

Nested Mobile Network상의 Route Optimization을 위한 MANET Protocol 적용 방안 연구

최승원*, 김상복*, 김영범**

*건국대학교 전자·정보통신공학부 대학원

e-mail : *swchoi@konkuk.ac.kr*, *sangbok@swu.ac.kr*

**건국대학교 전자공학부

ybkim@konkuk.ac.kr

A Study of method to apply MANET Protocol
for Route Optimization in Nested Mobile Network

Seung-Won Choi*, Sang-Bok Kim*, Young-Beom Kim**

*Graduate School of Electronic Eng., Konkuk University

**Dept. of Electronic Eng., Konkuk University

I. 서 론

Abstract

무선 네트워크 이동성 기술에 대한 연구가 수년간 진행되어 오면서 Mobile Network에 PAN(Personal Area Network)과 유사한 형태의 Nested Mobile Network에 대한 관심이 높아지고 있으며, 이러한 Nested Mobile Network에서의 경로최적화(Route Optimization : RO) 기술에 대한 연구가 활발하게 진행되고 있다. NEMO(Network MObility)의 RO를 위해 제안된 논문 중에 ORC(Optimized Route Cache Protocol)에 대한 제안이 있었다.[1] NEMO Basic Support가 표준안으로 채택되면서 연구 대상에서 거론되지 않고 있지만, 복잡한 이동성 기술인 Nested Mobile Network상의 RO를 위해 다시 검토해 볼 수 있을 것이다. 또한 동일 저자에 의해 제안된 Nested Mobile Network 내부에 Ad-hoc Routing 알고리즘인 OLSR(Optimized Link State Routing Protocol)을 적용한 제안이 발표되었다.[2]

본 논문에서는 ORC와 Nested Mobile Network상의 OLSR Scheme을 적용하여 RO를 위한 방안을 제안하고자 한다.

무선 네트워크 이동성 기술에 대한 연구가 수년간 진행되어 오면서, Mobile Network의 다양한 모델들이 연구 대상으로 주목 받고 있으며, 그 중 PAN(Personal Area Network)과 유사한 형태의 Nested Mobile Network에 대한 관심이 높아지고 있다. 이러한 Nested Mobile Network에서의 주된 이슈 중 하나가 RO에 대한 연구이며, Mobile Network가 Nested Mobile Network로 구성되어 있을 경우 내부 Mobile Node들과 통신하는 CN(Correspondent Node)들 간의 연속적인 패킷 전송을 위해 RO를 할 수 있도록 한 RRH, RBU+와 같은 제안이 발표되었다.

일반적인 NEMO를 지원하는 ORC(Optimized Route Cache Protocol)와 같은 제안이 있었다. NEMO Basic Support가 표준안으로 채택되면서 연구대상에서 거론되지 않고 있지만, 복잡한 이동성 기술인 Nested Mobile Network상의 RO를 위해 다시 검토해 볼 수 있을 것이다.

최근 동일 저자에 의해 제안된 Nested Mobile Network 내부에 Ad-hoc Routing 알고리즘인 OLSR을 적용한 제안이 발표됨으로서 MANET을 이용하여 RO에 대한 새로운 방향 제시를 시도했다고 할 수 있다.

본 논문에서는 ORC Protocol과 Nested Mobile

Network상에 OLSR Scheme을 적용하여 RO를 위한 방안을 제안하고자 한다.

본 논문은 4장으로 구성되었고, 2장에서는 관련연구로 ORC Protocol과 Nested Mobile Network 내부에 OLSR을 적용한 제안을 설명하였으며, 3장에서는 개선방안으로 제안 모델에 대해 설명하였으며, 4장에서 결론을 맺고자 한다.

II. 관련 연구

2.1 ORC(Optimized Route Cache) Protocol

그림 1은 ORC Protocol의 동작원리를 보여주고 있다.

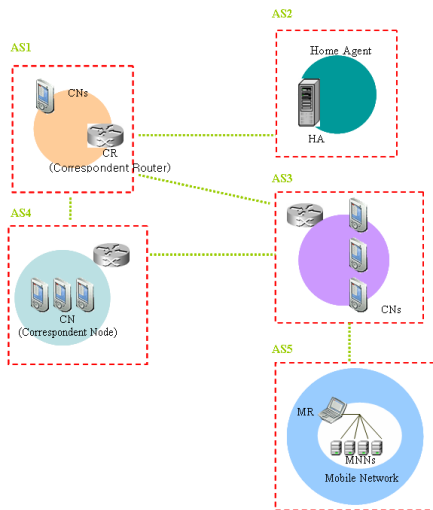


그림 1. ORC Protocol의 동작원리

5개의 AS(Autonomous System)는 BGP(Border Gateway Protocol)에 의해서 연결되어 있으며, Mobile Router(MR)가 이동될 때마다 CR(Correspondent Router)은 Mobile Router로부터 BU(Binding Update) Notification Message를 받거나 BU Message를 Binding Cache에 Cache Entry로 저장한다. BU Message 전송 후, CRs은 Mobile Network로 전송되는 packet을 가로채고 Binding Cache Entry에 있는 MN(Mobile Node)의 CoA(Care-of-Address)정보에 의해 MN의 상위MR의 CoA로 tunneling한다. 모든 CR은 IGP(Interior Gateway Protocol) 또는 EGP(Exterior Gateway Protocol)와 관계없이 Routing Protocol에 의해 Mobile Network로 향하는 packet을 intercept하기 위해 AS에 존재하는 CR들에게 Anycast Packet을 사용하여 전송한다. CR은 자신이 전송 받은 Binding

Information이 유효할 때 자신의 Binding Cache Entry에 의한 Proxy Route를 다른 CR들에 전송한다. Binding Cache Entry 삭제 후에, CR은 routing table에서 proxy route를 제거한다. Proxy route는 Binding Cache 정보는 아니며, CN의 목적지 MN에 대한 prefix와 MR의 CoA를 포함하고 있다.

2.2 Nested Mobile Network with OLSR

OLSR Routing Protocol은 HNA(Host & Network Association)-Message를 사용하여 MRs 간에 교환된 Prefix Information를 이용해 Nested Mobile Network 내부 MNs 간에 직접 연결이 가능하도록 한다. 이러한 Routing Protocol의 적용은 Mobile Network 내부의 BU 발생 횟수를 줄여 Bandwidth가 한정되어 있는 Wireless Medium에서 불필요한 Traffic을 감소시키며, 내부 MNs 간의 데이터 전송 시 경로를 줄일 수 있다. 이와 같이 Nested Mobile Network에 Ad-hoc Routing Protocol인 OLSR을 적용한 이유는 동적 환경에서의 RO가 뛰어나고 BU 발생 횟수를 줄일 수 있기 때문이다.

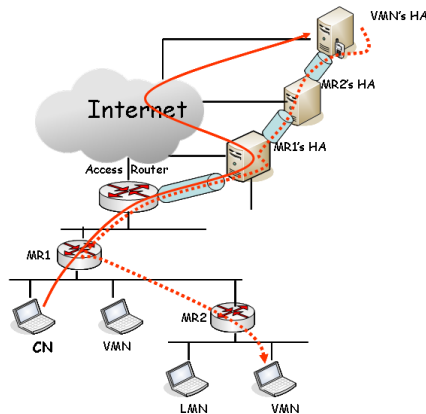
MIPv6[3]에 의하면 그림 2와 같이 Nested Mobile Network상의 서로 다른 MR에 연결되어 있는 CN과 VMN(Visited Mobile Node)이 Nested Mobile Network 상에서 통신을 하기 위해서는 NEMO 외부에 존재하는 각 HA(Home Agent)의 터널을 이용해 통신하게 된다. 이러한 복잡한 경로를 이용할 때, 패킷 전송의 전송 지연을 방지하기 위해 내부 Network Nodes간의 패킷 전송은 내부에 OLSR Routing Protocol을 적용하게 된 것이다.

Nested Mobile Networks에 Ad-hoc Routing Protocol인 OLSR을 적용한 방식에는 Nested NEMO 내부에 적용한 경우와 Nested NEMO 외부에 적용한 경우로 나누어진다.

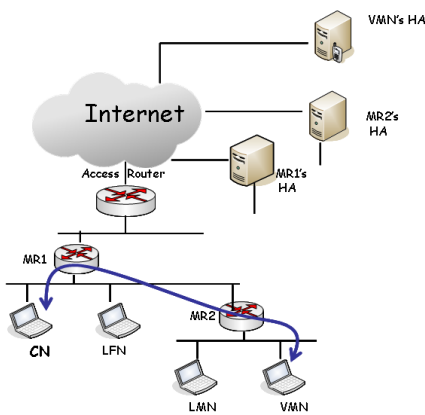
2.2.1 OLSR networks in Nested NEMO

Nested NEMO Network에서 MRs은 OLSR을 이용함으로써 Ad-hoc network과 같이 구성이 가능하다. OLSR은 Light-Weight Topology Discovery(TC)와 HNA 메시지를 이용하여 MRs이 Nested NEMO 안에서 Ad-hoc Network과 에서와 마찬가지로 최적의 경로를 제공한다.

OLSR Routing Protocol은 HNA Message를 이용하여 Mobile Network Node들의 Network Prefix가 주기적으로 변함을 알 수 있으며, TC Message를 이용하여 Topology 정보를 갱신한다.



(a) Nested NEMO에 OLSR을 적용하기 전



(b) Nested NEMO에 OLSR을 적용한 후
그림 2. Nested NEMO 내부 CN과 VMN간의 통신

MRs는 TC Message 안에 Multipoint Relay(MPR)을 포함하여 Broadcast하고, MR이 아닌 Mobile Network nodes에 대한 Prefix Information은 HNA Messages를 통해 Broadcast 한다.

같은 Nested NEMO 안에서의 MN들은 OLSR에 의해 MR들 간의 통신으로 직접 이루어진다. AR은 OLSR 기능이 꼭 필요하진 않지만, 인터넷쪽으로부터 OLSR이 가능한 노드로 데이터가 도착했을 때, Top Level Mobile Router(TLMR)을 통해 전송하고자 하는 Node의 경로를 정하는 Gateway 역할을 하게 된다.

2.2.2. Nested NEMO 밖에서의 Tunnel Optimization

Internet에 있는 Node 즉 CN과 Nested NEMO Network에 있는 MN과 통신하는데 있어서 불필요한 Encapsulation과 Dog-Legged Routing 또는 Pinball Routing을 피하기 위해서 몇 가지 방법이 제시되었다.

현재 이와 같은 문제의 해결방법으로 제시되는 것 중 하나는 RO를 통해 필요한 Tunneling 수를 줄이고자 Nested NEMO에서 Nesting Level 깊이에 따른 HA와의 Tunnel을 형성하고, BU Message 전송을 통

해 Tunnel을 줄이는 제안 등으로 접근하고 있다.

Nested NEMO에 Ad-hoc Routing을 적용한 이 제안에서는, Internet에 있는 HA가 Encapsulation 해서 보낼 때, AR(or TLMR)이 Encapsulation Header 정보를 가지고 목적지인 MN으로 경로 설정을 통해 바로 전송이 가능하게 하는 것이다. 그렇지만, Nested NEMO Network안에 있는 MR들이 OLSR Routing Protocol을 사용하여 Network을 형성하면서 Nested NEMO Network 안에 Correct Routing을 하는데 있어서 Encapsulation Information이 더 이상 필요하지 않게 된다.

이와 같이, OLSR Protocol을 Nested NEMO Network 안에 적용시킴으로써, 현재 Basic NEMO에서 사용되는 Binding Signaling을 간단하게 이용해 RO문제를 효과적으로 줄일 수 있다.

III. 제안모델

기존의 제안된 모델들의 단점을 해결하기 위해서 II장에서 정리한 ORC와 OLSR Routing Protocol을 같이 적용한 모델을 제안하고자 한다.

그림 3은 제안 모델의 구성도이며, 기존의 ORC 모델은 MN 또는 MR의 BU Message에 의해 CR의 Binding Cache Information을 유지하였다면, 제안 모델은 내부에 OLSR Routing 정보에 의해 내부 경로에 대한 정보를 유지한다.

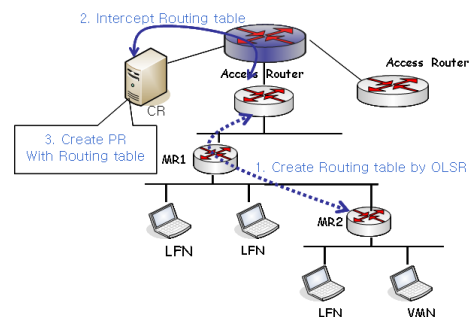


그림 3. 제안 모델의 구성도

[1]에 의하면, CR은 Mobile Network으로 향하는 Packet을 가로채기 위해서 Proxy Route를 전파하는데, Proxy Route에는 목적지의 Mobile Prefix와 다음 Hop의 주소를 포함하고 있다. 이와 같이 자신이 속한 도메인인 AS내에 존재하는 AR을 Mirroring 하고, 패킷을 Intercept하여 해당 패킷의 헤더에 Encapsulation 헤더를 추가하여 전송하는 역할을 한다.

이와 같이 CR은 AR의 Packet을 가로채어 재전송하

는 역할과 AR을 모니터링 하는 역할 등을 수행하는 기능을 가지고 있기 때문에 제안하고자 하는 모델에서의 AR에 존재하는 Routing Table을 자신의 Proxy Route로 업데이트 하는 기능을 실행할 수 있도록 할 수 있다.

그림 3에 도식된 그림에 의한 제안 모델이 CR에서 Proxy Router를 생성하는 과정은 다음과 같다.

1. 내부 MRs은 자신의 Routing Protocol인 OLSR에 의해 Routing Table을 생성한다.
2. CR은 자신이 Mirroring하고 있는 AR에 있는 Routing Table을 주기적으로 Intercept 한다.
3. CR은 해당 정보에 의해 Proxy Route를 생성한다.
4. Correspondent Router Discovery에 의해 anycast로 요청되어지는 "Correspondent Router Discovery and Reply"에 대한 응답메세지로 정보를 전송하여 CR 간의 정보교환을 수행하게 된다.

이와 같은 CR의 동작으로 다른 도메인 내에 존재하는 CR들과 해당 정보를 공유하는 Delegator역할이 CR에 의해 이루어지게 된다.

CR과 내부의 OLSR에 의한 Routing Table의 생성으로 MN 또는 MR의 이동시 이들 MN 또는 MR은 BU과정 없이 통신을 할 수 있으며, HA의 터널생성과 Pinball Routing 문제를 해결 할 수 있게 된다.

IV. 결론

본 논문에서는 Nested Mobile Network을 지원하기 위한 Route Optimization 방안을 제안하기 위해 ORC와 Ad-hoc Routing Protocol인 OLSR이 적용된 Nested NEMO 환경을 결합하는 방안을 제안하였다. Nested NEMO 환경을 구현하는데 있어 궁극적인 문제는 Pinball Routing 문제를 해결하는데 있기 때문에 이 두 가지 Scheme에 대한 결합으로 인한 새로운 모델의 제안은 검토할 수 있는 것이다. 본 논문에 기존 모델과의 비교를 통한 성능평가 부분을 수행하지 못한 부분에 대해서는 다음 연구를 통해 좀 더 상세히 연구되어야 할 것이다. 또한 ORC Router가 제안된 기능을 수행하는데 있어 발생될 수 있는 문제점도 다음 연구를 통해 분석되어야 할 것이다.

참고문헌

[1] Ryuji Wakikawa, "Optimized Route Cache Protocol (ORC)", Internet Draft, draft-wakikawa-nemo-orc-01, Oct 2004.

[2] Ryuji Wakikawa, "ROUTE OPTIMIZATION IN NESTED MOBILE NETWORKS (NEMO) USING OLSR ", NCS 2005.

[3] D. Johnson, C. Perkins, "Mobility support in IPv6", RFC 3775, June 2004.

[4] T. Clausen, P. Jacquet, "Optimized Link State Routing Protocol," RFC 3626, <http://ietf.org/rfc/rfc3626.txt>, 2003

[5] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[6] Ryuji Wakikawa, Susumu Koshiba, Keisuke Uehara, Jun Murai, "ORC: Optimized Route Cache Management Protocol for Network Mobility." Proc. of ICT2003, Nov 2003.

[7] Thierry Ernst, "Network mobility in IPv6", PhD Thesis, University Joseph Fourier Grenoble, Francee, available at <http://www.inria.fr/rrrt/tu-0714.html>, October 2001.

[8] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. "Nemo Basic Support Protocol" (work in progress). Internet Draft (draft-ietf-nemo-basic-support-02), Internet Engineering Task Force, December 2003.