

CDN 인프라를 기반으로 한 멀티미디어 콘텐츠의 효율적인 전송 방안에 관한 연구

김선호[†] · 김기영^{**} · 신용태^{***}

요 약

최근 인터넷 이용자의 급증과 멀티미디어 콘텐츠의 증가로 인한 네트워크의 정체 현상을 해결하기 위한 방안으로 CDN 기술이 대두되고 있다. 본 논문에서는 멀티미디어 콘텐츠를 안정적이고 신속하게 전송할 수 있는 방안으로 CDN 인프라를 기반으로 하여 CDN 서버 그룹간의 멀티캐스트 전송과 CDN 서버와 사용자 사이에 다중 코딩과 다중 경로 전송을 제안하였다. 이러한 연구는 멀티캐스트 전송에 있어서의 비 신뢰성 문제를 해결하고 멀티미디어 콘텐츠의 전송 효율을 높여 웹 콘텐츠 분야의 새로운 비즈니스 모델 창출에 기여할 수 있을 것이다.

A Study on Efficient Transmission of MultiMedia Contents based on CDN Infrastructure

Seonho Kim[†], Kiyoung Kim^{**} and Yongtae Shin^{***}

ABSTRACT

The CDN is proposed as a solution of network bottleneck problem resulting from the rapid growth of Internet user and multimedia contents on the Internet. This paper suggests MDC and path diversity between CDN Edge server and end-user, and multicast transmission among CDN servers based on CDN infrastructure. It may be a solution for reliable and fast transmission of rich multimedia contents. These research will support to solve the unreliable transmission problem of multicasting and to create the new business models in the field of web contents.

Key words: CDN, MDC, SDC, Multicast

1. 서 론

최근 네트워크 사업자인 ISP(Internet Service Provider)와 최종사용자 사이의 가입자망이 ADSL, VDSL과 같은 초고속 통신망으로 대체되어 최종사용자가 사용할 수 있는 대역폭이 확대됨에 따라 인터넷 이용자가 급증하게 되었고 웹 콘텐츠에 있어서도

HTML 기반의 텍스트 위주에서 벗어나 전자상거래, 뉴스 등 대용량의 멀티미디어 콘텐츠에 대한 수요가 증가하고 있다[1]. 그러나, 멀티미디어 콘텐츠는 대역폭을 많이 차지하며 전달되는 조건이 복잡하여 특정한 인터넷 노드에 과중한 부하가 걸리게 되며 특히 사용자의 가입자망에서 콘텐츠 제공자(Content Provider)의 웹 서버에 이르는 구간을 일컫는 미들 마일(Middle-Mile)[2]에서는 여러 ISP가 많은 라우터를 통하여 서로 중첩되는 과정에서 콘텐츠가 심하게 손상되고 전송이 지연되는 등의 문제가 있다.

이러한 문제를 해결하기 위한 방안으로 CDN(Content Delivery Network)이 대두되었다. CDN은 CP(Content Provider) 서버의 콘텐츠를 다수의 지역

본 논문은 한국과학재단 특정기초(과제번호 :R01-2001-000-00362-0) 연구비 지원에 의해 수행되었음.

접수일 : 2003년 1월 22일, 완료일 : 2003년 4월 7일

[†] 정희원, 동덕여자대학교 정보과학대학 강의전임교수

^{**} 숭실대학교 컴퓨터학과 박사과정

^{***} 숭실대학교 컴퓨터학부 부교수

에 설치된 캐시 서버로 미리 복제해 놓고 사용자가 콘텐츠 전송을 요구하면 현재 사용자의 위치와 가장 가까운 캐시 서버를 하나 골라 이 캐시 서버로 하여금 콘텐츠를 신속한 경로를 통해 제공하는 기술이다. 이 기술이 적용되면 복잡한 인터넷의 미들 마일을 지나는 트래픽의 총량을 줄일 수 있으며 콘텐츠의 전송 속도를 향상시킬 수 있다.

한편, 원격 진료, 주문형 비디오, 화상 회의 등 멀티미디어 정보를 네트워크에 효율적으로 전송하기 위해서는 멀티캐스트 기반의 서비스 구현이 요구되고 있으며 현재 인터넷에서는 일-대-다 혹은 다-대-다 간의 통신을 제공하는 멀티캐스트 서비스를 사용하기 위하여 가상 네트워크인 Mbone(Multicast Backbone)을 통한 IP Multicast 서비스가 제공되고 있다. 그러나, IP Multicast는 IP Unicast 서비스와 마찬가지로 best-effort 전송을 기반으로 하며 신뢰성을 보장하는 메커니즘은 제공하지 못한다. 또한 인터넷 상에서 신뢰성을 보장하기 위해 사용하는 프로토콜인 TCP는 IP Multicast를 지원하지 못한다. IP Multicast 서비스는 어느 정도의 패킷 손실에는 영향을 받지 않는 응용에는 적합한 서비스를 제공할 수 있지만, 데이터의 손실 없는 전송이 보장되어야 하는 응용에서는 상위 계층에서 신뢰성을 보장하는 추가적인 방법이 제공되어야 한다.

그러므로, 본 논문에서는 CDN 인프라에 기반 하여 멀티미디어 데이터를 하나의 서버가 아닌 사용자와 가까운 2개의 Edge 서버를 선택하여 2개의 경로에 의해 데이터를 전송함과 동시에 다중 코딩(Multiple Description Coding)을 이용하여 미디어 스트림을 여러 개의 스트림으로 분할하여 코딩하고 각 코딩된 스트림을 독립된 경로로 전송함으로써 전송 에러의 발생률을 줄이고 발생된 에러에 대해서는 자체 복구할 수 있는 신뢰성 있는 멀티미디어 전송을 제안한다.

2. 관련연구

2.1 CDN : Content Delivery Network

CDN은 기간 망과 가입자망간의 연결을 물리적인 망의 증설을 통해 개선하는 것이 아니라 병목현상의 대상인 데이터 트래픽, 즉 콘텐츠를 인터넷 네트워크의 주요 지점으로 분산시킴으로써 해결하고자 하는

기술이다. 이 기술이 적용되면 복잡한 인터넷의 미들 마일을 지나는 트래픽의 양을 줄일 수 있으며 원본 콘텐츠를 분산 설치된 여러 서버에 분배, 동기화하여 CP(Content Provider) 서버의 부하를 줄이고 각 사용자와 최단 경로를 통해 콘텐츠를 전송하므로 콘텐츠의 전송 속도를 향상시킬 수 있으며 콘텐츠 전송 품질과 안정성을 보장할 수 있다.

CDN 아키텍처는 일반적으로 사용자, CP의 원본 콘텐츠가 저장되어 있는 오리진 서버, 원본 콘텐츠를 복사해 놓은 서버, 사용자의 요청에 대한 라우팅, CP의 콘텐츠 분배, CDN 네트워크 사용에 대한 지불 시스템 등으로 구성된다[3]. 이러한 CDN 요소들과 여러 CDN 간의 연동을 위한 프로토콜에 관한 연구가 IETF의 워킹 그룹인 CDI(Content Distribution Internetworking)에서 진행 중이며 연동을 위한 모델, 아키텍처, 콘텐츠 분배 및 라우팅 등에 관한 드래프트들이[4] 토의 중에 있다. 또한, 하나의 CDN 네트워크에 사용자가 집중될 경우 동적으로 사용자로 하여금 다른 CDN으로 옮겨 갈 수 있도록 하여 확장성과 성능을 향상시키고자 하는 연구[5], CDN 네트워크 상에 최적의 콘텐츠 배치를 위한 연구[6], 그리고 콘텐츠의 요청에 대해 가장 적절한 단말 서버 선택을 위한 연구[7] 등 인터넷을 통한 고품질의 멀티미디어 서비스 전송을 위한 CDN 인프라에 대한 연구가 학계와 업계에서 활발히 진행 중에 있다.

CDN 서버들은 일반적으로 그림 1에서 보듯이 CP 오리진 서버로부터 계층 구조를 이루며 정책적, 지역적으로 분산된다.

CDN 네트워크에서의 콘텐츠 전송 과정이 기존의 콘텐츠 전송 과정과 다른 점은 기존 방식에서는 그림

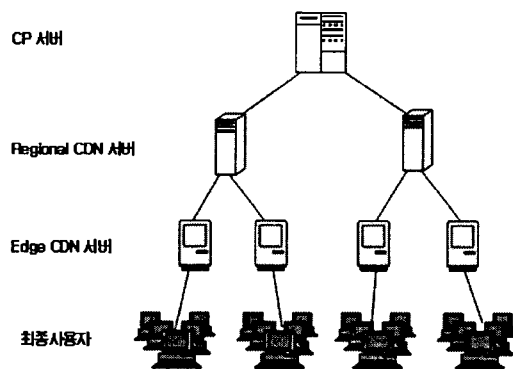


그림 1. CDN 서버 구조

2와 같이 사용자가 CP 서버에 콘텐츠를 요청하면 (①), CP 서버로부터 여러 네트워크를 거쳐 사용자에게 콘텐츠가 전송된다(②).

반면, CDN에서의 콘텐츠 전송 과정은 그림 3과 같이

- ① 사용자가 CP 서버에 콘텐츠 요청하고
- ② CP 서버는 콘텐츠 대신에 콘텐츠의 복사를 가지고 있는 사용자와 가장 가까운 Edge 서버의 위치 정보를 전송하면
- ③ 사용자는 Edge 서버에 콘텐츠를 요청하게 되고
- ④ Edge 서버로부터 사용자에게 콘텐츠가 전송된다.

즉, 중간 경로를 거치지 않고 사용자와 가까운 Edge 서버로부터 사용자에게 직접 전송함으로써 그 결과, 콘텐츠의 손실이 줄고 CP 서버의 부하가 경감되며 동영상 등 대용량의 콘텐츠를 빠른 속도로 서비스 하는 것이 가능하다.

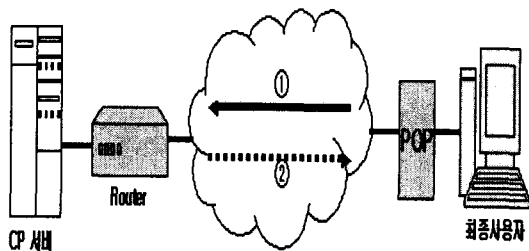


그림 2. 기존 네트워크에서의 콘텐츠 전송 과정

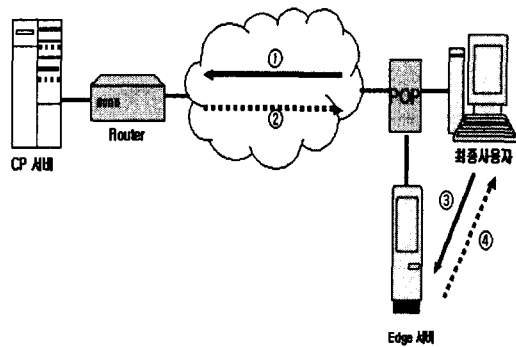


그림 3. CDN 네트워크에서의 콘텐츠 전송 과정

2.2 신뢰성 있는 멀티캐스트

멀티캐스트는 UDP 기반의 best-effort 서비스를 기반으로 하여 단지 데이터를 전송할 뿐이며 수신자들이 데이터를 제대로 받았는지에 대한 확인은 하지

않는다. 이를 해결하고자 유니캐스트 전송에서의 TCP 프로토콜과 같은 기능을 할 수 있는 신뢰적인 멀티캐스트 전송을 위한 연구의 필요성이 대두되었다. IETF의 RMT(Reliable Multicast Transport) 워킹 그룹에서는 신뢰성을 보장하는 멀티캐스트 서비스에 필요한 오류 제어와 혼잡 제어에 관한 토의가 활발히 진행되고 있다.

멀티캐스트는 일-대-다 또는 다-대-다 전송을 지원한다. 이를 위해서는 멀티캐스트 호스트 그룹의 정의가 있어야 하며 해당 세션에 대한 그룹의 가입/탈퇴 기능을 수행하는 그룹관리 메커니즘이 필요하다. 멀티캐스트는 IGMP(Internet Group Management Protocol)를 이용하여 그룹관리를 하는데 IGMP 프로토콜은 그룹에 대한 가입/탈퇴 처리를 담당한다.

IGMP는 멀티캐스트 라우터가 해당 서브넷에 존재하는 호스트 그룹 멤버의 존재를 인식하기 위해 기본적으로 query와 report 메시지를 사용한다. query 메시지는 IGMP 프로토콜이 해당 그룹에 가입할 호스트가 있는지 확인하거나 현재 그룹의 멤버를 확인하기 위해 서브넷에 주기적으로 전송된다. IGMP가 해당 서브넷에 query 메시지를 보내면 해당 멀티캐스트 그룹에 가입을 원하는 호스트는 report 메시지를 보내어 그룹에 가입할 수 있다. 일정시간 동안 query 메시지에 대한 응답을 하지 않으면 그룹의 탈퇴가 이루어진다[8].

멀티캐스트에서 신뢰성 제공을 위한 다양한 방법이 제안되어 왔는데 본 논문의 제안을 위해 고려한 대표적인 RMT(Reliable Multicast Transport) 기법은 다음과 같다.

2.2.1 Tree-based ACKs

멀티캐스트 사용자의 논리적 트리를 구성하고 트리 구조를 이용하여 오류를 복구하는 기법이다. 트리를 통해 각 수신자들은 부모-자식(parent-children) 관계를 형성하여 각 부모 노드들은 자식 노드에 대한 오류 복구 기능을 제공한다. 부모 노드가 재전송을 요구하는 패킷을 가지고 있지 않은 경우에는, 상위 부모 노드에게 재전송을 요청하여 결국 데이터 송신자에게 재전송 요구가 전달될 수 있다.

즉, 그룹의 대표 노드인 DR(Designated Receiver)은 각 자식 노드로부터 ACK 패킷 및 손실 정보를

받은 뒤 오류를 복구하고, 자기 서브 그룹에 대한 데이터 송수신 상태 정보를 상위 부모 노드에게 전달한다. 이러한 상태 정보에 입각하여 송신자는 흐름 및 혼잡제어를 수행하게 된다. 그러나, 각 수신자의 정보를 송신자에게 보낼 필요가 있는 경우 해당 제어 패킷이 트리를 따라 송신자에게 전달되는 동안 일정 시간의 지연이 발생하는 문제점이 있다. 대표적인 프로토콜로는 RMTP(Reliable Multicast Transport Protocol)[9]가 있다.

2.2.2 Timer-based NACKs

오류 복구 기능을 송신자가 아닌 가까이 있는 수신자의 도움으로 해결한다. 오류 재전송을 요구하는 수신자가 NACK 패킷을 전체 그룹에 전송하면 가까이에 성공적인 수신자가 있을 경우 그 수신자가 NACK 패킷에 응답한다. 이 경우 같은 NACK 패킷을 여러 수신자들이 동시에 발생시켜 NACK 패킷이 폭주할 우려가 있으므로, 각 수신자는 타이머(timer)를 이용하여 적절한 시간 동안 다른 NACK 패킷이 이미 발생되었는지를 파악한다. 이 방법은 NACK 억제제를 위해 사용되는 타이머의 동작이 전체 성능에 영향을 주며 모든 수신자에게 멀티캐스트 송신 능력을 요구한다. 대표적인 프로토콜로는 SRM(Scalable Reliable Multicast)[10]이 있다.

2.2.3 FEC(Forward Error Correction)

ACK/NACK 기반의 오류 복구에서는 수신자의 패킷 수신 정보를 송신자에게 전달하여 송신자가 패킷을 재전송 하는데 반해 FEC는 송신자가 데이터 송신 단계에서부터 패리티 비트(parity bits) 등의 부가 정보를 추가하여 전송하는 방식이다. 즉, 패킷 손실이 발생했을 경우 수신자로부터의 피드백 없이 다른 패킷의 부가 정보를 이용하여 손실된 패킷을 복구하는 것이다. 재전송 및 오류 복구 등으로 인한 추가 지연시간이 소요되지 않지만 별도의 FEC 코딩(coding)이 요구되며 불필요한 오버헤드가 발생할 수 있다. 대표적인 프로토콜로는 RMDP(Reliable Multicast Dissemination Protocol)[11]가 있다.

2.3 MDC : Multiple Description Coding, 다중 코딩

MDC는 하나의 프레임을 여러 개의 패킷으로 분

할하고 각 패킷에는 자신의 정보와 다른 패킷에 관한 부가 정보를 추가하여 전송도중 어느 하나의 패킷을 잃더라도 정상적으로 수신된 패킷으로 복구할 수 있고 모두를 수신하게 되면 송신 시와 똑같은 품질의 콘텐츠를 생성할 수 있는 코딩 기술이다. 그림 4에서 보는 것과 같이 하나의 스트림을 Even 샘플과 Odd 샘플로 양자화 하여 스트림 1에는 고화질의 Even 샘플과 부가적으로 저화질의 인접한 Odd 샘플과 Even 샘플의 차이를 추가하고 스트림 2는 그 반대로 생성하는 것이다. 이 경우 원래 스트림에 비해 25% 정도의 부가 정보가 추가되지만[12] 만약 스트림 1에서 손실이 일어난 경우 스트림 2의 정보를 가지고 고화질의 Odd 샘플과 저화질의 Even 샘플을 재생할 수 있으며 스트림 1, 2가 모두 수신되면 고화질의 Even 샘플과 고화질의 Odd 샘플을 재생함으로써 높은 품질의 결과를 얻을 수 있다.

MDC 방식은 패킷 손실 발생 시에 재요청을 통해 패킷을 회복하지 않고 손실이 발생하지 않은 스트림의 패킷을 이용하여 손실에 대한 에러 회복을 하기 때문에 에러 회복을 위한 제어 패킷이 발생하지 않으므로 대역폭의 절약과 전송 지연을 줄일 수 있다. 따라서 수신 노드의 수가 증가하여도 재전송에 의한 에러 회복 절차를 수행하지 않기 때문에 많은 수의 수신 노드가 참여하는 환경에서 기존 방법보다 네트워크 자원 사용을 효율적으로 사용할 수 있다.

기존의 Layered Coding 방식은 패킷을 하나의 기본 레이어와 여러 개의 보조 레이어로 나누고 사용자의 환경에 따라 레이어를 선택하도록 하는 코딩 기술으로써 일반적으로 기본 레이어에는 기본 품질을 보장할 수 있는 중요한 멀티미디어 데이터를 포함하고 보조 레이어에는 높은 품질의 서비스를 위한 상태정보 데이터를 포함한다. 그러나, 인터넷은 기본적으로

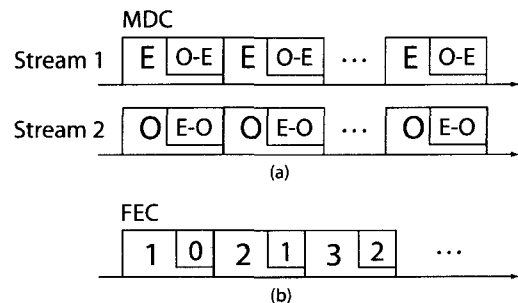


그림 4. MDC와 FEC

로 QoS를 보장하지 못하기 때문에 각 레이어에 대한 우선 순위를 부여할 수 없으므로 기본 레이어를 손실하게 되면 보조 레이어는 의미가 없어지는 문제가 있다. 그리고, FEC 방식은 네트워크 지연에 의해 연속적인 패킷의 손실이 발생할 경우에는 그 부분에 대한 복구는 할 수가 없는 단점이 있다.

3. 제안하는 콘텐츠 전송 모델

CDN 네트워크는 대용량의 멀티미디어 콘텐츠 전송에 있어 지연을 줄이고 웹 서버로의 로드 집중 문제를 해결하는 하나의 방법으로 제시되었다. 멀티미디어 데이터는 지연에 매우 민감하므로 최단 경로를 찾는 것이 중요하고 데이터 볼륨이 크기 때문에 높은 대역폭을 요구하며 재전송을 최소화해야 하는 특성을 갖고 있기 때문에[13] 이러한 특성을 고려한 새로운 전송 모델이 필요하다.

본 논문에서는 다음과 같은 네트워크 모델을 가정한다.

① CDN 서버들은 지역적으로 트리 기반의 멀티캐스트 그룹을 형성하며, 설정된 멀티캐스트 트리는 거의 변동이 없으며, IGMP 프로토콜에 의해 그룹관리를 한다.

② CDN 기술은 미리 CDN 서버들에게 콘텐츠의 복사를 분배해 놓는 것이므로 서버들 간의 전송에 있어서는 지연에 크게 민감하지 않다. 미들 마일에서는 트래픽을 줄여 네트워크의 효율을 높이는 것이 더 중요하므로 서버 간 콘텐츠 전송은 SRM 프로토콜에 기반 하여 CP 오리진 서버에서 CDN 서버 그룹으로 멀티캐스트 전송을 하며 전송 중 오류는 NACK를 사용하여 제어한다.

③ 모든 CDN 서버들은 콘텐츠의 동기화를 유지하고 있다.

이러한 가정 하에 그림 5와 같은 콘텐츠 전송 모델을 제안한다.

- ① 사용자는 CP 서버에 콘텐츠를 요청한다.
- ② CP 서버는 IP 주소에 기반 하여 사용자와 최단 거리의 Edge 서버 k 개를 선정하여 위치 정보를 전송한다. (k 는 2개 이상의 다중 서버를 의미하며 본 연구에서는 2로 설정한다.)
- ③ 사용자와 최단 거리의 Edge 서버 두 곳에 콘텐

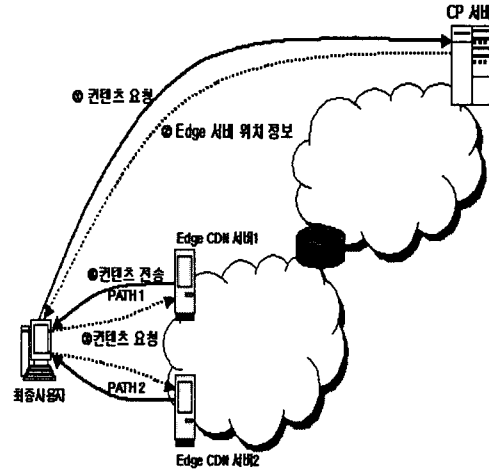


그림 5. 제안하는 콘텐츠 전송 모델

츠를 요청한다.

④ 각 Edge 서버에서는 MDC 방식으로 코딩된 스트림을 각각 독립된 경로로 전송하면 사용자는 2개의 Edge 서버로부터 받은 중복된 스트림은 폐기하고 나머지 스트림을 디코딩하여 재생한다.

이러한 모델은 CP 서버에서 Edge CDN 서버까지 멀티캐스트에 의해 콘텐츠를 미리 전송해 놓음으로써 전송효율과 확장성을 향상시키고 다중 코딩(Multiple Description Coding)을 이용하여 전송 중 에러를 재전송하지 않고 부가 정보를 이용하여 복구함으로써 전송 성능을 향상시키고 또한, 하나의 경로에서 지연이 발생할 경우 모든 스트림이 손실될 수 있으므로 2개의 Edge 서버에서 다중 경로로 스트림을 전송받음으로 해서 신뢰성을 향상시킬 수 있다.

다중 코딩은 기존의 MPEG와 같은 단일 코딩(SDC: Single Description Coding)과는 달리 미디어 스트림을 여러 개의 스트림으로 분할하여 코딩하고 수신된 코딩 스트림은 기본 데이터(baseline quality video)가 되어 디코딩 하는데 사용될 수 있으며 각 스트림은 서로 독립적인 특성을 갖고 있다. 사용자가 미디어 스트림을 요청하면 사용자와 가까이 있는 2개의 Edge 서버로부터 다른 경로를 통해 스트림을 수신하게 됨으로 이러한 구조는 CDN과 관련하여 사용자에게 응답 시간을 줄이고 CP 서버의 부하를 감소시키고 네트워크나 서버의 다운에 강하고 사용자의 수에 확장성을 제공하는 장점이 있다.

기존의 FEC, Layered Coding, MPEG와 같은 단일 코딩 방식은 지연으로 인한 패킷 손실이 잦은 네

트위크 환경에서는 안정적인 재생을 보장하지 못하며 손실된 스트림을 효율적으로 재구성할 수 없는 단점이 있지만 다중 코딩은 각 코딩된 스트림이 서로 독립적이며 스트림 내에 부분적으로 손실이 있더라도 나머지 부분으로 복구가 가능하며 더 추가적으로 스트림이 수신되면 미디어의 질을 더욱 향상시킬 수 있는 장점이 있다. 또한, 2개의 Edge 서버를 선택함으로써 모든 경로가 끊기거나 모든 경로에서 데이터 손실이 일어날 확률을 감소시켜 신뢰성을 향상시켰다.

4. 분석 및 평가

4.1 분석환경

본 절에서는 제안한 시스템의 우수함을 객관적으로 입증하기 위해 한 개의 ES(Edge Server)에서 SDC(단일 코딩)을 사용하여 전송하는 방식과 제안한 방식을 비교 분석한다.

분석을 위한 임의의 네트워크 N은 2개의 Edge 서버(ES)와 n개의 수신 노드로 구성되며, 수신 노드의 집합을 $R_G = \{ R_1, R_2, \dots, R_n \}$, ($0 < i < n$)라 정의한다.

분석환경의 토폴로지는 그림 6과 같으며, R1과 수신 노드간의 링크에서 평균 전송 데이터 크기, 지연에 대해 비교 분석하였다. 제안한 시스템이 두 개의 ES를 기준으로 동작하고 서로 다른 경로를 사용하지만, 같은 경로를 사용하는 부분에서 에러가 발생하는 경우에는 데이터를 수신하지 못하기 때문이다.

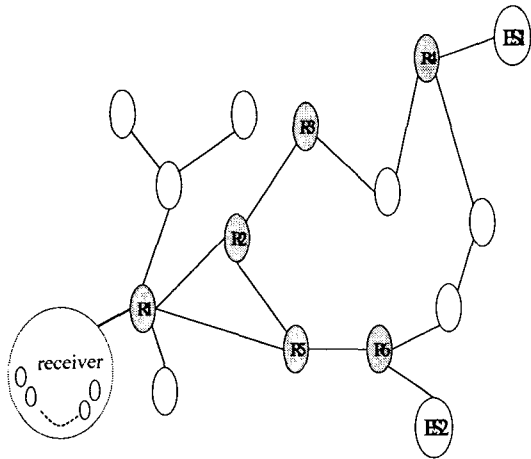


그림 6. 성능분석 토폴로지 환경

한편, CP 서버에 콘텐츠 요청 지연 시간을 C_{rc} , ES 서버에 콘텐츠 요청 지연 시간을 C_{re} 로 정의하였으며 에러가 발생하지 않는다고 가정하였다. 수신해야 할 콘텐츠의 크기는 패킷 단위이며 콘텐츠 전송에 소요되는 시간을 C_{ct} 로 정의하였다. 또한 공유 링크에서의 패킷 손실 확률은 P_{sle} 이며, ES 서버에서의 에러 발생 확률을 P_{se} 로 정의하였다. 시뮬레이션은 VC++을 이용하였으며, 링크에 패킷의 도착은 기존 방식과 제안하는 방식 모두 $e^{-\lambda t}$ 이며, 분석에 사용된 변수와 의미는 표 1과 같다.

표 1. 실험변수

변수	의 미
D_l	링크의 지연시간
C_{rc}	CP서버에 콘텐츠요청 소요시간
C_{re}	ES서버에 콘텐츠 요청 소요시간
C_{ct}	콘텐츠 전송에 소요되는 시간
F_{rt}	ES서버 에러 회복 지연 시간
α	C_{re} 의 평균 소요시간 비례상수
P_{le}	링크에서 에러가 발생할 확률
P_{sle}	공유 링크에서 에러가 발생할 확률
P_{se}	ES 서버에서 에러가 발생할 확률

기존 방식의 평균 전송 시간(D_{ST})은 CP, ES에 콘텐츠 요청 소요 시간, 패킷 에러에 따른 지연, 서버 에러에 따른 지연으로 구성되며 수식 (1)과 같다. 제안한 방식의 평균 전송시간(D_{MT})은 $n(=2)$ 개의 ES에 콘텐츠 전송 요청에 따른 지연 시간을 고려하여 αC_{re} 가 된다. $\alpha(=1.2)$ 는 한 개의 ES 사용을 기준으로 했을 때 상대적 비례 지연 상수이다. 또한 ES1과 R1사이의 공유 링크에서 에러가 발생한 경우, ES1에서 R2, ES2에서 R2까지의 경로에서 동시에 에러가 발생할 확률은 $P(P_{le}^2 | P_{sle})$ 가 된다. 따라서 평균 전송 시간은 수식 (2)와 같이 표현할 수 있다.

$$(1) D_{ST} = C_{rc} + C_{re} + C_{ct}(1 + P_{le}) + F_{rt}P_{se}$$

$$(2) D_{MT} = C_{rc} + \alpha C_{re} + C_{ct}(1 + P(P_{le}^2 | P_{sle})) + F_{rt}P_{se}^2$$

한편, m개의 수신 노드에서 데이터 수신 시 발생하는 평균 데이터의 크기는 단일 스트림을 사용하는 경우 수식 (3)과 같으며, 제안하는 방식의 경우에는 수식 (4)와 같이 표현될 수 있다.

$$(3) Error_{S_{ms}} = \sum_{i=1}^m \left\{ \binom{m}{i} (1 - P_{sle})^{m-i} (P_{sle})^i i \right\}$$

$$(4) Error_{M_{ms}} = \sum_{i=1}^m \left\{ \binom{m}{i} (1 - P_{sle})^{m-i} (P_{sle})^i \frac{k}{2kC_2} \right\}$$

그림 7은 수신 노드의 증가에 따른 평균 에러 발생 확률에 대한 결과이다. CP에 콘텐츠 요청시간(C_{rc})을 1ms, ES에 콘텐츠 요청 시간(C_{re})을 1ms, 서버의 에러 확률(P_{se})을 0.001로 가정하여 비교 분석하였으며, 그림7-a는 P_{sle} 의 값이 0.01, 그림7-b는 P_{sle} 의 값이 0.001일 때의 결과이다.

수신 노드의 크기가 증가할수록 평균 에러 발생 확률이 높아지는 것을 확인할 수 있으며, 제안한 방식이 더 낮은 값을 갖는다. 이는 공유 링크에서 에러가 발생한 경우에도 동일한 패킷을 모두 손실할 확률이 작기 때문이다.

그림 8은 평균 트래픽 발생량을 비교한 것이다. 실험변수 값은 그림 7-a와 같으며 패킷의 크기는 1KB로 가정하였다.

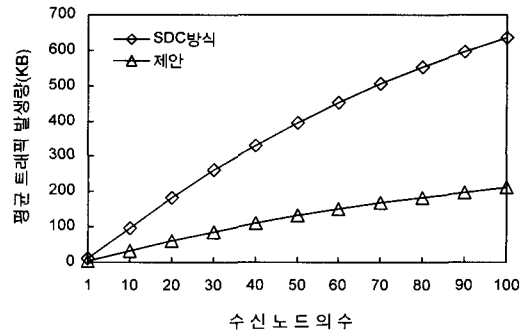


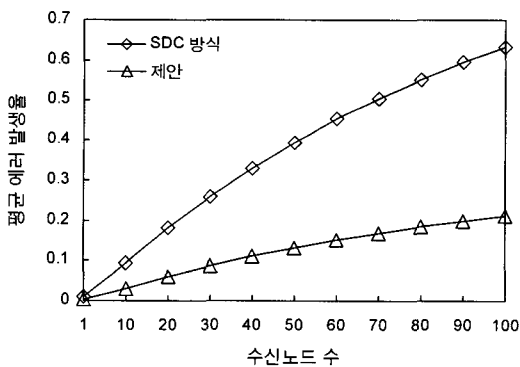
그림 8. ES와 수신 노드간의 평균 트래픽 발생량

평균 트래픽 발생량은 수신 노드에 비례하지만 제안하는 방식의 경우 기존 방법보다 60% 정도 낮은 결과를 보인다. 제안하는 방식은 두 개의 경로를 통해 데이터를 전송하며 동일한 패킷에 대해 에러가 발생한 경우에만 재전송이 이루어지기 때문이다.

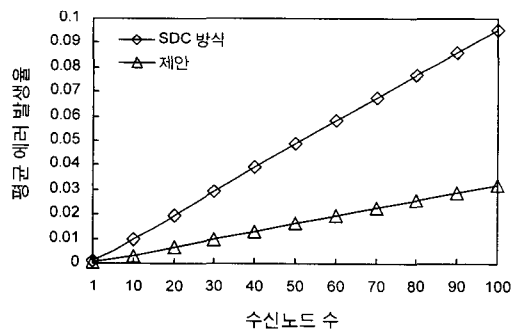
그림 9는 RTT(Rout Trip Time)이 10ms, 링크 에러(P_{sle})가 0.01일때, 수신 노드 수에 따른 평균 전송 지연 시간을 나타낸다.

제안하는 방식은 수신 노드가 증가하여도 2ms 정도의 평균 지연 시간을 유지하는 것을 확인할 수 있다. 반면, 기존 방법은 수신 노드의 증가에 따라 평균 지연 시간이 지속적으로 증가한다. 제안하는 방법은 다른 경로를 통해 전달되는 두 개의 스트림 중 최소한 개의 패킷을 수신하면 재생이 가능하기 때문이다.

한편, R1과 수신 노드간의 링크에서 존재하는 패킷이 k 개인 경우 평균 트래픽 발생량은 그림 10과 같다. 공유 링크에서 에러 발생 시 패킷의 수가 많을수록 평균 트래픽 발생량은 감소하는 것을 확인할 수 있다. 이는 공유 링크에 의해 제안한 방법의 효율이 감소하지 않기 때문이다.



(a) 평균 에러 발생 확률($P_{sle}=0.01$)



(b) 평균 에러 발생 확률($P_{sle}=0.001$)

그림 7. 링크 에러에 따른 평균 에러 발생 확률

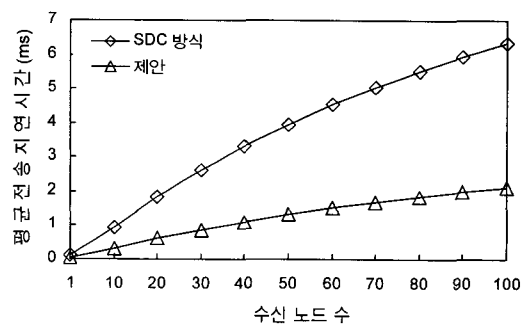


그림 9. 평균 전송 지연 시간

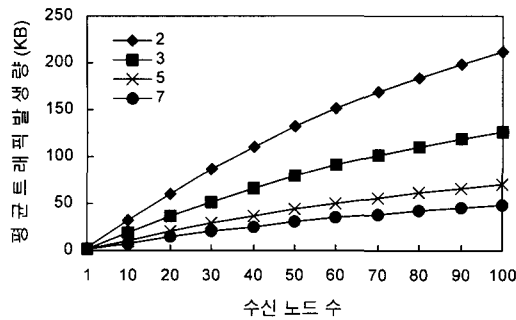


그림 10. 패킷 수에 따른 ES와 수신 노드간의 평균 트래픽 발생량

5. 결 론

CDN 네트워크는 콘텐츠를 다수의 지역 서버에 분산시켜 인터넷상에 발생하는 순간적인 사용자 집중으로 인한 서버의 부하를 줄이고 대용량 콘텐츠의 전송 지연과 손실을 줄여 서비스의 질을 높일 수 있는 개선된 인프라이며 본 연구는 CDN이라는 새로운 개념의 통신망에서 대용량의 멀티미디어 데이터를 효율적으로 전송할 수 있는 방안을 제안하였다.

사용자와 가까운 2개의 단말 서버(Edge Server)를 선택하여 2개의 경로에 의해 데이터를 전송함과 동시에 다중 코딩(Multiple Description Coding)를 이용하여 미디어 스트림을 여러 개의 스트림으로 분할하여 코딩하고 이 코딩의 복사를 단말 서버에서 사용자에게로 전송하게 하여 에러를 감소시켰다. 분석 결과, 제안한 전송 방식은 하나의 서버에서 단일 코딩 방식을 사용한 전송에 비해 평균 에러 발생 확률이 더 낮으며 평균 전송 데이터의 크기도 60% 정도 낮은 결과를 보였다.

그러나, 본 논문에서는 단말 서버 선택 시 최단 거리의 단말 서버를 선택하도록 하였는데 이것은 특정 서버에 로드가 집중될 수 있으며 사용자 입장에서는 다중 서버에서 동일 스트림을 받아야 하므로 자원의 낭비가 있을 수 있고 중복된 스트림을 계속 체크하여 버려야 하는 추가 부담의 문제가 있다. 그러므로, 향후 최단 거리뿐 아니라 단말 서버의 로드, 네트워크 트래픽 등 여러 가지 변수를 고려한 최적의 단말 서버 선택을 위한 연구를 진행할 예정이며 이러한 연구는 멀티미디어 콘텐츠의 안정적이고 신속한 전송에 기여할 것이다.

현재 인터넷상에서 멀티미디어 콘텐츠의 급증을

고려해 볼 때, CDN은 차세대 인프라로 확장될 것이며 서로 다른 관리하에 있는 CDN 네트워크간의 콘텐츠 공유 및 분배 정책, 실시간 서비스의 처리, 보안 메커니즘 등 여러 측면의 연구가 더욱 필요할 것이다.

참 고 문 헌

- [1] Stardust.com, "The Ins and Outs of Content Delivery Networks", White Paper, Stardust.com, Dec. 2000.
- [2] Cisco Systems, "Content-Delivery Networks for Host Services.", White Paper, Cisco Systems, 2000.
- [3] G. Peng, "CDN: Content Distribution Network", in *Stony Brook University Tech Reports*, TR-125, Jan. 2003.
- [4] <http://www.ietf.org/html.charters/cdi-charter.html>
- [5] A.Biliris, C. Cranor, F. Douglass, M. Rabinovich, S. Sibal, O.Spatscheck, W. Sturm. "CDN Brokering" *Proceedings of WCW'01*, Jun. 2001.
- [6] J. Kangasharju, J. Roberts, K. W. Ross, "Object Replication Strategies in Content Distribution Networks", *Proceedings of WCW'01*, Jun. 2001.
- [7] <http://www.akamai.com>
- [8] W. Fenner, "Internet Group Management Protocol, Version 2", IETF RFC 2236, Nov. 1997.
- [9] B. Whetton, J. Conlan. "A Rate Based Congestion Control Scheme for Reliable Multicast", Technical White Paper, Oct. 1998.
- [10] M. Kadansky, D. Chiu., "Tree-based Reliable Multicast", IETF Internet Draft, Jan. 2000.
- [11] R. Yavatkar, G. James, S. Madhu "A Reliable Dissemination Protocol for Interactive Collaborative Applications", *Proceeding of ACM Multimedia*, Nov. 1995.
- [12] J.G. Apostolopoulos, "Error-resilient video compression via multiple state streams," *Proceeding of International Workshop on*

Very Low Bitrate Video Coding(VLBV'99), Oct. 1999.

- [13] J.C. Pasquale, G.C. Polyzos, and G. Xylomenos, "The Multimedia Multicasting Problem," *Multimedia Systems*, vol.6, no.1, pp. 43-59, Jan. 1998.



김 선 호

1987년 이화여자대학교 수학교육전공 학사
 1992년 이화여자대학교 교육대학원 전자계산교육전공 석사
 1987년~1989년 대우전자부품(주) 전산실

1990년~1993년 한국생산성본부 정보화사업부
 1999년~현재 송실대학교 컴퓨터학과 박사과정
 1998년~현재 동덕여자대학교 정보과학대학 강의전임교수

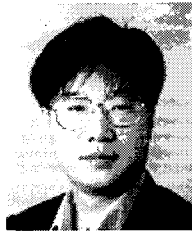
관심분야 : 멀티캐스트, Mobile IP, CDN, DRM
 E-mail : shkim98@dongduk.ac.kr



김 기 영

1996년 상지대학교 전자계산학과 학사
 1995년~1997년 삼보정보통신기술연구소
 1999년 송실대학교 컴퓨터학과 공학석사
 2003년 송실대학교 공학박사

관심분야 : 멀티캐스트, 실시간 프로토콜, Mobile-IP, 정보보호
 E-mail : ganet89@kingdom.ssu.ac.kr



신 용 태

1985년 한양대학교 산업공학과 학사
 1990년 Univ. of Iowa 컴퓨터학과 석사
 1994년 Univ. of Iowa 컴퓨터학과 박사
 1994년~1995년 Michigan State

Univ. 전산학과 객원교수
 1995년 3월~현재 송실대학교 컴퓨터학부 부교수
 관심분야 : 멀티캐스트, 그룹통신, 인터넷 보안, 이동 인터넷 통신
 E-mail : shin@comp.ssu.ac.kr

교 신 저 자

김 선 호 156-743 서울 동작구 상도동 송실대학교 컴퓨터학과 통신연구실