

단방향 링크 Ad-hoc 망을 위한 라우팅 프로토콜*

박상호**

요약

본 논문에서는 다중 홉 ad-hoc 네트워크가 단방향 경로를 포함하고 있는 경우 단방향 링크를 통하여 패킷을 전송하기 위한 효율적인 라우팅 프로토콜을 제안한다. 제안하는 프로토콜은 노드에서 패킷의 송신과 수신이 가능하도록 두개의 라우팅 테이블로 구성되어 있다. 프로토콜은 양방향 링크들로만 구성된 네트워크와 단방향 링크를 포함하는 네트워크 모두에 운용이 가능하다. 제안하는 프로토콜은 양방향 링크뿐만 아니라 단방향 링크도 고려한 방식이므로 패킷의 송신과 수신을 위한 경로 생성 시 단방향 링크에 대한 경로생성이 내재된다.

Routing Protocol for Ad-hoc Networks with Unidirectional Links

Sangho Park**

ABSTRACT

We proposed a simple and efficient routing protocol designed specifically for multi-hop ad-hoc networks. The proposed protocol is composed of the two routing tables which work together to allow nodes to send and receive packets. All aspects of the protocol can be operated on networks entirely consisting bidirectional links and networks with unidirectional links. Finding unidirectional link is embedded in the routing protocol since we considered not only bidirectional links but also unidirectional links when we developed a routing method.

Key words : Ad-hoc Networks, Unidirectional Links, Routing Algorithm

* 이 논문은 2003년도 안동대학교 학술연구지원사업에 의하여 연구되었음.

** 안동대학교 전자정보산업학부

1. 서 론

Ad-hoc 네트워크 라우팅 프로토콜[1, 2]는 네트워크 내의 노드들이 양방향 링크로 연결되어 있음을 전제로 하고 있다. 그러나 유선망과 달리 무선망에서는 전파의 전파특성과 단말기의 송신전력의 차이에 의하여 네트워크에 단 방향 링크가 발생하는 경우가 있다. 단 방향 링크는 두 노드 간에 한 쪽 방향으로만 통신이 가능한 경우이다. 두 노드가 양방향 링크로 연결되어 있는 경우에는 노드 간에 하나의 경로만 설정하면 되지만, 두 노드가 단 방향 링크로 연결된 경우에는 노드간의 통신을 위하여 전송을 위한 경로와 수신을 위한 경로를 각각 설정해야 한다.

Prakash[3, 4]는 단 방향 링크를 갖는 ad-hoc 네트워크를 위한 라우팅 프로토콜을 제안하였다. 단 방향 링크를 갖는 ad-hoc 네트워크에서는 $O(n)$ 보다 큰 테이블을 유지해야 함을 보였다. [3, 4]에서 각 노드는 $O(n^2)$ 크기의 저장 공간을 필요로 한다. Lau[5]은 위성을 이용한 DBS(direct broadcast system)과 이동 무선망의 단 방향 링크를 위한 라우팅 프로토콜을 제안하였다. 네트워크의 각 노드들은 두 개의 라우팅 테이블 FT(From Table)과 TT(To Table)을 가지고 있으며 각 노드는 크기가 $O(n)$ 인 테이블을 유지하고 있다.

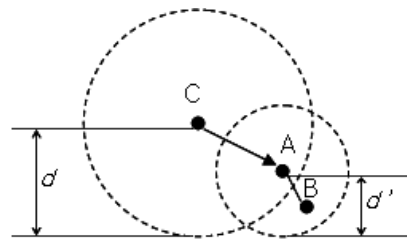
양방향 링크만을 고려하는 프로토콜[1, 2]에서는 두 노드 간에 단 방향 링크가 발생하면 그 링크는 연결되어 있지 않은 것으로 간주하고 다중 홉(multi-ho)을 통하여 두 노드를 연결할 수 있는 양방향 경로를 찾는다. 단 방향 링크를 고려하는 프로토콜에서는 두 노드 간에 단 방향 링크가 발생하면 두 노드 간에 링크가 형성된 방향으로만 그 링크를 사용하고, 링크가 형성되지 않은 방향으로만 다중 홉을 이용하여 새로운 링크를 찾아 두 노드간의 링크를 형성하여야하므로 노드 간에 링크를 형성하는 과정에 단 방향 링크를 찾는 과정이 내재되어있어야 시그널링을 위한 신호의 양을 줄이고

링크를 찾는 시간을 절약 할 수 있다.

본 논문에서는 단방향 링크를 갖는 ad-hoc 네트워크에서 패킷전송을 위하여 출발지와 목적지 간의 경로를 찾는 알고리즘을 제안한다. 제안하는 라우팅 알고리즘은 거리벡터를 이용한 라우팅 방식 [1, 2]를 대상으로 하며 라우팅 알고리즘[5]과 같이 양방향으로 라우팅 테이블을 유지하고 각 노드의 테이블 크기는 $O(n)$ 이다.

2. 단 방향 링크를 갖는 네트워크

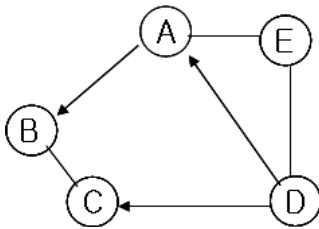
Ad-hoc 네트워크와 같은 다중 홉 무선 네트워크에서 단 방향 링크가 발생하는 이유는 무선 단말 장치의 송신전력의 차이에 의한 것과 각 노드에 영향을 주는 채널 간섭의 세기의 차이에 의한 것이 주요한 요인이다. (그림 1)에서 노드 C의 전파지역은 반지름이 d 인 원이고 노드 A의 전파지역은 반지름이 d' 인 원이다. A와 C의 송신전력의 차이에 의하여 C에서 A로 신호가 전달되나 A에서 C로는 신호가 전달되지 않는다. 따라서 A와 B는 양방향 링크가 형성되나, C에서 A는 단 방향 링크가 발생하게 된다.



(그림 1) 송신전력의 차이에 의한 단 방향 링크의 발생

네트워크가 양방향 링크들로만 구성되어 있다고 가정한 프로토콜[1, 2]에서는 단 방향 링크가 발생하면 그 링크를 네트워크에서 제외하고 두 노드 간에 새로운 경로를 찾게 된다. 그러나 단 방향 링

크가 존재하는 것을 가정하면 단 방향 링크는 그대로 사용하고 발생한 단 방향 링크에 대한 역방향 경로만 새로이 찾으면 된다. 단 방향 링크가 발생했을 때, 단 방향 링크를 포함하는 라우팅 방법은 양방향 링크만을 가정한 라우팅 방법보다 네트워크의 연결성을 향상시킬 수 있다.



(그림 2) 단 방향 링크가 있는 네트워크

(그림 2)에 단 방향 링크가 존재하는 네트워크를 나타내었다. 그림에서 화살표가 있는 링크는 단 방향 링크를 나타내고, 화살표가 없는 링크는 양방향 링크를 나타낸다. 양방향 링크만 고려하는 경로설정 방법에서 단 방향 링크는 양방향으로 패킷의 전송이 불가능하므로 두 노드 간에 링크가 설정되지 않은 것으로 간주하게 된다. (그림 2)에서 A-B로 패킷을 전송하는 경우 단 방향 링크를 이용하면 A-B로 하나의 홉으로 통신이 가능하나 단 방향 링크를 제거하고 양방향 링크만 사용하면 A-E-D-C-B로 네 개의 홉이 필요하다. 그러나 단 방향 링크를 이용한 라우팅은 양방향 링크로만 되어있는 라우팅보다 복잡하며 부하(overhead)가 크다[5].

양방향 링크 네트워크를 가정한 라우팅 프로토콜에서는 hello 신호를 이웃한 노드에 정기적으로 발생시켜 응답신호가 없으면 양방향 링크가 아님을 알 수 있다. 본 논문에서 제안하는 프로토콜은 양방향 링크 뿐 만 아니라 단 방향 링크도 고려한 방식이므로 패킷의 전송과 수신을 위한 경로 생성시 단 방향 링크로 인하여 송신경로와 수신경로가 다르면 두 개의 경로를 각각 찾으므로 단 방향 링

크의 존재와 경로발견이 프로토콜 내에 내재되게 된다.

3. 단 방향 링크를 갖는 ad-hoc 망의 라우팅 프로토콜

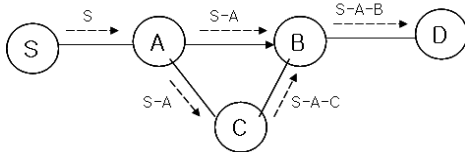
최근 단 방향 링크를 갖는 무선 ad-hoc 네트워크를 위한 경로설정 알고리즘이 제안되었다[3-5]. 노드가 n 개이며 단 방향 링크가 있는 네트워크를 위한 [3, 4]의 방법은 각 노드를 위하여 크기가 $n \times n$ 인 행렬 D 를 구성한다. 단 방향 링크를 발견하고, 단 방향 링크를 이용한 경로를 발견하기 위하여 D 는 네트워크 내의 각 노드 i 와 노드 j 를 위한 경로정보 $D[i, j]$ 로 구성되어있다. $D[i, j]$ 와 $D[j, i]$ 가 같으면 노드 i 와 j 는 양방향 링크로 연결되어 있고, 다르면 두 노드 간에는 단 방향 링크로 연결된 경우이다. 각 노드는 이웃노드로부터 정보를 수집하여 크기가 $O(n^2)$ 인 행렬 D 의 내용을 갱신한다.

본 논문에서는 각 노드가 크기가 $n \times n$ 인 모든 노드에 대한 경로정보를 저장하는 대신 크기가 n 인 이웃노드에 대한 송신경로를 위한 라우팅 테이블과 크기가 n 인 이웃노드에 대한 수신경로를 위한 라우팅 테이블을 유지함으로써 단 방향 링크를 활용한 경로를 설정할 수 있는 방법을 제안한다. 제안된 프로토콜은 정기적인 테이블 갱신(table driven)에 의한 라우팅과 패킷전송의 요구가 발생하였을 때 테이블을 갱신하는 경우(on-demand driven)에 적용 가능하다. 네트워크를 구성하고 있는 노드들은 IP를 갖지 않는 무선 네트워크로 가정하였다.

프로토콜은 두개의 단계로 나눌 수 있는데 하나는 송신을 위한 경로발견 단계이고 다른 하나는 수신을 위한 경로응답 단계이다. 경로발견 단계에서는 패킷전송을 하고자하는 출발지 S (source)에서 최종목적지 D (destination)까지의 경로거리가

가장 짧은 최단 경로를 찾게 된다. 여기서 경로거리는 문제정의를 간단히 하기 위하여 경로에서 거쳐야하는 노드의 수 즉 홉의 수로 한다.

노드 S는 route request를 이웃 노드들에 브로드캐스트한다. route request에는 출발지 주소, 목적지 주소를 포함하며 동일한 경로발견 요구가 노드에 도착 되었을 때 중복작업을 피하기 위한 request id와 송신경로 기록을 포함하고 있다. 노드 S에서 route request 발생 시 송신경로기록에는 신호를 발생한 노드인 S가 기록되어 있다. 노드 S의 route request를 수신한 노드들은 송신경로를 기록한 다음 이웃노드에 route request를 발송한다. 만약 노드들이 동일한 request id를 갖는 두 개 이상의 신호를 수신하면 경로거리가 가장 작은 request 신호를 취한다. 각 노드에서 송신경로를 기록한 다음 route request를 이웃노드에 브로드캐스트하는 과정을 목적지에 도달할 때 까지 반복한다. 이 과정을 (그림 3)을 이용하여 살펴보자.



(그림 3) 단 방향 경로를 갖는 네트워크에서 경로를 발견하는 과정

노드 S가 노드 D와 패킷을 송수신하기 위한 경로를 찾으려면, 출발지 주소 S, 도착지 주소 D, request id, 경로기록이 포함되어있는 route request 패킷을 발신한다. S에서 발생된 request 신호를 노드 A가 수신한다. 노드 A는 route request 패킷에서 경로기록에 A를 추가하여 경로기록을 S-A로 갱신하고 경로기록과 request id를 버퍼에 저장한 후 request 패킷을 발송한다. 노드 A에서 발송된 패킷은 노드 B와 C에 도착한다. 노드 B는 경로기록을 S-A에서 S-A-B로 갱신하고 경로기록과 request id를 버퍼에 저장한 후 패킷을 발송한다.

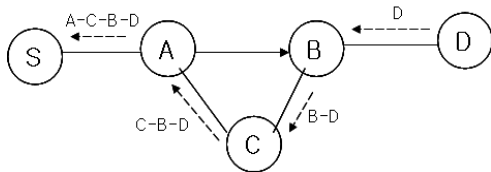
노드 C는 경로기록을 S-A에서 S-A-C로 갱신하고 경로기록과 request id를 버퍼에 저장한 후 패킷을 발송한다. 노드 B가 발송한 request는 노드 C와 D가 수신하게 되는데 C는 이미 request id가 동일한 패킷을 수신하였으며 지금 수신한 request와 이전에 A로부터 받은 request를 비교하면 지금 B로부터 받은 request의 경로거리가 크므로 B로부터 수신된 request를 무시한다. 노드 C가 발송한 request는 노드 A와 노드 B에서 수신하게 되나 앞과 같은 이유로 그 route request는 모두 무시된다. 목적지 노드 D는 노드 B로부터 route request를 받으면 route reply 패킷을 브로드캐스트한다. 라우팅 테이블을 생성하는 알고리즘을 설명하기 위하여 몇 가지 표기방법을 정하였다. 노드 A와 B사이의 경로를 $p(A, B)$, 경로거리를 $d(A, B)$ 라고 정의하자. $p(A, B, C, D)$ 는 경로 A-B-C-D의 경로이며 $d(A, B, C, D)$ 는 A-B-C-D의 경로거리이다. route request에 의하여 송신 경로를 생성하는 알고리즘은 (그림 4)와 같다.

1. route request packet을 초기화한다.
2. route request를 이웃 노드에 브로드캐스트한다.
3. $p(A, B)$ 에서 노드 B가 노드 A로부터 패킷을 받으면
 - (1) 송신경로기록에 노드 B를 추가 한다.
 - (2) 노드 B가 목적지 노드인가 판단한다.
4. 만약 3번 과정에서 두 개 이상의 노드로부터 동일한 request id를 가지는 packet을 수신하면 $d(A, B)$ 를 비교하여 경로거리가 큰 작은 패킷을 취하고 경로거리가 큰 패킷은 버린다.
5. 만약 노드 B가 목적지 노드가 아니면 목적지 노드에 도달할 때까지 과정 2번부터 반복한다.

(그림 4) route request에 의하여 송신경로를 생성하는 알고리즘

목적지 노드 D에서 출발지 노드 S로 수신경로를 찾을 때 각 노드들은 IP 주소를 갖지 않는다. 따라서 route request에 기록되어 있는 송신경로에 역으로 각 노드들을 찾아갈 수는 없으므로 노드

D 는 출발지 노드 S 로 route reply를 브로드캐스트한다. route reply에는 출발지 주소, 목적지 주소를 포함하며 동일한 경로발견 요구가 노드에 도착 되었을 때 중복작업을 피하기 위한 reply id와 request id, 그리고 경로발견 단계에서 찾아진 송신 경로 기록과 경로응답 단계에서 찾게 될 수신 경로 기록을 포함하고 있다. 수신경로를 찾는 과정은 송신경로를 찾는 과정과 유사하다. 이 과정을 (그림 5)를 이용하여 살펴보자.



(그림 5) 단 방향 링크를 갖는 네트워크에서 경로응답 과정

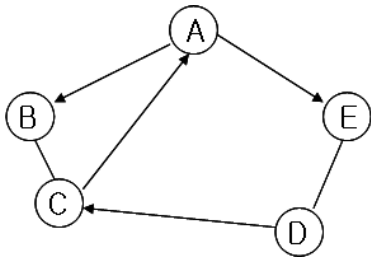
D 에서 발생된 route reply 패킷을 노드 B 가 수신한다. 노드 B 는 route reply 패킷에서 수신 경로 기록에 B 를 추가하여 경로기록을 $D-B$ 로 갱신하고 경로기록과 reply id를 버퍼에 저장한 후 reply 패킷을 발송한다. 노드 B 에서 발송된 패킷은 노드 C 에 도착한다. 노드 C 는 경로기록을 $D-B$ 에서 $D-B-C$ 로 갱신하고 경로기록과 reply id를 버퍼에 저장한 후 패킷을 발송한다. 노드 A 는 경로기록을 $D-B-C$ 에서 $D-B-C-A$ 로 갱신하고 경로기록과 reply id를 버퍼에 저장한 후 패킷을 발송한다. 노드 A 가 발송한 route reply 패킷은 노드 S 와 B 가 수신하게 되는데 B 는 request id가 동일한 패킷이 수신경로를 찾는데 B 의 버퍼에 기록된 송신경로를 참조할 때 송신경로에서 자기보다 이전 노드에서 route reply 패킷이 도착하였으므로 최단 경로에 의한 패킷전송이 아님을 알 수 있으므로 수신된 패킷을 무시하고 이웃 노드에 route reply 패킷을 발송하지 않는다. route reply에 의하여 수신경로를 생성하는 알고리즘은 (그림 6)과 같다.

1. route request packet에 수신경로 발견을 위한 정보를 포함시켜 route reply 패킷을 생성한다.
2. route reply 패킷을 이웃 노드에 브로드캐스트한다
3. $p(A, B)$ 에서 노드 B 가 노드 A 로부터 패킷을 받으면
 - (1) 수신경로기록에 노드 B 를 추가 한다.
 - (2) 노드 B 가 출발지 노드인가 판단한다.
4. 만약 3번 과정에서 두개 이상의 노드로부터 동일한 request id를 가지는 packet을 수신하면
 - (1) 경로거리 $d(A, B)$ 를 비교하여 경로거리가 큰 작은 패킷을 취하고 경로거리가 큰 패킷은 버린다.
 - (2) 송신경로 상에서 경로발견 단계에서 A 가 B 의 경로 상 이전 노드이면 그 패킷은 무시한다.
5. 노드 B 가 출발지 노드가 아니면 출발지 노드에 도달할 때까지 2번부터 반복한다.

(그림 6) route reply에 의하여 수신경로를 생성하는 알고리즘

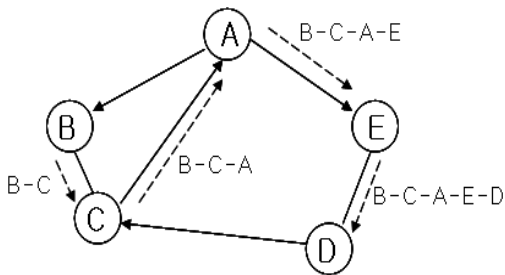
n 개의 노드로 구성된 네트워크에서 DSDV[1]와 AODV[2]와 같이 거리 벡터(distance vector)에 기반한 프로토콜들은 양방향 링크에 대한 경로를 발견하기 위한 라우팅 테이블을 생성하므로 이웃 노드들에 대해 하나의 거리벡터 정보가 필요하여 각 노드의 라우팅 테이블의 크기는 $O(n)$ 이다. 본 논문에서 제안하는 방법은 네트워크에서 단 방향 링크가 존재하는 경우도 고려하므로 송신과 수신을 위한 경로정보가 각각 필요하다. 각 노드는 이웃 노드에 대해 크기가 $O(n)$ 인 거리벡터 테이블 RDT와 RRT를 유지하여야 하므로 각 노드는[5]와 같이 크기가 $O(n)$ 인 라우팅 테이블을 유지하여야 한다. [3,4]의 방법은 각 노드가 이웃노드 뿐 아니라 네트워크 내의 모든 노드에 대한 경로정보를 가지고 있어야 하므로 테이블의 크기가 $O(n^2)$ 이다.

(그림 7)의 단 방향 링크를 갖는 네트워크를 이용하여 노드 B 와 노드 D 간에 경로를 설정하기 위하여 라우팅 테이블을 생성하는 과정을 아래에 설명하였다. 노드 B 에서 노드 D 로 패킷을 전송하기 위한 라우팅 테이블을 생성하기 위하여 B 에서 route request를 브로드 캐스트하면 A 와 B 사이의 경로



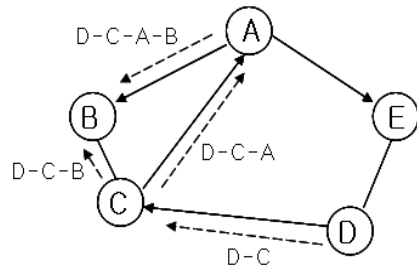
(그림 7) 단 방향 링크 네트워크 예

는 A에서 B로의 단 방향 경로이므로 노드 B에서 발송한 패킷은 C로 전달된다. 같이하여 C에서 이웃 노드들을 향하여 발송한 신호는 A와 B에 수신된다. B에 전달된 route request는 무시된다. A에서 발송한 패킷은 E, 마지막으로 E에서 발송한 패킷은 D로 전달되어 목적지 노드에 도달한다. 이러한 과정을 (그림 8)에 나타내었다.



(그림 8) 노드 B에서 노드 D까지의 경로를 찾는 과정

최종 목적지 D에서 route reply 패킷을 생성하여 이웃 노드에 브로드캐스트하면 C와 E로 전달된다. E에서 발송된 reply 패킷은 노드 D로 전달되나 D에서 무시되고 C에서 발송한 패킷은 노드 B와 A로 전달된다. 노드 A에서 발송한 reply 패킷은 노드 B에 도착하나 노드 C에서 도착한 패킷의 경로거리가 노드 A에서 도착한 패킷의 경로거리보다 작으므로 출발지 노드 B는 노드 C에서 발송된 패킷을 취한다. 이러한 과정을 (그림 9)에 나타내었다.



(그림 9) 노드 D에서 노드 B로 응답하는 경로를 찾는 과정

송신경로를 찾는 과정과 수신경로를 찾는 과정에서 각 노드의 버퍼에는 경로기록이 저장되어 있다. 노드 A에는 송신경로 B-C-A와 수신경로 D-C-A, 노드 B에는 송신경로 B와 수신경로 D-C-B, 노드 C에는 송신경로 B-C와 수신경로 D-C, 노드 D에는 송신경로 B-C-A-E-D와 수신경로 D, 노드 E에는 송신경로 B-C-A-E가 저장되어 있다. 노드 E는 수신경로 발견 시 노드를 거치지 않았으므로 수신경로는 비어있다. 각 노드에서는 버퍼에 저장되어 있는 경로정보를 이용하여 RDT와 RRT 라우팅 테이블을 생성한다. RDT의 형식은 {DN, NN, m}이다. 여기서 DN은 목적지 노드(destination node), NN은 이웃노드(neighbor node), m은 경로거리(metric)이다. RRT의 형식은 {SN, NN, m}이다. 여기서 SN은 출발지 노드(source node)이다. <표 1>에 (그림 7)를 이용하여 노드 B와 노드 D간의 경로설정 후 생성한 라우팅 테이블을 보였다. <표 1>에서는 아직 모든 노드들에 대한 경로가 설정되지 않았으므로 라우팅 테이블이 완성되지 않았다.

Table driven 방식에서는 모든 노드에서 route request 패킷을 발생하여 라우팅 테이블을 완성하고, on-demand driven 방식에서는 패킷전송의 요구가 있을 때 경로에 대한 정보를 추가하면 라우팅 테이블을 완성할 수 있다. <표 2>에 (그림 7)의 네트워크에서 노드 A의 라우팅 테이블 RDT와 RRT를 나타내었다. RDT에서 {C, B, 2}는 A에서

C까지의 최단경로는 경로거리가 2이며 그 경로 상에서 메시지를 전달할 다음 노드는 B라는 의미이다. RRT에서 {E, D, 3}은 노드 E에서 노드 A까지의 최단경로는 거리가 3이며 그 경로 상에서 E의 다음 노드는 D라는 의미이다. <표 3>은 노드 B, <표 4>는 노드 C, <표 5>는 노드 D, 그리고 <표 6>은 노드 E의 라우팅 테이블을 각각 보였다.

<표 1> (그림 7)을 이용하여 노드 B와 노드 D 간의 경로설정 후 생성한 노드 B의 라우팅 테이블

RDT			RRT		
DN	NN	m	SN	NN	m
A	C	2	A	-	-
B	B	0	B	B	0
C	C	1	C	-	-
D	C	4	D	C	2
E	C	3	E	-	-

<표 2> 노드 A의 라우팅 테이블

RDT			RRT		
DN	NN	m	SN	NN	m
A	C	2	A	-	-
B	B	0	B	B	0
C	C	1	C	-	-
D	C	4	D	C	2
E	C	3	E	-	-

<표 3> 노드 B의 라우팅 테이블

RDT			RRT		
DN	NN	m	SN	NN	m
A	C	2	A	-	-
B	B	0	B	B	0
C	C	1	C	-	-
D	C	4	D	C	2
E	C	3	E	-	-

<표 4> 노드 C의 라우팅 테이블

RDT			RRT		
DN	NN	m	SN	NN	m
A	C	2	A	-	-
B	B	0	B	B	0
C	C	1	C	-	-
D	C	4	D	C	2
E	C	3	E	-	-

<표 5> 노드 D의 라우팅 테이블

RDT			RRT		
DN	NN	m	SN	NN	m
A	C	2	A	-	-
B	B	0	B	B	0
C	C	1	C	-	-
D	C	4	D	C	2
E	C	3	E	-	-

<표 6> 노드 E의 라우팅 테이블

RDT			RRT		
DN	NN	m	SN	NN	m
A	C	2	A	-	-
B	B	0	B	B	0
C	C	1	C	-	-
D	C	4	D	C	2
E	C	3	E	-	-

4. 결 론

본 논문에서는 다중 홉 이동 ad-hoc 네트워크를 위한 간단하며 효율적인 라우팅 프로토콜을 제안하였다. 제안한 프로토콜은 단 방향 링크를 갖는 네트워크에서도 라우팅을 가능하게 한다. 제안된 프로토콜은 패킷의 송신과 수신을 위하여 두개의 라우팅 테이블로 구성되어 있다. 두 개의 라우팅 테이블을 유지함으로써 양방향 링크 만 포함하는

네트워크뿐만 아니라 단 방향 링크를 포함하는 네트워크에서도 효율적으로 패킷의 경로를 설정할 수 있다. 각 노드의 라우팅 테이블의 크기는 $O(n)$ 이다.

참 고 문 헌

[1] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", *Comp. Commun. Rev.*, pp. 234-244, Oct. 1994.

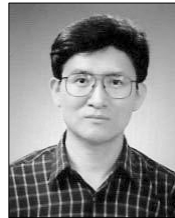
[2] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing", *Proc. 2nd IEEE Workshop on Mobile Comp. Sys. and Apps.*, pp. 90-100, Feb. 1999.

[3] R. Prakash and M. Singhal, "Impact of Unidirectional Links in Wireless ad-hoc Networks", *Proc. DIMACS Workshop on Mobile*

Networks and Computing, Rutgers University, NJ, March 1999.

[4] R. Prakash, "A Routing Algorithm for Wireless Ad Hoc Networks with Unidirectional Links", *Wireless Networks*, Vol. 7, No. 6, pp. 617-625, 2001.

[5] F. C. M. Lau, G. Chen, H. Huang, and L. Xie, "A Distance-vector Routing Protocol for Networks with Unidirectional Links", *Comp. Commun.*, Vol. 23, pp. 418-424, 2000.



박 상 호

1979년 경북대학교 전자공학
(공학사)

1981년 영남대학교 전자공학과
(공학석사)

1989년 Syracuse University
ECE(MS)

1995년 State University of New York at Buffalo,
ECE(Ph.D.)

1996년~현재 안동대학교 전자정보산업학부 부교수