

# PMIPv6 네트워크에서 비용효과적인 도메인 간의 분산 이동성 제어기법

장 순 호<sup>†</sup> · 정 종 필<sup>††</sup>

요 약

PMIPv6(Proxy Mobile IPv6)는 MN(Mobile Node)의 이동성을 지원하기 위한 망 기반의 이동성관리 기술이다. PMIPv6의 구성 요소는 네트워크 기반 이동성 지원 프로토콜로서 도메인을 관리하는 LMA(Local Mobility Anchor)와 MN의 이동을 감지하고 MN의 위치 정보를 등록하는 MAG(Mobile Access Gateway) 그리고 MN으로 구성되어 있다. PMIPv6는 LMA와 MAG 사이의 터널을 형성하고 패킷이 LMA를 통하여 전달되도록 설계하여, LMA 병목현상 및 종단 간 지연이 증가하는 문제점이 있다. 이러한 문제점을 해결하기 위해 경로 최적화 수행 가능 감지, 경로 최적화 절차 등의 많은 연구가 진행되고 있으나 추가적인 시그널링으로 인한 오버헤드가 증가하고 다중 LMA 환경에 적용하기에 어려움이 있다. 따라서 본 논문에서는 다중 LMA 환경을 고려한 PMIPv6 기반의 향상된 경로 최적화 방안을 제안한다. PMIPv6 도메인의 모든 LMA는 PIG(Proxy Internetworking Gateway)에 연결되어 PMIPv6 도메인 간 분산 이동성 제어를 수행한다. PMIPv6 도메인 내 모든 LMA의 정보를 MAG가 유지하도록 하여 신속하게 경로 최적화를 수행하도록 하였으며, LMA에 경로 최적화 상태 정보를 PIG로부터 수신 받아 저장하여 추가적인 시그널링없이 경로 최적화를 지원하도록 한다.

키워드 : 모바일네트워크, 분산 이동성 제어, Proxy Mobile IPv6, PIG

## Cost-Effective Inter-LMA Domain Distributed Mobility Control Scheme in PMIPv6 Networks

Soonho Jang<sup>†</sup> · Jongpil Jeong<sup>††</sup>

ABSTRACT

Proxy Mobile IPv6 (PMIPv6) is designed to provide network-based mobility management support to an MN without any involvement of the MN in the mobility related signalling, hence, the proxy mobility entity performs all related signalling on behalf of the MN. The new principal functional entities of PMIPv6 are the local mobility anchor (LMA) and the mobile access gateway (MAG). In PMIPv6, all the data traffic sent from the MN gets routed to the LMA through a tunnel between the LMA and the MAG, but it still has the single point of failure (SPOF) and bottleneck state of traffic. To solve these problems, various approaches directed towards PMIPv6 performance improvements such as route optimization proposed. But these approaches add additional signalling to support MN's mobility, which incurs extra network overhead and still has difficult to apply to multiple-LMA networks. In this paper, we propose a improved route optimization in PMIPv6-based multiple-LMA networks. All LMA connected to the proxy internetworking gateway (PIG), which performs inter-domain distributed mobility control. And, each MAG keeps the information of all LMA in PMIPv6 domain, so it is possible to perform fast route optimization. Therefore, it supports route optimization without any additional signalling because the LMA receives the state information of route optimization from PIG.

Keywords : Mobile Networks, Distributed Mobility Control, Proxy Mobile IPv6, PIG

### 1. 서 론

최근 스마트폰 등 다양한 모바일 기기가 등장하고 페이스북, 트위터 등의 소셜 네트워크 서비스의 등장과 카카오톡 등 각종 모바일 앱 등이 등장함에 따라 모바일 인터넷 트래픽이 날로 급증하고 있다. 보고서에 의하면 2014년이 지나면 약 16억의 모바일 인터넷 사용자를 예상하고 있다[1]. 더

※ 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2011-0027030).

† 정 회 원 : 성균관대학교 정보통신대학원 석사과정

†† 정 회 원 : 성균관대학교 정보통신공학부 공학박사(교신저자)

논문접수 : 2012년 1월 2일

수정일 : 1차 2012년 3월 16일

심사완료 : 2012년 3월 16일

육이 모바일 기반의 멀티미디어 앱의 활성화로 인해 모바일 인터넷 트래픽은 더욱 부하가 가중될 것이다. 이러한 모바일 트래픽을 효율적으로 처리하기 위한 이동성 제어 기술에 대한 수요가 요구되고 있다.

현재의 인터넷 이동성 제어 기술은 계층적 망구조를 기반으로 하는 집중형(centralized) 방식의 특징을 지니고 있는데, 이러한 방식으로는 급격히 증가하는 모바일 인터넷 트래픽 수요를 대처하기가 어렵다. 이를 위한 대안기술로서 분산형(distributed) 이동성 제어기법이 IETF(Internet Engineering Task Force) 표준화기구에서 논의되고 있다. PMIPv6(Proxy Mobile IPv6)은 MN(Mobile Node)를 대신하여 액세스 망에서 MN의 이동성을 지원하기 위한 네트워크-기반 이동성관리 프로토콜이다[2]. 기존의 PMIPv6 프로토콜에서는 MN이 통신하고자하는 CN(Correspondent Node)와 동일한 MAG(Mobile Access Gateway)내에 있더라도 데이터 패킷은 MN의 LMA(Local Mobility Anchor)를 통과하여 전송되기 때문에 패킷전송 비용이 증가하는 문제가 발생한다. 기존의 분산 최적화 기법은 동일한 PMIPv6 도메인 내에서 분산 이동성 제어 연구에 한정되어 있어 CN이 같은 도메인에 있는 경우만을 고려하는 한계점이 있었다.

<표 1> 약어 표

약어	의미
AAA	Authentication, Authorization, Accounting
BCE	Binding Cache Entry
BU	Binding Update
CN	Correspondent Node
CoA	Care of Address
DD-PMIP	Data-driven Distributed PMIP
HA	Home Agent
HNP	Home Network Prefix
HoA	Home Address
IETF	Internet Engineering Task Force
LBA	LMA Binding Ack
LBQ	LMA Binding Query
LBU	LMA Binding Update
LIT	LMA Information Table
LMA	Local Mobility Anchor
LRI	Local Routing Optimization Indication
MAG	Mobile Access Gateway
MIPv6	Mobile IPv6
MN-ID	MN-Identifier
MN	Mobile Node
NAI	Network Access Identifier
NetLMM	Network-based Local Mobility Management
PBA	Proxy Binding Ack
PBQ	Proxy Binding Query
PBU	Proxy Binding Update
PMIPv6	Proxy Mobile IPv6
PIG	Proxy Internetworking Gateway
PQA	Proxy Query Ack
PS	Policy Store
RA	Router Advertisement
S-PMIP	Signal-driven PMIP
SD-PMIP	Signal-driven Distributed PMIP
SMA	Session Mobility Anchor
TAP	Topological Anchor Point

본 논문에서는 PMIPv6 도메인을 관리하는 PIG(Proxy Internetworking Gateway)를 별도로 두어 도메인 간의 이동성 제어를 관리할 수 있도록 하였다. PMIPv6 도메인을 관리하는 LMA와 PIG간 비용에 PMIPv6 도메인 내에서의 이동성 비용을 합산하여 기존의 단순한 PMIPv6 도메인 내에서의 분산 이동성 제어가 아니라 PMIPv6 도메인 간의 분산 이동성 제어에 최적화됨을 보여준다.

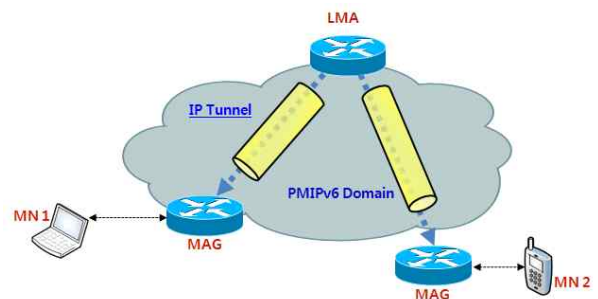
본 논문의 구성은 다음과 같다. 2장에서는 현재 도메인 간 연동기법에 대해서 설명하고, 3장에서는 도메인 내에서의 분산제어를 확대하여 도메인 간 분산제어 기법을 제안하고 성능분석을 수행하고, 제안 기법의 동작 및 절차에 대해서 설명한다. 4장에서는 제안기법의 성능을 분석하고, 5장에서는 결론을 맺는다.

## 2. 관련 연구

### 2.1 PMIPv6 도메인 내의 분산처리 기법

#### 2.1.1 PMIPv6의 구성 및 기본동작

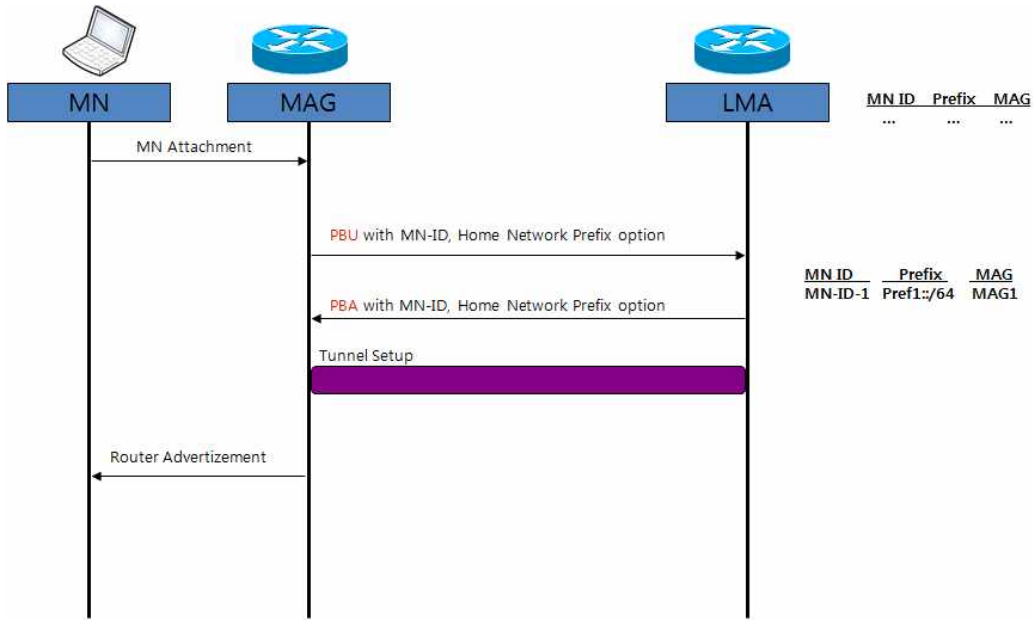
PMIPv6의 도메인 구조는 (그림 1)에서와 같이 크게 LMA와 MAG, MN으로 구성된다. LMA는 PMIPv6에서 MN에 대한 일종의 HA(Home Agent)역할을 하며 세부적으로는 MN에게 HNP(Home Network Prefix)를 할당하고 자신이 관리하는 도메인 내의 모든 MN들의 주소와 위치정보를 유지하여 연결을 보장하는 역할을 한다. MAG는 MN을 대신해서 LMA와의 터널링을 통해 MN의 이동성을 지원하기 위한 시그널링을 수행하는 역할을 하고 MN의 네트워크에 대한 연결 기능과 라우팅 기능을 담당한다. PMIPv6 도메인 내에서 MN은 NAI(Network Access Identifier) 같은 MN-ID(MN-Identifier)로 식별되고, 일단 MN이 성공적으로 접속 인증을 수행하게 되면 네트워크는 MN이 항상 홈 네트워크에 있는 것처럼 만들어 준다. MN이 PMIPv6 도메인 내의 어느 곳으로 이동하던지 항상 동일한 HNP를 할당받기 때문에 MN은 PMIPv6 도메인을 하나의 링크로 간주하게 되는 것이다.



(그림 1) PMIPv6 도메인의 기본 구조

#### 2.1.2 PMIPv6의 초기화 과정

PMIPv6의 초기화 과정은 (그림 2)와 같다. MN이 MAG의 링크에 접속하면, MAG는 L2핸드오버 과정에서 접속한 MN

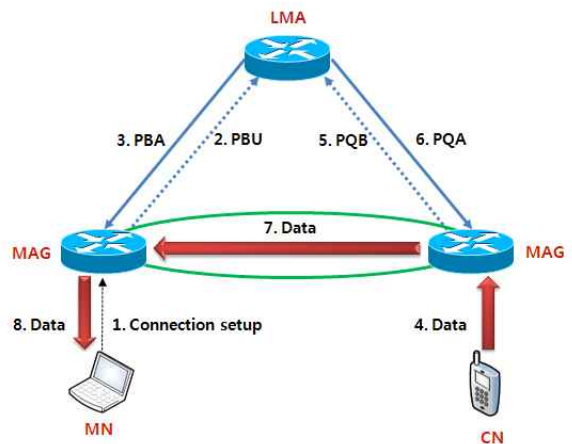


(그림 2) PMIPv6 핸드오버 절차

의 MN-ID를 알게 된다. MAG는 MN-ID를 이용해 인증을 수행하며, 이때 MN-ID에 해당하는 Policy Profile를 PS(Policy Store)로부터 받게 된다. MAG는 Profile에 포함된 LMA의 주소로 MN-ID를 PBU(Proxy Binding Update) 메시지에 담아서 전송한다. PBU 메시지를 수신한 LMA는 BCE(Binding Cache Entry)에 해당 MN에 대한 정보가 있는지 확인한 다음 없으면 NM의 정보를 생성하고, 있을 경우에는 기존의 정보를 갱신한다. 그 다음 LMA는 해당 MN에 대한 HNP를 PBA(Proxy Binding Ack) 메시지에 담아서 MAG로 보낸다. 이 때, LMA와 MAG 사이에는 양방향 터널이 생성되게 된다. MAG는 PBA를 받자마자 MN-HNP 정보를 추출하여 RA(Router Advertisement) 메시지에 담아서 MN에게 보낸다. RA 메시지를 받은 MN은 Policy Profile에서 정의된 주소 설정 방법에 따라 RA 메시지에 포함된 MN-HNP를 이용하여 자신의 HoA(Home Address)를 생성한다. MN은 이제 생성된 HoA를 사용하여 CN과 통신할 수 있다. 이 때, MN에서 보내지거나 MN을 향하는 모든 패킷은 LMA를 거치게 된다. LMA가 MN의 HNP에 대한 TAP(Topological Anchor Point) 이기 때문에 외부에서 MN을 향하는 모든 패킷은 LMA에 도착하게 되고 LMA는 이것을 MAG와의 사이에 설립된 양방향 터널을 통해 MN으로 포워딩한다. MN에서 보내지는 패킷 역시 MAG가 LMA로 터널링 한 뒤, LMA가 외부 헤더를 제거하고 CN으로 전송한다.

2.1.3 S-PMIP( Signal-driven PMIP )

S-PMIP[3]는 데이터 영역으로부터 제어 영역이 분리된 부분적으로 분산된 구조이다. (그림 3)은 S-PMIP 기법의 동작절차를 보여준다. PBU 동작은 PMIPv6에서 수행된 것과 동일하게 MAG와 LMA 사이에서 수행된다. 그러나 패킷 전달 동작에서 CN의 MAG는 데이터를 전달하기 전에



(그림 3) S-PMIP 동작

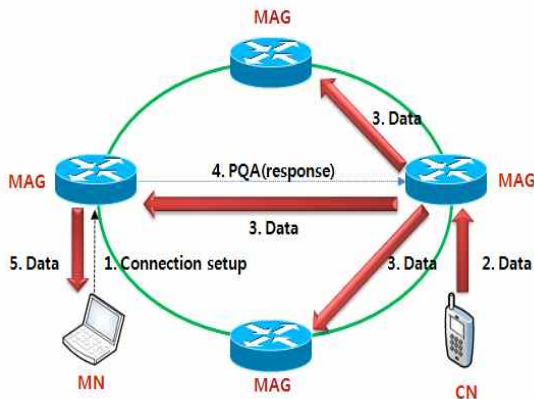
LMA에 연결된 MN의 CoA(Care of Address)를 우선 찾는다. 이렇게 해서 CN의 MAG는 최근 정의된 PBQ(Proxy Binding Query) 명령을 LMA에 보내고, LMA는 최근 정의된 PQA(Proxy Query Ack) 메시지를 MAG에 보낸다. 이렇게 새롭게 정의된 PBQ와 PQA 메시지는 다음 섹션을 지정하게 된다. 결국, CN의 MAG는 MN의 MAG와 더 나가 MN에 데이터를 직접 전달하게 된다.

S-PMIP의 동작 절차는 (그림 3)과 같이 MN이 기동되면 MAG를 통해 PMIPv6에 연결되고 HoA를 얻는다. MAG는 PBU 메시지를 LMA로 보내 MN의 CoA와 HoA를 연결한다. PBU 요청을 수신하면, LMA는 BCE와 관련된 정보를 생성하고 PBA 메시지를 MAG로 보낸다. CN은 MN에 데이터 패킷을 보낸다. CN의 MAG는 PBQ 메시지를 MN의 CoA를 찾는 LMA에 보낸다. PBQ 메시지를 수신한 LMA는 MN의 CoA를 포함한 PQA 메시지를 CN의 MAG에 보낸다.

이 처리과정동안 CN의 MAG는 데이터 손실을 막기 위해서 CN으로부터 수신된 데이터 패킷을 저장한다. 결국 터널이 구축되면, CN의 MAG는 저장된 데이터 패킷을 먼저 보낸다. 결국 CN의 MAG는 MN의 MAG에 데이터 패킷을 보낸다. 마지막으로 데이터 패킷은 MN에 전달된다.

2.1.4 DD-PMIP( Data-driven Distributed PMIP )

DD-PMIP[3] 완전 분산 처리된 데이터 기반 멀티캐스트/브로드캐스트 방식이다. 이 기술에서는 LMA와 바인딩 업데이트가 사용되지 않는다. CN의 MAG는 바인딩 쿼리 작업을 사용하지 않고, 멀티캐스트 하여 도메인의 모든 MAG에 데이터 패킷을 보낸다.



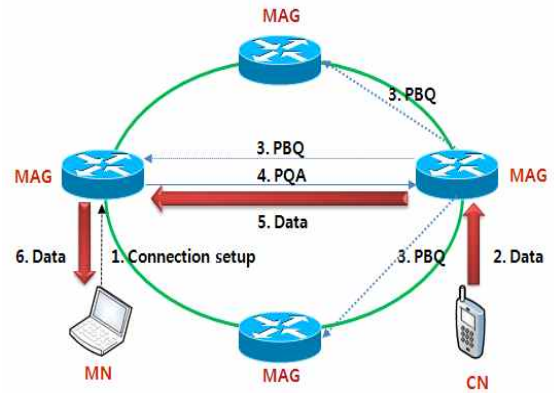
(그림 4) DD-PMIP 동작

DD-PMIP의 동작 절차는 (그림 4)와 같이 MN이 기동되면 MAG를 통해 PMIPv6에 연결되고 HoA를 얻는다. CN은 MAG에 데이터 패킷을 보낸다. CN의 MAG는 도메인 내 모든 MAG에 브로드캐스트로 데이터 패킷을 보낸다. 데이터 패킷을 수신한 MN의 MAG는 PQA 메시지를 CN의 MAG에 보낸다. 마지막으로 데이터 패킷은 MN에 전달된다.

2.1.5 SD-PMIP( Signal-driven Distributed PMIP )

SD-PMIP[3]은 완전히 분산된 기술로 P2P 검색 기술과 유사하다. SD-PMIP의 동작은 MN이 기동되면 MAG에 연결된다. CN은 MN에 데이터 패킷을 보낸다. MN의 위치를 찾고, CN의 MAG는 PBQ 메시지를 도메인 내 모든 MAG에 멀티캐스트 방식으로 보낸다. 멀티캐스트 전송은 미리 정의된 멀티캐스트 주소를 통해 수행된다. MN의 MAG만 유일하게 PQA 메시지로 CN의 MAG에 응답한다. 도메인 내 모든 다른 MAG는 PBQ 메시지를 무시한다. MN의 MAG에 데이터 패킷이 전송되고 MN에 데이터가 전송된다.

SD-PMIP의 동작 절차는 (그림 5)와 같이 MN이 기동되면 MAG를 통해 PMIPv6에 연결되고 HoA를 얻는다. CN은 MAG에 데이터 패킷을 보낸다. CN의 MAG는 도메인 내 모든 MAG에 멀티캐스트로 PBQ 제어신호를 보낸다. 데이터 패킷을 수신한 MN의 MAG는 PQA 메시지를 CN의 MAG에 보낸다. CN의 MAG는 MN의 MAG로 데이터 패킷을 전송하고, 마지막으로 데이터 패킷은 MN에 전달된다.

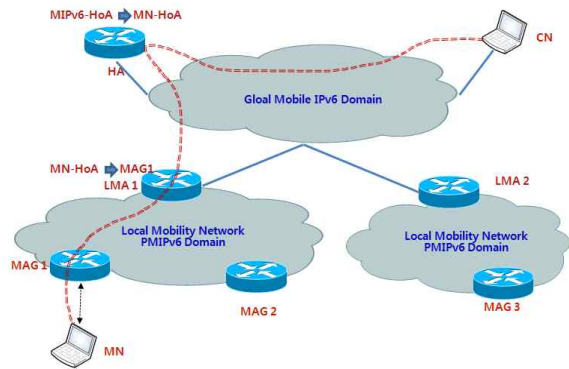


(그림 5) SD-PMIP 동작

2.2 PMIPv6 도메인 간 경로최적화 기법

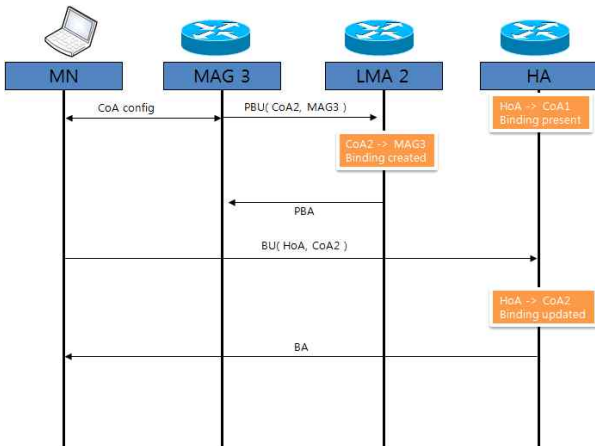
2.1 절에서 도메인 내에서 경로최적화 기법에 대해서 언급하였다. S-PMIP 기법의 경우 모든 제어신호가 LMA를 통해야 하기 때문에 과도한 시그널에 의해 네트워크의 트래픽이 증가한다. DD-PMIP의 경우는 브로드캐스트 방식으로 PMIPv6 도메인 내 모든 MAG에 데이터를 전송하는 기법으로 불필요한 데이터 전송으로 인해 네트워크 내 트래픽이 증가하는 문제가 있다. SD-PMIP의 경우 앞의 두 가지 기법의 문제점을 개선한 기법으로 PMIPv6 도메인 내에서 효율적인 분산제어가 가능하나 도메인 내로 국한된다. 2.2 절에서는 도메인간의 경로 연동 기법에 대해서 살펴본다.

2.2.1 PMIPv6-MIPv6 연동기법



(그림 6) PMIPv6-MIPv6 연동

[4]에서는 PMIPv6와 MIPv6의 연동방안을 연구하고 있다. PMIPv6 도메인 내의 지역 이동성 지원을 위해 PMIPv6를 사용하고, 도메인 외부로의 전역 이동성을 위해서는 MIPv6를 사용하는 방안으로 (그림 6)과 같다. PMIPv6에서 관리되는 주소 MN-HoA는 HA에 CoA로 등록된다. 이것은 HA가 MN-HoA를 가리키는 MIPv6-HoA에 대한 BCE를 가지고 있다는 것을 의미한다. MN으로 보내지는 패킷들은 이 BCE를 이용해 PMIPv6 도메인의 LMA로 보내지고, MAG를 거쳐 MN에게 전송된다. MN이 PMIPv6 도메인 내에서 이동하는 경우 기존 PMIPv6의 핸드오버 절차와 동일하다.

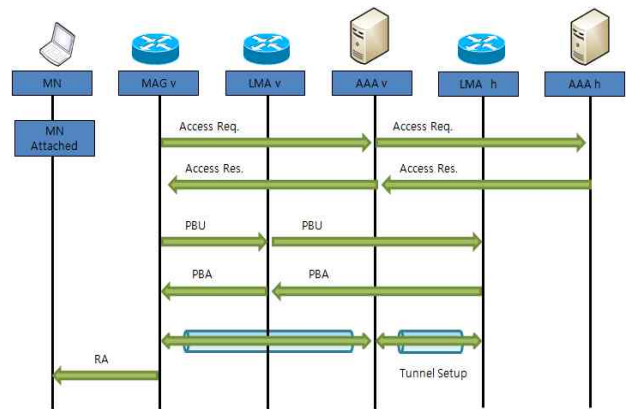


(그림 7) MIPv6와 PMIPv6 핸드오버 절차

MN은 같은 MN-HNP를 사용하기 때문에, HA에게 바인딩 업데이트를 위한 BU(Binding Update)를 보낼 필요가 없다.

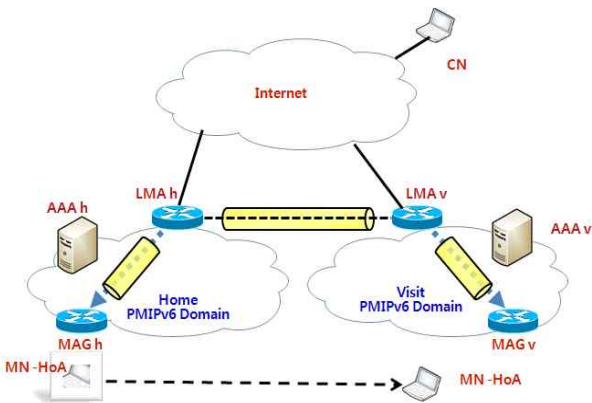
하지만 MN이 MAG1에서 MAG3으로 이동하게 되면, 새로운 PMIPv6 도메인으로의 이동이기 때문에 (그림 7)과 같이 초기화 과정을 수행해서 LMA2로부터 새로운 MN-HNP를 할당받아야 한다. MN은 이를 이용해 생성한 새로운 CoA를 BU에 담아 HA에게 전송하고, HA는 바인딩 업데이트를 수행한다. PMIPv6-MIPv6 연동을 통해 PMIPv6 도메인 간 핸드오버를 지원하는 이 방법은 MN이 MIPv6(Mobile IPv6) 프로토콜 스택을 반드시 가져야 한다는 문제점이 있다. 또한 MAG와 MN 사이에서도 캡슐화된 패킷이 전송되기 때문에 무선 구간을 효율적으로 활용하지 못하게 된다.

AAAv는 수신한 MN-ID에 해당하는 MN의 정보가 없기 때문에, MN이 이동전에 위치하던 Home PMIPv6 도메인의 AAAh에게 AAA인증과 Policy Profile을 요청한다. 이때, AAAv는 MN-ID로 사용되는 NAI(Network Access Identifier)를 보고 MN이 속해 있던 도메인과 그 도메인의 AAA 서버의 주소를 알아야만 한다. AAAv로부터 요청 메시지를 받은 AAAh는 AAA 인증을 수행한 뒤, MN의 Policy Profile과 LMAh의 주소를 AAAv에게 전송한다. LMAh의 주소가 필요한 이유는 MN이 홈 도메인에서 사용하던 MN-HNP를 방문 도메인에서도 계속 사용할 수 있도록 하기 위함이다. AAAv는 AAAh로부터의 메시지에 LMAv의 주소를 추가하여 MAGv에게 전송한다. 이를 수신한 MAGv는 LMAv에게 MN-ID와 LMAh의 주소를 담은 PBU 메시지를 전송한다.



(그림 9) J-H Na 기법의 핸드오버 절차

2.2.2 J-H Na 기법



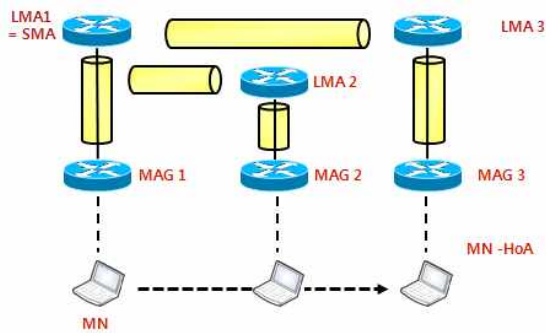
(그림 8) J-H Na 기법

Na의 기법[5]에서는 LMA와 MAG, AAA(Authentication, Authorization, Accounting) 서버가 PMIPv6 도메인 간 핸드오버를 위해 추가적인 역할을 수행하게 된다. Na의 기법의 구성은 (그림 8)과 같다. MN이 PMIPv6 홈도메인에서 PMIPv6 방문도메인으로 이동하여 MAGv에 연결되면, MAGv는 AAAv에게 MN-ID를 보내 AAA 인증을 수행하고, 해당 MN의 Policy Profile을 받고자 한다. 하지만,

PBU 메시지를 받은 LMAv는 다시 MN-ID를 포함한 PBU 메시지를 LMAh에게 보낸다. LMAh는 PBU 메시지를 받은 다음, MN을 위해 이전에 설립되었던 터널을 제거하고 LMAv로 새로운 터널을 설립한다. 또한 LMAv에게 MN-HNP를 담은 PBA 메시지를 전송한다. PBA 메시지를 받은 LMAv는 MN-HNP를 등록하고, 다시 MN-HNP를 PBA 메시지에 담아 MAGv에게 보낸다. 이로써, PMIPv6 도메인 간 핸드오버를 위해 LMAh와 LMAv사이, LMAv와 MAGv 사이에 연쇄된 양방향 터널이 생성된다. PBA 메시지를 담은 MAGv는 MN-HNP를 RA 메시지에 담아 MN에게 전송한다. 이 MN-HNP는 MN이 PMIPv6 홈 도메인에서 LMAh로부터 할당받았던 것과 동일한 것이기 때문에 MN은 여전히 같은 네트워크에 있다고 간주하게 되고, PMIPv6 도메인 간 핸드오버가 완료된다. 이 기법은 MN에 대한 AAA 인증과 Policy Profile을 받기 위해 AAAh로부터의 응답을 기다려야 하기 때문에 긴 핸드오버 지연이 발생하는 문제점이 있다.

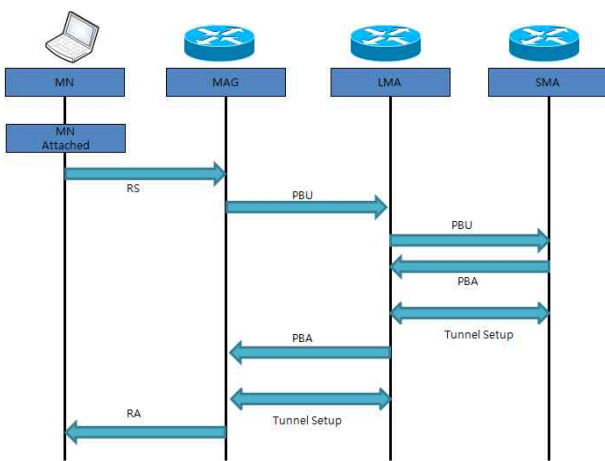
2.2.3 N.Neumann 기법

[6][7]은 PMIPv6 도메인 간 핸드오버를 위해 SMA(Session Mobility Anchor)라는 네트워크 엔티티를 정의하



(그림 10) N-Neumann 기법

었다. SMA는 MN이 이동성 세션동안 연결된 첫 번째 PMIPv6 도메인의 LMA로, 이동성 세션은 MN이 네트워크로 부티의 이동성 지원을 필요로 하는 기간이다. SMA는 이동성 세션이 지속되는 동안 MN과 관련된 모든 들어오고 나가는 패킷들을 처리하게 된다. MN이 PMIPv6 도메인 사이를 이동할 시의 동작은 (그림 11)과 같다. MN이 처음 접속한 PMIPv6 도메인 내에 있을 경우 SMA는 기존의 PMIPv6의 LMA와 동일하게 동작한다. 하지만 MN이 새로운 PMIPv6 도메인으로 이동하면 SMA는 새로운 도메인의 LMA와 터널을 형성하고 MN을 향하는, 또는 MN으로부터 나가는 패킷들은 이 터널을 통해 전송된다. 새로운 도메인의 LMA는 기존의 PMIPv6 LMA와 동일한 역할을 수행하여 MN에게 이동성을 제공한다. MN이 또 다른 PMIPv6 도메인으로 이동하게 되면 SMA는 이전 도메인의 LMA와 형성했던 터널을 제거하고 방금 이동한 새로운 도메인의 LMA와 다시 터널을 형성한다. Neumann의 기법에서는 SMA를 정하는 방법에 따라 Direct Location과 Indirect Location 두 가지 방법으로 구분하고 있다.



(그림 11) N-Neumann의 기법 핸드오버 절차

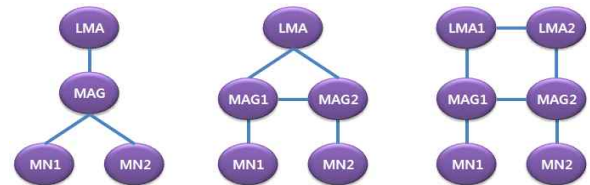
### 3. 비용효과적인 도메인 간 분산 이동성 제어기법

네트워크 기반 이동성관리 프로토콜인 PMIPv6는 하나의 도메인 내에서의 이동만을 지원할 뿐, 도메인 간 핸드오버

가 발생할 경우에는 이동성을 보장하지 않는다. 이 문제를 해결하기 위해 앞서 살펴보았던 방법들이 IETF NetLMM(Network-based Local Mobility Management) WG에서 제안되었다. 하지만, PMIPv6와 MIPv6의 연동 기법은 MN이 MIPv6 프로토콜 스택을 필요로 한다는 문제점이 발생하게 된다. 그리고 Na의 기법은 이전 도메인의 AAA 서버에서 Policy Profile을 받아와야 하기 때문에 긴 핸드오버 지연시간이 발생하게 된다. 또한 Neumann의 기법 역시 도메인 간 핸드오버를 위해 VMA라는 공동 데이터베이스를 거쳐야하기에 긴 핸드오버 지연시간이 필요하다. 이러한 문제점을 보완하기 위해 이 논문에서는 PIG를 이용한 두 가지 전역 이동성 제어 기법을 설명한다. 본 논문에서 제안한 기법의 구현을 위해 단말의 이동성을 도메인 내(Intra)와 도메인 간(Inter)로 구분하고 2010년 2월 NetLMM WG에서 제안한 [9] 기법을 이용한다. [9]에서 제안한 경로 최적화 기법은 기존의 경로 최적화 메시지 내에 (그림 12)와 같은 Local Routing Optimization Mobility Option을 추가하여 이 옵션 내에 있는 LRI(Local Routing Optimization Indication) 필드 값을 이용한다.

0	1	2	3
0 1 2 3 4 5 6 7	8 9 0 1 2 3 4 5	6 7 8 9 0 1 2 3 4 5 6 7 8 9	0 1
Type = TBD	Length	Reserved	LRI

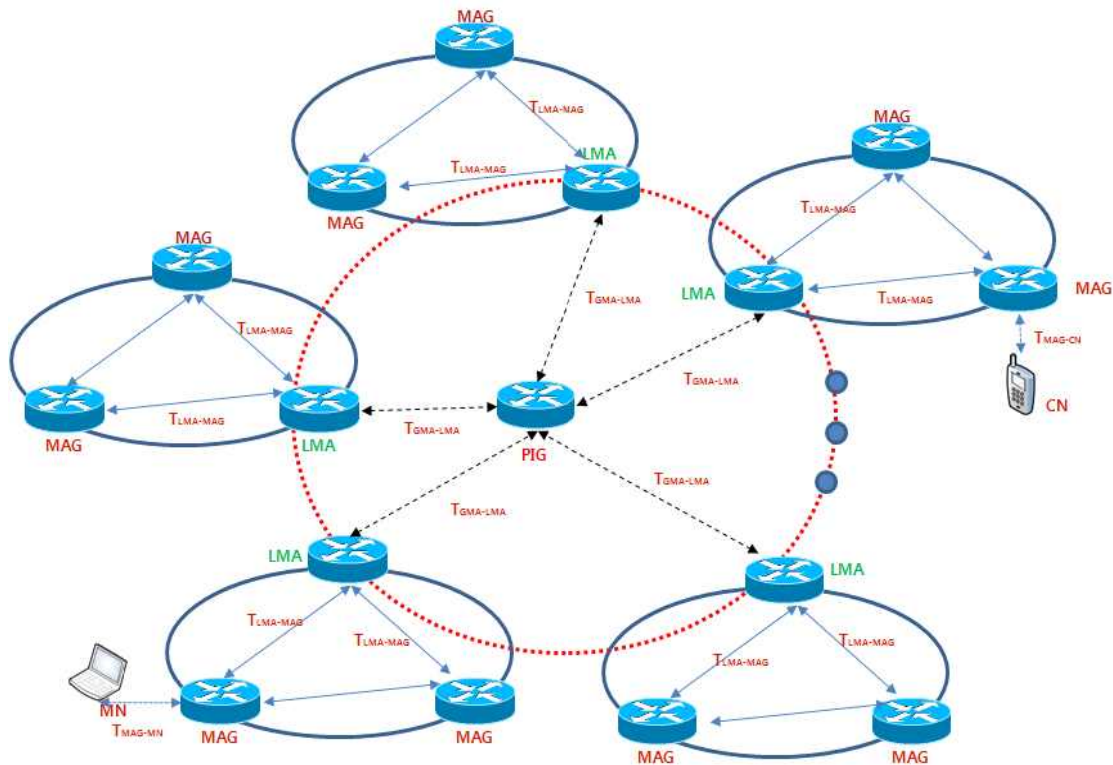
(그림 12) Local Routing Optimization Mobility Option[9]



(그림 13) PMIPv6의 네트워크 구성 환경

이 LRI 필드 값은 (그림 13)과 같이 경로 최적화가 이뤄지는 네트워크 구성 환경에 따라 다르게 설정된다. 두 MN이 하나의 LMA와 MAG에 부착되어 있는 Intra-MAG Local Routing의 경우에는 LRI 필드 값이 01로 설정되고, 두 MN이 하나의 LMA 상에 각각 다른 MAG에 부착되어 있는 Intra-LMA Local Routing의 경우에는 LRI 필드 값이 10으로 설정된다. 그리고 두 MN이 각각 다른 LMA와 MAG에 부착되어 있는 Inter-LMA Local Routing의 경우에는 LRI 필드 값이 11로 설정되며, 경로 최적화 상태를 알 수 없거나 불가능할 경우에는 LRI 필드 값이 00으로 설정된다. 이 필드값을 이용하여 제안한 기법을 Intra와 Inter의 이동성으로 구분한다.

LMA 주된 역할이 패킷 포워딩인데 이를 데이터 신호라고 하고, 어떤 패킷을 어디로 보낼지 정하는 라우팅테이블 관리를 맡는 부분을 제어신호라고 한다. PIG를 이용한 신호 분산 제어방법은 부분적으로 분산처리된 구조로 부분적으로



(그림 14) PIG를 이용한 전역 이동성 제어 구조

데이터부로부터 제어신호를 분리하는 방법이다. PIG 에 LMA를 관리하는 LIT(LMA Information Table)을 정의하여 데이터 패킷을 전달할 목적지를 쉽게 찾아 갈 수 있는 방법이다. PIG를 이용한 데이터신호 분산 제어방법은 PIG로부터 LMA 정보를 수신 받아서 네트워크에 있는 LMA들에게 RO\_REQ 명령을 보내면 MN이 포함된 PMIPv6 도메인의 LMA만 RO\_REP 명령으로 응답한다. 이때, RO\_REP 정보에 포함된 MAG 주소로 터널을 형성하여 데이터를 전달한다.

3.1 PIG를 이용한 제어신호 분산 제어방법

PMIPv6는 도메인 외부로의 이동성을 지원하지 않는다. 전역 이동성을 지원하기 위해서 PMIPv6 네트워크에 각 도메인의 LMA 정보를 관리하는 LIT를 PIG에 정의하여 다른 도메인에 있는 CN의 데이터를 가져올 수 있는 방법이 있다.

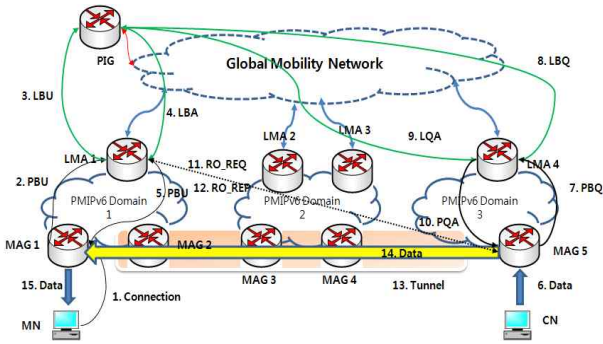
제안 기법의 동작 절차는 다음과 같다. 우선, MN 이 MAG1에 연결을 시도한다. MAG1은 LMA1에 PBU 메시지를 전송한다. PBU 메시지를 수신한 LMA1은 LMA1이 포함되어 있는 도메인 이름과 LMA1관련 정보를 PIG에 LBU(LMA Binding Update) 명령을 통해 업데이트를 한다. PIG는 LMA 관련정보를 LIT Table에 업데이트를 하고 LBA(LMA Binding Ack)명령을 LMA1에 보낸다. LBA 명령을 수신한 LMA1은 MN-HNP를 생성하고 BCE에 MN의 정보를 추가한다. CN은 MN에 보낼 데이터 패킷을 MAG5에 전송한다. 데이터를 수신한 MAG5는 PBQ 명령을 LMA4로 보낸다. PBQ 명령을 수신한 LMA4는 LBQ(LMA

<표 2> LMA Information Table 의 구조

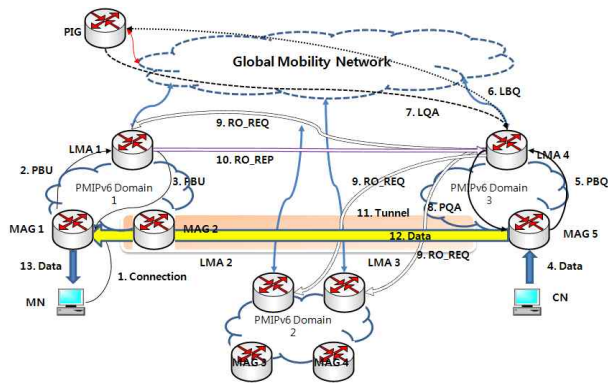
PMIPv6 Domain	LMA Name	LMA HNP Range	LMA Address
PMIPv6 Domain 1	LMA 1	aaaa::/64	LMA1_address
PMIPv6 Domain 2	LMA 2	bbbb::/64	LMA2_address
	LMA 3	cccc::/64	LMA3_address
PMIPv6 Domain 3	LMA 4	dddd::/64	LMA4_address
...	...	...	...

Binding Query) 명령을 PIG로 보낸다. LBQ 명령을 수신한 PIG는 데이터를 요청한 LMA1이 어느 도메인에 속해 있는지 도메인 정보와 LMA 정보를 LMA4로 송신한다. LMA4는 PQA 명령을 MAG5에 전송한다. 이때 PQA 정보내에 데이터를 요청한 도메인의 LMA1정보가 포함되어 있다. LMA1의 정보를 받은 MAG5는 LMA1에 RO\_REQ 명령을 보낸다. RO\_REQ 명령을 받은 LMA1은 RO\_REP 명령을 통해 MN의 MAG주소인 MAG1 주소를 알려준다.

MN의 MAG1 주소를 알게 된 MAG5는 MAG1에 최적화 터널을 형성하기 위한 Tunneling Request 명령을 보내고 MAG1에게 Tunneling Response 명령을 받으면 MAG1과 MAG5사이 터널이 형성된다. 형성된 터널을 통해 데이터를 전송한다.



(그림 15) PIG를 이용한 신호 분산처리



(그림 16) PIG를 이용한 데이터 분산제어

3.2 PIG 를 이용한 데이터신호 분산 제어방법

이 제안 기법은 전역 이동성 네트워크에 있는 모든 PMIPv6 도메인들은 PIG에 등록되어 있다고 가정한다.

PIG 를 이용한 데이터신호 분산 제어방법의 동작은 다음과 같다. 우선, MN 이 MAG1에 연결을 시도한다. MAG1은 LMA1에 PBU 메시지를 전송한다. PBU 메시지를 수신한 LMA1은 LMA1이 포함되어 있는 도메인 이름과 LMA1관련 정보를 PIG에 LBU 명령을 통해 업데이트를 한다. CN은 MN에 보낼 데이터 패킷을 MAG5에 전송한다. 데이터를 수신한 MAG5는 PBQ 명령을 LMA4로 보낸다. PBQ 명령을 수신한 LMA4는 LBQ 명령을 PIG로 보낸다. LBQ 명령을

수신한 PIG는 LIT에 있는 도메인 정보와 LMA 정보를 LMA4로 송신한다. LMA4는 PQA 명령을 MAG5에 전송한다. LMA4는 PIG에서 수신 받은 LMA들에 RO\_REQ 명령을 보낸다. RO\_REQ 명령을 받은 LMA들 중 MN이 있는 LMA만 RO\_REQ 명령을 LMA4로 전송한다. MN의 MAG1 주소를 알게 된 MAG5는 MAG1에 최적화 터널을 형성하기 위한 Tunneling Request 명령을 보내고 MAG1에게 Tunneling Response 명령을 받으면 MAG1과 MAG5사이 터널이 형성된다. 형성된 터널을 통해 데이터를 전송한다.

4. 성능 평가

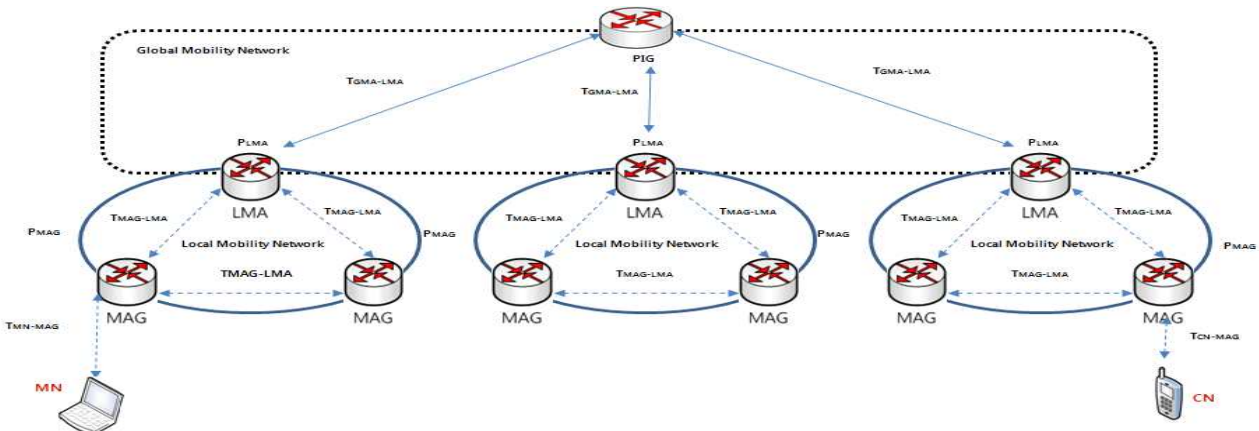
4.1 성능평가 구성

PMIPv6 도메인 내에서 지역 분산 이동성 제어와 PMIPv6 도메인 간의 전역 이동성 제어를 결합한 구조이다. PMIPv6의 모든 도메인은 PIG를 중심으로 결합되어 있고 PMIPv6 도메인은 도메인을 총괄하는 LMA와 MAG로 구성되어 있다. 도메인 간 전역 이동성제어는 PIG를 통해 수행된다.

<표 3>은 비용 분석을 위한 매개변수를 정의한다.

<표 3> 비용 분석을 위한 매개변수

매개변수	설명
$T_{a,b}$	node a와 b 사이에 패킷 전송 비용
$P_c$	node c를 위한 바인딩 업데이트 또는 록업 처리 비용
$T_{setup}$	MN 과 MAG 사이의 연결 설정 시간
$N_{Host/MAG}$	MAG 당 활동하는 호스트 수
$N_{MAG}$	PMIPv6 도메인 내 MAG 수
$H_{a-b}$	네트워크 내에서 노드 a와 b 사이의 Hop 수
$S_{control}$	제어 패킷의 크기 (바이트)
$S_{data}$	데이터 패킷의 크기(바이트)
$\alpha$	LMA가 바인딩 업데이트 하는데 사용하는 비용
$\beta$	LMA 또는 MAG 위에서 록업 비용
$\tau$	유선 링크에서 패킷전송비용
$k$	무선 링크에서 패킷전송 비용



(그림 17) 제안기법의 네트워크 구성



4.2 비용분석

4.2.1 S-PMIP-INTRA

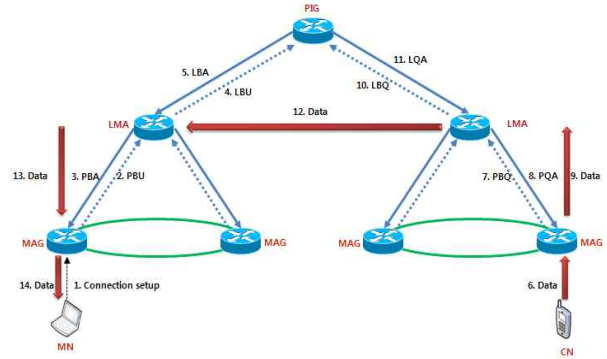
S-PMIP의 바인딩 업데이트 작업은 다음과 같이 수행된다. MN이 새로운 MAG 지역에 들어왔을 때, MN은 연결을 설정하고 MN의 HoA를 얻는다[3]. 이 동작이 대략  $T_{setup}$  만큼 소요된다고 가정한다. 그 후, MN의 MAG가 PBU 및 PBA 제어 메시지를 교환하여 LMA와 PBU 작업을 수행하고, LMA는 데이터베이스를 업데이트한다. 이 동작은  $2T_{MAG-LMA}$ 와  $P_{LMA}$ 가 걸린다.  $T_{MAG-LMA} = \tau H_{MAG-LMA}$ 와  $P_{LMA} = \alpha \log(N_{MAG} \times N_{Host/MAG})$ 와 같다. 그것은 LMA ( $P_{LMA}$ )와 바인딩 업데이트에 대한 처리 비용은 데이터베이스를 구현하는 트리 기반 데이터 구조를 사용하여 로그 규모의 도메인에서 활성화된 호스트의 총 개수 ( $N_{MAG} \times N_{Host/MAG}$ )에 비례한다. 따라서 PMIPv6의 바인딩 업데이트 비용은 다음과 같이 표현할 수 있다.

패킷을 전달하는데, 바인딩 검색에 대한 제어 작업은 데이터 패킷 전달에서 분리된다. 따라서 CN에서 MN으로 패킷 전달 비용은 다음과 같이 계산한다. 첫째, CN의 데이터 패킷이  $S_{data} \times T_{CN-MAG}$ 는 MAG에 전달된다.

CN의 MAG는 LMA에서 PBQ 및 PQA 메시지를 교환하여 MN의 CoA얻는다. 이때 소요 비용은  $S_{control} \times 2T_{MAG-LMA} + P_{LMA}$ 와 같다. CN의 MAG가 MN의 MAG에

데이터 패킷을 전달하고, 소요비용은  $S_{data} \times (T_{MAG-MAG} + T_{MAG-MN})$  이다. 따라서 S-PMIP의 패킷 전달 비용은 다음과 같다.

4.2.2 S-PMIP-INTER



(그림 18) S-PMIP 도메인간의 분산제어

S-PMIP의 도메인 간 바인딩 업데이트 작업은 다음과 같이 수행된다. MN이 새로운 MAG 지역에 들어왔을 때, MN은 연결을 설정하고 MN의 HoA를 얻는다[3]. 이 동작이 대략  $T_{setup}$  만큼 소요된다고 가정한다. 그 후, MN의 MAG가 PBU 및 PBA 제어 메시지를 교환하여 LMA와 PBU 작

<표 4> S-PMIP의 PMIPv6 도메인 내의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$
2, 3	MN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$

$$\begin{aligned}
 BUC_{S-PMIP-INTRA} &= T_{SETUP} + S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA} \\
 &= T_{SETUP} + S_{CONTROL} \times 2\tau H_{MAG-LMA} + \alpha \log(N_{MAG} \times N_{Host/MAG})
 \end{aligned}$$

<표 5> S-PMIP의 PMIPv6 도메인 내의 패킷 전달 비용

단계	동작	비용
4	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
5, 6	CN의 MAG는 PBQ 메시지를 LMA로 보내고 PQA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$
7, 8	CN의 MAG는 데이터를 MN의 MAG로 보내고, MN의 MAG는 MN으로 데이터를 보낸다.	$S_{DATA} \times (T_{MAG-MAG} + T_{MN-MAG})$

$$\begin{aligned}
 PDC_{S-PMIP-INTRA} &= S_{DATA} (T_{CN-MAG} + T_{MAG-MAG} + T_{MAG-MN}) + S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA} \\
 &= S_{DATA} (\kappa H_{CN-MAG} + \tau H_{MAG-MAG} + \kappa H_{MN-MAG}) + S_{CONTROL} \times 2\tau H_{MAG-LMA} + \beta \log(N_{MAG} \times N_{Host/MAG})
 \end{aligned}$$

총 소요 비용은  $TC_{S-PMIP-INTRA} = BUC_{S-PMIP-INTRA} + PDC_{S-PMIP-INTRA}$  이다.

〈표 6〉 S-PMIP의 PMIPv6 도메인 간의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$
2, 3	MN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$
4, 5	MN의 LMA는 LBU 메시지를 PIG로 보내고 LBA 메시지를 수신 받는다. PIG는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{LMA-PIG} + P_{PIG}$

$$\begin{aligned}
 BUC_{S-PMIP-INTER} &= T_{SETUP} + S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA} + 2T_{LMA-PIG} + P_{PIG} \\
 &= T_{SETUP} + S_{CONTROL} \times 2\tau X(H_{MAG-LMA} + H_{LMA-PIG}) + \alpha \log(N_{MAG} \times N_{Host/MAG}) + \alpha \log(N_{LMA} \times N_{MAG/LMA})
 \end{aligned}$$

〈표 7〉 S-PMIP의 PMIPv6 도메인 간의 패킷 전달 비용

단계	동작	비용
6	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
7, 8	CN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$
9	CN의 MAG는 LMA에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{MAG-LMA}$
10, 11	CN의 LMA는 LBQ 메시지를 PIG로 보내고 LQA 메시지를 수신 받는다. PIG는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{LMA-PIG} + P_{PIG}$
12	CN의 LMA는 MN의 LMA로 데이터 패킷을 전달한다.	$S_{DATA} \times T_{LMA-LMA}$
13, 14	MN의 LMA는 MN의 MAG로 데이터 패킷을 전달한다. MN의 MAG는 MN으로 데이터 패킷을 전달한다.	$S_{DATA} \times (T_{MAG-LMA} + T_{MN-MAG})$

$$\begin{aligned}
 PDC_{S-PMIP-INTER} &= S_{DATA} (T_{CN-MAG} + T_{MAG-LMA} + T_{LMA-LMA} + T_{LMA-MAG} + T_{MAG-MN}) \\
 &\quad + S_{CONTROL} \times (2T_{MAG-LMA} + 2T_{LMA-PIG}) + P_{LMA} + P_{PIG} \\
 &= S_{DATA} (\kappa H_{CN-MAG} + \tau H_{MAG-MAG} + 2\tau H_{MAG-LMA} + \tau H_{LMA-LMA} + \kappa H_{MN-MAG}) \\
 &\quad + S_{CONTROL} (2\tau H_{MAG-LMA} + 2\tau H_{LMA-PIG}) + \beta \log(N_{MAG} \times N_{Host/MAG}) + \beta \log(N_{LMA} \times N_{MAG/LMA})
 \end{aligned}$$

총 소요 비용은  $TC_{S-PMIP-INTER} = BUC_{S-PMIP-INTER} + PDC_{S-PMIP-INTER}$  이다.

업을 수행하고, LMA는 데이터베이스를 업데이트한다. 이 동작은  $2T_{MAG-LMA}$ 와  $P_{LMA}$ 가 걸린다. MN의 LMA가 LBU 및 LBA 제어 메시지를 PIG와 교환하여 PIG는 데이터베이스를 업데이트한다. 따라서 S-PMIP의 도메인 간 바인딩 업데이트 비용은 다음과 같이 표현할 수 있다.

패킷을 전달하는데, 바인딩 검색에 대한 제어 작업은 데이터 패킷 전달에서 분리된다. 따라서 CN에서 MN으로 패킷 전달 비용은 다음과 같이 계산한다. 첫째, CN의 데이터 패킷이 MAG에 전달된다. CN의 MAG는 LMA에서 PBU 및 PBA 메시지를 교환하여 MN의 CoA를 얻는다. CN의 MAG가 LMA에 데이터 패킷을 전달한다. CN의 LMA는 LBQ와 LQA 메시지를 PIG와 주고받는다. CN의 LMA는 MN의 LMA로 데이터 패킷을 전달한다. MN의 LMA는 MN의 MAG로 데이터 패킷을 전달한다. 최종적으로, MN의 MAG는 MN으로 데이터 패킷을 전달한다. S-PMIP의 도메인 간 패킷 전달 비용은 다음과 같다.

#### 4.2.3 DD-PMIP-INTER

DD-PMIP은 MAG와 LMA간에 바인딩 업데이트가 수행되지 않는다. 따라서 DD-PMIP의 바인딩 업데이트 비용은 다음과 같다.

〈표 8〉 DD-PMIP의 PMIPv6 도메인 내의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$

$$BUC_{DD-PMIP-INTER} = T_{SETUP}$$

DD-PMIP은 데이터 구동 방식이며, 따라서 CN에서 MN으로 패킷 전달 비용은 다음과 같이 계산한다. 첫째, CN의 데이터 패킷이 동일 MAG에 전달된다. CN의 MAG는 멀티 캐스트에 의해 동일 도메인의 모든 MAG에 데이터 패킷을 보냅니다. MN의 MAG만 PQA 메시지를 CN의 MAG에 전달한다. MN의 MAG는 MN에 데이터 패킷을 전달한다. 따라서 DD-PMIP의 패킷 전달 비용은 다음과 같이 표현할 수 있다.

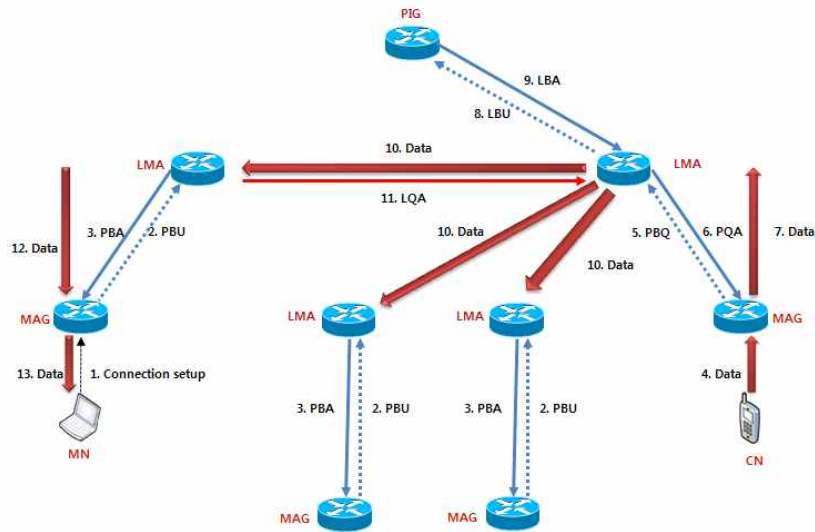
〈표 9〉 DD-PMIP의 PMIPv6 도메인 내의 패킷전달 비용

단계	동작	비용
2	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
3	CN의 MAG는 도메인 내 모든 MAG에 멀티캐스트로 데이터 패킷을 전달한다.	$S_{DATA} \times (T_{MAG-MAG} \times N_{MAG})$
4	MN의 MAG만 PQA 메시지로 응답한다. 나머지 MAG는 데이터를 무시한다.	$S_{CONTROL} \times T_{MAG-MAG}$
5	MN의 MAG는 데이터 패킷을 MN으로 전달한다.	$S_{DATA} \times T_{MN-MAG}$

$$PDC_{DD-PMIP-INTRA} = S_{DATA}(T_{CN-MAG} + T_{MAG-MAG} \times N_{MAG} + T_{MAG-MN}) + S_{CONTROL} \times T_{MAG-MAG} + P_{MAG}$$

$$= S_{DATA}(\kappa H_{CN-MAG} + \tau H_{MAG-MAG} \times N_{MAG} + \kappa H_{MN-MAG}) + S_{CONTROL} \times \tau H_{MAG-MAG} + \beta \log(N_{Host/MAG})$$

총 소요 비용은  $TC_{DD-PMIP-INTRA} = BUC_{DD-PMIP-INTRA} + PDC_{DD-PMIP-INTRA}$  이다.



(그림 19) DD-PMIP 기법의 도메인 간 확장

4.2.4 DD-PMIP-INTER

DD-PMIP 기법을 도메인 간으로 확장하였을 경우 구성은 (그림 19)와 같다. MN이 MAG에 연결되고 MN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신받는다. DD-PMIP의 도메인 간 바인딩 업데이트 비용은 다음과 같다. CN에서 MN으로 패킷 전달 비용은 다음과 같이 계산한다. CN의 MAG는 PBQ 메시지를 LMA로 보내고 PQA 메

시지를 수신 받는다. CN의 MAG는 LMA에 데이터 패킷을 전달하고, CN의 LMA는 LBQ 메시지를 PIG로 보내고 LQA 메시지를 수신 받는다. CN의 LMA는 멀티캐스트로 데이터 패킷을 PIG 내 모든 LMA로 전송한다. MN의 LMA만 LQA 메시지로 응답하고 나머지 LMA는 데이터를 무시한다. MN의 LMA는 MN의 MAG로 데이터 패킷을 전달한다. MN의 MAG는 데이터 패킷을 MN으로 전달한다.

〈표 10〉 DD-PMIP의 PMIPv6 도메인 간의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$
2, 3	MN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$

$$BUC_{DD-PMIP-INTER} = T_{SETUP} + S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$$

$$= T_{SETUP} + S_{CONTROL} \times 2\tau H_{MAG-LMA} + \alpha \log(N_{MAG} \times N_{Host/MAG})$$

〈표 11〉 DD-PMIP의 PMIPv6 도메인 간의 패킷전달 비용

단계	동작	비용
4	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
5, 6	CN의 MAG는 PBQ 메시지를 LMA로 보내고 PQA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times T_{MAG-LMA} + P_{LMA}$
7	CN의 MAG는 LMA에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{MAG-LMA}$
8, 9	CN의 LMA는 LBQ 메시지를 PIG로 보내고 LQA 메시지를 수신 받는다.	$S_{CONTROL} \times T_{LMA-PIG} + P_{PIG}$
10	CN의 LMA는 멀티캐스트로 데이터 패킷을 PIG 내 LMA로 전송한다.	$S_{DATA} \times (T_{LMA-LMA} \times N_{LMA})$
11	MN의 LMA만 LQA 메시지로 응답하고 나머지 LMA는 데이터를 무시한다.	$S_{CONTROL} \times T_{LMA-LMA}$
12	MN의 LMA는 MN의 MAG로 데이터 패킷을 전달한다.	$S_{DATA} \times T_{MAG-LMA}$
13	MN의 MAG는 데이터 패킷을 MN으로 전달한다.	$S_{DATA} \times T_{MN-MAG}$

$$\begin{aligned}
 PDC_{DD-PMIP-INTER} &= S_{DATA}(T_{CN-MAG} + 2T_{MAG-LMA} + T_{LMA-LMA} \times N_{LMA} + T_{MAG-MN}) \\
 &\quad + S_{CONTROL}(T_{MAG-LMA} + T_{LMA-PIG} + T_{LMA-LMA}) + P_{LMA} + P_{PIG} \\
 &= S_{DATA}(\kappa H_{CN-MAG} + 2\tau H_{MAG-MAG} + \tau H_{LMA-LMA} \times N_{LMA} + \kappa H_{MN-MAG}) \\
 &\quad + S_{CONTROL}(\tau H_{MAG-LMA} + 2\tau H_{LMA-MAG} + \tau H_{LMA-LMA}) + \beta \log(N_{MAG} \times N_{Host/MAG}) + \beta \log(N_{LMA} \times N_{MAG/LMA})
 \end{aligned}$$

총 소요 비용은  $TC_{DD-PMIP-INTER} = BUC_{DD-PMIP-INTER} + PDC_{DD-PMIP-INTER}$  이다.

4.2.5 SD-PMIP-INTRA

SD-PMIP의 도메인 내 바인딩 업데이트 비용은 다음과 같다.

〈표 12〉 SD-PMIP의 PMIPv6 도메인 내의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$

$$BUC_{SD-PMIP-INTRA} = T_{SETUP}$$

SD-PMIP의 패킷전송 비용은 다음과 같다. CN은 MAG에 데이터 패킷을 보낸다. CN의 MAG는 도메인 내 모든 MAG에 멀티캐스트로 PBQ 제어신호를 보낸다. 데이터 패킷을 수신한 MN의 MAG는 PQA 메시지를 CN의 MAG에 보낸다. CN의 MAG는 MN의 MAG로 데이터 패킷을 전송하고, 마지막으로 데이터 패킷은 MN에 전달된다.

4.2.6 SD-PMIP-INTER

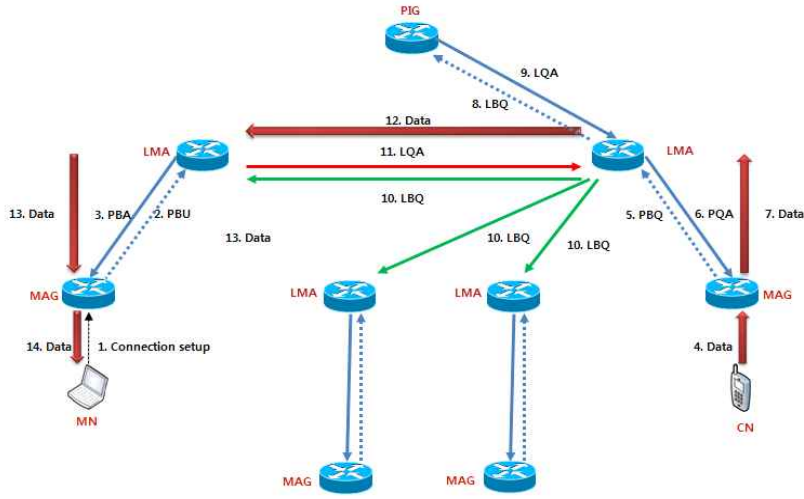
SD-PMIP의 도메인 간 바인딩 업데이트 비용은 다음과 같다.

〈표 13〉 SD-PMIP의 PMIPv6 도메인 내의 패킷전달 비용

단계	동작	비용
2	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
3	CN의 MAG는 도메인 내 모든 MAG에 멀티캐스트로 PBQ 메시지를 전달한다.	$S_{CONTROL} \times (T_{MAG-MAG} \times N_{MAG})$
4	MN의 MAG만 PQA 메시지로 응답한다. 나머지 MAG는 데이터를 무시한다.	$S_{CONTROL} \times T_{MAG-MAG}$
5	CN의 MAG는 MN의 MAG로 데이터 패킷을 전송한다.	$S_{DATA} \times T_{MAG-MAG}$
6	MN의 MAG는 데이터 패킷을 MN으로 전달한다.	$S_{DATA} \times T_{MN-MAG}$

$$\begin{aligned}
 PDC_{SD-PMIP-INTRA} &= S_{DATA}(T_{CN-MAG} + T_{MAG-MAG} + T_{MAG-MN}) + S_{CONTROL} \times 2T_{MAG-MAG} \times (N_{MAG} + 1) + P_{MAG} \\
 &= S_{DATA}(\kappa H_{CN-MAG} + \tau H_{MAG-MAG} + \kappa H_{MN-MAG}) + S_{CONTROL} \times \tau 2H_{MAG-MAG} \times (N_{MAG} + 1) + \beta \log(N_{Host/MAG})
 \end{aligned}$$

총 소요 비용은  $TC_{SD-PMIP-INTRA} = BUC_{SD-PMIP-INTRA} + PDC_{SD-PMIP-INTRA}$  이다.



(그림 20) SD-PMIP 기법의 도메인 간 확장

〈표 14〉 SD-PMIP의 PMIPv6 도메인 간의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$
2, 3	MN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$

$$\begin{aligned}
 BUC_{SD-PMIP-INTER} &= T_{SETUP} + S_{CONTROL} \times 2T_{MAG-LMA} \\
 &= T_{SETUP} + S_{CONTROL} \times 2\tau H_{MAG-LMA} + \alpha \log(N_{MAG} \times N_{Host/MAG})
 \end{aligned}$$

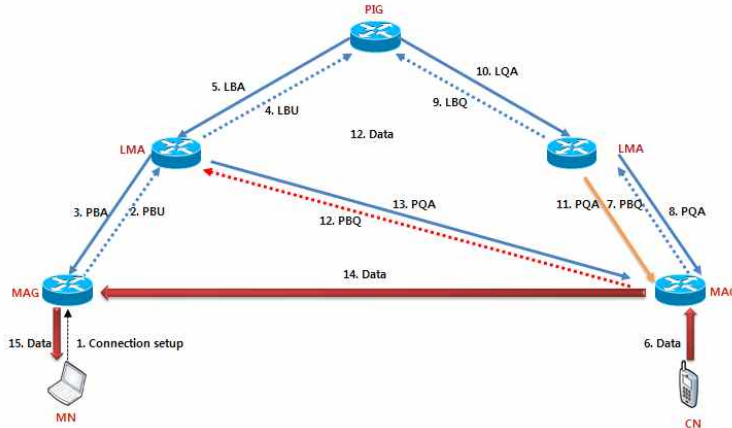
SD-PMIP의 도메인 간 패킷전송 비용은 다음과 같다.

〈표 15〉 SD-PMIP의 PMIPv6 도메인 간의 패킷전달 비용

단계	동작	비용
4	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
5, 6	CN의 MAG는 PBQ 메시지를 LMA로 보내고 PQA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times T_{MAG-LMA} + P_{LMA}$
7	CN의 MAG는 LMA에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{MAG-LMA}$
8, 9	CN의 LMA는 LBQ 메시지를 PIG로 보내고 LQA 메시지를 수신 받는다.	$S_{CONTROL} \times T_{LMA-PIG} + P_{PIG}$
10	CN의 LMA는 멀티캐스트로 LBQ 메시지를 PIG 내 LMA로 전송한다.	$S_{CONTROL} \times (T_{LMA-LMA} \times N_{LMA})$
11	MN의 LMA만 LQA 메시지로 응답하고 나머지 LMA는 데이터를 무시한다.	$S_{CONTROL} \times T_{LMA-LMA}$
12	MN의 LMA는 CN의 LMA로 데이터 패킷을 전달한다.	$S_{DATA} \times T_{LMA-LMA}$
13	MN의 LMA는 MN의 MAG로 데이터 패킷을 전달한다.	$S_{DATA} \times T_{MAG-LMA}$
14	MN의 MAG는 데이터 패킷을 MN으로 전달한다.	$S_{DATA} \times T_{MN-MAG}$

$$\begin{aligned}
 PDC_{SD-PMIP-INTER} &= S_{DATA} (T_{CN-MAG} + 2T_{MAG-LMA} + T_{LMA-LMA} + T_{MAG-MN}) \\
 &\quad + S_{CONTROL} (T_{MAG-LMA} + T_{LMA-PIG} + T_{LMA-LMA} \times (N_{LMA} + 1)) + P_{LMA} + P_{PIG} \\
 &= S_{DATA} (\kappa H_{CN-MAG} + 2\tau H_{MAG-MAG} + \tau H_{LMA-LMA} + \kappa H_{MN-MAG}) \\
 &\quad + S_{CONTROL} (\tau H_{MAG-LMA} + 2\tau H_{LMA-PIG} + \tau H_{LMA-LMA} \times (N_{LMA} + 1)) \\
 &\quad + \beta \log(N_{MAG} \times N_{Host/MAG}) + \beta \log(N_{LMA} \times N_{MAG/LMA})
 \end{aligned}$$

총 소요 비용은  $TC_{DD-PMIP-INTER} = BUC_{DD-PMIP-INTER} + PDC_{DD-PMIP-INTER}$  이다.



(그림 21) PIG를 이용한 신호분산 제어

4.2.7 PIG를 이용한 신호 분산 제어방법

SD-GMIP의 도메인 간 바인딩 업데이트 비용은 다음과 같다.

<표 16> SD-GMIP의 PMIPv6 도메인 간의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$
2, 3	MN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$
4, 5	MN의 LMA는 LBU 메시지를 PIG로 보내고 LBA 메시지를 수신 받는다. PIG는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{LMA-PIG} + P_{PIG}$

$$\begin{aligned}
 BUC_{SD-GMIP-INTER} &= T_{SETUP} + S_{CONTROL} (2T_{MAG-LMA} + 2T_{LMA-PIG}) + P_{LMA} + P_{PIG} \\
 &= T_{SETUP} + S_{CONTROL} (2\tau H_{MAG-LMA} + 2\tau H_{LMA-PIG}) + \alpha \log(N_{MAG} \times N_{Host/MAG}) + \alpha \log(N_{LMA} \times N_{PIG/LMA})
 \end{aligned}$$

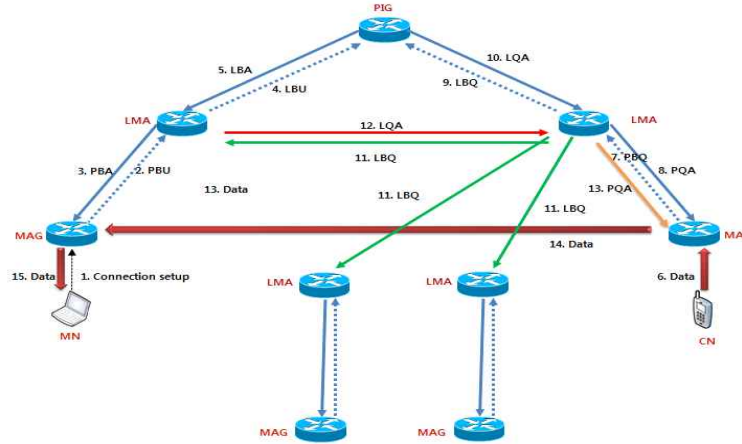
SD-GMIP의 도메인 간 패킷전송 비용은 다음과 같다.

<표 17> SD-GMIP의 PMIPv6 도메인 간의 패킷전달 비용

단계	동작	비용
6	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
7, 8	CN의 MAG는 PBQ 메시지를 LMA로 보내고 PQA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$
9, 10	CN의 LMA는 LBQ 메시지를 PIG로 보내고 LQA 메시지를 수신 받는다.	$S_{CONTROL} \times 2T_{LMA-PIG} + P_{PIG}$
11	CN의 LMA는 CN의 MAG로 PIG에서 수신한 정보를 PQA 메시지로 전송한다.	$S_{CONTROL} \times T_{MAG-LMA}$
12, 13	CN의 MAG는 MN의 LMA로 PBQ 메시지를 보내고 PQA 메시지로 수신 받는다.	$S_{CONTROL} \times 2T_{MAG-LMA}$
14	CN의 MAG는 MN의 MAG로 데이터 패킷을 전달한다.	$S_{DATA} \times T_{MAG-MAG}$
15	MN의 MAG는 데이터 패킷을 MN으로 전달한다.	$S_{DATA} \times T_{MN-MAG}$

$$\begin{aligned}
 PDC_{SD-GMIP-INTER} &= S_{DATA} (T_{CN-MAG} + T_{MAG-MAG} + T_{MAG-MN}) \\
 &\quad + S_{CONTROL} (4T_{MAG-LMA} + 2T_{LMA-PIG}) + P_{LMA} + P_{PIG} \\
 &= S_{DATA} (\kappa H_{CN-MAG} + \tau H_{MAG-MAG} + \kappa H_{MN-MAG}) \\
 &\quad + S_{CONTROL} (4\tau H_{MAG-LMA} + 2\tau H_{LMA-PIG}) + \beta \log(N_{MAG} \times N_{Host/MAG}) + \beta \log(N_{LMA} \times N_{PIG/LMA})
 \end{aligned}$$

총 소요 비용은  $TC_{SD-GMIP-INTER} = BUC_{SD-GMIP-INTER} + PDC_{SD-GMIP-INTER}$  이다.



(그림 22) PIG를 이용한 데이터 분산 제어

4.2.8 PIG를 이용한 데이터 신호 분산 제어방법

DD-GMIP의 도메인 간 바인딩 업데이트 비용은 다음과 같다.

<표 18> DD-GMIP의 PMIPv6 도메인 간의 바인딩 업데이트 비용

단계	동작	비용
1	MN이 MAG에 연결된다.	$T_{SETUP}$
2, 3	MN의 MAG는 PBU 메시지를 LMA로 보내고 PBA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$
4, 5	MN의 LMA는 LBU 메시지를 PIG로 보내고 LBA 메시지를 수신 받는다. PIG는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{LMA-PIG} + P_{PIG}$

$$\begin{aligned}
 BUC_{DD-GMIP-INTER} &= T_{SETUP} + S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA} + P_{PIG} \\
 &= T_{SETUP} + S_{CONTROL} \times 2\tau H_{MAG-LMA} + \alpha \log(N_{MAG} \times N_{Host/MAG}) + \alpha \log(N_{LMA} \times N_{PIG/LMA})
 \end{aligned}$$

DD-GMIP의 도메인 간 패킷전송 비용은 다음과 같다.

<표 19> DD-PMIP의 PMIPv6 도메인 간의 패킷전달 비용

단계	동작	비용
6	CN이 MAG에 데이터 패킷을 전달한다.	$S_{DATA} \times T_{CN-MAG}$
7, 8	CN의 MAG는 PBQ 메시지를 LMA로 보내고 PQA 메시지를 수신 받는다. LMA는 database를 업데이트한다.	$S_{CONTROL} \times 2T_{MAG-LMA} + P_{LMA}$
9, 10	CN의 LMA는 LBQ 메시지를 PIG로 보내고 LQA 메시지를 수신 받는다.	$S_{CONTROL} \times 2T_{LMA-PIG} + P_{PIG}$
11	CN의 LMA는 멀티캐스트로 LBQ 메시지를 PIG 내 LMA로 전송한다.	$S_{CONTROL} \times T_{LMA-LMA} \times N_{LMA}$
12	MN의 LMA만 LQA 메시지로 응답한다.	$S_{CONTROL} \times T_{LMA-LMA}$
13	CN의 LMA는 CN의 MAG로 MN의 MAG 주소를 전송한다.	$S_{CONTROL} \times T_{MAG-LMA}$
14	CN의 MAG는 MN의 MAG로 데이터를 전송한다.	$S_{DATA} \times T_{MAG-MAG}$
15	MN의 MAG는 데이터 패킷을 MN으로 전달한다.	$S_{DATA} \times T_{MN-MAG}$

$$\begin{aligned}
 PDC_{DD-GMIP-INTER} &= S_{DATA}(T_{CN-MAG} + T_{MAG-MAG} + T_{MAG-MN}) \\
 &\quad + S_{CONTROL}(3T_{MAG-LMA} + 2T_{LMA-PIG} + T_{LMA-LMA} \times (N_{LMA} + 1)) + P_{LMA} + P_{PIG} \\
 &= S_{DATA}(\kappa H_{CN-MAG} + \tau H_{MAG-MAG} + \kappa H_{MN-MAG}) \\
 &\quad + S_{CONTROL}(3\tau H_{MAG-LMA} + 2\tau H_{LMA-PIG} + \tau H_{LMA-LMA} \times (N_{LMA} + 1)) \\
 &\quad + \beta \log(N_{MAG} \times N_{Host/MAG}) + \beta \log(N_{LMA} \times N_{PIG/LMA})
 \end{aligned}$$

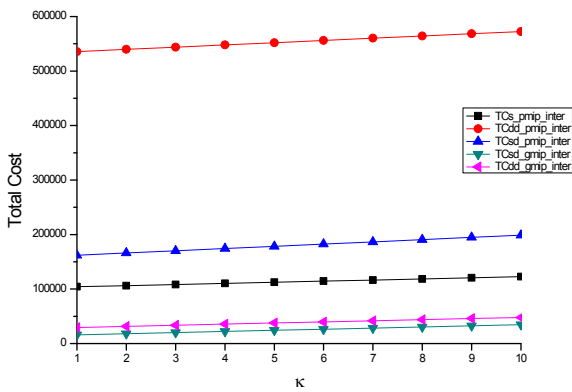
총 소요 비용은  $TC_{DD-GMIP-INTER} = BUC_{DD-GMIP-INTER} + PDC_{DD-GMIP-INTER}$  이다.

### 4.3 성능평가 결과

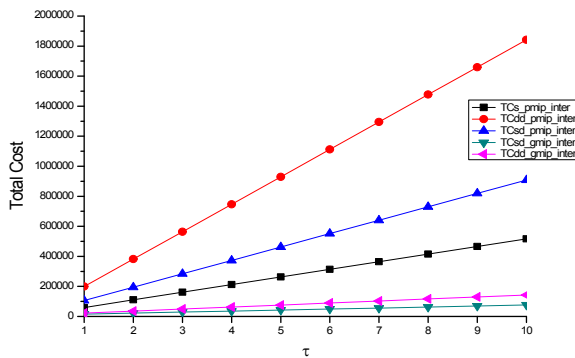
성능평가를 위해 매개변수 값을 <표 20>과 같이 설정하였다. 부분적인 매개변수 값은 [8]에서 유도된 값을 사용하였다. (그림 23)과 (그림 24)는 유/무선 링크 상에서 데이터 전송 비용의 차이를 비교하였다. 그림에서 보는 것과 같이 기존의 분산기법의 확장 보다 제안된 기법의 비용이 훨씬 효과적으로 나타났다. 무선링크 상에서는 총 비용이 모든 기법에서 선형으로 증가하나 무선에서는 바인딩 업데이트 비용의 차이로 그 차이가 크게 나타나는 것을 볼 수 있다.

<표 20> 성능평가를 위한 매개변수 값

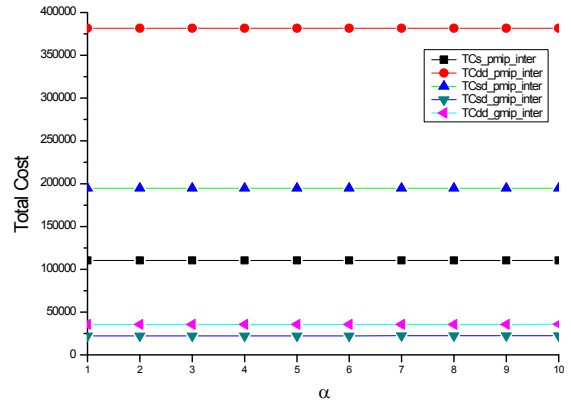
Parameter	Default	Minimum	Maximum
$\kappa$	4	1	10
$\tau$	2	1	10
$\alpha$	3	1	10
$\beta$	2	1	10
$N_{Host/MAG}$	100	100	1000
$N_{MAG}$	10	10	100
$N_{LMA}$	10	5	50
$H_{MAG-LMA}$	10	1	10
$H_{PIG-LMA}$		1	
$\gamma$	0.3	0	2.0
$H_{MN-MAG}, H_{CN-MAG}$		1	
$T_{setup}$		500(ms)	
$S_{data}$		1024(bytes)	
$S_{control}$		50(bytes)	



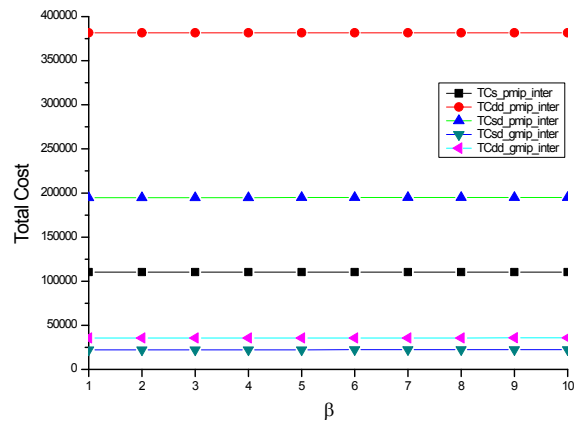
(그림 23) 무선링크에서 단위 전송 비용



(그림 24) 유선링크에서 단위 전송 비용

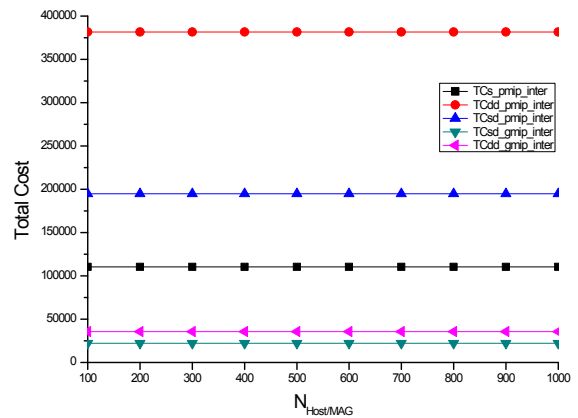


(그림 25) LMA에서 바인딩 업데이트 비용



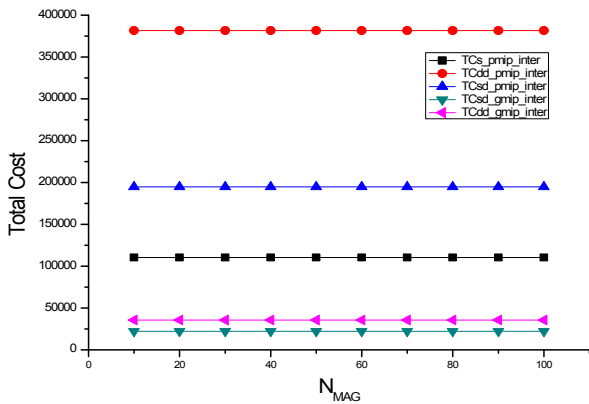
(그림 26) LMA 또는 MAG에서 록업 비용

(그림 25)과 (그림 26)은 바인딩 업데이트 비용과 록업 동작비용의 합계를 나타낸 그림이다. 그림에서 보는 것과 같이 모든 기법들이 거의 영향을 받지 않는다. 이것은 바인딩 업데이트와 록업 동작이 총 비용에 중요한 영향을 주지 않는 것을 알 수 있다.



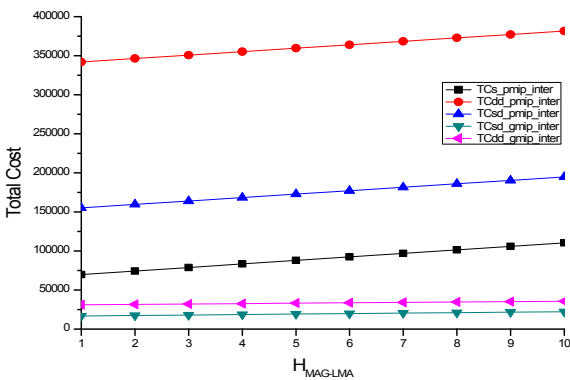
(그림 27) MAG 내 호스트 수



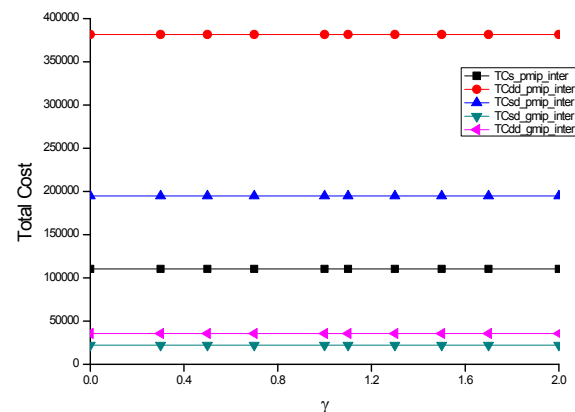


(그림 28) PMIP 도메인 내 MAG 의 수

(그림 27)은 MAG 내 포함된 호스트 수에 의한 총 사용 비용을 분석하였다. 그림에서 보는 것과 같이 MAG 내 호스트 수는 비교한 기법 모두에 거의 영향을 주지 않는 것을 알 수 있다.



(그림 29) MAG와 LMA 사이의 홉카운트 수



(그림 30) H\_MAG-LMA 안의 H\_MAG-MAG 홉카운트비율

모든 분석 결과를 보면, 기존의 도메인 내 분산 이동성 제어의 확장을 적용한 기법 보다 본 논문에서 제안한 기법의 성능이 모든 분석에서 비용 효과적으로 나타났다. 특히

PIG를 이용한 시그널 분산 제어 기법이 전체 비용측면에서 가장 효과적으로 나타났다.

### 5. 결 론

PMIPv6는 단말의 핸드오버 시 과도한 시그널 전송 문제와 지연시간 문제를 해결하기 위해 IETF NetLMM WG에서 제안되었고, 최근 PMIPv6 도메인 내에서 효율적인 데이터 처리를 위해 분산제어가 연구되고 있다. 본 논문에서는 최근 연구되는 PMIPv6 도메인 내에서의 분산제어의 영역을 PMIPv6 도메인 간으로 확장하여 이동성 제어에 대해서 분석하였다. SD-GMIP, DD-GMIP는 기존에 연구되었던 SD-PMIP와 DD-PMIP의 지역적인 분산제어를 전역 이동성 제어로 확장하여 연구한 구조이다. 최근에는 다양한 PMIPv6 도메인 간의 핸드오버 기법이 제안되었으나 본 논문에서 제시한 기법이 적용되면 별도의 핸드오버 절차를 수행하지 않더라도 LMA에 대한 정보들이 PIG에 포함되어 있어 끊임없이 네트워크를 이용할 수 있다. 비용측면의 분석을 통하여 기존의 분산 기법의 확장보다 PIG를 이용한 분산 제어 기법의 성능이 훨씬 우수한 것으로 증명된다. 그중에서도 LMA 간 데이터를 이용한 신호 분산 기법보다 LMA 간 시그널링을 이용한 분산 기법의 성능이 더 좋은 성능을 보여준다.

### 참 고 문 헌

- [1] Morgan Stanley Report, Internet trends, available from [http://www.morganstanley.com/institutional/techresearch/pdfs/Internet Trends041210.pdf](http://www.morganstanley.com/institutional/techresearch/pdfs/Internet%20Trends041210.pdf), April 2010.
- [2] S. Gundavelli (Editor) et al., "Proxy Mobile IPv6," RFC5213 August 2008.
- [3] Heeyoung JUNG, Moneeb GOHAR, Ji-In KIM and Seok-Joo KOH, "Distributed Mobility Control in Proxy Mobile IPv6 Networks," Proceedings of IEICE TRANS. COMMUN., Vol.E94-B, No.8, August 2011.
- [4] G. Giaretta, "Interactions between PMIPv6 and MIPv6: scenarios and related issues," draft-ietf-netlmm-mip-interactions-04, June 2009.
- [5] J-H. Na, et. al., "Roaming Mechanism between PMIPv6 Domains," draft-park-netlmm-pmipv6-roaming-01, July 2008.
- [6] N. Neumann, X. Fu, J. Lei, and G. Zhang, "I-PMIP: an inter-domain mobility extension for proxy-mobile IP," IWCMC'09, pp.994-999, June 2009.
- [7] N. Neumann, X. Fu, J. Lei, and G. Zhang, "Inter-Domain Handover and Data Forwarding between Proxy Mobile IPv6 Domains," draft-neumann-netlmm-inter-domain-01, January 2009.
- [8] Y.H. Choi and T.M. Chung, "Using correspondent information for route optimization scheme on proxy mobile IPv6," J. Netw., Vol.5, No.8, pp.984 - 89, August 2010.

- [9] Q. Wu et al., "An Extension to Proxy Mobile IPv6 for Local Routing Optimization," draft-wu-netext-local-ro-05, February 2010.



### 장 순 호

e-mail : prolike@naver.com  
2011년~현 재 성균관대학교 정보통신  
대학원 석사과정  
관심분야: 모바일컴퓨팅, Proxy Mobile  
IPv6, 변전소 통신네트워크,  
USN 네트워크 보안 등



### 정 중 필

e-main : jpjeong@skku.edu  
2008년 성균관대학교 정보통신대학  
(공학박사)  
2009년 성균관대학교 컨버전스연구소  
연구교수  
2010년~현 재 성균관대학교 산학협력단  
산학협력중점교수  
관심분야: 모바일컴퓨팅, 센서 이동성, 차량 모바일 네트워크,  
스마트기기 보안, 네트워크 보안, IT융합, 인터랙션  
사이언스 등