

오버레이 네트워크 상에서의 콘텐츠 중심의 라우팅

이문영⁰, 고동일, 조기덕, 권태경, 최양희

서울대학교 전기컴퓨터 공학부

{mylee, diko, kdcho, tk, yhchoi}@mmlab.snu.ac.kr

A Content-Centric Routing on the Overlay Networks

Munyoung Lee⁰, Dong-il Diko Ko, Kideok Cho, Taekyoung Kwon, Yanghee Choi

School of Computer Science and Engineering, Seoul National University

요 약

패킷 네트워크 이론과 TCP/IP를 기반으로 탄생한 인터넷은 월드 와이드 웹 (WWW)의 등장으로 인하여 엄청난 성장을 이룩하였으며, 가장 중요한 통신 인프라 역할을 수행하고 있다. 그러나 현재 인터넷의 사용 패턴은 일대일 통신이 아닌 대용량 콘텐츠 검색과 획득에 집중되고 있는 실정이다. 종단간 패러다임을 기반으로 하고 있는 현재 인터넷은 이러한 인터넷 사용패턴을 효율적으로 지원할 수 없다는 단점이 있다. 따라서 본 논문에서는 현재 인터넷의 사용패턴을 잘 지원할 수 있는 새로운 네트워크인 콘텐츠 중심 네트워크에서의 라우팅 기법인 Content-Centric Routing (CCR)을 제안한다. CCR은 Reverse Path Forwarding (RFP)과 Store & Forward 방식을 채용함으로써, 콘텐츠 중심 라우팅 기법 중 가장 대표적인 기법인 Combined Broadcast and Content Based (CBCB) 라우팅 방식보다 대역폭 사용의 효율성 및 노드 실패/토폴로지 변화에 강한 대처 능력을 보인다.

1. 서론

인터넷은 패킷 네트워크 이론과 TCP/IP를 기반으로 탄생하였으며 1990년대 초반 월드 와이드 웹 (WWW)이 등장함에 따라 엄청난 성장을 하였고, 이제는 현실 세상과 떼어놓을 수 없는 중요한 통신 인프라가 되었다.

인터넷은 TCP/IP의 종단간 (end-to-end) 패러다임을 기반으로 동작하기 때문에 1:1의 호스트 대 호스트 (host-to-host) 통신을 기본으로 가정하고 있다. 그러나 현재의 인터넷 사용은 인터넷 설계 방식과는 다른 1:N 혹은 M:N 통신 방식을 통한 대용량 콘텐츠의 검색과 획득에 집중되고 있다. 예를 들어 P2P 트래픽이 인터넷 트래픽의 대부분을 차지하고 있다는 최근의 연구결과[1]는 이러한 패러다임 변화의 좋은 근거가 될 수 있다. 다시 말하면, 현재의 인터넷 사용 패턴은 컴퓨터 간의 1:1 대화 (conversation) 중심에서 데이터 분배 (dissemination) 중심으로 패러다임이 바뀌고 있다.

현재의 인터넷이 연결과 대화 중심의 1:1 통신을 가정하고 있기 때문에 N개의 데이터를 M명의 사용자에게 전달하는 경우 네트워크 대역폭 사용량은 $N \times M$ 이 되어 비효율적이다. 이러한 문제점을 해결하기 위해 IP 멀티캐스트 등이 개발되었지만 확장성 부재와 혼잡 제어의 어려움 등으로 널리 보급되고 있지 못한 실정이다. 따라서 데이터 분배 중심의 패러다임을 기반으로 하여 현재 인터넷의 사용 패턴을 효율적으로

지원할 수 있는 콘텐츠 중심의 새로운 네트워크가 필요하다.

따라서 본 논문에서는 콘텐츠 중심의 네트워크 패러다임을 지원할 수 있는 새로운 라우팅 기법인 콘텐츠 중심 라우팅 (Content-Centric Routing, CCR)을 제안한다. 본 논문은 다음과 같이 구성된다. 우선 CCR을 설명하기 위해 2장에서는 콘텐츠 기반 네트워크 관련 연구 중 가장 대표적인 연구인 콜로라도 대학의 Combined Broadcast and Content Based (CBCB) 라우팅 기법을 간단히 설명한 뒤에 CBCB 라우팅의 문제점을 알아본다. 3장에서는 CBCB 라우팅 기법의 문제점을 해결하기 위하여 본 논문에서 제안하는 콘텐츠 중심 라우팅 기법, CCR에 대해서 설명한다. 4장에서 CCR과 CBCB 라우팅을 비교 분석한 결과를 제시한 뒤에, 5장에서 본 논문을 정리하고 향후 연구 주제를 제안하며 본 논문을 마친다.

2. 관련 연구

콘텐츠 기반의 네트워크와 관련된 연구는 아직 초기 단계에 머무는 상황이다. Van Jacobson은 현재 사용자들의 인터넷 사용 방식을 고려한다면 대화방식이 아니라 데이터 자체에 초점을 맞추는 데이터 분배 (dissemination) 방식이 필요하다고 주장하였다[2]. 또한,

콜로라도 대학에서는 속성 (attribute)과 값 (value)으로 이루어진 메시지를 사용하여 라우팅을 하는 기법을 제안하였다[3]. 다음 절에서는 가장 대표적인 콘텐츠 기반 네트워크 라우팅 기법인 콜로라도 대학의 Combined Broadcast and Content Based (CBCB) 라우팅 기법[4]에 대해 자세히 소개한다.

2.1. Combined Broadcast and Content Based Routing (CBCB) 라우팅

CBCB 라우팅은 인터넷 오버레이 위에서 구성된 콘텐츠 기반의 네트워크 오버레이를 바탕으로 데이터의 전송이 일어나는 라우팅 기법이다. CBCB 라우팅에서는 기존의 인터넷과 같이 데이터를 받을 목적지 노드의 IP주소를 통해 통신이 이루어지는 대신에 소스 노드가 자신이 관심 있는 데이터를 정의하는 predicate을 뿌리게 되며, 이를 바탕으로 라우팅이 이루어지게 된다.

CBCB 라우팅에서는 이처럼 predicate을 기반으로 라우팅을 지원하기 위해서 오버레이 네트워크 위에 다시 브로드캐스트 계층 (broadcast layer)과 콘텐츠 기반 계층 (content based layer)이라는 오버레이 계층을 구성하였다. 각 계층 (layer)의 역할은 다음과 같다.

- 브로드캐스트 계층: 각 노드로부터 모든 목적지까지 가는 브로드캐스트 트리를 구성한다. 즉, 브로드캐스트 트리를 통해 특정 노드로부터 다른 모든 노드로 가는 경로를 알 수 있다.
- 콘텐츠 기반 계층: 브로드캐스트 계층 위에서 구성되며 실제로 콘텐츠 기반의 라우팅이 일어나는 계층이다. 브로드캐스트 트리 중 일부분을 이용하여 콘텐츠를 전송하는 트리를 구성하게 된다.

오버레이 상에 위와 같은 두 개의 계층을 구성한 CBCB 라우팅은 다음과 같이 동작한다. 먼저 데이터를 원하는 노드는 잠재적인 전송자 (Sender)들에게 자신이 어떠한 속성을 가진 콘텐츠에 관심이 있는지를 알리기 위해 predicate이 포함된 Receiver Advertisements (RA)를 뿌리게 되며, RA메시지는 브로드캐스트 트리를 통해 오버레이 상의 모든 노드에게 전달된다. 이 때 중간에 있는 노드들은 어느 방향 (interface)에서 어느 predicate이 왔다는 것을 테이블로 유지한다. 추후 전송자가 해당 데이터를 배포하게 되면 중간 노드는 테이블을 참고하여 데이터에 관심을 보인 RA가 온 방향으로 데이터를 전달하게 된다. 예를 들어 노드 A가 뉴욕 타임즈에 관심 있다는 RA를 뿌리게 되면 이를 받은 중간 노드의 테이블에 뉴욕 타임즈에 관한 내용이 저장되어, 나중에 전송자가 뉴욕 타임즈를 배포하면

테이블을 참고하여 노드 A까지 전달되게 된다.

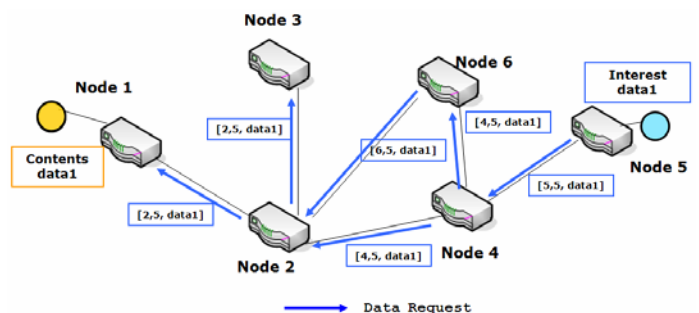
또한 중복되는 RA의 전송을 막기 위해 ingress filtering을 사용한다. 이는 이미 RA를 통해 등록한 predicate이 포함할 수 있는 predicate을 새로 받은 경우에 RA를 더 이상 전달하지 않는 방법을 의미한다. 예를 들어, 뉴욕 타임즈를 요청하는 RA를 이미 받은 중간 노드가 뉴욕 타임즈의 스포츠면을 요청하는 RA를 받게 되는 경우 뉴욕 타임즈가 뉴욕 타임즈 스포츠면을 포함하기 때문에 더 이상 RA를 전달하지 않게 된다. 다만 중간 노드는 스포츠면을 요구한 인터페이스 쪽으로 스포츠면만을 전달해야 함을 기록한다.

2.2. CBCB 라우팅의 문제점

CBCB 라우팅은 콘텐츠 기반의 라우팅을 처음으로 제안하였다는 의의가 있지만, 사전에 수신자가 RA를 통해 자신이 관심 있는 데이터의 정보를 등록해야 하므로 특정 시간마다 뉴스 같은 정보를 제공하는 주기적인 데이터 (periodic data)를 지원하는 데에만 적합한 모델이라는 한계가 있다. 또한 오버레이 네트워크를 구성하는 노드의 실패 (failure)가 발생했을 때는 모든 노드의 브로드캐스트 트리를 재구성해야 해야 한다. 따라서 브로드캐스트 트리 유지를 위한 오버헤드가 상당히 크기 때문에 노드 실패가 빈번히 발생하는 실제의 네트워크에서는 적용하기 어렵다는 단점이 있다.

하지만 콘텐츠 기반의 네트워크에서는 주기적인 데이터뿐 아니라 영화 파일과 같은 정적인 데이터도 지원할 수 있어야 하며, 노드 실패가 빈번한 상황에서도 정상적으로 동작할 수 있는 라우팅 기법이 필요하다. 따라서 본 논문에서는 이러한 문제점들을 해결할 수 있는 콘텐츠 중심의 라우팅 (Content Centric Routing) 기법을 제안한다.

3. 콘텐츠 중심 라우팅 (Content-Centric Routing, CCR)



[그림 1] CCR에서의 데이터 요청 과정

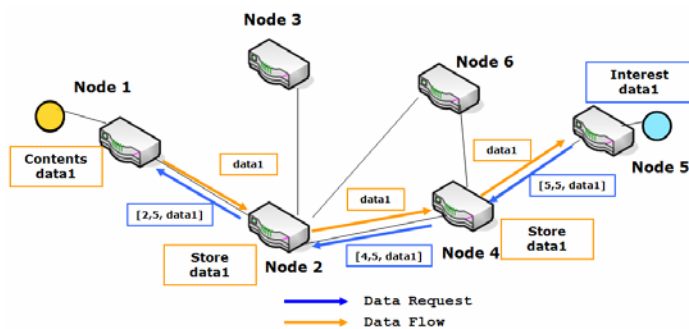
3.1. CCR의 동작 알고리즘 및 시나리오

[표 1] CCR의 동작 알고리즘

Algorithm.

1. A node requests data
2. Data requests flood restrictedly -via RPF (Reverse path forwarding)
3. A node having data sends data to reverse path
4. Intermediate routers store the data and forward it
5. Node receives the data

[표 1]에 나타난 바와 같이 CCR의 동작은 데이터를 요청하는 단계와 데이터를 받아오는 단계로 나눌 수 있다. [그림 1]은 CCR에서 노드 5가 data1을 요청하는 과정이 나타나있다. [4,5,data1]은 노드 5의 data1에 대한 요청을 노드 4가 전달한다는 것을 나타낸다. 우선 노드 5는 자신이 원하는 데이터의 파일 이름인 data1 등의 정보를 데이터 요청(data request) 패킷에 담아 콘텐츠 네트워크에 요청하게 된다. 데이터 요청을 목적지 노드까지 전달하기 위해 CCR에서는 Reverse Path Forwarding(RPF) 방식을 사용한다. RPF에서는 노드 실패에 유연하게 대처하기 위해 별도의 브로드캐스트 트리를 구성하지 않고 제한된 플러딩 방식으로 데이터 요청을 목적지 노드까지 전달하게 된다. 예를 들어 [그림 1]에서 노드 2의 경우 노드 4에게서 받은 데이터 요청을 노드 6에게도 받게 되므로 불필요한 전달을 막기 위해 노드 2는 노드 6에게서 받은 데이터 요청은 다른 노드로 다시 전달하지 않는다.



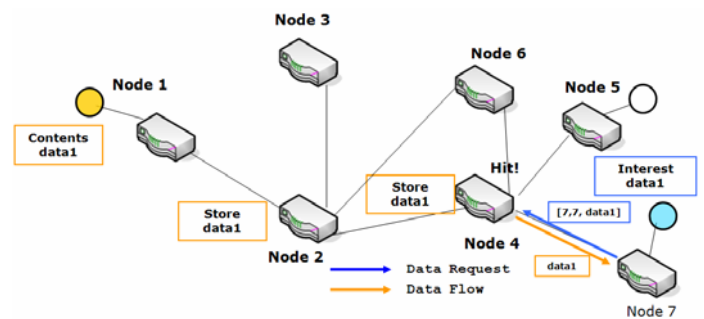
[그림 2] CCR에서의 데이터 전달 과정

CCR에서 데이터를 받아오는 과정이 [그림 2]에 나타나 있다. 위와 같은 과정으로 해당 데이터 (data1)를 가진 소스 노드(노드 1)가 요청을 받으면 데이터 요청을 전달하는 과정에서 생성된 역방향 경로 (reverse path)를 사용하여 data1을 노드 2에게 전송하게 된다. 이 때 노드 2는 전달 중인 데이터를 로컬 저장소에 저장한

뒤에 다음 노드인 노드 4로 전달한다. 이렇게 저장된 데이터는 추후 data1에 대한 요청이 올 경우, 소스 노드까지 데이터 요청을 계속 전달하지 않고 노드 2가 바로 data1을 보내주는데 사용된다. 역방향 경로를 통해 노드 4에게 전달된 data1은 최종적으로 처음 data1을 요청한 노드 5에 도착하게 되고 데이터 전송 과정은 끝나게 된다.

3.3. CCR의 장점

2장에서 설명한 대표적인 콘텐츠 중심 라우팅 기법인 CBCB 라우팅 기법과 비교해볼 때 CCR의 장점은 크게 두 가지를 들 수 있다.

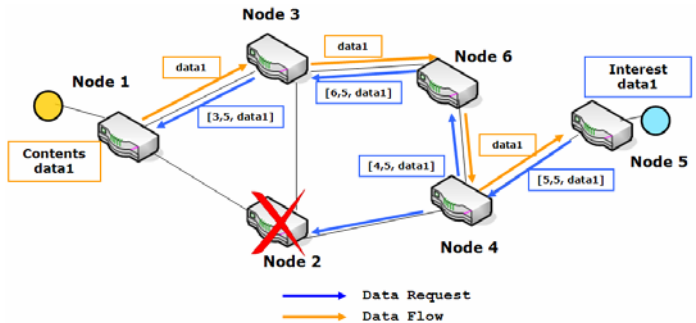


[그림 3] Store & forward을 통한 네트워크 대역폭(bandwidth) 절약

첫 번째는 Store & forward을 통한 네트워크 대역폭 (bandwidth)의 절약이다. CBCB 라우팅에서는 근처에 있는 주기적인 데이터가 아닌 정적인 데이터를 전송하는 경우, 두 노드가 같은 데이터를 요청해도 소스 노드는 목적지로 데이터를 각각 보내야 하므로 네트워크 대역폭 사용이 비효율적이다. 하지만 CCR에서는 중간 노드가 데이터를 저장하고 있기 때문에 저장하고 있는 데이터에 대한 요청을 받으면 중간 노드가 직접 전달이 가능하다. [그림 3]에 CCR에서 store & forward 방식을 통해 네트워크 대역폭을 절약하는 예가 나타나 있다. [그림 2]에 나타난 바와 같이 노드 4는 노드 5의 data1에 대한 데이터 요청을 처리하면서 data1을 자신의 로컬 저장소에 저장해두었다. 이 상황에서 노드 7로부터 data1에 대한 요청을 받은 경우, 노드 4는 데이터 요청을 더 이상 전달하지 않고 자신의 로컬 저장소에 저장되어 있는 data1을 직접 노드 7로 보내주게 된다. 따라서, 노드 4는 데이터 요청을 다른 노드로 전달하지 않으며, data1도 소스 노드로부터 재전송 될 필요 없이 노드 4와 노드 7 사이에서만 전송이 일어나므로 네트워크 대역폭을 절약할 수 있다.

둘째로 오버레이 네트워크 상에서 노드 실패가 발생하거나 토폴로지의 변화가 생겼을 때 빠른 복구가 가능하다. CBCB 라우팅에서는 노드 실패가 발생하거나

토폴로지가 바뀌었을 때 모든 노드가 자신의 브로드캐스트 트리를 새롭게 재구성해야 하는 오버헤드 발생한다. 하지만 CCR에서는 RFP에 기반한 제한된 플러딩 기법을 통해 데이터를 요청/전달하므로 브로드캐스트 트리를 유지하는 등의 오버헤드가 발생하지 않는다. [그림 4]에서 노드 2의 실패 상황이 발생하는 경우, 노드 5의 데이터 요청은 노드 6, 노드 3을 거쳐서 결국 소스 노드에게 전달 될 수 있으므로 노드 2의 실패 여부와 상관 없이 데이터를 받을 수 있다. 이와 같이 노드 실패가 발생하는 경우나 토폴로지의 변화가 생기는 경우에도 소스 노드까지 다른 경로를 이용해서 데이터 요청을 전달할 수 있으므로 CCR이 CBCB 라우팅에 비해 좀 더 노드 실패에 강한 것을 알 수 있다.



[그림 4] 노드 실패와 토폴로지 변화에 대응

위와 같이 CCR은 store & forward 방식과 Reverse Path Forwarding (RFP)을 채용하여 대역폭 사용의 효율성 및 노드 실패/토폴로지 변화에 강한 대처 능력을 보인다. 따라서 다양한 속성을 가진 콘텐츠가 네트워크를 통해 전달되고, 수많은 노드가 존재하여 노드 실패/토폴로지의 변화가 빈번히 발생할 콘텐츠 중심의 네트워크에는 CBCB 라우팅 기법보다 CCR이 적합할 것으로 보인다. 다음 장에서는 보다 정확한 정량적 성능 비교를 위한 실험 결과를 보인다.

4. 실험 결과

4.1. 실험 설정

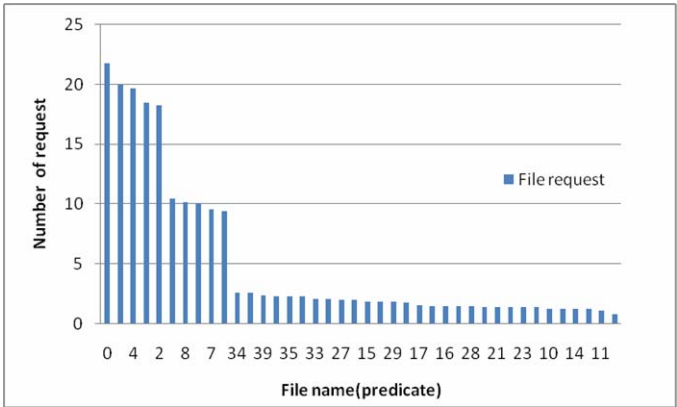
CCR과 CBCB 라우팅의 성능을 비교 평가하기 위해 JAVA를 이용하여 오버레이 네트워크를 구현하였고, 그 위에 CCR과 CBCB 라우팅 모듈을 구현하여 시뮬레이션을 수행하였다. 다양한 네트워크 환경에서의 라우팅 동작을 확인해보기 위하여 무작위로 생성된 토폴로지에 대해 실험을 하였으며, 임의의 시간에 노드 실패가 발생하도록 하였다.

CBCB 라우팅은 요청 노드가 RA를 보내면 이를 받은 소스 노드가 바로 데이터를 전송하는 것이 아니라 데이터에 대한 관심 여부만을 등록 한 뒤, 추후 데이터

배포 시 전송이 이루어진다. 이에 반해 CCR에서는 데이터 요청이 들어오는 경우 소스 노드가 바로 데이터를 전송하게 된다. 따라서 CCR과 비교를 위해 CBCB 라우팅에서도 소스 노드가 RA를 받으면 바로 데이터를 보내주는 것으로 가정하였다. 이는 영화/음악 다운로드와 같은 콘텐츠 기반 네트워크의 일반적인 사용 방식에 더욱 적합한 모델이다. 또한 모든 노드들은 사전에 자신의 데이터를 업로드 해놓았다고 가정하였으며, CBCB 라우팅의 predicate으로 파일 이름을 사용하였다.

실제 사용자의 콘텐츠 요청 패턴은 인기 있는 콘텐츠의 대한 요청이 전체 데이터 요청의 대다수를 차지하는 지프 확률 분포 (Zipf probability distribution)[5]를 보이게 된다. 따라서 본 실험에서는 지프 확률 분포를 반영하여 [그림 5]와 같이 파일의 인기도에 따라 데이터 요청의 빈도수를 조절하여 특정 인기 있는 콘텐츠에 대한 요청이 대다수를 차지하도록 조절하였다. 파일의 크기는 10Mbytes, 제어 패킷의 크기는 1Kbytes로 가정하였다.

CCR의 성능을 확인하기 위한 평가 기준으로는 데이터 요청 당 평균 네트워크 대역폭 (bandwidth) 사용량, 노드 실패 발생 시 평균 전송 시간, 그리고 요청당 제어 패킷의 숫자인 오버헤드를 사용하였다.



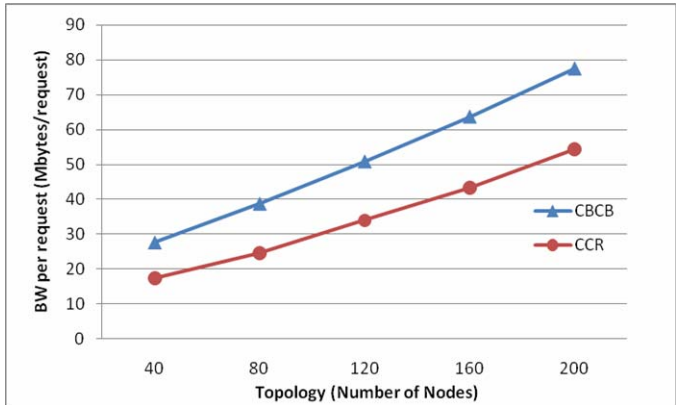
[그림 5] 40개 파일에 대한 노드의 요청 분석

4.2. 네트워크 대역폭 사용 (Bandwidth usage)

이 실험에서는 CCR과 CBCB 라우팅에서 데이터 전송 시의 네트워크 대역폭 사용량을 측정하였다. 본 실험에서 네트워크 대역폭 사용 (bandwidth usage)은 파일에 대한 요청이 발생했을 때 오버레이 네트워크의 링크에 지나가는 제어 패킷과 데이터 패킷의 크기로 정의하였다.

[그림 6]에 CCR과 CBCB 라우팅의 전체 노드 수에 따른 요청 당 대역폭 사용량을 나타내었다. [그림 6]의 결과를 보면 전체 노드 숫자가 증가함에 따라 데이터를 받아오는 평균 홉 수가 증가하여 CCR과 CBCB 라우팅

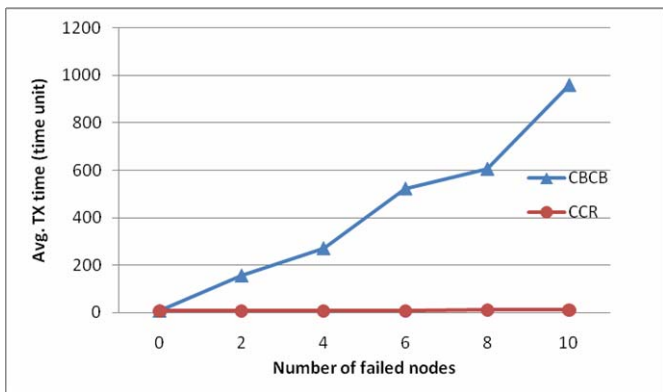
모두 대역폭 사용량은 증가하는 것을 확인할 수 있다. 그러나 전체 노드의 숫자와 상관없이 CCR이 CBCB에 비해 약 70% 정도의 네트워크 대역폭만 사용하고 있는 것을 확인할 수 있다. 이는 3.3에서 설명한 대로 CBCB 라우팅에서는 소스 노드에서 모든 요청에 대해 매번 데이터를 보내야 하지만 store & forward 기법을 사용하는 CCR에서는 중간 노드가 해당 데이터를 저장하고 있을 때에는 직접 전달이 가능하므로 네트워크 대역폭을 절약할 수 있기 때문이다.



[그림 6] 전체 노드 수에 따른 요청 당 대역폭 사용량

4.3. Failure effects

노드 실패에 따른 네트워크의 견고함(robustness)을 확인해보기 위해 임의의 시간에 0~10개의 노드 실패를 발생 시킨 후 평균 전송 시간이 어떻게 달라지는지를 알아보았다. 데이터를 요청한 후 데이터를 받을 때까지 걸리는 시간을 전송시간으로 정의하였다.



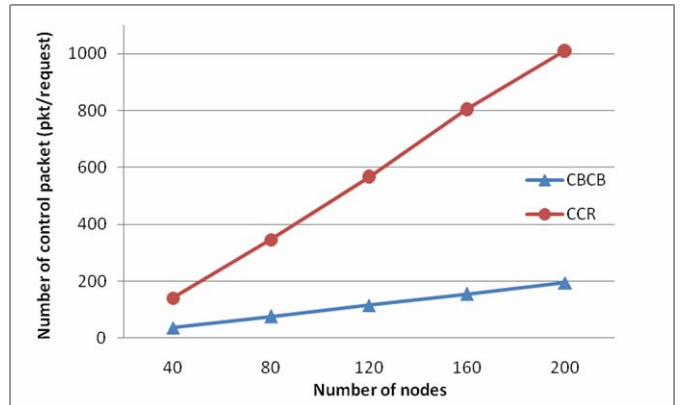
[그림 7] 노드 실패 발생에 따른 평균 전송 시간

[그림 7]에 노드 실패 발생에 따른 평균 전송 시간이 나타나 있다. [그림 7]과 같이 노드 실패가 발생하지 않았을 때는 CCR과 CBCB 라우팅의 평균 전송 시간이 차이가 거의 없지만 노드 실패가 많아질수록 CBCB 라우팅에서는 평균 전송 시간이 커지는 것을 확인할 수 있다. 이에 반해 CCR은 노드 실패가 발생하는 숫자와

상관없이 일정한 평균 전송 시간을 보인다. 이는 CCR에서는 노드 실패가 일어나도 다른 경로를 이용하여 전송을 재개 할 수 있지만, CBCB 라우팅에서는 브로드캐스트 트리를 재구성해야 하는 오버헤드로 인하여 평균 전송 시간이 급격하게 증가하게 되기 때문이다.

4.4. 오버헤드 (Overhead)

이번 실험에서는 CCR과 CBCB 라우팅에서 하나의 데이터를 받아오기 위해 발생하는 오버헤드를 비교하였다. 데이터 요청 당 발생하는 제어 패킷의 숫자를 오버헤드라고 정의하였다.



[그림 8] 전체 노드 수에 따른 데이터 요청 당 제어 패킷 수

[그림 8]에 전체 노드 숫자에 따른 데이터 요청 당 제어 패킷의 숫자가 나타나 있다. [그림 8]의 결과를 살펴보면 CCR이 CBCB 라우팅 기법에 비해 오버헤드가 큰 것을 확인할 수 있고, 노드 수가 증가함에 따라 이 차이가 더 커지는 경향을 보인다. 이는 CCR에서는 제한된 플러딩을 사용하기 때문에 브로드캐스트 트리를 따라 제어 패킷이 전달되는 CBCB 라우팅에 비해 더 많은 제어 패킷을 발생시키기 때문이다. 그러나 데이터 패킷 (10Mbytes)에 비해 제어 패킷 (1Kbytes)의 크기가 매우 작기 때문에 전체 네트워크 대역폭 사용량은 4.2절의 결과처럼 CCR이 더욱 우수하다. 하지만 제어 패킷의 숫자가 늘어나는 것이 좋은 현상은 아니므로 오버헤드를 줄이기 위한 연구를 진행 할 계획이다.

5. 결론

현재의 인터넷이 종단간 통신을 가정하고 설계된 반면에 실제 인터넷의 사용은 대용량 데이터를 검색/획득하는 콘텐츠 중심의 방식에 집중되고 있다.

따라서 본 논문에서는 실제 인터넷 사용 패턴을 잘 반영하는 효율적인 콘텐츠 중심의 라우팅 기법인 CCR을 제안하였다. CCR은 기존에 제안된 콘텐츠 중심 라우팅 기법 중 대표적인 CBCB 라우팅과 비교했을 때 대역폭 사용의 효율성 및 노드 실패/토폴로지 변화에 강한 대처 능력 측면에서 더 좋은 성능을 보였다. 앞으로 우리는 데이터 요청 당 사용된 제어 패킷의 숫자를 줄이기 위한 연구를 수행할 계획이며, 실제 구현을 통해 그 가능성을 검증할 예정이다.

참고 문헌

- [1] An iPoque's Press Release
http://www.ipoque.com/news_&_events/internet_studies/internet_study_2007
- [2] Van Jacobson, A New Way to look at Networking.
<http://video.google.com/videoplay?docid=-6972678839686672840>
- [3] Content-Based Networking in University of Colorado
<http://serl.cs.colorado.edu/serl/cbn/>
- [4] A. Carzaniga, M.J. Rutherford, and A.L. Wolf, "A Routing Scheme for Content-Based Networking". Proceedings of IEEE INFOCOM 2004. Hong Kong, China. March, 2004.
- [5] George K. Zipf (1949) Human Behavior and the Principle of Least-Effort. Addison-Wesley.