

효율적이고 확장성 있는 실시간 트리 구성을 위한 오버레이 멀티캐스트 메커니즘

준회원 남 윤 승*, 임 동 기*, 양 현 중*, 종신회원 남 지 승*

Efficient and Scalable Overlay Multicast Mechanism for Real-time Tree Construction

Yun-Seung Nam*, Dong-Gee Im*, Hyun-Jong Yang* *Associate Members,*
Ji-Seung Nam* *Lifelong Member*

요 약

인터넷 방송에서 그룹간의 통신을 위해서는 효율적이고 확장 가능한 멀티캐스트 메커니즘이 필요하다. 오버레이 멀티캐스트의 성능 향상을 위해서는 멀티캐스트 트리의 최적화가 요구된다. 이러한 최적화 문제는 NP-complete로 알려져 있다. 따라서 오버레이 멀티캐스트 트리의 각 노드들이 out-degree가 제한되어 있을 경우, 새로운 참여자는 이미 그룹에 참여된 사용자들 중 자신에게 적합한 부모노드를 효율적으로 찾아 그룹참여를 하여야 한다. 본 논문에서는 트리기반의 오버레이 멀티캐스트 구성 시, 새로운 사용자는 루트노드와의 지연시간을 측정하여 level을 설정한다. 이 후 새로운 사용자는 ACK-SEND기법을 사용하여 후보 부모노드를 효과적으로 찾고 level값을 비교하여 자신에 적합한 위치를 찾아 참여하게 된다. 각각의 노드들은 제공자 노드와 가까운 노드일수록 트리 깊이가 낮은 곳에 위치하게 된다. 또한 장애 발생 시, ACK-SEND기법을 사용하여 빠른 복구를 보장할 수 있다. 결국 신규 노드는 효율적이고 빠르게 멀티캐스트 트리에서 적합한 위치를 찾아 참여가 이뤄지는 장점이 있다.

Key Words : Real-time Tree Construction, Overlay Multicast, Application Layer Multicast, IPTV Service

ABSTRACT

In the internet broadcast, efficient and scalable mechanism of multicast is needed for the communication between groups. Furthermore, Optimization of the multicast tree is required to improve the performance of overlay multicast. This optimization is well-known as NP-complete. If a node in the tree has limited out-degree, a user who wants to join the group has to find parent user who has already joined. In this paper, the users who want to join the group need to setup their level using delay test from source node. And then new users can find candidate parent nodes effectively using ACK-SEND approach and take proper position by comparing level. The closer node of the user to root node should be located in lower level. Also, even if a barrier is caused, fast recovery will be guaranteed using ACK-SEND approach. Through this, the newcomer node can find their location in the multicast tree and join the group fast and effectively.

※ 본 연구는 대한민국 지식경제부 ITTA(Institute for Information Technology Advancement)의 대학정보통신연구협의회 ITRC (Information Technology Research Center)의 지원으로 수행되었습니다. (ITTA-2009-(C1090-0903-0011))

* 전남대학교 전자컴퓨터공학과 멀티미디어 정보통신 연구실(namyunseung@gmail.com)

논문번호 : KICS2009-09-417, 접수일자 : 2009년 9월 17일, 최종논문접수일자 : 2009년 11월 3일

I. 서 론

일반적으로 멀티미디어 서비스를 위해서 QoS (quality of service)와 멀티캐스트를 IP계층에서 제공하도록 하는 것이 가장 중요한 특징이다. 이러한 기능은 종단 시스템에서 단독으로 제공되지 못하기 때문에 IP계층으로부터 제공받아야 한다. 이는 종단 호스트의 IP유니캐스트 위에 멀티캐스트를 구현함으로써 제공 가능하다.^[1]

IP멀티캐스트^{[2],[3]}는 다수의 참가노드들에게 다양한 종류의 멀티미디어 스트리밍 전송을 위한 가장 효과적인 방법이다. 동일한 데이터를 참가노드 수만큼 복사하지 않고 전송하여 중복을 줄일 수 있기 때문이다. 하지만 멀티캐스트 장비의 설치비용, inter-domain 연결, 보안등의 문제점에 의해서 현실적인 서비스가 제한되었다^[4].

이처럼 현실성이 부족한 IP멀티캐스트의 대안으로, IP계층 대신 응용계층에서 멀티캐스트 기능을 구현하도록 하는 오버레이 멀티캐스트(Overlay Multicast) 또는 응용 계층 멀티캐스트 기술이 제안되었다. 오버레이 멀티캐스트는 오버레이 노드들끼리 유니캐스트 터널을 형성하여 데이터를 전달하며 전송하게 되어 호스트들이 라우터의 기능을 대신하게 된다. 따라서 현재의 IP망에 적용이 가능하다는 장점을 가지고 있다. 다음 그림 1은 오버레이 멀티캐스트의 전송 방식을 보여주고 있다.

그룹 참여노드는 Host 1,3,4,6,7이다. 소스노드는 자식노드인 Host 1,6에게 데이터를 전송하게 된다. 이 후 Host 3은 1,4에게 6은 7에게 데이터를 전달하게 된다.

오버레이 멀티캐스트는 IP멀티캐스트보다 오버레이 노드들의 그룹 참여 및 탈퇴가 이뤄질 경우 쉽

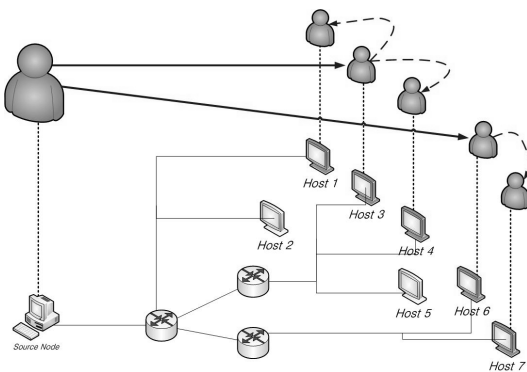


그림 1. Overlay multicast

게 충돌이 발생할 수 있고 전송 지연이 증가하는 단점을 가지고 있다^[5]. 하지만 사용자 호스트 및 네트워크 환경의 성능 향상으로 추후 발전 가능성은 충분하다.

현재 오버레이 멀티캐스트 트리 구조에 대한 알고리즘은 많이 연구되었다. 이중 Tree-based 방식은 TBCP와 HMTP가 대표적이다. 이는 새로운 참가자가 이미 참가중인 노드들과 지연시간을 비교하여 적합한 부모노드를 찾아가며 트리를 구성해 나간다.

본 논문에서는 조인 과정 시 기존 Tree-based 방식에서 사용되는 적합한 부모노드를 찾아가는 알고리즘보다 간단하고 빠른 그룹참여 및 트리복구 알고리즘을 제안하였다. 또한 트리 구성 시 신규 그룹 참여를 원하는 노드는 루트노드와의 지연시간을 이용하여 자신에게 적합한 후보 부모노드를 찾아가며 트리 위치를 알아낸다. 이는 그룹 참여시간을 줄이고 트리구성을 최적화하여 실시간 스트리밍 방송 서비스에 적합한 오버레이 멀티캐스트 트리구성 알고리즘일 것으로 보인다.

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

II. 관련 연구

일반적으로 오버레이 멀티캐스트 알고리즘은 토폴로지 구성 형태에 따라 트리 기반 오버레이(Tree-based overlays), 그물망 기반 오버레이(Mesh-based overlays), 다중 트리/그물망 오버레이(Multiple tree/mesh overlays), 링 과 다중 링 오버레이(Ring and multi-ring overlays) 그리고 분산된 해쉬 테이블(Distributed Hash Tables)로 분류할 수 있다^{[6]-[10]}.

트리 기반의 오버레이 멀티캐스트 구성법중 대표적으로 TBCP와 HMTP가 있다.

다음 그림 2에서는 TBCP의 조인 수행 과정을 보여주고 있다. 과정은 다음과 같다^[11].

- (a) 새로운 참여자 N이 후보 부모노드를 찾기 위해 루트노드인 P에게 메시지를 전송한다.
- (b) P는 현재 자신이 가지고 있는 자식노드들의 리스트를 N에게 전송한다. 이때 타이머를 설정하여 다른 새로운 참여자의 참여를 대기하도록 한다.

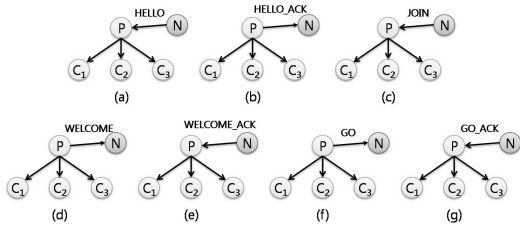


그림 2. TBCP Join Procedure

- (c) N은 P에서 보낸 리스트와의 지연시간을 측정하고 이 정보를 P에게 전송한다. 이후 P는 N의 적절한 위치를 찾기 위해서 모든 구성을 평가한다.
- (d),(e) 만약 N이 P에게 적합하면 받아들인다.
- (f),(g) N이 P에게 적합하지 않을 경우 가야할 위치를 알려준다.

HMTP는 group-shared기반의 end-host 멀티캐스트 프로토콜이다. 그림 3에서는 HMTP의 조인 수행 과정을 보여주고 있다. 과정은 다음과 같다¹²⁾.

- (a) 새로운 참여자 N은 RP에게 루트노드 P를 알아낸다. 루트노드는 N의 잠재적 부모이다.
- (b) N은 P노드의 모든 자식 노드 리스트를 받게 되고 리스트에 포함된 노드들과 지연시간을 측정한다.
- (c) N은 가장 가까운 노드를 선택하여 자신의 새로운 잠재적 부모로 선택하고, 다시 (b)를 수행한다.(이때 유효하지 않는 잠재노드는 제외한다.)
- (d) N은 부모노드에게 Join을 요청을 하고 거절되면 유효하지 않는 잠재노드로 표시하여 (c)를 다시 수행한다.
- (e) Join 요청이 허락되면 N은 부모노드와 유니캐스트 터널을 형성하고 그룹에 참여한다.

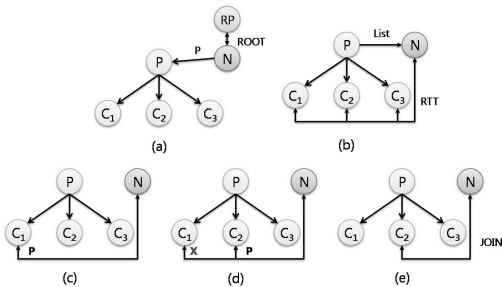


그림 3. HMTP Join Procedure

트리 기반의 구성법을 사용한 두 방식은 모두 후보 부모노드를 찾기 위해 각각의 후보노드들과 지연시간을 측정 및 평가한다. 하지만 이때 발생하는 시간 측정에 많은 시간이 소요된다. 또한 참여순서에 의한 트리구성으로 적합한 트리구성을 할 수 없게 된다.

III. 제안하는 Overlay Multicast 메커니즘

3.1 개요

오버레이 멀티캐스트를 실시간 멀티미디어 방송 서비스에 적용하기 위하여 몇 가지 해결해야 할 문제점들 있다. 첫 번째로, 새로운 참가자의 그룹참여 및 기존 참여자의 탈퇴에 의한 트리복구 지연 시간은 멀티미디어 스트리밍을 전송받는 시간 및 품질과 직결된다. 두 번째로, 오버레이 노드들을 경유하여 데이터를 전송할 때 발생하는 시간 지연이다. 이는 멀티미디어 스트리밍은 실시간성이 요구되기 때문에 시간 지연은 품질에 심각한 영향을 미친다. 따라서 빠른 그룹 참여, 복구 그리고 전송지연을 최소화 하는 것이 요구된다.

제안하는 알고리즘은 새로운 멤버 참가 시 기존의 Tree-based 구성법에서 발생하는 자식노드의 리스트 전송 및 지연시간 측정 방식을 개선하였다. 또한 신규 참여노드는 기존의 트리에서 자신에게 적합한 위치에 삽입 혹은 참여하게 된다.

새로운 노드는 루트노드를 임시 및 후보 부모노드로 선택하고 참여를 요청하게 된다. 부모노드가 새로운 참여자를 허락할 수 없을 경우, 자식노드들에게 새로운 노드의 IP주소와 level값을 포함한 메시지를 전송하게 된다. 이때 level값은 루트노드와 신규 노드와의 지연시간을 이용한 값이다. 이는 그룹 가입 시 기존의 참여자들의 level값과 비교하며 트리에서 자신에게 적합한 위치를 찾기 위해서이다. level은 루트노드와의 지연시간이 짧을수록 작은 값을 얻게 된다. 부모노드로부터 메시지를 받은 자식노드들은 자신의 level에 새로운 노드 level의 차이값을 포함시킨 메시지를 새로운 노드에게 전달한다. 새로운 노드는 가장먼저 도착한 메시지를 받게 되면 level값을 확인 후 적합한 후보 부모노드를 결정한다. 만약 level 차이가 음수이면 메시지를 전송한 노드가 자신보다 level이 낮음을 의미하고 자식노드로 참여한다. level 차이가 0이상이면 메시지를 전송한 노드가 자신보다 높거나 같은 level의 노드임으로 부모노드로 참여한다. 참가중인 노드의 그룹

탈퇴 시에는 그룹 참여 방식을 활용한 빠른 트리 재구성을 보장하도록 제안한다.

3.2 트리 최적화

제안한 오버레이 멀티캐스트 트리 구성 알고리즘은 기존의 트리 기반 오버레이 구성방법을 개선한다. 물리적인 네트워크 환경에서 오버레이 멀티캐스트 트리 구성은 참여노드 순서에 의한 트리 구성이 아닌, 최단 거리에 의한 최적화 구성이 이루어져야 한다.

그림 4에서 (a)는 그룹에 참여한 순서에 의해 완성된 토폴로지와 트리구성을 보여주고, (b)는 루트노드와의 비용 값을 이용하여 구성한 토폴로지와 트리 구성을 보여주고 있다. (a)에서 순서에 의한 트리 구성 시 N1~N3의 순서에 의해서 트리깊이를 증가시키며 참여하게 되는데 이때 완성된 트리의 총 비용은 $Cost_{total} = 40$ 이 된다. (b)의 경우 N1~N3의 순서에 의해서 참여하는 노드들은 트리 참여시 루트노드와 비용을 계산하여 자신의 level값을 결정한다. 이 후 이미 참여된 노드들과 level값을 비교하며 자신의 트리 위치를 결정하여 참여한다. 결국 루트노드와의 거리가 가까운 노드일수록 트리 깊이가 낮은 곳에 위치하게 된다. 이때 완성된 총 비용은 $Cost_{total} = 25$ 가 된다.

다음 그림 5는 그림 4.(b)의 최적화 트리구성 과정을 보여준다. 각 노드는 N1~N3노드의 순서로 루트노드에 참여요청을 하게 된다. 이때 참여하는 각각의 노드는 루트노드와의 비용 값을 이용하여 level을 결정한다. 루트노드에 N1이 참여한 후 N2노드 참여시 N1노드의 level값과 비교된다. 이때

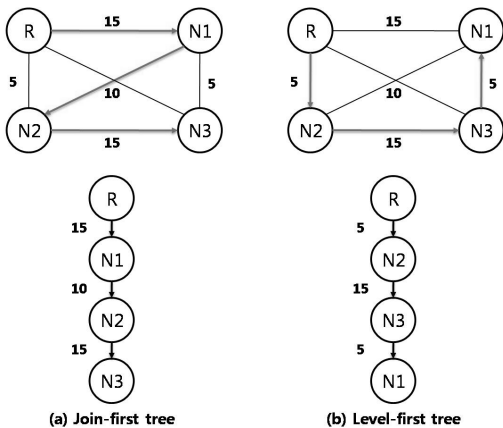


그림 4. 가입순서에 의한 트리구성과 제안한 최적화 기법에 의한 트리구성

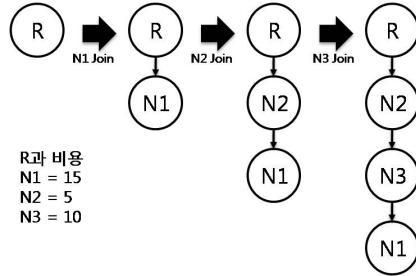


그림 5. 그림 4(b)의 최적화 트리구성 과정

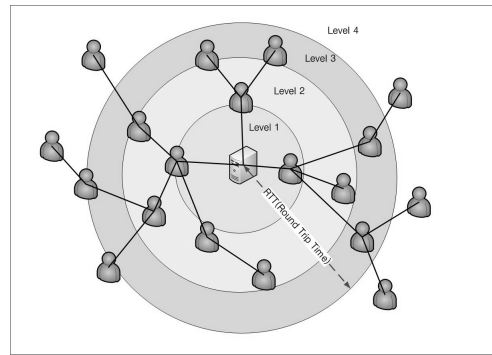


그림 6. 그림 4(b)의 최적화 트리구성 모습

N2의 비용이 N1보다 적기 때문에 스위칭이 이뤄진다. 따라서 N1노드는 N2의 자식 노드가 되고 루트노드는 N2의 부모노드가 된다. N3노드 참여시 위의 과정이 반복되어 이뤄진다.

위의 그림 6은 트리 최적화 구성모습이다. 사용자들은 중간의 서버로부터 지연시간에 의한 level값에 위치해 있다. 즉, 루트노드와 가까운 사용자일수록 트리 깊이가 낮은 곳에 위치하여 구성된다.

3.3 그룹 참여

트리기반 구성법인 TBCP와 HMTP의 경우 그룹 참가 시 새로 참여한 노드는 자신의 부모노드를 찾기 위해서 루트노드 혹은 부모노드로부터 후보 부모 리스트를 받아서 각각의 노드와 지연시간을 측정 평가하여 가장 가까운 노드를 찾게 된다. 하지만 이 경우 리스트의 모든 노드들과 지연시간을 측정 및 평가해야하는 이는 트리 구성 및 복구 시 많은 시간 지연이 소요되는 문제점을 가지고 있다.

본 논문에서 제안하는 신규 노드의 그룹 참여 방법은 후보 리스트들과의 지연시간 측정과정 없이 부모노드가 자식노드들에게 새로운 참가노드에게 ACK 메시지를 전송하도록 지시하는 방법이다. 이때 새로운 노드는 가장먼저 도착한 ACK 메시지를 전송

한 노드가 자신에게 가장 적은 지연시간을 가진 후보 부모노드임을 알게 된다.

그림 7은 제안한 알고리즘의 그룹참여 과정을 보여주고 있다. 그룹 참여 방법은 다음과 같다.

- (a) 새로운 노드 N은 그룹참가를 위하여 RP로부터 알아낸 R(루트)노드를 임시 및 후보부모노드로 선택하고 Hello 메시지를 전송한다.
- (b) R노드는 N에게 지연시간을 전송하고 N은 R과의 지연시간 값을 이용하여 자신의 level값을 결정한다. 이때 R노드와의 지연시간이 작을수록 낮은 level값을 얻게 된다.
- (c) N노드는 자신의 level값을 포함한 Join 요청 메시지를 전송한다.
- (d) R노드의 자식노드 수가 out-degree값보다 작으면 N노드에게 Join을 허락한다. N은 허락된 R노드를 부모노드로 선정한다.
- (e) R노드의 자식노드 수가 out-degree값과 같으면 자식노드들(C_x)에게 N노드의 level값과 IP가 포함된 ACK 메시지를 전송한다.
- (f), (e)의 메시지를 받은 C_x들은 자신의 level과 N노드의 level차이 값을 포함한 ACK 메시지를 N노드에게 전송한다.
- (g) N노드는 가장먼저 메시지가 도착한 C₁노드를 후보 부모노드로 선정하고 level차이 값을 분석한다. 만약 level차이 값이 음수인 경우 C₁의 R노드와의 지연시간이 자신보다 적음을 의미함으로써 C₁노드에게 Join요청을 한다. 이후 C₁노드는 (d)의 과정으로 돌아간다.
- (h) C₁노드의 level차이 값이 0보다 크거나 같으면 N은 자신의 임시 부모였던 R과 C₁노드에

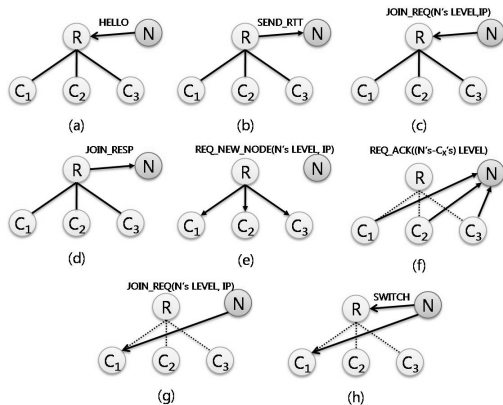


그림 7. 제안한 알고리즘 그룹 참여

게 Switch요청을 한다. 이후 N은 R의 자식노드가 되고 C₁은 N의 자식노드가 된다. 이후 N은 임시 및 후보부모노드인 R을 부모노드로 결정하고 R은 C₁대신 N을 자식노드로 허락한다. C₁은 R을 임시부모노드로 N을 부모노드로 결정한다.

(e)와 (f)의 과정으로 기존 트리 기반 구성법에서 모든 노드와의 지연시간을 검사하여 그룹참가를 하던 방법과는 다르게 새로운 노드의 Join요청과 함께 ACK 메시지를 그룹에 참여한 노드들끼리 전달하게 된다. 새로운 노드는 먼저 도착한 메시지를 전송한 노드를 지연시간이 짧은 노드로 선정하고 부모노드를 찾아가게 된다. 이를 통해서 신규노드는 그룹 참여 지연시간을 단축할 수 있다.

또한 (h)의 과정을 통해서 트리 구조의 최적화가 가능하게 된다. 따라서 루트노드에서 각각의 노드들까지 스트리밍 전송 시 지연시간을 최소화할 수 있다. 이는 오버레이 멀티캐스트에서 QoS의 문제점을 해결하는데 도움이 될 수 있다.

3.4 트리 복구

본 논문에서 제안한 알고리즘의 트리의 트리 복구과정은 다음 그림 8과 같다.

- (a) 만약 C₁노드가 그룹 탈퇴를 원하면 부모노드 P와 자식노드 C₂₋₄노드에게 Leave를 요청한다. 만약 Leave 요청 없이 C₁노드가 탈퇴할 경우 C₂₋₄노드는 C₁노드로부터 데이터가 일정시간동안 없음을 알고 인지하게 된다.
- (b) C₂₋₄노드는 C₁노드가 탈퇴하였음을 알게 되면 그룹 참여시 C₁노드를 발견하기 위해 정한 임시 부모노드 P에게 그룹 참여과정인 Join요청 메시지를 전송하고 다시 그룹참여 과정으로 돌아간다. 이후 트리는 재구성 된다.

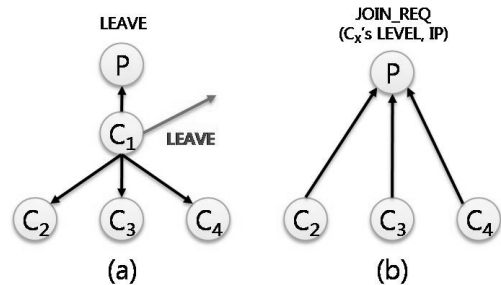


그림 8. 노드 탈퇴 시 트리 복구과정

C₂~C₄노드는 새로운 부모노드를 찾게 되고 자식 노드들에게 새로운 부모노드를 알려주어 임시부모노드를 바꾸도록 한다. 또한 C₂~C₄노드의 부모노드들은 C₂~C₄노드에게 자신의 부모노드를 알려주고 임시부모노드를 바꾸도록 한다. 이는 탈퇴한 노드의 자식노드들만 부모노드를 찾아 가입하게 됨으로써 효율적인 트리 복구를 보장한다.

IV. 성능 평가

4.1 시뮬레이션 환경

본 논문에서는 제안된 모델의 성능을 평가하기 위해 GT-ITM을 이용하여 연결 대역폭을 10Mbps로 최대 1000개의 노드를 가진 네트워크 토폴로지를 구성하였다. 구성된 토폴로지를 통해 ns-2 시뮬레이션을 사용하여 동일한 환경에서 테스트 하였다. 첫 번째와 두 번째 결과는 하나의 노드가 가질 수 있는 자식노드의 최대 수(out-degree)를 3으로 하였고 참가노드 수를 10에서부터 100개의노드까지 10씩 증가하며 그룹에 참가시켰을 때 트리 구성시간과 오버헤드 수를 보여준다. 세 번째 결과는 100개의 노드를 참여시킬 때 자식노드의 최대수를 2에서 5까지 변화시켜 테스트 하였다. 각각의 테스트는 기존의 트리 기반의 구성법인 TBCP, HMTP, 트리 최적화 방법을 사용한 경우와 사용하지 않은 경우의 ACK-SEND 알고리즘을 비교 분석 하였다. 마지막 네 번째 결과는 100개의 노드가 참여한 ACK-SEND 트리에서 최대 50개까지 임의 노드 탈퇴 시 복구 알고리즘을 테스트 하였다.

4.2 결과

첫 번째 시뮬레이션인 그림 9는 그룹 참가노드를 10씩 증가시키며 100개의 노드까지 증가하였을 때 각 노드가 트리에 참가되는데 소요되는 전체 시간의 평균값이다.

TBCP와 HMTP의 경우 후보 부모노드들과의 지연 시간을 검사하는데 소요되는 시간 때문에 제안된 알고리즘에 비해서 많은 시간이 소요되는 것을 확인하였다. 또한 트리 최적화를 사용할 경우 소요되는 지연시간에 의해서 사용하지 않은 경우보다 높은 지연시간이 소요되었다. 이는 기존의 지연시간을 측정하는 구성법보다 ACK-SEND를 통한 부모노드 발견이 보다 효율적인 것을 말해주고 있다. 따라서 실제 오버레이 멀티캐스트 서비스에서 신규 참여자의 그룹 참여 지연시간을 최소화 할 수 있다.

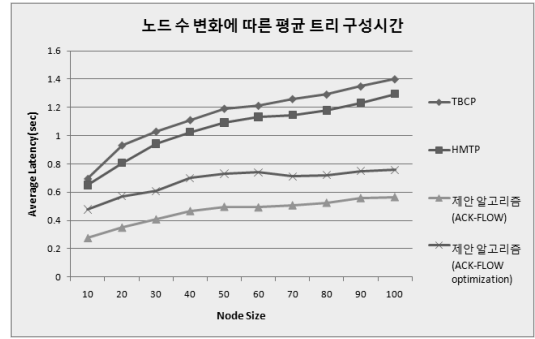


그림 9. 노드 수 변화에 따른 평균 트리 구성시간

두 번째 시뮬레이션 결과인 그림 10은 노드수를 10에서부터 100까지 10씩 증가시키며 전체노드에서 발생하는 Control Overhead의 수를 나타내고 있다.

TBCP의 경우 노드크기가 증가할수록 모든 자식노드와의 지연시간 측정 및 평가에 필요한 제어 메시지 증가에 의해서 오버헤드가 급속히 증가하고 있다. HMTP의 경우 ACK-SEND기법 보다 오버헤드수가 많이 발생하였다. 이는 HMTP의 경우 지연시간 측정을 위한 후보 부모노드의 리스트전송 및 Join을 위한 제어메시지의 증가를 의미한다. 또한 트리 최적화 구성법을 사용한 경우 루트 노드와의 지연시간 측정 및 노드 스위칭 과정에서의 제어 메시지가 증가하여 보다 많은 오버헤드 값을 나타내고 있다.

세 번째 시뮬레이션 결과인 그림 11은 100개의 노드를 참가시킬 때 out-degree를 2에서부터 5까지 변화시키며 각 노드가 트리에 참가하는데 소요되는 전체 시간의 평균값이다.

out-degree가 클수록 부모노드를 찾기 위해 비교하는 노드수가 감소하기 때문에 트리를 구성하는데 소요되는 시간이 감소하게 된다. 또한 ACK-SEND

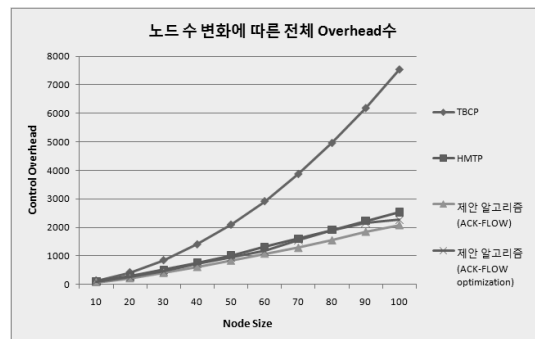


그림 10. 노드 수 변화에 따른 전체 오버헤드

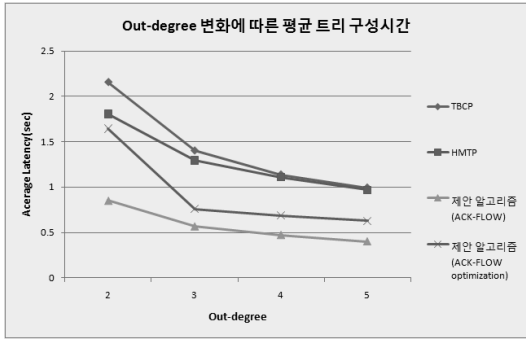


그림 11. out-degree 변화에 따른 평균트리 구성시간

트리 최적화 기법을 사용할 경우 out-degree=2인 경우에서 많은 스위칭 과정이 발생하여 최적화 기법을 사용하지 않은 경우보다 많은 지연시간이 소요되었다. 하지만 out-degree=3부터 차이 폭이 현저히 줄어들었다. 또한 지연 시간에 의한 level설정 범위에 따라 스위칭 발생을 최소화하며 더욱 최적화된 트리구성이 가능할 것으로 보인다.

네 번째 시뮬레이션 결과인 표 1은 100개의 노드가 참여한 트리에서 탈퇴노드의 수를 5개부터 전체 참여노드 수의 절반인 50개까지 임의의 시간에 그룹 탈퇴를 하였을 때, 트리 재구성을 위한 자식노드들의 재가입 횟수와 평균 지연시간을 보여주고 있다. 탈퇴하는 부모노드의 수가 증가 할수록 트리 재구성을 위한 자식노드들의 재가입 횟수가 증가하게 된다. 또한 평균 복구시간은 첫 번째 시뮬레이션의 평균 트리구성 시간과 비슷하거나 보다 짧은 시간이 소요됨을 볼 수 있다.

표 1. 재가입 노드의 수 및 평균 트리 복구 시간

탈퇴노드 수	재가입 횟수	평균복구시간(sec)
5	11	0.4791
10	21	0.4734
20	42	0.65
30	61	0.5727
40	82	0.602
50	99	0.6944

V. 결 론

오버레이 멀티캐스트의 성능 향상을 위해서는 트리 구성의 최적화가 필요하다. 본 논문에서는 기존의 트리 기반 오버레이 멀티캐스트 구성법을 개선한 ACK-SEND기법의 알고리즘을 제안하였다. 또한

최적화기법을 사용하여 트리 구성 및 복구를 최적화 하였다. 이러한 방법은 기존의 트리 기반의 트리 구성 방법보다 특별히 많은 제어 정보를 필요하지 않으며 스트리밍 전송, 트리 관리, 성능 향상 및 장애 복구 등에 있어서 기존의 방법보다 더욱 효과적이며 간단하게 적용될 수 있다.

시뮬레이션을 통해서 기존의 트리 기반의 구성법보다 빠른 트리구성 및 복구가 가능하고 오버헤드도 크게 증가하지 않는 것을 확인할 수 있었다. 제안된 오버레이 멀티캐스트 메커니즘은 실제 오버레이 멀티캐스트 스트리밍 전송 및 실시간 서비스에서 지연시간을 최소화한 서비스를 제공할 수 있다.

참 고 문 헌

- [1] Y. Chu, S. G. Rao, H. Zhang, "A Case for End System Multicast," in *proc. IEEE JOURNAL Comm*, 20(8), pp.1456-1471, Oct., 2002.
- [2] S. Deering, D. Cheriton, "Multicast routing in Datagram internetworks and extended LANS," in *proc. ACM Trans. Comp, Syst*, 8(2), pp.85-100. May, 1990.
- [3] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C.Liu, L.wei, "The PIM Architecture for Wide-Area Multicast Routing", in *proc. IEEE/ACM Trans. Networking*, 4(2), pp.784-803, Dec 1997.
- [4] C.K.YEO, B.S.Lee, M.H.Er "A survey of application level multicast techniques", in *proc. ELSEVIER, Comp Com*, 27, pp.1547-1568, May, 2004.
- [5] X. Zhang, J. Liu, B. Li, T-S, P. Yum, "CoolStreaming/DONet: A Data-driven Overlay Network for Peer-to-Peer Live Media Streaming", in *proc. IEEE INFOCOM*, 3, pp.2102-2111, Mar., 2005.
- [6] Al Hamra A. and Felber P.A., "Design Choices for content Distribuion in P2P Networks," in *proc. ACK SIGCOMM Computer Communication Review*, 35(5), pp.29-40, Oct., 2005.
- [7] EL-Sayed A., Roca V. and Mathy L., "A Survey of Proposals for and Alternative Group Communication Service," *IEEE Network*, 17(1), pp.46-51, Jan./Feb., 2003.
- [8] Ganjam A. and Zhang H., "Internet Multicast

