

HMIPv6의 중첩된 이동 네트워크에서 MAP의 캐시 정보를 이용한 명시적 경로 지정(Explicit Path Assignment) 알고리즘

(Explicit Path Assignment(EPA) Algorithm using the Cache Information of MAP in Nested Mobile Network of HMIPv6)

송지영[†] 김병기^{**}
(Jiyoung Song) (Byunggi Kim)

요약 HMIPv6는 노드와 라우터의 이동성으로 중첩된 형태의 이동 네트워크가 형성될 수 있다. 중첩된 이동 네트워크에서는 여러 개의 서브넷이 존재하고 또 여러 개의 이동 라우터(MR)와 이동 노드(MN)가 활동한다. 중첩의 깊이가 커지면 패킷이 경유하는 이동 라우터의 수가 증가하고, 이로 인해 데이터의 전송 지연이 커질 수 있다. 본 논문에서는 Mobility Anchor Point(MAP)의 바인딩 캐시 정보를 활용하여 MAP에서 목적지 이동 노드까지의 경로를 구하고, 그 경로 정보를 패킷의 헤더에 포함하여 처리하는 EPA 알고리즘을 제안한다. EPA 알고리즘을 적용하면 MAP 도메인내의 불필요한 패킷의 양을 줄일 수 있다. 또 중간 이동 라우터에서는 간단한 패킷 치환 작업만으로 패킷의 재전송을 지원할 수 있어 전송 지연을 줄일 수 있다.

키워드 : 중첩 이동 네트워크, HMIPv6, 이동 라우터

Abstract In HMIPv6, the nested mobile network because of the mobility of node and router can be constituted. Many subnets exist and many mobile router(MR)s and mobile node(MN)s activate in the nested depth is deeper, the number of mobile router that packet goes through, increases and data transmission delay owing to this might be larger. This paper proposes EPA algorithm which finds out the path from Mobility Anchor Point(MAP) to a destination mobile node using the binding cache information of MAP and processes the path information by adding it to packet header. If we apply EPA algorithm, the quantity of unnecessary packet within MAP domain can be decreased. Also, the transmission delay can be decreased in a intermediate mobile router because it supports packet re-transmission just by simple packet substitution.

Key words : Nested Mobile Networks, HMIPv6, Mobile Router

1. 서론

HMIPv6는 IETF의 NEtwork MObility Working Group에서 이동노드의 핸드오프 시간과 오버헤드를 줄이는 것을 목적으로 제안되었다. HMIPv6은 지역 Home Agent(HA)인 MAP를 두어 도메인내의 노드들의 주소를 관리하고 목적지 이동 노드까지의 데이터 터

널링을 통해 최적의 경로를 제공해준다[1-3]. 이동 네트워크는 이동 라우터가 이동의 단위가 되며 고정 단말 또는 이동 단말, 다른 이동 네트워크의 서브넷을 포함할 수 있다. 이동 라우터가 새로운 MAP 도메인으로 이동하면 이동 라우터안의 모든 노드는 위치 등록이 필요하다. 중첩된 이동 네트워크는 최상위 이동 라우터를 MAP으로 지정한다. MAP과 이동 라우터의 바인딩 캐시의 정보를 노드의 이동시 마다 갱신함으로써, MAP이 도메인 내의 모든 이동 라우터와 노드들의 위치 정보를 관리 할 수 있도록 한다. MAP은 자신의 도메인내의 모든 노드들과 Correspondent Node(CN)와의 데이터 송신 및 수신 역할 뿐만 아니라, HA와의 바인딩 정보 교환 등 그 역할과 기능이 커지고 있다[4].

· 본 연구는 숭실대학교 교내연구비 지원으로 이루어졌음

† 정 회 원 : 숭실대학교 컴퓨터학과

jysong1@kopo.ac.kr

** 종신회원 : 숭실대학교 컴퓨터학부 교수

bgkim@ssu.ac.kr

논문접수 : 2006년 5월 10일

심사완료 : 2006년 11월 15일

본 논문은 HMIPv6의 중첩된 이동 네트워크 환경에서 MAP이 지역 HA 역할을 하는 점을 이용한다. MAP의 바인딩 캐시 정보를 활용하여 MAP에서 목적지 이동 노드까지의 경로를 구하고, 그 경로 정보를 패킷의 헤더에 포함하여 처리하는 EPA 알고리즘을 제안한다. EPA 알고리즘을 적용하면 MAP 도메인내의 불필요한 패킷의 양을 줄일 수 있다. 또 중간 이동 라우터에서는 간단한 패킷 치환 작업만으로 패킷의 재전송을 지원할 수 있어 전송지연을 줄일 수 있다. 2장에서는 HMIPv6와 관련된 연구에 대하여 알아보고, 3장에서는 본 논문에서 제안하는 변형된 MAP 캐시 구조를 설명한다. 4장에서는 EPA 알고리즘에 대하여 설명한다. 5장에서는 분석 및 성능평가 결과를 보이며, 마지막으로 6장에서 결론을 맺는다.

2. 관련연구

중첩된 이동 네트워크에는 여러 개의 MAP이 존재할 수 있다. MR중에서 액세스 라우터와 직접 연결된 최상의 MR을 MAP으로 지정한다. 최상위 MAP 지정 방법은 HMIPv6를 적용하여 지역 MR과 노드들의 위치 관리 및 데이터 전송의 효율성을 증대시키며, MAP을 기반으로 한 노드들의 독립성을 향상시킬 수 있다. 이 방법은 실제 HMIPv6를 적용하여 이동 네트워크 시스템을 구현할 때 가장 먼저 고려되는 방법이기도 하다. HMIPv6에서 이동 노드는 Router Advertisements (RA)와 MAP option을 수신하여 기존의 MAP 도메인인지 또는 새로운 MAP 도메인인지를 감지한다. 만약, 이동 노드가 새로운 MAP 도메인으로 이동한 경우에는 MAP과 HA, CN에게 Binding Update (BU)를 전송하여 자신의 위치를 알려야 한다[5-7].

IETF에서 제안된 네트워크 이동성 지원 프로토콜은

이동 라우터와 HA 사이에, 양방향 터널링으로 패킷을 전송한다. 그러나 중첩된 이동 네트워크에서 패킷은 이동 라우터와 HA를 모두 거쳐서 전송되는 Pinball Routing 문제가 발생한다. 이것은 중첩의 깊이가 깊어질수록 패킷의 전달 경로가 증가한다. 이로 인해 전송 지연이 늘어나기 때문에 오류의 가능성도 높아진다. 이 문제를 해결하기 Reverse Routing Header(RRH)는 중첩된 이동 네트워크 환경에서 이동 라우터가 자신의 HA에게 Care-Of-Address(CoA)를 최상위 이동 라우터로부터 자신에게 이르는 이동 라우터의 리스트를 알려주는 방식이 있다[8]. [8]에서는 중첩된 이동 네트워크의 최상위 라우터와 이에 속한 모든 라우터들이 각각의 HA 사이에 터널링을 설정하여 최적화된 경로로 데이터가 전송될 수 있도록 제안하였다. 이 밖에도 이동 라우터나 이동 노드가 이동할 때마다 새로 속하게 되는 네트워크의 프리픽스를 할당받아 CoA를 생성하고 자신의 하위에 속하는 모든 이동 네트워크에 광고하는 “Prefix 위임” 기능을 이용하여 경로 최적화 방안을 연구하였다[9].

기존 논문들과 달리 본 논문에서는 지역 HA인 MAP의 바인딩 캐시 정보를 활용하여 명시적 경로를 결정하고, 패킷이 경유하는 중간 이동 라우터의 처리 지연을 최소화하는데 중점을 둔다. MAP의 바인딩 캐시에 저장된 이동 노드의 정보를 활용하여 명시적 경로 지정이 가능함을 보인다. 또 분석과 성능평가를 통해 전송 지연을 줄일 수 있음을 보인다.

3. MAP의 바인딩 캐시 구조 변경

중첩된 이동 네트워크는 이동 노드의 움직임으로 중첩의 깊이가 커질 수 있다. 논문에서는 그림 1과 같은 중첩된 이동 네트워크의 구조를 고려한다. 이동 노드 MN1이 MAP_A 도메인에서 MAP_B 도메인으로 이동

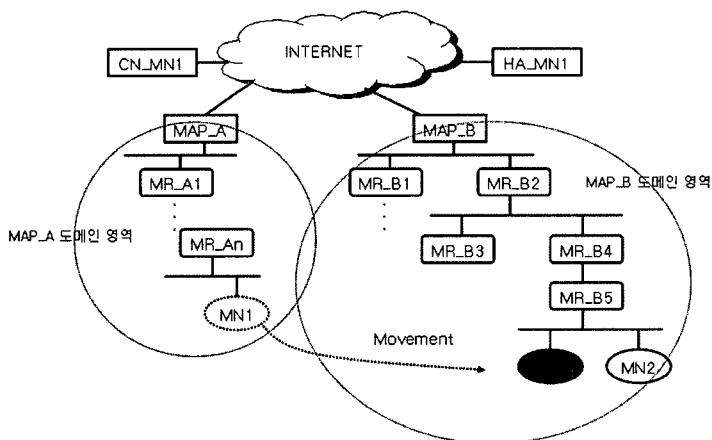


그림 1 중첩된 이동 네트워크의 구조

하는 것을 고려한다.

본 논문에서는 바인딩 캐시의 구조를 변경하여 이동 노드의 상위 노드 정보를 추가로 저장한다. 그림 2와 같이 바인딩 캐시 테이블의 구조는 {Node_ID, Node_RCoA, Node_LCoA, Up_Node_ID, Network Prefix}로 구성한다. 기존의 바인딩 캐시 테이블에 새롭게 추가되는 필드는 Up_Node_ID이다. Up_Node_ID는 이동 노드 자신의 상위 노드(또는 MR)의 정보로서, MAP 도메인 내의 목적지 이동 노드로 전송되는 패킷의 명시적 경로를 결정하는데 이용한다.

그림 2는 MN1이 지역 BU를 수행한 후의 MAP의 바인딩 캐시 테이블의 구조를 보여주고 있다. 이동 노드가 새로운 MAP 도메인으로 진입하게 되면 이동 노드는 무선 링크를 생성한 후에(①), 새로운 이동 라우터로부터 RA 메시지를 수신한 후 On-Link Care-of-Address(LCoA)를 생성한다(②). 그 다음 이동 노드는 Regional Care-of-Address(RCoA)와 LCoA를 구성하여 지역 BU 메시지를 MAP으로 전송한다. 이때 송신 주소는 LCoA를 사용하고, 목적지 주소는 MAP option에서 얻은 Global IP Address를 사용한다(③). 이동 노드는 지역 BU가 성공한 후 HA와 CN에게 BU를 위해 자 HA와 CN에게 RCoA와 홈 주소를 전송한다(④). HMIPv6에서 이동 노드는 위의 3단계를 수행하지만, 본 논문에서는 그림 3과 같이 지역 BU 메시지를 확장하여 상위 노드의 정보를 전송한다. 이를 위해 BU 메시지에 새로운 "U"플래그를 추가한다. "U"플래그를 "1"로 하여(Node_LCoA, Up_Node_ID)를 MAP으로 전송하면,

MAP은 바인딩 캐시 테이블의 상위 노드 정보를 갱신한다(⑤).

그림 1에서 MN1을 포함한 MAP_B 도메인내의 모든 라우터 및 이동 노드들은 지역 BU를 수행하는 경우 4가지 정보를 전송한다. Up_Node_ID 필드는 기존의 바인딩 캐시 테이블에 새롭게 추가된 필드로서 상위 노드의 정보를 저장하며, MAP에서 목적지 이동 노드까지의 명시적 경로를 지정하는데 사용한다. MAP은 INNER 패킷을 수신한 후 패킷의 목적지 노드를 분석하여 MAP에서 목적지 노드까지의 명시적 경로를 결정한다. MAP은 패킷이 목적지 노드로 재전송할 때 경유하는 모든 이동 라우터의 정보를 Type 2 Routing Header(RH2)에 명시하여 전송한다.

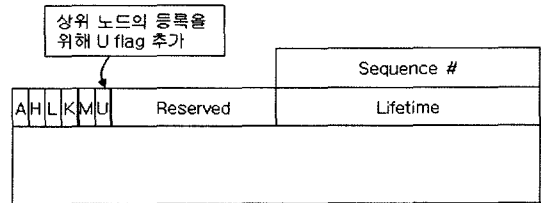


그림 3 상위 노드 정보 전송을 위한 지역 바인딩 메시지 확장

4. MAP 도메인에서의 명시적 경로 지정 방법

이동 네트워크에서 패킷은 이동 라우터 내의 모든 이동 노드로 전송한다. 패킷을 수신한 이동 노드 중 라우

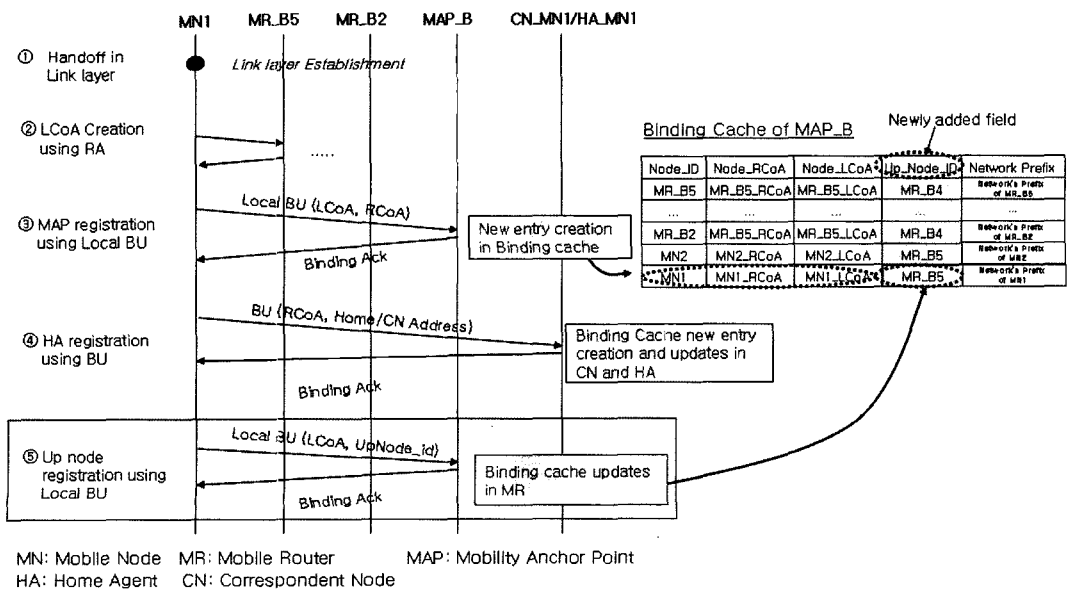


그림 2 이동 노드의 MAP 등록 과정과 바인딩 캐시 구조

터는 목적지가 자신이 아닐 경우 패킷을 재전송 한다. 중첩된 이동 네트워크에서는 여러 개의 이동 네트워크가 존재하므로 불필요한 패킷이 네트워크상에 존재한다. 그림 4의 구조에서는 MAP_B에서 MN1으로 전송되는 패킷은 MR_B1의 서브넷으로 전달하지 않는다. 따라서 불필요한 패킷의 재전송을 줄일 수 있다. 또한 패킷의 헤더에 명시적인 경로를 포함하여 전송하므로, 중간 이동 라우터는 자신의 라우팅 테이블을 검색할 필요가 없어 패킷 헤더의 처리 시간을 줄일 수 있다. 중간 이동 라우터는 MAP에서 지정한 명시적 경로대로 패킷 헤더의 정보를 변경하여 재전송하는 작업을 수행한다.

그림 4에서 MN1의 상대 노드인 CN_MN1이 MN1에게 데이터를 전송한다면 패킷의 전달 경로는 CN_MN1 → MAP_B → MR_B2 → MR_B4 → MR_B5 → MN1이다. MAP은 자신의 도메인으로 들어온 패킷의 최종 목적지를 분석하고, 바인딩 캐시 정보를 이용하여 위와 같은 명시적 경로를 지정하여 재전송한다.

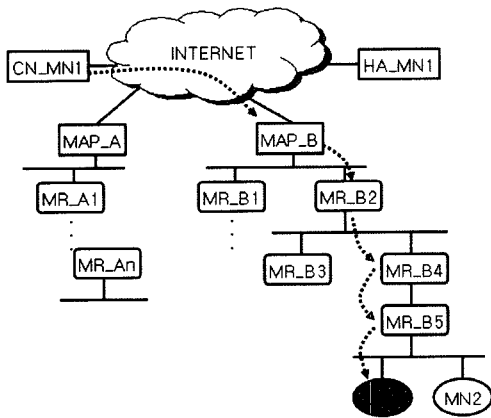


그림 4 CN에서 MN1으로의 패킷 전송 경로

CN_MN1에서 MN1으로 전송되는 패킷은 먼저 지역 HA인 MAP_B로 전달된다. MAP_B는 바인딩 캐시 정보를 활용하여 목적지 노드인 MN1까지의 명시적 경로를 결정한다. MAP_B는 패킷 헤더의 RH2를 확장하여 명시적인 경로를 표시한다. 확장된 RH2는 여러 개의 주소를 가질 수 있으므로 목적지 노드까지의 명시적 경로 표시가 가능하다. MAP에서 MN1까지의 중간 이동 라우터는 패킷의 RH2를 검사하여 다음 이동 라우터나 이동 노드의 LCoA가 있으면 자신의 주소를 패킷의 목적

지 주소와 교체한 후 재전송한다.

4.1 Explicit Path Assignment 알고리즘

그림 4에서 HA_MN1 또는 CN에서 MN1으로 전송한 패킷의 목적지 주소는 MAP_B가 된다. MAP_B에 도착한 패킷은 목적지 주소를 수정한 뒤에 MAP_B도 메인내로 재전송한다. 그림 5는 MAP_B가 수신한 패킷의 헤더를 나타낸다. 목적지 주소는 지역 HA인 MAP_B의 주소이며, 최종 목적지가 MN1임을 나타내고 있다.

다음은 MAP_B에서 MN1까지의 명시적 경로를 결정하는 알고리즘이다. 스택의 크기는 MAP_B의 중첩의 크기만큼 준비한다.

```

Empty a stack:
set src_addr = DEST_field;
set dest_addr = Final_DEST_field;
search (bind_cache, dest_addr)
{
  if (bind_cache.Node_RCoA == dest_addr) { %Part 1.
    get up_node = bind_cache.Up_Node_ID;
    get node_LCoA = bind_cache.Node_LCoA;
    push a stack (node_LCoA);
  }
  if (up_node == MAP) exit;
  else {
    search (bind_cache, up_node) %Part 2.
    {
      if (bind_cache.Node_ID == up_node) {
        get up_node = bind_cache.Up_Node_ID;
        get node_LCoA = bind_cache.Node_LCoA;
        push a stack(node_LCoA);
      }
      if (up_node == MAP) exit;
    }
  }
}
pop a stack to dest_addr; %Part 3.
pop a RH2 until empty a stack ;
    
```

그림 6 EPA 알고리즘

EPA 알고리즘의 동작은 바인딩 캐시에 Up_Node_ID 필드를 추가함으로써 가능하다. MAP은 수신한 패킷의 최종 목적지 노드와 바인딩 캐시의 RCoA를 비교한다. 목적지 노드와 바인딩 캐시의 RCoA가 같으면, 바인딩 캐시 테이블의 Up_Node_ID와 LCoA 정보를 가져온다. 바인딩 캐시의 Up_Node_ID는 up_node에, Node_LCoA는 node_LCoA에 저장된다. up_node는 목적지 노드의 상위 노드의 정보를 검색하는데 이용된다. node_LCoA

SRC	DEST	Final_DEST				
HA_MN1 or CN	MAP_B_ADDR	EXTENSTION	RH TYPE 2	MN1_RCoA	HA_MN1_ADDR	INNER PACKET

그림 5 MAP_B가 수신한 패킷 헤더

는 패킷의 최종 목적지 노드이거나 중간에 경유하는 이동라우터의 LCoA이므로 준비한 스택에 저장한다. 이때 up_node가 MAP이면 알고리즘 수행을 중지하고, 그렇지 않으면 바인딩 캐시에서 up_node의 정보 검색을 계속한다. up_node가 MAP이면 MAP 자신이므로 알고리즘 실행을 끝낸다. 따라서, 스택에는 MN1_LCoA → MR_B5_LCoA → MR_B4_LCoA → MR_B2_LCoA의 순으로 저장된다. 그림 6의 EPA 알고리즘에서 Part 1 부분은 중첩의 깊이가 1인 경우 실행되며, Part 2는 다중 중첩의 경우에 실행하는 부분이다. 마지막으로 Part 3 부분은 스택에 저장된 정보 중 맨 처음 POP 되는 정보는 목적지 주소로 사용되며, 나머지 정보는 RH2 헤더에 순서대로 저장한다. 그림 7은 EPA 알고리즘 실행 후의 스택의 모양과 POP 연산 후 MAP_B에서 캡슐화된 헤더의 모양을 나타낸다.

MAP_B는 EPA 알고리즘 실행 후, 그림 7의 (b)와 같이 패킷의 헤더를 확장 한 후 다음 목적지인 MR_B2로 전송한다. MR_B2를 비롯한 중간 이동 라우터의 RH2 처리 과정은 그림 8과 같다. MR_B2가 MAP_B로

부터 수신한 패킷은 (a)와 같다. MR_B2는 수신한 패킷의 목적지 주소를 송신 주소로 치환하고, RH2 헤더의 첫 번째 주소 필드를 목적지 주소로 치환한다. 따라서, MR_B2는 (b)와 같이 송신주소는 MR_B2가 되고 목적지 주소는 MR_B4로 하여 전송한다. MR_B4도 MR_B2와 동일한 과정을 수행하여 다음 경유지로 전송한다. 패킷이 MN1에 패킷이 도착하면 (d)와 같이 RH2에는 더 이상의 정보가 없으며, MN1은 최종 목적지가 자신임을 인식하고 INNER 패킷을 처리한다.

HMIPv6의 전송 방법은 이동 라우터가 자신의 영역 내의 모든 이동 라우터나 이동 노드에게 패킷을 전송하지만, EPA 알고리즘을 적용하면 목적지로 경유하는 이동 라우터만 패킷을 수신하여 재전송하게 된다. 따라서, 상대적으로 명시적 경로에 포함되지 않는 이동 라우터는 불필요한 패킷을 처리 및 재전송 부담을 줄일 수 있다. 따라서 그림 1과 같은 중첩된 이동네트워크 구조에서는 MN1로 향하는 모든 패킷은 MR_B2로만 전송되며, MR_B1의 모든 서브넷에는 전송되지 않으므로 불필요한 무선 데이터의 오버헤드를 줄일 수 있다.

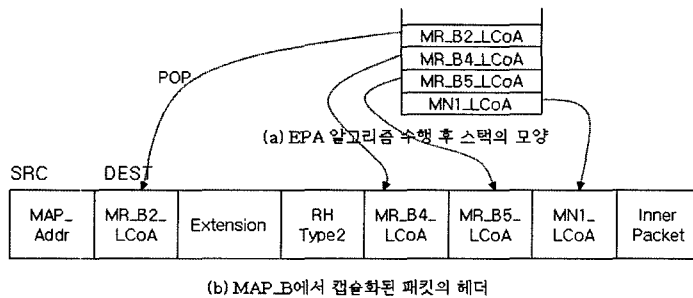


그림 7 스택의 내용과 MAP_B에서 확장 처리된 패킷 헤더

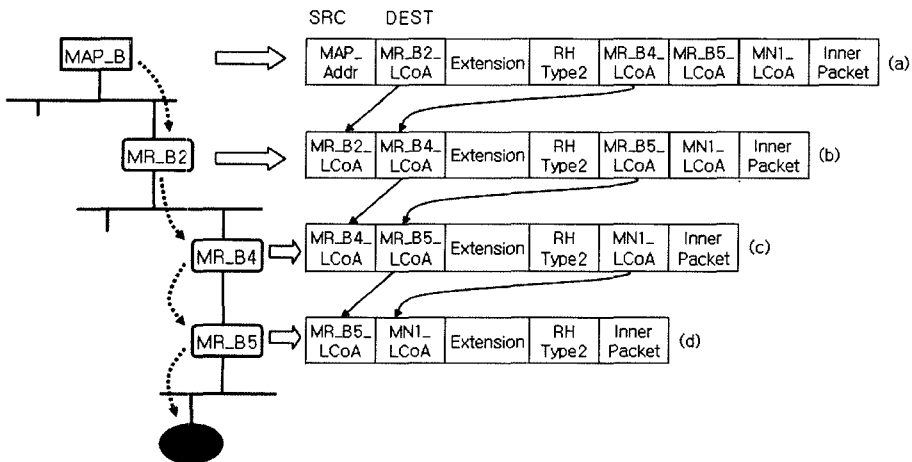


그림 8 중간 이동 라우터의 확장 헤더 처리 과정

5. 분석 및 성능 평가

HMIPv6의 중첩된 이동네트워크에서는 중첩의 깊이에 따라 전송 지연이 결정된다. 본 논문은 중간 이동 라우터의 패킷의 헤더 처리를 단순화하여 전송 지연 시간을 감소시켰다.

본 논문에서 제안한 명시적 경로 지정 방법인 EPA 알고리즘과 기존의 HMIPv6 전송 방식의 비교를 통해 성능을 평가한다. CN에서 목적지 MN까지의 전송 시간(T_{EPA})라고 하면, T_{EPA} 는 CN에서 MAP까지의 전송 시간($T_{CNtoMAP}$)과 MAP에서 목적지 노드까지의 전송 시간($T_{MAPtoMN}$)으로 구할 수 있다.

$$T_{EPA} = T_{CNtoMAP} + T_{MAPtoMN} \quad (1)$$

여기서 $T_{MAPtoMN}$ 는 다음과 같이 구할 수 있다.

$$T_{MAPtoMN} = T_{DecisionPath} + \sum_{i=1}^n i \cdot (T_{HeaderSwitch} + T_{trans} + T_{LinkDelay}) \quad (2)$$

$T_{DecisionPath}$ 는 MAP에서 명시적 경로 알고리즘을 수행하는 시간을 나타내며, n 은 중첩된 이동 네트워크의 깊이를 말한다. $T_{HeaderSwitching}$ 은 중간 이동라우터에서의 발신지 주소와 목적지 주소를 교환하는데 걸리는 시간이다. T_{trans} 는 CN에서 MAP까지의 전송 지연 시간이며, $T_{LinkDelay}$ 는 무선 링크의 전송 지연 시간이다.

T_{EPA} 를 이용하여 전송 시간에 대한 확률식은 다음과 같이 구할 수 있다. T_E 를 MAP에서 MN까지의 전송 시간에 대한 랜덤 변수라 하고, 평균시간 T_{EPA} 를 가지는 지수분포를 따른다고 가정할 때, T_E 에 대한 확률밀도함수 $f_{T_E}(t)$ 는 다음과 같다.

$$f_{T_E}(t) = \begin{cases} \frac{e^{-t/T_{EPA}}}{T_{EPA}}, & t \geq 0 \\ 0, & t < 0. \end{cases} \quad (3)$$

기존의 HMIPv6에서의 CN에서 목적지 MN까지의 전송지연을 T_{HMIP} 이라고 하면, (4)와 같이 구할 수 있다.

$$T_{HMIP} = T_{CNtoMAP} + T_{MAPtoMN} \quad (4)$$

$T_{CNtoMAP}$ 은 상대노드에서 MAP까지의 전송지연 시간이고, $T_{MAPtoMN}$ 은 (5)와 같이 구할 수 있다.

$$T_{MAPtoMN} = T_{BindCache} + \sum_{i=1}^n i \cdot (T_{MRBindCache} + T_{trans} + T_{HeaderProcess} + T_{LinkDelay}) \quad (5)$$

식 (5)에서 $T_{BindCache}$ 는 HMIPv6에서 바인딩 캐시를 탐색하여 MN의 RCoA가 자신의 MAP 도메인 내에 있는지를 검색하는 시간이다. $T_{MRbindCache}$ 는 중간 이동 라우터에서 패킷의 목적지를 분석하여 자신의 라우팅

영역에 있는 MN인지를 검색하는 시간을 말한다. $T_{HeaderProcess}$ 는 중간 이동 라우터에서 라우팅 테이블 검색 후, 패킷의 헤더를 처리하는 시간이다.

T_H 를 기존 HMIPv6의 확률변수라고 하고 평균시간 T_{HMIP} 을 가지는 지수분포를 따른다면 T_H 에 대한 확률 밀도 함수는 다음과 같다.

$$f_{T_H}(t) = \begin{cases} \frac{e^{-t/T_{HMIP}}}{T_{HMIP}}, & t \geq 0 \\ 0, & t < 0. \end{cases} \quad (6)$$

본 논문에서는 제안된 EPA 알고리즘과 기존의 HMIPv6의 전송지연에 따른 블로킹 확률을 분석한다. 이동 노드가 새로운 MAP이나 이동 라우터로 진입하여 패킷을 전송 받은 경우, 이동 노드의 평균 전송 지연 허용 시간에 대한 파라미터를 T_{dt} 라 한다면, EPA 방식의 호 블로킹 확률은 다음과 같이 구할 수 있다.

$$P_{EPA} = \Pr\{T_E > T_{dt}\} = \int_0^{\infty} P\{T_E > t\} f_{T_E}(t) dt \quad (7)$$

$$= \int_0^{\infty} [1 - F_{T_E}(t)] \cdot f_{T_E}(t) dt$$

또 기존의 HMIPv6 방식의 전송 지연에 따른 호 블로킹 확률은 다음과 같이 구한다.

$$P_{HMIP} = \Pr\{T_H > T_{dt}\} = \int_0^{\infty} P\{T_H > t\} f_{T_H}(t) dt \quad (8)$$

$$= \int_0^{\infty} [1 - F_{T_H}(t)] \cdot f_{T_H}(t) dt$$

본 논문에서는 이동 노드의 전송 지연의 시간에 대한 분포를 지수분포와 감마 분포를 고려하였다[10,11]. 이동 노드가 전송 지연에 대해 지수 분포를 보인다고 가정하면, 이에 대한 확률밀도 함수는 다음과 같다.

$$f_{T_{\alpha-\exp}}(t) = \begin{cases} \mu e^{-\mu t}, & t \geq 0 \\ 0, & t < 0. \end{cases} \quad (9)$$

여기서 $1/\mu$ 는 이동 노드의 평균 전송 지연이다.

다음으로 T_{dt} 가 감마분포를 보인다면, T_{dt} 에 대한 확률 밀도 함수는 다음과 같다.

$$f_{T_{\alpha-\gamma}}(t) = \begin{cases} \frac{\eta e^{-\eta t} (\eta t)^{w-1}}{\Gamma(w)}, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (10)$$

여기서 w 는 감마분포에 대한 shape 파라미터이며, η 는 감마분포의 scale 파라미터로 $\eta = \sigma w$ 로 나타내고, $\Gamma(w)$ 는 감마함수로 다음 식과 같다. 또한 위의 감마분포에 대한 확률 밀도 함수는 w 가 정수일 때 Erlang 분포가 된다.

$$\Gamma(w) = \int_0^{\infty} e^{-t} (t^{w-1}) dt. \quad (11)$$

따라서, EPA 알고리즘과 HMIPv6의 전송지연으로 인한 호 블로킹 확률 P_{EPA} 와 P_{HMIP} 의 전송 지연에 대

한 시간 함수를 얻을 수 있다. 먼저 제안한 EPA 알고리즘의 P_{EPA} 의 전송지연에 대한 시간의 함수 $f_{T_{a-exp}}(t)$ 와 $f_{T_{a-gam}}(t)$ 는 다음의 (12)와 (13) 같이 구해진다.

$$\begin{aligned}
 P_{EPA(exp)} &= \Pr\{T_E > T_{dt}\} = \int_0^\infty P\{T_E > t\}f_{T_a}(t)dt \quad (12) \\
 &= \int_0^\infty e^{-t/T_{EPA}} \cdot f_{T_{a-exp}}(t)dt \\
 &= \int_0^\infty \mu e^{-t/T_{EPA}} \cdot e^{-t\mu} dt \\
 &= \frac{\mu T_{EPA}}{1 + \mu T_{EPA}}
 \end{aligned}$$

그리고

$$\begin{aligned}
 P_{EPA(gam)} &= \int_0^\infty e^{-t/T_{EPA}} \cdot f_{T_{a-gam}}(t)dt \quad (13) \\
 &= \int_0^\infty e^{-t/T_{EPA}} \cdot \frac{\eta e^{-nt}(\eta t)^{w-1}}{\Gamma(w)} dt
 \end{aligned}$$

그다음 기존의 HMIPv6에서 P_{HMIP} 의 전송지연에 대한 시간의 함수 $f_{T_{a-exp}}(t)$ 은 다음과 같이 얻을 수 있다.

$$\begin{aligned}
 P_{HMIP(exp)} &= \Pr\{T_H > T_{dt}\} = \int_0^\infty P\{T_H > t\}f_{T_a}(t)dt \\
 &= \int_0^\infty e^{-t/T_{HMIP}} \cdot f_{T_{a-exp}}(t)dt \\
 &= \int_0^\infty \mu e^{-t/T_{HMIP}} \cdot e^{-t\mu} dt \\
 &= \frac{\mu T_{HMIP}}{1 + \mu T_{HMIP}} \quad (14)
 \end{aligned}$$

그리고 전송지연에 대한 감마분포에 대한 $f_{T_{a-gam}}(t)$ 는 다음과 같다.

$$\begin{aligned}
 P_{HMIP(gam)} &= \int_0^\infty e^{-t/T_{HMIP}} \cdot f_{T_{a-gam}}(t)dt \quad (15) \\
 &= \int_0^\infty e^{-t/T_{HMIP}} \cdot \frac{\eta e^{-nt}(\eta t)^{w-1}}{\Gamma(w)} dt
 \end{aligned}$$

여기서 EPA 알고리즘과 기존의 HMIPv6의 수식을 비교하면, $T_{CNtoMAP}$ 는 MAP 도메인의 외부 환경으로 동일한 파라미터라고 할 수 있다. 제안한 EPA 알고리즘은 MAP에서 MN까지의 경로인 식 (2)와 식 (4)의 $T_{MAPtoMN}$ 부분에 대한 자세한 분석이 필요하다.

$T_{DecisionPath}$ 는 MAP에서 목적지 MN까지의 명시적 경로를 결정하는 EPA 알고리즘을 수행하는데 걸리는 시간이며, $T_{BindCache}$ 는 MAP에 도착한 패킷을 분석하여 목적지 필드의 RCoA가 MAP의 바인딩 캐시에 있는지를 검색하는 시간이다. 두 개의 파라미터 모두 수행시간에는 큰 차이가 없다. $T_{HeaderSwitching}$ 는 EPA 알고리즘의 파라미터로 중간 이동 라우터가 수신한 패킷의 헤더 부분을 치환하는데 걸리는 시간이며, HMIPv6의 $T_{MRbindCache}$ 는 중간 이동라우터에서 수신한 패킷을 분석하고, 라우팅 테이블을 검색하여 패킷의 헤더를 다시 생성($T_{HeaderProcess}$)하는데 필요한 시간이다. 따라서 $T_{HeaderSwitching}$ 보다는 $T_{MRbindCache}$ 가 더 많은 시간을 필요로 한다. 무선 링크의 지연시간인 $T_{LinkDelay}$ 같다고 볼 수 있다. 결국 전송지연은 T_{EPA} 보다 T_{HMIP} 이 더 크다고 할 수 있다. 또한 이동 네트워크의 중첩의 깊이가 커질수록 T_{EPA} 보다 T_{HMIP} 의 차이는 더 커진다.

시뮬레이션은 IEEE 802.11b 기반의 환경으로 하였으며, 시스템 파라미터는 표 1과 같다.

그림 9는 이동 네트워크의 중첩의 깊이에 따른 전송지연을 보여주고 있다. 중첩이 1인 경우, 즉 중첩 네트워크

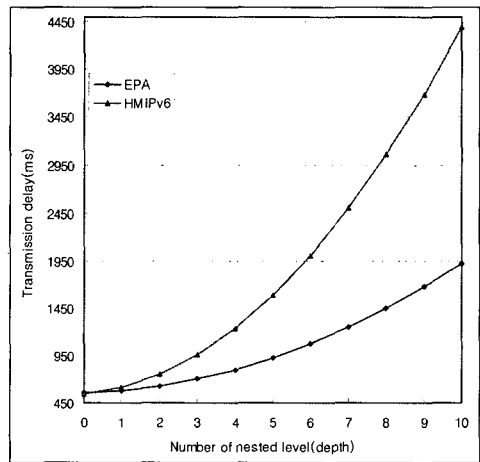


그림 9 중첩의 깊이에 따른 전송 지연

표 1 시스템 파라미터

파라미터	값	파라미터	값
트래픽 형식	UDP	$T_{CNtoMAP}$	500ms
$T_{DecisionPath}$	50ms	$T_{HeaderSwitching}$	5ms
T_{Trans}	payload/rate	$T_{LinkDelay}$	20ms
$T_{BindCache}$	40ms	$T_{MRbindCache}$	40ms
$T_{HeaderProcess}$	10ms	n	0~10
전송속도	2Mbps	메시지 크기	100byte

워크가 아닌 경우에는 기존의 HMIPv6가 10ms 더 작지만, 중첩의 깊이가 커질수록 제안한 EPA 알고리즘이 더 작은 전송 지연을 보여주고 있다. 중첩의 깊이가 6이상이면 기존 HMIPv6의 전송 지연 시간은 EPA 보다 2배 이상 커짐을 볼 수 있다.

P_{EPA} 와 P_{HMIP} 에 대한 지수 분포의 호 블로킹 확률 $P_{EPA(exp)}$ 와 $P_{HMIP(exp)}$ 은 시스템 파라미터를 이용하여 식 (12)과 (14)에 의해 얻어진다. 그림 10은 중첩의 깊이가 2와 4일 때 지수 분포의 호 블로킹 확률을 보이고 있다. 제안한 EPA 알고리즘이 기존의 HMIPv6보다 호 블로킹 확률이 작음을 알 수 있다. 중첩의 깊이가 커질수록 호 블로킹 확률 차이가 커짐을 볼 수 있다. 그림 11은 중첩의 깊이가 6과 8일 때의 결과를 보여주고 있다.

본 논문에서는 MAP에서 이동 노드까지의 전송 지연에 대하여만 고려하였으며, 이를 바탕으로 전송지연에 따른 호 블로킹 경우를 가정하였다. 앞서도 언급한 바와 같이 만일 w 가 정수라면 감마 분포는 Erlang 분포를 보인다. P_{EPA} 와 P_{HMIP} 에 대한 감마 분포의 호 블로킹 확률 $P_{EPA(gam)}$ 와 $P_{HMIP(gam)}$ 은 시스템 파라미터를 이용하여 식 (15)과 (16),(17),(18)에 의해 얻어진다. 따

라서 $w=1$ 인 경우 $P_{EPA(gam)}$ 를 나타내는 식 (15)는 다음과 같이 기술되어진다.

$$P_{EPA(gam)} = \int_0^\infty e^{-t/T_{EPA}} \cdot \frac{\eta e^{-nt}(\eta t)^0}{\Gamma(1)} dt \quad (15)$$

$$= \int_0^\infty \eta e^{-t/T_{EPA}} \cdot e^{-nt} dt$$

$$= \frac{\sigma T_{EPA}}{1 + \sigma T_{EPA}}$$

그리고 $w=2$ 인 경우는 아래와 같다.

$$P_{EPA(gam)} = \int_0^\infty e^{-t/T_{EPA}} \cdot \frac{\eta e^{-nt}(\eta t)}{\Gamma(2)} dt \quad (16)$$

$$= \frac{\eta^2}{\Gamma(2)} \int_0^\infty t \cdot e^{-t(1/T_{EPA} + \eta)} dt$$

$$= \left(\frac{2\sigma T_{EPA}}{1 + 2\sigma T_{EPA}} \right)^2$$

위와 마찬가지로 기존의 HMIPv6에 대하여 $w=1$ 과 $w=2$ 인 경우는 식 (17)과 (18)로 각각 기술 할 수 있다.

$$P_{HMIP(gam)} = \int_0^\infty e^{-t/T_{HMIP}} \cdot \frac{\eta e^{-nt}(\eta t)^0}{\Gamma(1)} dt \quad (17)$$

$$= \int_0^\infty \eta e^{-t/T_{HMIP}} \cdot e^{-nt} dt$$

$$= \frac{\sigma T_{HMIP}}{1 + \sigma T_{HMIP}}$$

$$P_{HMIP(gam)} = \int_0^\infty e^{-t/T_{HMIP}} \cdot \frac{\eta e^{-nt}(\eta t)}{\Gamma(2)} dt \quad (18)$$

$$= \frac{\eta^2}{\Gamma(2)} \int_0^\infty t \cdot e^{-t(1/T_{HMIP} + \eta)} dt$$

$$= \left(\frac{2\sigma T_{HMIP}}{1 + 2\sigma T_{HMIP}} \right)^2$$

그림 12와 그림 13은 감마 분포에서 $w=1$ 일 때 중첩의 깊이(n)별로 EPA 알고리즘과 기존의 HMIPv6의 실험 결과를 보이고 있다. 제안한 알고리즘의 호 블로킹 확률이 더 낮음을 볼 수 있다. 그림 13에서는 $n=8$ 인 EPA 알고리즘이 $n=6$ 인 HMIPv6보다 호 블로킹 확률이 낮음을 보여주고 있다. 이는 중첩의 깊이가 커질수록 제안한 EPA 알고리즘의 성능이 좋음을 알 수 있다.

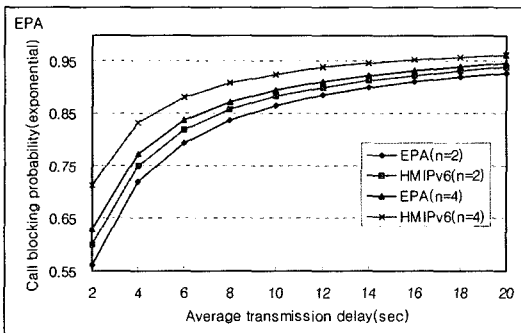


그림 10 평균 전송 지연에 대한 지수 분포의 호 블로킹 확률($n=2, n=4$)

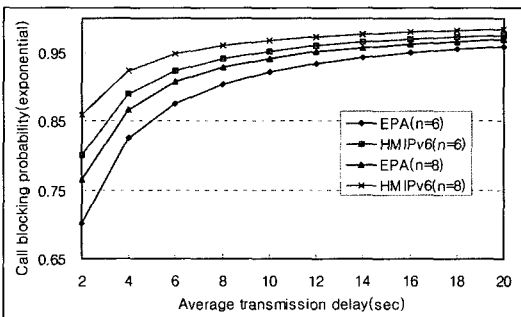


그림 11 평균 전송 지연에 대한 지수 분포의 호 블로킹 확률($n=6, n=8$)

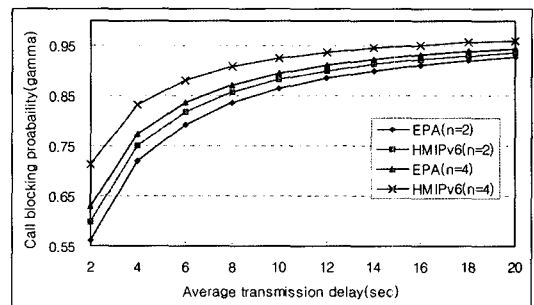


그림 12 평균 전송 지연 시간에 대한 호 블로킹 확률 ($w=1, n=2, n=4$)

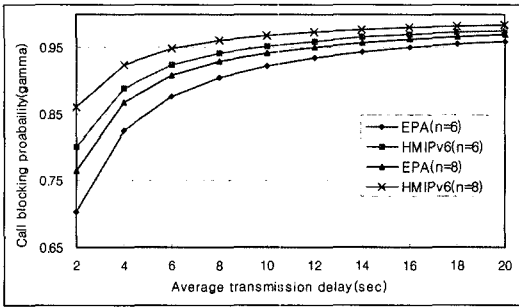


그림 13 평균 전송 지연 시간에 대한 호 블로킹 확률 ($w=1, n=6, n=8$)

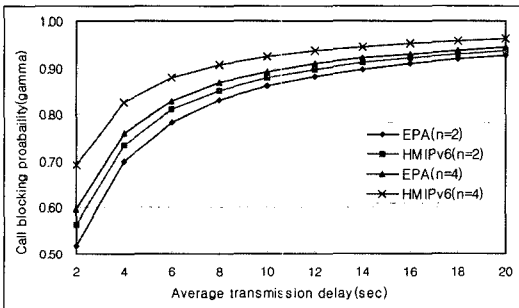


그림 14 평균 전송 지연 시간에 대한 호 블로킹 확률 ($w=2, n=2, n=4$)

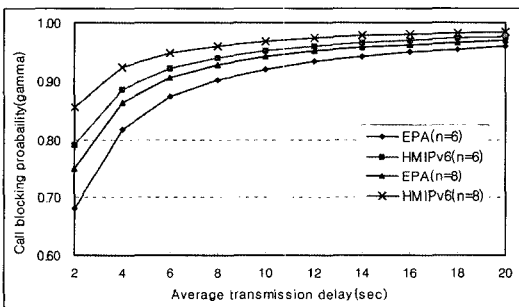


그림 15 평균 전송 지연 시간에 대한 호 블로킹 확률 ($w=2, n=6, n=8$)

그림 14와 그림 15는 감마 분포에서 $w=2$ 일 때의 중첩의 깊이에 따른 호 블로킹 확률을 보여주고 있다. $w=1$ 일 때와 마찬가지로 제안한 EPA 알고리즘의 호 블로킹 확률이 낮음을 볼 수 있다.

6. 결론

본 논문에서는 HMIPv6 기반의 중첩된 이동 네트워크 환경에서 MAP의 바인딩 캐시 정보를 활용하여 MAP에서 목적지 이동 노드까지의 명시적 경로 지정하

는 EPA 알고리즘을 제안하였다. 명시적 경로를 결정하기 위해 MAP의 바인딩 캐시 구조를 변경하여 이동 노드의 상위 노드 정보를 저장하도록 하였다. MAP은 EPA 알고리즘을 이용하여 MAP에서 목적지 이동 노드까지의 명시적 경로를 결정한 후, RH2의 확장 헤더에 전송 경로를 저장하여 재전송한다. 중간 이동 라우터에서는 패킷의 목적지 주소와 RH2 헤더의 간단한 치환으로 헤더 처리 과정을 단순화하였다. HMIPv6의 라우팅 테이블 검색 및 패킷 재조립 처리 시간을 줄임으로써 전송 지연 시간을 감소시켰다. 또 성능평가를 통해 EPA 알고리즘이 이동 네트워크의 중첩의 깊이가 커질수록 기존의 HMIPv6 보다 전송지연이 현저히 작고, 호 블로킹 확률이 낮음을 보였다. 향후 이동 네트워크인 점을 감안하여 이동 라우터 및 이동 노드의 이동성 및 움직임을 고려한 MAP 정보 관리 방안과 본 알고리즘을 적용하기 위해 추가로 소요되는 시그널링 분석에 대한 연구가 필요하다.

참고 문헌

- [1] V. Devarapalli, A. Petrescu, R. Wakikawa, P. Thubert, "Network Mobility (NEMO) Basic Support Protocol," IETF, RFC3963, 2005.
- [2] H. Soliman, C. Castelluccia, K. El Malki, L. Bellier, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)," IETF, RFC4140, 2005.
- [3] C. Castelluccia, "HMIPv6 : A Hierarchical Mobile IPv6 Proposal," ACM SIGMOBILE Mobile Computing and Review, Vol.4, No.1, pp.48-59, January, 2000.
- [4] E. Perera, V. Sivaraman, A. Seneviratne, "Survey on network mobility support," ACM SIGMOBILE Mobile Computing and Communications Review, Vol.8, No.2, pp.7-19, April, 2004.
- [5] Li-Wei Huang, Jen-yi Pan, Ko-Shung Chen and Chiao Yu, "Hierarchical Mobility Management in IPv4/IPv6 Co-existed Networks," ICWN'05, pp.1023-1028, June, 2005.
- [6] Tao Lin, Chunjian Pan and Ziaiang Hou, "An efficient mobility management mechanism in Hierarchy Mobile IPv6 networks," IEEE VTC'04, pp.4974-4977, September, 2004.
- [7] Taewan You, Sangheon Pack, and Yanghee Choi, "Robust Hierarchical Mobile IPv6(RH-MIPv6)," IEEE VTC'03, pp.2014-2018, October, 2003.
- [8] P.Thurbert, M.Molteni, "draft-thurbert-nemo-reverse-routing-header-01.txt," Internet Draft, IETF, 2002, Work in progress.
- [9] Kyeong-Jin Lee, Jae-Hoon Jeong, Jung-Soo Park, Hyoung-Jun Kim, "Route Optimization For Mobile Nodes in Mobile Network based on Prefix Delegation," Internet Draft, IETF, 2004.

- [10] F. Khan, and D. Zeglache, "Effects of cell residence time distribution on the performance of cellular mobile networks," *IEEE VTC'97*, pp.949-953, May, 1997.
- [11] Y. Xiao, and J. Rosdahl, "Throughput and Delay Limits of IEEE 802.11," *IEEE Communication Letters*, Vol.6, No.8, pp.355-357, August, 2002.



송·지·영

1999년 청운대학교 컴퓨터과학과(공학사). 2001년 숭실대학교 대학원 컴퓨터학과(공학석사). 2001년~현재 숭실대학교 대학원 컴퓨터학과 박사과정. 2003년~현재 한국폴리텍대학 조교수. 관심분야는 이동통신, HMIPv6, 인터넷 QoS, 정보보호



김·병·기

1979년 한국과학기술원 전산학과 석사
 1997년 한국과학기술원 전산학과 박사
 1979년~1982년 경북대학교 전자공학과 전임강사. 1982년~현재 숭실대학교 컴퓨터학부 교수. 관심분야는 유비쿼터스 디지털방송, HMIPv6, 멀티캐스트 프로토콜