

P2P 기반 CDN 시스템에서 네트워크 토폴로지와 전송 성능과의 관계 분석

황희선[○], 김태형
한양대학교 전자컴퓨터공학부
{hshwang[○], tkim}@cse.hanyang.ac.kr

Relationship Analysis between Network Topology and Delivering Performance in P2P based Content Delivery Network

Hee-Seon Hwang[○], Tae-Hyung Kim
Dept. of Computer Science & Engineering, Hanyang University

요 약

CDN(Content Delivery Network)은 기존의 중앙 집중형 구조를 갖는 multimedia streaming 서비스의 서버 과부하 문제와 서버 주위의 네트워크 전송 속도 급감 문제를 해결 하기 위해 서버와 클라이언트 사이에 콘텐츠 캐시 역할을 하는 edge server를 둬으로써 원래 서버로의 요청을 분산 처리 하는 콘텐츠 전송 기술이다. 그러나 여전히 대규모 전송이 필요할 경우 원하는 수준만큼의 scalability를 보장할 수 없다. 이에 대한 근본적인 해결방법으로 본 논문에서는 P2P 기반의 CDN 시스템 구축을 위한 첫 단계로서 참여 노드간의 네트워크 토폴로지와 전송 성능간의 관계를 분석해보고자 한다. 이를 위해 다양한 구조의 전송 성능을 평가할 수 있는 환경을 구축하였으며 이를 이용하여 여러 구조에서의 전송 성능 측정 결과를 제시하여 최적의 P2P 전송 네트워크를 결정할 수 있도록 한다.

1. 서론

기본적인 멀티미디어 스트리밍 서비스는 중앙에 서비스를 제공 하는 서버가 있고 클라이언트들이 서비스를 요청을 하여 서버가 전송 서비스를 제공하는 중앙 집중형 구조를 갖는다. 이 구조는 클라이언트 수가 급증할 때 전송요청을 처리하기 위해 서버에 과부하 문제가 발생할 수 있으며 서버 주위의 네트워크 대역폭 수요가 커져서 전송속도가 급감하는 문제를 야기할 수 있다.

이러한 문제를 해결 하기 위한 기술로 CDN 구축에 대한 연구가 최근 활발히 진행되고 있다 [1]. CDN은 원래의 콘텐츠 서버에 대한 요청을 분산 처리할 수 있는 edge server를 사용자와 인접하여 위치시키고 콘텐츠에 대한 캐시 서비스를 제공함으로써 전송 성능 향상을 도모할 수 있다는 장점이 있다.

CDN은 두 가지 핵심 모듈로 구성 된다. 하나는 클라이언트의 요청을 받아 콘텐츠를 서비스 하는 캐시 서버이고, 나머지 하나는 콘텐츠를 여러 캐시 서버에 전송 하고 시스템 동작 상태를 모니터링 할 수 있게 하는 콘텐츠 전송 및 관리 모듈이다. 이러한 구조는 서버와 클라이언트 사이에 캐시 서버를 둬으로써 server의 과부하와 대역폭 부족 문제는 해결해 주지만, 여전히 많은 클라이언트에게 서비스 해주기 위한 서버 확장 작업이 scalable 하지 않으므로 운영상 문제점이 있다.

이 문제를 해결하기 위해 최근 많은 주목을 받고 있는 P2P(Peer-to-Peer) 컴퓨팅 기술을 살펴 볼 필요가 있다. P2P 기반 시스템은 근본적으로 scalable 한 솔루션을 제공하기 때문이다 [2]. 일반적으로 P2P 시스템은 어떤 클라이언트의 요청을 처리 하는 서버가 고정되어 있지 않으므로 완전한 분산 처리 시스템이다. 따라서, 콘텐츠 전송 성능을 향상시키고 scalable 한 솔루션을 제공하기 위해서는 P2P 기술이 CDN 시스템 구축에 필수적

로 적용되어야 할 것으로 생각되며 이를 위해 본 논문에서는 완전히 분산된 P2P 네트워크를 구성하는 데 있어서 네트워크 토폴로지가 전송 성능에 미치는 영향을 분석해 본다.

P2P 기반 CDN 시스템에서의 콘텐츠 Server는 클라이언트와 그들의 보유 파일간의 매핑 테이블을 갖고 있다가 클라이언트의 요청을 그 콘텐츠를 가진 다른 클라이언트에게 라우팅시켜 서비스 받게 함으로서 서버를 확장 하지 않고도 많은 클라이언트에게 서비스 할 수 있다. 이 방식은 기존 CDN에 비해 두 가지 장점을 갖는다.

첫째는 클라이언트가 캐시서버가 아닌 인접 서버로부터 콘텐츠를 전송 받기 때문에 콘텐츠 제공자들이 서버 확장성 문제에서 해방 될 수 있다. 둘째로 캐시 서버보다 더 인접한 클라이언트로부터 콘텐츠를 제공 받기 때문에 더욱 고품질의 콘텐츠 서비스가 가능하다.

CDN 서버가 클라이언트의 요청을 다른 클라이언트로 라우팅 시키는 과정에서 CDN 서버와 클라이언트들 사이에 일정하지 않은 네트워크 토폴로지가 형성 된다. 이 구조 상에서는 클라이언트는 서버 데이터를 다른 클라이언트 노드들을 거쳐서 받아야 하므로 서버에서 직접 데이터를 받는 것보다 전송 지연이 생길 수 있다. 이 전송 지연을 최소화 하기 위해서 서버와 클라이언트 노드 사이에 만들어 지는 토폴로지를 최적화 하는 것은 중요한 문제이다.

본 논문에서는 여러 토폴로지를 CDN에 적용 시켜 봄으로써 보다 효율적인 전송 능력을 갖는 네트워크 토폴로지를 형성 하는 방법을 소개 한다. 이를 위해 여러 토폴로지를 구성 할 수 있고, 각 토폴로지 상에서 파일 전송 시간을 체크하는 소프트웨어를 구현 하여 성능 평가를 다양 하게 수행 할 수 있는 환경을 구축 하였다. 이 환경을 사용 해서 각각의 토폴로지를 갖는 P2P 기반 컨

텐츠 전송 시스템들의 성능을 토폴로지의 관점에서 비교, 분석하여 multimedia streaming 서비스를 위한 최적의 토폴로지를 형성 하는 방법을 제시 하는 것이 본 논문의 목적이다.

2. 성능 평가 시스템 환경

성능 평가를 위한 시스템 환경은 마스터 서버와 일반 노드들로 구성 된다. 아래 그림1에서 볼 수 있듯이 마스터 서버는 사용자가 원하는 토폴로지를 입력으로 받아 각 노드에 적절한 제어신호를 전달함으로써 토폴로지를 구성하도록 한다.

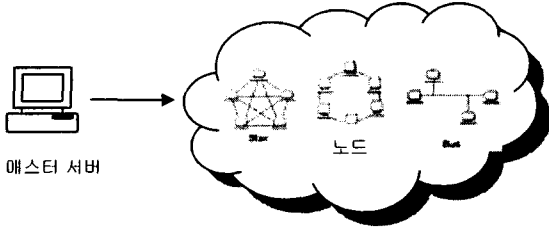


그림1. 평가 시스템 구조

2.1 마스터 서버

하나의 네트워크 구조에 따른 전송 성능 평가를 주관하는 마스터 서버는 다음의 세 가지의 기능을 갖는다.

첫째는, 프로그램 초기화 기능이다. 현재 접속중인 모든 클라이언트와 연결을 끊고 할당 받았던 소켓과 메모리를 해제하고 프로그램을 초기화하여 토폴로지 구성과 전송시간 측정을 준비한다. 그 다음은 토폴로지 생성 기능으로서, 각 클라이언트들이 서버에 접속 하면 그 클라이언트들이 개방하고 있는 서버의 주소와 포트 번호를 질의하여 테이블에 저장 한다. 사용자로부터 노드들간의 연결 리스트 받아 파싱한 후 각각의 노드들에게 그 노드가 연결 해야 하는 하위 노드들의 주소와 포트 번호를 보내어 하위 노드에 연결 하도록 한다. 예를 들어, 사용자가 그림 2와 같은 토폴로지를 원한다면 1,2,2,5,2,2,3,4,5,2,6,7과 같은 형태로 입력한다. 이것은 1번 노드에 2개의 하위노드2번과 5번이 있고, 2번 노드에 2개의 하위노드 3번과 4번이 있고, 5번 노드에 2개의 하위노드 6, 7번이 있다는 의미이다. 마지막으로 생성된 네트워크 구조대로 연결된 각 노드들에게 실제 컨텐츠 파일을 모두 전달하는데 걸리는 시간을 측정하는 것까지 마스터 서버가 담당한다.

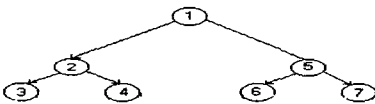


그림 2. 토폴로지에

2.2 토폴로지를 구성 하고 있는 노드

상위 노드에게는 클라이언트 역할을 하고 하위 노드들에게는

서버 역할을 한다. 마스터 서버에 접속해서 자신이 개방하고 있는 서버의 포트번호를 알려 줌으로서 토폴로지의 구성 노드로서 등록 된다. 이들 역시 다음의 세가지 동작을 수행한다. 첫째, 마스터 서버와 연결을 끊고 할당된 메모리와 자원을 해제하며 프로그램을 초기화하여 다음 명령을 기다린다. 그리고, 토폴로지를 구성하고 있는 노드들 사이에 존재하는 파일의 검색을 수행한다. 검색된 결과는 사용자에게 전달되며 사용자가 원하는 컨텐츠에 대한 전송을 명령하면 해당 컨텐츠가 참여하고 있는 모든 노드에 전달될 수 있도록 하며 이에 따른 각 노드별 완료시간을 마스터 서버에 보고함으로써 전체 전송시간을 측정할 수 있도록 한다.

검색 방식과 파일 다운로드 방식은 누텔라 방식을 따른다. 검색 패킷들의 범람을 막기 위해 TTL항목이 패킷에 포함 되고, 중복 노드 처리를 막기 위해 각 쿼리 패킷과 쿼리 응답 패킷은 GUID라 하는 유일한 키를 갖게 된다.

이와 같은 동작 형태는 완전하게 분산된 연결을 형성하므로 본 논문에서 전송 성능과 토폴로지 간의 관계를 분석하려는 목적에 부합되는 형태이기 때문이다. 즉, Gnutella는 TTL에 따라 무작위로 네트워크를 구성하지만 최적화된 구조를 갖도록 유도할 수 있다면 전송성능의 향상을 도모할 수 있기 때문이다. 물론 Gnutella는 최소한의 제한조건만 유지하면서 무작위로 네트워크를 구성하므로 최적화된 구조를 형성하는 것이 가능하지 않다. 그러나 3절의 실험결과에 따르면 구조와 전송성능은 분명한 차이를 만들므로 향후 효율적인 P2P 네트워크를 구성하는 방법에 대한 연구가 필요할 것으로 생각된다.

3 실험 결과

Windows 2000/98 환경의 8대의 컴퓨터에서 실험하였다. 프로그램의 구현 언어는 Visual C++이며 한 대의 PC에 마스터 서버를 구동 시킨 후 나머지 7대에 각각의 노드 수만큼 분산 시켜서 구성노드 프로그램을 구동 시킨다. 원하는 토폴로지는 마스터 서버가 사용자로부터 입력 받아 구성되며 검색결과에서 사용자가 선택한 파일을 모든 노드에 전송한다. 아래 그림은 각각의 토폴로지 상에서 140 MB 파일을 전송 시키는데 걸리는 시간을 지표로 나타낸 것이다. 트리는 Root노드로부터 전송을 시작 하고 Mesh 형은 Mesh를 구성 하고 있는 중간 노드부터 전송을 시작 한다. 이 실험 결과를 통해 토폴로지 구성 가이드 라인을 제시 한다.

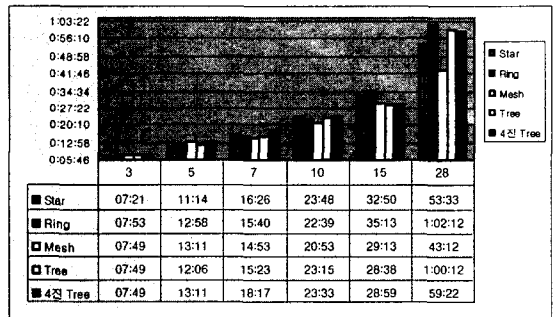


그림 3. 다양한 구조에서의 전송성능

우선, 연결 노드수가 증가함에 따라 전송 시간도 증가한다. 그러므로 특정 레벨을 구성 하는 노드들에 새로운 노드들을 연결해야 할 때 새로운 노드들을 그 레벨의 기존 노드들에 균형 있게 연결을 해야 한다. 예를 들어 N-level에 세 개의 노드 a, b, c 가 있고 각각의 노드가 2개의 노드에 연결을 가질 수 있는 경우 새로운 노드가 들어 왔을 때 a에 연결 되도록 배치 하고, 두 번째 노드가 들어 왔을 때 b 노드에 연결 되도록 배치 하고, 세 번째 노드가 들어 왔을 때 c 노드에 연결 되도록 배치 하고 그리고 네 번째 노드가 들어 오면 다시 a 노드에 배치 하도록 한다.

둘째, 버스 형과 버스를 제외한 토폴로지를 보면 연결 Hop수를 늘렸을 때 증가한 전송 시간이 직접 연결된 노드 수를 늘렸을 때 증가한 전송 시간보다 길다. 그러나 이진 트리와 4진 트리를 보면 Hop수는 2진 트리가 더 많은데 전송 시간은 2진 트리가 더 적게 걸리는 경우를 볼 수 있다. 즉, 연결 노드 수가 증가 할 때의 전송 시간 증가율과 Hop수가 증가할 때의 전송 시간 증가율을 비교해서 전자가 후자를 넘지 않는 선에서 연결 노드의 개수를 정해야 한다.

셋째, 트리형과 그물형의 결과를 보면 그물형이 전송 시간이 덜 걸린다. 이것은 트리형은 항상 파일을 주는 노드가 상위노드로 고정 되어 있어 전송 시간이 고정 되어 있지만, 그물형은 네 방향 중 먼저 오는 것에서 파일을 받을 수 있기 때문이다. 그러므로 트리형과 같이 전송 방향이 한 방향인 경우 한 노드에 연결되는 노드는 전송 시간이 비슷한 노드끼리 연결 되어야 한다는 말이다. 그렇지 않으면 전송 시간이 빠른 노드가 있어도 전체 전송 시간을 줄이진 못한다.

아래 표는 동일한 토폴로지에서 그 토폴로지의 구성 노드들 간의 전송 시간과 콘텐츠 서버와의 파일 전송 시간을 계산하여 최적의 노드 배치와 최악의 노드 배치 상황 하에서 전송 시간을 구해본 결과이다. 노드간의 전송 시간이 다르며 그 시간에 따라 토폴로지에 배치해야 함을 보여주는 예이다. 사용자의 토폴로지 입력은 1,2,2,3,2,2,4,5,3,2,6,7로 7개의 노드로 트리 모양의 토폴로지이다.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|------|-----|-----|-----|-----|-----|-----|-----|
| Root | 191 | 126 | 158 | 125 | 124 | 128 | 126 |
| 1 | | 191 | 222 | 174 | 182 | 197 | 205 |
| 2 | 191 | | 147 | 125 | 126 | 131 | 150 |
| 3 | 222 | 147 | | 124 | 147 | 131 | 129 |
| 4 | 174 | 125 | 124 | | 124 | 127 | 124 |
| 5 | 182 | 126 | 147 | 124 | | 145 | 125 |
| 6 | 197 | 131 | 131 | 127 | 145 | | 125 |
| 7 | 205 | 150 | 129 | 124 | 123 | 125 | |

표 1. 8개 노드간 전송에 필요한 시간

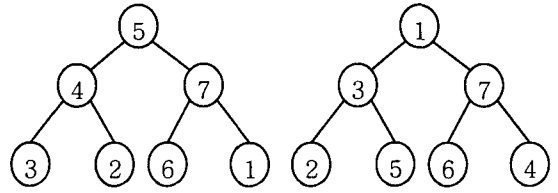


그림 4. 최적의 노드 배치
전송 시간 : 14분 53초

그림 5. 최악의 노드 배치
전송 시간 : 16분 23초

4 결론 및 향후 과제

본 논문에서는 P2P 기반의 CDN을 구축하기 위한 기초 연구로서 네트워크 토폴로지와 전송성능과의 관계를 분석하였다. 첫 번째 실험 결과 Mesh 구조가 가장 좋은 전송 성능을 보였고, Star와 4진 Tree 구조에서 상대적으로 낮은 전송 성능을 보였다. 두 번째 실험 결과 같은 토폴로지라 하더라도 노드들의 배치에 따라 성능에 차이가 난다는 것을 알 수 있었다.

향후 과제로는 형태가 정해지지 않은 토폴로지를 갖는 Gnutella 와 같은 P2P 기반 시스템에 특정 토폴로지를 적용하여 그 파생 효과를 검증 하는 연구가 필요하다.

5 참고 문헌

[1] 최승락, 양철웅, 이중식, " CDN의 핵심 구성 기술들과 경향," 한국정보과학회지 제20권 제9호, pp.5-11, 2002.9.
 [2] 김필우, " 차세대 인터넷 Peer to Peer" 한빛 미디어
 [3] <http://www.gnutella.com>
 [4] <http://www.web-biz.pe.kr/web/p2p1.html>
 [5] D. A. Tran, K. A. Hua and T. T Do., " ZIGZAG: An Efficient Peer-to-Peer Scheme for Media Streaming," IEEE INFOCOM2003, April 2003.
 [6] H. Deshpande, M. Bawa, H. Garcia-Molina, "Streaming Live Media over a Peer to Peer Network," Technical Report, Stanford University, August 2001.
 [7] 김화중, " 컴퓨터 네트워크 프로그래밍" 홍릉출판사, pp.43-92