

Mobile Ad Hoc Networks(MANET)에서의 고정셀을 이용한 Cluster Based Routing Protocol *

정종광⁰ 김재훈
아주대학교 정보통신전문대학원
{charlie⁰, jaikim}@dmc.ajou.ac.kr

Cluster Based Routing Protocol Using Fixed Cell in Mobile Ad hoc Networks (MANET)

JongKwang Jeong⁰ Jai-Hoon Kim
Graduate School of Information and Communication, Ajou University

요 약

Mobile Ad Hoc Network에서는 무선으로 연결된 호스트들이 쉽게 이동할 수 있으며, 미리 설치된 유선망을 이용하는 셀룰러망과 달리 이동 호스트 사이의 통신만으로 이루어진 망이다. Mobile Ad Hoc Network에서는 각각의 노드들의 이동성이 높기 때문에 이 각각의 노드들의 라우팅 경로를 결정하는 것이 중요하다. 이에 따라 Mobile Ad Hoc Network를 위한 많은 라우팅 프로토콜이 제안되었다. 본 논문에서는 기존에 제안된 Cluster Based Routing Protocol(CBRP)를 변형하여 마치 셀룰러망에서의 셀과 같이 고정된 위치를 하나의 셀로 정의하고 그 하나의 셀이 클러스터를 형성하여 라우팅 오버헤드를 줄일 수 있는 기법을 제안한다

1. 서 론

Mobile Ad Hoc Network(MANET)는 미리 설치된 유선망에 의해 동작하는 셀룰러망과는 달리 각각의 호스트들 사이의 무선통신만으로 동작하는 망으로써 이동 호스트가 자신의 전송 범위 밖의 다른 목적지 호스트에 패킷을 전달해야 하는 경우 다른 노드가 이를 중계하여 목적지에 이를 전달할 수 있게 된다. 따라서 이 각각의 노드들은 호스트의 기능과 더불어 라우터의 기능을 함께 수행하여야 한다. 그러나 이 MANET에서는 호스트 자체가 이동을 하기 때문에 항상 이것들의 위치를 정확히 예측하기가 어렵다. 따라서 한 노드에서 다른 노드로 패킷을 보내고자 할 때 라우팅 경로를 찾고 이를 관리하는 것은 그만큼 복잡하고 어렵다. 이에 따라 MANET에서의 좀 더 효율적인 라우팅을 구현하기 위해 많은 프로토콜이 제안되었다. 종래의 유선망에 사용되던 라우팅 프로토콜은 주기적으로 갱신되는 메시지로 인한 오버헤드와 토폴로지의 변화에 대한 적응성이 낮기 때문에 무선 환경에 알맞지 않다. 이를 위해 여러 가지 라우팅 프로토콜이 IETF의 MANET 워킹 그룹에 제안되어 연구되고 있다.

본 논문에서는 기존에 제안된 Cluster Based Routing Protocol(CBRP)를 변형하여 마치 셀룰러망에서의 셀과 같이 고정된 위치를 하나의 셀로 정의하고, 그 하나의 셀이 클러스터를 형성하여 라우팅 오버헤드를 줄일 수 있는 기법을 제안한다. 각각의 클러스터에는 클러스터 헤더가 존재하여 분산 환경에서 조정자 역할을 수행하여 클러스터 내의 라우팅 뿐만 아니라 클러스터내의 한 노드가 데이터 다른 클러스터의 노드에게 패킷을 보내는 경우에도 라우팅 경로를 설정하도록 도와주는 역할을 한다. 클러스터 헤더는 일반 노드들에 비해 더 많은 기능을 수행한다. 따라서 클러스터 헤더가 때때로 병목이 될 수도 있다. 본 논문에서는 기존 cluster protocol에서 존재하는 문제를 해결하기 적당한 셀의 크기를 찾는 것을 목적으로 한다. 적당한 셀의 크기는 시뮬레이션과 수학적 방법을 통하여 구할 예정이다.

2. 관련연구

MANET 은 Mobile IP 와 같이 반드시 고정된 기지국만이 이동 서비스를 지원하는 형태의 네트워크가 아니다. 노드 자신이 Ad-hoc 형태로 네트워크 라우팅 인프라 구조를 형성한다. MANET 을 구성하는 각 노드들은 자유 자재로 이동하는데 따른 제약 사항이 없으며 이에 따라 노드의 빠른 이동에 따른 위치의 변화에 적응 가능한 라우팅 프로토콜이 요구 된다. 또한 무선 이동 호스트들은 대역폭과 전력의 사용에 제한을 가지므로 기존의 유선망에서의 라우팅 프로토콜들은 확장되거나 보완되어야 한다. 이동 Ad-hoc 네트워크에서 동작하는 라우팅 프로토콜은 기존의 고정된 네트워크에서 동작하는 라우팅 프로토콜과 구별되는 몇 가지의 추가된 요구사항을 가지게 된다. 따라서 여러 프로토콜들이 제안 되었으며 현재 IETF 에서 Ad Hoc Network 에서의 라우팅 표준을 제정하기 위해 워킹 그룹을 결성해 연구를 진행 중에 있다. 많은 연구들이 진행되었으나 이중 cluster 관련 알고리즘으로는 National University of Singapore 의 Mingliang Jiang, Jinyang Li, Yong Chiang Tay 가 제안한 Cluster Based Routing Protocol(CBRP)가 있다[1]. 이 프로토콜은 전체 망을 클러스터 단위로 분할하고 클러스터 헤더에 의해 라우팅 수행하게 하여 Route request 가 클러스터 헤더들에게만 전달되며, 게이트웨이 노드들을 경유하여 라우팅 경로가 설정된다. 그리고 또 다른 알고리즘으로는 University of Illinois, Urbana-Champaign 의 Raghupathy Sivakumar, Prasun Sinha, Vaduvur Bharghavan 가 제안한 Core-Extraction Distributed Ad Hoc Routing(CEDAR)가 있다[2]. 이것은 CBRP 와 유사하다. 네트워크에서의 서브넷들은 core 로 구분되며 각 노드들은 적어도 하나의 노드가 core 내의 있는 노드와 인접해야 한다. core 는 주기적으로 각 노드와 그 이웃 노드들 간에 메시지 교환으로 결정되며, 각각의 core 노드들은 인접한 core 까지의 경로를 지역화된 브로드캐스트를 이용하여 결정한다.

본 연구는 정보통신부에서 지원하는 대학기초연구지원사업(2001-103-3)에 의하여 지원되었음

따라서 core 내의 한 노드가 목적인 노드까지의 라우팅 경로 설정은 core 에 의해 결정된다. 따라서 라우팅 경로를 설정하거나 관리 시 각 노드의 부담은 적지만 반대로 core 노드는 더 많은 부담을 가지게 되는 단점이 있다. 본 논문에서는 위에서 언급한 CBRP 알고리즘을 변형하여 네트워크를 셀룰러망에서와 같이 고정된 셀로 분할하여 각각의 셀이 클러스터를 구성하여 라우팅 경로를 설정 시 오버헤드를 줄일 수 있다. 또한 정적인 클러스터 구조로 인하여 동적인 클러스터 구조에 비해 이동 환경에서의 제약 사항 중에 하나인 주파수 대역폭의 재사용이 이루어 질 수 있다는 장점을 갖는 알고리즘을 제안한다.

3. Mobile Ad hoc Networks(MANET)에서의 고정셀을 이용한 Cluster based Routing Protocol

클러스터 구조의 프로토콜은 MANET 환경에 맞도록 설계된 라우팅 프로토콜로 네트워크를 구성하는 노드들을 본포에 따라 여러 개의 중복되거나 분리된 클러스터로 나누어 관리한다. 그리고 하나의 클러스터에 클러스터 헤더가 정해져서 각 영역에 속한 노드들의 정보를 관리한다. 클러스터 사이의 라우팅은 각 클러스터 헤더가 가지고 있는 정보들을 이용하여 동적으로 이루어진다. 노드들을 그룹으로 관리함으로써 라우트 발견 절차 시에 트래픽의 양을 효과적으로 줄이며 속도도 향상시킨다. MANET 은 이동에 따른 동적인 위치의 변화 때문에 네트워크의 범위가 커질수록 오버헤드가 심각하게 증가된다.

3.1 구성 및 동작

MANET 내의 모든 노드들은 송수신기를 장착하고 있으며, 각 노드들은 패킷을 브로드캐스트 한다. 각 노드는 이웃 테이블을 관리하며 이웃 테이블의 노드에게 헬로우 메시지에 포함하여 주기적으로 방송한다. 이웃 노드 B로부터 헬로우 메시지를 수신한 노드 A 는 자신의 이웃 테이블을 다음과 같이 갱신한다. 노드 B의 정보가 자신의 이웃 테이블에 존재하는지 확인하여 존재하지 않다면 항목을 추가하고, B 의 이웃 테이블이 A 의 정보를 포함하고 있으면 A 는 B 로의 링크를 양방향이라고 표시한다. 그리고 노드 B 가 클러스터 헤더이면 A 의 테이블에 B 를 클러스터 헤더로 표기한다. 이웃 테이블의 모든 항목은 타이머에 의해 관리된다. 이웃 노드들의 위상이 변하지 않을 때 각 노드는 서로 양방향 링크를 가지는 모든 노드들의 정보를 가지며 또한 자신으로의 링크를 가지는 노드에 대한 정보도 가지게 된다. 프로토콜의 동작은 다음과 같다.

- ① 클러스터의 헤더 선출은 클러스터 내에서 최저 ID 를 가지는 노드가 선출된다. 노드는 헬로우 메시지로부터 클러스터 형성을 위한 정보를 얻는다. 자신과 양방향으로 연결된 모든 노드들 중에서 가장 작은 값의 ID 를 가지는 노드가 클러스터 헤더가 되며, 새로운 클러스터 헤더는 자신의 광고 메시지 중 첫 번째 항목을 '클러스터 헤드' 로 기입한 후 브로드캐스트 한다. 역으로 클러스터 헤더와 양방향 링크를 가진 모든 멤버 노드들은 그 클러스터의 멤버로 속하게 된다.
- ② 클러스터 관리: 클러스터는 클러스터 헤더에 의해 식별되기 때문에 될 수 있으면 클러스터 헤더는 위치가

자주 변하지 않아야 한다. 클러스터 관리 절차는 다음과 같다.

- i. 노드 제거: 노드 X 가 클러스터 헤더로의 양방향 링크를 잃게 되거나 또는 노드가 다른 셀로 이동하여 링크가 끊어진 경우 호스트 클러스터로부터 제거된다. 두 경우 모두 클러스터 헤더와 노드 X 의 이웃 테이블은 타이머에 의해 상대방의 정보가 갱신되게 된다.
- ii. 노드 추가: 이미 존재하는 클러스터에 새로운 노드가 추가된 경우 헤더보다 더 낮은 ID 를 가지고 있어도 그 노드는 해당 클러스터의 멤버 노드가 된다. 이유는 헤더가 자주 바뀌게 되는 것을 막기 위함이다. 이 노드는 새로운 호스트 클러스터를 알게 되며 새로운 호스트 클러스터 헤더는 새로운 멤버 노드에 대한 이웃 테이블을 갱신하게 된다. 노드가 초기화될 때 헬로우 메시지의 멤버상태는 일정주기 동안 '미결정'으로 설정된다. 클러스터 헤더와 양방향 링크를 가진다는 것을 발견할 때 이 항목을 '클러스터 멤버'로 바꾼다. 어떠한 클러스터 헤더와도 양방향 링크를 가지지 않을 때 스스로 클러스터 헤더가 되며 이를 헬로우 메시지로 브로드캐스트 한다.

3.2 라우팅 고려 사항

라우팅은 소스 라우팅을 기반으로 하며 라우트 발견, 라우팅, 잘못된 라우트의 삭제 등 세 단계로 나눌 수 있다. 라우트 발견 단계 동안에 패킷 흐름의 증가를 최소화하기 위해 클러스터 구조를 도입하였다.

1) 게이트웨이 발견

클러스터 헤더는 자기 이웃한 클러스터 헤더로 가기 위한 정보를 필요로 한다. 이를 위하여 셀 내에서 다른 셀로 링크가 연결되어 있는 노드를 게이트웨이 노드라고 한다. 클러스터 X 와 클러스터 Y 에 각각 속한 두 개의 노드 사이에 링크가 설정되어 있는 경우 X 와 Y 는 게이트웨이 발견에 의해, 클러스터 X 의 헤더는 링크로 연결된 이웃 클러스터 헤더에 대한 라우팅 정보를 알게 된다. 이를 위해 인접 클러스터 테이블이 헤더에 의해 관리된다. 이 테이블은 노드에 의해 주기적으로 수신하는 헬로우 메시지에 의해 갱신된다. 헤더는 인접 클러스터 헤더와 여러 링크를 가질 수 있지만 그 중 하나만 인접 클러스터 테이블에 기록된다. 이 테이블은 노드의 클러스터 헤더에 주기적으로 보내진다. 클러스터 헤더는 자신의 인접 클러스터 테이블을 구성하는 데에 멤버 노드들이 보낸 인접 클러스터 테이블을 이용한다. 클러스터 헤더는 인접 클러스터 테이블에서의 연결관계(어느 노드로부터 어느 노드로 연결되었는지)를 알기 위해 메시지를 이웃 클러스터에 전송한다. 결과적으로 클러스터 헤더는 이웃 클러스터의 링크 관계를 정확히 알게 된다.

2) 라우트 발견

경로 발견은 목적지 D 노드로 패킷을 보내고자 하는 노드 S 에 의해 D 로의 소스 라우트를 획득하는 방법이다. 노드 D 로의 경로를 발견하기 위해 소스 노드 S 는 처음에 소스의 주소만 기록된 소스 라우트 리스트가 담긴 라우트 요청 패킷(RREQ)을 전송한다. 이 패킷을 전달하는 노드는 리스트에 자신의 주소를 추가한다. 각 노드는 하나의 라우트 요청 패킷에 대해 단 한번만 전달할 수 있다. CBRP 에서 라우트 요청 패킷은 목적지인 D 노드까지 전달 되는데 다음과 같은 형태의 라우트를 따를 것이다.

S - CH1 - G1 - G2 - CH2 - G2 - G3 - ... - D

(S: 소스, CH: 클러스터 헤더, G: 게이트웨이, D: 목적지)

소스 S는 항상 CH1이라는 클러스터 헤더에 라우트 요청 패킷을 전송한다. 각 클러스터 헤더는 소스 라우트 리스트에 나타나지 않은 이웃 노드(양방향으로 연결된)로 게이트웨이를 통해 라우트 요청 패킷을 전달한다. 이러한 절차는 목적지 노드 또는 목적지까지의 라우트를 제공할 수 있는 노드를 찾을 때까지 반복된다. 라우트 요청의 목적지인 D 노드가 패킷을 수신할 때 D 노드는 라우트 요청 패킷에 기록된 소스 라우트 리스트의 역 순서를 S로의 소스 라우트로 선택할 수 있다. 노드 D는 발견된 소스 라우트를 라우트 응답 패킷(RREP)에 복사하여 라우트를 요청한 노드 S에 전송한다. 이 소스 라우트는 소스 S가 D로 패킷을 전달하기 위해 거쳐야 하는 클러스터의 순서를 가지고 있다. 라우트 응답을 전달 하는 동안, 중간 클러스터 헤더들은 다른 클러스터에서 들어오는 링크를 나가는 링크로 패킷의 IP 헤더를 갱신한다. 또한 중간 클러스터 헤더들은 자신이 가지고 있는 클러스터 위상과 클러스터 사이의 게이트웨이 정보를 이용하여 가능한 최적의 라우트를 가지도록 라우트 응답 패킷에 기록된 소스 라우트 리스트를 변경할 수 있다. 노드가 라우트 요청 패킷이나 응답 패킷을 전달 하면서 획득한 소스 라우트는 라우트 캐시에 저장하여, 라우트 발견의 비용을 줄인다. 노드가 패킷을 보내고자 할 때 자신의 라우트 캐시를 검사하고 적당한 라우트가 캐시에 없을 때 라우트 발견 절차를 수행한다. 그리고 라우트 삭제는 소스 노드 S와 D 사이의 위상이 변하여 더 이상 이 경로를 사용할 수 없을 때 라우트는 삭제된다. S가 계속하여 D에 패킷을 전송하고 싶을 때 새로운 라우트 발견 절차를 다시 시작한다.

4. 성능평가 및 분석

본 논문에서는 time complexity와 communication complexity를 P. Krishna의 "A Cluster-Based Approach for Routing in Dynamic Network"에서 분석한 것과 같은 방법으로 worst-case performance를 여러 다른 알고리즘과 비교 분석하였다. 비교 대상 알고리즘으로는 Distributed Bellman-Ford (DBF), Ideal Link State (ILS), Diffusing Update Algorithm (DUAL), NP 그리고 Flooding등이다. 그리고 성능평가의 기준인 Time complexity는 네트워크에서 topology 변화 후 그것을 reconverge 하는데 걸리는 step 수이다. 그리고 Communication complexity는 네트워크에서 topology 변화 후 그것을 reconverge 하는데 걸리는 메시지의 수라고 정의한다. 그리고 여기서 라우팅은 complexity 자료는 [1] 을 참조하였다. Complexity의 parameter의 복잡도는 다음 파라미터에 따라 결정된다.

- N: 네트워크에 있는 노드의 수.
- E: 네트워크에 있는 링크의 수.
- d: 네트워크의 지름. 네트워크의 지름은 어떤 두 노드 사이에 longest shortest path 로 정의된다.
- D: 노드의 최대 도수(degree)
- B: 네트워크에서 유일한 경계(boundary) 노드의 최대 한계 값. 중첩된 클러스터 구조는 한 개 이상의 경계 노드를 가질 수 있다. 그렇지만 그 중 하나의 노드만이 경계 노드로 사용되며, 클러스터간 메시지를 전달하는 역할을 하며 나머지는 경계 노드가 아닌 것으로 간주된다.

- x: 토폴로지 변화에 따른 영향을 받게 되는 노드의 수
- l: 네트워크 세그먼트에 영향 받는 반지름

<표 1> topology update의 복잡도

Protocol	TC	CC
DBF	$O(N)$	$O(N^2)$
ILS	$O(d)$	$O(E)$
DUAL	$O(x)$	$O(Dx)$
NP	$O(l)$	$O(x)$
Flooding	0	0
Cluster	$O(l)$	$O(x)$

<표 2>데이터 패킷을 보내기 위한 복잡도

Protocol	TC	CC
DBF	$O(d)$	$O(d)$
ILS	$O(d)$	$O(d)$
DUAL	$O(d)$	$O(d)$
NP	$O(d)$	$O(d)$
Flooding	$O(d)$	$O(E)$
Cluster	$O(d)$	$O(d)$

5. 결론 및 향후 과제

Mobile Ad Hoc Network에서는 제한된 자원과 노드들의 이동성 때문에 적합한 프로토콜을 찾는 것이 중요하다. 따라서 많은 프로토콜이 제안되었으며, 본 논문에서는 셀룰러망에서와 같이 고정된 셀을 두고 거기에 클러스터를 구성 하여 라우트 경로의 설정과 같은 topology의 변화에도 유연한 알고리즘을 제안하였다. 그리고 고정된 셀 개념을 두어 셀룰러망에서와 같이 bandwidth의 재사용이 가능하다는 장점을 가지고 있다. 향후 과제로서 시뮬레이션을 통한 optimal한 cell의 크기를 찾고 기존에 제안된 여러 알고리즘과 라우팅 오버헤드에 관한 성능 분석을 예정하고 있다.

6. 참고문헌

- [1] P. Krishna, N.H. Vaidya, M. Chatterjee, D.K. Pradhan, "A Cluster-based Approach for Routing in Dynamic Networks," SIGCOMM Computer Communications Review (CCR), 1997
- [2] M. Jiang, J. li, and Y. C. Tay, "Cluster based routing protocol (CBRP) functional specification (Internet-Draft)," Aug.1998.
- [3] R. Silvakumar, P. Shinha, and V. Bharghavan, "Core Extraction Distributed Ad hoc Routing (CEDAR) Specification (Internet-Draft)," Oct. 1998.
- [4] S. Corson and J. Macker "Mobile ah hoc networking (manet): Routing protocol performance issues and evaluation consideration (Internet-Draft)" Mar. 1998.