

라우팅 테이블의 크기를 고려한 랑데부 라우터 선정 알고리즘

조 기 성[†] · 장 희 선^{††} · 김 동 휘^{†††}

요 약

공유형 트리(Shared Tree)를 구성하여 멀티캐스트 서비스를 제공하는 CBT(Core Based Tree)와 PIM-SM(Protocol Independent Multicast-Sparse Mode) 프로토콜에서는 랑데부 라우터(RP: Rendezvous Point)의 위치에 따라 전체 네트워크의 효율성이 결정된다. 본 논문에서는 멀티캐스트 데이터 전송을 위하여 필요한 라우팅 테이블의 크기(테이블 내 entries의 수)를 최소화하고, 소스-멤버 사이의 비용을 일정한 범위 내로 관리할 수 있는 새로운 RP 선정 알고리즘을 제안한다. 성능분석 결과, 비용만을 고려하는 기존의 방법에 비하여 총비용은 5.37%, 라우팅 테이블의 크기는 13.35%까지 줄일 수 있음을 알 수 있다.

키워드 : 멀티캐스트, 랑데부 포인트, 코어 라우터, 라우팅 테이블, 서비스품질

A Rendezvous Router Decision Algorithm Considering Routing Table Size

Kee-Seong Cho[†] · Hee-Seon Jang^{††} · Dong-Whee Kim^{†††}

ABSTRACT

Depending on the location of the rendezvous point (RP), the network efficiency is determined in the core based tree (CBT) or protocol independent multicast-sparse mode (PIM-SM) multicasting protocol to provide the multicast services based on the shared tree. In this paper, a new algorithm to allocate the RP using the estimated values of the total cost and the size(number of entries) of the routing tables is proposed for efficiently controlling the cost and the number of routing table entries. The numerical results show that the proposed algorithm reduces the total cost in 5.37%, and the size of routing tables in 13.35% as compared to the previous algorithm.

Key Words : Multicast, Rendezvous Point, Core Router, Routing Table, QoS

1. 서 론

최근 컴퓨터와 정보통신 기술의 발달로 인터넷이 크게 보급되어 문자, 숫자 등의 단순한 정보뿐만 아니라 오디오, 비디오, 그래픽 등의 멀티미디어 데이터의 처리와 통신이 가능하게 되었다. 아울러 멀티미디어 데이터 전송 기술은 고대역을 요구하는 인터넷 방송, 화상회의, 인터넷 기반의 전자상거래, 동영상 광고, 실시간 방송, 원격 의료 서비스 및 원격 교육 방송 등의 다양한 서비스에 활용될 수 있다.

대표적인 광대역 인터넷 서비스, 예를 들어 실시간 방송이나 원격 교육 방송 서비스는 전송되어야 할 데이터의 양과 특정한 다수의 수신 그룹 등과 같은 제약사항으로 많은 대역폭을 소모하게 된다. 더욱이 이와 같은 서비스를 유니

캐스트(Unicast) 전송을 이용하여 제공하는 경우 다수의 경로 전송에 따른 대역폭 낭비가 큰 문제가 된다. 따라서 이런 문제점을 해결하기 위하여 멀티캐스트(Multicast) 전송 기술을 사용하며, 멀티캐스트에서는 네트워크를 통해 송신 노드(소스, Source, Sender)가 동일한 패킷을 다수의 수신 노드(멤버, Member, Receiver)들에게 동시에 전달함으로써 대역폭과 지연시간 관점에서 패킷 전달의 효율을 높인다 [1,2].

대표적인 멀티캐스트 서비스를 사용하는 어플리케이션으로 IPTV나 원격 교육 방송 서비스를 들 수 있으며, 여기에서는 서비스를 원하는 사용자가 일정한 절차를 통해 서비스 그룹에 가입(Join) 하고 가입자 정보를 바탕으로 라우팅 프로토콜[3-7]을 사용하여 멀티캐스트 트리(Tree)를 생성하며, 이를 바탕으로 라우팅 테이블을 생성하고 가입자를 관리하게 된다. 이 경우 라우팅 프로토콜에 따라 트리 구성 방식을 소스 기반 트리(Source-based Tree)와 센터 기반 트리(Center-based Tree)로 나눈다. 소스 기반 트리에서는 송신

[†] 정희원 : 한국전자통신연구원 팀장

^{††} 정희원 : 평택대학교 경상학부 교수(e-비즈니스연구소 소장)

^{†††} 종신희원 : 대구대학교 컴퓨터·IT 공학부 교수

논문접수 : 2006년 7월 27일, 심사완료 : 2006년 11월 12일

호스트로부터 각 멤버에 이르는 트리, 즉 송신 호스트의 수만큼 트리를 관리해야 하며 DVMRP(Distance Vector Multicast Routing Protocol)[5], MOSPF(Multicast Extension to Open Shortest Path First)[6,7] 및 PIM-DM (Protocol Independent Multicast-Dense Mode) 프로토콜[9]에서 이용되고 있다. 반면, 센터 기반 방식에서는 공유형 트리[3]를 사용하며 확장성과 트리 관리 측면에서 유리한 CBT(Core Based Protocol)[9]와 PIM-SM(Protocol Independent Multicast-Sparse Mode)[3] 프로토콜에서 이용되고 있다. 그리고 센터 기반 방식에서 송신 호스트들은 코어(Core) 혹은 랑데부 라우터(RP: Rendezvous Point)라고 불리는 중심 라우터(Center Router)를 선정하고 이 중심 라우터로부터 각 멤버까지 공유형 트리를 구성한다.

따라서 공유형 트리를 이용하는 프로토콜에서는 어떤 노드를 RP로 선정하느냐에 따라 소스-멤버들 사이의 비용(거리, 홉수, 대역폭, 링크 가중치 등)과 대역폭 사용 효율에 영향을 미치게 되며 이를 위하여 여러 가지 알고리즘들이 제안되었다[4,8,9]. 대표적으로 이들은 후보 노드와 멤버들 사이의 최대 거리, 평균 거리, 공유형 트리 구성시 연결 링크들의 수 등을 구하여 이를 최소화하는 노드를 RP로 지정한다. 특히, Thaler와 Ravishankar[4]이 제안한 비용 예측값을 이용한 알고리즘에서는 노드와 각 멤버까지의 거리에 대한 예측 값을 이용하여 RP를 지정하며 비용과 지연시간 면에서 다른 알고리즘들에 비하여 성능이 우수함이 입증되었다. 그러나 제안한 알고리즘에서는 주어진 소스와 멤버에 따라 멀티캐스트 라우팅을 위하여 필수적으로 관리해야 할 라우팅 테이블의 크기(테이블 내 entries의 수)를 고려하지 않았다.

본 논문에서는 여러 개의 소스와 멤버가 SSM(Source-specific Multicasting)[10] 방식을 통해 정보를 전달하고 이를 위하여 공유형 트리를 이용하는 센터 기반 프로토콜에서 멀티캐스트 라우팅을 위한 RP 선정 알고리즘을 새로 제안한다. 여기에서는 일정한 범위 내에서의 비용을 만족하는 라우터들 중에 라우팅 테이블의 크기를 최소화하는 노드를 RP로 선정함으로써 기존의 알고리즘 보다 효율적인 알고리즘을 제안하고자 한다. 즉, 제안한 알고리즘에서는 소스-멤버 사이의 비용(지연)이 일정한 범위 내에 존재하는 후보 노드들 중 라우팅 테이블의 크기를 최소화하는 후보 노드를 RP로 선정함으로써 총비용과 라우팅 테이블의 크기를 줄일 수 있다. 서론에 이어 2장에서는 멀티캐스트 라우팅 프로토콜에 대하여 설명하고 3장에서는 RP 선정 알고리즘에 대하여 설명한다. 그리고 시뮬레이션 결과를 4장에 보이며, 끝으로 결론과 추후 연구 내용을 논한다.

2. 멀티캐스트 라우팅

멀티캐스트 라우팅 프로토콜은 멀티캐스트 송신 호스트에서 그룹 멤버에 이르는 경로를 정하기 위한 프로토콜이다. 그룹별로 송신 호스트와 그룹 멤버를 연결시켜주는 트리가

구성되며, 이에 따라 멀티캐스트 라우터는 forwarding table을 작성하고 이 테이블에 따라 수신된 데이터그램을 소속 그룹별로 내보낼 인접 라우터(인터페이스)를 지정한다. 멀티캐스트 라우팅 프로토콜은 트리의 Root 위치와 멤버 호스트들의 조인 정보 가입 형태에 따라 <표 1>과 같이 나눌 수 있다[3,9].

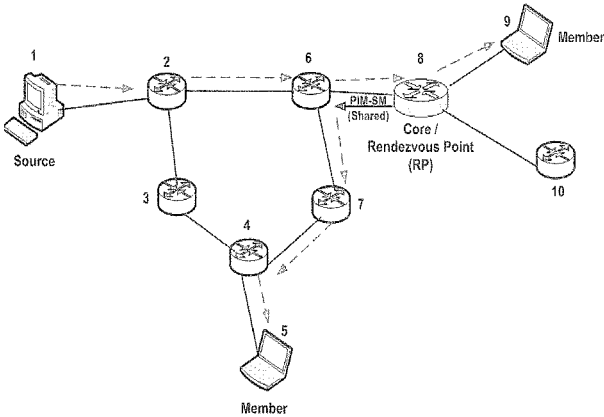
먼저, 루트의 위치를 기준으로 보면 멀티캐스트 데이터그램 전송을 위한 트리 구성시 소스 기반 트리(SBT: Source-based Tree)를 구성하는 경우와 센터 라우터를 지정하여 구성하는 경우로 나눌 수 있다. 소스 기반 방식에서는 소스별로 트리를 구성하며 센터 기반 트리에서는 센터 라우터를 중심으로 공유형 트리를 구성하여 각각의 소스가 이를 공유하게 된다. 그리고 멤버와 연결된 라우터들의 멤버 가입 형태에 따라 묵시적 가입(Implicit Join)과 명시적 가입(Explicit Join)으로 분류된다. 묵시적 가입에서는 라우터가 멤버로서 가입하는 절차가 없으며 데이터그램이 도착하면 먼저 다른 라우터로 보내고 그 라우터로부터 해지를 요청하는 메시지가 없으면 그 라우터는 계속 멤버로 간주한다. 반면, 명시적 가입은 데이터그램을 송신하기 이전에 라우터들이 송신 호스트나 중심 라우터들에게 Join 메시지를 보냄으로써 멤버로서의 가입의사를 전달한다. 묵시적 방법에 비하여 명시적 방법에서는 데이터그램 전송 이전에 각 멤버에 이르는 트리를 구성할 수 있어 멤버가 없는 지역으

<표 1> 멀티캐스트 라우팅 프로토콜

조인 가입 형태	루트 위치	소스 기반 트리	센터 기반 트리
Implicit (묵시적 가입)		DVMRP PIM-DM	-
Explicit (명시적 가입)		MOSPF PIM-SM(SBT)	CBT PIM-SM(Shared)

<표 2> 트리구성 형태에 따른 프로토콜의 장단점

트리 구성	소스 기반 트리	센터 기반 트리
장점	<ul style="list-style-type: none"> - 소스가 적은 경우 유리 - 트래픽 분산 처리 - 송수신 호스트 사이 최적의 경로가 설정되므로 지연시간이 짧음 	<ul style="list-style-type: none"> - 그룹별로 한 개의 트리만 관리(라우팅 테이블 관리를 위한 메모리가 적음) - 확장성(Scalability) 우수 - 멤버가 없는 지역으로 데이터그램을 보내는 비효율성 제거 - 비디오 게임, 화상회의(소스가 많은 경우)에 유리
단점	<ul style="list-style-type: none"> - 송신 호스트의 수만큼 트리 관리(라우팅 테이블 관리를 위한 메모리 소비가 큼) - 트리 유지/관리 및 네트워크 토폴로지 파악을 위한 제어 트래픽(오버헤드) 발생 - 대규모 망에 부적합 - 데이터그램 전송을 위한 대역폭 사용이 큼 	<ul style="list-style-type: none"> - 송수신 호스트 사이 지연시간이 김 - 센터 라우터로의 트래픽 집중



(그림 1) 센터 기반 트리

로의 비효율적인 전송을 방지할 수 있다. <표 2>는 트리 구성 형태에 따른 프로토콜의 장단점을 나타낸다.

(그림 1)은 소스가 하나이며, 멤버가 2개인 네트워크에서 센터 기반의 트리 구성 예를 보여준다. 여기에서 CBT에서는 Core, PIM-SM에서는 Rendezvous Point(RP)라고 불리는 센터 라우터를 8번 노드로 가정한다. CBT의 경우 라우팅 절차를 요약하면 다음과 같다.

- (1) IGMP 프로토콜을 이용하여 각각의 멤버는 RP로 Join 메시지를 전송하며 이 경우 RP까지의 최단경로를 이용한다. 5번 노드는 5→4→7→6→8의 경로를 따라, 9번 노드는 9→8의 경로를 따라 RP로 Join 메시지를 전송한다.
- (2) Join 메시지의 전송 경로를 따라 양방향(멤버↔센터)의 트리를 구성한다.
- (3) 소스에서 RP까지의 최단 경로로 데이터그램을 전송한다. 즉 1번 소스는 1→2→6→8의 경로를 따라 RP로 데이터그램을 전송한다.
- (4) 위의 정보를 토대로 각각의 노드는 포워딩 테이블을 구성한다.
- (5) 데이터그램 전송시 멤버로 이르는 분기를 만나는 경우 분기를 따라 데이터그램을 전송한다. 예를 들어 1→2→6의 경로로 도착하는 데이터그램은 8번 RP로 전송하기 전에 6→7→4→5의 경로로 패킷을 전송한다.

PIM-SM(Shared)에서 멤버의 Join 가입 절차와 트리 설정 절차는 CBT와 동일하나, CBT는 양방향 트리를 구성하고 PIM-SM(Shared)에서는 단방향(센터→멤버) 트리를 구성한다. 따라서 PIM-SM(Shared)에서는 송신 호스트가 센터 라우터로 터널링(tunneling)을 통해 캡슐화된 데이터그램을 보내고 센터 라우터에서 다시 단방향 트리를 따라 각각의 멤버 호스트에게 전달된다. 즉, (그림 1)에서 CBT와 달리, PIM-SM(Shared)에서는 1→2→6의 경로로 도착하는 데이터그램을 7번 노드로 분기하지 않고 8번의 RP로 전송한 후 다시 6번 노드로 데이터그램을 전송하여 6→7→4→5의 경로로 패킷을 전송한다.

3. RP 지정 알고리즘

PIM-SM(Shared) 프로토콜의 경우 RP의 위치에 따라 소스-멤버 사이의 비용(거리, 홉수, 대역폭, 링크 가중치 등)이 달라지므로 RP 지정을 위한 여러 가지 알고리즘들이 제안되었다[4,8]. 이들 중 대표적인 알고리즘을 요약하면 다음과 같다.

- Actual Cost 방법 : 모든 멤버들을 연결하는 공유형 트리 구성시 연결된 링크의 수를 최소화하는 노드를 RP로 선정
- Max Dist 방법 : 멤버까지의 최대 거리를 최소화하는 노드를 RP로 선정
- Avg Dist 방법 : 멤버까지의 평균 거리를 최소화하는 노드를 RP로 선정
- Max Diameter 방법 : 멤버까지의 거리 중 가장 큰 두개의 거리 값의 합을 최소화하는 노드를 RP로 선정

그리고 Thaler와 Ravishankar[4]은 RP와 멤버들 사이의 비용을 예측하여 다음과 같은 알고리즘을 제안하였으며, 이는 위의 다른 방법들에 비하여 비용이나 지연시간 면에서 성능이 우수함이 입증되었다.

EstCost 알고리즘

- (1) 네트워크 상에 존재하는 각각의 노드(모든 라우터)에 대하여 다음의 비용 값을 계산한다.

$$Est\ Cost = \frac{Est\ Cost_{min} + Est\ Cost_{max}}{2} \quad (1)$$

$Est\ Cost_{min}$ 은 노드에서 각각의 멤버(M : 멤버의 집합)까지 이르는 비용($d(i,j)$ 는 노드 i 와 노드 j 사이의 비용) 중 최대값을 의미하며, 이는 해당 노드가 RP로 선정되는 경우 비용의 하한값(Lower Bound)을 나타낸다. 즉, 해당 노드가 RP인 경우 RP에서 가장 비용이 큰 멤버까지 도달하는 경로상에 모든 멤버들이 존재하는 경우가 비용의 최소값을 갖게 된다.

$$Est\ Cost_{min} = \max_{m \in M} d(RP, m) \quad (2)$$

$Est\ Cost_{max}$ 은 노드에서 각각의 멤버까지 이르는 비용들의 합을 의미하며, 이는 해당 노드가 RP로 선정되는 경우 비용의 상한값(Upper Bound)을 나타낸다. 즉, 해당 노드가 RP인 경우 RP에서 모든 멤버까지 이르는 경로가 서로 다른 경우가 비용의 최대값을 갖게 된다.

$$Est\ Cost_{max} = \sum_{m \in M} d(RP, m) \quad (3)$$

- (2) EstCost 값이 가장 작은 노드를 RP로 선택한다.

이와 같이 EstCost 알고리즘에서는 모든 노드에 대하여 식(1)의 비용 값을 구하고 이 값이 가장 작은 노드를 RP로 지정한다. 그러나 이는 멀티캐스트 그룹과 소스/멤버의 수가 많은 경우 현실적으로 적용이 불가능하다. 이러한 문제점을 해결하고 보다 동적인 상황을 반영하며, 실제로 프로토콜로 구현하기 위해서 타이머와 최대 후보 목록의 수를 관리하는 방안을 제시하였다[4, Hill-Climbing Protocol(HILLCLIMB) 프로토콜(p.293)]. HILLCLIMB 프로토콜에서는 주어진 타이머 시간 동안 일정한 수(25개)의 RP 목록을 관리하고 이 중에서 비용이 가장 작은 노드를 RP로 지정하며, 지정된 시간이 지난 후에 목록에 저장된 라우터들의 이웃 노드들을 대상으로 위의 알고리즘을 적용하여 후보 목록을 갱신하는 과정을 반복한다. 그러나 본 논문에서는 모든 노드들에 대하여 RP 지정을 위한 기준 파라미터값(식(1)의 EstCost 값)을 모두 구할 수 있다고 가정하여 시뮬레이션을 수행하였다.

EstCost 알고리즘에서는 RP 지정을 위해 멤버까지의 비용만을 고려하고 소스와 RP까지의 비용 및 각 노드에서 관리해야 할 라우팅 테이블(Routing Table)의 크기(테이블 내 entries의 수)를 고려하지 않았다. 여기서 라우팅 테이블은 멀티캐스트 라우팅 프로토콜에 의하여 구성되는 테이블(포워딩 테이블)을 의미하며, (Group id, Source id, In interface, Out interface)의 엔트리(entries)를 저장함으로써 In interface로 들어오는 데이터그램을 Out interface로 보내게 된다. 따라서 멀티캐스트 라우터에서 관리하는 라우팅 테이블의 엔트리 개수, 즉 라우팅 테이블의 크기는 멀티캐스트 그룹의 수와 그룹당 소스의 수가 많은 경우 커지게 된다[9].

멀티캐스트 그룹의 수를 N , 그룹당 평균 소스의 수를 n , 그리고 소스에서 RP까지의 평균 홉 수를 d 라고 할 때 소스에서 RP까지의 경로(트리) 설정을 위해 필요한 평균 라우팅 테이블의 크기는 $N \cdot n \cdot d$ 이 되며 전체 라우터의 수가 r 인 경우 소스에서 RP까지의 경로 설정을 위해 필요한 라우팅 테이블의 크기는 다음과 같다.

$$RT_1 = \frac{N \cdot n \cdot d}{r} \quad (4)$$

소스에서 RP까지의 경로 설정 후 멀티캐스트 그룹내의 소스들은 동일한 공유 트리(Shared Tree)를 따라 데이터를 전송한다. 공유 트리 경로 상에 존재하는 라우터의 수를 s 라고 할 때 총 엔트리의 수는 $N \cdot s$ 가 되고 전체 네트워크를 고려하여 RP에서 멤버까지 이르는 경로 상의 라우팅 테이블의 크기, RT_2 를 구한다.

$$RT_2 = \frac{N \cdot s}{r} \quad (5)$$

결국, PIM-SM(Shared)의 경우 라우팅 테이블의 크기(RT)는 다음과 같다.

$$RT = RT_1 + RT_2 = N \cdot n \cdot \frac{d}{r} + N \cdot \frac{s}{r} \quad (6)$$

본 연구에서는 식(6)에서 구한 라우팅 테이블의 크기를 고려하여 RP를 선정하는 알고리즘을 새로 제시한다. 제안하는 알고리즘에서는 각각의 노드들에 대해 식(1)의 비용을 구하고 일정한 범위내 비용 값을 만족하는 노드들 중 식(6)의 RT 값을 최소화하는 노드를 RP로 선정한다. 여기서 RP-멤버 사이의 공유형 트리 상에 존재하는 라우터의 수(s)는 아직까지 기존 프로토콜에서 구체적으로 그 정보를 알 수 없으나 멤버들이 RP로 join 메시지를 전송하는 IGMP 프로토콜상에 해당 정보를 넣음으로써 RP가 공유형 트리 구성의 정보를 얻을 수 있을 것이다. 그리고 본 논문에서는 모든 노드들에 대하여 RP 지정을 위한 파라미터 값(EstCost 및 RT)을 모두 구할 수 있다고 가정하여 시뮬레이션을 수행한다. 각각의 노드에 대한 비용 예측값의 상한을 ϵ 으로 제한하는 경우($EstCost \leq \epsilon$) 구하고자 하는 문제를 다음 (P)로 표현할 수 있다.

$$(P) \quad \begin{aligned} \text{Min } RT &= N \cdot n \cdot \frac{d}{r} + N \cdot \frac{s}{r} \\ \text{st } \max_{m \in M} d(RP, m) + \sum_{m \in M} d(RP, m) &\leq 2\epsilon \end{aligned} \quad (7)$$

위 문제 (P)에 대해 해를 구하기 위한 MinRT 알고리즘은 다음과 같다.

MinRT 알고리즘

- (1) 네트워크 상에 존재하는 각각의 노드(모든 라우터)에 대해 EstCost값을 구한다.
- (2) EstCost 값이 ϵ 보다 작은 노드를 선택한다.
- (3) 위 (2)에서 구한 노드들 중 RT 값이 가장 작은 노드를 RP로 선택한다.
- (4) 만약 위 (2)에서 ϵ 보다 작은 노드가 없는 경우 EstCost 값이 가장 작은 노드들 RP로 선택한다.

즉, 제약식을 통하여 RP와 멤버까지 예측 비용의 최대값을 ϵ 으로 제한하고 이를 만족하는 노드들 중 RT 값이 가장 작은 노드를 RP로 선택함으로써 기존의 EstCost 알고리즘보다 제안한 알고리즘에서는 일정한 범위 내로 링크 비용을 줄일 수 있고(RP-멤버 사이의 지연을 일정한 범위 내로 제한) 또한, 라우팅 테이블의 크기를 최소화할 수 있다. 제안한 알고리즘에서 RP와 멤버까지의 예측 비용을 ϵ 으로 제한한 이유는 만약, 비용을 제한하지 않고 RT 값만을 최소화하는 노드를 RP로 선정하는 경우 식(6)에서 소스와 RP 사이의 홉 수를 가능한 줄이고자 소스들에 가까운 지점에 위치하고 있는 노드를 RP로 선정하게 되어 RP와 멤버 사이의 비용이 크게 증가하게 됨으로서 소스와 멤버 사이의 전체 비용이 증가하게 된다. 이러한 문제를 방지하기 위해 RP와 멤버 사이의 예측 비용을 ϵ 으로 제한하고 이들 중 RT 값을 최소화하는 노드를 RP로 선정함으로써 총 비용을 일정한 범위 내로 줄이면서 라우팅 테이블의 엔트리 수를 최소화할 수 있다.

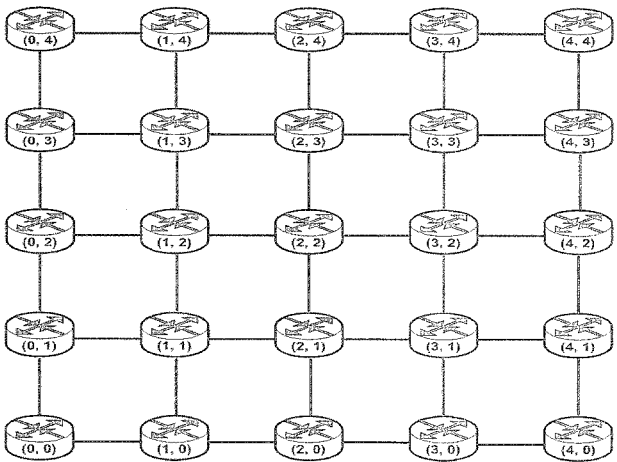
한편, 제안한 MinRT 알고리즘에서는 ϵ 값에 따라 제안한 알고리즘의 성능이 결정된다. ϵ 값이 너무 작은 경우 제약식을 만족하는 후보 RP가 존재하지 않을 수 있으며, 반면 ϵ 값이 너무 큰 경우 모든 노드들에 대해서 RT 값을 비교하여 그 값이 가장 작은 노드를 선택하게 된다. 따라서 멀티캐스트 서비스를 제공하는 네트워크의 구조를 고려하여 적절한 ϵ 값을 사전에 정의해 주어야 한다.

4. 성능분석

성능분석을 위하여 (그림 2)와 같은 Mesh 네트워크를 가정하고 그룹이 한 개($N=1$)인 경우를 고려한다. 노드 사이의 비용을 1(unit cost)이라 할 때 노드 (i,j) 와 노드 (k,l) 사이의 거리(비용), $d(i,j,k,l)$ 는 다음과 같다.

$$d(i,j,k,l) = \begin{cases} 0, & \text{if } i=j=k=l \\ |k-i|, & \text{if } j=l, k \neq i \\ |l-j|, & \text{if } i=k, j \neq l \\ |k-i|+|l-j|, & \text{otherwise} \end{cases} \quad (8)$$

앞에서 설명하였듯이 제안한 MinRT 알고리즘에서는 상수 ϵ 값에 따라 알고리즘의 성능이 변하게 되며, 제약식을 만족하는 후보 RP를 구하기 위하여 본 논문에서는 ϵ 을 다음과 같이 가정한다. (그림 2)의 Mesh 네트워크에서 노드간 최소 거리 비용을 가정하는 경우, 임의의 RP 노드와 멤버 사이의 최소 비용은 0(RP와 멤버가 동일한 경우)이며 최대 비용은 8((0,0)과 (4,4)노드 사이의 비용)이 된다. 따라서 평균적으로 임의의 노드와 모든 멤버들 사이의 비용은 $8 \times$ 멤버의 수/2가 된다. 그리고 RP와 멤버들 사이의 평균 비용의 한계를 이 값의 1/2로 하여 ' $\epsilon=2 \times$ 멤버의 수'로 가정한다. 사전에 정확한 ϵ 값을 산정하기 위해서는 망 구성, 노드간 케이블 특성 등의 네트워크 설계와 서비스 지연시간을 고려하여 적절한 한계 범위를 가정할 필요가 있으나 본 분석에서는 소스와 멤버가 랜덤하게 지정되는 경우를 가정하여 RP



(그림 2) Mesh 네트워크

와 멤버 사이의 비용 값을 예측하여 그 한계 범위를 가정하였다. 그리고 (그림 2)의 Mesh 네트워크에서 시뮬레이션을 수행한 결과 모든 경우에서 앞에서 가정한 ϵ 범위 내에 존재하는 후보 RP를 구할 수 있었다. 한편, MinRT 알고리즘에서 RT_2 를 구하기 위해 공유 트리 경로 상에 존재하는 라우터의 수(s)는 멤버까지의 거리에 비례하고 멤버의 수에 반비례한다고 가정하여 s 를 다음과 같이 구한다.

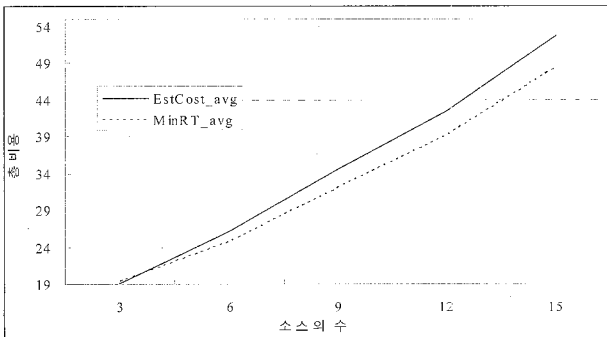
$$s = \frac{\sum_{m \in M} d(RP, m)}{\#of\ members} \quad (9)$$

멤버가 5개인 경우, 소스의 수에 따른 총비용과 라우팅 테이블의 크기를 비교(소스/멤버를 랜덤하게 지정하고, 각각의 경우에 대하여 10번의 시뮬레이션을 수행)하면 (그림 3)~(그림 5)와 같다. (그림 4)~(그림 5)에서는 알고리즘의 성능을 비교하기 위하여 총비용과 라우팅 테이블의 크기에 대한 차이($= \frac{(Est\ Cost - MinRT)}{Est\ Cost} \times 100\%$)를 나타내었다. 즉,

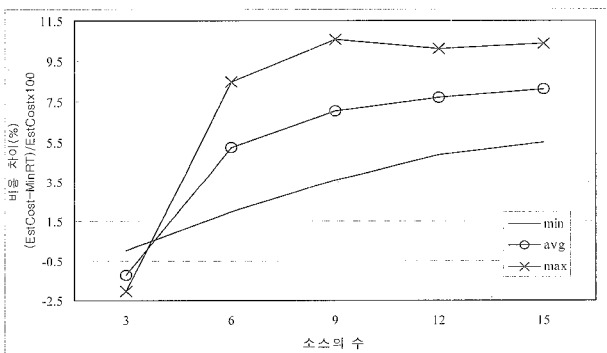
$$\frac{(Est\ Cost - MinRT)}{Est\ Cost} \times 100\%$$

은 두 알고리즘(EstCost 및 MinRT)에 대한 소스-멤버 사이의 총비용과 라우팅 테이블의 크기에 대한 상대적인 차이를 나타낸다. 상대 차이 값이 클수록 제안한 MinRT 알고리즘의 성능이 우수함을 나타내며, 이 값이 음수인 경우 기존 EstCost 알고리즘의 성능이 우수함을 의미한다. 한편, 알고리즘에서 라데부 라우터의 결정 알고리즘에 대한 판단 기준값(EstCost 및 RT)이 동일한 경우 총비용과 라우팅 테이블의 크기에 대한 최소(min), 평균(avg), 최대(max) 값을 기준으로 비교하였다. (그림 3)은 각각의 알고리즘에 대한 총비용(평균값)을 나타낸다. 소스의 수가 커짐에 따라 MinRT 알고리즘에서의 총비용이 작아짐을 알 수 있다. (그림 4)에서 소스의 수가 3인 경우를 제외하면 MinRT에서의 총비용 값이 EstCost에 비하여 항상 작으며, 소스의 수가 증가함에 따라 제안한 MinRT 알고리즘에서의 총 비용 값이 EstCost 알고리즘 보다 작음을 알 수 있다. 이는 EstCost 알고리즘에서 RP-멤버 사이의 예측 비용만을 고려하여 RP를 선정하는 반면, MinRT 알고리즘에서는 RP-멤버 사이의 비용을 일정한 범위 내로 한정하고 이들 중 RT 값이 가장 작은 노드를 선택함으로써 총 비용의 감소 효과를 얻을 수 있기 때문이다. 즉, 멤버의 수가 5개로 고정된 경우 EstCost 알고리즘에서는 멤버의 위치를 고려하여 멤버들과 가까운 위치에 있는 노드를 RP로 선정함으로써 RP-멤버 사이의 비용을 최소화 시키지만, 전체 비용(소스-RP-멤버)을 고려하지 않음으로써 소스의 수가 증가함에 따라 성능이 떨어진다. 반면, MinRT 알고리즘에서는 RP-멤버 사이의 비용을 일정한 범위($\epsilon=10$)내로 한정하고 이를 만족하는 노드들 중 RT 값이 최소화되는 노드를 선택하게 된다. 이 경우 RT 값을 구성하는 요소 중 소스와 RP사이의 홉 수에 비례하는 값인 RT_1 값을 최소화함으로써 소스-RP 사이의 비용을 최소화하는 효과를 얻어 이에 따라 전체 비용이 작아지는 결과를 얻게 된다.

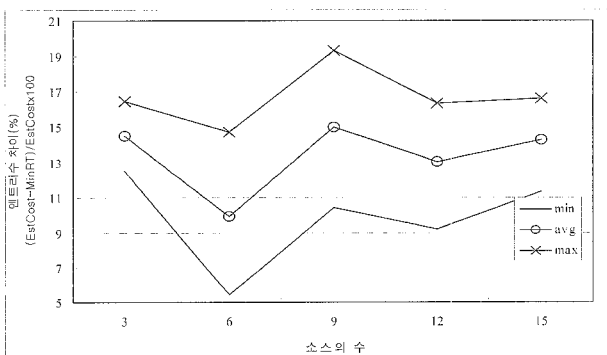
라우팅 테이블의 크기 비교 결과, (그림 5)로부터 제안한 알고리즘에서의 라우팅 테이블의 크기가 기존의 방법보다 항상 작음을 알 수 있다. 평균(avg) 값을 기준으로 볼 때, 총 비용은 5.37%, 테이블의 크기는 13.35% 정도 감소함을 나타낸다. 랜덤한 소스/멤버 선정과 식(9)에서 구한 공유 트리 상에 존재하는 라우터의 수(s)에 대한 예측 값의 오차로 인하여 소스의 수가 6개인 경우 MinRT 알고리즘에서 다소 성능이 떨어짐을 알 수 있다. 그러나 일반적으로 소스의 수가 증가함에 따라 MinRT 알고리즘에서의 라우팅 테이블의 크기가 작아짐을 알 수 있다. 이는 언급하였듯이 소스-RP 사이의 홉 수를 고려하여 RP를 선정함으로써 전체 라우터에서 관리하는 라우팅 테이블의 크기를 효과적으로 줄일 수 있음을 나타낸다. 향후 s에 대한 정확한 검증 과정(실제 서비스 제공 환경)을 거친 후 보다 실제적인 시뮬레이션을 통한 비교, 분석이 필요하다.



(그림 3) 소스의 수에 따른 총비용(평균값)

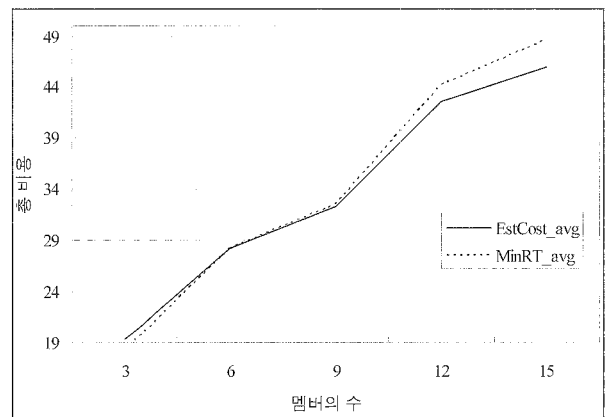


(그림 4) 소스의 수에 따른 총비용 차이

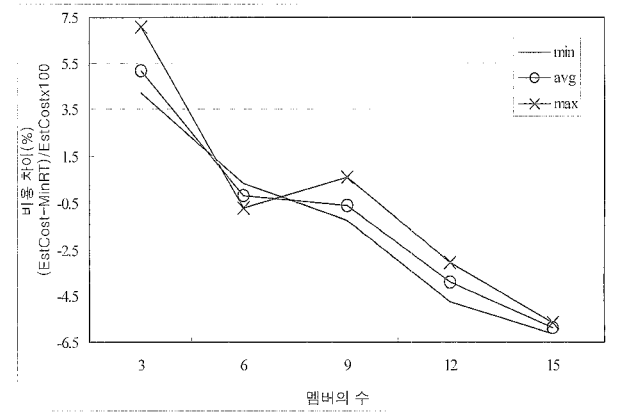


(그림 5) 소스의 수에 따른 라우팅 테이블의 크기 차이

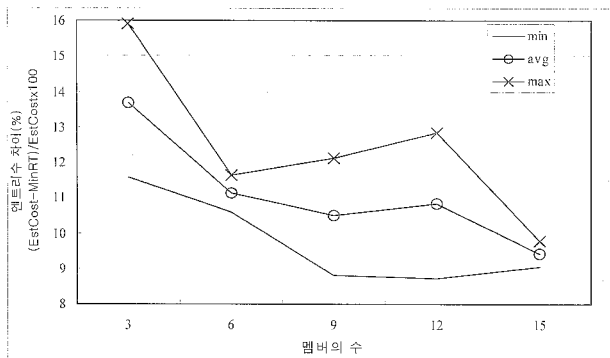
(그림 6)은 5개의 소스가 있는 경우 멤버의 수가 증가할 때의 각각의 알고리즘에 대한 총비용의 평균값을 나타낸다. 멤버의 수가 많아짐에 따라 MinRT 알고리즘의 총비용이 커짐을 알 수 있다. (그림 7)과 (그림 8)은 5개의 소스를 가정할 때 멤버의 수에 따른 결과를 보여준다. 기존 알고리즘에서는 멤버의 수가 증가함에 따라 EstCost로 구한 기준 값(RP-멤버 사이의 예측 비용)이 전체 비용(소스-RP-멤버 사이의 총비용)을 반영하게 되어 MinRT 알고리즘에 비하여 총비용이 작게 된다. 즉, 소스의 수가 5개로 고정된 경우 멤버의 수가 증가함에 따라 EstCost 알고리즘에서는 멤버들과 가까운 위치에 RP 노드를 두지만 MinRT 알고리즘에서는 소스와 RP 사이의 홉 수를 고려하여 RP를 결정하기 때문에 RP와 멤버들 사이의 비용이 EstCost 알고리즘 보다 증가하게 되며, 이로 인하여 전체 비용이 다소 증가하게 된다. 그러나 (그림 8)은 MinRT에서의 라우팅 테이블의 크기가 기존의 EstCost에서 보다 항상 작음을 나타낸다. 그리고 멤버의 수가 증가함에 따라 MinRT 알고리즘에서의 성능이 다소 떨어짐을 알 수 있다. 이는 MinRT 알고리즘에서 RT₁ 값을 기준으로 RP를 선정함으로써 RP와 멤버들 사이의 거리가 증가하게 되고 이로 인하여 공유 트리 상에 존재하는 라우터 수의 증가로 인하여 라우팅 테이블의 크기가 증가하기 때문이다. 평균 값(avg)을 기준으로 볼 때 MinRT에서의



(그림 6) 멤버의 수에 따른 총비용(평균값)



(그림 7) 멤버의 수에 따른 총비용 차이



(그림 8) 멤버의 수에 따른 라우팅 테이블의 크기 차이

총비용은 1.08% 정도 증가하지만 엔트리의 수는 11.11% 정도 감소함을 알 수 있다.

본 논문에서는 (그림 2)와 같이 5×5=25개의 라우터를 가정하여 분석하였다. 그러나 일반적인 네트워크 구조와 노드들이 많은 대규모 네트워크하에서 알고리즘의 성능에 대한 비교, 평가가 향후 보다 정밀하게 이루어질 필요가 있다. 예측컨대, 제안한 MinRT 알고리즘은 멤버의 수보다 소스의 수가 많은 경우 보다 성능이 우수한 것으로 판정되었으며, 라우팅 테이블의 엔트리 수는 기존의 EstCost 알고리즘보다 항상 작음을 알 수 있다. 본 논문에서 분석한 자료로부터 전체 25개의 노드들 중 소스가 5개(20%)인 경우 멤버의 수가 9개 이상(36%)으로 증가할 때 MinRT에서의 총비용이 EstCost 알고리즘에서의 총비용보다 커지게 됨을 알 수 있다. 이와 같이 본 논문에서 고려한 알고리즘의 성능은 네트워크의 규모보다 전체 노드들 중 소스 및 멤버의 비율에 따라 그 성능이 크게 좌우될 것으로 판단된다. 왜냐하면, 네트워크의 규모가 커지는 경우 소스/멤버들이 RP로부터 보다 멀리 분포하게 되고 이에 따라 총비용 및 RT 값이 두 알고리즘에서 모두 증가하게 되며 이로 인해 RP 선정을 위한 판단 기준(EstCost 및 RT의 상대적인 비교를 통해 RP를 선정)에 영향을 미치지 않는 때문이다.

5. 결 론

네트워크 자원의 증가와 서비스 제공 기술의 발달로 특정 다수를 대상으로 IPTV나 원격 교육 방송을 제공하는 그룹 통신 기법인 멀티캐스트의 필요성이 증대되고 있다. 지금까지 이러한 멀티캐스트 서비스를 위하여 여러 가지 프로토콜이 제안되고 있으며 최근 센터 또는 랑데부 라우터(RP: Rendezvous Point)를 중심으로 공유형 트리를 구성하여 가입자 관리 측면에서나 자원 사용면에서 효율성을 높이고자 하는 CBT 및 PIM-SM 프로토콜 사용이 권고되고 있다. 공유형 트리 구성 프로토콜에서는 RP 라우터를 지정하여 소스에서 RP 라우터까지 소스 기반 트리를 구성하고 RP에서 각각의 수신 호스트(멤버)까지 공유형 트리를 이용하여 데이터를 전송하게 된다. 따라서 RP의 위치에 따라 소스에서 멤버까지의 비용과 관리해야 할 라우팅 테이블의 크기

(entries의 수)가 달라지게 된다.

본 논문에서는 기존의 비용 값만을 이용하는 방법과 다르게 일정한 범위 내 비용을 만족하는 노드들 중에 관리해야 할 라우팅 테이블의 크기를 예측한 값이 가장 작은 라우터를 RP로 선택함으로써 보다 효율적인 RP 선정 알고리즘을 제안하였다. 성능분석을 위하여 Mesh 형의 네트워크를 구성하고 소스와 멤버의 수에 따른 알고리즘의 총비용과 라우팅 테이블의 크기를 비교하였다. 기존 방법과 비교하여 새로 제안한 알고리즘에서는 소스의 수가 증가함에 따라 비용이 5.37%, 라우팅 테이블의 크기가 13.35% 정도 감소함을 알 수 있었다. 반면, 멤버의 수가 증가하면서 기존의 방법에서는 RP와 멤버들 사이의 비용을 충분히 예측하므로 비용 감소의 효과가 다소 줄어들지만 제안한 알고리즘에서는 항상 최소의 라우팅 테이블의 크기를 보장함을 알 수 있다. 결국, 본 논문에서 제안한 알고리즘은 소스의 수가 많은 경우 유리하며, 멤버의 수가 많은 경우라도 작은 총비용(지연)의 증가로 라우팅 테이블의 크기를 최소화 하고자 하는 경우 적용을 고려 할만하다. 본 논문에서는 Mesh 형태의 네트워크를 가정하여 시뮬레이션을 수행하였으나 향후 보다 일반적인 네트워크 형태에서 지연시간을 고려한 알고리즘의 성능을 비교, 분석할 필요가 있으며 멤버의 분포에 따른 성능 평가와 함께 최적 알고리즘과의 비교가 이루어져야 하고 나아가 알고리즘을 구현하기 위하여 필요한 오버헤드 메시지를 반영하여 실제 네트워크 환경에서의 적용 방안을 검토하여야 한다. 아울러 현실적으로 RP 라우터로 지정될 수 있는 노드들이 그다지 많지 않은 네트워크 상황(네트워크상에 라우터는 아주 많지만 RP는 그 중 아주 일부인 경우)에서는 본 논문에서 제안한 방법보다는 후보 라우터들이 RP인 경우를 가정하여 총 비용을 비교함으로써 보다 효율적으로 RP를 지정할 수 있을 것으로 사료된다. 이런 상황에서는 본 논문의 결과가 현실성이 떨어지나 반면, 향후 네트워크가 복잡하고 소스 및 멤버가 동적으로 바뀌는 멀티캐스팅 서비스 제공 하에서 본 논문에서 제안한 방법론을 적용할 만하다.

참 고 문 헌

- [1] 김백현, 황태준, 김익수, "주분형 비디오 분배를 위한 웹-캐스팅 멀티캐스트 전송 기법," 정보처리학회논문지B, 제12-B권, 제7호, pp.775-782, 2005. 12.
- [2] 조성근, 최성곤, 최준근, "BcN 환경에서 효과적인 멀티캐스트 서비스를 위한 연결 수락 제어 방안," 정보처리학회논문지C, 제12-C권, 제6호, pp.793-798, 2005. 10.
- [3] 차동완, 정용주, 윤문길, "인터넷 정보 기술," 홍릉과학출판사, 2004.
- [4] D.G. Thaler and C.V. Ravishankar, "Distributed Center-Location Algorithms," *IEEE Journal on Selected Areas in Communications*, Vol.15, No.3, pp.291-303, April, 1997.
- [5] D. Waitzman, C. Partridge, and S. Deering, "Distance Vector Multicast Routing Protocol," RFC 1075, IETF,

November, 1988.

- [6] J. Moy, "Multicast Routing Extensions for OSPF," *ACM Communications*, Vol. 37, No. 8, pp.61-66, August, 1994.
- [7] J. Moy, "OSPF version 2," RFC 1583, IETF, March, 1994.
- [8] Sung Mo Chung and Chan-Hyun Youn, "Core Selection Algorithm for Multicast Routing under Multiple QoS Constraints," *Electronics Letters*, Vol.36, No.4, pp.378-379, February, 2000.
- [9] T. Billhartz, J.B. Cain, E. Farrey-Goudreau, D. Fieg, and S.G. Batsell, "Performance and Resource Cost Comparisons for the CBT and PIM Multicast Routing Protocols," *IEEE Journal on Selected Areas in Communications*, Vol.15, No.3, pp.304-315, April, 1997.
- [10] RFC 3569, "An Overview of Source-Specific Multicast (SSM)," July, 2003.

조기성



e-mail : chokis@etri.re.kr
 1982년 경북대학교 전자공학과(학사)
 1984년 경북대학교 대학원 전자공학과
 (공학석사)
 1984년~현재 한국전자통신연구원 팀장
 관심분야: BcN 통합제어기술 등

장희선



e-mail : hsjang@ptu.ac.kr
 1990년 울산대학교 산업공학과(학사)
 1992년 KAIST 산업공학과(공학석사)
 2002년 KAIST 산업공학과(공학박사)
 1992년~2002년 한국전자통신연구원
 2003년~현재 평택대학교 경상학부
 교수(e-비즈니스연구소 소장)

관심분야: 트래픽 엔지니어링, 성능분석 등

김동휘



e-mail : dwkim@daegu.ac.kr
 1981년 경북대학교 전자공학과(학사)
 1983년 경북대학교 대학원 전자공학과
 (공학석사)
 1995년 와세다대학 이공학 연구과
 (공학박사)

1983년~1996년 한국전자통신연구원
 1996년~현재 대구대학교 컴퓨터·IT 공학부 교수
 관심분야: BcN 통합제어기술 등