

Bulut Bilişimde GreenCloud QoSbox Benzetimi

GreenCloud Simulation QoSbox in Cloud Computing

Shafiullah Malekzai, Doğan Yıldız, Serap Karagöl

Elektrik ve Elektronik Mühendisliği Bölümü, Ondokuz Mayıs Üniversitesi, Samsun, Türkiye
shafimalekzai001@gmail.com, {dogan.yildiz, serap.karagol}@omu.edu.tr

Özetçe—Bulut bilişim, ağ bant genişliği ihtiyacının artmasıyla yerel kaynakların, erişim gereksinimlerini karşılamakta yetersiz kalması sonucu ortaya çıkan bir teknolojidir. Bu teknolojiye hizmetlerin çeşitliliği fiziksel yerleştirmedeki kısıtlamaları ortadan kaldırmıştır. Hizmetlerin çeşitliliği aynı zamanda sanal donanım veya yazılım türleri ile kullanılabilirlik kolaylığına sahiptir. GreenCloud benzetimcisi; anahtarlar, yönlendiriciler, bağlantılar...vb. gibi veri merkezi sanallaştırılmış bileşenleri ile enerji duyarlı bulut bilişim uygulamalarının performans sonuçlarını gösteren, Network Simulator (NS-2) gibi bulut bilişim benzetimcilerinin genişletilmiş versiyonudur. Bu çalışmada bulut bilişim ortamında ağ performans ve davranışları, QoSbox tabanlı GreenCloud benzetimcisi yardımıyla gözlemlenmiş ve bulut kullanıcılarının talepleri başına sanallaştırılmış ağ sağlayan bir bulut mimarisi tanıtılmıştır.

Anahtar Kelimeler — Bulut Bilişim; GreenCloud; NS-2; QoSbox.

Abstract— Cloud computing was introduced as a new technology when local resources could not any more fulfill access requirements along with the increase of network bandwidth and variety of services breaks constraints of physical placement and deliver diversities of services in its virtualized hardware and software type. GreenCloud simulator as an extension of the Network Simulator (NS-2) cloud computing simulators, execute an energy-aware cloud computing scenario with data center virtualized components (switches, routers, links, etc.). This paper converses a cloud architecture that delivers virtualized network as per demand from cloud user, with the explanation of QoSbox-based GreenCloud simulation that to demonstrate the performance and behavior of the network performance in cloud computing environment.

Keywords — Cloud Computing; GreenCloud; NS-2; QoSbox.

I. GİRİŞ

Bulut bilişim kavramı 21. yüzyılın başından itibaren ele alınmaya başlanan bir kavramdır. Bulut bilişim; veri depolama, yazılım uygulamaları, yapay zeka, işlem gücü ve işlem kapasitesi gibi ihtiyaçlara ve uygulamalara

internet ortamı üzerinden erişime imkan tanımaktadır. Bireyler ve işletmeler için hiyerarşik hizmetleri aracılığıyla talebe bağlı bilişim hizmeti sağlamaktadır. Bulut bilişim; dağıtılmış hesaplama, paralel hesaplama ve dağıtımli hesaplama kavramlarının geliştirilmiş şeklidir. Bulut bilişim için henüz kesin bir tanımlama mevcut olmamasına rağmen bu kavram günümüzde ticari ve teknik açılardan ele alınmaktadır [1, 2]. Bulut bilişimin en dikkat çekici özelliği kullanıcıların sadece yazılım hizmeti kullanımları kadar ödeme yapmaları, donanım harcamaları için ödeme yapmamalarıdır. ‘Ne kadar kullanım o kadar ödeme’ mantığı dolayısıyla hasılat getiren dilimlenmiş servisler pratik olarak üçe ayrılmaktadır:

1. Hizmetsel Yazılım (Software as a Service (SaaS))
2. Hizmetsel Platform (Platform as a Service (PaaS))
3. Hizmetsel Altyapı (Infrastructure as a Service (IaaS))

Tüm bu hizmetler kullanıcılara uygulamaları çevrimiçi çalıştırma ve verileri çevrimiçi saklama imkanı vermektedir. Bununla birlikte bu hizmetlerin her biri kullanıcıya farklı seviyelerde denetim ve esneklik imkanı sağlamaktadır.

En basit anlamda SaaS, kullanıcılara hali hazırda var olan çevrim içi uygulamaları çalıştırabilme imkanı sağlamaktadır. Bulut bilişimde SaaS hizmeti yardımıyla kullanıma hazır uygulamalara internet üzerinden çok kolay bir şekilde erişilebilmektedir. Örneğin; Google, Microsoft ve Zoho şirketlerinin her biri bir çevrimiçi kelime işlemcisi, bir hesap çizelgesi ve bir sunum paketi (Microsoft Office Web Uygulamaları) sunmaktadır. Pixlr, Jay Cut, Aviary gibi bazı yaratıcı SaaS araçları; ticari uygulamalarda kullanılmak üzere Employease, Netsuite, Salesforce tarafından sunulmuştur [3].

SaaS’ın en büyük avantajı, bütün uygulamaların ya bedava ya da aboneliğe göre ücretlendirme esaslı olmasıdır. SaaS hizmetine internet üzerinden herhangi bir bilgisayardan erişilebilir ve SaaS, işbirlikçi çalışma için mükemmel bir hizmet çeşididir. Ancak genel uygulamalar birçok durumda ticari uygulamalar için uygun değildir.

Diğer yandan PaaS; kullanıcılara, tedarikçiye özel araçlar ve diller kullanarak kendi özel bulut uygulamalarını oluşturma imkanı vermektedir. Bu ise

kullanıcılara kendilerine ait yeni çevrimiçi uygulamalarını oluşturabilecekleri bir alan sunmaktadır. Örneğin, bir Google ürünü olan App Engine kullanıcıya Google altyapısındaki kendi web uygulamasını geliştirme ve çalıştırma ortamı sunmaktadır. Aynı zamanda Microsoft da bu imkanları kullanıcıya Azure isimli uygulamasıyla kendi bulut alanında sunmaktadır.

Salesforce da force.com isimli önerisiyle, PaaS konusunda önemli bir konumdadır. Hem App Engine hem de force.com, ücretsiz olarak kullanıcıya kendi uygulamasını oluşturma ve oluşturduğu bu uygulamadan başka kullanıcıların da faydalanmasına imkan tanımaktadırlar.

IaaS kullanıcıya herhangi bir uygulamayı kendi istediği bir bulut donanımında, sağlayıcının bulut donanımına erişim sağlayarak ihtiyacına göre çalıştırabilme imkanı sunar. Bu ise Bilişim Teknolojisi (Information Technology (IT)) maliyetini azaltmak için var olan uygulamanın bir şirketin veri merkezinden taşınabilmesi anlamına gelmektedir.

Bulut altyapısının en temel birimi sunucudur. Günümüzde sunucular fiziksel ve sanal olabilmektedir. Fiziksel sunucular münferit kişisel bilgisayarlardır. Ancak sanal sunucular ise birçok kullanıcıyı sanallaştırma adı verilen işlem ile yazılım kontrollü olarak gerçek fiziksel sunuculara aktarma işlevini gerçekleştirirler [3].

Ağda hizmet farklılaşması konuları, Servis Kalitesi (Quality-of-Service (QoS)) kavramı dahilinde incelenmektedir. QoS konusundaki sorunların, ağın kapasitesinin artırılarak çözülebileceği düşünülebilir. Ne yazık ki, bu durum sistemi çok daha zor sorunlarla karşı karşıya bırakabilir. Uçtan uca trafik akışı tarafından alınan QoS, uçtan uca güzergahında en küçük kapasiteli (dar boğaz) bağlantıdan alınan QoS ile sınırlandırılmıştır. Bu nedenle, bazı bağlantıların kapasitelerini arttırmak, sadece dar boğazın yerini ağın diğer taraflarına, yani çekirdekten kenara doğru değiştirecektir. Son birkaç yıldır, herhangi bir QoS hizmeti olmadan internet ağ bağlantıları kapasitesi giderek artmıştır ve bu durum önerilen çözümün beklenenin aksine bu bağlantılardaki trafiğin daha da artmasına sebep olmuştur.

Günümüzde, bilgisayarların artan işlem gücü ve yukarıda belirtilen ölçeklenebilirlik konularının yardımıyla bilgisayarlar, ölçeklenebilir QoS mimarilerinin prototiplerini uygulamak için yönlendiriciler gibi kullanılabilir.

GreenCloud benzetimcisi, veri merkezinde neredeyse tüm sanallaştırılmış altyapı unsurları ile enerji farkındalıklı bulut bilişim senaryoları uygulamak ve iş yükü eğilimli, önceliği güç tüketimi ve maliyeti olan benzetim mekanizmasını bulut bilişim ortamında yönetir [4, 5].

Bu çalışmada, UNIX tabanlı bilgisayarda trafik sınıflarına sekme başı hizmet garantisi sağlayan yapılandırılabilir IP yönlendiricisi olan QoSbox [6] yapısının tasarımı ve uygulanması incelenmiştir.

II. QOSBOX UYGULAMASI İÇİN GENEL BİLGİLER

A. Bulut Bilişimde QoS

QoS'un bulut ortamındaki farklı dört işlevi yerine getiren bileşenleri mevcuttur. Bunlar, kaynak sağlayıcılar (Amazon AWS, Microsoft Windows Azure ve Google bulut), ekipman sağlayıcıları (Katman2 ve Katman3 tipi bağlantılar), kaynak tüketici (Kullanıcılar) ve aracı yazılım (kaynak sağlayıcı ve tüketici arasındaki bağlantı) şeklindedir [7].

B. GreenCloud Benzetimcisinde QoSbox

QoSbox ve QoS mimarisi arasındaki temel fark, QoSbox'ın istenilen hizmet garantilerini uygulamak için artımlı bir şekilde bir ağ dağıtması için izin veren trafik şekillendirme veya kabul denetimi gibi dış bileşenlere ihtiyaç duymamasıdır. Bunun yerine, dinamik paket yönlendirme ve bırakma kararını anlık gelen trafiğine göre vermektedir. QoSbox aynı anda hem mutlak sınırları hem de oransal hizmet garantilerini; kuyruk gecikmesi, kayıp oranları ve iş/zaman oranı konularında zorlayabilir [8].

Aslında; QoSbox her bir sekme servisi için trafik akışı, gecikmeler ve kayıplar konusundaki farklılaşma sınıflarını belirleyen yapılandırılabilir IP yönlendiricisidir [9]. QoSbox'taki hizmet farklılaşmasını sağlayan mekanizmalar, QoSbox'ın çıkış kuyruklarındaki paket zamanlaması ve paket bırakma işlemleridir.

QoSbox, NS-2 benzetim ortamını temel alan GreenCloud benzetimcisinin dahilindedir. Ortak Arabellek Yönetimi ve Planlama (The Joint Buffer Management and Scheduling (JoBS)); NS-2 benzetimcisinde görevi, her bir düğüm için bağımsız gecikme farklılaşması ve kesin, göreceli kayıp oranı sağlamak olan anahtar uygulamalardan biridir [8]. Bu nedenle, veri iletimi sırasında, JoBS algoritması her sekme için hizmet vermektedir. Örneğin, bu çalışmada QoS kısıtlamalarına ihtiyaç duyulmuştur ve QoS kısıtlamaları aşağıdaki gibi üçe ayrılır:

- Sınıf1 gecikmesi $\approx 2 \times$ Sınıf2 gecikmesi
- Sınıf1 gecikmesi $\approx 10^{-1} \times$ Sınıf3 gecikmesi Kayıp oranı
- Sınıf3 gecikmesi ≤ 5 ms

Yukarıdaki üç kısıtlamanın ilk ikisi göreceli kısıtlamalara aittir. Kısıtlamalar, herhangi mutlak ve göreceli kısıtlamaların kombinasyonu olarak ele alınabilir. Özel olarak, JoBS aşağıdaki gibi sıralanan beş kısıtlama türlerini tamamen destekler:

- Göreceli Gecikme Kısıtlaması (Relative Delay Constraint (RDC)): Bu kısıtlama, sınıflar arasındaki orantısal gecikme farklılaşmasına dikkat çeker. Örneğin, Sınıf1 ve Sınıf2 arasındaki ilişki aşağıdaki gibi olacaktır:
$$\frac{\text{sınıf2 gecikmesi}}{\text{sınıf1 gecikmesi}} \approx \text{Sabit sayı} \quad (1)$$
- Mutlak Gecikme Kısıtlaması (Absolute Delay Constraint (ADC)): Bu kısıtlama SınıfN'deki ADC

gereksinimli gecikmenin GecikmeN (d_N)’deki gecikme değerinden daha iyi bir değer sağlaması gerektiğine işaret eder.

- İlişkili Kayıp Oranı Kısıtlaması (Related Loss Rate Constraint (RLC)): Bu kısıtlama sınıflar arası ilintili kayıp oranı değişimine işaret eder.
- Mutlak Kayıp Oranı Kısıtlaması (Absolute LossRate Constraint (ALC)): Bu kısıtlama SınıfN’deki ALC gereksinimli gecikmenin Kayıp oranıN (L_N) eşik değerinden daha iyi bir değer sağlaması gerektiğine işaret eder.
- Mutlak Oran Kısıtlaması (Absolute Rate Constraint (ARC)): Bu kısıtlama SınıfN’deki ARC gereksinimli çıkış değerinin μ_N gibi bir alt değere sahip olması gerektiğine işaret eder.

JoBS algoritmasındaki QoS kısıtlamaları için aşağıdaki karşılaştırmalar yapılabilir.

- ALC > ADC
- ARC > Görelî Kısıtlamalar

JoBS tarafından mutlak ve görelî kısıtlamalar aynı anda ortaya çıkarılmamışsa, öncelik mekanizması ADC’nin dikkate alınacağına işaret eder.

III. QOS TABANLI AĞ İÇİN BENZETİM SONUÇLARI

A. GreenCloud Benzetimi

Bu benzetim, GreenCloud Virtual Machine (VM) makinası ile [10] çalışmasında gösterilmiştir. Bu benzetim VM ortamında faaliyet göstermektedir (VMware workstation) ve Linux OS işletim sisteminde çalışmaktadır. (home/greencloud/src/scripts/) klasöründeki ‘topology.tcl’ dosyasından, veri merkezi ağ topolojisine ulaşılmaktadır.

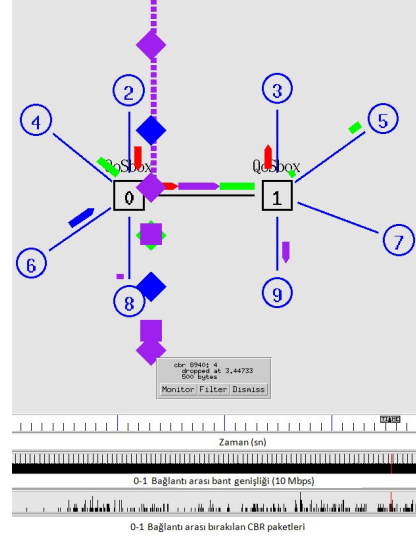
B. QoSbox Benzetimi

Bulut benzetiminin performans değerlendirmesi GreenCloud benzetimi kullanılarak yapılmıştır. Benzetimin temel amacı, QoS garantili GreenCloud benzetiminde QoSbox yerleştirerek uygun bulut bilişim senaryoları altında QoS özelliklerinin sağlanıp dağıtılmasıdır. Çalışmanın temel hedefleri:

- GreenCloud benzetiminde QoSbox kullanımı;
- Ağ topolojisini (kullanıcı sayısını) değiştirerek, ağ performansının etkilenip etkilenmediğini görmek;
- Farklı kullanıcılar için farklı hizmet türlerini tanımlayıp ağ performansının etkilenip etkilenmediğini görmek.

QoSbox benzetimine ‘jobs-lossdel.tcl’ dosyasından ‘/home/greencloud/greencloud/build/ns2.35/tcl/ex/jobs’ adresi üzerinden ulaşılır. Sonuçlar *ns jobs-lossdel.tcl* komutunun çalışması ile elde edilmektedir. Aşağıdaki Şekil 1’de gösterildiği gibi QoSbox benzetimi üretilmiş olup veri akışı Yönlendirici 0’dan Yönlendirici 1’e (soldan sağa) 4 akışın yönetimi ile iletilmektedir. Veri akışı kaynaktan alıcıya doğrudur. Bir tek akış, bir kaynağı

(çift düğüm olarak işaretlenen) ve bir alıcıyı (tek düğüm olarak işaretlenen) iki terminalde yansıtmaktadır. Şekil 1’de dört farklı renk dört farklı akışı göstermektedir. Mesela, akışlar düğüm 2’den 3’e, 4’ten 5’e, 6’dan 7’ye ve 8’den 9’a akmaktadır.



Şekil 1. Dört akışlı QoSbox benzetim sonucu

Şekil 1’de belirtilen QoSbox benzetim topoloji tanımlaması aşağıdaki gibidir. Aynı zamanda, ‘jobs-lossdel.tcl’ dosyası topolojinin nasıl düzenlediğini tanımlamaktadır.

```
# nam stuff
$ns duplex-link-op $source(1) $score_node(1) orient 300deg
$ns duplex-link-op $source(1) $score_node(1) color "blue"
$ns duplex-link-op $source(2) $score_node(1) orient 330deg
$ns duplex-link-op $source(2) $score_node(1) color "blue"
$ns duplex-link-op $source(3) $score_node(1) orient 30deg
$ns duplex-link-op $source(3) $score_node(1) color "blue"
$ns duplex-link-op $source(4) $score_node(1) orient 60deg
$ns duplex-link-op $source(4) $score_node(1) color "blue"
$ns duplex-link-op $score_node(1) $score_node(2) orient right
$ns duplex-link-op $score_node(1) $score_node(2) color black
$ns duplex-link-op $score_node(1) $score_node(2) queuePos 0.5
$ns duplex-link-op $score_node(2) $sink(1) orient 60deg
$ns duplex-link-op $score_node(2) $sink(1) color "blue"
$ns duplex-link-op $score_node(2) $sink(2) orient 30deg
$ns duplex-link-op $score_node(2) $sink(2) color "blue"
$ns duplex-link-op $score_node(2) $sink(3) orient 330deg
$ns duplex-link-op $score_node(2) $sink(3) color "blue"
$ns duplex-link-op $score_node(2) $sink(4) orient 300deg
$ns duplex-link-op $score_node(2) $sink(4) color "blue"
$ns color 1 red
$ns color 2 green
$ns color 3 blue
$ns color 4 purple
puts "ngol'n"
$ns run
```

Aynı zamanda sınıf sayısına da eşit olan akış sayıları aşağıdaki gibi yapılandırılabilir:

```
# Number of monitored flows (equal to # of classes)
set N_CL 4
puts "Number of classes: $N_CL"

set N_USERS 4
puts "Number of flows: $N_USERS"

set START_TM 0.0
puts "Monitored flows start at: $START_TM"

set max_time 10.0
puts "Max time: $max_time (sec)"
```

Her bir akışta konumlandırılmış birden çok trafik sınıfı (hizmet çeşidi) bulunabilmektedir. Yukarıdaki gibi dört akış sayısı için dört sınıf tanımlanmıştır:

```
# Classes:
# 1: ADC=5 ms, ALC=1 %, no ARC, no RDC, no RLC
# 2: no ADC, no ALC, RDC = 4, RLC = 2
# 3: no ADC, no ALC, RDC = 4, RLC = 2
# 4: no ADC, no ALC, RDC = 4, RLC = 2
```

Akış sayısı kendi sınıf sayısı ile beraber tanımlanır ve hem kullanıcı sayısının hem de hizmet çeşidinin

değişebileceğini gösterir. Örneğin, sınıf sayıları akış sayılarına eşit olduğu bir durum aşağıdaki gibidir:

```
# Number of monitored flows (equal to # of classes)
set N_CL 4
puts "Number of classes: $N_CL"

set N_USERS 4
puts "Number of flows: $N_USERS"
```

Yukarıdaki parametreler bu çalışmadaki benzetim için aşağıdaki gibi tanımlanabilir. Burada, sınıf sayıları akış sayılarına eşittir.

```
# Number of monitored flows (equal to # of classes)
set N_CL 3
puts "Number of classes: $N_CL"

set N_USERS 3
puts "Number of flows: $N_USERS"
```

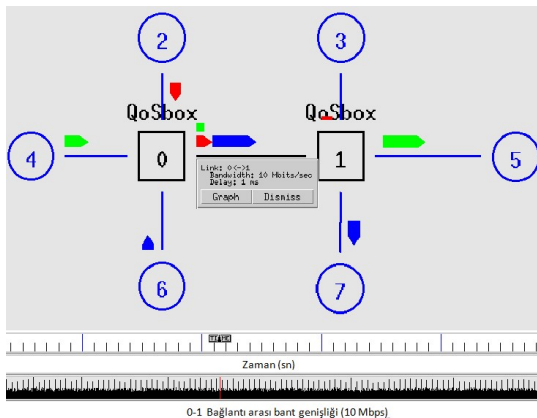
Sınıf sayıları akış sayılarıyla eş zamanlı olarak değişir. Yukarıdaki gibi üç akış sayısı için aşağıda üç sınıf tanımlanmıştır:

```
# Classes:
# 1: ADC=5 ms, ALC=1 %, no ARC, no RDC, no RLC
# 2: no ADC, no ALC, RDC = 4, RLC = 2
# 3: no ADC, no ALC, RDC = 4, RLC = 2
```

‘jobs-lossdel.tcl’ dosyasındaki örnek topoloji aynı zamanda yukarıdaki parametrelere göre düzenlenebilir.

```
# nam stuff
$ns duplex-link-op $source(1) $score_node(1) orient 300deg
$ns duplex-link-op $source(1) $score_node(1) color "blue"
$ns duplex-link-op $source(2) $score_node(1) orient 300deg
$ns duplex-link-op $source(2) $score_node(1) color "blue"
$ns duplex-link-op $source(3) $score_node(1) orient 300deg
$ns duplex-link-op $source(3) $score_node(1) color "blue"
$ns duplex-link-op $score_node(1) $score_node(2) orient right
$ns duplex-link-op $score_node(1) $score_node(2) color black
$ns duplex-link-op $score_node(1) $score_node(2) queuepos 0.5
$ns duplex-link-op $score_node(2) $sink(1) orient 60deg
$ns duplex-link-op $score_node(2) $sink(1) color "blue"
$ns duplex-link-op $score_node(2) $sink(2) orient 30deg
$ns duplex-link-op $score_node(2) $sink(2) color "blue"
$ns duplex-link-op $score_node(2) $sink(3) orient 30deg
$ns duplex-link-op $score_node(2) $sink(3) color "blue"
$ns color 1 red
$ns color 2 green
$ns color 3 blue
puts "ngo!\n"
$ns run
```

QoSbox benzetimi için elde edilen yeni sonuçlar Şekil 2’deki gibidir ve bu şekilden topolojinin değiştiği görülmektedir. Şekil 2’de üç farklı renk üç farklı akışı göstermektedir. Mesela, akışlar düğüm 2’den 3’e, 4’ten 5’e, 6’dan 7’ye akmaktadır.



Şekil 2. Üç akışlı QoSbox benzetim sonucu

‘jobs-lossdel.tcl’ dosyasından, bant genişliği, gecikme, iş/zaman oranı ve ağ geçidindeki ara bellek boyutları gibi QoS faktörlerine erişilebilir.

```
puts "ntoPOLOGY SETUP"
# Number of hops (n)
set hops 2
puts "Number of hops: $hops"
# Link latency (ms)
set DELAY 1.0
puts "Link latency: $DELAY (ms)"
# Link bandwidth (Mbps)
set BW 10000.0
puts "Link capacity: $BW (Mbps)"
# Buffer size in gateways (in packets)
set GW_BUFF 50
puts "Gateway Buffer Size: $GW_BUFF (packets)"
# Packet size (in bytes); assume common for all sources
set PKTSZ 500
set MAXLEN 50
puts "Packet Size: $PKTSZ (bytes)"
```

IV. SONUÇ

[8] çalışmasında QoSbox benzetim sonuçları, 0-1 bağlantısında hiçbir Sabit Bit Oranı (Constant Bit Rate) CBR bırakması olmadan iki akış ve iki hizmet sınıfı kullanılarak elde edilmiştir. Fakat, bizim çalışmamızda [8] çalışmasından farklı olarak, 0-1 bağlantısında daha fazla hizmet sınıfları ve akışlar kullanılarak (üç tane) hiçbir CBR bırakması olmadan optimum sonuçlar elde edilmiştir. Bu çalışmanın sonucunda elde edilenler, bu konu üzerinde şimdiye kadar yapılan çalışmalardan farklıdır. JoBS kullanıcı sayısından bağımsızdır. Burada önemli olan durum ise zorlama girişimi ile garanti sayılarından oluşan kısıtlama sistemidir. Genellikle, daha fazla hizmet sınıfları kısıtlamaların doyumluğa erişmesini zorlaştırmaktadır ve bu durum bazı kısıtlamaların dinlenme konumuna geçmesiyle sonuçlanır. Tipik olarak, mutlak kısıtlamalara göreli kısıtlamalara göre daha fazla öncelik tanınmaktadır. JoBS, göreli garanti durumuna ihtiyaç duyulan yerlerde dar boğaz durumu olduğu sürece tasarrufludur. Sanal bir ağ, artan akış trafiğinin etkilerini öngörmeden gelişirse bunun sonucu olarak QoS problemlinden bütün ağ kullanıcıları etkilenecektir.

KAYNAKÇA

- [1] Virtualization & Cloud Computing Group, “Virtualization and Cloud Computing [M]”, Beijing Electronic Industry Publish. 2009: 127-135.
- [2] Armbrust M, Fox A, Griffith R, “Above the Clouds: A Berkeley View of Cloud Computing [J]”, 2009. 2(23);, pp.45-48. Rudnick, A. I., Polifroni, Thayer, E. H., and Brennan, R. A. "Interactive problem solving with speech", J. Acoust. Soc. Amer., Vol. 84, 1988, p S213(A).
- [3] explainingcomputers.com/cloud/index.html (Visited date: 25 January 2016)
- [4] Baris Aksanli, Jagannathan Venkatesh, and Tajana Šimunic’ Rosing, “Using Datacenter Simulation to Evaluate Green”, University of California, San Diego, 2007, pp.25.
- [5] D. Kliazovich, P. Bouvry, and S. U. Khan, “A Packet-level Simulator of Energy-aware Cloud Computing Data Centers Energy Integration”, University of Luxembourg.
- [6] Nicolas Christin J’org Liebeherr, Department of Computer Science, “The QoSbox: A PC-Router for Quantitative Service Differentiation in IP Networks”, University of Virginia, 2009, pp.7-23.
- [7] VARIA, J, 2009. “Cloud Architectures Amazon Web Services” White Paper, Amazon.
- [8] Jiang Zhuhua, “GreenCloud for Simulating QoS-Based NaaS in Cloud Computing”, IEEE, “Computational Intelligence and Security (CIS), 2013 9th International Conference”, pp 766-770, Leshan, 2013.
- [9] CSIM Development Toolkit for Simulation and Modeling, 2010. Available at www.buyya.com/papers/CloudSim2010.pdf (Visited date: December 2015).
- [10] <http://greencloud.gforge.uni.lu> (ziyaret tarihi: Aralık 2015).