

이동성이 잦은 Mobile IP를 위한 효율적인 연결 설정 기법

노경택*, 김혜영**

A Method of efficient connection setting for Mobile IP with high mobility

Kyung-Taeg Rho*, Hye-Young Kim**

요 약

IETF에서 제안한 Mobile IP는 효율적이거나, 이동 호스트들이 어느 특정한 지역 내에서 많은 움직임을 가지면서 통신할 경우 비효율적인 면이 있다. 이는 이동 호스트를 등록하고 인터넷 환경 하에서 보안된 경로를 설정하며 이 경로에 데이터를 전송하는 데 많은 지연을 유발하기 때문이며, 이동 호스트들이 어느 특정한 지역 내에서 잦은 이동을 갖는 경우, 이러한 비효율은 더 심각해진다고 할 수 있다. 따라서 본 논문에서는 특정지역 내의 이동성 관리에 있어서 위와 같은 문제점을 보완하기 위한 방법의 하나로 경계 라우터와 포인터 포워딩 및 캐시 방법을 결합한 Anchor 체인 방법에 의한 Anchor 외부 에이전트를 사용한 방법을 제안한다.

Abstract

Although Mobile IP proposed in IETF is effective, it has inefficiency in case mobile hosts communicate with each others while they are roaming frequently in a specific area. It occurs lots of latency because mobile hosts must be registered and establish an secure path under the internet environments and transmitting data on the path. Additionally this inefficiency is more aggravated in case mobile hosts has high mobility. Thus this paper propose a method using Anchor foreign agent by Anchor chain method which combine an pointer forwarding and a cache method plus a border router as a way to complement the above problem which exists in an mobility management in a specific area.

▶ Keyword : Mobile IP, Care-of Address, tunneling

• 제1저자 : 노경택

• 접수일 : 2004.09.29, 심사완료일 : 2004.11.16

* 서울보건대학 인터넷정보과, ** 고려대학교 컴퓨터과학과 박사

I. 서론

다양하고 빠른 변화를 하는 사회에서 컴퓨터 사용자들은 많은 이동성을 가지게 되며, 최근 웹 기반 응용 프로그램들의 빠른 발전으로 인터넷상에서의 컴퓨터들간의 통신이 증가하고 있다. 그러나 인터넷의 통신프로토콜인 TCP/IP는 기본적으로는 이동호스트를 지원하도록 설계되지는 않았다. 그러므로 Mobile IP 프로토콜은 독립적으로 변화하는 이동호스트들의 위치를 알고 데이터를 전송할 수 있는 이동성 관리 기능을 가져야 한다. 즉 Mobile IP 프로토콜은 이동호스트에게 끊임 없는 인터넷의 연결성을 제공해야 한다. 이를 위한 기능은 크게 세 가지로 나누어 볼 수 있다. 그 첫째는 이동호스트들의 위치를 등록하여 그 위치를 알기 위한 등록과 둘째로는 이동호스트의 변하는 위치를 관리하는 것으로 바인딩 업데이트, 그리고 셋째로는 이동호스트의 목적지에 실제 데이터를 전송하는 것으로 볼 수 있다.

IETF Mobile IP에서는(1)(7)(8) 이를 위하여 이동호스트 외에 홈 에이전트와 외부 에이전트를 두고 있으며 이 둘간의 터널링을 통하여 실제 데이터를 전송하고 있다. 즉, 통신하고자 하는 호스트나 이동호스트간의 통신은 세 개의 네트워크를 통해 이루어진다. 즉, 통신하고자 하는 호스트의 서브네트워크, 홈 에이전트의 서브네트워크, 그리고 위치하고 있는 이동호스트들의 현재 서브네트워크로 이루어진다. 이를 피하기 위해서 라우팅 최적 기법(2) 제안되었다. 하지만 이 기법도 여전히 통신하고자 하는 호스트가 이동호스트의 COA를(Care-Of-Address) 홈 에이전트로부터 받아야 한다. 또한 터널링에 따른 통신 대역폭과 프로토콜 오버헤드는 여전히 존재하게 된다. 특히 이동호스트들이 어느 특정 지역 내에서 잦은 움직임이 있는 경우에는 위의 오버헤드는 더욱 심각하게 된다.

본 논문에서는 움직임이 많은 특정 지역내의 이동호스트들에 대해 홈 에이전트와 외부 에이전트사이의 메시지 교환을 줄이고 외부 에이전트와 홈 에이전트 사이의 안전한 터널, 안전한 키를 재설정하는 낭비를 줄여 효율적인 통신과 인터넷을 통한 전송 오버헤드를 줄이는 방법으로 경계 라우터와 Anchor 외부 에이전트를 사용한 방법을 제안한다.

II. 관련연구 및 연구동기

2.1 이동호스트 탐색

홈 에이전트와 외부 에이전트는 그들이 서비스하고 있는

각 링크에서의 가용성을 알아보게 된다. 이동호스트 탐색은 이동호스트가 홈 네트워크나 외부 네트워크에 연결되어 있는지를 찾아보는 방법으로 이를 통해서 이동호스트가 하나의 네트워크에서 다른 네트워크로 이동했음을 감지할 수 있게 된다. 새로 도착한 이동호스트는 예상되는 에이전트가 있는지 알기 위해 링크에서 solicitation을 보낼 수 있다. 모든 이동호스트는 에이전트 solicitation을 구형해야 한다. 이 solicitation은 에이전트 부재시와 COA가 링크 계층 프로토콜이나 다른 방법에 의해 결정되지 못했을 때만 보내져야 한다.

2.2 위치등록과 라우팅

이동호스트가 홈 네트워크의 하위 네트워크를 떠나 다른 곳으로 이동했을 때 자신의 COA를 홈 에이전트에 등록한다. 이동호스트는 두 가지 경우로 등록을 하게 되는데 하나는 이동호스트의 홈 에이전트에의 등록을 전달하는 외부 에이전트를 경유하는 것이고 다른 하나는 이동호스트가 외부 에이전트가 재부팅된 것을 감지했을 때 홈 에이전트로 직접 등록하는 것이다. 현재 등록 지속시간이 거의 만기되었을 때 이동호스트는 자신의 외부 에이전트에 (재)등록해야 한다. 이동호스트에 의해 전송된 멀티캐스트와 브로드캐스트 패킷을 제외한 패킷들은 터널링 없이 목적지로 바로 경로지정이 된다. 이동호스트로 보내진 패킷들은 홈 에이전트로 라우팅이 지정되고 여기서 복사본이 COA로 터널링된다. 원래의 패킷은 이 터널링에서 추출되며 이동호스트로 전달된다.

2.3 경계 라우터(Border Router)

Mobile IP 프로토콜에서는 등록과 패킷전송의 오버헤드 때문에 지연이 일어날 수 있다. Chu-Sing, Kun-da, Chun(6)은 이동호스트들의 움직임은 지역적 성질을 가진다는 관찰 결과 하에 지역적 통신의 오버헤드를 줄이고 인터넷상에서의 지연을 제거하는 기법을 제안했다. 이동호스트가 어떤 특정 네트워크사이에서 이동하는 경우에 어떤 범위 내에서 외부 에이전트의 가까이에 가상의 홈 에이전트를 둬으로써 등록 지연을 줄일 수 있다. (6)에서 그들은 여러 개의 네트워크를 하나의 관리자 도메인으로 묶는 시도를 했고 경계 라우터에 가상의 홈 에이전트를 위치시켰다. 이는 등록 지연을 줄이는 것 외에 통신하고자 하는 호스트에서 캐시를 두지 않고도 데이터 지연을 제거하는 부가적인 이익을 가질 수 있다. 즉, 통신하고자 하는 호스트가 도메인을 벗어날 때 경계 라우터와의 연결은 재사용될 수 있다. 그러므로 라우터

로의 데이터 경로는 통신하고자 하는 호스트가 이동 호스트와 같은 도메인에 위치할 때 짧아질 수 있다

2.4 로컬 anchoring 방법

이동 호스트와 홈 에이전트 사이에 중재하는 것을 추가함으로써 바인딩 오버헤드를 제거하는 많은 방법들이 있다.

Bejerano 와 cidon 은 바인딩 오버헤드와 전달의 비용을 줄이기 위한 목적으로 포인터 포워딩과 캐시방식을 결합한 이동성 기법을 제안했다. 이동 호스트는 물리적인 장치이므로 보통 지역성을 가진다. 지역적인 이동에서 이 기법은 이동 호스트를 홈 에이전트에 연결한 Anchor 체인을 사용하여 결합 비용을 줄였다. 각 Anchor는 어느 정도의 정확성에서 이동 호스트의 위치를 정의한다. Anchor는 이동 호스트가 방문한 외부 에이전트고 첫 번째 앵커는 이동 호스트의 홈 에이전트가 된다. 모든 앵커는 어느 정도의 정확성으로 이동 호스트의 위치를 기록하고 체인에서 연속적으로 외부 에이전트를 가리킨다. 이동 호스트가 붙어있는 외부 에이전트에 도착할 때까지 이 체인을 추적하면 정확성은 높아진다. 이동 호스트로 목적지가 정해진 패킷들은 접근 지점이라 불리는 체인에서의 Anchor중 하나로 보내진다. 위의 패킷들은 이동 호스트로 전달하는 외부 에이전트에 닿을 때까지 한 Anchor에서 다른 Anchor로 전달된다. 이들 Anchor들은 또한 이동 호스트로의 데이터그램의 효율적인 전달에도 쓰인다. 통신하고자 하는 에이전트와의 통신을 초기화시키자마자 이동 호스트는 Anchor체인의 기록을 이용하여 통신하고자 하는 에이전트를 지원한다. 통신하고자 하는 에이전트가 이동 호스트에게 메시지를 보내려고 시도할 때 접근 지점이라 불리는 외부 에이전트를 선택하고 메시지를 그 곳으로 보낸다. 메시지는 체인을 따라 외부 에이전트에서 이동 호스트에 닿을 때까지 전달된다.

이 기법에서는 선택된 접근 지점이 이동 호스트 위치를 추적하는 포인터 체인을 포함한다. 그러므로 이 접근 지점은 처음에 이동 호스트가 홈 에이전트에게 메시지를 보내지 않고 현재 이동 호스트의 근처에 있는 것이 선택될 수 있다. 이 기법은 바인딩 오버헤드와 전달에 대한 오버헤드를 줄일 수 있다.

이 논문에서 우리는 특정 지역에서 이동성을 관리할 목적으로 지역 Anchoring과 경계 라우터를 결합한 기법을 제안한다. 논문의 구성은 다음과 같다. 3절 본문에서 이동 호스트의 등록에 대해 자세히 다루고, 4절 결론에서는 제안한 방법의 시뮬레이션 결과를 보이며 결론을 내리면서 마지막으로 향후 연구에 대해 생각해 볼 것이다.

III. 본 론

IETF의 Mobile IP는 아래의 그림 1과 같이 나타낼 수 있다. 이동 호스트가 이동한 네트워크의 외부 에이전트에서 홈 에이전트에 자신의 위치를 등록하기 위해서 외부 에이전트와 홈 에이전트 사이에 메시지를 주고받게 됨으로써 등록 지연이 발생하게 된다. 만일 홈 에이전트와 외부 에이전트가 같이 위치하고 있다면 이런 지연은 감소하게 될 것이다.

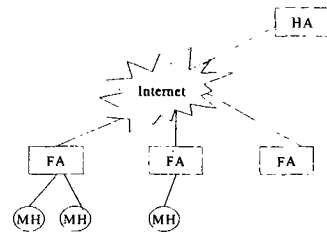


그림 1 IETF Mobile IP
Figure 1. IETF Mobile IP

즉, 위에서 설명한 등록 지연은 홈 에이전트와 외부 에이전트 사이의 거리에 비례한다[3][4]. 그러므로 네트워크를 여러 개의 군집된 서브 네트워크로 나누어서, 나누어진 네트워크간의 경계에 경계 라우터를 두어 가상 홈 에이전트를 둬으로써 등록 지연을 감소시킬 뿐 아니라, 핸드오프 동안에 버퍼링의 기능을 가짐으로써 통신하고자 하는 호스트에 캐쉬 없이도 데이터 지연을 없애는 장점을 갖는다. 또한 Anchor 외부 에이전트를 둬으로써 등록 후에 발생하는 홈 에이전트와 외부 에이전트간의 터널링에서 오는 지연을 피할 수 있는 장점을 갖는다. 아래의 (그림 2)는 위의 내용을 간략하게 보여 주고 있다.

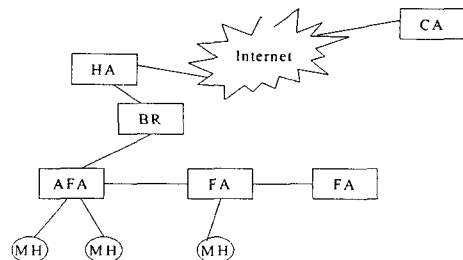


그림 2 제안된 Mobile IP
Figure 2. Proposed Mobile IP

다음은 어떻게 이동 호스트를 등록하고 또한 Anchor 외부 에이전트를 어떻게 관리하는지 즉, 언제 Anchor 외부 에이전트가 새로운 Anchor 외부 에이전트가 되며, 이동호스트의 위치를 어떻게 탐색하는지를 설명하고 핸드오프시에 어떻게 버퍼링을 하는지를 보이며 마지막으로 IETF Mobile IP와의 상호운영 방안을 설명한다.

3.1 등록(registration)

이동호스트가 이동하거나 혹은 이동호스트가 갖는 정보 중의 유효기간이 만료되기 바로 직전이나 혹은 주어진 주기에 의해서 등록을 하게 되는데 이동 호스트가 홈 에이전트에 있을 때 m_1 이고 이동 호스트의 현재 위치가 m_ℓ 이라 하면 $dk = \text{dist}(mk, mk+1)$ 로 mk 와 $mk+1$ 사이의 k 번째 포인트의 크기를 갖는 지리적 거리를 의미한다. $k \geq 1$ 이며 $dk \leq dk-1 / \beta$ 일 때 그 Anchor의 범위내에 있다고 한다[5]. 따라서 Anchor 외부 에이전트에서만 등록이 일어나고, 그렇지 않으면 새로운 Anchor 외부 에이전트를 계산하여 등록을 하게 된다. 이 때 이동호스트의 현재 zone과 이전의 zone을 비교하여 같지 않은 경우는 새로운 경계 라우터에 등록 해 주어야 하며 또한 홈 에이전트에게도 등록을 해 주어야 한다. 다음은 등록을 위한 알고리즘을 나타낸다.

```

registration(h, mh) {
    ha_list: /* HLR에 등록된 내용 */
    mh_hip: /* 이동호스트의 home agent */
    mh_coa: /* 이동호스트의 care-of-address */
    afa, bn, hn: /* 등록할 수 있는 최대수 */
    br_list: /* boarder router에 등록된 내용 */
    alfa_list: /* afa에 등록된 내용 */
    m_hip: /* 이동호스트의 홈 주소 */
    m_ccoa: /* 이동호스트의 현재 coa */
    m_zone_id: /* zone의 식별자 */
    m_br: /* boarder router사용여부 */
    m_br_id: /* boarder router 식별자 */
    m_anchor: /* anchor 사용여부 */
    m_anchor_id: /* anchor 식별자 */
    m_pcoa: /* 이전의 coa */

    if (length(ha_list) < h) ha_list = search_br(mh_mip,
    ha_list, h);
    pending_ha = ha_list;
    while (registration_state != SUCCESS)
    { retries = 0;
      while (retries ++ < MAX_RETRY)
      {
        for (all ha in pending_ha)
          lifetime를 검사시킨다.
      }
    }
}
    
```

```

if ( 정해진 lifetime내에 모든 pending_ha로부터 acknowledgment를 받음)
    { registration_state = SUCCESS;
      return;
    }
else if ( 정해진 lifetime내에 모든 ha로부터 acknowledgment를 받음)
    { registration_state = PARTIAL_SUCCESS;
      pending_ha의 list를 update한다.;
    }
}
) /* end of while */
pending_ha = search_br(mh_hp, ha_list, pending_ha, h);
ha_list를 update한다;
) /* end of registration */
    
```

3.2 새로운 Anchor 외부 에이전트

다음은 새로운 Anchor 외부 에이전트에서 처리하는 과정을 알고리즘으로 나타낸다.

```

new_AnchorFA (u, l, {m1...mℓ})
{
    u_flag = FALSE;
    while ( k > 1) and ( dist(u, mk) > dist((mk, mk-1) / β) ) do
    { Anchor FA로부터 mk를 읽음;
      u_flag = TRUE;
      k = k - 1;
    }
    return(m_flag, u, {m1...mℓ})
}
    
```

3.3 이동호스트의 위치 탐색

통신하고자 하는 호스트는 이동 호스트의 위치 탐색하는 알고리즘을 통해서 통신하게 될 것이다. 결국 이 과정을 통해서 통신하고자 하는 호스트는 Anchor 외부 에이전트로부터 이동 호스트의 정보를 얻게 될 것이다. 이 정보를 ha_list로 관리하고 있다. 그 후에 통신하고자 하는 호스트는 이동 호스트의 정보를 얻는 데, 터널링 없이 곧바로 이동 호스트와의 통신이 가능하게 된다. 다음은 위의 이동 호스트의 위치 탐색을 위한 알고리즘을 나타낸 것이다.

```

find(mh_hip, n)
{ pending_ha = NULL;
  if (ha_list내에 이동호스트의 정보가 없음)
    ha_list = query_br(mh_mip, pending_ha, n);
  if (ha_list내에 이동호스트의 정보가 없음)
    ha_list = query_afa(mh_mip, pending_ha, n);
  pending_ha = select(ha_list, mh_mip);
  retries = 0;
  while (retries ++ < MAX_RETRIES)
  { if (정해진 lifetime이 경과하기 전에 어느 pending_ha에 request reply에 대한 receive가 있음)
    }
}
    
```

```

{ coa를 갱신함:
  lifetime을 갱신함:
  break:
}
서로 통신을 한다:
} /* end of while */
} /* end of find */
    
```

3.4 핸드오프(handoff)시의 버퍼링(buffering)

이동 호스트가 주기적인 advertisement 메시지를 받음으로써 자신의 zone의 변화를 알게 되면, binding update extension으로 등록 요청을 하게 된다. 이 메시지는 Anchor 외부 에이전트나 경계 라우터, 혹은 홈 에이전트에 직접적으로 보내진다. 이때의 binding update extension 메시지는 경계 라우터 정보를 갖고 있게 되며, 경계 라우터가 binding update extension으로 등록 요청을 받으면, 이전의 zone에서의 경계 라우터와 홈 에이전트, 그리고 이동 호스트의 이동성을 알기 위한 정보를 추출해 낸다. 새로운 메시지가 새로운 경로를 통해 재경로 설정을 할 수 있도록 메시지를 이전 zone의 경계 라우터에게 보낸다. 이런 과정을 통해 잃어버릴 수 있는 메시지를 줄일 수 있게 된다.

3.5 상호 운영성

IETF의 Mobile IP와 본 논문에서 제안한 Mobile IP와는 상당히 다른 구조적 차이를 보이고 있다. 따라서 본 논문에서 제안한 구조를 갖는 이동 호스트가 IETF의 Mobile IP의 구역으로 이동하거나, 혹은 그 반대 경우를 위한 상호 운영성을 고려해야 한다.

이를 위해서 등록시에 경계 라우터와 Anchor 외부 에이전트의 사용여부를 표시하는 부분을 등록추가에 두고 있다. 따라서 이 부분에 대한 검사를 통해 상호 운영성을 갖도록 하였다.

IV. 결론 및 향후 연구과제

우리는 binding operation의 전체 통신 비용과 접근 비용을 시뮬레이션하였다. 그림 3과 그림 4는 binding operation의 시뮬레이션 결과를 나타낸 것으로 제안된 방법이 낮은 통신 오버헤드와 이동 호스트의 접근 비용을 가짐을 보여 주고 있다.

또한 다양한 DMR(Delivery Movement Ratio=delivery/movement ratio)에 대한 평균 전송비용을 시뮬레이션 하였다. 그림 5와 6은 제안한 방법으로 다양한

DMR값에 대한 delivery operation의 평균 통신비용과 평균 접근 비용을 나타내고 있다. 이동 호스트는 속도 S를 가고, 임의의 경로를 따라 움직이고 통신하고자 하는 호스트는 그래프들을 따라 규칙적으로 분산되어 있다. 그림 4와 (그림 5)에서는 제안된 방법이 낮은 통신 오버헤드와 낮은 이동 호스트들의 낮은 접근 비용을 갖는다는 것을 보여 주고 있다.

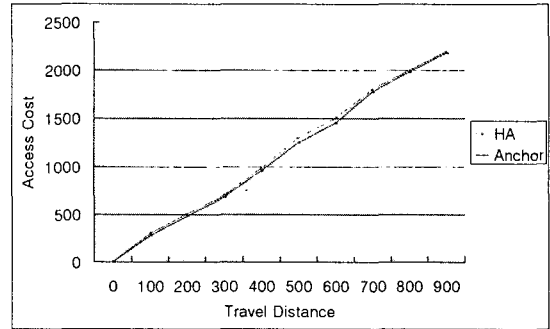


그림 3 binding operation의 총 접근 비용
Figure 4. Total access costs of binding operation

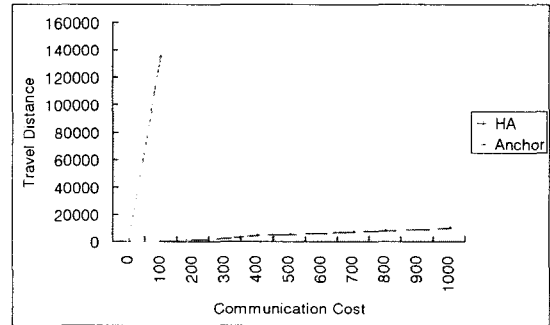


그림 4 binding operation의 총 통신비용
Figure 3. Total communication costs of binding operation

본 논문에서는 특정 지역안에서 잦은 이동성을 가지면서 통신할 경우 등록에 따른 지연과 새로운 등록시에 발생하게 되는 안전한 경로의 생성에 따른 오버헤드등 비 효율성을 감소시키려는 목적으로 경계 라우터와 Anchor 외부 에이전트를 사용한 방법을 제안하였다. 따라서 특정지역 내에서의 움직임에 대해서는 경계 라우터와 Anchor 외부 에이전트의 사용으로 재경로설정외 오버헤드를 줄여 줄 뿐만아니라 효율적인 연결설정을 보여 주고 있다. 또한 홈 에이전트의 예기치 못한 오류시에 경계 라우터에 의한 부분적인 서비스 제공이 가능한 효과를 얻을 수도 있다.

그러나 본 논문에서 제안한 방법이 위의 실험 결과에서 보여 준 바와 같이 특정지역내의 잦은 이동성이 아닌 경우에는 커다란 효율성을 갖지 못함을 알 수 있다. 따라서 향후의 연구과제로 이동 호스트의 이동성에 대한 분석을 통해 경계 라우터와 Anchor 외부 에이전트를 설정하는 방법을 들 수 있을 것이다.

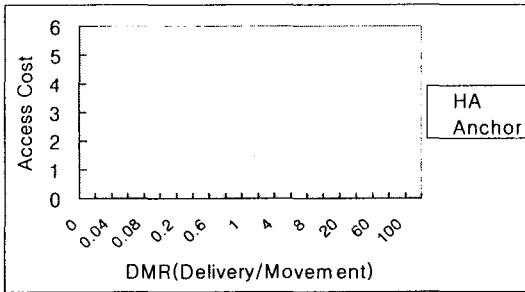


그림 5. delivery operation의 평균 통신 비용
Figure 5. Average communication costs of delivery operation

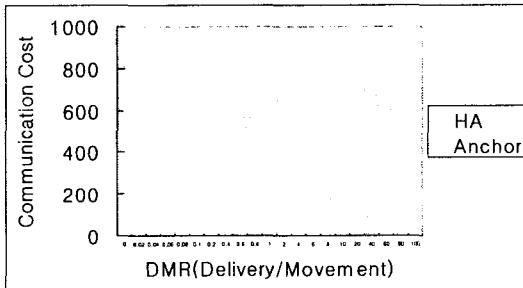


그림 6. delivery operation의 평균 접근비용
Figure 6. Average access costs of delivery operation

참고문헌

- [1] C. Perkins, editor IP Mobility Support RFC 2002, Feb. 2000
- [2] Johnson, D.B. C. Perkins(eds.), Route Optimization in Mobile IP, Internet Draft, draft-route optim-09.txt, 15, Feb. 2000
- [3] A. Bar-Noy and I. Kessler, Tracking Mobile users in Wireless Communication Network, IEEE Information '93, Vol. 3, pp. 1232-1239
- [4] T. Imielinski and B. R. Badrinath, Querying in Highly Mobile Distributed Environments, Proceeding of the 18th VLDB, August 1992, pp. 41-52
- [5] Yigal Bejerano and Israel Cidon, An Anchor Chain Scheme for IP Mobility Management
- [6] Chu-Sing Yang, Kun-Bunda Wu, and Chun-Wei Tseng, Database and Expert Systems Applications, 1998. Proceedings. Ninth International Workshop on , 1998
- [7] 노경택, "Mobile-IP에서의 이동성 관리와 Smooth Handoffs", 한국OA학회논문지 VOL. 4. NO. 4. Dec. 1999
- [8] 노경택 "NEMO와 결합된 Hierarchical Mobile IPv6에서 멀티캐스팅을 이용한 핸드오프 지원기법" 한국컴퓨터정보학회논문지 VOL. 9. NO. 3. Sep. 2004.

저 자 소 개



노 경 택
고려대학교 컴퓨터학과 학사수료
서울보건대학 인터넷정보과 교수
〈관심분야〉 wireless sensor networks,
mobile communications,



김 혜 영
1989년 9월~1998년 2월
현대전자소프트웨어
2001년 3월~2004년 2월
청강문화산업대학 교수
고려대학교 컴퓨터학과 학사
〈관심분야〉 mobile communications,
mobile game programming