

논문 2005-42TC-1-10

VoIP 시뮬레이션을 지원하는 네트워크 설계 및 분석 도구의 구현

(Implementation of a Network Design and Analysis Tool Supporting
VoIP Simulations)

최 재 원*, 이 광 휘**

(Jae-Won Choi and Kwang-Hui Lee)

요 약

본 논문에서는 네트워크를 설계하고 분석할 수 있는 실용적인 시뮬레이션 도구의 구현에 대하여 기술하였다. 본 시뮬레이터의 특징은 쉽고 직관적인 사용법, 장비와 프로토콜의 실제 동작 구현, 시뮬레이션용 트래픽의 실제 생성과 전송, VoIP의 지원 등이다. 특히, 본 논문은 VoIP를 지원하는 시뮬레이터의 구현과 적용 방법에 초점을 맞추었다. 음성 트래픽의 특성만을 정의하여 전송하고, 일반 데이터와 동일하게 처리하여 지연과 같은 품질만 분석하는 기존의 도구와는 차별을 두었다. 이를 위하여 통화를 연결하고 해제하는 호 신호와 음성 정보 트래픽을 구분하여 생성하고 처리하도록 하였다. 또한, VoIP 게이트웨이와 게이트키퍼 등의 장비를 탑재하여 호 처리율이나 실패율과 같은 장비의 성능과 함께 PSTN과 인터넷 상호 간의 음성 트래픽 전송 품질을 분석할 수 있도록 하였다. 본 시스템의 구현 방법과 적용 예를 통하여 시뮬레이션 도구의 활용화 방안을 제시하였다.

Abstract

In this paper, we have described the implementation of a practical simulation tool to design and analyze communication networks. Especially, this study is focused on the implementation and application methods of a simulator supporting VoIP. The key characteristics of this particular system are its easy and intuitive usage, the real behaviors implementation of equipment and protocols, the actual generation and transmission of traffic for simulation, supporting of VoIP and so forth. Our system is distinguished from the existing tools which define only the nature of voice traffic, process those packets in the same way as general data, and analyze only the quality of packet transmission such as delay. Our tool presented in this paper generates and processes packets in different way according to the types of traffic distinguishing call signal from voice information traffic. Also, we equipped this system with the various devices such as VoIP gateway and gatekeeper, which enabled this system to analyze the performance of devices and the quality of voice traffic transmission between PSTN and Internet. By presenting the implementation methods and application of this system, we managed to propose the utilization scheme of a simulation tool.

Keywords : Network Simulation, VoIP, Traffic Generation, NetDAS

I. 서 론

현재 운용되고 있는 네트워크를 효율적으로 관리하고 향후의 확장 계획을 체계적으로 수립하기 위해서는 신

뢰성 있는 성능 분석과 예측이 필요하다^{1), 2)}. 즉, 네트워크 관리자 또는 설계자는 경험적인 지식보다는 검증된 도구를 이용하여 네트워크를 모델링하고 설계하며 이에 대한 성능을 분석할 수 있어야 한다. 이를 위한 시뮬레이션 도구는 쉽게 사용 가능해야 하며, 최근의 응용 패턴이나 특성들을 반영할 수 있는 것이어야 한다.

본 논문에서는 네트워크를 설계하고 분석할 수 있는 실용적인 시뮬레이션 도구의 구현에 관하여 언급한다. 본 논문에서 구현한 시뮬레이터의 특징은 쉽고 직관적

* 학생회원, ** 정회원, 창원대학교 컴퓨터공학과
(Dept. of Computer Engineering, Changwon
National University)

※ 본 논문은 한국과학재단의 산학협력연구로 수행된
결과의 일부임(I01-2001-000-00061-0).

접수일자: 2004년10월12일, 수정완료일: 2004년12월1일

인 사용법, 장비의 실제 동작 구현, 시뮬레이션용 트래픽의 생성과 전송, VoIP(Voice over IP)^[3]의 지원 등이다. 시뮬레이션 도구의 구현에 관한 연구로서 최근에 발표된 NCTUns^[4]는 FreeBSD 또는 Linux의 실제 프로토콜 스택과 응용 프로그램을 직접적으로 사용하는 것이 특징이며, 다양한 프로토콜과 무선 환경을 지원한다. 하지만, 본 연구에서 주안점을 두고 있는 VoIP를 직접적으로 지원하지는 않으며, PSTN(Public Switched Telephone Network)과의 연동도 가능하지 않다. 가장 널리 알려진 상업용 시뮬레이터인 OPNET^[5]의 경우 프로파일이라는 특성을 이용하여 세부적인 설정이 가능하므로 다양한 실험을 할 수 있다는 장점이 있다. 또한, 가장 대표적인 공개용 시뮬레이터인 NS-2^[6, 7]는 주로 프로토콜을 개선하여 그 동작을 검증하고 분석하기 위한 도구로서, 사용자는 개발된 코드를 직접 원하는 형태로 수정하고 스크립트를 작성하여야 한다. 이러한 도구들은 설정 과정이 복잡하거나, 프로토콜 또는 프로그래밍에 대한 지식과 경험을 요구하므로 사용에 어려움이 따른다. 이에 반하여, 본 연구에서 구현한 시뮬레이터는 설정 과정을 단순화하고 기능을 직관적으로 구성함으로써 사용자 친화적이다. 또한, 기존의 시뮬레이션 도구들이 이벤트나 수학적 계산에 입각하여 성능을 분석하는데 반하여, 본 도구는 실제 장비의 동작 방식과 프로토콜의 기능들을 모두 직접 구현하여 탑재함으로써 실질적인 시뮬레이션의 수행이 가능하도록 하였다. 즉, 단순한 논리적 트래픽이 아닌 시뮬레이션용 트래픽을 실제로 생성하여 전송하며, 프로토콜이 탑재된 모든 노드는 스케줄링, 스위칭, 라우팅 등을 실제 장비와 동일한 방식으로 수행한다. VoIP 기능에 대하여 OPNET과 NS-2는 음성 트래픽의 특성을 정의하여 전송할 수 있다. 하지만, 이러한 기존의 도구들은 지연과 같은 음성의 품질만을 분석하며 관련된 장비에 대한 사항들은 전혀 포함되어 있지 않다. 이에 본 도구는 VoIP 성능 측정을 위한 게이트웨이와 게이트키퍼 등의 장비를 탑재하고 음성 통화의 연결을 위한 호 신호와 음성 트래픽을 추가하였다. 이를 통하여 음성에 대한 품질뿐만 아니라 호 처리율과 같은 장비의 성능까지 분석할 수 있도록 하였다.

본 논문의 구성은 다음과 같다. II장에서 본 시뮬레이션 도구의 전체적인 구조에 대하여 설명한다. III장은 세부 모듈의 구현 방법들을 기술하고, IV장에서 VoIP 시뮬레이션에 대한 구현을 설명한다. V장에서는 구현한 도구를 이용한 실험 예와 그 결과를 분석한다. 마지

막으로 VI장에서 결론을 맺고 향후 연구에 대하여 언급한다.

II. 시스템의 전체 구조

본 논문에서 제안하여 구현한 시뮬레이션 시스템은 네트워크를 설계하고 그 성능을 분석하기 위한 도구로서 VoIP 시뮬레이션을 지원한다. 사용자는 노드와 링크를 이용하여 전체적인 토폴로지를 형성하고 각 장비의 성능을 결정하게 된다. 그런 후, 트래픽의 발생 패턴을 지정하면 설계된 네트워크는 자동 설정 과정을 거쳐 시뮬레이션이 수행되고 최종적으로 그 성능이 분석된다. 그림 1은 제안 시스템의 전체적인 구조와 흐름을 도시화한 것으로서, 점선 화살표는 명령의 흐름을 나타낸 것이고, 실선 화살표는 프로세스의 흐름과 진행 순서를 표시한 것이다. 아래에서 중요 모듈의 기능에 대하여 설명한다.

1. 디자인 매니저(Design Manager)

네트워크를 설계하기 위한 환경에 해당하는 부분으로서 토폴로지와 장비의 종류 및 속성들을 지정하고 트래픽을 발생시키는 사용자 인터페이스를 제공한다. 디자인 매니저는 노드나 링크 등에 대한 클래스 구조와 장치 정보 베이스(device information base)를 통하여 사용자에게 의해 설계된 네트워크 정보들을 개별적인 객체로 생성하고 시스템이 구동되는 동안 유지 및 관리한다. 사용자는 실제 장비의 모델명을 선택하는 것으로서 장비의 복잡한 성능 및 파라미터 설정 과정을 생략할 수 있게 되는 것이다. 물론, 사용자는 이 값들을 변경할 수 있으며, 해당 장비가 장치 정보 베이스에 존재하지 않는 경우 새로운 모델을 정의할 수도 있다.

NS-2는 토폴로지 생성기의 사용이 가능하기는 하지만, 네트워크 설계를 위해서는 스크립트를 작성하여야

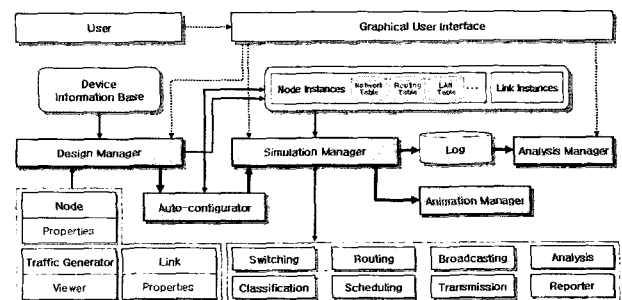


그림 1. 제안 시스템의 구조

Fig. 1. Architecture of proposed system.

한다. 하지만, 본 시스템은 OPNET과 마찬가지로 GUI 방식의 네트워크 설계, 장비의 속성 및 성능 지정, 트래픽의 생성 등이 가능하다. 본 시스템에서 장치 정보 베이스는 실제 상용 제품의 모델명과 그 성능 정보를 유지하고 있는 데이터베이스이다. OPNET도 유사한 방법으로 실제 장비를 지정할 수 있도록 하고 있다. NS-2에는 이러한 기능이 없으며, 기본적으로는 장비의 성능에 대한 분석 결과를 제공하지 못 한다. 또한, OPNET과 마찬가지로 본 시스템도 서브넷의 구성이 가능하나, NS-2에는 이러한 노드의 정의가 없다.

2. 자동 구성기(Auto-configurator)

설계된 네트워크를 시뮬레이션하기 위하여 필요한 환경이나 구성 정보 등을 자동으로 설정하는 일종의 전처리기이다. 자동 구성기가 수행하는 기본적인 작업은 아래와 같다.

- 노드와 링크에 대한 유일한 식별자(id: identifier)의 할당
- 각 인터페이스를 구별하는 유일한 id의 할당
- 유일한 IP 주소의 할당
- 유일한 물리(MAC) 주소의 할당
- 고유한 id와 주소에 대한 할당 오류 검사
- 링크와 같은 논리적 설계 오류 검사

본 시스템은 실제 장비의 운용과 작동 방식을 그대로 반영하였기 때문에 IP 주소나 물리 주소와 같은 값들의 지정이 필요하다. OPNET과 NS-2도 원활한 시뮬레이션의 수행을 위하여 논리적 주소를 할당한다. 또한, 본 시스템은 시뮬레이션 수행 전에 설계상의 각종 오류 검사를 자동으로 수행하게 된다. OPNET은 사용자가 원할 경우에 이러한 기능이 수행된다. NS-2는 이러한 기능이 없으므로 실행 시에 스크립트 또는 소스 코드의 오류가 발생하게 된다.

3. 시뮬레이션 매니저(Simulation Manager)

노드의 동작에 따른 패킷의 흐름을 가시화하는 애니메이션 매니저(Animation Manager)와 함께 실제 시뮬레이션을 수행하는 일을 담당한다. 디자인 매니저와 자동 구성기에 의하여 생성되고 유지되는 정보들을 수집한 후 시뮬레이터가 초기화되고 애니메이션 매니저가 가동된다. 중단 노드는 트래픽을 발생시키고 중간 노드는 패킷 처리를 위한 스케줄링, 테이블 갱신, 경로 산정,

전송 메커니즘 등을 이용하여 패킷을 전달함으로써 시뮬레이션을 수행한다. 또한, 이러한 과정에서 동반되는 패킷 처리와 전달에 관한 모든 이력들을 로그 파일로 기록한다. 이와 관련된 좀 더 구체적인 사항은 3장에서 언급한다.

본 시스템은 시뮬레이션과 동시에 패킷의 흐름을 애니메이션으로 가시화하여 보여 준다. OPNET은 기본적으로 애니메이션을 표현하지 않고, 사용자가 지정할 경우 시뮬레이션이 종료되고 난 후에 로그로 기록된 패킷의 흐름을 애니메이션으로 확인할 수 있다. NS-2 또한 시뮬레이션 동안의 패킷 흐름을 트레이스 파일로 기록한 후 *nam*^[8]이라는 도구를 이용하여 애니메이션을 볼 수 있다.

4. 분석 매니저(Analysis Manager)

시뮬레이션 과정에서 생성된 로그 기록을 이용하여 네트워크 장비와 패킷 흐름에 대한 성능을 분석한다. 분석 결과는 수치 정보와 함께 그래프로 도식화하여 보고서 형식으로 출력할 수 있도록 함으로써 예측 및 정책 결정의 자료로 활용될 수 있도록 한다.

OPNET은 분석 결과를 자체 뷰어를 통하여 제공하며, 사용자가 지정할 경우 HTML 형식으로 변환이 가능하다. NS-2는 시뮬레이션 과정을 트레이스 파일로 저장하고 변환 과정을 거치면 *xgraph*^[9]라는 도구로 분석 결과 그래프를 볼 수 있다. 반면에, 본 시스템은 시뮬레이션 과정이 기록된 로그 파일을 분석하여 수치와 그래프 정보가 포함된 HTML 형식의 보고서를 생성하고 웹 브라우저를 통하여 이를 보여 준다.

III. 동작 원리와 구현 방법

본 장에서는 제안 시스템의 핵심 기능과 모듈에 대하여 기본 설계 방법과 그 동작 원리를 설명한다.

1. 시간 관리와 스케줄링

본 시스템에서는 시뮬레이션의 특성상 가상의 시간 개념을 도입하여 사용되 기본 단위를 ns(nanosecond)로 하였다. 스케줄링의 문제에 있어서, OPNET이나 NS-2는 전체 스케줄러가 이벤트를 기반으로 시간에 따른 작업을 하나씩 처리하는 구조를 가진다. 반면에, 본 시스템의 스케줄링은 본질적으로 라운드 로빈 방식을 따른다. 즉, 노드와 링크 같은 모든 객체에 대하여 특정한 시간 단위(quantum)만큼 처리 시간을 할당하여 작

업을 수행한다. 여기서, 각 라운드마다 할당되는 고유한 시간 단위를 T_Q 라 하고, 특정한 객체에서의 처리 시간 또는 지연 시간을 T_P 라 한다면, 본 시스템에서는 이 시간 단위를 $T_Q < \min\{T_{Pi}\} (i=1, 2, \dots, n)$ 을 만족하는 T_Q 의 최대값으로 지정하였다. T_Q 의 값이 T_P 보다 큰 경우, 동일한 객체 내에서의 처리에는 문제가 없으나, 라운드 로빈에 따라 다음 번 순서에 해당하는 객체로 처리가 넘어갔을 때 특정한 처리가 수행되어야 할 시간이 이미 경과해 버리게 되는 상황이 발생하기 때문이다.

2. 주소의 할당

시스템이 각 장치들을 효율적으로 구분하고 제어할 수 있도록 하기 위하여 자동 구성기는 각각에 대하여 고유한 id 와 주소를 부여한다. 각 노드에 대하여 노드 id 를, 모든 인터페이스에 대하여 그룹 id 와 인터페이스 id 를 각각 할당한다. 그리고 이들을 결합시켜 아래 그림 2와 같이 IP 주소와 물리 주소를 구성한다.

OPNET이나 NS-2가 논리적 일련 번호를 할당하는 것과 달리 본 시스템은 실제 환경과 동작의 구현을 위하여 토폴로지를 분석하고 네트워크를 구분한다. 그런 후, B 클래스에 해당하는 32 bit의 명시적인 IP 주소가 부여되고, 48 bit의 물리 주소가 할당된다.

3. 패킷의 구조와 트래픽의 생성

시뮬레이션을 실제 동작과 동일하게 수행하기 위하여 본 시스템에서는 실질적인 패킷을 직접 생성하여 목적지로 전송한다. 시뮬레이션의 효율을 높이기 위하여 아래 그림 3과 같이 이더넷 프레임의 구조를 일부 변경하였으며, 구현상으로는 바이트 배열 형태이다. 물론, 필요한 경우 새로운 형태의 패킷 구조를 정의할 수 있도록 하였다.

트래픽의 양상에 따라 네트워크는 다른 성능을 보일 수 있기 때문에 시뮬레이션에서 트래픽의 발생 패턴은 중요한 요소이다^[10]. 따라서 시뮬레이션 시스템은 크기,

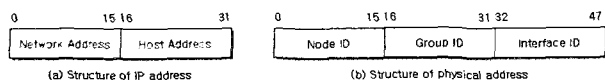


그림 2. IP 주소와 물리 주소의 구조 (단위 : bit)
Fig. 2. Structures of IP and physical addresses (unit : bit).

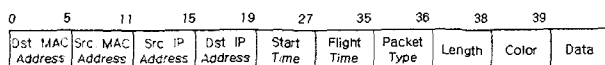


그림 3. 시뮬레이션 패킷의 구조 (단위 : byte)
Fig. 3. Structure of simulation packet (unit : byte).

시간 간격, 목적지 등이 구분되는 트래픽을 다양한 방법으로 생성할 수 있어야 한다. 본 논문에서는 표 1과 같은 종류와 방법으로 트래픽 생성 수단을 제공한다. 그림 4는 트래픽을 생성하는 화면의 예를 보여 주는 것으로, OPNET처럼 수학적 분포 함수^[11]를 지정한다.

OPNET과 NS-2는 기본적으로 프레임 계층에서만 트래픽을 생성하는 반면에, 본 시스템은 혼합 모드를 비롯하여 응용 계층의 트래픽 생성을 지원한다. 또한, NS-2는 음성 패킷을 생성하는 방법이 제공되지 않는다. 그리고 OPNET은 중단 노드에서 음성 정보에 해당하는 트래픽을 생성하고 코딩을 수행한다. 중간 노드에서는 일반 데이터 트래픽과 특별히 구분되지 않고 동일하게 처리된다. 반면에, 본 시스템은 호를 설정하거나 해제하기 위한 신호 패킷을 추가적으로 지원하며, 게이

표 1. 트래픽 생성의 종류와 방법
Table 1. Types and methods of traffic generation.

종류	계층 모드	설명
데이터	프레임 단일	링크를 통해 최종적으로 전송되는 프레임의 크기를 기본 단위로 하여 생성한다.
	프레임 혼합	링크를 통해 최종적으로 전송되는 프레임의 크기를 기본 단위로 하여 생성되되, 버스티한 특성과 정도를 반영하도록 여러 가지 분포 함수를 혼합하여 지정한다.
	응용 단일	중단 사용자가 전송하고자 하는 데이터의 크기를 기본 단위로 하여 생성한다. 이렇게 생성된 트래픽은 하위 계층에서 MTU에 따라 분할 또는 패딩 된다.
음성	호 신호	음성 통화의 연결을 설정하거나 해제하기 위한 신호를 생성한다.
	음성 정보	하나의 통화동안 발생하는 음성 트래픽을 위의 프레임 혼합과 유사한 방법으로 지정한다.

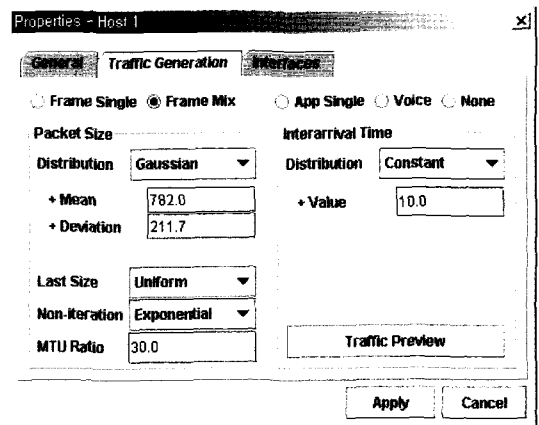


그림 4. 트래픽 생성 예
Fig. 4. Example of traffic generation.

트웨이나 게이트키퍼와 같은 장비에서 성능에 따라 호의 설립에 대한 수락 여부 등이 결정된다. 그리고 음성 정보 또한 다른 패킷과는 달리 이들 장비에서 데이터 패킷으로의 미디어 변환 과정 등을 거치게 된다.

4. 노드의 구조와 패킷의 처리

호스트와 호스트 그룹 등의 종단 노드는 데이터 패킷과 음성 패킷을 발생시킬 수 있다. 또한, OPNET이나 NS-2와는 달리 IP 전화기와 일반 음성 전화기가 추가되었다. 그리고 중간 노드에 대해서는 VoIP 처리를 위한 게이트웨이와 게이트키퍼 장비가 추가되었다.

모든 노드는 링크 상으로 패킷을 전송하고 링크 상의 패킷을 수신하는 기능을 가진다. 라우터 노드에는 라우팅 프로토콜이 구현되어 탑재되어 있다. 라우터들은 동적으로 라우팅 정보를 실제 교환하며 테이블을 갱신한다. 또한, 들어오는 패킷들을 구분하여 시뮬레이션용 데이터 패킷, 통화 연결을 위한 호 신호, 음성 패킷, 라우팅 패킷 등의 종류를 판별하고 적절한 처리를 수행한다.

5. 로그의 기록과 성능 분석

본 시스템은 모든 패킷 흐름에 대한 로그를 기록한다. 이는 OPNET이나 NS-2와 그 방법과 정보가 유사하다. 모든 패킷의 입력과 출력에 대하여 수신 시간, 처리 시간, 송신 시간, 입출력 인터페이스 등이 기록된다. 또한, 노드에 대해서는 매 순간의 처리율, 메모리 사용률, 패킷 손실 정보 등이 모두 로그로 기록된다.

라우터와 같은 프로세싱 노드는 메모리 사용량과 노드에서의 지연 시간 이외에 패킷에 대한 초당 처리 성능을 분석 결과로 제시할 수 있어야 한다. 이 성능을 위하여 PPS(Packet Per Second) 단위가 사용되며, 아래 식 (1)과 같이 표현된다.

$$PPS = 1s / (Inter Frame Gap + Preamble Time + Frame Time) \quad (1)$$

Inter Frame Gap : 프레임 사이의 시간 간격

Preamble Time : 프레임 앞에 붙는 시간

Frame Time : 프레임이 전송되는데 걸리는 시간

본래의 성능은 장비의 백플레인과 같은 요인에 의하여 결정이 되며, 시뮬레이션 시에 실제 처리 성능을 기록에 남긴 후 분석을 수행하게 된다. 종단간의 지연 시간은 패킷이 지나가는 모든 노드와 링크 상에서의 지연 시간의 합으로 계산되며, 아래 식 (2)와 같다.

$$E2E Delay = Serialization Delay + Propagation Delay + Switching Delay \quad (2)$$

Serialization Delay : 패킷을 링크 상에 직렬화하는 시간

Propagation Delay : 매체를 통해 1bit를 전송하는 시간

Switching Delay : 패킷이 장비에 입력되어 출력되기까지의 시간

다른 지연 시간은 고정된 값이거나 특정한 요인에 비례 또는 반비례하게 되므로 스위칭 지연 시간을 줄이는 것이 중요하다. 이는 토폴로지의 설계와 장비의 성능 등이 좌우하게 된다.

IV. VoIP의 지원

IP 텔레포니는 인터넷 응용 중의 하나로서 IP와 인터넷을 이용하여 음성을 전송하는 기술이다. 이러한 서비스를 제공하기 위해서는 특별한 장치와 관련 기술이 뒷받침되어야 한다. 본 연구에서는 이러한 VoIP 성능을 분석할 수 있는 시뮬레이션 기능을 구현하였다. 본 장에서는 이에 대하여 구체적으로 설명하도록 한다.

1. VoIP 시뮬레이션 관련 기존 연구

VoIP의 지원과 관련하여 기존 도구들과의 차이점을 본 연구자가 현재까지 파악한대로 설명하면 다음과 같다.

NS-2는 새로운 프로토콜이나 메커니즘의 개발을 실험하기 위하여 적합하다. 이러한 점으로 인하여 본 시스템과는 시뮬레이션의 목적이나 용도가 다르다. 특히, 소스 코드를 원하는 형태로 수정하지 않는 한 노드의 종류를 구분하지 못하며, 음성 트래픽의 패턴을 정의하고 전송할 수 없다. 또한, 실제와 동일한 장비, 특히 VoIP와 관련된 장비의 성능을 시뮬레이션할 수 없다.

OPNET은 NS-2와 달리 범용 네트워크와 장비의 성능 측정도 가능하다. OPNET의 경우, 종단 노드는 음성 트래픽을 정의하고 발생시킬 수 있다. 하지만, 음성 트래픽은 라우터 등의 프로세싱 노드에서 일반 데이터 트래픽과 동일하게 처리된다. 단지 음성 트래픽에 대하여 패킷의 TOS(Type of Service) 값 등 QoS 관련 파라미터만이 구분될 수 있으며, 프로세싱 노드는 이 값과 큐 스케줄링 방식에 따라 패킷을 처리하게 된다. 또한, PSTN과 연동이 되지 않고, 연결 설정을 위한 시그널 프로토콜로 SIP(Session Initiation Protocol)^[12]만을 지

원한다. 즉, VoIP 게이트웨이와 게이트키퍼 장비가 포함되어 있지 않아 이에 대한 성능 측정을 할 수 없으며, IP 망 내에서의 음성만을 시뮬레이션할 수 있다. 음성에 대한 분석 결과로 송/수신 패킷량, 음성 전송에 대한 종단간 지연 시간과 지연 변이 등을 제공한다. 또한, SIP에 대한 분석 결과로 연결된 통화의 수, 요청이 거부된 통화의 수, 통화 지속 시간 등을 볼 수 있다.

위와 같이 기존의 시뮬레이션 도구들은 VoIP를 완전하게 지원하지 못 하고 있다. PSTN과의 연동을 통하여 음성 통화를 다양하게 모델링하고 VoIP 관련 장비의 성능을 분석할 수 있도록 하기 위하여 본 연구에서 적용한 방법들을 설명하면 아래와 같다.

2. 호 신호와 음성 트래픽의 생성

호스트는 일반 데이터 패킷과 음성 트래픽을 모두 생성시킬 수 있고, IP 전화기와 일반 전화기는 음성 트래픽만을 생성시킬 수 있도록 노드를 설계하였다. 본 절에서는 그 중에서도 음성 통화와 관련된 부분만 설명하고자 한다.

음성 통화를 위해서는 통화를 설정하기 위한 신호, 실제 음성 트래픽, 통화를 해제하기 위한 신호를 각각 생성할 수 있어야 한다. 그림 5는 이러한 신호와 트래픽의 관계를 간략하게 보여 준다. 시그널링 프로토콜을 이용하여 통화가 연결 또는 해제되며, 이는 특별히 통화의 종단을 초래하는 장애가 발생하지 않는 한 하나의 쌍으로 이루어진다. 하나의 호 설정 신호와 하나의 호 해제 신호 사이에서 하나의 통화 세션이 이루어진다. 하나의 통화는 실제 음성이 활성화 되어 있는 시간이 있을 수 있으며, 그렇지 않은 시간도 있을 수 있다. 그리고 하나의 통화가 끝나면 임의의 시간이 경과한 후

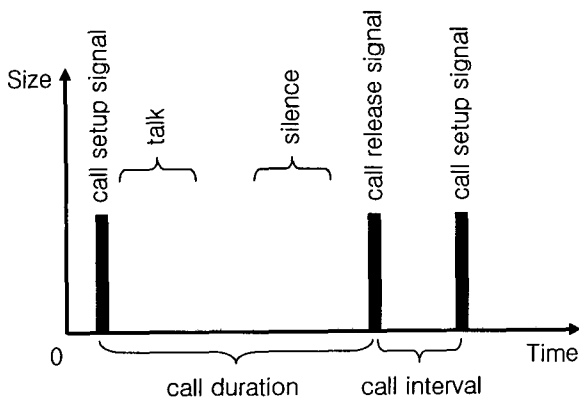


그림 5. 호와 음성의 발생
Fig. 5. Generation of call and voice.

또 다른 통화가 시도될 수 있다.

본 시스템에서는 음성 트래픽의 발생을 위하여 위에서 설명한 호 신호와 음성 트래픽, 그리고 이들 각각의 시간 간격을 개별적인 분포 함수를 이용하여 지정하고 이에 대한 파라미터를 입력할 수 있도록 하였다. 즉, 시뮬레이션하고자 하는 음성 트래픽의 특성을 정의할 수 있도록 하였다.

3. PSTN 연동을 위한 VoIP 장비의 시뮬레이션

본 연구에서는 단순히 IP 망에서 음성 데이터를 생성하여 다른 데이터 정보와 동일한 방식 또는 우선순위 기법으로 처리하는 기존의 시뮬레이션 도구와 차별성을 두고자 하였다. 실제로 이와 관련되어 적용되고 있는 VoIP 게이트웨이와 게이트키퍼 장비를 추가하고, IP 전화기는 물론 PSTN을 구성하기 위한 일반 음성 전화기 노드를 추가하였다. 이를 통하여 패킷 스위칭 네트워크인 IP 망과 순수 음성 통화를 위한 회선 교환 네트워크인 PSTN의 상호 연동을 시뮬레이션할 수 있게 된다. 즉, IP 망에서 호스트와 호스트 간의 음성 트래픽 전송 뿐만 아니라, IP 망과 PSTN, 즉 호스트 또는 IP 전화기와 일반 음성 전화기와의 음성 통화에 대한 시뮬레이션이 가능하게 된다.

게이트웨이는 PSTN 미디어와 인터넷 IP 패킷간의 미디어 변환 기능, SS7^[13]과 H.323^[14] 또는 SIP간의 시그널링 변환 기능 등을 수행한다. 게이트키퍼는 H.323과 IP의 주소 변환, H.323에 대한 수락 제어, 단말기의 대역폭 요구 관리 등을 수행한다. 이 두 장비에서 중요한 성능 정보는 사용자 수용 용량, 단위 시간당 호 처리 성능, 코덱, 프로토콜 등이다. 그림 6은 VoIP를 위한 게이트웨이 장비의 설정 예를 보여 준다. 호 신호에 의하

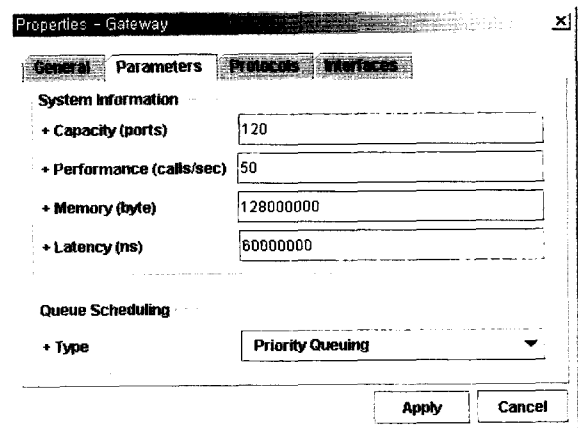


그림 6. VoIP 게이트웨이의 설정 예
Fig. 6. Example of VoIP Gateway configuration.

여 장비의 처리 성능에 따라 호의 수락 여부가 결정되고, 음성 트래픽은 코덱을 이용하여 인코딩 된다. 일반적으로 코덱은 실제 장비에서 하드웨어적으로 구현되어 있기 때문에 인코딩 시간은 고정된 값을 가진다. 따라서 본 연구에서도 이를 위한 시간은 상수값으로 입력받을 수 있도록 구현하였다.

VoIP 장비에 대한 분석 결과는 프로세싱 노드에 대한 일반적인 성능, 즉 자원 사용률, 패킷 처리율, 패킷 손실, 노드 상에서의 지연 시간 등과 더불어 호 처리율, 호 실패율 등을 제시한다. 그리고 음성 품질에 대한 분석 결과로 종단간 지연 시간과 지연 변이 등을 보여줄 수 있다.

V. 실험 예 및 결과 분석

본 장에서는 구현한 시스템을 이용하여 네트워크를 설계하고 분석한 예를 설명하고자 한다. 그런 후, 본 시스템의 제약 사항을 논의한다. 본 시뮬레이터는 IP 망을 비롯한 기존의 다양한 망에 대한 성능 분석이 가능하지만, 본 논문에서는 VoIP의 기능과 성능에 대한 시뮬레이션에 대해서만 언급하도록 한다. 기타의 기능은 참고 문헌 [15]에서 구체적으로 언급하고 있다.

1. 실험을 통한 구현 결과

본 실험에서 설계한 네트워크는 3개의 서브넷을 가지며, 라우터 8개, 스위치 30개, 게이트웨이 9개, 게이트키퍼 6개 등으로 구성된다. 종단 노드는 호스트 그룹에서 지정한 속성을 포함하여 호스트 893개, IP 전화기 24개, 음성 전화기 84개이다. 시뮬레이션을 위한 트래픽의 발생은 다양한 패턴으로 설정하였다. 호스트는 데이터 패킷과 음성 패킷을 모두 발생하도록 하였는데, VoIP의 성능 측정을 위하여 음성 패킷 위주로 발생시켰다. IP 망에 접속되는 IP 전화기와 PSTN 망에 접속되는 일반 전화기는 음성 패킷만이 생성된다. 또한, 실험을 위하여 호스트들과 전화기들이 비교적 많은 양의 트래픽과 음성 통화를 발생하도록 설정하였다. 모든 노드와 링크는 장애가 발생하지 않는다는 가정을 하였으며, 라우팅 프로토콜은 RIP(Routing Information Protocol)로 설정하였다.

본 실험은 시뮬레이션 시간을 30분으로 설정하여 수행하였다. 물론, 사용 가능한 프로토콜은 RIP 이외에도 OSPF(Open Shortest Path First)와 같은 다양한 프로토콜들이 있으며, 장애가 발생하는 경우에 대한 사항은

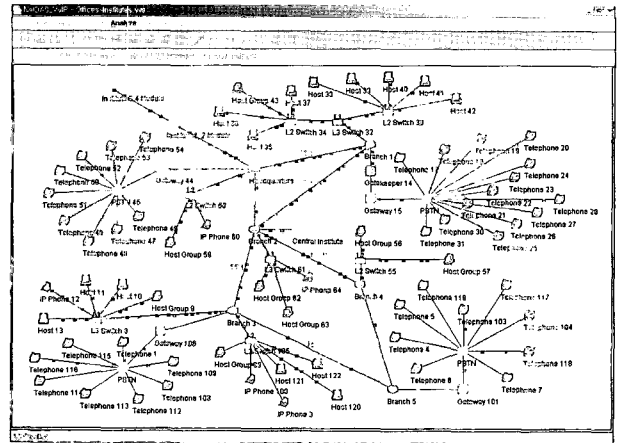


그림 7. 시뮬레이션 화면 예
Fig. 7. Example screen of simulation with animation.

Layer	Packets	...
Layer 1	1000000	...
Layer 2	1000000	...
Layer 3	1000000	...
Layer 4	1000000	...
Layer 5	1000000	...
Layer 6	1000000	...
Layer 7	1000000	...
Layer 8	1000000	...
Layer 9	1000000	...
Layer 10	1000000	...
Layer 11	1000000	...
Layer 12	1000000	...
Layer 13	1000000	...
Layer 14	1000000	...
Layer 15	1000000	...
Layer 16	1000000	...
Layer 17	1000000	...
Layer 18	1000000	...
Layer 19	1000000	...
Layer 20	1000000	...
Layer 21	1000000	...
Layer 22	1000000	...
Layer 23	1000000	...
Layer 24	1000000	...

그림 8. 분석 결과 화면 예
Fig. 8. Example screen of analysis results.

본 논문이 아닌 다른 논문에서 언급하도록 한다. 그림 7은 위에서 설명한 것과 같이 설계한 네트워크의 시뮬레이션 화면을 보여 주고 있다. 시뮬레이션을 가동시키면 먼저 라우팅 프로토콜이 동작하여 네트워크 정보들을 교환한다. 이와 같이 신호 정보들을 주고받은 후 네트워크의 안정화가 이루어지고 나면 실제 시뮬레이션이 동작한다. 일반 데이터 패킷, 음성 패킷, 라우팅 패킷, 호 신호 등이 색깔로 구분되어 시뮬레이션의 수행과 동시에 애니메이션으로 표현된다.

위의 그림 8은 HTML 문서로 표현되는 분석 결과 화면의 일부이다. 실험 결과의 일부를 설명하면 아래와 같다. 패킷 처리율, 장비 사용률, 패킷 손실, 노드 지연, 링크 사용률과 같은 장비의 성능에는 특별한 문제가 없었으며 종단간의 전달 지연에도 특이점은 없었다. 그러나 특정한 지역의 게이트웨이와 게이트키퍼에서 호의 실패율이 5% 정도로 나타났다. 또한, 이 지역과의 음성 통화에서 패킷 전송의 종단간 지연 시간이 평균 300ms 정도로 나타나 통화 품질에 문제가 있는 것이 발견되었다. 이러한 문제를 해결하기 위하여 이 지역에 회선을 증설시키고 장비를 추가하였다. 그 결과, 호의 실패율은 0이 되었으며, 음성 패킷의 종단간 지연 시간도 150ms 이내가 되어 적절한 조치가 이루어졌음을 확인할 수 있

었다.

2. 시스템의 제약 사항

본 연구에서 개발한 시뮬레이션 시스템의 수행 환경에는 제한이 없다. 자바 언어로 구현되어 플랫폼에 독립적으로 실행되기 때문이다. 단지, 1절에서 예시한 실험은 Intel Pentium IV 2.0GHz, 256MB DDR DRAM 하드웨어 환경을 가지는 Windows 2000 Advanced Server 운영 체제 하에서 수행되었다.

시뮬레이션에 걸리는 시간은 시뮬레이션이 실제로 수행되는 시스템의 성능 및 실험 당시의 시스템 부하에 따라 다르다. 다만, 시뮬레이션의 수행 시간만으로 본다면, 기존의 시뮬레이션 도구에 비하여 상대적으로 수행 시간이 오래 걸린다고 할 수 있다. 이는 두 가지 요인에 기인한다. 먼저, 이벤트 위주로 시뮬레이션이 진행되는 일부 도구들과는 달리, 본 시스템에는 노드와 프로토콜의 동작 메커니즘들이 실제와 동일하게 구현 및 탑재되어 있고, 이 과정들이 모두 수행되기 때문이다. 또한, 시뮬레이션과 애니메이션이 완전히 분리된 기존의 도구들과는 달리, 본 시스템은 기본적으로 이들을 동시에 수행하기 때문이다. 위와 같은 이유로 인하여 동일한 조건에서 시뮬레이션 시간을 비교하는 것은 불가능하다. 그럼에도 불구하고, 동일한 망을 모델링하여 비교해 본 결과, 망의 크기에 따라 다소의 차이는 있지만 평균적으로 다음과 같은 결과를 얻었다. 본 시스템의 수행 시간은 OPNET보다 약 10배가 더 길었다. 하지만, 시뮬레이션 수행 시간과 애니메이션 수행 시간을 합하였을 경우에는 OPNET이 본 시스템보다 약 3배의 시간이 더 소요되었다.

확장성 문제에 관한 한 구현상의 제한은 없다. 즉, 본 도구를 이용하여 설계하는 네트워크의 수, 서브넷의 수, 노드의 수, 발생 트래픽의 양 등은 한계를 가지지 않는다. 이 문제는 본 도구를 수행하는 실제 컴퓨터 시스템의 하드웨어 사양이나 운영 체제에 영향을 받을 수 있다. 다만, 본 시스템은 논리적으로 모든 IP 노드들에 대하여 B 클래스에 해당하는 주소를 할당하므로, 이에 기인하는 제한점은 있을 수 있다. 즉, 토폴로지 상으로 볼 때 전체적으로 최대 2^{16} 개의 네트워크를 구성할 수 있으며, 하나의 네트워크 내에는 최대 2^{16} 개만큼의 노드를 배치할 수 있다.

VI. 결론 및 향후 연구

본 논문에서는 네트워크를 설계하고 분석할 수 있는 시뮬레이션 도구의 개발에 대하여 설명하였다. 그 중에서도 VoIP를 다양한 환경에서 시뮬레이션할 수 있는 기능에 특히 초점을 맞추었다. 본 시스템은 일반적인 네트워크뿐만 아니라, 최근 인터넷의 응용으로 주목받고 있는 VoIP까지 분석할 수 있는 도구이다. 음성 트래픽 정보를 발생시켜 이에 대한 전송 품질만을 분석할 수 있는 기존의 도구와는 달리, 시뮬레이션 시 VoIP 게이트웨이와 게이트키퍼 장비 등을 분석할 수 있는 기능을 추가하였다. 이를 통하여 통화 품질은 물론 호의 처리율과 실패율 등을 분석 결과로 제시할 수 있게 되었다. 실험을 위하여 다양한 데이터 패킷, 호 신호, 음성 트래픽 등을 발생시켰으며, 예로서 설계한 네트워크에 대한 시뮬레이션 결과를 통하여 본 시스템의 실제 적용 방법과 활용 방안을 소개하였다.

본문의 실험 예에서 알 수 있는 것처럼 본 시스템은 네트워크의 설계와 확장 분야, VoIP 도입 분야 등에서 성능 분석 및 예측 도구로서의 역할을 수행할 수 있을 것이다. 향후에는 좀 더 다양한 응용 서비스를 분석할 수 있는 기능을 탑재하여야 하며, 시스템의 효율을 향상시킬 필요가 있다.

참고 문헌

- [1] Teresa C. Mann-Rubinson, Kornel Terplan, *Network Design: Management and Technical Perspectives*, CRC Press, Aug. 1998.
- [2] Robert S. Cahn, *Wide Area Network Design: Concepts and Tools for Optimization*, Morgan Kaufmann Publishers, May 1998.
- [3] Uyless black, *Voice over IP*, Prentice Hall PTR, 1999.
- [4] S.Y. Wang, C.L. Chou, C.H. Huang, C.C. Hwang, Z.M. Yang, C.C. Chiou, and C.C. Lin, "The Design and Implementation of the NCTUns1.0 Network Simulator," *Computer Networks*, Vol. 42, No. 2, pp. 175-197, June 2003.
- [5] OPNET, <http://www.opnet.com/>
- [6] NS-2, <http://www.isi.edu/nsnam/ns/>
- [7] Lee Breslau, et al., "Advances in Network Simulation," *IEEE Computer*, Vol. 33, No. 5, pp. 59-67, May 2000.
- [8] Nam, <http://www.isi.edu/nsnam/nam/>
- [9] Xgraph, <http://www.isi.edu/nsnam/xgraph/>

[10] Uie-Soo Jun and Kwang-Hui Lee, "HTTP Traffic Modeling and Analysis with Statistical Process," *Journal of KSII*, Vol. 5, No. 4, pp. 63-76, Aug. 2004.

[11] Sheldon M. Ross, *Introduction to Probability Models*, 7th ed., Academic Press, Feb. 2000.

[12] J. Rosenberg, et al., *SIP: Session Initiation Protocol*, IETF RFC 3261, June 2002.

[13] P.J. Kuhn, C.D. Pack and R.A. Skoog, "Common Channel Signaling Networks: Past, Present, Future," *IEEE Journal on Selected Areas in Communications*, pp. 381-393, April 1994.

[14] ITU-T Recommendation H.323, *Packet-based multimedia communications systems*, Nov. 2000.

[15] Jae-Won Choi and Kwang-Hui Lee, "Design and Implementation of a Network Simulation System," in *Proc. of PDCN 2004*, pp. 245-250, Innsbruck, Austria, Feb. 17-19, 2004.

저 자 소 개



최 재 원(학생회원)
 2000년 2월 창원대학교
 컴퓨터공학과 (공학사)
 2002년 2월 창원대학교
 컴퓨터공학과 (공학석사)
 2002년 3월~현재 창원대학교
 컴퓨터공학과 박사과정

<주관심분야: QoS/네트워크 관리, 라우팅 알고리즘, 네트워크 시뮬레이션>



이 광 휘(정회원)
 1983년 2월 고려대학교
 전자공학과 (공학사)
 1985년 2월 고려대학교
 전자공학과 (공학석사)
 1989년 2월 고려대학교
 전자공학과 (공학박사)

1991년~1992년 영국 Wales 대학(Swansea) 및
 Newbridge Networks사 연구원

1994년~1995년 영국 런던대학(UCL) 연구원

1997년~1999년 영국 Reading 대학 연구원

2000년~현재 영국 Nortel Networks사 연구원

1988년~현재 창원대학교 컴퓨터공학과 교수

<주관심분야: QoS/네트워크 관리, 정책기반 네트워크, 라우팅 알고리즘, 트래픽 엔지니어링, 모바일 컴퓨팅>

