

지연시간을 고려한 최소비용의 동적 멀티캐스트 라우팅 알고리즘

(A Distributed Low-cost Dynamic Multicast Routing Algorithm with Delay Constraints)

申民宇*, 林炯錫**

(Min-Woo Shin and Hyeong-Seok Lim)

요약

화상회의와 같은 실시간 멀티미디어 응용들은 종단간 엄격한 지연시간 준수를 요구하며 많은 네트워크 자원을 소비한다. 이러한 응용들의 효율적 관리를 위해 멀티캐스트 프로토콜은 주어진 종단간 지연시간을 만족하면서 멀티캐스트 트리의 비용을 최소화하는 라우팅 알고리즘을 필요로 한다. 하지만 그런 트리를 찾는 문제는 계산비용이 매우 높다고 알려져 있다. 그러므로 본 논문에서는 주어진 종단간 지연시간을 만족하는 멀티캐스트 트리를 찾는 계산비용을 감소하면서 트리의 평균 비용을 최소화하는 휴리스틱 분산형 멀티캐스트 라우팅 알고리즘을 제시한다. 또한 시뮬레이션을 통해 제시한 알고리즘이 기존 알고리즘에 비해 트리의 평균 비용 측면에서 우수함을 보인다.

Abstract

Many real-time multimedia applications, such as video conferencing have stringent end-to-end delay constraints and consume large amount of network resources. In order to support these applications efficiently, multicast routing algorithms computing least cost multicast trees that satisfy a given end-to-end delay constraint are needed. However, finding such a tree is known to be computationally expensive. Therefore, we propose a heuristic distributed multicast routing algorithm that reduces a "finding multicast tree" that satisfies a given end-to-end delay constraint and minimizes the average resulting tree cost. Also, simulation results show that the proposed algorithm has much better average cost performance than other existing algorithms.

Keywords : Routing, Delay Constraints, Distributed Algorithms

I. 서론

최근 들어 화상회의나 원격교육과 같은 실시간 멀티

미디어 응용들이 많이 소개됨에 따라 네트워크 라우팅 알고리즘의 중요성이 부각되고 있다. 이러한 응용들은 보통 엄격한 지연시간 준수를 요구하며 많은 네트워크 자원을 소비한다. 그러므로 이런 응용들의 효율적인 관리를 위해 주어진 종단간 지연시간을 만족하면서 멀티캐스트 트리의 비용을 최소화할 수 있는 멀티캐스트 라우팅 알고리즘이 필요하다.

보통 멀티캐스트 라우팅 알고리즘은 임의의 그래프에서 특정 그룹에 속한 노드들을 최소의 비용으로 연결하는 문제로 생각할 수 있으며 경로설정의 계산방식에 따라 중앙 집중형과 분산형으로 나뉜다. 중앙 집중

* 正會員, 東新大學校 컴퓨터應用學科
(Dept. of Computer Science, Dongshin Univ.)

** 正會員, 全南大學校 電算學科
(Dept. of Computer Science, Chonnam National Univ.)

※ 이 논문은 2000년도 한국학술진흥재단의 지원에 의하여 연구되었음(KRF-2000-041-E00299)

接受日字:2002年2月5日, 수정완료일:2002年3月26日

형 라우팅 방식은 한 노드가 전체 네트워크에 대한 모든 상태 정보를 유지하며 그 정보를 이용하여 목적 노드까지의 경로를 설정한다. 그러므로 라우팅 속도는 빠르지만 전체 네트워크 상태 정보의 유지에 따른 오버헤드가 따른다. 반면 분산형 라우팅 방식은 각 노드가 네트워크 상태의 부분적인 정보만을 유지하며 메시지 교환에 의하여 라우팅 경로를 설정한다. 따라서 중앙 집중형에 비하여 라우팅 속도는 느리지만 네트워크 정보 유지에 대한 오버헤드는 감소한다. 또한 멀티캐스트 라우팅 알고리즘은 멀티캐스트 세션 동안 세션에 참여하는 멤버들의 변화 유무에 따라 동적 라우팅 알고리즘과 정적 라우팅 알고리즘으로 분류된다.

지난 수년동안 중단간 지연시간을 고려한 많은 멀티캐스트 라우팅 알고리즘이 제안되었지만¹⁻³⁾ 대부분 멀티캐스트 그룹의 동적인 변화는 고려하고 있지 않다. 현재 인터넷(Mbone)상에서 사용중인 멀티캐스트 라우팅 프로토콜들은 멀티캐스트 그룹의 동적인 변화를 잘 지원할 수 있지만 대부분의 멀티캐스트 라우팅 알고리즘들⁴⁻⁹⁾은 최선형(best-effort) 서비스만을 우선으로 설계되고 있으므로 멀티캐스트 그룹의 동적인 변화나 응용들의 실 시간성을 보장하지는 못한다. 뿐만 아니라 현존하는 대부분의 동적 멀티캐스트 라우팅 알고리즘들은 경로선택 기준으로 오직 하나의 인자(링크 비용 또는 링크 지연시간)만을 고려하고 있다. 처음으로 WAVE¹⁰⁾가 하나 이상의 인자, 즉 링크의 비용과 지연시간을 고려한 알고리즘을 제안하였지만 주어진 중단간 지연시간을 만족하면서 최소비용을 찾는 알고리즘이라고는 할 수 없다. 최근에 Quan Sun이 주어진 중단간 지연시간을 만족하면서 최소비용 경로를 찾는 유니캐스트 라우팅 알고리즘(DCR)¹¹⁾을 제안하고, 이를 기반으로 효율적인 동적 멀티캐스트 라우팅 알고리즘(DCDMR)¹²⁾을 제안하였다. 하지만 DCDMR은 특정한 경우 임의의 노드에서 데이터 중복현상이 발생할 수 있고, 멀티캐스트 트리의 전체 비용이 DCR의 성능에 전적으로 의존하는 단점을 가지고 있다.

본 논문에서는 DCDMR의 문제점을 해결하고 멀티캐스트 그룹 멤버의 동적인 변화에 잘 대처할 수 있으며 현존하는 거리벡터(distance vector) 기반의 라우팅 프로토콜에 잘 부합되는 분산형 동적 멀티캐스트 라우팅 알고리즘을 제시한다.

본 논문은 2장에서 네트워크 모델 및 문제정의, 3장에서 관련연구, 4장에서 제안 알고리즘 기술, 5장에서

시뮬레이션 결과, 6장에서 결론 및 연구과제로 구성된다.

II. 네트워크 모델 및 문제정의

본 논문의 네트워크 모델과 문제정의는 [12]에 정의된 것과 유사하다. 각 링크(E)는 전송 지연시간, 비용인자를 가지며 네트워크는 이런 링크들로 각 노드(V)들을 연결한 방향 그래프 $G=(V, E)$ 로 표현된다. 각 링크들의 인자를 위해 다음과 같은 두 함수가 사용된다.

- ▶ $D(e)$: 해당 링크의 데이터 패킷 전송지연시간
- ▶ $C(e)$: 해당 링크의 자원(resource) 활용도

소스노드 $s(s \in V)$ 와 목적노드의 집합 $D(D \subseteq V - s)$ 가 주어졌을 때, s 를 루트로 하고 D 에 속한 모든 노드를 연결한 트리 $T(T \subseteq G)$ 를 멀티캐스트 트리라 하며, 이때 트리 T 의 비용은 다음과 같이 정의된다.

$$C_T = \sum_{e \in T} C(e)$$

D 에 속한 노드들을 보통 멀티캐스트 그룹 멤버라 하고, $|D|$ 를 멀티캐스트 그룹 크기라 한다. $P_T(s, v)$ 를 멀티캐스트 트리 T 상에서 s 에서 $v(v \in D)$ 까지의 유일한 경로라 하면, 다음 조건을 만족하는 트리 T 를 지연시간을 고려한 멀티캐스트 트리(DCMT, Delay-Constrained Multicast Tree)라 한다.

$$\sum_{e \in P_T(s, v)} D(e) < \Delta \text{ (주어진 지연시간 제한조건)} \quad \forall v \in D$$

위의 조건을 만족하면서 최소비용을 갖는 멀티캐스트 트리를 찾는 문제는 NP-complete^[11]임이 알려져 있다.

본 논문은 DCMT이면서 최소비용을 갖는 멀티캐스트 트리를 찾는 문제로서 동적인 환경을 고려한 경우이므로 구체적인 문제정의를 하면 다음과 같다.

문제정의 : 방향 네트워크 $G=(V, E)$ 에 양의 값을 갖는 링크 비용 C 와 링크 지연시간 D (C 와 D 는 시간에 따라 변할 수 있음), 그리고 전송 지연시간 제한조건 (Δ)이 주어진 경우, T_0 를 소스노드 $s(s \in V)$ 를 포함하는 초기 DCMT라 하자. 이때 $r(t_i)$ 를 시간 t_i 에 그룹 멤버를 추가하거나 삭제하고자 하는 임의의 요청이라 하면, 일련의 요청 $\{r(t_1), r(t_2), \dots, r(t_i), \dots\}$, ($t_1 <$

t_2, \dots, t_i, \dots)가 발생할 경우, 모든 가능한 DCMT중 비용이 최소가 되는 일련의 DCMT $\{T_1, T_2, \dots, T_i, \dots\}$ 를 찾아라(단 T_i 는 $r(t_i)$ 의 발생시 비용이 최소가 되도록 T_{i-1} 을 수정한 DCMT).

위의 문제를 해결하는 가장 간단한 방법은 매번 요청이 발생할 때마다 [2]처럼 정적 DCMT를 구하는 알고리즘을 적용하는 것이다. 하지만 이런 방법은 트리의 재조정을 위해 현재의 멀티캐스트 세션들이 상당 부분 파괴될 가능성이 있으므로 이런 방법은 실시간 멀티미디어 응용에는 적합하다고 할 수 없다. 따라서 기존 멀티캐스트 세션의 파괴를 가능한 줄일 수 있는 효율적인 동적 멀티캐스트 라우팅 알고리즘이 필요하다. 다음 장에서는 이런 알고리즘들 중 하나이면서 본 논문의 모태가 된 DCDMR에 대해 살펴본다.

III. 관련연구

DCDMR 알고리즘은 경로설정을 위해 네트워크를 구성하는 각 노드 ($v_i \in V$)가 라우팅 정보로서 모든 전출링크에 대한 지연시간 및 네트워크의 제한된 정보인 지연벡터(delay vector)와 비용벡터(cost vector)를 유지

한다. 이 벡터들은 $|V|-1$ 개의 entry로 구성되며 한 entry는 네트워크를 구성하는 임의의 노드 $v_j \in V(v_j \neq v_i)$ 에 대한 정보를 포함한다. 노드 v_i 에서 임의의 노드 v_j 를 위한 하나의 entry가 포함되어야 할 지연벡터 및 비용벡터의 정보는 다음과 같다.

■ 지연벡터 정보

- ① 노드 v_i 에서 v_j 까지의 최소 지연시간 경로 ($P_{id}(v_i, v_j)$)상의 링크 지연시간의 합 : $D(P_{id}(v_i, v_j))$
- ② 노드 v_i 에서 v_j 까지의 최소 지연시간 경로상의 링크 비용의 합 : $C(P_{id}(v_i, v_j))$
- ③ 노드 v_i 에서 v_j 까지의 최소 지연시간 경로상의 다음 노드의 주소 : $id(P_{id}(v_i, v_j))$

■ 비용벡터 정보

- ① 노드 v_i 에서 v_j 까지의 최소 비용 경로 ($P_{ic}(v_i, v_j)$)상의 링크 지연시간 합 : $D(P_{ic}(v_i, v_j))$
- ② 노드 v_i 에서 v_j 까지의 최소 비용 경로상의 링크 비용의 합 : $C(P_{ic}(v_i, v_j))$

$req \leftarrow$ 기존 멀티캐스트 트리 T 에 참여하고자하는 요청 메시지
 $v_n \leftarrow req$ 를 발생시킨 노드
 $D(P_T(s, v_i)) \leftarrow T$ 상에서 s (소스노드)에서 v_i (T 의 임의의 노드)까지의 지연시간
(단, 이 값은 각 v_i 가 이미 알고 있다고 가정)
 $v_p \leftarrow v_n$ 과 지연시간으로 최단경로에 위치한 T 의 임의의 노드

- step1) Set $v_i \leftarrow v_p$
step2) If $v_i = s$ and $D(P_{id}(s, v_n)) \geq \Delta$, there exists no delay-constrained path from s to node v_n and DCDMR terminates; otherwise go to next step
step3) If $D(P_T(s, v_i)) + D(P_{id}(v_i, v_n)) < \Delta$, node v_i computes a path from itself to the node v_n satisfying the delay constraint of $\Delta - D(P_T(s, v_i))$ using DCR; otherwise go to next step
step4) v_i sends the received req to v_i 's parent node v_p on the existing tree.
Set $v_i \leftarrow v_p$ and go to step2)
-

그림 1. DCDMR 알고리즘
Fig. 1. DCDMR algorithm.

③ 노드 v_i 에서 v_j 까지의 최소 비용 경로상의 다음 노드의 주소 : $id(P_{ic}(v_i, v_j))$

위의 정보들을 활용하여 DCDMR 알고리즘은 임의의 노드에서의 경로설정을 다음과 같이 처리한다.

그림 1의 step 3)에서 보듯이 DCDMR의 실질적인 노드간의 경로설정은 지연시간을 고려한 분산형 유니캐스트 알고리즘인 DCR을 사용하는데, 경로설정 요청 메시지를 받은 임의의 노드에서 목적 노드까지의 개략적인 DCR 알고리즘은 다음과 같다.

■ DCR 알고리즘

설명의 편의를 위해 현재 경로설정 요청 메시지를 받은 노드를 *active_node*, 목적노드를 d, 소스 노드에서 *active_node*까지 이미 설정된 경로의 지연시간의 합을 *delay_so_far*라 하자. 이때 *active_node*에서의 다음노드 선택은 $delay_so_far + D(P_{ic}(active_node, d))$ 가 지연 시간 제한조건을 만족할 경우 $id(P_{ic}(active_node, d))$ 가 되고, 그렇지 않을 경우에는 $id(P_{id}(active_node, d))$ 가 된다. 그 후 선택된 노드에 $(d, \Delta, delay_so_far)$ 를 포함하는 경로설정 요청 메시지를 보내고 위의 과정을 목적노드가 경로설정 요청 메시지를 받을 때까지 반복한다.

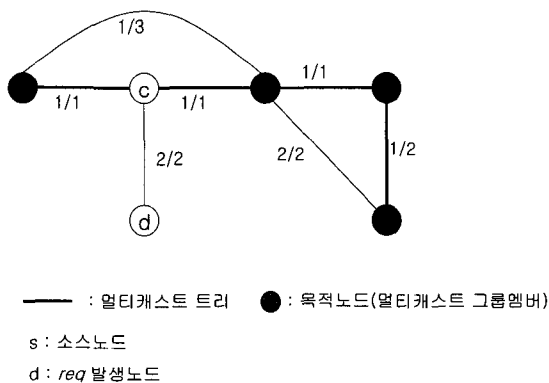


그림 2. Δ 가 5인 경우 메시지 중복의 예(단 링크에 부여된 값은 비용/지연시간임)
Fig. 2. An example of the packet duplication with a delay bound of 5(links are cost/delay).

DCDMR 알고리즘은 매우 간단하면서도 효율적인 알고리즘이지만 전체 멀티캐스트 트리의 비용이 전적

로 DCR에 의존적이며 특정한 경우에 메시지 중복 현상이 발생할 수 있는데 그림 2는 노드 d가 기존 멀티캐스트 트리에 참여 하고자 할 때 발생하는 메시지 중복현상을 보여준다.

그림 2에서 보듯이 c 노드가 d 노드와 지연시간으로 최단경로의 위치에 있지만 $D(P_{ic}(s, c)) + D(P_{id}(c, d)) > \Delta$ 이므로 알고리즘의 step 4)에 따라 d 노드로부터 받은 req를 단지 자신의 부모노드인 b 노드에 전달한다. b 노드, a 노드도 같은 상황이므로 req는 s 노드까지 전달된다. s 노드에서는 $D(P_{id}(s, d)) < \Delta$ 를 만족하므로 소스 노드를 s로 하고 목적 노드를 d로 하여 DCR이 수행된다. 그 결과 $s \rightarrow b \rightarrow c \rightarrow d$ 의 경로가 설정되는데 이렇게 되면 노드 b에서 메시지 중복이 발생됨을 알 수 있다.

그러므로 본 논문에서는 DCDMR의 데이터 중복 현상을 해결하고, 멀티캐스트 트리의 평균 비용 개선을 위해 경로의 비용 측면에서 DCR보다 효율적인 유니캐스트 알고리즘을 제시한다.

IV. 제안 알고리즘

본 논문에서 제안한 알고리즘(DDMR, Delay constrained Dynamic Multicast Routing algorithm)은 경로 설정을 위한 라우팅 정보로 3장에서 언급한 비용벡터 및 지연벡터를 이용한다. 이 벡터들은 실제로 DVMRP와 같은 프로토콜에서 거리벡터(distance vector)를 수집하는 절차를 이용하면 쉽게 획득할 수 있다. 그러나 본 논문은 이런 벡터들을 입력으로 하는 라우팅 알고리즘을 다룬 것임으로 지연벡터나 비용벡터 유지에 대한 자세한 절차는 언급하지 않기로 한다.

DDMR은 DCDMR처럼 임의의 멀티캐스트 그룹 멤버들 간의 경로 설정이 유니캐스트 라우팅 알고리즘에 기반 하므로 먼저 DDMR 알고리즘에 사용되는 유니캐스트 라우팅 알고리즘 즉, DDUR(Distributed Delay constrained Unicast Routing algorithm)에 대해 먼저 언급한 후 DDMR 알고리즘에 대해 살펴보도록 한다.

1. DDUR 알고리즘

Quan Sun의 DCR 알고리즘은 임의의 노드에서 경로 설정이 목적 노드까지의 최소비용 경로 또는 최소 지연시간 경로상의 다음 노드로만 제한되기 때문에 경로 설정의 계산 시간은 빠른 반면 경로의 비용 측면에서

는 다소 비효율적이라 할 수 있다. 따라서 DDUR 알고리즘에서는 다음 노드의 선택 기준을 보다 확장함으로써 전체적인 경로의 비용을 감소시킨다. DDUR 알고리즘은 소스 노드에서 목적 노드까지의 경로 설정이 매번 하나의 노드씩 결정되며, 경로설정 요청 메시지를 받은 임의의 *active_node*에서 목적 노드로의 다음 노드 선택 기준은 다음과 같다.

■ $D(P_{id}(active_node, d)) \leq \Delta$ 이면 경로 설정의 다음 노드는 DCR 알고리즘과 같이 $id(P_{ic}(active_node, d))$ 가 된다.

■ $D(P_{ic}(active_node, d)) > \Delta$ 인 경우는 *active_node*와 이웃한 노드들을 $v_i, (i=1, \dots, k)$ 라 할 때, 각 v_i 에 비용계산 요청 메시지를 보내고 비용계산 요청 메시지를 받은 v_i 는 아래와 같은 방법으로 비용을 계산한 후 계산된 비용을 포함하는 응답 메시지를 *active_node*에 되돌려 보낸다. *active_node*는 응답 메시지 중 목적노드까지 최소 비용을 유도할 수 있는 v_i 를 선택하여 경로 설정의 다음 노드로 선택한다.

(v_i 의 비용 계산) 만약 $delay_so_far + D(active_node, v_i) + D(P_{id}(v_i, d)) > \Delta$ 이면 목적 노드까지의 경로를 구할 수 없으므로 "path not found"라는 응답 메시지를 *active_node*에 보낸다. 그렇지 않은 경우, $delay_so_far + D(active_node, v_i) + D(P_{ic}(v_i, d)) \leq \Delta$ 를 만족하면 v_i 의 비용을 $C(active_node, v_i) + C(P_{ic}(v_i, d))$ 로 설정하고, 만족하지 않으면 $C(active_node, v_i) + C(P_{id}(v_i, d))$ 로 설정한다.

그 후 선택된 노드에 경로설정 요청 메시지를 보내고 위의 과정을 목적노드가 경로설정 요청 메시지를 받을 때까지 반복한다. DDUR은 임의의 노드에서 목적 노드까지의 최소 비용 경로가 지연시간 제한 조건을 만족하면 DCR과 같은 경로를 구하게 되지만 그렇지 않은 경우, 시뮬레이션 결과 약 40%정도의 비용 개선 효과가 있음을 알 수 있었다. 그림 3은 서로 다른 경로를 구한 경우, DDUR에 대한 DCR의 초과비용 비율의 시뮬레이션 결과를 보여준다. 그러나 최악의 경우 메시지 복잡도 측면에서 보면 DCR은 $O(|V|)$ 인 반면, DDUR은 경로설정 과정의 모든 노드에서 비용 계산 요

청 메시지가 발생할 경우, 자신과 이웃한 모든 노드들로 요청 메시지를 전달해야 되므로 $O(|E|)$ 가 된다(단, $|E|$ 는 네트워크를 구성하는 링크의 수).

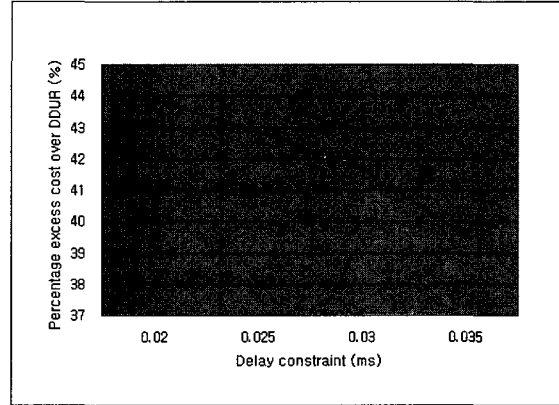


그림 3. DDUR에 대한 초과 비용 비율(노드수가 50인 경우)

Fig. 3. Percentage excess cost over DDUR(number of nodes = 50).

2. DDMR 알고리즘

DDMR 알고리즘은 데이터 중복이 발생하지 않을 경우 DCDMR 알고리즘과 유사하다. 즉, DCDMR의 step3)에서 DCR을 사용하는 대신 DDUR을 사용하면 된다. 그러나 DDUR 수행 도중 기존의 멀티캐스트 트리에 속한 노드를 만나면 DCDMR처럼 데이터 중복이 발생하는데 이런 경우 데이터 중복을 제거하고 멀티캐스트 트리의 재조정을 위해 알고리즘의 수정이 필요하게 된다. 이를 위해 멀티캐스트 트리를 구성하는 각 노드들은 경로상의 다음노드에 대한 정보뿐만 아니라 목적노드 리스트를 추가로 유지해야 한다. 목적노드 리스트의 유지는 경로설정 요청메시지를 받을 때마다 목적노드 리스트에 목적노드를 포함하도록 DDUR 알고리즘을 약간 수정함으로써 가능하다. 그림 4는 데이터 중복 발생 시 데이터 중복을 해결하는 알고리즘을 기술한 것이다(단 3장에서 사용한 용어를 그대로 사용하기로 한다).

위 알고리즘을 그림 2에 적용시켜보면 b 노드에서 데이터 중복이 발생되므로 먼저 $slack(=1)$ 을 구하고 $D(P_r(s, b))$ 의 값을 3에서 2로 수정한다. 그 후 부모노드와의 연결을 끊고 b 노드에서 유지하고 있는 목적노드 리스트 e,d에 대하여 b 노드를 *active_node*로 하여

- $v_d \leftarrow D(P_T(s, v_i)) + D(P_{id}(v_i, v_n)) < \Delta$ 인 노드 v_i 에 대해 DDUR 수행 중 데이터 중복을 발생시킨 노드
 - $D_{DDUR}(v_i, v_d) \leftarrow$ DDUR적용 시 두 노드 (v_i, v_d) 사이의 지연시간
 - $slack \leftarrow D(P_T(s, v_d)) - (D(P_T(s, v_i)) + D_{DDUR}(v_i, v_d))$
- F1) v_d 에서의 $slack$ 을 계산한다.
 F2) $D(P_T(s, v_d))$ 를 $D(P_T(s, v_d)) - slack$ 으로 수정한다.
 F3) v_d 와 부모 노드와의 연결을 끊는다.
 F4) v_d 를 *active_node*로 하여 v_d 의 각각의 목적노드 리스트들에 대해 DDUR 알고리즘 수행한다.

그림 4. 데이터 중복 제거 알고리즘
 Fig. 4. An algorithm of removing packet duplication.

DDUR을 수행한다. 그 결과 멀티캐스트 트리가 그림 5 처럼 재조정된다. 그림 5에서 보면 데이터 중복 해결 과정에서 노드 b까지의 지연시간이 감소됨으로써 노드 b와 노드 e의 경로가 $b \rightarrow c \rightarrow e$ 에서 $b \rightarrow e$ 로 변하게 되고 그로 인해 전체 멀티캐스트 트리의 비용이 감소됨을 알 수 있다.

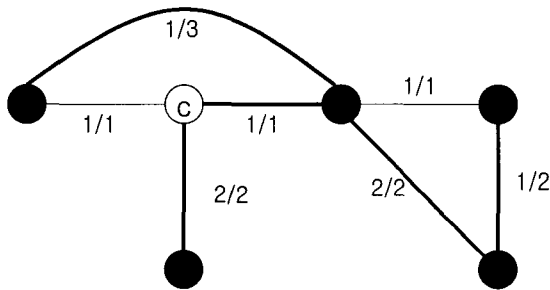


그림 5. 데이터 중복 해결 후의 멀티캐스트 트리
 Fig. 5. Multicast tree after removing packet duplication.

V. 시뮬레이션

본 논문에서는 시뮬레이션을 위해 North Carolina State University에서 개발한 멀티캐스트 라우팅 시뮬레이터 MCRSIM^[13]을 사용하였다. 또한 네트워크 생성을 위해 [14]에서 제시된 랜덤 그래프 생성기를 사용하였다. 이 생성기는 처음에 좌표상의 임의의 위치에 노드들을 생성시키고 아래와 같은 확률로 노드들간의 링크를 연결한다.

$$P(u, v) = \beta \exp \frac{-l(u, v)}{La} \quad \alpha, \beta \in (0, 1]$$

여기서 $-l(u, v)$ 는 노드 u 에서 노드 v 사이의 거리이고, L 은 임의의 두 노드사이의 최대 거리이다. α 와 β 는 $(0, 1]$ 사이의 값으로 α 가 클수록 먼 거리에 있는 노드간의 링크 존재 비율이 증가하고 β 가 클수록 그래프의 평균 차수가 증가한다. 네트워크 환경은 링크들의 수용능력이 동일하게 155Mbps인 전 이중성 ATM망을 가정하였고 각 노드들은 non-blocking ATM 스위치를 나타내었다. 또한 각 링크는 1Mbps에서 125Mbps 사이의 background traffic을 갖도록 하였고 각 링크의 비용 $C(e)$ 는 reserved bandwidth를, 지연시간 $D(e)$ 는 propagation delay를 사용하였다. 그리고 현 네트워크 라우터들의 평균 차수가 4~5정도임을 감안하여 노드의 평균 차수를 5로 설정하였다. 실험에는 멀티캐스트 트리의 평균 비용 비교를 위해 본 논문의 동기가 된 DCDMR과 정적 멀티캐스트 라우팅 알고리즘으로서 비용 측면에서 최적으로 알려진 BSMA^[2]를 포함시켰다.

초기 멀티캐스트 트리 T_0 는 무작위로 선택된 s 노드만을 포함시켰으며, 그 후 일련의 요청 $\{r_1, r_2, \dots, r_m\}$ (단, $r_i \neq r_j$ if $i \neq j$ for $i, j = 1, 2, \dots, m$)을 랜덤하게 발생시켰다. 각 요청 r_i 에 따른 새로운 멀티캐스트 트리 T_i 는 T_{i-1} 에 DCDMR 및 DDMR을 적용하여 생성시켰지만, 정적 알고리즘 BSMA를 위해서는 일련의 요청 순서와 상관없이 바로 T_m 을 생성하였다. 이런 방식으로 T_m 의 비용에 대해 95% 신뢰수준을 사용하여 실험을 반복하였다.

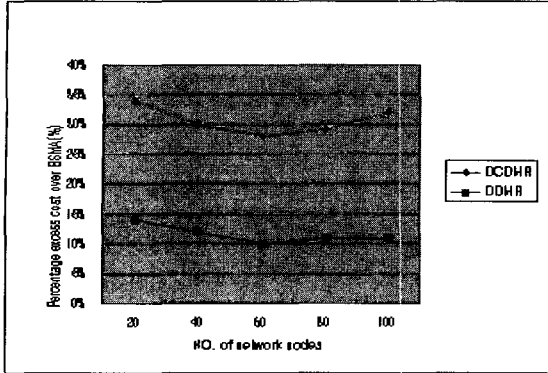


그림 6. req가 15인 경우 노드 수의 변화에 따른 BSMA에 대한 초과 비용 비율

Fig. 6. percentage excess cost over BSMA versus number of req., = 15.

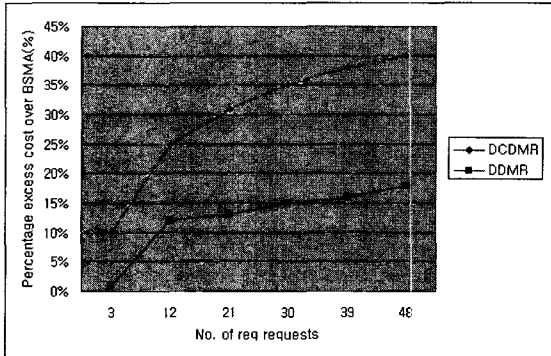


그림 7. 노드 수가 60인 경우 req.변화에 따른 BSMA에 대한 초과 비용 비율

Fig. 7. Percentage excess cost over BSMA versus number of req., for number of network nodes = 60.

그림 6은 비교적 적은 수의 요청($req = 15$)이 발생한 경우, 네트워크를 구성하는 노드 수의 변화에 따른 BSMA에 대한 초과 비용 비율을, 그림 7은 네트워크를 구성하는 노드 수를 고정시킨 상태에서 요청 수의 변화에 따른 BSMA에 대한 초과 비용 비율을 보여주고 있다(단, 두 경우 모두 $\Delta = 50ms$). 그림에서 보듯이 시뮬레이션의 결과, 요청의 수나 노드의 수가 증가하더라도 BSMA에 대한 초과 비용 비율이 크게 증가하지 않고 두 경우 모두 DCDMR에 비해 약 20% 정도의 비용 개선 효과가 있음을 알 수 있었다.

VI. 결론 및 추후연구

본 논문에서는 지연시간을 고려한 최소비용 동적 멀

티캐스트 라우팅 알고리즘(DDMR)을 제안하였다. DDMR은 멀티캐스트 그룹 멤버의 변화에 따른 기존의 멀티캐스트 트리의 변화 정도가 비교적 적은 동적 알고리즘이며, 소스노드가 경로설정을 위해 많은 계산을 필요로 하지 않기 때문에 확장성도 좋은 알고리즘이라 할 수 있다. 또한 시뮬레이션을 통한 비교 분석 결과, 멀티캐스트 트리의 평균 비용 측면에서도 라우팅 환경이 비슷한 DCDMR에 비해 약 20%정도의 성능 향상을 보임을 알 수 있었다.

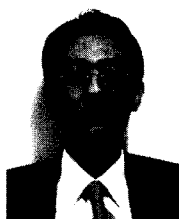
앞으로의 연구 방향은 DDUR이나 DDMR 알고리즘의 특성을 지연시간을 고려한 브로드캐스팅(broadcasting) 알고리즘에 적용한다거나 또는 Mobile이나 Ad Hoc 네트워크 환경으로 확장하는 것이라 할 수 있다.

참고 문헌

- [1] Kompella, V. P., et al., Multicast Routing for Multimedia Communication, *IEEE/ACM Transactions on Networking*, 1(3), pp. 286~292, 1993.
- [2] Zhu, Q., et al., A Source-Based Algorithm for Near-Optimum Delay Constrained Multicasting, *IEEE INFOCOM'95*, pp. 377~385, 1995.
- [3] Wi, S. and Choi, Y., A Delay Constrained Distributed Multicast Routing Algorithm, *12th International Conference on Computer Communication, ICC'95*, pp. 833~838, 1995.
- [4] S. Deering., Multicast routing in Datagram Internetworks and Extended LANs, *ACM Transactions on Computer Systems*, Vol. 8, No. 2, pp. 85~110, May, 1990.
- [5] D. Waitzman, C. Partridge, and S. Deering, Distance Vector Multicast Routing Protocol, RFC1075, Nov., 1988.
- [6] S. Deering, Host Extensions for IP Multicasting, RFC1112, Aug., 1989.
- [7] D. Estrin, Y. Rekhter, and S. Hotz, A unified Approach to Inter-Domain Routing, RFC1322, May. 1992.
- [8] A. Ballardie, P. Tsuchiya, and J. Crowcroft, Core Based Trees(CBT) - Scalable Multicast Routing, Internet Draft, Apr., 1993.

- [9] A. Ballardie, P. Francis, and J. Crowcroft, Core Bases Trees(CBT) - An Architecture for Scalable Inter-Domain Multicast Routing, *ACM SIGCOMM'93*, Sep., 1993.
- [10] Biersack, E. and Nonnenmacher J., WAVE: A New Multicast Routing Algorithm for Static and Dynamic Multicast Groups, in *proceedings of 5th Workshop on Network and Operating System Support for Digital Audio and Video*, 1995.
- [11] Q. Sun and H. Langendorfer., A New distributed Routing Algorithm for Supporting Delay-Sensitive Applications, *International Report, Institute of Operating Systems and Computer Networks*, TU Braunschweig, Germany, March 1997.
- [12] Q. Sun and H. Langendorfer., A Distributed Delay-Constrained Dynamic Multicast Routing Algorithm, in *European Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services (IDMS'97)*, 1997.
- [13] Salama, H.F., et al, Evaluation of Multicast Routing Algorithms for Real Time Communication on High-Speed Networks, in *High Performance Networking VI, IFIP 6th International Conference on High Performance Networking*, pp. 27~42, 1995.
- [14] Waxman, B.M., Routing of Multipoint Connections, *IEEE Journal on Selected Areas in Communications*, 6(9), pp. 1617~1622, 1988.

저 자 소 개



申民宇(正會員)

1990년 전남대학교 전산학과 학사.
1992년 전남대학교 전산학과 석사.
1996년 12월~1998년 8월 (주)멀티
정보 연구개발 팀장. 2000년 2월~현
재 동신대학교 컴퓨터응용학과 전임
강사. <주관심분야: 그래프이론, 라

우팅 프로토콜 및 알고리즘>



林炯錫(正會員)

1983년 서울대학교 컴퓨터공학과 졸업(공학사). 1985년 한국과학기술원 전산학과 졸업(공학석사). 1993년 한국과학기술원 전산학과 졸업(공학박사). 1996년~1997년 Purdue Univ. 방문교수. 1987년 현재 전남대학교

컴퓨터 정보학부 교수. <주관심분야: 그래프이론, 분산 및 병렬처리, 암호이론>