

무선 ad-hoc 네트워크 환경을 위한 MHCRP 라우팅 프로토콜

전형국*, 엄영익**

*성균관대학교 전기 전자 및 컴퓨터 공학부
e-mail:{cell24, yieom}@ece.skku.ac.kr

MHCRP Protocol in Wireless Ad-hoc Networks

Hyung Kook Jun*, Young Ik Eom**

*School of Electrical and Computer Engineering,
Sungkyunkwan University

요약

무선 ad-hoc 네트워크는 중앙의 특별한 관리 체계 없이 유선망과 기지국을 가지지 않는 이동 호스트로 구성된 네트워크를 말한다. 이러한 ad-hoc 네트워크는 잦은 망 구성의 변화, 라우터의 수, 사용 자원 등 기존 유선 네트워크와는 다른 특징들을 가지게 된다. 따라서 기존의 유선망에서 사용하던 라우팅 프로토콜을 ad-hoc 네트워크에 적용시키는 데에는 많은 문제점들이 발생한다. 이 논문에서는 ad-hoc 네트워크의 환경을 고려하여 적은 지연시간과 오버헤드를 발생시키는 MHCRP 라우팅 프로토콜을 제안하고자 한다. 제안하는 모델은 ad-hoc 네트워크 내의 이동 호스트들을 멀티 홉을 갖는 클러스터로 묶어, 클러스터 헤드로 하여금 자신의 멤버 호스트와 이웃 클러스터의 헤드의 정보를 유지하게 하여, 경로 설정에 대한 요구가 있을 때 적은 지연 시간과 적은 패킷으로 목적지까지의 최단 경로를 설정할 수 있는 계층적인 경로 설정 방법을 제시한다.

1. 서론

ad-hoc 네트워크는 중앙의 특별한 관리 체계 없이 유선망과 기지국을 가지지 않는 이동 호스트의 집합을 말한다. 따라서 ad-hoc 네트워크는 기존의 유선망의 도움 없이 필요시에 빠르게 설치되어야 하는 네트워크의 설정에 용이하다. 이러한 ad-hoc 네트워크는 기존의 유선망과 비교되는 많은 특징들을 가지고 있다. 첫째, ad-hoc 네트워크 내의 모든 이동 호스트들은 라우터의 역할을 수행한다. 둘째, 이동 호스트의 이동으로 네트워크의 토폴로지가 동적으로 변한다. 셋째, 무선 매체들이 간섭 현상을 일으키고 작은 무선 대역폭과 도달할 수 있는 거리의 한계, 잦은 끊김이 발생한다. 넷째, 기존의 유선 네트워크 보다 각 호스트들의 메모리, 배터리, 시스템 구성 요소 등이 부족하다.

위와 같은 특징들은 ad-hoc 네트워크 라우팅 프로토콜을 설계하는데 있어서 고려되어야 할 중요한 요소이다. 따라서 기존 유선 네트워크의 라우팅 프로토콜을 ad-hoc 네트워크에 적용시킬 경우에는 많은 문제점들이 발생하게 된다.

이 논문에서는 위에서 열거한 ad-hoc 네트워크의 특징들을 고려하여, ad-hoc 네트워크의 라우팅 프로토콜을 설계하는데 있어서 동적으로 변화하는 네트워크 환경에

적응하면서 적은 지연 시간과 적은 오버헤드를 가지고 ad-hoc 네트워크 내의 경로 정보를 얻을 수 있는 방법을 제시하고자 한다.

2. 관련연구

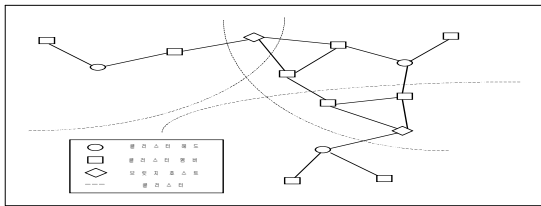
기존의 ad-hoc 네트워크의 라우팅 프로토콜은 크게 순항적(proactive) 설정 방식과 반응적(reactive) 설정 방식 2가지로 분류할 수 있다. 순항적 설정 방식은 각 이동 호스트가 네트워크 내의 경로들에 대한 정보를 항상 유지하고 있어서, 목적지에 대한 경로 정보가 요구되었을 때 즉시 경로 정보를 사용할 수가 있다. 그러나 경로에 대한 최신 정보를 유지하기 위해 많은 네트워크 트래픽을 유발하고 경로 정보의 방송 주기가 길어지면 대부분의 경로 정보는 쓸모 없게 되는 문제점이 있다. 대표적인 프로토콜로는 DSDV, OLSRP, WRP 등이 있다. 반면에 반응적 설정 방식은 특정 목적지 경로에 대한 경로 요구가 있을 때 경로 설정 절차를 수행한다. 따라서 각 이동 호스트들이 모든 라우터의 경로 정보를 유지하지 않기 때문에 상대적으로 적은 메모리 공간을 필요로 한다. 하지만 경로 설정 과정에서 발생하는 지연 시간을 가지게 되며, 또한 경로 발견을 위한 네트워크 검색으로 인해서 상대적으로 많은 네트

워크 트래픽이 발생하게 된다. 대표적인 프로토콜로는 DSR, AODV, ZRP, TORA, RDMAR, STAR등이 있다.

3. MHCRP

3.1 개요

이 논문에서 제안하는 시스템 모델은 그림 2와 같으며 모델에서 사용되는 노드들에 대한 설명은 다음 절에서 계속된다. 모델에서 클러스터의 크기를 2홉으로 가정하였으며 클러스터의 크기는 나중에 설명될 알고리즘에 의해서 동적으로 변경될 수 있다.

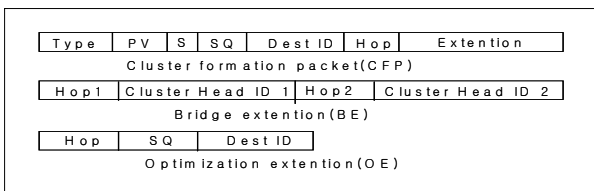


(그림 1) 멀티 홉 클러스터 기반의 ad-hoc 네트워크

그림 2에서 ad-hoc 네트워크 내의 각 이동 호스트들 지역에 기반하여 헤드와 멤버를 갖는 클러스터로 형성한다. 각 지역별로 클러스터가 형성되면 각 클러스터 헤드는 클러스터 유지에 필요한 멤버들에 대한 정보와 이웃 클러스터에 대한 정보를 가지게된다. 이러한 클러스터의 형성은 ad-hoc 네트워크에 계층적인 구조를 형성시켜 네트워크 관리와 경로 정보에 대한 효율성을 제시해줄 수 있다. 이 논문에서는 클러스터를 구성하는데 있어서 고정된 크기의 클러스터를 구성하는 대신에 네트워크의 변화에 적응할 수 있고 동적으로 변하는 멀티 홉 클러스터기반의 라우팅 프로토콜을 제시한다.

3.2 패킷 포맷

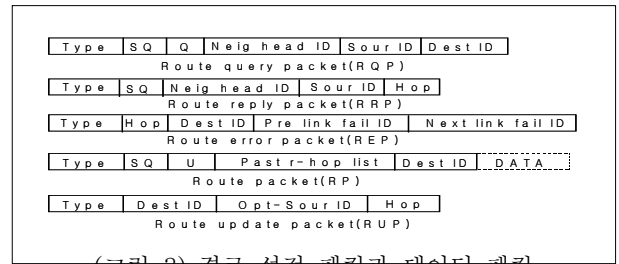
각 이동 호스트는 클러스터를 구성하기 위하여 CFP(cluster formation packet)를 사용한다. 클러스터 구성이 완료되면 클러스터와 클러스터를 연결해주는 bridge 노드를 선출해야 하는데 CFP의 BE(bridge extension)이 이 역할을 수행한다. 경로 설정 도중 경로의 최적화를 하기 위해서는 OE(optimization extension)의 필드가 이용된다. 다음 그림 2은 CFP와 BE, OE의 패킷 형식이다.



(그림 2) CFP 패킷의 확장 필드

클러스터의 구성이 완료되면 목적지 경로를 알고 싶은 송신 노드는 경로 설정 작업을 수행해야 한다. 그림 3의 5가지 패킷은 경로 설정 작업을 수행하기 위해 ad-hoc 네트

워크 내의 이동 호스트들이 사용하게 될 패킷 형식이다.



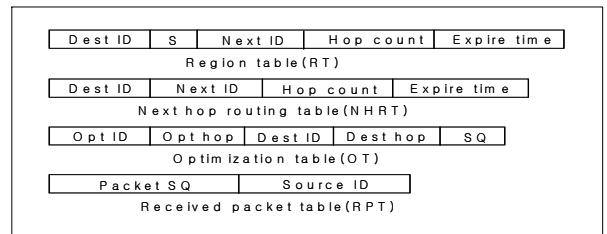
(그림 3) 경로 설정 패킷과 데이터 패킷

RQP와 RRP는 경로 설정을 위해서 사용되는 패킷으로, RQP는 경로 요구 노드에서 RRP는 중간 노드나 목적지 노드에 의해서 사용된다. REP는 경로가 실패했을 경우 경로 실패를 발견한 노드에 의해서 사용되며, RP는 실제 데이터가 전송되는 패킷의 형식, RUP는 경로 최적화 과정이나 경로 복구를 위해서 사용되는 패킷이다.

3.3 자료구조

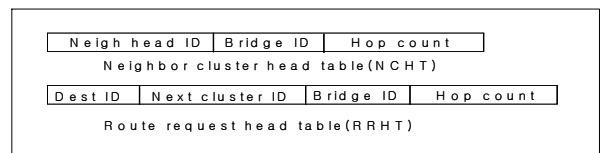
각 이동 호스트는 그림 4의 테이블을 유지한다.

RT는 CFP 패킷을 통해서 획득한 지역 호스트에 대한 라우팅 정보를 저장하는 테이블이다. NHRT는 경로 설정을 통해서 얻어진 네트워크 내의 라우팅 정보를 나타내며, OT는 최적화 과정을 위해 OE 패킷의 정보를 저장한다. RPT는 중복 수신된 패킷을 제거하기 위해서 사용된다.



(그림 4) 각 이동 호스트의 유지 테이블

클러스터 헤드는 추가로 그림 5의 테이블을 유지한다. NCHT는 이웃 클러스터의 헤드 정보를 유지하는 테이블이고 RRHT는 경로 설정 과정에서 응답 패킷이 송신지 노드로 전송되는데 사용된다.



(그림 5) 클러스터 헤드의 추가 유지 테이블

3.4 클러스터 구성

3.4.1 클러스터의 형성

처음 ad-hoc 네트워크가 구성될 때 각 호스트는 자신의 이동성을 바탕으로 우선순위를 결정한다. 우선순위가

결정되면 각 이동 호스트는 클러스터를 구성하기 위하여 CFP 패킷에 자신의 우선순위를 포함하여 이웃 호스트에게 방송한다. CFP패킷의 Type 필드 값은 1이며 PV 필드는 각 호스트의 우선순위 값을 나타낸다. S 필드는 각 호스트의 상태를 나타내며 4가지 상태 값을 갖는다. 클러스터의 멤버나 헤드가 아닌 노드는 0, 클러스터 헤드인 경우는 1, 클러스터의 멤버인 경우는 2, 브릿지 노드인 경우는 3의 값을 갖는데, 초기의 각 노드는 0의 값을 갖는다. SQ는 일련번호(sequence number)로 중복 수신된 패킷을 제거하기 위해서 사용된다. Dest ID는 CFP 패킷을 만든 노드의 주소를 나타낸다. Hop 필드는 ad-hoc 네트워크 내에 형성되는 클러스터의 크기를 결정해 주며, 각 노드가 Hop에 표시된 거리만큼의 지역적인 토폴로지 정보를 얻을 수 있게 해준다. 알고리즘 1은 클러스터 형성과 함께 각 이동 호스트가 지역적인 토폴로지 정보를 얻기 위한 방법을 보여준다.

```

node A broadcasts CFP packet;
while (CF time) {
  if (node A receives other host's CFP) {
    if (packet's hop < 0 || previous received packet)
      discard CFP packet;
    else {
      if (CFP packet's Dest ID exists in RT) {
        if (packet's hop < hop in RT) {
          update hop for Dest ID;
          rebroadcast CFP packet;
        }
        else {
          rebroadcast CFP packet;
        }
      }
      else {
        record CFP packet's route information in NHRT;
      }
      if (node A priority > CFP packet's PV) {
        node A becomes cluster head;
      }
      else {
        node A becomes cluster member;
      }
    }
  }
  if (node A don't receives other host's CFP) {
    node A becomes cluster head;
  }
}

```

(알고리즘 1) 클러스터의 형성 알고리즘

이 논문에서 제안되는 클러스터의 크기는 2 홉의 크기로 가정한다. 클러스터가 형성되면 클러스터 헤드는 CFP 패킷의 S 필드의 값을 1로 설정하고 멤버인 경우는 2로 설정하여 주기적으로 CFP 패킷을 방송하게 된다.

3.4.2 클러스터의 유지

형성된 클러스터는 시간이 지남에 따라 변화하게 된다. 특히 새로운 클러스터의 생성과 삭제는 이동 호스트들에게 추가적인 오버헤드를 발생시키는 요인이 된다. 따라서 이동 호스트의 특성으로 인한 변화되는 클러스터의 변화를 최소화하기 위한 방법이 제시되어야 한다. 알고리즘 2는 클러스터 변화를 최소화하기 위해 사용되는 알고리즘이다.

3.4.3 브릿지(bridge) 호스트의 선출과 유지

클러스터 헤드와 멤버를 갖는 클러스터가 형성되면 클

러스터와 클러스터간의 데이터를 전송해줄 브릿지 노드를

```

if (cluster head A finds a new host in his region by CFP packet) {
  if (a new host is a cluster head) {
    compare its PV to new cluster head's PV;
    if (its own PV > a new cluster head's PV) {
      continue to act cluster head;
      record CFP packet's route information in RT;
    }
    else {
      give up cluster head;
      record CFP packet's route information in RT;
      trigger cluster formation procedure;
    }
  }
  else {
    record CFP packet's route information in RT;
  }
}
if (cluster head loses its member by expire time in RT) {
  clear the entry of the member;
}
if (cluster member loses all its own cluster heads) {
  become cluster head;
}
if (cluster member loses a host in his region by expire time in RT) {
  clear the entry of the host;
}
}

```

(알고리즘 2) 클러스터 유지 알고리즘

선출해야 한다. 클러스터가 형성되면 멤버 호스트중에서 두 개 이상의 클러스터 헤드를 갖는 호스트는 BE에 자신이 연결된 클러스터와 클러스터까지의 거리를 기록하여 CFP 패킷과 함께 이웃 호스트에게 주기적으로 전송한다. BE에 있는 cluster head ID 필드는 자신과 연결된 클러스터 헤드의 ID를 나타내고 hop은 그 클러스터 헤드까지의 거리를 나타낸다. BE는 브릿지 후보 호스트에게 유효한 필드이다. 알고리즘 3은 브릿지 호스트 선출 과정이다.

```

candidate bridge host A broadcasts CFP packet with BE;
if (candidate bridge host A receives CFP packet with BE for the same cluster head pair) {
  if (its own hop < other candidate bridge host hop) {
    set CFP packet's S field to 3;
    continue to broadcast CFP packet with BE;
  }
  else if (its own hop == other candidate bridge host hop) {
    if (its own PV > other candidate bridge host's PV) {
      set CFP packet's S field to 3;
      continue to broadcast CFP packet with BE;
    }
    else {
      continue to broadcast CFP packet without BE;
    }
  }
  else {
    continue to broadcast CFP packet without BE;
  }
}
else {
  set CFP packet's S field to 3;
}
}

```

(알고리즘 3) 브릿지 노드 선출 알고리즘

브릿지 호스트가 선출되면 브릿지 호스트는 BE 필드와 함께 S 필드의 값을 3으로 설정하고 방송하게 된다.

4 경로 설정과 유지

4.1 경로 설정

이 논문에서는 CBRP에서와 같이 클러스터 정보를 이용하여 네트워크 오버헤드를 줄이는 방법을 사용한다.

목적지에 대한 경로 정보가 필요한 송신 노드는 자신의

영역안에 자신이 원하는 목적지가 있는지를 확인한다. 만약 원하는 목적지 정보가 없으면 RQP 패킷을 작성하여 자신의 클러스터 헤드에게 전송하게 된다. 이때 사용되는 RQP 패킷의 Type 필드는 4이며 Neig head ID는 이웃 클러스터의 ID, SQ는 일련번호를 나타낸다. Source ID는 RQP 패킷을 생성한 호스트의 주소 Dest ID는 경로 정보가 필요한 특정 목적지를 나타낸다. 알고리즘 4는 각 호스트에서 RQP 패킷의 처리를 보여준다.

```
// source node operation
check whether the destination is within its region by RT;
if (the path to the destination exists in RT)
    unicast data packet;
else
    send RQP packet to its cluster heads;
// intermediate node operation
check whether the destination is within NHRT;
if (the destination exists in NHRT)
    send RRP packet to source node;
else
    send RQP packet to next host;
// cluster head operation
cluster head receives RQP packet;
check whether the destination is within its region by RT;
if (the path to the destination exists in RT)
    record RQP packet's information in RRHT;
    unicast RQP packet;
else
    record RQP packet's information in RRHT;
    send RQP packet to its neighbor cluster head;
// destination node operation
send RRP packet to source node;
```

(알고리즘 4) 각 호스트에서의 RQP 패킷의 처리

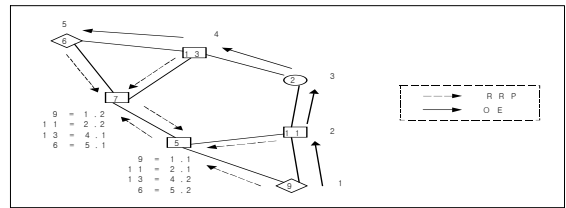
RQP 패킷을 수신한 목적지 노드나 목적지에 대한 경로 정보를 가지고 있는 중간 노드들은 RRP를 작성하여 송신 노드로 전송하게 된다. RRP 패킷은 RRHT에 의해서 송신 노드로의 역경로를 알 수 있다. RRP 패킷의 Type 필드는 5이면 Neig head ID는 RQP를 송신한 이웃 클러스터 헤드의 주소이다. SQ는 일련 번호이고, Q는 일반적으로 0으로 설정되고 1인 경우는 경로 재 검색을 위해서 사용되는데 이 예제는 뒤에서 설명한다. Dest ID는 목적지의 주소 hop은 목적지에서 송신지로 가는 경로의 hop을 나타낸다. 알고리즘 5는 RRP 패킷을 전송받은 호스트들의 동작 과정이다.

```
//destination operation
destination host send RRP packet to neighbor cluster head that RQP packet is received from;
// intermediate node receives RRP
record RRP packet's information in NHRT;
send CFP packet with OE;
send RRP packet to next hop host for reaching neighbor cluster head;
// cluster head receives RRP packet
record RRP packet's information in NHRT;
record RRP packet's information in RRHT;
if (source node exists in its region)
    send RRP packet to source node;
else
    send RRP packet to its neighbor cluster head;
```

(알고리즘 5) 각 호스트에서의 RRP 패킷의 처리

그림 6은 멀티 홉 클러스터 기반의 라우팅 프로토콜의

경로 최적화 과정을 보여준다.



(그림 6) 경로 최적화 과정

RRP 패킷을 전송받은 노드들은 OE를 생성하여 RRP 패킷의 Hop, SQ, Dest ID를 복사한다. 이 OE를 CFP 패킷과 함께 자신의 region 안에 있는 호스트에게 전송한다. OE를 전송받은 호스트들은 optimization table(OT)에 수신된 OE정보를 기록하여 최적화를 수행한다. 그림에서 5번 노드와 7번 노드의 OT 모습을 보여주고 있다. 5번 노드는 9번 노드와 6번 노드로부터 받은 OT정보를 통하여 9, 11, 2, 4, 5로 가는 4홉의 거리(5-1) 보다 적은 3홉의 거리 (1+2)를 알게 된다. 7번 노드에서도 9번 노드와 6번 노드로부터 받은 거리가 기존 RRP의 4홉의 거리보다 (2+1)이 적다는 것을 알게 된다. 따라서 5번 노드와 7번 노드는 NHRT를 RRP의 경로를 수정하고 7번 노드는 6번 노드에게 RRP를 전송한다. RRP의 패킷의 Type 필드는 8이며 Dest ID는 목적지 노드의 주소, Opt-Source ID는 RRP 패킷을 생성한 노드로 여기서는 7번 노드가 된다. Hop은 새로 적용될 목적지까지의 거리가 된다. 패킷을 전송받은 6번 노드는 자신의 NHRT를 수정하고 RRP를 송신지 노드로 전송한다. RRP 패킷은 각 노드를 거치면서 Dest ID에 대해서 NHRT의 경로 엔트리의 hop을 수정하게 된다.

4.2 경로 유지

기존의 ad-hoc 네트워크 라우팅 프로토콜은 목적지에 대한 첫 경로가 설정되고 난 다음 설정된 경로에 변화가 생기거나 경로 실패가 발생하면 새로 경로 발견 과정을 수행해야 했다. 멀티 홉 클러스터 라우팅 프로토콜에서는 각 노드들이 region 정보를 항상 유지함으로써 region의 크기에 해당하는 지역 경로 변화를 해결할 수 있다. 다음은 데이터 패킷의 전송과 경로 유지에 대한 설명이다.

Route packet(RP)는 MHCPR에서 사용되는 데이터 패킷의 형태를 보여준다. Type 필드 값은 7이며 U는 지역적인 경로 복구가 이루어졌을 때 1로 설정되고 그 외의 경우는 0이 된다. Past r-hop list는 RP 패킷이 지나온 경로 중 부분 호스트의 리스트로 현재 호스트로부터 클러스터 크기의 홉 수만큼 지나온 리스트의 목록이 된다. Dest ID는 목적지 노드의 주소의 값이 되고 DATA는 실제 데이터 정보가 된다.

4.3 지역적인 경로 최적화

처음 데이터가 전송되고 난 후 시간이 경과됨에 따라서 처음 설정된 경로 정보는 이동 호스트의 이동으로 최적화되지 않은 경우가 발생하게 된다. Past-r hop list 정보가 최적화되지 않은 경로 정보를 지역적으로 최적화하기 위해서 사용되는 정보이다. 각 호스트는 자신이 유지하고 있는 region 정보와 이 축적된 경로 목록을 비교하여, 목록에 있는 노드에 대해서 region 정보가 더 작은 hop을 가지고 있는 경우 각 호스트는 그 노드에게 RUP 패킷을 전송하여 기존의 경로 정보를 수정하게 한다. 이 RUP 패킷은 경로를 따라 송신 노드 방향으로 전송되어 목적지에 대한 hop 수를 수정하게 해준다. 위의 그림에서 클러스터의 크기가 2홉이고 데이터가 9, 11, 2, 13, 6의 경로를 따라서 전송된다고 가정하면 6번 노드는 2와 13, 13번 노드는 11과 2를 Past r-hop list로 가지게 된다. 이 정보는 지역 정보(RT)와 비교되어 데이터가 전송되는 도중 경로 최적화 작업을 수행하게 된다. 클러스터가 커지는 경우 지역적으로 복구될 수 있는 경로의 크기는 더욱 증가 될 것이다.

4.4 경로 복구

이동 호스트의 이동은 위에서 설명한 것 같이 경로에 대한 변화와 함께 경로 실패를 발생시키게 된다. 기존의 라우팅 프로토콜들은 경로 실패가 발생하는 경우 송신 노드가 새로 경로 발견 절차를 수행하여 추가적인 오버헤드를 발생시키는 어려움이 있었다. MHCRP에서는 새로운 경로 발견 절차를 수행하기 전에 경로 복구를 위해서 두 단계의 사전 경로 복구 과정을 수행한다.

먼저 경로 실패가 발생하면 경로 실패를 발견한 노드는 송신지 노드에게 Route error packet(REP)를 전송한다. REP의 Type 필드의 값은 6이며 Hop은 경로 실패가 발생한 노드의 거리를 나타낸다. Dest ID는 목적지 주소, Pre link fail ID와 Next link fail ID는 경로가 실패된 링크 정보를 나타낸다. REP를 전송받은 노드는 경로 복구 시간(route restore time) 동안 경로 발견 절차를 수행하지 않는다. 경로 실패를 발견한 노드는 경로 실패가 일어난 next hop 노드를 자신의 RT에서 찾는다. 만약 RT에서 next hop 노드를 발견하면 RP 패킷의 U필드를 1로 설정하여 next hop 노드에게 전송한다. U 필드의 설정은 목적지 노드가 패킷을 전송 받은 경우 새로이 RRP 패킷을 송신노드로 전송하여 경로 최적화 작업을 수행하게 한다.

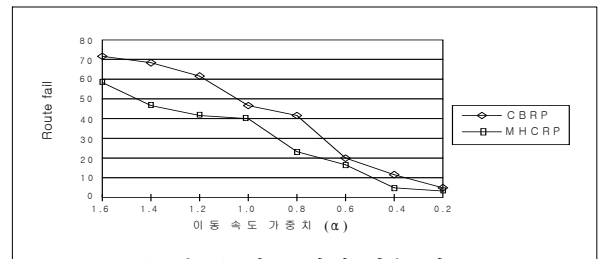
RT에 next hop 노드가 존재하지 않으면 경로 실패를 발견한 노드는 자신의 클러스터 헤드에게 REP 패킷을 전송한다. 이때 REP 패킷의 Q 필드의 값을 1로 설정하게 되는데 이 REP 패킷은 모든 이웃 클러스터 헤드에게 전송되는 것이 아니라 초기 목적지 경로 발견을 위해서 설정된 클러스터 헤드들, 즉 목적지 노드의 경로 정보를 획득하기 위해서 목적지에 대한 응답 패킷을 전송하는데 사용되었던 클러스터 헤드들에게로만 전송되게 된다. 이 정보는 REP 패킷과 RRP 패킷이 전송되는 도중 RRHT에

저장되게 된다. 이러한 경로 복구 방식은 이동 호스트들의 이동이 지역에 기반하기 때문에 새로운 경로 설정 보다 빠른 경로 복구를 보여준다.

위에서 설명한 경로 복구가 실패하면 송신지 노드는 경로 설정 과정을 재 수행하게 된다.

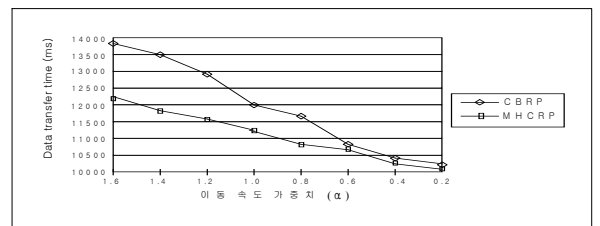
5. 시험 및 평가

본 논문에서 제안하는 시스템 모델의 평가를 위하여 기존 경로 설정 방법인 CBRP와의 비교를 통해서 시스템의 성능을 평가한다. 그림 7과 8은 1000개의 데이터 패킷을 전송하는 경우 경로 실패의 회수와 데이터 전송 시간을 비교한 그래프이다.



(그림 7) 경로 실패 회수 비교

결과를 통해서 이동 호스트의 이동 속도가 빨라질수록 데이터 전송 시간에 큰 영향을 미치게 된다.



(그림 8) 데이터 전송 시간 비교

6. 결론

기존 클러스터 기반의 ad-hoc 네트워크 라우팅 프로토콜은 클러스터의 크기를 고정된 크기로 설정함으로써 네트워크 변화에 따른 오버헤드와 지연시간에 대해서 능동적으로 대처할 수 없었다. 따라서 이 논문에서는 ad-hoc 네트워크의 이동 호스트를 클러스터링 방식으로 멀티 홉을 가지는 클러스터를 구성할 수 있는 방법을 제시하였다.

[참고문헌]

- [1] J. J. Garcia-Luna-Aceves, M. Spohn, and D. Beyer, "Source Tree Adaptive Routing (STAR) Protocol," Internet Draft, IETF, Oct. 1999.
- [2] C. E. Perkins, "Mobile Ad hoc Networking Terminology," Internet Draft, IETF, Nov. 1998.
- [3] T. Imielinski and H. F. Korth, Mobile Computing, Kluwer Academic Publishers, 1996, pp. 153-181.
- [4] S. Corson, and J. Macker, "Mobile Ad hoc Networking(MANET)," Internet Draft, IETF, Oct. 1998.
- [5] D. B. Jonson, "Routing in Ad hoc Networks of Mobile Hosts," In Proceeding of The IEEE Workshop on Mobile Computing Systems and Application, Dec. 1994.
- [6] M. S. Corson, S. Papademetriou, and P. Papadopoulos, V. Park, A Qayyum, "An Internet MANET Encapsulation

Protocol(IMEP) Specification," Internet Draft, IETF, Aug. 1999.