



**GAZİ ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ - BİLGİSAYAR MÜHENDİSLİĞİ**

**171180010 – Cansu AYTEN**

**BM402 BİLGİSAYAR AĞLARI**

**ÖDEV 2: Paket Anahtarlama Ağılarda Kuyruk Yönetimi**

**Mart 2022**

## İÇİNDEKİLER

### Sayfa

İÇİNDEKİLER.....	i
1. PAKET ANAHTARLAMA.....	1
2. PAKET ANAHTARLAMALI AĞLARDA KUYRUK YÖNETİMİ .....	2
2.1. Kuyruk Yönetimi Yapıları .....	2
2.1.1. Drop Tail Algoritması .....	2
2.1.2. Random Early Detection (RED) Algoritması .....	3
2.1.3. BLUE Algoritması .....	4
2.1.4. REM (Random Exponential Marking) Algoritması.....	5
2.1.5. CHOKe Algoritması.....	6
2.1.6. GREEN Algoritması .....	6
KAYNAKÇA .....	7

## 1. PAKET ANAHTARLAMA

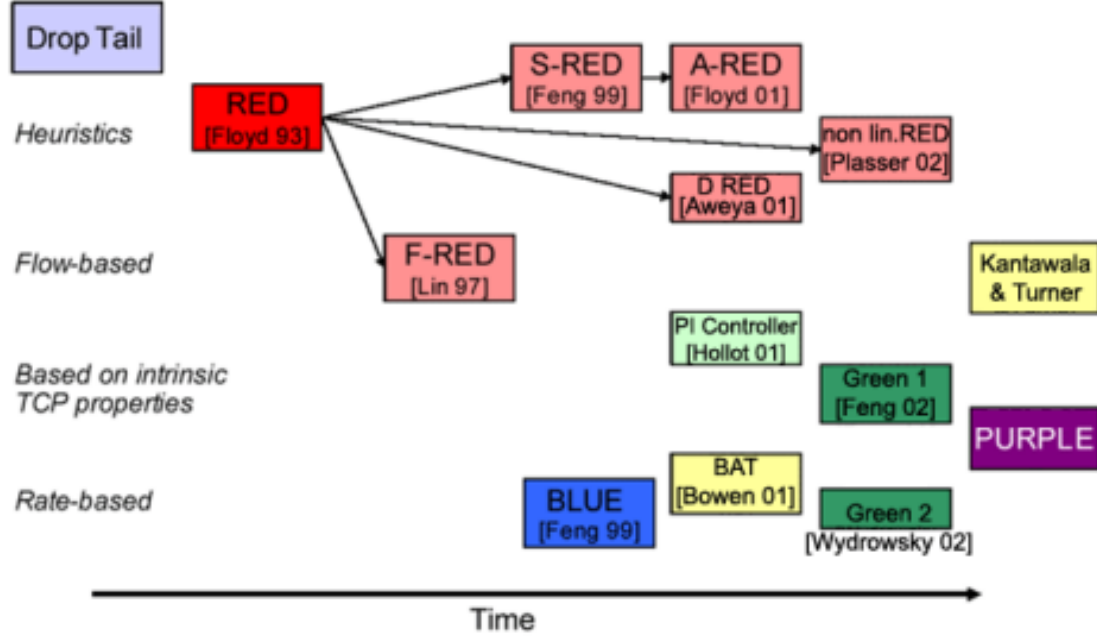
Paket anahtarlama, aktarılabacak olan bir verinin tek seferde değil de daha küçük parçalara bölünerek gönderildiği yöntemdir. Bu yöntemde aktarılabacak olan verilerin daha küçük birimlere bölünmesiyle oluşan yapı paket olarak adlandırılır. Paketler donanıma bağlı olarak herkeste farklı yapıya sahip olabilir. Paket anahtarlama kaynaklar verimli bir şekilde paylaşılır. Paketler ayrılmadan önceki boyuttan daha küçük olduklarından daha hızlı hareket edebilirler. Bant genişliği etkin olarak kullanılır. Ama kaynakların bu şekilde paylaşımını yönetmek zordur. Paket olarak bölünen veriler bir noktadan diğer noktaya taşınırken farklı yollar kullanabilirler. Bu yollar belirlenirken paketler üzerindeki başlık bilgisinden yararlanılır. Bir bağlantıyı birden çok paket kullanabilir. Paketler farklı yollar kullandığı için ulaşacakları noktaya sırasız bir şekilde varabilirler. Bu nedenle varış noktasında paketlerin sıralanması gerekir. Burada da başlık bilgisinden ve birtakım farklı bilgilerden yararlanılmaktadır. Paket anahtarlama birtakım dezavantajlara sahiptir. Bunlara kayıp paketler, değişen gecikme vb. örnekler verilebilir. Ağ tıkanıklığı gibi durumların kayıp paket ve jitter (değişen gecikme) gibi sorunlara yol açtığı söylenebilir. Ağ meşgul ya da dolu olsa dahi paketler kabul edilir, tıkanıklık meydana gelir. Bu durum paketlerin ulaşım süresini artırır. Tıkanıklığın olması durumunda hızlı bir şekilde giderilmeye çalışılır. Hatasız ve eksiksiz gelen paketler sıraya sokulduktan sonra kullanıma hazırdır. Hatalı gelmesi durumunda yalnızca o paketin yeniden gönderilmesi trafiğin yükünü hafifletir [1,2,3,4].

Trafiğin kontrolü ve bazı programların daha verimli çalışabilmesi için Hizmet kalitesi (Quality of Service – QoS) tekniği kullanılabilir. Bu teknikte paketler üzerinde bir değer ile önceliklendirme yapılır. Önceliklendirmeye birlikte zaman kaybının azaltılması amaçlanır. Kesinti veya gecikmeden çok fazla etkilenebilecek ve bu nedenle verimsiz çalışabilecek olan kısacası kesinti veya gecikme durumuna dayanıksız olan programlar daha öncelikli olur. Bu nedenle bazı paketler daha öncelikli olacak ve trafik bu şekilde ayarlanacaktır [5]

İnternette paket anahtarlama yöntemi kullanılmaktadır. Çünkü bir diğer yöntem olan devre anahtarlama yönteminde iki nokta arasındaki kaynaklar tek bir bağlantıya ayrılır. Kaynaklar başka bağlantılar tarafından diğer bağlantı sona erene kadar kullanılamaz. Sırası gelenin kullanması ve diğerinin beklemesi nedeniyle kaynak paylaşımı da zor değildir. Fakat verimsiz bir yöntemdir. Paket anahtarlama devre anahtarlama bulunan boş kalma sonucu olan verimsizliğin çözülüp daha verimli hale getirilmesi nedeniyle İnternette paket anahtarlama yöntemi kullanılmıştır.

## 2. PAKET ANAHTARLAMALI AĞLARDA KUYRUK YÖNETİMİ

Literatürdeki aktif kuyruk yönetimi algoritmaları (AQM) Şekil 1’de özetlenmiştir. Bu araştırma ödevinde ise sırayla Drop-Tail algoritması, RED algoritması, BLUE algoritması, REM algoritması, CHOKe algoritması ve son olarak GREEN algoritması açıklanmıştır.



Şekil 1: Aktif Kuyruk Yönetimi Algoritmalarının Zamana Göre Gelişimi [6]

### 2.1. Kuyruk Yönetimi Yapıları

#### 2.1.1 Drop Tail Algoritması

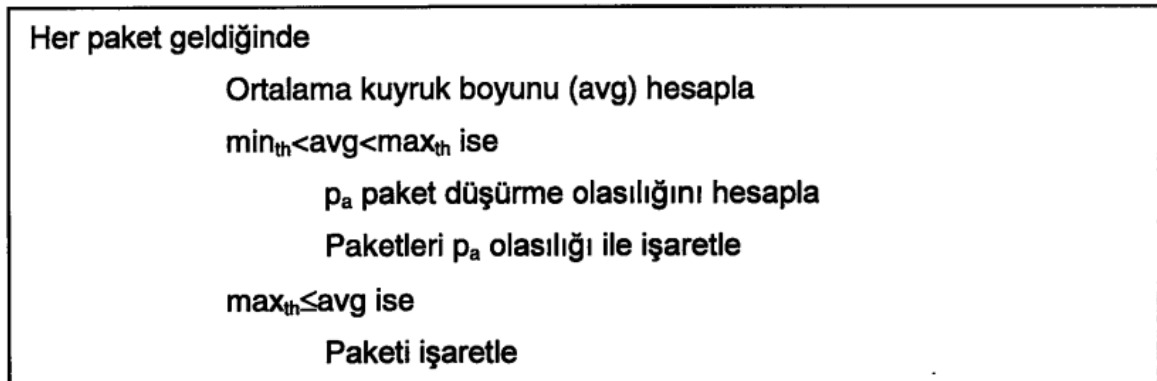
Kuyruk (queue) veri yapısı bazı elemanların depolanmasını sağlar. Bu yapıda elemanlar art arda tutulur. FIFO (First In First Out) yani ilk giren ilk çıkar mantığına uyacak şekilde çalışır. Drop-Tail da bu mantık çerçevesinde ilk gelen ilk hizmet alır (FCFS) şeklinde çalışmaktadır. Paketlerin ne zaman bırakılacağına karar veren basit bir yöntemdir. Bu yapıda ilk gelen paket ilk yönlendirilecektir. Kuyruk maksimum uzunluğuna ulaşana kadar gelen paketler kabul edilir. Maksimum uzunluğa ulaştığı durumda kuyruktaki sıra azalincaya kadar gelen paketler reddedilir. Uzun yıllarca kullanılan bu yapı İnternet tarafından da kullanılmıştır. İki temel dezavantaja sahiptir [6,7,8]:

1. Kilitlenme: Kilitlenme nedenleri senkronizasyon ve diğer zamanlama sorunlarıdır. Drop tail yapısında kuyruk bazen yalnızca bir ya da birkaç bağlantı tarafından kaplanır. Bu durum diğer bağlantıların paketlerinin kuyrukta yer edinmesine engel olur.

2. Kuyruk Doluluğu: Drop tail yönteminde kuyruk dolu olup gelen paketin reddedilmesi durumunda tıkanıklık sinyali verildiği için kuyruklar uzun süre dolu ya da neredeyse dolu durumunu korur. Kuyruk yönetimindeki en önemli amaçlardan biri de kuyruk boyutunu kontrol etmek, kararlı durumdaki kuyruk boyutunu azaltmaktır.

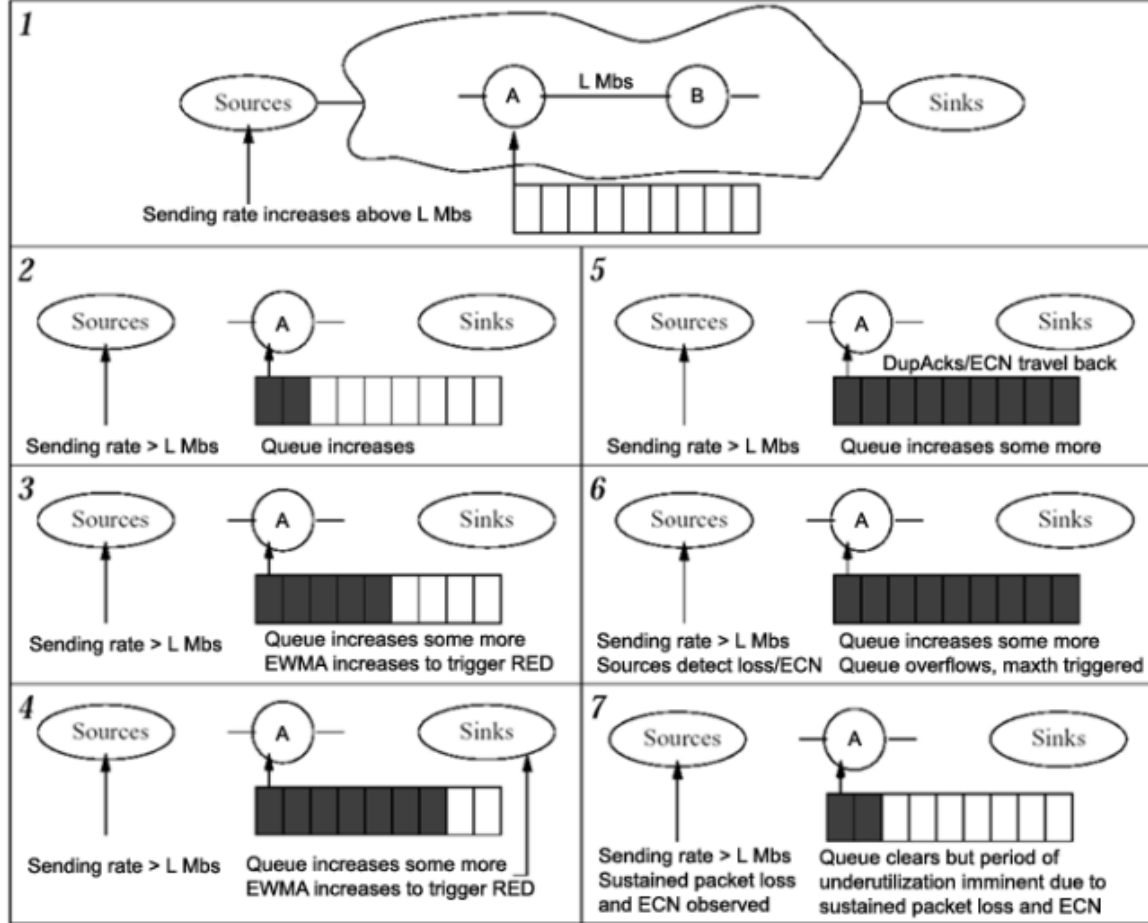
### 2.1.2 Random Early Detection (RED) Algoritması

TCP'nin drop-tail yöntemindeki en büyük sorunlarından biri kuyruk taşması nedeniyle meydana gelen paket kaybının anlaşılmasıyla birlikte kaynakların iletim hızlarının düşmesidir. Yönlendiricide kuyruğun dolu olması sebebiyle reddedilen paket ile bunun kaynakta algılanması arasında dikkate alınması gereken bir süre geçebilir. Bu sürede gönderenin ağı desteklemediği yüksek bir hızda iletme devam etmesiyle birçok paket düşebilir. RED tıkanıklığı erkenden algılayıp son ana bilgisayarlara tıkanıklık bildirimini gönderip bu sorunu hafifletici bir çözüm sunar. Bu sayede iletim hızı kuyruk henüz taşmadan düşürülebilir. Kısacası RED yöntemi tıkanıklıktan kaçınmak amacıyla geliştirilmiş bir yöntemdir denilebilir. Paketleri kuyruklar taşmadan olasılıklı olarak bırakmak amacıyla istatistiksel yöntemler kullanır. Böylelikle bir kaynağı kuyruğu sabit tutmaya yetecek kadar yavaşlatır. Böylelikle yüksek hızda iletim yapılması sonucundaki kayıp paket sayısı azalmış olur. RED paketlerin ne zaman bırakılacağıyla ve hangi paketlerin bırakılması gerektiğiyle ilgilenir. Bu yöntem ortalama kuyruk boyutunu takip ederek her yeni paketin gelmesinin ardından yeniden bir ortalama boyut hesaplar. Bu ortalama boyutun belirli bir eşiği aşıp aşmamasına göre paket hakkındaki karar verilir –paketler işaretlenir-. Karar verilirken iki eşik parametresi mevcuttur. İlki minimum eşiktir. Ortalama kuyruk boyutu bu eşiğin altında ise hiçbir paket bırakılmaz. Diğeri ise maksimum eşiktir. Ortalama kuyruk boyutu bu eşiğin üstünde ise tüm paketler düşürülmelidir [6,9].



Şekil 2: RED Algoritması [7]

Tıkanıklık bildirimi konusunda tek yapılabilir yöntem paket düşürülmesi değildir. Farklı bir yöntem olan açık tıkanıklık bildirimi (ECN, Explicit Congestion Notification) de bu amaçla kullanılabilir. Böylelikle tıkanıklık olduğunda paketler düşürülmeden de akışlar bilgilendirilebilir [7].



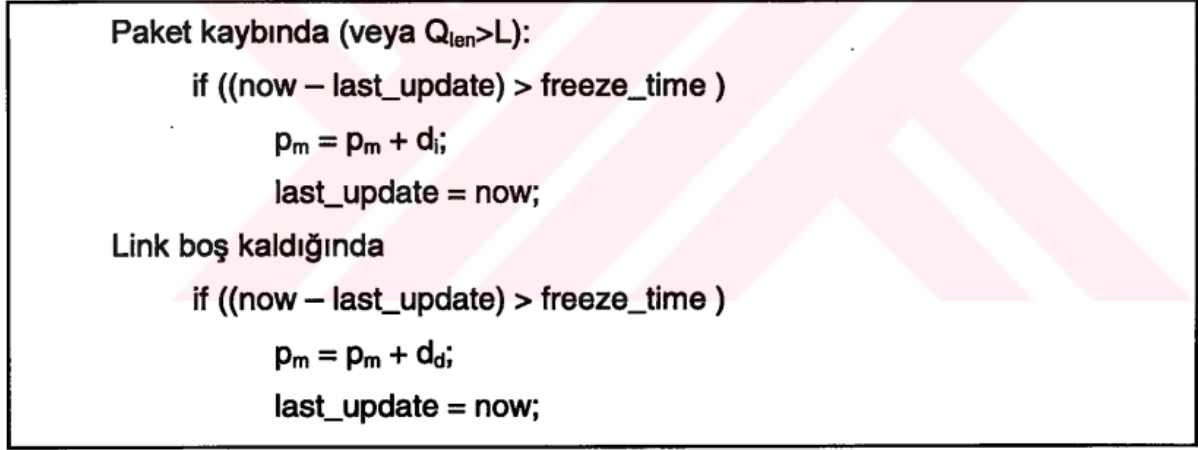
Şekil 3: RED Örneği [6]

RED bazı dezavantajlara sahiptir. İlki uygun parametreleri bulmanın zor olmasıdır. Uygun parametreden kasıt ne zaman paketler bırakılmaya başlanmalı ya da ortalama kuyruk uzunluğu için zaman ölçeği ne olmalı gibi sorulardır. Diğerleri ise parametreler doğru ayarlanmazsa algoritma düzgün bir şekilde çalışmaz [10].

### 2.1.3 BLUE Algoritması

BLUE yöntemi kuyruk boyutuyla ilgilenmek yerine paket düşürmeleriyle ve hattın kullanım geçmişiyile ilgilenerek kuyruk yönetimini sağlamaktadır. RED'den önemli farklarından biri budur. Bu yöntemde paketleri işaretlemek amacıyla kullanılan olasılık  $p_m$  olarak belirtilir. Bu

olasılık kuyruğun taşması ya da boş kalmasına bağlıdır. Kuyruk dolu ise paketler düşeceğinden işaretleme olasılığı olan  $p_m$  artırılır. Böylelikle tıkanıklık önlenmeye çalışılır. Kuyruk boş hale gelmekteyse  $p_m$  değeri düşürülür. Kuyruğun doluluk durumuna bağlı olarak işaretleme yapılmaktadır [7]



Şekil 4: Blue Algoritması [7]

Algoritmadaki parametrelerden biri freeze\_time'dır. Bu parametre kuyruğun güncel olmasını sağlar. Bu güncellemeler kuyruktaki anlık değişimlerin önemsenmemesini ve kuyruktaki değişimlere gereksiz tepki verilmemesini önlemek için belirli aralıklarla yapılır. Bu parametre dışında paket işaretleme olasılığını artırıp azaltmakta kullanılan iki parametre daha bulunmaktadır. Bunlar  $d_i$  ve  $d_d$ 'dir.  $d_i$  paket işaretleme olasılığını artırmaya yarar,  $d_d$  ise tam tersi şekilde azaltmaya yarar [7].

Bu algoritmadaki bazı dezavantajlara örnek olarak erken tıkanıklık tespitinin olmaması ve yavaş yanıt verilmesi verilebilir. Avantajı ise yüksek verimliliğidir [9].

#### 2.1.4 REM (Random Exponential Marking) Algoritması

REM yöntemi ihmal edilebilir paket kaybı ve gecikmeye erişimi bununla birlikte bant genişliğinin yüksek düzeyde kullanımını hedefler. Bu yöntemde tıkanıklık ölçüsü, bant genişliğine daha çok talep olması ve kullanıcı sayısına bağlıdır. Performans ölçüsü ise kullanıcı sayısından bağımsızdır, kullanıcılar hedefledikleri performans etrafında dengede tutulmalıdır. REM ile RED arasındaki farklardan biri paket işaretleme yönteminin farklılığıdır. Paket işaretleme yöntemi konusunda REM her kuyruk tıkanıklığını ölçmek için bir değer tutar. Bu değer price olarak adlandırılır ve paket işaretleme olasılığı belirlenirken kullanılır. REM'deki amaç giriş oranı ile bağlantı kapasitesini yakın tutmaktır. Eğer giriş oranı bağlantı kapasitesi

farkı pozitif ise price artırılır, negatif ise price azaltılır. Bu şekilde price değeri güncellenmektedir. Diğer bir farklılık ise tıkanıklık belirleme yöntemidir. Bahsedildiği üzere REM’de bu yöntem price değeri iken RED’de ortalama kuyruk boyutudur. Tıkanıklık ile performans ölçüsünü birbirinden ayırmak için paket kaybı, gecikme ve ortalama kuyruk boyutu gibi yapılarla ilgilenmez [8].

### **2.1.5 CHOKe Algoritması**

CHOKe yöntemi uygulanması basit bir yöntem olup yanıt vermeyen akışlar ile de ilgilenir. Bu yöntem Drop-tail ve RED algoritmalarının birleştirilmesi ile oluşan algoritma üzerinde birtakım küçük değişiklikler sonucu oluşmuştur. Bir paket geldiğinde ortalama kuyruk boyutu belirlenen eşik değerinin üzerindeyse tampondan rastgele bir paket seçilerek gelen paketle karşılaştırılır. Bunun sonucunda eğer iki paket de aynı akıştan geliyorsa düşerler, aynı akıştan gelmiyorsa RED ile hesaplanan olasılıkla kuyruğa eklenirler. Bu yöntemde bir kuyruқта yanıt vermeyen akışların yanıt veren akışlardan fazla olması ve karşılaştırılırken seçilme olasılıklarının daha fazla olması yöntemin arkasında yatan fikirdir. Yanıt vermeyen akışlarla ilgilendiği için bu tür akışlardan gelen paketlere daha çok bakılabilir. Tıkanıklık olması durumunda yanıt vermeyen akışlardan arabellekte daha çok paket varsa bu akışları kontrol eder, izlemez [11].

### **2.1.6 GREEN Algoritması**

GREEN yönteminde TCP paketlerinin proaktif şekilde atılması için TCP’nin kararlı davranışı ile temin edilen bilgiler kullanılır. Bu yöntem tıkanıklığı önlemeyi hedefler. Tıkanıklığın tahmin edilmesi yerine aktif bir şekilde tıkanıklığın bulunması ile algoritma çalışmaktadır. Bu yöntemde kuyruk boyutu düşük tutulmaktadır. Bu sayede gecikme (jitter) azalır [8].



## KAYNAKÇA

1. Paket Anahtarlama (2022). *Vikipedi*.  
[https://tr.wikipedia.org/wiki/Paket\\_anahtarlama](https://tr.wikipedia.org/wiki/Paket_anahtarlama)
2. Türk, E. (2019). Devre Anahtarlama ve Paket Anahtarlama  
<https://esenbaharturkay.medium.com/devre-anahtarlama-ve-paket-anahtarlama-fda60601590b>
3. Oktuğ, S. (t.y.) ANAHTARLAMA : DEVRE ANAHTARLAMA. MESAJ ANAHTARLAMA. PAKET ANAHTARLAMA.  
<http://android.eng.ankara.edu.tr/wp-content/uploads/sites/656/2018/02/2-T%C3%BCrk%C3%A7e-kaynak-B%C3%B6l%C3%BCm-2.pdf>
4. Paket Anahtarlama Nedir? (t.y.)  
<https://www.netinbag.com/tr/internet/what-is-packet-switching.html>
5. QoS (port önceliği). (2021). *Vikipedi*  
[https://tr.wikipedia.org/wiki/QoS\\_\(port\\_%C3%B6nceli%C4%9Fi\)](https://tr.wikipedia.org/wiki/QoS_(port_%C3%B6nceli%C4%9Fi))
6. Özkes, S. (2005). EVALUATION OF ACTIVE QUEUE MANAGEMENT ALGORITHMS  
<https://ticaret.edu.tr/uploads/kutuphane/dergi/f7/M00105.pdf>
7. Okuroğlu, B. (2000). TCP/IP'DE KUYRUK YÖNETİM MEKANİZMALARI VE İNTERNET İÇİN KALİTELİ HİZMET
8. Şimşek, M. (2012). PAKET ANAHTARLAMALI AĞLARDA HİZMET KALİTESİ TABANLI ULAŞTIRMA PROTOKOLÜ VE KUYRUK YAPISI GELİŞTİRİLMESİ VE UYGULAMASI  
<https://dspace.gazi.edu.tr/bitstream/handle/20.500.12602/177692/fc8c0233f00a171a587c6b6bcdcb93e.pdf?sequence=1&isAllowed=y>
9. Socrates, C. Beulah Devamalar, P.M. Kannamma Sridharan, R. (2014). Congestion Control for Packet Switched Networks: A Survey  
<http://www.ijsrp.org/research-paper-1214/ijsrp-p3608.pdf>
10. Rexford, J. (t.y.) Queue Management  
<https://www.cs.princeton.edu/courses/archive/spr12/cos461/docs/lec08-queues.pdf>
11. Prakash, S.J. Amalarethinam, G. Raj, G.D. (2011). QoS Congestion Control AQM Algorithms: A Survey.  
[https://www.researchgate.net/publication/269938906\\_QoS\\_Congestion\\_Control\\_AQM\\_Algorithms\\_A\\_Survey](https://www.researchgate.net/publication/269938906_QoS_Congestion_Control_AQM_Algorithms_A_Survey)