

다중 사용자의 멀티미디어 요구 서비스를 위한 오버레이 멀티캐스트 트리의 구성과 복구 방안

정회원 강 미 영*, 준회원 양 현 중*, 종신회원 남 지 승*

An Approach for Multi-User Multimedia Requests Service to Overlay Multicast Trees

Mi-Young Kang* *Regular Member*, Hyun-Jong Yang* *Associate Member*,
Ji-Seung Nam* *Lifelong Member*

요 약

인터넷의 대부분의 데이터가 이미지, 음성, 비디오 등의 고용량 데이터임을 감안할 때 이를 고속으로 처리할 수 있는 네트워크 상에서의 멀티미디어 데이터의 처리 요구가 증가하고 있다. 네트워크 상에서 IP 멀티캐스트의 대안으로 고려되어지는 오버레이 멀티캐스트는 하드웨어적인 인프라의 구축 없이도 시스템의 자원과 네트워크 대역폭을 효율적으로 사용할 수 있는 기법이다. 하지만 고용량의 멀티미디어를 요구하는 다중 사용자들에 대한 요구 서비스에 대한 적합한 모델과 중간 노드의 이탈 시 발생하게 되는 멀티캐스트 트리의 복구에 대한 적절한 모델이 필요하다. 본 논문에서는 시간적 스케일링을 통해 패킷 간 지연(Jitter)을 적용하여 현재 네트워크 상태를 파악한다. 파악된 네트워크 정보에 따라 제안한 다중 사용자 서비스 제어 알고리즘을 적용함으로써 멀티미디어 요구에 대한 지연시간의 영향을 최소화 하였다. 시뮬레이션 결과를 통해 제안된 모델이 기존의 기법들 보다 적은 복구 시간이 소요되고 멀티미디어 서비스 요구에 대한 다중 사용자 노드의 이탈로 인해 많은 수의 노드가 영향을 받는 상황일수록 더욱 효과적인 방안임을 보여주고 있다.

Key Words : Multi-User, Overlay Multicast, Application Layer Multicast, Resource Reservation

ABSTRACT

In the Internet, as computer resource is developed, multimedia data request being increase more and more. It is effective way that process both high capacity-data and real-time data. Overlay Multicast is an effective method for efficient utilization of system resources and network bandwidth without using hardware customization. Overlay Multicast is an effective method for multimedia data service to multi-users. Multicast tree reconstruction is required when a non-leaf host leaves or fails. In this paper, relay-frame interval is selected as revealed network-state with jitter. In our proposal, multi-user service control algorithm gives a delay effect in multimedia request time. The simulation results show that our proposal takes shorter period of time than the other algorithms to reconstruct a similar tree and that it is a more effective way to deal with a lot of nodes that have lost their multi-user nodes.

I. 서 론

네트워크의 거의 모든 데이터의 추세가 멀티미디어

어 데이터로서 이는 동시성과 연속성을 가진 미디어들의 조합이다. 오늘날 인터넷에선 이런 멀티미디어를 매개로 한 통신이 주를 이루고 있다. 멀티미디어

* 전남대학교 공과대학 컴퓨터공학과 멀티미디어 정보통신 연구실 (mykang017@empal.com)

논문번호 : KICS2008-08-356, 접수일자 : 2008년 8월 18일, 최종논문접수일자 : 2008년 10월 28일

어 영상은 현재의 인터넷 미디어가 그렇듯 실시간으로 서비스 되어야 하나 기존의 통신 프로토콜은 이를 만족시켜주지 못한다. 데이터나 시스템의 특성상 데이터 전송이 어떤 시간 안에 이루어져야 한다는 시간 제약성을 갖는 통신을 실시간 통신이라 하는데 실시간 통신 프로토콜로는 RTP, RSVP, RTSP 등이 있다. 이러한 실시간 통신 프로토콜을 사용하여 다자간 화상회의, 의료 정보 시스템, 교통제어와 같은 실시간 시스템, 멀티미디어 시스템 구축이 가능하다^[11].

다양한 멀티미디어 서비스의 제공과 네트워크 효율성 관점에서는 QoS와 멀티캐스트가 연구의 중요한 이슈로 자리 잡고 있다. 본 논문에서는 오버레이 멀티캐스트를 기반으로 다중 사용자들에게 멀티미디어 서비스를 제공한다. 오버레이 기반 멀티캐스트는 기존의 IP 멀티캐스트가 실제로 사용되기 위해서 각각의 라우터에서 IP 멀티캐스트 패킷을 처리 할 수 있도록 구현되어야 한다는 점과는 다르게 사용되고 네트워크 구조의 변경 없이 단지 종단 노드가 자신이 받은 패킷을 자신의 다음 노드에게 전달해주는 방법을 사용한 어플리케이션 소프트웨어의 설치만으로 멀티캐스트를 사용할 수 있는 기법이다.

네트워크 상에서 IP 멀티캐스트의 대안으로 고려되어지는 오버레이 멀티캐스트는 하드웨어적인 인프라의 구축 없이도 시스템의 자원과 네트워크 대역폭을 효율적으로 사용할 수 있는 기법이다. 하지만 고용량의 멀티미디어를 요구하는 다중 사용자들에 대한 요구 서비스를 제공하기 위해서는 응용계층에서부터 최종 목적지에 도착하기까지의 트래픽은 경로상의 몇몇 종단 시스템들을 경유해야만 하므로 오버레이 멀티캐스트는 IP 멀티캐스트에 비해서 전송 지연이 증가하게 된다. 이러한 전송 지연의 증가는 실시간 멀티미디어 전송에 있어 중요한 문제가 되며, 이 문제를 해결하기 위한 연구가 이루어져 왔다^{[12][13]}.

이때 트리를 구성할 시 비용에 가장 큰 주안점을 둔다면 최소 신장 트리(Minimum Spanning Tree) 알고리즘인 Kruskal 또는 Prim 알고리즘 등의 영향을 받을 것이다. 반면에 지연에 주안점을 두고 트리를 구성한다면 최단경로우선(Shortest Path Tree) 트리 알고리즘인 Dijkstra와 Bellman-Ford 알고리즘이 사용될 것이다. 본 논문에서는 시간적 스케일링을 통해 패킷 간 지연(Jitter)을 적용한 다중 사용자 서비스 제어 알고리즘을 제안한다.

II. 관련 연구

알고리즘들은 중앙형으로 실행하느냐 아니면 분산형으로 실행하느냐에 따라 트리 구성 시 두 가지 모두 약간의 문제점을 안고 있다. 중앙형 알고리즘의 문제점은 중앙노드의 링크에 문제가 생겼을 경우 일어나게 되는 심각한 과부하와 네트워크 토폴로지에 대한 전반적인 정보가 필요하다는 것이다. 전통적인 TBCP[12]와 HMTTP[13]은 오버레이 멀티캐스트 라우팅 트리를 구성하는데 분산 알고리즘을 사용하였다. 그렇지만 고용량의 멀티미디어를 요구하는 다중 사용자들에 대한 요구 서비스에 대한 적합한 모델과 중간 노드의 이탈 시 발생하게 되는 멀티미디어 트리의 복구에 대한 적절한 모델의 필요성 측면에서는 본 연구의 다중 사용자 요구 서비스를 위한 멀티캐스팅 적용에 있어서 적절하지 않다는 사실을 확인하였다. TBCP 연구에 있어 가장 큰 단점은 사실상 부모노드는 직접적으로 연결된 자식노드에 대한 정보만을 가지고 있다는 점이다.

멀티캐스트를 이용하여 일반 데이터를 전송하는 경우 최적의 경로를 통해 모든 데이터가 전송되는 것이 일차적인 목표이지만 실시간 멀티미디어 서비스와 같은 스트리밍 서비스를 하게 되는 경우에는 사용자에게 중간 노드들의 이탈로 인한 트리 구성의 변화에도 종단 노드들이 최대한 영향을 받지 않도록 QoS(Quality of Service)를 보장하는 것이 최우선이기 때문에 어느 정도 패킷 손실이 있을 지라도 빠른 시간 안에 트리를 복구해야만 한다. 이와 같이 멀티캐스트 트리를 복구하기 위한 기법으로는 크게 노드가 트리를 이탈한 후에 복구를 시작하는 Reactive approach와 사전에 복구 계획을 세워두어 노드가 이탈하게 되면 즉시 트리를 재구성하게 되는 Proactive approach로 나누어 볼 수 있다.

본 논문에서는 다양한 메커니즘들 중에서 분산형 알고리즘 기반의 트리 우선 특성을 가지고 노드들간의 지연 시간이 최소인 경로를 찾아 멀티캐스트 트리를 구성한 후 트리의 빠른 복구를 위해 Proactive approach를 사용하여 멀티미디어 서비스를 요구하는 다중 사용자들에게 적합한 서비스인 오버레이 기반 멀티캐스트 알고리즘을 제안한다.

TBCP는 오버레이 기반 멀티캐스트에서 데이터 전송을 위한 트리 구성을 담당하는 프로토콜이다. TBCP의 주된 전략은 트리를 구성하는데 있어서 제한된 최소한의 멤버/위상 정보만으로 최대한 빨리 트리를 구성할 수 있다는 것이다. 노드들간에

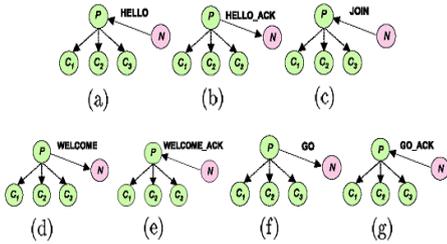


그림 1. TBCP 조인 과정

TBCP를 디자인하는데 있어서 가장 기본적으로 요구되는 사항은 네트워크에서 노드간의 정보를 얻을 수 있는 측정값이 필요하다는 것이다. 확장성의 관점에서, 스페닝 트리가 계산되기 전에 전체 모든 호스트들의 정보를 알고 있는 상태에서 트리를 구성하는 방법은 현실적으로 불가능하다. 또한 확고함과 능률성으로 보더라도 중앙에 집중되는 알고리즘을 피할 수 있어서 선호 되어진다. 따라서 TBCP에서 네트워크 라우터와의 상호작용이나 네트워크 위상의 특별한 정보 없이 노드들간의 멀티캐스트 세션을 이루는 분산된 스페닝 트리를 구성한다. 오직 그룹 멤버들간의 부분적인 정보만을 가지고 트리를 구성하는 것이다. TBCP는 트리 우선 방식이며, 새 멤버의 조인 시에 근접한 최상의 위치를 알아내서 위치시키는 전략을 가진 분산된 오버레이 스페닝 트리 구성 프로토콜이다. [그림 1]에서는 TBCP 조인 과정을 보여주고 있다.

HMTP는 집합에 필요한 시간을 단축하고 가능하면 빠르게 트리를 구성하는데 중점을 두고 있어서, 확장성을 제공하고 부모 노드의 빠른 스위칭을 기반으로 실시간 방송 서비스나 데이터/콘텐츠 배포 등의 일대다 응용 서비스에 적합하다. [그림 2]에서는 HMTP의 트리 구성 과정을 보여주고 있다.

그림에서 RP(Rendezvous Point)는 종단 노드들을 이용해서 트리를 생성하기 전에 노드들의 특성을 연결 및 해제 시키는 라우터 역할을 수행한다.

- (a) 새로운 노드 H가 RP에게 루트 A를 알아온다. 루트 노드 A는 노드 H의 잠재적 부모가 된다.
- (b) 노드 H는 잠재적 부모의 모든 자식의 리스트를 받고, RTT를 측정한다.
- (c) 잠재적 부모와 그 자식들 중 가장 작은 RTT를 갖는 노드를 택하여 새로운 잠재적 부모로 설정하고, (b)를 다시 수행한다. 이때, 무

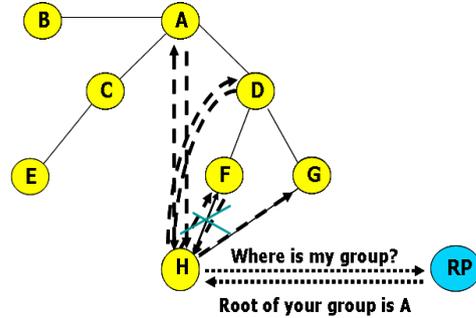


그림 2. HMTP 조인 과정

효로 표시된 노드는 제외된다.

- (d) 잠재적 부모에게 조인 요청을 한다. 만약 거절당하면, 유효하지 않은 잠재적 부모로 표시하고 (c)절차로 돌아간다. 수락받으면, 유니캐스트 터널을 형성하고 세션에 참여한다.

HMTP는 그룹 웨어드 트리 기반의 종단 노드 멀티캐스트 프로토콜이다. 라우팅의 비효율성을 줄이기 위해서, 오버레이 멀티캐스트도 가능한 한 하부의 네트워크 토폴로지와 비슷해야 한다. 그러나 종단 노드는 하부의 네트워크 정보를 알지 못하므로, 종단과 종단간의 거리 정보를 이용한다. 트리의 경우 메쉬 보다 Fail에 더욱 더 민감하기 때문에, 각 멤버 노드들은 주기적으로 이웃과 메시지를 교환하여 트리 구조를 유지한다.

Reactive approach에서는 중간 노드가 이탈하였을 경우 일반적으로 해당 노드의 조부모 노드나 루트 노드에 Rejoin 시키는 방법으로 트리를 복구하는 반면에 Proactive approach에서는 각각의 노드가 백업 루트를 가지고 있음으로서 부모 노드의 이탈에 대비하게 된다. Okada의 제안 기법[14]에서는 우선 최소 RTT 값으로 멀티캐스트 트리를 구성한 후에 두 번째 최소 RTT 값을 가지는 노드를 후보 부모 노드로 저장하여 백업 경로를 지정해 두는 방법을 사용한다. Yang의 제안 기법[15]에서는 기본적으로 중간노드가 자신이 가지고 있는 자식 노드들을 위한 백업 경로를 사전에 선정하는 방법을 사용한다. 또한 Kusumoto의 제안기법[16]은 모든 노드들이 항상 1이상의 여유 Out Degree 값 즉, 잠재적 부모노드로서 자식 노드에게 서비스를 할 수 있는 상태를 나타내는 Out Degree를 가지면서 자식 노드와 조부모 노드 사이의 연결 작업만으로 백업 경로를 설정한다.

Ⅲ. 제안하는 Overlay Multicast 모델

3.1 개요

제안 모델에서 오버레이 멀티캐스트 기반 하의 다중 사용자의 멀티미디어 서비스를 제공하는 모든 노드들은 세션 정보에 있어서 클라이언트의 요청 정보 이외에 그 클라이언트를 통하여 전송되는 논리적 데이터 전송 경로가 포함되어 있다는 점에서 유니캐스트 방식과 다르다.

새로운 노드는 세션 관리 서버에서 제공되는 프로그램 리스트를 보고 서비스 받고자 하는 프로그램을 선택한다. 세션 관리 서버는 새로운 노드의 연결 정보를 해당 멀티미디어 서버에 전송하여 주게 된다. 멀티미디어 서버는 서비스를 수행하는 제공자 입과 동시에 랑데부 기능을 수행하는 루트 노드이다. 루트 노드는 최적의 부모 노드를 선택할 수 있는 잠재적 부모 노드 목록을 응답메시지와 함께 새로운 노드에게 전송하여 줌으로써 고품질 멀티미디어 서비스를 수행하는 해당 멀티미디어 서버와 새로운 노드간에 세션 연결이 이루어진다.

멀티미디어 서버는 파일, 캡코더 그리고 TV 수신 안테나를 통하여 MPEG2-TS 패킷을 획득하며, 획득된 MPEG2-TS 패킷은 각 노드에게 중계 전송된다. 이때 실시간 다중 사용자 서비스를 위해 제안한 오버레이 멀티캐스트 기반의 다중 사용자 서비스 제어 알고리즘이 적용된다. 오버레이 멀티캐스트 특성 상 트리에 구성원이 되는 노드들은 멀티미디어 서비스를 받고 있음과 동시에 다른 노드에게 릴레이 해 줄 수 있는 기능까지 가지고 있어야 한다.

루트 노드는 새로운 노드에게 현재 그룹에 가입할 수 있는 정보를 주기 위해 PPL (Potential Parent List)을 전송해 주는데 이 PPL의 구조는 [그림 3]과 같다. PPL은 서비스를 제공해 줄 수 있는

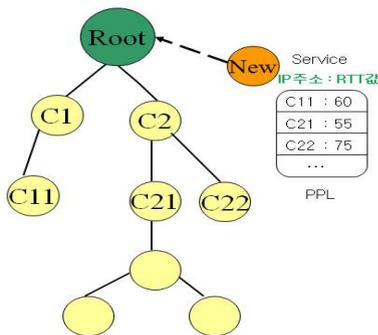


그림 3. 루트 노드의 잠재적 부모 리스트의 구조

노드의 IP 주소와 그 노드가 부모 노드를 거쳐 루트 노드까지 가지는 RTT 값으로 구성된다.

루트 노드는 멀티캐스트 그룹에 새로운 노드가 가입하고 떠나거나 노드들이 가지고 있는 정보가 변경될 때마다 이 정보를 갱신하여 항상 최신의 트리 정보를 유지하게 된다.

3.2 다중 사용자를 위한 트리 구성 계획

다양한 메커니즘들 중에서 분산 형 알고리즘 기반의 트리 우선 특성을 가지고 있으며 상위 노드로부터 자식 노드에게 서비스를 할 수 있는 상태를 나타내는 Out Degree를 순서대로 모두 소진해 가는 트리 밸런스를 기본으로 하고 있는 다중 사용자 서비스 제어 알고리즘은 새로운 노드의 조인이 용이하며, 노드들 간의 지연 시간이 최소인 경로를 찾아 설정함으로써 멀티미디어 서비스의 품질을 보장한다. IP 멀티캐스트에서는 트리를 구성하는 중간 노드들이 라우터이기 때문에 고려할 필요가 없는 부분이 오버레이 멀티캐스트에서는 중간 노드를 포함한 모든 노드들이 언제든지 멀티캐스트 그룹을 자유롭게 이탈할 수 있는 중단 노드들로 구성이 된다는 점에서 새로운 문제점을 안고 있다. 따라서 오버레이 멀티캐스트 트리를 구성하고 있는 중간 노드가 멀티캐스트 그룹을 이탈하였을 경우 트리를 재구성 해야 하며 성능 저하에 대한 대책도 간구하여야 한다. 따라서 제안한 메커니즘에서는 중간 부모 노드들의 이탈로 인해 많은 수의 자식 노드가 영향을 받는 상황에 대해서도 효과적인 방안을 제공하고 있다.

아래 [그림 4]와 [그림 5]는 새로운 노드가 멀티캐스트 그룹에 가입하는 과정인 업무 수행 흐름도를 보여주고 있다.

새로운 노드는 제일 먼저 루트 노드에게 서비스를 요청하고 루트 노드는 자신의 Out Degree 정보를 확인한다. 만약 자신이 직접 서비스를 제공할 수 있는 상태라면 바로 서비스 요청을 수락하는 Ack 메시지를 전송하여 서비스를 제공하게 되지만 그렇지 못한 경우에는 그룹 멤버들로부터 수집한 PPL을 대신 전송하게 된다.

이를 수신한 새로운 노드는 리스트 내에 있는 노드들과의 RTT 값을 체크하는 작업을 수행하게 되고 이 때 얻어진 값을 PPL 내의 각 노드의 값과 합산하여 전체적인 경로의 RTT 값을 계산한다. 이렇게 얻어진 값은 적은 순서대로 정렬이 되어 순차적으로 서비스를 요청하게 된다.

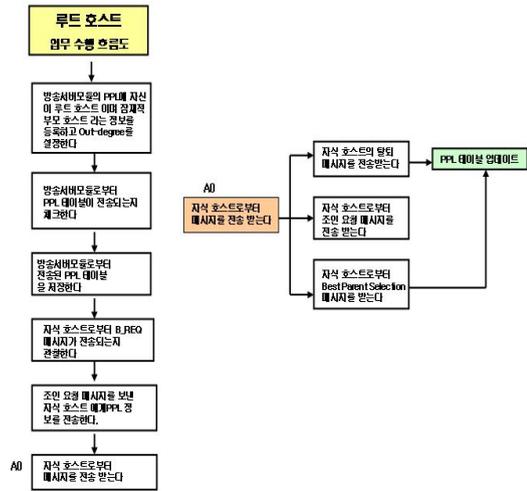


그림 4. 루트 노드의 업무 수행 흐름도

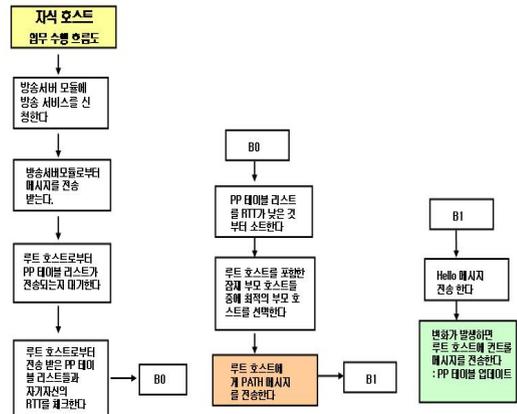


그림 5. 자식 노드의 업무 수행 흐름도

위의 작업을 거쳐 새로운 노드는 자신의 부모 노드를 찾아 서비스를 받게 되면 새롭게 가입한 노드는 자신이 가지고 있는 전체 RTT값과 Service Out Degree 값에 여유가 있음을 알려주는 메시지를 전송하게 된다. 이러한 방법으로 루트 노드는 항상 그룹 멤버들의 최신의 정보를 가지고 멀티캐스트 트리를 구성할 수 있는 준비를 할 수 있게 된다.

3.3 다중 사용자를 위한 트리 복구 계획

오버레이 멀티캐스트 환경에서는 중간 노드를 포함한 모든 노드들이 언제든지 멀티캐스트 그룹을 자유롭게 이탈할 수 있는 오버레이 멀티캐스트 환경에서의 서비스 사용자 이탈은 전체 성능에 매우 중요한 영향을 미친다. 특정 노드가 서비스 도중 이탈했을 경우 특정 노드는 자신의 자식 노드들 모두

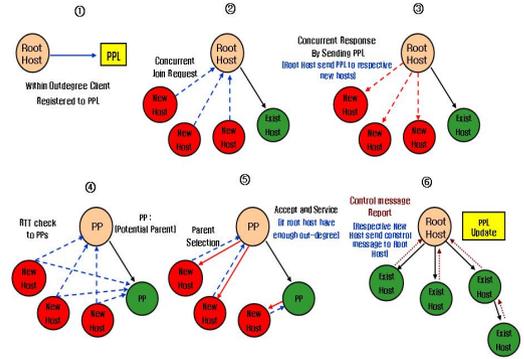


그림 6. 다중 사용자 노드의 복구 기법

에게 서비스가 중단 될 수 있는 상황을 초래할 수 있기 때문이다. 제안 모델에서는 이런 예외상황을 피드백 되어오는 컨트롤 메시지 정보에 의해 특정 노드들의 이탈을 모니터링하며 다중 사용자의 멀티미디어 요구 서비스가 즉시 복구 될 수 있도록 하기 위해 [그림 6]과 같은 복구 기법을 사용한다.

다중 사용자 멀티미디어 서비스를 제공하는 중간 노드가 정상적인 탈퇴 또는 임의적인 이탈을 하였을 경우 자식 노드들은 부모 노드로부터 멀티미디어 서비스를 제공받지 못하는 상황이 발생하게 된다. 이때 자식 노드들은 리JOIN 요청을 루트 노드에 보내며 루트 노드는 서비스 리JOIN 요청을 보낸 각각의 자식 노드들에게 현재 관리하고 있는 최신의 PPL 정보를 보냄으로써 리JOIN 요청에 대한 응답을 하게 된다. 각각의 리JOIN 메시지를 보낸 자식 노드들은 PPL 정보를 받고 멀티미디어 서비스를 받을 수 있는 최적의 부모 노드를 선택하기 위해 PPL에 등록 되어진 잠재적 부모 노드와 자기 자신이 각각의 RTT 체크 루틴을 수행함으로써 가장 최적의 부모 노드를 선택하게 된다. 본 제안 기법에서 복구 메커니즘을 사용할 때 처음에 설정한 복구 노드는 시간이 흐르면서 부분적인 트리의 구조가 변화함에 따라 최적의 경로를 가지는 복구 노드일 확률이 감소하게 된다. 때문에 항상 최적의 경로를 가지는 멀티캐스트 트리를 구성하고자 복구 노드를 갱신하는 작업을 할 수 있다. 이 작업은 노드의 이탈이 발생했을 때 그 영향을 받는 모든 노드들이 가지고 있는 복구 연결을 갱신하면 되는 데 이탈한 노드가 상위 노드일수록 이 작업에 부하가 크다는 단점이 있다. 이러한 이유로 상위 노드부터 순차적으로 새로운 복구 노드를 찾는 작업을 수행하여 동시에 많은 수의 노드들이 복구 노드를 찾는 부하를

줄었고 한 순간에 하나의 노드만이 작업을 하여 복구 노드를 찾는 과정 중에 부모 노드가 이탈하여 장시간 서비스를 받지 못하는 확률을 최대한 감소시켰다.

IV. 성능 평가

4.1 시뮬레이션 환경

본 논문에서는 제안된 모델의 성능을 평가하기 위해 ns2 환경 상에서 최대 1000개의 노드를 생성하였고 한 개의 노드가 가지는 최대 Out Degree 값은 100Mbps의 네트워크 환경에서 25Mbps 정도의 대역폭이 필요한 HD급 멀티미디어 요구에 대한 서비스를 한다고 가정하여 3으로 정의 하였고 노드를 거치면서 발생 할 수 있는 작업(Processing) 지연 시간은 모든 노드가 동일한 환경 하의 시스템이라고 가정하여 환경 요소에서 제외 하였다.

제안된 알고리즘의 성능 비교를 위해 관련 연구에서 살펴보았던 TBCP의 제안 기법과 HMTP의 제안 기법을 동일 환경 하에서 프로그래밍 후 테스트 하였다. 단, 복구 알고리즘 상의 비교를 위해 멀티캐스트 트리의 구성 알고리즘은 본 논문에서 제안한 각 경로가 가지는 최소 RTT 값을 이용하여 트리를 구성하는 방법으로 통일하였다.

4.2 결과

첫 번째 시뮬레이션인 [그림 7]에서는 트리를 구성하는 노드의 수를 0개부터 100개까지 증가시켜가며 이 때 구성이 된 트리내의 링크간의 RTT 총합의 변화 값을 측정하였다. 다중 사용자 요구 트리 구성 시 RTT 변화를 분석할 수 있는데 시간이 증가할수록 차이를 보이고 있음을 알 수 있다. 이 중에서 TBCP의 성능이 비교적 좋지 않음을 볼 수 있

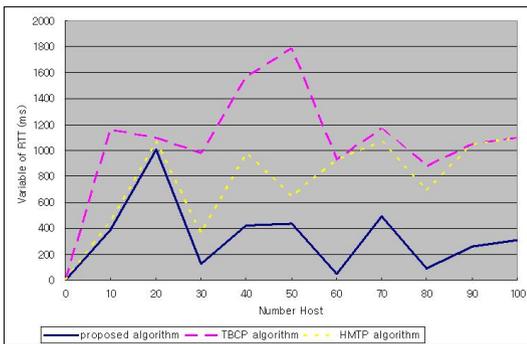


그림 7. 다중 사용자 요구 트리 구성 시 RTT 변화

는데 이는 트리 구성 시 TBCP 기법의 특성을 보여주고 있다. TBCP 기법은 트리 구성 시 새로운 노드의 가입 요청을 일정시간 제한 한 다음 순차적으로 서비스를 수행하기 때문이다. 더 나은 성능을 보이고 있는 HMTP는 동시적으로 새로운 노드들이 가입 요청을 하였을 때 서비스 요청에 대해서 동시적으로 수행하지만 루트 노드의 모든 자식 노드 리스트를 체크한 다음에 최적의 부모 노드를 선택함으로써 TBCP 기법 보다는 더 나은 성능을 보이고 있지만 트리의 상위 부모 노드의 서비스 기능 여부에도 불구하고 하위 수준에 인접한 위치에 존재하는 노드들끼리 그룹핑이 발생하므로 HMTP 기법에서 발생하는 그룹핑은 트리의 평균 깊이를 증가시키며 그 결과 제안한 다중 사용자 서비스 제어 알고리즘에 비해 RTT 값을 증가시키는 결과로 이어진다. 반면 제안된 기법에서는 노드들이 동시에 서비스 요청을 하여도 최적의 부모 노드를 선택할 수 있는 권한을 주며 상위로부터 Out Degree를 모두 소진하면서 서비스를 수행하므로 RTT 값의 차이를 줄일 수 있게 되었다.

두 번째 시뮬레이션인 [그림 8]에서는 노드의 수를 0개부터 100개까지 트리의 수를 증가시켜가며 구성한 후 랜덤하게 가입한 노드를 이탈시켜 복구 후 변화된 RTT 값을 측정한 것이다. 적은 수의 노드로 트리가 구성이 되었을 때에는 제안된 기법과 HMTP 기법 사이에 차이는 근소하지만 노드의 수가 증가 할수록 제안된 기법의 성능이 더 좋을 수 있는데 이는 복구가 필요한 노드의 경로가 제안된 기법에서는 비교적 상위 노드에 분포하고 있지만 HMTP의 기법에서는 부모 노드의 자리를 차지한 한 개의 자식 노드를 제외하면 모두 하위에 위치한 노드에게서 다시 서비스를 받게 되는 확률이 크기 때문이다.

[그림 7~8] 시뮬레이션 결과로부터 노드가 증가할 때 RTT 값이 급격하게 떨어지는 현상을 모니터링 할 수 있다. 이는 노드 수가 적은 상태에서의 트리 구성이나 노드의 트리 이탈 시에 상위에 위치한 노드의 영향을 받을 확률이 높다. 하지만 노드의 수가 많아지면서는 하위에 위치한 노드들 사이의 트리 구성이나 노드들의 트리 이탈 확률이 높다. 따라서 노드가 증가할수록 RTT 값이 급격하게 떨어지는 현상이 일어난다.

세 번째 시뮬레이션인 [그림 9]에서 측정된 트리 비용은 오버레이 기반 멀티캐스트에 의해 생성된 멀티미디어 요구 서비스 전송 시 트리의 링크에서

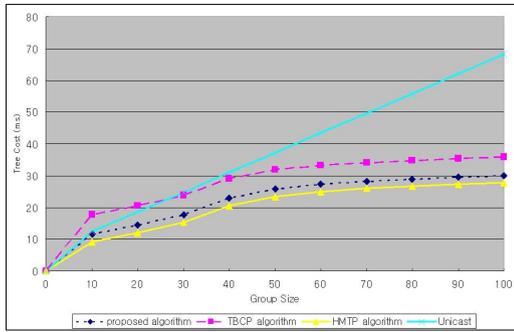


그림 9. 그룹 호스트 수에 따른 총 트리 비용

발생하는 전송 지연을 의미한다. 총 트리 비용은 루트 노드에서 멀티캐스트 그룹에 속한 모든 노드들에 멀티미디어 요구 서비스를 전송할 경우 발생하는 모든 트리 비용의 합을 의미한다.

위의 시뮬레이션 결과인 그룹 노드 수의 변화에 따른 총 트리 비용을 보면 제안하는 알고리즘과 HMTP 알고리즘 그리고 TBCP 알고리즘 모두 유니캐스트에 비해 그룹 노드의 수가 증가함에 따라 성능향상을 보이고 있음을 알 수 있다. 또한 HMTP는 제안한 알고리즘에 비해 조금 더 뛰어난 성능을 보이고 있으나 그룹 노드의 수가 증가 함에 따라 점점 그 격차가 감소하고 있음을 알 수 있다.

V. 결 론

본 논문에서는 오버레이 멀티캐스트에서 고용량의 멀티미디어를 요구하는 다중 사용자들에 대한 요구 서비스에 대한 적합한 모델과 중간 노드의 이탈 시 발생하게 되는 멀티캐스트 트리의 복구에 대한 모델을 제안하였다. 제안된 모델에서 노드들이 동시에 서비스 요청을 하였을 시 최적의 부모 노드를 선택할 수 있는 권한을 주며 상위로부터 Out Degree를 모두 소진하면서 서비스를 수행할 수 있도록 하였다 또한 복구 노드 갱신작업을 통해 항상 최적의 경로를 가지는 트리를 구성하게 되었다. 하지만 Out Degree의 한계로 인하여 가입을 허용할 수 없는 경우 새로운 노드의 멤버 가입에 지연이 발생할 수도 있다. 본 논문에서는 PPL이 전체적으로 하나로 작성되어 루트 노드로부터 관리하는 방법 하 에서 최적의 서비스를 받는 방법을 제안하였지만, 향후 PPL을 지역적으로 관리함으로써 최적의 서비스를 수행할 수 있는 연구가 필요하다.

시뮬레이션을 통해서 기존의 제안 기법들과 비교

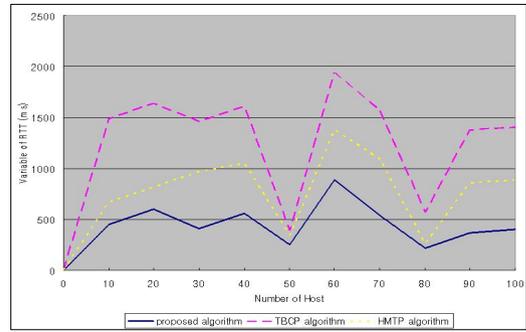


그림 8 다중 사용자 요구 트리 복구 시 RTT 변화

한 결과 본 논문의 제안 기법이 트리 구성 작업과 복구 작업에서 가장 적은 시간이 소요되었고 RTT 값의 변화 또한 가장 작았음을 볼 수 있었다. 따라서 제안된 모델은 멀티캐스트 그룹의 이탈 빈도가 낮은 안정적인 노드들로 구성이 된 트리뿐만이 아니라 이탈이 빈번한 노드들로 멀티캐스트 트리를 구성해야 되는 상황에서도 효율적으로 적용될 수 있는 기법이 될 수 있음을 알 수 있다.

참 고 문 헌

- [1] Chunlei Liu, "Multimedia Networking : Goals and Challenges", http://www.cis.ohiostate.edu/~jain/cis788-97/ip_multimedia/index.htm, 2000
- [2] Schulzrinne, Casner, Frederick, and Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC 1889, February 1996
- [3] Y. Chu, S. G. Rao, H. Zhang, "A Case for End System Multicast," in Proceedings of ACM SIGMETRICS 2000, June.
- [4] D. Pendarakis, S. Shi, D. Verma, M. Waldvogel, "ALMI: An Application Level Multicast Infrastructure", 3rd USENIX Symposium on Internet Technologies and Systems, Mar 2001.
- [5] Y. Chawathe, S. McCanne, E. Brewer, "Scattercast: An Architecture for Internet Broadcast Distribution as an Infrastructure Service" PhD Thesis, University of California, Berkeley, 2000
- [6] <http://www.icir.org/yoid/>
- [7] J. Jannotti, D. Gifford, K. Johnson, M. Kaashoek, J. O'Toole, "Overcast: Reliable

Multicasting with an Overlay Network”, 4th Symposium on Operating Systems Design & Implementation, Oct. 2000

[8] H. Deshpande, M. Bawa, H. Garcia-Molina, “Streaming Live Media over Peers”, Technical Report 2002-21, Stanford University, Mar. 2002

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

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

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

[12] Dolejs, O., Hanzak, Z., “Optimality of the Tree Building Control Protocol”, In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, Las Vegas, USA, ISBN 1-892512-90-4, CSREA Press, June 2002.

[13] L.Mathy, R. Canonico, and D. Hutchison, “An Overlay Tree Building Control Protocol”, NGC2001, Nov. 2001.

[14] Y. Okada, M. Oguro, J. Katto, S. Okubo, “A New Approach for the Construction of ALM Trees using Layered Video Coding”, in Proceedings of ACM Multimedia 2005.

[15] M. Yang, Z. Fei., “A Proactive Approach to Reconstructing Overlay Multicast Trees”, in proceedings of INFOCOM 2004, March.

[16] T. Kusumoto, Y. Kunichika, J. Katto, S. Okubo, “Proactive Route Maintenance and Overhead Reduction for Application Layer Multicast”, ICAS/ICNS 2005.

강 미 영 (Mi-Young Kang)

정회원



2001년 전남대학교 컴퓨터공학과(공학석사)

2001년~2002년 한국전자통신 연구소 위촉연구원

2003~현재 전남대학교 컴퓨터공학과 박사과정

<관심분야> 통신프로토콜,

인터넷 실시간 서비스, 라우팅

양 현 종 (Hyun-Jong Yang)

준회원



2008년 전남대학교 전자컴퓨터 정보통신 공학부

2008년~현재 전남대학교 전자컴퓨터 공학과 석사과정

<관심분야> 컴퓨터 네트워크, 통신 프로토콜, 라우팅

남 지 승 (Ji-Seung Nam)

중신회원



1992년 Univ. of Arizona, 전자공학과(공학박사)

1992년~1995년 한국전자통신 연구소 선임연구원

1999년~2001년 전남대학교 정보통신특성화 센터장

2001년~2005년 전남대학교 인

터넷창업보육 센터장

1995년~현재 전남대학교 컴퓨터공학과 교수

<관심분야> 통신 프로토콜, 인터넷 실시간 서비스, 라우팅