

---

# 로컬 변환에 의한 오버레이 멀티캐스트 트리의 성능 최적화 기법

## Performance Optimization Technique for Overlay Multicast Trees by Local Transformation

---

강미영, 이형욱, 곽용완, 남지승  
전남대학교 컴퓨터공학과

Mi-Young Kang(kmy2221@yahoo.co.kr), Hyung-Ok Lee(narcis@freechal.com),  
Young-Wan Kwag(youngwan38@empal.com), Ji-Seung Nam(jsnam@chonnam.ac.kr)

---

### 요약

오버레이 멀티캐스트는 하드웨어적인 인프라 구축 없이도 시스템의 자원과 네트워크 대역폭을 효율적으로 사용할 수 있는 기법이며 중간 노드의 이탈이 발생하게 될 경우 멀티캐스트 트리를 재구성 한다. 그러나 빈번한 멀티캐스트 트리의 재구성은 심각한 성능 저하를 가져오게 된다. 본 논문에서는 이러한 성능저하를 보완하기 위해 각 자식 노드들로부터 소스 노드에게 주기적으로 피드백 되어오는 정보(RTCP-probing)를 기반으로 트리 성능 최적화 알고리즘을 제안한다. 제안된 모델은 서비스 하는 부모노드가 트리의 성능을 저하 시키는 원인으로 판단되어질 때 수행하는 메커니즘이다. 메커니즘을 수행하여 성능 최적화 트리를 구성함으로써 전체적인 서비스 트리의 성능을 향상시켰다. 시뮬레이션 결과를 통해 제안된 모델뿐만이 아니라 기존의 기법들에 적용하였을 때도 효과적인 방안을 보여주고 있다.

■ 중심어 : | 오버레이 멀티캐스트 | 스왑 | 로컬 변환 | ChiPaRoS |

### Abstract

Overlay Multicast is an effective method for efficient utilization of system resources and network bandwidth without a need for hardware customization. Multicast tree reconstruction is required when a non-leaf node leaves or fails. However frequent multicast tree reconstruction introduces serious degradation in performance. In this paper, we propose a tree performance optimization algorithm to solve this defect by using information(RTCP-probing) that becomes a periodic feedback to a source node from each child node. The proposed model is a mechanism performed when a parent node seems to cause deterioration in the tree performance. We have improved the performance of the whole service tree using the mechanism and hence composing an optimization tree. The simulation results show that our proposal stands to be an effective method that can be applied to not only the proposed model but also to existing techniques.

■ keyword : | Overlay Multicast | SWAP | Local Transformation | ChiPaRoS |

---

## I. 서론

IP 멀티캐스트가 실제로 사용되기 위해서는 각 라우

터에서 IP 멀티캐스트 패킷을 처리 할 수 있도록 구현되어야 한다. 그러나 라우터의 구현 및 확장, 혼잡 제어, 신뢰성 있는 전송 등의 문제로 인해 실제 인터넷에서는

적용되어지지 못하고 있는 실정이다.

IP 멀티캐스트의 대안으로 제시되어지는 오버레이 멀티캐스트(Overlay Multicast)는 기존 라우터들을 변경할 필요 없이 응용 계층에서 라우팅을 실시하며 멀티캐스트의 기능을 각 노드(end-host)에 구현한 것으로서 기존 노드가 자신이 받은 패킷을 자신의 다음 노드에게 전달해주는 방법을 사용한 어플리케이션 소프트웨어의 설치만으로 멀티캐스트를 사용할 수 있는 기법이다.

그러나 오버레이 멀티캐스트는 트리를 구성하고 있는 중간 노드가 멀티캐스트 그룹을 이탈할 하였을 경우 트리를 재구성 해야만 된다는 문제점을 가지고 있다. IP 멀티캐스트는 트리를 구성하고 있는 중간 노드들이 라우터이기 때문에 이와 같은 문제를 고려할 필요가 없지만 오버레이 멀티캐스트는 중간 노드를 포함한 모든 노드들이 언제든지 멀티캐스트 그룹을 자유롭게 이탈할 수 있는 노드(end host)로 구성이 된다는 점에서 새롭게 생겨난 문제점인 것이다.

기존의 오버레이 멀티캐스트에 관한 연구는 효율적인 멀티캐스트 트리 구성을 위해 대부분 라우팅 프로토콜의 설계 분야에서 이루어져왔다[1-9]. 반면 멀티캐스트 트리의 성능 최적화를 위해 트리를 재구성 하는데 중점을 둔 연구는 상대적으로 많이 이루어 지지 않았는데 이는 멀티캐스트 환경에서 하고자 하는 서비스의 종류에 따라 그 비중이 달라지는데도 한 원인이 있다.

멀티캐스트를 이용하여 일반 데이터를 전송하는 경우 최적의 경로를 통해 모든 데이터가 전송되는 것이 일차적인 목표이지만, 실시간 방송 서비스와 같은 스트리밍 서비스를 하게 되는 경우에는 중간 노드들의 이탈에 의한 트리 구성의 잦은 변화로 발생하게 되는 성능 저하에 각 노드(end host)들이 최대한 영향을 받지 않도록 QoS (Quality of Service)를 보장 하는 것이 최우선이기 때문에 어느 정도 패킷 손실이 있을 지라도 빠른 시간 안에 성능 최적의 트리를 재구성 해야만 한다.

본 논문에서는 트리 구성 알고리즘으로 mTBCP[10]와 전통적인 TBCP 그리고 HMTP를 사용하여 멀티캐스트 트리를 각각 구성하였다.

트리를 구성한 후 노드의 잦은 이탈로 인한 성능 저하를 보완하는 로컬 변환 메커니즘을 적용하여 성능 최

적의 트리를 재구성하는 오버레이 멀티캐스트 성능 최적화 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 기술하며 3장에서는 제안된 모델의 개요 및 전체적인 동작 절차를 설명한다. 4장에서는 시뮬레이션을 통하여 기존 방안과의 비교 및 성능을 평가하고 5장에서 결론을 맺는다.

## 1. 관련연구

Ondrej Dolejs와 Zdenek Hanzal는 오버레이 멀티캐스트 트리 구성과 트리 구성의 최적 상태에 이르는 특성에 대해 잘 설명하고 있다[11]. 만약에 트리를 구성할 때 비용(cost)에 가장 큰 주안점을 둔다면 최소 신장 트리(minimum spanning tree) 알고리즘인 Kruskal 또는 Prim 알고리즘 등의 영향을 받을 것이다. 반면에 지연(delay)에 주안점을 두고 트리를 구성한다면 최단경로 우선(shortest path tree) 트리 알고리즘인 Dijkstra와 Bellman-Ford 알고리즘이 사용될 것이다. 알고리즘들은 또한 중앙형(centralized)으로 실행하느냐 아니면 분산형(distributed)으로 실행되느냐에 따라 두 가지 경우 모두 약간의 문제점을 안고 있다는 사실을 알게 되었다. 중앙형 알고리즘의 문제점은 중앙 노드(central node)의 링크가 Failure 되었을 때 일어나게 되는 심각한 과부하와 네트워크 토폴로지에 대한 전반적인 정보가 필요하다는 것이다.

전통적인 TBCP는 오버레이 멀티캐스트 라우팅 트리를 구성하는데 분산 알고리즘을 사용하였다[12]. 그렇지만 본 연구에 있어서 성능 최적화 부분에 대한 단점을 발견하였으며 개인 방송 서비스를 위한 신뢰성 있는 멀티캐스팅 적용에 적절하지 않다는 사실을 확인하였다. TBCP 연구에 있어서 가장 큰 단점은 사실상 부모 노드는 자식노드의 정보를 직접적으로 연결된 자식 노드 (First Layer) 정보만을 가지고 있다는 점이다. 즉, 부모노드와 New 노드(Join 하려는 노드) 사이의 정보; 자식노드들과 New 노드 사이의 정보; 자식노드들과 부모노드 사이의 정보만을 유지 하고 있다. 즉, TBCP 알고리즘은 깊이를 가진 하위 자식 노드들에 대한 정보를 가지고 있지 않다.

오버레이 멀티캐스트 기반 알고리즘은 중간 노드의 이탈이 발생하게 될 경우 멀티캐스트 트리를 재구성 하게 되는데 TBCP의 경우 자식 노드들에 대한 정보가 빈약하므로 중간 노드들의 탈퇴로 인한 빈번한 멀티캐스트 트리의 재구성은 심각한 성능 저하를 가져오게 된다. 따라서 최적의 서비스를 제공 하는데 있어서 한계성을 가지고 있다.

본 논문에서 제안한 오버레이 멀티캐스트 트리는 PPL(Potential Parent List) 정보를 관리하고 있다. 즉, 부모노드가 될 수 있는 모든 노드들과 New 노드(Join하려는 노드) 사이의 정보를 가지고 있으므로 제안한 알고리즘은 깊이를 가진 하위 자식노드들에 대한 정보를 가지고 있게 된다. 따라서 만약 중간 노드의 이탈이 발생하게 될 경우 PPL(Potential Parent List) 정보를 이용하여 멀티캐스트 트리를 재구성 하게 되며 빈번한 트리 재구성으로 인한 성능 저하 부분은 로컬변환 메커니즘을 적용하여 수행함으로써 성능 최적의 서비스를 제공하였다.

## II. 본 론

### 1. 오버레이 멀티캐스트 모델

오버레이 멀티캐스트 개인 방송 시스템은, 효율적으로 디자인 되어 졌으며 3가지 주요 기술(오버레이 기술, 멀티캐스트 기술, 개인방송 기술)을 이용함으로써 여러 가지 한계를 극복하도록 구현 되었다. [그림 1]은 오버레이 방송 시스템의 프레임워크를 보여주고 있다. [그림 1]에서 방송 서비스를 받고자 하는 New 노드(Join을 원하는 노드)는 웹 서버로부터 서비스 받고자 하는 채널정보를 선택하게 되며 개인 방송 시스템은 방송 서비스를 수행하기 위한 작업을 준비하게 된다. 다음 단계로 웹 서버는 해당 서비스 채널인 방송서비스 제공자(Broadcasting Service Provider=소스루트)와 New 노드의 연결을 중재하여 준다. 소스루트는 New 노드에게 가장 빠르게 연결하여 최적의 서비스를 받을 수 있는 잠재적 부모노드(Potential Parent List) 테이블 리스트를 보내준다. New 노드는 전송 받은 테이블 리스트 정

보를 가지고 각각의 잠재적 부모 노드에 대한 RTT(Round-Trip-Time) 체크루틴을 수행함으로써 최적의 부모노드를 선택하게 되며, 부모노드로부터 오버레이 멀티캐스트 기반의 방송서비스를 받게 된다.

제안 시스템에서 New 노드가 실제적으로 접속(access)하여 서비스를 받는 세션에 대한 구조를 자세히 살펴보면, 먼저 New 노드는 New\_B\_REQ(New Client Broadcast Request) 메시지를 방송서비스 제공자(Broadcasting Service Provider)에 전송한다. 그러면 mTBCP 기반 오버레이 멀티캐스트 트리 관리 모듈은 방송서비스 제공자에 속한 모듈로서 New\_B\_REQ를 받은 방송서비스 제공자는 mTBCP 모듈을 이용하여 New 노드에게 서비스가 가능한 부모 노드 리스트인 PPL을 응답 메시지와 함께 보내게 된다. PPL 정보를 받은 New 노드는 RTT 체크 루틴을 수행하여 최적의 부모노드를 선택함으로써 성공적으로 Join하게 된다. New 노드는 선택한 부모 노드에 성공적으로 Join 했다는 메시지를 방송서비스 제공자에 보내게 되며, 개인방송국의 서브 모듈인 mTBCP기반 오버레이 멀티캐스트 트리 관리 모듈은 PPL 정보를 업데이트 한다.

New 노드의 Join은 기존에 제안하였던 mTBCP[11] 알고리즘을 사용하여 수행한다.

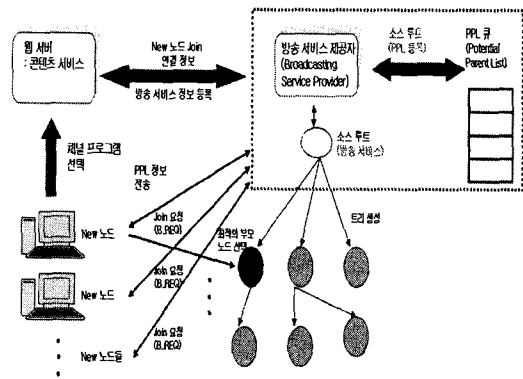


그림 1. 개인 방송 시스템 전체 프레임 워크

### 2. 잠재적 부모 노드(PPL) 구조

오버레이 멀티캐스트 트리에서 소스루트는 New 노드가 멀티캐스트 그룹에 가입하고자 할 때 일차적으로

접속하게 되는 랑데부 (rendezvous) 기능을 수행한다. 이때 소스루트는 방송 서비스 그룹에 가입할 수 있는 정보를 주기 위해 잠재적 부모 노드 목록인 PPL(Potential Parent List)를 New 노드에게 전송해 주는데 이 PPL 구조는 아래 그림과 같다.

PPL은 방송 서비스를 제공해 줄 수 있는 잠재적 부모 노드의 IP 주소, 상위 부모 노드를 거쳐 소스루트까지 가지는 RTT값과 여유 Out-degree로 구성된다.

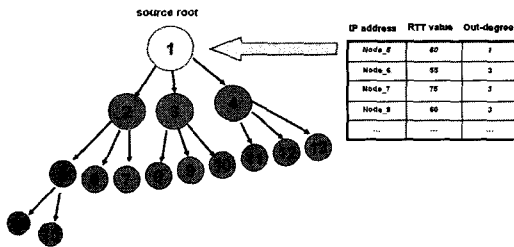


그림 2. 잠재적 부모 노드 PPL 구조

New 노드는 잠재적 부모 노드들 중 최적의 부모 노드를 선택하거나 기존의 노드가 방송 서비스 그룹에서 떠나는 경우 호스트들이 가지고 있는 정보가 변경될 때마다 소스루트에 PPL Update 메시지를 보냄으로써 항상 최신의 트리 정보를 유지할 수 있게 한다.

아래 그림은 방송하고자 하는 소스루트가 트리를 구성할 때 제일 먼저 PPL 모듈에 자신을 등록 시키는 과정을 보여주고 있다.

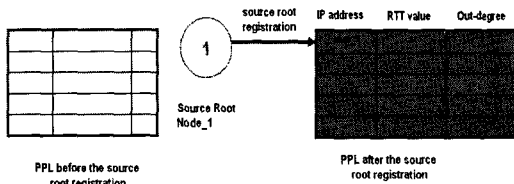


그림 3. 트리 구성 초기화

### 3. 성능 최적화 계획

본 논문에서 제안하는 성능 최적화 기법인 로컬 변환 메커니즘은 피드백 되어오는 정보(RTCP-probing)에 의해 부모노드들의 성능/능력 등의 모니터링이 가능하며 부모노드로서의 능력이 현저히 떨어진다고 판단되

어질때 제안하는 메커니즘이 자동으로 수행되어진다.

다음은 메커니즘이 동작되기 위해 기본적으로 보고 되는 피드백 정보의 지연시간 산출 공식이다.

◆RTCP report delays at instantaneous times  
→ Delay rates

◆Delay is the sum up of:

- Packetization delay →  $d_p$
- Processing delay →  $d_c$
- Queuing delay →  $d_q$
- Transmission delay →  $d_t$
- Propagation delay →  $d_o$

◆Therefore, total delay,  $D_r = d_p + d_c + d_q + d_t + d_o$

◆ Instantaneous node delay,  $D_i = \text{arrive time}(t_a) - \text{previous time}(t_p)$   
→  $D_i = t_a - t_p$

◆ Delay rate = total arrive time / total number of nodes

그림 4. 피드백 정보 지연시간 산출

최초로 구성된 트리는 가장 최적화 된 트리의 상태를 유지하고 있다. 그러나 오버레이 멀티캐스트의 특성을 반영하는 자유로운 노드의 임의 이탈은 시간이 지날수록 최적화 트리를 유지하지 못하게 되는 상황을 유발시킨다. 이는 주기적으로 보고되어지는 RTCP-probing 정보에 의해 발견할 수 있으며 이때, 제안하는 알고리즘을 적용하였을 때 전체적인 서비스 성능이 향상된 모습을 시뮬레이션을 통하여 결과를 확인하였다. 비효율적인 서비스 노드는 RTCP-probing에 의해 보고되는 피드백 정보에 의해 감지되며 방송 서비스를 받고 있는 자식노드들이 모두 부모 노드보다 성능이 더 좋다고 판단되어졌을 경우, 로컬 변환 모듈인 ChiPaRoS(Child-Parent-Role-Swapping) 알고리즘을 수행할 준비 상태로 진입한다. ChiPaRos 알고리즘은 아래의 2가지 상황에 대해서만 스왑 모듈을 가동한다.

• Step 1 : Child-Parent-Role-Swapping 수행

RTCP-probing의 보고에 의해 효율적인 부모 노드로서의 서비스 역할을 하지 못한다고 판단이 되면 자식노드들의 상태(서비스 수행능력 및 여유 Out-degree 등)를 판단하게 된다. 자식노드들 중 최적의 성능을 가지고 있는 노드와 여유 Out-degree가 가장 많은 자식노드가 동일한 노드라면 곧바로 부모 노드와 자식 노드 사이의 스왑 모듈을 수행한다.

- Step 2 : Child-Parent-Role-Swapping 수행  
 자식노드들 중 최적의 성능을 가지고 있는 노드와 여유 Out-degree가 가장 많은 자식 노드가 동일한 노드가 아닐 때 여유 Out-degree가 가장 많은 자식 노드와 부모 노드사이에 스왑 모듈을 수행한다.

ChiPaRoS 알고리즘 수행의 예를 [그림 5]과 [그림 6]에서 자세히 보여주고 있다.

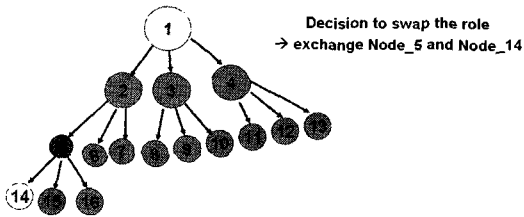


그림 5. ChiPaRoS 수행(Step 1)

만약 소스루트(노드 1)에서는 자식노드들인 노드 14, 노드 15, 노드 16보다 부모노드인 노드 5의 성능이 좋지 않음으로 인해서 전체적인 서비스가 좋지 않은 상태일 경우 : 최적의 성능을 가지고 있는 노드와 여유 Out-degree가 가장 많은 자식노드가 노드 14로서 동일한 노드라면 Child-Parent-Role-Swapping 알고리즘을 수행한다.

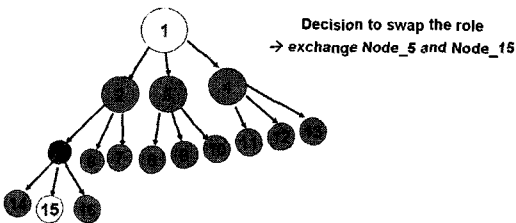


그림 6. ChiPaRoS 수행(Step 2)

만약 소스루트(노드 1)에서는 자식노드들인 노드 14, 노드 15, 노드 16보다 부모노드인 노드 5의 성능이 좋지 않음으로 인해서 전체적인 서비스가 좋지 않은 상태로 진행되는 경우, 최적의 성능을 가지고 있는 노드는 노드 14 이며 여유 Out-degree가 가장 많은 자식노드는 노드 15 로서 동일한 노드가 아니라면 여유

Out-degree가 가장 많은 자식노드와 부모 노드 사이에 스왑이 발생하는 Child-Parent-Role-Swapping 알고리즘을 수행한다.

#### 4. 성능 평가

본 논문에서는 제안된 모델의 성능을 평가하기 위해 네트워크 시뮬레이터인 ns2[12]를 사용하였다. 토폴로지 생성은 GT\_ITM의 Transit-Stub 모델을 적용하여 1,000개 노드로 구성된 네트워크 토폴로지를 생성 하였으며 한 개의 노드가 가지는 최대 Out-degree 값은 100Mbps의 네트워크 환경에서 25Mbps 정도의 대역폭이 필요한 HD 급 영상을 서비스 한다고 가정하여 3 으로 정의 하였다.

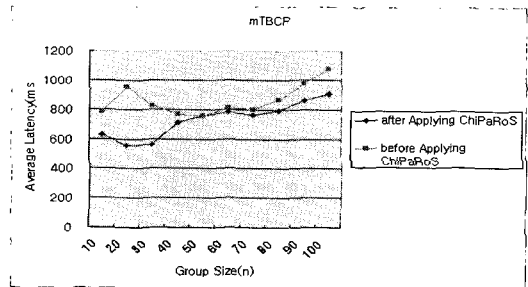


그림 7. 트리 재구성 시 평균 Latency 변화 (mTBCP)

제안된 알고리즘의 성능 비교를 위해 트리구성은 mTBCP 기법과 관련 연구에서 살펴보았던 전통적인 TBCP 기법을 사용하여 멀티캐스트 트리를 구성하였다.

첫 번째 시뮬레이션 [그림 7]은 mTBCP를 사용하여 트리를 구성한 후 그룹의 크기를 증가 시켜가면서(10~100) 노드의 이탈을 랜덤하게 발생하게 하였다. 그런 다음 트리의 재구성 상태를 모니터링 하였다. 이때 로컬 변환 모듈을 가동 시켰을 때(after Applying ChiPaRoS)와 로컬 변환 모듈을 가동시키지 않았을 때(before Applying ChiPaRoS)의 트리의 재구성 상태를 트리의 평균 Latency 값을 측정하여 비교하였다. 로컬변환 모듈을 수행하였을 때 훨씬 좋은 트리의 재구성 상태를 확인할 수 있었으며, 이는 전체적인 서비스의 성능이 향상됨을 의미한다.

두 번째 시뮬레이션 [그림 8]은 전통적인 TBCP를 사

용하여 트리를 구성한 후 그룹의 크기를 증가 시켜가면서(10~100) 노드의 이탈을 랜덤하게 발생하게 하였다. 그런 다음 트리의 재구성 상태를 모니터링 하였다. 이때 로컬 변환 모듈을 가동 시켰을 때(after Applying ChiPaRoS)와 로컬 변환 모듈을 가동시키지 않았을 때(before Applying ChiPaRoS)의 트리의 재구성 상태를 트리의 평균 Latency 값을 측정하여 비교하였다. 제안한 로컬변환 모듈을 수행하였을 때 훨씬 좋은 트리의 재구성 상태를 확인할 수 있었으며, 이는 전체적인 서비스의 성능이 향상됨을 의미한다.

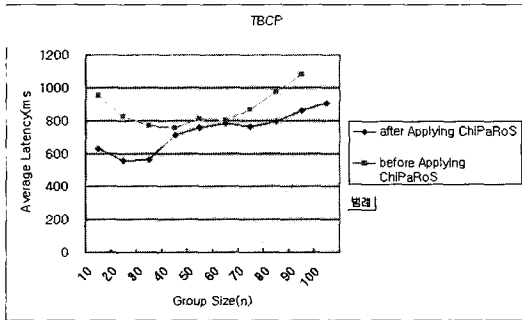


그림 8. 트리 재구성 시 평균 Latency 변화 (TBCP)

오버레이 멀티캐스트 기반으로 트리를 구성하였을 때 중간 노드들의 이탈에 따른 트리 재구성으로 인한 성능 저하 부분은 발생 할 수밖에 문제이다.

따라서 본 논문에서는 다양한 트리구성 기법에 제안한 성능최적화기법인 ChiPaRoS 알고리즘을 적용하여 수행함으로써 노드 이탈에 따른 트리 재구성으로 인한 성능 저하 부분을 보완하여 향상된 서비스를 제공하였다.

### III. 결론

본 논문에서는 방송 시스템에서 방송을 서비스 하고자 하는 경우에 방송 서비스를 받고자 하는 자식 노드의 최적화 서비스 부분을 고려하기 위한 로컬 변환 모델을 제안하였다. 제안된 모델에서 부모노드가 부모노드로서의 역할을 하지 못할 경우 자식노드와의 스왑 과

정을 통해 좀 더 효율적인 서비스를 받게 하였다. 이러한 방법은 소스루트의 부하를 약간 가중시키지만 서비스 측면에서는 최적의 서비스를 받을 수 있는 트리 상태를 유지하게 된다.

실험을 통해서 기존의 제안 기법들과 비교한 결과 본 논문의 제안 기법을 성능 최적화 작업에 적용하였을 경우 비효율적인 트리 상태에서 벗어나 효율적인 트리 상태에 접근하는 것을 볼 수 있었다. 평균 Latency 값 또한 많은 수의 노드가 영향을 받을 때에는 가장 작은 값을 가짐을 확인 하였다. 따라서 제안된 모델은 멀티캐스트 그룹의 이탈이 빈번한 노드들로 멀티캐스트 트리를 구성해야 되는 상황에서 효율적으로 적용될 수 있는 성능 최적화 기법이 될 수 있음을 알 수 있다.

### 참고 문헌

- [1] Y. Chu, S. G. Rao, and H. Zhang, "A Case for End System Multicast," in Proceedings of ACM SIGMERTICS, June 2000.
- [2] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel, "ALMI: An Application Level Multicast Infrastructure," 3rd USENIX Symposium on Internet Technologies and Systems, Mar. 2001.
- [3] Y. Chawathe, S. McCanne, and E. Brewer, "Scattercast: An Architecture for Internet Broadcast Distribution as an Infrastructure Service," PhD Thesis, University of California, Berkeley, 2000.
- [4] <http://www.icir.org/yoid/>
- [5] J. Jannotti, D. Gifford, K. Johnson, M. Kaashoek, and J. O'Toole, "Overcast: Reliable Multicasting with an Overlay Network," 4th Symposium on Operating Systems Design & Implementation, Oct. 2000.
- [6] H. Deshpande, M. Bawa, and H. G. Molina, "Streaming Live Media over Peers," Technical Report 2002-21, Stanford University, Mar. 2002.

[7] S. Zhuang, B. Zhao, A. Joseph, R. Kartz, and S. Shenker, "Bayeux: An Architecture for Scalable and Fault-Tolerant Wide-Area Data Dissemination," ACM NOSSDAV, June 2001.

[8] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, "Application-level Multicast using Content-Addressable Networks," In Proceedings of NGC, 2001.

[9] D. Tran, K. Hua, and T. Do, "ZIGZAG: An Efficient Peer-to-Peer Scheme for Media Streaming," in proceedings of IEEE INFOCOM, Apr. 2003.

[10] M. Y. Kang, O. F. Hamad, C. U. Pom, and J. S. Nam, "FTTH-Enhanced Mini-System mTBCP-Based Overlay Construction and Evaluation," in proceedings of SOFSEM Lecture Notes in Computer Science, Springer-Verlag, Berlin Heidelberg, Vol.4362, pp.612-623, Jan. 2007.

[11] O. Dolejs and Z. Hanzalek, "Optimality of the Tree Building Control Protocol," In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications(fPDPTA'02), Las Vegas, USA, ISBN 1-892512-90-4, CSREA Press, June 2002.

[12] <http://www.isi.edu/nsnam/ns>

**저자 소개**

**강 미 영(Mi-Young Kang)**

정회원



- 2001년 : 전남대학교, 컴퓨터공학과(공학석사)
- 2001년 4월 ~ 2001년 9월 : 한국 전자통신연구소 위촉연구원
- 2003년 ~ 현재 : 전남대학교 컴퓨터공학과(박사과정)

<관심분야> : 통신 프로토콜, 인터넷 실시간 서비스, 라우팅

**이 형 옥(Hyung-Ok Lee)**

준회원



- 2006년 2월 : 전남대학교, 컴퓨터 정보통신공학과 졸업
- 2006년 ~ 현재 : 전남대학교, 컴퓨터 정보통신공학과 석사과정

<관심분야> : 컴퓨터 네트워크, 통신 프로토콜

**곽 용 완(Young-Wan Kwag)**

정회원



- 2004년 2월 : 전남대학교, 컴퓨터 공학과(공학석사)
- 2006년 ~ 현재 : 전남대학교, 컴퓨터공학과 박사과정

<관심분야> : 컴퓨터 네트워크, 통신 프로토콜

**남 지 승(Ji-Seung Nam)**

정회원



- 1992년 : Univ. of Arizona, 전자 공학과(공학박사)
- 1992년 ~ 1995년 : 한국전자통신연구소 선임연구원
- 1999년 ~ 2001년 : 전남대학교 정보통신특성화 센터장

- 2001년 ~ 2005년 : 전남대학교 인터넷창업보육 센터장
- 1995년 ~ 현재 : 전남대학교 컴퓨터공학과 교수

<관심분야> : 통신 프로토콜, 인터넷 실시간 서비스, 라우팅