연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 기법

(Grid-based Cloaking Area Creation Scheme supporting Continuous Location-Based Services)

이 아 름* 김 형 일* 장 재 우**
(Ah-Reum Lee) (Hyeong-il, Kim) (Jae-Woo Chang)

요 약 최근 PDA, 휴대폰, GPS와 같은 모바일 기기 및 무선 통신 기술의 발달로 인하여 위치 기반 서비스의 이용이 확산되었다. 하지만 이러한 서비스는 사용자의 정확한 위치정보를 가지고 LBS 서버에 연속적으로 서비스를 요청하기 때문에, 심각한 개인 정보 누출의 위협이 될 수 있다. 따라서 모바일 사용자의 안전하고 편리한 위치기반 서비스 사용을 위한 개인 정보 보호 방법이 필요하다. 이를 위해 본 논문에서는 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 기법을 제안한다. 제안하는 기법은 연속적인 위치기반 서비스를 효율적으로 지원하기 위하여 그리드 기반의 셀 확장을 통해 빠르게 Cloaking 영역을 설정한다. 아울러, 모바일 사용자의 위치 노출 확률을 최소로 하는 Cloaking 영역 설정을 위하여, 가중치를 부여하여 프라이버시 보호 수준을 계산한다. 마지막으로 성능평가를 통해서 제안하는 기법이 서비스 시간, 프라이버시 보호 수준에서 기존 연구보다 우수함을 보인다.

키워드 : 개인 정보 보호, 연속적인 위치기반 서비스, Cloaking 기법

Abstract Recent development in wireless communication technology and mobile equipment like PDA, cellular phone and GPS makes location-based services (LBSs) popular. However, because, in the LBSs, users continuously request a query to LBS servers by using their exact locations, privacy information could be in danger. Therefore, a mechanism for users' privacy protection is required for the safe and comfortable use of LBSs by mobile users. For this, this paper propose a grid-based cloaking area creation scheme supporting continuous LBSs. The proposed scheme creates a cloaking area rapidly by using grid-based cell expansion to efficiently support the continuous LBSs. In addition, to generate a cloaking area which makes the exposure probability of a mobile user to a minimum, we compute a privacy protection degree by granting weights to mobile users. Finally, we show from a performance analysis that our cloaking scheme outperforms the existing cloaking schemes, in terms of service time, privacy protection degree.

Keywords : Privacy Protection, Continuous Location-Based Services, Cloaking Scheme

1. 서 론

최근 PDA, 휴대폰, GPS와 같은 모바일 기기 및 무선 통신 기술의 발달로 인하여 위치 기반 서비스의 이용이 확산되었다. 위치기반 서비스(Location-Based Service)란 유무선 통신망을 통해 얻은 위치정보를 유용한 정보와 결합하여 사용자가 필요로 하는 부가적인 응용 서비스를 제공하는 것이다[1,2]. 위치기반 서비스에서 모바일 사용자는 GPS 등과 같은 위치 측위 시스템을 이용하여, 사용자의 위치 정보를 LBS 서버에 보내어 교통 정보, 친구 찾기, 인접한 POI(Point Of Interest) 등 다양한 종류의 위치 기반 서비스를 이용할 수 있다. 그러나 이와 같이 사용자의 정확한 위치정보를 가지고 LBS 서버에 서비스를 요청하는 것은 심각한 개인 정보 누출의 위협이 될 수 있다. 왜냐하면 LBS 서버에 보내진 사용자의 위치 정보가 유/무선 통신상에서 유출되거나 LBS 서버를 관리하는 관리자를 통해 악용된다면, 이를 통해 서비스 이용자들이 어떤 장소에 자주 방문하는지, 또한 이러한 방문이 어떤 시간대

† 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. 2009-0059417)
† 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음(IITA-2008-(C1090-0801-) 0047))

* 전북대학교 전자정보공학부 컴퓨터공학과 석사과정, {arlee, khi}@dblab.chonbuk.ac.kr
** 전북대학교 전자정보공학부 컴퓨터공학과 교수, jwchang@jbnu.ac.kr(교신저자)

논문접수 : 2009.07.01 수정일 : 1차 2009.08.21 / 2차 2009.09.11 심사완료 : 2009.09.18

에 주로 이루어지는 지를 파악하여, 생활 스타일, 질병 정보 등의 사생활 정보를 획득할 수 있기 때문이다. 실제로 위치 기반 서비스를 이용한 스토킹이나 개인정보 유출 피해 사례가 빈번히 발생하고 있다[3,4]. 따라서 사용자의 안전하고 편리한 위치기반 서비스 사용을 위한 개인 정보 보호 방법이 요구된다.

이러한 개인 정보 보호를 위한 연구 중에는 K-Anonymity를 만족하는 Cloaking 기법에 대한 연구가 존재한다. 이 기법은, 사용자가 LBS 서버에 질의(서비스) 전송 시, 사용자의 좌표정보를 숨기기 위해 K-Anonymity를 만족하면서, 최소 크기를 가지는 질의 영역(이하 Cloaking 영역)을 설정하는 것을 말한다. 여기서 K-Anonymity는 Cloaking 영역에 질의를 요청한 사용자와 그 사용자를 제외한 K-1명의 다른 사용자의 위치 정보를 포함하는 것이다. 이러한 Cloaking 영역의 설정으로 사용자의 위치 노출 확률을 최소화 시킬 수 있다. 그러나 K-Anonymity를 만족하는 기존의 Cloaking 영역 설정 기법은 사용자가 이동하면서 연속적으로 서비스를 요청하는 경우, 생성되는 모든 Cloaking 영역에 질의를 요청한 사용자는 존재하지만, 다른 사용자는 이동 경로에 따라 Cloaking 영역에 포함되지 않는 경우가 존재한다. 이러한 문제점을 극복하기 위해, Cloaking 영역의 프라이버시 보호 수준을 고려하여 연속된 위치기반 서비스를 제공하는 Toby Xu와 Ying Cai의 연구[5]인 Advanced KAA가 제안되었다. Advanced KAA는 연속적으로 서비스를 요청하는 사용자의 프라이버시를 보호하면서 최소의 Cloaking 영역을 설정하기 위해 Cloaking 영역의 프라이버시 보호 수준을 계산하여 Cloaking 영역을 설정한다. 그러나 Advanced KAA는 두 가지 문제점을 지니고 있다. 첫째, 최소의 Cloaking 영역을 설정하기 위하여 생성할 수 있는 모든 후보 영역들을 고려하기 때문에 Cloaking 영역을 설정하는 시간이 증가하는 문제점을 지니고 있다. 둘째, 프라이버시 보호 수준을 계산하기 위해 무작위 데이터 샘플링을 통하여 사용자의 확률 값을 계산하기 때문에, 이전 Cloaking 영역에 포함된 많은 사용자가 설정한 Cloaking 영역에 포함되지 않는 경우가 발생한다. 따라서 서비스를 요청한 사용자의 프라이버시를 침해하는 문제점을 지닌다.

따라서 본 논문에서는 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 기법을 제안한다. 이 기법은 연속적인 위치기반 서비스를 효율적으로 지원하기 위하여 그리드 기반의 셀 확장을 통해 빠르게 Cloaking 영역을 설정한다. 아울러, 모바일 사용자의 위치 노출 확률을 최소로 하는 Cloaking 영역 설정을 위하여, 사용자에게 가중치를 부여하여 프라이버시 보호 수준을 계산한다.

본 논문의 구성은 다음과 같다. 2장에서는 K-anonymity를 고려하면서 연속적인 위치기반 서비스를 지원하는 기존 Cloaking 기법을 소개한다. 3장에서는 기존 연구의 문제점을 개선한 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 기법을 제안하고, 4장에서는 기존 연구와 제안하는 기법과의 성능비교를 수행한다. 마지막으로 5장에서는 결론 및 향후 연구에 대해 기술한다.

2. 관련 연구

기존 K-anonymity를 만족하는 Cloaking 영역을 생성하는 연구들[6,7,8,9,10,11]은, 사용자가 연속적으로 질의를 요청했을 경우 사용자의 프라이버시를 보호해주지 못한다. 연속적인 질의의 경우, 모든 Cloaking 영역에는 실제로 질의를 요청한 사용자가 항상 포함되어 있기 때문이다. 즉, 기존 연구들은 매 시간에서의 Cloaking 영역이 K-anonymity를 만족하도록 보장해주지만, 연속적인 질의의 경우 질의를 요청한 사용자의 개인 정보가 노출될 가능성이 존재한다. 그림 1은 이러한 문제점을 보여준다. 질의를 요청한 사용자(N)가 k=5를 요청한다고 가정한다. 매 시간에서의 Cloaking 영역은 사용자가 요구하는 k를 만족한다. 하지만, 초기 Cloaking 영역(T=0) 안에 포함된 사용자들 중, T=2초일 때의 Cloaking 영역에 남아있는 사용자는 N 뿐이다. 이러한 경우, 상대방은 N이 질의를 요청한 사용자라는 것을 알 수 있다.

이러한 문제를 해결하기 위해, 초기 Cloaking 영역에 포함된 모든 사용자들이, 이후 Cloaking 영역에 항상 포함되도록 할 수 있다. 하지만, 초기 Cloaking 영역에 포함된 사용자들의 움직임에 따라 Cloaking 영역이 비효율적으로 커질 수도 있다는 단점이 존재한다. 그림 2는 이러한 문제점을 보여준다. N이 k=5를 요청했다고 하자.

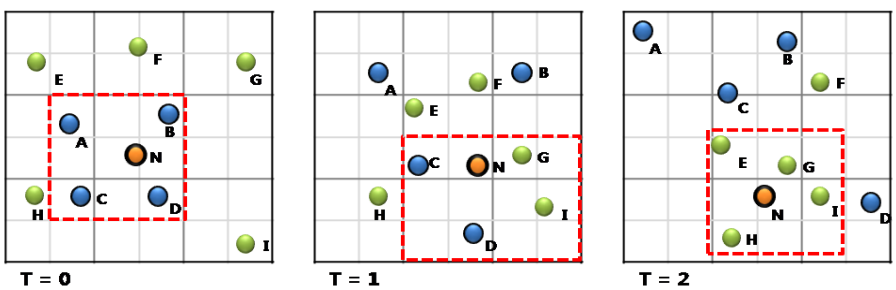


그림 1. 연속적인 질의처리 수행 시 발생하는 문제

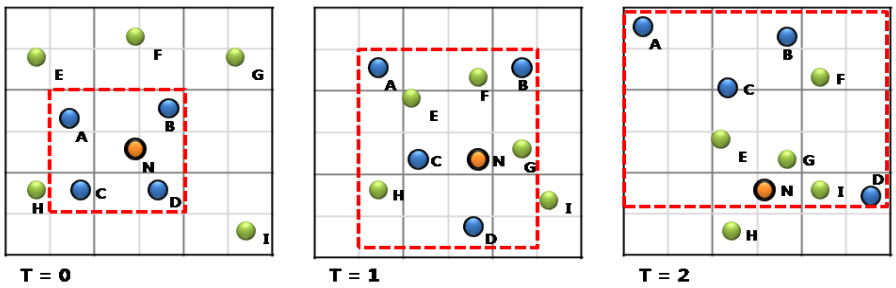


그림 2. 연속적인 질의처리를 위해 모든 사용자를 유지할 때 발생하는 문제

T=0일 때 Cloaking 영역에 포함된 사용자는 S(A) = {N, A, B, C, D}이므로, 이들은 매 시간 Cloaking 영역에 포함되어야 한다. T=1, T=2의 Cloaking 영역은 이들을 경계로 한 Cloaking 영역을 생성한 모습을 보여준다. 이 때, 매 시간에서의 Cloaking 영역은 초기 Cloaking 영역에 속한 사용자들을 모두 포함하기 때문에, 요구한 k를 만족한다. 하지만, 시간이 흐름에 따라 Cloaking 영역이 커짐을 알 수 있다.

이러한 문제점을 다룬 연구로는 Toby Xu와 Ying Cai의 연구[5]가 유일하다. 일반적으로, 초기 Cloaking 영역에 포함된 사용자들을 항상 포함하며 Cloaking 영역을 생성한다면, 시간이 지날수록 Cloaking 영역의 크기는 커지며, 새로 포함된 사용자의 수 또한 증가하게 된다. 이들은 새로 포함된 사용자들이 Anonymity에 기여하는 수준을 계산하여, K-anonymity를 만족하는 최소의 원을 찾아, 이를 Cloaking 영역으로 결정하는 Advanced KAA(K-anonymity Area)를 제안하였다. Advanced KAA는 Cloaking 영역의 프라이버시 보호 수준을 측정하기 위해서 엔트로피(entropy)를 사용한다. 엔트로피는 Cloaking 영역 안에 있는 사용자들 중에서 실제로 질의를 요청한 사용자가 누구인지 알기 위해 추가로 필요한 정보의 양을 의미한다. 사용자 N을 위한 Cloaking 영역을 A, Cloaking 영역 A에 포함된 사용자들의 집합을 $S(A) = \{N_1, N_2, \cdots, N_m\}$라 할 때, A의 엔트로피는 식 (1)과 같이 계산한다.

$$H(A) = -\sum_{i=1}^{m} p_i \log p_i \quad (1)$$

여기에서 H(A)는 Cloaking 영역 A의 엔트로피를 나타내고, $p_i (1 \leq i \leq m)$는 각 사용자가 실제 질의를 요청한 사용자 N이 될 확률을 의미한다. 또한, 식 (1)에 의해 구해진 엔트로피를 식 (2)에 대입하여 A의 Anonymity Degree를 구한다.

$$D(A) = 2^{H(A)} \quad (2)$$

여기에서 D(A)는 A의 Anonymity Degree를 의미한다. 예로, N이 요청한 k=4이며, 초기 Cloaking 영역에 속한 사용자의 수가 4라고 가정하자. 초기 Cloaking 영역에서, 각 사용자가 N이 될 확률 p_i는 1/4로 모두 같다. 식 (1)에 의해 엔트로피 H(A) = log4 = 2가 된다. 또한, 식 (2)에 의해 A의 D(A) = 2^2 = 4가 된다. 이는 N이 요청한 K-anonymity를 만족함을 알 수 있다.

초기 시간이 아닌 경우에 각 노드의 확률을 구하기 위해서는 다음과 같은 방법을 사용한다. 먼저, 이전 시간의 어떤 사용자가 나중 시간의 어떤 사용자의 위치로 움직일 수 있는 α개의 샘플을 생성하여, 이를 통해 이동확률 매트릭스(transition matrix) M을 생성한다. M은 이러한 모든 움직임을 반영하며, M의 각 셀이 갖는 값 n_{xy}는 α개의 샘플 중에서, 이전 시간의 사용자 x가 나중 시간의 사용자 y로 움직인 샘플의 수를 의미한다. 나중 시간 사용자들의 확률 집합은 식 (3)을 통해 구해진다.

$$P_t = P_{t-1} \times M \quad (3)$$

여기에서 P_t는 나중 시간 사용자들의 확률 집합을 의미하고, P_{t-1}은 이전 시간 사용자들의 확률 집합을 의미한다.

3. 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 기법

본 장에서는 제안하는 기법의 설계 시 고려사항에 대하여 기술한다. 또한, 이를 통해 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 알고리즘을 제안한다.

3.1 설계 시 고려사항

본 연구에서는 Cloaking 영역을 설정하는 주체가 Anonymizer인 중앙 집중(Centralized)방식을 사용하며, 시스템 구조는 그림 3과 같다. Anonymizer란, 모바일 사용자와 LBS 서버 중간에 존재하는 신뢰할 수 있는 서버이다. 시스템 구조는 크게 모바일 사용자와 Anonymizer, LBS 서버로 구성된다. 모바일 사용자는 Anonymizer로 사용자의 위치 정보가 포함된 질의와 서비스 시간을 전송하고, 서비스 시간동안 주기적으로 모바일 사용자의 위치정보를 갱신한다. Anonymizer는 전송받은 모바일 사용자의 질의를 수행하기 위해 세션 ID를 임의로 생성하

며, 이때 생성된 세션 ID는 해당 모바일 사용자의 서비스 시간동안 지속해서 사용된다. 또한, Anonymizer는 k-1명의 인접한 다른 모바일 사용자들의 위치정보를 사용하여 (k-anonymity) Cloaking 영역을 설정한 후, 세션 ID를 통해 Cloaking영역을 LBS 서버로 전송한다. 본 연구에서는 모바일 사용자를 이동 객체로 고려하기 때문에 특정시간에 설정된 Cloaking 영역은 질의를 요청한 모바일 사용자에게만 유효하다. 즉, k-1명의 인접한 모바일 사용자들은 Cloaking 영역의 설정을 위해서만 사용될 뿐, 질의 결과를 위한 서비스 요청을 하지 않는다. 한편, LBS 서버는 전송받은 Cloaking 영역을 바탕으로 요청된 질의를 처리한다. Anonymizer는 LBS 서버에서 전송된 질의 수행 결과를 저장된 사용자의 위치정보를 바탕으로 필터링 한 뒤, 서비스를 요청한 사용자에게 정확한 질의 결과를 전송한다.

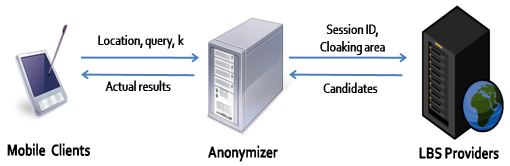


그림 3. 제안하는 기법의 시스템 구조

또한, 본 연구에서는 GPS와 같은 위치 측위 시스템을 이용하여 획득한 사용자의 위치 정보를 LBS 서버에 보내어 질의를 수행하는 위치기반 서비스를 고려한다. 하지만 상대방(Adversary)이 사용자의 정확한 위치 정보를 알게 될 경우, 사용자의 프라이버시를 침해할 위협 수준이 높아진다. 여기서 상대방이란 개인 정보를 획득하여 이를 악의적인 목적에 이용하는 사람을 의미한다. 즉, 상대방은 유선 또는 무선 통신을 감청함으로써 LBS 서버로부터 데이터를 획득하거나, 그가 알고자 하는 정보에 대한 사전 지식을 가지고 사용자의 개인 정보를 획득한다. 심지어는 서비스 제공자 자체가 상대방이 되어 사용자의 개인 정보를 상업적 목적으로 사용하거나 사용자 위치 정보를 제 3자에게 판매한다[12]. 사용자는 서비스를 이용하기 위해 지속적으로 자신의 위치정보와 질의를 LBS 서버에 전송한다. 상대방은 사용자가 전송한 정보와 유지하고 있는 POI(e,g, 주유소, 레스토랑) 정보를 획득한 후, 이를 통해 사용자의 개인정보를 유추할 수 있다. 이와 같은 문제점을 해결하기 위하여, 사용자는 인접한 다른 사용자의 정보를 포함한 Cloaking 영역을 설정하여 자신의 위치노출 확률을 줄인다. 하지만 이동 중인 사용자가 연속적으로 서비스를 요청하는 경우, 질의를 요청한 사용자의 이동 방향에 따라 Cloaking 영역이 연속적으로 설정되지만, 다른 사용자는 질의를 요청한 사용자와 다른 이동 경로를 가질 수 있으므로, 설정되는 Cloaking 영역에 포함되지 않는 경우가 존재한다.(그림 1 참고). 이와 같은 경우, 상대방은 연속적으로 전송되는 Cloaking 영역에서 질의를 요청한 사용자를 구분할 수 있고 이를 통해 사용자의 정확한 위치 및 개인정보를 유추할 수 있다. 이러한 문제점은 Cloaking 영역의 프라이버시 보호 수준을 고려하여 연속된 위치기반 서비스를 제공함으로써 해결할 수 있다. 연속적인 위치기반 서비스를 제공하는 기존의 연구는 Advanced KAA가 유일하다. 그러나 Advanced KAA는 다음의 두 가지 문제점을 지닌다. 첫째, 최소의 Cloaking 영역을 설정하기 위하여 생성할 수 있는 모든 후보 영역들을 고려하기 때문에, Cloaking 영역을 설정하는 시간이 증가하는 문제점을 지니고 있다. 예를 들어, 그림 4와 같이 Cloaking 영역을 설정할 수 있는 임시 영역이 존재할 경우, 임시 영역에서 생성할 수 있는 모든 후보 영역을 고려한 후, 최소 크기를 갖는 Cloaking 영역을 설정한다. 따라서 임시 영역 안의 모든 후보 영역을 계산하기 위한 Cloaking 영역 설정 시간이 NP-hard, 즉 polynomial 만큼 늘어나고, 이는 전체 서비스 시간을 증가시키는 문제점을 갖는다. 따라서 효율적으로 모바일 사용자에게 서비스를 제공하기 위하여 Cloaking 영역 설정 시간을 줄이는 방법이 필요하다.

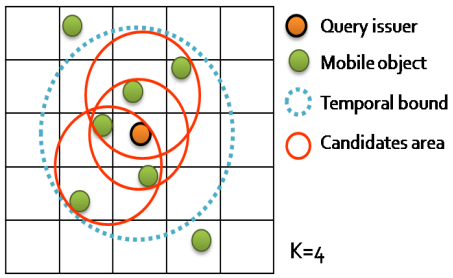


그림 4. Advanced KAA의 후보 영역을 고려한 Cloaking 영역 설정

둘째, 프라이버시 보호 수준을 계산하기 위해 무작위 데이터 샘플링을 통하여 사용되는 사용자의 확률 값을 계산하기 때문에, 이전 Cloaking 영역에 포함된 많은 사용자들이 설정되는 Cloaking 영역에 포함되지 않는 경우가 발생한다. 예를 들어, 초기에 Cloaking 영역 설정 시 포함되었던 사용자들의 확률 값이 서비스 수행 시간 동안 무작위 데이터 샘플링에 의해 감소함에 따라 최악의 경우, 마지막에 생성되는 Cloaking 영역에 서비스를 요청한 사용자만 남는 경우가 발생하게 된다. 이는 사용자의 프라이버시를 침해하는 문제점을 갖게 된다. 이를 해결하기 위해 이전 Cloaking 영역에 포함되는 사용자들이 설정되는 Cloaking 영역에 최대한 포함될 수 있도록 하는 방법이 필요하다.

따라서 본 연구에서는 위와 같은 문제점을 해결하기 위해, 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 알고리즘을 제안한다.

3.2 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 알고리즘

본 절에서는 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 알고리즘에 대해 제안한다. 제안하는

알고리즘은 모바일 사용자가 연속적인 질의요청 시 최소의 Cloaking 영역을 설정하는 부분과 서비스 수행 시간 동안 사용자의 프라이버시를 보장하면서 Cloaking 영역을 설정하는 부분으로 구성된다. 먼저 수행단계 1은 사용자가 서비스를 요청한 초기, 즉 T=0인 경우에 그리드 기반의 셀 확장을 통해 Cloaking 영역을 빠르게 설정하는 방법을 기술한다.

수행단계 1. 서비스 요청시 K-anonymity를 만족하는 Cloaking 영역 설정(T=0)

질의를 요청한 사용자가 위치한 그리드 셀을 중심으로, 사방으로 한 셀씩 확장된 영역에 위치한 셀을 탐색한다. 탐색 중, 확장된 영역 내에 포함된 사용자의 수(k′)가 사용자가 요구한 k 값보다 크면 셀 탐색을 중단하고, 셀 탐색을 통해 선택된 그리드 셀을 포함하는 최소 경계 사각형을 임시 Cloaking 영역으로 설정한다. 이때, 임시 Cloaking 영역을 이루고 있는 셀 개수(C)와 k′를 구한다. 예를 들어, 사용자가 k=4를 요청한 경우, 그림 5와 같이 먼저 사용자가 위치한 셀을 검색하고, 그것을 중심으로 한 셀씩 영역을 확장한다. 그림 5와 같이 사용자가 위치한 셀을 기반으로 두 번의 확장을 수행하여 임시 Cloaking 영역을 설정하고 C와 k′ 값을 구한다.

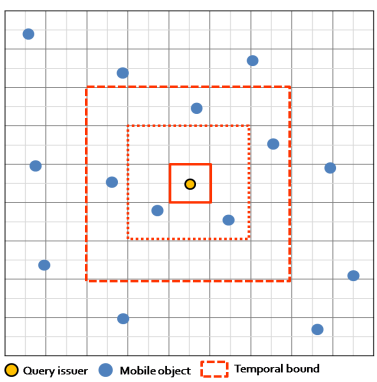
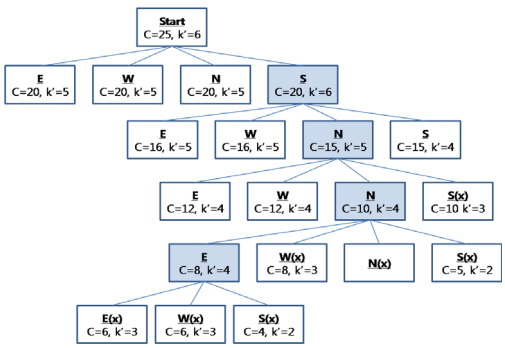
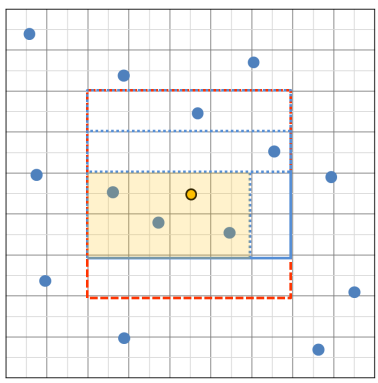


그림 5. 임시 Cloaking 영역 설정

사용자가 요구한 k를 만족하는 최소의 Cloaking 영역을 설정하기 위해, 임시 Cloaking 영역 설정 시 구해진 C와 k′값을 최초의 한계 값으로 설정한다. 그 후, 임시 Cloaking 영역의 각 방향에 존재하는 행 또는 열을 감소시킬 경우의 C와 k′를 구한다. 이는 임시 Cloaking 영역을 줄이기 위해 사용되며, 임시 Cloaking 영역의 셀을 감소시키기 위한 우선순위는 다음과 같다. k를 만족하면서 C가 적을수록 우선순위가 높고, C가 같은 경우 k′가 많을수록 우선순위가 높다. 우선순위가 가장 높은 방향으로 임시 Cloaking 영역을 감소시키고, 이는 사용자가 요구한 k값보다 작아지기 전까지 수행한다. 우선순위에서 사용자 수가 k값보다 크면서, 셀 개수가 적은 것을 선택하는 것은 Cloaking 영역에 대한 질의 처리 후 결과 후보 집합이 불필요하게 많아지는 것을 방지하기 위함이다. 아울러, 셀 개수가 같을 때 사용자의 수가 많은 것이 우선순위가 높은 이유는 Cloaking 영역 내 가능한 많은 사용자를 포함하여 사용자의 프라이버시 보호 수준을 높이기 위함이다. 예를 들어, 그림 5에서 임시 Cloaking 영역의 최초 한계 값은 C=25, k′=6이 된다. 그리고 임시 Cloaking 영역의 각 방향의 한계 값을 계산하면서 임시 Cloaking 영역을 축소시키는 경우를 트리로 나타내면 그림 6(a)와 같다. 트리의 각 노드의 값은 임시 Cloaking 영역에 존재하는 셀의 개수와 사용자의 수를 나타내며, 트리는 너비 우선 탐색으로 확장이 진행되고, 최초에 지정된 한계 값을 계속해서 갱신하면서 수행된다. 한계 값을 이용한 트리 탐색을 통해 최소의 Cloaking 영역을 갖는 Cloaking 영역을 설정할 수 있다. 그림 6(b)는 그림 6(a)의 트리를 이용하여 설정한 최종 결과를 나타낸다.



(a) Cloaking 영역을 축소하기 위한 정보를 담는 트리



(b) 최소 Cloaking 영역을 설정 결과

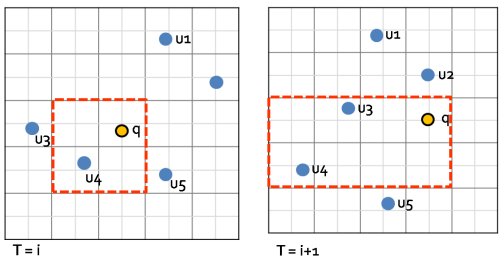
그림 6. 서비스 요청 시 최소의 Cloaking 영역을 설정하기 위한 예제

수행단계 1에서는 사용자의 질의요청 시에 빠르게 Cloaking 영역을 설정하였다면, 수행단계 2에서는 사용자가 요청한 서비스 시간 동안, 사용자의 위치 노출 확률을 최소로 하는 Cloaking 영역을 설정한다. 따라서 이전 Cloaking 영역에 포함된 사용자들에게 가중치를 부여하여, 서비스를 요청한 사용자가 요구하는 프라이버시 보호

수준을 유지하는 Cloaking 영역을 설정하는 방법을 기술한다.

수행단계 2. 연속적인 위치기반 서비스를 지원하기 위한 Cloaking 영역 설정기법(T>0)

사용자가 요청한 서비스 시간(T) 동안, 설정된 Cloaking 영역 안에 존재하는 사용자들은 각자의 이동경로를 따라 움직이게 된다. 따라서 이를 고려한 Cloaking 영역을 설정하기 위해, 설정된 Cloaking 영역 안에 존재하던 사용자를 모두 포함하는 임시 Cloaking 영역을 설정한다. 예를 들어 그림 7(a)는 T=i, k=2일 때 설정된 Cloaking 영역을 나타낸다. 이후, T=i+1일 때 사용자들이 그림 7(b)와 같이 이동할 경우, 사용자(q)와 u4를 포함하는 최소 경계 사각형을 임시 Cloaking 영역으로 설정한다.



(a)T= i일 때의 분포 (b) T= i+1일 때의 분포

그림 7. 사용자의 위치변화를 고려한 임시 Cloaking 영역 설정

임시 Cloaking 영역이 설정되면, 효과적인 Cloaking 영역 설정을 위하여 Cloaking 영역의 프라이버시 보호 수준을 측정하는 엔트로피를 사용한다. 엔트로피의 값을 계산하기 위해 이전 Cloaking 영역에 속했던 사용자가 설정할 Cloaking 영역에 포함될 확률을 구한다. 사용자의 확률을 구하기 위해서는 이동확률 매트릭스를 생성해야 하는데, 엔트로피 값을 구하기 위한 기존의 이동확률 매트릭스는 이전 Cloaking 영역에 속한 사용자의 정보를 반영하지 못하는 문제점을 지니고 있다. 따라서 사용자가 Cloaking 영역에 속했던 횟수를 고려하여 α개의 샘플을 생성하고, 이를 이동확률 매트릭스 생성 시 반영한다. 예를 들어, 그림 7의 T=i일 때 설정된 Cloaking 영역에 포함된 q와 u4가 Cloaking 영역에 속한 횟수가 각각 3회, 2회라고 하고, α=20일 경우 생성되는 이동객체 매트릭스는 다음과 같다. 그림 8의 (a)가 기존의 이동확률 매트릭스(M)라면, (b)는 Cloaking 영역에 속했던 횟수를 고려한 이동확률 매트릭스(M′)이다.

그림 8의 매트릭스에 존재하는 샘플의 수와 식 (3)을 이용하여, 설정할 Cloaking 영역에 존재하는 사용자의 확률을 계산하면 다음과 같다. 이전 Cloaking 영역에 포함되었던 사용자인 q와 u4의 확률 값은 0.5라고 가정한다.

	q	u3	u4
q	8	5	7
u4	6	8	6

(a) 기존 이동확률 매트릭스(M)

	q	u3	u4
q	12	3	5
u4	3	7	10

(b) 가중치를 고려한 이동확률 매트릭스(M′)

그림 8. 사용되는 이동객체 매트릭스

M.q = 0.5*8/20 + 0.5*6/20 = 0.35,
M.u3 = 0.5*5/20 + 0.5*8/20 = 0.325,
M.u4 = 0.5*7/20 + 0.5*6/20 = 0.325

M′.q = 0.5*12/20 + 0.5*3/20 = 0.375,
M′.u3 = 0.5*3/20 + 0.5*7/20 = 0.25,
M′.u4 = 0.5*5/20 + 0.5*10/20 = 0.375

이와 같이, M′ 매트릭스를 사용하여 사용자의 확률 값을 계산하게 되면 Cloaking 영역에 포함되었던 사용자의 확률 값을 증가시킬 수 있다. 이를 통해, Cloaking 영역에 포함되었던 사용자들이 설정할 Cloaking 영역에 포함될 확률을 높임으로써, 서비스를 요청한 사용자의 프라이버시를 보장한다. 임시 Cloaking 영역에 존재하는 사용자의 확률 값을 계산하면 질의를 요청한 사용자가 위치한 그리드 셀을 중심으로 셀을 확장하고, 확장한 셀 안에 포함된 사용자의 수가 요구한 k값과 일치하거나 큰 경우에는 식 (1)을 이용하여 엔트로피를 계산하고, 이를 식 (2)에 대입하여 Cloaking 영역의 프라이버시 보호 수준을 계산한다. 계산된 프라이버시 보호 수준이 사용자가 요구한 k값보다 클 경우, 사용자들이 존재하는 그리드 셀을 포함하는 최소 경계 사각형을 Cloaking 영역으로 설정한다. 예를 들어, 그림 7(b)과 같은 임시 Cloaking 영역과 M′를 이용하여 계산된 사용자의 확률 값을 사용하고 사용자가 요구하는 k=2인 경우, q와 u3이 선택된다. 두 사용자의 확률 값을 가지고 Cloaking 영역의 프라이버시 보호 수준을 계산하면 다음과 같다.

$H(A) = -(0.375*\log 0.375 + 0.25\log 0.25) = 1.03064$
$D(A) = 2^{1.03064} = 2.04293$

계산된 프라이버시 보호 수준이 사용자가 요구한 2보다 크기 때문에 q와 u3을 포함하는 최소 경계 사각형을 Cloaking 영역으로 설정한다. 그림 9는 최종적으로 설정된 Cloaking 영역을 나타낸다.

앞에서의 수행단계를 고려하여 설계한 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 알고리즘은 그림 10과 같다. 먼저, Cloaking 영역을 요청하는 사용자의 위치정보, 프라이버시 보호 수준 k,

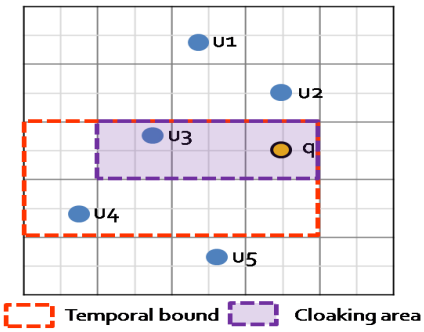


그림 9. 프라이버시 보호 수준을 고려한 Cloaking 영역 설정

서비스 시간 T를 입력받는다. 다음으로 질의를 요청한 사용자가 위치한 셀을 검색하여, 그것을 중심으로 사방으로 한 셀씩 확장하여 영역을 탐색한다. k개 이상의 사용자를 포함하는 셀을 탐색한 뒤, 그들과 질의 요청자가 위치한 셀을 포함하는 최소경계사각형을 임시 Cloaking 영역을 설정한다. 이후 T=0이면, 임시 Cloaking 영역의 셀 개수와 포함된 사용자의 수를 구하고 가지치기를 통해 사용자가 요구한 k값을 만족하면서 최소의 크기를 갖는 Cloaking 영역을 설정한다. T>0 경우, 계산된 프라이버시 보호 수준이 사용자가 요구한 k값보다 클 경우, 이를 결과 Cloaking 영역으로 설정한다. 이후 사용자가 요구한 서비스 시간이 지나면 알고리즘을 종료한다.

```
연속적인 위치기반 서비스를 지원하는 그리드 기반
        Cloaking 영역 설정 알고리즘
//입력: 질의 요청자(q)의 좌표정보, k(프라이버시 보호
        수준), T(요청한 서비스 시간)
//출력: T에 해당하는 k를 만족하는 Cloaking 영역
1. q가 위치한 셀 위치 검색
2. for('t'(현재 서비스 시간) < T)
3. {
4.      if(T == 0)
5.      {
6.      영역 확장을 통한 주변 셀 탐색을 통해 k-1명의
        사용자가 포함된 셀 선택
7.      선택된 셀들과 질의가 발생한 셀을 포함하는 최
        소경계 사각형을 임시 Cloaking 영역으로 설정
8.      임시 Cloaking 영역에 포함된 셀개수(C)와 사
        용자 수(k')를 최초의 한계 값으로 설정
9.      임시 Cloaking 영역의 각 방향을 고려하여 사
        용자가 요구하는 k값을 만족하면서 선택할 수
        있는 최소의 Cloaking 영역을 선정하여, 결과
        Cloaking 영역으로 반환
10.     }
12.     else
13.     {
14.     //T가 0보다 큰 경우의 Cloaking 영역 설정
15.     사용자들의 위치 정보를 갱신함
16.     T=0일 때 설정된 Cloaking 영역에 포함되었던
        모든 사용자들을 검색
17.     검색된 사용자를 포함하는 최소 경계 사각형을
        임시 Cloaking 영역으로 설정
18.     사용자의 count를 고려한 이동객체 매트릭스를
        생성하여 확률 값 계산
19.     q가 위치한 셀 위치를 기반으로 k-1명의 사용
        자가 포함된 셀 선택
20.     do
21.     {
22.     선택한 셀을 포함하는 영역을 임시
        Cloaking 영역으로 설정하고, entropy를 계산
23.     entropy를 Anonymity degree(D)로 변환시킴
24.     if(D>= k)
25.     {
26.     임시 Cloaking 영역을 결과 Cloaking 영역으로
        반환
27.     }
28.     임시 Cloaking 영역을 확장하여 이웃한 사용자
        를 검색
29.     }while(D < K)
30.     }
31. }
32. Cloaking 영역 설정 알고리즘 종료
```

그림 10. 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 알고리즘

4. 성능평가

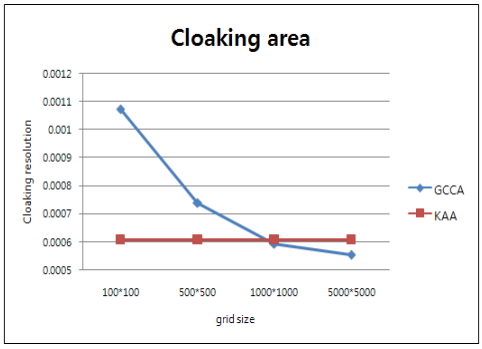
본 연구에서 제안하는 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 알고리즘(Grid based Continuous Cloaking Algorithm(이하 GCCA))은 Microsoft Visual Studio.NET 2003으로 구현하였으며, Intel Core2 Duo CPU E4500 2.20GHz와 Ram 2G, Window XP상에서 성능을 평가하였다. 성능평가에 사용된 이동객체 데이터는 Network-based Generator of Moving Objects[13]를 사용하여 독일 올덴버그의(15×15㎢) 실제 도로 네트워크를 기반으로 생성하였다. 또한, 전체 영역 대비 설정된 Cloaking 영역의 비율을 쉽게 파악하기 위하여 Cloaking 영역의 최대 크기는 1로 설정하였다.

성능평가 대상으로는 연속 질의 처리를 지원하는 유일한 연구인 Toby Xu와 Ying Cai의 연구[5] Advanced KAA(이하 KAA) 기법을 구현하여 비교하였다. <표 1>은 성능평가에 사용된 변수들이다.

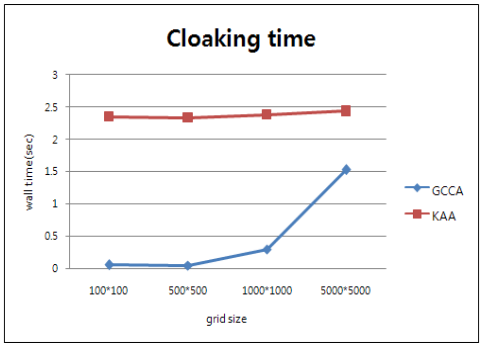
표 1. 실험 환경

parameter	range	default
Node number	2000 ~ 8000	5000
Anonymity level	2 ~ 20	10
Session life time	3 ~ 7	5
Number of service	50	50
Cell size	100*100 ~ 5000*5000	1000*1000

KAA와의 성능 비교에 앞서, 제안하는 GCCA의 성능은 셀 사이즈의 영향을 받는다. 그림 11은 그리드 셀 사이즈에 따른 Cloaking 영역의 크기 및 영역 설정 속도를 비교한 것이다. Cloaking 영역을 설정함에 있어서 영역의 크기와 영역 설정 속도 모두 서비스의 질을 좌우하므로, 영역과 시간 양 측면에서 좋은 성능을 보이는 1000×1000을 기본 그리드 셀 사이즈로 하여 성능평가를 수행하였다.



(a)



(b)

그림 11. 그리드 셀 사이즈에 따른 Cloaking 영역 및 영역 설정 시간 비교

4.1 K-anonymity(이하 k) 변화에 따른 성능평가

그림 12는 k 변화에 따른 Cloaking 영역의 크기를 비교한 것이다. 두 방법 모두 k가 증가할수록, Cloaking 영역은 증가하였다. 특히, k가 커질수록 GCCA는 KAA보다 Cloaking 영역이 증가하였으며, GCCA의 평균 Cloaking 영역은 0.000778로 KAA의 0.000677보다 약 1.15배 커짐을 알 수 있었다. 이는 KAA가 Cloaking 영역으로 K-anonymity를 만족하는 가장 작은 원을 찾는 반면, GCCA는 초기 Cloaking 영역에 포함된 모바일 사용자들을 포함하는 Cloaking 영역을 생성하기 때문이다. 따라서 k가 커질수록 GCCA가 유지하려는 모바일 사용자들이 많아져, Cloaking 영역이 크게 설정되었다.

그림 13은 k 변화에 따른 Cloaking 영역 설정 시간을 보여준다. 두 방법 모두 k가 증가할수록, Cloaking 영역 설정 시간이 증가하였다. 하지만, GCCA는 그리드 기반으로 빠르게 모바일 사용자를 검색하기 때문에, 모바일 사용자 쌍을 통해 후보 원 영역을 생성하는 KAA에 비해 빠른 영역 설정 성능을 보였다. GCCA의 평균 Cloaking 영역 설정 시간은 0.4273으로 KAA의 2.8673에 비해 약 6.7배 좋은 성능을 보였다.

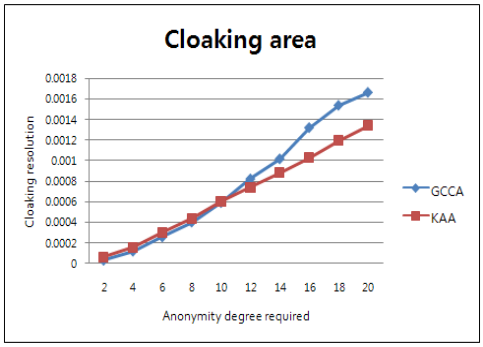


그림 12. k 변화에 따른 Cloaking 영역 크기 비교

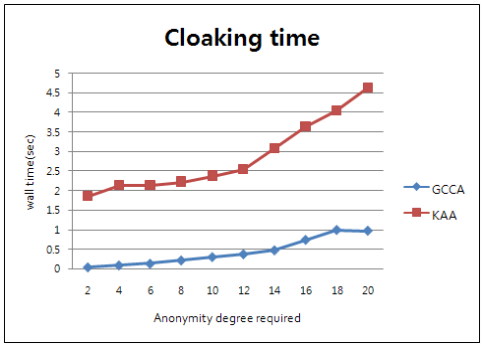
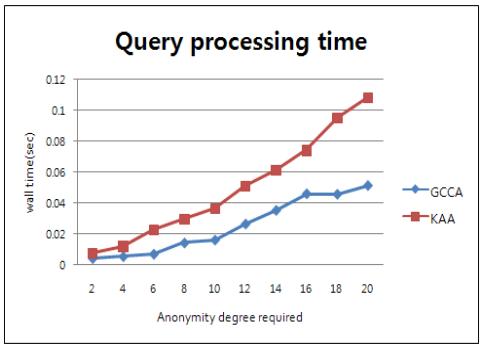


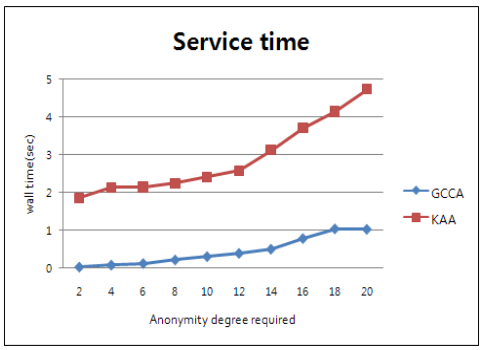
그림 13. k 변화에 따른 Cloaking 영역설정 시간 비교

그림 14는 k 변화에 따른 질의 처리 시간과 총 서비스 시간을 비교한 것이다. 그림 12를 통해 알 수 있듯이, Cloaking 영역은 KAA가 GCCA에 비해 작게 설정되었다. 하지만, KAA의 경우, 원 영역에 외접하는 사각형을 통해 질의 처리가 수행되기 때문에, 오히려 질의 처리 시간에서 GCCA보다 성능이 저하됨을 그림 14(a)를 통해

알 수 있다. 평균적으로는 GCCA가 0.0252, KAA는 0.0500로 GCCA가 약 1.98배 좋은 성능을 보였다. 그림 14(b)는 Cloaking 영역 설정 시간과 질의 처리 시간을 합한 총 서비스 시간을 나타낸다. Cloaking 영역 설정 시간이 질의 처리 시간에 비해 상대적으로 크기 때문에, 총 서비스 시간은 그림 13과 거의 유사하게 나타났다. GCCA의 평균 총 서비스 시간은 0.4526으로 KAA의 2.9173에 비해 약 6.4배 보다 좋은 성능을 보였다.



(a)



(b)

그림 14. k 변화에 따른 질의 처리 시간 및 총 서비스 시간 비교

그림 15는 k 변화에 따른 프라이버시 수준을 측정한 것이다. 여기에서 프라이버시 수준은 초기 Cloaking 영역에 속한 모바일 사용자 중에서, 이후 Cloaking 영역에 포함되어 있는 모바일 사용자의 수를 고려하여 측정한 값이다. 프라이버시 수준은 다음의 <식 4>를 통해 계산하였다.

$$\text{프라이버시 수준} = 1 - (m-1)/(n-1) \quad (4)$$

여기에서 n은 초기 Cloaking 영역에 포함된 모바일 사용자의 수, m은 이후 Cloaking 영역에 포함된 모바일 사용자의 수를 의미한다. 또한, $0 \leq$ 프라이버시 수준 ≤ 1의 범위를 가지며, 프라이버시 수준 = 1은 질의 요청자가 100% 확률로 노출됨을 의미한다. 그림 15에서 볼 수 있듯이, GCCA가 KAA보다 항상 더 낮은 노출 위험을 보였다. 평균적으로 GCCA는 0.132417, KAA는 0.182769로, GCCA가 약 1.4배 좋은 성능을 보였다. 이는 GCCA가 초기 Cloaking 영역에 포함된 모바일 사용자에게 가중치를 둬 Cloaking 영역을 설정했기 때문이다.

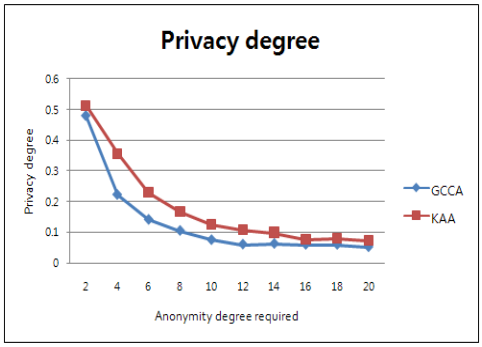


그림 15. k 변화에 따른 프라이버시 수준 비교

4.2 모바일 사용자 수 변화에 따른 성능평가

그림 16은 모바일 사용자 수 변화에 따른 Cloaking 영역의 크기를 비교한 것이다. 두 방법 모두 모바일 사용자 수가 증가할수록, Cloaking 영역은 감소하였다. 이는 모바일 사용자 수가 증가할수록 단위 면적 당 밀집도가 증가하여, 보다 좁은 면적에서 K-anonymity를 만족할 수 있는 충분한 모바일 사용자를 찾을 수 있기 때문이다.

특히, 모바일 사용자의 수가 5000 이상일 때는 GCCA가 KAA와 영역이 유사하거나 더 작아짐을 알 수 있었다. 이는 모바일 사용자가 조밀하게 모여 있을수록, 원이 영역 측면에서 갖는 장점이 사라지기 때문이다. 달리 말하면, KAA는 모바일 사용자들이 원주에 위치하는 원을 생성하기 때문에, 모바일 사용자가 밀집해 있을수록 원이 생성되지 않는 경우가 많아, 보다 큰 영역이 선택될 가능성이 높아지기 때문이다. 또한, 성능평가에 사용된 데이터가 실제 도시를 기반으로 한 데이터임을 고려하면, 모바일 사용자 수가 많은 경우가 실세계에 더 유사하다고 볼 수 있다. 따라서 제안하는 GCCA가 실용적임을 유추할 수 있었다. 평균적으로는 GCCA가 0.0008, KAA가 0.0007로, KAA가 약 1.14배 좋은 성능을 보였다.

그림 17은 모바일 사용자 수 변화에 따른 Cloaking 영역 설정 시간을 보여준다. GCCA는 모바일 사용자의 수가 증가할수록 Cloaking 영역의 생성 시간이 짧아짐을 확인할 수 있었다. 이는 모바일 사용자의 밀집도가 높을수록, 보다 적은 셀을 탐색하여 K-anonymity를 만족시키는 사용자를 탐색할 수 있기 때문이다.

반면에 KAA의 경우는 모바일 사용자의 수가 증가함에 따라, Cloaking 영역 생성 시간도 증가하였다. KAA는 최소의 Cloaking 영역을 생성하기 위해, 모바일 사용자들로 생성 가능한 모든 원을 고려하는데, 모바일 사용

자의 밀집도가 높아질수록 생성 가능한 원의 개수 또한 증가하기 때문이다. GCCA의 평균 Cloaking 영역 설정 시간은 0.2781로 KAA의 2.3456에 비해 약 8.43배 보다 성능을 보였다.

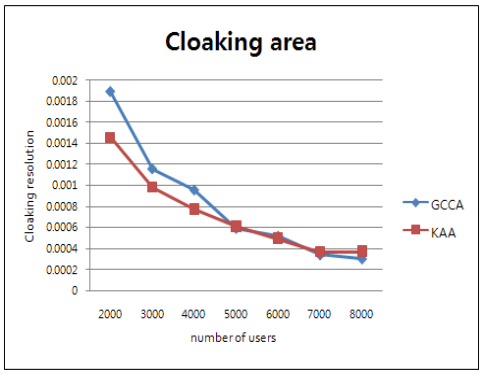


그림 16. 모바일 사용자 수 변화에 따른 Cloaking 영역 크기 비교

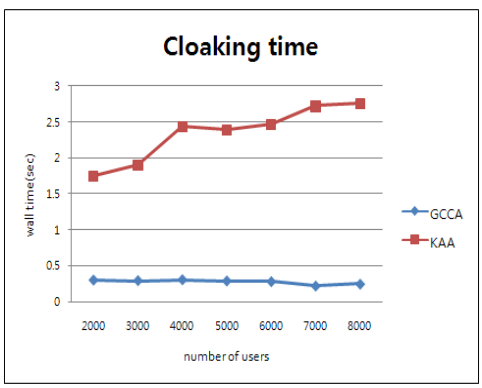


그림 17. 모바일 사용자 수 변화에 따른 Cloaking 영역 설정 시간 비교

그림 18은 모바일 사용자 수 변화에 따른 총 서비스 시간을 보여준다. 질의 처리 시간에 관한 성능은 4.1절에서 언급한 것과 같은 경향을 보이므로 생략한다. GCCA의 평균 총 서비스 시간은 0.3053으로 KAA의 2.3980에 비해 약 7.9배 좋은 성능을 보였다.

그림 19는 모바일 사용자 수 변화에 따른 프라이버시 수준을 측정한 것이다. 두 기법 모두 모바일 사용자의 수가 증가할수록, 사용자의 노출 확률이 증가하였다. 이는 모바일 사용자가 밀집해 있을수록 Cloaking 영역이 작게 생성되기 때문에, 새로운 사용자가 Cloaking 영역에 새로 포함되거나, Cloaking 영역에 포함되었던 사용자가 나가게 될 확률이 높아지기 때문이다. 평균적으로는 GCCA가 0.0767, KAA가 0.1254로, GCCA가 약 1.6배 더 낮은 노출 위험을 갖는 것을 알 수 있었다.

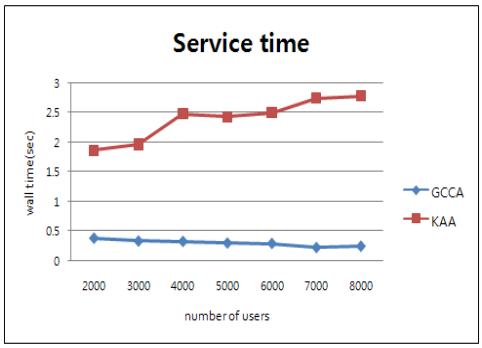


그림 18. 모바일 사용자 수 변화에 따른 총 서비스 시간 비교

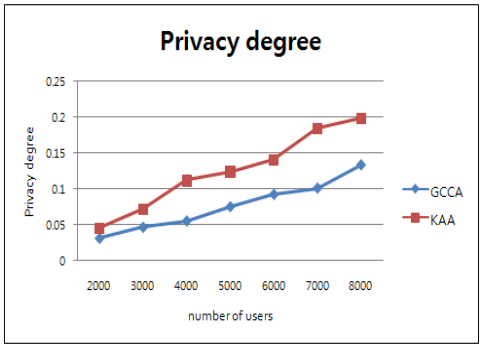


그림 19. 모바일 사용자 수 변화에 따른 프라이버시 보호 수준 비교

4.3 Session life time 수 변화에 따른 성능평가

그림 20은 session life time 수에 따른 Cloaking 영역의 크기를 비교한 것이다. 두 기법 모두 움직이는 모바일 사용자를 대상으로 하기 때문에, 서비스 횟수가 증가할수록 Cloaking 영역이 커지는 양상을 보였다. 앞서 말했듯이, KAA는 최소의 Cloaking 영역을 찾기 때문에, KAA의 영역이 조금 작은 것을 볼 수 있었다. 평균적으로는, GCCA가 0.00059 KAA가 0.00057로 GCCA가 약 1.03배 더 큰 Cloaking 영역을 설정하였다. 그림 21은 session life time 수에 따른 Cloaking 영역 설정 시간을 보여준다. 두 기법 모두 서비스 횟수가 증가할수록, Cloaking 영역이 넓어지고, 영역에 포함되는 모바일 사용자의 수가 증가하기 때문에, Cloaking 영역 설정 시간이 증가였다. 하지만, GCCA가 보다 완만한 상승 곡선을 보여, Cloaking 영역 생성 시간 측면에서 KAA에 비해 우수함을 알 수 있었다. GCCA의 평균 Cloaking 영역 설정 시간은 0.3288로 KAA의 2.4863에 비해 약 7.5배 좋은 성능을 보였다.

그림 22는 session life time 수에 따른 총 서비스 시간을 비교한 것이다. GCCA의 총 서비스 시간은 0.3465로 KAA의 2.5254에 비해 약 7.3배 좋은 성능을 보였다.

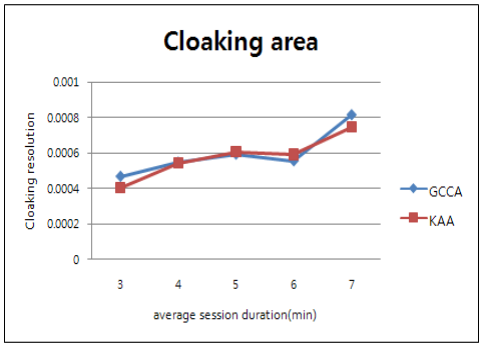


그림 20. session life time 변화에 따른 Cloaking 영역 크기 비교

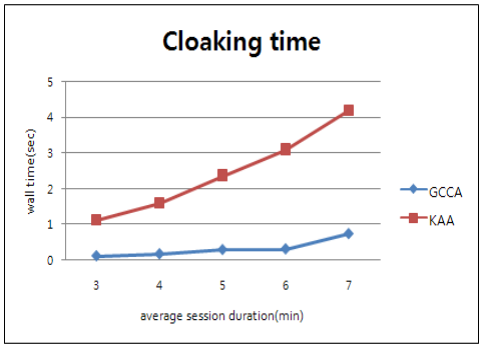


그림 21. session life time 변화에 따른 Cloaking 영역 설정 시간 비교

그림 23은 session life time 수에 따른 프라이버시 보호 수준을 측정한 것이다. 두 기법 모두 서비스 횟수가 증가할수록, 사용자의 노출 확률이 증가하였다. 하지만, 제안하는 GCCA가 KAA보다 항상 더 낮은 노출 위험을 갖는 것을 볼 수 있었다. 평균적으로 GCCA는 0.087, KAA는 0.135로, GCCA가 약 1.5배 좋은 성능을 보였다.

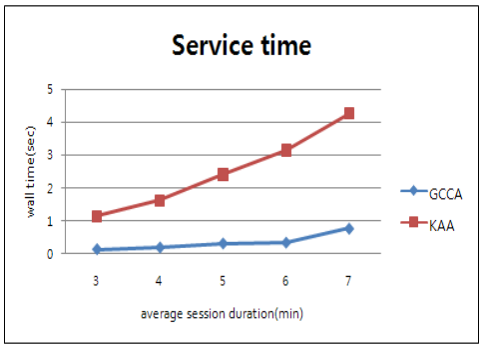


그림 22. session life time 변화에 따른 총 서비스 시간 비교

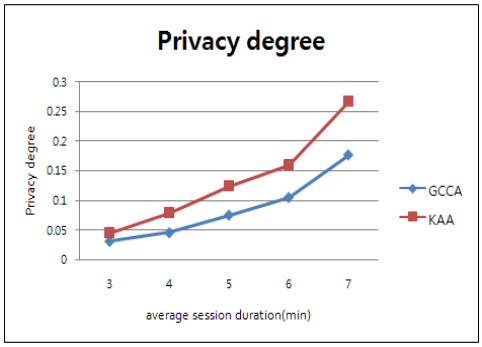


그림 23. session life time 변화에 따른 프라이버시 보호 수준 비교

5. 결론 및 향후연구

본 논문에서는 위치기반 서비스에서 연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 기법을 제안한다. 제안하는 기법은 연속적인 위치기반 서비스를 효율적으로 지원하기 위하여 그리드 기반의 셀 확장을 통해 빠르게 Cloaking 영역을 설정하였다. 아울러, 모바일 사용자의 위치 노출 확률을 최소로 하는 Cloaking 영역 설정을 위하여 Cloaking 영역에 포함되었던 사용자들에게 가중치를 부여하여 이동확률 매트릭스를 생성하였고, 이를 바탕으로 엔트로피를 계산하였다. 마지막으로 기존 연구인 Advanced KAA와의 성능비교를 통하여 제안하는 기법이 우수함을 보였다. 제안하는 기법은 Advanced KAA가 모든 후보집합을 고려하여 Cloaking 영역을 생성하기 때문에 Cloaking 영역이 약 1.1배 크게 설정되었지만, 그리드 기반의 셀 확장을 통해 기존 연구에 비해 약 6.7배 Cloaking 영역을 빠르게 설정하였다. 또한, 전체 서비스를 수행하는 시간 역시 기존 연구에 비해 약 6.4배 빠름을 보였다. 아울러, 제안하는 기법은 질의 요청자가 노출될 확률을 기존 연구에 비해 1.5배 줄여 보다 높은 프라이버시 보호 수준을 보장함을 보였다.

향후 연구는 분산 환경에서 연속질의처리를 수행하는 그리드 기반 Cloaking 기법을 연구하는 것이다.

참 고 문 헌

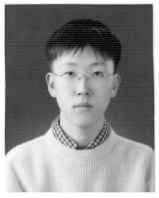
[1] 이준석, 김서균, “위치기반서비스(LBS)의 기술 동향 및 국내외 산업 동향 분석,” 정보통신연구진흥원 계간 제 5권 제 2호 (통권 16호), 2003.

[2] 이낙훈, 박주훈, 안병익, “위치기반 응용 서비스(항법, 디렉토리, 위치추적)를 지원하는 LBS 표준 참조 시스템,” 한국공간정보시스템학회 학술대회 논문집, pp 33-38, 2004.

[3] Voelcker, J, “Stalked by Satellite: An Alarming Rise in GPS-enabled Harassment”, IEEE Spectrum, Vol.47 NO.7, 2006, pp.15-16

[4] J. Warrior, E. McHenry, and K. McGee, “They Know Where You Are”, IEEE Spectrum, Vol.40 No.7, 2003, pp. 20-25.
[5] Toby Xu and Ying Cai, “Location Anonymity in Continuous Location-based Services”, ACMGIS, 2007, pp. 221-238.
[6] Gedik, B., Liu, L, “Location Privacy in Mobile Systems: A Personalized Anonymization Model”, ICDCS, 2005, pp. 620-629.
[7] Gruteser, M., Liu, X. “Protecting Privacy in Continuous Location-Tracking Applications”, IEEE Security and Privacy Vol.2 No.2, 2004, pp. 28 - 34.
[8] Mokbel, M.F., Chow, C.Y., Aref, W.G., “The New Casper: Query Processing for Location Services without Compromising Privacy”, VLDB, 2006, pp.763-774.
[9] Z. Xiao, X. Meng and J. Xu, "Quality Aware Privacy Protection for Location-based Services", Database Systems for Advanced Applications, vol.4443, 2007, pp. 434-446.
[10] Cheng, R., Zhang, Y., Bertino, E., Prabhakar, S. “Preserving User Location Privacy in Mobile Data Management Infrastructures”, Privacy Enhancing Technology Workshop, Vol.4258, 2006, pp. 393-412.
[11] 김지희, 이아름, 김용기, 엄정호, 장재우, “위치기반 서비스에서 개인 정보 보호를 위한 K-anonymity 및 L-diversity를 지원하는 Cloaking 기법”, 한국공간정보시스템학회, 제10권 제4호, 2008, pp.1~10.
[12] Gruteser, M., Grunwald, D. “Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking”, MobiSys, 2003, pp.31-42.
[13] T. Brinkhoff, “A Framework for Generating Network-Based Moving Objects”, GeoInformatica, Vol.6 No.2, 2002, pp.153-180

이 아 름
2008년 전북대학교 컴퓨터공학과(공학사)
2008년~현재 전북대학교 컴퓨터공학과 석사과정
관심분야 : 공간 데이터베이스, 질의처리, 위치 보안을 위한 cloaking



김 형 일
2009년 전북대학교 컴퓨터공학과(공학사)
2009년~현재 전북대학교 컴퓨터공학과 석사과정
관심분야 : 공간 데이터베이스, 위치 보안을 위한 cloaking

장 재 우
1984년 서울대학교 전자계산기공학과 (공학사)
1986년 한국과학기술원 전산학과 (공학석사)
1991년 한국과학기술원 전산학과 (공학박사)
1996년~1997년 Univ. of Minnesota, Visiting Scholar
2003년~2004년 Penn State Univ., Visiting Scholar.
1991년~현재 전북대학교 컴퓨터공학과 교수
관심분야 : 공간 네트워크 데이터베이스, 하부저장구조, 센서네크워크