이학석사 학위논문

Ad-hoc 무선망에서 AODV와 DSR 라우팅 프로토콜에 대한 연구

2002년 2월

부 경 대 학 교 대 학 원

전 자 계 산 학 과

고 영 웅

이학석사 학위논문

Ad-hoc 무선망에서 AODV와 DSR 라우팅 프로토콜에 대한 연구

지도교수 박 승 섭

이 논문을 이학석사 학위논문으로 제출함

2002년 2월

부 경 대 학 교 대 학 원

전 자 계 산 학 과

고 영 웅

고영웅의 이학석사 학위논문을 인준함

2001년 12월 26일

주 심 이학박사 윤 성 대 (인)

위 원 공학박사 조 우 현 (인)

위 원 공학박사 박 승 섭 (인)

<차 례>

<그림 차례>

A Study on AODV and DSR Routing Protocols in Ad-hoc Mobile Networks

Young-Woong Ko

Dept. of Computer Science Graduate School of
Pukyong National University

Abstract

Ad-hoc networks consist of a set of mobile hosts that communicate using wireless links, without the use of other communicate support facilities (such as base stations, etc.). The topology of an Ad-hoc network changes due to the movement of mobile host, which may lead to sudden packet loss. Recently, the large amount of research has focused on the routing protocols needed in such an environment, however the researches of routing protocol performance analysis were not exhausted.

Accordingly, we simulated Ad-hoc mobile network that using AODV and DSR routing protocols. We use Ad-hoc mobile network topology size and node speed as simulation estimation-factor to analysis routing protocol performance. As the result of the simulation, we know that AODV is more sensitive than DSR about node speed, number of the node.

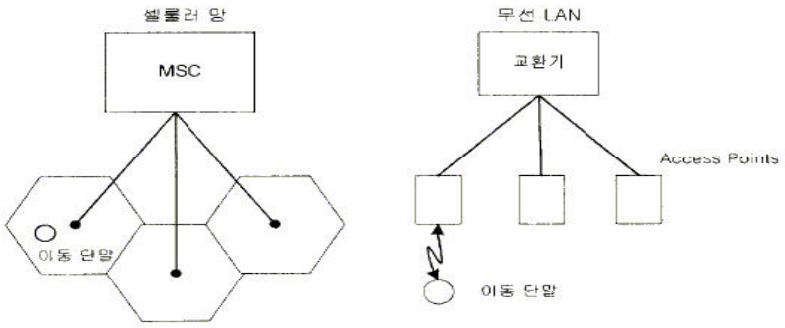
1. 서 론

최근 들어 아주 빠르게 발전하고 있는 무선 통신 환경에서, 차세대 통신 서비스는 현재의 단순한 음성 위주의 서비스가 아닌 PDA(Personal Digital Assistants)와 휴대형 컴퓨터 등의 이동 컴퓨팅 단말의 진보와 함께 고속의 멀티미디어 데이터 서비스로 발전해 가고 있다.

이동 무선 네트워크(mobile wireless network)는 1970년대에 나타나기 시작했으며, 그 이후로 컴퓨터 산업에서 중요한 분야로서 인식되어 왔다. 이동 통신은 음성, 데이터 화상, 동화상 등의 정보들을 교환 할 수 있는 수단을 제공하여, 이동하는 사용자들이 시간이나 위치 접속 방법에 상관없이 정보를 주고받을 수 있도록 하였다.

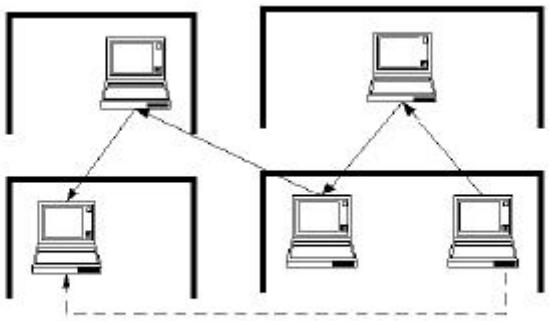
이동 무선 네트워크에는 현재 두 가지 종류가 연구되고 있다. 첫째는 하부구조(infra-structure)기반의 네트워크로서, 셀(cell) 기반 네트워크라고도 불린다. 여기에는 고정되고 유선으로 연결된 게이트웨이가 존재한다. 유선으로 고정된 게이트웨이와 이동 사용자들은 기지국(base station)을 통해 연결된다. 이 네트워크에 있는 이동 호스트(mobile host)는 통신 범위 안에 들어있는 기지국 중 가장 가까운 기지국을 통해 유선 네트워크에 연결하고 통신을 한다. 이동 호스트가 어느 기지국과의 통신 범위를 벗어나고 새로운 기지국의 범위 안으로 들어갔을 때, 기존 기지국에서 새로운 기지국으로 handoff가 일어난다.

이 과정으로 네트워크에서의 끊김없는 통신이 계속 이어지게 된다.



(그림 1) 기존의 무선망 구성도

두 번째 타입의 이동 무선 네트워크는 하부구조가 없는 이동 네트워크이며, 이는 일반적으로 Ad-hoc Network이라 불린다. Internet Engineering Task Force (IETF)에서 규정하고 있는 Ad-hoc 무선망은 전통적인 유선 환경과는 달리 고정된 구조를 가지지 않고 노트북이나 PDA와 같은 이동 컴퓨팅 단말 장치 만으로 구성 된 이동이 자유로운 네트워크이다.
그러므로 Ad-hoc 무선망은 병원, 전시장, 생산 현장 등과 같은 긴박한 상황이나 지속적인 망 연결이 필요 없는 환경에서 적용 가능하다. Ad-hoc 무선 네트워크에서 각각의 이동 노드는 단지 호스트가 아니라 하나의 라우터로 동작하며, 다른 노드에 대해 다중 경로를 가질 수 있다. 또한 동적으로 경로를 설정할 수 있기 때문에 기반구조 없는 네트워킹이라고도 한다.



(그림 2) Ad-hoc 무선망의 동작

특정 시점에서 이동노드의 통신 연결상태는 노드 사이의 위치나 전송 전력레벨, 안테나 패턴, 채널간의 상호 레벨의 함수로 나타낸다. 라우터들 사이의 이동성과 다른 연결 요소들의 가변적인 요소는 잠재적으로 속도감 있고 예측 불가능하게 변하는 망을 가져올 수 있다. 이에 따른 패킷의 손실이 무선 네트워크에서 자주 발생하게 된다[1][18].

TCP는 널리 알려져 있고, 가장 많이 사용되는 인터넷 프로토콜이다. TCP를 이용하여 여러 가지 어플리케이션을 사용할 수 있도록 하기 위해 Ad-hoc 무선망에서도 TCP를 사용하여야 한다. 그러나 Ad-hoc 무선망은 기존의 일반 유선 망과 달리 end-to-end 전달 방식을 사용하는 것이 아니라 홉 단위로 데이터를 전달하는 hop-by-hop 전달 방식을 사용하므로 신뢰성 있는 서비스를 제공하지 못한다. 기존의 연구에서는 실시간 처리를 하는 특정한 하나의 인터넷 트래픽에 관한 연

구나 Ad-hoc 무선망에서 사용되는 라우팅 프로토콜의 각각에 대한 성능을 분석하는 연구가 있어왔다[2],[3]. 또한 인터넷 트래픽을 시뮬레이션 파라미터로 사용하여 라우팅 프로토콜의 성능을 분석한 연구, [6]에서와 같이 Ad-hoc 무선망에서 특정한 TCP 버전과 제시한 알고리즘과의 분석에 대한 연구가 있었지만, 두 개 이상의 라우팅 프로토콜에서 인터넷 트래픽을 사용하여 라우팅 프로토콜을 비교한 연구는 현재 미비한 실정이다.

따라서, 본 논문에서는 Ad-hoc무선 네트워크에서 대표적인 On-Demand 라우팅 프로토콜인 AODV와 DSR을 사용하였다. 이 두 On-Demand 라우팅 프로토콜의 성능을 평가하기 위해 현재 무선 네트워크에서 많이 사용되는 TCP Sack와 대표적인 실시간 트래픽인 CBR을 사용하였다. 본 모의 실험에서는 이동 노드의 수와 이동 노드의 이동 속도를 변화시키는 방식으로 실험을 하였다. 본 논문의 구성은 1장 서론에 이어, 2장에서는 실험에서 사용되는 TCP의 버전별 특징과 Ad-hoc 무선망의 개념, Ad-hoc 무선망에서 사용되는 대표적인 On-Demand 라우팅 프로토콜인 AODV와 DSR의 메커니즘에 대해서 설명한다. 3장에서는 본 논문에서 사용된 NS-2 시뮬레이터와 시뮬레이션 환경에 대해 설명을 하고있으며, 4장에서는 모의 실험 결과 분석에 대해서 설명하며, 마지막 5장에서 본 논문의 결론에 대해 서술하였다.

2. 관련 연구

본 장에서는 모의 실험에 사용된 TCP의 버전별 특징, Ad-hoc 무선 네트워크의 기본 개념과 특징, Ad-hoc 네트워크에 사용되는 라우팅 프로토콜의 종류와 특징에 대해서 설명한다.

2.1 TCP

TCP는 인터넷에서 종단 시스템간의 신뢰성을 보장하는 데이터 전송알고리즘으로 인터넷 서비스의 성능에 많은 영향을 미친다. 현재 TCP에는 망의 혼잡 상황에 대한 대처방식에 따라 여러 가지 버전이 있으며, 가장 많이 사용되는 TCP 버전인 Tahoe, Reno, Sack 버전에 대해서 설명한다.

2.1.1 Tahoe 버전

Tahoe 버전은 초기에 slow-start를 시작하여 패킷을 송수신하다가 혼잡이 발생하면 ssthredhold(Slow-Start threshole)를 패킷이 손실되기 전의 반으로 재 설정한 후, 다시 Slow-Start를 실행하고 만일 cwnd (Congestion Window)값이 현재 설정된 ssthreshold값과 같아지면 그 이후에는 혼잡회피 상태로 들

어가게 된다. Tahoe는 한 윈도우 내에 처음 발생한 손실된 패킷을 재 전송한 후 차례로 Slow-Start를 실행하게 되고 이후의 모든 패킷을 재 전송하게 되므로 수신 호스트가 이미 수신한 패킷을 다시 수신하게 되는 문제점이 있으나 안정성이 높다.

2.1.2 Reno 버전

Reno 버전은 기존의 slow-start, 혼잡회피 알고리즘이외에 빠른 회복과 빠른 재전송 알고리즘을 추가한 버전이다. 빠른 재전송은 만일 수신 호스트에서 순서 번호에 어긋나는 세그먼트가 수신되면, 수신 호스트는 수신되지 않은 순서번호로 확인 응답을 보내게 된다. 이 상황은 올바른 순서번호의 세그먼트를 수신할 때까지 계속된다. 따라서 송신 측이 중복된 확인 응답이 3개 이상 수신하게 되면 세그먼트가 손실되었다는 것으로 인식하여 세그먼트를 재전송 함으로서 타이머가 만료될 때까지 기다리는 시간을 줄이는 방법이다. 빠른 회복은 세그먼트 손실 시 폭주 윈도우의 크기를 1로 설정하여 다시 slow-start 단계로 들어가는 방법을 개선하여 세그먼트 손실 시 윈도우의 크기의 1/2을 현재 윈도우 크기로 설정하고, 바로 충돌 회피 단계를 시작하여 slow-start 단계를 제거한다. 그러나 Reno 버전은 한 ,두개의 세그먼트가 손실되었을 때는 성능이 우수하나 여러 개의 패킷이 손실되면 폭주 윈도우의 크기가 계속 줄어

들어 일정 수 이상을 전달 할 수 없게 되는 상황이 발생할 수 있다.

2.1.3 Sack 버전

Sack 버전은 Reno 버전의 세 개 이상의 확인응답을 수신했을 때 재전송 하는 방식을 개선하여 선택적 확인응답을 사용하는 방식으로 동일한 확인응답이 두 개만 수신되어도 바로 세그먼트 손실로 인정하고 바로 재 전송하는 알고리즘을 추가한 것이다. Sack 버전은 세그먼트 손실에 대한 응답을 두 개만 수신하면 재전송하기 때문에 Reno 버전보다 빠른 재전송을 할 수 있다.

2.2 Mobile Ad-hoc 무선 네트워크 (MANET)

Ad-hoc 무선 Network는 제한된 무선 채널을 통해 이동 단말간의 통신을 수행하는 멀티 홉(multi-hop) 무선 네트워크이다.

2.2.1 Ad-hoc 무선 네트워크의 개요

일반적인 셀룰러 망의 경우 BS(Base Station: 기지국), MSC(Mobile Switching Center), HLR(Home Location Register), VLR(Vistor Location Register) 등과 같은 망 구성 요소들을 가지고 있다. 이 요소들은 망 관리 기능을 수행한다. 그러나 이동 Ad-hoc Network의 경우 유선 백본(backbone)망

이나 기지국, MSC 등과 같은 망 구성 요소들이 없기 때문에 망 관리 기능을 수행하기 위해서는 보다 복잡하고 분산된 알고리즘이 필요하다. 특히 HLR 및 VLR에 의해 지원되는 이동성 관리 기능 및 무선 접속 MAC(Medium Access Control) 기능을 위한 기존의 알고리즘들이 Ad-hoc 환경에서는 적용될 수 없다[11][18].

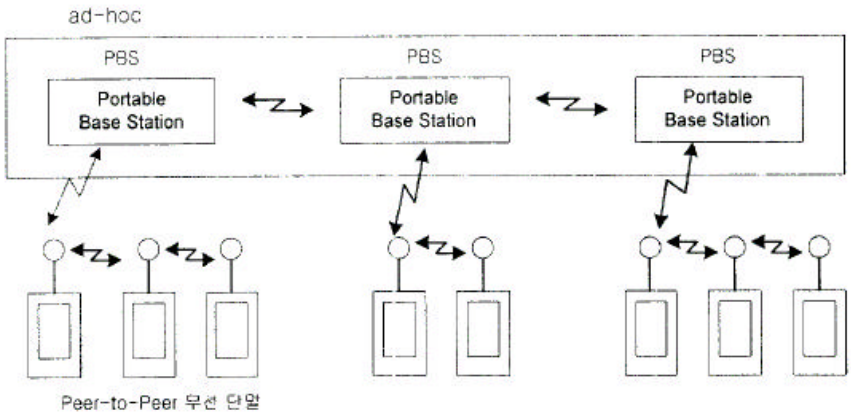
기존의 셀룰러 망의 경우, 각 셀에 기지국이 있어서 셀 안의 모든 이동 단말들의 이동성 관리 및 MAC 기능을 담당하고, 또한 각각의 기지국들은 MSC가 통합하여 관리한다. 마찬가지로 기존의 무선 LAN의 경우에도 시스템 내에 교환기가 존재하고 교환기에 붙어 있는 각각의 access point를 통해 이동 단말들이 통신을 할 수 있다. 반면, Ad-hoc 환경의 경우, 기존 무선망의 기지국, MSC 또는 교환기 등과 같은 장비들이 없이 단말들 자체로서 모든 통신이 이루어져야 하므로 각 단말들이 때로는 중계기 기능을 수행하거나 때로는 peer-to-peer 이동 단말로서 통신이 이루어져야 한다.

Ad-hoc 망에서는 모든 망 구성 요소간의 통신이 제한된 자원을 가지고 수행되기 때문에 효율적인 망 관리 작업을 위해서 무선 스펙트럼을 유지하고 전송 전력을 줄이는 것과 같은 많은 제약 조건이 따른다. 그러므로, Ad-hoc 망에서 두 단말간의 통신을 위해서는 중간 단말을 통해 중계기의 역할을 수행하는 멀티 홉 경로 설정이 필요하다. 더불어 네트워크 토폴로지(topology)가 수시로 변화하기 때문에 최적화 된 경로를 찾고 유지하는 것은 매우 중요하고 어려운 일이다. 그래서

Ad-hoc 경로 설정 알고리즘은 토폴로지 변화에 민첩하게 대응할 수 있어야만 한다[18].

2.2.2 Ad-hoc 무선 네트워크의 특징과 응용분야

Ad-hoc 무선망은 고정된 유선망을 가지지 않고 이동 노드들로만 구성된 망이다. 그러므로 유선망을 구성하기 힘들거나 망을 구성한 후 단기간에 사용되는 경우에 아주 적합하다. Ad-hoc 무선망에서는 각각의 이동 노드에 대한 이동 제약이 없고, 유선망이나 기지국과 같은 기반 구조를 필요로 하지 않기 때문에 병원, 전시장, 생산현장과 같은 여러 환경에서 적용이 가능하며, 천재지면 및 각종 재해로 인해 기간망이 파괴된 경우 복구용 통신 네트웍으로도 활용되고 있고, 최근 들어서는 가정용/사무용 소규모 네트웍 구축과 같은 상업적 용도로의 전환을 위한 연구도 활발히 진행되고 있다[11]. 그러나 Ad-hoc 무선망은 유선망과 같은 기반구조를 가지지 않고 무선이라는 특성 때문에 노드의 이동이 자유롭다는 장점 이외에 유선망의 대역폭보다 훨씬 낮은 2Mbps로 제한되어 있고, 이동 노드의 이동으로 인해 토폴로지가 자주 변화하기 때문에 패킷의 손실이 자주 발생한다는 단점이 있다[6]. 그림 3은 Ad-hoc 무선망의 구성도를 보여주고 있다.



(그림 3) 이동 Ad-hoc 망 구성도

2.3 라우팅 프로토콜의 종류

기존의 유선망과 마찬가지로 Ad-hoc 무선망에서도 어떤 라우팅 프로토콜과 트래픽을 사용하는가에 따라 망의 성능이 좌우된다. Ad-hoc 무선망에서 사용되는 라우팅 알고리즘은 유선망에서 사용되는 라우팅 알고리즘을 기반으로 개발되었다.

Ad-hoc 무선망에서 사용되는 라우팅 프로토콜은 크게 Table-Driven 방식과 On-Demand 방식 두 가지 부류로 나눌 수 있다. Table-Driven 프로토콜은 각각의 이동 노드가 무선망 내의 모든 경로의 정보를 유지하고 있기 때문에 경로 요구 시 최적의 경로를 설정 할 수 있다. 그러나 최신의 정보를 유지하기 위해, 제어 패킷을 통해 주기적으로 정보를 갱신해야 하므로 실제 전달하는 데이터 외에 많은 망 트래픽을 유발하는 단점을 가진다. 대표적인 Table-Driven 라우팅 프로토콜에

는 DSDV, WRP, GSR, FSR 등이 있다. On-Demand 라우팅 프로토콜은 특정 목적지에 대한 경로를 요구하였을 때만 경로 설정 단계가 수행된다. 경로 설정 단계가 완료되어 새로운 경로가 발견되면 경로 유지 단계가 수행된다. 경로 유지는 목적지에게 더 이상의 요구 패킷이 없거나, 목적지까지의 경로를 사용할 수 없을 때까지 계속 수행된다. 이런 프로토콜의 단점은 경로 설정 과정에서 지연이 발생한다는 점과 홉 거리에 따른 관점에서 볼 때 최적의 경로를 보장하지 못한다는 점이다. 대표적인 On-Demand 라우팅 프로토콜에는 CBRP, DSR, AODV, TORA 등이 있다[3][4].

2.4 AODV(Ad-hoc On-Demand Distance Vector) 라우팅 알고리즘

AODV는 Ad-hoc 무선망을 위해 설계된 라우팅 프로토콜이다. 이 프로토콜은 DSR과 DSDV의 조합으로 이루어진 프로토콜로서 경로 탐색과 경로 유지와 같은 On-Demand 라우팅 메커니즘은 DSR에서 가져왔고 hop-by-hop 라우팅, 순서번호, 주기적인 패킷 갱신과 같은 메커니즘은 DSDV에서 가져왔다. AODV의 장점은 DSR과 같이 각각의 패킷에 경로 정보가 포함되지 않으므로 데이터를 전달할 때 오버헤드가 감소되고, 각각의 노드가 망 전체의 라우팅 테이블을 유지하지 않고 각 노드에 인접한 이웃 노드의 라우팅 정보만을 유지하며, 그 정보를 자신의 이웃에게만 방송하기 때문에 DSDV보다는 메모리

사용율이 낮다는 장점을 가진다. 그러나 hello 메시지와 같은 주기적인 메시지를 교환해야 하므로 전체 망 사용율이 증가되는 단점을 가진다.

2.4.1 AODV의 기본 메커니즘

AODV의 기본 메커니즘은 크게 경로 탐색 단계와 경로 유지 단계로 나눌 수 있다. 본 장에서는 경로 탐색 단계와 경로 유지 단계에 대해 설명한다.

2.4.1.1 경로 탐색 단계

경로 탐색 단계는 소스 노드가 자신의 라우팅 테이블에서 라우팅 정보를 가지고 있지 않는 다른 노드와 통신하기를 원할 때 초기화된다. 모든 노드는 노드 순서번호와 방송 ID 두 개의 카운터를 가지고 있다. 어떤 목적지까지 데이터를 전달하고자 할 때, 소스 노드는 자신의 이웃에게 경로 요구(RREQ: Route Request)메시지를 방송함으로서 경로 발견 단계를 시작한다.

목적지 노드나 목적지 노드의 현재 경로를 알고 있는 중간노드가 RREQ를 수신했을 때, 이전에 수신된 RREQ와 동일한 패킷을 수신하게되면 중복된 RREQ를 폐기하고 RREQ를 이웃 노드로 방송하지 않는다. 소스에서 목적지까지 RREQ가 전달될 때 RREQ를 수신한 모든 노드에서 소스까지 자동적으로

역 방향의 경로가 설정된다.

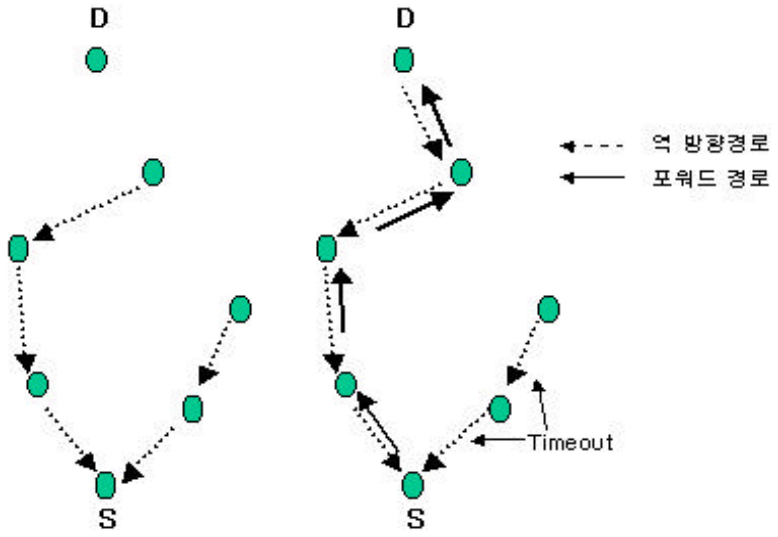
RREQ가 목적지에 대한 현재의 경로를 가지고 있는 중간 노드까지 도착하게되면, 중간 노드는 자신이 수신한 RREQ를 조사한다. 만일 중간 노드가 통신하고자 하는 목적지까지의 경로를 가지고 있다면 중간 노드는 자신의 라우트 엔트리와 수신한 RREQ에 있는 목적지 순서번호를 비교하여 현재의 경로인지를 결정하게된다.

만일 목적지로 향하는 RREQ 메시지의 목적지순서 번호가 중간 노드에 기록되어 있는 목적지순서 번호보다 크면 중간 노드는 자신이 가지고 있는 경로 정보를 사용하지 않고 목적지를 향해 다시 RREQ를 방송한다. 중간 노드는 자신의 목적지 순서번호가 수신한 RREQ가 가지고 있는 순서번호보다 크거나 같을 때만 응답하게 된다. 중간 노드가 현재의 목적지에 대한 경로를 가지고 있다면, 중간 노드는 목적지를 대신해서 RREQ를 방송한 소스 노드로 경로응답(RREP: Route Reply) 메시지를 단방향 전송한다.

목적지로 향하는 경로를 설정 할 수 있는 노드까지 방송 패킷이 도착하면 RREQ의 소스를 향해서 역 방향 경로가 설정된다.

각각의 노드는 RREP를 보낸 쪽으로 포인터를 설정하고, 소스와 목적지까지의 라우트 엔트리에 대한 타임아웃 정보를 갱신하며, 목적지에 대한 가장 최근의 목적지 순서 번호를 기록한다. RREP가 소스 노드까지 도착하게 되면 소스 노드는

RREP를 수신하자마자 데이터를 전송할 수 있고, 만일 소스 노드가 현재의 경로보다 더 나은 경로를 알게 되면 자신의 라우팅 정보를 갱신 할 수 있다[16]. 그림 4는 AODV 라우팅 시 역 방향 경로와 포워드 경로의 생성을 보여준다.



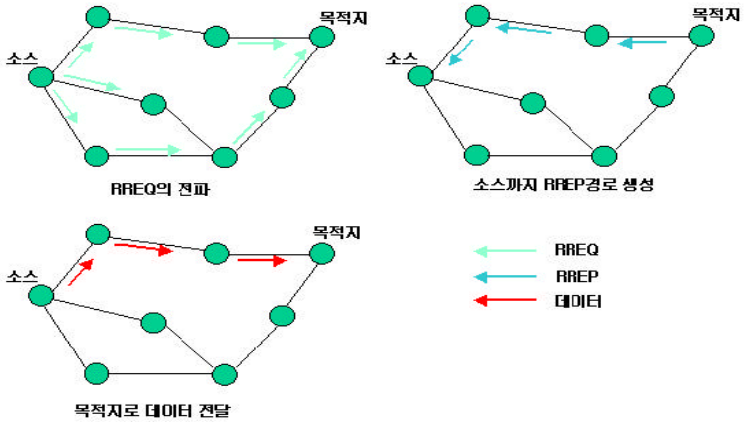
(그림 4) 역 방향 경로와 포워드 경로의 생성

2.4.1.2 경로 유지 단계

데이터 전송 중 소스 노드가 이동하게 되면 소스 노드는 경로 발견 단계를 다시 초기화함으로서 새로운 목적지에 대한 경로를 설정 할 수 있다. 그러나 목적지나 중간 노드가 이동하게 되면 특수한 RREP 메시지가 소스 노드로 전달되게된다. hello 메시지로 불리는 주기적 메시지는 현재의 링크가 양방향

링크라는 것을 보장해줄 뿐만 아니라 링크 실패도 감지할 수 있도록 해 준다. 어떤 노드가 이동해서 목적지 노드로 데이터를 전달할 수 없게되면, 상향 스트림이 파괴된 이전의 노드는 새로운 순서번호와 ∞의 홉 카운트를 가지는 경로 실패 (RERR: Route ERRor) 메시지를 전파한다[2].

이러한 과정은 소스 노드가 RERR 메시지를 수신 할 때까지 계속 전파된다. 노드가 여전히 목적지까지의 경로를 필요로 하면, 소스 노드는 RERR 메시지를 수신하자마자 또다시 경로 발견 단계를 시작한다. 이 때 새로운 RREQ 메시지는 현재 설정하는 경로가 이전에 사용하던 경로가 아니라 새로 설정되는 경로이며, 유효한 경로라는 것을 알리기 위해 이전에 알고있던 목적지 순서 번호보다 하나 더 큰 목적지 순서 번호와 방송 ID로 설정하여 경로 발견 단계를 시작한다. 그림 5는 ADOV에서 데이터 전달이 어떻게 발생하는지를 보여준다.



(그림 5) AODV의 데이터 전달과정

2.5 DSR (Dynamic Source Routing) 알고리즘

DSR의 특징은 소스 라우팅을 사용한다는 것이다[3][6]. 소스 라우팅은 송신 노드가 목적지까지의 완전한 hop-by-hop 경로를 알고 있어야 한다는 것이다. 각각의 경로들은 라우트 캐쉬에 저장되어 있고, 데이터 패킷은 IP 헤더 내부에 있는 라우트 캐쉬의 소스 경로에 따라 전달된다.

2.5.1 DSR의 경로 탐색 단계

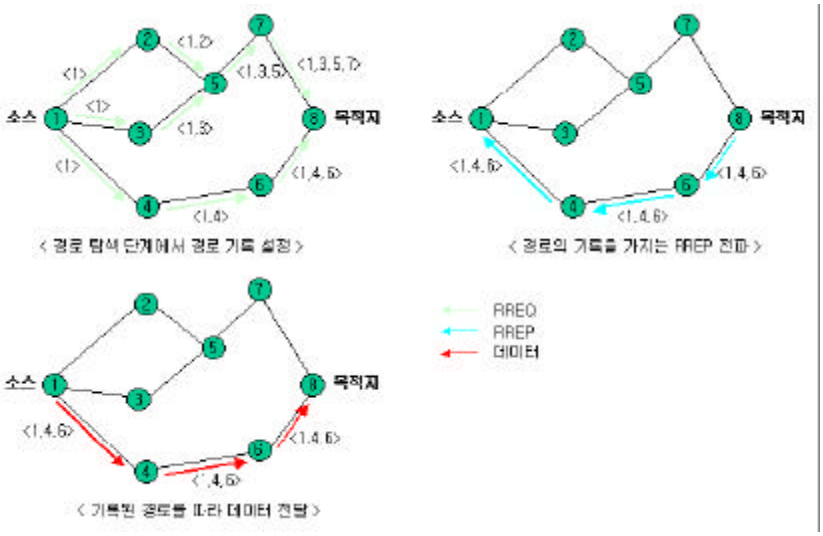
Ad-hoc 망에 있는 어떤 소스 노드가 경로를 알 수 없는 목적지까지 데이터를 전달 하고자 할 때, 소스 노드는 동적으로 목적지까지의 경로를 결정하기 위해 경로 탐색 단계를 시작한다. 경로 탐색 단계는 경로 요구 (RREQ: Route Request) 메시지를 망 전체로 전달함으로써 시작된다. 각 노드가 RREQ를 수신하자마자 RREQ 메시지를 다시 자신의 이웃들에게 방송한다. RREQ를 수신한 노드가 자신의 이웃에게 RREQ를 다시 방송하지 않는 경우는 RREQ를 수신한 노드가 목적지이거나 RREQ를 수신한 노드의 라우트 캐쉬에 목적지까지의 경로를 가지고 있는 경우이다. RREQ 메시지를 수신한 노드는 그에 대한 응답으로 경로 응답(RREP : Route Reply) 메시지를 RREQ를 송신한 최초의 노드까지 전달하게 된다. 여기서

RREP가 전달되는 방향은 RREQ가 방송되는 방향과 반대 방향으로 전달된다. RREP 메시지에 의해 기억된 경로는 차후 데이터 전달에 사용되기 위해 소스 노드에 저장된다[17].

2.5.2 DSR의 경로 유지 단계

만일 소스 경로 상에 있는 어떤 연결이 실패하게되면, 소스 노드는 경로 에러(RERR: Route Error) 메시지를 통하여 경로 실패를 알 수 있다. 소스 노드가 경로 실패를 인식하게되면 소스 노드는 자신의 캐쉬로부터 실패한 경로를 제거하게 된다. 만일 소스 노드가 여전히 목적지까지의 경로가 필요하다면, 소스 노드는 경로 탐색 단계를 다시 시작하게 된다. DSR은 소스 라우팅과 라우트 캐쉬방식으로 구성되어있고, 라우팅 루프를 감지하기 위한 특별한 메커니즘을 사용하지 않는다.

그림 6은 DSR 라우팅 프로토콜이 소스에서 목적지까지 경로 탐색 단계를 거쳐 경로를 설정 한 후, 설정된 경로를 통해서 데이터가 전달되는 과정을 나타낸다.



(그림6) DSR의 데이터 전달과정

3. 시뮬레이션

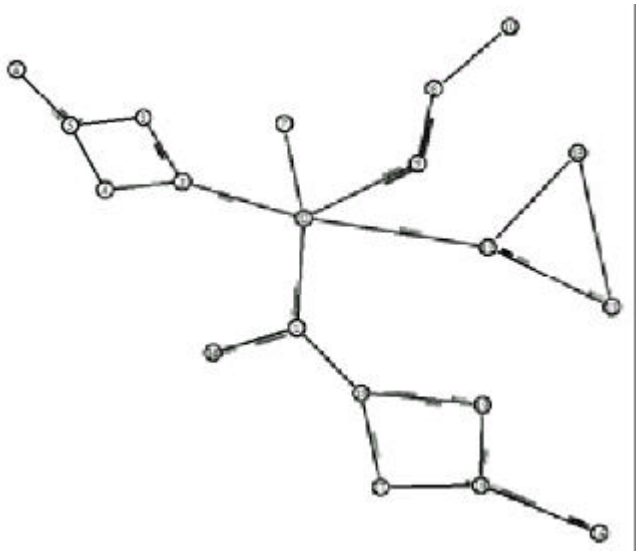
본 장에서는 시뮬레이션 모델과 파라미터에 대하여 설명한다. 시뮬레이션 도구로서는 버클리 대학에서 개발한 분산 객체 네트워크 시뮬레이터인 NS-2(Network Simulator)를 사용하였다[5].

3.1 NS-2 (Network Simulator-2) 시뮬레이터

NS-2는 콜롬비아 대학에 의해 개발된 시뮬레이션 테스트베드인 NEST 를 기반으로 UC Berkeley에서 1988년 REAL이라는 네트워크 시뮬레이터를 개발하였다. 그리고 1989년에 LBNL(Lawrence Berkeley National Laboratory)의 네트워크 연구그룹은 이 REAL을 기반으로 연구를 시작하여 NS-1라고 불리는 시뮬레이션 도구를 개발하였다. LBNL에 의해 개발된 NS-1은 확장된 Tool Command Language(tcl)을 시뮬레이션 기술 언어로 사용하며, 수행하고자 하는 시뮬레이션은 tcl 프로그램으로써 정의되게 된다. 이 tcl 대신에 MIT에 의해 개발된 otcl(Object tcl)을 사용하여 새로운 구조를 갖도록 NS-1을 개선한 것이 NS-2이다.

3.2 시뮬레이션 망 모델

본 논문에서는 Ad-hoc 무선망에서 두 개의 On-Demand 라우팅 프로토콜인 AODV와 DSR을 사용했을 때 각각의 라우팅 프로토콜에서의 인터넷 트래픽의 처리율과 성능을 분석하기 위해 무선에서 가장 많이 사용되는 비 실시간 트래픽인 TCP Sack버전과 대표적 실시간 트래픽인 CBR 트래픽을 사용하였고, MAC 계층의 인터페이스로 IEEE에서 규정한 802.11을 사용하였다. 망 모델은 다양한 트래픽 패턴과 이동 패턴에 의해서 임의적으로 생성하였다. 모의 실험에서 노드는 'random waypoint' 모델에 다라 이동하며, 각각의 시나리오는 유휴시간에 의해서 구별되는데 본 논문에서 유휴시간은 0으로 설정하였다. 실험이 시작되면 각각의 노드는 유휴시간 없이 임의적인 목적지를 선택하고 목적지를 향해 시나리오에 의해 설정된 속도로 이동한다. 그리고 목적지에 도달하면 다시 임의적인 다른 목적지로 이동하는 망을 사용하였다. 모의 실험에 사용된 전체 무선망의 대역폭은 IETF에서 규정된 2Mbps, 전파의 전달 형태는 직접파와 반사파를 같이 사용하는 Two Ray Ground 방식을 사용하였다. 이동 노드가 사용하는 안테나는 일반적인 무선 랜 카드에 장착되어있는 Omni 안테나를 사용하였으며, 이동 노드의 무선 주파수의 최대 전달범위는 250m로 설정하였다. 그림 7은 본 논문에서 사용된 망 모델의 예이다.



(그림 7) 모의실험에서 사용된 무선망의 예

3.3 시뮬레이션 파라미터

본 논문에서 두 라우팅 프로토콜을 분석하기 위해 비 실시간 트래픽인 TCP 중에서 일반적으로 무선 네트워크에 자주 사용되는 TCP Sack버전을 사용하였으며, 어플리케이션으로는 FTP를 사용하였다. 그리고 실시간 트래픽으로는 음성, 영상 등을 전달 할 때 사용되는 CBR(Constant Bit Rate)을 사용하였으며, CBR의 어플리케이션으로서 Telnet을 사용하였다.

Ad-hoc 무선망의 라우팅 프로토콜은 대표적인 On-Demand 라우팅 프로토콜인 AODV와 DSR을 사용하였고, TCP Sack버전과 CBR 두 가지 트래픽을 사용하여 두 프로토콜의 성능을

분석하였다.

본 실험에서는 AODV와 DSR 사이의 처리율 관계를 분석하기 위해 500×500(m)의 토폴로지 크기에 이동 노드의 수를 10, 20, 30, 40개로 가변 시키고, 이동 노드의 이동 속도를 20m/sec로 고정한 Ad-hoc 망을 사용하였다. 그리고 AODV와 DSR에서 이동 노드의 속도와 처리율의 관계를 분석하기 위해서 이동 노드의 수를 10개로 고정하고 노드의 이동 속도를 10 30m/sec로 변화시켰으며, 각각의 실험에서 TCP 데이터 패킷과 CBR 데이터 패킷의 크기를 512 Bytes로 고정하였고, TCP 데이터 패킷을 사용한 모의 실험 시간은 100초 동안 실험하였다. 또한 CBR 트래픽을 사용한 모의 실험에서 사용된 망 역시 토폴로지의 크기는 500×500(m), 이동노드의 개수를 10, 20, 30, 40개로 변화시켜서 300초 동안 모의 실험하였다. 그리고 CBR 트래픽의 성능 분석 요소로서 패킷 처리율의 진동을 기준으로 하였다.

모의 실험에서는 이동 노드의 유휴 시간(Pause time)은 0초로 설정하여 500×500m 크기의 토폴로지에 지속적으로 노드가 이동하는 망을 설계하였다.

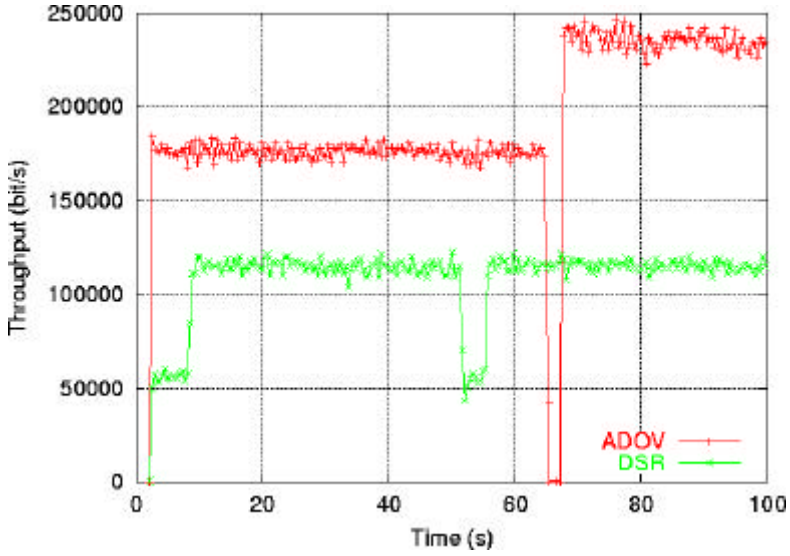
4. 시뮬레이션 결과분석

본 장에서는 대표적인 인터넷 프로토콜인 TCP와 CBR 트래픽을 AODV와 DSR에 각각 적용했을 때 수신 노드에서 트래픽을 처리한 처리율을 기준으로 분석하였으며, TCP와 CBR의 처리율을 분석하는데 있어서 망의 지연율은 고려하지 않았다. 그리고 처리율은 단위 시간당 목적지 노드가 수신한 패킷 양에 시간을 나눈 값을 처리율로 나타내었다.

4.1 이동 노드 수의 변화에 따른 AODV와 DSR에서의 TCP 처리율의 변화

그림 8, 그림 9, 그림 10, 그림 11은 Ad-hoc 무선망에서 이동 노드의 수를 10, 20, 30, 40개로 점차 증가시키고 이동하는 노드의 이동 속도를 20m/sec로 고정하였을 때, AODV와 DSR에서 처리되는 패킷의 처리율을 나타낸다.

그림 8에서는 이동 노드의 수가 10개, 이동 속도를 20m/sec로 고정하였을 때 AODV와 DSR 라우팅 프로토콜에서 TCP Sack 버전의 트래픽이 처리된 처리율을 나타낸다. 그림에서 살펴보면, 모의 실험 초기에 AODV는 약 180kbps,

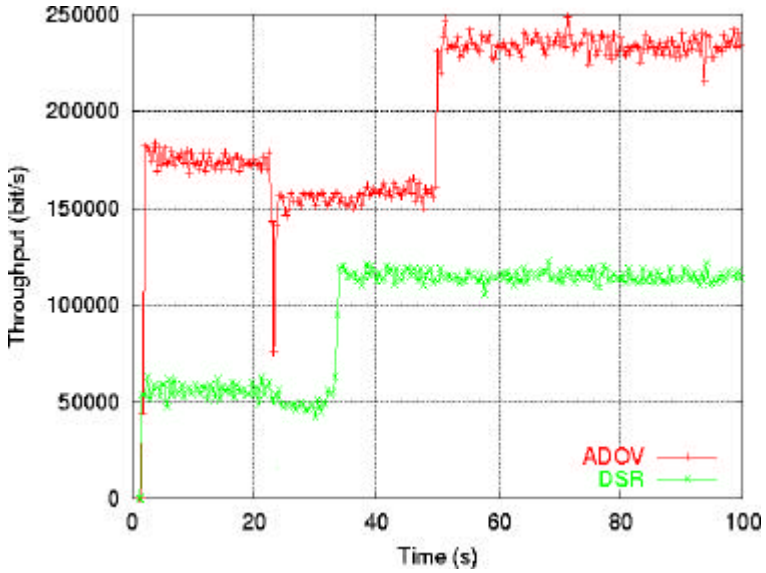


(그림 8) 이동 노드의 개수가 10개, 이동 속도가 20m/sec일 때 AODV와 DSR에서의 처리율

DSR의 경우 140kbps의 처리율을 보인다. 그러나 DSR 라우팅 프로토콜을 사용하는 노드는 약 50초에서 55초 사이, AODV 라우팅 프로토콜을 사용하는 노드는 약 63초에서 65초 사이에 큰 처리율의 변화가 발생한다는 것을 알 수 있다. 이는 DSR의 경우 50 55초 사이 AODV는 63 65초 사이에 목적지 노드나 데이터를 목적지까지 중계하던 이웃 노드가 이동하면서 발생된 경로 재 설정 시간이다. 이때 AODV 라우팅 프로토콜을 사용하는 노드의 경우 일시적이나마 목적지 노드의 처리율이 0으로 떨어짐을 알 수 있다. 그러나 DSR의 경우 처리율은 AODV의 2/3 정도의 수준밖에 되지 않지만 목적지나 데이터를 중계하던 이웃노드가 이동하여 경로가 재 설정되어도 데이

터의 처리율은 0이 되지 않고 50kbps정도의 처리율을 유지한다는 것을 알 수 있다. 이는 각각의 노드가 이미 거쳐갔던 목적지까지의 경로를 라우팅 캐쉬에 저장하기 때문에 그만큼 빨리 경로를 재 설정하여 데이터를 전달할 수 있다는 것이다.

그림 9는 이동 노드의 수를 20개로 변화시키고 노드의 이동 속도를 20m/sec으로 고정했을 때 AODV와 DSR을 사용하는 목적지 노드에서 수신한 TCP Sack의 처리율을 나타낸다.

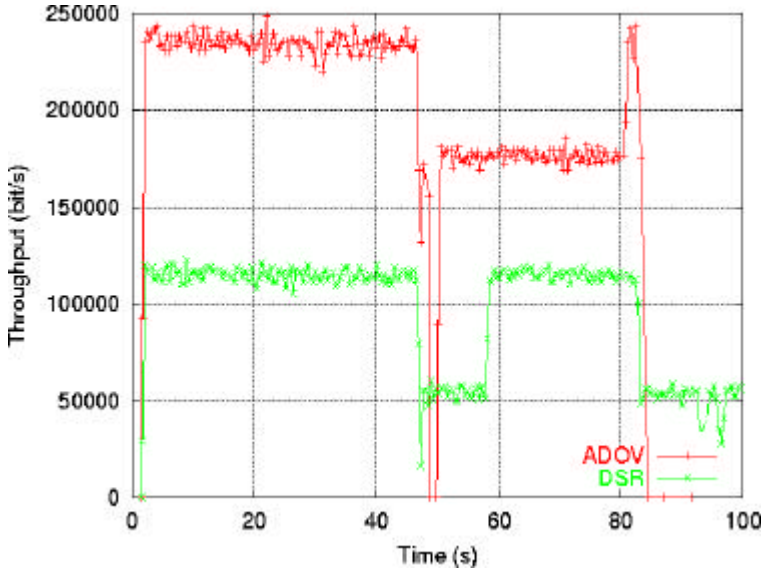


(그림 9) 이동 노드 개수가 20개, 이동 속도가 20m/sec일 때 AODV와 DSR에서의 처리율

AODV를 사용하였을 때 목적지 노드는 시뮬레이션 시간 20초 부근에서 약간의 처리율 변화를 나타내고 있으나 DSR의 경우 처리율이 저하되는 변화는 보이지 않는다. 그러나 전체적인 처리율 면에서 볼 때 AODV의 처리율이 DSR보다 높다는 것을

그림을 통해 알 수 있다.

그림 10은 이동 노드의 수가 30개, 이동속도는 20m/sec 일 때 AODV와 DSR의 처리율을 나타낸다. AODV에서는 시뮬레이션 시간 45초와 80에 한 번씩 잠시나마 처리율이 0으로 떨어지지만, DSR의 경우 처리율의 변화가 발생하더라도 극단적인 변화가 발생하지 않고 낮은 처리율이라 하더라도 일정한 처리율을 유지함을 알 수 있다. 그러나 이동 노드 수가 늘어남에 따라 AODV의 최대 처리율은 별 변화가 없는 반면에 DSR에서 최대 처리율은 노드수가 늘어남에 따라 약간 감소한다는 것을 알 수 있다.

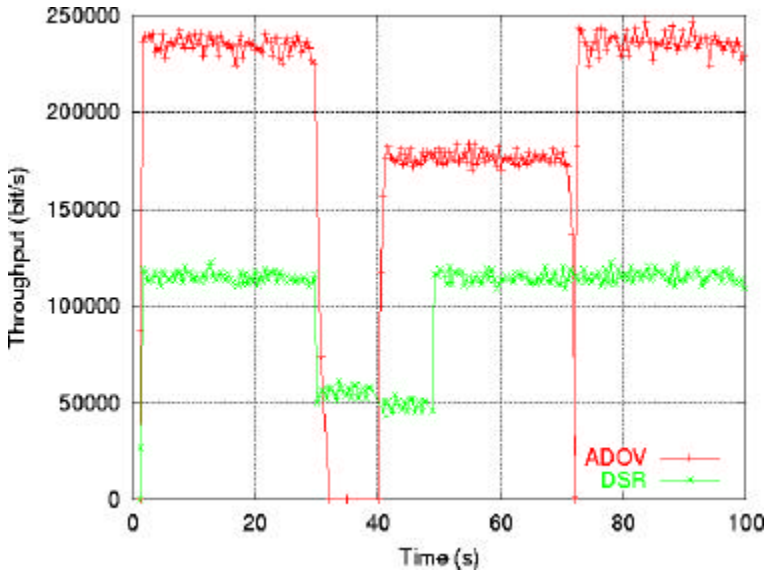


(그림 10) 이동 노드의 개수가 30개, 이동 속도가 20m/sec일 때 AODV와 DSR의 처리율

이는 DSR 라우팅 프로토콜로 경로를 설정하고 데이터를 전

달 할 때 각각의 데이터 헤더에 데이터가 이동했던 모든 경로를 가지고 이동하기 때문이다. DSR에서는 데이터 헤더 내부의 오버헤드 때문에 노드 수가 증가할수록 처리되는 데이터가 줄어든다는 것을 의미한다.

그림 11은 이동 노드 수가 40개로 증가하고, 이동 노드의 이동 속도는 20m/sec로 고정되었을 때 AODV와 DSR 각각의 처리율을 나타낸다. AODV에서는 30 40초, 68초 부근에서 노드 이동에 의한 경로 재 설정 관계로 처리율이 0으로 떨어진다는 것을 알 수 있다.



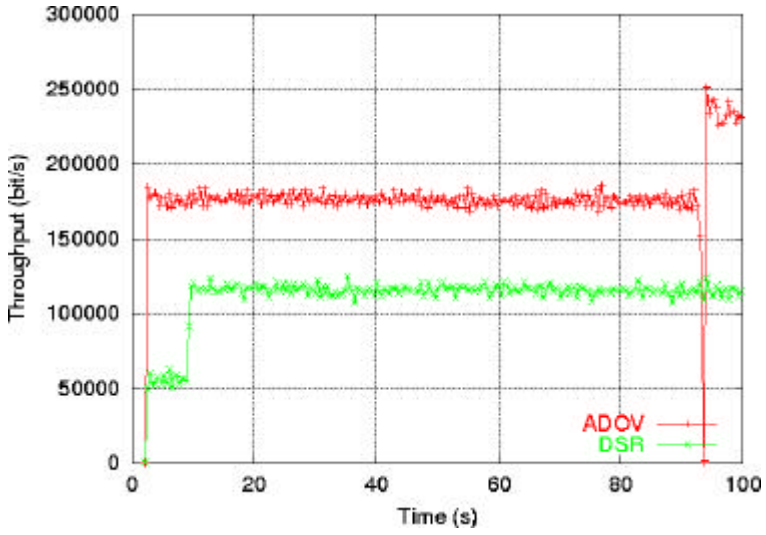
(그림 11) 이동 노드의 개수가 40개, 이동 속도가 20m/sec일 때 AODV와 DSR의 처리율

반면에 DSR에서는 그림 10에서와 같이 약간의 처리율의 변화는 보이지만 처리율이 0으로 떨어지지는 않는다. 그러나

최대 처리율은 노드가 증가할수록 DSR이 감소한다는 것을 알 수 있다.

4.2 이동 속도 변화에 따라 AODV와 DSR에서의 TCP 처리율 변화

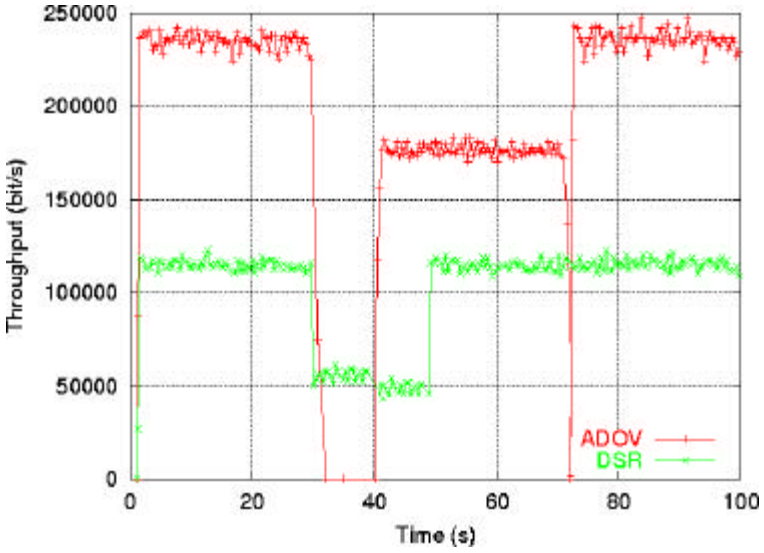
그림 12, 그림 13, 그림 14는 이동노드의 수를 20개로 고정하고, 이동노드의 이동 속도를 각각 10 30m/sec로 변화했을 때 AODV와 DSR에서 TCP 처리율의 변화를 나타내는 그림이다. 그림 12는 이동 노드의 수가 20개이고, 노드의 이동 속도는 10m/sec일 때 AODV와 DSR에서의 처리율을 나타낸다.



(그림 12) 이동 노드의 개수가 20개, 이동속도가 10m/sec일 때 AODV와 DSR의 처리율

AODV에서는 시뮬레이션 시간 92초에서 노드의 이동으로 인한 처리율의 감소를 보이고 있다. 반면에 DSR의 경우, 시뮬레이션 초기에 약간 낮은 처리율을 보이다가 거의 일정한 처리율을 보인다는 것을 알 수 있다.

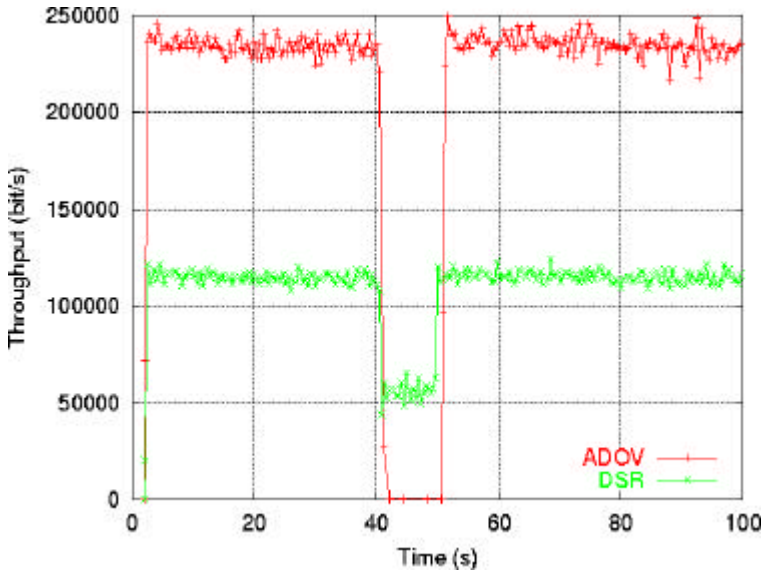
그림 13은 이동 노드 수가 20개, 노드의 이동 속도를 20, 30m/sec로 변화 시켰을 때 AODV와 DSR의 처리율 변화를 나타낸다. 그림 13에서 AODV는 전체 시뮬레이션 시간 중에서 두 번의 급격한 처리율 변화를 보이는 반면에, DSR은 처리율의 변화를 보이긴 하지만 AODV보다는 일정한 대역을 확보한 상태로 데이터를 처리한다는 것을 알 수 있다.



(그림 13) 이동 노드의 개수가 20개, 이동 속도가 20m/sec일 때 AODV와 DSR의 처리율

그림 14에서도 그림 13과 같이 AODV에서는 처리율의 급격

한 변화를 보이는 반면, DSR에서는 일정 대역을 유지한다는 것을 알 수 있다.



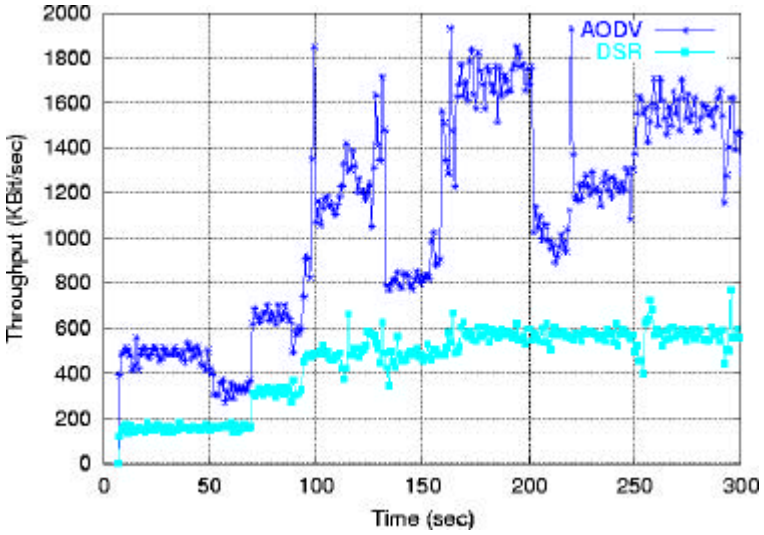
(그림 14) 이동 노드의 개수가 20개, 이동 속도가 30m/sec일 때 AODV와 DSR의 처리율

그림 12, 그림 13, 그림 14에서 볼 때 DSR은 노드의 이동 속도와는 상관없이 그리 큰 처리율의 변화를 보이지 않는 반면에, AODV에서는 노드의 이동 속도와 노드 개수에 따라 처리율의 급격한 변화가 발생한다는 것을 알 수 있다. 이는 DSR에서는 이동노드의 유휴시간(Pause time)이 길어도 처리율에 그다지 큰 영향을 미치지 않고, AODV에서는 유휴 시간에 따라서 안정된 처리율을 보일 것이라는 것을 간접적으로 알 수 있다.

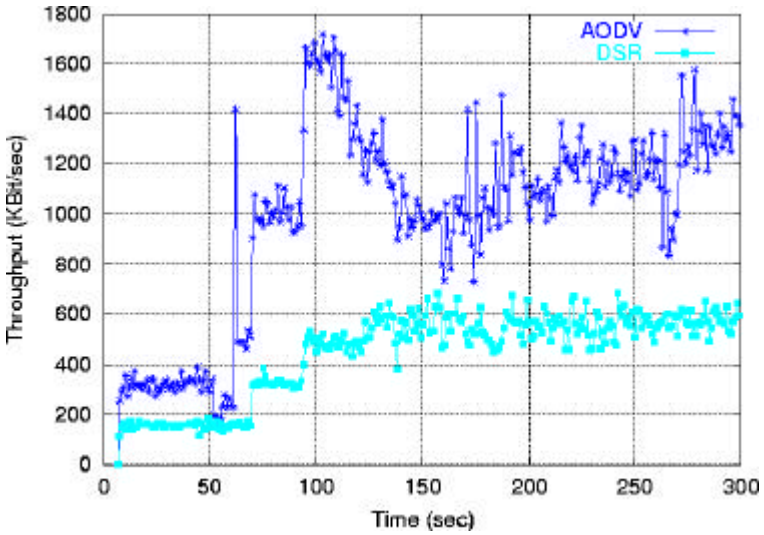
4.3 AODV와 DSR 상에서 CBR 트래픽의 처리율 변화

그림 15, 그림 16, 그림 17, 그림 18은 500×500(m) 크기의 Ad-hoc 망에서 노드의 이동 속도는 10m/sec로 고정시키고, 노드 수를 10, 20, 30, 40개로 변화 시켰을 때 AODV와 DSR 라우팅 프로토콜의 CBR 트래픽 처리율을 나타내고 있다.

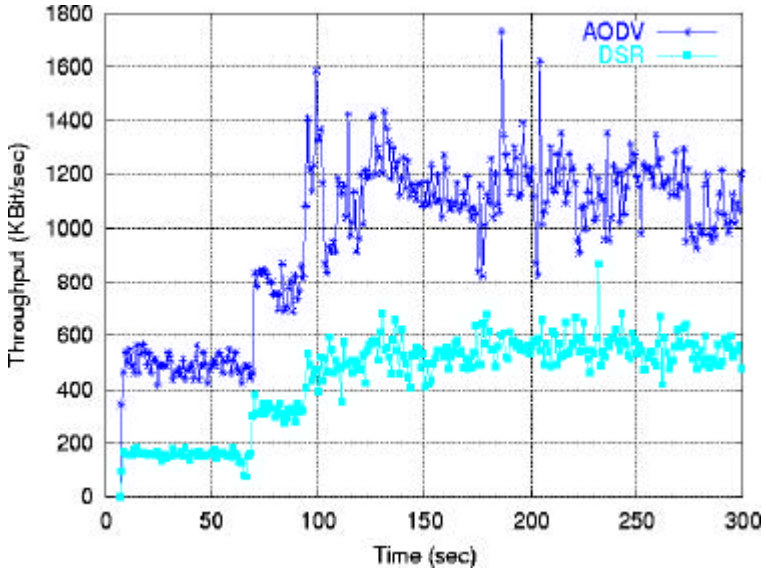
각각의 그림에서 살펴보면 AODV는 TCP 트래픽을 사용했을 때처럼 극단적인 데이터의 끊김 현상이 발생하지 않는다. 그러나 DSR보다 트래픽 처리율이 높으며, 처리율의 변화가 크다는 것을 알 수 있다. 이는 AODV가 데이터를 전송하기 위해 경로 탐색 단계를 시작할 때, DSR과는 달리 각각의 노드가 자신의 바로 인접한 이웃 노드의 라우팅 테이블만을 유지함으로 인해 DSR보다 낮은 라우팅 오버헤드를 가지기 때문이다. 반면에 DSR은 AODV보다 처리율은 떨어지지만, 처리율의 변화가 적으므로 보다 안정적으로 데이터를 처리하고 있다는 것을 알 수 있다.



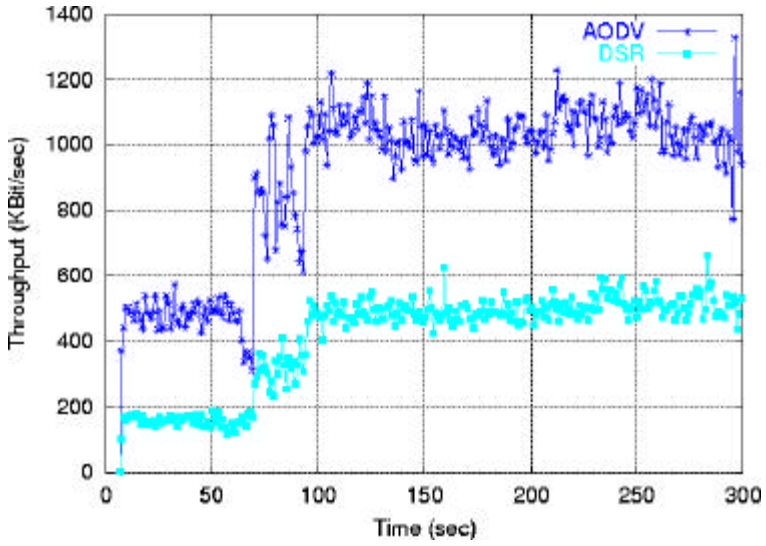
(그림 15) 이동 노드의 개수가 10개일 때 AODV와 DSR의 처리율



(그림 16) 이동 노드의 개수가 20개일 때 AODV와 DSR의 처리율



(그림 17) 이동 노드의 개수가 30개일 때 AODV와 DSR의 처리율



(그림 18) 노드수가 40일 때 AODV와 DSR의 처리율

5. 결 론

최근 몇 년 동안 Ad-hoc망에서 다양한 라우팅 프로토콜이 연구되어 왔다. 그러나 이러한 프로토콜 상에서 인터넷 트래픽 성능에 관한 연구가 미진했으며, 특히 AODV상에서 트래픽 분석은 거의 없었다.

본 논문에서는 인터넷 트래픽으로써 TCP Sack 버전과 실시간 트래픽 CBR을 사용하였고, 분산객체 네트워크 시뮬레이터인 NS-2를 사용하여 Ad-hoc Routing Protocol들 중에서 On-Demand 방식인 AODV와 DSR을 이용하여 두 라우팅 프로토콜의 성능을 분석하였다. 본 논문에서 시뮬레이션은 노드의 수와 노드의 이동 속도를 변화 시켜 실험하였다. 성능 평가 요소로는 시간에 따른 처리율을 사용하였으며, 지연은 고려하지 않았다.

모의 실험 결과 AODV 라우팅 프로토콜은 TCP와 CBR 트래픽을 사용했을 때, DSR 보다는 높은 처리율을 보였다. 반면 노드의 이동 속도와 노드 수 변화에 민감한 반응을 보였다.

DSR은 노드의 이동 속도와 노드의 개수가 변화하여도 처리율이 일정한 수준을 유지하여 안정적이었다. 그러나 전체적인 처리율 면에서는 AODV보다 다소 떨어진다는 것을 알 수 있었다. AODV는 노드의 이동 속도와 이동 노드의 개수가 증가함에 따라서 극단적인 데이터의 끊김이 발생하는 등의 민감한

반응을 보인 반면, DSR은 노드의 이동속도나 노드 수가 증가하더라도 일정 크기의 대역을 유지하여 데이터를 처리한다는 것을 알 수 있었다.

참고 문헌

[1] S. Corson and J. Macker, “Mobile Ad-hoc Networking(MANET)", Internet Dreft, IETF, Oct, 1998.

[2] Charles E. Perkins, Elizabeth M. Royer and Samir Das, "Ad-hoc On-Demand Distance Vector (AODV)Routing", Oct, 99 IETF Draft, pp. 33 38.

[3] Josh Broch, D. A. Maltz. "The dynamic source routing protocol for Ad-hoc networks", draft-ietf-manet-dsr-01, Dec 1998.

[4] Elizabeth M. Royer, Chai-Keoeng Toh, “A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks," IEEE, April, 1999.

[5] The VINT Project", ns Notes and Documentation", February 25, 2000.

[6] D. Johnson and D. Maltz. "Dynamic source routing in Ad-hoc wireless networks.", Mobicom computing, 1996.

[7] J. Broch, D. A. Maltz, D. B. Johonson, Y, C. Hu, and J. Jetcheva, "A performance Comparison of Multi-hop wireless Ad-hoc Network Routing Protocols," In proceeding of the Fourth Annual ACM/IEEE

international Conference on Mobile Computing and Networking, Oct. 1998.

[8] IEEE standard Department, "Wireless Lan Medium access control(MAC) and physical layer(PHY) specifications", IEEE standard 802.11, 1997.

[9] Gavin Holland, Nitin Vaidya, "Analysis of TCP Performance over Mobile Ad Hoc Networks", Proceedings of Mobicom'99, Seattle.

[10] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang. MACAW: A media access protocol for wireless LAN‘s. In Proceedings of the SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications, pages 212, August 1994.

[11] James Geier, Martin DeSimio, and Byron Welsh. Network routing techniques and their relevance to packet radio networks. In *ARRL/ CRRL Amateur Radio* 9th Computer Networking Conference, pages 105'17, September 1990.

[12] C. Hedrick. Routing Information Protocol. Internet Request For Comments RFC 1058, June 1988.

[13] Allman M " On the generation and use of TCP acknowledgements ", ACM Computer.

[14] Communication Review, October 1998, 17 pages

Presents a simulation study of TCP acknowledgement generation and utilization techniques.

[15] Biaz S, Vaidya N et al "TCP over wireless networks using multiple acknowledgements", Texas A&M University, Technical Report 97-001, January 1997, 10 pages.

[16] 고영웅, 마주영, 유동철, 박승섭, “Ad-hoc 무선망에서 AODV 라우팅 프로토콜을 이용한 TCP 트래픽의 성능 분석”, 인터넷 정보학회 논문지, 2권 3호, pp. 9 16, 2001년 8월

[17] 이규남, 고영웅, 박승섭, “AODV와 DSR 라우팅 프로토콜 상에서 인터넷 트래픽의 성능비교”, 한국 멀티미디어학회 춘계학술대회 논문집, 4권 1호 , pp. 261 264, 2001년 6월

[18] 류동열, “TORA상에서의 TCP 트래픽 성능분석에 관한 연구", 부경대학교 공학석사 학위논문, pp. 4 9, 2001년 6월

감사의 글

길다면 길었던 2년이 어느덧 지나갔습니다. 지난 2년 동안 여러모로 도움을 주신 많은 분들이 생각납니다. 그 동안 많은 지도와 조언을 아끼지 않으시고 부족한 면을 채워주셨던, 박승섭 교수님께 진심으로 감사를 드립니다. 또한 귀중한 시간을 할애하여 논문이 완성되기까지 도와주신 윤성대, 조우현 교수님께도 감사를 드리며, 여러 가지 면에서 지도해주신 박만곤, 윤성대, 박흥복, 여정모, 박지환, 김창수, 이경현, 김영봉 교수님들께 감사드립니다.

불철주야로 연구에 매진하고 동고동락을 같이했던 육동철 선배님께 감사드리며, 이번에 같이 졸업하여 미국으로 떠날 산업 대학원의 동렬 선배와 이미 미국으로 출국하신 박봉주 선배님, 매사에 아낌없이 도와주었던 동기 마주영, 저의 절친한 일반 대학원 동기 대업, 금호, 찬희, 형욱 형에게 감사 드립니다. 그리고 이미 졸업하신 박희대 계장님, 전흥진 선배님, 오명석 선배님과 김명희 선배님께 감사드립니다. 또한 이규남 선생님, 문병윤 선생님께 감사드리고 후배 영하와 연구실의 학부생인 도기, 정섭, 승주, 현정에게 많은 감사를 드립니다.

연구실은 달랐지만 언제나 저에게 힘이 되었던, 황순환 선배, 상일씨, 현호 형님께 감사 드립니다. 그리고 마지막으로 저에게 오늘이 있게 해 주신 아버지와 어머니, 두 동생에게 감사를 드리며, 저의 친구들인 동호, 유근, 현호, 맹기형, 창호, 문성, 손 문, 성이, 은희 그리고 내가 아는 모든 분들에게 감사드립니다.