

# Bilgisayar Ağlarında Tıkanıklık Denetimi ve Çözüm Yöntemleri

Mehmet ŞİMŞEK, M. Ali AKCAYOL

Gazi Üniversitesi, Mühendislik Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, Maltepe, Ankara.  
[mehmet.simsek@gazi.edu.tr](mailto:mehmet.simsek@gazi.edu.tr), [akcayol@gazi.edu.tr](mailto:akcayol@gazi.edu.tr)

**Özet**— Bu çalışmada, bilgisayar ağlarında tıkanıklıktan kaçınma ve tıkanıklık denetimi üzerine yapılmış çalışmalar ve sonuçları incelenmiştir. Ayrıca, kablosuz ağlar için geliştirilmiş olan TCP sürümleri sunulmuş ve kablosuz ağlar için geliştirilen tıkanıklıktan kaçınma ve tıkanıklık denetimi protokolleri incelenmiştir. Yapılan incelemeler neticesinde tıkanıklık denetimi ve giderilmesi konusundaki açık çalışma alanları ortaya konulmuştur.

**Anahtar kelimeler**— Tıkanıklık, TCP, Protokol, Kablosuz Ağlar

## Congestion Control and Solution Methods in Computer Networks

**Abstract**— In this paper congestion avoidance and control issues and their results has been reviewed. Although, TCP versions, developed for wireless networks are presented and congestion avoidance and control protocols also reviewed. Open issues on congestion avoidance and control were presented as a result of the study.

**Keywords**— Congestion, TCP, Protocol, Wireless Networks

### 1. GİRİŞ

Bilgisayar ağlarında tıkanıklık, ağın performansını etkileyen en önemli unsurdur. Bir bilgisayar ağında fazla yükten dolayı paketlerin bekleme süreleri artarsa, paket kayıpları yaşanır ve ağın etkinliği azalır bu ağ tıkanmış demektir. Ağın etkinliğinin azalması, kaybolan paketlerin yeniden gönderilmesini gerektirmekte ve ağa fazladan trafik yükü getirmektedir [1-3].

Tıkanıklık göreceli bir durumdur. Örneğin paketlerin bir ağda kaybolma olasılığı 1/500 olsun. Gönderdiği paketlerin 1/1000 oranında bir hata ile iletilmesini isteyen bir kullanıcı bu ağa tıkanık derken, gönderdiği paketlerin 1/100 oranında hata ile iletilmesinden memnun olan bir kullanıcı için bu ağ tıkanık değildir. Yine aynı şekilde, ağ elektronik posta göndermek için kullanan bir kullanıcı gönderdiği e-postanın ertesi gün iletilmesinden memnun olabilirken, gerçek zamanlı video aktarımı yapan başka bir kullanıcı aynı ağın tıkalı olduğunu söyleyebilmektedir. Bazı kaynaklarda tıkanıklık, ağ yükünün artması ile kullanıcının istediği performansın düşmesi şeklinde tanımlanmaktadır [3]. Diğer bir tıkanıklık tanımı ise şu şekildedir: Ağdaki aşırı yüklenme sonucunda kuyrukta bekleyen paketlerin sayısının anormal şekilde artması,

yeni gönderilen paketlerin işleme alınamaması ve işleme alınamayan paketlerin yeniden gönderilmesi sonucunda kuyruk uzunluğunun sürekli olarak artması [3, 4].

Temel olarak iki çeşit tıkanıklık önleme yöntemi vardır. Bunlar, rezervasyon tabanlı ve isteğe bağlı metotlardır [4].

Bir kullanıcının ihtiyacı olan hizmet kalitesi için önceden ağ üzerinde kaynak ayrılabilir. Kullanıcı istediği zaman bu kaynakları kullanarak ağ üzerinden bilgi transfer eder. Bu yöntem rezervasyon tabanlı olarak adlandırılabilir. Alternatif olarak kullanıcı gerek duyduğu anda ağdaki kaynakları rezerve etmek isteyebilir. Fakat ağdaki mevcut yük dolayısıyla istediği kalitede hizmet alamayabilir. İkinci metot rezervasyonsuz ağlarda kullanılır. Bu durumda kullanıcılar kendilerini değişen ağ durumuna uyarlamak durumundadırlar.

İsteğe bağlı planda ise ağı kullanan kişiler sürekli olarak ağı gözlemlemeli ve tıkanıklığı önlemek için ağın durumunu değiştirmelidirler. Her iki yöntemin de avantajları ve dezavantajları vardır. Birinci yöntemde hizmet kalitesi garanti edilir fakat kullanıcı sayısı sınırlandırılmak zorundadır. İkinci yöntemde kullanıcı

sayısı sınırlı değildir fakat her kullanıcı aldığı hizmetten ödün vermek durumunda kalabilir.

Rezervasyon tabanlı ve isteğe bağlı yöntemlerde tam bir dışta bırakma söz konusu değildir. İki yöntemin bir arada uygulandığı karma sistemler de tasarlanabilir. Bazı kullanıcılar için kesin bir hizmet kalitesi sağlanması gerekiyor ise o kullanıcılara kaynaklar rezerve edilebilir. Diğer kullanıcılar ise kalan kaynakları paylaşabilirler. Bundan başka, her kullanıcı için kaynakların küçük bir kısmı rezerve edilir. Kalan kaynaklar bir havuzda toplanır ve kullanıcılar bu kaynaklara erişmek için birbirleri ile yarışabilirler [3, 4].

Bir ağda tıkanıklığın olup olmadığı aşağıdaki durumlarla ifade edilir:

- Eğer düğümdeki çıkış tamponu dolu ise ve giriş tamponunda yer yoksa tıkanıklık meydana geldi denebilir.
- Düğüm, çıkış hattını gözlemler. Eğer çıkış hattı üzerinde %80 gibi bir eşik değerinin üstünde yoğunluk varsa bu hat tıkanacak demektir.
- Gidiş-Dönüş gecikme süreleri incelenebilir. Kuyruk uzunluğu arttıysa ve gidiş-dönüş gecikme süresi yükseldiyse tıkanıklık meydana gelebilir demektir.
- Düğüm bir sayaç tutar. Eğer gönderilen paket gereken süre içerisinde kabul edilmezse tıkanıklık vardır.

## 2. TIKANIKLIK DENETİMİ VE GİDERİLMESİ

Bilgisayar ağlarındaki tıkanıklık problemi sistematik bir biçimde ilk olarak Jacobson ve Karels tarafından ele alınmıştır. 1988 yılında V. Jacobson, "Congestion Avoidance And Control" isimli bir makale yayınladılar. Bu çalışmada, BSD (Berkeley Software Distribution) TCP 4.0'da yer alan yedi algoritmadan beşi tanımlandı [2].

Bu yedi algoritma:

- Round-trip time variance estimation.
- Exponential retransmit timer backoff.
- Slow-Start.
- More aggressive receiver ACK policy.
- Dynamic window sizing on congestion.
- Karn's clamped retransmit backoff.
- Fast Retransmit

1993 yılında Lawrence Berkeley Laboratuvarı'ndan S. Floyd ve V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance" isimli bir makale yayınladılar [5]. Bu makalede Random Early Detection (RED) olarak adlandırılan bir kuyruk yapısı tanımlanmaktaydı. Bu kuyruk yapısının bulunduğu gatewayler tıkanıklık olup olmayacağına ortalama kuyruk uzunluğuna göre karar veriyorlardı. Belirlenen bir kuyruk uzunluğunu geçtikten sonra gelen paketler atılıyorlardı veya kuyruk uzunluğunun bir fonksiyonuna göre atılma olasılıkları hesaplanıyordu.

1994 yılında Arizona Üniversitesi'nden L. S. Brakmo ve arkadaşları, "TCP Vegas: New Techniques for Congestion Detection and Avoidance" isimli bir makale yayınladılar [6]. Vegas ismini verdikleri bu yeni TCP versiyonu BSD 4.0'da kullanılan TCP Reno versiyonuna göre %40-70 daha fazla paket üretilmesini (throughput) sağlıyor, paket kayıp oranında da %20-50 arasında azalma sağlıyordu.

1999 yılında M. Allman, V. Paxson ve W. Stevens "TCP Congestion Control" isimli bir Request For Comments (RFC) dokümanı yazdılar [12]. Bu çalışmada TCP'nin 4 tıkanıklık denetimi algoritması olan slow start, congestion avoidance, fast retransmit ve fast recovery algoritmaları detayları ile tanımlanmıştır.

Yine 1999 yılında H. Balakrishnan ve arkadaşları, "An Integrated Congestion Management Architecture for Internet Hosts" isimli bir makale yayınladılar [13]. Bu makaledeki motivasyonları, bilgisayar ağlarındaki tıkanıklık denetimini uçtan uca gerçekleştirmekti. Böylece uzun vadede internetin sürekliliğini sağlamayı amaçlıyorlardı.

2000 yılında Sally Floyd "Congestion Control Principle" isimli bir RFC dokümanı yazdı [14]. Bu doküman daha önceki RFC dokümanlarının bir incelemesi şeklindeydi. Önceki RFC dokümanlarından farklı olarak, yeni transport protokollerinin geliştirilmesi için öneriler, bant genişliğinin bağlantılar arasında adaletli olarak paylaşılabilmesi için öneriler ve tıkanıklık denetimini etkileyen uygulama katmanı konuları yer almaktadır.

Şimdiye kadar anlatılmış olan tıkanıklıktan kaçınma ve tıkanıklık giderme yöntemleri geleneksel kablolu ağlar düşünülerek geliştirilmiştir. 3 numaralı başlık altında, kablosuz ağlar için geliştirilmiş olan TCP sürümleri ve geliştirilmiş bazı tıkanıklık denetimi yöntemlerine yer verilecektir.

## 3. KABLOSUZ AĞLARDA TIKANIKLIK DENETİMİ

Geleneksel TCP'nin kablosuz ağlarda kullanılmasının çeşitli sakıncaları vardır. Bunlar aşağıda özetle anlatılmaktadır.

### *Paket Kayıplarını Yanlış Yorumlama*

Geleneksel TCP, kablolu ağlar için hazırlanmıştır ve paket kayıplarını tıkanıklık belirtisi olarak değerlendirir. Bir paket kaybolduğu zaman tıkanıklık çözümü için geliştirilmiş olan algoritma işletilir. Hareketli tasarsız ağlarda, kablolu ağlara göre çok daha fazla paket kaybı yaşanmaktadır. Tasarsız ağlarda sık sık düğümlerin hareketliliğinden ötürü yol kırımları meydana gelmektedir, radyo sinyallerinin girişimde bulunması, tek yönlü bağlantılar gibi nedenlerden dolayı paket kayıpları yaşanmaktadır. Bu kayıpların tıkanıklık olarak değerlendirilmemesi gerekmektedir.

Hareketli tasarsız ağlarda düğümlerin hareketliliği ile ilgili olarak bir kısıtlama olmadığı için sık sık bağlantı kopmaları yaşanmaktadır. Bir bağlantı koptuktan sonra, yeni bir yolun temin edilmesi gerekmekte ve ağ üzerinde çalıştırılan yönlendirme algoritması bunu yapmaktadır. Yeni yolun kurulma süresi eğer tekrar iletim zaman aşımı aralığı (retransmit timeout interval) RTO'dan büyükse, gönderici düğüm bir tıkanıklık olduğu düşünüp, tıkanıklık algoritması çalıştıracaktır ve kaybolan paketleri yeniden gönderecektir. Bu da bant genişliğinin azalmasına ve bataryanın boş yere kullanılmasına neden olacaktır.

#### Yol Uzunluğunun Etkisi

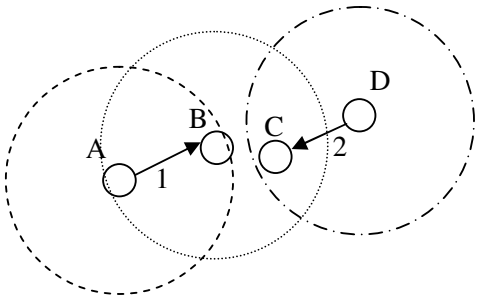
Hareketli tasarsız ağlarda yol uzunluğunun büyük olması demek, hedefe daha fazla düğüm üzerinden geçilerek gidilmesi demektir. Bu da, düğümlerin hareketliliğinden dolayı daha fazla yol kırılmasına yol açar.

#### Tıkanıklık Penceresinin Yanlış Kullanılması

Tasarsız ağlarda bir yol kırılması meydana geldiği zaman tıkanıklık önleme algoritması işletilir ve tıkanıklık pencere boyutu azaltılır ve RTO'yu artırır. Yeni yol kurulduğu zaman, tıkanıklık penceresi buna hemen uyum sağlayamaz. Yeni kurulmuş olan yolun kapasitesi yüksek olmasına rağmen bunun küçük bir kısmı kullanılabilir. Bundan dolayı, sık sık yol kırılmaları hattın verimliliğini düşürür.

#### Asimetrik Bağlantı Davranışı

Hareketli tasarsız ağda kullanılan radyo kanalı farklı özelliklere sahip olabilir. Radyo dalgalarının yayılımının çevresel etkisi, konumdan dolayı meydana gelen çekişme gibi nedenlerden ötürü hat tek yönlü olabilir. Bu da, ACK paketlerinin iletilmesini engeller. Bu durumda, tıkanıklık algoritmasını çalıştırmak mümkün değildir.



Şekil 1. Asimetrik bağlantı davranışı

Şekil 1'de asimetrik bağlantı davranışına bir örnek verilmektedir. Bütün düğümlerin aynı kanalı kullandıklarını düşünelim. İlk bağlantı için A kaynak ve B hedef düğümler, ikinci bağlantı için D kaynak ve C hedef düğümler olsun. A düğümünün gönderdiği paketlere karşılık B düğümünün ACK göndermek istemesi durumunda B düğümünün radyo sinyallerinin kapsama

alanı nedeniyle C düğümü de kapsama alanına girecek ve D-C arası olan bağlantının radyo sinyalleri ile girişim (interference) meydana gelecektir. Böyle bir durumda ortamı ele geçirme çekişmeleri neticesinde B düğümünün ACK gönderme prosedürü düzgün çalışmayacaktır.

Yukarıda sayılan nedenlerde dolayı yeni çalışmalar gereği duyulmuştur.

Chandran ve arkadaşları 1998 yılında "A Feedback Based Scheme for Improving TCP Performance in Ad-Hoc Wireless Networks" isimli bir makale yayınladılar [15]. Bu çalışmada önerilen geri Beslemeli TCP (Feedback-Based TCP, TCP-F), tasarsız ağlarda performansı arttırmak için geleneksel TCP'nin değiştirilmesi ile oluşturulmuştur. Bu protokol, geri besleme tabanlı bir yaklaşım kullanır. TCP-F, oluşabilecek bağlantı kopmalarını kaynak düğümüne bildirebilecek bir yönlendirme yöntemine ihtiyaç duyar. Bu protokolün amacı, bağlantı kopmalarından meydana gelen paket kayıplarını azaltmaktır. Bir paket kaybolduğu zaman tıkanıklık önleme algoritması işletilir ve pencere boyutu küçültülür [15, 16].

G. Holland ve N. Vaidya 1999 yılında, "Analysis of TCP Performance over Mobile Ad Hoc Networks" isimli bir makale yayınladılar [17]. Bu çalışmada önerilen TCP-ELFN (Explicit Link Failure Notification) protokolü, TCP-F ile benzer özelliklere sahiptir. Fakat açık bağlantı başarısızlığı bildirimini yakalaması ve yeniden yol kurulduğunda bunu tespit etmesi açısından farklıdır. ELFN mesajı, bağlantı kopukluğunu tespit eden düğüm tarafından, kaynak düğümüne bu kopukluğu bildirmek için üretilir. Bu, iki yolla olur: ilk yol, ICMP paketi ile "hedefe ulaşılamıyor" mesajı yollamaktır. Diğer yol ise, Yol Hatası mesajı ile bu durumu bildirmektir [16,17].

Dongkyun ve arkadaşları 2001 yılında "TCP-BuS: Improving TCP Performans in Wireless Ad Hoc Networks" isimli bir makale yayınladılar [18]. Bu çalışmada önerilen Tamponlama Yapabilen ve Sıra Bilgisi Kullanan TCP (TCP With Buffering Capability and Sequence Information, TCP-BuS), aradaki düğümün, bağlantı kopukluğunu tespit etme ve bunu kaynak düğümüne geri bildirme yöntemleri konusunda, TCP-F ve TCP-ELFN protokollerine benzer. TCP-BuS, dayanışma tabanlı yönlendirme protokolü üzerine kurulmuştur. Bu yüzden, bu protokolda olan belirli paketleri kullanır. Aktarımın olduğu her ara düğüm, aktarılan paketleri bir tampondan tutar. Aktarılan paket, hemen yok edilmez.

J. W. Kang ve B. Nath 2004 yılında "Resource-controlled MAC-layer Congestion Control Scheme in Cellular Packet Network" isimli bir makale yayınladılar [19]. Çalışmaları diğer çalışmalardan farklı olarak tıkanıklık durumu algılandığında aktarım hızını düşürmek değil daha fazla bant genişliğini kullanmak şeklindeydi. Yaptıkları çalışmada bir hücre içerisinde bulunan hareketli terminallere bir fiziksel kanalın zaman paylaşımı olarak baz istasyonu tarafından kullanıldığını varsayımlardır. Ayrıca her bir hareketli

terminale  $n$  adet fiziksel kanal atandığını varsaymışlardır. Hareketli terminal üzerinde çalışan MAC katmanı paket zamanlayıcısı gönderilecek olan paketleri fiziksel kanallar üzerine eşit olarak dağıtır. Eğer hareketli terminalin belirli bir süre gönderecek paketi yoksa gereksiz olan fiziksel kanallar serbest bırakılır ve ihtiyacı olan hareketli terminaller tarafından kullanılır.

S.W. Ng ve E. Chan 2005 yılında “Equation-based TCP-friendly congestion control under lossy environment” isimli bir makale yayınladılar [20]. Bu çalışmada, paket kaybının sıkça yaşandığı kablosuz ortamlarda TCP penceresinin boyutunun azaltılması ile ilgili değişik yaklaşımlar ortaya konmuştur. Ayrıca, gerçek zamanlı çoklu ortam uygulamalarında, paket kayıpları yaşandığında TCP gibi davranış sergileyebilen biri basit (SRP) diğeri karmaşık (CRP) olarak isimlendirilen iki adet aktarım protokolü önerilmiştir. Bu proc kaybı yaşandığı zaman TCP'nin pencere boyutlandırma prosedürüne benzer bir prosedür izleyerek bant genişliğinin daha adil kullanılmasını sağlarlar. CRP protokolü, paket kayıplarının daha sık yaşandığı kablosuz ortamlarda, SRP protokolü ise paket kayıplarının daha seyrek yaşandığı kablosuz ortamlarda daha iyi sonuçlar vermiştir. Bunun nedenini yazarlar, CRP protokolünde paket iletim zaman aşımı (timeout) değerlerinin de hesaplamaya dâhil edilmesi olarak açıklamaktadırlar.

M. Şimşek 2006 yılında yaptığı “Kablosuz Ağlarda Yönlendirme Protokolü ve Tıkanıklık Denetimi” isimli tez çalışmasında [21], M. Şimşek ve M. A. Akcayol 2008 yılında yaptıkları “Kablosuz Ağlarda Sezgisel Bir Yönlendirme Protokolü Ve Tıkanıklık Denetimi” isimli çalışmalarında [22] hareketli ad hoc ağlarda kullanılabilecek bir tıkanıklık haberdar yönlendirme protokolü önermişlerdir. Yaptıkları çalışmalarında tıkanıklığa karar vermek için düğümün arayüz kuyruk uzunluğunun belirli bir eşik seviyeye ulaşmasını şart koşmuşlardır. Eğer kuyruk uzunluğu eşik seviyeye ulaşmışsa tıkanıklık meydana geleceğine karar verilerek, geliştirilen tıkanıklık önleme yöntemi devreye sokulmaktadır. Tıkanıklığa karar verilmesini sağlayan sözde kod aşağıdaki gibidir

```
if ( KAU < (KMU*8/10) ){ Tıkanıklık yok }
else{ Tıkanıklık meydana gelecek }
```

Yukarıdaki sözde kodda KAU kuyruk anlık uzunluğunu, KMU da kuyruk maksimum uzunluğunu ifade etmektedir. Yapılan karşılaştırma neticesinde eğer tıkanıklık meydana geleceğine karar verilmişse, tıkanıklık olacağını belirleyen düğüm hedef düğüme giden ve arayüz kuyruk uzunluğu düşük olan bir başka yol arar. Yeni yol bulunursa iletişim bu yola taşınır, yeni bir yol bulunamazsa düğüm aynı bağlantı üzerinde kendinden önce bulunan düğüme bir tıkanıklık bildirimini gönderir. Tıkanıklık bildirimini alan düğüm aynı işlemleri tekrar eder. Bu işlem yeni bir yol bulunana kadar veya tıkanıklık bildirimini kaynak düğüme ulaşana kadar devam eder. Kaynak düğüm de yeni bir yol bulamazsa diğer tıkanıklık

BİLİŞİM TEKNOLOJİLERİ DERGİSİ, CİLT: 1, SAYI: 3, EYLÜL 2008  
giderme yöntemleri devreye girer. Yukarıda anlatılan sürecin algoritması aşağıdaki gibidir.

```
While (Tüm paketler bitene kadar)
{
    if (K1<E_D)
    {
        Paketi ilk arayüzden yolla
    }
    else
    {
        yeni_yol ara;
        if (yeni yol varsa)
        Paketi ikinci arayüzden yolla;
        else
        {
            Paketi ilk arayüzden yolla;
        }
    }
}
Önceki düğüme tıkanıklık bildirimini gönder.
```

Geliştirilmiş olan bu yeni yönlendirme protokolü ile tıkanıklık nedeni ile yaşanan paket kayıp miktarlarında %10-15 arası iyileştirme sağlanmıştır. Ayrıca, ara düğümlerde yapılan tıkanıklık önleme işlemi sayesinde tıkanıklık problemi yerel olarak çözülmeye çalışılmıştır.

D. Kliazovich ve F. Granelli 2006 yılında “Cross-layer congestion control in ad hoc wireless networks” isimli bir makale yayınladılar [23]. Bu çalışmalarında kablosuz yerel alan ağlarında çalışan katmanlar arası bir tıkanıklık kontrol yöntemi önermişlerdir. Bu yöntem, kullanılan gerçek bant genişliği ve paket gecikmeleri gibi veri bağı katmanı bilgilerini kullanarak TCP'nin pencere boyutlandırma davranışını belirlemektedir. Yaptıkları benzetimler neticesinde, geliştirdikleri TCP modifikasyonunun standart TCP, TCP Vegas ve TCP-Westwood sürümlerinden daha iyi throughput sağladığını görmüşlerdir.

T. Tunalı ve K. Anar 2006 yılında “Adaptive available bandwidth estimation for internet video streaming” isimli bir makale yayınladılar [24]. Bu çalışmalarında, internet üzerinden video aktarımı için kullanılabilir bant genişliği ölçüm yöntemi önermişlerdir. Yaptıkları çalışma ile kablosuz ortamlardan kaynaklanan paket atılma durumu ile tıkanıklıktan kaynaklanan paket atılma durumu ayırt etmeye çalışmışlardır. Kullanılabilir bant genişliğini çok kısa sürede ölçebilen gözlemleme paketleri ile tıkanıklık durumuna karar verilerek video aktarımını bant genişliğine göre ayarlamak mümkün olmaktadır. Geliştirdikleri yöntemin kısıtları, gözlemleme paket büyüklüğü ve gözlemleme sıklığından kaynaklanan yük olarak sıralanabilir. Ayrıca, deneysel sonuçlar neticesinde elde ettikleri veriler ışığında şu söylenebilir: kullanılabilir bant genişliği dinamik bir süreç olsa da kısıtlı sürelerde bant genişliğini ölçerek, bunu gerçek kullanılabilir bant genişliğini hesaplamada kullanmak mümkündür.

S. Toklu 2007 yılında yaptığı “Wap Trafikinde Tıkanıklık Denetimi Ve Ulaşım Katmanı Protokolleri” isimli tez çalışmasında önceliklendirilmiş bir kuyruk yapısı önermiştir [25]. Bu kuyruk yapısı WAP gatewaylerde çalışmakta olup gelen trafiğin TCP veya UDP olmasına göre bir önceliklendirme yapmaktadır. Gateway üzerindeki kuyruk uzunluğu belirlenmiş bir eşik değerin üzerine çıktığı anda yöntem devreye girmekte ve alınan UDP paketlerini atmaktadır. Buradaki temel motivasyon, TCP paketleri ile taşınan bilgilerin daha önemli olması ve UDP paketlerinin atılmaları durumunda kaynak tarafından yeniden gönderilmemeleridir. Tıkanıklığı engellemek için TCP paketleri kasıtlı olarak atılmış olsalardı, TCP paketlerini oluşturan kaynak düğüm atılan paketler için ACK alamadığından bu paketleri yeniden gönderecek ve gatewaydeki kuyruk anormal biçimde artacaktır. Yukarıda anlatılan sürecin sözde kodu aşağıdaki gibidir.

```

if ( KAU < (EŞİK) )
{
    Gelen paketi kuyruğa al
}
else
{
    if (alınan paket == TCP) Gelen paketi kuyruğa al
    else Gelen paketi at
}

```

Yukarıdaki kodda KAU kuyruk anlık uzunluğu ve EŞİK de belirlenen kuyruk eşik değeridir.

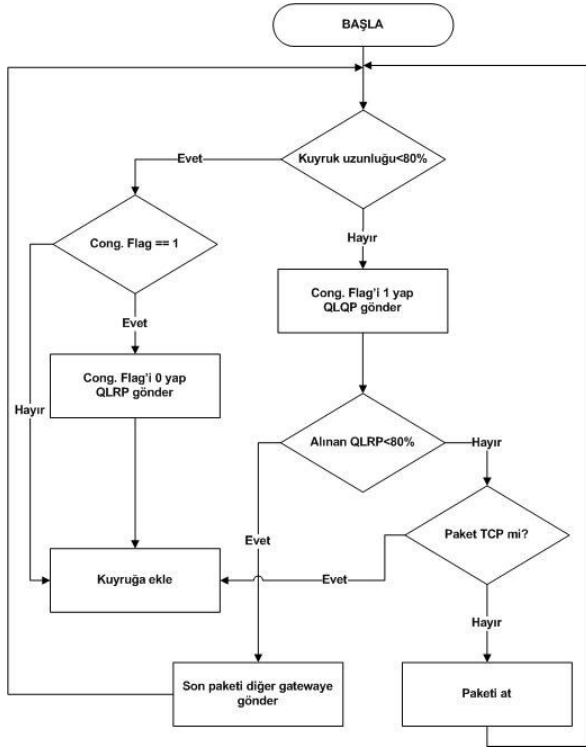
Önerilmiş olan bu yeni kuyruk yapısı ile daha çok TCP paketinin ağda taşınması sağlanmış, bir bakıma TCP bağlantılarının servis kalitesi artırılmıştır.

J. Chen ve ark. 2008 yılında “TCP with delayed ack for wireless networks” isimli bir makale yayınladılar [26]. Yaptıkları çalışmalar ve analiz sonuçları ile optimal TCP throughput elde etmek için topolojinin ve bağlantı desenlerinin birlikte değerlendirilerek alıcı tarafta TCP pencere boyutunun belirlenmesi gerektiğini ortaya koydular. Ayrıca, kablosuz ortamlarda TCP bağlantısının kurulduğu yolun uzunluğu arttıkça sinyal girişimlerinin de artacağı ve dolayısı ile TCP pencere boyutunun belirlenmesinde yol uzunluğunun da belirleyici bir kriter olması gerektiği ortaya konmuştur.

S. Toklu ve arkadaşları 2008 yılında yaptıkları “A Priority Based Protocol And Load Balancing For Queue Management On Wireless Networks” isimli çalışmalarında [25]’deki yöntemi geliştirerek WAP gatewayler arası yük dağılımı ve yönlendirme yapmışlardır [27]. Önerilen sistem gateway üzerindeki kuyruk uzunluğunun belirli bir eşik seviyesine ulaşması ile başlar. Önerilen yöntemin çalışma şeması Şekil 2’de gösterilmektedir. Queue Length Query Packet – QLQP, diğer gatewaylerdeki kuyruk doluluk oranının sorgulanması için bir oluşturulan bir UDP paketidir. QLRP – Queue Length Report Packet de QLQP paketine karşılık olarak gönderilen ve gatewaydeki o anki kuyruk uzunluğunu bildiren bir UDP paketidir.

Şekil 2’de de gösterildiği gibi herhangi bir düğümde gelen paketler öncelikle WAP gatewaye gelmektedir. Kuyruk doluluk oranı %80’den küçükse paket kuyruğa alınmakta ve sunucuya gönderilmektedir. Kuyruk doluluk oranı %80’den büyükse Cong.Flag 1 yapılmakta ve QLQP paketi bütün Gatewaylere gönderilmektedir. Daha sonra, kuyruk doluluk oranı %80’den küçük olan bir Gateway varsa alınmış olan paket bu Gatewaye gönderilmektedir. Eğer yoksa paketin TCP paketi olup olmadığı incelenmekte ve TCP paketi değilse atılıp, TCP paketi ise kuyruğa alınmaktadır. Cong.Flag değişkeninin kullanılmasının nedeni, %80 seviyesine geçişte ve bu seviyeden düşüşte sürekli olarak QLQP paketi gönderilmesini engellemektir.

G. J. Yua ve C. Y. Chang 2008 yılında “Congestion control of bluetooth radio system by piconet restructuring” isimli bir makale yayınladılar [28]. Bu çalışmalarında bluetooth haberleşmesi esnasında yaşanabilecek tıkanıklık durumlarına bir çözüm getirmişlerdir. İki veya daha fazla bluetooth cihazının bir araya gelmesi ile piconet, iki veya daha fazla piconetin bir araya gelmesi ile de scatternet yapısı oluşur. Her bir piconete bir master cihaz bulunur. Diğer cihazlar slave olarak adlandırılırlar ve aralarındaki haberleşmeyi master cihaz üzerinden yaparlar. Piconetlerin aralarında haberleşebilmesi için bluetooth cihazlarından bazıları aktarıcı (relay) görevi görür. Eğer trafik akışı düzenlenmezse scatternet üzerinde ciddi tıkanıklık durumları meydana gelebilir. Bu çalışmada piconetin yeniden yapılanması ile akış kontrolü sağlanmış, bluetooth cihazlarının rol değişimi gerçekleştirilmiştir. Böylelikle tıkanıklık problemi giderilmeye çalışılmıştır. Geliştirdikleri yöntemin avantajları: haberleşme darboğazı çözümü, haberleşme yolunun kısaltılması, aktarım gecikmesinin azaltılması ve piconetin yaşam süresinin uzatılması olarak sıralanabilir. yazarlar geliştirdikleri piconet yeniden yapılandırma protokolü ile master cihazın yükünü paylaştırmışlardır. Piconet içerisinde en yüksek miktarda veriyi üreten cihaz yardımcı master cihaz olarak belirlenmiş ve slave cihazlar ile doğrudan haberleşmesi sağlanmıştır. Bu sayede master cihazın yükü azaltılmıştır. Geliştirilen yöntemin sağladığı diğer bir fayda da master cihazın güç tüketiminin azaltılmasıdır.



Şekil 2. [22] çalışmasında önerilen yöntemin çalışma şeması

#### 4. SONUÇ

İncelenen çalışmalar ve dokümanlar neticesinde bilgisayar ağlarında tıkanıklık denetimi ve giderilmesi konusunun açık bir çalışma alanı olduğu görülmektedir. Özellikle kablosuz ağlara yönelik tıkanıklık denetimi ve giderme yöntemlerinin geliştirilmesi ve yeni yöntemler önerilmesi gerekmektedir. Yapılan çalışmaların çok büyük bir kısmının TCP tabanlı olduğu görülmektedir. Bu çalışmaların UDP tabanlı bağlantılar için kullanılmayacağı açıktır. UDP tabanlı uygulamalar için tıkanıklık denetimi ve giderme konusu da açık olan bir çalışma alanıdır. Ayrıca tıkanıklıktan haberdar uygulamalar geliştirilmesi tıkanıklık sorununa yeni yaklaşımlar getirecektir.

Kablolu ağlar ile ilgili birçok standart tıkanıklık denetimi ve giderme yöntemi olmasına rağmen kablosuz ağlara özel bir standart bulunmamaktadır.

Son zamanlarda gelişen GSM teknolojileri ile birlikte artması beklenen çoklu ortam uygulamaları ile GSM şebekeleri üzerinde yoğunluk yaşanması beklenmektedir. GSM şebekeleri ile ilgili olarak tıkanıklık denetimi ve giderme konusu da önu açık bir çalışma konusudur.

#### 5. KAYNAKLAR

- [1] V. Jacobson, "Congestion Avoidance and Control", Proceedings of ACM SIGCOMM, Stanford, Palo Alto California, 314-329, 16-18 August, 1988.
- [2] W. R. Stevens, "TCP Slow-Start, Congestion Avoidance, Fast Retransmission, and Fast Recovery Algorithms", IETF RFC 2001, Virginia, 3-5 (1997).

- BİLİŞİM TEKNOLOJİLERİ DERGİSİ, CİLT: 1, SAYI: 3, EYLÜL 2008
- [3] S. Keshav, "Congestion Control in Computer Networks", Doktora Tezi, Bilgisayar Bilimleri, University of California at Berkeley, 1991.
  - [4] M. Welzl, "Network Congestion Control, managing internet traffic", Wiley, New Jersey, A.B.D., 2005.
  - [5] S. Floyd, V. Jacobson, "Random early detection gateways for congestion avoidance", IEEE/ACM Transactions on Networking (TON), 1(4), 397 – 413, 1993
  - [6] L. S. Brakmo, S. W. O'malley ve Larry L. Peterson, "TCP Vegas: New techniques for congestion detection and avoidance", ACM SIGCOMM Computer Communication Review, 24(4), 24-3, 1994
  - [7] R. Jain., "A Delay-Based Approach for Congestion Avoidance in Interconnected Heterogeneous Computer Networks", ACM Computer Communication Review, 19(5), 56-71, 1989.
  - [8] Z. Wang, J. Crowcroft, "A New Congestion Control Scheme: Slow Start and Search (Tri-S)", ACM Computer Communication Review, 21(1), 32-43, 1991.
  - [9] Z. Wang, J. Crowcroft, "Eliminating Periodic Packet Losses in 4.3-Tahoe BSD TCP Congestion Control Algorithm", ACM Computer Communication Review, 22(2), 9-16, 1992.
  - [10] J. C. Hoe, "Improving the start-up behavior of a congestion control scheme for TCP", Proceedings of ACM SIGCOMM, Stanford Palo Alto California, 270-280, 28-30 August, 1996
  - [11] Web: [www.cs.berkeley.edu/~kfall/EE122/lec22/sld008.htm](http://www.cs.berkeley.edu/~kfall/EE122/lec22/sld008.htm)
  - [12] M. Allman, V. Paxson ve W. Stevens, "TCP Congestion Control", IETF RFC 2581, Virginia, 7-9, 1999.
  - [13] H. Balakrishnan, H. S. Rahul ve S. Seshan, "An integrated congestion management architecture for Internet hosts", ACM SIGCOMM Computer Communication Review, 29(4), 175-187, 1999
  - [14] Floyd S., "Congestion Control Principle", IETF RFC 2914, Virginia, 6-7, 2000.
  - [15] K. Chandran, S. Raghunathan, S. Venkatesan, ve R. Prakash, "A Feedback-Based Scheme for Improving TCP Performans in Ad Hoc Wireless Networks", IEEE Personal Communications Magazine, 8(1), 34-39, 2001,
  - [16] C. S. R. Murthy, B. S. Manoj, Ad-Hoc Wireless Networks-Architectures And Protocols, Prentice Hall, New Jersey, A.B.D., 2004.
  - [17] G. Holland ve N. Vaidya, "Analysis of TCP Performance over Mobil Ad Hoc Networks", Proceedings of ACM MOBICOM, Seattle Washington, A.B.D., 219-230, 15-19 August, 1999.
  - [18] Dongkyun Kim, C. K. Toh, Y. Choi, "TCP-BuS: Improving TCP Performans in Wireless Ad Hoc Networks", Journal of Communications and Networks, 3(2), 12-14, 2001.
  - [19] J. Kang, B. Nath, "Resource-controlled MAC-layer Congestion Control Scheme in Cellular Packet Network", IEEE Vehicular Technology Conference, Milano, İtalya, 1988-1992, 17-19 Mayıs, 2004
  - [20] S.W. Ng, E. Chan, "Equation-based TCP-friendly congestion control under lossy environment", Journal of Systems Architecture, 51(9), 542-569, 2005
  - [21] M. Şimşek, Kablosuz Ağlarda Yönlendirme Protokolü ve Tıkanıklık Denetimi, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, 2006
  - [22] M. Şimşek, M. A. Akçayol "A Heuristic Routing Protocol And Congestion Control At Wireless Networks", J. Fac. Eng. Arch. Gazi University, 23(2), 57-63, 2008.
  - [23] D. Kliazovich, F. Granelli, "Cross-layer congestion control in ad hoc wireless networks", Ad Hoc Networks, 4(6), 687-708, 2006
  - [24] T. Tunali, K. Anar, "Adaptive available bandwidth estimation for internet video streaming", Signal Processing: Image Communication, 21(3), 217-234, 2006
  - [25] S. Toklu, "Wap Trafığında Tıkanıklık Denetimi ve Ulaşım Katmanı Protokolleri", Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, 2007
  - [26] J. Chen ve ark., "TCP with delayed ack for wireless networks", Ad Hoc Networks, 6(7), 1098-1116, 2008
  - [27] S.Toklu, M. Şimşek, O. Yıldız, İ. Bay, A. Şimşek, M. A. Akçayol, "A Priority Based Protocol And Load Balancing For Queue Management On Wireless Networks ", WORLDCOMP08(ICWN'08), Las Vegas Nevada, A.B.D., 14-17 July, 2008. (baskıda)
  - [28] G. J. Yua, C. Y. Chang, "Congestion control of bluetooth radio system by piconet restructuring ", Journal of Network and Computer Applications, 31(3), 201-223, 2008