

NS-2 ve NS-3 Ağ Simülatörlerinin Ölçeklenebilirlik Analizi ve Karşılaştırma

Ünal ÇAVUŞOĞLU, Ahmet ZENGİN

Bilgisayar Mühendisliği Bölümü, Sakarya Üniversitesi Teknoloji Fakültesi, Sakarya, Türkiye
unalc@sakarya.edu.tr, azengin@sakarya.edu.tr
 (Geliş/Received: 22.07.2012; Kabul/Accepted: 20.09.2012)

Özet- Bu makalede, NS-2 ve NS-3 ağ simülatörlerinin ölçeklenebilirlik analizi gerçekleştirilmiştir. NS-2 ve NS-3 de oluşturulan ağlar; düğüm sayısı, ağ çıkışı, simülasyon zamanı, işlemci ve bellek miktarları gibi faktörler kullanılarak karşılaştırılmıştır. Ayrıca NS-2 ve NS-3 ağ simülatörleri hakkında kısaca bilgi verilmiştir. Kullanılan farklı donanım özelliklerine sahip bilgisayarlar ve farklı yapıdaki ağlar üzerinde yapılan testler sonucu ağ simülatör programlarının ölçeklenebilirlik kabiliyetleri değerlendirilmiştir.

Anahtar kelimeler- Ağ simülatörleri, NS-2, NS-3, ölçeklenebilirlik

Scalability Analysis and Comparison of NS-2 and NS-3 Network Simulators

Abstract- In this article, the NS-2 and NS-3 scalability of the network simulators using a computer analysis of two different and two different networks was designed. Networks created in NS-2 and NS-3, when compared with by using such factors as the number of nodes, the simulation time, the amount of processor and memory. In addition, the network simulator NS-2 and NS-3 is an overview of programs. Networks and different components used in computers with different hardware features of the result of tests carried out on the scalability of the network capabilities of the simulator programs are evaluated.

Keywords- Network simulator, NS-2, NS-3, scalability

1. GİRİŞ

Bu çalışmada NS-2 ve NS-3 ağ simülatör programları üzerinde iki farklı topolojide senaryolar geliştirilerek, simülatörlerin ölçeklenebilirlik analizleri gerçekleştirilmiştir. Testler sonucu elde edilen analiz sonuçlarına göre ağ simülatörlerinin ölçeklenebilirlik ve performansları hakkında değerlendirmelerde bulunulmuştur. Farklı donanımsal özelliklere sahip bilgisayarlar üzerinde ağ topoloji yapılarında çok sık kullanıma sahip ızgara ve yıldız topolojileri üzerinde testler yapılmıştır. 16x16'lık ızgara simülasyonunun NS-2 ve NS-3 kodları Bilgisayar 1 ve Bilgisayar 2 de ayrı ayrı çalıştırılarak simülasyonun tamamlanma süreleri ve CPU, bellek kullanım durumları incelenmiştir. Izgara simülasyonunda NS-2 simülatörü her iki bilgisayarda da daha hızlı çalışırken, NS-3 simülatöründe simülasyonun tamamlanması daha uzun sürmüştür. Tasarlanan bir diğer simülasyon olan yıldız topolojide 1000 düğüm için yine NS-2 simülatöründe simülasyonun tamamlanma süresinin NS-3 simülatörüne göre daha kısa olduğu gözlemlenmiştir. Bunun yanı sıra yıldız topoloji simülasyonunda Bilgisayar2 1500 düğüme kadar çıkabilirken Bilgisayar1 1000 düğümün üstüne çıkamamıştır. Kaynak kullanımı açısından değerlendirildiğinde ise NS-3 ağ simülatörünün daha verimli bir şekilde kullanım gerçekleştirdiği tespit edilmiştir. Yapılan deneyler sonucu elde edilen bu veriler ışığında ağ simülatörlerinin karşılaştırılması yapılmıştır.

Makalede 1. Bölümde girişi takiben, 2. Bölümde ölçeklenebilirlik, modelleme ve simülasyon, NS-2 ve NS-3 ağ simülatörleri, animasyon programları hakkında genel bilgilendirme yapılmış, 3. Bölümde performans deneyleri gerçekleştirilmiş, 4. bölümde iki simülatör karşılaştırılmış olup, 5. bölümde ise sonuç ve değerlendirme yapılmış ve gelecekteki çalışmalar hakkında bilgi verilmiştir.

2. GENEL BİLGİLER

2.1. Ölçeklenebilirlik

Ölçeklenebilirlik, donanımın veya yazılımın bilgi işlem gereksinimlerini karşılamak üzere kolayca genişletilebilme yeteneğidir [1]. Her ağ topolojisi veya bu topolojiler üzerinde kullanılan cihazlar, analiz programları, daha iyi performans ve ölçeklenebilirlik sağlayacak şekilde tasarlanmalıdır. Bununla birlikte, donanım yapılandırılmaları ve gerektiği gibi iyi tasarlanmamış veya gerektiği gibi sınanmamış uygulamalardaki performans sorunları, ölçeklenebilirliği olumsuz etkileyebilir. Yöneticiler, donanımı, dağıtılmış uygulamaları ve özel bileşenleri de kapsayan altyapılarını dikkatle planlayarak ve değerlendirerek, ağ yapılarının ölçeklenebilirliğini önemli ölçüde arttırabilirler. Günümüz teknolojisinin çok hızlı gelişmesi sonucunda kurulacak olan sistemlerin veya tasarlanacak programların

ölçeklenebilir olması çok önemli bir kriter haline gelmiştir.

2.2. Modelleme ve Simülasyon

Simülasyon dinamik bir sistemin özelliklerini ve davranışlarını bilgisayar aracılığıyla değerlendiren bir tekniktir. Kullanıcısına değişik tasarım ve işletim stratejilerinin genel sistem performansı üzerindeki etkisini gösterir. Sonuçta elde edilenler, istenen model karakteristiklerine ait birer tahmindir. Diğer bir tanımla simülasyon, incelenen bir gerçek dünya sisteminin belli bir zaman diliminde istenilen gerçek karakteristiklerini tahmin etmek amacıyla sistemin matematiksel, mantıksal bir modelinin geliştirilmesi ve bu sistem üzerinde deneyler yapılması sürecidir [2]. Analitik yaklaşımların aksine simülasyon modelleri, karmaşık problemlerin modellenmesi ve çözümünde daha başarılı olurlar. Değişkenler arasındaki etkileşimleri simülasyon modellerinde gözlemek daha kolaydır. Ancak yoğun bilgisayar kullanımını gerektirir. Gerçek sistemden toplanan bilgiler, bilgisayarda geliştirilen modellere uygulanarak sayısal birtakım sonuçlara ulaşmak hedeflenir. Bunların değerlendirilmesi ve yorumlanması yapılarak sistem performans ölçütlerine ait birtakım tahminlerde bulunulur. Simülasyon modelleri aracılığı ile en kötü durum senaryoları da incelenebilir.

2.3. NS-2 Ağ Simülatörü

NS (Network Simulator), ağ simülasyonu oluşturmak ve gerçekleştirmek için ilk olarak 1989 yılında geliştirilmeye başlanmıştır. Akademik araştırmalar için büyük öneme sahip açık kaynak kodlu bir ayrık olay ağ simülatörü olarak tanımlanmaktadır [3]. NS kullanımı, 1995 yılında DARPA'nın sponsorluğunda bir ivme kazanmıştır ve günümüzde de simülatörün geliştirilmesi geniş bir kullanıcı grubu tarafından sürdürülmektedir. NS ile kablolu (wired) ya da kablosuz (wireless) ağlarda istenilen miktarda düğümler ve bu düğümler arası linkler tanımlanabilir, yönlendirme algoritmaları ile çoklu gönderim (multicast) protokolleri kullanılabilir ve ad-hoc network, WiFi, WiMax, vb. gibi bir takım popüler kablosuz ağ uygulamalarının modellemeleri ve simülasyonu gerçekleştirilebilir. NS simülatörü, ağ araştırma ve eğitimini destekleyerek protokol tasarımı, trafik araştırması sağlamakta, ücretsiz açık kaynak kodu ile karşılaştırmalı ve model paylaşımı güvenilir bir deney ortamı sunmaktadır. NS programlama, yapılan çalışmalarda belli bir algoritmaya uygun olarak oluşturulmaktadır. Öncelikle olay programlayıcısı kurulur, izleme açılır ve kapanır, ağ modeli oluşturulur, ilgili yönlendirme protokolü kurulur, sisteme hatalar girilir, iletim bağlantısı ile trafik oluşturulur ve uygulama gerçekleştirilerek veriler iletilir. NS simülatörünün ikinci versiyonu NS-2 olarak adlandırılmaktadır [4]. NS-2, C++ tabanlı bir simülatördür ve TCL dilinin nesneye yönelik bir versiyonu olan OTCL ile bir simülasyon ara yüzü vasıtasıyla ağ simülasyonu gerçekleştirmek mümkündür. C++, her paket işlemi için hızlı, detaylı bir kontrol sağlarken, OTCL ise mevcut C++ nesnelerini kullanarak simülasyon senaryo ayarları, periyodik veya tetiklemeli

olay programlama ve kolay yazılım imkanlarıyla öne çıkmaktadır. NS-2, Linux platformunda çalışmaktadır ve Cygwin programı vasıtasıyla Microsoft Windows ortamında da simülasyonlar gerçekleştirilebilmektedir. NS-2 simülatörünün son sürümü olan 2.35, Kasım 2011'de kullanılmaya başlanmıştır [5]. NS-2 ayrık olaylı bir ağ simülatörüdür. Basit ağ protokollerinin modellenmesine odaklanır. NS-2 ağ simülatörü nesne yönelimli bir yazılım içerir ve modüler bir yapıya sahiptir. NS-2'nin yardımcıları NAM (Network Animation Ortamı) ve TCL Programlarıdır. TCL, bir derleyici programıdır ve NS-2'de veya TCL de yazılan kodlar bu derleyici tarafından derlenir. NS-2 programının asıl görevi Internet protokollerinin dizaynidir. Özellikle TCP protokolleri için kullanılmaktadır. Bununla birlikte hiyerarşik yapıların oluşturulması için yardımcı ve hareket ettirici bir programdır. NS-2 İnternet katmanında farklı yeni protokoller oluşturmak (yönlendirme, iletim, uygulama), Internet protokollerinin (ATM, kablolu LANs, kablosuz LANs) etkileşimlerini göstermek için ve internetteki yeni mimari yapıların performanslarını belirlemek ve etkilerini göstermek için kullanılır.

2.4. NS-3 Ağ Simülatörü

NS-3 internet sistemleri için ayrık olay (discrete-event) mantığı ile çalışan bir ağ simülatörüdür [6]. NS-3 simülatörü, NS simülatörünün 3.versiyonu olarak 1 Temmuz 2006 tarihinden itibaren geliştirilmeye başlanmıştır ve projenin tamamlanması için 4 yıllık bir süre öngörülmüştür. Günümüzde tasarımı devam etmekte olan NS-3 simülatörü, University of Washington, Georgia Institute of Technology, ICSI ve INRIA gibi kurumlar tarafından geliştirilmeye devam etmektedir. NS-3 simülatörü araştırma geliştirme ve akademik faaliyetlerde kullanılmak üzere özellikle internet tabanlı sistemler için geliştirilen bir ayrık olay ağ simülatörü olarak tanımlanmaktadır. GNU GPLv2 (general public license) lisanslı yani bedelsiz ve açık kaynak kodlu bir yapıya sahiptir. NS-3, C++ ve Python dilleri kullanılarak yazılmaktadır. Yine NS-2 simülatöründe olduğu gibi Linux işletim sistemi üzerinde çalıştırılmaktadır ve Cygwin programı vasıtasıyla Windows tabanlı sistemlerde de kullanılabilir. NS-3'te kod yapıları Doxygen isimli bir yazılım dokümantasyon programı vasıtasıyla uygulanmaktadır. NS-2'ye aşına olanlar için NS-3'e geçişte en gözle görülür değişiklik script dilinin seçimidir. NS-3 simülatörü NS-2'nin genişletilmiş hali değildir. Her iki simülatörde C++ dilinde yazılmış fakat NS-3, NS-2 uygulamalarını desteklemeyen tamamen yeni bir simülatördür. NS-2'den bazı modeller zaten NS-2'den NS-3'e bağlanmıştır. NS-3 geliştirilmeye devam ederken NS-2' de varlığını sürdürecektir ve geçiş ve entegre mekanizmaları çalışacaktır. NS-3 açık kodlu ve araştırmacılara yazılımlarına katkıda bulunma ve paylaşma için açık bir çevre sağlamak için çalışmaktadır. NS-3 projesinde Waf sistem mimarisi kullanılmıştır. Waf, Python temelli yeni bir sistemdir. NS-3 IP adreslemeyi, daha çok internet protokolleri ve tasarımları ve daha detaylı 802.11 modeli vb. kullanarak düğümler üzerinde çoklu ara yüzleri hatasız işlemek gibi özellikleri destekler.

NS-3 aşağıdaki özellikleri desteklemektedir:

- Sanal ağların yapısı ve olay zamanlayıcı, topoloji üreticileri, zamanlayıcılar, rastgele değişkenler gibi nesnelere için destek ve internet tabanlı ve diğer paket ağ sistemleri odaklı diğer nesnelere için ayrık olay ağ simülasyonu desteği sağlama,
- Ağ simülasyonu desteği; Simülatör işlemleri için gerçek ağ paketlerini üretme ve yayma,
- Dağıtık simülasyon desteği; çoklu işlemciler ya da dağıtık makineler kullanılmadan simülasyon yeteneği,
- Ağ simülasyonları için animasyon desteği ve
- Simülasyon çıkışında bulunan, hesaplama, kayıt tutma, izleme istatistikleri için destek sağlar.

Tablo 1. NS-3 Büyüme eğilimi

Sürüm	C++ kod satır sayısı (wc src/klasörü)	Programcı sayısı	İndirme sayısı
NS-3.4	110000	27	1700
NS-3.8	250000	55	86014

Tablo 1’de NS-3’ün gelişimini göstermek için kod satır sayısı, programın gelişimi için çalışan yazar sayısı ve programın resmi sitesinden sürümlerinin indirme sayıları görülmektedir. NS-3’e olan ilginin gün geçtikçe arttığını bu rakamlar açık şekilde ortaya koymaktadır.

2.5. NAM (Network Animator)

NAM network simülasyon izlerini ve gerçek dünya veri paketlerini izlemek için bir animasyon tabanlı programdır [7]. NAM; topoloji düzeni, paket seviye animasyon ve çeşitli data keşif araçlarını destekler. NAM’ın geliştirmesine LBL(Lawrence Berkeley National Laboratory)’de başlanmıştır. Son zamanlarda gelişme göstermektedir. NAM geliştirilme çabası devam etmekte olan bir proje olan VINT ile sağlanmaktadır. NAM’ı kullanmak için ilk adım bir iz (trace) dosyası oluşturmaktır. İz dosyası gibi topoloji bilgilerini içermelidir. Simülasyon boyunca kullanıcı, topoloji konfigürasyonu, düzen bilgisi ve paket izlerini izleme olaylarını (tracing events) kullanarak üretebilir. İzleme dosyası üretildiği zaman, her şey anımsamak için hazır demektir. NAM iz dosyasını okur, topolojiyi oluşturur, bir ekran çıkar ve simülasyonu gösterir. NAM kullanıcı arayüzü animasyonun birden fazla görünümünü sağlar. NAM düğüm(node), bağlantı (link), kuyruk (queue), paket(packet), temsilci (agent) ve monitor yapı bloklarını kullanarak animasyon yapar.

3. PERFORMANS KARŞILAŞTIRMA DENEYLERİ

3.1. Simülasyon Parametreleri

Tablo 2’de özellikleri görülen bilgisayarlar üzerinde NS-2 ve NS-3 ağ simülatör programları kurularak ızgara ve yıldız topoloji yapısında hazırlanan ağ modelleri analiz edilmiştir. Simülatörlerin ölçeklenebilirlik ve performans

analizini farklı donanımlar üzerinde gerçekleştirilerek daha sağlıklı değerler üretilmesi amaçlanmıştır.

Tablo 2. Simülasyonda kullanılan bilgisayarların özellikleri

Bilgisayar 1	Bilgisayar 2
1GB RAM Intel® Core™2 Duo İşlemci 1,5 GHz işlemci hızı VMware 512 MB ram, 10 GB harddisk	4 GB RAM Intel® Core™ i7 İşlemci 1,73 GHz işlemci hızı Virtual Box 1GB ram, 8 GB harddisk

Diğer simülasyon parametreleri uygulanan ağ topolojilerinin ilgili kısımlarında tablolar halinde belirtilmiştir.

3.2. Izgara Ağ Simülasyonu

Bu simülasyonda 16x16’lık ızgara modelinin hem NS-2 hem de NS-3 simülatörleri kullanılarak ve iki farklı bilgisayarda çalıştırılarak ölçeklenebilirlik analizi incelenmiştir. Bilgisayarların işlemci özellikleri Tablo 2’de belirtilmiştir. Izgara ağ modelinin özellikleri Tablo 3’de görülmektedir.

Tablo 3. Izgara ağ simülasyonu özellikleri

Trafik	TCP ve FTP trafikleri rastgele seçilen düğümler arasında üretilmiştir.
Bant Genişliği	5 mbps
Gecikme	2 ms.
Simülasyon Süresi	20 sn
Düğüm Sayısı	256

Tablo 3’de gerçekleştirilmiş olan ızgara ağ simülasyonuna ait özellikler görülmektedir. Izgara ağ topolojisi üzerindeki trafik Tcp ve Ftp protokolleri ile rasgele seçilen düğümler üzerinde, 5 Mbps bant genişliğinde ve 256 düğüm üzerinde gerçekleştirilmiştir.

Tablo 4. Bilgisayar1 simülasyon sonuçları

Simülatör	Simülasyon Süresi	Kaynak Kullanımı		Düğüm Sayısı
		CPU	BELLEK	
NS-2	2.5 Dk.	% 98	%36	256
NS-3	4 Dk.	%97	%27	256

Tablo 5. Bilgisayar2 simülasyon sonuçları

Simülatör	Simülasyon Süresi	Kaynak Kullanımı		Düğüm Sayısı
		CPU	BELLEK	
NS-2	35 sn.	% 96	%27	256
NS-3	1 Dk.	%99	%18	256

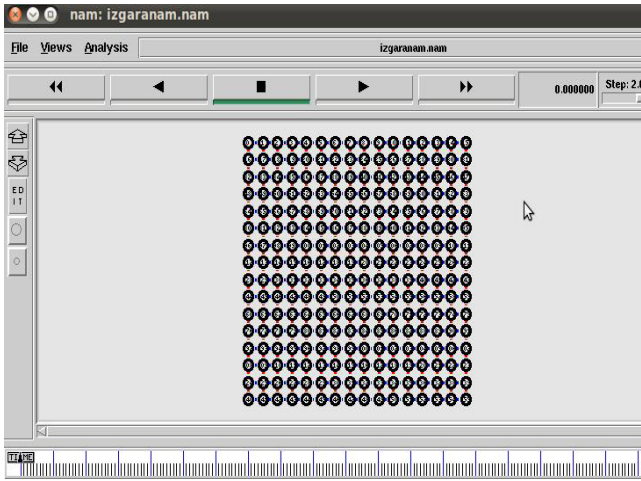
Tablo 4 ve Tablo 5’de farklı donanımsal özelliklere sahip bilgisayar1 ve bilgisayar2 makinelerinde 256 düğüme sahip bir ızgara ağ simülasyonunun çalıştırılması sonucu

elde edilen simülasyon süreleri, işlemci ve bellek kullanım değerleri görülmektedir.

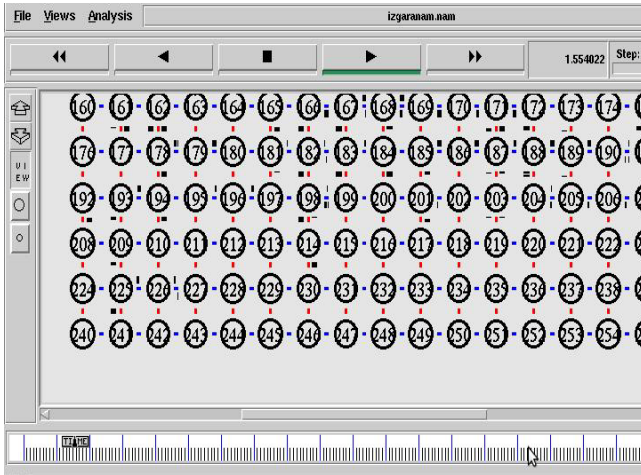
Tablo 4 ve 5'deki kullanım değerleri incelendiğinde, donanımsal olarak daha güçlü olan bilgisayar2 üzerinde işlemler daha kısa sürede tamamlanmıştır. Izgara ağ modelinin NS-2 ve NS-3 ağ simülatörleri üzerinde çalıştırılması sonucu her iki donanım üzerinde; NS-2 üzerinde simülasyon daha kısa sürede tamamlanmış, her iki simülatör işlemcinin hemen tamamını kullanmış ve NS-3'ün belleği daha etkili kullandığı tespit edilmiştir.

3.2.1. NS-2 Simülasyon Ekran Çıktıları

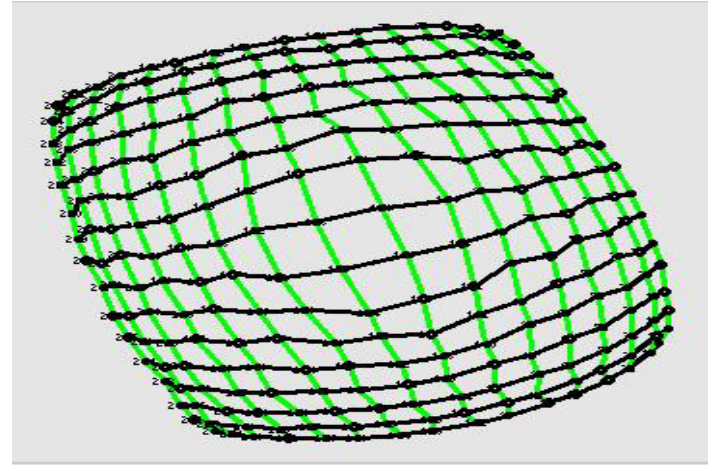
Şekil 1, 2 ve 3'te gerçekleştirilen simülasyon deneylerinin NS-2 ekran çıktıları verilmektedir.



Şekil 1. Bilgisayar 1'de 16x16 lık NS-2 ızgara ağ görüntüsü



Şekil 2. Bilgisayar 1'de 16x16'lık NS-2 ızgara ağ simülasyonunun 1.5 sn'deki trafik durumu



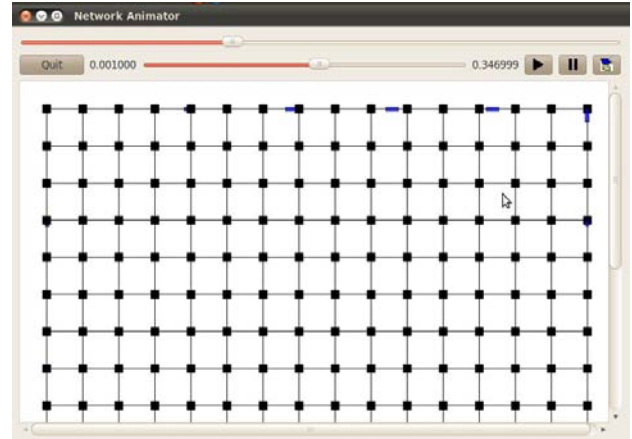
Şekil 3. Bilgisayar 2'de 16x16'lık NS-2 ızgara ağ görüntüsü

3.2.2. NS-3 NetAnim Ekran Çıktıları

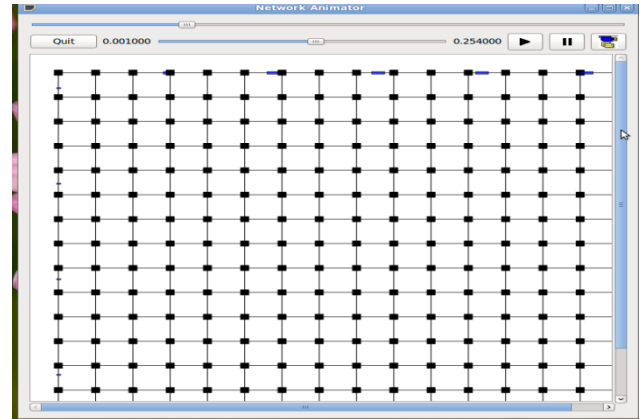
NS-3'te simülasyonun çalışmasını görebilmek için .tr uzantılı izleme dosyası NetAnim klasörüne kopyalanır ve aşağıdaki komutlar konsolda yazılır.

- cd NetAnim
- ./NetAnim izgara3.tr

Şekil 4 ve 5'de gerçekleştirilen simülasyon deneylerinin NS-3 ekran çıktıları verilmektedir.



Şekil 4. Bilgisayar 1'de 16x16 lık NS-3ızgara ağ simülasyonunun 3.46 s'deki trafik durumu



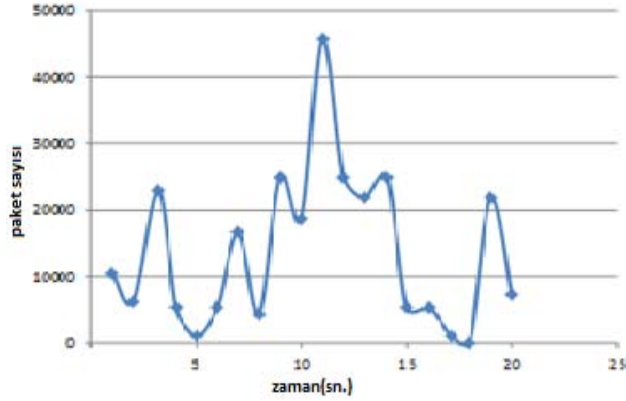
Şekil 5. Bilgisayar 2'de 16x16 lık NS-3 ızgara ağ simülasyonunun 2.5s'deki trafik durumu

3.2.3. Izgara Ağ Modeli İçin Ağ Çıkışı

Izgara örneğinde 172. düğüm üzerindeki 1sn aralıklarla throughput elde etmek için NS-2'de konsola aşağıdaki komut yazılır.

```
perl throughput.pl izgaratr.csv 172 1>172izgara-ns2
```

Daha sonra Microsoft Excel'de üretilen dosya açılarak gerekli düzenlemelerden sonra şekil 6' daki grafik elde edilir. Grafiği incelediğimizde 20 sn'lik simülasyon süresince 172 nolu düğüm üzerinde veri paketlerinin değişimini gözlemleyebiliyoruz.



Şekil 6. Izgara ağındaki 172. düğüm üzerindeki ağ çıkışı grafiği

3.2.4. Izgara Ağ Modeli Sonuçlarının Değerlendirilmesi

Simülörler için ızgara ağ modeli örneklerini inceleyecek olursak, 2. bilgisayarın özellikleri 1. bilgisayara göre daha iyi olduğu için, 2.bilgisayarda NS-2 ve NS-3 uygulaması daha kısa zamanda gerçekleşmiştir. Ancak aynı koşul altında, tek bir bilgisayarda değerlendirecek olursak, her iki bilgisayarda da simülasyon NS-3'te daha uzun zamanda gerçekleşmiştir. NS-2 ve NS-3'de simülasyon çalışırken birbirine yakın yüzdelerde işlemci kullanımı olmuştur. Her iki bilgisayarda da işlemcinin %99'u simülasyon çalışırken kullanılmıştır. Bellek kullanımı NS-3'te daha düşük olduğu gözlenmiştir.

Bu bilgilerin tümü değerlendirildiğinde iki simülör arasında performans farklılığının olduğu açıkça görülmektedir. Bu performansın bir parçası olan ölçeklenebilirlik düşünüldüğünde ise NS-3 simülörünün var olan şartları daha iyi değerlendirerek daha iyi sonuçlar elde ettiği söylenebilir. Koşulların farklılığına bağlı olarak uygulanan örneklerde farklı performanslar elde edilmiştir. Ancak aynı koşullar düşünüldüğünde NS-3 simülörünün NS-2 simülörüne göre daha ölçeklenebilir olduğu söylenebilir.

3.3. Yıldız Ağ Topolojisi

Yıldız topolojisi, en yaygın ve gerçekçi bilgisayar ağı topolojilerindendir. Bütün bilgisayarların tek bir merkeze bağlanarak oluşturduğu topolojidir. Genelde merkezde dağıtıcı veya anahtar denilen cihazlar kullanılır. Ortak yol topolojisine göre daha performanslı

olan topolojidir. Merkezdeki dağıtıcı veya anahtarda oluşacak problem bütün ağı etkiler. Böylece merkezdeki problem diğerlerine de akseder. Avantajları, bir istasyon arızalanması diğer istasyonları etkilemez, ağa yeni bir istasyon eklemek kolaydır, ağ yönetimi çok kolaydır, kurulan ağ elemanlarına göre yüksek hızlar elde edilebilir. Bu simülasyonda yıldız topoloji modeli hem NS-2 hem de NS-3 simülörleri kullanılarak ve iki farklı bilgisayarda çalıştırılarak ölçeklenebilirlik analizi yapılmıştır.

Tablo 6. Yıldız ağ topoloji özellikleri

Trafik	TCP ve FTP trafikleri rastgele seçilen düğümler arasında üretilmiştir.
Bant Genişliği	5 mbps
Gecikme	2 ms.
Simülasyon Süre	10 sn
Düğüm Sayısı	1000

Tablo 6'da gerçekleştirilmiş olan yıldız ağ simülasyonuna ait özellikler görülmektedir. Yıldız ağ topolojisi üzerindeki trafik TCP VE FTP protokolleri ile rasgele seçilen düğümler üzerinde, 5 mbps bant genişliğinde ve 1000 düğüm üzerinde gerçekleştirilmiştir.

Tablo 7. Bilgisayar1 simülasyon sonuçları

Simülör	Simülasyon Süresi	Kaynak Kullanımı		Düğüm Sayısı
		CPU	BELLEK	
NS-2	2 Dk.	% 99	%48	1000
NS-3	10 Dk.	%100	%34	1000

Tablo 8. Bilgisayar2 simülasyon sonuçları

Simülör	Simülasyon Süresi	Kaynak Kullanımı		Düğüm Sayısı
		CPU	BELLEK	
NS-2	37 sn.	% 99	%35	1000
NS-3	4 Dk.	%100	%24	1000

Tablo 7 ve Tablo 8'de farklı donanımsal özelliklere sahip bilgisayar1 ve bilgisayar2 makinelerinde 1000 düğüme sahip bir yıldız ağ simülasyonun çalıştırılması sonucu elde edilen simülasyon süreleri, işlemci ve bellek kullanım değerleri görülmektedir.

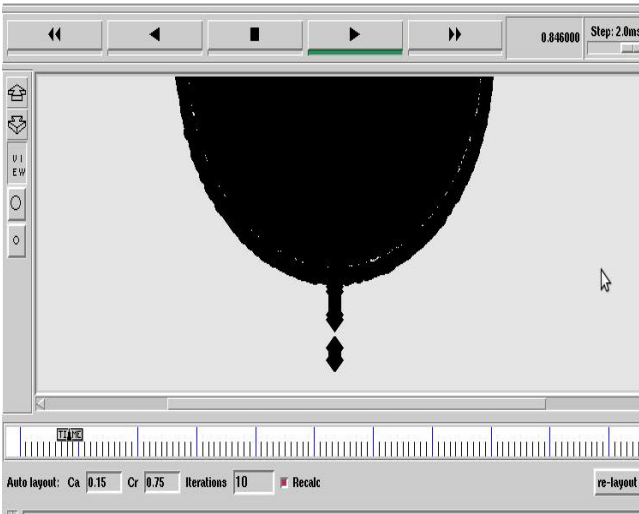
Tablo 7 ve 8'deki değerler incelendiğinde, donanımsal olarak daha güçlü olan bilgisayar2 üzerinde işlemler daha kısa sürede tamamlanmıştır. Izgara ağ modelinin NS-2 ve NS-3 ağ simülörleri üzerinde çalıştırılması sonucu her iki donanım üzerinde; NS-2 üzerinde simülasyon daha kısa sürede tamamlanmış, her iki simülör işlemcinin hemen hemen tamamını kullanmış ve NS-3'ün belleği daha etkili kullandığı tespit edilmiştir.

3.3.1. NS-2 Simülasyon Ekran Çıktıları

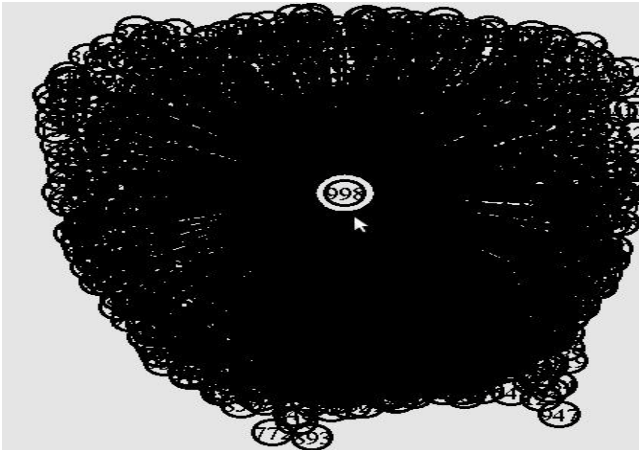
Şekil 7,8 ve 9'da gerçekleştirilen simülasyon deneylerinin NS-2 ekran çıktıları verilmektedir.



Şekil 7. Bilgisayar 1'de NS-2 yıldız topoloji



Şekil 8. Bilgisayar 1'de NS-2 yıldız topoloji simülasyonunda paket kayıpları



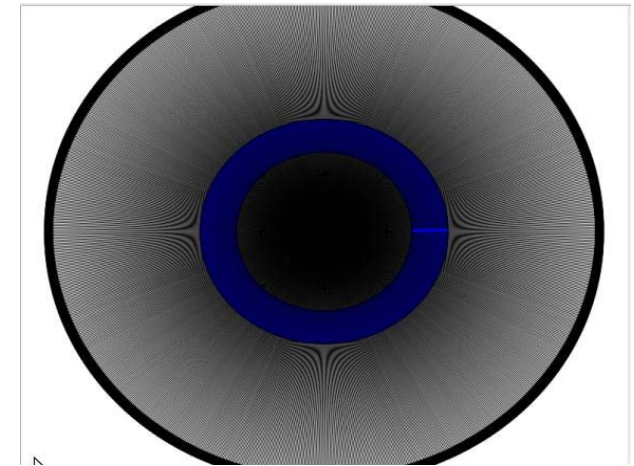
Şekil 9. Bilgisayar 2'de NS-2 yıldız topoloji

3.3.2. NS-3 Simülasyon Çıktıları

Şekil 10 ve 11'de gerçekleştirilen simülasyon deneylerinin NS-3 ekran çıktıları verilmektedir.



Şekil 10. Bilgisayar 1'de NS-3 Yıldız topoloji görünümü



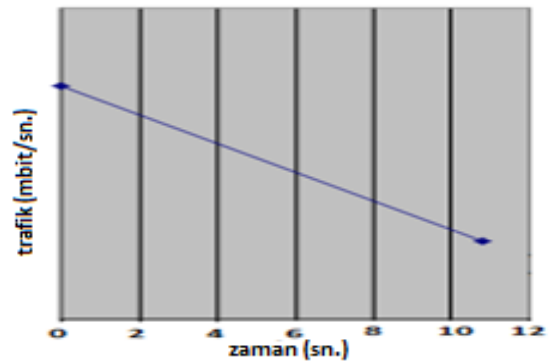
Şekil 11. Bilgisayar 2'de NS-3 Yıldız topoloji görünümü (trafik sırasında)

3.3.3. Yıldız Ağ Modeli İçin Ağ Çıkışı

Yıldız topoloji simülasyonunda 1. düğüm üzerindeki 0.1sn aralıklarla throughput elde etmek için NS-2'de konsola aşağıdaki komut yazılır.

```
perl throughput.pl yıldızizleme.tr 1 0.1> yıldızth
```

Daha sonra Microsoft Excel'de üretilen dosya açılarak gerekli düzenlemelerden sonra grafik elde edilir.



Şekil 12. Yıldız topolojide 1. düğüm üzerindeki 0.1 sn aralıkla izleme grafiği

3.3.4. Yıldız Ağ Modeli Sonuçlarının Değerlendirilmesi

Simülasyonlar için yıldız topoloji örneklerini inceleyecek olursak, ızgara örneğinde olduğu gibi, 2. bilgisayarın özellikleri daha iyi olduğu için, NS-2 ve NS-3 uygulaması 1. bilgisayara göre daha kısa zamanda gerçekleşmiştir. Ancak aynı koşul altında, tek bir bilgisayarda değerlendirecek olursak, her iki bilgisayarda da simülasyon NS-3'te daha uzun zamanda gerçekleşmiştir. NS-2 ve NS-3'de simülasyon çalışırken birbirine yakın yüzdelerde işlemci ve bellek kullanımı olmuştur. Her iki bilgisayarda da işlemcinin %100'ü simülasyon çalışırken kullanılmıştır. Bunun yanı sıra bilgisayar2, simülasyonda NS-2 ve NS-3'de 1500 düğüme kadar çıkabilirken bilgisayar1 1000 düğümün üstüne çıkamamıştır.

4. SİMULATÖRLERİN KARŞILAŞTIRILMASI VE DEĞERLENDİRME

NS-2 OTCL'de yazılmıştır ve simülasyonlar Network Animator (NAM) programı kullanılarak görselleştirilebilmektedir. NS-2'de bir simülasyonu sadece C++ programından çalıştırmak mümkün değildir. NS-3'de simülasyon opsiyonel Python bağlayıcıları ile tamamen C++'da yazılmıştır. Simülasyon yazılımları bu yüzden C++ veya Python'da yazılmış olabilir. Bazı simülasyon sonuçları NAM tarafından görselleştirilebilir, fakat yeni görselleştirme programları da geliştirilmektedir. NS-3 pcap iz dosyaları üretir, diğer yardımcı programlar da bu iz dosyalarını analiz edebilir. NS-3 NS-2'nin sahip olduğu modellerin hepsine sahip değildir, ama diğer taraftan NS-3'ün yeni yetenekleri vardır (IPv6 adreslemeyi, daha çok internet protokolleri ve tasarımları ve daha detaylı 802.11 modeli vb. kullanarak düğümler üzerinde çoklu arayüzleri hatasız işlemek gibi). NS-2 modelleri genelde NS-3'e bağlanabilir. NS-3 için birçok yeni gelişim vardır. Tablo 9'da her iki simülasyonun karşılaştırılması yapılmıştır.

Tablo 9. NS-2 & NS-3 karşılaştırması [8]

	NS-2	NS-3
İlk sürüm tarihi	2006	2008
Temel yapı	NS-1&REAL	NS-2 GTNets YANS
Mimari	OTCL & C++	C++&PYTHON Scr.
Kurum	DARPA, VINT SAMAN&NSF	NSF CISE & INRIA
Destek	gönüllüler, USC ISI & sourceforge	NSF.INRIA, GT, WashU & gönüllüler
Script dili	OTCL	Python
Görsel destek	NAM	NS-3-viz, pyviz, NAM, iNSpect
Ölçeklenebilirlik	Sıralı	Dağıtık

4.1. Genel Yapı

NS-2 ve NS-3 ayrıık olay simülasyonlarıdır, fakat yapı olarak aralarında ciddi farklılıklar bulunmaktadır. NS-3

gerçek ağ bileşenleri, protokoller ve API'ler ile yapılacak uygulamalara çok daha iyi sonuçlar üretebilmektedir. IP ve UDP protokolleri detaylı olarak analiz edilebilmektedir. NS-3'ün bir diğer avantajı derlenmiş olan kodların tekrar kullanılabilir olmasıdır. NS-3 kod geliştiricileri düşük seviye kod(makine diline yakın) uygulamalar ile soket haberleşmesi, paket iletimi ve adresleme gibi işlemleri çok daha erken safhalarda gerçekleştirebilirler. Tüm bunların sonucu olarak aynı protokol yapıları üzerinde gerçekleştirilen işlemler değerlendirildiğinde iki simülasyon arasında yapısal farklar bulunmaktadır. NS-3 gelişmiş yapı ve detaylı analiz özellikleri ile ön plana çıkmaktadır. M. H. Rehmani'nin 2010 yılında yapmış olduğu çalışmada NS-2 ve NS-3'ün ağ katman yapılarının karşılaştırılması gerçekleştirilmiştir [9].

4.2. Kullanılabilirlik ve Adaptasyon

Kullanılabilirlik programlama dili ve derleme açısından değerlendirildiğinde NS-2 C++ ve OTCL üzerinde çalışmaktadır. NS-3 ise sadece C++ dilini kullanır, bu sebepten dolayı üzerinde çalışması ve derlemesi daha kolaydır. Ayrıca NS-3'de isteğe bağlı olarak Python ile de çalışmak mümkündür. NS-2 derleme aşamasında geleneksel GNU derleme sistemini kullanırken NS-3 Python dilinde yazılmış olan son zamanlarda yaygın olarak kullanılan WAF sistemini kullanır. Dokümantasyon açısından da NS-2'nin yeterli ve tüm modüler yapısını açıklayacak yeterli dokümana sahip olduğu söylenemez fakat NS-3 çok daha genç ve dinamik bir proje olduğundan çok ciddi bir dokümantasyonu ve sürekli olarak artan bir desteğe sahiptir. İki simülasyonun modüler yapısına bakıldığında NS-2'nin modülleri arasında kolay bir değişim gerçekleştirmek zordur. NS-3'de katmanlar arası sınırlar ve yapı açık ve net bir şekilde belirlenmiştir. NS-3 uygulama ve ağ katmanında çok daha esnek bir yapıya sahiptir.

4.3. Bileşen ve Modelleme

NS-3'de yeni bileşen ve düğüm özellikleri ekleme ve oluşturma işlemlerinin çok daha kolay yapılabilmesi amaçlanmıştır. Bu özellik bir ağ simülasyon programı için ciddi ve tercih edilmesini sağlayacak olan bir sebeptir. NS-3 yapısına yeni protokoller düzenli ve kolay bir şekilde eklenebilir, modelleme gerçekleştirilebilir. Özellikle son zamanlarda kablosuz ağ yapıları ile ilgili ciddi çalışmalar bulunmaktadır.

4.4. Kurulum, Kontrol ve Analiz

NS-2 ve NS-3 temel ağ bileşenlerini ağ senaryoları oluşturmak için kullanılmaktadır. Özellikle NS-3 ağ oluşturma, kurulum ve bağlantıların daha kolay yapılabilmesi için çeşitli yardımcı programlar sunarak işlemleri kolaylaştırmaktadır. Ancak her iki simülasyonun de veri depolama ve yapılan deneyler üzerinde kontrol imkanları sınırlıdır. NS-3 NS-2'ye göre güçlü bir çerçeveye sahiptir. NS-3 paket izleme için pcap gibi, SQLite gibi harici yazılım izleme araç yüzlerini kullanır. Belli bir standart yapının olması NS-3 için bir avantajdır.

4.5. Gelişim Durumu

NS-3 projesinin gelişimi boyunca NS-2 sürekli olarak desteklenmiştir fakat NS yazılım geliştirme grubu sadece NS-3 üzerine odaklanmaktadır. NS-2 üzerinde bu gelişim süresince sadece temel güncelleme işlemleri gerçekleştirilmiştir. NS-3 ün gelişimi için planlanan NSF projesi ancak 2010 yılı sonu itibarıyla tamamlanamamıştır. NS-3 üzerinde çalışmakta olan bazı modüler yapılar NS-2 üzerinden alınmıştır. Son zamanlarda NS-3 için yeni bir çerçeve geliştirme çalışmasının gerçekleştirileceği NSF tarafından ilan edilmiştir. Bu yapı simülasyon üzerinde gerçekleştirilecek olan simülasyonun sonuçlarının daha iyi bir şekilde analiz edilebilmesi için gerekli geliştirmelerin yapılmasına odaklanacaktır. NS-3 için yapılan geliştirme çalışmasının bir kısmı Google Summer of Code (GSOC) tarafından desteklenmektedir.

4.6. Verimlilik ve Performans

NS-2 ve NS-3 verimlilik ve performans açısından değerlendirildiğinde paket kayıpları NS-2 de NS-3'e göre daha fazla olduğu, işlem zamanı ve kaynak kullanımları açısından ise NS-3'ün daha verimli ve performanslı olarak işlemleri gerçekleştirmektedir.

5. SONUÇLAR VE ÖNERİLER

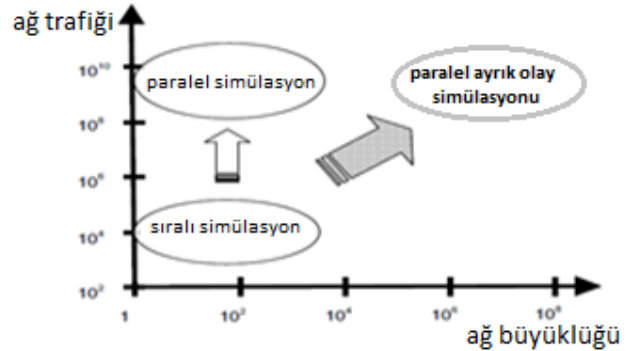
Bilgisayar ağlarının simülasyonu için kullanılan birçok simülasyon arasında açık kaynak kodlu yazılımlar olan NS-2 ve NS-3 diğer birçok lisanslı ve yüksek maliyetli ürünün yanında büyük ölçekli ağların analiz ve simülasyonunda tercih edilmektedir. NS-3 NS-2'nin tüm özelliklerini bünyesinde barındırmamaktadır. Fakat NS-3 ü ön plana çıkaran, önemini artıran en önemli nokta özellikle internet tabanlı uygulamalarda yeni ve etkili yeteneklere sahip olmasıdır.

Bu makalede NS-2 ve NS-3 üzerinde farklı parametreler kullanılarak yapılmış olan testler neticesinde iki farklı ağ yapısı kurularak analizler gerçekleştirilmiştir. Bu bilgilerin tümü değerlendirildiğinde her iki uygulama için farklı donanıma sahip bilgisayarlar üzerinde işlem süresi, işlemci ve bellek kullanım değerleri tespit ölçülmüştür. Koşulların farklılığına bağlı olarak uygulanan örneklerde farklı performanslar elde edilmiştir. Simülasyonun gerçekleştirme süresi değerlendirildiğinde NS-3 simülasyonu, NS-2 simülasyonuna göre ızgara ağ yapısında ortalama 2-3 kat, yıldız ağ topolojisinde ortalama 8 kat daha fazla sürede işlemi tamamlamıştır. İşlem süresi arasındaki bu farkın bir sebebi NS-3'e sonradan eklenen ve NS-2'nin yapısında bulunmayan katmanlara ait işlemlerin yapılmasıdır.

İşlemci kullanımı açısından bakıldığında her iki simülasyonun tüm simülasyon işlemlerinde işlemcinin neredeyse tamamına yakınına kullandığı gözlemlenmiştir. Bellek kullanımı açısından değerlendirildiğinde NS-3, NS-2'ye göre tüm simülasyon işlemlerinde belleği daha az miktarda, etkili ve verimli kullandığı görülmüştür. Büyük ölçekli yapılarda bellek kullanımı açısından değerlendirildiğinde NS-3'ün ciddi oranda öne çıkacağı

görülebilir. NS-3'de düğüm yapısı NS-2'ye göre daha sadeleştirilmiş ve dinamik bir yapı haline getirilmiş, sadece gerekli olan bileşenler ile ilişkilendirilmiş olduğundan dolayı belleği daha efektif kullanmıştır. Tüm parametreler ve sonuçlar değerlendirildiğinde NS-3 ağ simülasyonunun mevcut imkânları daha verimli kullanıp, ağır yapısı büyütüldüğünde daha performanslı ve ölçeklenebilir olduğu sonucuna varılmıştır.

NS-2 ve NS-3 simülasyonlarının karşılaştırılması amacıyla Debajyoti Pal yayımlanmış olduğu makalede [10], ağ simülasyonlarını paket kaybı, işlem süresi ve bellek kullanımı açısından test etmiştir. Testler sonucu elde edilen sonuçlarda ağ büyüklüğüne bağlı olarak NS-2 de daha fazla paket kaybı gerçekleştiği, NS-3 ün belleği çok daha verimli kullandığı, işlem sürelerinin iki simülasyonda da birbirine yakın olduğu tespit edilmiştir. Ayrıca Debajyoti Pal'ın yapmış olduğu çalışmada NS-2 ve NS-3 simülasyonlarının yanı sıra diğer yaygın olarak kullanılan ağ simülasyonlarının de testleri gerçekleştirilmiş, birbiri ile karşılaştırılarak değerlendirme yapılmıştır. Bu makalede yapılan test sonuçları ile Debajyoti Pal'ın yayımlanmış olduğu makaledeki sonuçlar kıyaslandığında bellek kullanımı açısından paralel sonuçlar, işlem zamanı açısından bu makaledeki testlerde NS-3 ağ simülasyonunun işlemleri NS-2 ağ simülasyonundan daha uzun zamanda gerçekleştirdiği tespit edilmiştir.



Şekil 13. Ağ boyutu ve trafik artış grafiği [11]

Şekil 13'deki grafikte sıralı, paralel işlem zamanlı ve paralel ayrıklı olay simülasyonlarının büyüyen ağ boyutları ile birlikte değerlendirildiğinde nasıl bir artış ve ihtiyacı ortaya çıkardığı açıkça görülmektedir. Günümüzün gittikçe büyüyen ve karmaşıklaşan ağ yapılarını simüle edecek ve daha gerçekçi sonuçlar üretip sağlıklı analizlerin ortaya çıkmasını sağlayacak ağ simülasyonlarında duyulan ihtiyaç artmaktadır. Bizim makalemizde üzerinde durduğumuz ve özellikle ölçeklenebilirlik açısından değerlendirdiğimiz NS-3 simülasyonu birçok açıdan gereksinimleri karşılayacak ve ortaya çıkan yeni ihtiyaç ve talepleri karşılayacak, gelişime açık altyapı ve mimariye sahip olduğu, NS-3 üzerinde geleceğe dönük çalışmaların bu kapsamda devam edeceği sonucuna varılabilir.

KAYNAKLAR

- [1] Ölçeklenebilirlik, [http://technet.microsoft.com/tr-tr/library/cc776523\(v=ws.10\)](http://technet.microsoft.com/tr-tr/library/cc776523(v=ws.10)), 2012
- [2] Modelleme ve simülasyon, <http://tr.wikipedia.org/wiki/Sim%C3%BClasyon>, 2012
- [3] NS-2, <http://www.isi.edu/nsnam/ns/>, 2012
- [4] NS-2, [http://en.wikipedia.org/wiki/Ns_\(simulator\)](http://en.wikipedia.org/wiki/Ns_(simulator)) <http://www.nsnam.org/docs/tutorial/tutorial.html>, 2012
- [5] NS-2, <http://www.isi.edu/nsnam/ns/ns-build.html>, 2012
- [6] NS-3, <http://www.nsnam.org/overview/what-is-ns-3/>, 2012
- [7] NAM(Network Animator), <http://www.nsnam.org/wiki/index.php/NetAnim>, 2012
- [8] R. Chaudhary, S. Sethi, R. Keshari, S. Goel, "A study of comparison of Network Simulator -3 and Network Simulator -2", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (1) , 3085 – 3092, 2012
- [9] NS-2,NS-3 comparion, http://www-npa.lip6.fr/~rehmani/ns3_v1.pdf, 2012
- [10] Debajyoti Pal, "A Performance Evaluation of Commonly Used Network Simulators" , **International Journal of Electronics Communication and Computer Engineering**, Volume 3, Issue 1, ISSN 2249 –071X ,2012s
- [11] R. M. Fujimoto, K. Perumalla, A. Park, H. Wu, M. H. Ammar, "Large-Scale Network Simulation: How Big? How Fast?" , **Proceedings of the 11TH IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer Telecommunications Systems (MASCOTS'03)** 1526-7539/03,2003