

콘텐츠 중심 네트워킹에서 효율적인 병렬 인터레스트 포워딩 기법 연구

이두호[○], 강효성, 홍충선*
 경희대학교 컴퓨터공학과
 { dooholee, kanghs, cshong }@khu.ac.kr

Efficient Parallel Forwarding Strategy in Content Centric Networking

Doo Ho Lee[○], Hyo Sung Kang, Choong Seon Hong*
 Department of Computer Science and Engineering, Kyung Hee University

요 약

ICT 기술의 발달로 수많은 장치들이 인터넷에 연결되고 방대한 양의 트래픽을 생산해내고 인터넷 이용자들은 양질의 콘텐츠 서비스를 원한다. 하지만 기존의 TCP/IP 네트워크가 급격하게 증대되는 네트워크 트래픽을 수용하기에 문제점이 많다. 따라서 대두되는 차세대 인터넷 기술로 네트워크 내에 중복되는 요청을 줄여주는 콘텐츠 중심 네트워킹(Content-Centric Networking, CCN)이라는 기술이 있다. 본 논문에서는 CCN 환경에서 높은 전송 효율을 보장하기 위한 인터레스트 병렬 포워딩 기법을 제안한다. 멀티패스를 이용하여 RTT와 대역폭에 따라 전송할 인터레스트 패킷의 양을 동적으로 조정한다. 따라서 콘텐츠 요청 시 효율적인 인터레스트 패킷 분배로 높은 전송 효율을 보장 받을 수 있을 것이다.

1. 서 론

현재 인터넷을 통한 멀티미디어 콘텐츠 서비스가 보편화 되어있고, 수많은 사람들이 멀티미디어 콘텐츠를 서비스를 이용하고 있다. 따라서 네트워크 안에서 무수히 많은 트래픽이 생성된다. 이러한 문제를 해결하고자 하는 IT 기술들도 충분히 발전하고 있으며 그중에서도 기존 TCP/IP 네트워크가 가지는 확장성, 보안성 등의 여러 문제를 해결하고 네트워크 안에서 무수히 복제되는 메시지의 중복을 막는 차세대 인터넷 기술, 콘텐츠 중심 네트워킹(Content-Centric Networking, 이하 CCN) 기술이 제안되었다 [1]. CCN은 송신자, 수신자 중심의 통신 구조를 가진 TCP/IP 네트워크와는 다르게 전송방기를 원하는 콘텐츠를 중심으로 네트워킹이 이루어진다. CCN에서 라우터는 캐싱 기능을 가지고 있어서 라우터를 지나가는 패킷을 콘텐츠 스토어에 저장했다가 이후에 같은 패킷에 대한 요청 메시지를 받으면 캐싱된 패킷을 전송해준다. 만약, 라우터의 콘텐츠 스토어에 요청된 패킷이 존재하지 않으면, 라우터는 요청 메시지를 주변의 모든 라우터에게 포워딩한다. 만일, 모든 라우터가 요청된 패킷을 캐싱하고 있지 않다면, 이 요청 메시지는 결국 콘텐츠를 보유한 서버까지 도달하게 된다. 이러한 CCN의 동작으로 인해 요청 메시지가 한 개의 서버로 몰려 생기는 병목현상이나 서버 과부하를 감소시킬 수 있다 [2].

하지만 CCN의 동작이 근본적인 전송속도를 빠르게 만들어 주는 것은 아니다. 네트워크 안에 분산 되어 있는 콘텐츠 서버들로부터 더 빠르게 전송 받을 수 있는 경로를 선택하는 것이 주요한 이슈이고, 운 좋게 중간 라우터에서 콘텐츠를 보유하고 있는 경우가 전송 딜레이가

줄어드는 경우다. 따라서 본 논문에서는 CCN 환경에서 다중 무선 인터페이스를 지원하는 모바일 디바이스가 동일한 콘텐츠를 보유한 여러 콘텐츠 서버로부터 동시에 콘텐츠를 전송받을 수 있는 포워딩 전략을 제안한다. 이 전략을 적용하면 다중 무선 인터페이스의 대역폭들을 충분히 활용하여 높은 전송 효율을 얻을 수 있을 것이다.

2. 관련 연구

2.1 콘텐츠 중심 네트워킹

CCN은 기존 TCP/IP 네트워크의 문제를 보완하기 위해 연구되고 있는 차세대 인터넷 패러다임으로 2009년 PARC연구소에서 처음 소개되었다. 기존의 위치 중심의 네트워킹에서 벗어나 무엇이 전송되는지에 초점을 맞추는 새로운 네트워킹 패러다임이다.

CCN 라우터는 그림 1 [1] 에서 보이는 것과 같은 구조를 가지고 있다. CCN 라우터는 콘텐츠의 청크를 저장

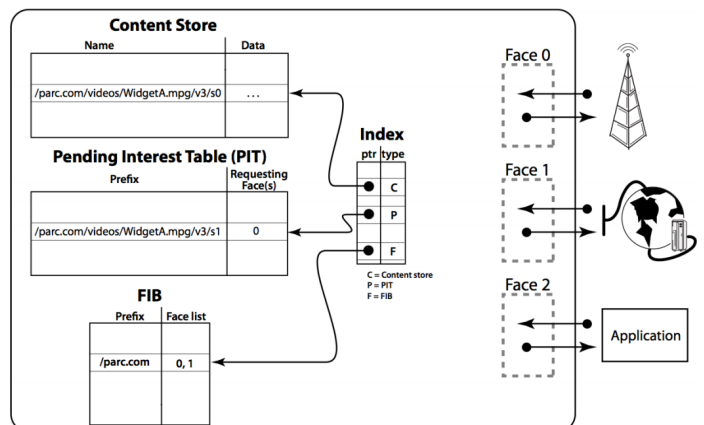


그림 1 CCN 포워딩 엔진 모델

본 연구는 미래창조과학부가 지원한 2015년 정보통신/방송(ICT) 연구개발사업의 연구결과로 수행되었음.
 *Dr. CS Hong is the corresponding author

할 수 있는 Content Store(CS)와 수신된 요청메시지를 관리하는 Pending Interest Table(PIT)과 전송되는 패킷을 관리하는 Forwarding Information Base(FIB)로 이루어져 있다. CCN에서 패킷이 드나드는 인터페이스를 Face라고 지칭하며 Face가 여러개 존재하면 뒤에 번호가 붙여진다. 콘텐츠를 요청하는 패킷은 인터레스트(Interest) 패킷이라고 불리며 이에 대응되어 전송되는 패킷은 데이터(Data) 패킷이라고 불린다.

2.2 포워딩 전략

효율적인 전송을 위한 포워딩 전략으로 크게 2가지 종류를 말할 수 있다. 첫 번째는, 멀티패스를 이용하는 방법으로 클라이언트가 동일한 콘텐츠를 갖는 여러 서버에 인터레스트 패킷을 나누어 포워딩하여 콘텐츠를 전송받는 방법이다. 두 번째는, 다중 인터페이스를 모바일 디바이스가 서버 혹은 AP에 연결되어 멀티패스를 생성하여 한 개의 인터페이스를 연결했을 경우보다 더 높은 대역폭으로 콘텐츠를 전송받는 방법이다. 이들과 관련된 연구로 [3]에서는 다중 인터페이스를 가진 디바이스가 각각 다른 네트워크에 연결 되어있는 경우에 어느 인터페이스를 통해 인터레스트 패킷을 보내는 것이 효율적인지 결정하는 알고리즘을 제안했다.

3. 결론

본 논문에서는 콘텐츠를 전송받는 효율을 높이기 위해 다중 인터페이스로 수립된 멀티패스 환경에서 각 패스의 실시간 RTT와 현재 대역폭 상태에 기반한 인터레스트 병렬 포워딩 방법을 제안한다. 본 방법을 제안하기 앞서 모바일 디바이스는 최소 2개 이상의 무선 인터페이스를 가지고 있다고 가정한다.

기존의 CCN은 인터레스트 패킷을 전송하고 가장 먼저 응답 메시지를 전송하는 경로를 최적의 패스로 간주하고 단일 경로를 수립한다. 이 후에 도착하는 응답은 버려진다. 하지만 본 제안에서는 멀티패스 수립을 위해 모바일 디바이스가 보유한 인터페이스의 수만큼 선착순으로 도착한 응답으로 패스를 수립한다. 이렇게 수립된 멀티패스는 각 인터페이스가 다른 네트워크에 접속되어 서로 다른 콘텐츠 공급자와 연결된 패스를 수립했다는 것을 가정으로 한다. 그리고 연결된 패스의 수만큼의 크기의 큐를 생성한다. 이 큐는 각 인터페이스에 동일한 콘텐츠를 요청하는 인터레스트를 어느 페이스에 얼마나 전송할지 결정하는 비율 값과 페이스 번호를 저장하고 한다. 인터레스트 패킷을 보내고 데이터 패킷을 받을 때마다 실시간으로 RTT가 계산되어 큐에 저장된다. 이 때, 가장 먼저 도착한 패스의 RTT, 즉 가장 작은 크기의 RTT가 큐의 가장 앞쪽부터 오름차순으로 저장된다. 그리고 RTT가 저장 될 때, 해당 RTT값을 가지는 인터페이스의 대역폭 또한 계산되어서 RTT값에 곱해진다. 이 값이 작을수록 더 많은 인터레스트 패킷을 보낼 수 있다는 것을 의미한다. 하지만, 작은 수에 더 많은 인터레스트 패킷을 보내는 비율을 계산하기 어렵다. 따라서 이렇게 계산된 값의 분자와 분모를 반전시켜 큰 값이 더 많은 인터레스트 패킷을 보낼 수 있게 변경한다. 하지만 이 값도 계산 과정에 의해 소수점이 생기기 쉽다. 따라서 이 값들을

정수비로 만들기 위해 큐에 저장된 각 비율을 가장 끝에 저장된 즉 크기가 가장 작은 비율로 나눈다. 그리고 소수점은 내림한다. 실시간으로 변하는 RTT와 이용 가능한 대역폭에 따라 전송할 인터레스트 패킷의 비율을 조정하는 수식은 (1) 이다.

$$FR_i = \left\lfloor \frac{RTT_i \times BW_i}{RTT_k} \right\rfloor \quad (i = 1 \dots k) \quad (1)$$

(1)에서 BW는 대역폭을 지칭하고, FR은 실제 인터레스트 패킷을 각 페이스로 전송할 포워딩 비율을 나타낸다. i는 큐의 인덱스를 의미하고 처음부터 끝까지 증가한다. k는 큐의 마지막 인덱스를 의미하고 k번째 RTT는 가장 늦게 도착한 RTT를 의미한다. (1)은 매번 데이터 패킷이 도착할 때마다 재계산되며 만약 RTT값이나 대역폭의 변화에 의해 비율 값이 달라지는 경우 내림차순 정렬을 통해 큐의 가장 앞쪽에 가장 큰 비율 값이 저장되게 한다. 큐에 담긴 FR의 수치만큼 인터레스트 패킷의 비율을 달리하여 각 인터페이스에 인터레스트 패킷을 전송한다. 만약 FR이 각각 2, 1 이라고 한다면, 먼저 큐의 첫 번째에 담긴 페이스로 청크 1, 2를 요청하는 인터레스트 패킷을 보내고 두 번째에 담긴 페이스로 청크 3을 요청하는 인터레스트 패킷을 보낸다. 그리고 과정이 반복된다. 만약 반복되는 과정에서 FR이 1, 3으로 바뀌게 된다면 즉시 전송할 인터레스트 패킷의 양을 변경하여 전송한다.

4. 성능 평가

4.1 시뮬레이션 환경

본 논문에서 제안하는 방법을 평가하기 위해 ccnsim [5] 시뮬레이터를 사용했다. ccnsim 0.3 버전을 사용하였으며 본 논문에서 제안하는 방법을 테스트하기 위해 코드를 수정했다. 표 1은 실험에 수행된 환경설정 매개 변수들을 보여준다. 본 실험은 2개의 서로 다른 무선 인터페이스를 가진 모바일 디바이스가 동일한 1개의 콘텐츠를 전송받을 때 걸리는 시간을 측정했고 서로 다른 Arrival rate 값을 가지고 3번 실험했다. 각각의 Arrival rate 값은 5, 10 그리고 15로 지정했다. 모바일 디바이스에 연결되는 네트워크는 LTE 네트워크와 Wi-Fi 네트워크

표 1 환경설정 파라미터

| Parameters | Value |
|----------------------|--------------------|
| Chunk size | 10 KB |
| Number of repository | 3 |
| Number of replicas | 3 |
| Number of clients | 3 |
| File size | 1 |
| Object size | 10,000 |
| Cache decision | LCE |
| Cache replacement | LRU |
| Zipf exponent | 1 |
| Cache size | 1 % of Object size |
| Arrival rate | 5, 10, 15 |

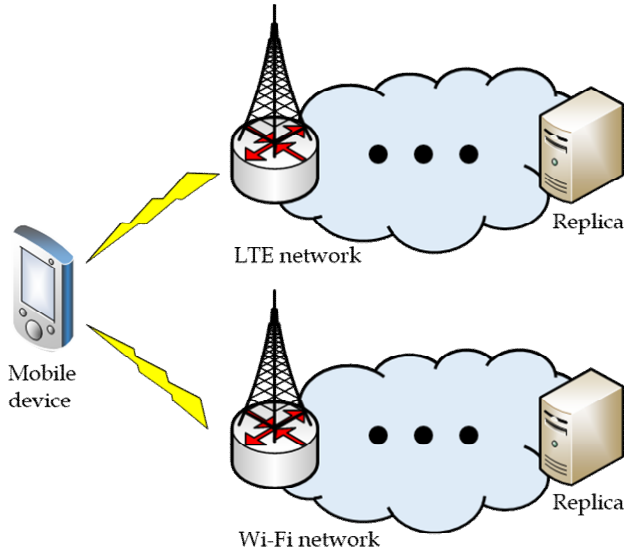


그림 2 서로 다른 2개의 무선 네트워크에서 동일한 콘텐츠를 전송받는 시나리오

워크로 가정한다. 따라서, LTE 네트워크의 대역폭은 OpenSignal에서 조사한 KT LTE의 대역폭 수치인 36 Mbps로 지정하고 [4], Wi-Fi 네트워크의 대역폭은 2.4 GHz 대역의 IEEE 802.11b의 대역폭 수치인 11 Mbps로 지정했다. 실험을 위한 토폴로지 구성은 그림 2에 나타나 있다. 각 네트워크에서 서버까지의 홉 수는 5이다.

4.2 시뮬레이션 결과

한 개의 콘텐츠를 전송받는데 걸리는 총 시간에서 총 전송받은 청크의 수를 나누어 한 개의 청크를 전송받는데 걸리는 평균 딜레이를 측정했다. 한 개의 LTE 인터페이스를 이용하여 총 청크 수 10,000개, 청크의 크기 10 KB인 콘텐츠를 전송받은 경우와 LTE 인터페이스와 Wi-Fi 인터페이스를 한 개씩 총 2개의 인터페이스를 이용하여 전송받은 경우를 비교한 결과가 그림 3에 나타나 있다. 그림 3에서 x축은 Arrival rate를 의미하고 y축은 1개의 청크를 전송받는데 소요되는 평균 딜레이를 의미하고, 한 개의 인터페이스를 이용하여 콘텐츠를 전송받

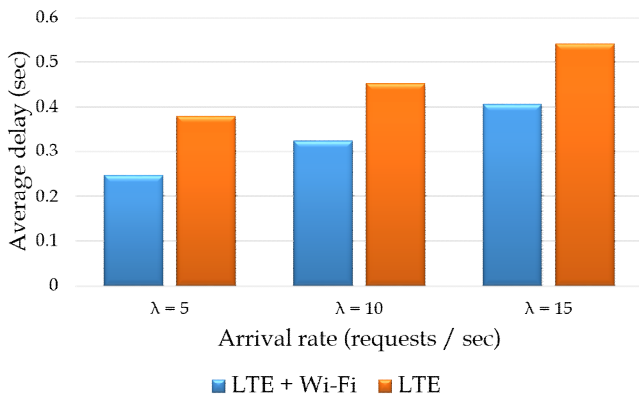


그림 3 청크 1개를 전송받는데 소요되는 평균 전송 딜레이

는 것보다 두 개의 인터페이스를 이용하여 콘텐츠를 동시에 전송 받는 것이 딜레이가 더 적다는 것을 알 수 있었다.

5. 결론

본 논문에서는 모바일 디바이스가 다중 인터페이스를 가지는 환경에서 실시간 RTT와 이용 가능한 대역폭에 기반 하여 멀티패스를 이용한 인터레스트 병렬 포워딩 방법을 제안했다. 이 제안된 방법을 ccnsim 시뮬레이터를 이용해 실험한 결과로 2개의 인터페이스를 이용하여 콘텐츠를 전송받은 결과가 더 우수한 것으로 나타났다.

향후 연구로는 다중 인터페이스의 대역폭을 더 효율적으로 활용하기 위해 전송할 인터레스트 패킷의 비율을 최적화 시키는 FR 값을 결정하는 수식을 더 연구할 예정이다. 그리고 명확한 비교를 위해 기존에 제안된 다른 다중 인터페이스를 이용한 포워딩 방법과 비교할 것이다.

6. 참고 문헌

- [1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, R. L. Braynard, "Networking Named Content", CoNEXT 2009, Rome, Dec, 2009.
- [2] Cheng Yi, Alexander Afanasyev, Lan Wang, "Adaptive Forwarding in Named Data Networking", ACM SIGCOMM Computer Communication Review, Vol 42, Number 3, July 2012
- [3] Klaus M. Scheider, Kai Mast, Udo R. Krieger, "CCN Forwarding Strategies for Multihomed Mobile Terminals", IEEE NetSys 2015 Workshops, March 2015
- [4] OpenSignal, <http://opensignal.com/reports/2015/09/state-of-lte-q3-2015/>
- [5] Giuseppe Rossi and D. Rossi, "ccnsim: an highly scalable ccn simulator", IEEE ICC 2013, June 2013