

무선 이동망을 위한 적응적 분산 동적 위치 관리 알고리즘의 설계 및 평가

천성광[†]·배인한^{††}

요약

미래의 PCS(Personal Communication Service) 네트워크 설계에서 하나의 중요한 문제는 위치 정보의 효율적인 관리이다. 이 논문에서, 우리는 n 위치 정보 데이터베이스들 중에 k 개에 이동 단말의 위치를 저장하는 적응적 분산 동적 위치 관리 알고리즘을 제안한다. 제안하는 알고리즘은 무선 이동망 내의 위치 정보 데이터베이스들의 공간 지역성과 원격 이동 단말들로부터 로컬 이동 단말의 위치 질의 선호도에 따라 사본 인자, k 를 적응적으로 선택한다. 제안하는 알고리즘의 성능을 분석적 모델과 시뮬레이션을 통하여 평가한다. 성능 평가 결과, 호 이동율에 관계없이 제안한 알고리즘의 성능이 Krishnamurthi의 알고리즘에 비해 성능이 우수함을 알 수 있었다.

Design and Evaluation of an Adaptive Distributed Dynamic Location Management Algorithm for Wireless Mobile Networks

Sung-Kwang Chun[†] · Ihn-Han Bae^{††}

ABSTRACT

An important issue in the design of future Personal Communication Service (PCS) networks is the efficient management of location information. In this paper, we propose an adaptive distributed dynamic location management algorithm that stores the position of the mobile terminal in k of the n location information databases (LIDs). The proposed algorithm chooses adaptively k , replication factor according to both the space locality of LIDs in wireless mobile networks and the location query popularity to local mobile terminal from remote mobile terminals. The performance of proposed algorithm is evaluated by both an analytical model and a simulation. Based on the results of performance evaluation, we know that the performance of the proposed algorithm is better than that of Krishnamurthi's algorithm regardless of call-mobility ratio.

키워드 : 무선 이동망(Wireless Mobile Networks), 위치 관리(Location Management), 분산 위치 관리 알고리즘(Distributed Location Management Algorithm)

1. 서론

셀룰러 통신 기술은 무선 이동망 기능의 주목할만한 발전으로 지난 십 년 동안에 급속하게 발전되었다. PCS 네트워크라 부르는 3세대 네트워크는 많은 사람들에게 다양한 서비스들을 제공할 것이다. PCS 네트워크에서 채널 할당, 위치 관리, 이동 호스트 전력의 최적 사용과 같은 다양한 설계 문제들에 대한 많은 연구가 진행되고 있다.

PCS 네트워크에서, 연결을 설정하기 위하여 이동 단말(MT, mobile terminal)의 위치를 찾는 위치 추적 메커니즘이 필요하다. 현재 방법들은 필요할 때 이동 단말이 네트워크에 그것의 위치를 보고할 것을 요구한다. 다소의 위치 관

리 방법들은 시간-기반 갱신 정책을 사용한다. 다른 방법들은 이동 단말의 현재 위치가 이전 갱신의 위치로부터 최소한 미리 정의된 거리만큼 떨어졌을 때만 이동 단말의 위치를 갱신할 것을 요구한다. 네트워크는 이동 단말의 위치를 위치 정보 데이터베이스(LID, location information database)들에 저장되고, 이 정보는 호 전달 동안에 검색된다. 다수의 LID들을 갖는 분산 데이터베이스 구조에서는 불필요한 성능 병목현상을 피하기 위하여 각 데이터베이스에 대한 위치 질의 부하를 균등화하는 이점이 있다. 이 논문에서는 LID들의 공간 지역성과 원격 이동 호스트들로부터의 로컬 이동 호스트의 위치 질의 선호도를 고려한 적응적 분산 동적 위치 관리 알고리즘을 제안하고, 그것의 성능을 분석적 모델과 시뮬레이션을 통하여 평가한다.

이 논문의 구성은 다음과 같다. 제2장에서는 분산 위치 관리에 대한 몇몇의 이전 연구들을 살펴보고, 제3장에서는

[†] 정 회 원 : 한국전자통신연구원 우정기술연구센터 연구원
^{††} 정 회 원 : 대구가톨릭대학교 컴퓨터정보통신공학부 교수
논문접수 : 2002년 1월 29일, 심사완료 : 2002년 8월 2일

본 연구에서 사용하는 PCS 모델을 설명하고, 제 4장에서는 본 논문에서 제안하는 적응적 분산 동적 위치 관리 알고리즘을 제안하고, 제 5장에서 제안하는 알고리즘의 성능을 분석적 모델과 시뮬레이션을 통하여 평가한다. 그리고 마지막으로 제 6장에서 결론을 맺는다.

2. 관련 연구

PCS 네트워크에서 위치 관리를 위한 다수의 방법들이 개발되었다. 그러한 방법들은 집중화된 데이터베이스 또는 분산 데이터베이스를 기초로 한다. 대표적인 위치 관리 방식으로는 갱신 회수와 검색 시간을 줄이기 위한 캐싱 방식[1], 포워딩 포인터 방식[2], 앵커 방식[3], 홈 이동 단말에 대해 VLR(visitor location register)을 사용하지 않는 단층 방식[4], 위치 데이터베이스를 계층적으로 배열한 계층 방식[5], 그리고 이동 단말의 위치 정보의 사본을 다수의 데이터베이스에 분산하여 관리하는 분산 방식[6-8] 등이 있다.

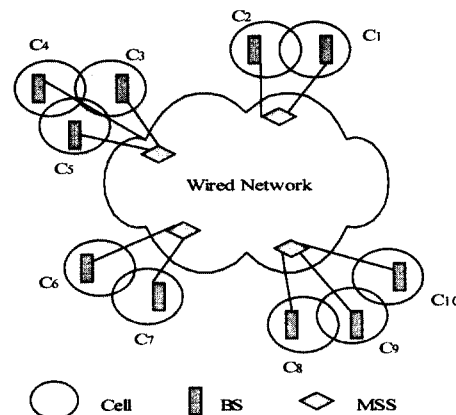
- 캐싱 방식 : 이동 단말에 대한 위치 질의가 STP(signal transfer point)를 통해 HLR(home location register)에 도착하면 STP는 이 이동 단말의 위치 정보를 캐시에 저장한다. 그리고 이후에 이 단말에 대한 위치 정보가 질의되면 HLR을 액세스하지 않고 캐시에 저장된 정보를 이용한다.
- 포워딩 포인터 방식 : 이동 단말이 최초의 위치 등록 시에는 HLR과 VLR에 위치 등록을 하고 등록된 지역을 바꿀때 이전 VLR로 메시지를 전송하여 현재 VLR을 가리키는 포워딩 포인터를 설정한다.
- 앵커 방식 : 이동 단말이 등록지역을 옮겼을때 새로 진입한 지역의 MSC(mobile switching center)는 해당 VLR에 그 이동 단말의 위치를 등록하며, 이 정보는 HLR이 아닌 그 이동 단말 부근에 위치한 LA(local anchor)로 전송되어 위치 정보가 갱신된다.
- 단층 방식 : 단층 방식에서는 HLR을 지역적으로 분산시켜 배치하여 VLR이 없어 신호 절차를 간소화시킬 수 있다. 따라서 위치 등록 메시지와 호 설정 요구 메시지는 HLR로 직접 전송된다.
- 계층 방식 : 위치 데이터베이스를 계층적으로 배열하여 이동 단말의 위치 추적과 호 설정 과정이 지역적으로 이루어진다. 각 데이터베이스는 현재 그 노드 아래에 있는 모든 이동 단말들에 대한 위치 정보를 가진다. 따라서 위치 추적 과정에서는 이전 노드와 새로운 노드의 공통 조상 노드까지의 데이터베이스를 갱신하며 위치 질의 메시지는 송신측과 수신측의 공통 조상까지 전송된다.
- 분산 동적 방식 : 이동 단말에 대한 위치 정보의 사본

이 위치 정보 데이터베이스들의 부분집합에서 관리되고, 이 부분집합은 이동 단말이 이동에 따라 동적으로 변한다.

본 논문에서는 제안하는 적응적 분산 동적 위치 관리 알고리즘은 분산 동적 방식에 속하며, 그 범주에 속하는 기존 연구들을 살펴보면 다음과 같다. 먼저, Parakash는 동적 위치 관리 방법을 제안하였다[6]. 여기서 n 데이터베이스들은 카디날리티(cardinality) $2\sqrt{n}-1$ 의 부분집합들로 분할되고, 부분집합들의 어떤 쌍은 최소한 하나의 공통 데이터베이스를 가진다. 위치 갱신 요청을 받았을때, 갱신을 위한 부분집합 선택에서 이동 단말의 ID와 그것의 현재 위치를 고려한 해시 함수를 사용한다. 호 전달 요청을 받았을때, 이동 지원국은 질의를 위한 부분집합을 선택하기 위하여 자신의 ID와 호출된 이동 단말의 ID를 갖는 해시 함수를 사용한다. 이 알고리즘은 위치 갱신과 질의에서 부하 균등을 이룰 수 있었다. 그리고 Krishnamurthi는 각 데이터베이스들의 갱신과 수정 부하가 균형을 이루는 분산 데이터베이스 구조에 기반 한 위치 관리 알고리즘을 제안하였다[7,8]. 여기서 어떤 이동 단말이 등록 지역을 떠나 다른 등록지역에 진입하면, 그것은 위치 갱신(LU, location update) 메시지를 발생시킨다. 이 메시지는 n LID들 중의 k LID들에서 그 이동 단말의 위치 정보를 갱신시킨다.

3. 시스템 모델

대부분 PCS 네트워크들은 대역폭 효율성을 위해 셀룰러 구조를 사용한다. 이 구조는 (그림 1)과 같이 유선 백본과 무선망으로 구성된다. 지리적 영역들은 셀의 집합으로 분할되고, 각 셀은 유선망과 무선망간의 무선 인터페이스 역할을 하는 기지국(BS, base station)에 의해 서비스된다. 다수의 BS들은 BS들과 백본간의 게이트웨이 역할을 하는 이동 교환국과 연결된다. 그 이동교환국은 PCS망의 각 등록된 이



(그림 1) PCS의 구조

동 단말에 대한 위치 및 서비스 정보를 저장하는데 사용되는 위치 정보 데이터베이스를 액세스한다. 분산 데이터베이스 구조는 증가된 제어 트래픽과 연결 지연 비용을 부담하여 결합 포용력, 확장성, 모듈성을 제공한다.

이 논문에서는 (그림 2)와 같은 일반적인 분산 LID 구조에 대한 논리적 설정을 가정한다. 여기서는 동등한 저장 및 액세스 기능을 갖는 n 데이터베이스들이 있다. 그러한 데이터베이스들은 유선망을 통하여 MSC와 서로 연결되어 있다. 어떤 MSC는 자신의 국부 LID를 알고 있고, LID들은 거리에 기초하여 논리적 링 구조로 배열되어 있다고 가정한다. 그리고 모든 MSC들은 같은 논리적 링 구조를 가지고 있다고 가정한다.

등록지역에 있는 이동 단말로부터 자주 질의되므로 cold 이동 단말보다 더 큰 사본 인수 k 가 선택되어야 한다. 그리고 로컬 LID가 관리하는 이동 단말들은 로컬 LID와 가까운 등록지역에 있는 이동 단말에 의해 자주 질의되는 공간 지역성이 있으므로 NLID는 LDLID 보다 더 큰 사본 인수 k 를 가져야 한다. 여기서 우리는 이동 단말의 선호도와 LID들의 공간 지역성을 고려하여 다음 네 가지 사본 인수를 사용한다.

- 이동단말이 hot인 경우 NLID에서의 사본인수, k_1
- 이동단말이 hot인 경우 DLID에서의 사본인수, k_2
- 이동단말이 cold인 경우 NLID에서의 사본인수, k_3
- 이동단말이 cold인 경우 DLID에서의 사본인수, k_4

어떤 이동 단말에 대한 위치 정보의 사본을 가지고 있는 LID들 간의 거리를 a 라 두면, n 개의 위치 데이터베이스와 사본 인수 k 를 갖는 시스템 (n, k) 에 대해 $n = k\alpha$ 이므로 $\alpha = \lfloor \frac{n}{k} \rfloor$ 이다. 그러므로 위치 갱신 단말이 hot이 $\alpha_1 =$

$$\lfloor \frac{2m+1}{k_1} \rfloor, \alpha_2 = \lfloor \frac{n-(2m+1)}{k_2} \rfloor$$

이고, 위치 갱신 단

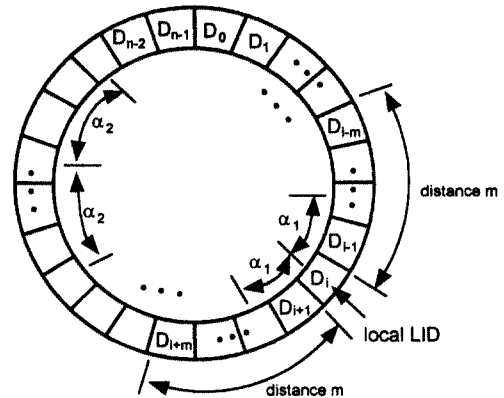
(그림 2) 분산 위치 데이터베이스 구조

4. 적응적 분산 동적 위치 관리 알고리즘

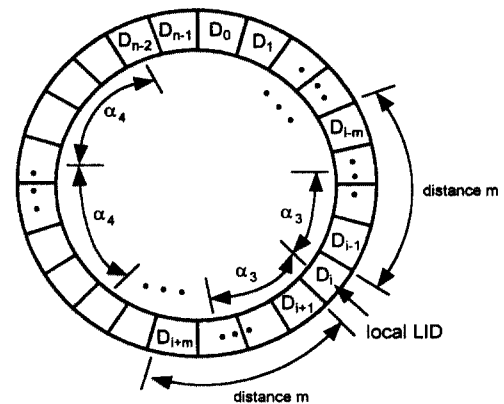
적용적 분산 동적 위치 관리 알고리즘은 크게 두 정책 : 위치 갱신 정책, 위치 질의 정책으로 구성된다. 제안하는 적응적 분산 동적 위치 관리 알고리즘 설계에서 핵심적인 문제는 최적 성능을 이루기 위하여 LID 개수 n 에서 어떤 이동 단말의 위치 정보를 저장하고 있는 LID 개수인 사본 인수 k 를 결정하는 것이다.

4.1 위치 갱신 정책

이동 단말이 등록지역을 떠나거나 다른 등록지역으로 진입할때, 그 이동 단말의 해당 MSC에 의해 위치 갱신 메시지가 발생한다. 그 MSC에 대응하는 LID를 로컬 LID라 한다. 본 논문에서, 우리는 사본 인수 k 의 결정에서 이동 단말의 선호도와 LID들의 공간 지역성을 이용한다. 각 이동 단말은 시간 간격 $[t-T, t]$ 동안에 다른 등록지역의 원격 이동 단말들로부터 받은 단위 시간 당 평균 질의의 개수가 정해진 임계치를 초과하면 hot 이동 단말이고, 아니면 cold 이동 단말이다. LID들의 논리적 링 구조에서 로컬 LID를 중심으로 앞쪽으로 m 거리 내에 있거나 뒤쪽으로 m 거리 내에 있는 LID들의 집합을 NLID(Near LIDs)라 한다. 그리고 로컬 LID에서 m 거리 이상 떨어진 LID들의 집합을 LDLID(Long Distance LIDs)라 한다. Hot 이동 단말인 경우, 다른



(a) Hot 이동단말인 경우



(b) Cold 이동단말인 경우

(그림 3) 갱신 정책의 예

말이 cold이면 $a_3 = \lfloor \frac{2m+1}{k_3} \rfloor$, $a_4 = \lfloor \frac{n-(2m+1)}{k_4} \rfloor$ 이다. 따라서 각각의 경우에, 갱신되어야 할 나머지 k_1-1 , k_2-1 , k_3-1 , k_4-1 데이터베이스들은 논리적 링 상에서 이전에 갱신된 데이터베이스로부터 거리 a_1, a_2, a_3, a_4 로 결정적으로 선택된다. 여기서 우리는 $a_1 < a_2 = a_3 < a_4$ 로 설정한다. 그리고 시스템내의 LID들은 D_0, D_1, \dots, D_{n-1} 로 라벨하면, $NLID = \{D_{i-m}, \dots, D_{i+m}\}$ 이고 $LDLID = \{D_{i-m-1}, \dots, D_{i+m+1}\}$ 가 된다(그림 3 참조).

(그림 4)는 적응적 분산 동적 위치 관리의 갱신 알고리즘을 보여준다. 여기서 MT_i 는 이동 단말의 식별자, D_i 는 MT_i 의 로컬 LID의 식별자, \oplus 는 집합 $\{0, 1, 2, \dots, n-1\}$ 상에서 정의된 modulo n 덧셈을, \ominus 는 식 (1)의 연산을 실행한다.

$$i \ominus j = \begin{cases} (i-j) & \text{if } (i-j) \geq 0 \\ n-(i-j) & \text{otherwise} \end{cases} \quad (1)$$

```

Update (MTi, Dj)
n : integer /* Number of LIDs */
m : integer /* Distance from LIDi for space locality */
k1 : integer /* In the case that MTi is hot,
               replication factor in NLID */
k2 : integer /* In the case that MTi is hot,
               replication factor in LDLID */
k3 : integer /* In the case that MTi is cold,
               replication factor in NLID */
k4 : integer /* In the case that MTi is cold,
               replication factor in LDLID */
x, y, r1, r2, cnt1, cnt2 : integer ;
begin
Delete old location information from LIDs ;
if (MTi == hot)
x := k1 ;
y := k2 ;
r1 := ⌊  $\frac{2m+1}{k_1}$  ⌋ ;
r2 := ⌊  $\frac{n-(2m+1)}{k_2}$  ⌋ ;
else
x := k3 ;
y := k4 ;
r1 := ⌊  $\frac{2m+1}{k_3}$  ⌋ ;
r2 := ⌊  $\frac{n-(2m+1)}{k_4}$  ⌋ ;
cnt1 := 1 ;
cnt2 := 0 ;
Update Di ;
while (cnt1 < x)
Update Di ⊕ (cnt1 × r1) ;
cnt1 := cnt1 + 1 ;
if (cnt1 <= x)
Update Di ⊖ (cnt1 × r1) ;
cnt1 := cnt1 + 1 ;
Else

```

```

Exit ;
done ;
while (cnt2 <= y)
Update Di ⊕ (cnt2 × r2) ;
cnt2 := cnt2 + 1 ;
if (cnt2 <= y)
Update Di ⊖ (cnt2 × r2) ;
cnt2 := cnt2 + 1 ;
Else
Exit ;
done ;
end.

```

(그림 4) 위치 갱신 알고리즘

4.2 위치 질의 정책

위치 질의 정책은 로컬 LID에서 시작하여 그 위치 정보가 발견될때까지 위치 정보 데이터베이스를 계속적으로 검색한다. 이 알고리즘은 로컬 LID를 i 라 두면, $D_i, D_{i+1}, \dots, D_{i+n}$ 순서로 순차 검색을 수행한다. 최적 경우에(위치를 찾는 목적 이동 단말이 hot이고, 그 목적 이동 단말의 위치 정보가 NLID내의 LID에 존재), 그 탐색이 최대 $\lfloor \frac{2m+1}{k_1} \rfloor$ 데이터베이스까지 계속된다. 그리고 최악 경우에(위치를 찾는 목적 이동 단말이 cold이고, 그 목적 이동 단말의 위치 정보가 LDLID내의 LID에 존재), 그 탐색이 최대 $\lfloor \frac{n-(2m+1)}{k_4} \rfloor$ 데이터베이스까지 계속된다.

(그림 5)는 적응적 분산 동적 위치 관리의 질의 알고리즘을 보여준다. 여기서 MT_i 는 목적 이동 단말을 찾는 소스 이동 단말의 식별자이고, D_i 는 MT_i 가 거주하는 로컬 LID의 식별자를 나타낸다.

```

Query (MTi, Dj) ;
cnt : integer ;
find : boolean ;
begin
find := false ;
cnt := 0 ;
while (find == false)
if (location information in Di ⊕ cnt)
return location information ;
find := true ;
else
cnt := cnt + 1 ;
done ;
end.

```

(그림 5) 위치 질의 알고리즘

5. 성능 평가

본 논문에서는 제안하는 적응적 분산 동적 위치 관리 알고리즘의 성능을 분석적 모델과 시뮬레이션을 통하여 평가한다.

5.1 분석적 모델

이 장에서 3가지 성능 평가 척도 : 평균 위치 갱신 비용, 평균 위치 질의 지연, 그리고 전체 위치 관리비용으로 제안하는 알고리즘의 성능을 평가한다. 성능 분석에서 사용되는 기호는 다음과 같다.

- X_i 는 i -번째 질의된 데이터베이스가 요구된 정보를 가지는지를 가리키는 이진 랜덤 변수
- T 는 위치 정보를 찾기 위한 질의의 개수를 나타낸다. 우리는 T 를 질의 지연(query delay)이라 한다.

[명제 1] 제안하는 알고리즘의 평균 위치 갱신비용은 $\Gamma = 2 \times \left\{ p_h \left(\lfloor \frac{2m+1}{\alpha_1} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_2} \rfloor \right) + (1-p_h) \left(\lfloor \frac{2m+1}{\alpha_3} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_4} \rfloor \right) \right\}$ 이다. 여기서 p_h 는 등록지역을 갱신하는 이동 단말이 hot일 확률을 나타낸다.

증명 : 시스템 (n, k) 가 주어지면, $\alpha = \lfloor \frac{n}{k} \rfloor$ 이고 $k = \lfloor \frac{n}{\alpha} \rfloor$ 이다.

(1) 등록 지역의 위치를 변경한 이동 단말이 hot인 경우의 위치 갱신 비용(Γ_h) :

① NLID에서 위치 등록 회수

$$\alpha_1 = \lfloor \frac{2m+1}{k_1} \rfloor \text{ 이므로 } k_1 = \lfloor \frac{2m+1}{\alpha_1} \rfloor \text{ 이다.}$$

② LDLID에서 위치 등록 회수

$$\alpha_2 = \lfloor \frac{n-(2m+1)}{k_2} \rfloor \text{ 이므로 } k_2 = \lfloor \frac{n-(2m+1)}{\alpha_2} \rfloor \text{ 이다.}$$

이동 단말의 새로운 위치 정보가 등록되는 위치 데이터베이스의 개수는 k_1+k_2 이므로 $\lfloor \frac{2m+1}{\alpha_1} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_2} \rfloor$ 이다. 따라서 등록 지역을 위치를 변경한 이동 단말의 예전 위치 정보는 위치 데이터베이스들로부터 제거되고 새로운 위치 정보는 등록되지므로 Γ_h 는 식 (2)와 같다.

$$\Gamma_h = 2 \times \left(\lfloor \frac{2m+1}{\alpha_1} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_2} \rfloor \right) \quad (2)$$

(2) 등록 지역의 위치를 변경한 이동 단말이 cold인 경우의 위치 갱신 비용(Γ_c) :

① NLID에서 위치 갱신 회수

$$\alpha_3 = \lfloor \frac{2m+1}{k_3} \rfloor \text{ 이므로 } k_3 = \lfloor \frac{2m+1}{\alpha_3} \rfloor \text{ 이다.}$$

② LDLID에서 위치 갱신 회수

$$\alpha_4 = \lfloor \frac{n-(2m+1)}{k_4} \rfloor \text{ 이므로 } k_4 = \lfloor \frac{n-(2m+1)}{\alpha_4} \rfloor \text{ 이다.}$$

$$k_4 = \lfloor \frac{n-(2m+1)}{\alpha_4} \rfloor \text{ 이다.}$$

이동 단말의 새로운 위치 정보가 등록되는 위치 데이터베이스의 개수는 k_3+k_4 이므로 $\lfloor \frac{2m+1}{\alpha_3} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_4} \rfloor$ 이다. 따라서 등록 지역을 위치를 변경한 이동 단말의 예전 위치 정보는 위치 데이터베이스들로부터 제거되고 새로운 위치 정보는 등록되지므로 Γ_c 는 식 (3)과 같다.

$$\Gamma_c = 2 \times \left(\lfloor \frac{2m+1}{\alpha_3} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_4} \rfloor \right) \quad (3)$$

$\Gamma = p_h \Gamma_h + (1-p_h) \Gamma_c$ 이므로 식 (2)와 식 (3)에 의해 평균 위치 갱신 비용은

$$\Gamma = 2 \left\{ p_h \left(\lfloor \frac{2m+1}{\alpha_1} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_2} \rfloor \right) + (1-p_h) \left(\lfloor \frac{2m+1}{\alpha_3} \rfloor + \lfloor \frac{n-(2m+1)}{\alpha_4} \rfloor \right) \right\}$$

[명제 2] 제안하는 알고리즘의 평균 위치 질의 지연은

$$\Theta = \frac{1}{2} \left[p_h \left\{ \frac{2m+1}{n} (\alpha_1+1) + \frac{n-(2m+1)}{n} (\alpha_2+1) \right\} + (1-p_h) \left\{ \frac{2m+1}{n} (\alpha_3+1) + \frac{n-(2m+1)}{n} (\alpha_4+1) \right\} \right]$$

이다. 여기서 p_h 은 소스 이동 단말이 위치를 찾는 목적 이동 단말이 hot일 확률을 나타낸다.

증명 : $j=1, 2, \dots, n$ 에 대해 $p_j = P(T=j)$ 이라 하자.

$$p_j = P\left(\sum_{i=1}^{j-1} X_i = 0, X_j = 1\right)$$

(1) 목적 이동 단말이 hot인 경우의 평균 질의 지연(Θ_h)

$$\Theta_h = \min\left(\frac{2m+1}{n} \sum_{j_1=1}^{\alpha_1} j_1 p_{j_1} + \frac{n-(2m+1)}{n} \sum_{j_2=1}^{\alpha_2} j_2 p_{j_2}\right)$$

$$p_{j_1} = \frac{k_1}{2m+1}, p_{j_2} = \frac{k_2}{n-(2m+1)},$$

$$k_1 = \frac{2m+1}{\alpha_1}, k_2 = \frac{n-(2m+1)}{\alpha_2}$$

따라서 Θ_h 는 식 (4)와 같다.

$$\Theta_h = \frac{1}{2} \left\{ \frac{2m+1}{n} (\alpha_1+1) + \frac{n-(2m+1)}{n} (\alpha_2+1) \right\} \quad (4)$$

(2) 목적 이동 단말이 cold인 경우의 평균 질의 지연(Θ_c)

$$\Theta_c = \min\left(\frac{2m+1}{n} \sum_{j_3=1}^{\alpha_3} j_3 p_{j_3} + \frac{n-(2m+1)}{n} \sum_{j_4=1}^{\alpha_4} j_4 p_{j_4}\right)$$

$$p_{j_3} = \frac{k_3}{2m+1}, p_{j_4} = \frac{k_4}{n-(2m+1)},$$

$$k_3 = \frac{2m+1}{a_3}, k_4 = \frac{n-(2m+1)}{a_4}$$

따라서 Θ_c 는 식 (5)와 같다.

$$\Theta_c = \frac{1}{2} \left\{ \frac{2m+1}{n}(a_3+1) + \frac{n-(2m+1)}{n}(a_4+1) \right\} \quad (5)$$

$\Theta = p_h \Theta_h + (1-p_h) \Theta_c$ 이므로 식 (4)와 식 (5)에 의해 평균 위치 질의 지연은

$$\Theta = \frac{1}{2} \left[p_h \left\{ \frac{2m+1}{n}(a_1+1) + \frac{n-(2m+1)}{n}(a_2+1) \right\} + (1-p_h) \left\{ \frac{2m+1}{n}(a_3+1) + \frac{n-(2m+1)}{n}(a_4+1) \right\} \right]$$

이다. ■

이동 당 호 도착율을 λ , 등록 지역의 이동 단말의 평균 거주 시간을 $\frac{1}{\mu}$ 라 두면, 적응적 분산 동적 위치 관리를 위한 전체 비용은 식 (6)과 같다.

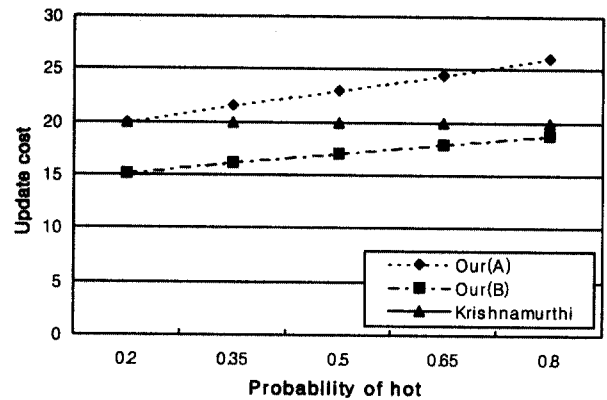
$$Total = Update + \frac{\lambda}{\mu} \times Query \quad (6)$$

식 (6)에서 $\frac{\lambda}{\mu}$ 를 호 이동율(CMR, call-mobility ratio)이라 하고, *Update*와 *Query*는 평균 위치 갱신 비용과 평균 위치 질의 비용을 각각 나타낸다.

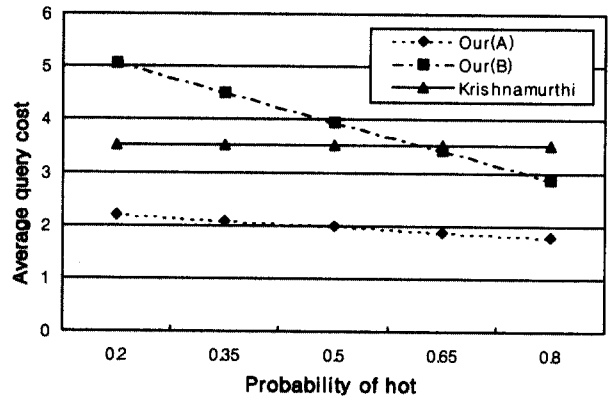
위치 관리 알고리즘들의 분석적 성능 평가 결과를 (그림 6)에서 보여준다. 위치 갱신 비용과 위치 질의 비용은 [명제 1], [명제 2]에 <표 1>의 매개변수 값을 적용하여 구하였고, 전체 비용은 <표 1>의 매개변수 값에 의해 도출된 [명제 1], [명제 2]의 값을 식 (6)에 적용하여 구하였다. (그림 6(a))는 등록 지역의 위치를 갱신한 이동 단말이 hot일 확률에 따른 위치 갱신 비용을 보여주고, (그림 6(b))는 소스 이동 단말이 위치를 찾는 목적 이동 단말이 hot일 확률에 따른 위치 질의 비용을 보여준다. 여기서 사례 A는 LID에서 위치 정보가 저장되는 사본의 개수가 많은 만큼 위치 갱신 비용은 많이 들고, 위치 정보의 사본을 저장하고 있는 LID들간의 거리가 짧으므로 위치 질의 비용은 적게 든다. 반면에 사례 B는 LID에서 위치 정보가 저장되는 사본의 개수가 작은 만큼 위치 갱신 비용은 적게 들고, 위치 정보의 사본을 저장하고 있는 LID들간의 거리가 길어지므로 위치 질의 비용이 많이 든다. 그리고 (그림 6(c))는 위치를 갱신한 이동 단말이 hot일 확률이 0.35이고 소스 이동 단말이 위치를 찾는 목적 이동 단말이 hot일 확률이 0.65일 때 호 이동율에 따른 전체 위치 관리 비용을 보여준다. 호 이동율에 관계없이 본 논문에서 제안하는 적응적 분산 동적 위치 관리 알고리즘이 Krishnamurthi의 알고리즘에 비해 성능이 우수함을 알 수 있다. 특히, 호 이동율이 높을수

<표 1> 분석적 평가를 위한 매개변수와 값

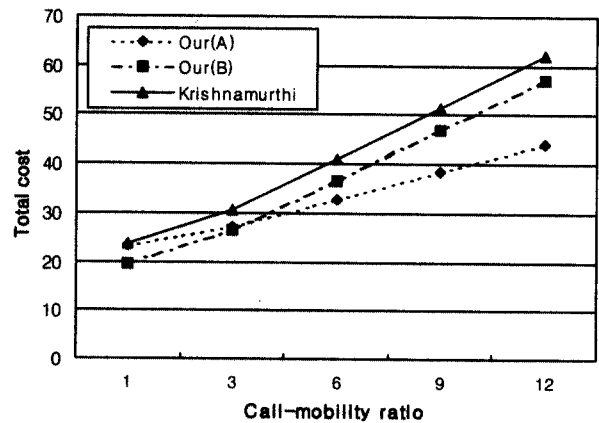
매개변수		값
n		60
m		10
a		6
사례 A	a_1	3
	a_2, a_3	5
	a_4	7
사례 B	a_1	4
	a_2, a_3	7
	a_4	9



(a) 위치 갱신 비용



(b) 위치 질의 비용



(c) 전체 비용

(그림 6) 분석적 성능 평가 결과

록 사례 A의 위치 질의 비용이 줄어들어 사례 A의 성능이 더 좋아짐을 알 수 있었다.

5.2 시뮬레이션

본 논문에서는 시뮬레이션을 통하여 제안하는 적응적 분산 동적 위치 관리 알고리즘의 성능을 평가하였다. 시뮬레이션에서 사용된 가정은 다음과 같다.

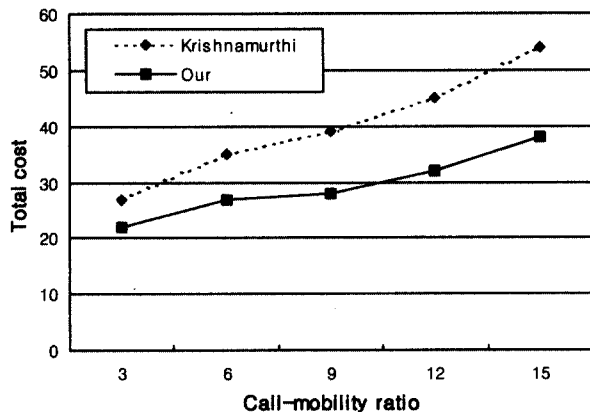
- n, m, 그리고 LID에서 사본인 수는 <표 1>의 사례 A에 따른다.
- 위치를 찾는 목적 이동 단말의 로컬 LID를 30이라 한다.

그리고 시뮬레이션에 사용된 매개변수와 값은 <표 2>와 같다. 여기서 목적 이동 단말의 위치를 찾는 소스 이동 단말의 위치는 공간 지역성을 고려하여 정규분포로 하였다.

<표 2> 시뮬레이션을 위한 매개변수와 값

매개 변수	값	비 고
위치 갱신 이동 단말이 hot일 확률(랜덤 분포)	random(0, 1)	0 : cold 1 : hot
목적 이동 단말이 hot일 확률(랜덤 분포)	random(0, 1)	0 : cold 1 : hot
소스 이동 단말의 위치 (가우스 분포)	normal(30.0, 1.0)	30.0 : 평균 1.0 : 표준편차

(그림 7)은 이동 단말이 10번의 위치 갱신을 하는 동안에 호 이동율에 따른 전체 위치 관리 비용을 보여준다. 분석적 평가 결과와 마찬가지로 호 이동율에 관계없이 제안하는 적응적 분산 동적 위치 관리 알고리즘이 Krishnamurthi의 알고리즘보다 성능이 우수함을 알 수 있다. 특히 호 이동율이 높을수록 적응적 분산 동적 위치 관리 알고리즘의 성능이 더 좋아짐을 알 수 있다. 이것은 호 이동율이 낮을 경우 위치 갱신 비용의 증가로 제안하는 알고리즘의 전체 위치 관리비용의 감소폭은 적으나, 호 이동율이 높아질수록 위치 질의 시 이동 단말의 공간 지역성을 이용하기 때문에 위치 질의 비용이 Krishnamurthi의 알고리즘에 비해 대폭 감소



(그림 7) 시뮬레이션 성능 평가 결과

하여 결국 제안하는 알고리즘의 전체 위치 관리비용이 큰 폭으로 줄어들기 때문이다.

6. 결 론

위치 정보의 효율적 관리는 미래의 이동망 설계에서 고려되어야 할 중요한 사항이다. 본 논문에서는 로컬 이동 단말에 대한 다른 등록 지역의 이동 단말로부터의 위치 질의 선호도와 MSC들과 대응하는 위치 정보 데이터베이스들의 공간 지역성을 이용하는 적응적 분산 동적 위치 관리 알고리즘을 설계하였고, 그리고 제안하는 알고리즘의 성능을 분석적 모델과 시뮬레이션을 통하여 각각 평가하였다. 성능 평가 결과, 호 이동율에 상관없이 제안하는 적응적 분산 동적 위치 관리 알고리즘이 Krishnamurthi의 알고리즘에 비해 우수한 성능을 보였다. 특히, 제안하는 알고리즘은 위치 질의 시 이동 단말의 공간 지역성을 이용하기 때문에 호 이동율이 높을수록 성능이 더 우수하였다.

참 고 문 헌

- [1] R. Jain, Y.-B. Lin, S. Mohan, "A Caching Strategy to Reduce Network Impacts of PCS," *IEEE Journal on Selected Areas of Communication*, 12(8), pp.1434-1445, October, 1994.
- [2] R. Jain, Y.-B. Lin, S. Mohan, "A Forwarding Strategy to Reduce Network Impacts of PCS," *In Proc. of IEEE Infocom*, pp.481-489, April 1995.
- [3] Joseph S. M. Ho, Ian F. Akyildiz, "Local Anchor Scheme for Reducing Signaling Costs in Personal Communication Networks," *IEEE/ACM Transactions on Networking*, Vol. 4, No.5, pp.709-725, 1996.
- [4] G. Dommetry, M. Veeraraghavan, M. Singhal, "Flat Location Management for PCNs," *In Proc. of Int. Conf. on Universal Personal Communications*, pp.146-152, 1997.
- [5] E. Pitoura and Ioannis Fudos, "An Efficient Hierarchical Scheme for Locating Highly Mobile Users," *The Computer Journal*, 44(2), pp.75-91, 2001.
- [6] R. Prakash, M. Singhal, "A Dynamic Approach to Location Management in Mobile Computing Systems," *Proc. SEKE '96*, pp.488-495, 1996.
- [7] G. Krishnamurthi, M. Azizoglu, A. K. Somani, "Optimal Location Management Algorithms for Mobile Networks," *Proc. MOBICOM98*, pp.223-232, 1998.
- [8] G. Krishnamurthi, A. K. Somani, "Effect of Failures on Optimal Location Management Algorithms," *In Proc. of FTCS 99*, pp.110-119, 1999.
- [9] S. K. Das and S. K. Sen, "Adaptive Location Prediction Strategies Based on a Hierarchical Network Model in a Cellular Mobile Environment," *The Computer Journal*, 42(6), 1999.

천성광

e-mail : skchun@etri.re.kr, ebxml@hanmail.net

1989년 계명대학교 물리학과(학사)

1991년 계명대학교 물리학과(석사수료)

1997년 대구가톨릭대학교 전자계산학과
(이학석사)

2000년 대구가톨릭대학교 전자계산학과
(박사수료)

1996~2001 (주)대송기업 전산실

1998~2001 대구산업정보대학 컴퓨터정보계열 겸임교수

1998~2001 상주대학교 외래교수

2001~현재 한국전자통신연구원 우정기술연구센터 연구원

관심분야 : 이동무선망, 무선인터넷, 멀티미디어 시스템, XML,
M-Commerce 등

배인한

e-mail : ihbae@cataegu.ac.kr

1984년 경남대학교 전자계산학과(학사)

1986년 중앙대학교 대학원 전자계산학과
(석사)

1990년 중앙대학교 대학원 전자계산학과
(박사)

1989년~현재 대구가톨릭대학교 컴퓨터정보통신공학부 교수

1996년~1997년 Computer and Information Science, The Ohio
State University (Postdoc)

2002년~현재 Computer Science, Old Dominion University(교환
교수)

관심분야 : 이동무선망, 무선인터넷, 멀티미디어 시스템, 분산 시
스템, M-commerce 등