

# 트리 기반 애드혹 멀티캐스트 라우팅 프로토콜을 위한 멀티캐스트 데이터 포워딩의 구현

김영민<sup>o</sup> 안상현

서울시립대학교 컴퓨터과학부

{bhole<sup>o</sup>, ahn}@venus.uos.ac.kr

## Implementation of Multicast Data Forwarding for Tree-Based Ad Hoc Multicast Routing Protocol

Youngmin Kim<sup>o</sup> Sanghyun Ahn

Dept. of Computer Science, University Of Seoul

### 요 약

이동 애드혹 환경에서의 멀티캐스트 라우팅 테이블을 이용하여 패킷을 전달하기 위해서는 멀티캐스트 데이터 포워딩이 지원되어야 한다. 무선 환경에서의 멀티캐스트 데이터 포워딩은 유선 환경에서의 멀티캐스트 데이터 포워딩과는 차이가 있다. 유선 환경에서 노드의 네트워크 인터페이스는 다른 노드의 네트워크 인터페이스와 1대1로 연결되고, 네트워크 인터페이스로 들어온 패킷이 다른 노드로 전달되어야 한다면 해당되는 다른 네트워크 인터페이스를 통해 전달된다. 그러나 이동 애드혹 환경에서 대부분의 노드는 하나의 네트워크 인터페이스를 가지며 패킷 진입 인터페이스와 진출 인터페이스가 같고 노드의 무선 네트워크 인터페이스는 이웃 노드의 네트워크 인터페이스들과 1대다의 관계를 갖는다. 이동 애드혹 환경에서 멀티캐스트 데이터 포워딩시에 이러한 특성을 고려하지 않을 경우 패킷 중복현상과 라우팅 루프 문제 등이 유발될 수 있다. 본 연구에서 제안하고 구현한 멀티캐스트 데이터 포워딩 기법은 리눅스 환경에서 넷플터[1]와 중복을 방지하기 위한 별도의 테이블을 사용하여 트리 기반 멀티캐스트 라우팅 프로토콜에 의해 결정된 경로를 이용한 효율적인 멀티캐스트 데이터 포워딩을 지원한다.

### 1. 서 론

이동 애드혹 네트워킹 기술은 네트워크 인프라가 갖추어지지 않은 환경에서 이동 무선 노드들만으로 구성된 네트워크이다. 휴대가 편리한 장비를 이용하여 원하는 곳에서 언제든지 네트워크로의 접속을 제공해 주는 이동 애드혹 네트워크는 많은 주목을 받고 있으며, 빠르게 성장하고 있다.

본 연구는 이동 애드혹 환경에서 멀티캐스트 패킷을 전송하기 위한 포워딩에 관한 것으로, 애드혹 멀티캐스트 라우팅 프로토콜에 의해 생성된 라우팅 테이블을 이용하여 멀티캐스트 패킷을 포워딩하고자 할 때 사용한다.

멀티캐스트 패킷 포워딩시에, 유선 환경에서는 멀티캐스트 트리상의 이웃 노드에게만 패킷을 전송하지만 무선 환경에서는 각 노드들이 일대일 인터페이스로 연결되어 있지 않으므로 다른 멀티캐스트 데이터 포워딩 방식이 필요하다. 무선 환경에서는 전파가 일정한 범위 내의 영역으로 전달되어 정당한 수신자가 아닌 모든 이웃한 노드들에게 패킷이 전달되며, 이 과정에서 자신에게 패킷을 보내준 노드에게도 전송하는 중복 현상이 발생하여 라우팅 루프를 생성할 수 있는 단점이 존재한다. 하지만, 멀티캐스팅의 측면에서는 한 번의 전송으로 여러 노드들이 동시에 수신 가능하므로 부족한 무선 대역폭 자원을 절약하면서 멀티캐스팅을 할 수 있는 장점이 있다. 본 연구에서는 이러한 무선의 특성을 고려하여 이동 애드혹 환경에서의 단순하고 효율적인 멀티캐스트 데이터 포워딩 기법을 제안하고 구현한다.

### 2. 고려사항

이 절에서는 이동 애드혹 환경에서 각 노드들이 멀티캐스트 데이터 포워딩을 처리할 때 고려해야 할 사항들을 유선 환경, 유니캐스트 데이터 포워딩과 비교하여 제시한다.

### 2.1 트리 기반 애드혹 멀티캐스트 데이터 포워딩

유선 환경에서의 멀티캐스트는 대부분 트리에 기반한 방식을 이용하지만, 이동 애드혹 환경에서는 트리 기반 방식뿐만 아니라 메쉬 기반 방식도 많이 제안되고 있다. ODMRP[2]와 같은 방식은 무선 환경의 특성을 적절히 활용하여 명시적으로 멀티캐스트 트리를 생성하지 않고 포워딩 노드를 선정하여 멀티캐스트 그룹에 대해 포워딩 하도록 정해진 노드만이 해당 그룹에 대한 포워딩을 담당하므로 멀티캐스트 라우팅 프로토콜에서 멀티캐스트 데이터 포워딩까지 고려한다. 하지만 멀티캐스트 데이터 포워딩을 OS에게 맡기는 MAODV[3]와 같은 트리 기반의 멀티캐스트 라우팅 프로토콜에서 OS가 제공하는 유선 환경에 맞추어진 멀티캐스트 데이터 포워딩 기법을 그대로 사용하면 2.2절에 언급할 문제로 인하여 멀티캐스트 데이터 포워딩을 정상적으로 수행할 수 없으므로 포워딩 방법을 수정해야 한다.

본 연구에서 제안하는 멀티캐스트 데이터 포워딩 기법은 노드들이 멀티캐스트 트리상의 이웃 노드에 대한 정보를 유지하는 트리 기반 멀티캐스트 라우팅 프로토콜에서 사용할 수 있다.

### 2.2 무선의 특성으로 인한 멀티캐스트 패킷 중복 수신 문제

유선에서의 유니캐스트 통신에서는 패킷 포워딩시 원하는 이웃 노드로만 전송할 수 있으나, 무선 환경에서는 이웃한 모든 노드에게 전송하므로 자신에게 패킷을 전송한 노드에게도 동일한 패킷을 전달하게 되며, 이러한 중복 전송을 앞으로 "역전송에 의한 중복"으로 표기한다. 유선 네트워크에서의 멀티캐스팅은 패킷이 들어오는 인터페이스와 나가는 인터페이스가 다르기 때문에 역전송에 의한 중복이 발생하지 않는다. 그러나, 무선 애드혹 네트워크에서의 멀티캐스트 통신은 들어오는 인터페이스와 나가는 인터페이스가 같고, 목적지 주소가 특정 노드의 주소가 아닌 멀티캐스트 그룹 주

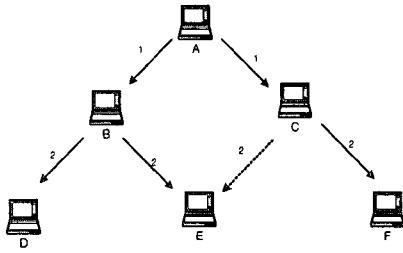


그림 1. 트리를 벗어난 전송에 의한 중복

소이므로 역전송에 의한 중복이 발생하며 패킷의 TTL값이 소진될 때까지 라우팅 루프를 형성한다. 그러므로, 이동 애드혹 네트워크상의 멀티캐스트 데이터 포워딩에서 역전송에 의한 중복을 방지하기 위한 처리를 요구하며, 추가적으로 트리상의 노드에게서만 패킷을 받도록 하는 기법이 요구된다.

그림 1과 같은 트리 기반의 애드혹 멀티캐스팅에서 실선은 멀티캐스트 트리를 나타내며, 노드 A가 멀티캐스트 그룹의 소스이고 노드 D, E, F가 그룹의 멤버라고 할 때, A가 보낸 패킷을 B, C가 수신하여 B는 트리상의 다음 노드인 D, E에게 전송하고 C는 F에게 포워딩해야 하지만, C의 전파범위에 E도 포함된다면 E는 B와 C로부터 패킷을 중복해서 받게 된다. 그림에서 점선으로 표시된 부분은 트리에 속하지 않으므로 E 노드에서는 B로부터 전송된 패킷만을 수신하도록 하는 방법을 제공해야 한다. 이러한 중복 현상은 앞으로 "트리를 벗어난 전송에 의한 중복"으로 표기한다.

### 3. 이동 애드혹 멀티캐스트 데이터 포워딩 구현

유선 환경에서의 포워딩은 라우팅 테이블 검색에 의해 수행되며, 포워딩시에 패킷을 수신한 인터페이스와 패킷을 전송할 인터페이스

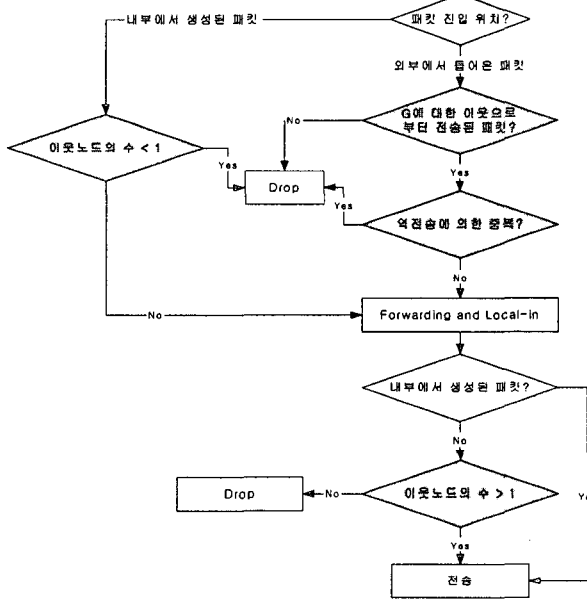


그림 2. 이동 애드혹 멀티캐스트 데이터 포워딩 순서도

가 다르므로 라우팅 경로만 정상적으로 설정되어 있다면 패킷 중복

수신이 발생하지 않는다. 그러나, 무선 환경에서의 멀티캐스트 데이터 포워딩에서는 2.2절에서 언급한 문제로 인하여 멀티캐스트 데이터 포워딩에 새로운 방법을 요구한다. 본 논문에서 구현하는 이동 애드혹 멀티캐스트 데이터 포워딩 (AMF: Ad-hoc Multicast data Forwarding) 방법은 포워딩 노드에서 패킷을 송수신하거나 소스 노드에서 패킷을 보낼 때와 목적지 노드에서 패킷을 받을 때, L2 계층에서 통신하는 두 노드들이 협업으로 멀티캐스트 데이터 포워딩을 수행한다. 송수신 노드에서는 패킷을 필터링하여 포워딩을 수행하며 중복 처리에도 필터링을 이용한다.

이동 애드혹 멀티캐스트 데이터 포워딩 구현은 리눅스 커널[4] 2.4 기반의 운영체제에서 NIST의 AODV[5] 구현에 IPv6 지원[6]과 MAODV 멀티캐스트 라우팅 프로토콜을 추가한 환경을 기반으로 SDR/NTE[7]를 사용하여 동작을 실험하였다. 다른 운영체제와 IPv4 기반의 네트워크에 대한 구현도 본 연구의 내용을 기반으로 수정하는데 무리가 없으며, 멀티캐스트 라우팅 프로토콜도 MAODV가 아닌 트리 기반의 다른 프로토콜을 사용할 수 있다.

리눅스 커널 2.4에는 패킷 필터링을 위해 사용할 수 있는 넷필터 (Netfilter)가 존재하여, 넷필터 훅(Hook)과 연결된 콜백(Callback) 함수를 이용해 대부분의 구현이 이루어진다. 그림 2는 AMF의 작동 방식을 보여주는 순서도이다.

#### 3.1 패킷 중복 처리 방법

AMF는 역전송에 의한 중복과 트리를 벗어난 전송에 의한 중복을 방지한다. ODMRP를 비롯한 기존의 애드혹 멀티캐스트 데이터 포워딩에서는 L3 소스 주소와 소스에서 각 패킷에 부여한 순서번호를 쌍으로 하는 엔트리들을 유지하는 테이블을 이용하여 두 가지 중복을 방지한다. 표 1은 유선에서 사용하는 멀티캐스트 데이터 포워딩 방법에 추가적으로 무선 애드혹 멀티캐스트 환경에서 발생할 수 있는 중복을 방지하기 위해 순서번호를 사용하는 방법에서 필요한 자료구조들이다.

그러나 이 방법은 소스에서 매번 L3 헤더에 순서번호를 부여해야 하고 포워딩 노드들에서는 매 패킷마다 L3 소스 주소와 순서번호 쌍을 유지해야 하므로, 적절한 시간에 불필요한 엔트리를 삭제하는 방법이 없을 경우 메모리 면에서도 낭비를 초래한다. 또한 IPv6 헤더에는 순서번호를 저장하기 위한 필드가 없으므로 확장 헤더를 사

표 1. 순서번호를 사용하는 방법에서 멀티캐스트 데이터 포워딩을 위해 필요한 자료구조들

(a) 멀티캐스트 데이터 포워딩 테이블의 엔트리	
Field Name	Data Type
Originator Address	L3(IPv4 or IPv6) Address
Multicast Group Address	L3(IPv4 or IPv6) Address
Incoming Interface	Integer
Outgoing Interface	Integer

(b) 중복 방지 테이블의 엔트리	
Field Name	Data Type
Originator Address	L3(IPv4 or IPv6) Address
Sequence Number 1	Integer
Sequence Number 2	Integer
...	...
Sequence Number n	Integer

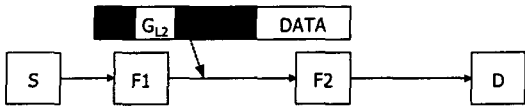


그림 3. 역전송에 의한 중복 방지

용해야 하는 부담이 있으며, 이동 애드혹 네트워크에서의 통신은 "Incoming Interface"와 "Outgoing Interface"가 같으므로 두 개의 필드를 사용하는 것은 낭비이다. 중복 방지 테이블은 역전송에 의한 중복을 방지하며, 트리를 벗어난 중복을 막을 수는 없으나 중복 자체는 방지할 수 있다. 그림 1을 예로 보충 설명하면 노드 E에서 노드 C로부터 온 패킷을 수신한 후, 뒤 늦게 노드 B에서 같은 순서번호를 가진 패킷이 도착한다면 이를 무시한다.

본 연구에서는 역전송에 의한 중복을 방지하기 위해 패킷의 L3 소스와 목적지 그룹에 대한 이전 노드의 L2 소스 주소를 엔트리로 사용한다. 그림 3을 예로 설명하면 D 노드는 그룹 G에 조인한 노드이고 S에서 D로 멀티캐스트 패킷 전송 중 F1에서 F2로 포워딩 되는 패킷을 보여준다. F2에서는 이 패킷을 수신하여 L3 헤더의 (S, G)를 보고 (S, G)에 대한 엔트리가 없으면 F1의 L2 주소 (S, G, F1<sub>L2</sub>)를 저장하고 패킷은 멀티캐스트 데이터 포워딩 테이블에 의해 처리한다. L3 헤더에 (S, G)를 갖는 패킷을 받으면 엔트리에 있는 L2 주소(F1<sub>L2</sub>)와 현재 수신한 패킷의 L2 소스 주소를 비교하여 같으면 수신하고 다르면 폐기한다. 소스 노드에서는 이전 노드가 없기 때문에 L2 소스 주소의 위치에 널(Null) 값을 넣어 중복을 방지하며, 엔트리 비교시 이전 노드의 L2 소스 주소 위치에 널 값이 있으면 소스 노드이므로 L3 헤더에 (S, G)를 갖는 패킷을 폐기한다.

트리를 벗어난 전송에 의한 중복을 처리하기 위해서는 테이블에 L3 목적지 그룹에 대한 트리상의 이웃 노드 L2 주소들을 엔트리로 등록하며, 멀티캐스트 라우팅 테이블을 이용하여 MAODV에서 이 등록을 담당한다. 패킷 수신시에 트리를 벗어난 전송에 의한 중복을 방지하기 위한 테이블(G<sub>L3</sub>, LIST<sub>L2neighbor</sub>)을 참조하여 패킷의 L3 목적지인 G에 대해 이전 노드의 L2 소스 주소가 이웃 노드 L2 주소의 리스트에 존재하는지를 검사하여 있으면 수신하고 없으면 폐기한다.

이상의 내용을 바탕으로 표 2는 본 논문에서 제안한 방법인 AMF에서 멀티캐스트 데이터 포워딩을 위한 필요한 자료구조들을 나타내며, 그림 4는 AMF의 멀티캐스트 데이터 포워딩 방식을 송수신 노드에서의 시간 흐름에 따라 보여준다.

표 2. AMF에서 멀티캐스트 데이터 포워딩을 위해 필요한 자료구조들

(a) 멀티캐스트 데이터 포워딩 테이블의 엔트리 (트리를 벗어난 전송에 의한 중복 방지)

Field Name	Data Type
Multicast Group Address	L3(IPv4 or IPv6) Address
Neighbor L2 Address 1	L2(MAC) Address
Neighbor L2 Address 2	L2(MAC) Address
...	...
Neighbor L2 Address n	L2(MAC) Address

(b) 역전송에 의한 중복 방지 테이블의 엔트리

Field Name	Data Type
Originator Address	L3(IPv4 or IPv6) Address
Multicast Group Address	L3(IPv4 or IPv6) Address
Previous Node of Multicast Flow	L2(MAC) Address

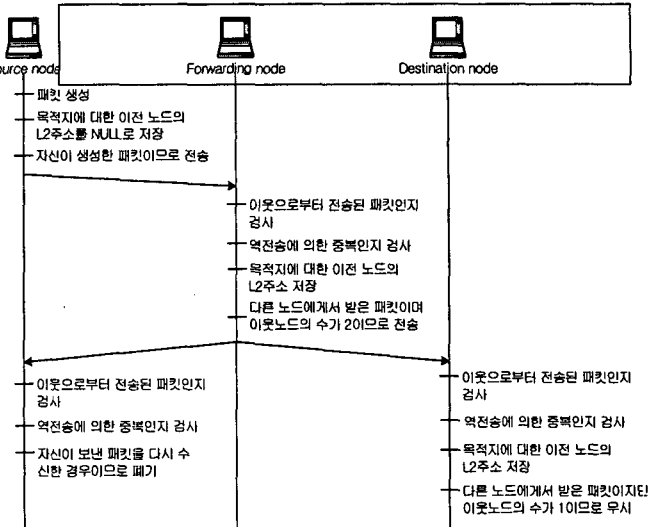


그림 4. 이동 애드혹 멀티캐스트 데이터 포워딩 동작 방식

4. 결론

이동 애드혹 환경에서의 멀티캐스트 데이터 포워딩은 유선 환경에서의 멀티캐스트 데이터 포워딩과는 많은 차이점이 있으므로, 이를 반영하지 않을 경우 많은 패킷 중복 현상으로 인한 라우팅 루프와 비효율적인 메모리 사용 등의 문제를 유발할 수 있다. 본 연구에서 제안한 멀티캐스트 데이터 포워딩 기법은 멀티캐스트 라우팅 프로토콜에서 생성한 라우팅 정보를 이용하여 "트리를 벗어난 전송에 의한 중복"을 방지하는 테이블을 생성하고, 멀티캐스트 트래픽이 이동하는 중에는 "역전송에 의한 중복" 방지 테이블을 생성하여 트리 기반의 멀티캐스트 라우팅 프로토콜의 포워딩을 효율적으로 지원한다.

본 연구에서는 무선 환경에서의 멀티캐스트 데이터 포워딩을 지원하기 위하여 리눅스 커널을 수정 하였으며, MAODV 구현물을 이용하여 이동 애드혹 환경에서 멀티캐스트 트래픽이 안정적으로 포워딩되는지를 확인하였다.

참고문헌

- [1] Netfilter, <http://www.netfilter.org/>
- [2] M.Gerla, S.-J. Lee, and W. Su. "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," Internet draft, draft-ietf-manet-odmrp-02.txt, 2000.
- [3] E. Royer, C. Perkins, "Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing," Internet Draft, draft-ietf-manet-maodv-00.txt, July, 2000.
- [4] Linux kernel, <http://www.kernel.org/>
- [5] C. Perkins, E. Royer, and S. Das, "Ad hoc on demand Distance Vector (AODV) routing," RFC 3561, July, 2003.
- [6] C. Perkins, E. Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing for IP version 6," Internet Draft, draft-ietf-manet-aodv6-01.txt, Nov. 2000.
- [7] UCL Network and Multimedia Research Group, <http://www-wmice.cs.ucl.ac.uk/multimedia/software/>