한국시뮬레이션학회 논문지
Vol. 15, No. 3, pp. 79-86 (2006. 9)

애드혹 네트워크에서의 라우팅 성능 개선

박노열[1†] · 김상경[2] · 김창화[2]

An Improvement of Routing Performance in Ad Hoc Networks

Noyeul Park · Sangkyung Kim · Changhwa Kim

ABSTRACT

This paper proposes a new protocol to enhance the routing performance and applies our proposal to AODV (Ad hoc On-demand Distance Vector routing) protocol, which is a typical on-demand routing protocol in MANET. In general, when a route repair procedure is performed due to the transfer or failure of a node, the source node re-initiate a route discovery procedure in most routing protocols of MANET. This causes to increase the control traffic overhead largely and degrade the routing performance. The main idea of our study is to provide a methodology that can minimize the control traffic overhead and enhance the performance by initiating a route repair procedure at destination node and utilizing the previous routing information maximumly. To evaluate the performance of the our mechanism, we have performed a series of simulations. The simulations include comparisons with AODV and AODV with local repair. The simulations have been executed under simulation environments taking into account mobility speed and network size. The simulation results show that our proposal overall outperforms the existing protocols.

Key words : ad hoc network, on-demand routing, improvement of route performance, AODV

요 약

본 논문에서는 대표적인 on-demand 방식의 AODV 프로토콜을 개선하여 성능이 우수한 새로운 프로토콜을 제안 한다. 일반적으로 노드의 이동이나 장애에 의하여 경로복구절차가 수행될 때 대부분의 애드혹 라우팅 프로토콜은 발신지 노드에 의한 경로 탐색 절차를 다시 사작하게 된다. 이것은 제어 트래픽 오버헤드를 크게 증가시키고, 라우팅 성능은 저하시키는 원인이 된다. 본 논문에서는 목적지 노드에서 경로 복구 절차를 시작하고 이전의 라우팅 정보를 최대로 이용함으로써 라우팅의 성능을 개선하는 알고리즘을 제안하며 시뮬레이션을 통하여 제안 프로토콜의 성능을 검증하였다. 시뮬레이션은 AODV 및 L-AODV 프로토콜과 비교 수행하였으며, 네트워크 크기, 노드 이동속도 등을 고려한 환경에서 수행하였다. 시뮬레이션의 결과는 제안 프로토콜이 기존 프로토콜에 비해 전반적으로 좋은 성능을 가짐을 보여주었다.

주요어 : 애드혹 망, 요구형 라우팅, 경로 복구, 라우팅 성능개선, AODV

1. 서 론

모바일 애드혹 네트워크는 노드의 예측 가능하지 않은 이동 패턴, 링크 품질, 잔존 배터리의 양 등의 조건에 따라 네트워크 토폴로지가 역동적으로 변할 수 있다는 특징을 갖는다. 따라서 애드혹 네트워크에서의 라우팅 프로토콜은 노드간 통신을 유지하기 위하여 네트워크 연결에 대한 지속적인 추적을 필요로 하며, 토폴로지 변화에 적응적으로 대처할 수 있어야 한다.

일반적으로 애드혹 라우팅 프로토콜은 table-driven 방식과 on-demand 방식의 두 범주로 나눌 수 있다.[1]

Table-driven 방식은 네트워크 내의 각 노드에 대한 라우팅 정보를 라우팅 테이블에 계속하여 유지하는 방식으로 빠른 경로 탐색의 장점이 있으나, 사용하지 않는 경로에 대한 정보도 유지 관리함으로써 라우팅 오버헤드가 증가하는 단점이 있으며, 대표적인 예로는 DSDV(Desti-

* 이 논문은 2005년도 강릉대학교 학술연구조성비 지원에 의하여 수행되었음.
2006년 8월 1일 접수, 2006년 9월 2일 채택
[1] 강릉대학교 컴퓨터공학과
[2] 강릉대학교 정보전자공학부
주 저 자 : 박노열
교신저자 : 김상경
E-mail; skkim98@kangnung.ac.kr

nation Sequenced Distance Vector routing)[2] 프로토콜이 있다.

On-demand 방식은 송신할 데이터를 갖는 발신지 노드가 데이터를 전송하고자 할 때만 목적지 노드까지의 경로를 탐색하여 경로를 설정, 유지함으로써 table-driven 방식보다 제어 트래픽에 의한 오버헤드를 줄일 수 있으며, 대표적인 프로토콜로는 DSR(Dynamic Source Routing)[3]과 AODV(Ad hoc On-demand Distance Vector Routing)[4] 프로토콜이 있으며, 특히 AODV 프로토콜에 대해 많은 연구가 진행되어 왔다.

AODV 프로토콜은 노드의 이동이나 장애로 인해 데이터 전송경로가 손실되는 경우 발신지 노드에서 새로운 경로탐색 절차를 시작하며, 새로운 전송경로를 설정하여 데이터 패킷을 전송한다. 이때 새로운 경로 탐색 절차를 수행하는 동안 제어 트래픽 오버헤드와 데이터 패킷의 손실이 발생해 데이터 전송에 좋지 않은 영향을 미치고 있다. 이를 개선하기 위해 AODV에 local repair 알고리즘[4]을 적용하였으나, 크게 성능이 개선되지는 않고 있다.

본 논문에서는 AODV 프로토콜을 개선하여 목적지 노드에서 경로 복구절차를 시작함으로써 기존 설정된 경로를 최대한 이용하여 경로 복구과정에서 제어 트래픽 오버헤드 및 데이터 패킷의 손실을 최소화하기 위한 라우팅 프로토콜을 제안하고, 시뮬레이션을 통해 데이터 패킷 전달률, 제어 트래픽 오버헤드, 종단간 지연시간, hop 수 등의 관점에서 AODV 및 local repair알고리즘을 적용한 AODV와 비교하여 성능분석을 수행하였다.

논문 구성은 다음과 같다. 2절에서는 AODV와 관련된 기존의 경로복구 라우팅 프로토콜을 분석하고, 3절에서는 본 논문에서 제안하는 라우팅 성능 개선 알고리즘에 대해 기술한다. 4절에서는 시뮬레이션을 통한 성능분석 결과를 설명하고, 5절에서는 결론 및 향후 연구 과제에 대해 기술하고자 한다.

2. 관련 연구

AODV에서의 경로 복구 절차는 설정된 경로를 통하여 데이터 패킷 전송을 수행하는 도중 경로상에 장애가 발생하여 더 이상 데이터 패킷을 전송 할 수 없는 상황이 발생하면, 장애 발생 지점에 따라 local repair를 수행하거나, 또는 발신지 노드에서 새로운 경로 설정 절차를 수행한다.

MDAODV(Maximally Disjoint Multipath Routing Protocol Based on AODV)[5]는 기존의 AOMDV[6] 프로토콜이 노드 비중첩은 제공하지 않고 링크 비중첩만을 제공하는 단점을 개선하기 위해 제안된 프로토콜로서, 최대 노드 비중첩과 링크 비중첩을 제공하는 핵심 아이디어는 RREQ(Route Request)에 소스 라우팅 주소 정보를 추가시켜 확장하는 것과 수신 노드가 미리 정해진 일정시간 동안 도착한 다수의 RREQ들 중에서 2개의 경로를 선택하는 것이다. 노드 고장이나 링크의 손상과 같은 예상치 못한 장애 때문에 이용할 수 없게 될 때까지 주 경로를 통해서 데이터 패킷을 전송한다. 주 경로를 이용할 수 없게 되면, 대체 경로를 통해 즉시 데이터 패킷 전송이 이루어진다. 만약 주 경로와 대체 경로 둘 다 이용할 수 없게 되면 새로운 경로를 다시 설정하기 위하여 경로 탐색 절차를 수행한다.

Dynamic Source Routing(DSR)[7]은 소스 라우팅 방법에 기초하고 있으며 모든 노드는 경로 정보를 라우팅 캐쉬에 유지하고 있다. DSR라우팅 프로토콜에서 경로 상의 링크 장애 발생시 RERR 메시지를 생성하여 발신지 노드로 전달한다. RERR 메시지를 수신한 발신지 노드는 새로운 경로 탐색을 위한 절차를 수행한다.

PATCH(Proximity Approach To Connection Healing Algorithm)[8]는 DSR 프로토콜을 개선하기 위하여 local repair 알고리즘을 적용한 프로토콜로서, 전송경로 설정 절차는 DSR과 동일한 방법으로 이루어지며, 설정된 경로를 통하여 데이터 패킷 전송 도중 경로상에 장애가 발생하여 더 이상 데이터 패킷을 전송할 수 없는 상황이 발생하면, 장애 발생 지점의 상위 노드에서 장애를 감지하게 되고, 장애를 감지한 노드는 local repair 알고리즘을 적용하여 경로 복구절차를 수행한다.

3. 제안 프로토콜

그림 1에서와 같이 설정된 경로를 통하여 데이터 패킷 전송을 수행하는 도중 그림 2와 같이 전송 링크의 장애 또는 목적지노드의 이동으로 인하여 전송 링크가 끊어지면 다음과 같은 과정을 통해 전송경로 복구절차를 수행하며, 단계적인 그림으로 나타내면 그림 3과 같다.

① 경로 장애 통지
② 목적지 노드에 의한 경로탐색
③ 경로복구

3.1 경로 장애 통지

제안 프로토콜은 AODV 프로토콜에서와 같이 hello 패킷을 이용하여 링크의 장애를 감지하게 된다. 링크장애

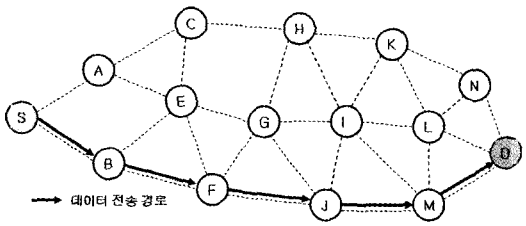

그림 1. 설정된 경로 예시

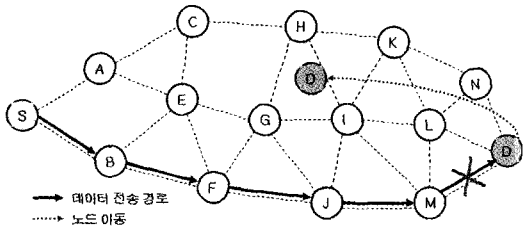

그림 2. 목적지노드의 이동에 의한 경로장애 예시

를 감지한 상위노드는 발신지 노드를 향하여 RERR (Router Error) 패킷을 전송하여 더 이상 데이터 패킷의 전송으로 인한 데이터 패킷 손실을 막을 수 있도록 라우팅 테이블의 상태를 수정한다. 그림 2에서와 같이 목적지 노드가 이 동하여 더 이상 데이터 패킷을 목적지 노드로 전송할 수 없는 경우 노드 M은 발신지 노드 S를 향하여 RERR 패킷을 생성하여 라우팅 테이블에 기록된 상위노드로 전달한다. 즉, RERR 패킷은 노드 M-J-F-B-S의 경로를 통하여 발신지 노드 에 도착하게 되고 RERR 패킷을 수신한 전송경로상의 노드들은 다음과 같이 동작한다.

- 수신한 RERR 패킷이 자신이 전송한 패킷이거나 중복 수신한 패킷이면 폐기한다.
- 노드 S가 아니고, 중복 수신한 패킷이 아니라면, 라우팅 테이블의 플래그를 RTF_IN_REPAIR"로 설정하고 라우팅 테이블에 기록된 상위 노드로 RERR 패킷을 전달한다.
- 수신 노드가 노드 S이면, 라우팅 테이블의 플래그를 "RTF_IN_REPAIR"로 설정하고 패킷을 폐기한다.

한편, 링크 장애를 감지한 하위노드는 목적지 노드를 향하여 RERR 패킷을 전송하여 목적지 노드가 경로 복구 절차를 시행토록 한다[1].

1) 그림 2의 예 에서는 링크장애 감지노드가 목적지 노드이므로 이 절차가 생략된다.

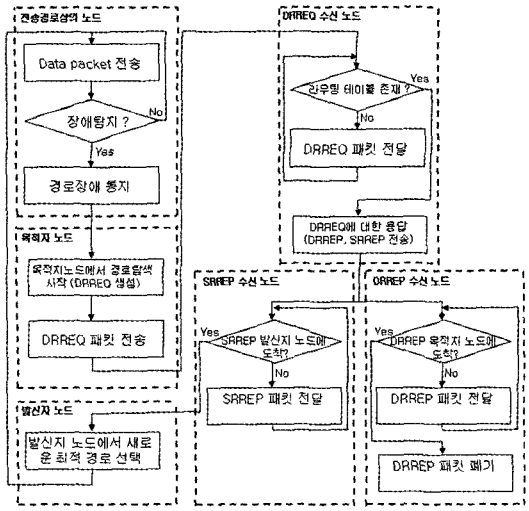

그림 3. 전송 경로 복구 절차

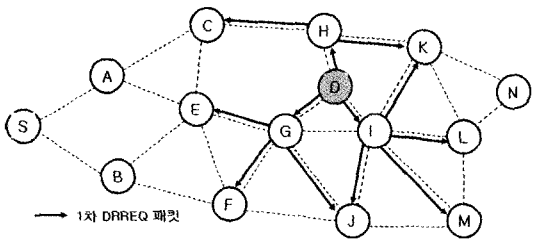

그림 4. DRREQ 패킷의 flooding 과정 예시

3.2 목적지 노드에 의한 경로탐색

경로 장애를 인지한 목적지 노드 D는 DRREQ(Destination Route Request) 패킷을 생성하여 이웃 노드들에 플러딩 하면서 경로 복구를 위한 탐색 절차를 시작한다. DRREQ 패킷은 그림 4과 같이 이웃노드로 플러딩 되며, 이 때 제어 패킷으로 인한 오버헤드를 최소화하기 위해 DRREQ 패킷의 전달 범위는 최초 2홉으로 설정하여 플러딩 하고 DRREQ 패킷에 대한 응답이 없을 경우 전달 범위를 1홉씩 증가하여 전송한다. DRREQ 패킷을 수신한 노드들은 다음과 같은 동작을 수행한다.

- 수신한 DRREQ 패킷이 자신이 전송한 패킷이면, 패킷을 폐기한다.
- 자신이 DRREQ 패킷을 중복 수신하였다면, 수신한 패킷을 폐기한다.
- 장애경로에 대한 라우팅 정보를 갖지 않고 수신한 DRREQ 패킷이 새로운 것이라면, DRREQ 패킷을 송신한 노드를 다음 홉으로 라우팅 테이블에 기록하고 DRREQ 패킷을 주변 노드로 플러딩 한다. 기록

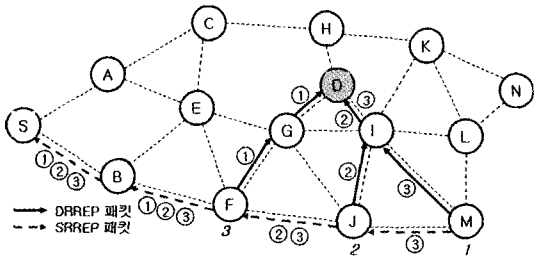

그림 5. 경로복구 응답 과정

된 정보는 경로복구 응답과정에서 상위 노드를 설정하는데 이용된다.

- 자신이 해당 목적지 노드로의 라우팅 테이블을 유지하고 있으면, 경로 탐색 응답을 수행하기 위해 DRREQ 패킷을 송신한 노드를 다음 홉으로 라우팅 테이블에 기록하고 경로 탐색 응답 과정을 수행하기 위해 DRREP 패킷과 SRREP(Source Route Request) 패킷을 생성하여 전송한다.

그림 4의 예에서는 목적지 노드 D와 목적지 노드에 대한 라우팅 정보를 가지고 있는 노드 F, 노드 J, 노드 M과 2홉 거리에 있으므로 DRREQ 패킷이 도착하기 위해서는 1번의 DRREQ 패킷의 플러딩이 필요하다.

경로탐색 절차에 대한 응답은 DRREP 패킷과 SRREP 패킷의 전송으로 이루어진다. DRREP 패킷은 그림 5와 같이 DRREQ 패킷이 전달된 순서의 역으로 전달되며, DRREP 패킷을 수신한 노드들은 패킷을 전송한 노드를 상위 노드로 기록하고 경로탐색 과정에서 기록한 다음 홉으로 전달한다. 목적지 노드가 수신하면, 경로탐색절차를 중단하고 수신한 DRREP 패킷을 폐기한다.

SRREP 패킷은 그림 5와 같이 노드의 라우팅 테이블에 기록된 상위노드 정보를 이용하여 노드 S에 도착할 때까지 전달되며, SRREP 패킷을 수신한 노드들은 다음과 같은 동작을 수행한다.

- 수신 노드가 노드 S가 아니면, 패킷을 전송한 노드를 데이터 패킷의 목적지 노드 D를 위한 다음 홉으로 수정하고 라우팅 테이블에 기록된 상위 노드로 전달한다.
- 수신 노드가 노드 S이면, 경로 복구과정을 수행한다.

3.3 경로복구

경로 복구절차는 SRREP 패킷이 발신지 노드에 도착하고, 발신지 노드가 수신된 SRREP 전달경로 중 하나를 데이터 전송을 위한 최종 전송경로로 선택함으로써 완료하게 되며, 전송경로를 선택하기 위해 발신지 노드는 다음과 같이 동작한다.

- 발신지 노드는 다음에 도착하는 SRREP 패킷들과 비교하여 더 효과적인 전송경로를 선택하기 위해 주어진 수신대기시간까지 임시 테이블에 저장한다.
- 주어진 수신 대기 시간이 종료되면, 발신지 노드는 수신된 각 SRREP 패킷 내의 destination hop count 필드의 내용을 참조하여 그 값이 가장 큰 SRREP 패킷이 전달된 경로를 최종 전송경로로 선택하게 된다.

Destination hop count 필드의 값은 장애 발생 전 설정된 경로상의 각 노드에서 목적지 노드까지의 거리를 나타내므로 값이 클수록 발신지 노드와 가까이 위치한다는 것을 의미한다. 즉, 발신지 노드 가까이에 위치한 노드를 선택할수록 전송 거리가 짧아져 더 효과적인 전송경로를 설정할 수 있다. 그림 5의 예 에서는 데이터 전송경로 복구를 위한 DRREQ 패킷에 대한 경로탐색 응답 과정을 나타내며, 목적지노드(노드 D)에 대한 라우팅 정보를 알고 있는 노드 F와 노드 J 그리고 노드 M에의해 ①, ② 및 ③의 경로로 각각 DRREP 패킷과 SRREP 패킷을 전달한다. 노드 아래의 이탤릭 숫자는 라우팅 테이블에 기록된 목적지 노드까지의 거리(hop count)를 나타낸다. 따라서 그림 5의 경우 hop count가 더 큰 노드 F에서 전송한 SRREP 패킷이 전달된 경로를 최종 전송경로로 선택하게 된다.

4. 성능분석

본 논문에서는 Network Simulator 2(NS-2)[9]를 이용하여 시뮬레이션을 수행하였으며, 제안 프로토콜의 성능을 AODV[4] 및 local repair 알고리즘을 적용한 L-AODV 라우팅 프로토콜과 비교 분석하였다. NS-2에서 제공하는 IEEE 802.11 무선 및 MAC 모델이 시뮬레이션 상의 노드들에 적용되었으며, 트래픽 데이터를 생성하기 위해 CBR (Constant Bit Rate) 발신지를 사용하여 각 시뮬레이션 환경에 따른 시나리오를 생성하였고, 트래픽의 페이로드 크기는 512바이트, 데이터 전달속도는 4패킷/초이다. 노드의 이동은 다음과 같은 Random Waypoint 모델[10]을 따르도록 설정하였다.

① 각 노드가 초기에 시뮬레이션 영역내의 임의의 장소에 위치한다.

② 시뮬레이션이 진행됨에 따라 각 노드는 현재 위치에서 일정시간 머문다.

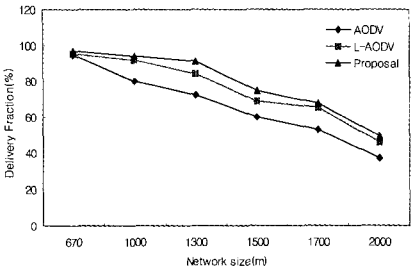

그림 6. 네트워크 크기에 따른 전달률

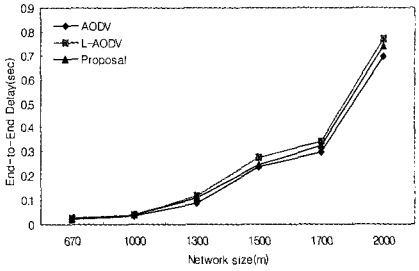

그림 7. 네트워크 크기에 따른 지연시간

③ 이후 임의의 새로운 위치를 목적지로 선정하여 0과 최대 속도 사이에서 균일하게 분포된 속도로 이동한다.

④ 목적지 노드에 도착하면 노드는 다시 정지시간 동안 머문 후 새로운 위치를 선정하여 이동하는 과정을 반복한다.

다양한 환경에서의 실험을 위하여 네트워크의 크기, 노드 수 등의 조건을 변화시키면서 성능분석을 실시하였다. 시뮬레이션의 결과는 최소 5회 이상의 실험을 통하여 얻어진 것이며, 성능 평가 척도는 패킷 전달율(packet delivery fraction), 종단간 평균지연시간(end-to-end delay), 제어 트래픽 오버헤드(control traffic overhead), 홉수(hop count)등을 적용하였다.

4.1 시뮬레이션 유형 I

시뮬레이션 네트워크의 크기를 변화시켜가면서 AODV 및 L-AODV와 비교 분석하였다. 본 시뮬레이션은 네트워크의 크기를 670 x 670 미터에서 2000 x 2000 미터로 변화시키며, 네트워크 내에 60개의 노드, 10m/s의 이동속도, 150 미터의 무선 전파범위로 네트워크를 모델링하였으며, 시뮬레이션 시간은 1,000초, 트래픽 발신지의 수는 20개로 설정하였다.

그림 6은 네트워크 크기에 따른 전달률을 보여준다. 전달률은 발신지 노드에서 전송한 데이터가 목적지노드에 정상적으로 도착한 비율을 나타내며, 데이터 패킷들은 노드 장애로 인한 경로 단절, 혼잡, 충돌 등에 의해서 목적지노드로 전달되는 도중 손실이 발생할 수 있다. 제안 프로토콜은 전반적으로 다른 2개의 프로토콜보다 높은 전달률을 보였으며, 네트워크 크기가 아주 작은 경우 AODV나 L-AODV 프로토콜과 전달률에 큰 차이가 없었고, 1500x1500의 네트워크 크기에서 AODV에 비해 약 24.5%. L-AODV에 비해 약 8.2%로 가장 높은 성능개선

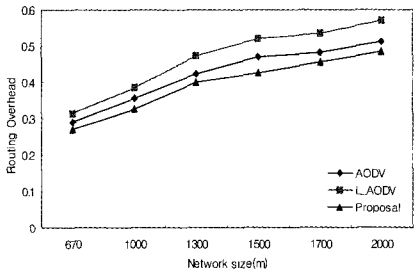

그림 8. 네트워크 크기에 따른 제어트래픽 오버헤드

효과를 보였는데, 이는 네트워크 크기에 적당한 수의 노드가 네트워크 전체에 고르게 위치하여 효과적으로 경로 설정 및 복구가 이루어 질수 있기 때문인 것으로 판단된다. 전체적으로 AODV에 비해 약 22.1%, L-AODV에 비해 약 5.14%의 전달률 개선효과를 보였다.

그림 7은 네트워크 크기 변화에 따른 데이터 패킷의 종단간 지연을 나타낸다. 3개의 프로토콜 모두 비슷한 수준의 지연을 나타내며, 네트워크의 크기가 증가함에 따라 지연시간이 급격히 증가하는 것으로 나타났으며, AODV에 비해 제안 프로토콜의 지연시간이 조금 높게 나타났다. 전송경로 단절시 경로복구를 수행하기 위한 제어 패킷인 SRREP 패킷을 발신지 노드에서 일정 시간동안 여러 개 수신하여 보다 효율적인 전송경로를 선택하기 위해 데이터 패킷의 전송을 일정시간 대기하게 되는데 이러한 대기시간에 의해 지연시간이 길어지는 것으로 판단된다.

그림 8은 각 라우팅 프로토콜의 제어 트래픽 오버헤드를 나타낸다. 제어 트래픽 오버헤드는 발신지 노드로부터 목적지 노드까지 데이터 트래픽을 성공적으로 전달하기 위해 얼마나 많은 제어 트래픽이 요구되는지를 나타낸다. 네트워크의 크기가 작은 경우 오버헤드가 적게 발생하고 네트워크 크기가 증가함에 따라 오버헤드가 증가하는 것으로 나타났다. 네트워크 크기가 작은 경우 노드가 밀집

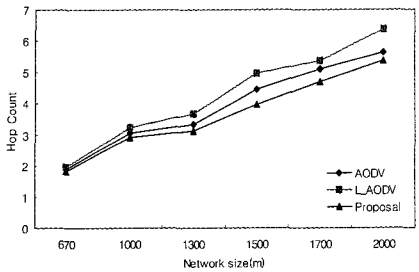

그림 9. 네트워크 크기에 따른 hop count

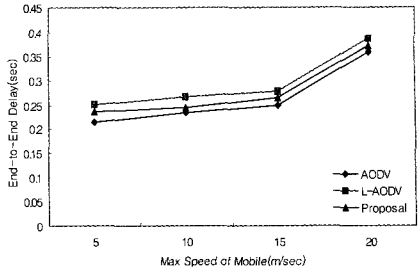

그림 11. 노드 이동속도에 따른 지연시간

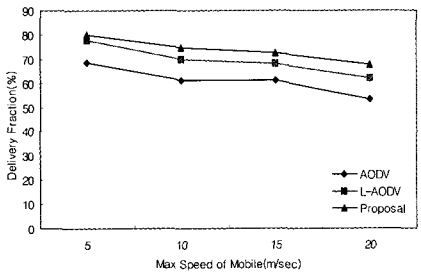

그림 10. 노드 이동속도에 따른 전달률

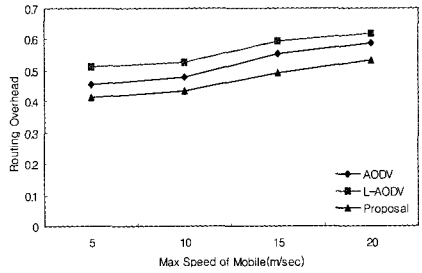

그림 12. 노드 이동속도에 따른 오버헤드

되어 있어 경로 장애 및 경로 재설정이 적게 일어나므로 오버헤드가 적게 발생하고, 네트워크 크기가 증가하면, 노드들이 서로 먼 거리에 위치하게 되어 경로 유지 가능성이 떨어지게 된다. 따라서 경로 유지 및 경로 복구를 위한 제어 트래픽의 발생으로 인하여 오버헤드가 증가하는 것으로 판단된다. 본 시뮬레이션에서는 네트워크의 크기에 따라 평균적으로 AODV에 비해 약 9.6%, L-AODV에 비해 약 15.6%의 개선효과를 나타냈다.

그림 9는 각 라우팅 프로토콜의 hop count를 나타낸다. hop count는 발신지 노드와 목적지 노드 간에 패킷 전달에 참여하는 노드의 수를 나타낸다. 네트워크의 크기가 작은 경우 3개의 프로토콜이 비슷한 hop count를 가지지만 크기가 증가하면서 차이가 뚜렷해짐을 보였다. 제안 프로토콜의 경우 경로 복구를 수행하는 경우에 기존 경로 중 발신지 노드와 가장 가까운 노드를 이용하여 경로가 복구되므로 hop count가 감소하는 것으로 판단된다. 전체적으로 제안 프로토콜은 AODV에 비해 약 6.2%, L-AODV에 비해 약 13.3%의 성능 개선효과를 보였다.

4.2 시뮬레이션 유형 II

네트워크 내의 노드 이동 속도를 변화시켜가면서 AODV 및 L-AODV와 비교 분석하였다. 본 시뮬레이션은 네트워크의 크기가 1500 x 1500 미터 크기 내에 60개의 노드가 5m/s에서 20m/s의 사이의 속도로 이동하는 네트워크를 모델링하였다. 노드의 무선 전파범위는 150 미터, 시뮬레이션 시간은 1000초이고, 트래픽 발신지의 수는 20개로 설정하였다.

그림 10은 네트워크 내의 노드 이동속도에 따른 패킷 전달률을 나타낸다. 노드의 이동 속도가 증가함에 따라 전달률이 감소하였으나, 제안 프로토콜은 전체적으로 AODV에 비해 약 21.18%, L-AODV에 비해 약 6.18%의 전달률 개선효과를 보였다.

그림 11은 네트워크 내의 노드 이동속도에 따른 데이터 패킷의 종단간 지연시간을 나타낸다. 3개의 프로토콜 모두 이동 속도가 15m/s이하일 때는 비교적 낮은 패킷 지연을 보이다가 이동 속도가 20m/s로 빨라지면서 지연 시간이 현저히 높아지는 것으로 나타났다. 전반적으로 AODV에 비해 L-AODV와 제안 프로토콜의 지연시간이 조금 긴 것으로 나타났다.

그림 12는 네트워크 내의 노드 이동 속도에 따른 각 라우팅 프로토콜의 제어 트래픽 오버헤드를 나타낸다. 노드의 이동 속도가 빨라질수록 발신지 노드에서 목적지 노드까지의 경로장애가 자주 발생하여 잦은 경로 복구절차 수행으로 인한 제어 트래픽이 증가하였으나, 제안 프로토콜

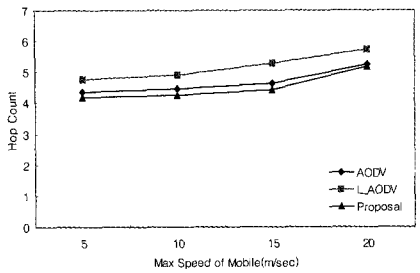

그림 13. 노드 이동속도에 따른 hop count

이 AODV에 비해 약 9.46%, L-AODV에 비해 약 16.8%의 성능개선 효과를 보였다.

그림 13은 노드의 이동 속도에 따른 각 라우팅 프로토콜의 hop count를 나타낸다. 제안 프로토콜은 AODV에 비해 약 3.6%, L-AODV에 비해 약 12.7%의 성능개선 효과를 보였다.

5. 결 론

본 논문에서는 모바일 애드혹 네트워크 환경에서 라우팅 프로토콜의 성능을 개선하기 위한 방법으로서 데이터 패킷 전송경로 장애 발생시 목적지 노드에서 기존에 기록된 라우팅 정보를 최대한 이용할 수 있는 경로 복구절차를 수행하여 데이터 패킷 전달률은 최대화하고, 제어 트래픽 오버헤드 및 데이터 패킷의 손실은 최소화하기 위한 새로운 on-demand 방식의 라우팅 프로토콜에 대해 제안했다.

제안 프로토콜의 핵심 아이디어는 노드들이 기존 위치에서 짧은 시간에 많이 멀어지지 않고, 목적지 노드가 발신지 노드 가까이로 이동할 수도 있으므로 목적지 노드에서 경로 복구절차를 시작하면, 기존에 설정된 라우팅 정보를 최대한 이용하여 경로 복구절차를 효과적으로 수행할 수 있다는 것이다.

NS-2 시뮬레이터를 이용하여 네트워크의 크기, 노드의 이동 속도 등을 변화시켜가면서 다양한 환경에서 성능 분석을 실시한 결과 제안 프로토콜은 기존의 AODV 프로토콜 및 L-AODV 프로토콜에 비해 데이터 패킷 전달률과 제어 트래픽 오버헤드, 홉 수 등의 성능 평가 척도에서 AODV 프로토콜에 비해 약 18.3%, 7.7%, 5.7%가 개선되었고, L-AODV에 비해서는 약 5.12%, 15.6%, 12.5%의 높은 성능 개선 효과를 나타냈다. 종단간 패킷 지연시간(end-to-end delay)은 AODV에 비해 약 12.7% 길게 나타났고, L-AODV에 비해서는 약 6.7% 짧게 나타났다.

향후에는 본 논문에서 제안한 프로토콜의 종단간 지연시간을 줄이고 보다 다양한 네트워크 환경에 적용하기 위한 연구를 진행할 계획이다.

참 고 문 헌

1. E. M. Royer, C-K Toh. "A review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks" *IEEE Personal Communication*, pp. 46-55 April 1999.
2. C. E. Perkins and T. J. Watson. (1994), "Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers," *Proc. of Int. Conf. on Communications Architectures(ACMSIGCOMM'94)*, Oct, pp. 234-244.
3. D. B. Johnson and D. A. Maltz. "Dynamic Source Routing in AdHoc Wireless Networks," *in Mobile computing, T. Imielinski and H. Korth, Eds. Kluwer Academic Publishers*, pp. 153-181, 1996.
4. C. Perkins, E. Belding-Royer, S. Das. (2003), "RFC3561 Ad hoc On-demand Distance Vector (AODV) Routing,"
5. 김중태, 모상만, 정일웅, (2005) "모바일에 드혹 네트워크에서의 AODV 기반 최대 비중첩 다중경로 라우팅 프로토콜," *정보처리학회논문지, 제12-C권 제3호*, pp. 429-436
6. M. K. Marina and S. R. Das, "Ad Hoc On-demand Multipath Distance Vector Routing", *http://www.ececs.uc.-edu/~sdas, Technical Report, Computer Science Department, Stony Brook University*, Apr, 2003.
7. D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *in Mobile computing, T. Imielinski and H. Korth, Eds. Kluwer Academic Publishers*, pp. 153-181, 1996.
8. Genping Liu, Kai-Juan Wong, Bu-Sung Lee, Boon-Chong Seet, Chuan-Heng Foh, Lijuan Zhu. (2003), "PATCH: A Novel Local Recovery Mechanism for Mobile Ad-hoc Networks," *IEEE vehicular technology conference*, pp. 2995-2999.
9. The Network Simulator ns-2, (2006), *http://www.isi.edu/nsnam/ns/*.
10. T. Camp, J. Boleng and V. Davies. (2002), "A survey of Mobility Models for Ad Hoc Network Research", *Wireless Communications & Mobile Computing (WCMC), Special issue on Mobile Ad Hoc Networking : Research, Trends and Applications*, pp. 483-502.

박 노 열 (eagle@kangnung.ac.kr)

2003년 한국방송통신대학교 컴퓨터과학과 이학사
2006년 강릉대학교 대학원 컴퓨터공학과 공학석사
2006년 강릉대학교 대학원 컴퓨터공학과 박사과정
1988년 서울대학교 중앙교육연구전산원
1990년~현재 강릉대학교 정보전산원

관심분야 : 애드혹 망, 센서 네트워크, 라우팅 등

김 상 경 (skkim98@kangnung.ac.kr)

1985년 고려대학교 전자공학과 공학사
1987년 고려대학교 대학원 전자공학과 공학석사
1989년 KT (선임연구원/부장)
1994년 TINA-C Core Team Member
2003년 고려대학교 대학원 전자공학과 공학박사
2004년~현재 강릉대학교 정보전자공학부 조교수

관심분야 : 이동망, 서비스 구조, 미들웨어 등

김 창 화 (kch@kangnung.ac.kr)

1985년 고려대학교 수학교육과 이학사
1987년 고려대학교 대학원 전산학 전공 이학석사
1990년 교려대학교 대학원 전산학 전공 이학박사
1989년~현재 강릉대학교 정보전자공학부 교수
1994년 Post-Doc./Visiting Professor, Toro-nto대학 Enterprise Integration 연구소
2002년 Visiting Scholar, Dept of Computer Science, Texas A&M University
2005년~현재 강릉대학교 해양센서네트워크시스템기술연구센터(ITRC) 소장
2005년~현재 한국시뮬레이션학회 논문지 편집위원
2006년~현재 강릉정보산업진흥원 이사

관심분야 : 분산시스템, 미들웨어, 지능형시스템