



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](#)

2009년 2월
석사학위 논문

모바일 애드혹 네트워크를 위한
고장 허용 라우팅 프로토콜

조선대학교 대학원

컴퓨터공학과

조 충 용

모바일 애드혹 네트워크를 위한
고장 허용 라우팅 프로토콜

A Fault-Tolerant Routing Protocol for Mobile Ad Hoc
Networks

2009년 2월

조선대학교 대학원

컴퓨터공학과

조 충 용

모바일 애드혹 네트워크를 위한
고장 허용 라우팅 프로토콜

지도교수 정 일 용

이 논문을 컴퓨터공학 석사학위신청 논문으로 제출함


2008년 12월


조선대학교 대학원


컴퓨터공학과

조 충 용

조충용의 석사학위 논문을 인준함

심사 위원장 조선대학교 교수 崔相同 

심사 위원 조선대학교 교수 李相晩 

심사 위원 조선대학교 교수 丁日鎭 

2009년 2월

조선대학교 대학원

목 차

ABSTRACT	iv
제1장. 서 론	1
제2장. 관련 연구	5
제1절. DSDV 프로토콜	7
제2절. AODV 프로토콜	11
1. 경로 발견	11
2. 경로 유지	22
3. 지역 경로 복구	26
제3장. 고장 허용 라우팅 프로토콜	29
제1절. 역 경로 설정	30
제2절. 우회 노드 탐색	35
제4장. 성능 평가	38
제1절. 모의실험 환경	38
제2절. 모의실험 결과 및 분석	42
제5장. 결 론	50
참고 문헌	52

그림 목 차

그림 1. 애드혹 네트워크에서의 멀티 홉 통신	3
그림 2. 애드혹 네트워크의 노드의 이동에 따른 변화	9
그림 3. 경로 요청 메시지 포맷	14
그림 4. AODV의 RREQ 패킷을 이용한 경로 탐색 과정	15
그림 5. 경로 응답 메시지 포맷	18
그림 6. AODV의 RREP 패킷을 이용한 역 경로 설정 과정	19
그림 7. 경로 응답 승인 메시지 포맷	21
그림 8. 경로 오류 메시지 포맷	24
그림 9. 경로 상의 단절 처리	28
그림 10. 경로의 단절에 따른 복구 과정	31
그림 11. 수정된 RREP 패킷의 구조	34
그림 12. 고장 허용 라우팅 프로토콜의 RSAL 패킷 구조	37
그림 13. 노드 이동성에 따른 패킷 전달율	43
그림 14. 노드 이동성에 따른 종단 간 전달 시간	44
그림 15. 노드 이동성에 따른 라우팅 오버헤드	45
그림 16. 데이터 전송률에 따른 패킷 전달율	47
그림 17. 데이터 전송률에 따른 종단 간 전달 시간	48
그림 18. 데이터 전송률에 따른 라우팅 오버헤드	49

표 목 차

표 1. MH4에 의한 경로 테이블 구조	10
표 2. 역 경로 엔트리 설정	20
표 3. 수정된 RREP에 의한 경로 테이블	32
표 4. 모의실험 환경	39

A B S T R A C T

A Fault-Tolerant Routing Protocol for Mobile Ad Hoc Networks

Chungyong Cho

Advisor : Prof. Ilyong Chung, Ph. D.

Department of Computer Engineering

Graduate School of Chosun University

A mobile ad-hoc network (MANET) is a self-configuring network composed of mobile nodes and wireless links. Mobile nodes are free to move randomly and are unstable. Since network topology changes rapidly and unpredictably in MANETs, the design of a robust routing protocol becomes extremely important.

In this paper, a fault-tolerant routing protocol based on AODV called FT-AODV is proposed for reliable and high-performance routing in MANETs. In FT-AODV, the RREP (Route Reply) packet is modified and a new RSAL (Route Salvage) packet is defined. When a link is broken, FT-AODV tries to find a detour path fast and allows the on-going data packet to be delivered to the destination.

The performance of the proposed FT-AODV is evaluated via tensive simulation, where the ns-2 network simulator is used. According to performance evaluation result, average end-to-end packet delay is reduced by 40% and routing load is decreased by 13% compared to AODV.

제1장 서 론

모바일 애드혹 네트워크[1]는 많은 기업체나 공공기관의 무선 네트워크 이용 방식을 변화시키고 있는 새로운 패러다임으로 무선 네트워크상에서 노드들이 AP (Access Point)를 통하지 않고 P2P 방식으로 서로 연결함으로써 무선 네트워크 통신이 가지는 거리상의 한계를 극복할 수 있다. 또한 노드의 이동이 자유롭기 때문에 네트워크의 토폴로지가 동적으로 변하는 특징을 가지고 있다. 이는 무선 네트워크의 구성 및 확장에 보다 나은 유연성을 제공함과 동시에 저렴한 비용으로 무선 네트워크의 구축 가능하게 한다. 모바일 애드혹 환경에서의 무선 노드는 사용 가능한 네트워크를 찾아 스스로 망 구성을 할 수 있다. 이 구조에서는 중간에서 제어하는 노드가 없으므로 각 노드들은 자신이 가질 수 있는 정보를 최대한 활용하여 네트워크에서 통신하며 먼 거리의 노드와의 통신에는 다른 노드들을 경유하여 통신한다. 이에 따라 모바일 애드혹 네트워크에 적합한 라우팅 방법이 필요하며 라우팅 방법은 홉 수, 에너지 소모, 전달 시간과 라우팅 오버헤드의 최소화를 고려해야 한다. 모바일 애드혹 네트워크는 자체적으로 끊어진 경로를 감지하고 복구하는 기능을 제공한다. 이러한 자가 망 형성 및 자가 망 관리 기능을 통해 모바일 애드혹 네트워크는 기존의 무선 사업자들이 무선 네트워크를 쉽게 확장하고 사업을 성장시킬 수 있게 할 뿐만 아니라, 무선 사업에 있어서 전례 없는 유연성과 단순성을 가져다주었다.

그림 1은 모바일 애드혹(MANETs: Mobile Ad hoc Networks) 환경에서 데이터를 멀티 홉으로 전달하기 위한 경로 설정으로 노드 A의 이동으로 인해 경로가 재설정되는 과정을 보이고 있다. 이와 같이 애드혹 환경에서는 무선 링크와 동적인 토폴로지라는 특징으로 인해 빈번하고 예측하기 어려운 경로의 변화를 겪는다. 여기에 에너지와 대역폭이라는 제약 조건으로 인해 전통적인 링크

상태 라우팅 알고리즘이나 거리 벡터 라우팅 알고리즘은 애드혹 환경에서는 효과적이지 않다. 따라서 애드혹 환경을 위한 많은 라우팅 알고리즘[2]이 제안되어 왔는데 애드혹을 위한 라우팅 알고리즘은 토폴로지, 경로의 선택, 경로를 요청한 노드의 위치 등 특수한 환경을 고려하며 경로 탐색에 있어서 빠르고 효과적이어야 한다. 애드혹 네트워크에서 라우팅 프로토콜을 결정하는 가장 주된 요소로는 노드가 네트워크의 토폴로지가 변경되는 동안 데이터가 지나갈 경로를 결정함에 있어서 자신과 연결되어 있으면서 그 정보를 전달할 이웃들의 정보를 아는 것이다. 더구나 노드의 수가 늘어날수록, 토폴로지의 크기가 커질수록 목적지까지의 경로를 찾는 데 있어서 노드와 노드 사이의 라우팅 정보의 교환이 매우 빈번하게 일어나게 되고 이에 따라 라우팅 정보들의 갱신과 이에 따른 오버헤드는 기하급수적으로 증가한다. 애드혹 네트워크에서 경로를 설정하고 유지하는 라우팅 알고리즘은 경로 발견 기법을 통해 경로를 찾아 설정하고 이후 전송되는 데이터는 이때 설정된 경로를 사용하여 전달된다. 유선에 비해 이동이 빈번하고 잔여 배터리 에너지 등의 조건에 따라 토폴로지 및 생성된 경로가 변화될 수 있기 때문에 라우팅 프로토콜은 주기적인 메시지 교환을 통해 경로를 유지하며 경로가 단절되었을 경우에 능동적으로 대처할 수 있어야 한다.

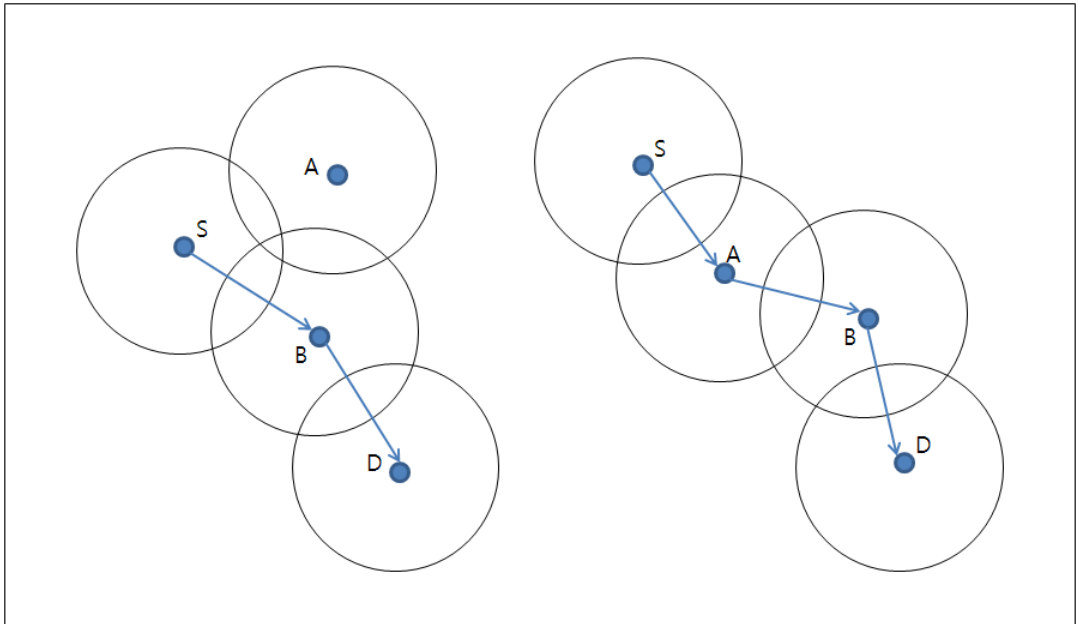


그림 1. 애드혹 네트워크에서의 멀티 홉 통신
Fig 1. An example of a multi-hop MANET

본 논문에서는 모바일 애드혹 네트워크에서 가장 많이 사용되고 있는 대표적 라우팅 프로토콜의 하나인 AODV 라우팅 프로토콜[8][9][10]을 기반으로 한 고장 허용 라우팅 알고리즘을 제안한다. 노드의 고장 확률을 p 라고 하면 두 개 노드가 동시에 고장 날 확률은 p^2 이 된다. 만약 노드 고장 확률이 1% ($p = 0.01$)라고 가정하면, 두 개 노드가 동시에 고장 날 확률은 0.01% ($p^2 = 0.0001$)로 현저하게 낮아진다. 따라서 단일 노드 고장에 대한 고장 허용 라우팅을 제공할 경우 네트워크의 신뢰성 및 전송 성능이 크게 향상될 수 있다. 제안 알고리즘은 기존의 방법[8][9][10]과 다르게 고장 검출 시 경로 발견 절차를 거치지 않고 우회 경로를 제공하여 오버헤드를 줄이고 신속하게 경로를 재설정할 수 있다. 경로 단절 시 경로의 재탐색을 위해 고장지점에서 목적지까지 RREQ (Route Request)를 플러딩하지 않고, 가까이 있는 노드 중 목적지로 패킷을 중계할 수 있는 노드를 찾아 경로를 복구하므로 기존의 방법보다 빠르고 제어 트래픽을 크게 유발하지 않음으로 네트워크 부하를 경감시켜 효과적으로 고장 노드를 우회하는 경로를 설정할 수 있다.

제2장 관련 연구

모바일 애드혹 네트워크에서의 라우팅 프로토콜은 크게 두 가지로 나뉘는데 하나는 사전 대응[2] 라우팅 방식, 다른 하나는 사후 대응[2] 라우팅 방식으로 분류할 수 있다.

사전 대응 라우팅 방식은 최단거리 탐색 방법인 Bellman-Ford [3] 알고리즘을 애드혹 네트워크에 적용한 방식으로, 애드혹 네트워크 내의 각 노드는 네트워크 상의 모든 도달 가능한 노드들까지의 라우팅 정보를 자신의 라우팅 테이블에 기록한다. 컨트롤 메시지의 주기적인 교환을 통해 전체의 라우팅 테이블을 유지하는 방식이며, 라우팅 정보를 항상 최신의 것으로 유지하는데 이를 위하여 각 노드들이 주기적으로 라우팅 정보를 갱신하거나 네트워크의 토폴로지의 변경이 있을 때마다 네트워크 전체로 변경 정보를 전파시켜 자신의 라우팅 정보를 변경하게 한다. 따라서 사전 대응 방식에서의 각 노드들은 항상 최신의 완성된 네트워크 토폴로지 정보를 가지고 있으며 이를 토대로 신속하게 목적지에 이르는 최적의 경로를 즉시 찾아서 정보를 전달할 수 있다는 장점이 있다. 그러나 토폴로지 변화가 빈번할 경우 그만큼 많은 양의 컨트롤 메시지를 발생시키며 변경된 라우팅 정보를 전체 네트워크에 전달하기 때문에 부하가 크다는 단점을 가진다. 사전 대응 방식의 라우팅 프로토콜로는 DSDV [4], TBRPF [5], OLSR [6] 등이 있다.

사후 대응 라우팅 방식은 전송할 패킷이 있을 때 경로를 찾아 설정하는 방법으로서 사전 대응 방식에서의 빈번한 갱신 부하를 줄일 수 있다는 장점이 있

다. 라우팅 정보는 경로 상의 각 노드에 저장되고 일정 시간 동안 사용되지 않은 경로는 테이블에서 삭제된다. 사후 대응 방식은 트래픽이 발생하는 시점에서 경로 탐색을 수행하기 때문에 사전 대응 방식에서 발생하는 제어 메시지 오버헤드 문제를 해결함으로써 네트워크 전체의 부하를 경감시킨다. 그러나 이러한 요구 기반 방식의 라우팅 프로토콜에서는 초기 경로 탐색에 따른 지연이 발생한다. 그럼에도 불구하고 사전 대응 방식보다 이동에 적합한 형태를 띠고 있기 때문에 애드혹 네트워크에 가장 적합한 방식으로 알려져 있다. 이 방식에 속하는 라우팅 프로토콜로는 DSR [7], AODV [8][9][10], TORA [11] 등이 있다.

제1절 DSDV 프로토콜

DSDV (Destination Sequenced Distance Vector) [4] 프로토콜은 거리 벡터 라우팅 알고리즘의 하나이며 최단 거리의 경로를 찾는 Bellman-Ford 알고리즘 [3]의 loop와 infinity 문제를 일련번호를 사용하여 해결한 프로토콜이다.

각 노드들은 주기적으로 컨트롤 메시지를 브로드캐스팅하며, 메시지에 일련번호를 포함하여 보내면 메시지를 받은 노드들은 자신의 라우팅 테이블에 있는 엔트리와 받은 메시지의 일련번호를 비교하여 받은 메시지의 일련번호가 크면 새로운 메시지로 간주하고 자신의 경로 테이블을 갱신한다. 표 1은 DSDV 프로토콜에서 그림 2의 MH₄(Mobile Host)에 의한 경로 테이블을 네트워크 전체에 전파하는 내용이며, 표 1은 그림 2에 대한 경로 테이블을 나타낸 것이다.

표 1에서 Destination은 목적지의 물리적인 주소를 나타내고, metric은 목적지까지의 hop 수를 나타낸다. 여기에서 sequence number는 Bellman-Ford [3] 알고리즘의 looping과 infinity 문제를 개선하기 위해 도입된 애트리뷰트로 기존의 라우팅 정보와 새로 갱신되는 라우팅 정보를 비교하여 인접한 노드에 라우팅 테이블을 전송할 때마다 1씩 증가한다.

DSDV [4]는 라우팅 테이블을 갱신하는데 있어서 패킷에서 운반되는 정보의 양을 줄이기 위해 전체 배포 방식과 부분 배포 방식을 사용한다. 전체 배포에서는 라우팅 테이블의 모든 정보를 전송하는 것이며, 부분 배포는 전체 배포가 성공한 이후에 변경된 항목만을 전송함으로써 네트워크에서의 트래픽 발생을 줄일 수 있다. DSDV는 네트워크의 트래픽을 줄이기 위해 부분 배포를 사용하고, 네트워크에 새로이 참여하는 노드가 있거나 네트워크를 이탈하는 노드가 발생

하는 등의 네트워크의 토폴로지가 크게 변하는 경우에는 전체 배포를 사용한
다.

DSDV [4] 프로토콜에서 각각의 노드들은 자신의 이웃 노드들과 주기적인 통신을 통해 노드의 추가 또는 이탈을 감지함으로써 자신의 테이블을 갱신하기 때문에 경로를 발견하는데 매우 효율적이며 빠른 탐색이 가능하다. Bellman-Ford [3]알고리즘의 문제점인 looping과 infinity 문제도 해결하였다. 그러나 경로를 갱신하는 과정에서 많은 컨트롤 메시지를 발생시켜 네트워크의 부담이 커지기 때문에 중대형 급의 모바일 애드혹 네트워크에는 적합하지 않다.

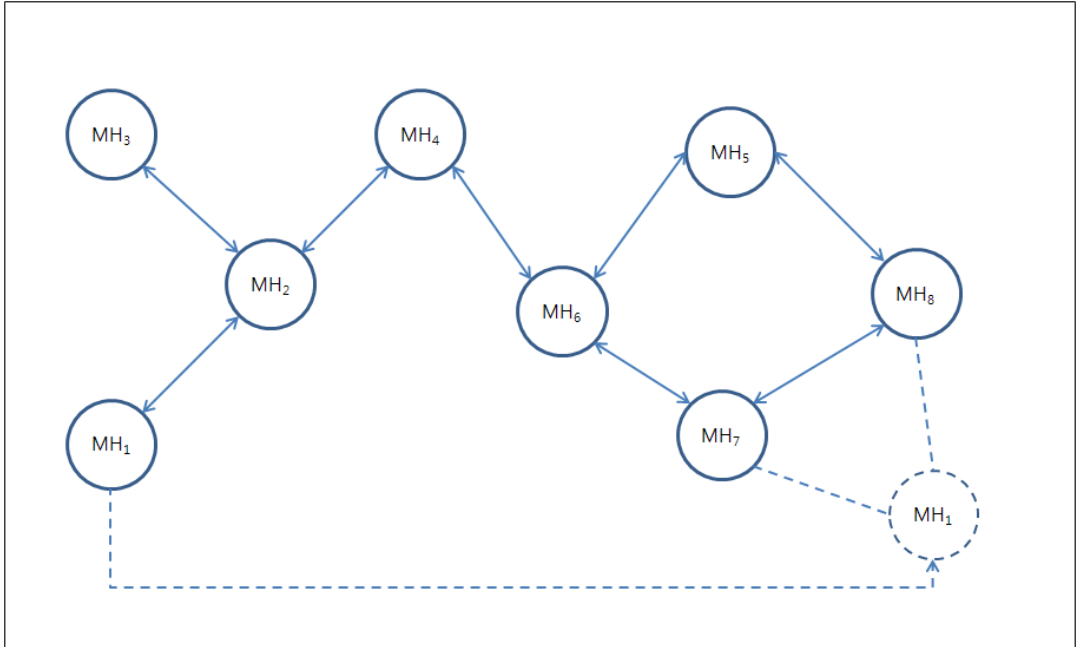


그림 2. 애드혹 네트워크의 노드의 이동에 따른 변화
 Fig 2. Node movement in an ad-hoc network

표 1. MH₄에 의한 경로 테이블 구조

Table 1. Structure of the MH₄ routing table

Destination	Metric	Sequence Number
MH ₁	2	S406_MH ₁
MH ₂	1	S128_MH ₂
MH ₃	2	S564_MH ₃
MH ₄	0	S710_MH ₄
MH ₅	2	S392_MH ₅
MH ₆	1	S076_MH ₆
MH ₇	2	S128_MH ₇
MH ₈	3	S050_MH ₈

제2절 AODV 프로토콜

AODV (Ad-hoc On-demand Distance Vector) [8][9][10]는 DSDV [4]와 DSR [7]의 조합을 기본으로 채택한 알고리즘으로 DSR의 경로 획득과 경로 유지 메커니즘에 DSDV의 hop-by-hop과 중복 방지를 위한 일련번호 개념과 주기적인 beacon 신호의 개념을 함께 적용한 라우팅 알고리즘이다. AODV는 경로를 탐색하는데 있어 요구되는 브로드캐스트의 수를 최소화하기 위해 요구 기반을 기본으로 하는 알고리즘으로 이는 DSDV 알고리즘이 갖는 완전한 전체 경로 목록을 유지하는 것과 큰 차이를 보이는 특징이다. AODV는 순전한 요구 기반 방식의 경로 탐색 알고리즘으로 경로에 참여하지 않는 노드는 라우팅 정보를 유지하지 않고, 라우팅 테이블의 교환에 참여하지도 않는다. AODV는 단지 경로 탐색과 경로 유지, 그리고 데이터의 전달에만 관여한다[2][9].

1. 경로 발견

어떤 소스노드가 어떤 목적지로 메시지를 보내려 할 때 경로를 가지고 있지 않다면 소스노드는 경로 탐색 과정을 먼저 수행하게 된다. 소스 노드는 RREQ (Route Request) 패킷을 자신의 이웃 노드들에게 전파하고 그 RREQ 패킷을 받은 이웃 노드는 또 다시 자신들의 이웃들에게 RREQ 패킷을 전달한다. 이러한 과정을 반복하면서 목적지 혹은 fresh enough route를 가진 중간 노드까지 경로 탐색을 수행한다.

AODV는 경로 탐색 시 순환 문제에 봉착하는 것을 방지하고 가장 최근의 라우팅 정보를 유지하기 위해 목적지 일련번호를 사용하며 각각의 노드는 브로드캐스트 ID처럼 RREQ 패킷에 RREQ ID를 유지한다. 목적지 일련번호는 경로를 요청하는 노드로 보내는 경로 정보와 함께 포함될 목적지 노드에서 생성된다.

그림 3과 같이 RREQ를 수신한 중계 노드가 목적지까지로 향하는 경로를 가지고 있으며 목적지까지의 일련번호가 같거나 크다면 RREQ에 대한 응답 메시

지를 보낼 수 있다. RREQ 패킷의 중계가 이루어지는 동안 중계 노드들은 브로드캐스트된 RREQ 패킷의 처음 도착한 내용에서 그들의 이웃 노드들의 주소와 경로 테이블을 기록한다. 그것에 의해 역 경로가 생성된다. 만약, 처음 도착한 RREQ의 브로드캐스트 된 패킷 이후에 수신된 같은 패킷들은 버려진다.

RREQ 패킷은 한 노드가 어떤 목적지 노드로 데이터를 전송하려고 할 때 경로를 획득하기 위한, 경로 요청 메시지로 자신의 이웃들에게 RREQ를 브로드캐스팅하고, RREQ를 받은 중계 노드가 목적지 노드이거나 목적지로 가는 충분한 경로를 가지고 있다면 경로 요청의 다음 단계로 진행한다.

유형 : AODV의 패킷의 종류를 구분하는 항목으로 RREQ 패킷의 경우는 1의 값을 가진다.

플래그

- J (Join) Flag : 멀티캐스트용으로 예약된 것으로 노드가 멀티캐스팅 그룹에 결합하기 위해 사용한다.

- R (Repair) Flag : 멀티캐스트용으로 예약된 것으로 노드가 멀티캐스트 트리에서 연결이 끊어졌을 때, 끊어진 경로를 복구하기 위해 사용한다.

- G (Gratuitous RREP) Flag : RREP 패킷이 목적지 노드에게 유니캐스트로 전송되어야 하는지에 대한 플래그이다.

- D (Destination only) Flag : 중간 노드가 RREQ 패킷에 대해 일일이 RREP를 전송한다면 목적지 노드는 경로획득에 대한 메시지를 받지 못해 소스노드에 대한 정보를 알 수 없게 된다. 따라서, 이 플래그를 사용하여 오직 목적지 노드만이 RREQ에 응답하도록 한다.

- U (Unknown sequence number) Flag : 어떤 문제가 발생하여 목적지의 일련번호를 알 수 없는 경우 사용한다.

예약 : 0으로 전송하며, 수신시에는 무시한다.

홉 카운트 : 발신자 IP 주소에서 요청을 처리하는 노드까지의 홉의 횟수로, 소스노드에서 홉 카운트를 0을 설정하고 중간 노드들이 소스노드가 브로드캐스트한 RREQ 패킷을 받으면 패킷에 저장된 홉 카운트에 1을 증가시킨다.

RREQ ID : RREQ의 일련번호로 RREQ를 식별한다. RREQ를 생성할 때마다 1씩 증가시키며 각 노드는 하나의 RREQ ID만을 유지한다. RREQ ID는 전송 노드의 IP 주소와 함께 선택될 때 특정 RREQ를 유일한 것으로 식별할 수 있도록 하기 위해 사용한다.

목적지 IP 주소 : 소스노드가 데이터패킷을 보내고자 하는, 최종 목적지 노드의 IP 주소이다.

목적지 일련번호 : 목적지를 향해 어떤 경로에 대해 발신자가 과거에 수신했던 마지막 일련번호로, 링크 단절과 같은 문제로 인해 경로 정보 교환 시 경로 상에서 발생할 수 있는 순환문제를 해결하며 순 경로를 설정할 때 사용한다. 소스 노드는 목적지 노드로부터 가장 최근에 받은 RREP 패킷의 일련번호를 설정한다.

발신자 IP 주소 : 경로를 요청한 노드(RREQ를 발신한 노드)의 IP 주소이다.

발신자 일련번호 : 경로 요청의 발신자를 향하는 경로 엔트리를 위해 사용되는, 소스 노드 자신의 일련번호로 RREQ를 발생시킬 때 값을 증가시켜 방송하며, 역 경로를 설정할 때 사용한다.

그림 4는 RREQ 패킷을 네트워크에 전파하면서 경로를 탐색하는 과정을 보여준다. 소스노드인 1번 노드가 목적지노드인 노드 8에게 데이터를 보내고자 할 때 먼저 노드 1은 노드 8까지의 경로를 탐색하기 위해 RREQ 패킷을 브로드캐스트한다. 노드 1로부터 브로드캐스트된 RREQ는 1-hop의 거리에 있는 노드 2와 노드 3에게 전달되고 노드 2와 노드 3은 다시 노드 5에게 전달한다. 노드 5는 노드 2와 노드 3을 통해 노드 1이 발송한 RREQ를 전달받게 되는데 이때 받은 패킷이 동일한 패킷이므로 먼저 받은 패킷을 처리하고, 나머지 패킷은 버려진다. 이러한 전달 과정을 반복하면서 RREQ 패킷이 노드 8로 전달이 되면 RREQ 메시지의 송신과정은 종료된다.

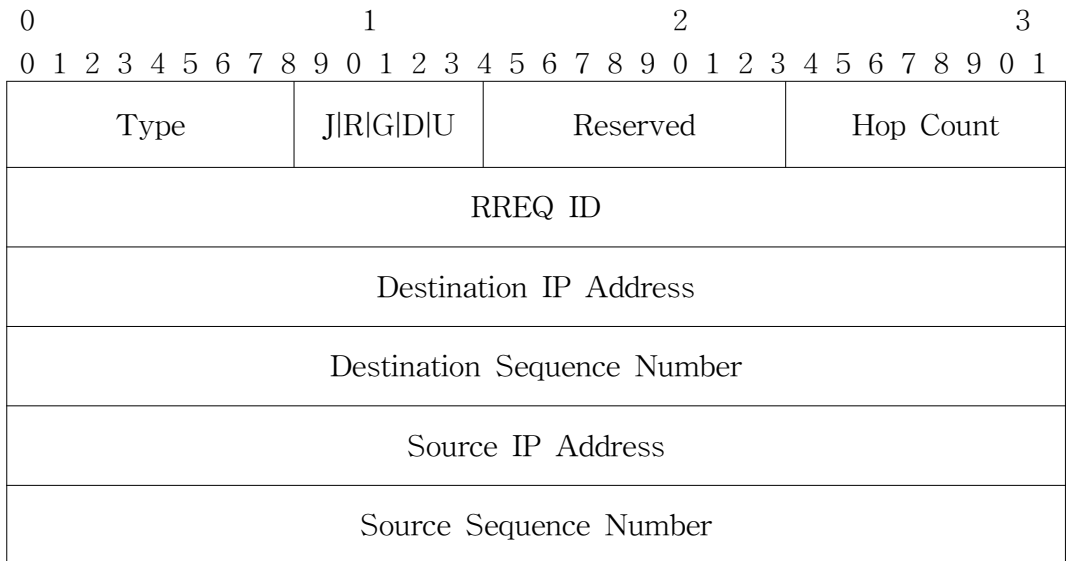


그림 3. 경로 요청 메시지 포맷
 Fig 3. RREQ message format

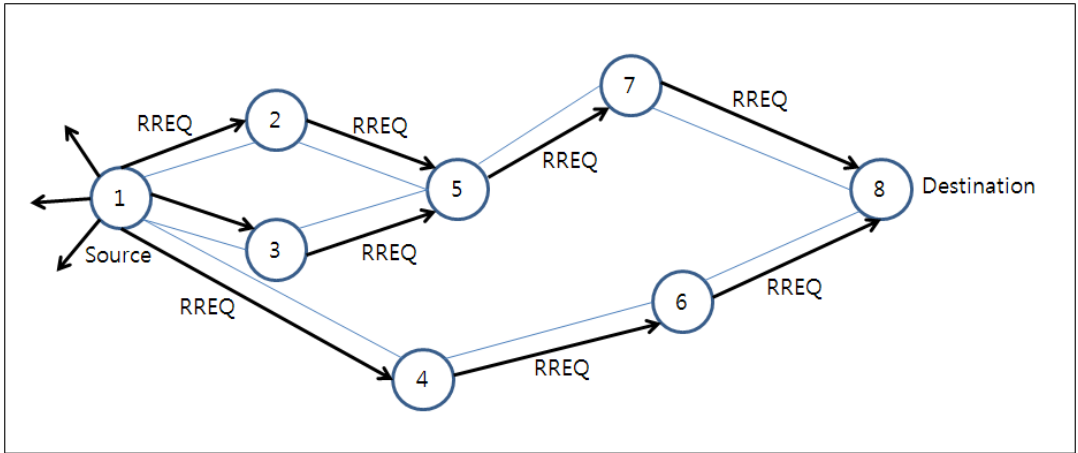


그림 4. AODV의 RREQ 패킷을 이용한 경로 탐색 과정
 Fig 4. Propagation of RREQ in AODV

RREQ 패킷이 목적지 노드나 혹은 목적지 노드까지의 경로를 가지고 있는 중계 노드에 도착하면 목적지 노드, 혹은 경로를 가진 중계 노드는 브로드캐스트가 아닌, 유니캐스트 방법으로 RREQ가 왔던 역 경로로 RREP (Route Reply)패킷을 자신의 이웃에게 송신한다. 이 동작 역시 앞에서 설명한 것과 같이 브로드캐스트 된 처음의 RREQ 패킷을 수신하였을 때 이루어지며 이후 수신한 동일한 RREQ 패킷은 무시된다. 어떤 노드가 RREQ 메시지를 수신하면, RREQ를 수신한 노드는 RREQ 메시지에 들어있는 소스 노드의 주소와 목적지 노드의 주소, 일련 번호, 홉 카운트 및 RREQ 메시지를 받은 이웃 노드의 IP 주소를 역 경로 엔트리에 저장하며 이러한 설정 과정을 역 경로 설정이라고 한다.

그림 5는 경로 응답 메시지의 포맷으로, 다음과 같은 필드를 가지고 있다.

유형 : AODV의 패킷의 종류를 구분하는 항목으로 RREP 패킷의 경우는 2의 값을 가진다.

플래그

- R (Repair) Flag : 멀티캐스트용으로 예약된 것으로 노드가 멀티캐스트 트리에서 연결이 끊어졌을 때, 끊어진 경로를 복구하기 위해 사용한다.

- A (Acknowledgement Required) : RREP를 송신한 후 상대 노드로부터 자신이 보낸 RREP 메시지에 대한 확인을 받고자 할 때 이 플래그를 설정한다. 이 값이 설정되어 있다면 RREP-ACK 메시지를 받을 수 있다.

예약 : 0으로 전송하며, 수신시에는 무시한다.

홉 카운트 : 소스 노드에서 목적지 노드까지의 홉 수이다.

목적지 IP 주소 : RREQ의 응답으로 RREP를 생성한 노드의 주소이다.

목적지 일련번호 : 해당 경로와 관련된 목적지 일련번호이다.

발신자 IP 주소 : 경로를 요청한 노드(RREQ를 발신한 노드)의 IP 주소이며

RREP 패킷의 목적지이다.

수명 : RREP를 받은 노드가 해당 경로를 유효한 것으로 간주하는 경로 수명의 시간.

그림 6은 AODV에서 RREP 패킷이 역 경로를 설정하는 과정을 보인 것으로 그림 6은 그림 4의, 각각의 노드들에 대한 역 경로 엔트리가 설정되는 과정으로 표 2에 경로 테이블에 기록되는 과정을 보이며, 역 경로 엔트리를 작성하는 목적은 RREP 메시지를 수신했을 때 RREQ 메시지를 발생한 이웃 노드에게 RREP 메시지를 전송하기 위해서이다.

그림 7은 경로 응답 승인(RREP-ACK) 메시지의 포맷을 나타낸 것으로 반드시 RREP 메시지의 플래그에 'A'비트가 설정된 결과로 전송되며 단방향 링크가 경로 발견 사이클이 완료되지 못하도록 할 위험이 있을 때 사용된다.

유형 : AODV의 패킷의 종류를 구분하는 항목으로 RREP-ACK 패킷의 경우는 4의 값을 가진다.

예약 : 0으로 전송하며 수신시에는 무시한다.

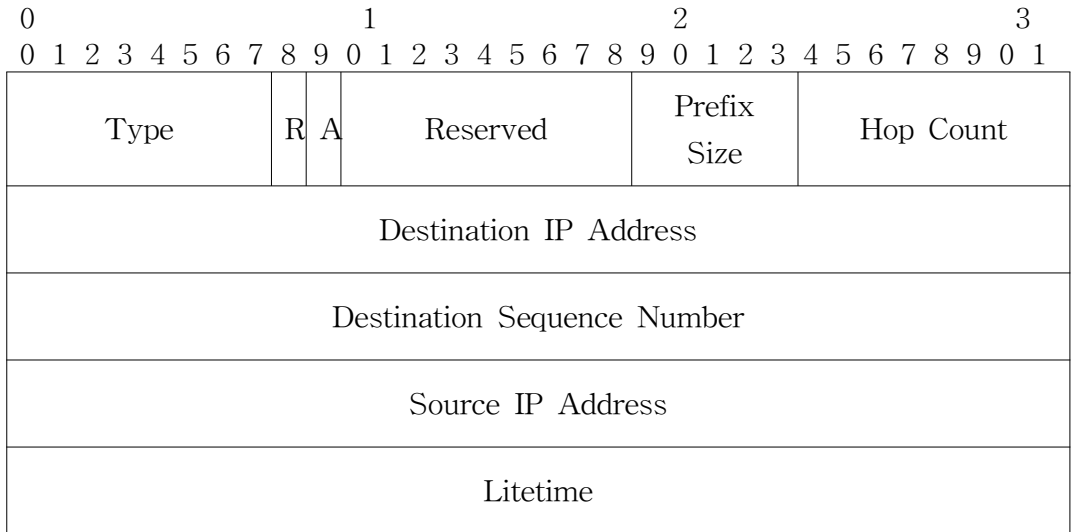


그림 5. 경로 응답 메시지 포맷
 Fig 5. RREP message format

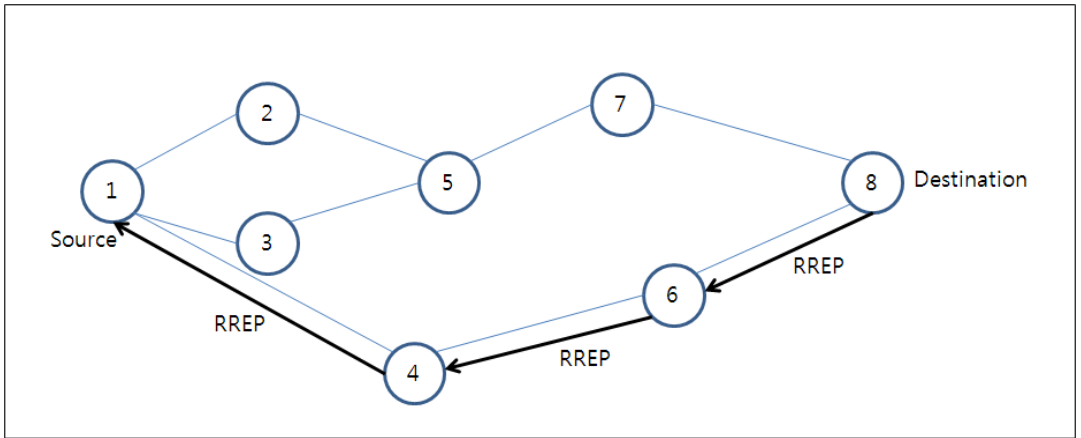


그림 6. AODV의 RREP 패킷을 이용한 역 경로 설정 과정

Fig 6. Path taken by RREP in AODV

표 2. 역 경로 엔트리 설정
 Table 2. Setup of reverse route entry

Node	Source Address	Destination Address	Source Sequence	Hop Count	Neighbor Address
4	1	8	1	1	1
6	1	8	1	2	4

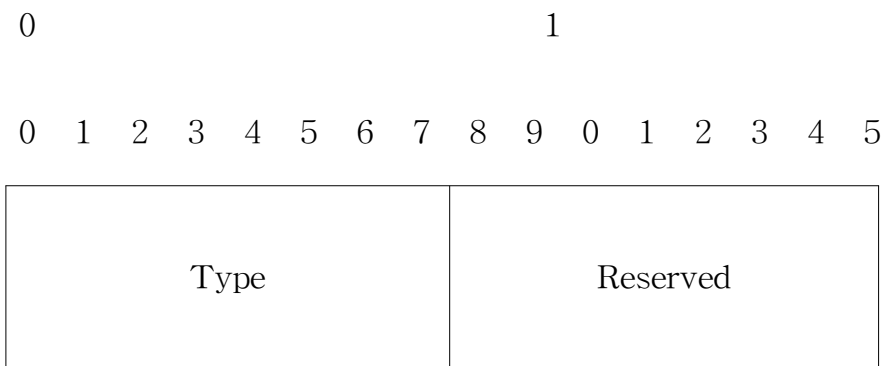


그림 7. 경로 응답 승인 메시지 포맷
 Fig 7. RREP-ACK message format

2. 경로 유지

AODV는 경로의 생존시간 동안 해당 경로가 사용되지 않을 경우 라우팅 엔트리에서 해당 경로를 삭제한다. 이는 RREQ에 의해 RREP가 전송되면서 경로가 생성되는, AODV의 대칭 링크만을 지원하는 특성 때문이다.

탐색된 경로들은 다음과 같이 유지 된다. 만약, 어떤 소스 노드가 이동을 한다면, 목적지까지의 경로를 찾기 위해 다시 경로 탐색 과정을 수행하는 경로 초기화 작업을 수행할 수 있다. 만약 어떤 노드가 경로를 따라서 이동한다면, 이동한 노드의 상위 노드는 이와 같은 움직임을 경로를 따라 알리고 RERR (Route Error)를 현재의 유효한 상위 노드들에 전송하여 해당 경로를 라우팅 테이블에서 삭제하며, 이 RERR은 최종적으로, 경로 탐색을 최초로 요청했던 소스노드에게로 전달된다.

RERR 메시지는 링크가 단절되었으며 링크의 단절로 인해 더 이상 통신이 불가능함을 알림으로써 기존의 경로를 초기화시키며, 소스 노드가 경로 단절로 인해 해제된 링크를 통해 여전히 데이터의 전송이 필요하다면 처음부터 경로를 탐색하도록 한다. 그림 8은 경로 오류 메시지의 포맷을 나타낸 것으로 다음과 같은 필드를 가지고 있다.

유형 : AODV의 패킷의 종류를 구분하는 항목으로 RERR 패킷의 경우는 3의 값을 가진다.

N (No Delete) Flag : 무삭제 플래그이다. 무삭제 플래그는, 노드가 지역 경로 복구를 수행할 때 설정하는 항목으로, 이 부분이 설정됨으로써 RERR의 본래의 목적인, RERR을 소스 노드로 보내어 기존 연결을 해제하는 작업을 하지 않고, 지역 경로 복구를 수행하는 동안 상위 노드들이 기존의 연결을 해제하지 못하도록 한다. 만약 지역 경로 복구가 수행되는 동안 기존의 연결이 해제되어

버린다면 지역 경로를 복구한다고 하더라도 소스 노드로부터 더 이상의 데이터는 오지 않을 것이므로 끊어진 경로를 복구하는 것은 의미가 없다.

예약 : 전송 시에는 0으로 보내며, 수신시에는 이를 무시한다.

DestCount : 메시지에 포함되어 있는, 도달할 수 없는 목적지의 수를 나타내며 최소한 1 이상이어야 한다. 하나의 링크를 여러 경로가 공유할 경우에 1 이상이 될 수 있다.

도달할 수 없는 목적지 IP 주소 : 링크 단절로 인해 도달할 수 없게 된 목적지의 IP 주소를 기록한다.

도달할 수 없는 목적지 일련번호 : 이전의 도달할 수 없는 목적지 IP 주소 필드에 있는 목적지에 대한 경로 테이블 엔트리에 있는 일련번호를 기록한다.

도달할 수 없는 추가적인 목적지 IP 주소 및 도달할 수 없는 추가적인 목적지 일련번호 : 도달할 수 없는 목적지의 수가 많으면 많을수록 그 수만큼 추가되며 필드의 역할은 앞에서 설명한 도달할 수 없는 추가적인 목적지 IP 주소와 도달할 수 없는 추가적인 목적지 일련번호와 같다.

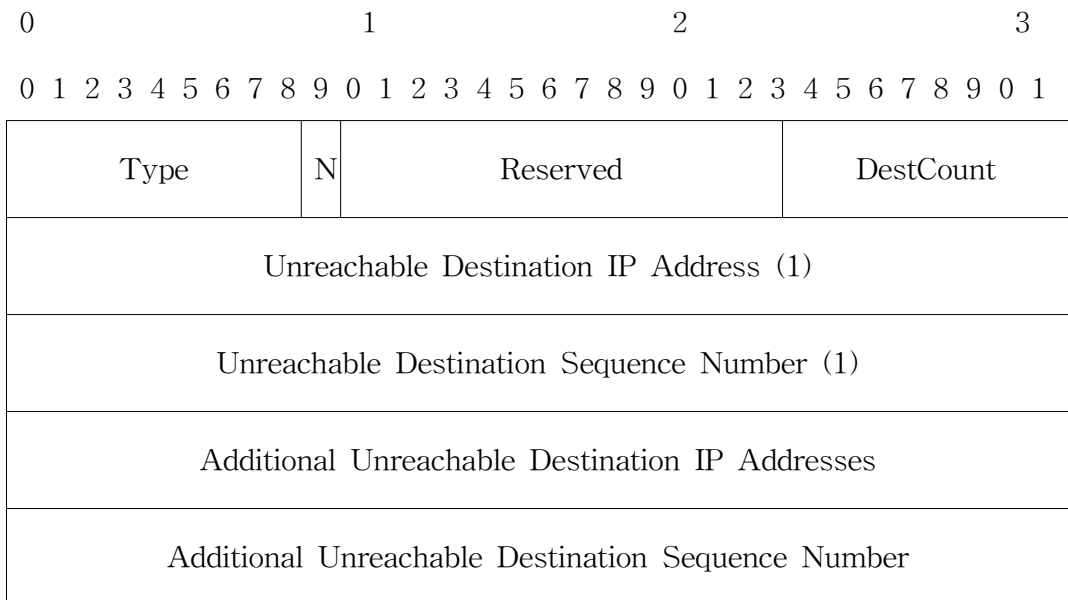


그림 8. 경로 오류 메시지 포맷
Fig 8. RERR message format

그러나 소스노드가 링크가 해제된 경로의 목적지로 계속 데이터를 전달해야 할 필요가 있을 경우, 소스노드는 다시 RREQ 패킷을 이전과 같이 브로드캐스트 함으로써 경로를 탐색할 수 있다.

또한, AODV 프로토콜[8][9][10]은 어떤 한 노드의 주변에 어떤 노드들이 있는지를 확인하여 이웃 노드들을 관리하기 위한 테이블을 가지고 있으며 주기적으로 1-hop간의 거리에 hello 메시지를 브로드캐스트 하여 이웃 테이블을 유지한다. 그러나 노드의 지역 연결을 유지하기 위한 방법으로 사용되는 이 hello 메시지의 브로드캐스트가 언제나 필요한 것은 아니다. 노드는 데이터 패킷의 재전송을 듣고 다음 노드가 여전히 자신과 통신이 가능하다는 것을 인지할 수 있다. 만약 이러한 재전송을 듣지 못한다면 노드는 hello 메시지를 포함한 관련 기술 중 한 가지를 선택하여 다음 홉이 여전히 전송범위에 있는지를 결정하려 할 것이다. hello 메시지는 자신의 hello 메시지에 대한 응답을 한 노드뿐만 아니라 다른 노드로의 통신을 감지함으로써 역시 자신의 이웃 테이블을 관리할 수 있다. 이러한 방법으로 더 많은 네트워크 연결 정보를 알 수 있다.

AODV 프로토콜[8][9][10]은 IEEE 802.11에서 제공하는 링크 계층의 통보 기능을 이용하여 연결 상태를 판단할 수 있다. 예를 들어, 허용된 최대 회수만큼 재전송을 시도한 후에도 RTS (Request To Send)를 보낸 후 링크 계층의 응답 (ACK)이 없거나 CTS (Clear To Send)를 얻지 못한다는 것은 활성화된 다음 홉에 대한 링크가 단절되었음을 의미한다. 링크 계층의 통보 기능을 이용할 수 없는 경우에는, 다음 홉으로부터 어떠한 패킷이라도 수신하거나 주기적인 hello 메시지를 수신한 경우, 다음 홉으로의 경로를 요청하는 다음 홉에 대한 RREQ 유니캐스트, 다음 홉으로 ICMP 에코 요청 메시지 유니캐스트 등의 방법으로 링크의 연결 상태를 확인할 수 없다. 이 방법들 중 어떤 것을 통해서도 다음 홉으로의 링크를 감지할 수 없는 경우 중계 노드는 그 링크가 손실되었다고 판단하며 링크 단절에 따른 절차를 수행한다.

3. 지역 경로 복구

링크 단절을 탐지한 상위 노드는 상황에 따라서 RERR 메시지를 소스 노드와 경로 상의 노드들에게 전송하여 기존의 경로를 무효화 하거나 지역 경로 복구를 수행하여 단절된 경로를 복구할 수 있다.

링크가 단절을 인지한 노드는 단절된 지점부터 목적지까지의 홉 수가 MAX_REPAIR_TTL [10]보다 큰 경우 RERR 메시지를 소스 노드와 경로 상의 노드들에게 전송하여 기존의 경로를 무효화 시킨 후 처음부터 다시 경로를 탐색한다. MAX_REPAIR_TTL보다 목적지까지의 홉 수가 더 크다는 것은 그림 9의 (a)에서 보는 바와 같이 단절된 지점이 전체 경로 중 목적지보다 소스 노드 쪽에 더 가깝다는 의미이다. MAX_REPAIR_TTL이 목적지까지의 홉 수보다 큰 경우는 단절된 지점이 전체 경로 중 소스 노드보다 목적지 노드에 더 가깝다는 의미로 그림 9의 (b)에서 보는 바와 같다. 이 경우 AODV 프로토콜은 지역 경로 복구[8][10][12]를 수행한다. 지역 경로 복구는 최초 경로를 탐색하는 과정과 유사하다.

단절된 링크를 복구하기 위해, 경로의 단절을 인지한 노드는 목적지에 대한 일련 번호를 증가시킨 다음 목적지에 대한 RREQ를 브로드캐스팅 한다. RREQ의 TTL은 초기에 $\max(\text{MIN_REPAIR_TTL}, 0.5 * \text{\#hops}) + \text{LOCAL_ADD_TTL}$ [10]로 설정한다. 여기에서 #hop은 originator라 불리는 경로가 단절된 지점의 상위 노드부터 패킷이 도달할 수 없는 지점까지의 홉 수이다. 소스 노드는 경로가 끊어졌는지에 대한 여부를 RERR 메시지가 오기 전까지는 알 수기 때문에 소스 노드는 경로 단절을 인지하지 못한 채 데이터를 송신한다. 따라서 이렇게 송신되는 데이터들이 단절된 경로로 인해 유실될 수가 있기 때문에 단절된 지점의 상위 노드는 지역 경로 복구가 진행되는 동안 자신

의 버퍼에 전달해야 할 데이터들을 저장한다. 단절 지점의 상위 노드는 끊어진 경로를 복구하기 위해 처음에 경로를 탐색하던 것처럼 RREQ를 브로드캐스트하고 응답을 기다린다. RREQ를 RREQ_RETRIES 시도 후에도 RREP를 받지 못하면 RERR 메시지를 소스 노드와 경로 상에 있는 노드들에게 전송하여 경로를 해제한다.

지역 경로 복구를 위한 RREQ 메시지가 목적지 노드로 도달하면 목적지 노드는 일련번호를 증가 시켜 RREP를 발송한다. 단절 지점의 상위 노드는 RREP를 수신하고 증가된 목적지의 일련번호를 확인하여 경로를 갱신하고 지역 경로 복구를 종료한다.

만약 목적지 노드가 아닌, 목적지로의 경로를 가진 중계 노드가 RREQ를 수신하고 RREP를 발송했다면 목적지의 일련번호가 증가하지 않았기 때문에 홉 수를 비교하게 된다. 홉 수가 이전에 알려진 경로의 홉 수보다 작다면 지역 경로 복구를 완료하며, 그렇지 않다면 이 RREP에 대한 정보를 무시하게 되어 지역 경로 복구가 완료되지 않는다.

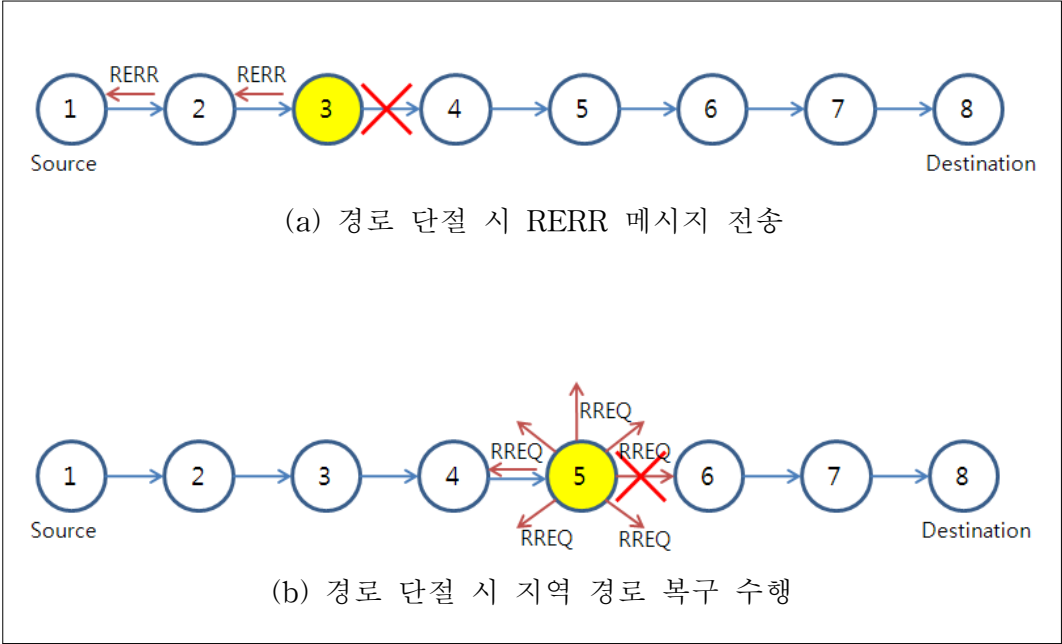


그림 9. 경로 상의 단절 처리
 Fig 9. Processing for broken links

제3장 고장 허용 라우팅 프로토콜

본 연구에서는 AODV 기반 고장 허용 라우팅 프로토콜(FT-AODV: Fault-Tolerant Routing Protocol)을 제안한다. 모바일 애드혹 네트워크에서 노드는 배터리가 소진되었거나 기타 고장을 일으키는 경우도 일어난다. 단일 노드의 고장이나 경로의 단절은 경로 재탐색에 따르는 전송 지연 시간과 라우팅 오버헤드의 증가를 가져왔다. 따라서 얼마나 빨리 노드의 고장이나 경로의 단절에 효과적으로 대처하는가 하는 것이 네트워크의 성능 향상에 기여할 수 있다. 노드의 고장 확률을 p 라고 하면 두 개 노드가 동시에 고장 날 확률은 p^2 이 된다. 만약 노드 고장 확률이 1% ($p = 0.01$)라고 가정하면, 두 개 노드가 동시에 고장 날 확률은 0.01% ($p^2 = 0.0001$)로 현저하게 낮아진다.

본 연구는 경로 상의 노드의 고장을 허용하기 위해 단절된 지점의 상위 노드가 단절된 지점의 하위 노드를 알고 있어야 한다. 이 정보를 이용하여 단절된 지점의 상위노드로부터 1-hop간의 거리에 있는 노드 중 단절된 지점의 하위노드를 이웃으로 가진 노드를 탐색한다. 단절된 지점의 하위노드를 탐색하기 위해서 모든 노드가 자신의 다음 홉뿐만 아니라 경로상의 2-hop 거리에 있는 노드의 정보를 알아야 한다. 이를 위해 RREP 패킷을 이용해 초기경로가 설정 될 시 RREP 패킷의 수정을 통해 하위 방향으로 2-hop간의 노드를 기록하여 이 정보를 이용하여 단절된 지점의 하위노드를 찾는다. 또한, 단절된 지점의 하위노드를 찾기 위해 RSAL (Route Salvage) 패킷을 제안하였다.

따라서 단일 노드 고장에 대한 고장 허용 라우팅을 제공하여 네트워크의 신뢰성 및 전송 성능이 크게 향상되었다.

제1절 역 경로 설정

그림 10에서 RREQ가 목적지인 노드 D까지 전달이 되면 노드 D는 RREQ가 왔던 경로를 따라 RREP 메시지를 발신지로 전송하는데 이 때 RREP에는 수신 노드의 현재 일련번호와 수신노드까지의 홉 수를 기록하여 유니캐스트로 전송을 한다. 이것은 기존의 AODV에서의 역 경로를 설정하는 과정이다. 기존의 AODV 프로토콜에서는 모든 경로를 전부 저장하고 있는 DSR 프로토콜과는 다르게 가볍게 설계되었기 때문에 패킷이 도착할 목적지 노드와 자신의 하위 방향으로의 다음 홉만 알고 있으며 또한 지역 경로 복구 시 노드 a가 RREQ를 목적지 노드로 다시 브로드캐스트하기 때문에 종단 간 전달 시간의 지연과 높은 오버헤드를 가지고 있다.

그림 10과 같이 제안하는 방법은 표 3과 같은 경로 테이블을 생성한다. 노드 b가 배터리가 소진되었거나 다른 곳으로 이동하는 등의 모바일 환경에서 쉽게 일어날 수 있는 고장으로 인해 전체 경로 중 노드 a에서 노드 c로의 구간이 단절되었을 경우, 역 경로가 설정될 때 노드 b가 자신의 다음 홉이 노드 c라는 사실을 RREP 패킷에 실어서 노드 a에게 보내어 노드 a가 단절된 지점인 노드 b 이후에 노드 c가 있다는 사실을 이미 알 수 있게 하여 RREQ 패킷을 목적지 노드인 D까지 브로드캐스팅을 하지 않고, 단지 1-hop 간에 있는 이웃 노드들에게 경로 상에 있는 노드 c를 이웃으로 가진 노드가 있는지를 확인하는 메시지를 전송한다. 이를 위해 그림 11과 같이 기존의 RREP 패킷에 32 bit Successor IP Address 필드인 Downstream IP Address를 추가하여 새로 추가된 필드에 노드 b의 다음 홉인 노드 c를 기록하고 수정된 RREP를 수신한 경우 갱신될 경로 테이블에도 Downstream IP Address를 기록할 필드인 Downsrc를 추가하였다.

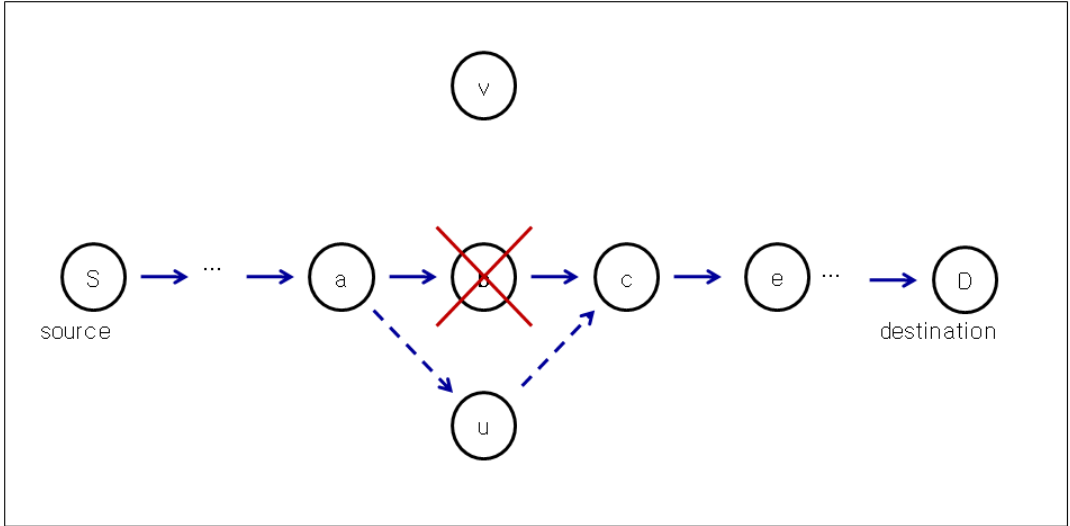


그림 10. 경로의 단절에 따른 복구 과정
 Fig 10. An example of local repair

표 3. 수정된 RREP에 의한 경로 테이블

Table 3. Routing table resulted from the modified RREP packet

Node	Destination	Nexthop	Downstream
a	D	b	c
b	D	c	e
u	D	c	e

그림 11은 고장 허용 라우팅 프로토콜을 위한 기존의 AODV의 RREP 패킷을 수정한 것으로, 그림 10에서 노드 a의 다음 홉인 노드 b의 고장이나 이동으로 인해 경로 a-b-c가 단절되었을 경우 노드 a에게 경로 상에 참여하는 노드 b의 다음 홉인 노드 c를 알게 하여 경로가 단절되었을 경우 노드 c를 이웃으로 가진 1-hop 간의 거리에 있는 노드들을 확인할 수 있도록 기존의 RREP 패킷에 32 bit successor IP address 필드를 추가하였으며 새로 추가된 downstream IP address 필드에 노드 b의 다음 홉을 기록한다.

RREQ는 경로 탐색을 위한 패킷으로 특별한 수정이 필요하지는 않다. 경로를 최초로 탐색할 때 소스 노드는 RREQ 패킷을 브로드캐스트하고 목적지 노드, 혹은 목적지로의 경로를 가진 중계노드가 RREQ 패킷을 수신하면 RREQ 패킷의 응답으로 RREP 패킷을 생성하여 소스노드로 전송하는데 이 때 역 경로가 생성이 된다. 각 노드들은 RREP 패킷을 수신하면 자신의 경로 테이블에서 목적지 노드로 가는 정보들을 추가 또는 갱신한다. 이 때, 각 노드들이 RREP를 상위 방향으로 전송하기 위하여 RREP 패킷에 정보를 기록하면서 추가로 목적지 노드로의 다음 홉을 추가된 필드인 downstream IP address에 함께 기록하여 전송한다. 패킷을 받은 노드는 내용을 확인하여 자신의 다음 홉과 다음 홉이 단절 되었을 경우 RSAL 패킷을 통해 요청할 downstream IP address를 자신의 경로 테이블에 기록한다.

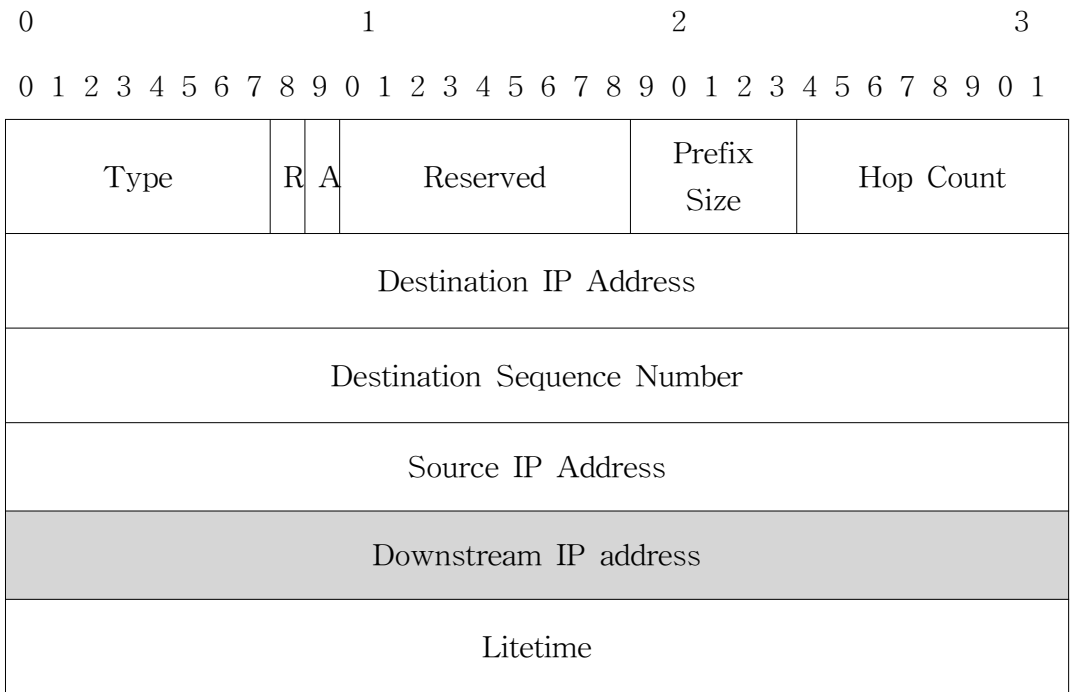


그림 11. 수정된 RREP 패킷의 구조
 Fig 11. Modified RREP message format

제2절 우회 노드 탐색

경로가 단절 되었을 때 단절된 지점의 상위 노드 a로부터 1-hop 간의 거리에 있는 노드 a의 이웃 노드 중 경로에 참여하는 노드 c를 이웃으로 가진 노드가 있는지를 확인하는 패킷을 전송하기 위해 RSAL 패킷을 제안한다. 노드 v와 노드 u는 노드 a의 1-hop 간의 전송 범위 이내에 있으며 이들은 경로에 참여하고 있는 노드 c를 이웃으로 가지고 있다. 노드 v와 노드 u는 노드 a가 발신한 RSAL 패킷을 수신하고 수신한 즉시 자신의 이웃 테이블에 경로 상에 참여하고 있는 노드 c가 있는지를 검사한다. 만약 이웃 테이블에 노드 c가 존재하지 않는다면 RSAL 패킷은 버려진다. 그러나 노드 c를 자신의 이웃으로 가지고 있다면 노드 v와 노드 u는 받은 RSAL의 패킷 내용 중 목적지 노드를 참조하여 자신의 경로 테이블에서 목적지 노드로 가는 경로가 있는지를 확인하여, 존재하지 않는다면 자신의 경로 테이블에 RSAL에 기록된 목적지 노드를 새로 추가하고, 자신의 경로 테이블에 해당 목적지 노드가 있으면 경로 테이블에 새로 추가할 필요는 없다. 그 다음, 자신의 역 경로 엔트리에 upstream 방향의 다음 노드를 노드 a로 수정하고 RSAL에 포함된 목적지인 노드 D로의 다음 홉을 노드 c로 수정한다. 그리고 RSAL에 대한 응답 메시지로 RREP 패킷을 자신의 상위 방향인 노드 a로 전송하여 끊어진 지점을 복구하게 된다.

그림 12는 고장 허용 라우팅 프로토콜의 RSAL 패킷의 구조를 나타낸 것으로 AODV의 패킷 유형은 5로 설정하고, 목적지 노드의 주소는 데이터가 최종적으로 도달해야 하는 목적지 노드의 IP 주소로 그림 10에서의 노드 D에 해당하고, 발신지 노드의 IP 주소는 처음에 데이터가 생성되는 소스 노드의 주소로 노드 S에 해당하며, 단절된 지점의 상위 지점인 originator IP address는 단절된 지점의 상위 노드인 노드 a, 그리고 단절된 지점의 하위 노드의 주소인

downstream IP address는 단절된 지점의 다음 홉인 노드 c가 된다.

그림 10에서와 같이 노드 b의 고장으로 인해 경로가 단절되었을 경우 노드 a는 RSAL패킷에 다음과 같은 내용을 포함하여 자신의 주변의 1-hop 거리에 있는 노드들에게 전송한다. RSAL을 수신한 노드 u와 노드 v는 먼저 자신의 이웃테이블에서 RSAL에서 요청하는 downstream IP address인 노드 c가 자신의 이웃인지를 검사한다. 만약 노드 c가 자신의 이웃이 아니라면 수신된 RSAL은 폐기된다. 반대로 노드 c를 자신의 이웃으로 가지고 있으면 노드 u와 v는 그 다음단계로 RSAL 패킷에서 요청하는 목적지 노드가 자신의 경로 테이블에 존재하는지를 확인한다. 만약 자신의 경로 테이블에 RSAL 패킷에서 요청하는 목적지 노드가 존재하지 않는다면 자신의 경로 테이블에 새로이 RSAL 패킷에서 요청하는 목적지를 추가하고 존재한다면 경로의 유효 시간을 갱신한다. 그리고 RREP 패킷을 노드 a에게 발송하여 경로가 자연스럽게 복구되도록 한다.

그림 10에서는 노드 u와 노드 v가 함께 이웃 노드로 노드 a, b, c를 가지고 있으며 노드 a로부터의 거리 또한 같으므로 각각 RSAL 패킷을 수신하고 RSAL 패킷에 대한 응답으로 노드 a에게 RREP 패킷을 발송한다. 이 때 노드 a는 기존의 AODV에서 같은 패킷을 처리하는 방법과 같이 홉 수와 일련번호를 가지고 도착한 RREP 패킷을 처리하게 된다.

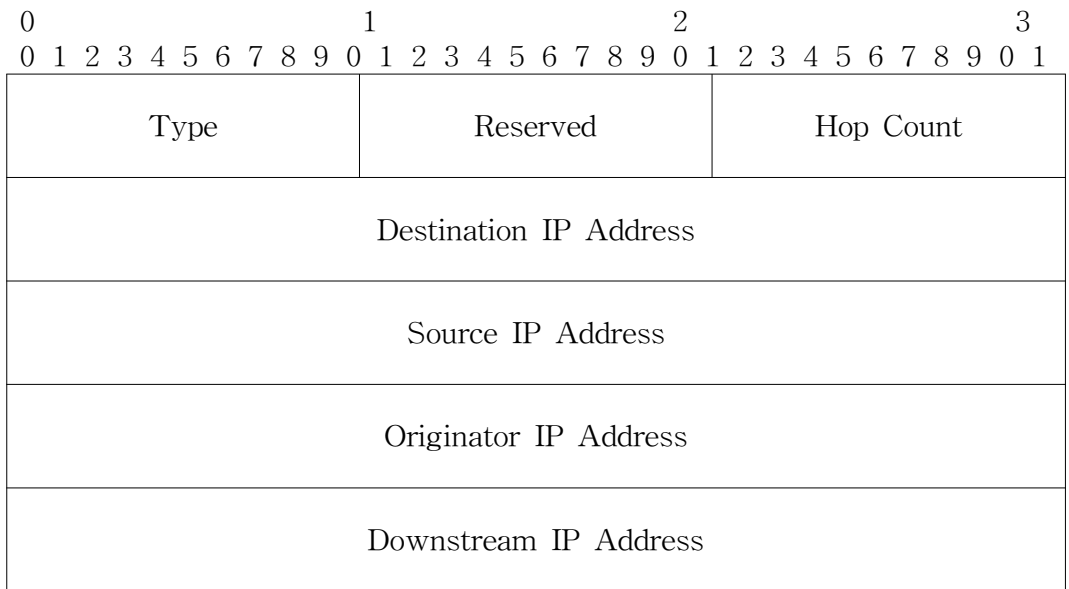


그림 12. 고장 허용 라우팅 프로토콜의 RSAL 패킷 구조
 Fig 12. RSAL message format for FT-AODV

제4장 성능 평가

제1절 모의실험 환경

본 연구에서 제안한 고장 허용 라우팅 프로토콜의 성능을 측정하기 위해 미국 버클리 대학에서 개발한 네트워크 시뮬레이터 NS-2.32 [13][15]를 사용하였다. NS-2는 네트워크의 성능 분석을 목적으로 하는 이벤트 기반의 시뮬레이터로 C++를 기반으로 하며 스크립팅을 위해 OTcl을 사용한다.

NS-2는 TCP, UDP, FTP, HTTP 등의 TCP/IP family와 라우팅 프로토콜, 모바일 애드혹 네트워크, WLAN, Mobile-IP와 Cellular 네트워크 등 시뮬레이션의 범위가 다양하며 소스코드가 공개되어 있기 때문에 누구나 수정하여 사용이 가능하다. NS-2 시뮬레이터는 현재도 계속해서 기능이 추가되고 있으며 성능 개선 작업도 진행되고 있다.

성능 개선을 확인하기 위하여 NS-2에 기본 탑재된 AODV 프로토콜과 본 연구에서 제안하는 FT-AODV 프로토콜을 비교하였다. 모의실험은 300×1500m²의 네트워크 내에서 50개의 노드를 임의의 위치에 배치시키고 세션의 수와 전송률을 일정하게 유지하면서 노드의 이동속도와 관계된 pause time을 변경하는 실험과 노드의 이동 속도와 세션의 수는 일정하게 유지하면서 초당 전송률을 달리하는 실험을 각각 50회 이상 진행하였으며 이에 대한 모의실험 환경은 표 4와 같다.

표 4. 모의실험 환경

Table 4. Simulation parameters

Parameter	Value
Simulator	NS-2.32
Topology size	300×1500 m ²
MAC type	IEEE 802.11
Propagation channel model	Two-ray ground reflection
Antenna model	Omni-directional antenna
Traffic source	Constant bit rate (CBR)
Number of nodes	50
Mobility model	Random waypoint model - Pause time: 0, 20, 50, 100, 300, 600, 900 (default: 100) secs - Speed: 0 ~ 10 m/s
Number of sessions	6
Transmission rate	1, 2, 4, 6, 8 (default: 4) packets/sec
Simulation time	900 secs (15 mins)

실험은 300×1500m²의 네트워크에 50개의 노드를 무작위로 배치시키고 총 6개의 세션이 연결되며 초당 4개의 패킷을 보내면서 노드의 정지 시간에 변화를 주는 실험과 노드의 정지 시간을 100초로 설정하고 총 6개의 세션이 연결되도록 하며 초당 전송하는 패킷의 수에 변화를 주는 실험으로 진행하였으며 각 실험마다 패킷 전달율, 데이터의 전달 시간 그리고 라우팅 오버헤드를 측정하였다.

패킷 전달율은 송신된 전체 CBR (Constant Bit Rate) 트래픽 중 목적지까지 도착한 데이터 패킷의 비율로 계산하며 식은 (1)과 같다.

$$Packet\ Delivery\ Ratio = \frac{Number\ of\ delivered\ packets}{Number\ of\ generated\ packets} \dots\dots\dots(1)$$

$$Average\ end-to-end\ delay = average\ value\ of\ delivered\ packet's\ timestamp - generated\ packet's\ timestamps \dots\dots\dots(2)$$

$$Normalized\ Routing\ Overhead = \frac{Number\ of\ routing\ packet\ transmissions}{Number\ of\ data\ packet\ transmissions} \dots\dots(3)$$

종단 간 지연시간은 경로 재탐색 시 소스 노드로부터 출발한 데이터를 중간 노드가 가지고 있는 buffering 시간, MAC 계층에서의 재전송 시간 그리고 데이터를 목적지까지 전달하는데 소요되는 시간을 포함하는 것으로 식 (2)와 같이 소스 노드로부터 출발한 데이터가 목적지 노드까지 도착하는데 걸리는 평균

시간을 측정하였으며 이 때 유실된 패킷은 평균 시간 측정 대상에서 제외하였다.

정규화된 라우팅 오버헤드는 식 (3)과 같이 전체 라우팅 컨트롤 패킷을 데이터 패킷으로 나눈 값으로 각각의 홉 간 전송을 하나의 전송으로 계산한 것이다. 패킷 전달율과 종단 간 전달 시간은 최선형 트래픽[14]의 측정을 위해 중요한 항목이며 라우팅 오버헤드는 라우팅 프로토콜의 효율성을 측정하기 위한 것이다.

제2절 모의실험 결과 및 분석

그림 13, 14, 15는 random waypoint model을 사용하여 배치된 노드 50개가 6개의 연결을 유지하고 초당 4개의 패킷을 전송하면서 이동 간 정지 시간을 10에서 900초로 변화를 주며 실험한 결과를 나타낸 그래프이다.

그림 13은 노드의 이동성에 따른 패킷의 전달율을 나타낸 것으로 네트워크 내에서 노드의 이동이 안정적일 수록 패킷의 전달율이 상승함을 보였으며 제안하는 FT-AODV의 패킷 전달율이 AODV에 비해 5% 이내의 소폭 향상이 있는 것으로 나타났다.

그림 14는 노드의 이동에 따른 데이터의 중단 간 전달 시간을 나타낸 것으로 네트워크 내에서 노드의 이동이 안정적일 수록 패킷의 전달시간은 감소함을 보였으며 네트워크 내의 노드들의 이동 속도가 증가할수록 데이터의 전달시간은 증가함을 보였다. 실험을 통해 제안하는 FT-AODV의 전달 시간이 AODV에 비해 평균 48% 감소함을 보였으며 이는 노드의 고장이나 경로의 유실로 인해 경로를 다시 탐색할 경우 목적지까지 RREQ 패킷을 전달하지 않고 고장이 난 지점에서 downstream node를 발견함으로써 자연스럽게 고장이 난 지점을 우회하여 통신이 중단되지 않는 데서 기인한다.

그림 15는 노드의 이동에 따른 라우팅 오버헤드를 나타낸다. 실험을 통해 네트워크 내에서 노드의 이동이 안정적일 수록 오버헤드 역시 감소함을 보였으며 네트워크 내의 노드들의 이동 속도가 증가할수록 오버헤드는 증가하였다.

실험을 통해 제안하는 FT-AODV의 오버헤드가 AODV에 비해 평균 23% 감소함을 보였다. 이는 고장이 난 지점에서 경로를 재탐색하기 위한 RREQ 플러딩이 이루어지지 않고 제안한 RSAL 패킷을 이용하여 자연스럽게 경로를 복구하기 때문이다.

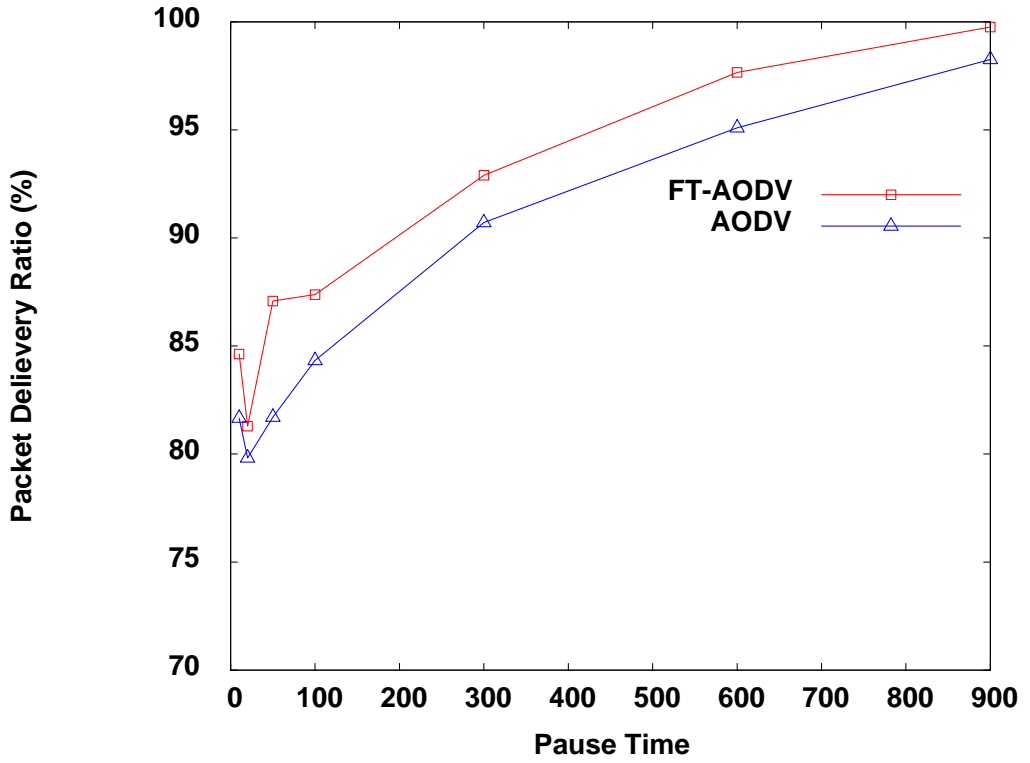


그림 13. 노드 이동성에 따른 패킷 전달율
 Fig 13. Packet delivery ratio versus node mobility

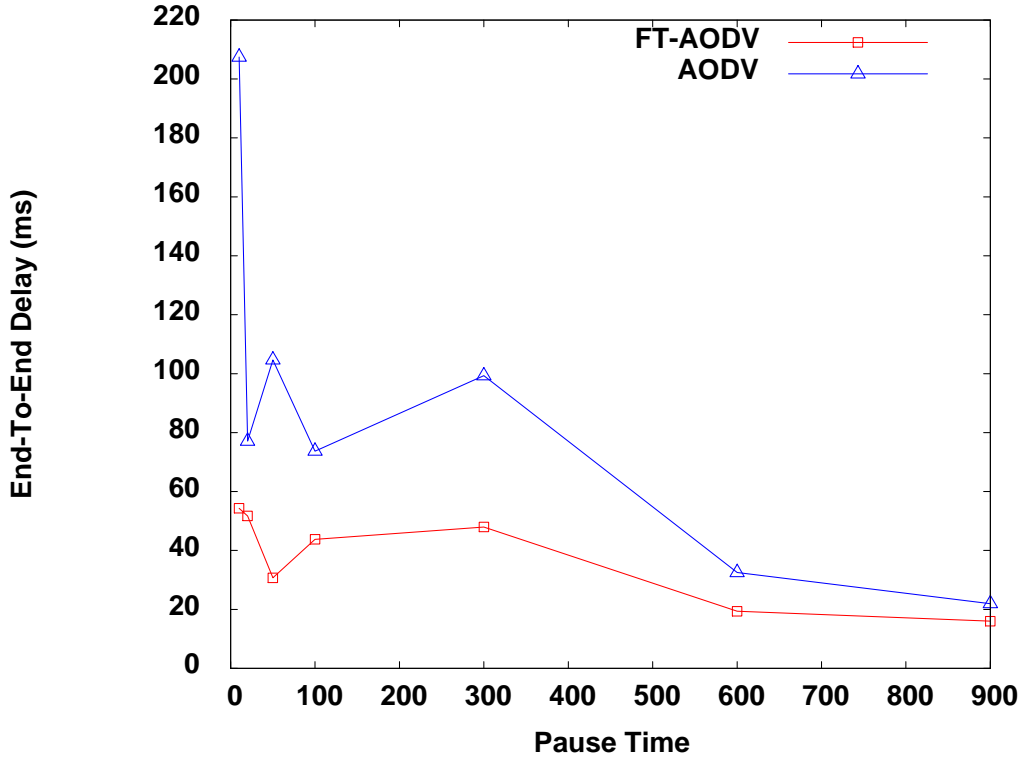


그림 14. 노드 이동성에 따른 중단 간 전달 시간
 Fig 14. End-to-end delay versus node mobility

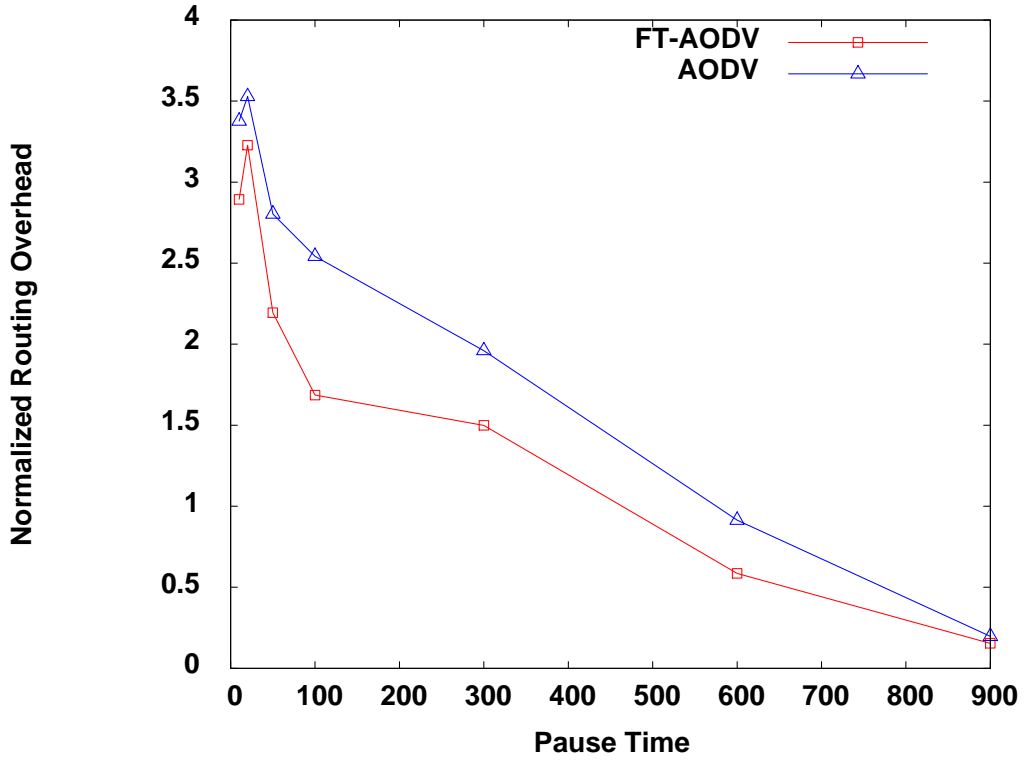


그림 15. 노드 이동성에 따른 라우팅 오버헤드

Fig 15. Routing load versus node mobility

그림 16, 17, 18은 random waypoint model을 사용하여 배치된 노드 50개가 총 실험 900초 중 100초의 정지시간을 갖고 6개의 연결을 유지하며 초당 패킷 전송률을 1에서 8개까지의 변화를 주며 실험한 결과를 나타낸 그래프이다.

그림 16은 데이터 패킷의 전송률에 따른 패킷의 전달율을 나타낸 것으로 초당 전송하는 패킷의 수가 증가함에 따라 전달율은 감소함을 보였으며 이는 많은 패킷이 전송될수록 네트워크 내에 충돌이 빈번히 일어남에 따라 유실되는 패킷이 증가하기 때문이다. 제안하는 FT-AODV의 패킷 전달율이 AODV에 비해 5% 이내의 소폭 향상이 있는 것으로 나타났다.

그림 17은 데이터 전송률에 따른 데이터의 종단 간 전달 시간을 나타낸 것으로 실험을 통해 제안하는 FT-AODV의 전달 시간이 AODV에 비해 평균 30% 감소된 결과를 보였다. 초당 전송하는 패킷의 수가 증가함에 따라 전달 시간은 증가함을 보였으며 이는 많은 패킷이 전송될수록 네트워크 내에 충돌이 빈번히 일어남에 따라 경로가 깨어지거나 변경되는 횟수가 증가함에서 기인하는 것으로 이로 인해 소스 노드로부터 출발하는 패킷이 단절된 지점의 상위 노드인 originator node에 buffering되어 머무는 시간의 총 합이 늘어남에 따라 종단 간 전달 시간도 함께 증가하였다.

그림 18은 데이터 전송률에 따른 라우팅 오버헤드를 나타낸다. 초당 전송하는 패킷의 수가 증가함에 따라 빈번한 경로 단절 및 변경으로 인해 오버헤드가 증가함을 보였으며 제안하는 FT-AODV는 목적지까지 RREQ 패킷을 플러딩하지 않고 1-hop간의 노드들에게 RSAL 패킷을 전송함으로 경로를 복구하기 때문에 경로 재탐색에 따르는 컨트롤 패킷을 줄일 수 있어 기존의 AODV에 비해 오버헤드를 11% 감소시키는 결과를 보여준다.

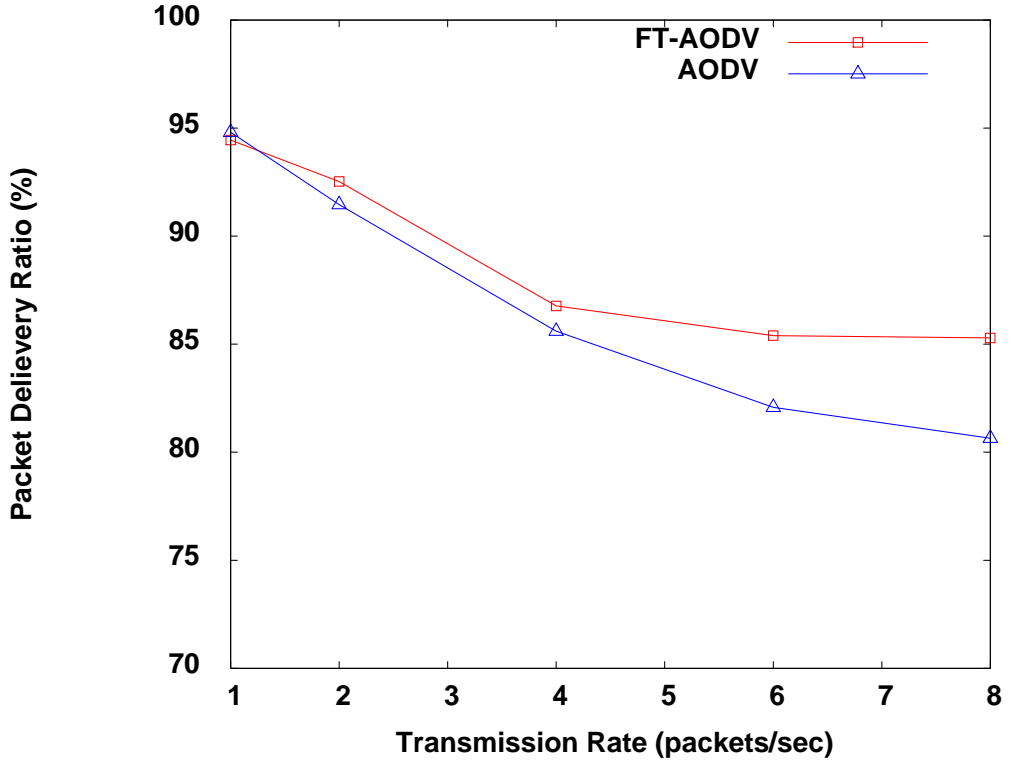


그림 16. 데이터 전송률에 따른 패킷 전달율
 Fig 16. Packet delivery ratio versus transmissstion rate

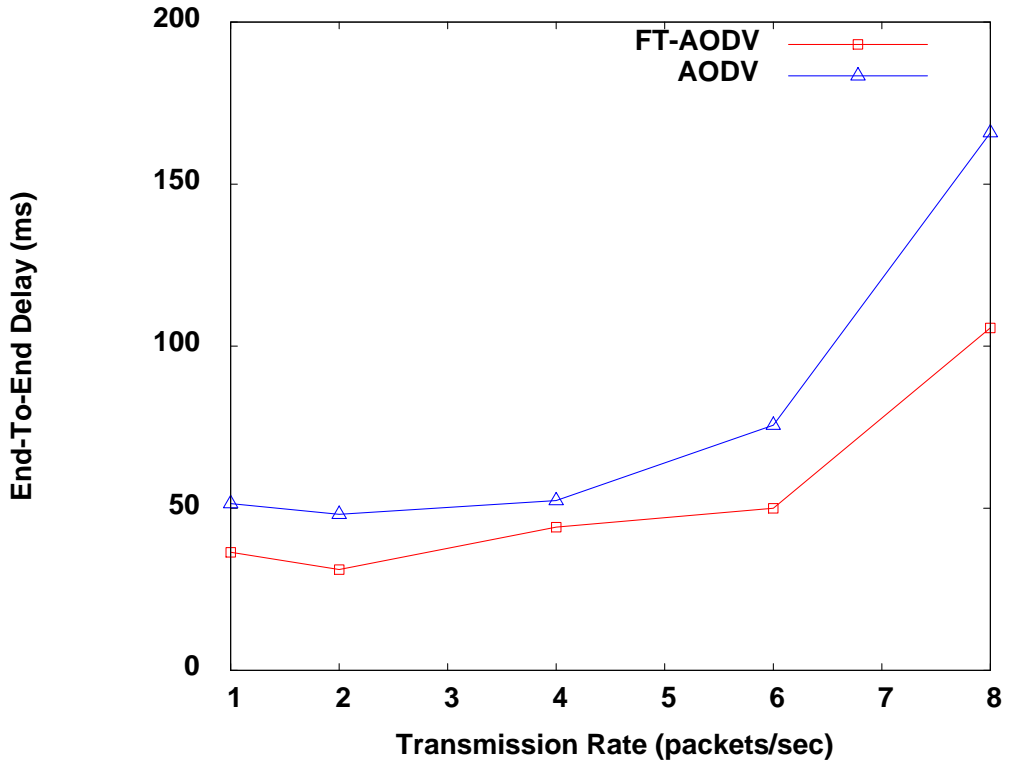


그림 17. 데이터 전송률에 따른 종단 간 전달 시간
 Fig 17. End-to-end delay versus transmission rate

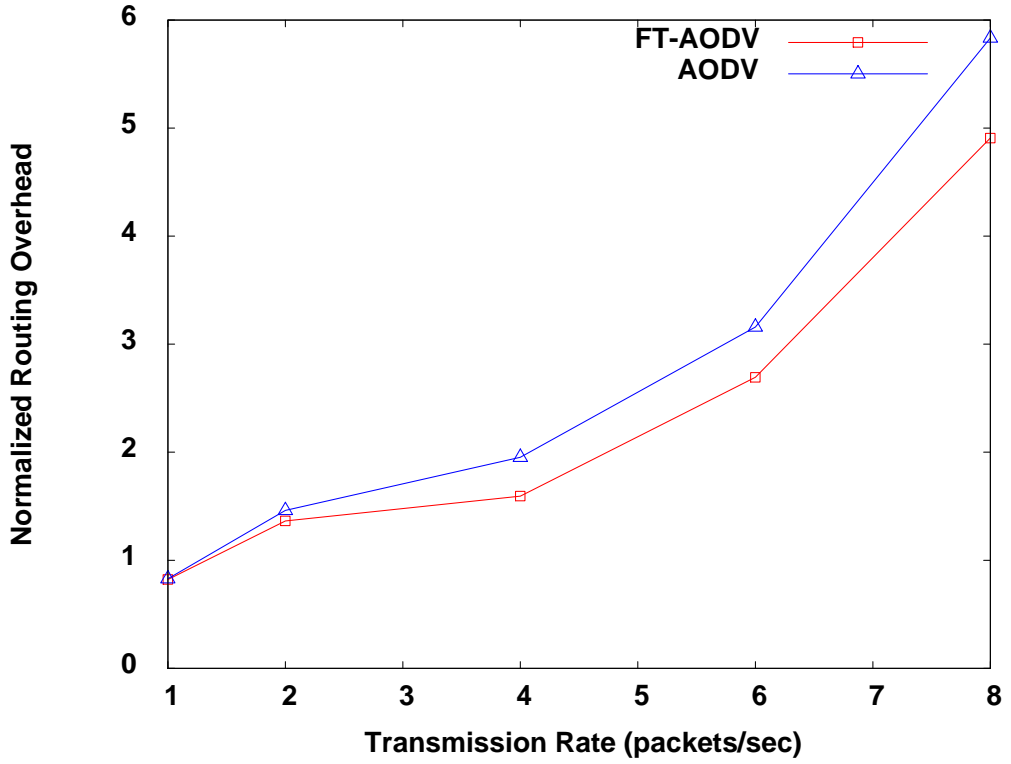


그림 18. 데이터 전송률에 따른 라우팅 오버헤드
 Fig 18. Routing load versus transmission rate

제5장 결 론

본 논문에서는 무선 통신 기술의 발달과 무선 통신 사용자 수의 증가와 더불어 많은 기업체나 공공기관의 무선 네트워크 이용 방식을 변화시키고 있는 모바일 애드혹 네트워크의 라우팅 프로토콜에 대해서 살펴보고 기존의 AODV 프로토콜을 기반으로 하는 고장 허용 라우팅 프로토콜을 제안하였다.

본 논문은 이동하는 특성을 가지는 모바일 애드혹 네트워크에서 단일 노드가 고장을 일으켰을 경우 쉽게 경로가 단절이 되고 이로 인한 데이터의 전달 시간과 경로 탐색에 소요되는 컨트롤 패킷의 증가로 인한 네트워크의 부하가 증가되는 문제를 개선하기 위한 방안을 제시하였다. 목적지와 다음 홉만을 알고 있는 AODV의 구조에서 단절된 지점의 다음 홉을 알게 하기 위하여 RREP 패킷 구조에 32-bit successor IP address를 추가하고 이와 함께 라우팅 테이블에 RREP 패킷에 실려 온 downstream IP address를 기록하기 위해 라우팅 테이블의 구조 또한 수정을 하였다. 또한, 노드의 고장이나 경로의 단절 시 단절된 지점의 상위 노드는 대체 노드를 찾기 위해 본 논문에서 제안한 RSAL 패킷을 1-hop간의 노드들에게 브로드캐스트 하여 대체 노드를 찾아 자연스럽게 통신이 지속될 수 있도록 하였다. 이로 인해 기존의 AODV 프로토콜에서 경로를 재탐색 하기 위해 목적지노드까지 RREQ를 플러딩하는 대신 1-hop간의 노드에 RSAL 만을 전송하므로 라우팅 컨트롤 패킷이 줄어들어 오버헤드가 감소하였으며 데이터의 평균 전달 시간 또한 감소하였다.

제안한 프로토콜의 검증과 성능 평가를 위하여 NS-2 시뮬레이터를 사용하였으며 이를 이용한 모의실험 결과 패킷 전달율은 기존 AODV 대비 5% 이내의 소폭 상승함을 보였고 데이터 패킷의 종단 간 전달 시간은 평균 40%, 라우팅 오버헤드는 평균 13% 감소되었다.

본 논문은 모의실험을 통해 라우팅 오버헤드의 감소를 보였으나 RSAL 패킷에 의한 오버헤드는 여전히 존재한다. 향후 오버헤드의 절감에 관한 연구가 더 이루어져야겠으며, 경로 단절 시 주변 노드들이 promiscuous mode를 통해 데이터의 흐름을 overhearing하고 스스로 복구하도록 하여 전반적인 성능을 향상시키는 것이 그 다음 과제라 할 수 있다.

참 고 문 헌

- [1] G. Grube and H. Ahmadi, "The Role of Ad-hoc Technology in the Broadband Wireless Networks of the Future," *Wireless Communications, John Wiley & Sons, Ltd*, pp. 133-154, Jan. 2007.
- [2] C. Cordeiro and P. Agrawal, "Ad hoc & Sensor Networks: Theory and Applications," *World Scientific Publishing Company*, pp. 19-79, May 2006.
- [3] T. M. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction To Algorithms Second Edition," *MIT Press and McGraw-Hill, ISBN 0-262-03293-7*, pp.588 - 592, Sept. 2001.
- [4] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *ACM SIGCOMM Computer Communication, Volume 24, Issue 4*, pp. 234-244, Oct. 1994.
- [5] R. Ogier, F. Templin and M. Lewis, "Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)," *IETF RFC3684*, Feb. 2004.
- [6] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," *IETF RFC3626*, Oct. 2003.
- [7] D. B. Johnson, D. A. Maltz and Y. C. Hu "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," *IETF Internet Draft*, Apr. 2003.
- [8] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," *Second IEEE Workshop on Mobile Computing Systems and*

- Applications*, pp. 90-100, Feb. 1999.
- [9] E. M. Royer and C. K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *IEEE Personal Communications*, pp. 46-55, Apr. 1999.
- [10] C. E. Perkins, E. M. Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," *IETF RFC 3561*, July 2003.
- [11] V. Park and S. Corson, "Temporally-Ordered Routing Algorithm (TORA) version 1: Functional specification," *IETF Internet*, July 2007.
- [12] M. Pan, S. Y. Chuang, and S. D. Wang, "Local Repair Mechanisms for On-Demand Routing in Mobile Ad Hoc Networks," *Proc. of 11th Pacific Rim Int. Symp. on Dependable Computing*, pp. 317-324, Dec. 2005.
- [13] NS-2.32 simulator, <http://www.isi.edu/nsnam/ns>, 2008.
- [14] C. E. Perkins, E. M. Royer, S. R. Das and M. K. Marina, "Performance comparison of two on-demand routing protocols for ad hoc Networks," *IEEE Personal Communications*, pp. 16-28, Feb. 2001.
- [15] 박성현, 이승형, "NS-2를 이용한 네트워크 시뮬레이션 방법론," 한국통신 학회지 제24권 제7호, pp. 79-88, 2007.

저작물 이용 허락서

학과	컴퓨터공학과	학번	20077107	과정	석사
성명	한글 조 충 용	한문 趙 忠 龍	영문 Chungyong Cho		
주소	광주광역시 서구 내방동 834-20				
연락처	E-mail : wings1999@gmail.com				
논문 제목	한글 모바일 애드혹 네트워크를 위한 고장 허용 라우팅 프로토콜				
	영문 A Fault-Tolerant Routing Protocol for Mobile Ad Hoc Networks				

본인이 저작한 위의 저작물에 대하여 다음과 같은 조건 아래 조선대학교가 저작물을 이용할 수 있도록 허락하고 동의합니다.

- 다 음 -

1. 저작물의 DB구축 및 인터넷을 포함한 정보통신망에의 공개를 위한 저작물의 복제, 기억장치에의 저장, 전송 등을 허락함.
2. 위의 목적을 위하여 필요한 범위 내에서의 편집과 형식상의 변경을 허락함. 다만, 저작물의 내용변경은 금지함.
3. 배포·전송된 저작물의 영리적 목적을 위한 복제, 저장, 전송 등은 금지함.
4. 저작물에 대한 이용기간은 5년으로 하고, 기간종료 3개월 이내에 별도의 의사 표시가 없을 경우에는 저작물의 이용기간을 계속 연장함.
5. 해당 저작물의 저작권을 타인에게 양도하거나 출판을 허락을 하였을 경우에는 1개월 이내에 대학에 이를 통보함.
6. 조선대학교는 저작물 이용의 허락 이후 해당 저작물로 인하여 발생하는 타인에 의한 권리 침해에 대하여 일체의 법적 책임을 지지 않음.
7. 소속 대학의 협정기관에 저작물의 제공 및 인터넷 등 정보통신망을 이용한 저작물의 전송·출력을 허락함.

동의여부 : 동의(0) 반대()

2009년 2월

저작자: 조 충 용 (인)

조선대학교 총장 귀하