



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

碩 士 學 位 論 文

Ad hoc 네트워크에서의 양방향
경로탐색을 이용한 경로복구 개선 방안

2009年

漢城大學校 大學院

情報 시스템 工學科

情報시스템工學專攻

韓 昊 延

碩 士 學 位 論 文

指導教授 南 斗 熙

Ad hoc 네트워크에서의 양방향
경로탐색을 이용한 경로복구 개선 방안

An improved route recovery using Bidirectional
Searching Method for Ad hoc Networks

2008年 12月 日

漢城大學校 大學院

情報 시스템 工學科

情報시스템工學專攻

韓 昊 延

碩 士 學 位 論 文

指導教授 南 斗 熙

Ad hoc 네트워크에서의 양방향
경로탐색을 이용한 경로복구 개선 방안

An improved route recovery using Bidirectional
Searching Method for Ad hoc Networks

위 論文을 工學 碩士學位 論文으로 提出함

2008年 12月 日

漢城大學校 大學院

情報 시스템 工學科

情報시스템工學專攻

韓 昊 延

韓昊延의 工學 碩士學位 論文으로 認准함

2008年 12月 日

審査委員長 이 기 원 (印)

審査委員 남 두 희 (印)

審査委員 김 승 천 (印)

- 목 차 -

I. 서론	1
1. 연구 배경	1
2. 연구목적	2
3. 연구 방법	2
II. 문헌고찰	5
1. 애드혹 네트워크	5
가. 개요	5
나. 특징	7
2. 라우팅 프로토콜	9
가. DSDV	9
나. DSR	13
다. AODV	20
3. 관련 연구	27
III. 제안 알고리즘	31
1. 경로 복구 절차 문제점	31
2. TW-AODV	33
3. 시뮬레이션	41
가. 실험 환경	41
나. 실험 결과	47
IV. 결론	59
V. 참고문헌	62

－ 표 목 차 －

<표 1> RREQ 메시지 필드 - AODV	22
<표 2> RREP 메시지 필드 - AODV	23
<표 3> RERR 메시지 필드 - AODV	24

- 그림 목 차 -

<그림 1> 연구 방법	4
<그림 2> 애드혹 네트워크	6
<그림 3> 애드혹 네트워크 활용	7
<그림 4> 애드혹 네트워크 범위	8
<그림 5> 멀티홉 라우팅	9
<그림 6> DSDV 수행과정	12
<그림 7> DSDV 알고리즘	13
<그림 8> RREQ 메시지 구조 - DSR	15
<그림 9> RREP 메시지 구조 - DSR	16
<그림 10> RERR 메시지 구조 - DSR	16
<그림 11> DSR 알고리즘 수행과정	19
<그림 12> RREQ 메시지 구조 - AODV	22
<그림 13> RREP 메시지 구조 - AODV	23
<그림 14> RERR 메시지 구조 - AODV	24
<그림 15> 경로 탐색 과정 - AODV	26
<그림 16> 경로 복구 과정 - AODV	27
<그림 17> 데이터 및 제어패킷 비율	32
<그림 18> AODV 지역 복구 적용 범위	33
<그림 19> B-RREQ 구조	34
<그림 20> B-RREP 구조	34
<그림 21> 경로 손실 감지	35
<그림 22> 에러 메시지 전송 단계	36
<그림 23> 지역 경로 탐색	37
<그림 24> 경로 정보 응답	38

<그림 25> 경로 정보 전달	39
<그림 26> 전체 경로 복원	39
<그림 27> 제안 알고리즘 수행과정	40
<그림 28> NS-2 시뮬레이터 구조	42
<그림 29> NS-2 시뮬레이터 작업 절차 개념도	43
<그림 30> NS-2 시뮬레이터 클래스 구조	44
<그림 31> 노드 50, 이동속도 5m/s - 오버헤드, 전송률	48
<그림 32> 노드 70, 이동속도 5m/s - 오버헤드, 전송률	50
<그림 33> 노드 90, 이동속도 5m/s - 오버헤드, 전송률	52
<그림 34> 노드 50, 이동속도 10m/s - 오버헤드, 전송률	54
<그림 35> 노드 50, 이동속도 15m/s - 오버헤드, 전송률	56

I. 서론

1. 연구 배경

언제 어디서나 정보를 이용할 수 있는 유비쿼터스의 시대적 흐름에 따라 통신 네트워크 환경이 변화하고 있다. 지금까지 네트워크 환경은 전통적인 유선 네트워크(Wired Network)에서 이동성을 보장한 무선 네트워크(Wireless Network)로 진화해왔다. 하지만 단순한 무선 네트워크 환경은 통신범위가 제한되어 있고 네트워크 기반시설에 필요하다는 점 때문에 제한적인 이동성을 제공하고 있다. 이에 새로운 무선 네트워크 환경인 애드혹 네트워크(Ad hoc Network)는 통신 허브와 같은 기반 시설 없이도 통신 노드들간의 자체적인 네트워크를 구성할 수 있다는 이점 때문에 큰 각광을 받고 있다.

애드혹 네트워크는 각 노드들의 자유로운 이동성을 보장하고 네트워크 전체를 관리하는 시설 없이 자체적인 망을 구성하므로 모든 통신 노드들의 위상이 동등하다. 애드혹 네트워크는 네트워크 관리에 대한 책임이 각 노드들에게 분산되어 있다는 점 때문에 기존의 네트워크에서 사용하였던 여러 기법들을 그대로 적용하기는 어렵다. 네트워크 설정 및 초기화(Configuration and Initialization), 라우팅 프로토콜(Routing Protocol), IP 주소 관리(IP Address Management), 보안(Security) 등이 애드혹 네트워크에서 주요 연구 주제로 다루어지고 있다.

특히, 라우팅 프로토콜 분야에서는 통신 노드들의 빈번한 이동으로 발생되는 링크 손실(Link Failure)에 대응하기 위한 경로 복구(Route Recovery)에 대한 연구가 활발히 진행되고 있다. 애드혹 네트워크에서 사용되는 여러 표준 프로토콜에서 이런 경로 복구에 대한 해결책과 여러 연구들에서 이를 개선하고자 하는 기법들을 제시하였다. 지금까지 제시된 기법들은 경로 복구를 위한 제어 패킷의 수가 많거나, 복구 시간이 지연되는 경우가 많아 실질적인 데이터 전송을 하는데 어려운 점이 많다. (박노열,

2. 연구목적

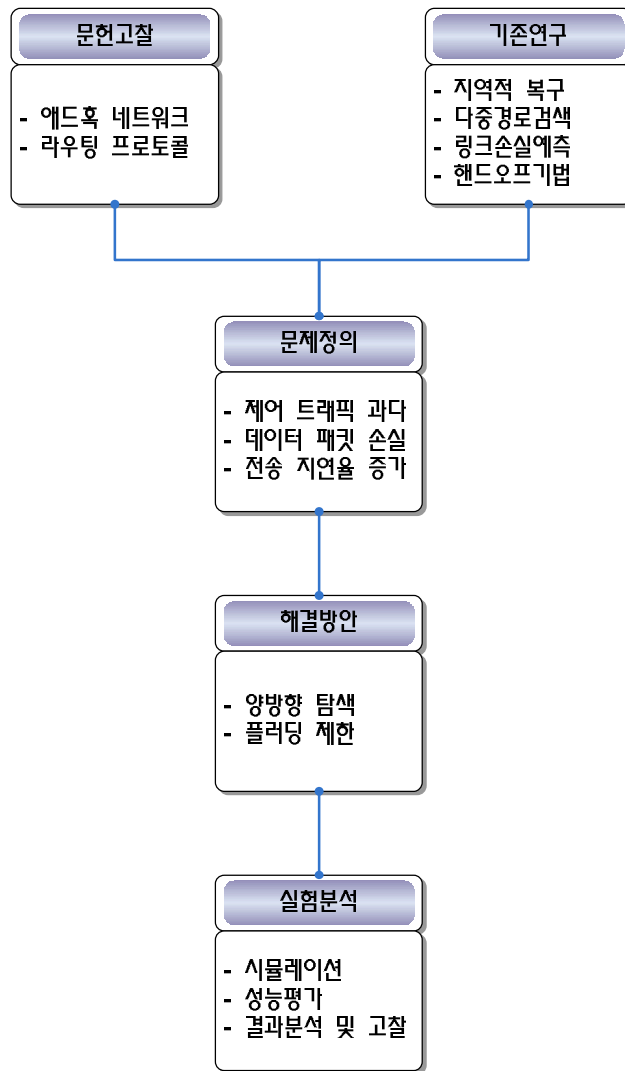
본 연구에서는 애드혹 네트워크에서 발생하는 경로 손실을 효율적으로 해결할 수 있는 경로 복구 기법을 제시한다. AODV(Ad hoc On-demand Distance Vector)는 기존 라우팅 프로토콜 중 대표적인 요구기반 방식으로써 이에 대한 여러 연구가 진행되고 있다. AODV 프로토콜에서는 링크 손실에 대처하기 위하여 주기적인 이웃 노드들에 대한 존재 확인과 손실 발생 시 에러 메시지를 출발지 노드에게 통보하여 경로를 재탐색하는 방법을 사용하고 있다. 이 과정에서 제어 패킷 브로드캐스트에 따른 오버헤드와 네트워크 혼잡으로 인한 데이터 패킷 손실 문제가 발생하고 있다. 이를 개선하기 위해 주변 노드를 이용한 지역적 복구 기법(local repair)을 적용하였으나, 목적지 노드와 가까운 경우에만 이용되고 있어 실질적으로 큰 성능 개선이 이루어지지 않고 있다. 본 연구에서 제시하는 기법은 AODV를 기반으로 하고 있으며, 링크 손실 위치에 관계없이 적용할 수 있는 지역적 복구 메커니즘을 제안한다.

제한된 브로드캐스트를 이용한 양방향 경로 복원 기법을 제안하여 제어 패킷 비율을 최소화함으로써, 실질적인 데이터 전송률을 개선하고자 한다. 애드혹 네트워크에서는 유선 네트워크보다 링크에 대한 신뢰성이 낮기 때문에, 네트워크 관리에 대한 정보는 대부분 브로드캐스트 방식으로 전송된다. 브로드캐스트 패킷은 전송 특성 상 노드들로 전송됨에 따라 패킷 수가 기하급수적으로 늘어나 심각한 네트워크 혼잡을 유발할 수 있다. 이러한 제어 패킷의 수를 감소시킴으로써, 전체적인 네트워크 처리량(Throughput)과 성능(Performance)을 향상시킬 수 있다.

3. 연구 방법

본 논문에서는 현재까지 제시된 이론과 연구를 바탕으로 문제점을 도출하고 이를 해결할 수 있는 방안과 실험 및 검증 과정으로 연구를 진행한다.

연구가 이루어지는 환경인 애드혹 네트워크의 전반적인 특성과 애드혹 네트워크에서 쓰이고 있는 라우팅 프로토콜에 대한 문헌을 고찰하고 경로 복구 기법에 관한 국내외 관련 논문들을 조사하여 기반 연구를 수행한다. 이를 토대로 현 상황에 대한 문제점 및 한계점을 제시하고 이를 해소할 수 있는 제안 알고리즘을 제시한다. 마지막으로 제안된 기법의 정량적 효과 척도를 가늠하기 위한 시뮬레이션과 결과 분석을 통해 최종 결론을 도출한다.



<그림 1> 연구 방법

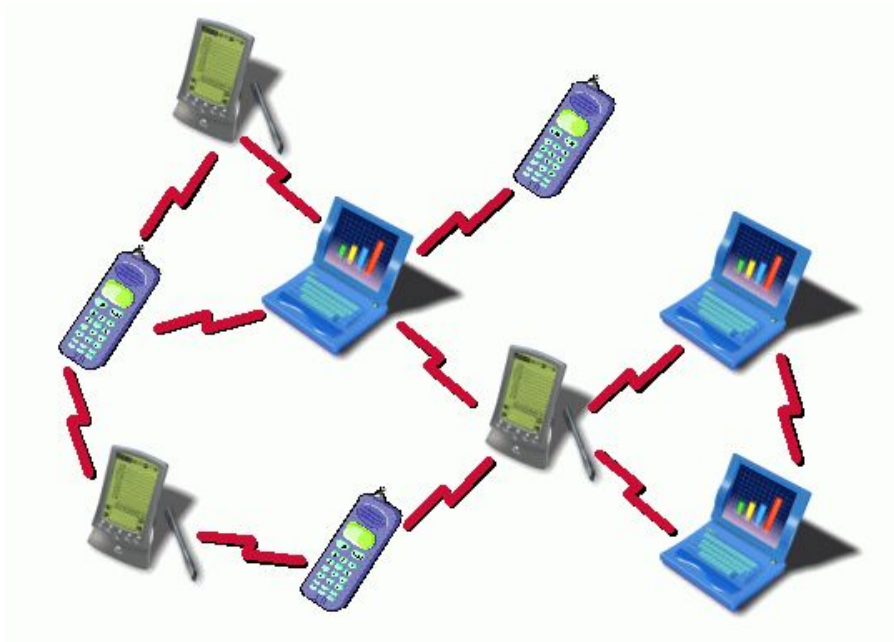
II. 문헌고찰

1. 애드혹 네트워크

가. 개요

무선통신(Wireless Communication) 방법은 통신 노드들끼리 와이어나 케이블 등으로 직접적으로 연결되어 있지 않아도 이용자들간의 자유로운 정보 전송을 가능하게 한다. 무선통신망 중 핸드폰을 이용한 CDMA망과 무선랜(Wireless LAN)망이 가장 널리 이용되고 있다. 이들은 통신 기지국이나 중계 서버, AP(Access Point)등과 같은 기반시설(Infrastructure)의 관리하에 운영되고 있다.

애드혹 네트워크는 기존 무선 네트워크망과는 달리, 사전의 네트워크 설정 작업이나 관리 시설 구축 없이도 통신 단말기 이용자들끼리 자체적으로 구성될 수 있다. 전체 네트워크에 대한 중심적인 역할을 하는 주체가 없고, 모든 노드가 동등한 위상을 갖는다. 또한 이동성이 보장되어 있어, 네트워크의 참여/이탈이 자유롭고 통신 거리에 상관없이 데이터 전송이 가능하다. 둘 이상의 통신 가능한 단말기만 있다면 언제 어디서나 손쉽게 네트워크를 구성할 수 있다.



<그림 2> 애드혹 네트워크

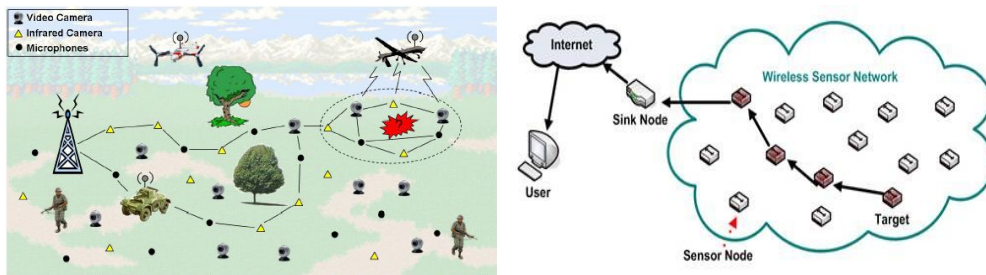
초기 애드혹 네트워크는 군사적인 목적으로 주로 사용되었다. 적진 환경에서의 빠른 네트워크 구성, 기반 시설 파괴에 따른 위험성 문제 등을 해결하기 위해 애드혹 네트워크가 고안되었다. 미 국방성 DARPA에서 1972년 PRnet(Packet Radio Network)을 개발한 것으로 시작으로 각각 1983년, 1994년에 SURAN(Survivable Radio Network)와 GloMo(Global Mobile) Information Systems을 발표하였다. 호주에서는 주로 지역 대학과 연구소들을 중심으로 하는 연구개발 프로젝트가 이루어졌다. 패킷 구조 연구(Packet Structure Research, 1991), 자가망 조직과 적응형 네트워크(Self-Organizing and Adaptive Links and Networks, 1993), 군사형 애드혹 네트워크에서의 라우팅 기법(Routing in Military Ad Hoc Networks, 2003) 등에 대한 연구가 진행되었다.

애드혹 네트워크는 신속한 네트워크 구성이 요구되는 경우나 네트워크 기반 시설 구축 및 관리 비용이 많은 경우에 주로 이용된다.

- 회의실에서 노트북이나 PDA와 같은 컴퓨터를 이용하는 사람들간의 문

서 교환이 가능하다.

- 정보 수집/제공을 위하여 특정 지역에 분산되어 설치된 센서들간의 자유로운 통신이 가능하다.
- 군사적 분쟁 중, 네트워크 구성이 용이하지 않은 낮은 곳에서의 전략 또는 상황 정보를 전송할 수 있다.
- 응급구조 상황이나 재난/구조 작업 지역에서의 네트워크 구성 및 활용이 용이하다.



<그림 3> 애드혹 네트워크 활용¹⁾

나. 특징

애드혹 네트워크는 고정적이고 경직된 구조의 기존 네트워크와는 달리 유동적이고 자유로운 형태를 취한다. 네트워크의 형상, 범위가 시시각각 변화하며,

애드혹 네트워크의 특징은 다음과 같이 5가지로 정리할 수 있다.

(1) 구축이 용이하다 (Easy to Construction)

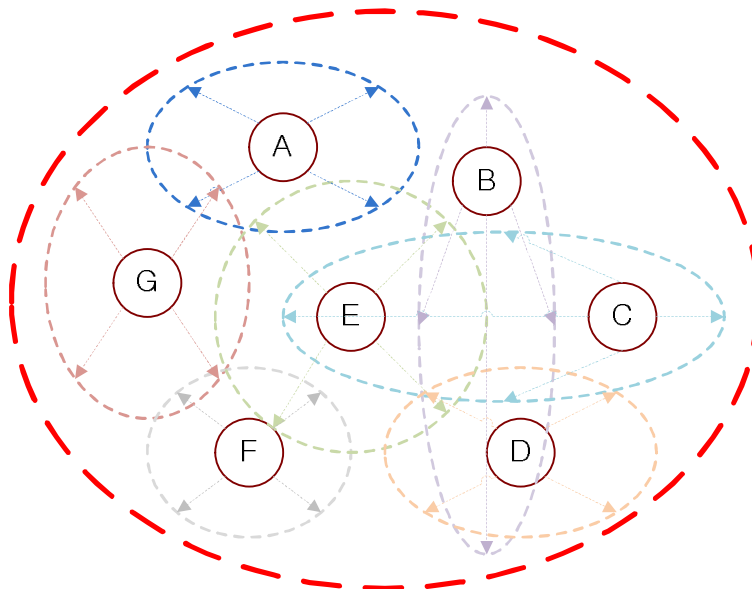
기존의 네트워크는 기반시설 구축, 주소 분배, 라우팅 설정 등과 같은 사전 작업을 필요로 한다. 애드혹 네트워크는 통신 노드들이 IP 주소를 관리하고, 라우팅 정보를 유지하고 있기 때문에, 이러한 사전 작업 없이도

1) <http://home.postech.ac.kr/~alpha44/cs499/intro.html>
http://www.ece.osu.edu/~ekici/res_wmsn.html

즉석에서 네트워크 구성이 가능하다. 또한 서버, 라우터, 브리지, 케이블 등과 같은 기반 시설을 필요로 하지 않기 때문에 네트워크 구축에 필요한 비용이 적다.

2) 향상된 이동성 (Advanced Mobility)

기존의 무선통신망에서도 노드들의 이동성을 보장한다. 하지만 외부 통신망과의 중계를 담당하는 서버의 전파 범위를 벗어난다면 통신이 불가능하다. 애드혹 네트워크에서는 참여 하는 노드들끼리 통신하므로 네트워크의 가장 외각에 위치하는 노드의 통신 범위에만 벗어나지 않는다면 얼마든지 이동이 가능하다. 네트워크에 참여하는 노드의 위치에 따라 통신 범위는 무한정 늘어날 수 있다.

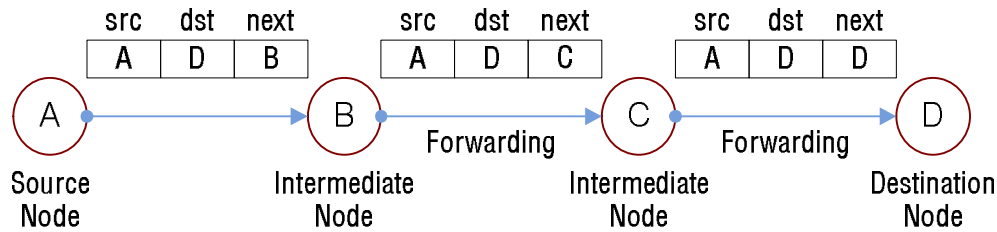


<그림 4> 애드혹 네트워크 범위

3) 멀티홉 라우팅 (Multi-hop Routing)

애드혹 네트워크에서는 중간 노드(Intermediate Node)들의 전달 (Forwarding)과 중계(Relay)를 통하여 데이터가 전송된다. 각 노드들의 전파 전송 범위가 정해져 있기 때문에 네트워크 범위가 정해져 있지 않은

애드혹 네트워크에서는 직접적인 데이터 전송이 불가능하다. 통신 노드들은 각자의 라우팅 정보를 유지하고 있어, 자신이 데이터의 최종 목적지가 아닌 경우, 목적지로 가기 위한 다음 홉으로 전달한다.



<그림 5> 멀티홉 라우팅

4) 적응형 네트워크 토폴로지 (Adaptive Network Topology)

통신 노드들의 시스템적인 상황, 이용자 및 주변 환경에 따라 네트워크 형상이 변화한다. 네트워크에 참여하고 있던 노드의 이동과 전원 상태, 통신 지역 등이 주요 변수로 작용한다. 애드혹 네트워크에서는 이러한 토폴로지 변화에 대응하기 위해 기존의 네트워크와는 다른 라우팅 및 네트워크 관리 기법들이 사용된다.

2. 라우팅 프로토콜

가. DSDV

1) 개요

대표적인 Proactive형 라우팅 프로토콜로써, 네트워크 상의 모든 노드가 자신이 아닌 다른 모든 노드들에 대한 경로 정보를 가지고 있다. 경로 정보들은 각 노드에 라우팅 테이블로써 유지되기 때문에, Table-Driven 방식이라고도 불린다. 노드들끼리 주기적인 제어 정보를 교환함으로써 라우팅 테이블을 갱신하고, 정보의 유효성을 판단하기 위해 목적지 일련 번호

(Destination Sequence Number)를 사용한다.²⁾

2) 특징

[1] 컨트롤 메시지의 주기적인 교환을 통해서 라우팅 테이블을 유지한다.

테이블 정보를 갱신하는 제어 정보는 발신 주기, 정보의 양에 따라 Incremental/Full Dump 두 가지 방식으로 나뉜다. Incremental Dump는 현재 유지하고 있는 라우팅 정보 중 오래된 정보들로만 컨트롤 메시지를 구성하여 이웃 노드에 알리는 방식이며, Full Dump는 모든 라우팅 정보를 전송하는 방식이다. 지속적으로 많은 양의 제어 정보가 교환되므로 네트워크 혼잡을 발생시킬 우려가 있다. 두 가지 갱신 방식의 비율을 효율적으로 조절함으로써, 네트워크 처리량을 향상시킬 수 있다.

[2] 모든 노드들은 전체 네트워크의 완성된 토폴로지 정보를 저장하고 있다.

DSDV 네트워크 상의 모든 노드들은 자신 이외의 다른 모든 노드들에 이르는 경로를 라우팅 테이블에 저장하고 있다. 그러므로 각 노드들은 전체 네트워크 상의 노드 연결 정보를 가지고 있으며, 이 정보들은 네트워크 크기에 비례한다.

[3] 데이터 송신 시, 추가적인 정보 교환없이 임의의 목적지에 이르는 경로를 즉시 찾을 수 있다.

정보 요청에 기반하여 경로 탐색을 수행하는 On-Demand 방식과는 달리, DSDV와 같은 Table-Driven 방식에서는 이미 모든 경로 정보가 테이블

2) Charles E. Perkins, Pravin Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing(DSDV) for Mobile Computers", London, United Kingdom, pp234-244

블로 유지되어 있으므로 추가적인 과정 없이 데이터 전송이 가능하다.

[4] 라우팅 루프 문제를 해결하기 위해 목적지 일련 번호를 사용한다.

주기적인 제어 정보 전송은 브로드캐스트 방식으로 이루어지므로, 직전에 보낸 제어 정보와 혼동을 일으킬 수 있다. 따라서, 각 노드들은 제어 정보에 목적지 일련 번호를 추가하고, 전송할 때마다 1씩 증가시켜 기존의 정보와 구분할 수 있게 한다.

3) 알고리즘

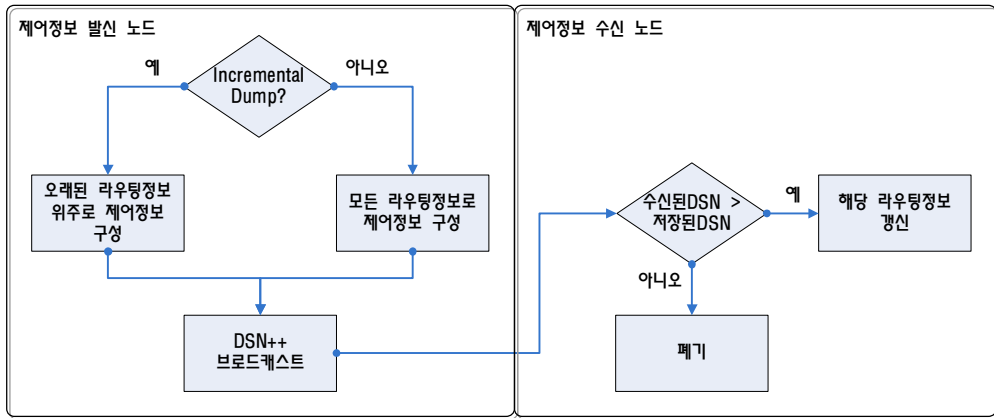
DSDV 프로토콜은 임의의 노드까지의 모든 경로 정보를 테이블로 유지하고 있기 때문에, 추가적인 경로 탐색 과정은 존재하지 않는다. 각 노드의 테이블 정보를 주기적으로 갱신하여 주변 노드의 상황을 반영하는 작업이 수행된다. 각 노드에서 라우팅 테이블을 유지하기 위한 일련의 작업 수행 절차는 아래와 같다.

1) 각 노드들은 주기적으로 자신의 존재를 알리는 컨트롤 메시지를 브로드캐스팅한다. 이 컨트롤 메시지에는 보낼 때 마다 1씩 증가하는 일련번호와 해당 노드가 가지고 있는 라우팅 정보가 포함된다.

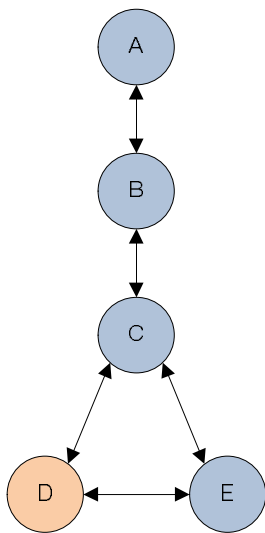
2) 컨트롤 메시지를 수신한 노드들은 자신의 라우팅 테이블에 있는 각 엔트리의 일련번호와 비교한다.

2)-1 수신된 메시지의 일련번호가 자신이 가지고 있는 정보의 일련번호보다 크면, 수신된 메시지의 라우팅 정보를 현재 노드의 라우팅 테이블에 복사한다.

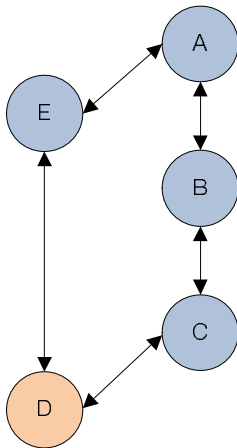
2)-2 반대로 수신된 메시지의 일련번호보다 작다면, 해당 정보를 폐기한다.



<그림 6> DSDV 수행과정



목적지	다음 홉	남은 홉수	일련번호
A	C	3	21
B	C	2	33
C	C	1	15
E	E	1	8



목적지	다음 홉	남은 홉수	일련번호
A	<u>E</u>	<u>2</u>	<u>22</u>
B	C	2	33
C	C	1	15
E	E	1	8

<그림 7> DSDV 알고리즘

위 그림은 주변 노드가 이동하여 라우팅 정보가 일련번호에 의해 갱신되는 과정을 보여주고 있다. 그림 옆 라우팅 테이블은 노드 D에 저장되어 있는 경로 정보를 나타낸다. 노드 E가 원래 위치에서 노드 A로 이동한 경우, 위치 변화에 대한 정보를 담고 있는 컨트롤 정보 교환이 이루어진다. D에서 A로 가는 경로에서 기존의 일련 번호 값이 노드 이동으로 인해 갱신되어 다음 홉이 노드 E로, 목적지까지의 홉 수가 2로 설정된다.

나. DSR

1) 개요

DSR은 전형적인 소스 라우팅 프로토콜로 데이터를 보내고자 하는 송신지에서 목적지까지의 모든 경로가 결정된다. 송신 노드는 목적지까지의 완전한 Hop-by-Hop 경로를 테이블로 유지하고 있으며, 중간 노드들은 데이터 패킷 헤더에 기록된 경로대로 패킷을 다음 노드로 전달한다. 경로 설정 과정은 요구에 따라(On-Demand) 이루어지며, 요청 메시지는 브로드캐스트로 응답 메시지는 유니캐스트 형태로 전송된다.³⁾

2) 특징

[1] 송신자 기반 라우팅 프로토콜

송신자는 어떤 목적지로 가기 위해 거치는 모든 중간 노드의 경로 정보를 저장하고 있다. 이러한 경로 정보는 데이터 패킷 전송 시 송신자 측에 의해 설정되며, 이를 전달하는 중간 노드는 자신의 정보와 다음 홉을 확인

3) D.B johnson, D. A. Maltz, Yih-chun Hu and J. G. Jetcheva, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks(DSR)", Internet Draft, IETF MANET Working Group, pp4-25

하여 목적지까지 전송한다.

[2] 통신의 필요성에 따른 경로 설정 메커니즘

DSR은 요구 기반형(On-Demand) 프로토콜이다. 네트워크 상의 모든 노드가 자신이 아닌 모든 노드에 대하여 경로 정보를 유지하는 테이블 기반(Table-Driven) 형태와는 달리 해당 목적지까지의 경로 정보가 필요할 때 경로 설정 과정을 수행한다. 이는 테이블 기반 프로토콜에서 사용하는 주기적인 경로 정보 교환으로 인한 오버헤드를 줄일 수 있고 각 노드에서 유지하는 라우팅 정보를 간략화할 수 있다는 장점이 있다.

[3] Overhearing을 통한 경로 정보 최적화

DSR은 통신 노드간의 주기적인 정보 교환이 일어나지 않기 때문에, 주변 노드에서 전송한 패킷을 임의적으로 수신하여 정보를 확인하여 현재 자신이 보유하고 있는 경로 정보를 최신의 정보로 갱신할 수 있다. 또한 자신이 전송한 패킷의 전달 과정을 확인함으로써 이웃 노드와의 링크 연결 여부를 판단할 수 있다.

3) 알고리즘

DSR 라우팅 프로토콜은 주로 RREQ(Route Request), RREP(Route Reply), RERR(Route Error) 메시지를 통해 경로 탐색 및 관리 과정을 수행한다. 각 메시지 구조 및 필드별 내용은 다음과 같다. (D.B johnson, 2008 : pp40-42)

0										1										2										3									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1								
Option Type										Opt Data Len										Identification																			
Target Address																																							
Route Address [1]																																							
Route Address [2]																																							
...																																							
Route Address [n]																																							

<그림 8> RREQ 메시지 구조 - DSR

- Option Type

메시지 처리 방법을 나타내며 1바이트로 구성된다. 처리방법은 무시(Ignore), 제거(Remove), 표시(Mark), 폐기(Drop)로 분류된다.

- Opt Data Len

8진수 부호 없는 정수로 Option Type과 Opt Data Len을 제외한 나머지 옵션의 길이를 의미한다. 경로 주소의 개수가 n개라면, $(4 * n) + 6$ 으로 정해진다.

- Identification

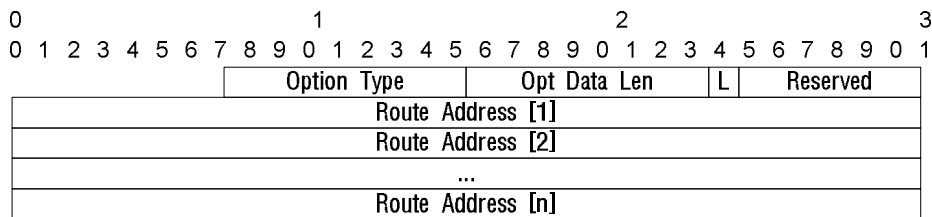
최초 RREQ 발신지에서 정해지는 식별자이다. RREQ 수신지에서는 이 필드를 참조하여 중복 수신 여부를 판단할 수 있다.

- Target Address

RREQ 메시지의 최종 목적지이며, 최초 RREQ 발신지에서 정해진다. RREQ메시지를 수신한 최종 목적지는 더 이상 플러딩하지 않고, 뒤에 이어지는 경로 주소를 통해 경로 응답 메시지를 생성한다.

- Route Address [1]-[n]

RREQ 메시지가 최초 발신지에서부터 전송되어져온 노드의 주소를 담고 있는 리스트이다. 주소의 개수는 Opt Data Len 필드를 통해 계산할 수 있다. ($n = (\text{Opt Data Len} - 6) / 4$) RREQ 메시지를 수신한 중간 노드들은 자신의 주소를 리스트에 추가하고 Opt Data Len 필드 값을 32만큼 증가시킨다. (D.B johnson, 2008 : pp42-44)



<그림 9> RREP 메시지 구조 - DSR

- Option Type, Opt Data Len

RREQ 메시지에서의 값과 동일하다.

- L (Last Hop External)

경로 주소에 포함된 마지막 홉이 현재 DSR 네트워크의 외부와 연결되어 있는지를 나타낸다.

- Reserved

예약된 필드로써 0으로 세팅되거나 무시된다.

- Route Address [1]-[n]

발신지로 전송되는 경로 주소 리스트이다. 1부터 n까지 전송 순서대로 주소들이 배치되며, Opt Data Len 값으로 전송 홉 수를 계산할 수 있다.

(D.B johnson, 2008 : pp44-46)

0	1	2	3
0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9	0 1
Option Type	Opt Data Len	Error Type	Reserved
Error Source Address			
Error Destination Address			
Type-Specific Information			

<그림 10> RERR 메시지 구조 - DSR

- Option Type, Opt Data Len, Reserved

RREQ, RREP 메시지와 동일하다.

- Error Type

발생한 에러의 종류를 나타낸다. 에러 코드는 다음과 같다.

1 = NODE_UNREACHABLE

2 = FLOW_STATE_NOT_SUPPORTED

3 = OPTION_NOT_SUPPORTED

- Error Source Address

최초로 링크 손실을 감지하고 RERR 메시지를 전송한 노드의 주소를 의미한다.

- Error Destination Address

RERR 메시지의 최종 목적지를 나타낸다.

- Type Specific Information

앞서 나온 Error Type 값에 따라 구체적인 정보가 저장된다. 이 필드에 저장되는 정보는 다음과 같다.

Error Type(1) = 도달할 수 없는 노드 주소(Unreachable Node Address, 32bit)

Error Type(2) = Empty

Error Type(3) = 지원하지 않는 에러 타입

경로 탐색 과정은 송신자에 의해 생성된 RREQ 메시지로 시작된다. RREQ에는 송신자의 식별자와 지역적으로 RREQ를 구분할 수 있는 식별자가 포함된다. 또한 RREQ는 송신지에서 목적지까지 전송되면서 지나간 중간 노드들의 경로를 기록하는 라우트 레코드(Route Record)를 가지고 있다. 각 노드들은 자신을 지나간 RREQ로부터 경로를 받아 자신의 라우트 캐쉬(Route Cache)에 해당 목적지까지의 전체 경로를 저장한다. 그리고 각 노드는 이미 수신된 RREQ를 확인하기 위해서 <송신자 ID, RREQ ID> 식별자 리스트 정보를 유지하고 있다. 구체적인 경로 설정 과정은 아래와 같다.

1) 임의의 목적지로의 경로 정보가 필요한 송신지는 자신의 IP주소와 요청 메시지의 식별자를 포함한 RREQ를 주변 노드에 브로드캐스트한다.

2) RREQ를 수신한 송신자가 아닌 중간 노드들은 다음과 같은 처리 과

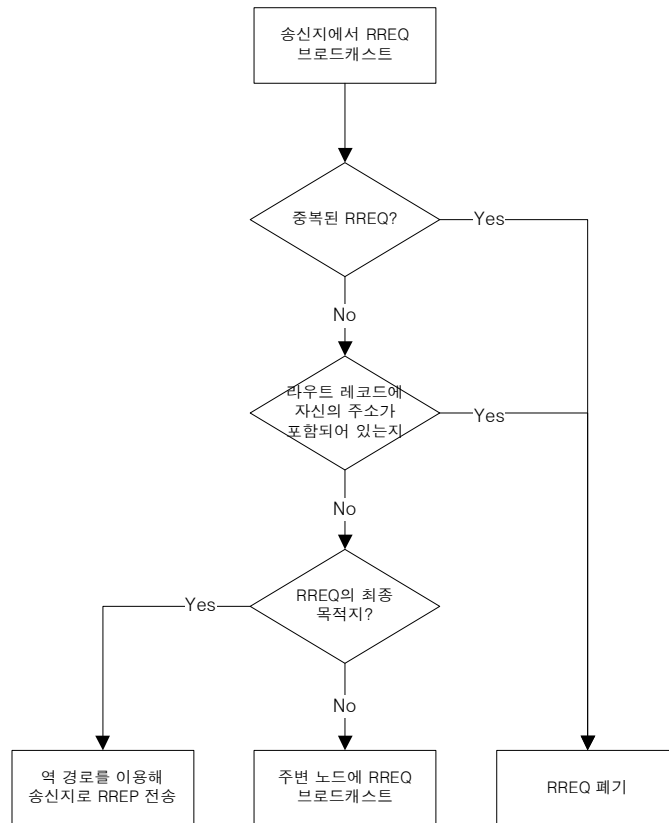
정을 수행한다.

2-1) 현재 수신된 RREQ의 식별자가 <송신자 ID, RREQ ID> 정보에 존재하는지 확인한다. 존재하는 정보일 경우, 중복된 RREQ이므로 폐기한다.

2-2) 식별자 리스트 정보에서 발견되지 않았을 경우, RREQ에 포함된 라우트 레코드 정보에서 자신의 주소가 있는지 확인한다. 자신의 주소가 존재한다면, 이 RREQ는 재수신된 루핑(looping) 메시지이므로 폐기한다.

2-3) 라우트 레코드 정보에 자신의 주소가 없는 경우, 자신이 RREQ의 목적지 노드인지를 확인한다. 만약, 자신이 RREQ의 최종 목적지 노드인 경우에는 RREQ의 라우트 레코드상에 기록된 경로 정보를 통해 경로 응답패킷(RREP)을 생성하여, 역 경로로 송신자에게 유니캐스트한다.

2-4) 자신이 RREQ의 최종 목적지가 아닌 경우, 자신의 주소를 RREQ의 라우트 레코드에 추가하고 주변 노드로 다시 브로드캐스트한다.



<그림 11> DSR 알고리즘 수행과정

DSR의 경로 관리 과정은 링크 손실 감지, 오류 패킷 전송, 라우팅 정보 삭제 순으로 진행된다.

링크 손실은 다음과 같은 세 가지 방법으로 확인할 수 있다. 첫째, 데이터 링크 계층(Data Link Layer)에서 응답신호를 사용한다. DSR 프로토콜 레벨에서 이루어지는 사항이 아닌 하드웨어와 가까운 곳에서 수행되어야 하므로 빠른 속도로 이웃 노드의 존재유무를 파악할 수 있다. 둘째, 패킷 감청(Overhearing)을 통해 링크 손실을 감지할 수 있다. 어떤 데이터 패킷을 다음 홉에 보낸 후, 다음 홉이 그 다음 홉으로 전달하는지를 확인한다. 목적지 주소가 자신이 아닌 패킷을 확인해야 하므로 첫 번째 방법과 마찬가지로 데이터 링크 계층에서의 추가적인 모듈 지원이 필요하다. 마지막으로, 명시적인 응답신호(Explicit Acknowledgement)를 요구하는 방법이 있

다. 응답신호 요구는 데이터 패킷의 R필드를 체크함으로써 이루어지며, 송신자 측에서 패킷을 전송할 때 세팅해야 한다. 이러한 패킷을 받은 수신자 측은 반드시 이에 대한 응답신호를 보내주어야 한다. 만약, 송신노드에서 응답신호를 수신하지 못한다면 송신노드는 해당 노드와의 링크가 손실되었다고 판단한다.

링크 손실을 감지한 노드는 RERR 패킷을 해당 경로의 송신자 측으로 전송한다. 이는 현재 노드까지의 경로의 반대 방향으로 유니캐스트된다. RERR 패킷을 수신한 노드는 자신이 최종 목적지가 아니면 해당 경로를 삭제하고 다음 노드로 전달한다. RERR 패킷이 최종 목적지에 다다르면, 해당 경로에 대한 모든 경로 정보의 삭제가 완료된다.

다. AODV

(1) 개요

AODV(Ad hoc On-Demand Vector)는 대표적인 On-Demand 방식의 라우팅 프로토콜로 DSDV와 DSR의 장점을 혼합한 방법이다. DSDV와는 달리 주변 노드에 대한 정보만을 라우팅 테이블로 유지하여, 제어 메시지로 인한 네트워크 트래픽을 줄인다. 또한 발신지를 중심으로 한 DSR과는 달리 라우팅 정보를 모든 노드에 균등하게 분산시킴으로써, 중형 이상의 대규모 네트워크에서도 효율적인 데이터 전송이 가능하게 한다.⁴⁾

(2) 특징

[1] Hop-by-Hop 라우팅 프로토콜

4) C. E. Perkins, E. M. Royer, and S. R. Das. "Ad hoc On-Demand Distance Vector(AODV) Routing", Internet Draft, IETF MANET Working Group, pp2-4

AODV는 다음 홉과 이전 홉에 대한 경로 정보만을 유지한다. 각 노드들은 어떤 발신지와 목적지간의 전체 경로 상에서, 최종 목적지로 가기위해 자신이 전달해야하는 다음 노드만을 기억하고 있다. 그러므로 라우팅에 대한 주도적인 역할을 하는 노드는 없으며, 모든 노드가 각각의 라우팅 정보를 유지하고 있다.

[2] 출발지 기반 경로 복구 메커니즘

링크 손실이 발생되면 이에 대한 에러 메시지가 발신지로 전송되면서, 경로 선상에 있는 노드들의 경로 정보가 삭제 또는 무효화된다. 또한 발신지는 에러 메시지를 수신하여 경로를 복구하기 위한 경로 요청 메시지를 전송한다. 이는 DSR에서 사용되었던 경로 복구 방법과 유사한 형태를 보인다.

[3] 지역 복구(Local Repair) 메커니즘

링크 손실이 발생한 지점이 전체 경로 선상에서 발신지보다 목적지에 가까운 경우, 지역 복구 기법으로 경로가 복원된다. 이 경우, 발신지에서부터 경로 탐색을 하게되면 제어 메시지로 인한 오버헤드가 심각해지기 때문이다. 지역 복구 기법은 링크 손실이 발생한 지점의 상위 노드가 최종 목적지 노드를 향해 경로 요청 메시지를 브로드캐스트 하는 방법으로 수행된다.

(3) 알고리즘

AODV 프로토콜에서는 RREQ, RREP, RERR 패킷이 주로 쓰인다. RREQ 패킷은 경로 요청 패킷으로써, 경로 탐색을 시작하는 출발지와 최종 목적지가 기록되어 출발지 노드에서부터 플러딩을 통해 이웃 노드로 전달된다. RREP 패킷은 RREQ 패킷에 대한 응답 메시지로써, 경로 설정 확인 정보와 목적지 일련번호 정보 등을 담고 있다. RERR 패킷은 이웃

노드와의 링크 손실(Link Failure)이 발생했을 때 출발지로 전송된다. 손실이 발생된 링크를 사용하는 전체 경로의 출발지로 전송된다. 각 메시지별 구조 및 세부 필드 내용은 아래와 같다. (C. E. Perkins, 2002 : pp7-11)

0	1	2	3
0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9	0 1
Type	J R G D U	Reserved	Hop Count
RREQ ID			
Destination IP Address			
Destination Sequence Number			
Originator IP Address			
Originator Sequence Number			

<그림 12> RREQ 메시지 구조 - AODV

<표 1> RREQ 메시지 필드 - AODV

필드명	내용
Type	RREQ를 의미하는 1값으로 설정
J, R, G, D, U Flags	J : 멀티캐스트를 위한 참여(Join) 플래그 R : 멀티캐스트를 위한 복구(Repair) 플래그 G : 요청에 의하지 않은 RREP의 유니캐스트 전송 여부 D : 목적지에 의한 RREQ의 응답 전송 여부 U : 알려지지 않은 목적지 일련번호 여부
Reserved	0으로 세팅되거나 무시됨
Hop Count	RREQ가 전달되면서 누적된 홉 수

- RREQ ID

최초 발신지에서 정해지는 값으로써, 특정 발신지가 전송한 RREQ의 중복성을 판단할 수 있다.

- Destination IP Address

경로를 얻고자 하는 최종 목적지를 의미한다.

- Destination Sequence Number

해당 목적지까지의 경로 정보의 유효성을 판단할 수 있는 값으로, 경로 정보가 갱신됨에 따라 증가한다. 높은 번호의 정보가 최신의 경로 정보를

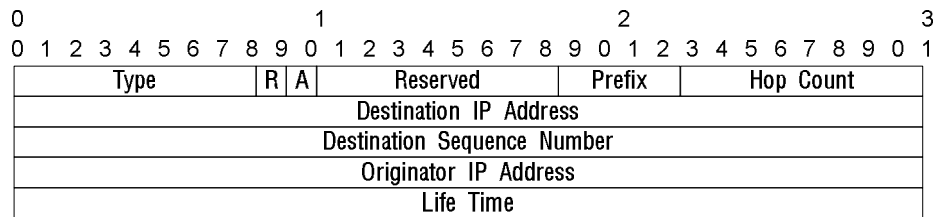
의미한다.

- Originator IP Address

RREQ 메시지를 생성하여 전송한 노드의 주소를 나타낸다.

- Originator Sequence Number

RREQ 발신지에 대한 일련번호를 나타낸다. 발신지-목적지간의 순 경로 외 역 경로를 선정할 때 이 값으로 경로 정보를 판단한다.



<그림 13> RREP 메시지 구조 - AODV

<표 2> RREP 메시지 필드 - AODV

필드명	내용
Type	RREP를 의미하는 2 값으로 설정
R, A	R : 멀티캐스트를 위한 복구(Repair) 플래그
Flags	A : 승인(Acknowledgement) 메시지 요구
Reserved	0으로 세팅되거나 무시됨
Prefix Size	일반적으로 0으로 설정됨
Hop Count	발신지부터 목적지까지의 홉 수

- Destination IP Address

RREP를 생성하여 전송하는 전체 경로의 목적지를 의미한다.

- Destination Sequence Number

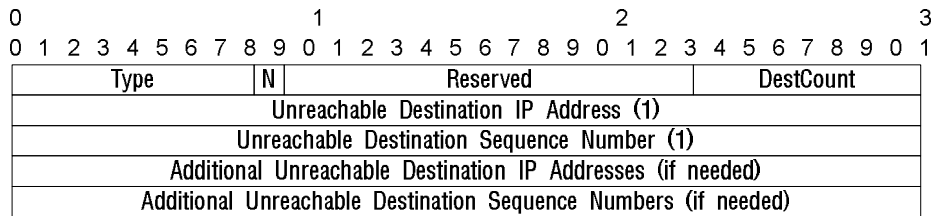
RREQ의 목적지 일련번호 의미와 동일하다.

- Originator IP Address

RREP의 최종 목적지를 의미하며, 이는 RREQ의 최초 발신지와 동일하다.

- Lifetime

경로 정보의 유효시간을 나타내며, 밀리초(milliseconds) 단위로 설정된다. RREP가 중간 노드들을 통해 전달되면서, RREQ가 전송되었던 경로를 이 시간만큼 유효화한다.



<그림 14> RERR 메시지 구조 - AODV

<표 3> RERR 메시지 필드 - AODV

필드명	내용
Type	RERR을 의미하는 3 값으로 설정
N Flag	N : 지역복구 수행에 따른 중간노드에서의 경로 삭제 여부
Reserved	0으로 세팅되거나 무시됨
DestCount	도달할 수 없는 목적지의 수

- Unreachable Destination IP Address

도달할 수 없는 목적지의 IP 주소를 나타낸다. DestCount값이 2이상인 경우, 추가적으로 목적지 주소가 더해질 수 있다.

- Unreachable Destination Sequence Number

링크 손실을 발견한 노드의 라우팅 정보에 기록된 목적지 일련번호를 의미한다. DestCount값에 따라 여러 개의 정보가 이어져 나올 수 있다.

경로 탐색 과정은 어떤 목적지로의 데이터 전송이 필요한 노드에서 RREQ를 생성하여 브로드캐스트하는 것으로 시작된다. RREQ는 브로드캐스트 형태로 전송되기 때문에, 다음 수신지가 불분명한 상태로 발신지에서 플러딩된다. RREQ를 수신한 노드는 바로 직전에 플러딩한 노드를 알 수

있으므로, 탐색하고자 하는 경로의 역 경로를 얻을 수 있다. RREQ를 수신한 최종 목적지는 이 역 경로들을 통해 발신지로 RREP를 전송한다. 세부 수행 과정은 다음과 같다.

- 1) 임의의 노드까지의 경로가 필요한 노드(출발지)는 경로 요청(RREQ) 패킷을 플러딩하고 경로 응답(RREP)를 기다린다.
- 2) 플러딩된 RREQ 패킷을 받은 중간 노드(Intermediate Node)는 RREQ 패킷에 기록된 출발지 정보를 참고하여 역 경로를 생성한다. 만약, 이전에 받았던 패킷과 동일하다면 폐기한다.

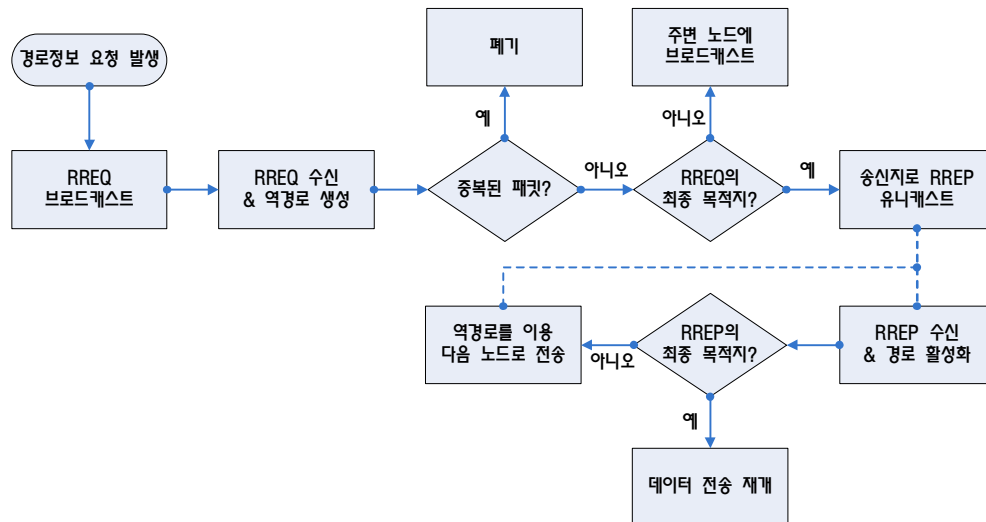
2-1) 현재 RREQ 패킷을 받은 노드가 RREQ에 기록된 최종 목적지와 동일한 노드라면 유효한 경로로 판정하고 역 경로를 이용해 출발지 노드로 RREP 패킷을 유니캐스트한다.

2-2) 최종 목적지가 아닌 경우, RREQ 패킷을 다시 주변 노드에 플러딩한다.

- 3) RREP를 수신한 노드는 바로 직전에 전송한 노드와의 경로를 활성화시키고 자신이 RREP의 최종 목적지인지를 확인한다.

3-1) 현재 RREP를 수신한 노드가 최종 목적지라면, 해당 목적지에 대한 데이터 전송을 시작하고, RREP를 폐기한다.

3-2) 최종 목적지가 아닌 경우, RREQ가 전송되면서 기록된 역 경로를 통해 다음 노드로 RREP를 전송한다.



<그림 15> 경로 탐색 과정 - AODV

경로 복구 과정은 주기적인 Hello 패킷 교환을 통한 링크 손실 감지에서 부터 시작된다. Hello 패킷은 자신과 인접한 노드들에게 자신의 존재를 알리는 메시지로써, TTL(Time To Live) 값을 1로 설정하여 주변 노드로 브로드캐스트된다. 링크 손실을 감지한 노드는 경로 선상의 위치에 따라 지역 복구 또는 발신지 기반 복구 방법을 선택하여 발신지로 RERR 메시지를 전송한다. 세부 수행 과정은 다음과 같다.

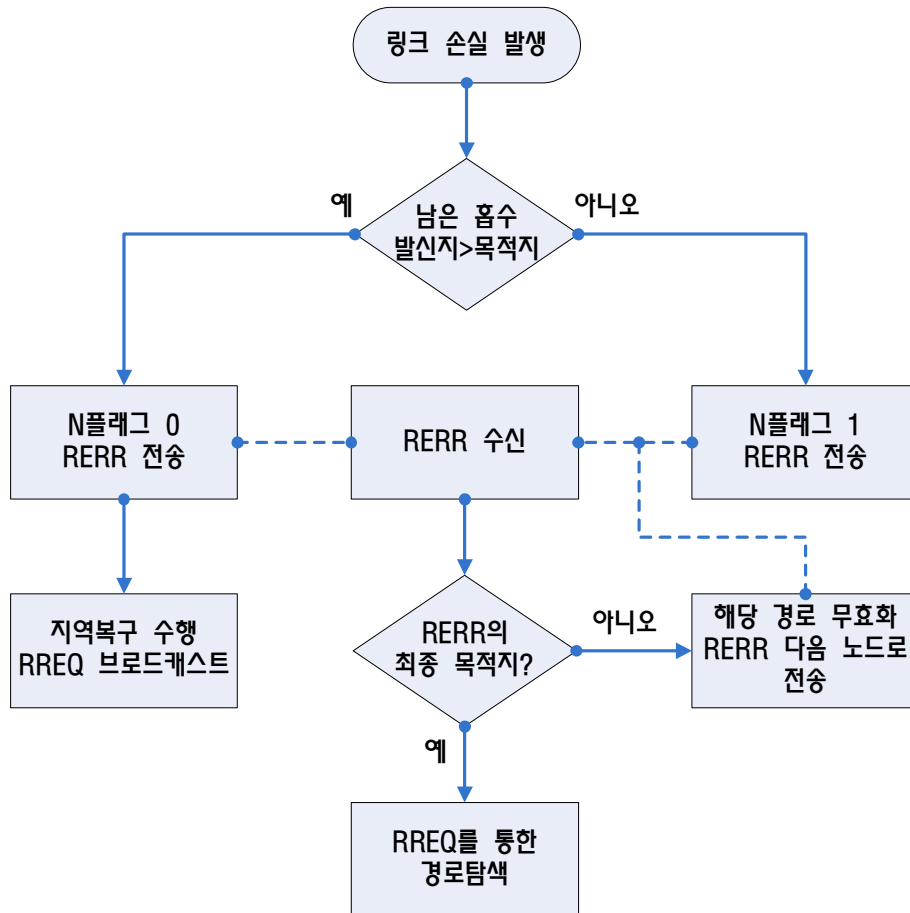
1) 링크 손실을 감지한 노드는 해당 링크를 사용하는 경로의 최종 목적지 까지 남은 홉 수와 발신지까지의 홉 수를 비교한다.

1-1) 목적지까지의 홉 수가 발신지까지의 홉 수보다 작은 경우, RERR 메시지의 N 플래그를 0으로 설정하고, 해당 링크를 사용하는 모든 경로에 대한 출발지로 RERR 메시지를 전송한다. 또한 지역 복구를 수행하기 위해, 최종 목적지를 향해 RREQ 메시지를 브로드캐스트한다. 이 후 과정은 경로 탐색 과정과 동일하다.

1-2) 발신지까지의 홉 수가 목적지까지의 홉 수보다 작은 경우, RERR 메시지의 N 플래그를 1로 설정하고, 해당 링크를 사용하는 모든 경로에 대한 출발지로 RERR 패킷을 전송한다.

2) RERR 패킷을 수신한 출발지 노드는 지속적으로 해당 목적지로의 경로

가 필요한 경우, RREQ 패킷을 플러딩하여 경로 탐색 절차를 수행한다.



<그림 16> 경로 복구 과정 - AODV

3. 관련 연구

애드혹 네트워크의 AODV 라우팅 프로토콜에 대한 많은 연구가 진행되고 있다. 그 중 링크 손실로 발생하는 문제점들을 해결하기 위한 여러 기법들이 제시되고 있다. 최근 국내/외로 진행되고 있는 여러 연구들은 크게 3가지 분야로 분류할 수 있다. 기본 AODV 프로토콜에서 제시하고 있는 지역적 복구 기법의 개선 방법과 다중 경로 탐색을 통한 우회로 선정, 링크 손실을 사전에 감지하여 대응하는 기법이다.

지역 복구 기법 개선 연구는 기본적인 지역 복구 알고리즘을 기반으로 기존 경로의 재사용, 플러딩 제한, 재탐색 위치 변경 등과 같은 기법을 적용하였다.

“서현곤(2004), AFLRS : 애드 혹 네트워크에서 AODV에 기반한 빠른 경로 복구 기법”에서는 기존의 AODV에서 사용하던 지역 복구 기법에서의 목적지 일련 번호 확인과 추가적으로 목적지 노드까지의 홉 수를 비교하여 링크 손실이 발생되기 전의 경로를 최대한 활용하는 경로 복구 기법을 제안하였다. 손실이 발생한 링크와 연결되어 있는 노드 중 목적지와 가까운 쪽을 단절 하위노드, 목적지와 먼 쪽 노드를 단절 상위노드로 정의하고 이들간의 경로 탐색을 수행한다. 종전의 지역 복구 과정에서 수행했던 목적지 일련번호 검사와 목적지 노드까지의 홉 수를 확인함으로써 경로 요청 메시지가 상위 노드쪽으로 전달되어 응답 메시지 루프가 발생하는 현상을 방지하였다.

“안수길(2005), 효율적인 지역복구를 위한 AODV 라우팅 프로토콜의 개선”에서는 지역적 복구 과정에 있어 RREQ 패킷의 플러딩을 제한하는 범위 제한적 지역 복구 기법을 제안한다. 단순히 현재 가지고 있는 경로의 목적지 일련 번호가 RREQ에 담긴 목적지 일련 번호보다 작다고 하여 폐기하는 것이 아니라 최종 목적지까지의 홉수를 비교하여 RREP 패킷을 생성하게 한다. 경로 복구 과정에서 RREQ 패킷의 플러딩 횟수를 감소시키고 목적지까지의 홉수가 적은 경로를 사용함으로써 제어 패킷 오버헤드와 데이터 패킷 전송률을 향상시킨다.

다중경로 탐색 기법 연구는 기본적인 AODV의 단일 경로 탐색에 대한 안정성을 문제점으로 제시하고, 경로 손실 시 차선택을 선택함으로써 데이터 전송률을 향상시키고자 했다. 이 분야의 많은 연구들에서는 다중 경로에서 가지는 노드/링크의 중복으로 인한 링크의 중복성을 해결할 수 있는 여러 기법들을 제시하였다.

“김동학(2007), AODV기반의 지역경로탐색을 이용한 노드 비중첩 다중 경로 탐색 기법”에서는 기존의 다중 경로 탐색 기법을 개선하여 여러 경

로들간의 중첩된 경로를 제거한 완전 비중첩 경로 탐색 기법을 제안하였다. 고전적인 AODV에서 사용하던 경로 요청 메시지 플러딩 과정을 통해 송신지-중간노드-수신지 간의 역경로를 구축하고 수신지에서 발생된 RREP 메시지들에 있는 경로가 서로 중첩되는 경우, 중간 노드에서 지역적 재탐색 기법을 사용하여 수신지로의 경로가 수립되지 않은 노드로 차선 경로를 구성한다. 경로 요청 메시지가 브로드캐스트 형식으로 전달됨으로써 발생된 제한적인 역 경로 정보를 중간 노드에서 재탐색을 수행함으로써 일종의 정보 보정 과정을 통해 노드 비중첩 경로를 구성한다.

링크 손실의 사전 감지를 통한 경로 관리 연구에는 이동 단말기의 전원 상태나 신호 세기(Signal Strength)를 이용한 여러 논문들이 제시되었다. 측정 수치가 경계값(Threshold) 이하로 떨어지게 되면, 링크 손실 가능성이 높은 것으로 판단한다. 경로가 손실되기 이전에 우회로를 탐색함으로써, 데이터 지연을 및 전송률을 감소시킬 수 있다.

“Srinath Perur(2002), A Preemptive Route Repair Strategy for AODV”에서는 이웃 노드들의 신호 강도 값을 유지하여 링크 손실이 발생되기 전에 우회로를 찾아 패킷을 전송하는 방법을 제안하였다. 각 노드들은 인접 노드들에 신호 강도 값을 포함한 주기적인 Hello 패킷을 전송함으로써 주변 노드들에 대한 신호 값을 리스트로 유지하고 있다. 어떤 노드로부터의 신호 값이 일정 값 이하로 떨어지면, 손실 위험상태로 간주하고 신호 값이 안정적인 이웃 노드들로 전송 경로를 구성하기 위해 TTL 값을 1로 설정하여 핸드오프 패킷을 브로드캐스트한다.

앞서 살펴본 논문들 외에도 여러 연구보고서를 통해 AODV 경로 손실로 인한 문제점을 해결하기 위한 방안들이 제시되고 있다. (이재석, 2006 : pp830-836), (이은주, 2008 : pp47-54) 하지만 아직까지 실용화에 성공하거나 여러 상황에 대하여 고른 성능 개선을 보이는 연구는 이루어지지 않았다. 라우팅 프로토콜이 네트워크 전체에서 데이터 전송에 대한 관리 법칙을 담고 있다라는 의미를 되새겨 본다면, 일반화된 방법과 이에 대한 엄격한 검증 절차가 뒤따라야 한다. 최근 발표된 관련 논문들을 분석한 결과, 다음과 같은 2가지 문제점을 지적하고자 한다.

첫째, 특정 경우에 대해서만 뛰어난 성능을 보이는 기법들이 제시되고 있다. (안태영, 2003 : pp265-267), (박노열, 2006 : pp 80-83) 프로토콜에 대한 연구는 몇몇 상황에 대처할 수 있는 방법이 아닌 모든 상황에 유연하게 대처할 수 있는 일반화된 규칙이 제시되어야 한다. 연구에서 제시된 상황을 제외한 나머지 경우에 대한 처리 방법들은 누락되거나 생략되어 있는 논문들이 많다.

기본적인 AODV에서 제시하고 있는 지역 복구의 한계점을 벗어나지 못하고 있다. 앞서 살펴본 논문을 포함한 여러 연구들에서는 AODV의 지역 복구 기법을 기반으로 약간의 수정을 가하는 것에 그치고 있다. (Michael Pan, 2005 : pp318-321) (Sang-min Lee, 2006 : pp1119-1120) 발신지보다 목적지에 가까운 곳에서만 지역 복구가 수행된다는 알고리즘 적용 범위를 해결하지 못하고 있다.

III. 제안 알고리즘

1. 경로 복구 절차 문제점

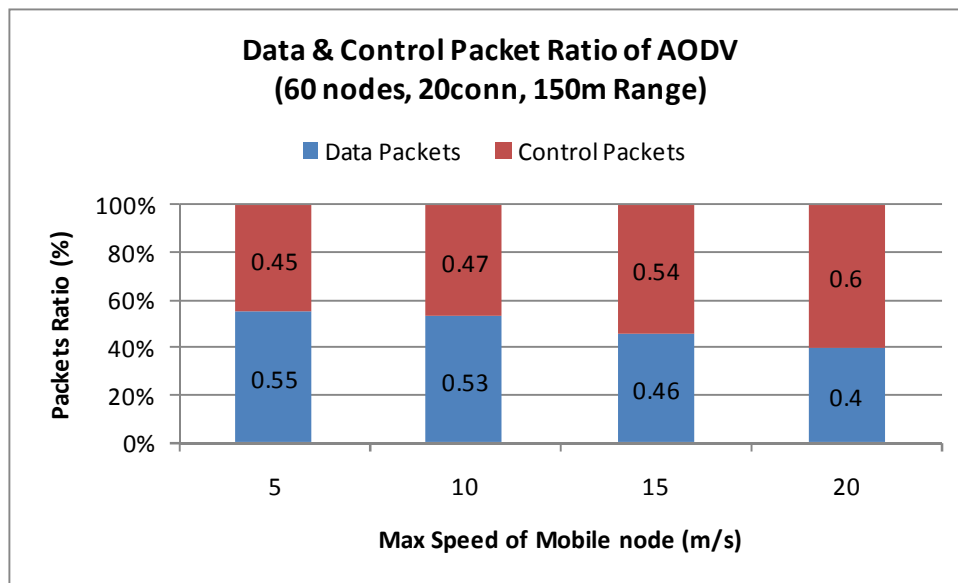
본 연구에서는 기존 애드혹 네트워크 라우팅 프로토콜의 경로 복구 과정에서 발생하는 문제점들을 해결하고자 한다. 기존 연구 분석과 관련된 문헌 고찰을 통해 다음과 같은 두 가지 문제점을 도출하였다.

1) 경로 복구 과정에서 발생하는 제어 패킷 오버헤드

애드혹 네트워크에서 전송되는 패킷의 약 40-50%는 네트워크 유지관리에 이용되는 제어 패킷에 속한다. 이러한 제어 패킷은 이웃 노드의 존재 확인, 경로 요청, 경로 정보 응답, 링크 손실 통보 등과 같은 정보를 담고 있다. 하지만 대부분 브로드캐스트 형태로 전송되기 때문에, 제어 패킷들은 노드를 거치면서 그 수가 기하급수적으로 늘어나게 된다. (박노열, 2006 : pp82-84)

일반적인 무선 네트워크의 MAC 계층에서는 충돌 회피 방법(Collision Avoidance)을 사용한다. 일시적으로 브로드캐스트 패킷의 수가 많아지면, 네트워크 혼잡이 발생하여 제어 패킷이 아닌 실질적인 데이터 패킷의 손실이 발생되게 된다. 특히 노드의 이동이나 전원 상태에 따른 경로 손실 및 토폴로지 변화가 일어난 경우, 데이터 전송 경로를 재설정하기 위한 다수의 제어 패킷들이 생성된다. (Cigdem Sengul, 2000 : pp1-4)

전체적인 네트워크 성능(Performance)과 처리량(Throughput)을 개선시키기 위해서는 네트워크 혼잡의 주요 원인이 되는 제어 패킷의 수를 감소시켜야 한다. 본 논문에서는 효율적인 경로 복구 기법 알고리즘을 제시함으로써, 제어 패킷으로 인한 오버헤드를 줄이고자 한다.

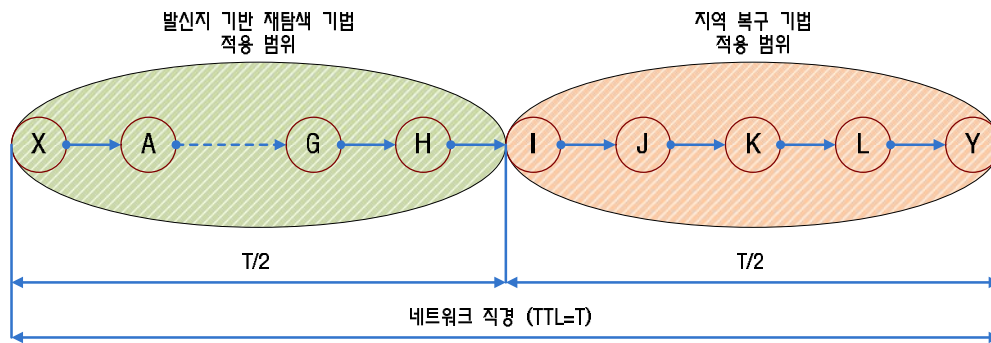


<그림 17> 데이터 및 제어패킷 비율

2) 제한된 지역 복구 기법

기존의 AODV 프로토콜에서는 링크 손실이 발생한 지점이 발신지보다 목적지와 가까운 경우에만 지역 복구가 수행되도록 명시하고 있다. 이는 전체적인 네트워크 측면에서 봤을 때, 절반에 해당하는 영역에서 링크 손실이 발생했을 때 지역 복구가 수행되고, 나머지 영역에서는 발신지 기반 재탐색 과정이 이루어진다고 할 수 있다. 발신지에서부터 경로를 다시 찾는 과정이 수행되면, 경로가 복구될 때 까지 해당 경로 상에서 전송되어야 하는 데이터 패킷이 지연될 뿐만 아니라, 브로드캐스트 패킷 전송으로 인하여 다른 링크에게까지 네트워크 혼잡을 유발시킬 수 있다.

지금까지 제시되었던 AODV의 지역 복구 기법에 관한 개선 연구들은 특정한 경우에만 주목한 방법이거나 기본적인 지역 복구 기법의 적용 범위의 한계점을 벗어나지 못하고 있다. 대부분 탐색 위치, TTL값 조정 등과 같은 약간의 수정 사항만을 반영하고 있어 여러 상황들에 대처해야 하는 알고리즘으로써는 아직 미흡한 점이 많다.



<그림 18> AODV 지역 복구 적용 범위

효율적인 경로 복구와 데이터 전송의 신뢰성을 향상시키기 위해 지역적 복구 기법의 적용 범위를 확대해야 한다. 네트워크 전체 범위에서 적용될 수 있는 일반화된 알고리즘과 이에 대한 검증 절차가 수행되어야 한다. 본 논문에서는 네트워크 전역에서 수행될 수 있는 양방향 지역 복구 기법을 제시함으로써, 이와 같은 문제점을 해결하고자 한다.

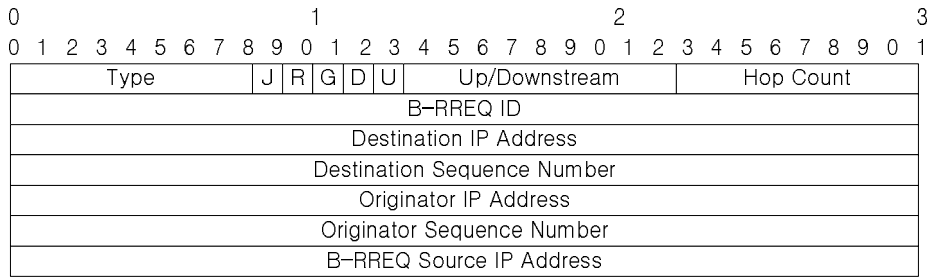
2. TW-AODV

본 연구에서는 기존의 AODV의 지역 복구 기법을 개선한 양방향 경로 탐색 기법(TW-AODV)을 제안한다. 이웃 노드와의 링크 손실을 감지한 노드 중 해당 경로에서 송신지와 가까운 쪽을 상류부 노드(Upstream node)로, 목적지와 가까운 쪽을 하류부 노드(Downstream node)로 정의하여 이 두 노드 사이에서의 플러딩 과정과 경로 추적을 통해 지역 복구를 시도한다.

양방향 경로 탐색과 경로 응답 메시지를 위해 기본적으로 AODV에서 쓰이는 메시지 외에 B-RREQ, B-RREP 2가지 메시지를 사용하였다.

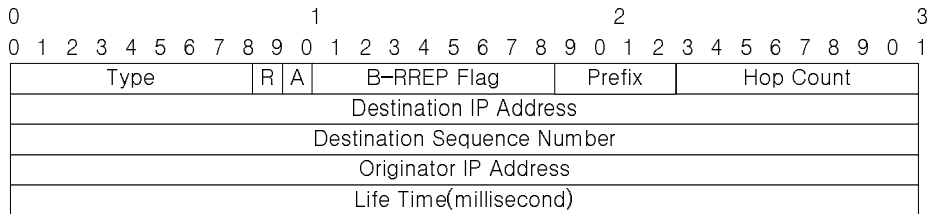
B-RREQ는 경로 손실을 감지한 상류부/하류부 노드에서 전송하는 메시지이다. 일반적인 RREQ 메시지에 발신지 IP 주소(B-RREQ Source IP Address)와 상류부/하류부 노드를 식별하기 위한 필드를 예약 필드에 추가하였다. 발신지 IP 주소는 브로드캐스트로 인한 중복 송신/수신 현상을

방지하는 역할을 한다.



<그림 19> B-RREQ 구조

B-RREP는 각각 상류부/하류부에서 전송한 B-RREQ 메시지를 수신하고 이에 대한 경로 정보를 다시 상류부/하류부로 전달하기 위한 메시지이다. 메시지 구조는 AODV의 RREP 메시지를 기본으로 하며, RREP와 구분하기 위한 식별자 필드를 예약 필드에 지정하였다. B-RREP 메시지는 B-RREQ가 전송되어온 역 경로를 통해 상류부 노드로 전달되고, 전체 경로의 송신지/목적지까지 전송된다. B-RREP 메시지의 구조는 아래 그림과 같다.

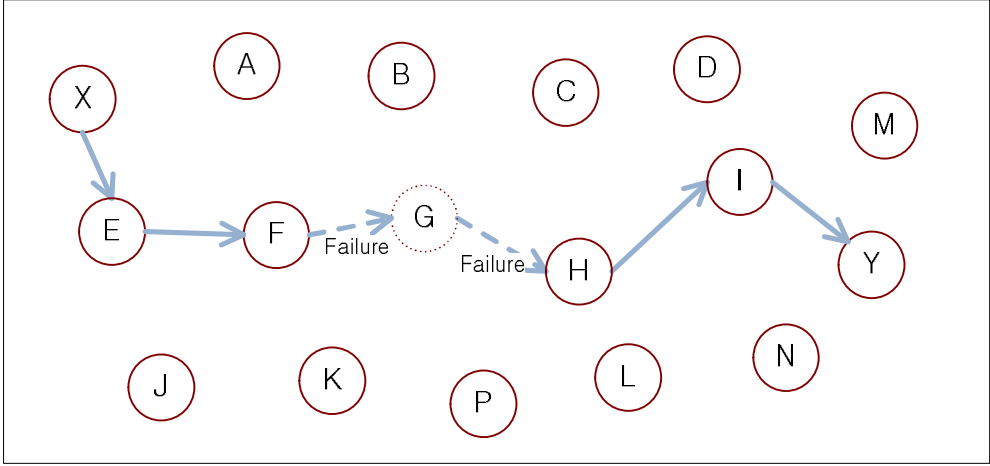


<그림 20> B-RREP 구조

본 연구에서 제안하는 기법은 총 6단계에 걸쳐 수행된다. 6단계는 경로 손실 감지, 에러 메시지 전송, 지역 경로 탐색, 경로 정보 응답, 경로 정보 전달, 전체 경로 복원 단계로 이루어진다.

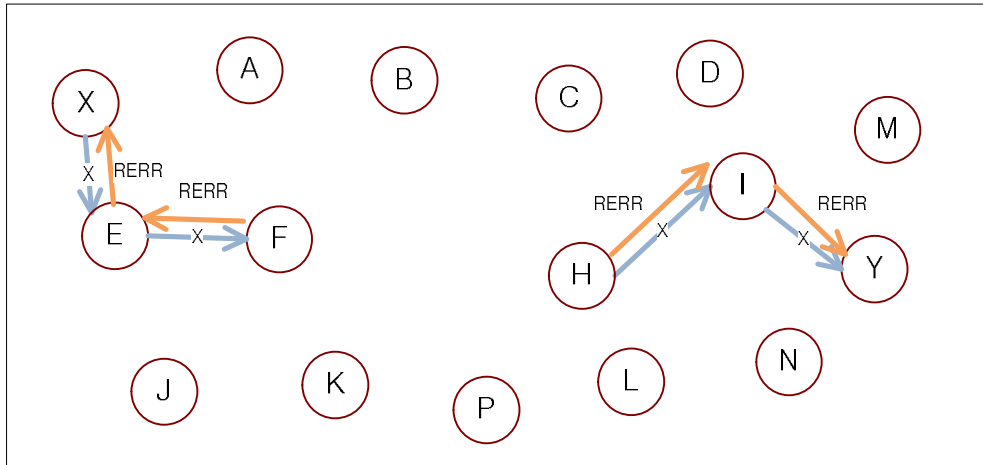
첫 번째 단계인 경로 손실 감지는 주기적인 패킷 교환을 통해 이루어진다. 일정 시간동안 이웃 노드로부터의 Hello 패킷이 수신되지 않을 시 해당 노드와의 링크가 손실되었다고 판단한다. 아래 그림에서 X->Y로의 전

체 경로 중 노드 G가 어떤 이유로 인해 네트워크 범위 내에서 사라진 경우, 노드 G와 연결된 노드 F, H는 링크 손실 사실을 확인할 수 있다. 여기서 손실된 노드 G를 기준으로 송신지 X와 가까운 노드 F를 상류부 노드, 목적지 Y와 가까운 노드 H를 하류부 노드로 정의한다.



<그림 21> 경로 손실 감지

에러 메시지 전송 단계에서는 상류부/하류부 노드에서 각각 송신지/목적지로 RERR 메시지를 전송한다. 이는 경로 복구 중 추가적인 데이터 전송을 막기 위한 방법으로써 RERR 메시지가 송신지/목적지로 전달되면서 상류부/하류부 노드를 제외한 경로 상의 모든 노드들에서 해당 경로에 대한 정보를 무효화한다. 아래 그림에서는 F, H 노드가 각각 X, Y 노드로 RERR 메시지를 전송하여 중간 경로들을 무효화하고 있다.

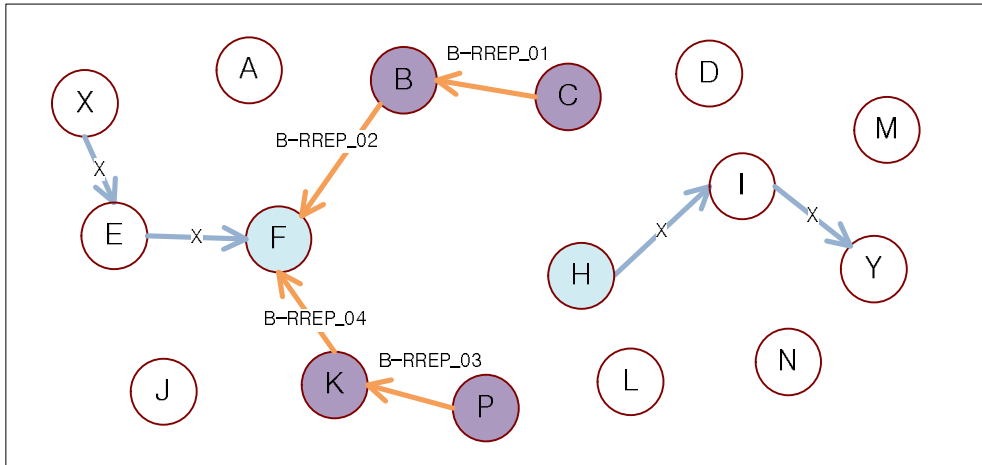


<그림 22> 에러 메시지 전송 단계

지역 경로 탐색에서는 실질적인 경로 설정을 위한 요청 패킷이 전송된다. 양방향 경로 탐색을 위한 B-RREQ 메시지는 상류부와 하류부 노드에서 생성되어 전송된다. B-RREQ 메시지는 TTL 값을 2로 주변 노드에 브로드캐스트 형식으로 전송된다. 이는 패킷의 플러딩 범위를 2홉으로 제한함으로써 1홉간의 경로 손실을 복구하는데 소요되는 브로드캐스트 전송을 최소화시키기 위함이다. 추가적으로 B-RREQ가 전체 경로의 역 경로로 전송되는 경우, 목적지까지의 남은 홉수를 확인하여 메시지를 폐기함으로써 불필요한 메시지 전송을 방지한다.

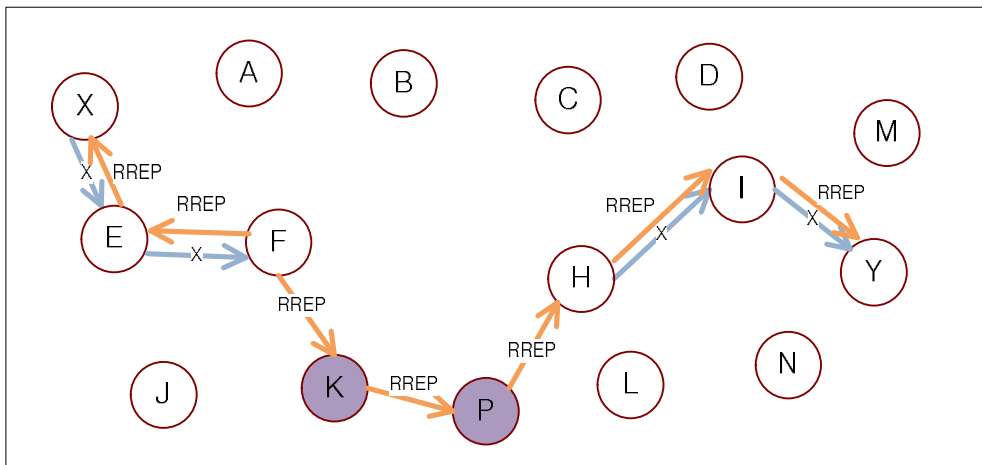
두 노드에서 전송되는 메시지는 전체 경로에 대한 목적지 일련번호는 같지만 B-RREQ 메시지 발신지는 서로 다르다. B-RREQ 메시지를 수신한 노드는 목적지 일련번호와 발신지를 확인하여, 기존의 수신된 메시지와 일련번호와 발신지가 같은 경우에는 중복된 수신이므로 폐기한다. 처음 수신된 메시지이면 버퍼에 저장한 후 B-RREQ 메시지가 전송한 직전 노드를 다음 홉으로 하는 역 방향 경로를 설정하고 다시 브로드캐스트한다. 일련번호는 같지만 발신지가 다른 경우에는, 반대쪽 노드에서 전송한 메시지이므로 경로 탐색이 완료되었다고 판단하고 전체 경로의 송신지, 목적지로 보내기 위한 B-RREP 메시지를 생성한다.

아래 그림에서는 B-RREQ(A)와 B-RREQ(B)가 각각 F, H로부터 2홉



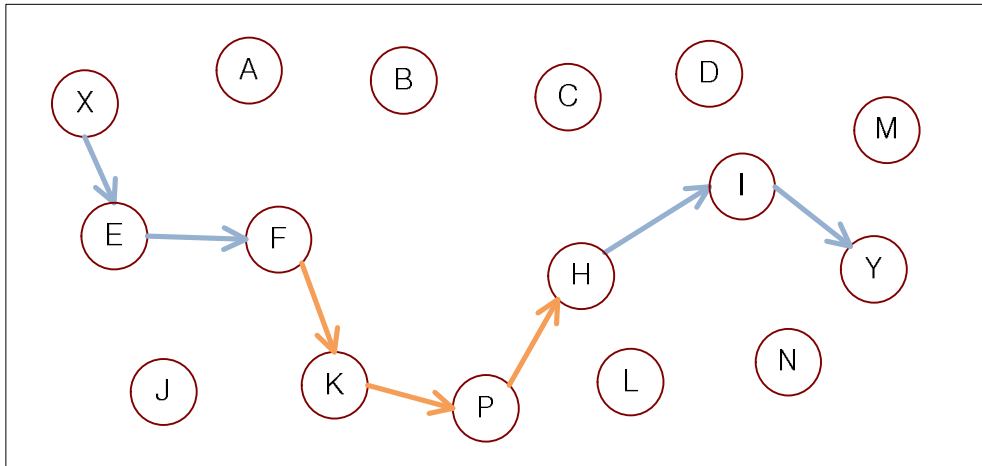
<그림 24> 경로 정보 응답

경로 정보 전달 단계에서는 상류부 노드가 수신한 응답 메시지를 전체 경로 상의 노드로 전파한다. 상류부 노드를 기준으로 송신지, 목적지 두 방향으로 B-RREP를 유니캐스트로 전송한다. B-RREP 메시지를 수신한 중간 노드는 기본적인 AODV에서의 동작과 동일하게 무효화된 경로를 복원시키고 전송되는 경로에 대한 역 경로를 저장한다. 또한 B-RREP의 목적지 일련번호를 증가시켜 새롭게 설정된 경로 정보라는 사실을 경로 상의 모든 노드들에게 알린다.

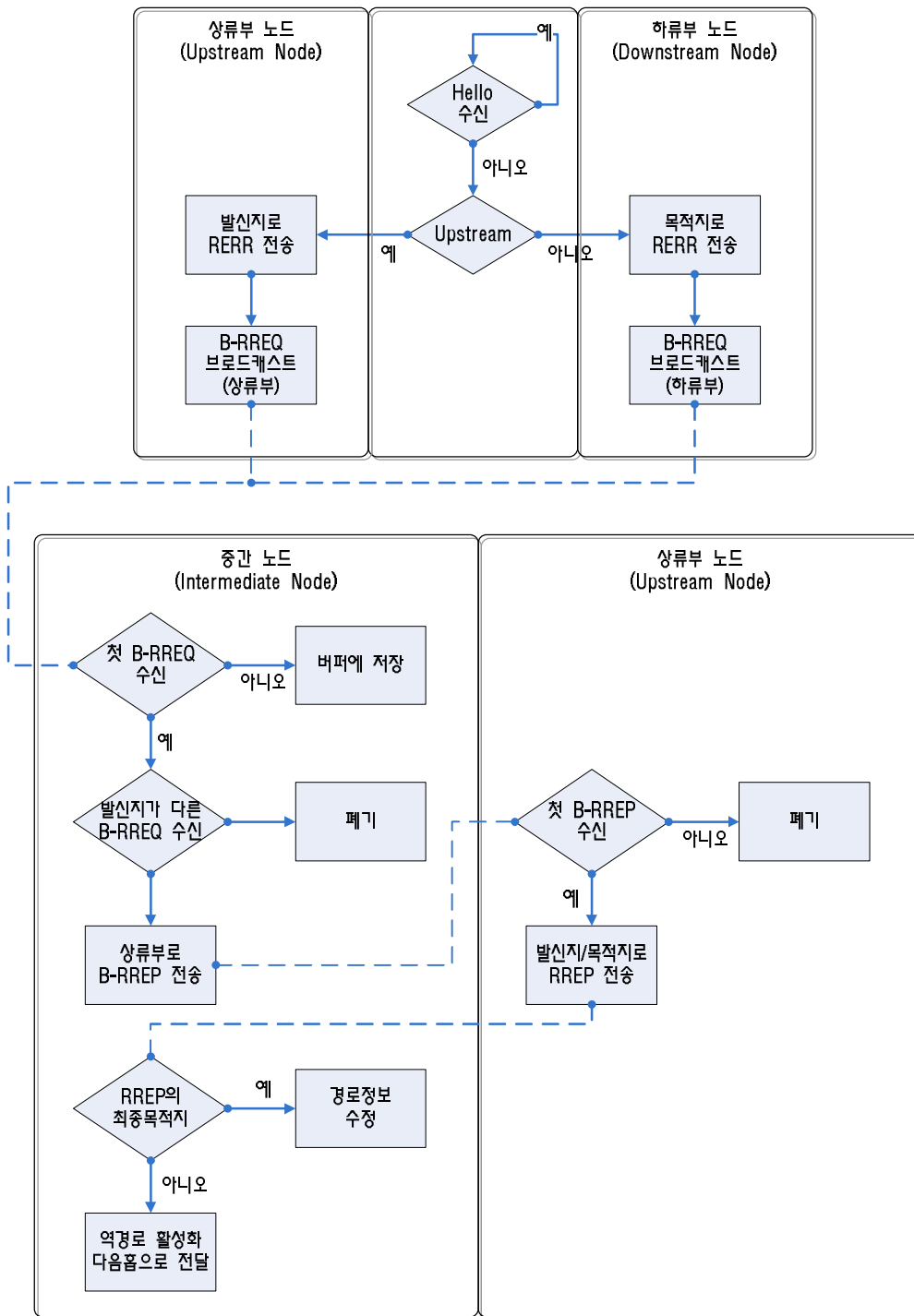


<그림 25> 경로 정보 전달

B-RREP가 각각 송신지와 목적지로 전달되면서 무효화되었던 경로와 B-RREQ로 설정되었던 경로가 유효화되어 전체 경로가 재설정된다. B-RREP 전송이 완료되면 송신지와 목적지간의 데이터 전송이 가능해진다. 아래 그림에서는 지역 복구에 대한 결과로 F->K->P->H 경로가 설정되었다.



<그림 26> 전체 경로 복원



<그림 27> 제안 알고리즘 수행과정

3. 시뮬레이션

가. 실험 환경

(1) NS-2 네트워크 시뮬레이터

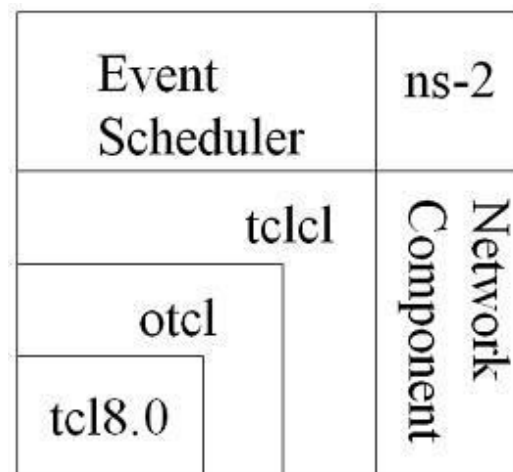
NS2(Network Simulator version 2) 시뮬레이터는 TCP(Transmission Control Protocol), UDP(User Datagram Protocol), FTP(File Transfer Protocol), HTTP(Hyper Text Transfer Protocol) 등과 같은 TCP/IP 프로토콜 패밀리와 라우팅 프로토콜(Routing Protocol), 그리고 멀티캐스팅 프로토콜(Multicast Protocol), RTP(Real Time Protocol) 등과 같은 다양한 인터넷 프로토콜을 시뮬레이션할 수 있다. 그리고 Ad Hoc 네트워크, 이동통신망의 기지국 모델, WLAN(Wireless Local Area Network), Mobile-IP 관련 프로토콜, 위성 네트워크까지 지원할 수 있는 적용 범위가 매우 넓은 네트워크 시뮬레이터이다. (<http://www.isi.edu/nsnam/ns/>)

NS2의 전신인 NS1은 LBNL 이라는 버클리 네트워크 연구 그룹이 발표한 연구 결과로 1995년에 VINT(Virtual InterNetwork Testbed) 프로젝트를 통한 자금 지원으로 완성되었다. 이듬해 1996년 NS1 시뮬레이터의 기능을 더욱 향상시킨 NS2가 발표되었다. NS1 시뮬레이터에서 사용한 시뮬레이션 언어인 TCL(Tool Command Language) 대신 MIT에서 개발하여 발표한 OTCL(Object Tool Command Language)을 사용하여 시뮬레이션 제어가 이루어지며, TCL 또한 지원함으로써 완벽한 이전 버전과의 호환성(Backward Compatability)을 갖추었다. NS2는 지금도 계속해서 기능이 추가되어 소스 코드가 인터넷상에서 소개 되고 있다.

NS2는 유료 시뮬레이터인 OPNET이나 QualNet에 비해 사용자를 위한 GUI가 갖추어져 있지 않고, 수동적인 소스 수정으로 인한 초보자가 접근하기 어려운 단점을 가지고 있다. 하지만 시뮬레이터 전체 소스가 공개되어 있어 누구나 사용할 수 있고 기존의 사용자층이 두터워서 다양하고 많은 자료와 라이브러리를 이용할 수 있다는 장점이 있다. 이러한 장점들 때

문에 지금까지도 학교나 연구소에서 가장 선호하는 네트워크 시뮬레이터로 알려져 있다.

NS2 시뮬레이터는 통신 상의 메시지 교환으로 이루어지는 이벤트(Event)를 제어하는 이벤트 스케줄러(Event Scheduler)와 다양한 네트워크 환경과 프로토콜을 지원하는 네트워크 컴포넌트(Network Component), 사용자로 하여금 시뮬레이션을 구동시키게 하는 TCL 스크립트 언어의 인터페이스를 제공하는 TCL 모듈로 구성되어 있다.



<그림 28> NS-2 시뮬레이터 구조

NS2 시뮬레이터의 구조상 특징은 이벤트 스케줄러와 IP 기반의 네트워크 컴포넌트를 기반으로 하여 이루어진 Event-Driven 방식을 채택한 네트워크 시뮬레이터라는 것이다. 이들은 C/C++ 코드를 기반으로 동작하는데, 이를 이용하여 사용자는 객체 라이브러리로부터 네트워크 객체를 확장하여 새로운 객체를 만들어낼 수 있어 이전에는 없던 새로운 시뮬레이션을 제어할 수 있다.

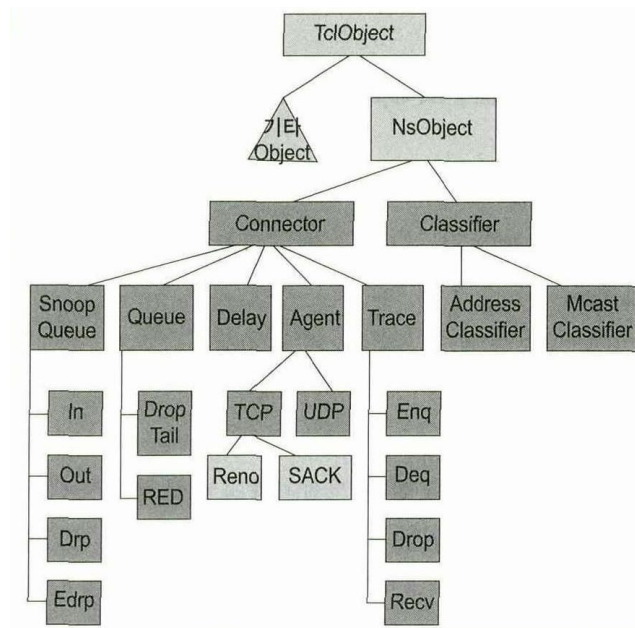
NS2 시뮬레이터의 수행절차는 아래 그림과 같다. 사용자 관점에서 볼 때 NS2는 시뮬레이션 이벤트 스케줄러와 네트워크 구성 모듈 라이브러리를 갖는 OTCL 스크립트 인터프리터이다. 사용자는 정해진 규칙에 의해

OTCL 스크립트 언어를 작성하여 구동시키면 NS2에서 이를 해석하고 내부 시뮬레이션 모듈을 이용하여 시뮬레이션을 수행한 후, 이에 대한 결과를 사용자에게 제공한다. 결과 출력은 텍스트 파일이나 Network Animator(NAM)으로 불리는 그래픽 시뮬레이션 디스플레이로 이루어지며 사용자가 작성한 OTCL 코드에서 출력 방법을 제어할 수 있다.



<그림 29> NS-2 시뮬레이터 작업 절차 개념도

NS2 시뮬레이터는 객체지향에 기반한 스크립트 언어 및 클래스 라이브러리를 제공하여 사용자 기호에 맞는 새로운 시뮬레이션을 구성할 수 있다. 네트워크를 구성하는 노드(Node), 링크(Link), 큐(Queue), 에이전트(Agent) 요소는 네트워크 컴포넌트 모듈에 구현되어 있으며, 이들은 TclObject라는 최상위 클래스로부터 파생된 상속(Hierarchy) 구조를 이루고 있다. NsObject를 상속받는 Connector와 Classifier를 활용하여 사용자가 원하는 필드 및 메소드들을 추가하여 세세한 시뮬레이션 제어가 가능하다.



<그림 30> NS-2 시뮬레이터 클래스 구조

NS2 시뮬레이터는 시뮬레이션 결과를 제공하기 위해 패킷 추적(Trace) 기능과 애니메이션 화면을 통한 NAM 디스플레이 기능을 제공한다. NAM이라는 애니메이션 툴은 스크립트로 작성한 소스 코딩에 대한 시뮬레이션 실행 및 결과 등을 GUI를 통해 디스플레이 장치에 직접 보여주는 기능을 제공한다. 패킷 추적 기능은 짧은 시간 단위로 나누어 그때마다의 패킷의 상태를 리포트함으로써 시뮬레이션의 시작 시점부터 종료 시점까지의 모든 패킷의 흐름을 정확하게 파악할 수 있게 해준다. 또한 자세하고 방대한 데이터를 시각적으로 표현하기 위하여 NS2 시뮬레이터에서는 xgraph를 통한 결과 그래프를 제공하여 성능의 변화 및 차이를 쉽게 알아볼 수 있도록 한다.

(2) 시뮬레이션 환경 및 시나리오

1. 실험목적

시뮬레이션내의 실험 환경 조건을 변화시킴에 따라 기존의 Ad hoc 네

트위크 라우팅 프로토콜과의 성능 비교 및 결과 분석을 통해 제안 알고리즘의 경로 복구 효율성을 판단한다.

2. 실험환경

Unix/Linux 기반의 NS2(Network Simulator version 2)를 사용하며 WLAN 표준 IEEE 802.11 무선 MAC 계층 모델을 통신 노드에 적용한다. 1000m x 1000m 면적에 50개의 노드를 무작위로 배치하고 5m/s 속도의 이동(10초)과 정지(5초)를 반복하여 180초간 시뮬레이션을 수행한다. 노드 간의 전송되는 데이터 패킷은 512byte이며 초당 4개의 패킷을 전송하는 CBR(Constant Bit Rate) 형태의 전송방법을 사용하며 UDP(User Datagram Packet) 프로토콜을 이용하여 전송한다. 시뮬레이션 시작 후 30초 이내에 데이터 전송이 이루어질 10쌍의 노드들이 랜덤하게 결정된다. 각 노드들의 전파 전송범위는 100m로 설정한다.

공간 면적 : 1000m ²
전체 노드 수 : 50
노드 이동 속도 : 5m/s
이동 패턴 : 10초 이동, 5초 정지
데이터 전송 : 4Packets/Sec, 512byte/Packet, UDP
통신 커넥션 수 : 10
전파 전송범위 : 100m

3. 시나리오

시뮬레이션 비교 대상은 기본적인 AODV와 다중경로 설정을 이용한 AOMDV(Ad hoc On-demand Multipath Distance Vector)로 선정한다. 고전적인 AODV에서는 경로 복구 시 local repair 방법과 송신자 기반 경로 탐색 기법을 사용하며 AOMDV는 차선 경로(Alternative Route) 캐싱을 통한 우회로 설정 방법을 사용한다. 기본적인 AODV 프로토콜과 이들의

단점을 개선한 AOMDV 프로토콜과의 비교를 통해 본 연구에서 제안하는 프로토콜의 경로 복구 효율성을 가늠해보고자 한다.

성능평가 척도로는 패킷 전송률(Packet Delivery Ratio), 제어 트래픽 오버헤드(Control Traffic Overhead)를 사용한다. 패킷 전송률은 출발지 노드에서 전송된 패킷과 목적지 노드에 도착한 패킷의 비율로 전송에 성공한 패킷의 비율을 의미한다. 패킷 전송률 값을 통해 경로 손실로 인한 복구 과정의 효율성을 판단할 수 있다. 제어 트래픽 오버헤드는 실질적인 데이터 패킷 전송이 아닌 전송 경로 관리를 위한 제어 패킷 수를 가늠해볼 수 있는 척도이다. 링크 손실 정도와 경로 복구 과정에 따라 제어 트래픽의 양이 결정되므로, 패킷 전송률 값과 함께 네트워크 성능에 미치는 영향을 판단하는데 중요한 자료가 된다.

1) 통신 노드 수의 변화

기본적인 실험 환경을 유지한 채 전체 노드 수를 50개에서 20개씩 90개까지 증가시키면서 위의 평가 척도들을 측정한다. 동일한 면적에서 전체 노드의 수가 증가하면 하나의 노드에 대한 이웃 노드의 수가 증가하게 된다. 이는 어떤 경로에 대해 여러 개의 우회로가 있음을 의미하므로 보다 효율적인 경로 선정 및 다중경로 프로토콜과의 제어 트래픽 정도를 비교해볼 수 있다.

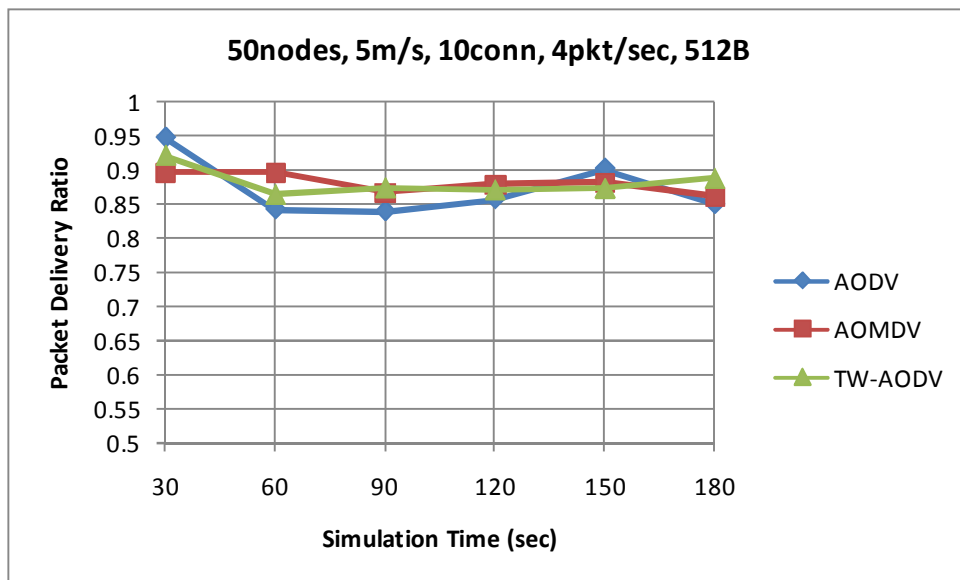
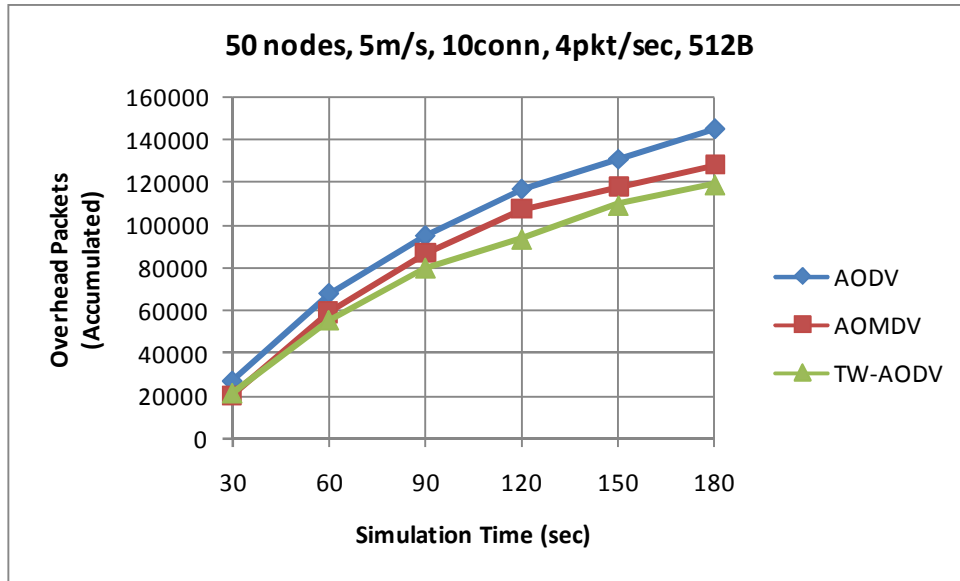
2) 노드 이동속도의 변화

초기 실험 환경 조건은 고정시키고 노드 이동 속도를 5m/s에서 15m/s까지 5m/s씩 증가시키면서 평가 척도를 측정한다. 각 노드들의 이동 속도가 증가하게 되면 노드간의 링크 손실 빈도가 증가하게 된다. 링크 손실시 발생하는 제어 트래픽과 손실 중에 생성되는 데이터 트래픽의 전달률 등을 확인함으로써 기존의 경로 복구 메커니즘과의 성능 평가를 수행해볼 수 있다.

나. 실험 결과

앞서 제시한 실험 환경 및 시나리오에 따라 시뮬레이션을 수행하였다. 패킷 전송률은 (목적지에 도달한 패킷 / 발신지에서 전송한 패킷)으로, 제어 트래픽 오버헤드는 AODV 패킷을 카운트하여 누적된 결과를 산출하였다. 각 실험은 180초간 20회 수행되었으며, 30초 마다 중간 결과 값들의 산술 평균 값을 그래프로 도식하였다.

1) 기본 실험 환경 - 노드 50, 이동속도 5m/s



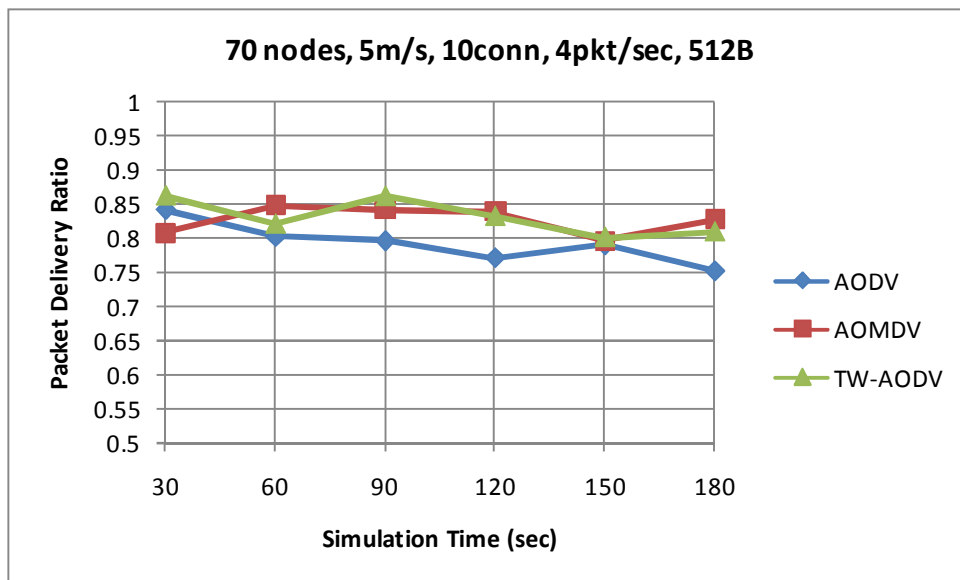
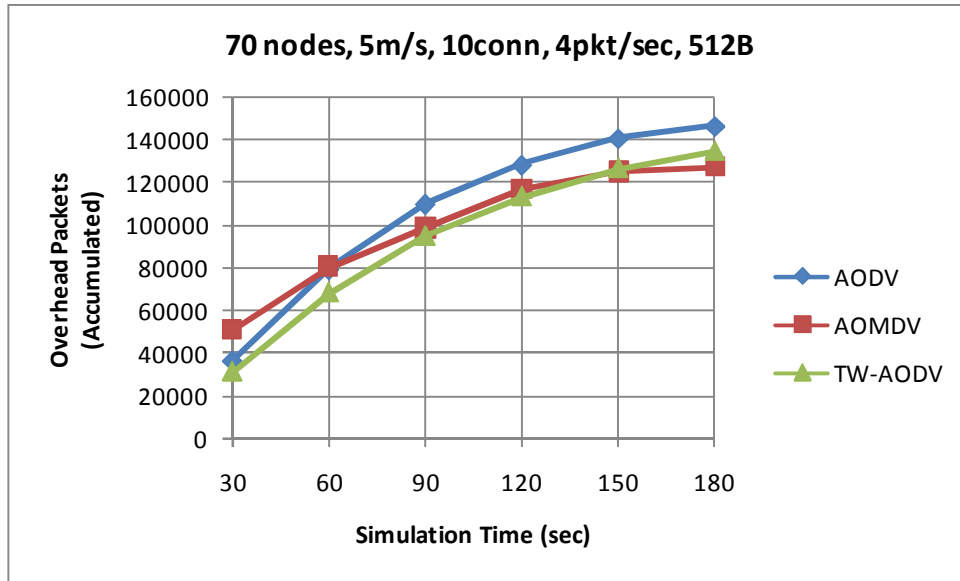
<그림 31> 노드 50, 이동속도 5m/s - 오버헤드, 전송률

기본 실험 환경은 다른 실험 조건들에 비해 링크 손실이 가장 적게 발생한다. (약 200회) 노드들의 이동 속도는 5m/s로써, 일반적인 사람들의 조깅하는 정도의 속도이다. 이 실험 환경에서는 세 가지 프로토콜이 거의 비슷한 성능을 보여주고 있다. 제어 트래픽 그림에 따르면 TW-AODV가

가장 적은 수의 제어 패킷을 생성함을 알 수 있다. 이는 송신자 기반의 재탐색 기법이 아닌 지역 복구 기법을 적용함으로써 경로 복구를 위한 브로드캐스트 패킷의 수가 상당수 감소된 것으로 판단된다. AOMDV의 경우, 우회로 관리에 의한 제어 트래픽은 발생하지만, 실질적인 데이터 전송률에 있어서는 다른 프로토콜과 비슷하거나 다소 우위에 있다.

2) 시나리오 1 - 통신 노드 수 변화 (70, 90)

첫 번째 시나리오에서는 기본 실험 환경에 비해, 전체 노드 수가 증가한다. 한 노드에 인접하는 이웃 노드의 수가 많아지므로, 다중 경로 프로토콜에서의 우회로 선정이 보다 잘 이루어질 수 있다. 또한 단위 공간에서의 노드 밀집도가 높아지므로, 브로드캐스트 패킷이 전송됨에 따라 증가하는 비율이 비약적으로 증가하여, 심각한 네트워크 혼잡을 유발시킬 수 있다.



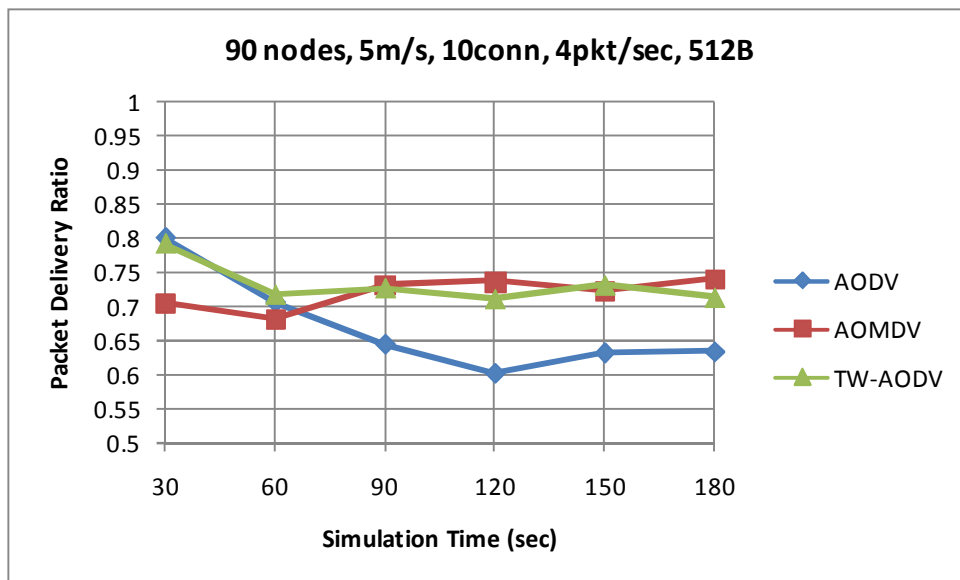
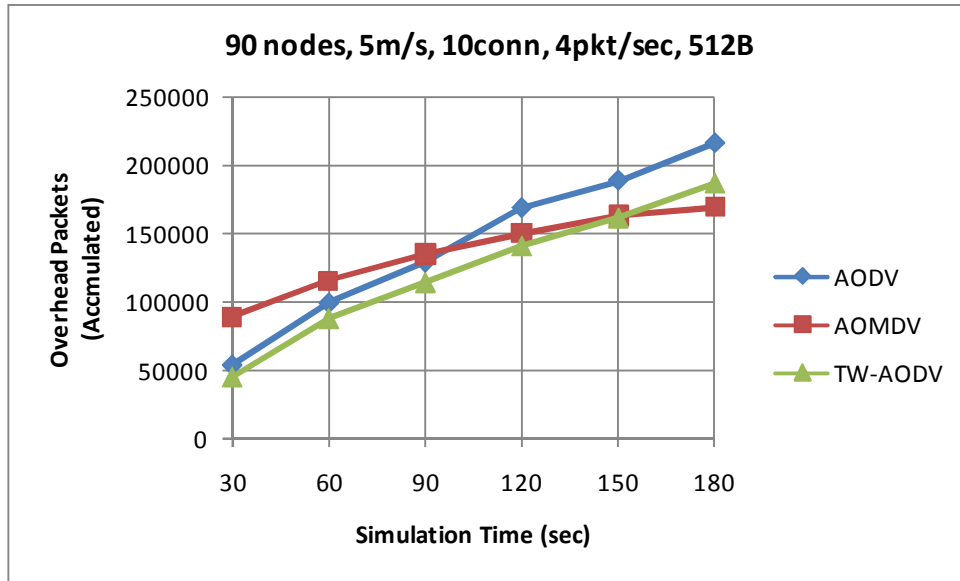
<그림 32> 노드 70, 이동속도 5m/s - 오버헤드, 전송률

전체 노드 수가 증가함에 따라, AODV가 가장 높은 제어 트래픽을 나타내었으며, 전송률에서도 80%에 못 미치는 성능 저하를 보였다. 이전 환경에 비해, 노드 밀집도가 높아 링크 손실 시 발생하는 제어 패킷 플러딩에 의한 오버헤드가 심화된 것으로 분석된다. 또한 오버헤드로 인한 네트워크

혼잡으로 데이터 전송 시 중간 노드에서의 데이터 손실 빈도가 높아진 것으로 판단된다.

AOMDV는 초기 네트워크 안정화 과정에서 우회로 탐색에 따른 오버헤드가 높게 나타났다. 이러한 대체 경로 탐색은 30초 전후에서 완료되는 것으로 판단된다. 경로 구성이 완료된 약 30초 이후로는 다른 프로토콜에 비해 적은 오버헤드와 안정된 전송률을 보이고 있다. 링크 손실이 발생한 경우, 대체 경로를 이용함으로써 경로 복구에 필요한 제어 트래픽과 데이터 손실을 감소시킨 것으로 분석된다.

TW-AODV는 AOMDV와는 달리, 단일 경로 탐색 기법을 이용함에도 불구하고, AOMDV와 비슷한 전송률을 보이고 있다. 시간이 지남에 따라, 전체적인 트래픽 오버헤드는 AOMDV에 비해 상대적으로 높아지지만, AOMDV와 비슷한 80% 이상의 전송률을 보이고 있다. 이는 경로 복구 과정에서 인한 데이터 손실이 적다는 것을 의미하므로, 기존의 경로 복구 방법에 비해 효율적으로 수행되고 있는 것으로 판단된다.



<그림 33> 노드 90, 이동속도 5m/s - 오버헤드, 전송률

전체 노드의 수가 90개로 노드의 밀집도가 가장 높은 환경이다. 앞선 결과에서와 마찬가지로 노드 수에 큰 영향을 받는 AODV가 가장 높은 오버헤드와 낮은 전송률을 보였다. 발신지 기반 재탐색 기법에 의한 경로 요청 패킷의 오버헤드가 높은 것으로 판단되며, 이로 인해 심각한 네트워크 혼

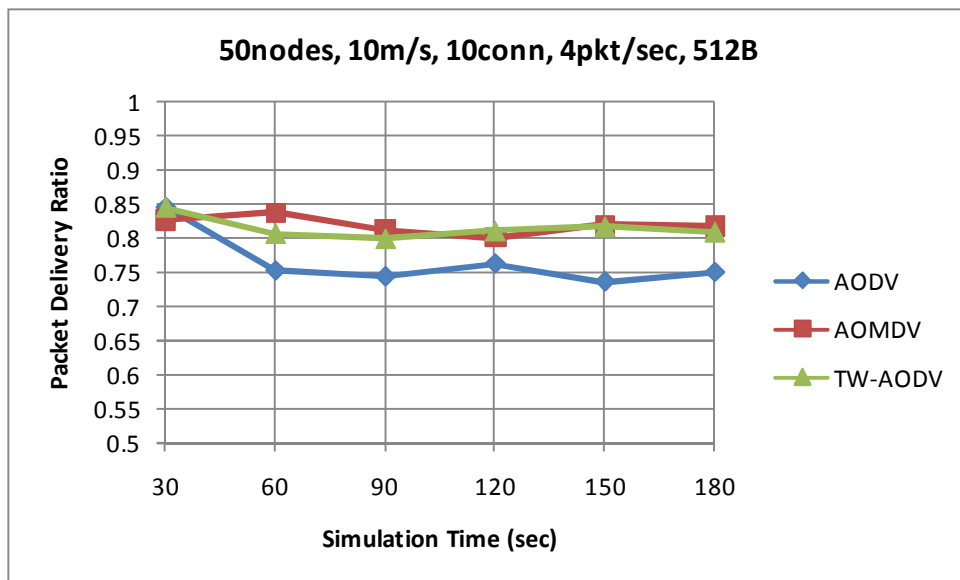
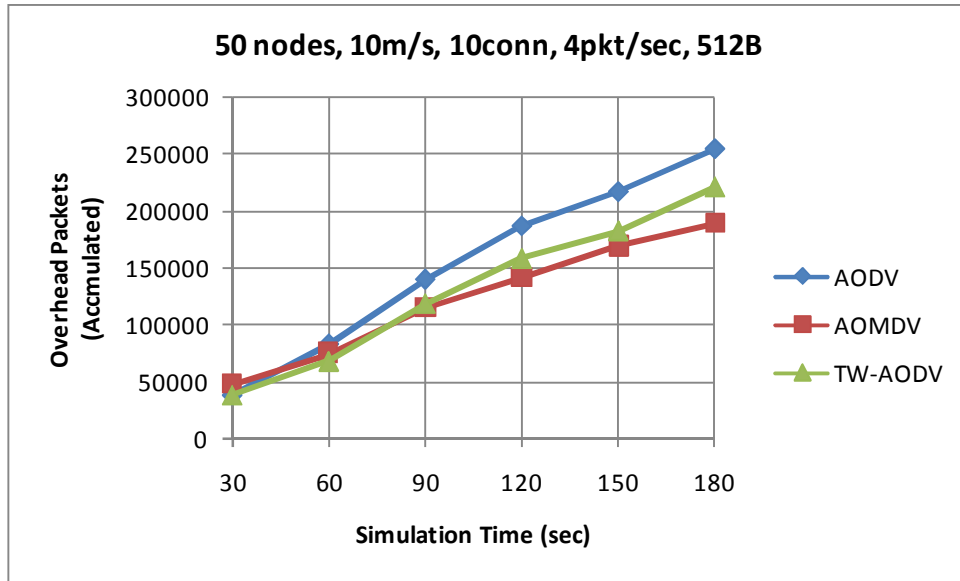
잡과 전송률 저하를 가져온 것으로 추측된다.

AOMDV도 이전 결과와 비슷한 수치를 보였다. 네트워크 형성 초기에는 대체 경로 탐색으로 인한 부하가 다른 프로토콜에 비해 높게 나타났지만, 시간이 지남에 따라 라우팅 경로가 안정화되어 75%에 이르는 전송률을 보였다. 이전 실험과는 다르게 노드 밀집도가 높아져 대체 경로 설정을 위한 시간이 60초 전후로 늦어진 것으로 판단된다.

TW-AODV는 앞선 실험보다 노드 수가 증가되었음에도 불구하고, 안정된 성능을 보여주고 있다. 경로 손실로 인해 제어 트래픽이 AOMDV에 비해 높게 발생하지만, 실질적인 데이터 전송률은 거의 비슷한 수치를 보이고 있다. 경로 복구 과정에서 발생하는 제어 패킷의 플러딩을 제한하고, 지역 복구의 적용 범위를 확대함으로써, 효율적인 경로 복구가 이루어지는 것으로 추측된다.

3) 시나리오 2 - 이동 속도 변화 (10m/s, 15m/s)

두 번째 시나리오는 기본 실험 환경에서 전체 노드의 수는 50개로 고정시키고, 각 노드들의 이동 속도만을 변화시킨다. 노드들의 이동 속도가 증가하면, 전송 경로 상에 있는 노드 간의 통신 반경(100m)을 벗어나는 빈도가 잦아진다. 이로 인해, 링크 손실 빈도가 높아지므로, 많은 경로 복구 작업이 수행된다.



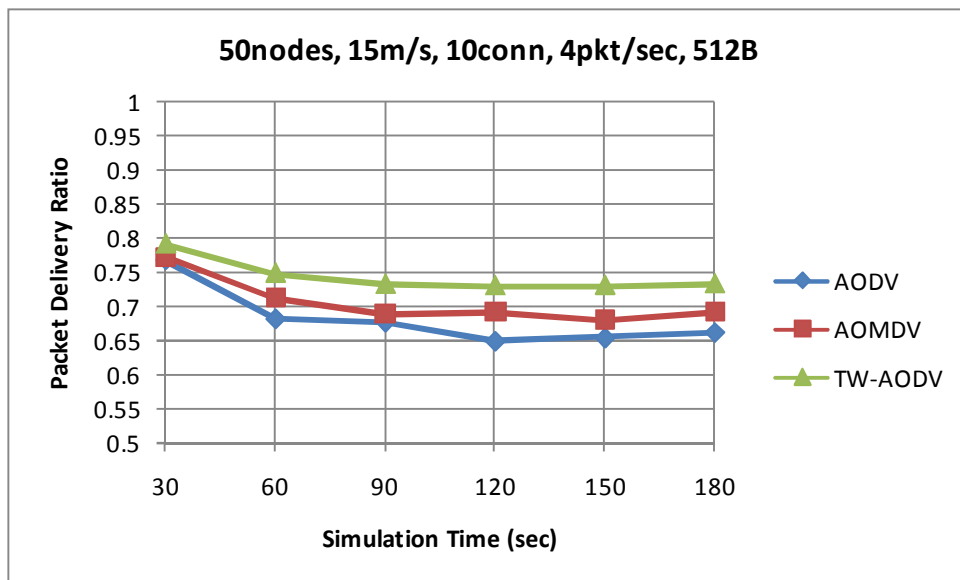
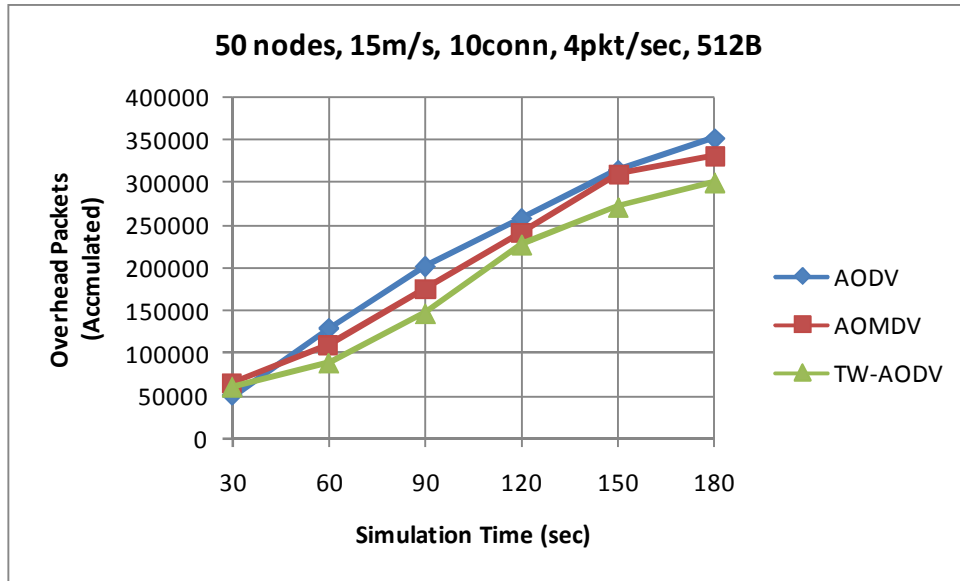
<그림 34> 노드 50, 이동속도 10m/s - 오버헤드, 전송률

노드의 수를 증가시킨 경우와 마찬가지로, 속도 변화에 대해서도 AODV의 성능 저하가 뚜렷이 나타났다. 네트워크 초기 형성 과정인 30초 이후, 링크 손실로 인한 네트워크 혼잡과 이에 따른 데이터 손실률이 증가하고 있다. 경로 복구를 위한 다수의 RREQ 패킷 브로드캐스트 전송이 성능 저

하의 주요 원인으로 판단된다.

AOMDV는 10m/s의 속도로 인한 링크 손실을 우회로를 이용하여 원활히 해결하고 있다. 다른 프로토콜에 비해 경로 복구 과정 수행 회수가 낮기 때문에, 제어 트래픽이 낮게 측정된 것으로 판단된다.

TW-AODV는 AODV와 마찬가지로 단일 경로를 이용하는 프로토콜이므로, 비슷한 회수의 경로 손실과 그에 따른 복구 과정을 수반해야 한다. 하지만 심각한 제어 트래픽을 유발하는 발신지 기반 재탐색 기법을 사용하지 않고, 간단한 지역 복구 기법을 적용함으로써, AODV와는 달리 낮은 제어 트래픽과 우수한 전송률을 보이고 있다.



<그림 35> 노드 50, 이동속도 15m/s - 오버헤드, 전송률

15m/s의 이동 속도에서도 AODV의 성능 저하가 두드러지게 나타났다. 상당수의 링크 손실이 발생함으로써, 심각한 네트워크 혼잡을 유발하고 있는 것으로 분석된다. 또한 약 65%의 전송률을 보여, 빠른 속도로 이동하는 도로 환경에서의 적응력은 저조한 것으로 판단된다.

AOMDV는 10m/s에 비해 월등하게 높은 오버헤드가 발생한 것으로 보인다. 이는 링크 손실에 대비하여 사전에 설정한 우회 경로들조차 손실이 발생되어, 이로 인한 경로 복구 과정이 다수 수행된 것으로 판단된다. AOMDV에서의 경로 복구 과정은 기본적으로 AODV와 동일하므로, 복구 과정에서 발생하는 제어 패킷이 네트워크 혼잡을 유발했다고 추측할 수 있다. 전송률도 이전 결과에 비해 큰 성능 저하를 보이고 있어, 차량이 주행하는 속도 정도로 노드들이 이동하는 네트워크 환경에서는 적합하지 않다고 할 수 있다.

반면, TW-AODV는 다른 프로토콜에 비해, 급격한 노드들의 이동에도 안정된 데이터 전송을 보여주었다. 이전 결과에 비해, 링크 손실 빈도가 증가하여 제어 트래픽이 상당수 늘어났지만, 기존의 AODV와 AOMDV보다는 낮은 오버헤드 비율을 보이고 있다. AOMDV에서 나타났던 우회로 손실에 대한 역효과와는 달리, 노드들이 빠르게 이동하는 환경에서도 제한된 플러딩을 통한 효율적인 지역 복구를 수행함으로써, 링크 손실 문제를 해결한 것으로 판단할 수 있다. 약 75%에 이르는 전송률을 보임으로써, 속도에 대한 적응력이 우수한 것으로 분석된다.

지금까지 노드 수 및 이동 속도 변화에 따른 시뮬레이션 결과를 살펴보았다. 전체 노드 수를 변화시키는 것은 각 프로토콜의 경로 탐색 과정에서 발생하는 네트워크 제어 트래픽 정도와 이로 인한 데이터 전송률을 파악할 수 있다. 또한 노드 밀집도를 높여, 다중 경로 탐색 프로토콜의 우회로 선정 기법과 본 논문의 알고리즘의 성능을 평가해볼 수 있다. 노드의 이동 속도 변화는 링크 손실 빈도를 증가시킴으로써, 경로 복구에 드는 비용을 확인할 수 있다.

본 논문에서 제시한 양방향 경로 탐색을 이용한 경로 복구 기법은 네트워크 상황 변화에 상관없이 안정된 성능을 보였다. 이 기법은 단일 경로를 이용하는 알고리즘이므로, AODV와 비슷한 수의 링크 손실 및 복구 과정을 수행하게 된다. 하지만 제한된 플러딩 기법과 양방향 탐색을 이용하여 지역 복구 범위를 확대시킴으로써, 기존 AODV와는 달리 현저하게 낮은 트래픽과 높은 데이터 전송률을 보였다. AOMDV는 여러 실험 환경에 걸

쳐 고른 성능을 보였으나, 노드들의 이동성이 높은 환경에서는 대체 경로의 안정성에 대한 문제로 오히려 좋지 않은 결과를 보였다. 반면, 논문에서 제시한 기법은 이러한 환경에서도 타 프로토콜에 비해 높은 전송률을 보임으로써, 경로 복구 과정의 효율성을 확인 시켜 주었다.

IV. 결론

최근 정보 기술 발달에 힘입어, 통신 환경에 대한 이용자들의 욕구가 다양해지고 있다. 지금까지 네트워크 환경은 전통적인 유선 네트워크에서 불필요한 선을 없애고 통신 노드의 이동성을 가미한 무선 네트워크로 변모해왔다. 현재 사용되고 있는 무선 네트워크는 통신 중계 서버나 AP 등과 같이 기반 시설에 의존하고 있기 때문에, 통신 범위와 이동성에 있어서 한계점을 드러내고 있다. 이에 반해 애드혹 네트워크는 통신 노드들만으로 네트워크 구성이 가능하므로, 기존의 무선 네트워크와는 달리 기반 시설을 필요로 하지 않는다는 장점이 있다. 또한 언제 어디서나 간단한 설정 작업만으로도 통신이 가능하기 때문에, 유비쿼터스 정보 사회에 적합한 네트워크 환경이라고 할 수 있다.

애드혹 네트워크는 군사 분야를 시작으로, 그 동안 여러 응용 분야에서 연구가 이루어져 왔다. 최근 실생활에 적용하기 위해 초기 네트워크 설정과 통신 보안, 라우팅 프로토콜, IP 주소 관리 등과 같은 주제들로 세계 각지에서 많은 연구가 진행되고 있다. 특히 라우팅 프로토콜 분야에서, 노드들의 이동으로 발생하는 링크 손실 문제를 해결하기 위한 많은 노력들이 이루어지고 있다. 지역 복구 방법의 개선, 다중 경로 탐색, 링크 손실 감지 방법 등을 통해 다수의 연구가 진행되고 있지만, 아직까지 뚜렷한 성능 개선을 보이고 있지 않다. 대부분의 연구가 특정한 상황을 대상으로 한 처리 방법만을 제시하거나 약간의 수정만을 가하는 것에 그치고 있어 경로 복구 문제에 대한 근본적인 해결점을 찾지 못하고 있다.

이에 본 논문에서는 대표적인 On-Demand 방식의 AODV를 기반으로 양방향 경로 복구 기법을 제안하여, 지역 복구의 범위를 확대함으로써 경로 복구 과정에서 발생하는 제어 트래픽 문제와 데이터 전송률을 개선하고자 하였다. 목적지와 가까운 경우에서만 수행되는 기본적인 AODV와는 달리, 손실된 링크를 기준으로 상류부/하류부 노드에서 각자 경로 오류 메시지와 요청 메시지를 전송하여 네트워크 전역에서 수행될 수 있는 지역

복구 기법을 제시하였다. 또한 2홉 범위로 플러딩을 제한하여, 1홉 범위의 링크 손실을 복구하기 위한 과도한 브로드캐스트 전송을 제어함으로써, 지역 복구에 소비되는 비용을 최소화하고자 하였다.

제시한 알고리즘의 성능 평가를 위해 여러 조건들을 변화시키며 시뮬레이션을 시행하였다. 첫 번째 시나리오에서는 전체 노드 수를 증가시켜 경로 복구 과정에서 발생하는 제어 트래픽과 많은 이웃 노드들의 우회로를 이용한 다중 경로 탐색 기법과의 성능을 평가하였다. 노드들의 이동 속도에 변화를 준 두 번째 시나리오에서는, 링크 손실 빈도를 높여 다른 알고리즘과의 경로 복구 과정의 효율성을 분석하였다. 두 가지 시나리오에 대하여 실험한 결과, 기본적인 AODV가 링크 손실과 경로 복구 과정에서 생성하는 브로드캐스트 패킷에 영향을 가장 많이 받는 것으로 나타났다. 첫 번째 시나리오에서는 노드 수가 증가할수록 다중 경로 탐색 기법이 다른 기법에 비해 다소 좋은 결과를 나타내었지만, 노드들의 이동 속도가 빠른 환경에서는 대체 경로에 대한 손실 문제가 발생함으로써, 현저한 전송률 저하 현상을 보였다. 반면 본 연구에서 제시하는 양방향 경로 탐색 기법은 노드 수나 이동 속도에 관계없이 고른 성능을 나타내었다. 특히 노드 수가 적은 중,소규모 네트워크에서 다른 프로토콜에 비해 뛰어난 성능을 보여, 회의실이나 실내의 이동 센서 네트워크에서 효과적인 라우팅이 가능할 것으로 판단된다. 노드의 이동 속도에 대한 적응력 또한 우수한 것으로 나타나 V2V(Vehicle-To-Vehicle), V2I(Vehicle-To-Infrastructure) 응용 분야에도 적용 가능성이 높은 것으로 분석된다.

본 논문에서 제시된 알고리즘을 관련 응용 분야에 적용하거나, 다른 알고리즘의 장점을 도입하는 방법으로 향후 연구 방향을 제시할 수 있다. 본 알고리즘의 주 비교 평가 대상인 AOMDV의 경우, 노드가 밀집된 공간에서 우수한 성능을 보여주는 점과 대체 경로를 이용함으로써 추가적인 제어 패킷이 필요 없다는 점 등을 도입할 수 있다. 특히 노드 및 링크가 중첩되지 않는 우회로 선정 기법에 대한 연구를 통해 링크 손실 문제에 보다 유연하게 대처할 수 있을 것이다. 교통 시뮬레이션과 연동한 차량 간 통신에서의 라우팅 기법, 주요 물품 관리 시스템 등과 같은 여러 응용 분

야에 적용함으로써, 제시된 경로 복구 기법의 실용화 가능성을 가늠해볼 수 있을 것이다.

V. 참고문헌

김동학, 김영락, 김종근, 「AODV기반의 지역경로탐색을 이용한 노드 비중첩 다중 경로 검색 기법」, 정보처리학회논문지 제14-C권 제1호, 2007

박노열, 김상경, 김창화, 「애드혹 네트워크에서의 라우팅 성능 개선」, 한국시물레이션학회 논문지 제15권 제3호 pp79-86, 09.2006

서현곤, 김기형, 서재홍 「AFLRS : 애드 혹 네트워크에서 AODV에 기반한 빠른 경로 복구 기법」, 정보과학회논문지, 제 31권 1호, 02.2004

안수길, 천수자, 안상현, 「효율적인 지역복구를 위한 AODV 라우팅 프로토콜의 개선」, 한국정보과학회 추계학술발표논문집 Vol. 30 No. 2, 2003

안태영, 서주하, 「지역적 복구를 이용한 개선된 AODV 알고리즘」, 한국정보과학회 가을학술발표논문집 Vol. 30 No. 2, 2003

이은주, 이권익, 좌정우, 양두영, 「차량간 통신을 위한 위치 정보 기반의 AODV 라우팅 프로토콜」, 한국콘텐츠학회논문지 '08 Vol. 8 No. 3 pp47-54, 2008

이재석, 박혜련, 이기현, 석정봉, 「Timestamp를 이용한 AODV 라우팅 알고리즘의 성능 개선」, 한국통신학회논문지 '06-8 Vol.31 No.8A pp830-836, 2006

C. E. Perkins, E. M. Royer, and S. R. Das. 「Ad hoc On-Demand Distance Vector (AODV) Routing」, <http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-10.txt>, Jan 2002. IETF Internet Draft

D.B johnson, D. A. Maltz, Yih-chun Hu and J. G. Jetcheva, 「The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks(DSR)」, Internet

Draft, IETF MANET Working Group, draft-ietf-manet-dsr-07.txt, 02. 2002

Charles E. Perkins, Pravin Bhagwat, 「Highly Dynamic Destination-Sequenced Distance-Vector Routing(DSDV) for Mobile Computers」, London, United Kingdom, Sep. 1994.

Mahesh K. Marina, Samir R. Das, 「Ad hoc on-demand multipath distance vector routing」, WIRELESS COMMUNICATIONS AND MOBILE COMPUTING Vol 6 Issue 7, pp969-988, 10. 2006

Mahesh K. Marina, Samir R. Das, 「On-Demand Multi Path Distance Vector Routing in Ad Hoc Networks」, Proceedings of the Ninth International Conference on Network Protocols, p.14, November 11-14, 2001

Srinath Perur, Abhilash P. and Sridhar Iyer, 「Router Handoff: A Preemptive Route Repair Strategy for AODV」, Personal Wireless Communications, 2002 IEEE International Conference, pp168-171, Dec. 2002

Micheal Pan, Sheng-Yan Chuang, Sheng-De Wang, 「Local Repair Mechanisms for On-Demand Routing in Mobile Ad hoc Networks」, Proceedings of the 11th Pacific Rim International Symposium on Dependable Computing, pp317-324, 2005

Sang-min Lee and Keecheon Kim, 「An Effective Path Recovery Mechanism for AODV using Candidate Node」, Frontiers of High Performance Computing and Networking - ISPA 2006 Workshops, LNCS 4331, pp1117-1125, 2006

Cigdem Sengul, 「Local Route Recovery in Mobile Ad hoc Networks」, Istanbul Technical University, 2000

Network Simulator 2 (<http://www.isi.edu/nsnam/ns/>)

ABSTRACT

An improved route recovery using Bidirectional Searching Method for Ad hoc Networks

Han, Ho-Yeon

Major in Information System Engineering

Dept. of Information System Engineering

Graduate School, Hansung University

In the growth of information technology, user requirements have been diversified in telecommunication environments. Because ad hoc network makes nodes communicate each other without infrastructure system and previous configuration, it comes into the spotlight that is suitable to ubiquitous society.

Of several research areas, routing protocol has been studied which can solve Link-Failure problem due to node movements. Though Many researches have been conducted by means of local repair improvement, multipath searching and link-failure detection, it couldn't present the primary solution.

This study proposes the Bidirectional Route Recovery method based on AODV which attempts to solve the control traffic problem and improve data delivery ratio. It can be used in all of the network area because upstream/downstream nodes send route error message and route request message to source/destination. Restricting the flooding of route request messages, it attempts to minimize the costs of local

repair.

It simulates this method for performance evaluation. In scenarios which variate the node numbers and node speed, it measures the control traffic and data delivery ratio. Proposed method shown the good performance regardless of the node numbers and node speed. Specially, it can apply to the small/medium network and V2V(Vehicle-to-Vehicle) application.