

논문 2005-42TC-9-4

모바일 Ad-hoc 네트워크를 위한 클러스터 기반 멀티캐스트 라우팅

(A Cluster-Based Multicast Routing for Mobile Ad-hoc Networks)

안 병 구*, 김 도 현**

(Beongku An and Dohyeun Kim)

요 약

본 논문에서 우리는 모바일 ad-hoc 네트워크를 위한 이동성을 사용하는 클러스터 기반의 멀티캐스트 라우팅(CMR: Cluster-based Multicast Routing) 방법을 제안한다. CMR의 주요한 특징은 다음처럼 요약 된다. a) 이동성 기반 클러스터링과 효과적으로 안정성과 확장성을 지원하기 위한 그룹 기반의 계층적 구조, b) 토폴로지의 견고함과 전송의 효율성을 동시에 지원하기 위한 그룹 기반의 메쉬 구조와 전송 트리 개념, c) 사전 결정적 기술들의 낮은 경로 수행 지연과 요구기반 방법의 낮은 오버헤드를 제공하는 개념의 조합. CMR의 성능평가는 모델링과 시뮬레이션을 통하여 이루어진다. 멀티캐스트 라우팅 프로토콜의 효율성 측정은 노드의 이동성, 멀티캐스트 그룹 사이즈, 송신자 노드 수 등의 함수로써 패킷 전송률, 확장성, 제어 오버헤드, 소스와 목적지 노드 사이의 시간 지연 등의 파라미터들을 사용한다.

Abstract

In this paper, we propose a Cluster-based Multicast Routing (CMR) suitable for mobile ad-hoc networks. The main features that our proposed method introduces are the following: a) mobility-based clustering and group based hierarchical structure in order to effectively support stability and scalability, b) group based mesh structure and forwarding tree concepts in order to support the robustness of the mesh topologies which provides limited redundancy and the efficiency of tree forwarding simultaneously, and c) combination of proactive and reactive concepts which provide low route acquisition delay and low overhead. The performance evaluation of the proposed protocol is achieved via modeling and simulation. The corresponding results demonstrate the proposed multicast protocol's efficiency in terms of packet delivery ratio, scalability, control overhead, end-to-end delay, as a function of mobility, multicast group size, and number of senders.

Keyword: Multicast Routing, Ad-hoc Networks

I. 서 론

모바일 ad-hoc 무선 네트워크에서는 무선 모바일 노드들의 모임이 하나의 영역으로 간주하여 이동성을 확

장하는 것을 목적으로 한다. 이 경우 무선 모바일 노드들은 자기 조직화된 ad-hoc을 형성하게 된다^[1]. Ad-hoc 네트워크에서 마이크로프로세서를 가진 모든 장치들은 이동성이 높은 이동성을 가지거나 고정적인 잠재 노드이다. 이것은 이동전화, 자동차, 모바일 정보센터, 인공위성, 데스크탑이나 핸드헬드 컴퓨터 등에 포함되어 있다. Ad-hoc 네트워크는 경로를 찾거나 트래픽 포워딩을 위한 특별한 라우터에 의존할 수 없기 때문에, ad-hoc 네트워크의 모바일 단말 시스템은 트래픽의 경로를 정하거나 링크와 이동성 패턴의 가장 동적인 상태로 네트워크를 적응시키기 위해 협조적으로 동작할 것이다. 모바일 ad-hoc 네트워크는 셀룰러 하부구조를

* 평생회원, 홍익대학교 컴퓨터정보통신공학과
(Dept. of Computer and Communications Eng.,
Hongik University)

** 정회원, 제주대학교 통신컴퓨터공학부
(Dept. of Communications & Computer Eng.,
Cheju National University)

※ 본 연구는 한국과학재단 특정기초연구지원사업
(R01-2005-000-10902-0)의 지원으로 수행되었음.
접수일자: 2004년11월30일, 수정완료일: 2005년9월6일

사용할 수 없거나 신뢰성 없는 응급구조, 탐사임무, 또는 법 집행과 같은 군이나 전술적 상황에서 특히 유용하다. 협동적 모바일 데이터 교환, 비행기내 회의 프로그램, 지능형 네트워크 서비스 등의 상업적 응용에서는 유비쿼터스 기반의 멀티캐스트 구조가 필요하다. 무선 네트워크 기반의 구조를 위한 멀티캐스트 프로토콜은 대체로 유선 네트워크를 확장하는 환경에만 초점이 맞추어져 있다. Ad-hoc 네트워크의 경로들이 주기적으로 바뀌고 경로의 재설정을 위한 고정된 기반구조가 존재하지 않으며, 전력과 적용 범위와 대역폭의 한계가 있기 때문에 기존의 네트워크 기반의 구조를 위해 설계된 멀티캐스트 라우팅 프로토콜은 대체로 다음의 이유들로 인해 모바일 ad-hoc 무선 네트워크에서 적용될 수가 없다.

본 논문에서는 모바일 ad-hoc 네트워크에 적합한 이동성 정보를 사용하는 클러스터 기반의 멀티캐스트 라우팅(CMR :Cluster-based Multicast Routing) 방법을 제시한다. 제안된 구조에서는 이동성 기반의 클러스터링과 이동성 정보(개별 이동성 및 그룹 이동성)와 위치 정보를 이용하여 그룹 이동성 기반의 계층을 형성하고 있다. 본 논문에서는 멀티캐스팅에 참여하는 모바일 ad-hoc 무선 네트워크의 노드들을 관찰을 통해 노드들은 그룹 기반으로 통신하고 동작하는 경향이 강하다는 사실을 알고 이것을 이용하여 제안된 구조를 제시하고 있다. 모바일 ad-hoc 무선 네트워크에서 주요 과제는 라우팅, 멀티캐스팅, 브로드캐스팅^[2] 및 대역폭 예약 등과 같은 기능을 위해 제공하기 위해서는 위치 관리와 자원 할당 기능이 요구된다. 셀룰러 네트워크에서는 제어 기지국(control station)을 통해 모든 제어기지국이 이러한 정보를 서로 알고 있다고 본다. 이러한 바탕으로 셀룰러 네트워크에서의 라우팅과 멀티캐스팅 문제는 간단하게 처리할 수 있다.

본 논문에서 제안된 방식은 노드의 클러스터를 생성함으로써 효과적으로 셀룰러 네트워크 개념을 모바일 ad-hoc 네트워크로 확장시킬 수 있다^[3]. 모바일 ad-hoc 네트워크에서의 통신이 종종 움직임을 통합하는 경향이 있는 팀들 사이에서 일어나는 반면 셀룰러 네트워크에서 통신은 그룹사이의 통신이라기보다 주로 점대점 통신이다. 똑같은 팀의 멤버 사이의 이러한 통합성은 네트워크를 각 그룹이 각자의 이동성 행동을 가지는 몇 개의 그룹으로 나누는 것을 가능하게 한다. 예를 들어 각 멀티캐스트 그룹은 공통적인 특성이 있는 특별한 사용자 그룹(예를 들면 전장터의 탱크 군대, 검색과 구출

작업의 검색 팀, 똑같은 회사에 속해서 움직이는 전문가들, 같은 학급 내의 학생, 기타 등등)과 대응한다. 다른 그룹이나 구조를 위해 다른 이동성 패턴을 정의할 수 있다. 이러한 관찰이 우리에게 이동성 정보를 사용하는 클러스터 기반의 멀티캐스트 라우팅 방법의 연구에 동기를 부여했다. 특히, 이 작업에서 우리는 양 단말과 관계된 노드들의 이동성 정도의 특성을 나타내기 위한 상대적 이동성 개념뿐만 아니라, 지리적 접근과 이동성 패턴과 같은 노드들 사이의 기능적 관계 등 네트워크의 물리적이고 논리적인 분할의 조합을 사용하는 이동성 기반의 클러스터링 Mobility Based Clustering) 접근을 사용하고 있다. 본 논문에서는 멀티캐스트 프로토콜을 지원하기 위한 하부구조로서 이동성 개념을 이용하여 클러스터링 하는 알고리즘을 제시한다. 또한, 제안된 클러스터링 방식을 바탕으로 생성된 노드그룹(클러스터) 사이에서 제안된 프로토콜은 메쉬 구조를 생성하고, 이것은 제한된 중복(중복이 없거나 브로드캐스팅 테크닉이 없는 것에 대해)을 지원하기 위해 그룹 노드 그래프(즉 클러스터헤드(cluster head))의 하위 그래프를 생성한다. 노드 사이의 가능한 경로의 최대 수는 이동성 패턴에 기초를 둔다. 추가로, 가장 짧거나 또는 다른 최상의 경로와 대립되는 것으로 가장 안정된 경로는 멀티캐스트 라우팅에 사용된다.

본 논문은 다음과 같이 구성된다. II장은 전체적인 CMR 기반을 설명한다. 특히, II장 1절은 CMR 개발을 위한 기반구조로서 이동성기반 클러스터링을 설명하고, II장 2절에서는 메쉬 구조의 개념과 멀티캐스트 그룹 구조와 제한된 메쉬 구조를 생성하기 위한 진행사항 설명하며, II장 3절은 데이터 포워딩 전략을 설명한다. III장은 제안된 방법의 성능평가와 결과 값을 포함하고, IV장은 이 논문의 결론을 설명한다.

II. CMR 기반구조

본 논문에서 제안된 이동성 정보를 사용하는 클러스터 기반의 멀티캐스트 라우팅인 CMR(Cluster based Multicast Routing) 방법은 두 개의 기본 구조로 구성되어 있다. 첫 번째 구조는 멀티캐스팅을 지원하는 기반 구조로서 이동성 기반의 계층적 클러스터링 알고리즘을 사용한다. 이것은 개별적, 그룹 이동성에 따른 계층적 클러스터를 구성 한다^[4]. 두 번째 구조에서 메쉬는 라우팅 기반구조의 작용을 하는 클러스터헤드(CHs)그래프의 하위그래프로 생성된다. 첫 번째로 제안된 알고리즘

의 하부구조 (이동성 기반의 계층 클러스터링)를 구성하고 다음으로 생성된 메쉬 토폴로지 상에서 데이터 전송 구조를 선택한다. 하지만 위에서 언급된 것처럼 메쉬는 클러스터헤드에 기반을 두고 있고, 네트워크의 구성원 전체에 두고 있지는 않다.

2.1 이동성 기반 클러스터링: 하부 기반구조

클러스터링은 특징점들이 비슷한 벡터들을 같은 그룹으로 형성하는 방법을 말한다. 유선 백본에 의해 연결된 고정 기지국이 있는 전통적 셀룰러 통신망(single hop)에서는 기지국에서 불과 한 홉 떨어져있는 두개의 모바일 노드 사이의 통신이 고정 기지국과 유선 백본에 의해 이루어진다. 이 경우 클러스터링은 효율적인 주파수 재사용을 위해서 시스템 내의 모든 기지국의 채널 그룹을 선택하고 할당하는 데 사용된다. 다중 홉 모바일 무선 통신망에서, 클러스터링은 중요한 특징(라우팅, 대역폭 할당, 이동성과 토폴로지 관리)의 개발을 위한 간편한 골격을 제공하기 위해 노드들을 그룹(클러스터)으로 모이게 하는 방법이다.

본 논문에서 사용된 클러스터링(즉, Mobility Based Clustering(MBC))은 노드들의 이동성(노드 각각의 이동성 및 그룹 이동성^[6])과 신뢰성 있는 GPS 시스템을 통한 위치정보에 기반을 두고 있다. 클러스터링 알고리즘은 네트워크의 분할을 위해서 물리적이고 논리적인 방법을 함께(combination of both physical and logical) 사용할 뿐만 아니라 유니캐스트 라우팅 프로토콜의 도움 없이 상대 노드들에 대한 이동성의 정도를 알기 위해서 상대적인 이동성의 정보를 사용한다. 이동성기반 계층적 클러스터링 알고리즘인 MBC는 노드들의 이동성 특성에 따라서 변화된 사이즈를 가지는 클러스터이다. 몇몇 클러스터 그룹은 각각의 그룹이 가지는 이동성에 따라서 하나의 그룹으로 합쳐질 수 있다. 주어진 시간동안 상대 노드들에 대한 이동성의 정도를 알기 위한 상대적인 이동성 개념의 도입은 그룹 지향적인 이동성^[6]을 가지는 노드들 사이의 다단계(multi-point) 통신 및 그들의 이동성 좌표를 지원하는 모바일 ad-hoc 네트워크에 주로 사용 되어 왔다.

우리의 목적은 멀티캐스트 서비스 지원을 위한 적응적 클러스터(adaptive clusters)의 구성이므로, 제안된 알고리즘은 모바일 노드 각각의 이동성 및 그룹 이동성 정보를 사용하며, 기반 라우팅 프로토콜에 의존하지 않는다.

MBC의 상세한 설명과 깊은 성능분석은 다음 논문

[6]에 자세히 설명되어 있다. MBC의 성능평가는 이 전략이 매우 안정적인 구조를 형성함을 알 수 있으며, 따라서 이 전략은 클러스터헤드 변화의 필요성을 상당히 많이 줄이므로 모바일 네트워크에서 이동성 관리(mobility management)와 멀티캐스터 기능들(multicast functions)의 실현을 위한 이상적인 전달 수단방법(vehicle)을 제공한다. 특히, 다음 논문 [15,16]에서 보여준 것처럼 MBC 알고리즘은 랜덤 이동성인 경우 lowest-ID^[7] 와 highest connectivity^[8] 클러스터링 알고리즘보다 약 50% 정도 더 안정성이 높다. 특히, 그룹 이동성을 가진 환경에서는 MBC 알고리즘이 lowest-ID 알고리즘과 highest-connectivity 알고리즘보다 약 10 배 이상 성능이 뛰어난을 알 수 있다. 그림 1은 제안된 클러스터링의 기본적인 개념을 보여준다.

그림 2는 멀티캐스트와 이동성 관리 서비스를 지원하기 위한 계층적 클러스터링 구조를 보여준다^[12].

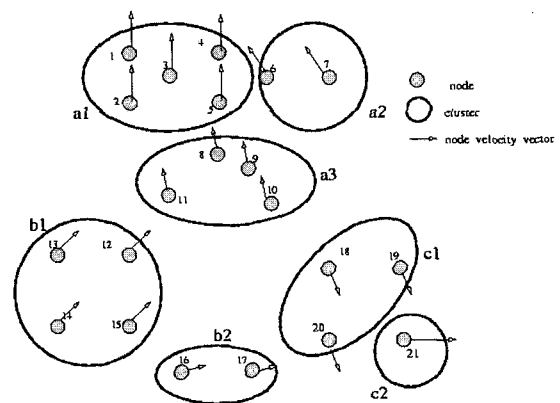


그림 1. 이동성 기반 클러스터링의 기본적인 개념
Fig. 1. Basic concept of the mobility-based clustering.

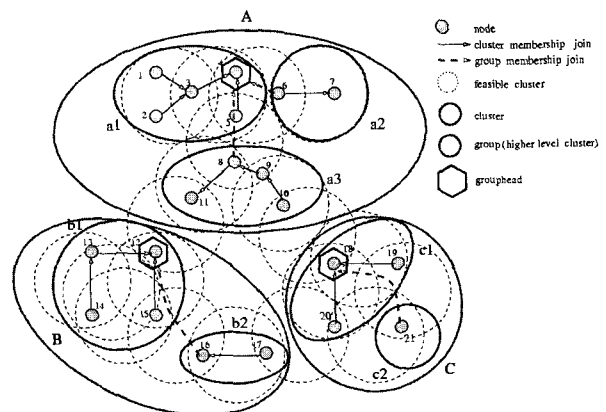


그림 2. 멀티캐스트를 지원하기 위한 이동성 기반 계층적 클러스터링 구조
Fig. 2. Hierarchical clustering structure for supporting multicast.

2.2 메쉬 구조

트리기반 멀티캐스트 라우팅 구조^[6]는 그 단순성으로 인해 컴퓨터 네트워크, 인터넷, 무선 다중 홉(multi-hop) 네트워크에서 광범위하게 사용되어왔다. 하지만 하부 토폴로지가 바뀔 때, 트리기반의 방법을 사용하는 라우터는 강제적으로 작동을 멈추고 트리를 재구성하는 동안 추가적 조정 트래픽이 발생한다. 메쉬 토폴로지(mesh topology)가 가지고 있는 고유의 중복(redundancy)은 이 문제를 극복하도록 할 수 있다. 하지만 동적인 모바일 무선 네트워크의 모든 멀티캐스트 구성원 사이에서 메쉬 구조를 만드는 것은 높은 오버헤드와 제어 트래픽을 생성할 수 있기 때문에 특히 대규모 네트워크에서는 실행 불가능하다. CMR에서 우리는 앞 절에서 설명된 하위의 이동성기반 클러스터링에 의해 생성된 클러스터헤드 상에서만 제한된 메쉬 구조를 제안한다. 그림 3은 클러스터헤드(CH)에 기반한 메쉬 구조를 보여준다. 여기서 메쉬 구조는 CHs만의 하부 그래프(subgraph)이고 전체적인 통신망의 하부 그래프가 아니라는 것에 주의를 기울여야만 한다. 그래서, 단지 CHs 만이 메쉬 구조(즉 한정된 메쉬 구조)의 구성원이 될 수 있다. CMR을 위한 하부구조가 매우 튼튼하기 때문에 파손된 모든 링크에 대한 멀티캐스트 경로의 재계산은 필요하지 않다.

$G_m = (V_m, E_m)$ 을 비 방향성(undirected)의 메쉬 그래프로 나타내고, V_m 은 CHs 집합이고 E_m 은 그 사이의 링크 집합이다.

V_m 내의 s 를 메쉬 내의 멀티캐스트 소스의 클러스터헤드로 나타내고, V_m 내의 r 을 메쉬 내에 대응되는 리시버로 나타내자. 주어진 CH들의 집합에서 (s (source), r (receiver))에 대해서 $M_{s,r} = (V_{s,r}, L_{s,r})$ 는 소스 s 와 리시버 r 사이에 존재하는 메쉬를 나타낸다. 여기서 $V_{s,r}$ 은 $M_{s,r}$ 에 있는 클러스터헤드의 집합을 나타내며, $L_{s,r}$ 은 $M_{s,r}$ 에 있는 링크들의 집합을 나타낸다. 우리의 프로토콜의 주요한 목적 중의 하나는 효과적인 비용으로 튼튼한 메쉬 구조를 만드는 것이다. 이것을 얻기 위해 Boundarycast와 Boundary-Crosscast^[9]가 사용되고 간단한 loop-free 방법이 소개된다. 다음 섹션에서 상세히 이 기술을 묘사한다.

2.2.1 멀티캐스트 그룹 구성과 유지

다음에서 우리는 제안된 멀티캐스트 그룹 구성과 유지

과정의 단계들을 소개한다. 그림 3은 이 절차에 관련된 주요한 단계의 도식적인 설명을 보여준다.

단계 1: 멀티캐스트 그룹의 멤버인 소스 노드가 멀티캐스트 패킷을 가지고 있을 때 소스노드는 패킷을 소스노드의 CH에게 보낸다. 제안된 이동성 기반의 클러스터링에서, 각 노드는 자신의 CH에 다음 노드 ID(node-id)를 가지고 있고, CH는 클러스터 내에 사전에 라우팅 정보를 유지한다. 멀티캐스트 멤버노드가 아닌(non-multicast) 구성원 노드가 멀티캐스트 그룹의 멤버가 되고 싶을 때, 그 노드는 자신의 CH에게 멀티캐스트 멤버십을 부탁한다. 만약 그것의 CH가 멀티캐스트 그룹의 멤버가 아니라면, CH는 이웃 CH들에게 2단계-5단계에서 설명된 것처럼 요청한다.

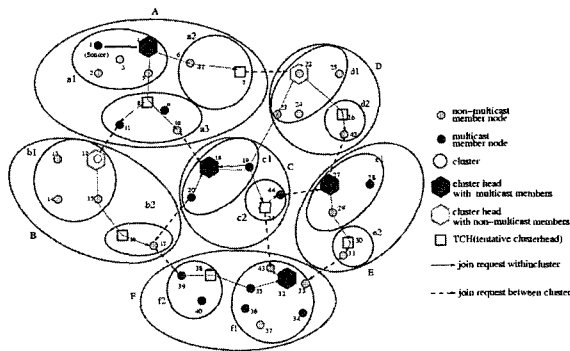
단계 2: CH는 JOIN REQUEST를 생성하고 그것을 Boundarycast와 BoundaryCrosscast 과정^[17]을 사용하여 이웃 CH들에게 알린다. JOIN REQUEST는 노드가 참여하길 원하는 멀티캐스트 그룹의 주소와 소스노드의 ID와 소스 CH의 ID로 구성된다. Boundarycast는 클러스터 내에서 JOIN REQUEST의 전송을 용이하게 하는데 이용된다. Boundarycast에서 각 CH는 전송을 제한하기 위해서 단지 CH의 경계 노드에게만 JOIN REQUEST를 보낸다. 경계 노드의 정보는 제안된 하부의 클러스터링 구조로부터 얻을 수 있고 각 노드에서 사전에 유지된다. 경계 노드가 CH로부터 JOIN REQUEST를 받을 때, 경계 노드는 Boundary Crosscast 작동을 실행한다. BoundaryCrosscast는 클러스터 사이의 라우팅에 사용된다. BoundaryCrosscast에서 각 경계 노드는 단지 그것의 이웃 클러스터 내의 경계 노드(한정된 검색)에게 JOIN REQUEST를 보낸다. 클러스터 간의 라우팅은 BoundaryCrosscast(한정된 경로검색 요청과 응답 방법)의 실행에 의해 경로를 '설정하는 요구기반(on-demand) 프로토콜을 사용하여 수행된다.

단계 3: 멀티캐스트 그룹의 멤버가 아닌 어떤 CH가 동일하지 않은(non-duplicate) JOIN REQUEST를 받으면, 그 CH는 상향 스트림(upstream)(즉, 소스 CH 방향) CH ID를 저장하고 그 메시지를 요청을 받은 곳을 제외한 모든 동일한 이웃 CH들에게 Boundarycast와 Boundary-Crosscast 기술들을 사용하여 메시지를 전송한다.

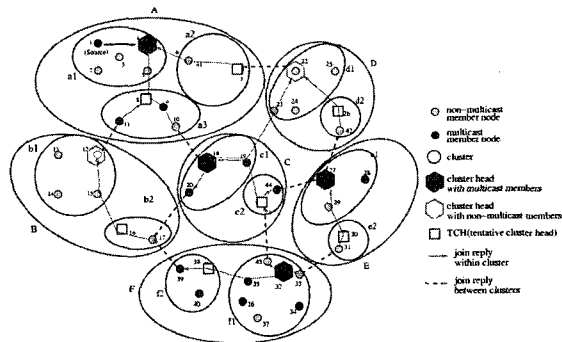
단계 4: 기존의 멀티캐스트 클러스터헤드(CH)멤버가

JOIN REQUEST 메시지를 받으면, 소스 CH에게 JOIN REPLY로 응답하고, CH는 자신의 멀티캐스트 멤버 테이블의 소스 목록을 생성하거나 갱신한다. 이 목록은 멀티캐스트 주소, 소스노드ID, 소스 CH의 ID, 그리고 마지막 JOIN REQUEST 를 받은 시간을 저장한다. 그리고 CH 는 단지 CH에게 처음 받았던 요청 메시지(동일하지 않은)를 이웃한 CHs에게 전송하고 같은 과정이 반복된다. 만약 소스 CH로부터 주어진 시기 내에 JOIN REQUEST 받지 않으면 그 목록은 멀티캐스트 멤버 테이블에서 제거된다.

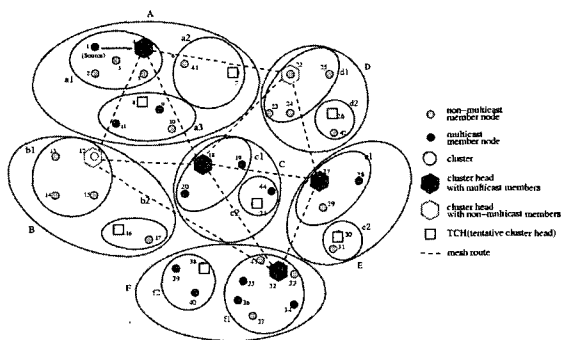
단계 5: CH들 사이의 어떤 노드(CH가 아닌)가 JOIN



(a)멀티캐스트 JOIN REQUEST



(b)멀티캐스트 JOIN REPLY



(c)메쉬구조

그림 3. 멀티캐스트 그룹 구성 및 유지
Fig. 3. Multicast group construction and maintenance.

REPLY를 받으면 자신의 ID가 다음 노드의 ID와 일치 여부를 확인한다. 동일한 경우 노드는 자신이 소스를 향한 경로상에 있다는 것을 알게 된다. 그리고 노드는 업스트림(upstream) (즉, 소스 CH 방향) 노드에게 JOIN REPLY 메시지를 전송한다. JOIN REPLY 메시지는 궁극적으로 멀티캐스트 소스 CH로 전달 된다.

단계 6: 어떤 멀티캐스트 소스가 멀티캐스트 그룹을 떠나고자 할 때는 소스 CH에게 메시지를 보내고 소스 CH는 JOIN REQUEST 메시지 전송을 멈춘다. 어떤 수신노드가 더 이상 특별한 멀티캐스트 그룹으로부터 메시지를 받고 싶지 않을 때는 그 노드는 그것의 CH에게 더 이상 멀티캐스트 그룹을 위한 멀티캐스트의 요청을 하지 않는다. 그 CH는 멀티캐스트 멤버가 지정된 시간 간격 내에 갱신되지 않으면 자신의 멀티캐스트 멤버 테이블에서 대응되는 엔트리를 제거한다.

2.3 데이터 전송 전략

CMR이 멀티캐스트 라우팅을 위해 메쉬(mesh) 기반을 생성한다고 해도, 멀티캐스트 데이터 전송은 소스기반의 트리구조에서 동작한다. 트리구조는 주어진 메쉬 구조상에서 만들어지고 메쉬의 안정성에 의존해 동적으로 생성될 수 있다. 데이터 전송 계획의 주요한 목적은 불필요한 경로(route)를 줄여서 대역폭 및 전송 과부하를 절약하기 위한 것이다. 우리의 전송전략에서, 메쉬 상에서 선택된 트리 경로는 이동성기반의 하부구조의 도움으로 일정한 지속시간(T)동안 안정할 수 있다.

$G_t = (V_t, E_t)$ 를 주어진 메쉬 구조상에서의 방향성(directed) 멀티캐스트 트리구조로 나타내자. 여기서 V_t 는 CH들의 집합이고 E_t 는 CHs 사이의 링크의 집합이다. 트리구조는 주어진 메쉬 구조의 하위그래프(i.e., $V_t \subseteq V_m$ and $E_t \subseteq E_m$)이고, V_t 에 존재하는 $s \in V_t$ 또한 트리내의 멀티캐스트 소스를 나타낸다. 트리 기반 상에서의 멀티캐스트 데이터 전송은 중복되는 전송을 절약하여 대역폭의 효율적인 사용을 이끈다. 다음에서 우리는 제안된 데이터 전송 전략의 주요한 단계를 설명한다. 그림4 는 이 프로세스에 관련된 주요한 단계의 도식적인 표시를 제공한다.

단계 1: 소스 CH는 메쉬 상에서 이웃 CH들에게 멀티캐스트 데이터를 발송한다.

단계 2: 어떤 노드 CH_i 는 첫째 데이터 패킷을 받으면 CH_i 는 그것이 메쉬상에서 다른 하향 스트림(upstream) 클러스터헤드 (소스 CH 방향)를 가지고 있는지 확인한다. 만약 그 노드가 다른 업스트림 CH를 가지고 있다면, 자신이 이미 데이터패킷을 받았다는 것을 알리기 위해 메쉬 경로 상의 다른 CH들 (모든 CH_k)에게 FORWARD REJECT 메시지를 전송한다. 데이터 패킷을 받는 어떠한 CH_i 는 일정 시간(T_r)후에 이 작동을 반복할 수 있는데, 여기서 T_r 은 네트워크의 안정성(이동성)에 의존한다. 비록 이 진행절차에 기초를 둔 현재 버전의 CMR 에서 우리는 트리구조 기반의 전송 토폴로

지를 생성하지만, 이것은 불안정한 환경 하에서 여기에서 설명 되어진 절차를 적용적으로 적용하는 것이 가능하다. 이런 의미에서 우리는 고정관념의 트리구조에서 벗어나 데이터 전송률을 높이기 위해서 더 많은 경로들을 가지는 구조들을 생각한다. 예를 들어, 만약 네트워크가 매우 불안정하다면, FORWARD REJECT 메시지가 얼마간의(전부가 아닌)다음 전송할 CH 들에게 전송되어 질 수 있다.

단계 3: 어떤 노드 CH_k 가 FORWARD REJECT 메시지를 받으면, CH_k 는 어떤 시간(T_f)동안 FORWARD REJECT 메시지를 보낸 CH_i 에게 더 이상 데이터 패킷을 보내지 않는다. 여기서 T_f 는 통신망의 안정성에 영향을 받는다.

단계 4: 자원(대역폭)은 선정된 전송 경로 상에서 사전에 설정(예약) 될 수 있다.

CMR 프로토콜의 현재 버전에서 우리는 모든 특정한 자원(bandwidth) 예약 계획(즉 단계 4)을 실행하지 않았다. 그 대신 우리는 선정된 트리에 관한 충분한 자원이 있다고 가정한다. CMR 프로토콜에서 수행될 수 있는 효과적인 자원 예약 계획의 설계와 개발을 위한 작업은 현재 연구 진행 중에 있다.

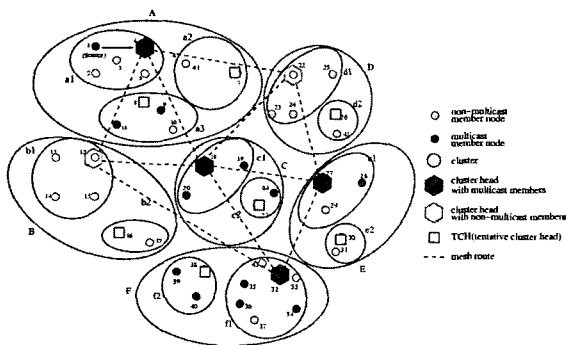
2.4 데이터 구조

다음에서 우리는 CMR 알고리즘의 실행에 필요한 데이터 구조에 대해 요약한다.

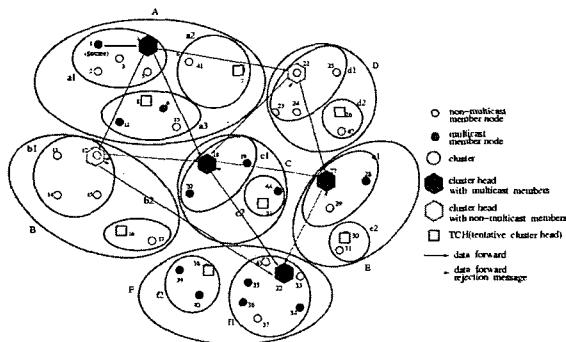
메쉬 테이블(mesh table): 메쉬 테이블은 수신자 노드로 향해서 다음 CH의 정보를 가지고 있다. 메쉬 테이블은 현재 CH ID와 다음 CH ID, 경로 정보(node-id 홉)로 구성된다. CHs 메쉬 테이블은 CH들 사이의 요청에 의해 생성되고 CH 멀티캐스트 멤버들에게 데이터를 전송하는데 사용된다.

클러스터 테이블(cluster table): CH는 자신의 클러스터 멤버쉽과 자신의 클러스터 내의 모든 노드들의 라우팅 정보를 유지한다. 이 정보는 테이블에서 정기적으로 갱신되고 유지된다.

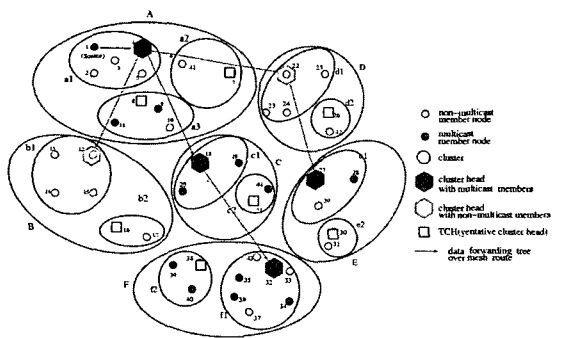
노드-CH 테이블(node cluster head table): 클러스터



(a) 메쉬구조



(b) 포워딩 트리 구성



(c) 포워딩 트리 위의 메쉬 구조

그림 4. 데이터 전송 전략

Fig. 4. Data forwarding strategy.

내의 각 노드는 그것의 CH로 가기위한 다음 노드 정보를 유지한다. 이 정보는 테이블에 정기적으로 갱신(update) 되고 유지된다.

멀티캐스트 멤버 테이블(multicast member table): 각 CH는 자신의 클러스터 안에서 멀티캐스트 그룹 멤버 정보를 가진다. 이 테이블은 마지막 JOIN REQUEST를 받을 때 멀티캐스트 주소, 소스 노드 ID, CH ID, 그리고 마지막 JOIN REQUEST를 받는 시간을 저장한다.

III. 성능 평가

이 절에서 우리는 제안된 멀티캐스트 라우팅 프로토콜의 성능을 평가한다. III장 1절에서 우리는 멀티캐스트 라우팅 프로토콜의 수행을 평가하는 데 사용되는 모델링과 시뮬레이션 구조를 설명하고, III장 2절에서는 성능평가 파라미터를 설명하며, III장 3절에서 결과를 이끌어내고 대응되는 특정 결과뿐만 아니라 이 연구를 두루 거쳐 사용된 성능 수치 값들을 보여준다. 우리의 멀티캐스트 라우팅 프로토콜 성능 평가에서 강조하는 점으로 효율성과 효과에 있다. 프로토콜 효과 외에 효율을 이해하는 것은 우리에게 상대적인 강점과 약점, 다양한 상황에 대한 응용성에 대해 연구하고 토론할 능력을 보여 준다. 또한 우리는 제안된 프로토콜을 ODMRP^[10] 프로토콜과 비교한다. 여기서 ODMRP는 메쉬 구조와 요구기반(on-demand) 및 포워딩 그룹 개념을 사용한다.

3.1 시뮬레이션 시나리오

제안된 멀티캐스트 라우팅 프로토콜의 성능평가는 Optimized Network Engineering Tool (OPNET)을 사용하는 모델링과 시뮬레이션을 통해 실행된다. 모바일 ad-hoc 네트워크는 1000 미터 x 1000 미터의 직사각형 지역 안에서 임의적으로 분포되어 있는 50개의 모바일 노드로 구성된다. 각 노드는 무한 버퍼와 store-and-forward 큐잉(queueing) 기지국(station)을 가지고, 신뢰할 수 있는 위치 선정 시스템(즉, GPS)의 도움으로 위치를 인식하게 된다고 가정한다. 그리고 모바일 노드는 $Z=300$ m의 일정한 무선 범위를 가진다고 가정한다. 따라서 우리는 다음과 같이 시뮬레이션 시나리오를 해석할 수 있다. 어떤 패킷은 노드의 전송 범위(발신기로부터 Z 반경)내에서 오류 없이 전송될 수 있다. 이 연구의 주요한 목적은 멀티캐스트 라우팅 전략의 효율과 효과

를 평가하는 것이고, 따라서 단순화된 모델이 데이터 링크 계층을 위해 가정된 반면 우리의 연구는 주로 네트워크 계층의 세부항목에 초점을 맞춘다. 이 연구를 통해서 우리는 노드의 이동성에 기인하여 한 노드가 다른 노드의 모바일 범위 안팎으로 움직임에 따라 링크가 실패하거나 재구성 된다고 추측한다.

모바일 노드는 네트워크를 두루 거쳐 움직이고 있다고 가정한다. 이 연구에서는 두 개의 다른 이동성 시나리오가 고려된다. 첫 번째는(이하 랜덤 이동성으로 나타낸다) 각 노드의 움직임의 속도와 방향은 각각 속도 범위 $[0, V_{max} \text{ km/h}]$ 과 방향범위 $[0, 2\pi]$ 로 균일하게 분배된다. 각 노드에서 속도와 방향이 매 Δ_t 초(시뮬레이션에서 $\Delta_t=5\text{seconds}$)에 독립적으로 갱신된다. 두 번째 이동성 시나리오에서(이하 그룹 이동성으로 나타낸다) 그룹기반의 이동성 패턴이 설계된다. 특히 노드들은 몇 개의 그룹으로 분류되는데, 여기서 같은 그룹 내의 노드들은 비슷한 이동성 특성을 가진다. 모든 모바일에서 각 모바일노드의 속도는 $[0, V_{max} \text{ km/h}]$ 의 범위 내에서 독립적으로 선택된다. 모바일 노드들의 움직임 방향은 시뮬레이션의 처음에 각 그룹(서로 다른 그룹은 서로 다른 초기 방향을 가진다.)별로 임의적으로 선택된다. 어떤 한 그룹에 속한 모든 모바일 노드들은 자신의 그룹에 원래 선정된 방향에 관하여 $(-\pi/6, +\pi/6)$ 의 범위 안에서 독립적으로 실제적인 움직임을 선택한다. 그 결과 똑같은 그룹에 속한 모바일 노드들은 그들이 움직이고 있는 방향에서 어떤 유사성을 보인다. 처음에 각 그룹은 5개의 노드로 구성되어 있었지만 시뮬레이션이 진행되어 가면서 그룹 당 노드의 숫자들은 동적으로 변화한다. 다음 절에서 이동성 시나리오들(랜덤 이동성 및 그룹 이동성)은 둘 다 V_{max} 의 3개의 다른 값, 20 km/hr, 50 km/hr 와 80 km/hr,에 대한 즉, 각각 느리고, 보통, 높은 움직임속도를 나타내는 결과를 나타낸다. 만약 모바일 노드가 주어진 네트워크 도달 범위 지역의 경계에 도착한다면 노드는 네트워크 내로 다시 들어간다. 멀티캐스트 구성원들은 임의로 선택된다. 결국 이 연구를 통해서, II장 1절의 이동성의 경계 값 Th_{mob} 은 다음과 같이 선택된다. m_{mob} 을 노드 m 이 받는 모든 이동성 정보의 평균값으로 나타내고, δ_{mob} 을 대응되는 표준편차로 나타내자. 그러면 Th_{mob} 은 다음과 같이 선택된다.

$$Th_{mob} = m_{mob} + k\delta_{mob} \quad (1)$$

여기서 k 는 상수이고 고려 하에 있는 네트워크와 요구되는 클러스터 안정성에 따라 변할 수 있다. 이 시뮬레이션에서 파라미터 k 는 실험에 기초를 두어 $k = 1.5$ 로 선택된다.

3.2 성능 평가 파라미터

우리가 이 논문에서 평가 목적을 위해 사용하는 성능 평가 파라미터 값은 다음과 같다:

패킷 전송 비율(packet delivery ratio): 멀티캐스트 수신자 노드에게 전달되어야 할 데이터 패킷의 숫자에 대한 멀티캐스트 수신자 노드에게 실제적으로 전달된 데이터 패킷의 숫자로 정의된다. 이 비율은 라우팅 전략의 효과를 나타낸다.

제어 오버헤드(control overhead): 경로 생성 프로세스에 관련된 제어 패킷의 평균으로 정의된다.

지연(delay): 멀티캐스트 소스노드와 수신자 노드 사이의 멀티캐스트 경로를 생성하는 평균 지연 시간으로 정의된다.

본 논문에서 성능평가 파라미터들은 주로 두 개의 기준으로 연구되었다. 이동성과 확장성(scalability)이다. 모바일 ad-hoc 네트워크에서 가장 중요한 특징 중의 하나가 이동성이기 때문에 우리는 멀티캐스트 라우팅 프로토콜의 성능을 이동성 속력(speed)의 함수로써 나타낸다. 게다가 각각의 프로토콜의 확장성을 연구하기 위해서 우리는 멀티캐스트 그룹 사이즈를 변경함으로써 프로토콜의 행동을 연구했다. 앞에서 언급했듯이 우리는 또한 같은 네트워크 시나리오 하에서 CMR의 성능을 대응하는 ODMRP^[18]의 성능과 비교한다. ODMRP는 채널 오버헤드를 줄이고 확장성을 향상시키는 요구 기반 라우팅 기술을 사용한다. ODMRP에서 그룹 멤버쉽과 멀티캐스트 경로들은 소스노드의 요청에 의해 설정되고 갱신된다.

3.3 수치 결과

그림 5와 그림 9는 각각 랜덤 이동성과 그룹 이동성 시나리오에서 프로토콜의 패킷 전송률을 이동성 함수로써 나타내는데, 여기서 초당 5개의 패킷 전송률 (packet

transmission ratio)을 사용했다. 이 그림들의 결과에서 보여지 듯 CMR은 임의적이거나 그룹지향의 패턴이 모두 지원되는 매우 동적인 상황에서 아주 잘 동작하는데 이유는 우리의 프로토콜에서 트리는 클러스터헤드 기반으로 생성된 메쉬 토폴로지의 하부구조에 기반을 두어 선택되고 만들어지기 때문이다. 따라서 링크 단절의 영향은 데이터를 적합한 목적지에 전송하는 이 프로토콜의 성능에는 매우 작은 영향을 미친다.

그림 6과 그림 10에서 우리는 랜덤 이동성과 그룹 이동성의 경우에 멀티캐스트 그룹 크기의 함수로써 20km/hr의 이동성 속도를 사용하여 패킷 전송률을 나타냄으로써 이 프로토콜의 확장성 특징을 나타낸다. 이러한 그림은 CMR이 멀티캐스트 멤버의 수에 의해 영향을 받지 않는다는 것을 보여준다. 제안된 프로토콜의 이러한 행동 추세는 플러딩(flooding) 기술과 ODMRP 프로토콜^[11]에 의해 나타난 것과 비슷하다. 멀티캐스트 그룹 사이즈에 따라 다른 행동을 보일 수 있는 다른 메쉬 기반 프로토콜과 달리 이 프로토콜이 멤버쉽 그룹 사이즈에 영향을 받지 않는 이유는 CMR에서의 메쉬 구조가 주로 하부의 이동성 기반의 클러스터링 알고리즘에 주로 의존하고 다른 프로토콜(CAMP^[19])처럼 수신자 노드의 구성원에 의존하지 않기 때문이다. 게다가 우리의 프로토콜이 하부구조를 만들기 위해 그룹 이동성 특성을 사용한다는 사실이 그룹 이동성 시나리오 하에서 CMR의 확장성과 안정성을 향상시키기 위해 기여하기 때문이다.

그림 7과 그림 11은 $V_{max} = 20\text{km/hr}$ 의 속도로 랜덤 이동성과 그룹 이동성 시나리오에서 송신자 노드(sender) 수의 함수로써 멀티캐스트 경로 생성과 유지와 관련된 제어 오버헤드(control overhead)를 나타낸다. 제어 오버헤드는 경로와 클러스터를 생성하고 유지하기 위해 다양한 노드들 사이에서 교환이 필요한 모든 신호(패킷)들을 포함한다. 우리가 그림 7과 그림 11로부터 보는 것처럼 CMR과 관련된 제어 오버헤드는 거기에 대응하는 ODMRP의 제어 오버헤드가 과감하게 증가하는 반면에 송신자 노드의 수가 증가하면서 아주 조금 증가한다. CMR가 하부구조로 클러스터링을 사용하고, 관련된 오버헤드는 주로 클러스터의 구성과 유지 보수 때문이고 송·수신 숫자에 어느 정도는 독립적이기 때문이다. 이런 경우에는 단지 작은 양의 추가적인 제어 신호만이 멀티캐스트 통신의 실제적 지원을 위해 교환되어야 한다. 그 결과 전체적인 오버헤드는 다른 요구기반(reactive) 접근들과 비교해서 소스 노드

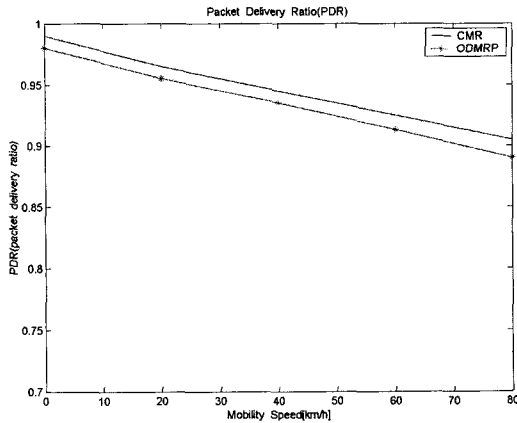


그림 5. 이동성 속력 함수에 대한 패킷 전송 비율 (랜덤 이동성)
 Fig. 5. Packet delivery ratio as function of mobility speed. (random mobility)

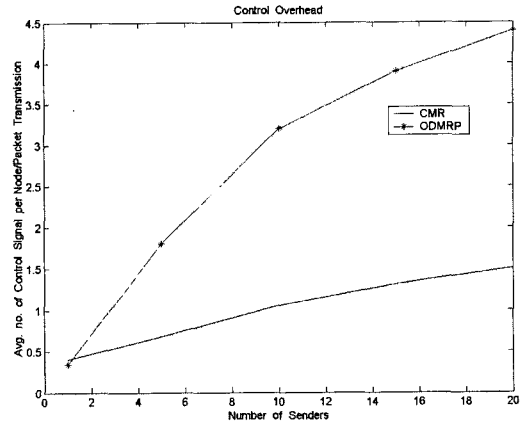


그림 7. 멀티캐스트 소스 크기 함수에 대한 경로 생성을 위한 컨트롤 과부하(랜덤 이동성)
 Fig. 7. Control overhead for route creation as function of number of senders. (random mobility)

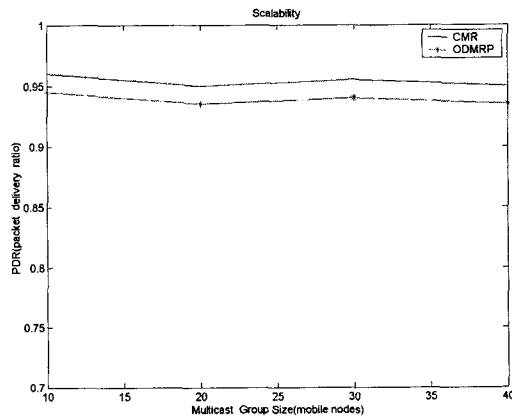


그림 6. 멀티캐스트 그룹 크기 함수에 대한 패킷 전송 비율(랜덤 이동성)
 Fig. 6. Packet delivery ratio as function of multicast group size. (random mobility)

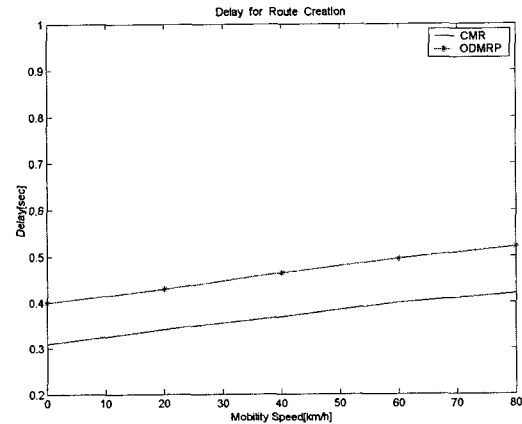


그림 8. 이동성 속력 함수에 대한 경로 생성을 위한 시간 지연(랜덤 이동성)
 Fig. 8. Delay for route creation as function of mobility speed. (random mobility)

(source node)의 수에 적은 의존성을 가진다. 예를 들어, 그림 7과 그림 11에서 관찰되는 것과 같이 ODMRP가 경로 생성에 요구기반(on-demand) 방식을 사용하기 때문에 대응하는 제어 신호의 수가 소스(source)의 수의 증가와 함께 크게 증가한다.

그림 8과 그림 12는 멤버들에게 멀티캐스트를 하기 위한 경로 생성에 걸리는 평균 지연시간을 나타낸다. 지연시간은 경로 생성을 위해 소스와 모든 멀티캐스트 멤버 사이에 걸리는 모든 지연 시간을 포함한다. 우리가 그림 8과 그림 12 으로부터 보는 것처럼 CMR의 지연은 대응되는 ODMRP 지연보다 작다. 앞에서 설명한 것처럼, 중요한 이유는 ODMRP가 단지 요구기반(reactive) 접근을 사용하는데 비해 CMR은 하이브리드(hybrid) 접근을 동시에 사용한다. 게다가 지연은 이동

성 속도가 증가하면서 아주 조금 증가한다. 중요한 이유 중의 하나는 CMR 전략의 통합된 부분인 MBC 방법이 클러스터헤드를 변경할 필요성을 현저히 줄이는 매우 안정된 하부구조를 생성하기 때문이다. 그러나, 하부구조가 안정될지라도 속도가 증가함에 따라 매쉬 구조와 생성된 경로가 망가질 가능성 또한 증가하고, 그리하여 지연 또한 증가할 수 있다.

그림 11과 그림 12에서 보듯이, 그룹 이동성 시나리오에서의 CMR의 성능(오버헤드와 지연의 조절)은 그에 대응되는 랜덤 이동성 시나리오(그림 7, 그림 8)보다 훨씬 우수하다. 그 주된 이유는 CMR의 설계 특징과 원리중 하나가 노드의 이동성 패턴에 의존하는 하부구조를 사용하는 것이고, 그래서 당연한 결과로 그것은 하부구조의 변경이 덜 민감한 그룹기반 동작과 통신의 경우처럼

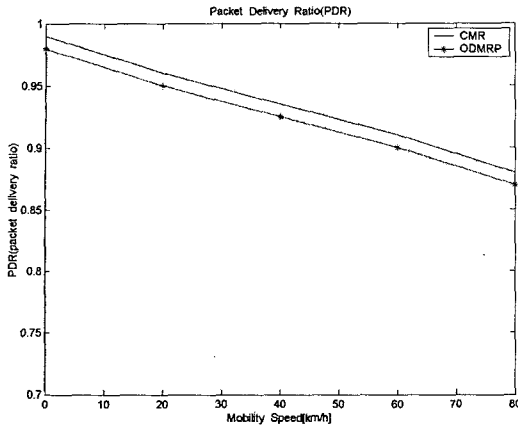


그림 9. 이동성 속력 함수에 대한 패킷 전송 비율 (그룹 이동성)
 Fig. 9. Packet delivery ratio as function of mobility speed. (group mobility)

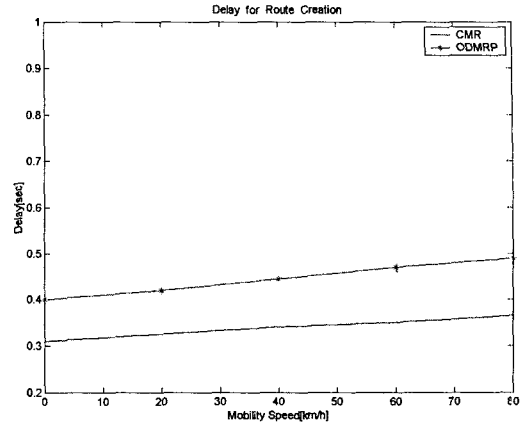


그림 12. 이동성 속력 함수에 대한 경로 생성을 위한 시간 지연(그룹 이동성)
 Fig. 12. Delay for route creation as function of mobility speed. (group mobility)

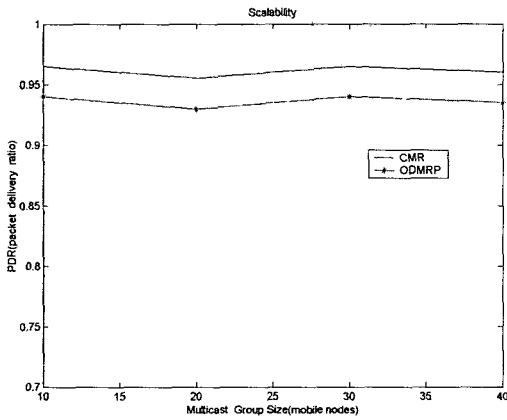


그림 10. 멀티캐스트 그룹 크기 함수에 대한 패킷 전송 비율(그룹 이동성)
 Fig. 10. Packet delivery ratio as function of multicast group size. (group mobility)

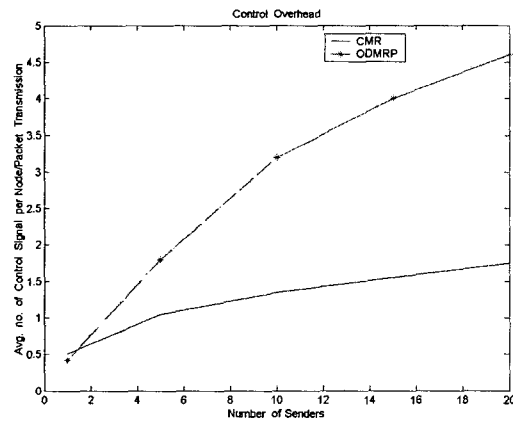


그림 11. 멀티캐스트 소스 크기 함수에 대한 경로 생성을 위한 컨트롤 과부하(그룹 이동성)
 Fig. 11. Control overhead for route creation as function of number of senders. (group mobility)

비슷한 이동성 특징을 나타낸다.

IV. 결론

본 논문에서 우리는 모바일 ad-hoc 네트워크를 위한 이동성을 사용하는 클러스터 기반의 멀티캐스트 라우팅 프로토콜(CMR)을 제안한다. CMR 프로토콜의 일부분으로서 우리는 물리적이고 논리적인 네트워크의 분할의 조합을 사용하는 이동성 기반의 클러스터링 (MBC) 접근을 사용했다. 이 구조는 주로 계층적 클러스터를 형성하고 멀티캐스트 프로토콜을 수행하기 위한 하부 구조를 제공한다. CMR의 성능평가는 노드의 이동성, 멀티캐스트 그룹 사이즈, 송신자 노드 수 등의 함수로써 패킷 전송률, 확장성, 제어 오버헤드, 소스와 목적지 노드 사이의 시간 지연 등의 파라미터들을 사용하여 제안된 멀티캐스트 라우팅 프로토콜의 효율을 측정한다. 성능평가 결과는 제안된 프로토콜이 그룹 지향의 이동성인 경우에서 더욱 우수한 성능을 나타낼 수 있다. 따라서, 모바일 ad-hoc 네트워크에서 멀티캐스트 응용이 그룹별로 움직이는 이동성을 가지는 경향이 매우 많이 존재하기 때문에 프로토콜은 모바일 ad-hoc 네트워크에서 다중 홉(multi-hop) 통신을 지원하는 이상적인 방법을 제공할 수 있음을 알 수 있다.

참고 문헌

[1] Elizabeth M. Royer and Chai-Keong Toh, "A Review of Current Routing Protocols for Ad Hoc

- Mobile Wireless Networks," IEEE Personal Commun., vol.6, issue 2, pp.46-55, April 1999.
- [2] Ivan Stojmenovic, Mahtab Seddigh, and Jovisa Zunic, "Dominating Sets and Neighbor Elimination-Based Broadcasting Algorithms in Wireless Networks," IEEE Transactions on Parallel and Distributed Systems, vol.13, no.1, pp.14-25, January 2002.
- [3] C.R. Lin and M. Gerla, "Adaptive Clustering for Mobile Wireless Networks," IEEE Journal on Selected Areas in Communications, vol.15, no.6, p.p. 1265-1275, September 1997.
- [4] M. Gerla and J.T. Tsai, "Multicluster, mobile, multimedia, radio network," ACM/Baltzer Journal Wireless Networks Journal, vol.1, no.3, p.p. 255-265, 1995.
- [5] Xiaoyan Hong, Mario Gerla, Guangyu Pei and Ching-Chuan Chiang, "A Group Mobility Model for Ad-Hoc Wireless networks," Proc. of ACM/IEEE MSWiM'99, August 1999.
- [6] Beongku An and Symeon Papavassiliou, "A Mobility-Based Clustering Approach to Support Mobility Management and Multicast Routing in Mobile Ad-hoc Wireless Networks," The International Journal of Network Management (IJNM), vol.1, no.6, pp.387-395, December 2001.
- [7] Anthony Ephremides, Jeffrey E. Wieselthier, and Dennis J. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling," Proc. of IEEE 75(1), pp.56-73, 1987.
- [8] Abhay K. Parekh, "Selecting routers in ad-hoc wireless networks," in ITS, 1994.
- [9] Zygmunt J. Hass, "A new routing protocol for the reconfigurable wireless networks," Proc. of ICUPC'97, vol.2, pp. 562-566, 1997.
- [10] Sung-Ju Lee, Mario Gerla, and Ching-Chuan Chiang, "On-demand Multicast Routing Protocol," Proc. of IEEE CNC'99, New Orleans, LA, pp.1298-1302, September 1999,
- [11] J.J. Garcia-Luna-Aceves and E. Madruga, "A Multicast Routing Protocol for Ad-hoc Networks," Proc. of INFOCOM'99, New York, NY, June 1999.

저 자 소개



안 병 구(평생회원)
1988년 경북대학교
전자공학과 (BS)
1996년 (미) Polytechnic
University, Dept. of
Electrical and Computer
Engineering (MS)

2002년 (미)New Jersey Institute of Technology
(NJIT), Dept. of Electrical and Computer
Engineering (Ph.D)

1990년~1994년 포항산업과학기술연구원(RIST),
선임연구원

1998년~2002년 Lecturer, New Jersey Institute
of Technology(NJIT). USA

2003년~현재 홍익대학교 컴퓨터정보통신공학과
조교수

Marquis Who's Who in Science and
Engineering

(세계과학기술 인명사전) 등재

<주관심분야: Wireless Networks, Ad-hoc
Networks, Sensor Networks, Multicast Routing
Protocols, Mobility Management, QoS,
Location-Based Technology>



김 도 현(정회원)
1988년 경북대학교
전자공학과 (BS)
1990년 경북대학교
전자공학과(MS)
2000년 경북대학교
전자공학과(Ph.D)

1990년~1995년 국방과학연구소(ADD) 연구원

1999년~2004년 천안대학교 정보통신공학부
조교수

2004년~현재 제주대학교 통신컴퓨터공학부
조교수

<주관심분야: Mobile Computing, Sensor
Networks, Mobility Management, Ubiquitous
Middleware and Services>