerle işaretleneceklerinin belirlenmesi ve kurulduğu düğüm ile, yönlendirilmenin yapılacağı düğümün yönlendirme tablolarının bu değerlere göre güncellenmeleri gerekmektedir. Kurulan yeni ışıkyolunun başındaki ve sonundaki iki düğümün, EAY'lar için yeni etiket değerleri belirlemeleri, bir işlem yüküne ve gecikmeye sebep olacağı için etiket kaydırma yöntemi kullanılır. Yeniden yönlendirilecek etiket anahtarlama yolların tümü, aynı sanal komşudan, aynı ışıkyolunu kullanarak, uyarılama algoritmasını çalıştıran düğüme gelmektedirler. Bu sebepten, bu ışıkyolunda EAY'lar için kullanılan etiketler, yeni ışıkyolunda EAY'ları ayrıştırmak ve tanımlamak için uygundur. Yönlendirmenin yapıldığı düğümde, EAY'ın önceki ışıkyolunda kullandığı etiket ve yeni ışıkyoluna ait arabirim numarası ile yönlendirme tablosuna bir satır eklenir. Bu eklemeyen sonra, yönlendirmede kullanılan eski satır tablodan silinerek geçici bir depolama alanına alınır. Bu sayede ilgili EAY'ın tek bir çıkışa yönlendirilmesi sağlanarak olası bir karışıklık engellenmiş olur. Eski etiket-arabirim satırının geçici bir tabloya alınmasının sebebi, ilerleyen zamanlarda bu yeni ışıkyolunun silinmesi gerektiğinde, EAY'ların hala aktif olabilme olasılığıdır. Bu durumda yeniden etiket belirleme işlemlerinin gerçekleştirilmemesi için eski yönlendirme bilgileri elde edilebilmelidir. Silme işlemi gerçekleşirse, tablo eski yönlendirme değerleri ile güncellenir. Bu sayede EAY'lar bir önceki güzergahlarına alınmış olur. Bu durumda o anda kullanılan değerleri tutmaya gerek yoktur. Gerektiği zaman mevcut yönlendirme bilgilerinden yine elde edilebilir.

Bir EAY için bir düğümde üç farklı ışıkyolu kurulmuş olabilir. Fakat bir EAY için bir anda o düğümde sadece bir ışıkyolu aktif olabilir. Diğer ışıkyolu ve etiket değerlerinden oluşan yönlendirme satırları, geçici depolama alanında tutulmaktadır. Işıkyolu silme aşamasında, eğer bu geçici depolama alanında bulunan bir arabirim değerine sahip bir ışıkyolunun silinmesi gerekirse, bu ışıkyolunun temel sanal topolojiye ait olup olmadığı kontrol edilir. Eğer ışıkyolu, temel sanal topolojiye ait değilse, sadece geçici tablodaki bu ışıkyoluna ait arabirime sahip satırlar silinir. Eğer o anda aktif olan ışıkyolunu silme kararı alınırsa, tablodaki mevcut değerler silinir ve

yerine geçici depolama alanına en son koyulan yönlendirme satırı getirilir. Bu sayede EAY'lar bir önceki güzergahlarına alınmış olur.

Etiket kaydırmaya örnek olarak Şekil 3.12'deki D_4 düğümünün tablosunun, ilk ışık yolu ekleme sonrasında geçirdiği değişim verilebilir. Başlangıç aşamasında D_4 düğümünün tablosu Tablo 3.2'deki gibi olduğu varsayılırsa, I_8 ışık yolu kurulduğunda yönlendirme tablosu Tablo 3.3'deki gibi olur.

Tablo 3.2 : D_4 Düğümünün Başlangıçtaki Yönlendirme Tablosu

Giriş		Çıkış	
Etiket	Arabirim	Etiket	Arabirim
1	1	3	3
2	2	4	3
-	-	5	3

Tablo 3.3 : D_4 Düğümünün Güncellenmiş Yönlendirme Tablosu

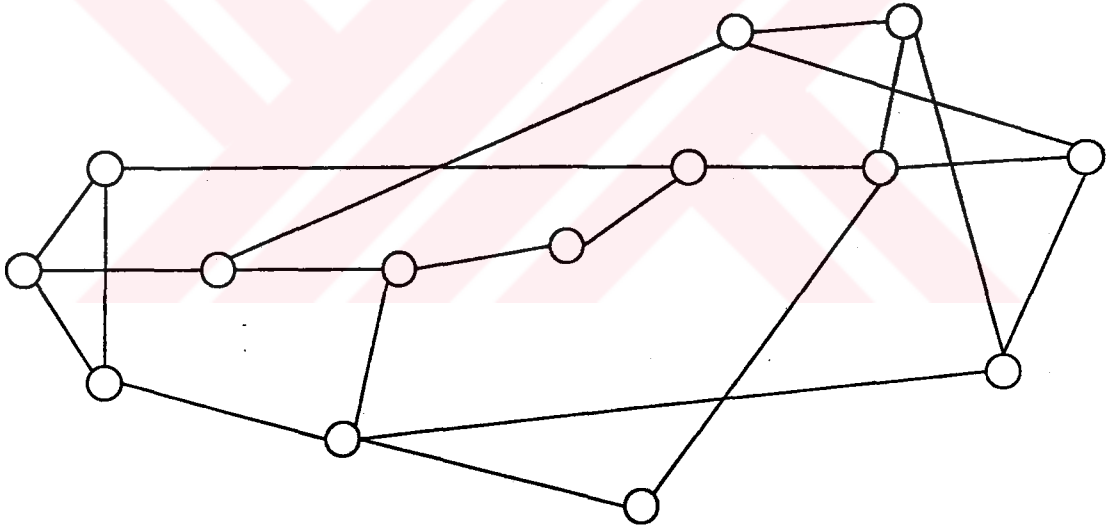
Giriş		Çıkış	
Etiket	Arabirim	Etiket	Arabirim
1	1	3	3
1	1	3	4
2	2	4	3
		5	3
-	-	5	4

4. SİMÜLASYON

Çalışmada önerilen yeni sanal topoloji uyarlaması yöntemi için simülasyon ortamı oluşturulmuş ve senaryolar çerçevesinde algoritmanın verimliliği araştırılmıştır. Elde edilen sonuçlar, grafiklerle beraber açıklanmıştır.

4.1 Simülasyon Ortamı

Simülasyon ortamı için 14 düğüm ve 21 fiberden oluşan Şekil 4.1'de görülen NSFNET fiziksel topolojisi kullanılmıştır. Her düğümde bir etiket anahtarlamalı yönlendirici ve optik bağdaştırıcı vardır. Fiberlerde 16 farklı dalga boyu, düğümlerde ise bu dalga boylarına ayarlanabilen 8 alıcı ve 8 verici bulunmaktadır.



Şekil 4.1 : Simülasyonda Kullanılan NSFNET Topolojisi

Temel sanal topolojiyi oluşturmak için, etiket anahtarlamalı yollar yaratılmış ve her bir EAY için de bir ışık yolu kurulmuştur. Başka bir deyişle başlangıç aşamasında ağın trafik ağırlıklı ortalama adım uzaklığı 1'dir. Simülasyon süresi boyunca ölçümler için etiket anahtarlamalı rastgele yollar yaratılmış, bu EAY'ların yük değerleri değiştirilmiş ve EAY'lar gene rastgele koparılmıştır. Başlangıç anından

itibaren 4 farklı zaman aralığında ağda 4 farklı trafik yoğunluğu oluşturulmuştur. T0'dan T1'e kadar olan birinci aralıkta, yeni EAY'lar oluşturulup zamanla yükleri değiştirilmiştir. T1'den T2'ye olan zaman aralığında mevcut EAY'ların yüklerinde fazla değişiklik yapılmamış, yeni EAY'ların yüklerinin artırılmasıyla, ağın trafik ağırlıklı ortalama adım uzaklığının artması sağlanmıştır. T2'den T3'e kadar olan üçüncü aralıkta EAY'ların ortalama yükleri azaltılmıştır. T3'den T4'e kadar olan son zaman aralığında ise rastgele EAY kurma, yük güncelleme ve koparma oranları eşitlenip denge durumu gözlemlenmiştir.

Simülasyonlar, 7 değişik ölçüm ve bu ölçümleri yansıtan 7 farklı grafik üzerinden yorumlanmıştır. İlk ölçüm metriği, trafik ağırlıklı ortalama adım uzaklığıdır. Bu metrik, trafiklerin adım sayısı ile yoğunlukları çarpımı toplamının, toplam trafik yoğunluğuna bölümü ile elde edilir. Metrik, ağdaki toplam trafiğin ortalama olarak ne kadar yönlendirmeye tabii tutulduğunu ve bunun sonucunda trafiklerin ağın tümüne ne kadar yük bindirdiklerini ifade eder. Yoğun trafiklerin adım sayılarının artması, bu değerin artmasında önemli rol oynar. Sanal topoloji düzenlemesinde, yoğun trafiklerin mümkün olduğu kadar az adımda iletilmesinin sağlanması esas olduğu için, bu metrik önemli bir verimlilik kistasıdır.

Simülasyonlarda ölçülen diğer değerlerden ikisi de, ışık yollarındaki en yüksek ve en düşük yük talep miktarlarıdır. Zamana göre ağdaki ışık yollarının en yüklüsünün yük talep miktarı ile en düşük yüklüsünün yük talep miktarı bir grafiğe yansıtılır. Buradaki amaçlardan biri, sanal ağdaki en yüksek yüklü ışık yolunun yükünün belirli bir değerin üzerinde çıkıp çıkmadığını gözlemlemektir. Çok yüklü olan ışık yolları, sanal ağda darboğazlar oluşturacak ve neticesinde yeni EAY isteklerinin reddedilmesine ve tıkanıklığa uğramasına zemin hazırlayacaktır. Diğer amaç ise sanal topolojideki en düşük yüklü ışık yolunun yük talebini gözlemlemektir. Sanal topolojide belirli bir değerin altında yüklü bir ışık yolu varsa, bu ışık yolu kaynakları boşa harcamış olacak, sanal topolojinin verimliliğini azaltacaktır.

Bir diğer ölçülen değer, ağdaki tıkanıklıktan etkilenmiş ışık yolu sayılarıdır. Sanal topolojideki ışık yollarının kapasitesi %100'ün üzerine çıkarsa, bu ışık yolunu kullanan EAY'lar, tıkanıklığa ve veri kaybına maruz kalacaklardır. İlgili grafik, yüksek yüklenme sebebiyle tıkanıklığa maruz kalan ışık yollarının zamana göre sayılarını gösterir. Aynı zamanda ağda tıkanıklıktan ötürü oluşan paket kayıpları da grafiklerle ifade edilmiştir. Bu grafiklerle beraber, bir de sanal ağ üzerindeki tüm

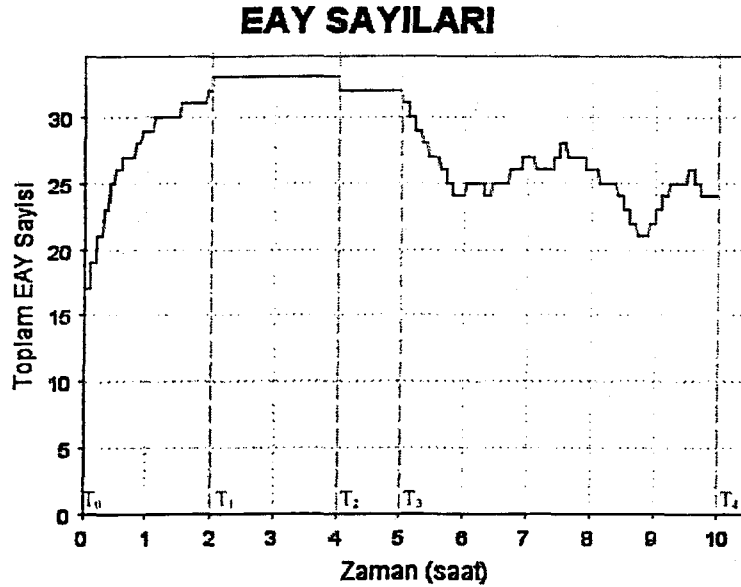
EAY sayılarını zamana göre veren bir grafik vardır. Bu grafik, sanal ağın ne kadar çok EAY kabul edebileceğini ve beraberinde, diğer grafiklerle beraber, EAY sayılarıyla diğer ölçümlerin arasındaki ilişkiyi vurgulamak için üretilmiştir.

Son olarak sanal topoloji uyarlaması yönteminin zamana göre ne kadar ışık yolu ekleyip sildiğini gösteren iki grafik üretilmiştir. Bu grafikler, uyarlama yönteminin, sanal ağdaki değişen trafik durumlarına göre nasıl tepki verdiğini ifade etmektedirler.

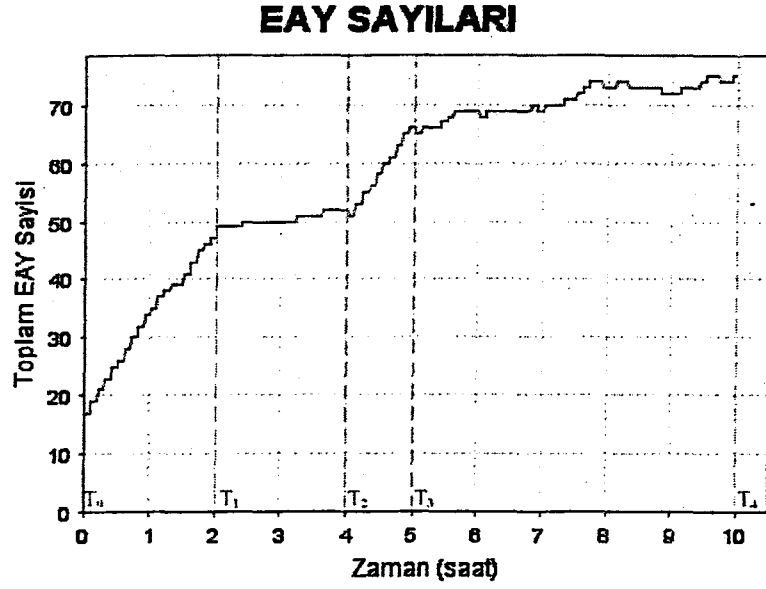
4.2 Uyarlamalı ve Uyarlamasız Sanal Ağ Değerleri

Bu bölümde uyarlama yapılmadan ve uyarlama yapılarak sanal topolojideki değişiklikler karşılaştırılmıştır. Sağlıklı bir karşılaştırma adına her iki durum için aynı ağ koşulları oluşturulmuştur. Bu koşullara örnek olarak, aynı zamanda aynı başlangıç-varış düğümü arasında aynı yüke sahip EAY istekleri üretilmiş, aynı zamanda aynı EAY'ların yükü aynı oranda değiştirilmiş ve aynı zamanda aynı EAY'lar koparılmıştır. Simülasyon 10 saat için çalıştırılarak ağdaki değişimler gözlemlenmiştir.

Simülasyon uyarlama olmadan çalıştırıldığında, toplam 89 adet etiket anahtarlama yol isteği, yeterli kaynak olmadığı için reddedilmiştir. Aynı şatlara sahip simülasyon, yüksek eşik değeri olan $W_H = 80$ ve düşük eşik değeri olan $W_L = 20$ iken çalıştırılmış, elde edilen grafikler uyarlamasız çalışmadan elde edilen grafiklerle karşılaştırılmıştır.



Şekil 4.2 : Uyarlamasız Simülasyon EAY Sayıları



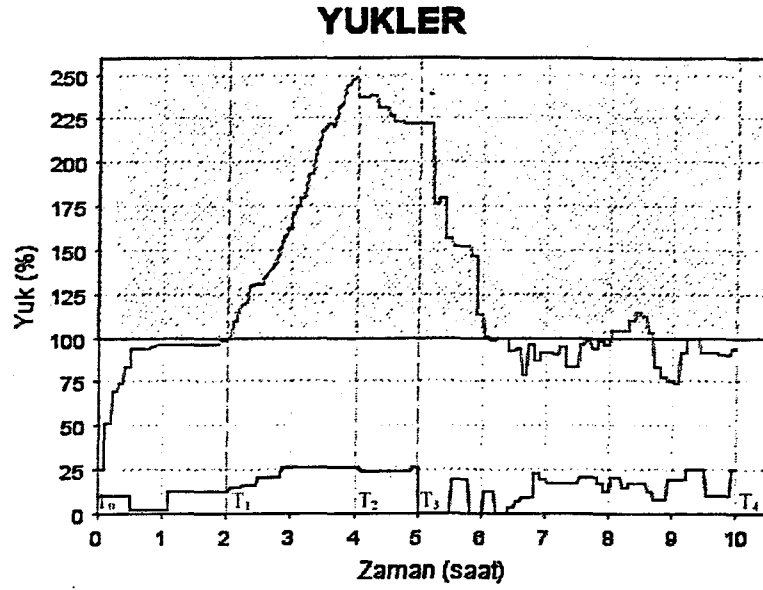
Şekil 4.3 : $W_H=80$, $W_L=20$ Uyarlamalı Simülasyonu EAY Sayıları

Şekil 4.2 ve Şekil 4.3'te simülasyon süreci boyunca ağı kabul edebildiği EAY sayıları görülmektedir. Uyarlamasız çalışmada 25. EAY kabulünden sonra, EAY kabulünde bir yavaşlama olmuş ve yaklaşık 34. EAY'dan fazla EAY kabul edilememiştir. Uyarlamanın çalıştığı Şekil 4.3'teki durumda ise bu yavaşlama gözlemlenmemiş ve simülasyon süresi boyunca belirli bir zamanda 75 EAY değerine ulaşılmıştır.

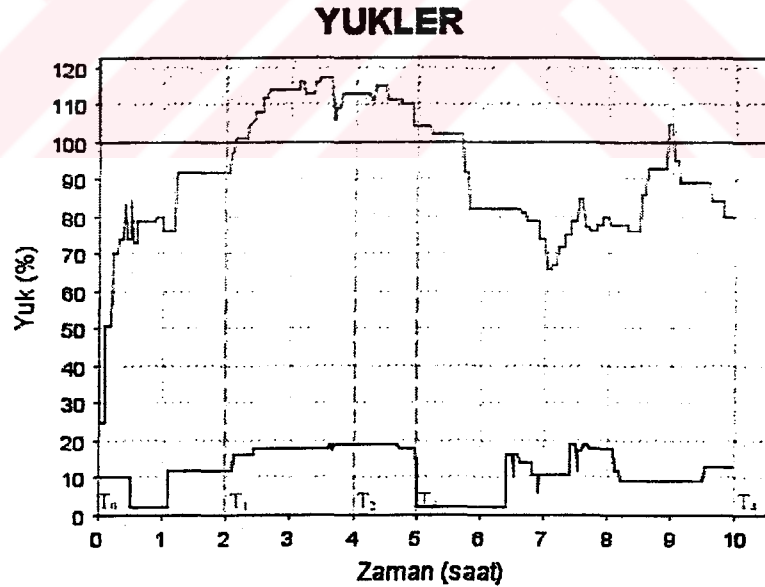
Şekil 4.4 ve Şekil 4.5'te simülasyon süresi boyunca ağıdaki en yüksek ve en düşük ışık yolu yük talepleri karşılaştırılmıştır. Yük talebi, ışık yolunu kullanan trafiklerin toplam ihtiyaç duydukları bant genişliğinin, ışık yolunun bant genişliğine oranının ifadesidir. Bir ışık yolunun kapasitesi %100 ile gösterildiğinde, kapasitesinden fazla miktarda trafik bu ışık yolu kullanmak istediğinde, %100 değerinden sonraki trafikler kayba uğrayacaktır. Bu sebepten %100 çizgisi üzeri bu kaybı ifade etmek için taranmıştır. %100 çizgisi altında kalan alanlar ise, gelen tüm talep karşılanabildiğinden, ışık yollarının o anki yük miktarlarını göstermektedir.

Şekil 4.4'teki grafiğe sahip uyarlamasız çalışmada 2. saatte en yüksek yüklü ışık yolunun yük talebi yaklaşık %100'e ulaşmışken, uyarlamalı çalışmada yüklenme %80 seviyelerinde kalmıştır. Bu grafiklerde esas ayrım yaklaşık 4. saatte başlamış, bu noktadan sonra uyarlamasız çalışmada ışık yolu yük talebi %250 seviyelerine ulaşırken, bu değer uyarlamalı çalışmada %120 seviyesinde kalmıştır. Dengeli

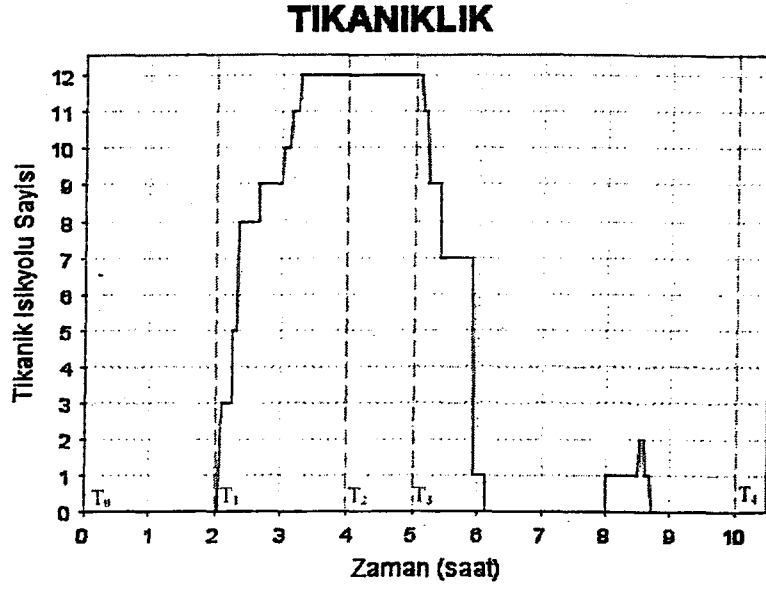
çalışma aralığı olan T3-T4 aralığında da uyarlamalı çalışmada ışıkyolu yük talebinin %100'ü geçtiği zaman aralığı sürelerinin, uyarlamasız çalışmaya göre daha az olduğu gözlemlenmektedir.



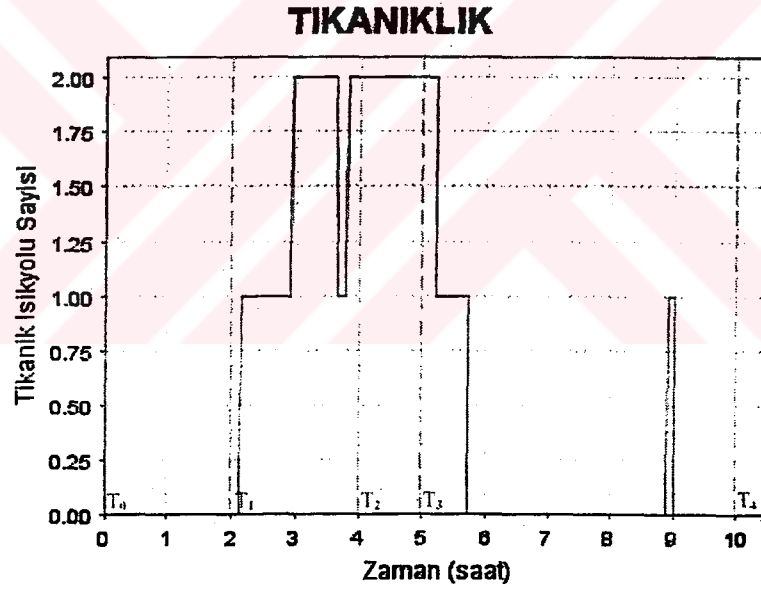
Şekil 4.4 : Uyarlamasız Simülasyon en Yüksek ve en Düşük Yük Talepleri



Şekil 4.5 : $W_H=80$, $W_L=20$ Simülasyonu en Yüksek ve en Düşük Yük Talepleri

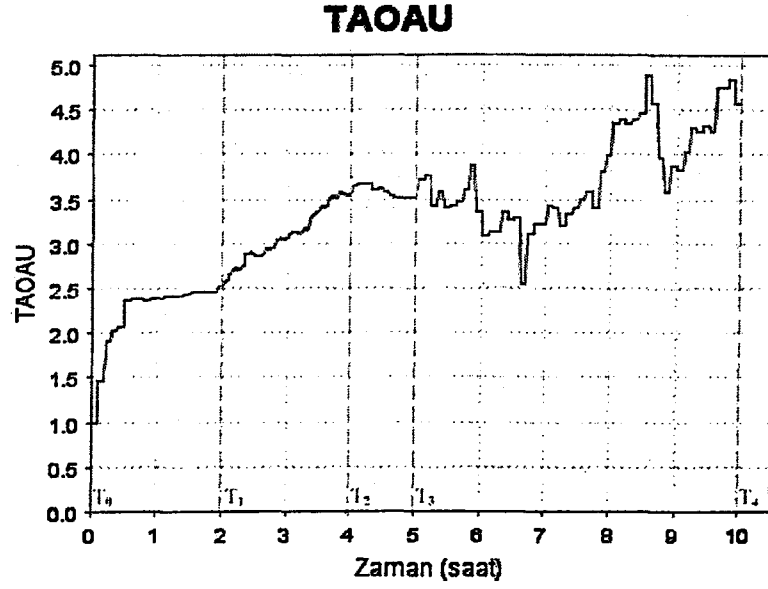


Şekil 4.6 : Uyarlamasız Simülasyon Tıkanık Işıkyolu Sayıları

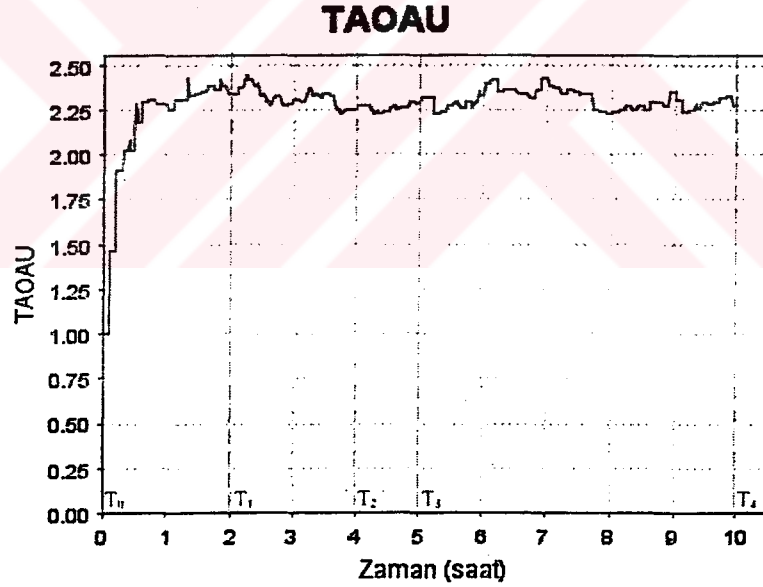


Şekil 4.7 : $W_H=80$, $W_L=20$ Simülasyonu Tıkanık Işıkyolu Sayıları

Şekil 4.6 ve Şekil 4.7’de simülasyon süresi boyunca ağda tıkanıklığa uğramış ışık yolu sayıları görülmektedir. Her iki durumda da 2. saat dolaylarında ağda tıkanıklık başlamıştır. Ancak uyarlamalı çalışmada simülasyon süresince en fazla 2 ışık yolu tıkanıklığa maruz kalırken, bu sayı uyarlamasız çalışmada 12’ye kadar çıkmıştır. Dikkat edilmesi gereken bir diğer nokta da, uyarlamasız çalışmada tıkanıklığın daha geniş bir zaman dilimine yayıldığıdır.



Şekil 4.8 : Uyarlamasız Simülasyon TAOAU Değerleri

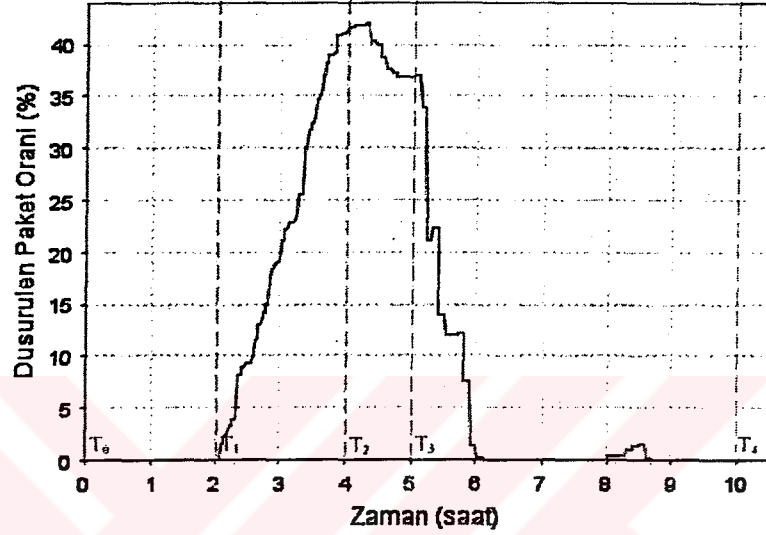


Şekil 4.9 : $W_H=80$, $W_L=20$ Simülasyonu TAOAU Değerleri

Şekil 4.8 ve Şekil 4.9'da simülasyon süresi boyunca ağdaki uyarlamalı ve uyarlamasız çalışmada, trafik ağırlıklı ortalama adım uzaklığı değerleri görülmektedir. Başlangıç durumunda tüm trafikler için bir ışık yolu oluşturulduğundan her iki durumda da TAOAU değerleri 1'dir. Zamanla trafik karakteristiklerinin değişmesiyle mevcut sanal topoloji verimliliğini yitirdiğinden, bu

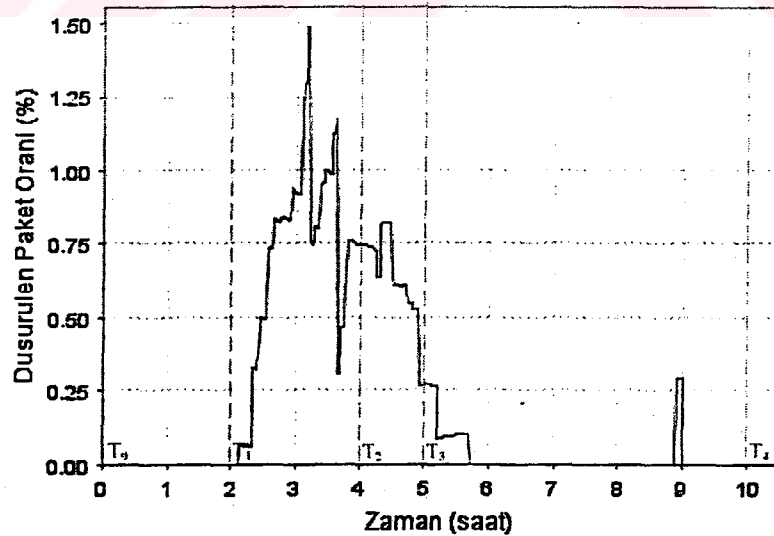
değerlerde artış gözlemlenmiştir. Uyarlamasız çalışmada TAOAU değeri simülasyon sonuna kadar artışını sürdürmüş, simülasyonun sonuna yaklaştığında da 5.0 değerine yaklaşmıştır. Uyarlamalı çalışmada ise, ağır trafiğinin arttırıldığı T1 anına kadar artıp 2.5 seviyesine yaklaşmış, fakat uyarlamanın etkisiyle bu değer aşılmamıştır. Simülasyonun ilerleyen zamanlarında, TAOAU değerinde düşüş gözlemlenmiştir.

DUSURULEN PAKETLER



Şekil 4.10 : Uyarlamasız Simülasyon Düşürülen Paket Oranları

DUSURULEN PAKETLER



Şekil 4.11 : $W_H=80$, $W_L=20$ Simülasyonu Düşürülen Paket Oranları

Şekil 4.10 ve Şekil 4.11’de uyarılama algoritmali ve algoritmasız yapılan simülasyonlardaki paket kayıp oranları gösterilmiştir. Bu grafiklere göre uyarılmalı simülasyonda kaybedilen paket oranı en çok %1.5 iken, uyarılmasız simülasyonda bu değer %45’e kadar çıkmıştır.

4.3 80-15 Uyarılması ile 70-15 Uyarılması Karşılaştırması

Uyarılmasız çalışma ile kıyaslanan yüksek ve düşük eşik değerleri sırasıyla 80 ve 15 olan uyarılmalı çalışmanın verimliliği, bu eşik değerlerinin değiştirilmesiyle artırılabilir. Örnek olarak bu değerleri 70 ve 15 yaparak elde edilen simülasyon sonuçların, 80-15 çiftine göre daha verimli oldukları gözlemlenmiştir.

Tablo 4.1 : $W_H=70$, $W_L=15$ Uyarılması Ağ Değişim Raporu

Eklenen Işıkyolu Sayısı	68
Silinen Işıkyolu Sayısı	34
Reddedilen EAY Sayısı	0
Reddedilen Işıkyolu Sayısı	3
EAY Yeniden Yönlendirme Sayısı	115

Tablo 4.2 : $W_H=80$, $W_L=15$ Uyarılması Ağ Değişim Raporu

Eklenen Işıkyolu Sayısı	52
Silinen Işıkyolu Sayısı	24
Reddedilen EAY Sayısı	7
Reddedilen Işıkyolu Sayısı	89
EAY Yeniden Yönlendirme Sayısı	76

Tablo 4.1 ve Tablo 4.2’de simülasyon sonucunda sanal topolojideki değişim raporları verilmiştir. Bu tablolara göre yüksek eşik değeri 70 olan çalışmada, bu değerin 80 olduğu çalışmaya göre 16 tane fazla ışıkyolu eklemesi yapılmıştır. Bu durumun nedeni, eşik değerinin aşağıya çekilmesiyle, bu değerin üzerinde kalan ışıkyolu

sayısının artmasıdır. Artışın neticesinde daha fazla ışık yolu eklenerek yüklü ışık yollarının sayısı azaltılmıştır.

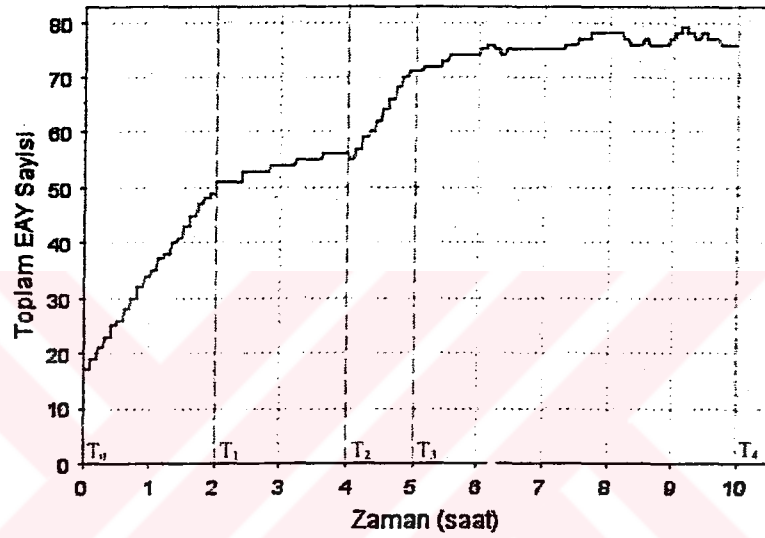
Tablolara göre silinen ışık yolu sayısında da yüksek eşik değeri 70 olan simülasyonun, bu değeri 80 olan simülasyona göre 10 tane fazla ışık yolu sildiği görülmektedir. İlk bakışta düşük eşik değeriyle çalıştırılan simülasyonda, eşik değeri aynı tutulduğu için aynı sayıda EAY'ın eşik değeri altında kalacağı düşünülmektedir. Oysa ki bu sonuçlara göre tersi bir durum gerçekleşmiştir. Bu durumun sebebi, 70-15 simülasyonunda eklenen ışık yolu sayısının daha fazla olmasındandır. Uyarlama algoritmasında, yüklü ve çok adımda aktarılan bir EAY'ın adım uzaklığı için birden fazla düğüm, ışık yolları kurarak, EAY'ın başlangıç düğümünü bitiş düğümüne yaklaştırır. Her bir işlemde adım uzaklığı azalmış olur. Bir düğüm, bu EAY için bir ışık yolu kurmuş ve EAY'ı bu ışık yoluna yönlendirmişse ve bir sonraki ışık yolu kurma işleminde, bu EAY için varışa daha yakın başka bir düğüme komşu bir ışık yolu kurması durumunda, ilk kurulan ışık yolunun yükü düşük eşik değerinin altında kalabilir. Bu durumda düğüm, bir sonraki işlem sırasında bu ışık yolunu silebilir. Neticede aynı düğümden aynı EAY için iki ışık yolu kurma işlemi, ilk ışık yolunun silinmesine sebep olmuştur. Bu durumların artması, toplamda silinen ışık yollarının sayısının artması sonucunu doğurmuştur.

Tablolarda reddedilen EAY sayılarına bakıldığında, 70-15 simülasyonunda hiç EAY reddedilmediği, ancak 80-15 çalışmasında 7 adet EAY'nin reddedildiği görülmektedir. Bunun olası sebebi, 70-15 çalışmasında daha fazla ışık yolu ekleme işlemi sayesinde çok yüklü ışık yollarının yükü azaltılmış ve yeni EAY'lerin kabulüne olanak sağlanmıştır.

Reddedilen ışık yolu sayılarına bakıldığında ise 80-15 simülasyonunda 89 ışık yolu reddedilmiş, 70-15 simülasyonunda ise 3 adet ışık yolu reddedilmiştir. Aslında 70-15 simülasyonunda daha fazla ışık yolu ekleme işlemi olduğu için daha çok kaynak tüketileceği ve dolayısıyla ışık yolu reddetme sayısının daha çok olacağı düşünülebilir. Ancak burada eklenen ışık yollarının bulunduğu bölge önem arz etmektedir. 80-15 simülasyonunda yoğunluğun çok olduğu bölgede işlem yapılmasından dolayı buradaki tükenen kaynaklar neticesinde daha fazla ışık yolu reddedilmiştir.

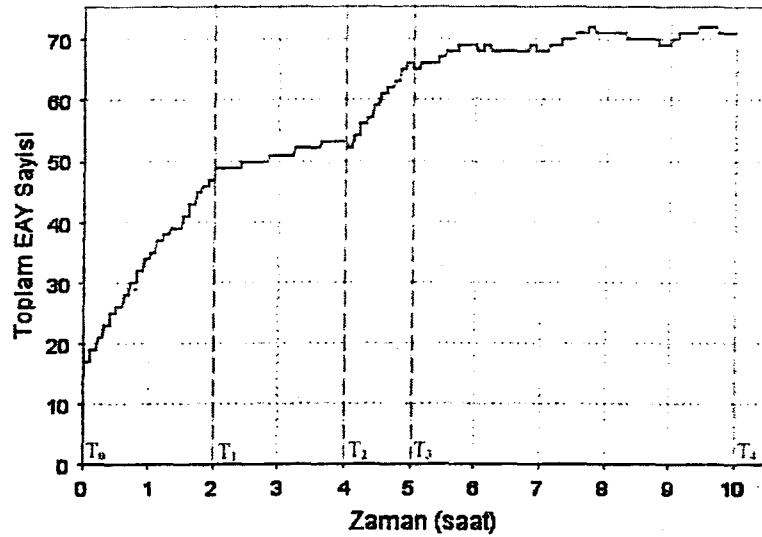
Tabloların son satırı EAY'leri yeniden yönlendirme sayılarını ifade etmektedir. Işıkyolu ekleme ya da silme işlemleri sırasında, EAY'ler yeni ışık yoluna ya da eski ışık yoluna alınmaktadır. Bu işlemler sırasında, her bir EAY için tablodaki bu değer bir arttırılmıştır. Bu değerler için EAY'lerin rahatsız edilme sayıları ifadesi de kullanılabilir. 70-15 simülasyonunda, 68 ışık yolu ekleme ve 34 silme, yani toplam 102 ışık yolu işlemi sırasında EAY'ler 115 defa rahatsız edilmiştir. 80-15 Simülasyonunda ise 52 ışık yolu ekleme ve 24 silme, yani toplam 76 ışık yolu işlemi sırasında EAY'ler 76 defa rahatsız edilmiştir.

EAY SAYILARI



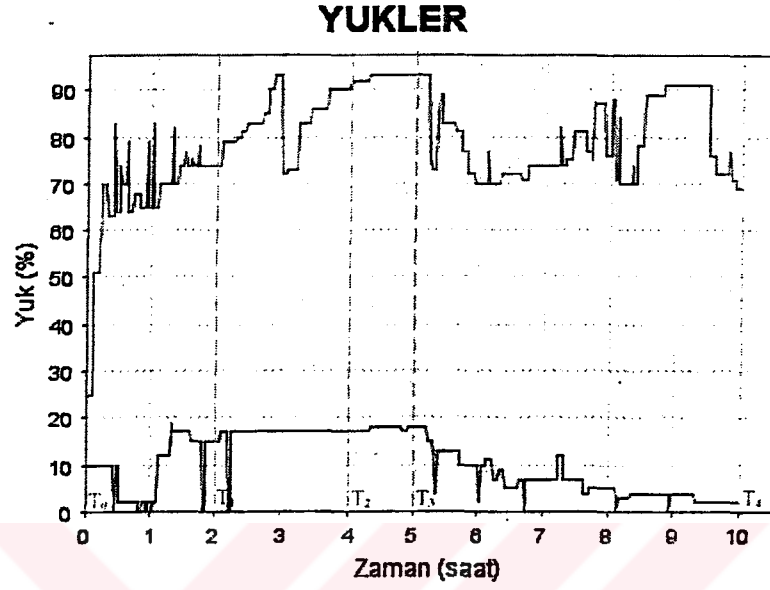
Şekil 4.12 : $W_H=70$, $W_L=15$ Uyarlaması EAY Sayıları

EAY SAYILARI

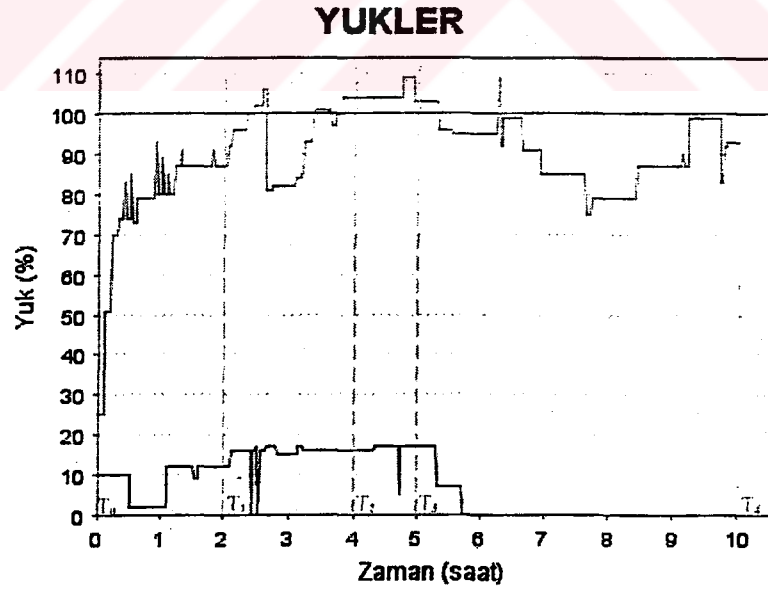


Şekil 4.13 : $W_H=80$, $W_L=15$ Uyarlaması EAY Sayıları

Simülasyon süresince ağdaki EAY sayılarını gösteren Şekil 4.10 ve Şekil 4.11'deki grafiklerde görüldüğü üzere, 70-15 simülasyonunda, 80-15 simülasyonuna göre daha çok EAY kabul edilmiştir.



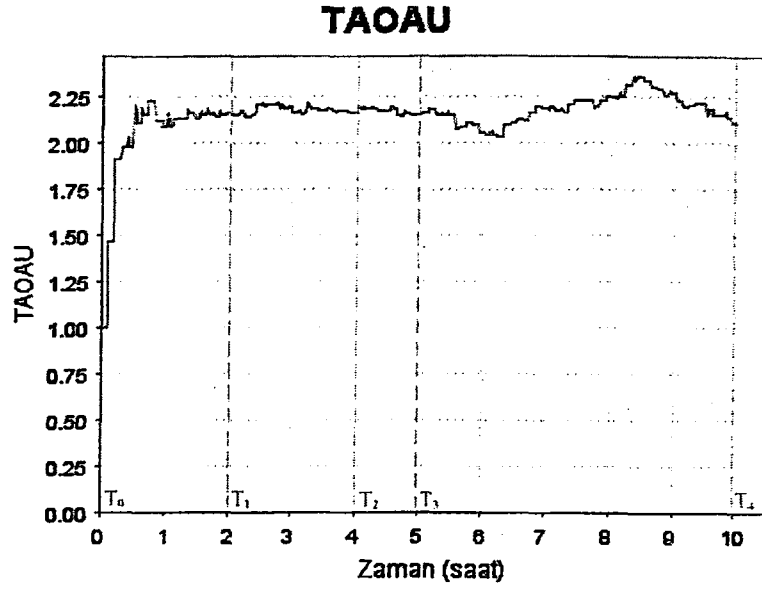
Şekil 4.14 : $W_H=70$, $W_L=15$ Uyarlaması en Yüksek ve en Düşük Yük Talepleri



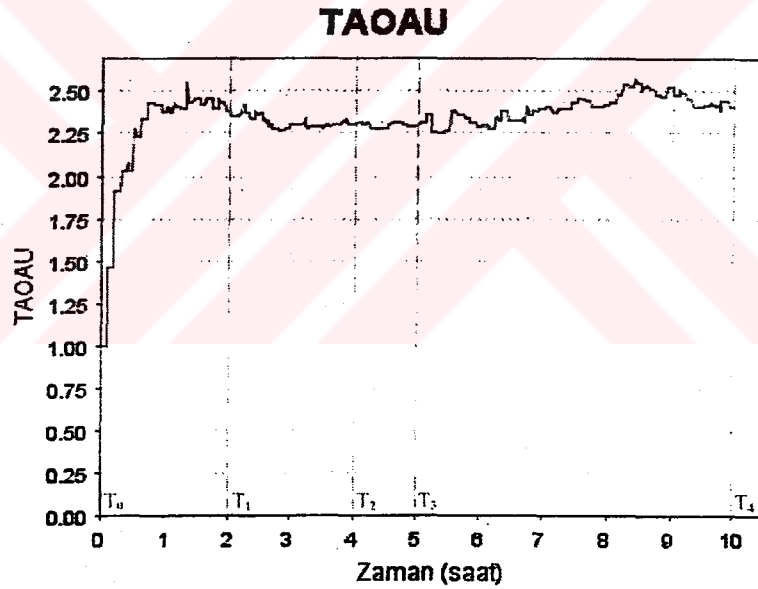
Şekil 4.15 : $W_H=80$, $W_L=15$ Uyarlaması en Yüksek ve en Düşük Yük Talepleri

Şekil 4.12 ve Şekil 4.13'te her iki simülasyon için sanal topolojideki en yüksek yüklü ve en düşük yüklü ışık yollarının yük talepleri görülmektedir. En yüksek yük talebine bakıldığında, 80-15 simülasyonunda %100 kapasitenin üzerine çıkmışken, 70-15 simülasyonunda kapasite aşımı olmamış ve simülasyon süresince hiçbir EAY tıkanıklık sorunu yaşamamıştır. Denge durumu olan T3-T4 zaman aralığında da, en yüksek yük talep miktarları, 70-15 simülasyonunda daha aşağıdadır. Her iki durumda da dikkat çeken nokta, en yüksek yük talep miktarlarının, yüksek eşik değerlerini aşmasıdır. Bunun sebebi, ışık yolu ekleme işlemleri sırasında sadece düğümü gelip geçen EAY'ler için ekleme işleminin yapılmasıdır. Toplamda fazla ışık yolu eklenmesine engel olup, rahatsız edilen EAY sayısını çok arttırmamak adına, düğümünden çıkan EAY'ler için ışık yolu ekleme yapılmamaktadır. Eğer ışık yolunun yükünün önemli bir kısmı, bu düğümünden çıkan EAY'ler tarafında yaratılıyorsa, bu durumda bu EAY'ler için ışık yolu kurulamayacaktır. Neticede bir noktadan sonra yüksek eşik değerinin aşılmasına engel olunamayacak, hatta 80-15 simülasyonunda da görüldüğü gibi, kapasite aşımı gerçekleşecektir.

En düşük yüklü ışık yollarına bakıldığında ise iki simülasyonun da sonuçlarının birbirine yakın olduğu görülmektedir. Bu grafikler için dikkat çekici nokta, sanal ağda düşük eşik değerinin altındaki ışık yollarını silip, kaynak ziyanına engel olmaya çalışılırken, simülasyon sonuçlarında genelde bu değerin altında ışık yolları bulunmaktadır. Bu durumun sebebi de başlangıç aşamasında oluşturulmuş olan sanal topolojiye ait ışık yollarının silinememesindendir. Eşik değerinden düşük yüke sahip bu ışık yolları genelde temel sanal topolojiye ait ışık yollarıdır.



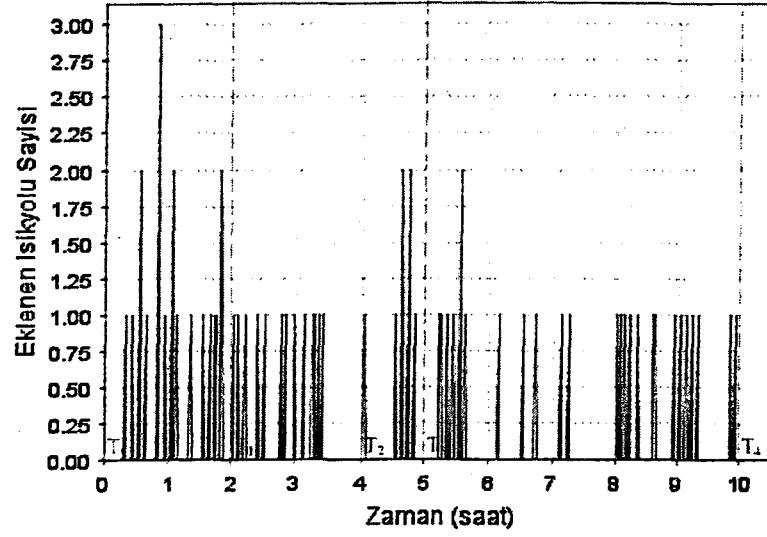
Şekil 4.16 : $W_H=70$, $W_L=15$ Uyarlaması TAOAU Değerleri



Şekil 4.17 : $W_H=80$, $W_L=15$ Uyarlaması TAOAU Değerleri

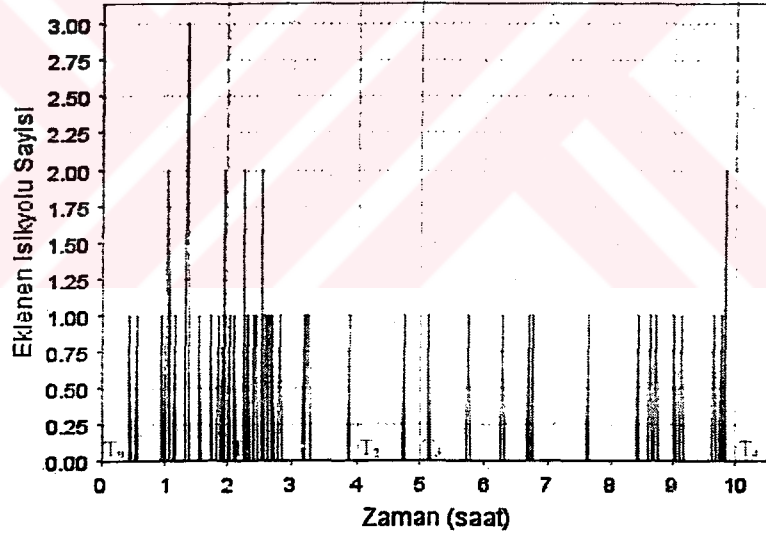
Şekil 4.14 ve Şekil 4.15'teki grafiklerde 70-15 ve 80-15 simülasyonlarının TAOAU değerleri bulunmaktadır. Bu grafiklerden görüldüğü gibi, 70-15 simülasyonu daha verimli çalışmış ve TAOAU değerleri simülasyon süresince yaklaşık 2.1 değerinde seyretmiştir. 80-15 Simülasyonunda ise, bu değer bir ara 2.5 seviyesine kadar çıkmış ve genelde 2.25 dolaylarında seyretmiştir.

ISIKYOLU EKLEME SAYILARI



Şekil 4.18 : $W_H=70$, $W_L=15$ Uyarlaması Işıkyolu Ekleme Sayıları

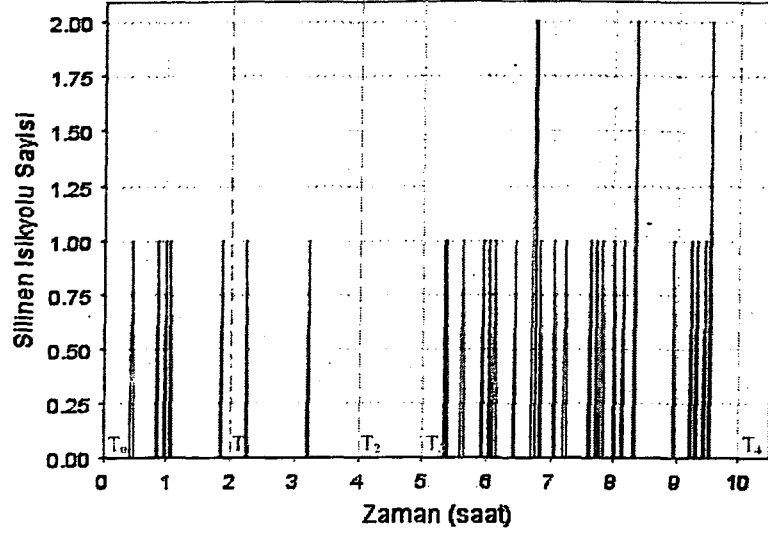
ISIKYOLU EKLEME SAYILARI



Şekil 4.19 : $W_H=80$, $W_L=15$ Uyarlaması Işıkyolu Ekleme Sayıları

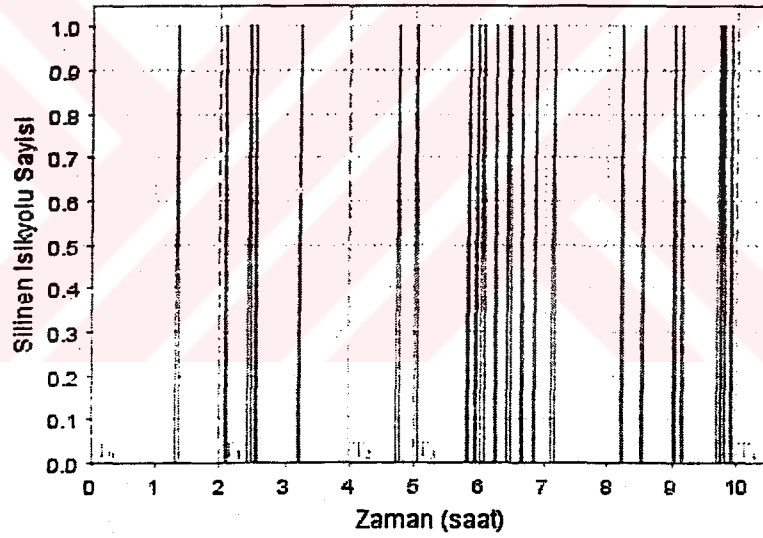
Şekil 4.16 ve Şekil 4.17’de simülasyon süresi boyunca sanal topolojiye eklenen ışıkyolu sayıları görülmektedir. Bu grafiklere göre en çok ışıkyolu, ağı yüklenen T_0 - T_1 zaman aralığında eklenmiştir. Ekleme işlemleri, ağı dengeli trafik oluşmaya başlayana kadar sayıca azalmış, denge durumu olan T_3 - T_4 aralığında ise en alt seviyeye inmiştir.

ISIKYOLU SILME SAYILARI



Şekil 4.20 : $W_H=70$, $W_L=15$ Uyarlaması Işıkyolu Silme Sayıları

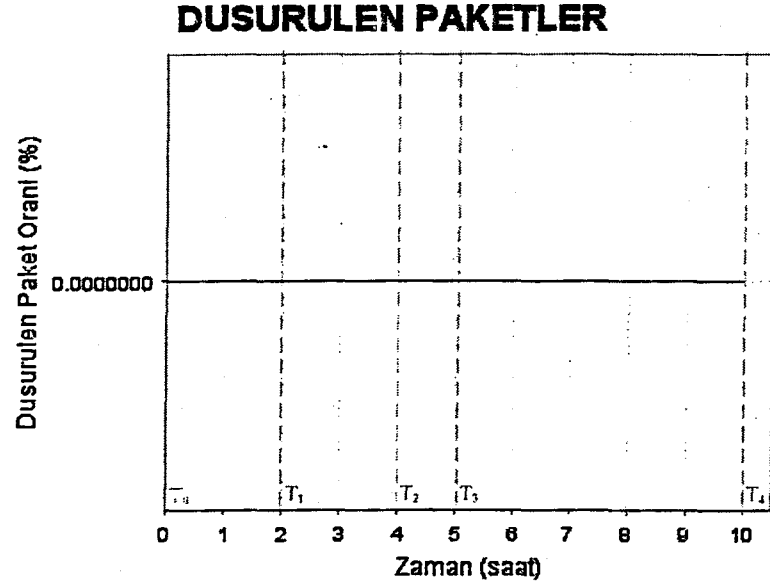
ISIKYOLU SILME SAYILARI



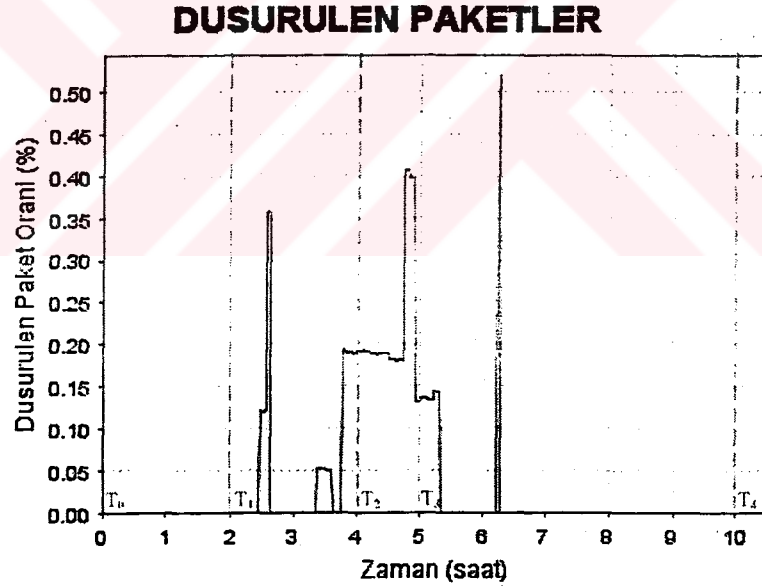
Şekil 4.21 : $W_H=80$, $W_L=15$ Uyarlaması Işıkyolu Silme Sayıları

Şekil 4.18 ve Şekil 4.19’da simülasyon süresi boyunca sanal topolojiden silinen ışık yolu sayıları görülmektedir. 80-15 Simülasyonunda ışık yolu silme sayıları denge durumu olan simülasyonun T₃-T₄ zaman aralığında daha sık gerçekleşmiştir. Denge durumunda ışık yolu silme işlemleri, 70-15 simülasyonunda da gerçekleşmiştir. Ancak 70-15 simülasyonunda ısınma aşaması olan T₀-T₁ zaman aralığında silinen ışık yolu sayısı, denge durumuna göre sayıca ikinci sıradadır. Bunun sebebi de, ağı yüklenme

aşamasında eklenen ışık yollarına kaydırılan EAY'lerin boşalttığı geçici ışık yolların silinmesidir.



Şekil 4.22 : $W_H=70$, $W_L=15$ Uyarlaması Düşürülen Paket Oranları



Şekil 4.23 : $W_H=80$, $W_L=15$ Uyarlaması Düşürülen Paket Oranları

Şekil 4.20 ve Şekil 4.21'de simülasyon süresince düşürülen paket oranları görülmektedir. 80-15 Uyarlamasında yüklenmenin olduğu T1-T2 aralığında en çok %0.5 oranında paket kaybı olmuştur. 70-15 simülasyonunda ise daha çok ışık yolu işlemi yapıp, ağ mevcut trafik akışına en uygun hale getirildiğinden, paket kaybı gözlemlenmemiştir.

4.4 Değişik Simülasyon Senaryoları

Önceki bölümlerde belirli bir senaryo üzerinde elde edilen sonuçlar verilmişti. Uyarlama algoritmasının verimliliği değişik başka senaryolar üzerinden de test edilmiş ve önceki bölümlerde verilen sonuçlara benzer sonuçlar elde edilmiştir. Bu benzerlik sebebiyle, bu bölümde tekrar grafiksel sonuçlar verilmemiştir.

Testler, önceki bölümden farklı olarak iki farklı uygulama şeklinde yapılmıştır. Birinci uygulamada başlangıç aşamasında EAY'ler oluşturulmuş ve zamanla bu EAY'lerin yükleri değiştirilmiştir. İkinci uygulamada ise yük miktarları sabit olan EAY'ler zamanla kurulmuş ve kaldırılmıştır. Her iki uygulamada da aktarılan veri miktarları merkezi ve yerel olarak arttırılmış ve sonra düşürülmüştür. Bu sayede ağda meydana gelecek bir yüklenmede, uyarlama algoritmasının başarısı sınanmıştır. Her iki durumda da algoritma önceki bölümlerde anlatılan senaryo sonuçlarına yakın sonuçlar üretmiş, yükü dengelemeyi başarabilmiştir.

5. SONUÇLAR VE TARTIŞMA

Bu çalışmada optik WDM ağlarının problemlerinden biri olan sanal topoloji uyarlamasına yeni bir yaklaşım getirilmiştir. Konu ile ilgili daha önce yapılan çalışmalarda bu problem genelde merkezi olarak ele alınmıştır. Bu yöntemlerde tüm sanal topoloji bilgisi bir merkezde toplanıp, uyarlama işlemleri bu veriler üzerinde yapılmaktadır. Önerilen yeni yöntemde ise sanal topolojiyi uyarlama görevi topolojideki düğümlere verilmiş, işlemler dağıtılmış hale getirilmiştir.

Dağıtılmış sanal topoloji uyarlaması yöntemi ile merkezi uyarlama yöntemlerinin dezavantajları ortadan kaldırılmıştır. Bu dezavantajlar arasında, merkezde bir arıza durumunda, uyarlama işlemlerinin kesintiye uğraması, ağdan sürekli veri toplama gerekliliği, eldeki tüm ağa ait çok miktarda veriden ötürü uyarlama algoritmasının getirdiği işlemsel yük bulunmaktadır.

Önerilen yeni dağıtılmış çözüm, ağın çok yüklenen ışıkyollarının rahatlatılmasını sağlayarak sanal topolojide meydana gelebilecek aksaklıkların önüne geçilmesini sağlamıştır. Aynı zamanda sanal topoloji uyarlama algoritmalarının hedeflerinden biri olan TAOAU değerinin düşürülmesi sağlanarak, sanal topolojinin verimliliği artırılmıştır.

5.1 Simülasyon Sonuçları

Önerilen yeni dağıtılmış sanal topoloji uyarlaması yönteminin denenmesi için bir simülasyon ortamı oluşturulmuştur. Bu ortamdan elde edilen sonuçlar aşağıdaki gibidir :

- Uyarlama yapılmadan ağda tıkanıklık oluşmuş, ışıyolu yükleri aşırı artmış ve yeni gelen EAY istekleri reddedilme oranı oldukça fazla olmuştur. Uyarlama yapılarak, reddedilen EAY sayısı düşürülmüş, ışıkyollarının yükleri azaltılıp, tıkanıklığın giderildiği gözlemlenmiştir.
- Uyarlamalı çalışmada, yüksek ve düşük eşik değerlerini değiştirerek sanal topolojinin verimliliğinin ayarlanabileceği sonucuna ulaşılmıştır.

- Uygulanan düşük ve yüksek eşik değeri çiftlerinden 70-15 en verimli sonuçları üretmiş, ağda hiç tıkanıklık yaşanmadığı gibi, TAOAU değeri de farklı eşik değerleri çiftlerinin sonuçları arasında en düşük seviye olan 2.1’de seyretmiştir.
- Uyarlama işlemleri sırasında, ağa yüklenildiği durumlarda, fazla miktarda ııkyolu ekleme işlemi yapılp, buna bağılı olarak aynı oranda EAY rahatsız edilmiştir.
- Sonuç olarak, řu ana kadar hep merkezi olarak ele alınan sanal topoloji uyarlaması probleminin, dağıtılmış bir yaklaşımla da çözülebileceğı anlaşılmıştır.

5.2 Gelecekteki Çalışma Önerileri

Önerilen yeni dağıtılmış çözümün eksik noktalarından biri, etiket anahtarlamaalı yolların kurulum aşamasının ele alınmamış olmasıdır. Bu aşama ele alınmadığından, uyarlama işlemlerinin başlayabilmesi için, sanal topolojinin oluşturulmuş ve bağılı olması önkoşulu getirilmiştir. Aynı zamanda başlangıçta oluşturulmuş olan bu sanal topolojideki ııkyollarının silinmesi engellenmiş, bu şekilde sanal topolojinin hep bağılı kalması sağlanmıştır. Etiket anahtarlamaalı yolların ve beraberinde bu kurulum sırasında ele alınması gereken ııkyolları oluşturmanın çalışması yapıldığı takdirde, sanal topolojideki önceden oluşturulmuş ııkyollarını silinememesi kısıtı ortadan kalkacak ve kaynaklar daha verimli bir şekilde kullanılabilir hale gelecektir. Ayrıca bu sayede, uyarlama yönteminin çalışması için aranan önkoşulların oluşturulmasına da gerek kalmayacaktır.

Gelecekte yapılabilecek çalışmalardan bir diğeri de, řu anda sanal topolojiye eklenen ııkyolu sayısının düşürölmesidir. Topolojiye fazla ııkyolu eklemek, EAY’lerin rahatsız edilme oranını arttıracığından, trafiklerdeki kayıpların artmasına neden olmaktadır. Bu sayının düşürölmesi bu kayıpların azalmasını sağlayacaktır.

KAYNAKLAR

- [1] Seo Seung-Woo, Bergman Keren, Prucnal Paul R., Transparent Optical Networks with Time-Division Multiplexing. IEEE Journal on Selected Areas in Communications, VOL. 14, NO. 5, June 1996 pp. 1039 - 1051
- [2] Chan Vincent W. S., Hall Katherine L., Modiano Eytan, Rauschenbach Kristin A., Architectures and Technologies for High-Speed Optical Data Networks. Journal of Lightwave Technology, VOL. 16, NO. 12, December 1998 pp.2146 2168
- [3] Mukherjee B., 2000. WDM optical communication networks: Progress and challenges, IEEE Journal on Selected Areas in Communications, 18, pp. 1810-1824.
- [4] Borella M.S., Jue J.P., Banerjee D., Ramamurthy B., Mukherjee B., Optical components for WDM lightwave networks. Proceedings of the IEEE Volume 85, Issue 8, Aug. 1997 pp. 1274 - 1307
- [5] Mukherjee B., Banerjee D., Ramamurthy S. and Mukherjee A., "Some principles for designing a wide-area WDM optical network", IEEE/ACM Transactions on Networking, 1996 pp. 684-696.
- [6] Colle D., Cheyns J., Develder C., Van Breusegem E., Ackaert A., Pickavet M., Demeester P. GMPLS extensions for supporting advanced optical networking technologies Transparent Optical Networks, 2003. Proceedings of 2003 5th International Conference on Volume 1, 29 June-3 July 2003 vol.1 pp. 170 - 173
- [7] Zhu Keyao, Mukherjee Biswanath, Traffic Grooming in an Optical WDM Mesh Network. IEEE Journal on Selected Areas in Communications, VOL. 20, NO. 1, January 2002 pp. 122 - 133
- [8] Comellas Jaume, Martínez Ricardo, Prat Josep, Sales Vicente, Junyent Gabriel, Integrated IP/WDM Routing in GMPLS-Based Optical Networks. IEEE Network March/April 2003 pp. 22-27
- [9] Gençata A., Mukherjee B., "Virtual Topology Adaptation for WDM Mesh Networks Under Dynamic Traffic", Proceedings of IEEE INFOCOM, Jun. 2002.
- [10] R.T.R. Almeida, K. Potratz, L.C. Bueno, M.E.V. Segatto, E. Oliveira, L.C. Calmon, "Addressing the electronic bottleneck to virtual topology design of optical networks", Proceedings of the 2003 SBMO/IEEE MTT-S International, Sept. 2003.

- [11] **Baldine I., Rouskas G. N.**, 2001. Traffic adaptive WDM networks: A study of reconfiguration issues. *IEEE Journal of Lightwave Technology*, **19**, 433-455.
- [12] **Ramamurthy B., Ramakrishnan A.**, 2000. Virtual topology reconfiguration of wavelength routed optical WDM networks, *IEEE GLOBECOM*, San Francisco, USA, November 27-December 1, pp. 1269-1275.
- [13] **Takagi H., Yongbing Zhang, Xiaohua Jia**, Virtual topology reconfiguration for wide-area WDM networks *Communications, Circuits and Systems and West Sino Expositions. IEEE 2002 International Conference on Volume 1*, 29 June-1 July 2002 pp. 835 - 839 vol.1
- [14] **Awduche Daniel, Rekhter Yakov**, Multiprotocol Lambda Switching: Combining MPLS Traffic Engineering Control with Optical Crossconnects. *IEEE Communications Magazine* March 2001 pp. 111-116
- [15] **Battou A.; Khan B.; Dardy H.; Guizani M.**, A control plane for wavelength routing networks *Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE Volume 3*, 25-29 Nov. 2001 pp. 1450 - 1454 vol.3

ÖZGEÇMİŞ

İsmail Caner Birol 1979 yılında İstanbul'da doğdu. İstanbul Teknik Üniversitesi Elektrik-Elektronik Fakültesi Bilgisayar Mühendisliği Bölümü'nden 2002 yılında mezun oldu. Bu tarihten beri Türk Hava Yolları Bilgi İşlem Departmanı'nda Analist Programcı olarak çalışmaktadır.

