

무선 Ad-hoc 네트워크 환경을 위한 우선순위 기반 소스 라우팅 프로토콜 설계

전대홍[○], 김인숙, 김문정, 염영익
 성균관대학교 정보통신공학부
 {babocdh, easy, tops, yieom}@ece.skku.ac.kr

The Design of Priority-Based Source Routing Protocol for Wireless Ad-hoc Networks

Dae Hong Chun[○], In-suk Kim, Moon Jeong Kim, Young Ik Eom
 School of Information and Communication Engineering, Sungkyunkwan University

요 약

최근 무선 환경에서 기지국 없이 이동 노드들만으로도 서로 통신할 수 있는 ad-hoc 네트워크에 대한 연구가 진행되고 있다. 이러한 무선 ad-hoc 네트워크 환경에서는 라우팅 프로토콜이 중요한 역할을 하게 되는데 그 중에서 DSR(Dynamic Source Routing)은 무선 ad-hoc 네트워크 환경에 적합한 소스 라우팅을 기반으로 한 라우팅 프로토콜이다. 그러나 DSR은 데이터 전송 경로를 선택할 때 그 경로의 신뢰성을 고려하지 않는다. 이를 해결하기 위해 본 논문에서는 무선 ad-hoc 네트워크 환경에서 노드의 우선순위에 기반한 신뢰성 있는 경로 설정 기법을 제안한다. 이 기법은 경로 설정 후에 데이터를 송수신하는 과정에서, 선택된 경로의 링크가 유실될 확률을 줄여줌으로써 더욱 신뢰성 있는 데이터 송수신을 보장한다. 그러므로 이 기법은 무선 ad-hoc 네트워크 환경에서 데이터의 손실을 줄일 필요가 있는 멀티미디어 통신 등의 응용분야에 적용시킬 수 있다.

1. 서론

최초에 군사적 목적으로 연구되었던 무선 ad-hoc 네트워크 환경은 최근 재해 지역에서의 통신, 응급 상황에서의 긴급구조 등의 활용을 위해 연구가 확산되고 있다. 무선 ad-hoc 네트워크 환경이란 특정 기지국을 이용하지 않고, 무선 이동 단말들로 구성된 네트워크를 말한다. 이러한 무선 ad-hoc 네트워크 환경에서는 이동 노드들끼리 경로를 탐색하고 유지시키는 라우팅 프로토콜이 중요하다[1].

현재까지 무선 ad-hoc 네트워크 환경을 위한 라우팅 프로토콜이 많이 연구되고 있다. 그 중 DSR은 무선 ad-hoc 네트워크 환경에 적합한 소스 라우팅을 기반으로 한 라우팅 프로토콜이다. 그러나 DSR은 데이터 전송 경로를 선택할 때 경로의 신뢰성을 보장하지 못한다[2].

본 논문에서는 노드들의 우선순위를 고려한 신뢰성 있는 경로 탐색 기법을 제안하고자 한다. 소스 노드는 경로를 선택할 때 이동이 적고 전원이 차단될 확률이 적은 신뢰성 있는 노드들을 이용하여 데이터 전송경로를 구성한다. 따라서 제안 기법은 데이터 전송 경로의 링크가 유실될 확률을 줄이고, 데이터 전송속도를 향상시키며, 추가적인 경로 요청을 줄인다.

본 논문의 2장에서는 기존의 라우팅 프로토콜의 종류와 특징에 대해서 알아보고, 3장에서는 제안 기법을 소개한다. 4장에서는 시뮬레이션을 통해 제안 기법을 평가하고, 5장에서는 결론 및 향후 과제를 알아본다.

2. 관련 연구

본 장에서는 무선 ad-hoc 네트워크 환경에 적합한 라우팅 프로토콜들에 대해서 알아보고, 그 중 본 논문과 관련이 있는 DSR에 대해 알아본다.

2.1 무선 ad-hoc 환경에서의 라우팅 프로토콜

무선 ad-hoc 네트워크 환경에서의 라우팅 프로토콜은 table driven 방식과 on-demand 방식으로 나뉜다. Table driven 방식은 네트워크 내에서 지속적으로 라우팅 정보를 관리함으로써 경로 요청이 있을 때 즉시 응답을 할 수 있으나 최신정보를 유지하기 위해 네트워크의 대역폭을 소모하는 단점을 가지며 OLSR, TBRP 등이 있다[3,4]. On-demand 방식은 경로 요청이 있을 경우에만 경로 설정을 통해 경로를 얻는 방식으로 라우팅 정보를 얻기 위한 시간 지연 때문에 실시간 통신에는 부적절하지만 네트워크 대역폭 소모가 적은 기법으로 AODV, DSR 등이 해당된다[5,6].

2.2 DSR(Dynamic Source Routing) 프로토콜

DSR은 On-demand 방식의 소스 라우팅을 기반으로 한 라우팅 프로토콜로 경로 설정과 경로 유지 단계로 구성된다[7]. 경로 설정 단계에서 소스 노드는 목적 노드까지의 경로를 얻기 위해 경로 요청 패킷을 브로드캐스팅하고, 중간 노드는 경로 요청 패킷 안에 자신의 주소를 기록한 후 다시 브로드캐스팅한다. 경로 요청 패킷을 전달받은 목적 노드는 경로 요청 패킷 안에서 노드들의 리스트를 추출하고, 그것을 다시 경로

응답 패킷에 담아 소스 노드에게 전송하게 된다.

경로 유지 단계에서 각 노드들은 이웃 노드의 상태를 확인하기 위한 주기적인 메시지를 사용하지 않는다. 대신 이웃 노드로의 데이터 전송이 실패할 경우, 데이터를 전송한 노드는 유실된 링크의 정보를 경로 오류 패킷에 실어서 소스 노드에 전송함으로써 경로 유지를 한다.

DSR은 경로 유지를 위해 적은 대역폭을 사용하는 효율적인 라우팅 프로토콜이다. 그러나 무선 ad-hoc 네트워크 환경에서 노드들의 이동이 활발할 경우, DSR은 경로의 신뢰성을 보장하지 못하는 문제점이 있다.

3. 제안 기법

3.1 개요

본 논문에서는 무선 네트워크 상에 이동이 적은 노드들이 존재하고 노드들 간에는 양방향 링크로 연결되어 있으며 감청 모드(promiscuous mode)를 사용한다고 가정한다. 이러한 노드들로 구성된 경로를 통해 데이터를 전송하면 전송 도중에 링크가 유실될 확률이 낮다. 따라서 본 논문에서는 오랜 시간 동안 경로를 유지시켜줄 수 있는 신뢰성 있는 노드들을 이용한 소스 라우팅 기법을 제안한다.

네트워크 상의 각 노드들은 이웃 노드의 우선순위에 대한 테이블과 경로를 저장하는 라우트 캐쉬 테이블을 유지한다. 또한 데이터가 전송될 때 전송 경로 상에 있는 각 노드들은 자신의 이웃 테이블에서 경로에 포함된 이웃 노드에 대한 우선순위를 증가시킨다.

소스 노드는 경로 요청을 한 후 목적 노드로부터 다수개의 경로 응답 패킷을 수신한다. 그리고 그 패킷들로부터 얻어진 경로들을 자신의 라우트 캐쉬에 저장한 후 최적 경로를 선택한다.

소스 노드로부터 경로 요청 패킷이 목적 노드로 전달되는 과정에서 중간 노드가 목적 노드까지의 경로를 자신의 라우트 캐쉬에 저장하고 있는 경우에는 중간 노드 역시 최적 경로를 선택하여 경로 응답 패킷을 소스 노드로 전송할 수 있다.

3.2 패킷 형식 및 자료구조

제안 기법에서 사용되는 패킷들의 형식은 그림 1에서 보인다.

Type	PacketID	DestAddr	NumAddr	AddrList
(a) 경로 요청 패킷 (RREQ : Route REQuest packet)				
Type	NumAddr	AddrList		
(b) 경로 응답 패킷 (RREP : Route REPLY packet)				
Type	NumAddr	AddrList	Data	
(c) 데이터 패킷 (DATP : DATa Packet)				
Type	ErrType	ErrSrcAddr	ErrDestAddr	
(d) 경로 오류 패킷 (RERR : Route ERRor packet)				

(그림 1) 패킷의 형식

그림 1-(a)에서 RREQ 안에는 중복 처리 방지를 위해 PacketID 필드를 두고, 목적 노드의 주소를 저장하기 위한 DestAddr 필드를 둔다. 그림 1-(a),(b),(c)의 RREQ, RREP, DATP에는 경로를 저장하기 위해 경로상의 노드의 개수인 NumAddr 필드와 그 노드의 리스트인 AddrList 필드를 둔다. 그림 1-(d)에서 RERR 안에는 유실된 링크의 양구간을 가리키는 ErrSrcAddr, ErrDestAddr 두 개의 필드를 둔다.

네트워크상의 각 노드들이 유지해야 할 이웃 테이블과 라우트 캐쉬 테이블의 구성은 그림 2와 같다.

NA (Neighbor Address)		PRI (Priority)	
(a) 이웃 테이블 (NT : Neighbor Table)			
SrcAddr	DestAddr	NumOfHops	SeqOfNode
ExpireTime			
(b) 라우트 캐쉬 테이블 (RCT : Route Cache Table)			

(그림 2) 이웃 테이블과 라우트 캐쉬 테이블의 구성

그림 2-(a)에서 NT에는 이웃 노드의 주소와 우선순위가 기록되며, 그림 2-(b)에서 RCT의 NumOfHops 필드에는 소스 노드에서 목적 노드까지의 경로의 홉수가 저장되고, SeqOfNode 필드에 경로상의 노드들의 순서가 저장된다. 각각의 엔트리에는 ExpireTime이 설정되어서 일정 시간동안 엔트리가 사용되지 않을 경우 자동으로 삭제된다.

3.3 알고리즘

본 절에서는 소스 노드가 다중 경로 중에 최적의 경로를 선택하는 작업과 경로 요청 패킷을 전달하는 중간 노드가 최적 경로를 선택해서 소스 노드에게 전송하는 작업, 그리고 네트워크 상의 각 노드들이 이웃 테이블을 유지하는 작업에 대해 설명한다.

(1) 소스 노드의 경로 선택 작업

소스 노드는 경로 요청 후 일정시간동안 다수개의 RREP를 수신할 수 있다. 이들 다중 경로들 중에서 최적 경로를 선택하는 동작과정을 알고리즘 1에서 보인다.

```
// Tout : timeout for waiting RREPs
// ROUTE : the route extracted from RREP
// HOP : the number of hops in ROUTE
// Mroute : the route which has minimal number of hops
// Mhop : the minimal number of hops
// Fnode : the node following source node in ROUTE
// Mpri : minimal priority of Fnodes
send RREQ:
while (during Tout) {
  if (received its RREP) {
    insert ROUTE into RCT:
    if (Mhop > HOP) {
      Mhop ← HOP:
      Mroute ← ROUTE:
    } else if (Mhop equals HOP) {
      if (Mpri > Fnode's priority) {
        Mpri ← Fnode's priority:
        Mroute ← ROUTE:
      }
    }
  }
}
select Mroute:
```

(알고리즘 1) 소스 노드의 경로선택

알고리즘 1에서 최적 경로 선택을 위해 소스 노드는 홉수가 가장 적은 경로를 최우선적으로 선택하고, 홉수가 가장 적은 경로의 수가 2개 이상일 경우에는 경로에 포함된 이웃 노드들 중에서 우선순위가 가장 높은 이웃 노드가 포함된 경로를 선택한다.

(2) 중간 노드의 경로 선택 및 경로 응답

중간 노드가 소스 노드의 경로 요청에 대해 경로 응답을 하는 동작과정을 알고리즘 2에서 보인다.

```
// ROUTE : the route from this node to DestAddr in RREQ
// HOP : the number of hops in ROUTE
// Mroute : the route which has minimal number of hops
// Mhop : the minimal number of hops
// Fnode : the node following current node in ROUTE
// Mpri : minimal priority of Fnodes
if (received RREQ and exists ROUTE in RCT) {
  extract AddrList from RREQ and discard RREQ:
  for (each ROUTE in RCT) {
    if (Mhop > HOP) {
      Mhop ← HOP:
      Mroute ← ROUTE:
    } else if (Mhop equals HOP) {
      if (Mpri > Fnode's priority) {
        Mpri ← Fnode's priority:
        Mroute ← ROUTE:
      }
    }
  }
  create RREP with (AddrList+Mroute) and send it to source node:
} else
  rebroadcast RREQ:
```

(알고리즘 2) 중간 노드에서의 경로 응답

알고리즘 2에서 RREQ를 수신한 중간 노드가 목적 노드까지의 경로를 자신의 라우트 캐쉬에 저장하고 있을 경우에는 자신의 노드에서 목적 노드까지의 최적 경로를 선택한 후 목적 노드를 대신하여 소스 노드에게 RREP를 전송한다. 이 때 최적 경로를 선택하는 방법은 알고리즘 1과 동일하다.

(3) 이웃 테이블 관리

네트워크상의 모든 노드들은 이웃 테이블을 유지해야 한다. 이웃 테이블에 엔트리를 등록, 수정, 삭제하는 과정을 알고리즘 3에서 보인다.

```
// Type : Type of Packet
// Sendnode : the Node which send RREQ to this node
// Errnode : ErrDestAddr field of RERR
switch (Type) {
  case RREQ : // insert entry
    insert address of Sendnode into NA field of NT:
    initialize NODE's priority to 1:
    break:
  case DATP : // modify entry
    for (each NODE in AddrList of DATP)
      if (exists NODE in NT)
        increase NODE's priority by 1:
    break:
  case RERR : // delete entry
    if (exists Errnode in NT)
      delete entry of Errnode from NT:
}
```

(알고리즘 3) 이웃 테이블에 엔트리 등록, 수정, 삭제

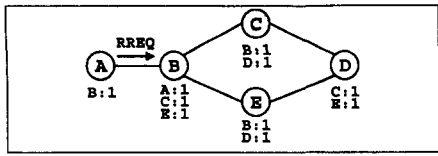
알고리즘 3에서 RREQ를 수신한 노드는 자신에게 RREQ를 전송시킨 노드를 자신의 이웃 테이블에 등록시키고 우선순위를 1로 초기화한다. DATP를 수신한 노드들은 DATP의 AddrList 필드 안의 노드들을 자신의 이웃 테이블에서 검색하여 등록되어 있는 노드들에 대해 우선순위를 1씩 증가시킨다. RERR를 전달받은 노드들은 RERR의 ErrDestAddr 필드를 확인하여 해당 노드를 자신의 이웃 테이블에서 삭제한다.

3.4 시나리오

본 절에서는 이웃 테이블을 초기화 및 수정하는 경우와 소스 노드 및 중간 노드가 최적 경로를 선택하는 시나리오를 보인다. 각 시나리오에서 노드의 아래에 그 노드의 이웃 테이블의 내용을 표시하였고, 이웃 테이블에서 영문자는 이웃 노드의 주소를 나타내며, 숫자는 해당 노드에 대한 우선순위를 나타낸다.

(1) 이웃 테이블을 초기화하는 경우

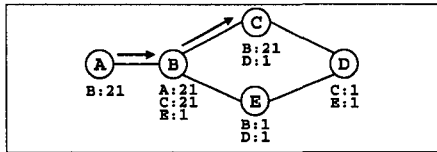
소스 노드가 경로를 요청하는 과정에서 네트워크상의 모든 노드들은 자신의 이웃 테이블에 이웃 노드들을 등록하고 그 이웃 노드의 우선순위를 초기화시킨다.



(그림 3) 노드 A에서 노드 C로의 경로 요청

그림 3에서 소스 노드 A가 목적 노드 C로의 경로 요청을 위해 RREQ를 브로드캐스팅 한 후의 각 노드의 이웃테이블의 상태를 보인다.

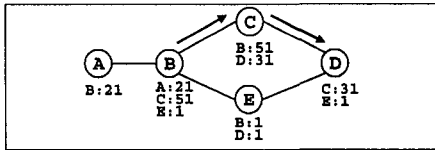
(2) 데이터 전송 시 이웃 테이블이 수정되는 경우
그림 3의 상태에서 소스 노드 A에서 목적 노드 C로 20패킷의 데이터를 전송한 후의 상태를 그림 4에서 보인다.



(그림 4) 노드 A에서 노드 C로의 데이터 전송

그림 4에서 노드 B는 자신의 이웃테이블의 정보를 참조하여 노드 C의 우선순위가 노드 E의 우선순위보다 높으므로, 노드 C로의 링크가 보다 더 신뢰성이 있다고 판단하게 된다.

(3) 소스 노드가 최적 경로를 선택하는 경우
그림 4의 상태에서 소스 노드 B는 목적 노드 D로의 경로를 요청하고 받은 경로 응답 중에서 자신의 이웃 테이블 정보를 참조하여 신뢰성이 높다고 판단되는 노드 C를 포함하는 경로를 최적의 경로로 선택하게 된다. 소스 노드 B가 목적 노드 D로 데이터를 전송시킨 후의 상태를 그림 5에서 보인다.



(그림 5) 노드 B에서 노드 D로의 데이터 전송

그림 5는 소스 노드 B에서 목적 노드 D까지 30개의 패킷 데이터를 전송한 후의 상태를 보인다.

(4) 중간 노드가 최적 경로를 선택하는 경우
그림 5의 상태에서 노드 A가 노드 D로 데이터를 전송시키기 위하여 경로를 요청하는 경우, RREQ를 수신한 노드 B는 목적 노드 D까지의 경로를 자신의 라우팅 캐쉬에 유지하고 있으므로 목적 노드 대신 경로 응답을 할 수 있다. 이 때 노드 B는 자신의 이웃 테이블을 참조하여 신뢰성이 높은 노드 C를 포함하는 경로를 최적 경로로 선택해서 소스 노드 A에게 경로 응답 패킷을 보낸다.

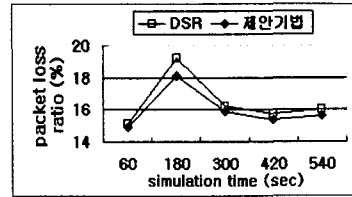
4. 성능평가

(1) 시뮬레이션 환경

제안 기법의 효율성을 측정하기 위하여 NS2-2.26을 이용한 시뮬레이션을 실시하였다[8]. 시뮬레이션 영역의 크기는 가로, 세로 각각 1000m이고, 총 노드 수는 20개로 하였다. DSR과의 성능을 비교하기 위하여 평균데이터 전송시간과 노드들의 패킷 손실율을 측정하였으며, 시뮬레이션 시간은 1분, 3분, 5분, 7분, 9분의 다섯 가지의 경우로 각 경우 당 시드를 달리하여 5회 실시한 후 평균을 내었다.

(2) 시뮬레이션 결과

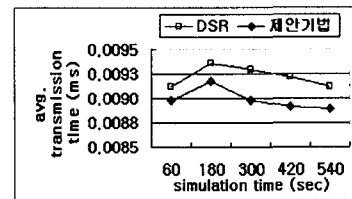
DSR과 제안 기법 간의 패킷 손실율의 비교를 그림 6에서 보인다.



(그림 6) 패킷 손실율

그림 6에서는 제안된 기법이 DSR에 비해 패킷 손실율이 감소되었음을 보인다. 이는 제안 기법이 신뢰성 있는 노드를 이용하여서 데이터 전송경로를 구성함으로써 링크가 유실될 확률을 줄였기 때문이다.

DSR과 제안 기법 간의 평균 데이터 전송시간의 비교를 그림 7에서 보인다.



(그림 7) 평균 데이터 전송시간

그림 7에서는 제안된 기법이 DSR보다 평균 데이터 전송시간이 짧아졌음을 보인다. 제안 기법은 DSR에 비해 데이터 전송 경로상의 링크가 유실될 확률이 적으므로 원활하게 데이터를 전송하고, 추가적인 경로 요청을 줄인다.

제안 기법은 이동성이 적고 신뢰성 있는 노드를 데이터 전송 경로 안에 포함시킴으로써 DSR에 비해 평균 데이터 전송시간과 패킷 손실율을 감소시킨다.

5. 결론

본 논문에서는 무선 ad-hoc 네트워크 환경에서 신뢰성 있는 우선순위 기반 소스 라우팅 기법을 제안하였다.

노드의 우선순위에 기반한 라우팅 기법은 패킷 손실율을 줄여줌으로써 데이터의 송수신이 빠르고 원활하게 수행되도록 한다. 또한 제안 기법은 추가적인 경로 요청을 줄여서, 네트워크의 대역폭 소모를 줄여준다.

그러나 제안 기법은 우선순위가 높은 일부 노드에서 많은 전력소모와 병목현상이 발생할 가능성이 있다. 향후 이를 보완하기 위한 기법들을 연구해야 할 것이다.

참고 문헌

- [1] S. Corson and J. Macker, "Mobile Ad hoc Network (MANET) : Routing Protocol Performance Issues and Evaluation Considerations," Internet RFC 2501, Jan. 1999.
- [2] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," Mobile Computing, chapter 5, pp. 153-181, Kluwer Academic Publishers, 1996.
- [2] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," Internet Draft, Jan. 2003.
- [4] R. Ogier, F. Templin and M. Lewis, "Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)," Internet Draft, Jul. 2003.
- [5] C. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," Internet RFC 3561, Jul. 2003.
- [6] D. B. Johnson, D. A. Maltz and Y. C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," Internet Draft, Apr. 2003.
- [7] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless AdHoc Networks," in AdHoc Networking, edited by Charles E. Perkins, Chapter5, pp.139-172, Addison-Wesley, 2001.
- [8] <http://www.isi.edu/nsnam/ns/tutorial>, "Tutorial for the Network Simulator ns"