

# 네트워크 트래픽 성능 향상을 위한 액티브 노드 및 액티브 네트워크 설계

최병선, 황영철, 이성현, 이원구, 이재광  
한남대학교 컴퓨터공학과 네트워크 실험실  
042-629-7559/042-629-7658

## Active Node and Active Network Modeling For Network Traffic Progress

Byung-Sun Choi, Young-Chul Hwang, Seong-Hyun Lee, Won-Goo Lee,  
Jae-Kwang Lee  
{ychwang, bschoi, shlee, wglee, jklee}@netwk.hannam.ac.kr

### Abstract

Computer simulation has used to a area of military training from about several years ago. War game model(or computer simulation) endow a military man with field training such as combat experience without operating combat strength or capabilities. To samely construct simulation environment against actual combat environment is to well construct DB to operate war game model, associate among federates on network. Thus, we construct virtual combat environment enabling to efficiently manage network traffic among federates(or active nodes) on active network that construct virtual military training space such as urgent combat field needed to rapidly transfer

combat information including image and video.

### 1. 서론

컴퓨터 시뮬레이션(혹은 가상 전장환경)처럼 미래엔 사람이 직접 전쟁터에서 전쟁을 하는 것이 아니라 첨단 과학기술로 만들어진 다양한 무기체계를 활용하여 상대의 정보를 입수하고, 목적과 용도에 따라 상대의 취약한 곳을 공격하게 될 것이다. 최근에는 컴퓨터 시뮬레이션을 통해서 실제 전투자산을 가동하지 않고 실전과 같은 전투경험을 부여하고 있다. 가상 모의훈련 전장환경은 컴퓨터를 이용한 가상의 전장환경을 제공하여, 군사작전의 분석 및 훈련용으로 사용된다. 하지만 환경적인 특성상 다량의 전장 데이터(이하 액티브 패킷)의 송수신을 고려했을 때, 종단장치(시뮬레이

※ 본 연구는 대학 IT 연구센터 육성/지원사업의 연구 결과로 수행되었음

선 서버와 IGI 호스트)간의 네트워크 트래픽 부하가 현저하게 증가하여 중간 노드(네트워크 서버, IGI 서버)들의 트래픽 처리율이 현저하게 저하될 수 있고, 이는 실제 전장환경과 유사한 환경을 구축하고 모의훈련 하고자 하는 가상 모의훈련 전장환경의 구성에 대한 실패를 초래할 수도 있다.

이러한 문제점 들을 해결하고자 하는 연구가 아직은 미비하지만, 향후 다각적인 측면에서 연구될 것이다[1].

본 논문의 구성은 다음과 같다. 2장에서는 지금까지 알려진 분산 시뮬레이션 및 액티브 네트워크 구조에 대해 살펴보고, 3장에서는 모델링을 수행할 대상 네트워크(혹은 가상 전장환경)의 성능향상을 위한 다양한 액티브 네트워크 기법에 대해서 설명할 것이다. 4장에서는 새로이 제시한 액티브 노드 및 네트워크에 대한 모델링을 하며, 5장에서는 결론을 맺고 향후 연구방향에 대해 논의한다.

## 2. 관련연구

### 2.1 분산 시뮬레이션 구조

분산 시뮬레이션의 어려움에도 불구하고 네트워크 기술을 통한 분산 시뮬레이션 기술은 끊임없이 발전해 왔다. 네트워크 기술을 이용한 분산 시뮬레이션은 개별 무기체계 수준의 장비조작 및 전투 기술 훈련에 사용해 오던 각종 시뮬레이션들을 하나의 컴퓨터 네트워크를 통해 연결함으로써 동일한 전장공간에서 시뮬레이터들을 운용할 수 있도록 하기 위한 시도로 출발했다[2]. 그러나 기존의 시뮬레이션 모형들은 대부분 각종 모의기능을 단일 모형에 통합하여 묘사하는 중앙집중식 모의구조를 갖는다. 이러한 중앙집중식 모의구조를 갖는 단일 모형으로는 보다 복잡하고 다양한 현대전을 실천과 유사한 훈련환경으로 조성하는 것이 불가능하게 되었다. 따라서 다양한 유형의 시뮬레이션

모델들을 상호연동하고 통합하는 방법을 통해 보다 복잡한 새로운 목적을 달성할 수 있도록 하기 위한 방안들이 요구되었다. 이러한 시도의 하나로 미 국방성에서는 1995년부터 표준연동구조(HLA: High Level Architect-ure)를 개발하게 되었고, 2000년에는 IEEE1516 국제표준으로 등록되었다 [3].

### 2.2 액티브 네트워크(Active Network)

#### 2.2.1 액티브 네트워크(Active Network)

액티브 네트워크는 노드에서 패킷에 포함된 프로그램의 실행이 가능하도록 한 네트워크이다. 따라서 기존 네트워크에서 노드는 단순히 데이터전송을 목적으로 패킷의 헤더 정보만을 처리한데 비해, 액티브 네트워크에서는 사용자가 노드 기능을 특성화시킬 수 있으므로, 사용자 중심의 네트워크 구성이 가능하게 된다. 이에 따라 액티브 네트워크는 표준화 작업없이 빠른 속도로 새로운 서비스들이 전개되도록 한다. 액티브 네트워크에서 프로그램 가능한 패킷은 일반 패킷과 구분하기 위해 접근 방법에 따라 스마트 패킷(Smart Packet) 또는 액티브 패킷(Active packet)이라 부른다. 본 논문에서는 통합 방법(Integrated approach)에서의 액티브 패킷을 사용하고, 액티브 패킷은 데이터와 프로그램 코드를 포함하게 되며, 실제로 이 패킷에는 가상 전장공간이라는 환경적 특성상 PDU(Protocol Data Unit)를 포함하게 된다. 아래 그림 2는 액티브 보안을 포함한 액티브 네트워크의 구성으로서 대응노드(ARN: Active Response Node)와 실행노드(AEN: Active Execution Node)를 기본 단위로 하는 기본 도메인(elementary domain)과 기본 도메인간 상호 연동된 ASM(Active Security Management) 도메인, 그리고 도메인을 관리하기 위한 중앙관제시스템(CCS: Central Coordinator System)으로 구성된 액티브 네트워크의 예이다. 아래 그

림 1의 액티브 네트워크 모델 간에는 유사한 형태의 구조와 관리라는 측면을 갖고 있지만, HLA 모델에서의 문제점을 액티브 네트워크 모델을 통해 극복할 수 있다는 측면 또한 내포하고 있다. 이에 우리는 두 모형간의 통합방안과 성능향상 방안에 대하여 제시하고, 모의실험을 통해 검증하고자 한다.

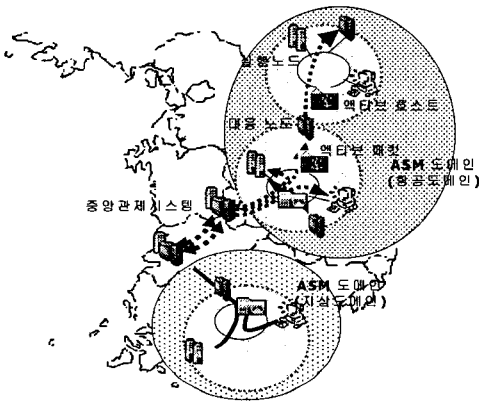


그림 154 액티브 네트워크 응용

### 2.2.2 액티브 라우팅(Active Routing)

현재 많은 연구소와 대학에서 액티브 네트워크 기술을 연구하고 있지만, 주로 액티브 노드 구조나 액티브 응용에 관한 연구를 하고 있으며, 액티브 네트워크 라우팅 프로토콜에 관한 연구는 미흡한 수준이다. 현재까지 제안된 액티브 노드들은 액티브 네트워크 토폴로지가 구성되어 있다고 가정하거나 혹은 정적으로 구성하는 방법을 사용한다, 정적인 구성 방법은 구현은 용이하지만 관리자의 많은 노력을 수반하며 특히 네트워크의 동적인 상황을 반영하지 못한다. 또한 액티브 네트워크 상의 특정 액티브 노드를 이용해 세션을 설정하는 액티브 응용[4]의 경우는 정확한 네트워크 토폴로지를 제공해 주어야 하며, 따라서 동적인 액티브 네트워크 라우팅 프로토콜이 요구된다.

## 3. 네트워크 트래픽 성능 향상 방안

현재 많은 연구소와 대학에서 액티브 네트워크 기술을 연구하고 있지만, 액티브 네트워크 상에서의 트래픽 제어, 액티브 라우터간의 동적 경로설정, 액티브 패킷에 대한 자원할당 및 효율적인 코드 캐싱과 같은 네트워크 트래픽 성능 향상에 관한 연구는 미흡한 수준에 있으며 반드시 해결해야 할 사항이다. 그림 2에서는 우리가 가상 전장공간 상에서 액티브 네트워크를 모델링하기 위한 우선적인 고려사항에 대해 보여주고 있다. 첫째, 네트워크 상의 모든 노드가 액티브 노드가 아니기 때문에 액티브 패킷을 다음 노드로 전달할 때 직접 다음 액티브 노드로 전달할 수 있는냐는 것. 둘째, 일반 IP 네트워크와 함께 구축되어 있는 액티브 네트워크 상에서 패킷을 어떠한 경로를 통하여 빠르게 전달시키며, 트래픽 제어를 어떻게 하느냐는 것. 셋째, 액티브 노드로 들어온 패킷에 대한 자원할당을 어떻게 효율적으로 할 수 있는냐는 것. 넷째, IGI 호스트로부터의 요청에 대한 응답으로 액티브 노드에서 멀티캐스트를 어떻게 효율적으로 처리할 수 있는냐는 것이다.

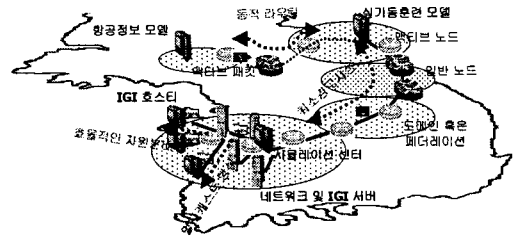


그림 2 가상 전장환경 상에서의 액티브 네트워크 구성

### 3.1 네트워크 상의 트래픽 제어

액티브 네트워크에서는 패킷 내에 프로그램이나 처리에 필요한 정보를 담아 전송하게 되는

데 네트워크 상에 있는 노드는 액티브 패킷(Active Packet)을 읽어 들여 각 라우터의 실행 환경(Execution Environment)에서 실행할 수 있고 필요한 프로그램(혹은 코드)을 다른 곳으로부터 다운로드하여 실행 가능하다. 또한 네트워크 내부 노드들은 서로에게 필요한 네트워크 상의 정보를 공유할 수 있다. 이 기술을 사용할 경우 가상 전장환경 상에서 혼잡이 발생하면 네트워크 내부에서 이를 감지하여 주변 라우터에 혼잡에 대한 정보를 알려줄 수 있으며 직접적으로 혼잡을 제어할 수 있다[6]. 액티브 네트워크 상에서 패킷이 액티브 노드(Active Node)에 수신되었을 경우 액티브 노드 상에 있는 혼잡정보(Congestion Information)와 액티브 패킷에 있는 혼잡 정보를 이용하여 패킷에 대한 처리를 수행하기 때문에 액티브 노드를 이용한 혼잡 제어는 액티브 노드의 위치와 노드 간의 통신이 중요하다고 할 수 있다[7]. 하지만, 본 연구에서는 액티브 노드의 위치에 관해서는 고려하지 않는다.

### 3.2 액티브 노드간의 동적경로 설정

현재 위게임 모델의 발전 과정으로 미루어 볼 때 기술적, 상업적인 이유로 가상 모의훈련 전장환경 상의 모든 노드가 액티브 노드로서 동작할 수 없고, 가상 전장공간의 모든 네트워크가 액티브 네트워크로 구축될 수 없다. 따라서 하나의 어플리케이션 작업 및 액티브 패킷 전송을 수행하기 위한 여러 개의 액티브 패킷들이 가상 전장공간상의 같은 액티브 노드를 경유한다는 것은 매우 중요한 문제이다. 즉, 송신지에서 목적지까지의 라우팅이 이루어진다면 액티브 패킷의 경로는 패킷마다 서로 다른 경로를 통해 전달될 가능성이 생긴다. 이 상황은 다음과 같은 액티브 어플리케이션에 있어서 심각한 문제를 유발한다. 액티브 네트워크 토폴로지 정보를 기반으로 몇몇 특정 액티브 노드를 세션에 포함

하는 액티브 어플리케이션의 경우, 반드시 같은 세션에 속한 액티브 패킷은 해당 액티브 노드를 순서대로 거쳐가야 한다. 혹은 이전 액티브 패킷의 결과가 다음 액티브 패킷의 처리과정에 영향을 주는 액티브 어플리케이션의 경우도 이전 액티브 패킷의 경로를 통해 전달되어야 한다. 또한 같은 세션에 속한 액티브 패킷이 서로 다른 액티브 노드를 거쳐 간다면, 각 액티브 노드마다 액티브 코드 요청 과정을 수반하게 되는 상황이 발생하게 되는 것이다. 따라서 송신 노드가 중간의 액티브 노드 위치에 대하여 모르더라도 액티브 패킷이 모두 같은 노드를 거치도록 하는 정적 및 동적 경로설정 기법의 적용을 통해 전송지연을 최소화하였다.

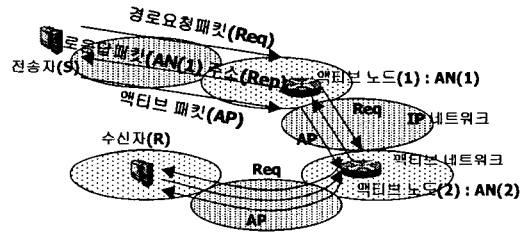


그림 156 액티브 패킷의 경로 설정 과정

### 3.3 액티브 노드에서의 효율적인 코드 캐싱 기법

본 논문에서는 무수히 발생하게 되는 코드 요청에 대한 캐시 적중률(hit-ratio)을 높이고 실행 코드를 효율적으로 캐시(cache)에 유지함으로써 코드 요청 횟수를 줄이기 위한 효율적인 캐싱 기법을 제안하였다. 기존의 캐싱 기법에 응용되는 기본적인 알고리즘으로는 FIFO(First-In-First-Out), LFU(Least Frequently Used), LRU(Least Recently Used) 기법 등이 있다. 본 논문에서 제안하는 캐싱 기법은 캐시 관리자가 코드의 참조 횟수, 즉 코드

의 실행 횟수에 대한 필드와 시간 제약(time limit)에 대한 필드를 참조하는 캐시를 관리하는 기법이다. 코드 정보 테이블(Code Information Table)에는 각 코드에 대한 식별자를 나타내는 필드(Code ID)와 코드 실행 횟수에 대한 정보를 나타내는 필드(Execution No), 시간 제약을 나타내는 필드(Time Limit No)로 구성된다. 코드 실행 횟수 필드는 코드가 최초 실행될 때 '1' 값을 가지며, 재실행 될 때마다 값을 '1' 씩 증가 시킨다. 이렇게 함으로서 코드 실행 횟수에 대한 정보를 유지한다. 시간 제약 필드는 시간 제약 값에 대한 정보를 주기적으로 갱신하여 유지한다. 코드가 최초로 실행된 후에 코드는 시간 제약에 대한 일정한 값(예를 들어 10)을 가지게 된다. 캐시 관리자는 코드가 실행된 후 일정 시간 마다(예를 들어 10시간) 시간 제약 값을 '1' 씩 감소시킨다. 시간 제약 값은 코드가 재실행 될 때 다시 초기 값(10)을 가진다. 제안한 알고리즘의 참조 횟수에 관한 알고리즘은 LFU 기법을 따른다. 그림 3은 제안한 기법의 동작순서를 순서도로 나타내었다. 순서도는 Execution No 필드를 참조하여 해당 코드를 삭제하는 과정과 Time Limit No 필드를 참조하여 해당 코드를 삭제하는 두 개의 과정이다.

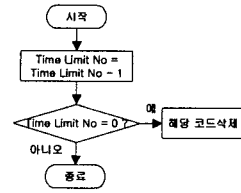
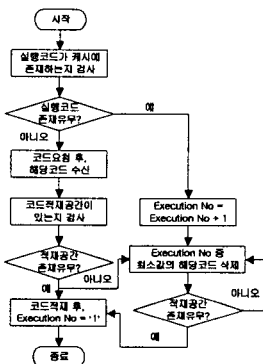


그림 4 코드 캐싱 기법의 동작 과정

액티브 노드에서는 다양한 패킷을 처리하기 위한 다양한 실행 코드가 필요하다. 이러한 실행 코드를 효율적으로 캐시에 유지함으로써 코드 요청 횟수를 줄임으로써 실행 코드 요청에 대한 패킷 처리 대기 시간을 감소시키고 패킷 처리 속도를 향상시킬 수 있을 것으로 생각된다.

## 4. 가상 전장환경에서의 액티브 네트워크 모델링

### 4.1 액티브 노드 모델링

본 논문에서는 이를 위해 액티브 패킷에 추가적인 IP 헤더를 캡슐화하여 전달하는 IP 터널링 방법을 사용하도록 제안한다. IP 터널링 방법은 현재 Mobile IP와 MBONE 등에서 사용되고 있는 방법이다. 다음으로, 다음 액티브 노드를 선택하는 과정에 있어서는 액티브 패킷의 처리 결과와 액티브 네트워크 토폴로지 정보를 고려해야 한다. 만약 처리 결과를 통해 라우팅 정보를 얻을 수 있다면, 이 라우팅 정보를 통해 다음 액티브 노드를 선택할 수 있다. 그렇지 않다면, 액티브 노드 자신이 유지하고 있는 라우팅 정보를 기반으로 다음 액티브 노드를 결정해야 한다. 만약, 인터넷의 모든 라우터가 액티브 라우터로서 동작한다면 기존 인터넷 라우팅 프로토콜에 의해서 만들어진 라우팅 정보를 그대로 사용할 수 있을 것이다. 그러나 액티브 노드들을 물리적으로 서로 연결되어 있지 못할 수도 있기 때문에 액티브 노드들만으로 구성된 액티브 네트워크

크 토폴로지 정보를 유지하고 액티브 패킷의 라우팅을 위한 액티브 라우팅 테이블을 별도로 구성해야 한다. 즉, 새로운 액티브 네트워크 라우팅 프로토콜이 요구된다. 이웃한 액티브 노드들이 물리적으로 직접 연결되어 있지 못하기 때문에 이웃한 노드에 대한 최선의 정보(이웃한 노드들의 리스트, 이웃한 노드와 링크 상태)를 유지하고 그에 따른 대응이 어렵기 때문에, 액티브 네트워크 라우팅 프로토콜은 일반적인 인터넷 라우팅 프로토콜보다 좀더 복잡할 것이다[5]. 기존의 액티브 노드에 있어, 패킷이 수신되면 우선 패킷 분류기가 액티브 패킷과 일반 패킷을 분류하여 액티브 패킷일 경우 패킷 내에 들어 있는 전장정보나 프로그램을 실행환경으로 보내어 처리하게 한다. 실행환경에서 처리 후 얻어진 정보는 상태 정보 표(State Information Table : SIT)에 저장하며 패킷 생성기를 이용하여 주변 라우터에 혼잡에 대한 정보를 제공할 수 있다. 본 논문에서 제안한 액티브 노드는 그림 5와 같이 기존의 액티브 노드에 하나의 작업을 수행하기 위해 필요한 일련의 액티브 패킷들이 같은 액티브 노드를 경유하도록 하기 위한 경로 설정 기법을 제공하는 경로설정 모듈, 네트워크 상의 트래픽을 제어하기 위한 혼잡제어모듈 등을 포함하고 있는 구조를 가지고 있다.

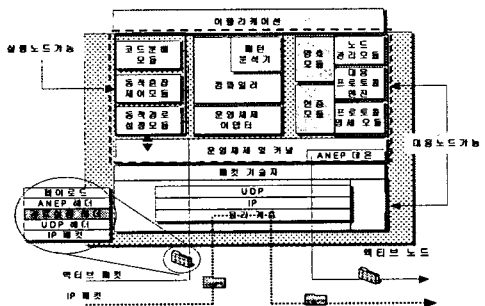


그림 5 액티브 노드 모델링

#### 4.2 액티브 네트워크 모델링

##### 4.2.1 가상 전장환경상의 액티브 네트워크 구성시 고려사항

가상 전장공간 상에서 액티브 네트워크를 구성하기 위해서는 다음의 두 가지 상황을 고려해 볼 필요가 있다. 첫째는 도메인 내부의 노드 중 일부가 일반 노드(Non-Active Node)인 경우, 둘째는 도메인과 도메인간의 노드가 일반 노드인 경우이다.

첫 번째 상황인 도메인 내부의 노드 중 일부가 일반 노드일 경우는 기존의 액티브 네트워크 기술에서 해결책을 다루고 있다. 즉, 캡슐이 액티브 노드를 지날 때는 해당 노드에서 필요한 처리를 하지만, 일반 노드를 통과할 때는 forward 시키도록 구성하고 있다. 두 번째 상황인 도메인과 도메인 사이의 중간 노드들이 일반 노드인 경우는 앞서 기술된 동적기법 등에서 이미 가정하고, 해결책을 제시하였으며, 앞의 상황보다 좀 더 신중한 모델링이 필요하다. 액티브 네트워크의 도메인 구성에서 액티브 노드를 관리 대상으로 편입하였다. 따라서 중간 노드가 액티브 노드일지라도 자신이 속한 도메인의 관리 제어를 받지 않는 액티브 노드는 정책상 일반 노드로 처리한다. 시뮬레이션 서버를 포함한 도메인(서버 도메인)에서 IGI 호스트를 포함한 도메인(이하 호스트 도메인)으로 캡슐이 전송될 때, 서버 도메인의 게이트웨이 역할인 노드는 서버 도메인이 관리하는 마지막 노드가 된다. 서버 도메인에서 마지막 노드를 지나는 순간부터 호스트 도메인 영역의 게이트웨이 역할의 노드에 도달하기까지 거치는 중간 노드는 모두 일반 노드로 인식을 하여 전송(forward) 처리하도록 한다. 이 경우는 서버 도메인과 호스트 도메인은 신뢰 관계(Trust)가 선행되어 있어야 하며, 노드의 신뢰 정보가 각각의 도메인에 소속된 노드 운영체제 컨트롤러 사이에 교환이 되어져야 한다. 도메인의 노드에서는 대상 도메인

으로 캡슐을 전송하기 전에 액티브 노드간에 신뢰 관계를 확인하여 이와 같은 캡슐 전송 처리를 한다.

#### 4.2.2 가상 전장환경에서의 액티브 네트워크 구성예

위에서 기술한 고려사항을 포함한 하부구조를 기반으로 창조 21 모델(하나의 커다란 도메인)이라는 가상 전장환경 상에서 액티브 네트워크 모델을 그림 6과 같이 구성하여 보았다.

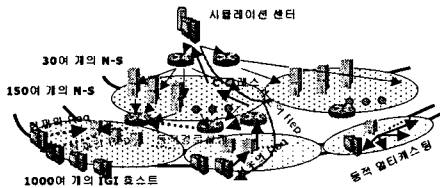


그림 6 도메인(혹은 시뮬레이션 모델) 안에서의 액티브 네트워크 구성

예를 들어, A라는 서버넷 안에 있는 IGI 호스트가 하나의 전장자료 혹은 서비스를 시뮬레이션 서버에 요청(최초의 Req)하게 될 경우에 기존의 위게임 모델에서는 시뮬레이션 서버가 모든 응답(최초의 R데)을 해야만 했다. 이것은 B라는 서버넷에 있는 IGI 호스트로부터 똑같은 서비스 요청(현재의 Req)을 받게 될 경우 시뮬레이션 서버가 똑같은 응답을 호스트에 전달해야만 한다. 이것은 시뮬레이션 서버를 포함한 전체 도메인에 커다란 부하를 가져오게 되며, 위게임 모델 자체가 운영되지 못하게 된다. 이에 그림 14에서와 같이 B라는 서버넷에 있는 IGI 호스트로부터 똑같은 서비스 요청(현재의 Req)이 있을 때, A 서버넷의 액티브 노드에서 이미 자료를 저장하고 있으므로 B 서버넷의 액티브 노드에 서비스(코드)를 전송하고, 그 서비스를 전달받은 노드에서 서비스를 요청한 IGI 호스트에 전달하면 된다. 이것은 요청이 작을

때에는 커다란 효율성을 갖지 못하지만, 요청(액티브 패킷)의 양일 많아질수록 가상 전장환경의 네트워크 트래픽을 현저하게 감소시킬 수 있을 것이다.

## 5. 결론 및 향후 연구

최근의 가상 모의훈련용 위게임 모델에서의 다양한 액티브 응용을 지원하기 위해서는 액티브 네트워크의 링크, 실시간 전송, 보다 효과적인 멀티캐스트 프로토콜이 요구된다. 전장 화상회의, 동영상 및 고용량 이미지와 같은 큰 용량의 데이터를 실시간으로 전송하기 위해서는 데이터의 특성을 고려해야 한다. 최근에는 데이터의 특성을 보다 효과적으로 적용하기 위하여 액티브 패킷을 사용해서 멀티미디어 데이터의 특성에 따라 대역폭과 전송지연을 미리 예측 처리하여 데이터를 전송함으로써 데이터 트래픽의 공정한 분배와 전송속도를 증가시킬 수 있게 되었다.

이에 본 논문에서는 기존 액티브 네트워크 기술들에 대한 통합을 통해 이러한 문제점들을 해결하고 가상 전장 환경에 적용한 후, 아래와 같은 모의실험을 통해 그 유효성을 검증 하였다. 우선, 기존의 혼잡제어 방식이 혼잡 발생 시점에서 혼잡에 대한 처리를 할 수 없다는 문제점을 가지고 있다. 즉 혼잡이 발생한 후 패킷 누락에 대한 신호가 전송자에게 도착 했을 경우에 혼잡을 감지하게 된다. 혼잡을 감지하는데 걸리는 이런 제어시간이 길어지면 길어질수록 혼잡은 더욱 악화된다. 혼잡 발생과 동시에 혼잡에 대한 처리를 할 수 있다면 전송자는 보다 빠르게 혼잡에 대처 할 수 있을 것이다. 끝으로, 액티브 패킷이 같은 액티브 노드를 지나도록 하기 위하여 액티브 노드를 찾기 위해 패킷을 보내는 방법을 적용하여 노드들이 중간 노드들의 위치나 상태에 대한 정보를 가지고 있지 않아도 되

고 기존의 IP 라우팅 프로토콜을 사용하며 IP 계층에서 조각화가 발생해도 같은 액티브 노드를 경유할 수 있도록 하였다.

향후 연구로는 지금까지의 개념을 좀더 확장하여 액티브 라우터와 일반 라우터간의 상호작용을 통해서 가능한 대역폭을 최대한 활용할 수 있는 방안을 연구하는 것이 필요하고, 광대역 네트워크를 구축했을 때의 네트워크 트래픽을 제어할 수 있는 연구가 요구되며. 모의 실험결과를 토대로 실제적인 네트워크 상의 모든 트래픽에 대해서 범용적으로 혼잡제어가 가능한 라우터 구성에 대한 연구가 이루어져야 할 것이다. 또한 본 연구에서 적용한 각각의 확장 기법을 확장하여 네트워크 레벨의 Qos를 지원하는 라우팅 기법과 네트워크 안정성을 위한 보안 라우팅 기법에 대하여 연구해야 할 것이다.

[6] Floyd, S., and Jacobson, v., "Random Early Detection Gateways for Congestion Avoidence," IEEE/ACM Transaction on Networking, August 1993.

[7] 안상현, 김경춘, 한민호, 나중찬, "액티브 라우터를 가진 IP 네트워크를 위한 OSPF 프로토콜의 확장 및 액티브 패킷 전달 방식", 정보과학회논문지, 제30권 제1호, 2003.

## 참고문헌

- [1] 장상철, 손미애, 정상윤, 장동욱, "차세대 시뮬레이션 연동체계 기술연구", 한국국방연구원, 2001.
- [2] 김대석, 양병희, 류제철, "합성전장훈련체계 개발방안 연구 I, II", 국방과 기술, 257호, 258호, 2000.
- [3] 서혜숙, 김태윤, "HLA 기반의 시뮬레이션 모형 개발 및 확장 연구", 한국정보처리학회 추계 학술발표논문집 제8권 제2호, 2001.
- [4] Sumi Choi et al., "Configuration Sessions in Programming Networks," Proc. IEEE INFOCOM, 2001.
- [5] Jonathan T. Moore and Michael Hick, "A Service Layer Routing Protocol for PLAN," PLAN Project, November 1997. <http://www.cis.upenn.edu/~switchware/PLAN>