HMIPv6 환경에서 에너지 효율적인 이동성 관리 기법

(Energy-Efficient Mobility Management Schemes in HMIPv6)

양 순 옥 † 김 성 석 † 황 종 선 †††
(Sun Ok Yang) (SungSuk Kim) (Chong-Sun Hwang)

요 약 Mobile IP에서 사용자의 이동성을 지원하기 위해 여러 종류의 메시지 -바인딩 갱신 메시지,바인딩 요청 메시지와 바인딩 응답 메시지- 가 사용된다. 끊김 없도록 이동성 관리를 하기 위해서는 그러한 메시지가 빈번하게 교환되어야 하지만, 이는 네트워크의 부하 증가 및 이동 노드의 에너지 효율성 악화를 유발한다. 따라서 서버는 사용자의 위치 추적을 수행하면서 위의 문제를 효과적으로 해결하기 위한 알고리즘이 필요하게 되었으며, 이것은 본 논문에서 주요한 대상이 된다. HMIPv6 환경에서 각 사용자는 과거 이동 로그를 지역적으로 저장한 후, 주기적으로 프로파일을 만든다. 이후 어느 영역으로 이동할 때마다 그 지역에서의 예상 상주 시간을 계산하는데, 이 프로파일 정보를 이용하여 메시지의 라이프타임으로 사용하도록 한다. 물론, 프로파일에서 이동 지역별 상주 시간뿐만 아니라 도착 시간까지 고려할 경우, 보다 정확한 라이프타임 값을 얻을 수 있다. HMIPv6 환경에서 제안한 기법들과 기존 기법 사이의 성능을 비교하기 위해 바인딩 갱신 메시지의 대역폭을 측정하는 다양한 실험을 수행하였다. 그 결과 이동 노드가 임의의 서브넷에서 13분 이상 상주시 80%이상의 이득을 얻을 수 있었다. 즉, 기존 기법과 동일하게 사용자의 위치를 관리하면서, 바인딩 갱신 메시지 수를 획기적으로 줄임으로써 네트워크 및 이동 단말의 에너지 효율성을 크게 향상시킬 수 있었다.

키워드 : 계층적 MIPv6, 바인딩 갱신, 이동 로그, 라이프타임, 이동성 패턴, 이동성 유형.

Abstract In Mobile IP, several types of messages - binding update, binding request and binding acknowledgement - are used to support user mobility. It is necessary to exchange those messages frequently for seamless mobility but it incurs both the increase of network overhead and poor usage of mobile node battery power. Thus, we need a mechanism that the server detects users location and also copes with the problems effectively, which is our main concern in this paper. Each user records all moving logs locally and periodically makes out profile based on them in HMIPv6. By using profile, estimated resident time can be computed whenever he enters an area and the time is set up as the binding update message lifetime. Of course, the more correct lifetime may be obtained if arrival time as well as average resident time is considered in profile. Through extensive experiments, we measure the bandwidth usage for binding update messages by comparing the proposed schemes with that in HMIPv6. From the results, Gain gets over 80% when mobile node stays more than 13 minutes in a subnet. Namely, we come to know that our schemes improve network usage and energy usage in mobile node by decreasing the number of messages while they also manage users locations like that in HMIPv6.

Key words : Hierarchial MIPv6, Binding Update, Moving Log, Lifetime, Mobility Pattern, Mobility Type

· 본 연구는 대학 IT연구센터 육성 · 지원사업의 연구결과로 수행되었음
† 비 회 원 : 고려대학교 컴퓨터학과
soyang@korea.ac.kr
†† 정 회 원 : 서경대학교 전자상거래학과 교수
(corresponding author임)
sskim@skuniv.ac.kr
††† 종신회원 : 고려대학교 컴퓨터학과 교수
hwang@disys.korea.ac.kr
논문접수 : 2004년 8월 12일
심사완료 : 2005년 5월 18일

1. 서 론

최근 몇 년 사이에 기술수준이 높은 통신 수단이 등장하면서 무선 네트워크 통신가입자 수가 급성장하였고, 언제 어디서나 통신하고자 하는 사용자의 요구가 점차 현실화되어 가고 있다. 게다가, 노트북과 PDA의 가격이 낮아지면서, 이동통신 시스템에서 멀티미디어 서비스에

대한 요구도 점차 증가하고 있으며, 이러한 경향과 더불어 인터넷 사용자들은 이동 중에도 인터넷에 지속적으로 연결되어 통신이 원활하게 이루어지길 원한다.

사용자의 이동성을 지원하는 MIPv6[1]에서 이동 노드(MN:Mobile Node)는 자신의 현재 위치를 알려주기 위해 한 장소에서 다른 장소로 이동할 때마다 바인딩 갱신(BU:Binding Update) 메시지를 홈 에이전트(HA:Home Agent)에게 보냄으로써, 이동 중에도 원활한 통신이 가능하게 한다. BU 메시지와 바인딩 요청(BR:Binding Request) 메시지는 MN과 상대 노드들(CNs:Correspondent nodes) 사이에 계속적인 연결 상태(connection) 유지 및 인터넷 위상(topology)내에서 노드들의 이동을 허용하기 위해 사용된다. MN이 BR을 수신하면, CN에게 응답으로 BU를 보내며, 최소 1초에서 최대 10초 간격으로 HA와 CN에게 주기적으로 BU를 재전송한다. 또한, BU 전송시 MN과 HA 또는 MN과 CN 사이에 연결 상태를 체크하기 위해서 라이프타임(lifetime)이 사용된다. 그런데, BU의 라이프타임이 짧으면, HA는 MN의 접속 단절을 빨리 알아낼 수 있으나, 그것은 무선 링크에서 BU 메시지의 엄청난 증가 뿐만 아니라 그것을 처리하기 위한 HA의 오버헤드 또한 증가시킨다. 이는 MN의 에너지 효율과 무선 대역폭 이용율을 악화시키는 주요한 원인 중 하나가 된다. 반면, BU의 라이프타임이 길면, HA와 CN들의 바인딩 캐시에 라이프타임이 갱신 될 때까지 바인딩 정보를 유지하게 된다. 이 경우, MN과 HA 사이에 BU 메시지 수는 감소시킬 수 있으나, 만약 현재 MN이 통신 중에 네트워크에 접속단절이 발생하면, 라이프타임이 갱신될 때까지 HA가 이 상황을 알아채지 못하고, 계속 패킷을 전송하여 패킷이 모두 손실되는 문제가 발생한다. 따라서, 만약 적절한 라이프타임이 설정되어 적용된다면, 이러한 문제를 해결할 수 있다.

만약, MN의 이동성 패턴에 대한 정보를 얻을 수 있다면, 현재 위치를 고려한 미래의 이동 행위(behavior) 예측이 가능하다. 우리는 이전 연구[2]에서, 동적으로 BU의 라이프타임을 결정하기 위해 MIPv6에서 프로파일 기반 전략(PBS:profile based strategy)을 제안하였다. 이 연구를 기반으로 본 논문에서는 HMIPv6[3]에서 사용자 이동성 패턴을 기반으로 하는 메시지 관리 기법을 제안한다. 이 기법에서 각 MN은 주기적으로 방문한 모든 도메인에 대해 라이프타임 값을 계산하고 프로파일에 그것을 유지 관리한다. MN이 임의의 서브넷을 방문할 때, 프로파일에 그 서브넷에 대한 정보가 존재한다면, 그 계산된 라이프타임을 BU의 라이프타임으로 적용한다. 만약 적절한 정보가 없을 경우, 기존 HMIPv6에서 사용하던 라이프타임 값을 그대로 적용한다.

또한, 방문한 서브넷 뿐만 아니라 서브넷 도착 시각에 따라서도 규칙적인 상주시간 패턴이 나타날 수 있다. 예를 들면, 사무직 근로자들은 같은 사무실이더라도 아침에 도착하여 근무하는 시간보다는 점심 먹은 후에 사무실에 돌아와서 더 오래 머무른다. 따라서, 우리는 각 서브넷 별로 도착 시간대를 고려한 확장된 기법도 함께 제안한다. 실험에선 HMIPv6와 제안 기법의 효율성을 비교 분석한다.

본 논문의 구성은 2장에서 관련 연구를, 3장에서 HMIPv6에서 프로파일 기반 라이프타임 할당 기법을 살펴본다. 그리고 4장에서는 BU 메시지 수를 고려하여 HMIPv6와 제안 기법의 실험 결과를 살펴보고, 5장에서 제안 기법의 결론을 제시한다.

2. 관련 연구

이동성 관리에 대한 연구가 PCS 환경에서는 많이 진행되었다. 사용자의 이동성을 반영한 프로파일 기반의 위치 전략을 제안하는 [4]은 네트워크에 부하를 주는 시그널량을 감소시키기 위해 제안되었다. 이 전략은 사용자가 가장 많이 발견된 위치부터 가장 적게 발견된 위치까지 순차적으로 등급을 부여한 프로파일을 시스템이 관리한다. 그리고, 프로파일이 생성된 후, 착호가 이동 단말에 발생하면 프로파일에 있는 각 위치를 순차적으로 패이징 함으로써, 기존에 지리적 기반의 위치 추적 구조에 비하여 대역폭과 시그널량이 감소하는 것을 보여주고 있다. 이동 단말의 이동 행위에 기반한 위치 추적 전략을 제안하는 [4]는 각 단말의 이동 행위로부터 시간대별 이동 확률을 계산한 후, 각 시간대별 최적의 패이징 영역을 찾아내는 기법을 제안하고 있다. 그러나, 이동 행위만을 고려했을 뿐, 각 위치별 상주 시간에 대한 고려는 하고 있지 않다.

반면, Mobile IP 환경에서는 이동성 관리에 대한 초기 단계의 연구가 진행 중이다[5]. [6]는 MIPv6에서 사용자의 이동성을 고려하여, 마이크로 영역(micro-area)과 매크로 영역(macro-area)으로 단순히 2가지 영역으로만 분리하여, 위치 정보 갱신을 중간 이동 에이전트(mobility agent)에게만 할 것인지, HA 및 CN까지 할 것인지를 구분하고 있다. 이 계층형 구조에서 두 개의 CoA를 관리하기 위해 두 개의 라이프타임을 부여한다. 만약 MN의 임시 주소가 만료되고, 중간 이동 에이전트에게 부여된 라이프타임이 만료되지 않으면, CN은 패킷을 중간 이동 에이전트에게 보낸다. 즉, 마이크로 영역 내에서만 이동을 하면, 중간 이동 에이전트에게만 BU 메시지를 보냄으로, HA 및 CN에게 보내는 것보다 BU 메시지의 전송 거리를 단축시키고 있지만, MN에서 발생하는 BU 메시지 수는 변함이 없다. 즉, MIPv6에서 경로를 최적화하기 위해 MN의 지역성을 이용하지만, 각 MN마다 고

유의 BU 메시지의 라이프타임 값을 결정하는 방법은 설명하지 못한다. 따라서, 본 논문에서는 HMIPv6에서 MN의 이동성 패턴을 고려하여 두개의 라이프타임을 선택적으로 적용하여, 효율적으로 BU와 BR 메시지의 수를 감소시키고자 한다.

3. 프로파일 기반의 라이프타임 할당 기법

이 절에서는 HMIPv6 환경에서 사용되는 BU 메시지와 제안 기법에서 사용되는 새로운 유형의 BU 메시지를 소개하고, 지역적인 프로파일 정보를 이용하여 BU 라이프타임을 할당하기 위한 알고리즘을 살펴보고자 한다.

3.1 바인딩 갱신 메시지

일정 서브넷에 비교적 규칙적인 이동성 패턴을 보이는 사무직 직원, 가정주부, 소매상인 등을 고려해 보자. 이들은 일정 시간 동안 그 서브넷을 떠나지 않는다. 반면, 택시기사, 세일즈맨 등은 완벽하게 불규칙한 움직임을 보인다. 만약, 과거 이동과 관련된 이러한 정보를 각 MN이 지역적으로 유지관리 한다면, MN이 서브넷에 진입할 때마다 적절한 BU 라이프타임을 부여하는 것이 가능하다.

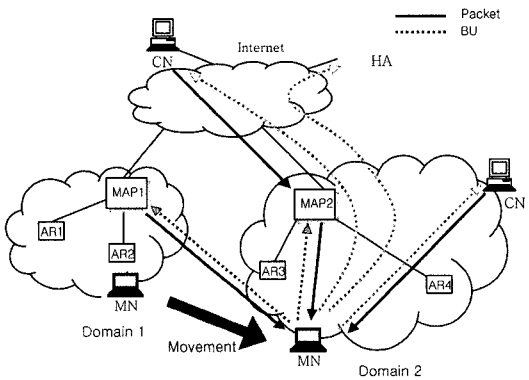


그림 1 HMIPv6에서 기본적인 라우팅 연산

그림 1은 지역적 이동성과 전역적 이동성을 차별화시킨 HMIPv6 환경을 보여주고 있다. 이것이 가능하도록 이동성 앵커 포인트(MAP:Mobility Anchor Point)를 도입하였고, MN과 CN이 통신을 하기 위해 두개의 주소가 사용된다. 하나는 도메인의 주소(RCoA:regional care-of-address)이고 다른 하나는 외부 서브넷에서 얻은 MN의 지역적 주소(LCoA:Local Care-of-Address)이다. 만약, MN이 분리된 MAP 도메인에 들어가게 되면, [3]에 설명된 방법에 의해 액세스 라우터(AR:Access Router)에 의해 라우터 광고(RA:Router Advertisement)를 받고 LCoA와 RCoA를 생성 후, 새로운 BU를 MAP과 HA와 CN들

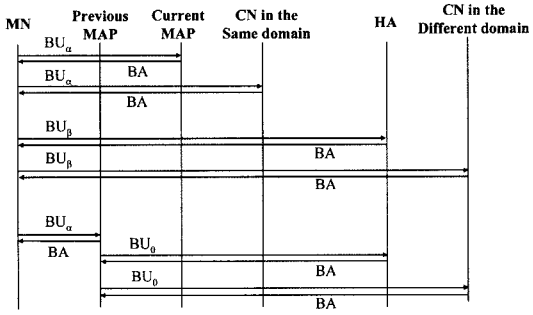


그림 2 바인딩 갱신 과정

에게 보낸다. MAP은 HA와 CN들이 보낸 모든 패킷들을 인터셉트한 후, MN의 LCoA에게 터널(tunnel)한다. 또한, 그림 2는 MN이 새로운 도메인으로 이동시, 발생되는 BU 메시지를 보여주고 있다. 프로파일 관련 정보가 있는 서브넷인 경우 외부 도메인의 CN과 HA에게는 BU_β를 보내고, 이후 BU_β가 만료하기 전에 MN이 서브넷을 떠날 경우에는 CN과 HA의 캐쉬에 남아있는 바인딩 정보를 삭제하기 위해 BU_0를 보내게 된다. 이외의 경우에는 모두 BU_α를 보낸다. 라이프타임 값에 따라 사용되는 세 종류의 BU 메시지는 다음과 같다.

(1) ***BU$_\alpha$***는 기존 MIPv6[1]에서 사용되는 라이프타임 값과 동일한 기본 라이프타임(LT_α)을 갖는다. MN이 새 MAP 영역으로 진입 후, MN은 이전 MAP에게 들어오는 모든 패킷을 새 CoA로 전달해 줄 것을 요청하는 BU_α를 보낼 수 있다.

(2) ***BU$_\beta$***는 지역 프로파일에 기반에서 계산된 라이프타임(LT_β)을 가지며, MN이 MAP과 HA 그리고 CN들에게 보낸다.

(3) ***BU$_0$***는 라이프타임 값(LT_0)이 0이며, 기존 MIPv6[1]에서 바인딩 캐시 정보를 삭제하기 위해서 사용된다. BU_β가 만료되기 전에 또 다른 도메인의 서브넷으로 이동할 때, MN은 HA와 CN들의 캐시에 남아 있는 부정확한 BU_β 정보를 제거하기 위해 사용 된다.

3.2 라이프타임 계산

MN이 임의의 서브넷을 떠날 때마다 그 방문과 관련된 정보, 즉 *이동 로그(Moving log)*를 기록한다. 이 로그에 각 서브넷을 방문할 때마다 서브넷 식별자와 도착시간 및 출발 시간의 정보가 정렬된 (l, AT, DT)형태로 저장된다. 각 서브넷 방문시 평균 상주 시간은 이동 로그를 이용하여 주기적으로 계산하여 프로파일에 저장한다. 그리고, 그 값은 BU_β을 위한 적응적 라이프타임을 계산하기 위해서 사용된다. 이러한 계산 과정에서 서브넷별 로그의 개수를 고려하여 방문 횟수가 경계값($Count_b$)보다 적은 로그는 신뢰할 만한 정보를 제공하지

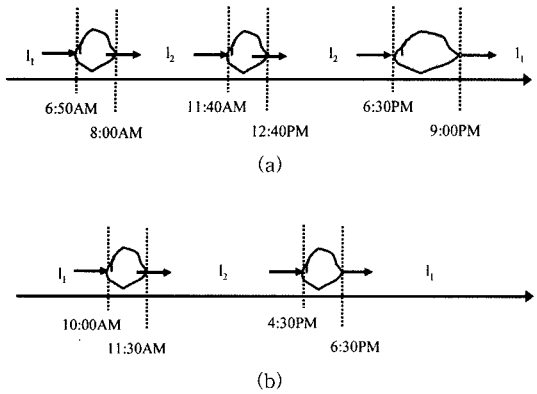


그림 3 규칙적인 이동 패턴의 예

못한다고 판단하고, 현재의 계산에서 배제한다. 의미 있는 라이프타임 값을 얻기 위해서 이동 로그 데이타로부터 규칙적인 이동 패턴이 존재해야 하는데, 그림 3은 몇몇 서브넷에서 MN의 일정 이동 패턴이 나타난 예를 보여주고 있다.

그림 3(a)의 경우, 회사원은 직장으로 출근하기 위해 l_1 위치에서 6:50 a.m.에 출발하여, 8:00 a.m.에 l_2 위치에 도착하고, 11:40 a.m.까지 그 곳에 상주하며 일을 한 후, 점심을 먹기 위해 예측할 수 없는 임의의 장소로 불규칙하게 이동한다. 12:40 p.m.에 l_2 위치로 되돌아온 후, 6:30 p.m.까지 그곳에 계속 머물며 일을 한다. 일을 마치고 사무실을 출발하여, 거의 규칙성을 찾아보기 힘든 임의의 장소에서 시간을 보낸 후 마지막으로 9:00 p.m.에 l_1 위치에 되돌아와 다음날 아침까지 그곳에 상주한다. 각 방문별로 보면, 약간의 차이는 있지만 각 위치에서 상주하는 시간은 거의 유사하다. 또한, 주부(그림 3(b))의 경우도 회사원의 이동 패턴과 유사하게 설명되며, 단지 자주 방문하는 서브넷의 수나 평균 상주 시간에서 약간의 차이가 있을 뿐이다.

결과적으로, 만약 이동 로그로부터 임의의 규칙성이 발견되면, 다음번 서브넷 방문시 상주 시간을 예측하기 위해 각 서브넷별 평균 상주 시간은 매우 유용할 수 있다. 물론, 택시 기사나 세일즈맨 같이 매우 불규칙한 이동성 패턴의 특성을 가지고 있어 일정한 상주 시간을 가지고 있지 않으므로 그림 3에 나타내지 않았지만, 이와 같은 경우도 고려되어야 한다. 다음 절에는 현재 서브넷에서 평균 상주 시간만을 고려한 기법과 상주 시간 뿐만 아니라 방문 시간대까지 고려한 확장 기법에 대해 살펴본다.

3.2.1 상주 시간 기반 기법(*RT*)

적응적 라이프타임(LT_β)을 계산 할 때, 각 서브넷별 평균 상주 시간과 로그의 수가 고려되어야 한다. MN이

```
// MN이 임의의 서브넷을 떠날때 마다 계산

If ( MN have visited subnet m ) {
t_n = DT_n - AT_n
If ( t_n > ρ * LT_α ){
        Record moving log
        Count_m = Count_m + 1
        Sum_m = Sum_m + t_n
        }
}

//주기적인 계산

Mean_m = Sum_m / Count_m
For ( all moving logs to subnet m ){
        Calculate Var_m          //식 (2)
        Determine mobility type
        }
If ( Count_m < Count_b ) LT = LT_α
Else{
    If ( Mean_m <= ρ * LT_α or mobility type C )
            LT = LT_α
    Else
            LT = Mean_m *V       // LT_β
    }
```

그림 4 상주 시간 기반 기법

서브넷 m으로부터 또 다른 서브넷으로 이동할 때, 현재 방문에 대한 이동 로그를 이용하여 모든 방문 서브넷에 대한 평균 상주 시간을 주기적으로 계산하는 *상주 시간 기반 기법(RT : resident time based scheme)*은 그림 4에서 볼 수 있다. 우선, 서브넷 m에 n번째 방문에 대한 상주 시간(t_n)은 식(1)과 같이 출발 시간(DT)에서 도착 시간(AT)을 뺀 값이 된다.

$$t_{nm} = DT_{nm} - AT_{nm} \quad (1)$$

그림 4에서 t_n과 $\rho * LT_\alpha$의 비교는 상주 시간이 작은 값, 즉, MN이 목적지까지 가는 도중에 서브넷을 잠시 거쳐가는 경우들을 이동 로그에서 배제하기 위한 것이다. 이론적으로, t_n과 LT_α를 비교하는 것이 합리적이만, 이렇게 하면 LT_α 값(즉, LT_α = 1~10 초)이 너무 작아 성능 차이가 매우 크다. 따라서, 우리는 t_n과 LT_α의 길이에 $\rho(\rho \geq 1)$배한 값을 비교한다. Sum_m과 $Count_m$은 각각 서브넷 m에 총 상주 시간과 총 방문 횟수를 의미한다. 계산 과정에서 만약 서브넷 m에 방문 횟수가 상수 값($Count_b$)보다 적으면, 서브넷 m에 있는 규칙성을 발견하기 곤란하기 때문에 BU_α가 사용된다. 또한, 계산된 평균 상주 시간은 단지 평균 값일 뿐, MN이 다음번 이동시 프로파일을 얼마나 잘 따를 것인가에 대한 보장이 없음으로 그대로 라이프타임 값에 적용하는 것은 곤란하다. 따라서, 본 논문에서는 평균 상주 시간뿐만 아니라, 분산(varience)을 고려한다. 즉, 모든 서브넷들을 프

로파일의 정확도에 따라 이동성 패턴을 이동성 유형 A, B, C로 분류한다. 이동성 유형 A는 MN의 프로파일 정보를 매우 신뢰할 수 있는 경우인 반면, 이동성 유형 C는 현재의 서브넷에서 규칙적이 이동성 패턴을 거의 찾아 볼 수 없는 경우이다. 그 차이를 정량화하기 위해 모든 서브넷에 대한 분산은 식 (2)을 이용하여 계산한다.

$$Var_m = \frac{1}{n}\sum_{i=1}^{n}(t_i - Mean_m)^2 \qquad (2)$$

만약, Var_m이 상수 δ_1 보다 적으면, 서브넷 m은 이동성 유형 A로 분류되며, MN이 그 서브넷에 다시 방문할 때, 계산된 평균 상주 시간이 높은 신뢰성을 가진다고 볼 수 있으므로 일정 비율$(0<V<1)$에 높은 값을 부여한다. Var_m이 상수 δ_1 보다 크고 상수 δ_2 보다 적으면, 서브넷 m은 이동성 유형 B로 분리된다. 이는 그 서브넷에서의 상주 시간에 편차가 큼을 의미하며, 따라서 계산된 평균 상주 시간을 부분적으로 이용해야 한다. 따라서, 이동성 유형 A보다 다소 낮은 값을 V에 부여한다. 그러나, Var_m이 상수 δ_2 보다 큰 경우 이동성 유형 C에서는 프로파일로부터 신뢰성 있는 정보를 제공받지 못하므로 차라리 BU_α 값을 이용한다. 이처럼 이동성 유형에 따라 서로 다른 상수 값(V)이 평균 상주 시간에 곱해져서 BU의 라이프타임 값이 결정된다. 프로파일 생성 후 MN이 서브넷 m을 방문할 때, 계산된 LT_β는 BU_β의 라이프타임 값이 된다. 만약, LT_β가 만료되기 전 MN이 또 다른 도메인 내에서 임의의 서브넷으로 이동하게 되면, MAP은 HA와 외부 CN들에게 BU_0를 보내야 한다. 그러나, 그 이외의 경우에 MN은 현재 네트워크의 프리픽스(prefix)와 MN의 주소의 프리픽스를 비교하여 두 값이 서로 다른 경우에만 BU_α을 보낸다.

3.2.2 시간대 기반 기법(*TR*)

몇몇 서브넷별 상주 시간은 종종 도착 시간에 의존한다. 예를 들면, 회사원은 일반적으로 사무실에서 오전보다는 오후에 더 오래 동안 상주한다. 이 경우, 우리는 상주 시간 기반 기법(*RT*)을 확장한 도착 시각의 시간대를 고려한 *시간대 기반 기법(TR : time region based scheme)*을 제안한다.

그림 5는 서브넷 m에서 도착 시간과 출발 시간을 나타내는 시간 다이어그램의 예를 보여주고 있다. Y축은

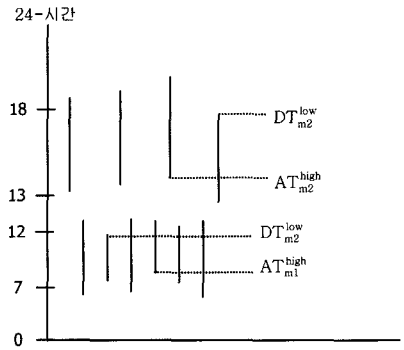


그림 5 서브넷 m에 MN의 도착 시간을 나타내는 시간 다이어그램

밤 12시를 0으로 하여 하루 24시간을 시간 단위로 나타낸다. 각 수직선은 이동 로그를 나타내는 것으로 그것의 길이는 상주 시간을 의미한다. 즉, 각 수직선의 아래쪽 끝은 매 방문별 도착 시간을, 위쪽 끝은 출발 시간을 의미하며, 기호 $AT_{mn}{}^{high}$와 $DT_{mn}{}^{low}$은 아래에서 설명된다. 이 그림에서 MN이 오전과 오후에 자주 서브넷을 방문하며, 같은 시간대이더라도 평균 상주 시간이 일치하지 않는다는 사실을 알 수 있다. 즉, 주기적인 계산 과정에서 단순히 서브넷만 아닌 *(서브넷 ID, 시간대)*별 평균 상주 시간까지 고려되어야 한다. 도착 시간대를 고려하면 하나의 서브넷에 여러 가지 평균 상주 시간이 존재할 수 있고, 이를 해결하기 위해 이동 로그로부터 시간대를 결정하기 위한 알고리즘이 필요하다. 그림 6은 고려되어야 할 5가지 경우를 보여주고 있다.

점선은 계산된 하나의 시간대를 나타내며, 실선은 현재의 방문 시간, 즉, 이동 로그에 기록된 시간을 나타낸다. 다음의 정보는 또한 각 도착 시간대별 프로파일에서 유지 관리되는 정보이다:

- $AT_{mn}{}^{high}$: 가장 늦은 도착 시간
- $DT_{mn}{}^{low}$: 가장 빠른 출발 시간
- $Count_{mn}$: n번째 시간대에 포함되는 방문 횟수
- $TotalCount_{mn}$: n번째 시간대에 고려되는 총 방문 횟수

위에서 n과 m은 서브넷 m에 n번째 도착 시간대를 나타내는 첨자이다. 본 논문은 방문 서브넷 뿐만 아니라

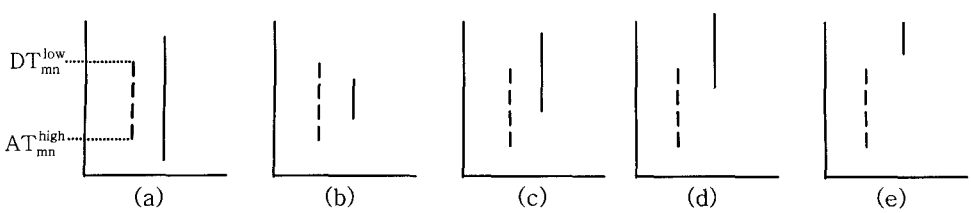


그림 6 다양한 방문 시간 사례

시간대를 고려하므로, 각 시간대별 방문 횟수($TotalCount_{mn}$, $Count_{mn}$)를 구분하여 관리한다. 그림에서 점선은 AT_{mn}^{high}과 DT_{mn}^{low}사이의 시간 간격을 나타낸다.

그림 6(e) 경우는 현재 상주 시간이 방문 시간대를 벗어나므로 시간대에 포함시키지 않는다. 또한, 그림 6(a), (b) 경우는 서브넷에서 MN이 너무 오래 또는 너무 짧게 머무는지를 체크할 필요가 있다. 만약, 상주 시간이 3/2 * $Interval_{mn}$(단, $Interval_{mn} = DT_{mn}^{low} - AT_{mn}^{high}$)보다 크거나 또는 1/2 * $Interval_{mn}$ 보다 작으면, 두 상주 시간의 차이가 너무 크기 때문에, 신뢰할 수 있는 정보를 제공할 수 없으므로 이러한 방문 로그는 완전히 배제된다. 게다가, MN의 상주 시간 길이 뿐만 아니라 도착 시간까지 고려되어야 한다. 그림 6(c), (d) 경우, 언뜻 보면 유사한 것처럼 보인다. 그러나, 이 두 경우를 모두 동일 시간대로 고려하면, 상주 시간대는 신뢰도가 떨어지는 정보를 제공하게 된다. 따라서, 정확도 있는 결정을 하기 위해 다음과 같은 처리 과정이 필요하다.

```
middle = Interval_mn /2
if ( middle ≥ 현재의 도착 시간)
        현재 방문 로그는 n번째 시간대에 포함된다
else
        현재 방문 로그는 배제된다
```

그 다음 만약 현재 로그가 서브넷 m의 n번째 시간대에 포함되면, $Count_{mn}$와 $TotalCount_{mn}$ 모두 1씩 증가시키고, 그렇지 않으면, 프로파일의 정확도를 결정하기위해 사용될 $TotalCount_{mn}$만 1 증가 시킨다(그림 6(a), (b) 참조). 물론, 만약 현재 방문 도착 시간이 AT_{mn}^{high}보다 빠르면, 그림 6(c), (d), (e)와 정반대의 경우로 고려되어야 한다. 알고리즘에서 배제된 이동 로그는 그림 6(a), (b) 경우를 제외하고는 또 다른 시간대 형성에 사용된다.

이와 같은 방식으로 도착 시간대가 결정 된 후, 이동성 유형을 결정하기 위한 대부분의 알고리즘은 *RT*와 유사하지만, 한 가지는 예외이다. *RT*에서는 방문한 서브넷별로 Var_m만 고려되었으나, *TR*에서는 시간대를 고려한 정보가 얼마나 유용한가를 평가하기 위해서 $Count_{mn}/TotalCount_{mn}$과 Var_m를 비교할 필요가 있다. 즉, Var_m이 c_1보다 적고 $Count_{mn}/TotalCount_{mn}$이 τ_1보다 크면, 그 도착 시간대는 이동성 유형 A로 판단된다. 그러나, 만약 Var_m이 c_2보다 크고 $Count_{mn}/TotalCount_{mn}$이 τ_2보다 작으면, 이동 패턴에 규칙성이 없는 것으로 판별한다(이동성 유형 C). 그 이외의 평균 상주 시간은 이동성 유형 B가 된다.

3.3 보안 및 접속단절

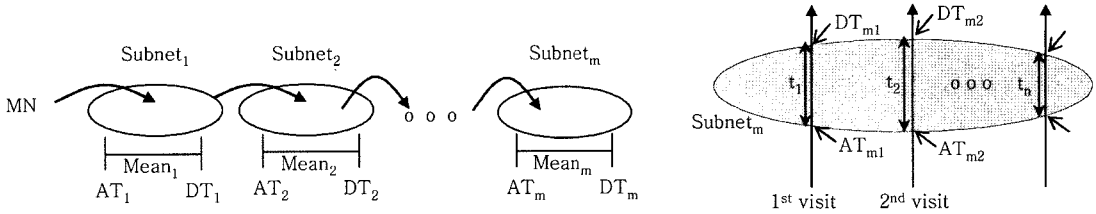
알고리즘 개발과정에서 고려해야할 사항은 보안 및 접속단절이다. 우선, 무선 환경에서는 악의적인 사용자에 의해 접속설정이 침해당할 수 있으며, BU 라이프타임이 길게 설정될 경우 그 사실을 오랫동안 모를 수 있다. 하지만, 이 경우 Mobile IP의 보안(security)과 관련된 연구 결과 [7]를 적용하면 자연스럽게 해결할 수 있다. [7]의 저자들은 일반적인 BU 메시지에 *Lifetime Credit Authorization option*을 추가하여, 안정적인 노드 즉, 특정 장소에 계속 머물러 있는 노드인 경우 *Kcredit*을 사용하여 라이프타임을 최대 8시간까지 확장 하고 있다. 하지만, 본 논문에서는 프로파일 정보를 이용하여 예상 상주 시간만큼만 *Kcredit*을 적용하면 되기 때문이다.

두 번째 접속단절의 경우, 패킷 손실을 고려대야 한다. 즉 CN이나 HA는 BU의 라이프타임 동안 MN에게 패킷을 보내지만 접속단절로 인하여 보내진 패킷은 전달되지 못한다. 그리고, 라이프타임이 길게 설정된다면 손실되는 패킷의 양이 커지게 된다. 이에 대하여 다음과 같이 두 가지로 나누어 생각할 수 있다. MN이 다른 영역으로 이동하는 경우, 새로운 BU 메시지를 전송해야 하므로 접속단절을 알 수 있게 된다. 반면, 한 영역 내에서 통신 중인 경우, 불가피하게 CN이나 HA로부터의 패킷을 수신하지 못하게 된다. 비록 손실된 양은 차이가 있지만, 접속단절로 인한 손실은 이전 Mobile IP에서도 고려 대상이였으며, 손실된 정보는 재전송 요청과 같은 방법을 사용하여 정보 손실을 보전할 수 있다. 그러나, 오늘날 발달된 이동 통신 기술로 인하여 정지 상태에서는 자발적으로 전원을 끄는 경우 이외는 접속 단절이 발생할 확률이 매우 낮다. 이는 접속단절 때문에 발생하는 패킷의 손실이 매우 적다는 것을 의미한다. 따라서 본 논문에서 이 문제를 논외로 한다.

4. 성능 평가

4.1 시뮬레이션 환경

그림 7은 본 논문에서 제안한 기법의 성능 평가를 위해 설정한 시뮬레이션 환경을 보여 준다. 이 그림에서 MN은 서브넷을 이동할 때마다 (l, AT_n, DT_n)를 포함한 로그 데이타, 즉, 서브넷의 주소, 도착 시간, 출발 시간을 로그로 남긴다. 여기서, $Mean_m$은 MN이 각 서브넷 m을 방문할 때마다 머무른 평균 상주 시간을 의미한다. *TR*의 경우, 각 MN은 *RT*와 동일한 로그 데이타를 수집하고, 추가로 도착 시간대와 관련된 데이타도 수집한다. 또한, MN이 각 서브넷에 매번 방문할 때 머무는 상주 시간은 분포에 모양을 결정하는 쉐이프 파라미터(shape parameter) α를 가진 감마분포를 따른다고 가정한다. 감마분포를 선택한 이유는 다른 분포에 비해 일반적인 경우를 표현할 수 있을 뿐만 아니라, 측정된 데이타가 특



(a) RT 모델 (b) TR 모델

그림 7 시뮬레이션 모델

정 분포를 따른다고 가정할 수 없는 경우에 적합하기 때문이다[8]. 감마분포에서 확률밀도 함수와 평균, 분산은 각각 다음과 같다.

$$f(t)=\frac{\lambda^{\alpha}}{\Gamma(\alpha)}(\lambda t)^{\alpha-1}\ell^{-\lambda t}, t \geq 0 \quad (3)$$

$$E(t)=\frac{\alpha}{\lambda} \quad (4)$$

$$V(t)=\frac{\alpha}{\lambda^2} \quad (5)$$

위의 식에서 확률변수 t는 상주 시간을 의미하며, 각 서브넷별로 매번 방문할 때마다 머무는 상주 시간의 평균은 식 (4)이고, 동일한 평균에서 분산을 다르게 했을 때 성능 차이를 비교하기위해 사용된 분산은 식 (5)와 같다. 일반적으로 이동 통신 시스템에서 상주 시간을 모델링하기위해 지수분포를 많이 사용하는데, 감마분포 확률밀도 함수 식 (3)에 α=1, λ=1/$E(t)$을 대입하면, 지수분포가 된다. 이와 같이 상주 시간을 감마분포와 지수분포 두 가지 로 가정한 후, MN으로부터 발생하는 BU 메시지 수를 측정하여 성능을 비교한다.

앞 절에서 상수로 설정한 파라미터의 값은 표 1과 같다. 물론, 사용자의 프로파일 정보의 파라미터들은 다양하게 존재할 수 있으며, 실험 결과에 여러 가지로 영향을 미칠 수 있다. 하지만, 본 논문에서는 사용자의 규칙적인 상주 시간에 따라 사용자의 이동성을 분류하여 각각 다른 라이프타임을 적용하는데 초점을 맞추고 있다. 따라서 이동성 유형과 관련 있는 파라미터 중심으로 기술하였다. 만약, 표 1의 파마미터 값 중에서 δ_1이 증가하거나 τ_1이 감소하는 경우 이동성 유형 A가 증가하게 된다. 그리고 δ_2가 증가하거나 τ_2가 감소하는 경우 이동성 유형 C가 감소하게 된다. 즉, 파라미터 값의 이러한 변화는 실험 결과의 이득을 증가시키는 효과가 있다. 반면, V_A와 V_B의 감소는 이동성 유형 A와 이동성 유형 B로 판별된 BU 라이프타임에 적용되는 평균 상주 시간의 반영 비율을 감소시켜 실험 결과의 이득을 감소시키는 효과가 있다.

4.2 시뮬레이션 결과

성능을 평가하기위해 MN이 임의의 서브넷에 상주하는 동안 기존 HMIPv6와 제안 기법을 적용한 경우 발생하는 BU 메시지에 할당된 총 대역폭을 비교하였다. 본 논문은 메시지 전달 과정에서 에러가 발생하지 않는다고 가정한다. 그리고, 결과에 대한 자세한 설명을 하기 전에 LT_a 값(즉, LT_a = 1~10초)이 적은 것에 주목해야 한다. 프로파일에서 규칙적인 이동성 패턴이 발견되는 것은 LT_a보다 상당히 오래 상주한다는 것을 의미한다. 즉, HMIPv6보다 제안 기법이 훨씬 성능이 좋을 것으로 예측할 수 있다. 따라서, 단순히 BU 메시지 수만을 비교하는 대신 HMIPv6와 제안 기법 사이의 BU 대역폭을 이용한 성능향상도(*G: Gain*)를 측정한다. G의 정의는 다음과 같다:

$$G=(BW_{HMIPv6}-BW_{proposal})/BW_{HMIPv6} \quad (6)$$

이 식에서 BW_{HMIPv6}과 $BW_{proposal}$은 각각 HMIPv6와 제안 기법에 할당된 대역폭을 의미한다. 제안 기법의 BU 발생량이 HMIPv6보다 매우 적으므로, G는 거의 1.0에 가까운 값이 된다.

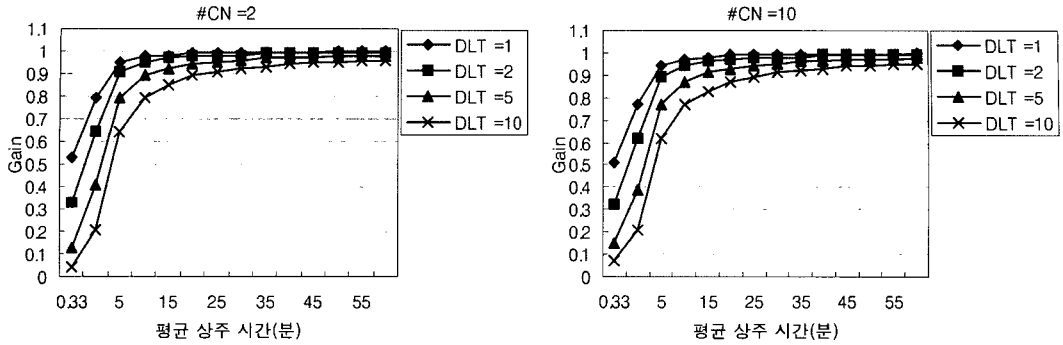
$$BW_{HMIPV}=Size_{BU}\times\kappa\times f_{ref}\times(f_{CN}\times(\#CN+1)) + (1-\kappa)\times(M\times CN+2) \quad (7)$$

$$BW_{proposal}=Size_{BU}\times\{\kappa\times(f_{RT-REF}\times(\#CN+1)) + (1-\kappa)\times(M\times CN+2)+f_{ADD}\}$$

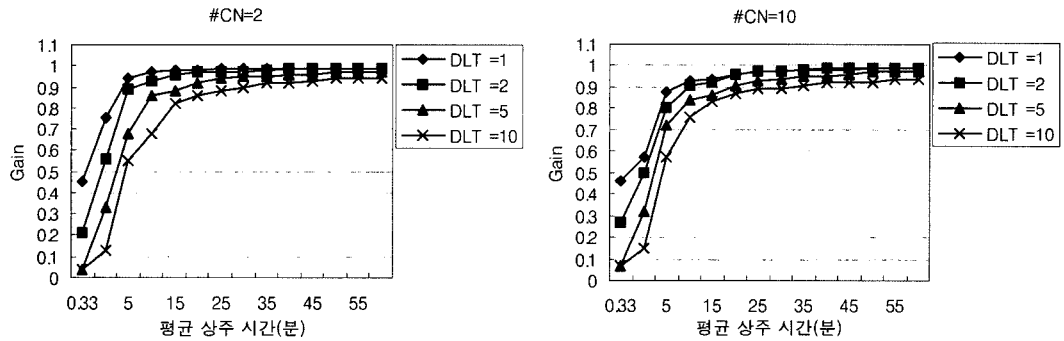
$$f_{ADD}=\begin{cases}\#CN+1 & \text{if } currently\ visited\ subnet\ is\ in\ the\ profile\\ 0 & otherwise\end{cases} \quad (8)$$

표 1 파라미터 값 세팅

파라미터	값	
$Count_b$	10	Count의 임계 값
δ_1	10	이동성 유형 A를 결정하기 위한 상수 값
δ_2	50	이동성 유형 C를 결정하기 위한 상수 값
τ_1	0.8	이동성 유형 A를 위한 $Count_{mn}/TotalCount_{mn}$의 상수 값
τ_2	0.6	이동성 유형 C를 위한 $Count_{mn}/TotalCount_{mn}$의 상수 값
κ	0.3	도메인 내 이동 율
V_A	1.0	이동성 유형 A의 V값
V_B	0.8	이동성 유형 B의 V값
ϕ	0.001	접속 단절 율



(a) 감마분포



(b) 지수분포

그림 8 RT에서 BU 대역폭 비교

식 (7)과 (8)에서 *#CN*은 현재 CN들의 수를 의미하고, $Size_{BU}$는 BU의 크기(68bytes = IPv6 header(40 bytes) + Binding Update Extension Header(28bytes))로 정의된다[5]. κ는 MN의 도메인 내부에서의 이동 비율을 의미하며, 도메인 간의 이동 비율은 1-κ로 도메인간 이동수를 서브넷을 이동한 총 수로 나눈 값이다. 도메인간 이동시 MN은 M번 연속적인 BU를 외부 도메인에 있는 CN들에게 보내고, 또 다른 BU를 HA에게 보낸 후 HA로부터 BA를 수신한다. 식 (7)에서 f_{REF}는 외부 도메인 내에서 서브넷을 따라 이동 할 때, MN이 외부 CN들과 HA에게 보내는 BU의 갱신 빈도를 나타낸다[9]. 식 (8)에서 $f_{RT\text{-}REF}$는 MN이 지역적 프로파일의 정보를 이용할 때 보내는 갱신 빈도를 나타내며, MAP에서 외부 CN들과 HA에게 추가적으로 f_{ADD}를 보내야 한다. 그러나, MN이 도메인간 이동일 때, BU에 할당된 대역폭은 두 기법이 동일하다. LT_a를 증가시키면서 감마분포에서 *RT*와 HMIPv6의 비교 결과는 그림 8에서 보여 주고 있고, 지수분포에서의 결과는 그림 9에서 보여 주고 있다. 본 논문에서 프로파일에 기록된 서브넷의 15%를 이동성 유형 A로, 또 다른 서브넷의 30%를 이동성 유형 B로, 나머지 55%를 이동성 유형 C로 가정한다.

외부 도메인 내에서 지역적으로 이동하는 경우, BU 대역폭은 *#CN*과 $f_{RT\text{-}REF}$에 의존한다. 그림 8에서 평균 상주 시간이 증가함에 따라 *G*가 거의 1.0에 접근하는 것을 볼 수 있다. 그 이유는 큰 값의 평균 상주 시간을 프로파일로부터 얻으면, BU의 라이프타임이 길게 설정되기 때문이다. 물론, 평균 상주 시간과 실제 라이프타임 사이에 차이는 매우 클 수 있다. 하지만, 이동성 유형 A 또는 B로 결정된 일부 서브넷에서 절약되는 대역폭의 양은 결코 적은 양이 아니다. 특히, MN이 빈번하게 이동하지 않음에도 불구하고, 대부분의 시그널링 양은 위치 갱신을 위한 BU에 의해서 발생된다. 따라서, 제안 기법에서 MN으로부터 주기적인 위치 갱신 BU 발생을 감소시킴으로써, *G*를 매우 향상시킬 수 있다.

만약, MN이 라이프타임이 만료되기 전에 현재의 서브넷에서 다른 곳으로 이동하게 되면, MAP은 BU_0를 보내서 외부 CN과 HA의 캐시에 남아 있는 바인딩 정보를 삭제해야 한다. 이 경우, 추가로 f_{ADD}가 전달되야 한다. MN이 이전 이동 패턴을 따르지 않을 경우, 특히, 적응적 라이프타임 보다 더 짧게 머무를 때, 문제가 발생할 수 있다. 하지만, 현재 상주 시간이 최소 몇 분 이상만 되면, 발생하는 총 BU 메시지 수는 상당 부분 감소시킬 수 있다.

전반적으로 *#CN*과 상주 시간 분포는 그림 8(a), (b)에

서 보는 바와 같이 전체적인 성능에 거의 영향을 주지 못한다. G는 평균 상주 시간이 0과 10분 사이에서 매우 빠르게 증가하고, 13분에서 0.8이 된다. 즉, MN이 13분 이상 서브넷에서 규칙적으로 상주하면, 프로파일의 정보가 매우 효과적이라는 것을 의미한다. 물론, 13분이라는 값은 다양한 실험 설정 값에 따라 달라질 수 있다. 하지만, 사용자가 어느 정도의 규칙성만 가진다면, MN이 얻을 수 있는 장점도 매우 크다고 볼 수 있다.

실험 결과로부터 한 가지 주목할 만한 것은 비록 실험에서 평균 상주 시간을 0.33에서 60분까지 변화를 시켰지만, 실제로는 평균 상주 시간이 60분보다 훨씬 클 수 있다. 본 논문에서는 사용자의 이동성 유형을 단지 세 가지로 나누었지만, 실제 사용자의 이동 및 상주 시간은 훨씬 다양할 수 있음을 의미한다. 따라서, 이동 로그로부터 프로파일을 생성하는 알고리즘을 향상시킨다면, 보다 더 신뢰성 있는 정보를 얻을 수 있을 것이다. 하지만, 이 경우 본 연구의 범위를 벗어나므로 더 이상 언급하지 않겠다.

두 번째 실험은 도착 시간대의 효과를 살펴보기 위한 것이다. RT와 TR은 첫 번째 실험에서와 동일한 방법으로 비교되었다. 두 기법의 차이는 프로파일의 정보가 더 세부적으로 분류되었는가 하는 것이다. 이 실험에서 프로파일에서 이동성 유형 A, B로 기록된 전체 서브넷의 40%은 하나의 시간대를 가지고, 또 다른 서브넷의 40%는 두 가지 시간대를 가지며, 나머지 20%는 세 가지 시간대를 가진다. 즉, 둘 이상의 시간대를 가진다는 것도 그 서브넷에 도착시간대별로 다른 라이프타임 값이 부여됨을 의미한다.

그리고, MN의 이동성이 규칙성을 보이는 서브넷만을 고려했으며, LT_a는 5초로 설정하고, 평균 상주 시간은 10분에서 60분까지 변화하는 경우만 가정하였다. 그림 9에서 평균 상주 시간이 30분 이상이 되면, 두 기법에서 G가 모두 1.0에 접근하고 있으며, 그 차이 또한 크지 않다. 그러나, 대부분 평균 상주 시간에서 RT보다 TR이 적은 대역폭을 사용한다. 그 이유는 서브넷 m에서 둘 또는 그 이상의 시간대가 존재하며 각 평균 상주 시간 사이에 차이가 매우 큰 경우, RT에서 Var_m은 매우 크지만 시간대별로 로그를 분류하여 평균 상주 시간을 적용하는 TR에서 Var_m을 감소시키기 때문이다. 다시 말해서, 다양한 방법에 의해 이동 로그를 분석한다면, 사용자의 이동성과 관계된 보다 의미 있는 정보를 얻을 수 있음을 알 수 있다. 이는 MN의 에너지의 효율적 사용이란 측면에서 볼 때 중요한 시금석이 될 수 있다.

5. 결 론

본 논문에서 우리는 HMIPv6에서 BU를 위한 적응적 라이프타임 결정하기 위한 상주 시간 기반 기법(RT)을 제안했다. 또한, 서브넷에 방문한 상주 시간 뿐만 아니라 MN의 도착시간까지 고려하여 시간대 기반 기법(TR)을 제안하였다. 제안 기법들은 각 MN의 이동성 패턴의 규칙성을 이용하여 동적으로 BU 메시지의 라이프타임 값을 설정하는 알고리즘과 프로파일 정보의 신뢰성을 평가하기 위한 방법을 제시하였다. 실험 결과 빈번한 BU 발생으로 인한 오버헤드를 감소시킬 수 있으며, 특히 MN에서 전송되는 메시지의 수를 줄임으로써 에너지 효율성을 높일 수 있었다.

논문에서 여러 차례 언급하였듯이 제안하는 기법에서의 성능향상은 사용자의 과거 이동성 기록으로부터 얻어진 정보에 의해 가능하였다. 하지만, 실제 이동성은 보다 복잡하며, 그 규칙성을 찾는 것은 그렇게 간단한 작업이 아니다. 이를 위해 과거 이동 로그를 분석하기 위한 다양한 접근 방법이 요구되며, 데이타 마이닝 기법도 그 한 가지 대안이 될 수 있다. 또한, 보다 심도 있는 실험을 통해 다양한 실험 인자 값을 결정하는 것도 성능에 큰 영향을 가져올 것이다.

참 고 문 헌

[1] D. B. Johnson and C. Perkins, "Mobility Support

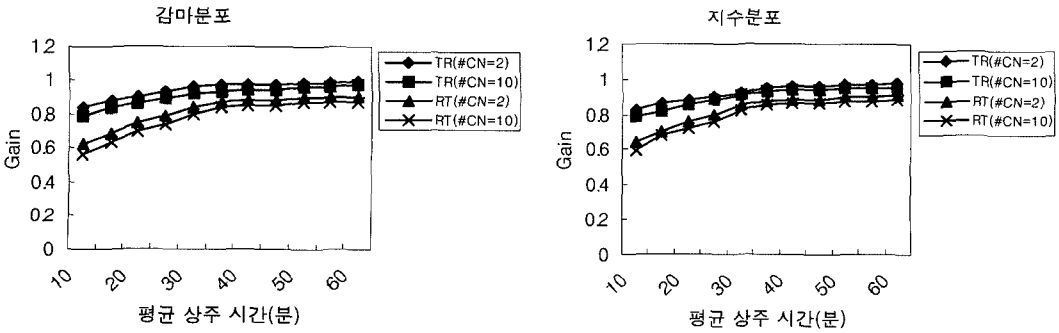
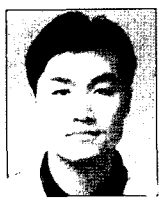


그림 9 RT와 TR의 비교

in IPv6," IETF Internet Draft, draft-ietf-mobileip-ipv6-16.txt, Mar. 2002.
[2] S. Yang, U. Song, J. Gil and C. Hwang, "A Profile-Based Dynamic Binding Update Strategies in Mobile IPv6," *in Proc. Of International Conference on Wireless Networks*, pp. 238-244, Jun. 2002.
[3] H. Soliman, C. Castellucia, K. E. Malki and L. Bellier, Hierarchical MIPv6 mobility management (HMIPv6), IETF Internet *Draft*, draft-ietf-mobileip-hmipv6-08.txt, Jun. 2003.
[4] G. P. Pollini and Chil-Lin I, "A Profile-Based Location Strategy and Its Performance," IEEE Selected Areas in Communications, Vol. 15, NO. 8, pp. 1415-1424, 1997.
[5] S. Tabbance, "Modelling the MSC/VLR Processing Load due to Mobility Management," in *Proc.* of IEEE International Conference on Uiversal Personal Communications., Vol. 1, pp. 741-744, 1998.
[6] Youn-Hee Han, J. M. Gil, and Hwang, "Route Optimization by the Use of Two Care-of Addresses in Hierarchical Mobile IPv6," IEICE Transaction on Communication, Vol. E84-B, No. 4, pp. 892-902, 2001.
[7] J. Arkko and C. Vogt, "Credit-Based Authorization for Binding Lifetime Extension," draft-arkko-mipv6-binding-lifetime-extension-00, May, 2004.
[8] Y. B. Lin, W. R. Lai and R. J. Chen, "Performance Analysis for Dual Band PCS Networks," *IEEE* Journal on Transactions on Computers, Vol. 49, No.2, pp 148-159, Feb. 2000.
[9] C. Castelluccia, "HMIPv6: A Hierarchical Mobile IPv6 Proposal," ACM SIGMOBILE Mobile *Computing and Communication Review*, Vol. 4, No. 2, Apr. 2000.

양 순 옥
1995년 고려대학교 자연자원대학 원예과학과 졸업. 2002년 고려대학교 대학원 컴퓨터학과(이학석사). 2002년부터 고려대학교 대학원 컴퓨터학과 박사과정 재학 관심분야는 이동 컴퓨팅 시스템에서의 사용자의 이동성 관리 및 고성능 컴퓨팅 등



김 성 석
1997년 고려대학교 전산과학과 졸업 1999년 및 2003년 고려대학교 컴퓨터학과 석사 및 박사 졸업. 2003년부터 서경대학교 전자상거래학과 교수로 재직. 관심분야는 분산시스템에서 캐싱 및 동시성 제어 기법, 이동 컴퓨팅 시스템에서 정보 서비스 등

황 종 선
정보과학회논문지 : 정보통신
제 32 권 제 2 호 참조