

이동 Ad Hoc 네트워크에서 부하분산 제어기법

송찬호

고려대학교 컴퓨터과학기술대학원

notdel@korea.ac.kr

A Load Balancing Control Method in Mobile Ad-Hoc Network

Chan-Ho Song

Graduate School of Computer Science and Technology, Korea Univ.

요 약

AODV나 DSR과 같은 Reactive 라우팅 방식은 트래픽이 발생하는 시점에서 경로를 탐색하는 방법으로서 최적 메시지 전송 경로가 유효하다면 트래픽의 증가와 과도한 에너지 소비를 고려하지 않고 계속적인 동일 전송 경로의 사용으로 인해 특정 노드에 대하여 에너지 소모가 심화될 뿐만 아니라 전송지연이 발생된다. 본 논문에서는 과도한 트래픽 상태를 분산 시켜 전체 네트워크의 균형적인 에너지 소비를 위하여 Reactive 라우팅 방법에서 특정 노드의 트래픽 상태, 에너지 상태, 라우팅 시간을 고려한 노드 상태 기반 부하 분산 방법(NSLB : Node Status-based Load Balancing Method)을 제안한다.

1. 서론

Ad Hoc 네트워크는 전시 상황, 재난 발생 지역에서와 같이 신속하고 일시적인 네트워크망 구성이 요구될 때 구축될 수 있다. Ad Hoc 네트워크는 기지국 혹은 액세스 포인트와 같은 중재자가 없이 이동 노드들간에 자체적으로 연결 설정이 이루어지며 이동 노드는 호스트이자 라우터로서 작동한다. Ad Hoc 라우팅 프로토콜은 IETF (International Engineering Task Force)/MANET(Mobile Ad Hoc Network)[1]의 주도로 표준화 작업이 진행 중이며 Table-driven 방식과 On-demand 방식[2]과 GPS 시스템에서 제공하는 지리 위치 정보에 의한 라우팅 방식이 있다. Table-driven 방식은 모든 이동 노드들이 라우팅 정보를 주기적으로 또는 네트워크 변경이 발생할 경우 네트워크 전체로 변경된 라우팅 정보를 전파하고 각 노드들은 자신의 라우팅 정보를 수신한 정보로 변경하여 최신의 라우팅 정보를 유지하도록 하는 방식이다. On-demand 라우팅 방식은 전송 요구가 발생할 때마다 경로 선정이 이루어지는 방식으로서 네트워크 변화에 따른 라우팅 테이블 변경 오버헤드를 줄일 수 있다는 장점이 있지만 경로를 선정할 경우 지연시간이 발생하는 단점을 지닌다. 또한 On-demand 라우팅은 소스 노드에서 목적 노드까지의 최적 경로 설정 후 메시지 전송이 발생하지만 네트워크 구조의 변화가 적은 안정된 상태라면 경로 상의 특정 노드를 라우터로 하여 지속적인 메시지 전송이 발생한다. 그 결과 발생한 과도한 트래픽으로 인해 전송지연을 유발하고 라우터로 이용되는 노드의 에너지 소모를 증가시킴으로써 라우터로서의 기능에 대부분

의 에너지를 낭비한다. 본 논문에서는 트래픽 집중을 완화시키는 방안으로 경로 선정에서 과도한 트래픽 상태에 있는 노드를 제외시키고 또 다른 경로를 선택하여 트래픽을 분산시킴으로써 균형적인 에너지 소모를 유도하는 노드 상태 기반 부하 분산 방법 (NSLB : Node Status-based Load Balancing Method)을 제안한다.

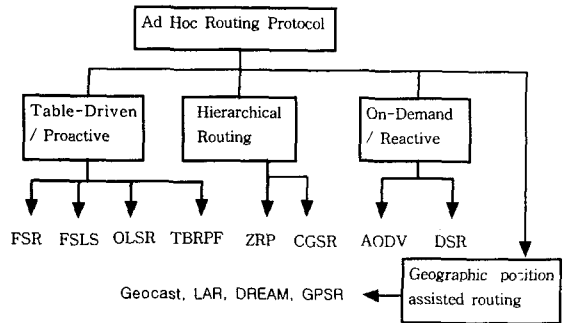


그림 1. Ad-Hoc 라우팅 프로토콜 종류

2. 관련연구

AODV(Ad hoc On-demand Distance Vector)[3]는 목적 노드까지 최단 경로를 라우팅 경로 탐색 과정을 통해 찾아내는 대표적인 요구기반 라우팅 프로토콜이다. 우선 데이터를 전송하고자 하는 노드가 자신의 라우팅 테이블에 원하는 목적 노드까지의 경로가 존재하는지 확

인하고 원하는 경로가 없다면 경로 탐색을 시작한다. 소스 노드는 RREQ(Route REQuest) 메시지를 생성하여 이웃 노드로 브로드캐스팅하고, RREQ 메시지를 수신한 중간 노드가 목적 노드로의 경로 정보를 가지고 있지 않을 경우 소스 노드까지의 역 경로를 라우팅 테이블에 저장한 후 RREQ 메시지를 이웃 노드로 다시 브로드캐스팅한다. 그러나 RREQ 메시지를 수신한 노드가 자신의 라우팅 테이블에서 목적 노드까지의 경로를 포함하고 있다면 RREQ 메시지의 소스 시퀀스 번호와 라우팅 엔트리의 목적 시퀀스 번호를 비교하여 목적 시퀀스 번호가 소스 시퀀스 번호 보다 크거나 같으면 RREP(Route REPLY) 메시지를 생성하여 응답함으로써 역 방향 경로가 설정되고 RREP 메시지를 수신한 소스 노드는 순방향 경로로 데이터를 전달 할 수 있게 된다. 반대로 시퀀스 번호가 작을 경우에는 역 경로 저장 후 이웃노드로 브로드캐스팅된다. 소스 노드와 목적 노드사이에서 단절이 발생할 경우 단절이 목적 노드로부터 가까운 위치에서 발생한다면 지역 경로 복구 방법으로 새로운 경로를 탐색한다. 목적 노드까지 지역 경로가 존재하지 않거나 목적 노드로부터 멀리 떨어진 위치에서 단절이 발생한 경우 경로 단절을 감지한 노드는 RERR(Route ERROr) 메시지를 생성하여 소스 노드에게 통보하고 RERR 메시지를 수신한 소스 노드가 아직 목적 노드까지의 경로를 필요로 한다면 경로 재설정을 위해 시퀀스 번호를 증가시켜 RREQ 메시지를 브로드캐스팅한다. 이런 기본 적인 On-demand 라우팅 프로토콜을 부하 분산에 적용한 것이 NSLB이며 성능분석은 향후 과제로 남겨둔다.

3. 노드 상태 기반 부하 분산 방법(NSLB)

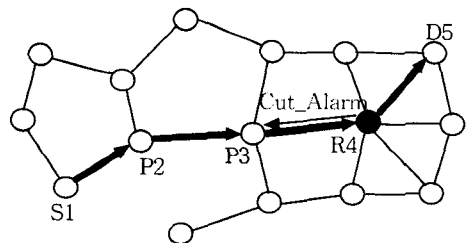
AODV와 같은 On-demand 라우팅 프로토콜은 노드마다의 이동성이 낮아 네트워크 망이 안정된 상태일 경우 특정노드의 트래픽이 고려되지 않은 상태에서 최단 경로만을 선택하여 데이터를 전송하기 때문에 특정 노드에 트래픽이 집중되어 전송 지연 및 과도한 에너지 소비가 발생하여도 새로운 전송경로를 검색하지 않는다. 이런 특정 노드의 과도한 에너지 소비를 차단하여 전체 네트워크의 수명을 연장시키기 위한 방안으로 노드 상태 기반 부하 분산 방법(NSLB)을 제안한다. NSLB는 각 노드마다 CPU 성능과 배터리 용량에 따라 결정된 최대 상태 값(max)과 현재 상태 값(state)을 부여한다. 현재 상태 값은 경로상의 특정노드가 라우터로서 작동할 경우 패킷 전송률이 높아지면 에너지 소모율이 증가하고 상대적으로 남은 에너지량이 감소하기 때문에 패킷 전송률(Fo), 에너지 소모율(Con), 남은 에너지량(Re), 라우팅시간(Rot)과 같은 factor들에 의해 결정되며 각 factor 마다 상태 값은 5 level로 분할하여 증감한다.

NSLB는 새로운 경로 탐색의 경우와 라우터로 작동하는 특정 노드가 과도한 트래픽 상태에 도달할 경우 각기 다른 방법으로 작동한다. 새로운 경로 선정시 최대 상태 값에 가중치(0.9)를 곱하여 판단하고 라우터로 작동하는 경우에는 최대 상태 값만으로 판단한다. 노드의 현재 상

태 값이 최대 상태 값보다 크거나 같고 남은 에너지량이 10% 이하일 경우 상태 테이블 값은 Refuse가 된다. 반대로 현재 상태 값이 최대 상태 값보다 낮은 경우 상태 테이블 값은 Pass가 될 것이다.

새로운 경로 탐색의 경우 송신 노드가 자신의 라우팅 테이블에서 목적 노드까지의 경로가 없다면 RREQ 메시지를 생성하여 이웃 노드로 브로드캐스팅하고 RREQ 메시지를 수신한 중간 노드는 목적 노드로의 경로 정보 포함여부와 상관없이 현재 상태 테이블의 값을 확인하여 현재 상태 값이 Refuse 상태라면 RREQ 메시지를 폐기하여 전송 경로 선정시 경로에서 제외된다. 반대로 상태 테이블 값이 Pass 상태라면 역 경로를 저장한 뒤 이웃 노드로 RREQ 메시지를 브로드캐스팅한다. RREQ 메시지를 수신한 중간 노드가 목적 노드로의 라우팅 정보를 포함하고 있다면 RREP 메시지를 생성하기 이전에 RREQ 메시지를 수신한 중간 노드에서 목적 노드까지 경로 상에 있는 모든 노드들의 상태 테이블을 확인한다. 모든 경로상의 노드 상태가 Pass 일 경우 RREP 메시지를 생성하여 소스 노드로 송신하며 경로 상의 노드 중 Refuse 상태인 노드가 있을 경우 지역 경로 탐색으로 새로운 경로를 선택한 다음 RREP 메시지를 소스 노드로 통보한다. RREP 메시지를 수신한 소스 노드는 즉시 데이터를 목적 노드까지 전송한다.

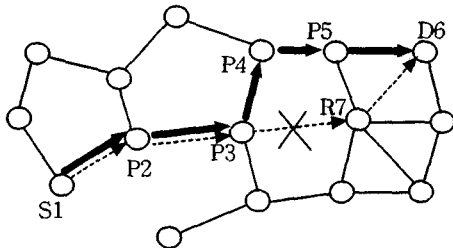
라우터로 작동되는 노드(R4)가 과도한 트래픽 상태에 도달하여 트래픽 상태 테이블이 Pass 상태에서 Refuse 상태로 전환될 경우 노드(R4)를 통해 데이터를 전달하는 경로 중에서 가장 높은 Fo 값을 나타내는 경로의 상위 노드(P3)에게만 Cut_Alarm 메시지를 송신한다.



P : Pass, R : Refuse, S : 소스 노드, D : 목적 노드
그림 2. Cut_Alarm 메시지 송신

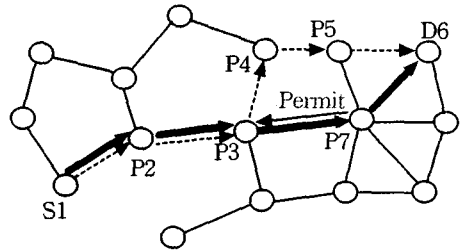
또한 노드는 Cut_Alarm 메시지를 1층 거리만큼만 전파시켜 추가적인 트래픽 발생을 최소화한다. Cut_Alarm 메시지를 수신한 상위 노드(P3)는 경로 단절의 상태로 동일하게 지역 경로 탐색 절차를 시작하며 새로운 경로가 선택될 때까지 전송 데이터를 버퍼에 저장한다. 새로운 지역 경로 탐색 후 선정된 경로로 버퍼에 저장된 데이터를 전달하며 만약 노드(P3)가 경로 탐색에 실패할 경우 RERR 메시지를 소스 노드에게 전송하여 새로운 경로를 탐색할 것을 통지한다. RERR 메시지를 수신한 소스 노드는 NSLB를 이용한 경로탐색을 시작하고 새롭게 선택된 경로를 통해 데이터를 전달한다. 현재 상태 값이 최

대 값 이하로 낮아져 시스템이 안정화 될 때까지 상태 테이블은 Pass 상태로 전환되지 않으며 새로운 전송 경로에도 포함되지 않을 것이다. 그림 3은 NSLB의 새로운 경로 선택을 통한 데이터 전송을 나타낸다.



→ 데이터 전송 경로, -----> 최단 경로
그림 3. NSLB 새로운 경로 선택

그림 3의 노드(R7)가 Refuse 상태에서 Pass 상태로 전환될 경우 Cut_Alarm 메시지를 수신한 상위 노드(P3)에게 Permit 메시지를 전송한다. 소스 노드(S1)는 최단 경로를 통하여 데이터를 다시 전송할 것이다.



→ 최단 전송 경로, -----> 우회 경로
그림 4. Permit 메시지 송신

```

factor_Max = Max / 4_factor
factor_value = factor_Max / 5_level

Fo = factor_value * current Packet_forward_level
Con = factor_value * current Energy_consume_level
Re = factor_value * current Remain_energy_level
Rot = factor_value * current Routing_time_level
state = SUM(Fo, Con, Re, Rot)

if node receive RREQ then
  {{ if max * 0.9 > state AND Re <= 10% then
    state_table = "Pass"
    else state_table = "Refuse"
  }}
  if state_table = "Refuse" then
    abolish the RREQ
  else if route_table = true then
    { if Source_Seq_Num <= Target_Seq_Num then
      { for I = current to dest
        if state_table = "Refuse" then
          { route_table = local_path_searching()
            exit()
          }
        node send RREP to source node
        source node send the packet
      }
      else broadcast RREQ to neighbor nodes
    }
  }
  if node acts as router then
  {{ if max > state AND Re <= 10% then
    state_table = "Pass"
    else state_table = "Refuse"
  }}
  if state_table changed to Refuse then
  { find Maximum Fo among Fos
    send Cut_Alarm message to previous_node in 1 hop
    node_buffer = received packet from source node
    route_table = local_path_searching()
    if local_path_searching = fail then
    { node send RERR to source node }
    source node send the packet
  }
  if state_table changed to Pass then
  { send Permit message to previous_node in 1 hop
    again send the packet through shortest path
  }}
  
```

표 1. NSLB 작동 알고리즘

4. 결론

네트워크 토폴로지의 변화가 심하지 않은 안정된 상태의 Ad-Hoc 네트워크일 경우 발견된 최단거리의 전송 경로에 트래픽이 집중되어 전송지연 등의 문제점이 발생하기 때문에 전송경로를 우회함으로써 언급한 문제를 해결할 수 있다. 그러나 기존 연구들의 경우 이러한 문제를 해결하기 위해 전송경로를 우회하기보다 계속적으로 패킷을 전송함으로써 트래픽을 가중시킨다. 이 논문에서는 네트워크 트래픽을 높일 수 있는 여러 요소들을 복합적으로 고려한 NSLB를 제안한다. NSLB는 Ad-Hoc 네트워크 트래픽을 고르게 분산시킬 수 있으며 분산된 트래픽으로 인하여 노드들이 에너지를 균형적으로 소비하는 장점을 가진다. 향후 연구 과제로는 기존 부하 분산 방법과 본 논문에서 제안한 NSLB와의 성능 비교에 대한 연구가 요구된다.

참고 문헌

- [1] M. S. Corson and J.P. Macker, "Mobile Ad Hoc Networking(Manet) Routing Protocol Performance Issue and Evaluation Considerations," IETF RFC 2501, Jan. 1999.
- [2] D. B. Johnson, D. A. Maltz, Yih-Chun Hu and J. G. Jetcheva "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks(DSR)", Internet Draft, IETF MANET Working group, draft-ietf-manet-dsr-07.txt, Feb. 2002.
- [3] C. E. Perkins, "Ad Hoc On-Demand Distance Vector(AODV) Routing", Internet Draft, IETF MANET Working group, draft-ietf-manet-aodv-12.txt, November 2002.
- [4] Vincent D. Park and M. Scott Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," Proc. of IEEE INFOCOM'97, pp.1405-1413, Apr. 1997.
- [5] Audrey Zhou and Hossam Hassanein, "Load-Balanced Wireless Ad Hoc Routing", Proc. Canadian Conference on Electrical and Computer Engineering, vol. 2, pp1157-1161, 2001.