

자율구성 계층구조 애드혹 네트워크를 위한 상호 연동방식의 토폴로지 탐색 및 라우팅 프로토콜

양 서 민* · 이 혁 준**

요 약

자율구성 계층구조 애드혹 네트워크(Self-organizing hierarchical ad hoc network, SOHAN)는 편평구조 애드혹 네트워크의 확장성을 향상시키기 위해 설계된 새로운 형태, 즉, 액세스 포인트, 전달 노드, 이동 노드의 3 계층의 애드혹 노드로 구성된 네트워크 구조이다. 본 논문에서는 SOHAN의 자율구성을 위한 토폴로지 탐색과 라우팅 프로토콜을 소개한다. 또한 높은 전송 용량을 갖는 최적의 클러스터 기반 계층구조 토폴로지를 형성하기 위한 링크 품질 및 MAC 지연 시간 기반의 크로스레이어 설계방식의 경로 척도를 제안한다. 토폴로지 탐색 프로토콜은 2.5 계층에서 MAC 주소를 기반으로 동작하는 라우팅 프로토콜을 위한 기본적인 정보를 제공한다. 이 라우팅 프로토콜은 AODV 프로토콜을 기반으로 하며, 계층구조의 장점을 활용하기 위해 토폴로지 탐색 프로토콜과 상호 연동하도록 설계된다. 시뮬레이션을 통해 전송용량, 종단간 지연시간, 패킷 전달률, 제어 오버헤드 관점에서 SOHAN의 우수한 성능과 확장성을 보인다.

A Joint Topology Discovery and Routing Protocol for Self-Organizing Hierarchical Ad Hoc Networks

Seomin Yang* · Hyukjoon Lee**

ABSTRACT

Self-organizing hierarchical ad hoc network (SOHAN) is a new ad-hoc network architecture designed to improve the scalability properties of conventional "flat" ad hoc networks. This network architecture consists of three tiers of ad-hoc nodes, i.e., access points, forwarding nodes and mobile nodes. This paper presents a topology discovery and routing protocol for the self-organization of SOHAN. We propose a cross-layer path metric based on link quality and MAC delay which plays a key role in producing an optimal cluster-based hierarchical topology with high throughput capacity. The topology discovery protocol provides the basis for routing which takes place in layer 2.5 using MAC addresses. The routing protocol is based on AODV with appropriate modifications to take advantage of the hierarchical topology and interact with the discovery protocol. Simulation results are presented which show the improved performance as well as scalability properties of SOHAN in terms of throughput capacity, end-to-end delay, packet delivery ratio and control overhead.

키워드 : 자율구성 애드혹 네트워크(Self-Organizing Ad Hoc Network), 이동 애드혹 네트워크(Mobile Ad-Hoc Network), 토폴로지 탐색 프로토콜(Topology Discovery Protocol), 무선 다중 홉 라우팅(Wireless Multi-Hop Routing), 확장성(Scalability), 크로스레이어 설계(Cross-Layer Design), 경로 척도(Path Metric)

1. 서 론

무선 다중 홉 네트워크는 지난 몇 년간 많은 관심을 끌고 있는 연구 분야 중 하나로 최근까지는 주로 라우팅 프로토콜을 중심으로 많은 연구가 이루어져 왔다. 대부분의 애드혹 라우팅 프로토콜들은 전통적인 편평구조 애드혹 네트워크에서 동작하도록 설계되었고, 지연시간, 패킷 전달율, 라우팅 오버헤드와 같은 사용자 성능 관점에서 최적화되었다.

그러나, 최근 들어서는 전력 소모나 전송 용량(throughput capacity)과 같은 무선 다중 홉 네트워크의 시스템 용량(system capacity)과 관련된 연구들에 대한 관심이 집중되고 있다. 편평구조 애드혹 네트워크는 근본적으로 확장성이 낮다는 문제를 가지고 있다. 이와 관련된 대표적인 연구로 Gupta와 Kumar는 애드혹 네트워크의 전송용량은 한계를 가지며, 노드수(n)가 증가함에 따라 $o(1/\sqrt{n})$ 만큼 감소한다는 것을 증명하였다[1]. 이에 따라 차세대 무선 네트워크 시스템에서는 무선 네트워크의 커버리지를 확장하고, 애드혹 네트워크의 전송 용량을 높이기 위하여 고정 무선 네트워크와 애드혹 네트워크의 연동을 고려하고 있다. 이는 계층구조에 기반하는 새로운 애드혹 네트워크 구조를 연구하는 동

※ 본 연구는 한국과학재단 목적기초연구(R01 2001 00349) 지원으로 수행되었음.
※ 이 논문은 2003년도 광운대학교 연구년에 의하여 연구되었음.
* 준 회원 : 광운대학교 대학원 컴퓨터공학과
** 정 회원 : 광운대학교 컴퓨터공학과 교수
논문접수 : 2004년 9월 1일, 심사완료 : 2004년 12월 8일

기가 된다.

S. Ganu는 액세스 포인트(Access Point, AP), 전달 노드(Forwarding Node, FN), 이동 노드(Mobile Node, MN)의 3계층(tier)으로 구성된 SOHAN(Self-Organizing Hierarchical Ad-Hoc Wireless Network)이라는 새로운 계층구조 애드혹 네트워크 구조를 제안하였다[2]. SOHAN에서 AP는 고속의 유선 링크에 연결되며, 하이브리드 유무선 다중 홉 네트워크를 구성하기 위하여 FN, MN과도 자율구성을 통해 연결된다. 이 구조는 애드혹 네트워크가 가진 강건성, 커버리지 확장 능력, 전력의 이득을 유지하면서도 시스템 용량을 증가시키도록 설계되었다.

SOHAN은 한 노드가 여러 무선 인터페이스를 가질 수 있는 다중 무선 방식 네트워크(multiple radio-based network)이기 때문에 FN은 두 개의 서로 다른 무선 인터페이스를 가질 수 있다. 여기서 하나는 MN으로 향하는 하향링크(downlink) 트래픽을 위한 것이고, 다른 하나는 FN간 트래픽이나 AP로 향하는 상향링크(uplink) 트래픽을 위한 것이다. 이와 같은 다중 무선 방식 네트워크에서는 네트워크 토폴로지 탐색(topology discovery)을 위하여 각 인터페이스의 서로 다른 채널을 통해 경로 요청(Route Request, RREQ) 메시지를 브로드캐스트 해야 하기 때문에 경로 요청 메시지를 플래딩하는 동작이 매우 비효율적이다. 따라서 이 네트워크에서는 이웃 노드와 통신할 수 있는 인터페이스와 채널을 데이터 트래픽 전송 전에 검색할 수 있으며, 애드혹 네트워크 환경에서 높은 전송 용량을 제공하는 네트워크 토폴로지를 구성할 수 있는 토폴로지 탐색 프로토콜이 필요하다. S. Ganu의 경우에는 네트워크 토폴로지를 결정하기 위한 탐색 척도로써 단순히 수신 신호 강도를 사용하는 MAC 계층 토폴로지 탐색 프로토콜을 제안하고 있다[2]. 그러나 이 연구에서는 탐색된 네트워크 토폴로지가 어떻게 네트워크의 확장성을 향상시킬 수 있는지에 대해 명확히 설명하지 못하고 있다. 또한 여기서는 토폴로지 탐색에 의해 생성된 이웃 노드 테이블을 주기적으로 교환하는 거리벡터(distance vector) 방식의 라우팅 프로토콜을 사용하고 있지만, 이는 MN으로부터 AP까지의 유출(outbound) 트래픽에 대해서만 지원하고 있다는 제약이 있다. S. Zhao는 DSR(Dynamic Source Routing)[3]이나 AODV(Ad-hoc On-demand Distance Vector)[4]와 같은 기존의 애드혹 라우팅 프로토콜들을 SOHAN 구조에 적용하여 이들에 대한 활용가능성을 평가하고, 시뮬레이션을 통해 기존의 편평구조 방식에 비하여 전송 용량 측면에서 성능 향상 결과를 보이고 있다[5]. 그러나, 여기서는 라우팅 프로토콜이 계층구조를 효율적으로 사용하는 방법에 관해서는 언급하고 있지 않다.

이전에 발표된 계층구조 애드혹 라우팅 프로토콜에 관한 연구들로는 편평구조를 가진 애드혹 네트워크상의 노드들을 클러스터링하여 논리적인 계층구조를 형성하는 방식들이 있

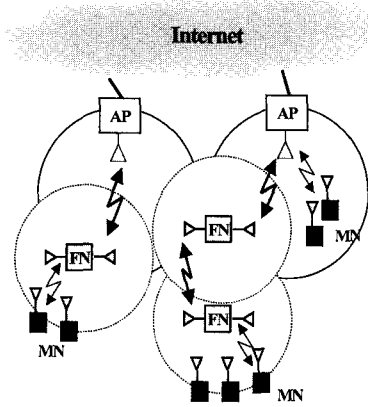
다[6-8]. 이 방식들은 편평구조 애드혹 네트워크에서의 제어 오버헤드를 줄이는 등의 라우팅 효율을 높이기 위하여 논리적인 계층구조를 활용한다. 그러나 이러한 방법들도 편평구조 애드혹 네트워크의 용량 한계를 극복하지는 못한다[1]. 그러나 SOHAN은 AP, FN, MN과 같은 명시적인 3계층(tier) 구조를 가지고 있고, AP들은 인터넷과 연결된 유선 백본망을 통해 서로 연결되기 때문에 기존의 편평구조 애드혹 네트워크와는 달리 물리적인 계층구조를 가지게 된다. 따라서 기존의 계층구조 라우팅 프로토콜은 SOHAN에 적용되기 어려우며, 이에 적합한 새로운 프로토콜의 설계가 요구된다.

본 논문에서는 SOHAN의 확장성을 증대시키고, 높은 전송 용량을 가진 최적의 클러스터 기반 계층구조 토폴로지를 구성하기 위한 토폴로지 탐색 프로토콜과 라우팅 프로토콜을 제안한다. 이 논문은 다음과 같이 구성된다. 2장에서는 자율구성 계층구조 애드혹 네트워크의 구조에 대해서 설명하고, 3장과 4장에서는 토폴로지 탐색 프로토콜과 라우팅 프로토콜에 대해 기술한다. 5장에서는 시뮬레이션 결과를 통해 SOHAN의 향상된 확장성을 보이고, 6장에서는 결론을 맺는다.

2. 계층구조 애드혹 네트워크 구조

SOHAN은 기존의 편평구조 애드혹 네트워크의 확장성을 향상시키기 위해 설계된 새로운 형태의 자율구성 계층구조 애드혹 네트워크 구조이다. 이 네트워크 구조는 (그림 1)과 같이 최하위계층에 위치하는 저전력 사용자 이동 노드(MN), 중간 계층에 위치하는 고전력 전달 노드(FN), 최상위계층에 위치하는 유선 액세스 포인트(AP)의 3계층(tier)으로 구성된다. MN은 단일 무선 인터페이스를 가지며, FN이나 AP와 연결된다. MN은 저사양의 사용자 기기로서 다른 노드를 위해 패킷을 중계하지 않는다. 또한 MN간은 직접 연결되지 않고, 자신과 최적의 링크 품질(quality)을 가진 AP나 FN을 통해 연결된다. 중간 계층(tier)에 위치하는 FN은 다중 홉 라우팅을 사용하여 다른 노드들에게 패킷을 전달하는 역할을 수행하며, 3계층상의 모든 노드와 직접 무선 연결을 가질 수 있기 때문에 스스로 여러 계층(layer)을 이룰 수 있다. 이 노드들은 단일 혹은 두 개의 무선 인터페이스를 가질 수 있으며, 여기서 하나는 FN-MN간 트래픽, 또 하나는 FN간 혹은 FN-AP간 트래픽을 위해 사용된다. 또한 FN은 고정된 형태 혹은 이동 형태로 사용될 수 있다. AP는 무선 인터페이스와 함께 인터넷과 연결할 수 있는 유선 인터페이스를 가지고 있기 때문에 액세스 라우터(Access Router, AR)로써 사용될 수 있다. 따라서 SOHAN과 인터넷 간에 전달되는 패킷들은 AP를 통해 라우팅된다. FN과 MN들은 애드혹 형태로 AP와 직간접적으로 통신할 수 있으며, AP들간은 유

선 링크를 통해 서로 연결되기 때문에 무선 링크에 비해 매우 빠른 전송 속도를 제공할 수 있을 뿐만 아니라 SOHAN과 인터넷 간의 상호연동(interworking)이 간단해진다.



(그림 1) 자율구성 계층구조 애드혹 네트워크 구조

3. 토폴로지 탐색 프로토콜

3.1 탐색 프로토콜

토폴로지 탐색은 노드들이 이웃 노드들과 결합(association)을 통해 논리적인 네트워크 토폴로지를 자율구성하는 과정이다. 우리가 제안하는 탐색 프로토콜은 FN과 MN이 AP를 기준으로 하는 논리적인 클러스터를 이루도록 하며, 이때 사용되는 기준은 클러스터 내의 각 노드와 AP 간의 다중 홉 통신비용이 최소화되도록 하는 것이다. 이런 방식으로 구성된 네트워크 토폴로지는 AP를 경유하는 클러스터 간 통신을 위해 최적화되며, 이를 통해 높은 전송 성능을 가진 AP간 우선 링크를 최대한 활용할 수 있도록 해준다.

토폴로지 탐색 프로토콜은 비콘(beacon)을 사용하는 방식이다. 여기서 비콘은 802.11과 같은 기존의 프로토콜에서 사용하는 비콘을 확장하거나 응용계층 비콘으로부터 만들어질 수 있다. AP나 FN은 한 홉 거리의 이웃 노드들을 인지하기 위하여 비콘을 사용한다. 비콘에는 최적의 네트워크 토폴로지를 결정하기 위해 필요로 하는 다양한 종류의 정보를 포함하고 있다. 시스템 시동 시부터 AP와 FN은 미리 정해진 채널을 통해 주기적으로 비콘을 전송하는 자율형성 과정을 시작한다. 여기서 AP는 언제나 비콘을 전송할 수 있지만, FN은 AP 혹은 다른 FN과 결합된 경우에만 비콘을 전송할 수 있다. 검색(scan) 단계에서 FN과 MN은 비콘을 수신하기 위하여 모든 채널에 대한 검색을 주기적으로 반복한다. 이때 비콘이 수신되면 FN과 MN은 비콘 정보를 이용하여 자신의 이웃 노드 테이블을 갱신하고, 최적의 부모 노드를 선택하여 결합 메시지(association message)를 전송하여 결합관계를 구성한다. 이와 같은 과정을 통해 AP가 루트(root)가 되고, FN이나 MN이 리프(leaf)가 되는 트리 구조

가 위에서부터 아래 방향으로 구성된다. 이 트리 구조 내의 모든 노드들은 루트 AP와 직간접적으로 결합된다. 토폴로지 탐색을 위한 비용 척도는 비콘이 전송되는 과정에서 아래 방향으로 누적되어 사용된다.

비콘 메시지의 구조는 (그림 2)와 같으며, 이 형식은 결합 메시지에도 그대로 사용된다. Msg Type 필드는 토폴로지 탐색 단계에서 사용되는 메시지의 타입을 지정한다. 여기서는 비콘, 결합요청, 결합응답, 결합해지의 4개 타입이 있다. Node Type 필드는 메시지를 전송한 노드의 타입 정보를 담고 있다. 여기서는 AP, FN, MN 중에 하나의 값을 가진다. Node ID 필드는 노드의 식별자로써 노드가 서로 다른 인터페이스 식별자(Interface ID)를 가진 복수의 인터페이스를 가지는 경우에도 노드를 인지하도록 한다. Channel 필드는 메시지를 전송하는데 사용된 무선 채널 번호를 담고 있으며, 이 값은 추후에 해당 이웃 노드와의 통신 채널로 사용된다. Cost to AP 필드는 비콘을 전송하는 노드로부터 AP까지의 라우팅 비용 척도에 대한 예상치를 나타낸다. 이 값은 AP까지의 라우팅 비용 관점에서 최선의 부모 노드를 선택하기 위해 사용되며, 이 값의 계산에 대해서는 다음절에서 자세히 언급한다. Beacon No. 필드는 메시지의 순서열 번호이고, 메시지의 중복을 감지하기 위해 사용된다. BSSID 필드는 BSSID(Basic Service Set ID)로 사용되는 루트 AP의 MAC 주소를 담고 있다. AP IPv6 Addr. 필드는 AP의 IPv6 주소를 담고 있으며, FN과 MN의 IPv6 주소 자동 할당(address autoconfiguration)[9]에 사용된다.

Msg Type	Node Type	Node ID	Channel	Cost to AP	Beacon No.	BSSID	AP IPv6 Addr.
----------	-----------	---------	---------	------------	------------	-------	---------------

(그림 2) SOHAN의 비콘과 결합 메시지 구조

이웃 노드 테이블은 각 노드에서 수신된 비콘과 결합 요청 메시지로부터 얻어진 정보들에 기반하여 구성된다. 여기에는 모든 유출 패킷에 대해 다음 홉(next hop) 노드가 되는 트리구조 상의 부모 노드에 대한 정보를 담고 있으며, 자식 노드에 대한 정보도 포함된다. 따라서 이웃 노드 테이블은 라우팅 프로토콜에서 초기 라우팅 테이블을 구성하기 위한 정보를 제공한다. 이는 각 노드에게 AP로 향하는 유출 트래픽에 대한 다음 홉 라우팅 정보를 제공하기 위하여 부모 노드가 기본 경로(default route)로 선택되게 된다. MN은 전력 소모를 줄이기 위하여 라우팅에 참여하지 않기 때문에 AP와 FN은 MN에 대한 패킷 라우팅을 위해 이웃 노드 테이블 내의 MN 엔트리를 이용한다. 같은 클러스터 내의 그 외 노드들에 대한 라우팅 테이블 엔트리는 우리가 수정한 AODV 라우팅 프로토콜에 의해 요구기반(on-demand) 방식으로 구성된다.

이웃 노드 테이블은 비콘과 결합 요청을 통해 새로운 이

웃 노드가 발견되었을 때 갱신된다. 노드가 이웃 노드로부터 비콘을 수신한 경우에는 비콘에 포함된 정보들이 이웃노드 테이블에 복사되고, 이웃 노드를 통해 도달할 수 있는 AP까지의 라우팅 비용이 계산된다. AP까지의 비용은 비콘에 포함된 이웃 노드로부터 AP까지의 라우팅 비용과 자신과 이웃 노드간의 링크 비용의 합으로 계산된다. 노드가 통신 범위를 벗어나는 경우에도 이웃 노드 테이블은 갱신된다. 엔트리에 존재하는 노드로부터 세 번의 연속된 검색(scan)과정동안 계속 비콘을 받지 못하면 엔트리는 즉각 제거된다. 각 노드는 주기적으로 자신의 이웃 노드들 중에 AP까지의 라우팅 비용이 가장 적은 노드를 후보 부모 노드로 선택한다.

노드는 후보 부모 노드를 선택한 후에 해당 노드에게 결합 요청 메시지를 보낸다. 결합 메시지를 받은 노드는 자신의 이웃 노드 테이블에 자식 노드 엔트리를 생성하고, 메시지로부터 전달 받은 정보를 저장한다. 또한 엔트리의 유효 시간을 나타내는 Lifetime 필드 값을 설정하고, 요청을 보낸 노드에게 결합 응답 메시지를 전송한다. 자식 노드가 결합 응답 메시지를 수신하면 후보 부모 노드를 부모 노드로 설정한다. 만일 자식 노드가 응답을 수신하지 못하면 세 번까지 결합 요청을 반복하여 송신하게 된다. 이때까지도 응답을 수신하지 못하면 자식 노드는 이웃 노드 테이블에서 후보 노드의 엔트리를 제거하고, 다음 후보 노드를 선택한 후에 결합 과정을 반복한다. 또한 네트워크의 단절에 대해 빠르게 대처하기 위하여 이미 결합관계에 있는 자식 노드가 부모 노드로부터 3회 연속 비콘을 수신하지 못하는 경우에도 새로운 결합과정을 시작하도록 한다.

이웃 노드 테이블의 각 엔트리는 (그림 3)과 같이 이웃 노드로부터 수신한 비콘 메시지의 여러 정보들을 포함하고 있다. 이웃 노드 테이블 내의 Node Type, Node ID, Channel, Cost to AP, Beacon No. BSSID, AP IPv6 Addr. 필드의 값들은 수신된 비콘 메시지로부터 복사된다. MAC Addr. 필드는 이웃 노드의 48-bit MAC 주소를 나타낸다. Interface ID 필드는 노드에 포함된 여러 무선 인터페이스들에 대해 수신된 비콘을 구분하고, 각 이웃 노드와 통신할 수 있는 인터페이스를 인지하기 위해 사용된다. Cost to Neighbor 필드는 이웃 노드와의 무선 링크 비용을 나타낸다. 우리는 휴리스틱(heuristic) 접근방식을 사용하여 이 값을 계산하며, 이에 대해서는 다음 절에서 자세히 설명한다. Assoc. 필드는 여기에는 노드와 가지고 있는 결합 관계의 종류를 나타내며, 여기에는 부모 관계, 자식 관계, 결합 없음 관계가 있다. 토폴로지 탐색 절차에 대한 자세한 설명은 (그림 4)에서 기술하고 있다.

MAC Addr.	Node Type	Node ID	BSSID	Life time	Interface ID	Channel	Cost to AP	Cost to Neighbor	Beacon No.	AP IPv6 Addr.	Assoc.
-----------	-----------	---------	-------	-----------	--------------	---------	------------	------------------	------------	---------------	--------

(그림 3) 이웃 노드 테이블 구조

3.2 탐색 척도

토폴로지 탐색 프로토콜의 중요한 목표 중에 하나는 높은 전송 성능을 가진 AP간 유선 백본 링크를 최대한 활용할 수 있도록 AP를 경유하는 클러스터간 통신에 최적화된 논리적인 토폴로지를 구성하는 것이다. 따라서 본 논문에서는 노드와 AP간의 통신 비용을 탐색 척도로써 사용한다. 각 노드에서는 노드로부터 유출(outbound)되거나 노드에게 유입(inbound)되는 데이터 트래픽을 전달할 수 있는 AP와 통신하기 위하여 경유해야 하는 부모 노드를 선택할 때 AP까지의 경로 척도를 기준으로 사용한다. 두 노드 간의 물리적인 링크 품질은 대칭(symmetric)이라고 가정한다. 따라서 AP와 특정 노드간의 유입/유출 트래픽은 모두 같은 경로를 사용한다.

AODV나 DSR과 같은 편평구조 애드혹 네트워크를 위한 전통적인 라우팅 프로토콜들에서 주로 사용되고 있는 최소 홉 수 척도는 높은 전송 성능을 가진 경로를 탐색하는데 많은 문제점을 가지고 있으며, 이에 대한 연구들이 널리 진행되었다[10-14]. 최소 홉 수에 기초하는 라우팅 프로토콜은 평균 홉 간 거리가 가장 긴 경로를 선택하게 되고, 이러한 경우에 고정된 송신 전력을 사용하게 되면 보다 낮은 수신 신호 강도와 높은 손실률로 인해 더 낮은 전송 성능을 나타낼 수도 있다. 따라서 본 논문에서는 경로 척도 계산을 위하여 최소 홉 수를 사용하지 않고, 지연시간과 링크 품질에 기초한 새로운 경로 척도를 정의한다. 이는 전달 지연 시간의 기대치(Expected Delivery Delay, EDD)라 하며, 두 노드간 링크에 대한 링크 EDD(EDD_{link})는 아래와 같이 정의된다.

$$EDD_{link} = c_1 D_q + c_2 \frac{D_c}{r}$$

여기서 c_1 과 c_2 는 상수이고, D_q 는 송신자 무선 인터페이스 내에서 패킷의 평균 큐잉 지연 시간이며, D_c 는 패킷 송신시에 발생하는 평균 MAC 경쟁 지연시간이다. 그리고, r 은 비콘의 전달율이다. D_q 는 각 노드의 상태와 연관된 정보이기 때문에 이 값은 노드에서 내부적으로 계산될 수 있다. 반면에 D_c 와 r 은 두 노드간의 링크와 연관된 정보이다. D_c 의 값은 링크의 다른 쪽에서 발생하는 패킷에 영향을 받기 때문에 실질적으로 비대칭적인 특성을 가진 것으로 생각될 수 있다. 그러나 특정 기간동안 링크의 양단에서 충분히 많은 수의 패킷이 발생한다면 D_c 의 값은 대칭적인 특성을 갖는다. 또한 두 노드간의 물리적인 링크는 대칭적인 채널 특성을 가진다고 가정하면 링크를 구성하는 두 노드 입장에서의 r 값은 동일하다. 따라서 이 두 값은 링크의 어느 방향에서라도 측정될 수 있다. 여기서 D_q 와 D_c 의 값은 인터페이스 드라이버로부터 직접 계산되고 얻어질 수 있지만, r 값은 일정 기간동안 각 이웃 노드로부터 수신된 비콘의 개수를 해

당 시간동안 전송되어야 하는 비콘의 개수로 나누어 계산한다. EDD_{link} 는 패킷 전송시 MAC에서 발생하는 모든 지연 시간을 나타내며, 직접적으로 전송 성능에 영향을 준다. 여기에서 전송과 전파(propagation)시 발생하는 지연시간은 값의 변화가 상대적으로 매우 적기 때문에 포함되지 않는다. 비콘 전달율의 역수는 해당 링크를 통해 패킷을 전송할 때 예상되는 재전송 횟수를 의미하게 된다[14].

경로 EDD(EDD_{path})는 경로상의 모든 링크에 대한 EDD_{link} 의 합으로 아래와 같이 표현된다.

$$EDD_{path} = \sum_{\text{all links } \in \text{ a path}} EDD_{link}$$

```

begin
1. {Initialization}
   Initialize channels with the predefined
   values for each node;
   case NodeType of
   AP:
     Start BeaconTimer( $T_{beacon}$ );
   FN, MN:
     Start AssociationTimer( $T_{association}$ );
   end { case };
2. {Main loop}
   repeat
3. {Sending beacon message}
   case NodeType of
   AP:
     if BeaconTimer is expired then
       BEACON.CostToAP :=  $D_{q,AP}$ ;
       Send BEACON to all neighbors;
       Restart BeaconTimer;
     end { if };
   FN:
     if PARENT entry exists and
       BeaconTimer is expired then
       BEACON.CostToAP := PARENT.CostToAP
       + PARENT.CostToNeigh;
       Send BEACON to all neighbors;
       Restart BeaconTimer;
     end { if };
   end { case };
4. {Receiving beacon message}
   if BEACON is received then
     case NodeType of
     AP:
       Discard the received message;
     FN, MN:
       Modify a neighbor entry by BEACON;
     end { case };
   end { if };
5. {Sending association request message}
   case NodeType of
   FN, MN:
     if AssociationTimer is expired then
       Find BestNeighbor based on
       CostToAP from NTable;
       Send ASSOC_req to BestNeighbor;
       if sending ASSOC_req is failed then
         if retry over three times then
           Delete ASSOC_req.Receiver entry
           from NTable;
           Restart AssociationTimer;
         else

```

```

       Resend ASSOC_req;
     end { if };
   end { if };
   end { case };
6. {Sending association ack message}
   if ASSOC_req is received then
     case NodeType of
     AP:
       Modify a neighbor entry by ASSOC_req;
       Set NTable[ASSOC_req.Sender].Assoc
       to CHILD;
       Send ASSOC_ack to ASSOC_req.Sender;
     FN:
       if PARENT entry exists and
         ASSOC_req.Sender  $\neq$  PARENT then
         Modify a neighbor entry by ASSOC_req;
         Set Table[ASSOC_req.Sender].Assoc
         to CHILD;
         Send ASSOC_ack to ASSOC_req.Sender;
       else then
         Discard the received message;
       end { if };
     MN:
       Discard the received message;
     end { case };
   end { if };
7. {Receiving association ack message}
   if ASSOC_ack is received then
     Modify a neighbor entry by ASSOC_ack;
     Set NTable[ASSOC_ack.Sender].Assoc
     to PARENT;
     Send DISASSOC to old PARENT;
     Restart AssociationTimer;
     if NodeType is equal to FN then
       Restart BeaconTimer;
     end { if };
   end { if };
8. {Receiving dis-association message}
   if DISASSOC is received then
     case NodeType of
     AP, FN:
       NTable[DISASSOC.Sender].Assoc := Null;
     MN:
       Discards the received message;
     end { case };
   end { if };
9. {Update MAC routing table using NTable}
   if NTable is modified then
     Modify the MAC RTable based on NTable;
   end { if };

   until the node shutdown;
end { Topology discovery procedure }

```

(그림 4) 토폴로지 탐색 절차

개념적으로 이 합은 각 링크들이 트리구조에 더해지면서 EDD_{link} 가 누적되어 얻어질 수 있다. 실제로 EDD_{link} 의 계산은 각 노드에서 이루어진다. 예를 들어 노드 X가 노드 Y로부터 비콘을 수신한다는 가정 하에서 먼저 우리는 X와 Y 중에 어느 노드가 계산한 D_c 와 r 을 사용할지를 결정해야 한다. AP의 경우에는 비콘을 수신하지 않기 때문에 r 을 계

산할 수 없다. 따라서 Y 가 r 을 계산할 수 있더라도 Y 로부터 계산된 D_c 값은 사용하지 않는다. 여기서 X 의 D_o 값은 부모 노드의 EDD_{path} 값과는 연관이 없지만 자신의 EDD_{path} 에는 포함된다. 노드 X 는 하향방향으로 비콘을 전송하기 전에 자신이 계산한 r 값과 D_c 값과 함께 노드 Y 로부터 수신한 EDD_{path} 를 사용하여 자신의 EDD_{path} 를 계산한다. AP의 경우에는 D_o 값만을 포함하여 전송한다.

경로 EDD는 노드로부터 AP까지 패킷을 전송하거나 혹은 그 반대의 경우에 발생하는 중단간 지연시간의 예측치이며, 전송 성능과 직접적인 연관 관계를 가지는 수치이다. 여기서 언급하는 재전송은 FEC(Forward Error Correction)가 사용되지 않는다는 가정 하에서 MAC 계층에서 발생하는 것을 나타낸다.

4. 라우팅 프로토콜

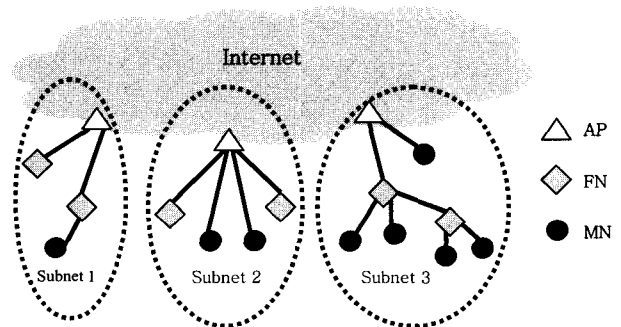
이전 장에서 설명했듯이 SOHAN의 토폴로지 탐색 프로토콜은 자율구성을 통해 전송 용량의 관점에서 최적의 토폴로지를 구성한다. 따라서 패킷 전송시에 토폴로지 탐색 프로토콜에 의해 발견된 토폴로지 정보의 잇점을 활용하기 위해서는 이를 사용하는 라우팅 프로토콜의 설계가 필요하다. 본 논문에서는 라우팅 프로토콜과 토폴로지 탐색 프로토콜이 2.5 계층에서 같이 동작한다. 이러한 방식은 다음과 같은 잇점을 가진다.

- 토폴로지 정보가 라우팅 프로토콜에 의해 쉽게 공유될 수 있다.
- 애드혹 라우팅이 IP 라우팅 계층에 투명하게 동작하기 때문에 애드혹 네트워크와 IP 기반 네트워크의 통합이 자연스럽게 이루어진다.
- 애드혹 네트워크 상에 형성된 클러스터는 단일 LAN 세그먼트로 인지되기 때문에 설치 과정이 단순해진다.

제안하는 라우팅 프로토콜은 AODV에 기반하며, 주소 체계와 라우팅 테이블 관리 방식을 수정하여 설계한다. 여기서는 먼저 이웃 노드 테이블에 기반하여 초기 라우팅 테이블을 구성하고, 이후에 요구기반으로 발견된 새로운 경로 엔트리들을 추가한다. 토폴로지 탐색 프로토콜에 의해 생성된 이웃 노드 테이블은 각 노드에서 AP까지 도달하기 위한 다음 홉 노드(즉 부모 노드)에 대한 정보를 포함하고 있다. 이 정보는 라우팅 테이블의 기본 경로를 초기화하고 주기적으로 갱신하기 위해 사용된다. 그 외의 다른 목적지 노드에 대한 엔트리들은 AODV 경로 탐색 동작에 의해 요구기반 방식으로 생성된다. MN은 라우팅에 참여하지 않기 때문에 다른 노드로 향하는 패킷을 중계하지 않고, 패킷의 소스(source) 혹은 목적지 역할만을 수행한다. 따라서 MN의 라우팅

테이블은 기본 경로 엔트리만을 가지며, 이는 다음 홉 경로로 사용하기 위한 부모 노드(FN이나 AP)에 대한 정보이다.

FN과 MN은 애드혹 라우팅을 위한 식별자로 MAC 주소를 사용하지만, 네트워크 계층에서는 IP 주소를 사용한다. 이 주소는 IPv6 주소 자동 할당과 같은 주소 할당 과정을 통해 얻어진다. MAC 주소에 기반한 라우팅 방식을 사용하기 위해서는 몇 가지 해결해야 할 문제가 있다. 네트워크 계층에서 IP 주소를 사용하여 통신 중단간에 패킷을 전달하기 위해서는 먼저 IP가 하부 계층에서 인식될 수 있는 MAC 주소로 변환되어야 한다. 이러한 주소 번역 동작은 단일 링크 상에 연결된 모든 노드들에 대한 브로드캐스트를 통해 이루어지며, 애드혹 네트워크에서는 이 동작이 네트워크 전체에 대한 플러딩을 통해 실행된다. 편평구조 애드혹 네트워크의 경우에 플러딩으로 인한 무선 자원의 손실은 중요한 문제이다. 이에 따라 본 논문에서 사용하는 플러딩은 하나의 AP와 직간접적으로 결합된 노드들의 집합에 대해서만 한정된다. 제한 영역 브로드캐스트(Limited-Scope Broadcast, LSBC)에서는 이웃 노드 테이블에 저장된 각 노드에 대한 AP와의 결합 정보를 이용한다. 이를 이용하여 LSBC 패킷을 받았을 경우에 각 노드는 자신이 소스 노드와 같은 AP와 결합된 경우에만 패킷을 중계한다. 노드는 플러딩 소스의 BSSID와 자신이 결합을 통해 연결된 AP의 BSSID를 비교하여 플러딩 소스와 같은 클러스터에 속하는지 검사할 수 있다. 소스 MAC 주소와 32비트 식별자는 특정 노드에 의해 이미 플러딩되었던 패킷인지를 구분하기 위해 사용된다. 각 노드는 과거에 처리된 플러딩 패킷의 리스트를 보관하고 있으며, 리스트의 각 엔트리에는 최근에 처리된 플러딩 패킷의 소스 MAC 주소와 식별자가 저장된다.



(그림 5) 트리 형태의 논리적인 토폴로지

제안하는 애드혹 라우팅 프로토콜은 계층구조 라우팅 프로토콜이다. SOHAN에서는 각 AP가 AR로써 설정되기 때문에 AP를 루트로 하는 트리는 (그림 5)와 같이 하나의 IP 서브넷으로 매핑된다. 이러한 방식에서 서브넷 내의 각 노드는 같은 프리픽스(prefix)로 구성된 IP 주소를 가지게 된

다. 만일 목적지 노드가 소스가 속한 서브넷의 외부에 있으면 패킷은 AP로 전달된다. 이후에 AP에서 목적지 네트워크까지는 IP 라우팅을 통해 전달된다. 따라서 애드혹 네트워크 내의 두 서브트리의 경계를 넘어 전달되는 패킷은 유선 링크를 통해 라우팅된다. 이런 방식은 AP들이 연결된 유선 링크가 무선 링크에 비해 매우 넓은 대역폭을 가진다는 가정에 기반한다. 그러나 이 방식에서 FN을 통해 전달되는 가까운 애드혹 경로가 존재하는 경우에는 최적의 경로를 선택하지 못할 수 있다. 제안하는 라우팅 프로토콜에서는 AP를 루트로 하여 구성되는 서브넷 트리와 연관되어 고려되어야 하는 세 가지 형태의 경로가 있다. 각 경우에 따라 패킷이 라우팅되는 과정은 아래와 같다.

4.1 유입(inbound) 경로

서브넷의 외부로부터 시작하여 FN이나 MN으로 전달되는 IP 패킷은 일반적인 IP 라우팅에 의해 AP로 전달된다. 모든 노드들은 AP로부터 단일 홉 거리에 있는 것으로 보이기 때문에 AP의 IP 계층에서는 애드혹 라우팅을 위해 MAC 라우팅 계층에게 패킷을 전달한다. MAC 라우팅 테이블을 검색하기 전에 목적지 IP 주소는 MAC 주소로 변환되어야 한다. 이를 위하여 IPv4에서는 ARP(Address Resolution Protocol) 테이블을 검색하고, IPv6의 경우에는 이웃 캐쉬(neighbor cache)를 검색하여 주소 번역을 수행한다. 만일 엔트리가 발견되지 않으면, LSBC를 통해 주소 번역 메시지를 플래딩한다. 이때 목적지 노드가 서브넷 내에 존재한다면 MAC 주소는 주소 번역 응답에 포함되어 유니캐스트(unicast)로 리턴된다. AP는 MAC 주소를 사용하여 목적지 노드에 대한 라우팅 테이블 엔트리를 검색한다. 목적지 노드로부터 주소 번역 응답 메시지가 소스에게 전달될 때 각 중간 노드들은 역 경로 설정(reverse path setup)을 통해 목적지 노드에 대한 MAC 라우팅 테이블을 추가한다. 이 때문에 주소 번역 동작이 성공한다면 이에 대한 라우팅 테이블 엔트리는 반드시 존재하게 된다. AP가 유출 데이터 패킷을 수신하는 경우에도 새로운 라우팅 테이블 엔트리가 추가된다. 또한 IPv6가 사용된다면 AP에 도착하는 라우터 간청(router solicitation) 메시지에 의해서도 엔트리가 추가된다. 따라서 AP는 서브넷 내의 목적지 노드에 대해 RREQ 메시지를 플래딩하지 않는다. AP는 라우팅 테이블에 기반하여 목적지 노드로 전달되기 위한 다음 홉 노드에게 패킷을 전달한다. 노드의 이동으로 인해 엔트리가 더 이상 유효하지 않게 되면 AP는 RERR 메시지를 수신하게 되고, 이후에 AP는 RREQ 메시지를 플래딩하여 다른 경로를 탐색한다.

4.2 유출(outbound) 경로

유출 패킷들은 토폴로지 탐색 프로토콜에 의해 결정된 경로를 통해 AP에게 전달된다. 소스 노드는 AP로 패킷을 전

달하기 위해서는 자신의 부모 노드에게 패킷을 전달해야 하며, 이를 위하여 기본 경로를 사용한다. 패킷 전송을 위해서는 먼저 목적지 노드가 소스 노드와 같은 서브넷에 존재하는 지를 확인해야 한다. 만일 목적지 프리픽스 검사를 통해 목적지 노드가 서브넷의 외부에 있다고 판단된다면 소스 노드는 기본 경로를 사용하여 AP로 패킷을 전송하며 이는 애드혹 MAC 라우팅을 통해 수행된다.

4.3 내부(internal) 경로

SOHAN의 같은 서브넷 내의 두 노드간에 전달되는 패킷은 순수 애드혹 라우팅에 의해 전송된다. 라우팅 계층에서는 같은 서브넷의 두 노드간에 일반적인 AODV 경로 탐색 동작과 토폴로지 탐색에서 사용되는 경로 척도를 통해 경로를 탐색하고 패킷을 전송한다. 주소 번역과 경로 탐색은 LSBC에 의해 수행된다.

5. 시뮬레이션 결과

5.1 시뮬레이션 모델

우리는 SOHAN을 위한 토폴로지 탐색 프로토콜과 라우팅 프로토콜의 성능을 평가하기 위하여 *ns-2* 시뮬레이터 [15]의 Monarch 확장판을 사용했다. 본 논문에서 제안하는 토폴로지 탐색 프로토콜과 라우팅 프로토콜은 802.11 MAC과 AODV 프로토콜을 수정하여 구현하였다. 또한 SOHAN의 각 계층을 구성하는 노드들이 구현되었다. AP는 100 Mbps의 유선 인터페이스와 802.11b 무선 인터페이스를 가지고, FN은 두 개의 802.11b 인터페이스로 구성되며, MN은 하나의 802.11b 무선 인터페이스를 가진다. 토폴로지 탐색 과정에서 사용되는 비콘 메시지와 이웃 노드 테이블은 MAC 계층에 추가되었다. AP와 FN은 기존의 애드혹 노드와 같이 패킷 중계 기능을 가지고 있으며, MN은 중계 기능이 없다. 또한 MN만이 데이터의 소스와 목적지가 될 수 있도록 설정하였다. LSBC 동작, 이웃 노드 테이블에 기반한 라우팅 테이블 초기화, 데이터 패킷과 주소 번역 응답 메시지에 기반한 역 경로 설정 기능은 AODV 프로토콜에 추가되었다. 이와 같이 수정된 AODV는 *ns-2*에서 LL(Link Layer)의 하부, MAC의 상위 계층에서 동작하도록 설정되었다.

[16]에서 제시하는 결과에 따라 본 논문에서는 FN과 MN 개수를 합한 총 애드혹 노드의 수가 AP 개수의 제곱이 되도록 설정하였다. 그리고, 모든 시나리오에서 MN의 개수는 FN 개수의 약 2배가 되도록 설정하였다. 모든 AP들은 지정된 위치에 고정되며, 전체 시뮬레이션 영역을 균등하게 나누는 사각형 서브 영역의 중심에 위치하도록 했다. FN과 MN은 시뮬레이션 영역 내에 불규칙하게 위치한다. FN과 MN은 이동 시나리오와 정지 시나리오를 가지며, 이동시에는 30초의 휴식 시간과 20m/s의 최대 이동 속도를 갖는 랜덤 중간

지점 모델(random waypoint model)을 사용했다. 트래픽 흐름(flow)의 소스와 목적지는 애드혹 네트워크 내에 존재하는 노드 중에서 무작위로 선택되기 때문에 시뮬레이션에 사용되는 트래픽 흐름은 클러스터내의 노드들간의 트래픽과 서로 다른 클러스터에 속한 노드들간의 트래픽이 혼재한다. 본 논문에서는 SOHAN과 인터넷간의 상호연동(interworking)에 대해서는 다루지 않기 때문에 애드혹 노드와 인터넷 호스트간의 트래픽은 포함되지 않는다. 트래픽 소스가 되는 MN은 512 바이트 크기의 패킷을 고정된 발생율에 따라 생성하며, 패킷 발생율은 초당 5개에서 80개까지 변화시켜 가면서 실험하였다. 트래픽 흐름의 개수는 20개로 고정하였다. <표 1>은 시뮬레이션에 사용된 주요 환경 변수들을 나타낸다.

<표 1> 시뮬레이션 환경 변수

Simulation area	1500m × 1500m				
Number of APs	4	6	8	10	12
Number of FNs	6	12	22	34	48
Number of MNs	10	24	42	66	96
Number of nodes	20	42	72	110	156
Number of pkts/sec generated	5, 10, 20, 30, 40, 50, 60, 70, 80 pkts/sec				
Packet size	512bytes				
Number of communication pairs	20				
Mobility model	Random waypoint model				
Max speed	0m/s, 20m/s				
Pause time	30sec				
Simulation time	1000sec				

5.2 성능 평가

시뮬레이션을 위한 시나리오들은 서로 다른 개수의 AP, FN, MN로 구성되고, 서로 다른 패킷 생성율을 가진다. 각 시나리오에 대하여 10개의 서로 다른 랜덤 노드 위치와 이동 패턴에 대해 반복하여 실험하였다. 시뮬레이션 결과를 평가하기 위하여 사용된 성능 척도는 다음과 같다.

- 시스템 전송용량(system throughput) : 목적지에 도착한 데이터의 총량을 나타내며, 시뮬레이션 시간에 대한 평균값이다.
- 평균 종단간 지연시간(average end-to-end delay) : 패킷이 소스로부터 생성되어 목적지에 도착할 때까지의 총 지연시간의 평균값이다. 이 지연시간에는 경로탐색을 위한 지연시간, 인터페이스 큐에서의 큐잉 지연시간, MAC에서의 재전송이나 경쟁으로 인한 지연시간, 전파와 전송을 위한 지연시간이 포함된다.
- 패킷 전달율(packet delivery ratio) : 소스에서 생성된 패킷 총량에 대한 목적지에 성공적으로 도착한 패킷 총량의 비율이다.

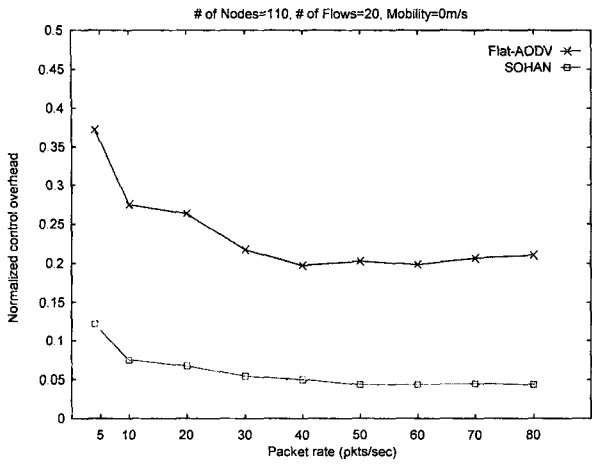
- 정규화 제어 오버헤드(normalized control overhead) : 목적지에 도착한 데이터 패킷의 총량과 이를 위해 사용된 총 제어 패킷의 총량의 비율이다. SOHAN에서의 라우팅은 MAC계층에서 동작하기 때문에 모든 제어 패킷의 크기는 IP 패킷이 아닌 MAC 프레임의 크기를 바이트 단위로 측정하였다. 제어 메시지에는 경로 탐색, 라우터 탐색, 주소 번역 메시지 뿐만 아니라 비콘과 결합 메시지도 포함된다.

편평구조 애드혹 네트워크에서는 IPv6 주소체계와 메시지 구조에 기반하는 AODV 라우팅 프로토콜을 사용하고, SOHAN과 동일한 노드 개수, 노드 위치, 트래픽 패턴을 갖는 시나리오를 사용하여 시뮬레이션한다. 시스템 부하와 확장성의 관점에서 두 네트워크의 성능을 비교한 결과는 다음과 같다.

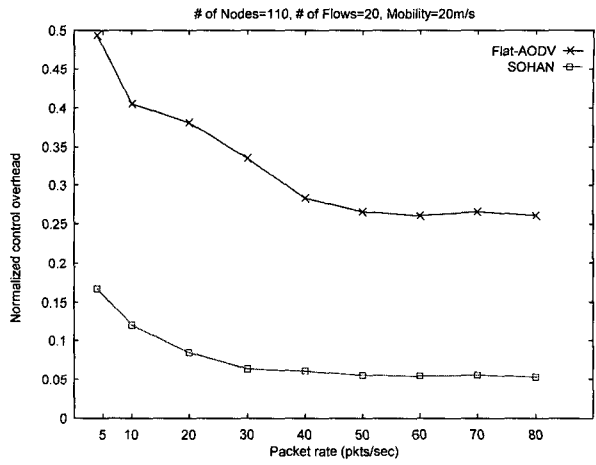
5.2.1 시스템 부하

시스템 부하에 따른 SOHAN의 성능을 평가하기 위하여 모든 시나리오에서 사용되는 총 노드 개수는 110개로 고정했다. SOHAN의 경우에 AP, FN, MN의 개수는 각각 10, 34, 66개다. 편평구조 네트워크의 경우에는 110개의 모든 노드들이 애드혹 노드이다.

(그림 6)에서와 같이 SOHAN의 제어 오버헤드는 편평구조 네트워크에서 AODV를 사용하는 경우에 비해 정지 시나리오에서 66.9%가 감소하고, 이동 시나리오에서 69.1%가 감소하는 결과를 보인다. 이와 같은 오버헤드의 감소는 네트워크 내에 플러딩되는 라우팅 패킷의 양이 줄어들기 때문이다. SOHAN에서는 경로를 탐색할 때 경로 탐색 메시지를 플러딩하지 않고 대부분 비콘과 결합 메시지에 의해 탐색된 토폴로지 정보를 이용하여 라우팅한다. 게다가 제어 패킷을 플러딩하는 경우에도 클러스터 내의 노드에 대해 제한적으로 플러딩하기 때문에 전체적인 오버헤드가 감소하게 된다. (그림 6)(a)와 (그림 6)(b)에서 보듯이 두 네트워크 모두에서 노드가 정지한 경우보다 이동하는 경우에 오버헤드가 증가하는 것을 확인할 수 있다. 이동 시나리오에서는 노드의 이동으로 인해 경로 단절과 경로 재설정이 발생한다. 이때 편평구조 애드혹 네트워크에서는 경로 재설정을 위해 RREQ를 전체 네트워크로 플러딩하기 때문에 정지 시나리오에 비해서 제어 오버헤드가 많이 증가한다. 그러나 SOHAN에서는 부모 노드를 통한 AP까지의 경로를 얻기 위하여 주기적으로 전송되는 비콘을 이용하기 때문에 제어 오버헤드의 양이 이동성에 거의 영향을 받지 않으며, RREQ를 플러딩하여 경로를 탐색해야 하는 경우에도 LSBC를 이용하므로 제어 오버헤드의 증가량이 상대적으로 적음을 알 수 있다. SOHAN에서의 이러한 제어 오버헤드 감소는 다른 성능 척도에도 많은 영향을 끼치게 된다.

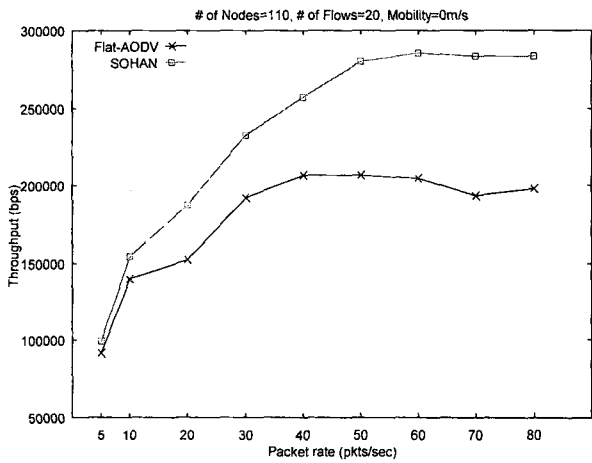


(a) 정지 시나리오

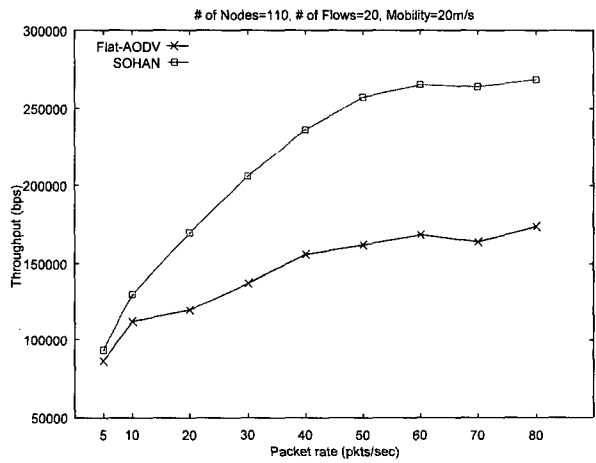


(b) 이동 시나리오

(그림 6) 트래픽 부하에 따른 정규화 제어 오버헤드

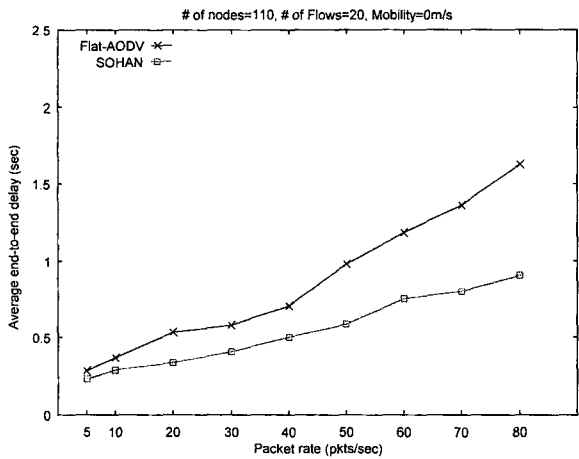


(a) 정지 시나리오

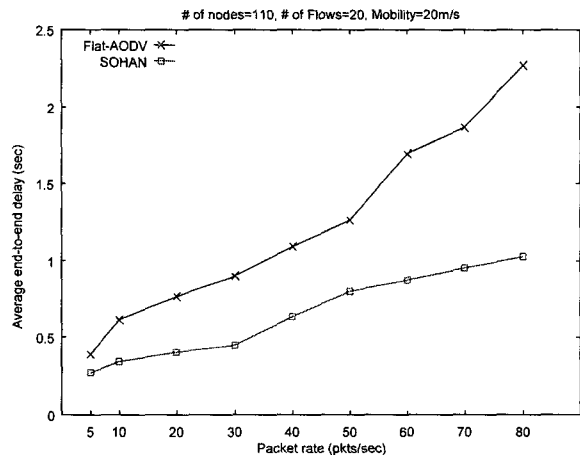


(b) 이동 시나리오

(그림 7) 트래픽 부하에 따른 단위시간당 전송량

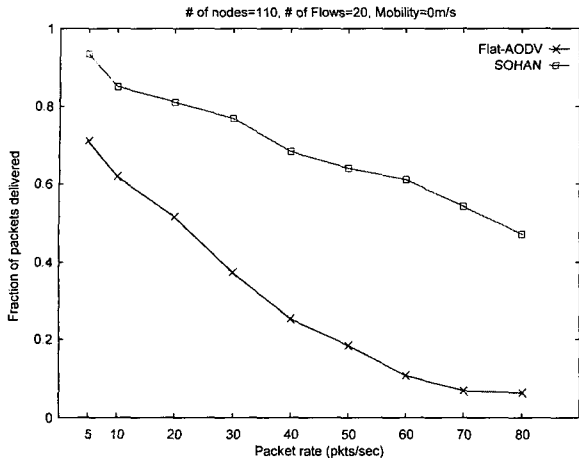


(a) 정지 시나리오

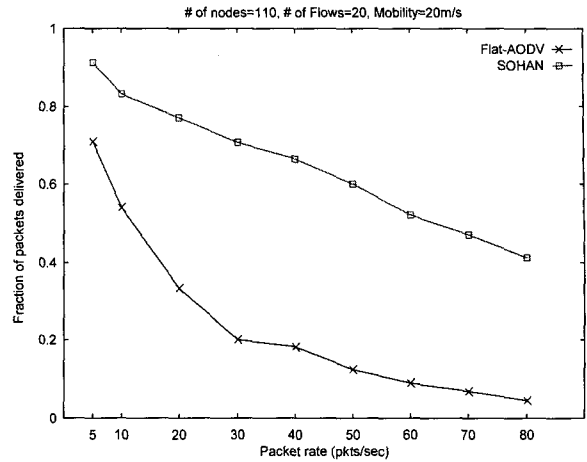


(b) 이동 시나리오

(그림 8) 트래픽 부하에 따른 중단간 지연시간



(a) 정지 시나리오



(b) 이동 시나리오

(그림 9) 트래픽 부하에 따른 패킷 전달율

(그림 7)은 시스템 부하의 증가에 따른 전송성능의 변화를 나타낸다. 정지 시나리오의 경우에 편평구조 네트워크는 패킷 발생율이 약 30pkts/sec 이상일 때부터 포화상태가 나타나지만, SOHAN의 경우에는 약 50pkts/sec 이상일 때부터 포화상태가 나타난다. 따라서 최대 시스템 용량은 편평구조의 경우에는 약 200Kbps 정도이고, SOHAN의 경우에는 약 280Kbps 정도임을 알 수 있다. 이동 시나리오의 경우에도 비슷한 경향을 보이고 있다. 이때 각 네트워크에서 사용된 전송 경로의 길이를 살펴보면, 편평구조의 경우에는 3.50 홉을 가지고, SOHAN의 경우에는 4.12 홉을 가진다. 이는 제한하는 라우팅 척도가 최소 홉 수 경로 대신에 높은 전송용량을 가진 경로를 선택하고, 이로 인해 SOHAN의 전송성능이 향상됨을 알 수 있다.

(그림 8)은 두 네트워크에 대한 종단간 지연시간의 변화를 나타내며, SOHAN의 종단간 지연시간이 편평구조 네트워크에 비해 완만한 증가를 보임을 알 수 있다. 이는 토폴로지 탐색 프로토콜에 의해 구성된 토폴로지가 MAC 지연시간이 적고 재전송이 적게 발생하는 링크들로 구성되어 있기 때문이다. (그림 9)는 두 네트워크에서의 패킷 전달율의 변화를 나타내며, 위에서 설명한 것과 같은 이유로 SOHAN의 패킷 전달율이 우수함을 알 수 있다.

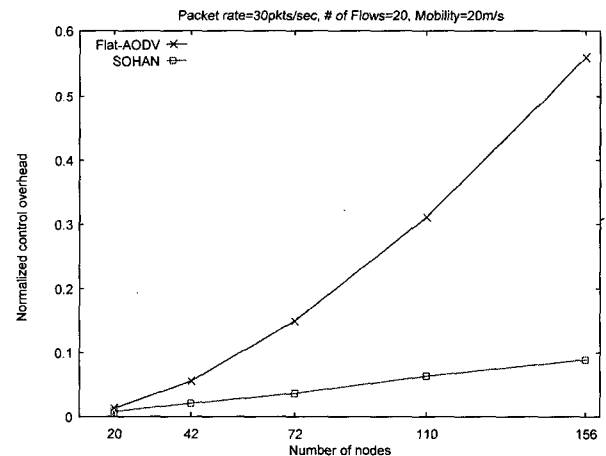
5.2.2 확장성

SOHAN의 확장성을 확인하기 위하여 이 시뮬레이션에서는 30pkts/sec의 발생율을 갖는 트래픽 흐름 20개를 사용하고, 노드 개수의 증가에 따른 성능을 측정하였다. (그림 10)에 나타난 결과를 보면 두 네트워크는 모두 선형적인 증가를 보이지만, SOHAN은 편평구조 네트워크에 비하여 노드 개수의 증가에 따른 오버헤드 증가 정도가 보다 완만함을 알 수 있다.

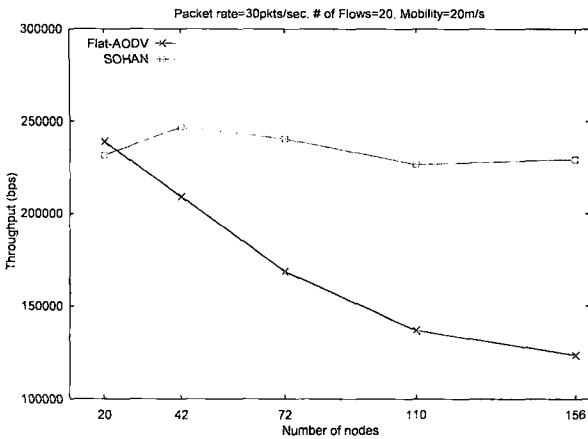
(그림 11)과 (그림 12)는 노드 개수에 따른 전송용량과

패킷 전달율의 변화를 나타낸다. 두 성능 그래프를 통해 SOHAN이 편평구조 네트워크에 비해 더 우수한 확장성을 가짐을 알 수 있다. 그러나 여기서 노드 개수가 20개인 경우에는 SOHAN의 전송성능과 패킷 전달율이 편평구조 네트워크 보다 오히려 낮게 나타남을 알 수 있다. 이 경우에 SOHAN에서 패킷 전달 기능을 가진 노드의 총 수는 AP와 FN을 합하여 10개 밖에 되지 않기 때문에 전체 시뮬레이션 영역을 덮지 못하게 되어 일부 MN에서는 AP나 FN과의 연결이 단절될 수 있다. 이 때문에 모든 노드가 패킷 전달 기능을 가진 편평구조 네트워크에 비하여 패킷 전달율과 전송률이 낮게 된다. 이런 성능 저하 현상을 막기 위해서는 주어진 서비스 영역에 대해 적당한 수준의 전송용량을 유지하기 위해 필요로 하는 최소한 AP와 FN 개수에 대한 향후 연구가 필요하다.

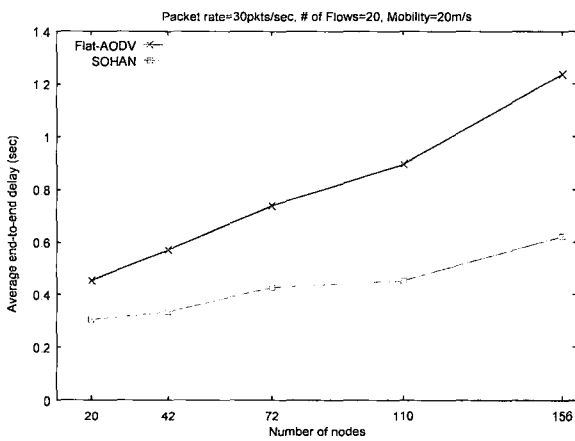
마지막으로 (그림 13)은 노드 개수 증가에 따른 지연시간의 변화를 나타내며, 이를 통해 편평구조 네트워크에 비하여 SOHAN이 우수한 확장성을 가짐을 알 수 있다.



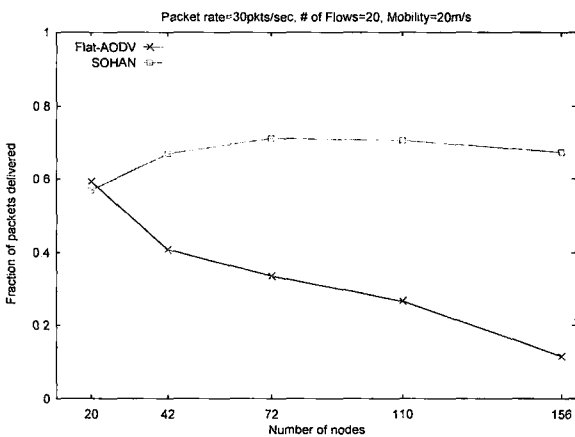
(그림 10) 노드 개수에 따른 정규화 제어 오버헤드



(그림 11) 노드 개수에 따른 단위시간당 전송량



(그림 12) 노드 개수에 따른 종단간 지연시간



(그림 13) 노드 개수에 따른 패킷 전달율

6. 결론

본 논문에서는 자율구성 애드혹 네트워크를 위한 MAC 토폴로지 탐색 프로토콜과 라우팅 프로토콜을 소개하였다. 또한 높은 전송 용량을 갖는 최적의 클러스터 기반 계층구조 토폴로지를 형성하기 위한 링크 품질 및 MAC 지연 시

간 기반의 크로스레이어 설계 방식의 경로 척도를 제안했다. 이를 통해 라우팅 프로토콜과 토폴로지 탐색 프로토콜이 밀접하게 결합되어 동작하게 되면 애드혹 네트워크의 성능 향상에 도움이 될 수 있음을 보였다. SOHAN은 향상된 처리 용량과 높은 확장성을 바탕으로 차세대 무선 네트워크 시스템을 위한 대규모 애드혹 네트워크 구축과 유비쿼터스 센서 네트워크 구축에 기여할 것으로 기대된다. 본 논문에서는 성능 척도 중 하나인 전력 소모에 대해서는 고려하지 않았으나 실제로 소모 전력은 배터리로 동작하는 이동 노드에게 심각한 제약이 될 수 있기 때문에 이것이 반영된 비용 계산이 필요하다. 향후에 홉 수나 전력 소모와 같은 여러 가지 비용 척도들과의 조합을 통한 향상된 경로 척도에 대한 연구가 좀더 필요하며, 다른 클러스터에 존재하는 두 노드간의 편평 라우팅 지원 문제와 이동 IPv4/v6와의 상호 연동에 대한 연구가 더 필요하다.

참고 문헌

- [1] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Transactions on Information Theory*, Vol.IT-46, No.2, pp.388-404, March, 2000.
- [2] S. Ganu, L. Raju, B. Anepu, I. Seskar and D. Raychaudhuri, "Architecture and Prototyping of an 802.11-based Self-Organizing Hierarchical Ad-Hoc Wireless Network (SOHAN)," submitted to *MobiHoc 2004*.
- [3] D. Johnson and D. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," draft-ietf-manet-dsr-09.txt, IETF Internet draft, 15, Apr., 2003.
- [4] C. Perkins, E. Belding Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," IETF RFC 3561, July, 2003.
- [5] S. Zhao, K. Tepe, I. Seskar and D. Raychaudhuri, "Routing Protocols for Self Organizing Hierarchical Ad Hoc Wireless Networks," in *Proc. of the IEEE Sarnoff Symposium*, Trenton, NJ, March, 2003.
- [6] E. M. Belding-Royer, "Hierarchical Routing in Ad hoc Mobile Networks," *Wireless Communication & Mobile Computing*, Vol.2, No.5, pp.515-532, 2002.
- [7] Z. Haas, M. Pearlman and P. Samar, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks," draft-ietf-manet-zone-zrp-04.txt, IETF Internet Draft, July, 2002.
- [8] M. Gerla, X. Hong and G. Pei, "Landmark Routing for Large Ad Hoc Wireless Networks," *Proceedings of IEEE Global Communications Conference (GLOBECOM 2000)*, San Francisco, CA, Nov., 2000.

[9] S. Thomson, Bellcore and T. Narten, "IPv6 Stateless Address Autoconfiguration," IETF RFC 2462, Dec., 1998.

[10] R. J. Punnoose, P. V. Nitkin, J. Broch and D. D. Stancil, "Optimizing wireless network protocols using real-time predictive propagation modeling," in Radio and Wireless Conference (RAWCON), pp.39-44, Aug., 1999.

[11] M. Yarvis, W. S. Conner, L. Krishnamurthy, J. Chhabra, B. Elliott and A. Mainwaring, "Real-world experiences with an interactive ad hoc sensor network," in Proc. of the Int. Workshop on Ad Hoc Networking, pp.143-151, Aug., 2002.

[12] W. H. Yuen, H. N. Lee and T. D. Anderson, "A Simple and Effective Cross Layer Networking System for Mobile Ad Hoc Networks," in 13th IEEE int. symposium on personal indoor and mobile radio communications (PIMRC 2002), Vol.4, pp.1952-1956, Sep., 2002.

[13] D. S. J. De Couto, D. Aguayo, B. A. Chambers and R. Morris, "Performance of Multihop Wireless Networks : Shortest Path is Not Enough," ACM SIGCOMM Computer Communication Review, Vol.43, pp.43-48, Jan., 2003.

[14] D. S. J. De Couto, D. Aguayo, J. Bicket and R. Morris, "A High-Throughput Path Metric for Multi-hop Wireless Routing," in Proc. of the Ninth Annual International Conference on Mobile Computing and Networking, MOBICOM 2003, pp.134-146, Oct., 2003.

[15] K. Fall and K. Varadhan, Eds., "The ns Manual," The

VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC. [Online]. Available WWW : http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf.

[16] B. Liu, Z. Liu and D. Towsley, "On the Capacity of Hybrid Wireless Networks," in Proc. of the IEEE Infocom 2003, Vol.2, pp.1543-1552, Mar., 2003.



양 서 민

e-mail : uniload@kw.ac.kr

1997년 광운대학교 컴퓨터공학과(학사)

1999년 광운대학교 대학원 컴퓨터공학과 (공학석사)

2004년 광운대학교 대학원 컴퓨터공학과 (공학박사)

관심분야 : 무선 네트워크, 이동 인터넷, 이동 애드혹 네트워크 등



이 혁 준

e-mail : hlee@daisy.kw.ac.kr

1987년 University of Michigan, Computer Science(학사)

1989년 Syracuse University, Computer Science(석사)

1993년 Syracuse University, Computer Science(박사)

1994년~1996년 삼성전자(주) 멀티미디어 연구소 선임연구원

1996년~현재 광운대학교 컴퓨터공학과 부교수

관심분야 : 무선네트워크, 이동 인터넷, 이동 애드혹 네트워크 등