

MANET에서 Interrupt message를 이용한 AODV 라우팅 프로토콜의 성능 개선

이 윤 경*, 김 주 균^o

Performance Enhancement of AODV Routing Protocol Using Interrupt Message in MANET

Yun-kyung Lee*, Ju-gyun Kim^o

요 약

MANET 환경에서 AODV는 요구기반 방식으로써의 장점을 갖지만 다른 라우팅 프로토콜들에 비해 제어 패킷 오버헤드가 크다는 단점도 가지고 있다. 본 논문은 로컬 연결의 탐지와 이웃 목록을 유지하기 위해 주기적으로 브로드캐스트하는 Hello message로 인한 문제점을 개선한다. 주기적인 Hello message는 패킷 전달률과 제한된 대역폭의 효율성을 감소시키고, 제어 패킷 오버헤드를 증가시켜 잔존 에너지량과 네트워크 수명을 감소시킨다. 또한 AODV에서의 주기적인 Hello message는 “Reactive”하지 않은 방식이기 때문에 많은 논쟁의 근원이었다. 이러한 문제점을 개선하기 위해 본 연구에서는 인터럽트 방식을 이용하여 주기적인 Hello message를 제거함으로써 제어 패킷 오버헤드를 줄여 성능을 개선하였다. 기존의 AODV와 제안된 mod_AODV의 성능은 QualNet 5.0을 이용하여 실험 분석하였고, 실험 결과는 시뮬레이션 파라미터들의 변화에 대한 모든 성능 매트릭들에서 mod_AODV가 전반적으로 우수한 성능을 보였다.

Key Words : MANET, AODV routing protocol, Hello message, Interrupt

ABSTRACT

In MANET, AODV(Ad hoc On-demand Distance Vector) has its advantages as on-demand approach but it also has a disadvantage that the control packet overhead is high compared to other routing protocols. This paper improves the problem caused by Hello messages that are broadcasted periodically to detect the local connectivity and maintain neighbor list. Periodic hello messages reduce the Packet delivery ratio and the efficiency in the limited bandwidth. And its increased Control packet overhead leads to decrease the Residual battery capacity and the Network lifetime. Further, non-reactive nature of periodic hello messages in AODV has also been the source of numerous controversies. In order to solve these problems, this paper improves the performance by using the interrupt driven approach which removes periodic hello messages and decreases the Control packet overhead. Performance comparisons between the traditional AODV and proposed mod_AODV done with network simulator QualNet 5.0 show that the mod_AODV performs better in most performance metrics under scenarios with various values of simulation parameters.

* First Author : 숙명여자대학교 컴퓨터과학과 운영체제 연구실, lyk4707@sookmyung.ac.kr, 학생회원

^o Corresponding Author : 숙명여자대학교 컴퓨터과학과 운영체제 연구실, jgkim@sookmyung.ac.kr, 정회원
논문번호 : KICS2013-07-288, 접수일자 : 2013년 7월 8일, 최종논문접수일자 : 2013년 10월 4일

I. 서 론

무선 네트워크는 그림 1과 같이 기존 통신 인프라(Base Station, Access Point)의 지원여부에 따라서 인프라가 있는 네트워크(Infrastructure)와 인프라가 없는(Infrastructureless, Ad hoc) 네트워크로 분류한다.

인프라가 있는 네트워크는 이동 단말들이 기존에 설치된 인프라의 도움으로 통신을 하는 구조로, 이동 전화망, 무선 LAN 등을 예로 들 수 있고, 자연 재해, 전쟁 등의 상황에서 기존 인프라의 고장이나 유선 단절 시 통신을 할 수 없다는 단점이 있다.

인프라가 없는 네트워크인 애드 혹 네트워크(Ad hoc network)에서 이동 단말들은 기존 인프라의 도움 없이 데이터를 송수신하는 주체로서의 호스트와 라우터로서의 역할을 병행하는 다중적인 역할을 담당하게 되고, 스스로 구성 되어 지면서 동적으로 변화되는 다중-홉 구조이고, 임시 구성용 네트워크며 자연 재해, 재난 지역 그리고 전쟁터 같은 인프라 없는 환경에 적용하도록 연구되어왔다^[1]. 인터넷의 급속한 성장으로 애드 혹 통신망은 근거리 지역망에서도 상업적으로 주목받기 시작했고, 그 예로 Bluetooth, HomeRF 등이 있다^[2]. 또한 무선 사설망에도 애드 혹 통신망 기술을 적용하기 위한 기술개발과 연구가 활발히 진행되었고, 이동 단말 사이의 효율적인 통신 제공을 위해 최적의 라우팅 기법이 요구되어 IETF(Internet engineering task force) MANET(Mobile ad hoc network) working group의 연구는 라우팅 프로토콜(Routing protocol) 분야로 집중되었다^[1].

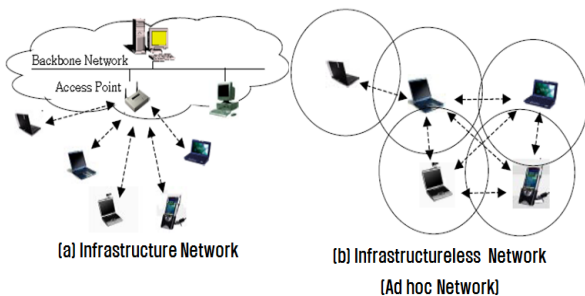


그림 1. 무선 네트워크의 분류
Fig. 1. Classification of a wireless network

이동 애드 혹 네트워크 (MANET:Mobile ad-hoc network)는 이동노드 간의 연결성, 전파 상태, 트래픽 및 사용자 이동 패턴에 따라 네트워크 토폴로지가 끊임없이 변화하므로 네트워크의 구성과 유지가

어렵고, 제반 기술의 적용이 용이치 않기 때문에 이동 애드 혹 네트워크에서의 라우팅 프로토콜은 이러한 동적인 네트워크 토폴로지의 변화에 신속하게 대처할 수 있도록 효율적으로 개선되어야 한다^[3]. 따라서 이동 애드 혹 네트워크에서의 잦은 위상변화에 대해 사용하지도 않는 경로발견으로 활용 가능한 자원의 많은 부분을 소비하는 테이블 관리 방식은 적합하지 않고, 요구 기반 방식(On-demand, Reactive)의 라우팅 프로토콜(DSR, AODV 등)이 더 적합하다고 할 수 있다^[4,5].

최근 스마트 폰의 활성화로 사용자가 이동 중에 애드 혹 네트워크를 형성해 데이터 통신 서비스를 이용하는 경우가 증가하고 있기 때문에 이러한 이동 애드 혹 네트워크 환경에 적합한 라우팅 프로토콜들 간의 성능 비교분석이 필요하다. 선행된 여러 라우팅 프로토콜들 간의 성능 비교 연구에 따르면 다양한 이동 패턴(Mobility pattern), 밀도(Density) 그리고 트래픽(Traffic)에서 요구 기반 방식의 AODV 라우팅 프로토콜이 패킷 수신률(Packet Delivery Fraction)이나 시스템 효율(Throughput)에서 좋은 성능을 보였고, 특히 많은 트래픽(High traffic)과 높은 이동성(High mobility)을 갖는 경우에도 AODV가 대체로 우수했음이 많은 연구들을 통해 제시되고 있다^[5-14].

AODV 라우팅 프로토콜은 연결 요청이 있을 때 경로정보를 획득하기 때문에 노드의 이동성에 관한 정확한 정보를 반영할 수 있어 데이터 전송률에서 테이블 기반 방식에 비해 우수함을 보이고, 패킷 헤더에 전체 경로를 포함하지 않아 같은 요구 기반 방식의 DSR의 단점인 소스 라우팅의 자원낭비 문제점을 개선하였지만, 목적지까지 하나의 경로만 유지하고, 획득한 경로가 변하지 않더라도 데이터 패킷이 전송되지 않으면 일정 시간 경과 후 해당 경로 정보를 삭제하므로 발생하는 제어 패킷의 수는 DSR보다 훨씬 많기 때문에 이 문제점에 대한 개선이 필요하다^[1].

개선을 하기 위한 애드 혹 네트워크 라우팅 프로토콜은 다음과 같은 문제를 고려해야 한다^[1].

첫째 애드 혹 노드들은 제한된 대역의 무선링크로 연결된다. 즉, 유선환경의 고정네트워크에 비해 훨씬 적은 용량의 대역을 사용하고, 다중접근, 페이딩, 잡음, 그리고 간섭 등의 무선특성까지 고려하면 최대 전송률 보다 훨씬 낮은 제한된 대역을 사용하게 되므로 이를 최대한 활용하기 위하여 제어패킷 오버헤드는 작아야 한다.

둘째 애드 혹 노드들은 저장과 배터리 용량 등의 자원이 제한되는 랩탑이나 이동 장치가 될 것이므로 과도한 플로딩(Flooding)의 사용과 주기적인 메시지는 최소화 하여야 한다.

셋째 애드 혹 노드들은 동적인 네트워크 토폴로지를 갖기 때문에 이러한 동적인 위상변화를 빠르게 반영하고 대체경로를 사용하여야 한다.

AODV의 제어패킷은 경로 탐지(Route discovery)를 위한 RREQ(Route request), RREP(Route reply) 등과 경로유지(Route maintenance)를 위한 RERR(Route error), 그리고 이웃 탐지 기법(Neighbor discovery method)으로 주기적인 Hello message 등이 사용되며 대부분의 선행 연구들이 경로 탐지에서의 RREQ와 RREP를 줄이기 위해 많은 노력을 해왔다.

요구 기반 방식의 AODV에서는 모든 노드가 주기적인 Hello message를 전송하는데 이것은 이웃(Neighbor) 노드들과의 로컬 연결에 대한 탐지와 이웃 목록(Neighbor list)을 유지하기 위한 것이다. 이러한 목적에서 주기적으로 브로드캐스트하는 Hello message에 의해 발생 가능한 문제점을 살펴보면 우선 각 이동 노드들이 주기적으로 전송하는 Hello message로 인해 제한된 대역에서 충돌, 간섭 등의 발생으로 패킷 손실을 유발해 패킷 전달률과 제한된 대역폭의 효율성을 감소시키는 것이다.

다음으로 주기적인 메시지로 인한 제어패킷 오버헤드의 증가는 잔존 에너지량(Residual battery capacity)의 감소와 이로 인한 전체 네트워크 수명(Network lifetime)을 감소시킨다.

마지막으로 대표적인 요구 기반 방식인 AODV에서 주기적인 Hello message는 “Reactive”하지 않은 방식이기 때문에 논쟁의 쟁점이 되어 왔다는 것이다.

따라서 본 연구에서는 요구 기반 방식의 AODV 라우팅 프로토콜의 제어 패킷 중 주기적인 Hello message로 인해 발생하는 문제점들을 해결하기 위하여 로컬 연결 탐지 기능과 이웃 목록의 유지 기능을 위해 인터럽트 방식을 도입하였다. 즉, 제안하는 개선된 mod_AODV는 인터럽트 방식을 적용하여 각 노드는 로컬 연결에 영향을 줄 수 있는 상황이 발생하면 해당하는 종류의 Interrupt message(START, S_ACK, LACK, MOVE)를 이웃 노드에게 지역적으로 브로드캐스트해 이웃 노드가 로컬 연결을 탐지하고 정확한 이웃 목록을 유지할 수 있도록 한다.

또한 [16]에서는 네트워크에서 발생하는 실패(Failure)를 두 가지로 분류하였는데, 갑작스런 물리적 고장(Physical fault)이나 단절(Disconnection)에 의한 예측 불가능한 실패(Unpredictable failure)와 노드의 이동 혹은 부족한 전력 등에 의한 예측 가능한 실패(Predictable failure)이다. 본 연구에서는 이 두 가지 분류 중 예측 가능한 실패를 연구 범위로 한정하였으며 이를 기반으로 로컬 연결에 영향을 줄 수 있는 상황의 종류와 이에 해당하는 Interrupt message들을 함께 정의하였다.

Interrupt message를 이용해 성능을 개선한 mod_AODV의 성능을 실험 분석하기 위해 QualNet 5.0을 이용하였고, 전체 노드 중 이동 노드의 비율, 최대 속도(Maximum velocity), 메시지 주기(Message interval), 휴지 시간(Pause time), 대역폭(Bandwidth) 그리고 트래픽을 증가시켜가면서 해당 시뮬레이션 파라미터들의 변화가 성능 매트릭들(Performance metrics)에 어떠한 영향을 미치는지를 기준으로 AODV와 mod_AODV의 성능을 비교 분석하였다.

실험에서 사용된 성능 매트릭은 IETF(Internet Engineering Task Force) MANET Working Group에 의해 제안된 패킷 전달률(Packet delivery ratio), 평균 종단간 지연(Average end-to-end delay) 그리고 제어 패킷 오버헤드(Control packet overhead)가 사용되었으며, 그 외에도 평균 종단간 지연과 밀접한 관련이 있는 단절된 링크의 수(Number of broken link), 네트워크 수명에 관련이 있는 잔존 에너지량 등을 사용하였다.

본 논문의 구성은 다음과 같다. II. 관련 연구에서는 이동 애드 혹 네트워크 라우팅 프로토콜들의 종류와 프로토콜들 간의 성능 비교에 관한 선행 연구들을 정리하였으며 III.에서는 Interrupt messages를 이용한 개선된 AODV에 대해 기술하고, IV. 실험 분석에서는 본 논문에서 인터럽트 방식을 적용해 개선한 mod_AODV와 기존 AODV 라우팅 프로토콜의 성능 비교 분석을 위한 시뮬레이션 환경 설정과 시뮬레이션 결과에 대해 기술하였고, V.에서 결론을 맺는다.

II. 관련 연구

2.1. 애드혹 네트워크의 라우팅 프로토콜들과 성능 비교

Ad-hoc 네트워크의 라우팅 프로토콜은 그림 2와

같이 크게 유선환경에서 사용되는 Bellman-Ford 알고리즘을 사용하는 테이블 관리 방식과 모든 단말이 이동하는 애드 혹 환경을 고려한 요구 기반 방식 그리고 이 두 가지 방식의 장점을 혼합한 하이브리드 방식(Hybrid)과 기타방식으로 분류할 수 있다¹¹⁾.

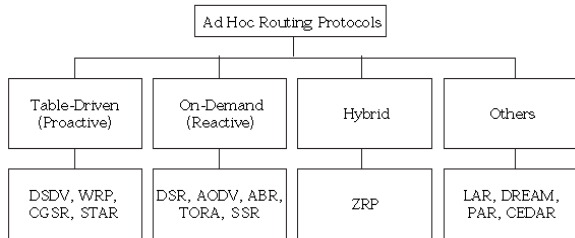


그림 2. 애드 혹 라우팅 프로토콜의 분류
Fig. 2. Classification of Ad Hoc routing protocols

2.1.1. 테이블 관리 방식(Table-driven)

테이블 관리 방식은 각 노드가 네트워크에 속한 모든 노드들에 대한 전체 경로를 테이블 각 엔트리에 저장하고 있다가 주기적으로 또는 네트워크 토폴로지가 변화할 때 라우팅 정보를 브로드캐스팅함으로써 항상 최신의 네트워크 정보를 유지하는 방식이다. 트래픽 발생으로 연결 요청이 있을 때 해당 경로 정보를 갖고 있어 연결설정이 빠른 장점이 있지만, 경로 관리를 위한 제어 메시지의 브로드캐스팅 오버헤드가 너무 크고, 이동 애드 혹 네트워크에서와 같은 잦은 위상변화에 대해 사용하지도 않는 경로발견으로 활용 가능한 자원의 많은 부분을 소비한다는 단점이 있다. 이 방식의 라우팅 프로토콜로는 DSDV (Destination Sequenced Distance Vector), WRP (Wireless Routing Protocol), CGSR(Clusterhead Gateway Switching Routing), STAR(Source-Tree Adaptive Routing) 등이 있다¹¹⁾.

2.1.2. 요구 기반 방식(On-demand 또는 reactive)

요구 기반 방식은 연결 요청이 있을 때 목적 노드로의 경로 설정을 위한 브로드캐스팅 오버헤드가 발생되고, 이로 인해 경로 발견을 위한 지연 시간이 커지는 단점이 있지만, 경로 설정 시 이동 노드의 이동성을 즉시 반영할 수 있어 정확한 경로 설정이 가능하다는 장점이 있다. 해당 하는 라우팅 프로토콜로는 표 1과 같이 DSR (Dynamic Source Routing), AODV(Ad-hoc On-demand Distance

Vector), ABR(Associativity-Based Routing), TORA(Temporally-Ordered Routing Algorithm), SSR(Scalable Source Routing) 등이 있다¹¹⁾.

표 1. 요구 기반 라우팅 프로토콜의 비교
Table 1. Comparison of On-demand Routing protocols

	Advantage	Disadvantage
DSR	<ul style="list-style-type: none"> possess Multiple path support Asymmetric link 	<ul style="list-style-type: none"> Source routing overhead Stale cache problem
AODV	<ul style="list-style-type: none"> support Multicast no path in the Data packet 	<ul style="list-style-type: none"> assume Symmetric links periodic Hello message
TORA	<ul style="list-style-type: none"> Multiple paths created support Multicast 	<ul style="list-style-type: none"> Synchronization problem
ABR	<ul style="list-style-type: none"> select a long-lasting path reset of partial route by intermediate node 	<ul style="list-style-type: none"> periodic Beacon
SSR	<ul style="list-style-type: none"> choose "stronger" connectivity routes 	<ul style="list-style-type: none"> periodic Beacon prohibit reset of partial route by intermediate node

2.1.3. 라우팅 프로토콜들 간의 성능 비교 연구

이동 애드 혹 네트워크 환경에서 라우팅 프로토콜들 간의 성능을 비교분석한 연구들은 크게 두 가지로 분류할 수 있는데 하나는 테이블 관리 방식과 요구 기반 방식 간의 비교 분석이고, 다른 하나는 요구 기반 방식의 라우팅 프로토콜들 간의 비교 분석이다.

우선 테이블 관리 방식과 요구 기반 방식 라우팅 프로토콜들 간의 비교 연구에 의하면 이동 망에서는 요구 기반 방식의 AODV가 패킷 전달률이 우수한 편이며, 특히 높은 이동성을 갖는 경우에는 AODV가 패킷 전달률이 높았다고 분석하고 있다¹⁵⁻⁹⁾.

또한 요구 기반 방식 라우팅 프로토콜들 간의 비교 연구에서는 다양한 이동 패턴, 밀도, 그리고 트래픽에서 AODV가 패킷 전달률이나 시스템 효율에서 좋은 성능을 보였고, 특히 많은 트래픽과 높은 이동성을 갖는 경우에도 AODV 라우팅 프로토콜이 우수했음을 제시하고 있다¹⁰⁻¹⁴⁾.

2.2. AODV 라우팅 프로토콜

애드 혹 네트워크 라우팅 프로토콜들 간의 성능을 비교분석한 선행 연구들에 의해 특히 많은 트래픽과 높은 이동성을 갖는 이동 애드 혹 네트워크 환경에서 우수한 성능을 보인 AODV는 1999년에 C. Perkins에 의해 테이블 기반 방식의 DSDV를 요구 기반 방식에 적용하기 위해 제안된 라우팅 프로토콜이다.

2.2.1. AODV의 특징

AODV는 연결 요청이 있을 때 경로 탐지 절차를 통해 얻어진 경로만을 일정시간 동안 유지한다는 점에서 테이블 기반 방식의 DSDV를 개선하였다. 또한 연결 요청 시 경로 정보를 획득하기 때문에 노드의 이동성에 관한 정확한 정보를 경로 설정에 반영할 수 있어 데이터 전송률에서는 우수함을 보인다.

같은 요구 기반 방식의 DSR과 비슷하게 동작하며 라우팅을 위한 제어 패킷도 RREQ, RREP, RERR 패킷이 사용된다. AODV는 패킷 헤더에 전체경로를 포함하지 않아 대역폭 측면에서 소스 라우팅 시 자원낭비 문제점을 개선하였고, 단지 통신할 필요가 있는 노드간의 경로만 라우팅 테이블에 관리함으로써 DSR 라우팅 프로토콜을 개선하였다. 또한 잘못된 경로 사용과 루프형성 문제를 해결하기 위하여 시퀀스 번호(Sequence Number)를 사용한다.

반면 AODV는 연결 요청 시 경로를 설정하기 때문에 평균 중단간 지연시간이 높고, 경로 추적 방식을 적용하는 다른 라우팅 프로토콜들에 비해 제어 패킷 오버헤드가 크다. 또한 AODV는 RREP 패킷이 RREQ 패킷으로부터 설정된 경로를 사용하기 때문에 대칭링크 만을 지원하고, 목적지까지 하나의 경로만 유지하며, 경로가 변하지 않더라도 데이터 패킷이 전송되지 않으면 일정시간 경과 후에는 경로를 자동 삭제하므로 발생하는 제어 패킷 수는 DSR 보다 훨씬 많은 단점을 갖는다^[1].

AODV의 제어패킷은 경로 탐지를 위한 RREQ, RREP 등과 경로유지를 위한 RERR, 그리고 주기적인 Hello message 등이 사용되며 대부분의 선행 연구들이 경로 탐지에서의 RREQ와 RREP를 줄이기 위해 많은 노력을 해왔다.

2.2.2. Hello message

애드 혹 네트워크에서는 동적인 위상 변화로 이동 노드들 간의 로컬 연결을 탐지하기 위해 2가지 방법을 사용할 수 있다. 하나는 링크 계층 피드백(link layer feedback)을 이용 하는 것이고, 다른 하나는 주기적인 제어 패킷의 브로드 캐스트를 이용한다. 그러나 실제 네트워크 환경에서는 링크 계층 피드백을 얻는 것이 어렵기 때문에 대부분의 프로토콜들은 주기적인 제어 패킷을 브로드캐스트함으로써 노드들 간의 로컬 연결을 탐지한다^[15].

특히 요구 기반 방식의 AODV는 경로 유지를

위해 주기적으로 Hello message를 브로드캐스트 하는데 이것은 이웃 노드들과의 로컬 연결에 대한 탐지와 이웃 목록을 유지하기 위한 것이다.

AODV에서 각 이동 노드는 제어 패킷(hello message 포함)의 수신을 통해 이웃 노드들과의 로컬 연결 여부를 결정하게 된다. 즉, 각 노드는 이웃노드들이 브로드캐스트한 제어 패킷(hello message포함)을 수신하면 해당 이웃 노드에 관한 라우팅 테이블 엔트리가 존재하는 경우 해당 엔트리의 수명(Lifetime)을 갱신하고, 존재하지 않을 경우 해당 엔트리를 생성하게 된다.

제어 패킷(Hello messages)을 통해 이웃 노드들과의 로컬 연결 여부를 결정하기 위해서는 2개의 변수 HELLO_INTERVAL과 ALLOWED_HELLO_LOSS를 사용한다. 각 이웃 노드는 로컬 연결을 유지하기 위해 미리 정해진 메시지 주기(HELLO_INTERVAL) 내에 특정 제어 패킷을 보내지 않았다면 대신 Hello message를 지역적으로 브로드캐스트하게 된다. 그 결과로 미리 정해진 메시지 주기마다 이웃노드로 부터 적어도 하나의 제어 패킷(hello message포함)이 수신된다. 그러나 만일 이웃노드로 부터의 제어 패킷(hello message포함)이 정해진 횟수의 메시지 주기(ALLOWED_HELLO_LOSS* HELLO_INTERVAL)안에 수신되지 않는다면 전송범위(Transmission range) 내에 해당 이웃노드는 존재하지 않는 것으로 판단하여 해당 라우팅 테이블 엔트리의 수명은 타임아웃이 되고 해당 엔트리는 삭제되어 로컬 연결은 단절된 것으로 처리 된다^[15].

각 노드가 이웃 노드와의 로컬 연결 여부를 탐지하고 이웃 목록을 유지하기 위해 주기적으로 브로드캐스트하는 Hello message에 의해 발생 가능한 문제점을 살펴보면 우선 각 이동 노드들이 주기적으로 전송하는 Hello message로 인해 제한된 대역에서 충돌, 간섭 등의 발생으로 패킷 손실을 유발해 패킷 전달률과 제한된 대역폭의 효율성을 감소시키는 것이다.

다음으로 주기적인 메시지로 인한 제어패킷 오버헤드의 증가는 잔존 에너지량의 감소와 이로 인한 전체 네트워크 수명을 감소시킨다.

마지막으로 주기적인 Hello message는 “Reactive” 하지 않은 방식이기 때문에 요구 기반 방식의 라우팅 프로토콜인 AODV의 성격에 자연스럽게 부응하지 못한다는 논쟁의 근원이 될 수 있다.

III. Interrupt message를 이용한 개선된 AODV

본 연구에서는 [16]에서 정의한 실패의 종류로 갑작스런 물리적 고장이나 단절에 의한 예측 불가능한 실패와 노드의 이동 혹은 부족한 전력 등에 의한 예측 가능한 실패 중에서 연구 범위를 예측 가능한 실패로 한정하였고, 이를 기반으로 Hello message의 주요 기능인 로컬 연결의 탐지 및 이웃 목록의 유지에 영향을 줄 수 있는 상황의 종류와 이에 해당하는 Interrupt message들을 함께 정의하였다. 또한 이 방식을 제안하는데 있어 이동 애드혹 노드들은 자신의 잔존 에너지량 즉, 배터리량을 모니터링 할 수 있고, 위치 이동 감지 센서를 통해 이동 사실을 감지해 낼 수 있다는 것을 전제로 한다.

표 2. 제어 패킷의 비교
Table 2. Comparison of Control packet

AODV	mod AODV
RREQ	RREQ
RREP	RREP
RERR	RERR
Hello message (Periodic- interval)	Interrupt message (Interrupt driven)

AODV와 mod_AODV의 제어 패킷의 비교는 표 2와 같다. 제어 패킷은 경로 탐지를 위한 RREQ, RREP와 경로 유지를 위한 RERR은 동일하지만, 로컬 연결의 탐지 및 이웃 목록의 유지를 위한 주기적인 Hello message가 mod_AODV에서는 성능 개선을 위해 Interrupt message들로 대체되었다. 즉, Hello message는 메시지 주기에 따라 주기적으로 브로드캐스트 되는 반면 Interrupt message를 이용해 성능을 개선한 mod_AODV는 각 이동 노드가 이웃 목록과 로컬 연결에 영향을 주는 상황이 발생되면 미리 정의된 해당 Interrupt message를 브로드캐스트해 이웃 노드들에게 로컬 연결을 탐지하도록 하고, 이웃 목록을 갱신하도록 한다.

mod_AODV에서 정의한 로컬 연결에 영향을 주는 상황과 Interrupt message의 분류를 살펴보면 표 3과 같고, 이를 기반으로 발생할 수 있는 4가지 종류를 아래에서 각 경우별로 설명한다.

표 3. 인터럽트 메시지의 분류
Table 3. Classification of Interrupt messages

State	Message type	Interval (Set timer)	neighbor's Lifetime
start (Join) network	START	X	MAX
receive "START"	S_ACK	X	MAX
low battery	LACK	X	0
movement	MOVE	O	ALLOWED_MESSAGE_LOSS * MESSAGE_INTERVAL

3.1. 네트워크가 시작되는 경우

네트워크가 시작되는 것은 모든 노드가 네트워크에 새로 진입하는 경우로 각 노드들은 이웃 노드들에게 자신의 존재를 알리고, 또한 이웃 노드들의 존재를 파악하여 이웃 목록과 로컬 연결 여부를 탐지하기 위해 Interrupt ("START") message와 Interrupt("S_ACK") message를 이용한다.

즉, 이 경우 각 노드들은 Interrupt ("START") message를 이웃 노드들에게 지역적으로 브로드캐스트하고, 이를 수신한 이웃 노드들은 각자의 라우팅 테이블에 메시지를 보낸 노드에 대한 정보를 생성 혹은 갱신하고, 해당 엔트리의 수명은 MAX값을 설정해 준다.

또한 Interrupt ("START") message를 수신한 이웃 노드들은 자신의 존재도 알리기 위해 Interrupt("S_ACK") message를 이웃 노드들에게 지역적으로 브로드캐스트하는데, 이 경우 해당 MESSAGE_INTERVAL에 다른 제어 패킷이 이미 브로드캐스트 되었다면 전송을 하지 않는다.

Interrupt("S_ACK") message를 수신한 노드들 역시 각자의 라우팅 테이블에 메시지를 보낸 이웃 노드에 대한 정보를 생성 혹은 갱신하고, 해당 엔트리의 수명은 MAX값을 설정함으로써 네트워크가 시작되는 경우에 각 노드들이 이웃 목록과 로컬 연결 여부를 탐지할 수 있게 된다.

3.2. 네트워크에 새로운 노드가 진입하는 경우

이미 시작된 네트워크에 새로운 노드가 진입을 하게 되면 해당 노드는 자신의 진입 사실을 이웃 노드들에게 알리고, 또한 이웃 노드들의 존재를 파악하여 이웃 목록과 로컬 연결 여부를 탐지하기 위해 Interrupt ("START") message와 Interrupt("S_ACK") message를 이용한다.

이 경우 이미 시작된 네트워크에 새로 진입한 노드는 Interrupt("START") message를 이웃 노드들에게 지역적으로 브로드 캐스트하게 되는데 이후 설

명은 1. 네트워크가 시작되는 경우와 동일하다.

즉, Interrupt (“START”) message를 수신한 이웃 노드들은 각자의 라우팅 테이블에 네트워크에 새로 진입한 노드에 대한 엔트리의 수명을 MAX값으로 하여 생성 혹은 갱신해 주고, 자신의 존재도 알리기 위해 Interrupt(“S_ACK”) message를 지역적으로 브로드캐스트하는데, 해당 MESSAGE_INTERVAL에 다른 제어 패킷이 이미 브로드캐스트 되었다면 전송을 생략한다. 새로 진입한 노드는 이웃 노드로 부터 Interrupt(“S_ACK”) message를 수신하면 라우팅 테이블에 해당 이웃 노드들에 대한 엔트리의 수명을 MAX값으로 생성 혹은 갱신함으로써 새로 진입한 곳에서의 이웃 목록을 유지하고 로컬 연결 여부를 탐지할 수 있게 된다.

3.3. 노드의 배터리가 부족한 경우

배터리 용량에 제한이 있는 이동 애드 혹 노드들은 배터리 부족으로 네트워크에서 갑자기 사라지게 되어 이웃 노드들과의 로컬 연결에 영향을 주게 된다. 따라서 각 노드는 자신의 잔존 에너지량을 체크하여 일정 기준 이하로 떨어지게 되면 곧 자신이 배터리 부족으로 인해 네트워크에서 사라질 것을 이웃 노드에게 알리기 위해 Interrupt(“LACK”) message를 이용한다. 만일 Interrupt(“LACK”) message를 전송한 노드가 데이터 전송을 위해 설정된 경로(Active route)에 속한 노드라면, 해당 노드의 상위 노드(Upstream node)는 다음 노드(Next node)의 배터리 부족으로 인해 설정된 경로가 끊어질 수 있다는 것을 예측할 수 있어 빠르게 적절한 대처를 할 수 있게 된다.

본 연구는 Interrupt message를 이용하여 이웃 목록 유지와 로컬 연결 여부를 판단하는 Hello message의 기능을 대신하는 것으로 범위를 한정하였고, 따라서 배터리 부족으로 사라질 이웃 노드를 탐지하여 이웃 목록 유지 및 로컬 연결 여부를 판단할 뿐 탐지 후 적절한 처리를 하는 것은 현재 다른 연구에서 진행 중임을 밝힌다.

이 경우 잔존 에너지량이 일정 기준 이하로 떨어진 노드는 Interrupt(“LACK”) message를 이웃 노드들에게 지역적으로 브로드 캐스트하고, 이를 수신한 이웃 노드들은 배터리가 부족한 해당 노드의 변경 사항을 각자의 라우팅 테이블에 반영하기 위해 해당 노드의 정보를 갱신하고, 엔트리 수명은 0 값으로 설정한다.

이후 일정시간(CHECK_ROUTE_TIMEOUT)이

지나면 해당 엔트리는 각 이웃 노드의 라우팅 테이블에서 자동 삭제되고, 더 이상 이웃 목록으로 유지되지 않으며, 로컬 연결은 단절된 것으로 처리된다. 또한 노드가 배터리를 재충전하고 네트워크에 다시 진입을 하게 되는 경우는 2. 네트워크에 새로운 노드가 진입하는 경우와 같이 처리 되어 다시 네트워크에 속함과 동시에 이웃 노드들과의 로컬 연결을 유지하게 된다.

3.4. 노드가 이동하는 경우

각 이동 노드는 자신의 위치 이동을 감지할 수 있어 이동을 하는 경우 이 사실을 이웃 노드에게 알리기 위해 Interrupt (“MOVE”) message를 이용한다. 다른 Interrupt(“START”, “S_ACK”, “LACK”) message들은 해당 상황이 발생되면 한번만 지역적으로 브로드캐스트되는데 반해 Interrupt (“MOVE”) message는 이동을 멈출 때까지 MESSAGE_INTERVAL마다 주기적으로 브로드캐스트 되어 이동하는 곳의 이웃 노드들에게 자신의 존재를 알리게 된다. 또한 이동의 유형을 세분화해보면 이동만 하는 경우와 이동을 하다 멈추는 경우, 그리고 이동을 하다 멈추고 다시 이동을 시작하는 경우로 나누어 생각해 볼 수 있다. 특히 휴지 시간(Pause time)을 반영한 경우는 해당 시간마다 주기적으로 이동을 멈추었다 다시 이동하기 때문에 이 경우를 세분화하여 분석하였다.

우선 이동만 하는 경우에서는 이동을 시작한 노드가 이웃 노드들에게 Interrupt (“MOVE”) message를 MESSAGE_INTERVAL마다 브로드캐스트하고, “MOVE” 메시지를 수신한 이웃 노드들은 각자의 라우팅 테이블에 이동 노드의 정보를 생성 혹은 갱신하는데 해당 엔트리의 수명은 ALLOWED_MESSAGE_LOSS * MESSAGE_INTERVAL을 값으로 설정해 준다. 이것은 허용 가능한 메시지 손실 횟수만큼의 주기 동안을 수명으로 설정해 놓고, 그 시간 안에 다음 “MOVE” 메시지가 수신된다면 해당 이동 노드는 아직까지 전송범위 내에서 이동하고 있는 것으로 판단되어 이웃 목록으로 유지가 되는 것이다. 만일 타임아웃이 될 때까지 다음 “MOVE” message가 수신되지 않는다면 해당 이동 노드는 전송범위를 벗어나 이동하고 있는 것으로 판단하여 라우팅 테이블 엔트리는 삭제되고, 더 이상 이웃 목록으로 유지되지 않는다.

다음으로 이동을 하던 노드가 이동을 멈추게 되면 Interrupt (“START”) message를 지역적으로 브

로드캐스트하여, 이동을 멈춘 곳의 이웃 노드들에게 자신의 존재를 알리는데 이후 설명은 2. 네트워크에 새로운 노드가 진입하는 경우와 동일하다.

마지막으로 이동을 멈추었다 다시 이동을 시작하는 경우에도 이동 노드는 Interrupt (“MOVE”) message를 MESSAGE_INTERVAL마다 브로드캐스트하여 자신의 이동 사실을 이동하는 곳의 이웃 노드들에게 알리게 된다.

IV. 실험 분석

4.1. 시나리오 사전 실험 결과

Interrupt message를 이용해 성능을 개선한 mod_AODV의 성능을 실험 분석하기에 앞서 기존의 AODV 라우팅 프로토콜의 성능 분석을 위한 최적의 시뮬레이션 파라미터 값을 선정하기 위해 사전 실험을 수행하였다. 사전 실험은 QualNet 5.0을 이용하여 수행하였고 실험 환경은 다음과 같다.

1500m × 1500m의 시뮬레이션 공간에 100개의 노드를 임의로 배치시키고, 100개의 노드 중 이동 노드의 비율(이하 #mobile로 표현하고 10, 20, 30, 40%로 증가), Maximum velocity(5, 10, 15, 20m/s로 증가), Hello message interval(1, 2, 3, 4, 5, 10, 40, 80, 120s로 증가), Pause time(30, 100, 500s로 증가), Bandwidth(2, 11Mbps로 증가)를 증가시키면서 그 외의 시뮬레이션 파라미터는 표 5와 같이 설정하여 작성된 900여개의 시나리오를 가지고 AODV 라우팅 프로토콜을 이용하여 사전 실험을 수행하였다.

표 4. 사전 실험 결과
Table 4. Preliminary experimental results

simulation parameter	PDR(Max) CPO(Min) Delay(Min)	PDR(Min) CPO(Max) Delay(Max)
#Mobile	10 %	40 %
Max. velocity	10 m/s	20 m/s
Hello interval	3 s	1 s
Pause time	100 s	30 s

사전 실험 결과는 표 4와 같이 성능 매트릭을 기준으로 분석하여 Packet Delivery Ratio(이하 PDR로 표현)가 최대, Control packet overhead(이하 CPO로 표현)와 Average end to end delay(이하 Delay로 표현)가 최소인 경우와 PDR이 최소, CPO와 Delay가 최대인 경우에 가장 많은 비율을 차지

하는 시뮬레이션 파라미터를 선정하였다.

Hello message interval은 1초에서 PDR이 최소, CPO와 Delay는 최대였으며, 3초 이후 PDR은 크게 증가 하지 않았고, CPO와 Delay는 크게 감소하지 않았기 때문에 Hello message interval은 1초(Min)와 3초(Max)를 선정하였다.

4.2. 시뮬레이션 환경

Interrupt message를 이용한 mod_AODV의 성능 분석을 위한 시뮬레이션 환경은 표 4. 사전 실험 결과에 의해 선정된 시뮬레이션 파라미터 값을 적용하여 표 5와 같이 설정하였다.

표 5. 시뮬레이션 파라미터
Table 5. Simulation parameter

Simulation parameter	Sim. 1	Sim. 2
Simulation Tool	QualNet 5.0	
Simulation Time	1000 s	
Simulation Terrain	1500 m X 1500 m	
Number of Node	100 node	
Number of CBR (CBR rate)	5000 (1packet/sec)	50000 (10packet/sec)
CBR packet size	512 bytes	
Channel Frequency	2.4Ghz	
MAC layer	IEEE802.11b	IEEE802.11g
Bandwidth (Transmission range)	2Mbps (449.403m)	24Mbps (252.717m)
	11Mbps (272.450m)	
Mobility model	Random waypoint	
Maximum Velocity	10(6~10), 20(16~20) m/s	
Pause time	30, 100 (s)	
Number of mobile node	10, 40 (%)	
Message interval	1, 3 (s)	

성능 분석을 위한 시뮬레이션은 두 가지로 구분되어 진행되었는데 시뮬레이션 1(Sim. 1)은 대부분의 연구에서 실험환경으로 채택한 것으로 MAC layer는 IEEE 802.11b DCF이고, 대역폭은 2Mbps(전송범위 449.403 m)와 11Mbps(전송범위 272.450 m) 각각에서 Traffic model로는 초당 CBR 1packet(5,000 packet)을 전송하는 것으로 설정하여 수행되었다. 이것은 대역폭을 2Mbps에서 11Mbps로 높이면서, 전송범위는 449.403m에서 272.450m로 감소되는 환경에서 제안하는 mod_AODV와 AODV의 성능을 실험 분석한 것이다. 시뮬레이션 2(Sim. 2)

는 최근 무선기기들이 IEEE 802.11g/n/ac까지 사용한다는 것을 고려하여 MAC layer는 IEEE 802.11g 이고, 대역폭은 시뮬레이션 1에서 11Mbps의 전송범위와 가장 비슷한 전송범위를 갖는 대역폭 24Mbps (전송범위 252.717m)에서 Traffic model로는 초당 CBR 10packet (50,000 packet)을 전송하는 것으로 설정하여 수행되었다. 즉, 시뮬레이션2는 시뮬레이션1과 비교해 IEEE 802.11g를 사용하여 전송범위는 비슷하게 유지하면서 대역폭과 트래픽 양을 높은 환경에서 제안하는 mod_AODV와 AODV의 성능을 실험 분석한 것이다.

Message interval은 AODV에서는 HELLO_INTERVAL로, mod_AODV에서는 MESSAGE_INTERVAL의 값으로 각각 적용하였다.

QualNet 5.0에서 제공하는 IEEE802.11b의 대역폭 2Mbps의 경우 전송 범위가 449.403m이고, 11Mbps의 경우 272.450m이다. 대역폭이 높아지면 전송 범위는 감소되고, 이를 고려하여 11Mbps의 전송 범위를 2Mbps의 것과 동일하게 맞추기 위해 노드 간의 거리를 보정하면 전송 범위가 겹치는 영역이 많을수록 불필요한 메시지를 제거하여 제어 패킷 오버헤드를 감소시킨 mod_AODV에만 좋은 결과가 나올 수 있기 때문에 거리 보정은 하지 않고, 현실 상황을 반영한 기본 전송범위 값을 그대로 사용하였다. 또한 시뮬레이션 2에서 사용하는 IEEE 802.11g에서 24Mbps의 경우에서도 위와 같은 이유로 거리 보정은 하지 않았고, 기본 전송범위 값도 그대로 사용하였다.

4.3. 성능 매트릭(Performance metrics)

본 연구에서 이웃 탐색 기법(Neighbor discovery method)으로 주기적인 Hello message를 사용하는 AODV와 Interrupt message를 이용하여 개선된 mod_AODV의 성능을 분석하기 위해 사용한 성능 매트릭들은 아래와 같다.

- 패킷 전달률(PDR : Packet Delivery Ratio)
목적지 노드에서 수신한 패킷 수 / 소스 노드에서 전송한 패킷 수
- 평균 종단간 지연(Average end to end delay)
인터페이스에서의 큐잉지연(queuing delay), MAC에서의 재전송 지연(retransmission delay) 그리고 경로 탐색 동안 버퍼링에 의해 발생 가능한

지연을 모두 포함하고, 전파(propagation)와 전송(transfer) 시간들도 포함된다.

- 단절된 링크의 수(Number of broken link)
링크가 단절되면 패킷을 큐(queue)에 넣고 경로를 다시 확보한 후에 재전송하기 때문에 지연 시간이 발생된다. 따라서 단절된 링크가 바로 패킷 손실로 이어지지 않는 경우도 있기 때문에 패킷 전달률에 직접적인 영향을 주기보다는 오히려 평균 종단간 지연시간과 밀접한 관련이 있어⁶⁾ 이 두 가지를 차트에 함께 제시한다.
- 제어 패킷 오버헤드(Control Packet Overhead)
데이터 패킷의 전달을 위해 경로 탐지와 경로 유지를 위해 사용되는 제어 패킷의 수
- 잔존 에너지량(Residual battery capacity)

4.4. 시뮬레이션 결과

시뮬레이션 결과는 각 성능 매트릭 별로 시뮬레이션 1(IEEE 802.11b DCF, CBR 1packet/sec)에서 대역폭이 2 Mbps와 11 Mbps의 결과와 시뮬레이션 2(IEEE 802.11g, CBR 10packet/sec)에서 24Mbps의 결과인 3가지 차트로 제시된다.

각 차트의 X축은 우선 Pause time이 30, 100 초로 구분되고, 이후 각 Pause time 내에서 이동 노드의 비율(#mobile)이 10, 40 %로 분류되며, 각 이동 노드의 비율 내에서 Message interval이 1, 3 s로 구분되고, Message interval에서 Maximum Velocity가 10, 20 m/s로 분류되어 시뮬레이션 결과 값을 나타낸다.

4.4.1. 패킷 전달률(Packet Delivery Ratio)

이웃 탐색 기법(Neighbor discovery method)으로 주기적인 Hello message를 사용하는 AODV와 Interrupt message를 사용하는 mod_AODV의 성능 분석을 위한 시뮬레이션 결과는 시뮬레이션 1(IEEE 802.11b DCF, CBR 1packet/sec)에서 대역폭이 2Mbps인 패킷 전달률의 그림 3과 11Mbps의 패킷 전달률인 그림 4와 같고, 시뮬레이션 2(IEEE 802.11g, CBR 10packet/sec)에서 대역폭 24Mbps의 패킷 전달률은 그림 5와 같다.

AODV와 mod_AODV 모두 그림 3, 그림 4 그리고 그림 5를 통해 Pause time이 100인 경우가 30인 경우보다 PDR 값이 대체로 높고, 이동 노드의

비율(#mobile)이 10%인 경우가 많은 노드가 움직이는 40%인 경우 보다 PDR값이 높았다.

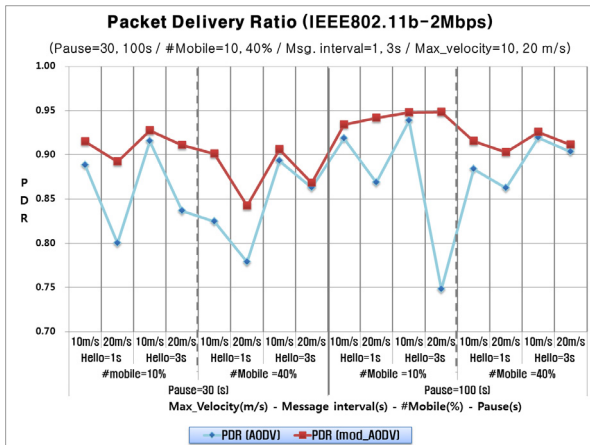


그림 3. 패킷 전달률 - IEEE 802.11b(2Mbps)
Fig. 3. Packet Delivery Ratio - IEEE 802.11b(2Mbps)

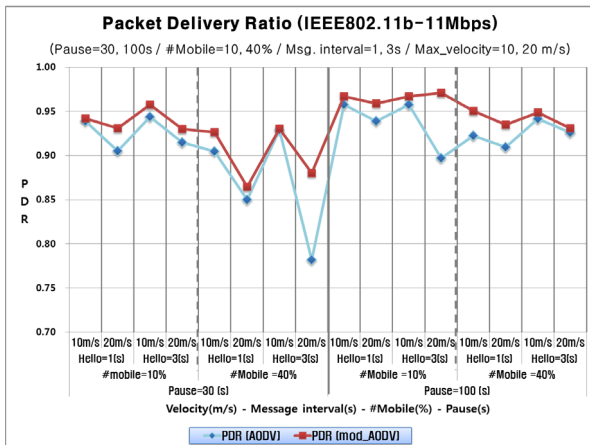


그림 4. 패킷 전달률 - IEEE 802.11b(11Mbps)
Fig. 4. Packet Delivery Ratio - IEEE802.11b(11Mbps)

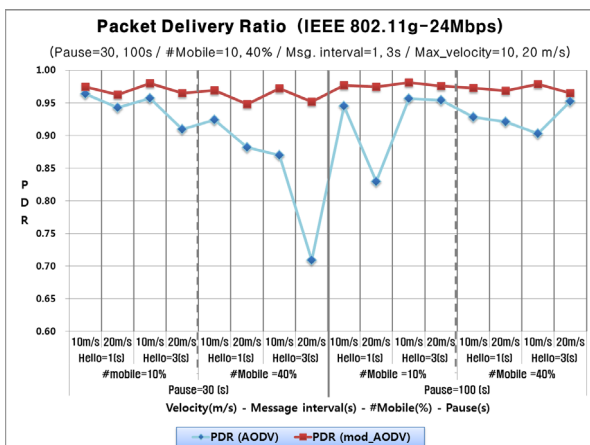


그림 5. 패킷 전달률 - IEEE 802.11g(24Mbps)
Fig. 5. Packet Delivery Ratio - IEEE 802.11g(24Mbps)

AODV는 Hello message, mod_AODV는 Interrupt message의 Message interval이 3초인 경우가 메시지가 더 자주 전송되는 1초인 경우보다 대체로 PDR값이 높았으며, Maximum velocity가 10 m/s인 경우가 20 m/s인 경우보다 PDR값이 더 높았다.

즉, AODV와 mod_AODV 모두 이동 노드의 비율이 적고 각 이동노드가 자주 멈추지 않으며, 움직이는 최대 속도가 작은 경우 그리고 메시지를 자주 전송하지 않는 경우가 PDR 값이 대체로 높았다.

다음으로 AODV와 mod_AODV를 비교하면 Message interval이 큰 경우 즉, 메시지 전송량이 적은 경우가 PDR 값이 높게 나왔듯이 AODV보다 인터럽트 방식을 이용하여 불필요한 제어 패킷의 양을 줄인 mod_AODV가 모든 세부항목에서 PDR 값이 더 높게 나오는 것을 그림 3, 그림 4 그리고 그림 5에서 모두 확인 할 수 있었다.

그림 4(11Mbps)에서는 대역폭의 증가와 함께 겹치는 전송범위가 줄어들면서 AODV와 mod_AODV의 PDR 값이 그림 3(2Mbps)의 값에 비해 전체적으로 상승하면서, PDR 값의 간격도 조금 좁혀졌지만 여전히 mod_AODV가 대부분의 구간에서 더 높게 유지 되는 것을 확인할 수 있다.

또한 그림 4에서 AODV와 mod_AODV는 Message interval이 1초인 경우의 PDR값이 3초인 경우 보다 높아지는 구간이 나타나는데, 이것으로 주기적인 제어 패킷을 발생시키는 이웃 노드 탐색 기법도 대역폭만 높아지는 경우에는 요구 기반 라우팅 프로토콜의 PDR값에 긍정적인 영향을 줄 수 있다는 것을 알 수 있다.

하지만 전송범위는 그림 4와 비슷하게 유지하면서 대역폭과 트래픽의 양을 함께 증가시키는 그림 5에서는 Interrupt message의 사용으로 불필요한 제어패킷을 제거한 mod_AODV가 그림 4에서처럼 트래픽의 증가에도 PDR 값이 전반적으로 증가함을 보이는 반면, 주기적인 Hello message를 사용하는 AODV의 경우 트래픽의 증가에 따라 PDR 값의 증가폭이 적거나 오히려 그림 4보다 더 감소하는 구간이 발생되었다.

따라서 대역폭이 증가한다고 하더라도 트래픽의 양을 고려한다면 주기적인 Hello message의 전송으로 인한 제어 패킷 오버헤드의 증가는 PDR 값에 좋지 않은 영향을 주는 것을 알 수 있고, 또한 불필요한 제어패킷 오버헤드를 제거한 mod_AODV는 트래픽의 증가에도 PDR 값을 더 향상시킬 수 있음

을 확인할 수 있었다.

4.4.2. 평균 종단간 지연(Average end to end delay)과 단절된 링크의 수 (Number of broken link)

앞에서 언급했듯이 링크가 단절되면 패킷을 큐에 넣고 경로가 확보되면 그 때 다시 패킷을 꺼내 전송하기 때문에 지연시간이 발생하게 된다. 이처럼 단절된 링크가 바로 패킷 손실로 이어지지 않는 경우도 발생하기 때문에 PDR 값에 직접적인 영향을 미치지 보다는 오히려 평균 종단간 지연시간과 밀접한 관련이 있다⁶⁾. 이러한 이유로 이 두 가지 성능 매트릭을 한 차트에 이중축 혼합형으로 제시하고자 한다.

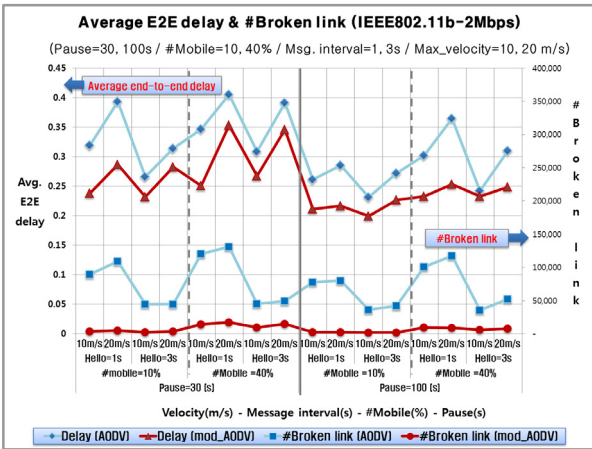


그림 6. 평균 종단간 지연과 단절된 링크의 수 - IEEE 802.11b(2Mbps)
Fig. 6. Average end to end delay&Number of broken link - IEEE 802.11b(2Mbps)

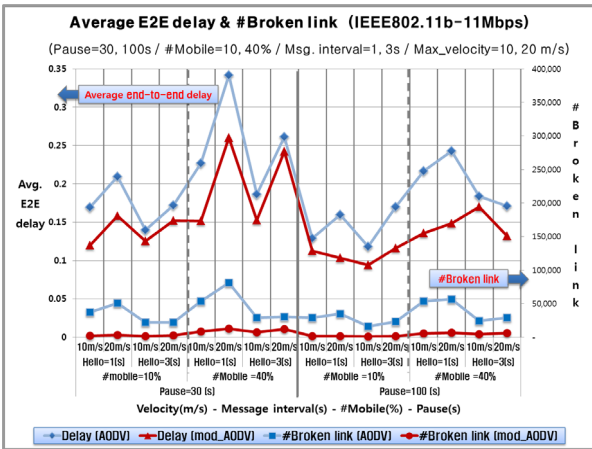


그림 7. 평균 종단간 지연과 단절된 링크의 수 - IEEE 802.11b(11Mbps)
Fig. 7. Average end to end delay&Number of broken link - IEEE 802.11b(11Mbps)

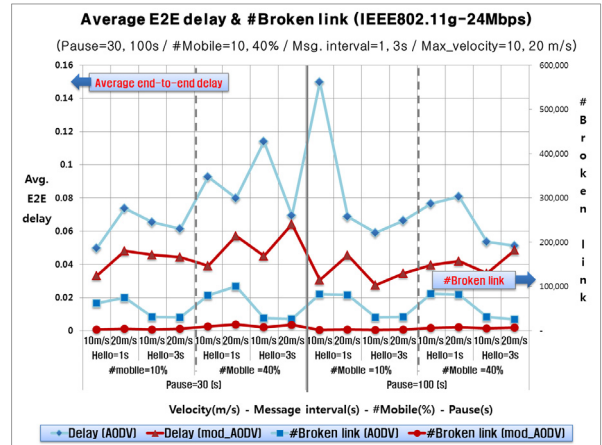


그림 8. 평균 종단간 지연과 단절된 링크의 수 - IEEE 802.11g(24Mbps)
Fig. 8. Average end to end delay&Number of broken link - IEEE 802.11g(24Mbps)

AODV와 mod_AODV의 성능 분석을 위한 시뮬레이션 결과는 시뮬레이션 1(IEEE 802.11b DCF, CBR 1packet/sec)에서 대역폭이 2Mbps일 때 평균 종단간 지연과 단절된 링크의 수는 그림 6과 같고, 11Mbps일 때는 그림7과 같으며, 시뮬레이션 2(IEEE 802.11g, CBR 10packet/sec)에서 대역폭 24Mbps의 평균 종단간 지연과 단절된 링크의 수는 그림 8과 같다. 이중 축 혼합형으로 제시되는 그림 6, 그림 7 그리고 그림 8에서 차트의 X축은 PDR의 경우와 같고, 왼쪽의 세로 주축은 평균 종단간 지연시간을 오른쪽의 세로 보조축은 단절된 링크의 수를 나타낸다.

그림 6, 그림 7 그리고 그림 8의 평균 종단간 지연시간과 단절된 링크의 수를 살펴보면 Pause time 값이 100 초인 경우가 30 초인 경우보다 낮고, 이동노드의 비율이 10%인 경우가 40%인 경우보다 낮으며, Message interval이 3초인 경우가 1초인 경우 보다 낮았다. 또한 Maximum velocity가 10 m/s인 경우가 20 m/s인 경우보다 지연시간과 단절된 링크의 수가 대체로 낮게 유지되는 것으로 나왔다. 즉, Pause time값이 크고 이동 노드의 비율이 낮은 경우 각 이동 노드가 저속으로 움직이고 Message interval이 커서 메시지가 많이 발생되지 않는 경우에 종단간 지연시간과 단절된 링크의 수가 낮게 유지된다.

다음으로 AODV와 mod_AODV를 비교하면 이웃 탐지 기법으로 Interrupt message를 사용하는 mod_AODV가 AODV에 비해 모든 세부항목에서 평균 종단간 지연과 단절된 링크의 수가 낮게 나오

는 것을 그림 6, 그림 7 그리고 그림 8 모두에서 확인할 수 있었다.

그림 7(11Mbps)에서는 대역폭의 증가와 함께 겹치는 전송범위가 줄어들면서 AODV와 mod_AODV의 평균 종단간 지연이 그림 6(2Mbps)에 비해 전체적으로 감소하면서 그 간격도 좁혀졌지만 여전히 mod_AODV가 대부분의 구간에서 더 낮게 유지됨을 확인할 수 있었다.

또한 PDR에서와 마찬가지로 그림 7에서도 AODV와 mod_AODV는 Message interval이 1초인 경우의 지연시간과 단절된 링크의 수가 3초인 경우보다 더 낮아지는 구간이 나타났다. 이것으로 주기적인 제어 패킷을 발생시키는 이웃 노드 탐색 기법도 대역폭만 높아지는 경우에는 지연시간과 단절된 링크의 수에 긍정적인 영향을 줄 수 있다는 것을 확인 하였다.

전송범위를 그림 7(11Mbps)과 비슷하게 유지하면서 대역폭과 트래픽의 양을 동시에 증가시키는 그림 8(24Mbps)에서는 이웃 노드 탐지 기법으로 Interrupt message를 사용하는 mod_AODV가 트래픽의 증가에도 평균 종단간 지연이 큰 폭으로 감소하고, 단절된 링크의 수에서는 약간 증가되는 구간이 발견되었다. 반면 주기적인 Hello message를 사용하는 AODV의 경우 트래픽의 증가에 따라 평균 종단간 지연이 감소되지만 mod_AODV와 격차가 그림 7보다 더 커지는 것을 관찰 할 수 있고, 단절된 링크의 수는 큰 폭으로 증가함으로써 대역폭이 증가한다고 하더라도 트래픽의 양을 고려한다면 주기적인 Hello message의 전송으로 인한 제어 패킷 오버헤드의 증가는 종단간 지연과 단절된 링크의 수에 좋지 않은 영향을 주는 것을 확인할 수 있고, 불필요한 제어 패킷 오버헤드를 제거한 mod_AODV는 트래픽의 증가에도 종단간 지연을 더 감소시킬 수 있음을 확인할 수 있었다.

오른쪽 세로 보조축의 단절된 링크의 수(#Broken link)는 Timeout에 의한 것과 MAC에 의해 수집된 결과 값의 합계로 결과에 포함되어 따로 제시되지 않았지만 MAC에 의해 수집된 단절된 링크의 수 역시도 mod_AODV의 값이 AODV에 비해 낮게 유지되었다.

4.4.3. 제어 패킷 오버헤드(Control Packet Overhead)

AODV와 mod_AODV의 성능 분석을 위한 시뮬레이션 결과는 시뮬레이션 1(IEEE 802.11b DCF,

CBR 1packet/sec)에서 대역폭이 2Mbps인 제어 패킷 오버헤드의 그림 9와 11Mbps의 제어 패킷 오버헤드인 그림 10과 같고, 시뮬레이션 2(IEEE 802.11g, CBR 10packet/sec)에서 대역폭 24Mbps의 제어 패킷 오버헤드는 그림 11과 같다.

그림 9, 그림 10 그리고 그림 11을 통해 제어 패킷 오버헤드를 살펴보면 Pause time값이 100 초인 경우가 30초에 비해 낮은 값을 나타내고, 이동 노드의 비율이 낮은 10%가 40%에 비해 제어패킷을 적게 발생시키며, Message interval이 3초인 경우가 1초에 비해 제어패킷 오버헤드가 적다.

또한 Maximum velocity가 10m/s인 경우가 20m/s인 경우에 비해 낮은 제어 패킷 오버헤드를 발생시킨다. 즉, 이동 노드들이 너무 자주 멈추지 않고, 이동 노드의 비율이 낮으며, 각 이동 노드가 저속으로 움직이고, Message interval이 커서 메시지가 많이 발생되지 않는 경우에 제어패킷 오버헤드가 낮게 유지 된다.

AODV와 mod_AODV를 비교하면 Interrupt message를 이용해 불필요한 주기적인 메시지를 감소시킨 mod_AODV가 모든 세부항목에서 제어 패킷 오버헤드 값이 AODV에 비해 더 낮게 유지됨을 그림 9, 그림 10 그리고 그림 11에서 모두 확인할 수 있다.

특히 AODV의 경우 Message interval 1초와 3초간의 결과 값의 격차가 현격하게 나타남을 확인할 수 있는 반면 mod_AODV의 경우는 불필요한 주기적인 메시지를 줄여 Message interval에 의한 결과 값의 차이가 일정하게 유지됨을 볼 수 있다.

그림 10(11Mbps)에서는 대역폭의 증가와 함께 겹쳐지는 전송범위가 줄어들면서 AODV와 mod_AODV의 제어 패킷 오버헤드 값이 그림 9(2Mbps)의 값에 비해 전체적으로 감소되었음을 볼 수 있고, 그림 9와 같이 mod_AODV가 AODV 보다 모든 구간에서 제어 패킷 오버헤드 값이 더 낮게 유지 되는 것을 확인할 수 있다.

또한 그림 11(24Mbps)에서는 전송범위는 유지하면서 대역폭과 함께 트래픽의 양을 증가시킴으로서 AODV의 제어 패킷 오버헤드 값이 mod_AODV에 비해 큰폭으로 증가됨을 관찰할 수 있고, 앞선 그림 9와 10에서처럼 여전히 mod_AODV가 전 구역에서 제어 패킷 오버헤드 값이 AODV에 비해 낮게 유지됨을 확인할 수 있었다.

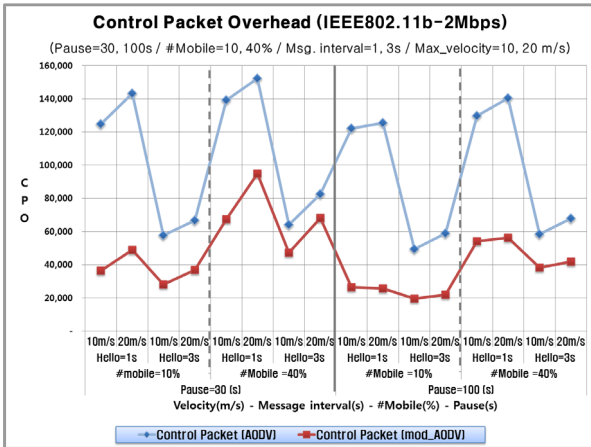


그림 9. 제어 패킷 오버헤드 - IEEE 802.11b(2Mbps)
 Fig. 9. Control Packet Overhead - IEEE 802.11b(2Mbps)

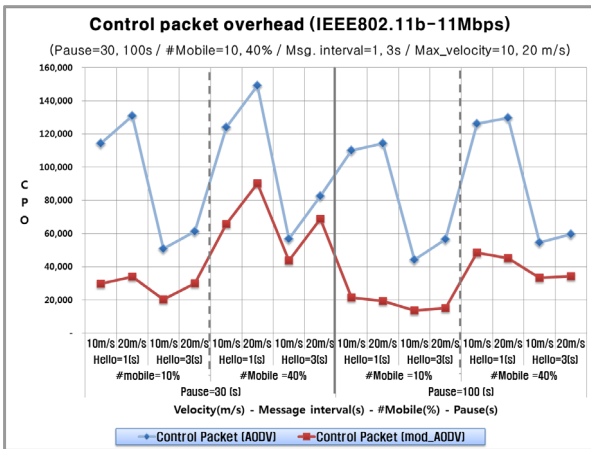


그림 10. 제어 패킷 오버헤드 - IEEE 802.11b(11Mbps)
 Fig.10. Control Packet Overhead - IEEE 802.11b(11Mbps)

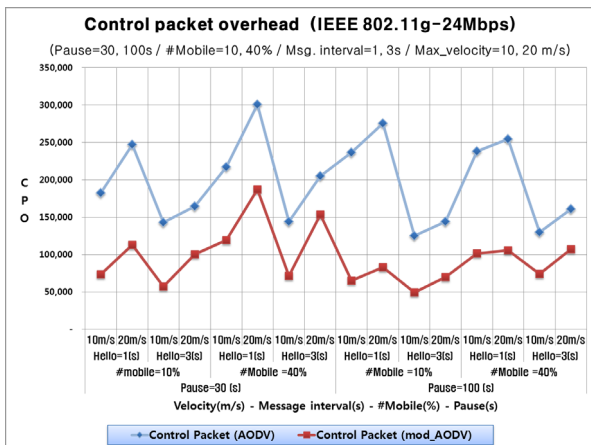


그림 11. 제어 패킷 오버헤드 - IEEE 802.11g(24Mbps)
 Fig. 11. Control Packet Overhead - IEEE 802.11g(24Mbps)

4.4.4. 잔존 에너지량(Residual battery capacity)

AODV와 mod_AODV의 성능 분석을 위한 시뮬레이션 결과는 시뮬레이션 1(IEEE 802.11b DCF, CBR 1packet/sec)에서 대역폭이 2Mbps인 잔존 에너지량의 그림 12와 11Mbps의 잔존 에너지량인 그림 13과 같고, 시뮬레이션 2(IEEE 802.11g, CBR 10packet/sec)에서 대역폭 24Mbps의 잔존 에너지량은 그림 14와 같다.

네트워크 수명과 연관이 있는 AODV와 mod_AODV의 잔존 에너지량은 그림 12, 그림 13 그리고 그림 14를 통해 살펴볼 수 있는데, Pause time값이 100초가 30초에 비해 대체로 높게 유지되고, 이동 노드의 비율에서는 10%가 40%에 비해 높았으며, Message interval의 경우는 3초가 1초에 비해 더 높았다.

또한 이동 노드들의 Maximum velocity에서는 10m/s가 20m/s에 비해 대체로 에너지 잔존량을 높게 유지했다. 즉, 이동 노드들이 너무 자주 멈추지 않고, 이동 노드의 비율이 낮으며, 각 이동 노드가 지속적으로 움직이는 경우, 그리고 Message interval이 커서 메시지 전송이 적은 경우에 잔존 에너지량이 높은 것으로 나타났다.

이웃 노드 탐색 기법으로 Interrupt message를 이용해 불필요한 메시지 전송을 제거하여 제어 패킷 오버헤드를 감소시킨 mod_AODV의 잔존 에너지량이 AODV에 비해 모든 세부항목에서 모두 높게 유지되어 네트워크 수명이 증가됨을 그림 12, 그림 13 그리고 그림 14에서 확인 할 수 있다.

그림 13(11Mbps)에서 대역폭의 증가와 함께 전송범위의 감소로 AODV와 mod_AODV의 잔존 에너지량이 그림 12(2Mbps)의 값에 비해 전체적으로 상승하면서, 잔존 에너지량의 간격도 조금 좁혀졌지만 여전히 불필요한 제어 패킷을 제거한 mod_AODV가 모든 구간에서 더 높게 유지됨을 확인할 수 있다.

특히 제어 패킷 오버헤드에서 Message interval 1초와 3초간의 현격한 차이를 보였던 AODV는 잔존 에너지량에서도 큰 차이를 보이는 반면, mod_AODV는 제어패킷 오버헤드에서와 같이 Message interval의 차이에 일정한 간격을 여전히 유지함을 볼 수 있다.

또한 그림 14(24Mbps)에서는 전송범위는 그림 13과 비슷하게 유지하면서 대역폭과 함께 트래픽의 양을 증가시킴으로서 AODV와 mod_AODV 모두 잔존 에너지량이 감소함을 관찰할 수 있고, 앞선 그림 12와 그림 13에서 처럼 여전히 mod_AODV가

전 구간에서 잔존에너지량이 높아 네트워크 수명이 길어질 수 있음을 확인 할 수 있다.

V. 결 론

이동 애드 혹 네트워크 환경에서 AODV 라우팅 프로토콜은 요구기반 방식으로써의 장점을 갖지만 다른 라우팅 프로토콜들에 비해 제어 패킷 오버헤드가 크다는 단점도 가지고 있어 선행 연구들은 RREQ와 RREP와 같은 경로 탐지를 위한 제어 패킷을 줄이기 위해 많은 노력을 해왔다.

이에 본 연구에서는 AODV의 제어패킷 중 로컬 연결의 탐지와 이웃 목록을 유지하기 위해 주기적으로 브로드캐스트하는 Hello message의 문제점을 분석하고, 이를 개선하고자 하였다.

개선된 mod_AODV는 인터럽트 방식을 도입하였으며, 각 노드는 로컬 연결에 영향을 줄 수 있는 상황이 발생하면 해당되는 Interrupt message(START, S_ACK, LACK, MOVE)를 이웃 노드에게 지역적으로 브로드캐스트한다. 이것을 수신한 이웃 노드가 로컬 연결을 탐지하고 정확한 이웃 목록을 유지할 수 있도록 개선하였다.

즉, mod_AODV는 인터럽트 방식을 이용하여 주기적인 Hello message를 제거함으로써 제어 패킷 오버헤드를 줄여 성능을 개선하였다.

AODV와 mod_AODV의 성능은 QualNet 5.0을 이용하여 수행하였고, 실험 평가는 전체 노드 중 이동 노드의 비율, Maximum velocity, Message interval, Pause time, Bandwidth, Traffic을 증가시켜가면서 해당 시뮬레이션 파라미터들의 변화가 성능 매트릭에 어떠한 영향을 미치는지를 기준으로 AODV와 mod_AODV의 성능을 비교 분석하였다. 또한 이 실험을 진행하기에 앞서 작성된 시나리오로 사전 실험을 수행함으로써 성능 분석을 위한 최적의 시뮬레이션 파라미터를 선정하였고, 이를 기반으로 수행한 실험 결과를 4가지 성능 매트릭을 기준으로 분석하였다. 분석결과 AODV에 비해 mod_AODV가 대체로 높은 패킷 전달률을 유지하였고, 단절된 링크의 수와 평균 종단간 지연에서도 AODV에 비해 낮은 값을 가졌다. 또한 제어 패킷 오버헤드에서도 mod_AODV가 AODV에 비해 낮은 값을 유지했으며 이로 인해 높은 잔존 에너지량을 보임으로써 4가지 성능 매트릭을 통한 분석 결과를 기반으로 mod_AODV가 AODV보다 전반적으로 우수함을 보였다.

본 연구의 의의는 첫째 요구 기반 방식의 프로토콜인 AODV의 제어 패킷 중 “Reactive”하지 않게 주기적으로 브로드캐스트하는 Hello message를 인

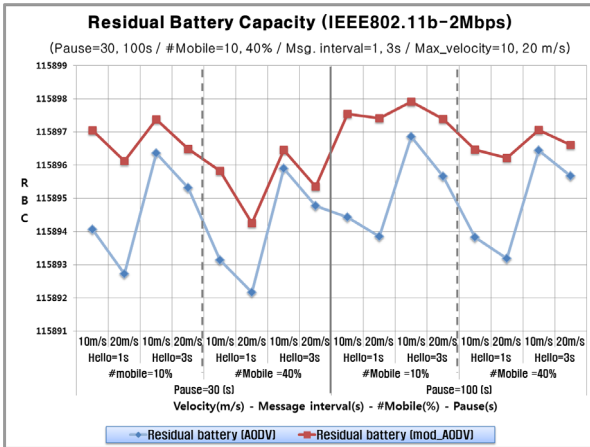


그림 12. 잔존 에너지량 - IEEE 802.11b(2Mbps)
 Fig. 12. Residual battery capacity - IEEE 802.11b(2Mbps)

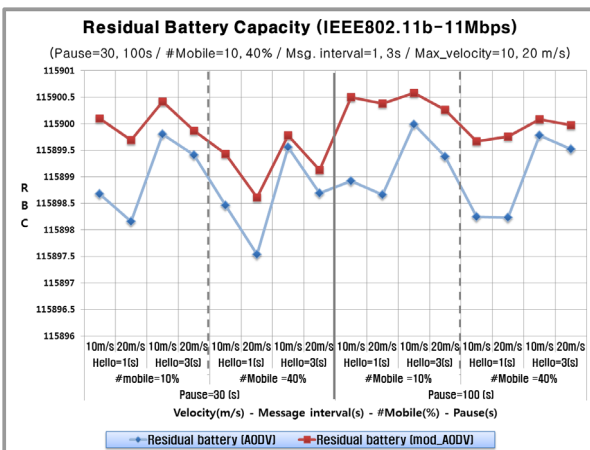


그림 13. 잔존 에너지량 - IEEE 802.11b(11Mbps)
 Fig. 13. Residual battery capacity - IEEE 802.11b(11Mbps)

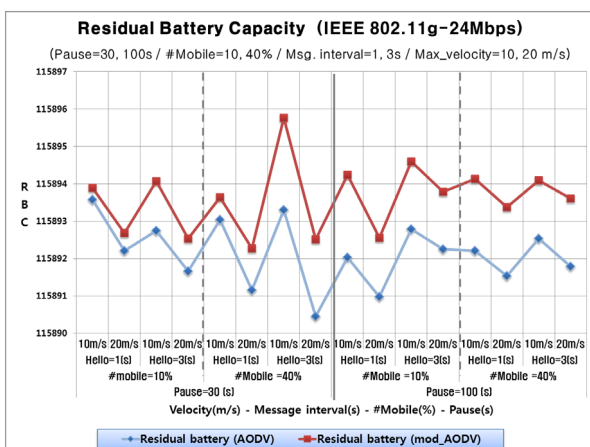


그림 14. 잔존 에너지량 - IEEE 802.11g(24Mbps)
 Fig. 14. Residual battery capacity - IEEE 802.11g(24Mbps)

터럽트 방식을 도입하여 적응적으로 개선하였다는 것이다. 즉, 주기적인 Hello message를 제거하고 대신 로컬 연결에 영향을 미치는 상황이 발생할 때만 해당 되는 Interrupt message를 지역적으로 브로드캐스트하게 함으로써 Hello message의 로컬 영역 탐지 기능과 이웃 목록의 유지기능을 대체하도록 하였다.

둘째 애드 혹 네트워크의 제한된 대역에서 주기적인 메시지를 최소화하여 제어 패킷 오버헤드를 줄임으로써 패킷 전달률을 증가 시키고, 평균 종단간 지연을 줄였으며, 이로 인해 잔존 에너지량을 향상시켜 네트워크 수명을 증가시키는 결과를 가져왔다.

셋째 로컬 연결에 영향을 줄 수 있는 상황과 Interrupt message의 종류를 미리 정의하였기 때문에 Interrupt message를 수신하면 그 종류에 따라 이웃 노드의 상태를 판단할 수 있어 처리 절차를 준비 할 수 있게 되었다는 것이다.

본 연구에서 제안한 mod_AODV 라우팅 프로토콜은 더욱 다양한 시나리오를 통한 실험분석으로 여러 가지 유용한 결과를 얻을 수 있을 것이다. 또한 미리 정의된 Interrupt message의 종류를 통해 이웃 노드와의 로컬 연결 상태를 탐지하고, 정확한 이웃 목록을 유지하는 것뿐만 아니라 Interrupt message의 종류에 따라 판단 및 예측을 통해 여러 가지 후속 처리에 관한 연구를 진행한다면 더욱 성능이 개선될 것이라고 판단된다.

References

[1] J.-C. Kim and Y.-Y. Kim, "The research trend in Ad Hoc network protocol," *Telecommun. Review*, vol. 12. no. 3, pp. 298-311, June 2002.

[2] C. W. Kim, *Technical Trends on Ad Hoc Networks(2004. 9)*, Retrieved May, 24, 2013, from <http://www.eic.re.kr>.

[3] H. Y. Kwon, J. W. Shin, J. H. Choi, and S. W. Nam, "Technical trends on mobile Ad Hoc networks," *Electron. Telecommun. Trends*, vol. 18, no. 2, pp. 11-24, Apr. 2003.

[4] S. Ahn, *Standardization Trends for IETF MANET working group(2004.10)*, Retrieved May, 24, 2013, from http://www.tta.or.kr/data/weekly_view.jsp?new

s_id=867.

[5] X. Zhang, Y. Wang, J.-G. Ki, and K.-T. Lee, "Analysis of MANET protocols using OPNET," *J. Inst. Webcasting, Internet Television, Telecommun.*, vol. 9, no. 4, pp. 87-97, Aug. 2009.

[6] H.-Y. Ahn, "Impact of mobility on the Ad Hoc network performance," *J. Inst. Webcasting, Internet, Telecommun.*, vol. 10, no. 5, pp. 201-208, Oct. 2010.

[7] P. Kuppusamy, K. Thirunavukkarasu, and B. Kalaavathi, "A study and comparison of OLSR, AODV and TORA routing protocols in ad hoc networks," in *Proc. 3rd Int. Conf. Electron. Computer Technol. (ICECT 2011)*, vol. 5, pp. 143-147, Kanyakumari, India, Apr. 2011.

[8] A. Shrestha and F. Tekiner, "On MANET routing protocols for mobility and scalability," in *Proc. Int. Conf. Parallel Distributed Comput., Applicat., Technol.*, pp. 451-456, Higashi Hiroshima, Japan, Dec. 2009.

[9] J. Nakasuwan and P. Raklua, "Performance comparison of AODV and OLSR for MANET," in *Proc. Int. Conf. Control Automation Syst. (ICCAS 2010)*, pp. 1974-1977, Gyeonggi-do, Korea, Oct. 2010.

[10] M. Barati, K. Atefi, F. Khosravi, and Y. A. Daftari, "Performance evaluation of energy consumption for AODV and DSR routing protocols in MANET," in *Proc. Int. Conf. Computer, Inform. Sci. (ICCIS 2012)*, pp. 636-642, Kuala Lumpur, Malaysia, June 2012.

[11] B. K. Panda, M. Das, B. Sahu, and R. Das, "Impact of mobility and terrain size on performance of AODV and DSR in Mobile Ad hoc network," in *Proc. Int. Conf. Wireless Optical Commun. Networks (WOCN 2012)*, pp.1-5, Indore, India, Sep. 2012.

[12] S. Mittal and P. Kaur, "Performance comparison of AODV, DSR and ZRP routing protocols in MANET'S," in *Proc. Int. Conf. Advances Comput., Control, Telecommun. Technol. (ACT '09)*, pp. 165-168, Trivandrum, India, Dec. 2009.

- [13] B. K. Panda, B. Dash, R. Das, and A. Sarangi, "Mobility and its impact on performance of AODV and DSR in mobile ad hoc network," in *Proc. 3rd Asian Himalayas Int. Conf. Internet (AH-ICI 2012)*, pp. 1-5, Kathmandu, Nepal, Nov. 2012.
- [14] D.-W. Kum, J.-S. Park, Y.-Z. Cho, and B.-Y. Cheon, "Performance evaluation of AODV and DYMO routing protocols in MANET," in *Proc. 7th IEEE Consumer Commun. Networking Conf. (CCNC)*, pp. 1-2, Las Vegas, U.S.A., Jan. 2010.
- [15] I. D. Chakeres and E. M. Belding-Royer, "The utility of hello messages for determining link connectivity," in *Proc. Int. Symp. Wireless Personal Multimedia Commun. (WPMC 2002)*. vol. 2, pp. 504 - 508, Honolulu, U.S.A., Oct. 2002.
- [16] H. Jeon, T. Kim, and J. Choi, "An approach for robust routing algorithms in Ad hoc network," in *Proc. 4th Int. Conf. Networked Comput., Advanced Inform. Manage. (NCM 2008)*, pp. 167-170, Gyeongju, Korea, Sep. 2008.

이 윤 경 (Yun-kyung Lee)



2000년 2월 숙명여자대학교
전자계산학과 졸업
2008년 2월 숙명여자대학교
전자계산교육과 교육학석사
2008년 9월~현재 숙명여자
대학교 컴퓨터과학과 박사과
정

<관심분야> 운영체제, MANET, Routing protocol

김 주 균 (Ju-gyun Kim)



1985년 서울대학교 계산통계학
과졸업
1985년~1986년 DEC Korea
근무.
1988년 서울대학교 계산통계학
과 계산학석사
1992년 서울대학교 계산통계학

과 계산학박사

1992년~현재 숙명여자대학교 이과대학 컴퓨터과학
전공 교수

<관심분야> 운영체제, 성능평가, Caching
Strategies, Flash Memory Technologies,
MANET.