

## MST 알고리즘 기반 콘텐츠 전송 네트워크에 관한 연구

이형옥\*, 강미영\*, 남지승°

## Content Delivery Network Based on MST Algorithm

Hyung-ok Lee\*, Mi-young Kang\*, Ji-seung Nam°

## 요약

스마트폰의 증가와 PC 성능 향상으로 유무선 통신망에 트래픽이 폭발적으로 증가하고 있다. 여기에는 페이스북, 유튜브와 같은 멀티미디어 서비스와 파일 공유가 큰 부분을 차지하고 있다. CDN(Content Delivery Network)은 원거리에 있는 콘텐츠 사업자의 웹 서버에 저장된 콘텐츠를 이용자 근처 CDN 서버에 미리 저장, 콘텐츠 요구 발생 시 최적의 CDN 서버로부터 콘텐츠를 제공하는 콘텐츠 전송 기술이다.

본 논문에서는 콘텐츠 요청 메시지 전달에 Minimum Spanning Tree(MST) 알고리즘을 응용한 SCRП(Shortest Core Routing Path) 알고리즘을 사용해 CDN 서버와 클라이언트의 콘텐츠 전달에 이용되는 전체 트래픽 양을 최적화하였다. 또한 HC\_LRU 캐시 알고리즘을 통해 캐시 적중률을 향상시킴으로써 콘텐츠 요청에 대한 평균 응답 시간을 단축시켰다. 제안한 SCRП와 HC\_LRU 알고리즘을 통해 트래픽을 지역화하고 병목현상을 방지하여 네트워크 자원을 효율적으로 사용하는 확장성 있는 콘텐츠 전송 네트워크 시스템을 구축할 수 있다.

**Key Words** : CDN, MST, SCRП, HC\_LRU, Cache

## ABSTRACT

The traffic in the wired and wireless networks has increased exponentially because of increase of smart phone and improvement of PC performance. Multimedia services and file transmission such as Facebook, Youtube occupy a large part of the traffic. CDN is a technique that duplicates the contents on a remote web server of content provider to local CDN servers near clients and chooses the optimal CDN server for providing the content to the client in the event of a content request.

In this paper, the content request message between CDN servers and the client used the SCRП algorithm utilizing the MST algorithm and the traffic throughput was optimized. The average response time for the content request is reduced by employing HC\_LRU cache algorithm that improves the cache hit ratio. The proposed SCRП and HC\_LRU algorithm may build a scalable content delivery network system that efficiently utilizes network resources, achieves traffic localization and prevents bottlenecks.

## I. 서론

스마트폰, 태블릿 등 슈퍼 단말의 폭발적인 증가는 오프라인으로 존재하던 영화, 비디오 등 멀티미디어

콘텐츠를 사용자의 손안에서 경험할 수 있도록 변화시키고 있으며, 이를 기반으로 콘텐츠 제작에서 유통, 소비까지의 콘텐츠 미디어 생태계가 인터넷으로 이동되어 모든 미디어가 이제는 온라인을 통해서 유통되

♦ First Author : Chonnam National University Department of Electronics and Computer Engineering, narcis99@nate.com, 학생회원

° Corresponding Author : Chonnam National University Department of Computer Engineering, jsnam@chonnam.ac.kr, 종신회원

\* Chonnam National University Department of Computer Engineering, kmy2221@nate.com, 정회원

논문번호 : KICS2015-10-324, Received October 6, 2015; Revised December 17, 2015; Accepted January 27, 2016

는 새로운 시대를 맞이하고 있다.

인터넷 기반 어플리케이션에는 확장성, 보안 및 신뢰성, 새로운 서비스에 대한 유연성 및 QoS 등의 요구사항이 제시된다. 기존 클라이언트-서버 방식에서 이러한 요구사항의 만족을 위해서는 복잡성 및 비용 문제가 제기된다. 반면, 모든 형태의 분산 자원 접근이 가능한 P2P 통신 방식에서는 보다 간단한 해결 방안을 제시함으로써 인터넷 기반 어플리케이션에 새로운 가능성을 제시하고 있다. 기존 클라이언트-서버 형태의 중앙 집중형 구조에서는 서버로 트래픽 집중으로 인한 병목 현상 및 queuing 지연 문제점을 내포하지만, P2P 구조에서는 각 피어로 분산된 트래픽 루트를 이용 가능하게 함으로써 효율적인 부하 분산이 가능하다. 최근에는 멀티미디어 서비스를 위한 P2P 기반의 어플리케이션들이 많이 제안되고 있다. 일반적으로 멀티미디어 서비스는 파일 공유 서비스와는 달리 대역폭, 지연, 그리고 지터 등의 측면에서 매우 엄격한 QoS 조건을 요구한다. 다수의 피어에 파일이 복제된 상태에서 P2P 방식으로 효율적인 부하 분산을 이용하는 개념은 미디어 스트리밍 및 VoD 분야에 많이 적용된다<sup>1-8)</sup>. 예로써, PeerCast, Peer-to-Peer-Radio, SCVI.net 등이 있다. 유사한 개념의 Kontiki network는 사내 e-Learning 교재 배포, antivirus 데이터 갱신, 컴퓨터 게임 업그레이드 등에 활용되고 있다<sup>9)</sup>.

본 논문에서는 이와 같은 통신망 부하 감소를 위한 P2P 트래픽 지역화 관련 기술 동향을 분석하여, CDN 기반의 콘텐츠 전달에 최적화된 오버레이 멀티캐스트 라우팅 및 데이터 전송 알고리즘을 개발하고 슈퍼 노드를 통한 콘텐츠 데이터의 효율적인 중계 전송 기술을 제안한다. 또한 슈퍼 노드를 분산 캐싱 서버로 사용하여 클라이언트의 데이터 요청을 마스터 서버와 분산 캐싱 서버로 분산시키므로써 데이터의 요청이 한 곳으로 집중되는 것을 방지함과 동시에 이러한 병목현상을 완화시켜 줄 것이다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 기술하고 3장에서는 제안하는 콘텐츠 전송 네트워크 기반 라우팅 프로토콜과 분산 캐싱 알고리즘에 대하여 자세히 기술한다. 4장에서 여러 가지 환경에서의 실험 및 평가를 실시한 후 5장에서 결론을 맺는다.

## II. 관련 연구

P2P 트래픽 지역화에 대한 방법은 많이 연구되어 왔으나, IP 홉 수 및 TTL 값 등을 사용하여 콘텐츠를 제공하는 P2P 상대노드를 지역에서 선별하기 위한 방

법은 아직 일반적이지 않다<sup>10)</sup>. P2P 트래픽 라우팅 체계 중 Chord는 노드의 IP 주소를 해시 함수로 해시한 결과 값에 따라 전후 노드를 링으로 구성하여 콘텐츠를 제공하지만 물리적인 거리를 고려하지 않으므로 실제 콘텐츠 제공 시 시스템의 성능을 감소시킬 소지가 있다<sup>11)</sup>. 그래서 물리적인 거리를 고려하여 라우팅하는 방법에 대한 연구가 이루어지고 있다<sup>12)</sup>. 한편, 글로벌 콘텐츠 제공시, IP주소의 첫 번째 옥텟을 이용하여 국가 수준 정도의 간단한 P2P 트래픽 지역화를 하는 방법도 있다<sup>13)</sup>.

ICT 시장은 스마트폰의 빠른 보급과 네트워크의 브로드밴드화(Wi-Fi, LTE 등)로 인하여 대용량의 고품질 동영상 콘텐츠와 음원 콘텐츠의 데이터 전송량이 급속하게 증가하고 있으며, 사용자들은 고품질의 멀티미디어 콘텐츠를 다양한 디바이스를 통해서 언제, 어디서나 적절한 비용으로 이용할 수 있는 통신서비스 사업자를 선택할 것이다. 이제는 통화 품질의 우수성이 아닌 YouTube, Netflix, 아프리카 TV 등과 같은 멀티미디어 콘텐츠를 소비자가 허용할 수 있는 적절한 가격으로 끊임 없이 전달해줄 수 있는나는 QoE(Quality of Experience)가 사용자가 통신서비스 사업자를 선택하는 기준이 되고 있다. 그렇다고 통신 서비스사업자들이 자사 콘텐츠도 아닌 타사 대용량 콘텐츠 전송을 위해 자사 코어망 용량 증설을 위해서 투자만 한다면 수익율은 크게 떨어질 수 밖에 없다. 또한 글로벌 고객들을 대상으로 서비스를 제공하는 글로벌 콘텐츠 제공업체들도 국가별, 통신서비스사업별로 품질이 다른 데이터 전송 네트워크의 QoS를 일정 수준 이상으로 보장할 수도 없다. 이런 문제점을 해결할 수 있는 솔루션이 바로 CDN(Content Delivery Network)이다. 현재까지의 CDN들은 대부분 계층적 중앙 집중 구조를 가지고, 관련 연구들도 이러한 구조를 가정하고 있다. 중앙집중형 CDN 구조에서는, 사용자의 매 서비스 요청에 대해서 서비스 제공자 측의 특정 노드가 콘텐츠 소스 선택 기능을 가지며, 사용자가 요청한 콘텐츠를 서비스할 수 있는 소스 노드들 중에서 소스 노드 선택 방안을 사용하여 사용자와 콘텐츠 소스 간의 선택 기준에 의거해서 소스 서버를 선택하여 사용자에게 알려준다. 중앙집중 구조는 나름대로의 장점도 있지만, 서비스 요청 및 제공을 위한 각종 부하가 집중되고, 또한 트래픽도 망의 각 링크에 병목을 야기하는 경우가 있으며, 확장성, 장애에 대한 관용성 보장 등의 측면에서 분산형 구조에 비해 불리할 수 있다. 이에 비해 분산형 서버 선택 구조를 사용하면, 사용자의 요청은 분산형 해시 테이블(DHT)

을 이용한 P2P 알고리즘이 동작되는 오버레이 네트워크 상에서 콘텐츠의 등록, 검색, 선택 등이 수행된다. P2P 방식에 대해서는 상당기간 많은 연구가 이루어졌고, 다양한 형태의 많은 방식이 제안되었다. 중앙 집중형(cnetralized)의 P2P는 중앙 집중형 CDN과 유사한 속성을 가질 것이고, Unstructured 형태의 P2P들은 탐색 과정에서 기본적으로 Flooding 방식에 의존하므로 트래픽을 무분별하게 발생 시킬 수 있다. 동작 시의 트래픽 발생을 보다 절제하면서도 효과적으로 동작하는 P2P들이 많이 제안되었고, 원형 주소 공간에 각각의 노드와 데이터의 키 값을 할당한 뒤 DHT(distributed hash table) 알고리즘을 이용하여 원하는 노드 정보를 얻는 Chord가 그 중 하나이다. 이러한 DHT 알고리즘을 통해 원활하게 피어의 참여와 이탈을 관리할 수 있고, 전체 라우팅 정보의 일관성을 유지하게 되며 DHT가 적용된 퍼거 테이블을 통해 라우팅 횟수를 줄어줄 수 있다. CAN에서는 d-차원의 테카르트 좌표 공간을 가상의 논리적 주소로 사용하는 방식을 제안하였다<sup>[14]</sup>. 이를 통해 장애에 강하며 확장성이 있고 자가 구성이 가능한 P2P 네트워크를 구성하였다. Chord에서는 DHT를 적용하여 서로 다른 동적 P2P 시스템에서의 부하 분산을 위한 알고리즘을 제안하였고, 이를 통해 급격한 피어들의 참여와 이탈에 따른 높은 부하를 고르게 분산시켰다<sup>[15]</sup>. 이러한 DHT를 이용한 P2P 방식들의 장점으로는 효율적이고 장애에 대한 관용성, 콘텐츠 소스의 locating 용이, self-organization, self-configuration, self-healing, 높은 확장성, 서비스 신뢰성, 그리고 유지 관리에 대한 용이성 및 로드 밸런싱 등이 있다. 이러한 P2P 방식들의 장점을 CDN에 적용하면 보다 비용 효율적인 CDN을 구축할 수 있고, 이를 위해 토폴로지 구성과 라우팅 방법에 관한 많은 연구들이 제안되고 있다<sup>[16]</sup>.

웹 캐시 연구는 주로 특정한 토폴로지(트리 토폴로지, 라인 토폴로지)를 갖는 웹 캐시에 대한 연구가 이루어 졌다. 계층적으로 형성된 캐시(hierarchical caching)에서 플루이드 모델(fluid model)을 적용하여 웹 캐시(Web caching)를 하는 알고리즘에 대한 연구를 적용하여 진행하였다<sup>[17]</sup>. 계층적으로 이루어진 캐시들이 협력을 하는 경우와 그렇지 않은 경우에 대한 알고리즘을 각각 제안하였고, 캐시간의 협력을 하는 경우의 성능이 더 우수함을 입증하였다. 이후, 연구결과에서 직관적으로 서술하고 넘어가는 부분을 엄밀하게 수학적으로 표현하고 증명한 연구도 진행되었다<sup>[18]</sup>. 유사한 연구로 콘텐츠를 계층적으로 캐시(hierarchical caching) 하는 경우와 중간캐시의 연결

없이 직접적으로 캐시와 연결되는 분산적 캐시(distributed caching)하는 경우에 대한 성능을 웹 캐시의 지연 시간과 대역폭 사용정도, 캐시에 걸리는 요청 수의 측면에서 비교하고, 각각의 장단점을 보였다<sup>[19]</sup>. 계층적 캐시방법은 대역폭의 사용정도가 낮은 반면, 상위 레벨에 존재하는 캐시에 높은 성능이 요구된다. 분산적 캐시 방법은 지연시간이 짧고 네트워크 전반에 걸쳐 트래픽이 분산된다는 장점이 있으나 대역폭의 사용량이 높다는 단점이 있다. 이 두 가지 방법을 혼합하여, 계층적인 캐시를 유지하면서 같은 레벨에 존재하는 캐시 간의 협동적, 분산적 캐시 방법을 취하는 방법을 제안하였고, 이 제안된 방법이 지연시간을 낮춘다는 것을 밝혔다.

### III. 제안하는 콘텐츠 전송 네트워크

최근 수년 간 인터넷 트래픽은 연 35% 증가를 보이고 있으며, 특히 실시간 대용량 멀티미디어 서비스의 증가가 두드러짐에 따라 인터넷 사업자 및 콘텐츠 제공 사업자가 주도하는 인터넷 서비스 환경에서 네트워크는 단순한 패킷 전달의 역할에 머무르고 있다. 즉, 3rd party 사업자는 네트워크에 트래픽을 일으키고 수익을 올리고 있으나 네트워크 투자비용에 대한 부담은 이루어지고 있지 않아 네트워크 사업자의 수익성은 급격히 악화되고 있는 상황이다. 이는 네트워크투자의 축소로 이어져 Content, Platform, Network, Terminal 사업자간 갈등을 야기하는 원인이 되고 있다. 이에 따라 현재의 단순 전달 역할에 최적 구조인 네트워크를 3rd party 사업자와 개방과 협력을 통해 트래픽 증가를 수익으로 전환하기 위한 네트워크의 전환이 필요한 상황이다. 다양한 콘텐츠를 슈퍼 단말들을 통해 고화질 실시간 스트리밍 형태로 서비스 받기를 원하는 사용자의 요구가 증가하고 있다. 그러나, 현재의 IP 네트워크는 Best Effort 구조를 가지고 있어 이를 통해 3D, Full HD 등 고품질 비디오 콘텐츠에 대한 실시간 스트리밍 서비스 제공은 QoS 보장 측면에서 한계를 드러내고 있다. 따라서 현재의 IP 네트워크에서 비용 효율적으로 고품질 비디오 트래픽을 전달하기 위해 최적화된 구조로의 진화가 필요하다.

제안하는 콘텐츠 전송 네트워크는 콘텐츠 및 서비스 전달에 최적화된 네트워크로 콘텐츠 인지 및 저장이 가능하며, 네트워크 사업자와 인터넷 사업자 간 개방형 에코환경 구현을 통해 수익 공유가 가능한 네트워크이다. 서비스 및 네트워크 계층이 결합되어지고 SCRIP 라우팅 기능을 도입하여 동일 콘텐츠가 중복으

로 전달되는 것을 방지함으로써, 네트워크 자원을 효율적으로 사용하는 기능을 제공한다. 제안하는 콘텐츠 전송 네트워크의 특성은 트래픽 지역화를 통한 백본 구간의 가용대역폭 확대, 유연한 콘텐츠 전달을 위한 동적 캐싱과 콘텐츠의 위치 파악 및 빠른 검색을 위한 콘텐츠 라우팅이다.

### 3.1 콘텐츠 전송 네트워크 시나리오

본 논문에서 제안하는 슈퍼 노드 기반 네트워크 상에서 콘텐츠를 전송하는 시나리오는 그림 1과 같다.

- ◇ 방송 시청을 원하는 사용자는 OTT 포털 페이지를 통하여 시청하고자 하는 콘텐츠에 접근한다.
- ◇ OTT 포털 페이지에서는 콘텐츠에 관한 식별자를 사용자에게 전송시키고, 슈퍼 게이트웨이에 콘텐츠를 요청하게 된다.
- ◇ 요청을 받은 슈퍼 게이트웨이는 슈퍼 노드들의 리스트와 콘텐츠 식별자를 사용자에게 전송한다.
- ◇ 슈퍼 노드 리스트를 전송받은 사용자는 슈퍼 노드들과의 RTT(Round Trip Time) 값을 측정해 가장 가까운 슈퍼 노드에 콘텐츠를 요청하게 된다.
- ◇ 콘텐츠 요청을 받은 슈퍼 노드는 자신에게 콘텐츠의 존재 유무를 검사한 후, 콘텐츠가 존재하면 사용자에게 응답 메시지를 보낸 후, 자신의 MST(Minimum Spanning Tree) 라우팅 테이블에 따라 요청 메시지를 플러딩하게 된다. 요청 메시지를 받은 슈퍼 노드들 또한 콘텐츠 유무를 확인한 후 응답 메시지와 요청 메시지를 전달한다.
- ◇ 사용자는 응답 메시지 중 가장 먼저 도착한 메시지를 보낸 슈퍼 노드에게 콘텐츠를 요청한다. 콘텐츠를 요청받은 슈퍼 노드는 사용자와 인접한 슈퍼 노드에게 데이터를 전송하게 된다. 이 때 데이터는

SCRP(Shortest Core Routing Path) 테이블에 따라 전송된다. 그리고 데이터를 전송받은 사용자와 인접한 슈퍼 노드는 사용자에게 데이터를 전송하게 된다.

### 3.2 콘텐츠 전송 네트워크 라우팅 프로토콜

본 논문에서 제안하는 콘텐츠 전송 네트워크 시스템의 구조는 그림 2와 같다.

콘텐츠 제공자인 CP(Content Provider)는 사용자가 처음 접속하여 콘텐츠를 요청하는 서버로 사용자에게 관리서버(Managed Server)의 정보를 전송해 준다. 관리 서버는 사용자에게 콘텐츠 식별자와 SN(Super Node)의 정보를 전송해 준다. 또한 제안한 알고리즘을 통해 각각의 슈퍼 노드들에게 SCRП 테이블 정보를 전송하게 된다. 각 슈퍼 노드들은 서비스 제공자에게 의해 고정된 라우팅 기능을 가진 서버로 콘텐츠 저장 및 전달, 캐싱 기능을 동시에 수행한다.

SCRП 테이블 정보는 콘텐츠를 검색할 때 슈퍼 노드들 간에 Query메시지를 플러딩하기 위해 사용된다. SCRП 테이블 정보를 얻기 위해서 MST(Minimum Spanning Tree) 알고리즘 중 Kruskal 알고리즘을 사용하였다. 본 논문에서 제안한 SCRП 알고리즘의 의사코드는 다음과 같다.

Kruskal 알고리즘의 return 값은 MST의 각 간선들로 이루어져 있고, 이를 이용하여 각 슈퍼노드에게 전달 될 SCRП 테이블을 작성하게 된다. SCRП 알고리즘의 원리는 다음과 같다.

Kruskal 알고리즘의 return 값인 T의 각각의 정점들의 degree를 체크한다. 여기서 degree는 정점의 반복 횟수를 나타낸다.(Fig. 3 : 2 line)

Degree의 수가 1인 정점들을 모은 후, T의 간선을 참고하여, 각 정점들 중 하나를 시작점으로 다른 점들

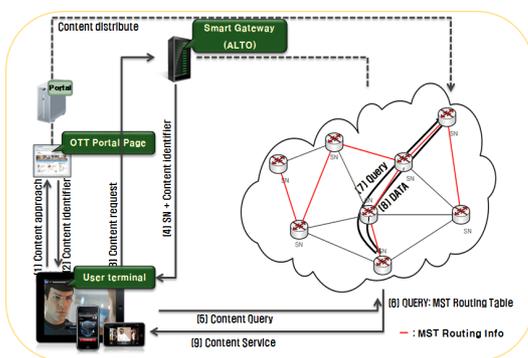


그림 1. 콘텐츠 전송 시나리오  
Fig. 1. Contents Delivery Scenario

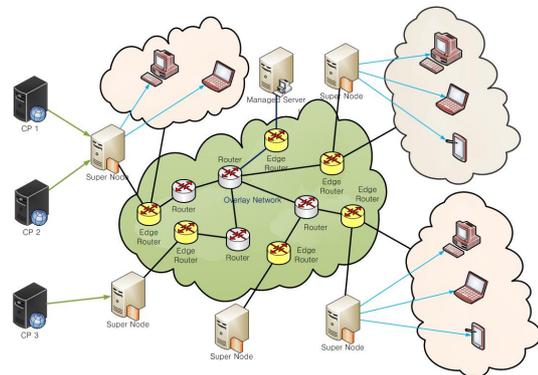


그림 2. 콘텐츠 전송 네트워크 구조  
Fig. 2. Contents Delivery Network Architecture

```

Function SCRP
  Count the degree of each vertex from T
  // T is return value of the Kruskal function
  S ← a set of vertices that the number of degree is one
  Make the Path that the one of S is a start point and the other one of S is an end point
  // i is total number of vertices in S
  // the number of Path = i(i-1)/2
  for each vertex v in G do
    Smart router SR(v) ← {v}
    for each vertex w in G do
      destination D(w) ← {v}
      find the Path has v, w
      SCRP(v, w) = vertex x beside v between v and w
  return SCRP
    
```

그림 3. SCRP 알고리즘 의사코드  
Fig. 3. SCRP Algorithm Pseudocode

끝점으로 하는 Path를 작성한다.(Fig. 3 : 5 line)  
 각 슈퍼 라우터 v는 목적지 w로 보내기 위한 최적의 라우터를 구하게 된다.(Fig. 3 : 9 line)  
 최적의 라우터는 작성한 Path 중에서 v,w를 갖는 Path를 찾은 후, v에서 w사이의 라우터 중 v에 인접한 라우터를 최적의 라우터로 선택하게 된다.(Fig. 3 : 13 line)

모든 슈퍼 라우터에 관해 모든 목적지를 구하기 위해 ④의 과정을 반복한다.

SCRP 테이블은 각각의 목적지로 가기 위해 인접한 라우터 중 최적의 라우터를 찾는 알고리즘이다. 이를 이용해 클라이언트는 MST에서 최단 경로를 통해 콘텐츠 요청 메시지를 전송하게 된다. 슈퍼 라우터 C에 대한 SCRP 테이블은 다음과 같다.

SCRP 테이블에는 각 목적지로 가기 위해 데이터를 전송해야 할 다음 라우터를 지정해 준다. 이렇게 목적지 슈퍼 노드로 데이터를 전송한 후, 목적지 슈퍼 노드는 콘텐츠를 요청한 클라이언트에게 유니캐스트로 데이터를 전송하게 된다.

그림 5는 제한한 콘텐츠 전송 네트워크 시스템의 Flowchart이다. 모든 슈퍼 노드들이 MST 트리에 조인을 하게 되면 MST Tree 모듈에서 SCRP 테이블 정보를 각각의 슈퍼 노드들에게 전송한다. 새로운 클라이언트가 관리 서버에 콘텐츠를 요청하면 콘텐츠 식별자와 모든 슈퍼 노드의 리스트를 클라이언트에게

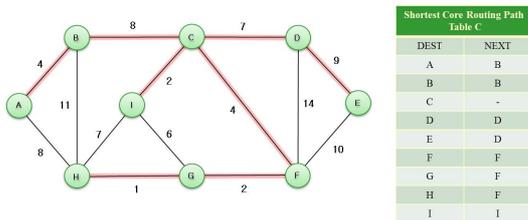


그림 4. SCRP 테이블  
Fig. 4. SCRP Table

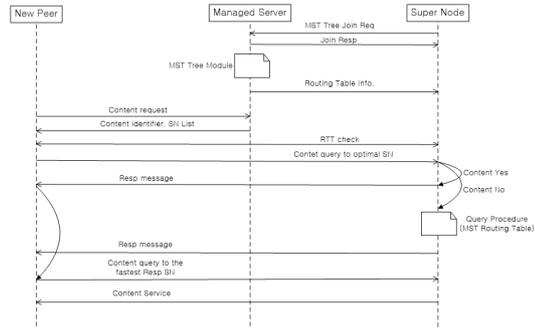


그림 5. 콘텐츠 전송 네트워크 시스템 흐름도  
Fig. 5. Contents Delivery Network System Flow Chart

전송하게 된다. 클라이언트는 전송 받은 리스트의 모든 슈퍼노드들과 RTT 측정을 하고 가장 작은 값을 가진 슈퍼 노드에게 콘텐츠를 요청하게 된다. 본 논문에서는 각 슈퍼 노드들과의 RTT값을 측정하여 최적의 슈퍼 노드를 선택했지만, 실제 네트워크 사업자들은 네트워크 트래픽 및 실시간 환경을 고려하여 최적의 슈퍼 노드를 클라이언트에게 전달할 수 있을 것이다. 콘텐츠 요청을 받은 슈퍼 노드는 콘텐츠를 보유하고 있으면 클라이언트에게 직접 콘텐츠를 전송하게 된다. 하지만 슈퍼 노드가 콘텐츠를 보유하고 있지 않으면 관리 서버로부터 전달 받은 SCRP 테이블에 따라 각 슈퍼 노드들에 콘텐츠 요청 메시지를 보낸다. 콘텐츠 요청 메시지를 받은 각 슈퍼 노드들은 콘텐츠가 존재할 경우 응답 메시지를 클라이언트가 처음 접속한 슈퍼 노드에게 보내게 되고, 가장 빠른 응답 메시지를 보낸 슈퍼 노드가 데이터를 전송하게 된다.

그림 6은 새로운 클라이언트가 접속했을 시, 전송하는 콘텐츠 검색 Query 메시지를 보여준다. 클라이언트는 슈퍼 노드들과 RTT 체크를 한 후, 최적의 슈퍼 노드인 C에게 콘텐츠 요청 메시지를 보내게 된다. C는 자신에게 콘텐츠 유무를 확인 후, 콘텐츠가 있으면 클라이언트에게 response 메시지를 전송 후, SCRP 테이블을 통해 Query 메시지를 플러딩한다. 클라이언트는 가장 먼저 도착한 response 메시지를 수신한 후 나머지 response 메시지는 discard한다. 그림 6(b)에서 콘텐츠를 가지고 있는 슈퍼 노드는 A, D, G이고, G의 response 메시지가 가장 먼저 도착했으므로 A와 D의 response 메시지는 discard하게 된다. G는 C에게 콘텐츠를 전송하게 되고, C는 이 데이터를 클라이언트에게 유니캐스트를 통해 전송하게 된다. C의 캐시에는 이 콘텐츠가 저장되어 새로운 클라이언트가 C에게 같은 데이터를 요청 시, 바로 전송을 해주게 된다.

콘텐츠 요청 메시지가 MST 테이블에 따라 전송되

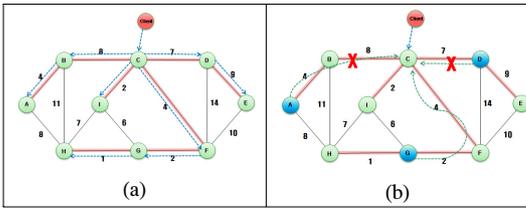


그림 6. 콘텐츠 Query 및 Response 메시지  
Fig. 6. Content Query and Response Message

므로 콘텐츠 요청이 많아질수록 전체 트래픽의 감소를 가져오게 된다. 또한 콘텐츠 요청이 증가할수록 캐시 사용의 효과가 극대화되기 때문에, 제안한 알고리즘을 대규모 네트워크에 적용했을 시, 트래픽을 지역화 시킴으로서 데이터 전송 효율을 높일 수 있을 것으로 생각된다.

### 3.3 콘텐츠 전송 네트워크 캐시 알고리즘

네트워크의 트래픽 감소 및 사용자가 콘텐츠를 받을 때까지의 시간을 감소하기 위해서는 캐시 네트워크 상에 존재하는 각 캐시가 적절한 콘텐츠를 저장하고 있어야 한다. 따라서 캐시에 저장된 콘텐츠에 대한 교환 정책은 사용자의 콘텐츠 요청 패턴과 깊은 상관 관계를 갖고 있다. 콘텐츠 교환 정책에는 대표적으로 LFU와 LRU가 있으며 여타 파생된 정책의 근간이 된다. 두 정책은 각각 요청의 빈도수(Frequency)와 최신성(Recency)을 기준으로 콘텐츠를 선택한다. LFU(Least Frequently Used)는 현재 캐시에 있는 콘텐츠 중 가장 요청 횟수가 적은 콘텐츠를, LRU(Least Recently Used)는 가장 요청되지 오래된 콘텐츠를 우선적으로 방출한다. RND(Random) 정책은 저장된 콘텐츠들 중에서 임의로 선택하여 방출하는 기법으로, 가장 구현이 간편하다. 이 외에도 콘텐츠의 빈도수와 최신성을 동시에 고려하거나 사이즈가 큰 콘텐츠를 캐시에서 우선적으로 삭제함으로써 좀 더 다양한 콘텐츠를 캐시에 저장하는 등 여러 가지 방법이 있다.

#### 3.3.1 HC\_LRU 캐시 알고리즘

본 논문에서는 HC\_LRU 캐시 알고리즘을 제안한다. HC\_LRU 캐시 알고리즘에서는 데이터 교체 정책 및 핫 데이터와 콜드 데이터의 유지를 위하여 블룸 필터 계수를 정의한다. 이 계수는 데이터가 참조되는 횟수를 가지고 있으며, 데이터가 참조되면 1씩 증가한다. 클라이언트가 슈퍼 노드에 있는 콘텐츠를 참조하면 계수를 1증가시키고 콜드 데이터 리스트에 캐시를 할당한다. 콜드 데이터 리스트가 다 차 있으면 콜

드 데이터 중 가장 낮은 참조 계수를 가진 콘텐츠 중 가장 오래된 콘텐츠를 삭제한다. 데이터의 참조 계수가 4가 되면 핫 데이터 리스트로 이동하게 되며, 핫 데이터 리스트의 비어 있는 공간 중 끝의 가장 가까운 곳에 할당하며, 계수는 0으로 초기화 한다. 그림 7과 같이 핫 데이터 리스트가 다 차 있다면 핫 데이터 리스트 중 앞부터 참조하여 계수가 가장 낮은 것과 교체한 후 교체된 데이터는 콜드 데이터 리스트에서 참조 계수 3을 가지게 된다. HC\_LRU 알고리즘은 새로운 데이터가 요청됐을 경우 가장 참조가 적게 되고, 최근에 참조되지 않은 데이터를 교체하므로 LFU 알고리즘과 LRU 알고리즘이 결합되었다고 할 수 있다.

그림 8은 콜드 데이터 리스트가 다 차있는 상태에서 새로운 데이터 k가 참조되었을 경우 캐시 메모리를 보여준다. 그림 7에서 보이는 것과 같이 콜드 데이터 리스트가 할당되어 있다고 가정하고 새로운 데이터 k를 캐시 메모리에 할당을 하기 위해서 콜드 데이터 리스트에 있는 데이터 중 가장 낮은 참조 계수인 1을 가지는 b와 a 데이터 중, 더 이전에 참조되었던

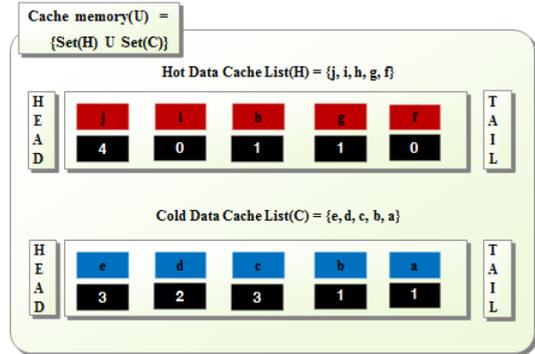


그림 7. Hot & Cold 캐시 관리  
Fig. 7. Hot & Cold Cache Management

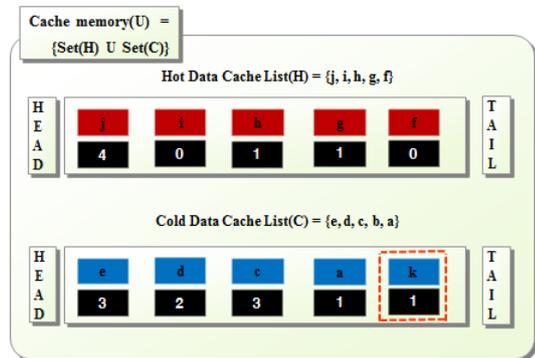


그림 8. Cold Data List에 새로운 데이터 추가  
Fig. 8. Insert a New Data in Cold Data List

데이터인 **b**를 콜드 데이터 캐시 리스트에서 삭제하고 새로운 데이터인 **k**를 리스트에 할당한다.

그림 9는 핫 데이터 리스트가 다 차있는 상태에서 콜드 데이터에 있던 데이터가 임계치를 넘는 참조를 가지면서 핫 데이터 리스트에 추가가 될 경우 캐시 메모리를 보여준다. 본 논문에서는 임계치를 4로 지정하였다. 그림 7에서 보이는 것과 같이 핫 데이터와 콜드 데이터 리스트가 할당되어 있다고 가정하고 데이터 **e**가 참조가 되었을 경우, 데이터 **e**는 핫 데이터 리스트로 이동하게 되고 참조계수는 0으로 리셋된다. 핫 데이터 리스트에는 가장 낮은 참조 계수인 0을 지니는 데이터 **i, f** 중, 더 오래전에 참조되었던 데이터인 **i**가 콜드 데이터 리스트로 이동하고 참조계수가 3으로 바뀐다.

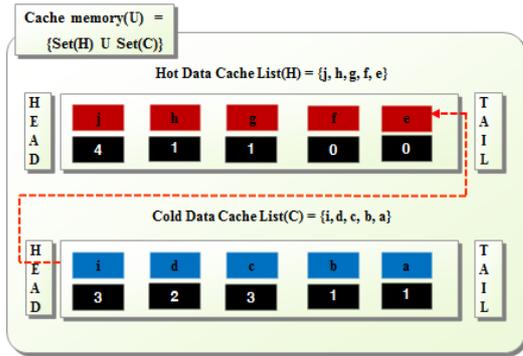


그림 9. Hot Data List에 새로운 데이터 추가  
Fig. 9. Insert a New Data in Hot Data List

#### IV. 실험

##### 4.1 시뮬레이션 환경

본 논문에서는 제안된 모델의 성능을 평가하기 위해 OMNeT++에 기반을 둔 P2P 시뮬레이터인 Oversim을 이용하였다. 또한 인터넷 토폴로지 모델을 위하여 GT-ITM을 사용하였다. 마스터 서버 1대, 관리 서버 1대, 슈퍼 노드는 4대에서 20까지 바꿔가며 성능 평가를 했다. 콘텐츠는 서로 다른 1000개를 사용했고 사이즈는 10MB로 통일했으며, 콘텐츠 요청 메시지는 60Bytes를 사용했다. 링크 용량은 100Mbps를 사용했다. 콘텐츠 요청 메시지는 초당 60회에서 100회로 하였으며, 메시지의 분포는 Zipf 법칙을 따랐다. 각각의 슈퍼 노드는 100개의 콘텐츠를 캐싱할 수 있고, 서로 다른 콘텐츠를 슈퍼 노드에 저장하였다.

표 1. 시뮬레이션 파라미터  
Table 1. Simulation Parameter.

Item	Specification
Simulation Tool	Oversim based on OmNeT++
Simulation Time	5000sec
Master Server	1
Managed Server	1
Super Node	4~20
Node(Client) Number	100~1,000
Put content Number per Super Node	100
Contents Number	1,000
Contents Popularity	Zipf-mandelbrot
Contents Request	60~100 request/sec
Average Link Bandwidth	100Mbps

##### 4.2 성능 평가

MST 알고리즘을 적용하기 위한 슈퍼 노드간의 가중치 값은 여러 파라미터들이 사용될 수 있다. 본 논문에서는 언더레이 네트워크 홉(hop)수와 전달 지연(delay)을 이용하여 가중치 값을 계산하였다.

슈퍼 노드 *i, j* 사이의 delay를  $RTT_{ij}$ , 홉 수를  $Hop_{ij}$ 라고 할 때, 두 노드 사이의 가중치 값의 Score Function은 다음과 같다.

$$Func(i, j) = \alpha(RTT_{ij}) + (1 - \alpha)(Hop_{ij}) \quad (0 \leq \alpha \leq 1) \quad (1)$$

그림 10은  $\alpha$  값에 따른 평균 응답시간의 변화를 10회 반복한 값이다. 실험에 따르면  $\alpha$  값이 0.8일 때 가장 빠른 응답시간을 보이고 있다.  $\alpha$  값이 1인 RTT만을 고려한 가중치 값이  $\alpha$  값이 0인 홉수만을 고려한 값보다 빠른 응답시간을 보인다. 하지만  $\alpha$  값이 0.8일

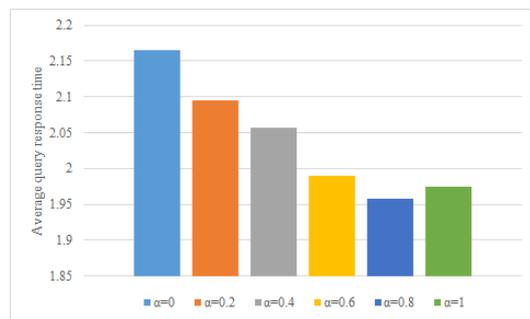


그림 10.  $\alpha$  값에 따른 평균 응답시간  
Fig. 10. Average Query Response Time for  $\alpha$  value

때 가장 빠른 응답시간을 가지는 것으로 보아 RTT 값과 홉수 간의 적절한 가중치 값을 계산하는 것도 중요한 것으로 생각된다.

#### 4.2.1 캐시 알고리즘에 따른 Hit Ratio

그림 11은 캐싱 정책에 따른 적중률을 나타내고 있다. 캐싱 정책은 FIFO, LFU, LRU, 그리고 본 논문에서 제안한 HC\_LRU를 사용하였다. FIFO는 먼저 들어온 데이터 블록을 대체하는 정책이고, LFU는 가장 빈도수가 적은 블록을 대체하는 정책이며 LRU는 가장 오래 전에 참조된 블록을 대체한다. HC\_LRU는 LRU를 기반으로 참조된 빈도를 추가하여 선호도가 높은 데이터가 대체 되는 것을 한번 더 보호하는 정책이다.

시뮬레이션에서는 초당 요청 수를 60에서 100으로 변경해가며 캐시 Hit Ratio를 측정하였다. 제안한 캐싱 정책이 FIFO, LFU, LRU 보다 높은 Hit Ratio를 보이고 있다. Hit Ratio가 높다는 것은 요청한 콘텐츠가 슈퍼 노드에 존재할 확률이 높다는 것을 의미하므로 제안한 캐싱 정책이 트래픽 관리를 더욱 효율적으로 할 수 있다는 것을 의미한다.

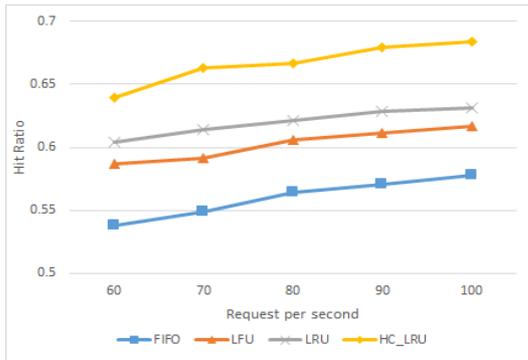


그림 11. 캐싱정책에 따른 캐시 적중률  
Fig. 11. Hit Ratio for Caching Policy

#### 4.2.2 평균 쿼리 응답시간

그림 12는 슈퍼노드 수에 따른 평균 쿼리 응답시간을 보여주고 있다. 제안한 캐싱 알고리즘을 사용한 CDN이 가장 빠른 평균 쿼리 응답시간을 보이는 것을 알 수 있다. MST와 SPT 알고리즘만을 적용한 CDN은 슈퍼노드의 수에 크게 영향을 받지 않는다. 그 이유는 두 방안은 슈퍼노드에서 캐싱을 하지 않고 마스터 서버에서 콘텐츠를 바로 전송 받기 때문이다. 그러므로 슈퍼노드의 수가 증가하더라도 초당 콘텐츠 요청 수가 100개로 일정했기 때문에 평균 쿼리 응답시

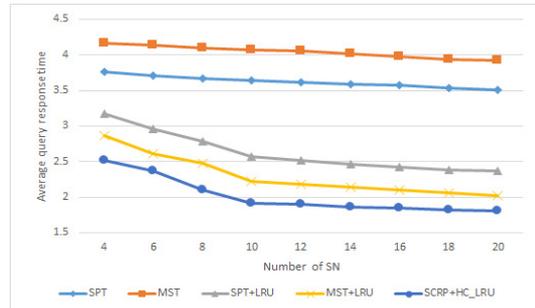


그림 12. 슈퍼노드 수에 따른 평균 쿼리 응답시간  
Fig. 12. Average Query Response Time for Number of Super Node

간은 큰 차이를 보이지 않는다. SPT+LRU, MST+LRU, SCRCP+HC\_LRU를 적용한 CDN은 슈퍼 노드의 수가 10개 이상일 때는 평균 쿼리 응답시간이 크게 차이가 나지 않는다. 슈퍼 노드는 서로 다른 100개의 콘텐츠를 캐싱하기 때문에 콘텐츠 수가 100개가 되면 모든 콘텐츠가 각 슈퍼 노드에 분산되어 있기 때문에 콘텐츠 요청이 마스터 서버까지 가지 않으므로 평균 쿼리 응답시간이 차이가 나지 않는 것으로 보인다.

그림 13은 클라이언트 수에 따른 평균 쿼리 응답시간을 보여준다. 제안한 캐싱 정책을 사용한 CDN이 가장 좋은 응답 시간을 보인다. 클라이언트의 수가 증가하더라도 초당 콘텐츠 요청수가 100개로 일정하므로 MST와 SPT를 적용한 CDN에서는 평균 쿼리 응답시간에 큰 영향을 받지 않는다. SPT+LRU를 적용한 CDN은 MST+LRU와 SCRCP+HC\_LRU를 적용한 CDN 보다 클라이언트 수가 증가 할수록 평균 쿼리 응답 시간의 기울기가 높아지는 것을 보인다. 캐시를 적용한 SPT 알고리즘이 MST 알고리즘보다 클라이언트의 수에 영향을 받는 것을 알 수 있다. 이는 MST

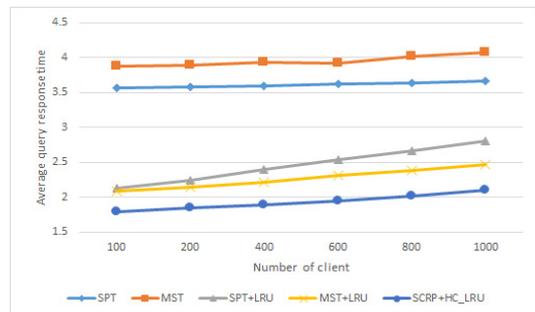


그림 13. 클라이언트 수에 따른 평균 쿼리 응답시간  
Fig. 13. Average Query Response Time for Number of Client

알고리즘을 사용할 경우 트래픽 병목현상을 방지함으로써 효율적으로 트래픽을 관리하여 클라이언트가 증가하더라도 SPT 알고리즘보다 크게 변동 없는 평균 쿼리 응답시간을 가지는 것이다.

그림 14는 초당 콘텐츠 요청 수에 따른 평균 쿼리 응답 시간을 보여준다. 여기서도 제안한 캐싱 정책을 사용한 CDN이 가장 빠른 평균 쿼리 응답시간을 보인다.

SPT와 MST만을 적용한 CDN은 캐싱정책을 적용한 CDN보다 가파른 평균 쿼리 응답시간의 상승을 보인다. 또한 SPT+LRU를 적용한 CDN이 MST+LRU와 SCRP+HC\_LRU를 적용한 CDN보다 가파른 평균 쿼리 응답 시간의 상승을 보인다.

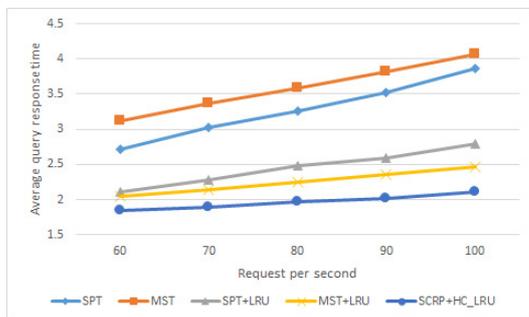


그림 14. 초당 콘텐츠 요청 수에 따른 평균 쿼리 응답시간  
Fig. 14. Average Query Response Time for Content Request per second

#### 4.2.3 최대 링크 이용률

그림 15는 초당 콘텐츠 요청 수에 따른 최대 링크 이용률을 보여준다. 최대 링크 이용률이란 시뮬레이션 시간 동안 각 링크 사용이 가장 많았을 경우의 이용률을 뜻한다. 캐싱을 사용하지 않았을 경우 SPT 알고리즘이 MST 알고리즘보다 더 낮은 최대 링크 이용률을 보인다. 최대 링크 이용률이 낮을수록 효율적인 트래

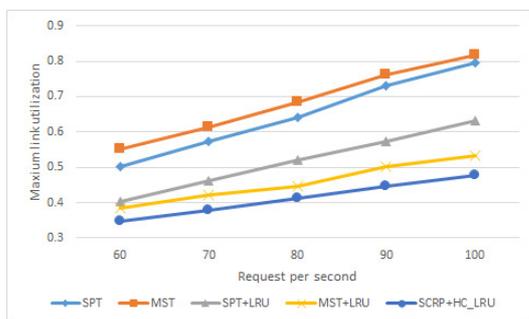


그림 15. 초당 콘텐츠 요청 수에 따른 최대 링크 이용률  
Fig. 15. Maximum Link Utilization for Content Request per second

픽 관리가 된다는 것을 뜻하므로 제안한 캐싱 정책을 사용한 CDN이 트래픽 관리에 가장 좋은 성능을 가지는 것을 의미한다.

## V. 실험

ICT 시장은 스마트폰의 빠른 보급과 네트워크의 브로드밴드화(Wi-Fi, LTE 등)로 인하여 대용량의 고품질 동영상 콘텐츠와 음원 콘텐츠의 데이터 전송량이 급속하게 증가하고 있다. 이제는 통화 품질의 우수성이 아닌 YouTube, Netflix, 아프리카 TV 등과 같은 멀티미디어 콘텐츠를 소비자가 허용할 수 있는 적절한 가격으로 끊임없이 전달해줄 수 있는나는 QoE(Quality of Experience)가 사용자가 통신서비스 사업자를 선택하는 기준이 되고 있다. 그렇다고 통신 서비스사업자들이 자사 콘텐츠도 아닌 타사 대용량 콘텐츠 전송을 위해 자사 코어망 용량 증설을 위해서 투자만 한다면 수익율은 크게 떨어질 수 밖에 없다. 또한 글로벌 고객들을 대상으로 서비스를 제공하는 글로벌 콘텐츠 제공업체들도 국가별, 통신서비스사업별로 품질이 다른 데이터 전송 네트워크의 QoS를 일정 수준 이상으로 보장할 수도 없다. 이런 문제점을 해결할 수 있는 솔루션이 바로 콘텐츠 전송 네트워크이다.

제안하는 콘텐츠 전송 네트워크는 콘텐츠 및 서비스 전달에 최적화된 네트워크로 콘텐츠 인지 및 저장 이 가능하며, 네트워크 사업자와 인터넷 사업자 간 개방형 에코환경 구현을 통해 수익 공유가 가능한 네트워크이다. 서비스 및 네트워크 계층이 결합되어지고 SCRP 라우팅 기능을 도입하여 동일 콘텐츠가 중복으로 전달되는 것을 방지함으로써, 네트워크 자원을 효율적으로 사용하는 기능을 제공한다. 또한, 클라이언트가 요청한 데이터가 캐싱 서버인 슈퍼 노드에 존재 하다면 스토리지 서버까지 가는 네트워크 비용을 줄이고 데이터 요청을 분산한다는 장점이 있다. 제안한 HC\_LRU 캐시 알고리즘은 핫 데이터, 콜드 데이터를 구분함으로써 핫 데이터를 캐싱 서버에 오래 상주할 수 있게 하도록 설계하였다. 시간 및 참조 지역성의 원리에 의해 같은 시간동안 접근이 자주 일어나는 핫 데이터를 저장함으로써 캐싱된 데이터를 효율적으로 활용한다. 또한 핫 데이터를 유지함으로써 캐시 적중률을 높였고, 그 결과 디스크의 I/O 연산을 줄일 수 있었다.

본 논문에서는 제안하는 구조인 SCRP+HC\_LRU를 적용한 CDN과 SPT를 적용한 CDN, MST를 적용

한 CDN, SPT+LRU를 적용한 CDN, MST+LRU를 적용한 CDN을 평균 요청 응답시간 및 최대 링크 이용률 면에서 각각 비교하였고, 캐싱 방안에 대해서도 제안한 HC\_LRU와 FIFO, LFU, LRU에 대해 시뮬레이션 하였다. 그 결과 제안하는 HC\_LRU 알고리즘이 캐시 적중률에서 FIFO, LFU, LRU보다 높은 것을 확인할 수 있었다. 또한 제안하는 콘텐츠 전송 네트워크가 평균 요청 응답시간 및 최대 링크 이용률에서 좋은 성능을 보였고, 이로 인해 외부로 나가는 트래픽을 감소시키고 병목현상을 방지하여 효율적인 네트워크 자원 사용이 가능할 것을 확인할 수 있었다. 향후, 모바일 동영상에 확산되며 급증하는 모바일 트래픽을 고려한 연구와 제안한 콘텐츠 전송 네트워크에 클라우드 기술을 도입한 연구를 통해 모바일 클라우드 CDN 기술에 관한 연구를 진행할 것이며 이는 향상된 CDN 기술 개발에 활용될 수 있을 것이다.

### References

- [1] X. Hei, et al., "A measurement study of a large-scale P2P IPTV system," *IEEE Trans. Multimedia*, vol. 9, no. 8, pp. 1672-1687, 2007.
- [2] Y. Liu, Y. Guo, and C. Liang, "A survey on peer-to-peer video streaming systems," *Peer-to-peer Netw. Appl.*, vol. 1, no. 1, pp. 18-28, 2008.
- [3] H. Schulze and K. Mochalski, *Internet Study 2008/2009*, IPOQUE Report, vol. 37, pp. 351-362, 2009.
- [4] J. Seedorf, S. Kiesel, and M. Stiernerling, "Traffic localization for P2P-applications: the ALTO approach," *IEEE Peer-to-Peer Comput.*, 2009, pp. 171-177, Sept. 2009.
- [5] B. Mathieu and P. Paris, "A topology-aware P2P video streaming system," *IEEE GHIS'09*, pp. 1-8, Jun. 2009.
- [6] Y. Tian, et al., "Improving stability for peer-to-peer multicast overlays by active measurements," *J. Syst. Architecture*, vol. 54, no. 1, pp. 305-323, 2008.
- [7] F. Wang, J. Liu, and Y. Xiong, "Stable peers: Existence, importance, and application in peer-to-peer live video streaming," *INFOCOM 2008*, Phoenix, AZ, Apr. 2008.
- [8] J. Park, "Efficient parent peer selection method in a wireless P2P system," *J. KICS*, vol. 39, no. 12, pp. 870-872, Dec. 2014.
- [9] T. A. Neves, et al., "Optimization in content distribution networks," in *Proc. Int. Conf. Eng. Optimization*, Rio de Janeiro, Brazil, Jun. 2008.
- [10] C. Huang, et al., "Understanding hybrid CDN-P2P: why limelight needs its own Red Swoosh," in *Proc. NOSSDAV ACM*, pp. 75-80, May 2008.
- [11] G. Pierre and M. Van Steen, "Globule: a collaborative content delivery network," *IEEE Commun. Mag.*, vol. 44, no. 8, pp. 127-133, 2006.
- [12] S. Gitzenis, G. S. Paschos, and L. Tassiulas, "Asymptotic laws for content replication and delivery in wireless networks," *IEEE INFOCOM*, pp. 531-539, Mar. 2012.
- [13] S. H. Kong and J.-Y. Lee, "Effective contents delivery system using service adaptive network architecture(SaNA)," *J. KICS*, vol. 39, no. 6, pp. 406-413, Jun. 2014.
- [14] S. Ratnasamy, et al., "A scalable content-addressable network," *ACM*, vol. 31, no. 4, pp. 161-172, 2001.
- [15] B. Godfrey, et al., "Load balancing in dynamic structured P2P systems," *INFOCOM 2004*, vol. 4, pp. 2253-2262, Mar. 2004
- [16] B. Azimdoost, C. Westphal, and H. R. Sadjadpour, "On the throughput capacity of information-centric networks," in *Proc. 25th Int. Teletraffic Congress 2013*, pp. 1-9, Sept. 2013.
- [17] H. Che, Y. Tung, and Z. Wang, "Hierarchical web caching systems: Modeling, design and experimental results," *IEEE J. Sel. Areas in Commun.*, vol. 20, no. 7, pp. 1305-1314, 2002.
- [18] C. Fricker, P. Robert, and J. Roberts, "A versatile and accurate approximation for LRU cache performance," in *Proc. 24th Int. Teletraffic Congress*, no. 8, Sept. 2012.
- [19] S. Han, H. Park and T. Kwon, "Shelf-Life time based cache replacement policy suitable for web environment," *J. KICS*, vol. 40, no. 6, pp. 1091-1101, Jun. 2015.

이 형 옥 (Hyung-ok Lee)



2008년 2월 : 전남대학교 컴퓨터 정보통신공학과 석사  
2015년 2월 : 전남대학교 전자 컴퓨터공학부 박사  
2015~현재 : 전남대학교 Post doc.  
<관심분야> 컴퓨터 네트워크, 통신 프로토콜

남 지 승 (Ji-seung Nam)



1992년 2월 : Univ. of Arizona 전자공학과 박사  
1992년~1995년 : 한국전자통신연구원 선임연구원  
1995년~현재 : 전남대학교 컴퓨터 공학과 교수  
<관심분야> 통신 프로토콜, 인터넷 실시간 서비스

강 미 영 (Mi-young Kang)



2001년 2월 : 전남대학교 컴퓨터공학과 석사  
2006년 2월 : 전남대학교 컴퓨터공학과 박사  
<관심분야> 라우팅 프로토콜, 인터넷 실시간 서비스