

# On-Demand 라우팅 프로토콜을 사용하는 에드혹

## 네트워크에 적합한 P2P 시스템

김영복<sup>o</sup> 장주욱  
서강대학교 전자공학과  
{virtuoso<sup>o</sup>,jjang}@eeca1.sogang.ac.kr

### P2P System for Ad-hoc Network Using On-Demand Routing Protocol

Youngbok Kim<sup>o</sup> Juwook Jang  
Sogang University Electronic Engineering Dept.

#### 요 약

지금까지의 P2P 알고리즘들은 링크가 비교적 안정적이고 움직임이 적은 유선망을 기반으로 만들어졌다. 하지만 무선의 경우 무선 전파의 손실률이 높아 링크의 안정성이 떨어지고 노드들의 이동성 때문에 링크의 연결 관계가 수시로 변하게 된다. 이러한 이유로 기존의 P2P 알고리즘들을 무선에 직접 적용하기 힘들다. 따라서 본 논문에서는 기존의 P2P 방식이 무선 에드혹 네트워크에서 나타낼 수 있는 문제점을 지적하고 이를 해결하기 위한 P2P 방식을 제안하였고 실험을 통하여 검증하였다.

#### 1. 서 론

P2P 네트워크에서 자료 검색 방식으로는 Napster[1], Gnutella[2], Chord[3] 등과 같은 방식들이 있다. Napster는 중앙 집중형 검색 방식으로 중앙 검색 서버에 접속하는 동시에 자료가 있는 위치를 알 수 있는 방식이다. 하지만 이 경우 중앙 서버의 신뢰도가 중요하고 별도의 서버를 두어야 한다. Gnutella는 별도의 서버를 두지 않는 분산형의 모델로써 각각의 노드(node)들이 가지고 있는 자료들을 색인 시켜놓고 원하는 자료를 검색하기 위해 쿼리(query)를 브로드캐스트(broadcast)하여 원하는 자료를 찾는 방식이다. 이 경우  $O(N)$ 에 해당하는 쿼리 메시지(message)를 보내야하므로 네트워크에 불필요한 메시지들이 대역폭을 소비할 수 있는 위험을 가지고 있다. Chord의 경우 위 두 가지 문제점을 해결하기 위하여 나온 검색 방식으로  $O(\log N)$ 의 쿼리 메시지만으로 검색을 하는 장점을 가지고 있다.

위의 대표적인 세 가지 경우 모두 유선망을 기본으로 한 P2P 알고리즘이며 링크의 상태가 안정적일 때 정상적으로 동작할 수 있다. 하지만 무선 에드혹 네트워크에서는 무선 특유의 전송 에러(error)와 노드들의 움직임에 따른 링크 에러가 많기 때문에 위의 알고리즘을 그대로 적용하기 힘들다.

또 실제 노드의 분포는 노드의 이동성으로 인하여 수시로 변하고 응용계층에서의 노드의 분포는 정적이기 때문에 그 차이는 노드의 움직임이 많으면 많을수록 커지게 된다. 이러한 경우 쿼리 메시지를 중계할 때 응용계층에서의 홉(hop)수보다 실제로는 더 많은 홉을 지나가게 되고 따라서 쿼리가 손실될 확률이 높아지게 되어 히트

률(Hit ratio)이 낮아지게 된다.

이를 해결하기 위한 방법으로 본 논문에서는 대표적인 On-demand 라우팅 프로토콜인 DSR[4]를 사용하여 쿼리 검색과 동시에 경로 설정을 하는 방법과 쿼리를 캐쉬(cache)하는 방법을 사용한 에드혹 망에 적합한 P2P 시스템을 제안한다.

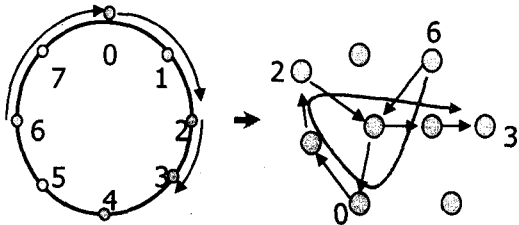
#### 2. 기존 P2P 시스템의 문제점

기존 P2P 방식의 문제점은 이동성이 높은 무선 에드혹 네트워크에서는 응용계층에서의 노드 분포와 실제 노드의 분포가 틀리다는 것이다. Chord와 같은 P2P 방식은 응용계층상에서의 계승자(successor)노드로 쿼리 메시지가 이동한다. 이때에 그림 1에서와 같이 6번노드의 계승자가 0이고 0의 계승자가 2, 2의 계승자가 30이어서 3번에서 원하는 자료가 검색이 된다면 응용계층상에서는 3홉을 지나온 것이지만 실제 에드혹 망에서는 여러 홉의 중간노드들을 건너온 것이다. 따라서 실제 3번 노드가 6번과 가까운 거리에 있어 쿼리의 경로가 짧아질 수 있음에도 불구하고 응용계층상에서의 쿼리의 경로는 실제 노드의 분포를 고려하지 않기 때문에 경로가 길어질 수 있다. 에드혹망에서 각 링크는 깨어질 확률이 높고 이러한 이유로 여러홉을 건너서 쿼리가 갈 경우 그만큼 쿼리가 손실될 확률이 높다.

또 On-demand 라우팅 프로토콜을 쓰는 에드혹 네트워크의 경우 응용계층상의 노드로 쿼리를 중계하기 위해서는 중계하려는 노드의 경로를 모를 경우 브로드캐스트

를 하여 경로를 찾은 후 쿼리를 보내야한다. 따라서 최악의 경우 중계하는 노드의 경로를 모두 모를 경우 중계 노드 수만큼의 브로드캐스트가 일어나게 된다. 이 경우 많은 양의 제어패킷이 생기기 때문에 네트워크의 자원을 낭비하게 된다.

이와 같은 문제점들이 발생하기 때문에 On-demand 라우팅을 하는 애드혹 네트워크의 특성을 반영한 새로운 P2P시스템이 필요하다.



응용계층상에서 쿼리의 경로 실제 노드 분포에 따른 쿼리의 경로

그림 1 응용계층상에서의 쿼리의 경로와 실제 노드 분포에 따른 쿼리의 경로

### 3. 제안하는 무선 애드혹 네트워크 P2P 시스템

On-demand방식의 라우팅 프로토콜 DSR에서는 경로 설정을 하기 위하여 브로드캐스트를 사용한다. 하지만 이 경우 한번 브로드캐스트를 한 패킷일 경우 다시 브로드캐스트 하지 않으므로 어느 정도 불필요한 브로드캐스트 패킷의 수를 줄일 수 있다. 대부분의 On-demand 라우팅 프로토콜 (AODV, DSR등)에서는 브로드캐스트는 경로 설정을 위하여 필요하다.

DSR에서 Route 경로를 찾기 위한 방법을 응용하여 P2P 검색에 사용할 수 있다. 쿼리를 브로드캐스트하면서 쿼리가 가는 경로를 각각 기록하고 원하는 자료가 있는 노드까지 쿼리가 도착하면 쿼리에 대한 응답을 보낼 때 쿼리가 지나온 경로를 포함하여 보내면 응답을 받은 노드는 이 경로를 가지고 라우팅 테이블(table)을 갱신한다.

예로 그림 2에서와 같이 1번 노드에서 키(key) K=30에 해당하는 자료를 찾기 위하여 Query 메시지를 브로드캐스트하면서 쿼리가 가는 경로를 저장한다. 이 때 8번 노드에서 K=30에 해당하는 자료를 가지고 있다면 쿼리가 8번까지 왔을 경우 쿼리에 대한 응답으로 1,2를 거쳐서 쿼리가 왔음을 응답 패킷에 넣어 1번 노드로 보내면 1번 노드는 8번 노드로 가는 라우팅 경로를 1-2-5-8 로 설정 한다.

이와 같이 라우팅 경로 설정과 쿼리 검색을 동시에 함으로서 쿼리 검색과 라우팅 경로 설정을 따로 할 때 발생하는 네트워크 제어 패킷의 수를 줄일 수 있다. 또 무선 통신의 특성상 패킷의 손실로 인하여 중간에 쿼리가 손실 되더라도 대체경로를 통하여 원하는 키값이 있는 노드에 쿼리가 도착할 확률이 높아지고 응용계층상에서

의 노드의 분포와 실제 네트워크에서의 노드의 분포가 같기 때문에 쿼리가 가는 출수가 짧아져 쿼리가 중간에 손실될 확률이 작아지게 되며 지연시간도 줄어들게 된다. 또 쿼리가 있을 때 브로드캐스트를 하기 때문에 노드의 이동성으로 네트워크의 노드 분포가 변하여도 쿼리 검색이 원활하게 이루어질 수 있다.

보다 빠른 쿼리 검색을 위하여 각 노드마다 키와 키를 가지고 있는 노드의 주소와 경로를 캐싱하는 방법도 사용할 수 있다. 이런 방식을 사용할 경우 해당 노드까지 쿼리가 도착하지 않더라도 중간 노드에서 해당 자료에 대한 키를 보고 키를 가지고 있는 노드를 알려줘 쿼리 시간을 보다 단축시킬 수 있다.

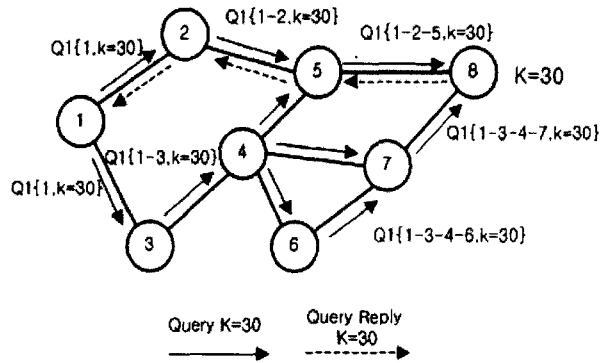


그림 2 제안된 쿼리와 경로 설정을 동시에 하는 방식

### 4. 실험

#### 4.1 실험 방법

Glomosim 2.0[5]을 사용하여 무선 환경에서 전파범위 250m 노드 수 100개, 라우팅 프로토콜은 DSR로 하였으며, 이동성 모델은 랜덤웨이 포인트(Randomway point) 모델을 이용하였고 노드의 정지 시간은 0~1500까지 다양하게 하였다. 전체 네트워크 영역 1500m\*1500m에서 실험을 하였고 전체 노드에서 무작위로 쿼리를 발생시키도록 하여 실험을 하여 제안된 방식의 P2P방식과 Chord 방식을 비교하였다.

#### 4.2 실험 측정계수

실험을 위하여 측정한 계수는 Hit Ratio와 컨트롤 오버헤드(Control Overhead), hit된 쿼리들의 평균 출수 이다. Hit ratio는 보낸 쿼리의 수와 hit된 쿼리수의 비율이고 컨트롤 오버헤드는 쿼리 하나당 발생한 제어패킷의 수이다.

#### 4.3 실험 결과

그림 3은 제안된 방식과 Chord방식의 Hit Ratio를 비교한 것이다. Chord방식에서는 계승자들로 쿼리가 가기 때문에 중간 계승자가 쿼리를 중계하지 못하게 되면 쿼리 검색이 끝나게 된다. 실험결과 쿼리의 Hit율은 노드의 이동성이 증가함에 따라서 현저히 떨어짐을 알 수 있고

움직임이 없을 경우에도 무선의 높은 데이터 손실률로 인하여 쿼리가 중간에 손실되는 것을 알 수 있다. 반면 제안된 방식은 높은 이동성인 환경에서도 쿼리가 가는 길이 다양하기 때문에 쿼리 검색의 Hit율이 높은 것을 볼 수 있고 움직임이 없는 경우 다양한 대체경로로 쿼리가 가기 때문에 높은 Hit율을 보이고 있다.

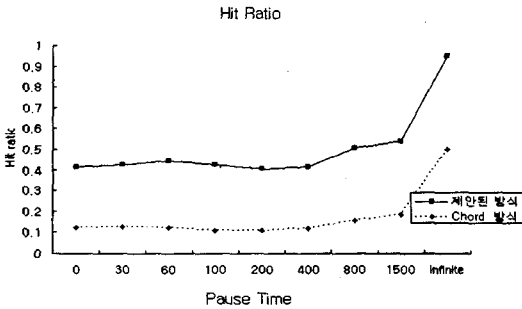


그림 3 제안된 방식과 Chord 방식의 Hit Ratio

그림 4는 제안된 방식과 Chord 방식의 컨트롤 오버헤드를 측정 한 것으로 제안된 방식과 차이가 없음을 알 수 있다. 이는 Chord방식의 경우 낮은 hit율을 보인 것으로 보아 대부분의 쿼리들이 손실되었기 때문에 다소 낮은 컨트롤 오버헤드를 나타내었다. 그러나 제안된 방식이 비슷한 컨트롤 오버헤드를 보이면서 높은 히트율을 보였으므로 성능향상을 보인 것을 알 수 있다. 제안된 방식에서는 1번의 Query로 1번의 Broadcast가 일어나지만 Chord의 경우 여러 홉을 지나야 할 경우가 많으므로 여러 번의 Broadcast가 일어날 수 있어 결과적으로 같은 Hit율에 더 높은 컨트롤 오버헤드를 보일 수 있다.

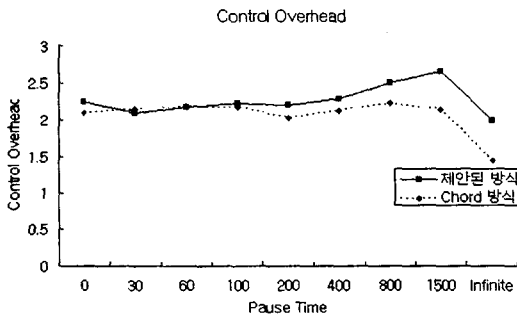
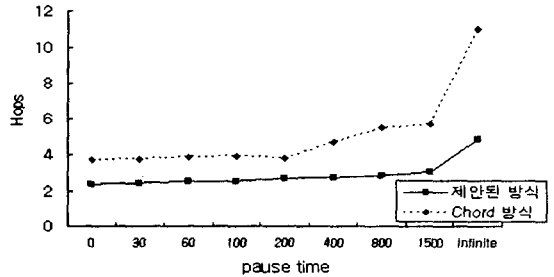


그림 4 제안된 방식과 Chord방식의 Control Overhead

그림 5는 Hit된 쿼리들의 평균 홉수를 나타낸 것으로 Chord의 경우 홉수가 작은 쿼리들만이 히트되는 반면 제안된 방식은 두배 정도의 홉을 지나도 쿼리들이 Hit되는 것을 알 수 있다. 또 움직임이 작은 경우를 살펴보면 Chord의 경우 대부분의 쿼리들이 평균적으로 제안된 방

식보다 여러홉을 지나는 것을 알 수 있다. 결과적으로 Chord가 여러 홉으로 쿼리가 가기 때문에 중간에 쿼리가 손실될 확률이 높고 컨트롤 오버헤드도 증가하는 것을 알 수 있고 제안된 방식이 보다 에드혹 네트워크에 적합함을 알 수 있다.

Hit된 Query의 평균 홉 수



5. 결론

지금까지 제안된 P2P방식은 유선망에 적합하도록 만들어졌기 때문에 무선 에드혹 네트워크에 적합하지 않다. 무선 에드혹 네트워크는 노드의 이동성으로 인하여 응용 계층에서의 노드의 분포와 실제 노드의 분포가 차이가 많이 나고 결과적으로 쿼리가 가는 홉수가 증가하게 된다. 쿼리의 홉수가 증가하면 손실률이 증가하여 Hit율이 낮아지게된다. 반면 본 논문에서 제안한 방식으로 쿼리 검색과 라우트 경로 설정을 동시에 하게 될 경우 쿼리가 최적화 된 경로로 가고 여러 가지 대체경로로 가게 되므로 이동성이 높은 무선 에드혹 네트워크에 적합하다.

6. 참고문헌

- [1] Napster. <http://www.napster.com/>
- [2] Gnutella. <http://gnutella.wego.com/>
- [3] Ion Stoica, Robert Morris, David Karger, M.Frans Kaashoof, Hari Balakrishnany, Chord : A Scalable Peer-to-peer lookup Service for Internet Applications MIT Laboratory for Computer Science, In ACM SIGCOMM Aug. 2001.
- [4] David B. Johnson, David A. Maltz Yih-Chun Hu, and Jorjeta G. Jetcheva, DSR :The Dynamic Source Routing Protocol for Multihop wireless Ad Hoc Networks, In Charles E.Perkins, editor, Ad hoc Networking (Boston, Ma: Addison-Wesley, 2001), pages 139-172.
- [5] L. Bajaj et al. Glomosim : A Scalable network simulation environment. Technical Report 990027, UCLA Computer Science Department, 1999.