

위치 기반 서비스에서 연속적인 질의 처리를 지원하는 분산 그리드 기반 Cloaking 영역 설정 기법[†]

김형일[○] 이아름 장재우^{*}

전북대학교 컴퓨터공학과, 영상정보신기술연구소^{*}
{hikim, arlee, jwchang}@dmlab.chonbuk.ac.kr

Distributed Grid-based Cloaking Area Creation Scheme supporting Continuous Query in Location-based Services

HyeongIl Kim[○] AhReum Lee JaeWoo Chang^{*}

Department of Computer Engineering, Chonbuk National University,
Center for Advanced Image and Information Technology^{*}

1. 연속적인 질의 처리를 지원하는 분산 그리드 기반 Cloaking 기법

최근 모바일 기기 및 무선 통신 기술의 발달로 인하여 위치 기반 서비스(Location-Based Service : LBS)의 이용이 확산되고 있다. LBS에서 사용자의 위치 정보를 보호하기 위한 기존의 연구는 K-anonymity를 만족하는 Cloaking 영역을 생성하는 기법이 대표적이다. 그러나 이와 같은 기법은 사용자가 이동하면서 연속적으로 서비스를 요청하는 경우, 설정되는 모든 Cloaking 영역에 질의 요청자는 존재하지만, k-1 명의 사용자는 포함되지 않을 수 있다는 문제가 존재한다. 이러한 문제점을 해결하기 위해, Toby Xu et al의 연구[1]와 A. Lee et al.의 연구[2]가 진행되었다. Toby Xu et al.의 연구에서는 Advanced K-Anonymity Area(이하 KAA)를 제안하였다. KAA는 연속적인 위치기반 질의를 지원할 수 있도록 엔트로피(Entropy)를 사용하여 설정되는 Cloaking 영역의 anonymity degree(이하 anonymity 정도)를 계산함으로써, 질의 요청자의 프라이버시를 보호할 수 있는 Cloaking 영역을 생성하는 기법이다. 하지만 KAA는 Cloaking 영역 설정에 많은 시간이 소요되고, 이전 Cloaking 영역에 속했던 사용자들에 대한 정보가 반영되지 못한다는 문제점을 갖는다. 한편, A. Lee et al.의 연구에서는 Grid-based Continuous Cloaking(이하 GCC)을 제안하였다. GCC는 KAA의 문제점을 고려하여 그리드 기반의 셀 확장을 통해 빠르게 Cloaking 영역을 설정하고, 영역에 포함되었던 사용자에게 가중치를 부여하여 이전 시간의 정보를 반영할 수 있는 기법이다. 그러나 이러한 연구는 모바일 사용자와 LBS 서버 중간에 존재하는 anonymizer를 통해 Cloaking 영역을 설정하는 중앙 집중 방식이다. 이와 같은 중앙 집중 방식은 anonymizer에서의 병목(bottle-neck) 현상으로 인한 성능 저하 문제 및 사용자의 위치정보를 관리하고 있는 anonymizer 자체의 보안 위협 문제를 지닌다. 이러한 문제를 해결하기 위해서는 분산 환경에서 연속적인 질의 처리를 제공할 수 있는 기법이 필요하다.

따라서 본 논문에서는 기존의 GCC를 확장하여, 분산 환경에서 연속적인 질의 처리를 지원하기 위한 Cloaking 영역 설정 기법인 DGCC(Distributed GCC)를 제안한다. 제안하는 기법은 Cloaking 영역 설정 시 필요한 정보를 분산 유지하고, Cloaking 영역 설정을 위한 연산을 분산적으로 수행한다. 아울러, 정보 병합 시, 각 클러스터 내 병합 노드를 통해 통신을 계층적으로 수행함으로써, 질의 요청자에게 집중되는 통신을 분산시킨다. 또한, 사용자 간 통신 시 상대방이 감청하는 것에 대비하여, 사용자의 위치를 힐버트 커브를 이용하여 힐버트 값으로 변환하고, 이를 인덱싱 키로 고려하여 가상 링 구조를 형성하는 클러스터 구성방법[3]을 사용한다. 제안하는 Cloaking 영역 설정 기법은 서비스 요청 시(T_0)와 서비스 수행시간 동안(T_i) Cloaking 영역을 설정하는 부분으로 나뉜다. 먼저, 서비스 요청 시 Cloaking 영역을 설정하는 방법은 수행단계 1과 같다.

수행단계 1. 서비스 요청 시, Cloaking 영역 설정

서비스 요청 시, 질의 요청자는 자신을 기준으로 정사각형 형태로 셀 확장을 하여, k-1명의 인접한 사용자를 포함하는 임시 Cloaking 영역을 설정한다. 이 후, 임시 Cloaking 영역 각 방향에서의 축소를 통해 k값을 만족하는 Cloaking 영역을 설정한다. 다음으로, 서비스 수행 시간동안에는 수행단계 2~5를 통해 분산적으로 Cloaking 영역을 설정한다.

수행단계 2. 임시 Cloaking 영역 설정

T_i 시간에 Cloaking 영역 안에 존재했던 사용자(이하 참여자)는 각자의 이동 경로를 따라 움직인 후, T_{i+1} 시간의 위치정보를 질의 요청자에게 전송한다. 질의 요청자는 전송받은 위치정보를 바탕으로 T_i 시간의 Cloaking 영역에 포함되었던 참여자들을 모두 포함하는 최소 경계 사각형을 임시 Cloaking 영역으로 설정한다. 질의 요청자는 T_{i+1} 시간에 임시 Cloaking 영역에 포함된 사용자를 모두 찾기 위하여, 클러스터의 대표 노드에 접근하여 임시 Cloaking 영역에 속한 셀에 사용자가 존재하는지를 검색하고, 이를 기반으로 임시 Cloaking 영역에 속한 사용자 및 사용자 위치에 관한 정보를 저장한다.

수행단계 3. 각 클러스터 내 병합노드의 선정

사용자들의 확률 값을 분산적으로 계산하기 위하여, 이전 시간에 Cloaking 영역에 속한 참여자들에게 임시 Cloaking 영역에 포함된 사용자들의 정보와 병합노드, 병합할 사용자 수에 대한 정보 및 이동확률 매트릭스(이하 M)를 구성할 샘플의 수 α 값을 전송한다. 여기서 병합노드는 해당 클러스터에 속한 참여자의 부분 확률 값을 합산하는 노드를 말하며, 병합 방식은 다음과 같다. 먼저, 참여자들을 각 클러스터에 따라 분류하고 무작위로 병합노드를 선정한다. 질의 요청자는 병합노드에게는 병합노드임을 알리고, 병합할 사용자의 수를 전송한다. 나머지 참여자들에게는 자신을 담당하는 병합노드를 알려준다. 이러한 과정을 통해 분류된 참여자들은 선정된 병합노드에게 자신의 정보를 전송하고, 병합노드들은 병합할 사용자 수만큼 통신을 기다린 후, 정보

[†] 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. 2010-0000202)

를 합산하여 질의 요청자에게 전송하는 계층적인 통신을 수행한다.

수행단계 4. 임시 Cloaking 영역 내 사용자의 확률 값 계산

설정되는 Cloaking 영역의 anonymity 정도를 계산하기 위해, T_i 시간의 참여자들이 T_{i+1} 시간에 설정된 임시 Cloaking 영역 안에서 어떤 사용자가 될 확률을 계산한다. 이를 위해 각 참여자는 α 개의 샘플로 이루어진 부분 이동확률 매트릭스(이하 M')를 생성한다. 이는 M 생성 및 사용자의 확률 계산을 분산적으로 수행하기 위함이다. 여기서 각 참여자가 생성한 M' 는 해당 참여자가 T_{i+1} 시간의 어떤 사용자의 위치로 이동하는 움직임을 반영한다. M' 가 생성되면 각 참여자는 유지하고 있던 확률 값을 식 (1)에 대입하여, T_{i+1} 시간의 각 사용자의 부분 확률 값을 계산한다. $UserProb_i$ 는 T_i 시간에 해당 참여자의 확률 값을 의미한다. 참여자들은 계산된 부분 확률 값을 병합노드로 전송하고, 각 클러스터 내의 병합노드는 부분 확률 값을 합산하여 질의 요청자에게 전송한다. 질의 요청자는 병합노드로부터 전송받은 부분 확률 값을 합산하여 임시 Cloaking 영역에 포함된 사용자의 최종 확률 값을 구한다.

$$Prob_{i+1} = UserProb_i * M'_{i+1} / \alpha \dots\dots\dots (1)$$

수행단계 5. Anonymity 정도를 고려한 Cloaking 영역 설정

임시 Cloaking 영역에 속한 사용자들의 확률 값이 계산되면, 질의 요청자는 영역을 확장하면서 사용자의 확률 값을 식 (2)에 대입하여 엔트로피($H(A)$)를 계산하고, 이를 식 (3)에 대입하여 anonymity 정도($D(A)$)를 계산한다. 이 과정에서 $k' \geq k$ 이면, 해당 사용자들을 포함하는 최소 경계 사각형을 Cloaking 영역으로 설정한다. 이후, 각 사용자에게 계산된 확률 값을 전송하여, 참여자들의 확률 값과 count 정보를 갱신하게 한다. 이는 이후 Cloaking 영역 설정 시 이용된다.

$$H(A) = - \sum_{i=1}^m p_i \log p_i \dots\dots\dots (2)$$

$$D(A) = 2^{H(A)} \dots\dots\dots (3)$$

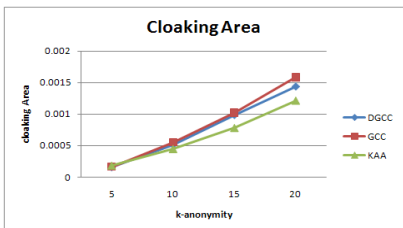
2. 성능 평가

제안하는 DGCC 기법은 Microsoft Visual Studio 6.0으로 구현하였으며, Intel Core2 Quad CPU Q6600 2.40GHz와 Ram 2GB, Windows XP professional 상에서 성능 평가를 수행하였다. 이동 객체 데이터는 Brinkhoff 알고리즘[4]을 사용하여 독일 울텐버그의(15×15km²) 도로 네트워크를 기반으로 5,000개를 생성하였다. 아울러, Cloaking 영역의 최대 크기는 1, 서비스 수행 시간은 5, 그리드 셀 사이즈는 512×512로 설정하여 성능평가를 수행하였다. 성능 평가는 k-anonymity 값을 5, 10, 15, 20으로 증가시키면서 Cloaking 영역 크기, Cloaking 영역 설정 시간, Cloaking 영역 내 사용자의 수를 비교하였다.

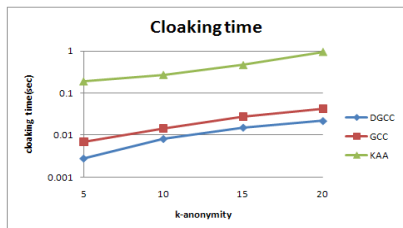
<그림 1>은 k 변화에 따른 Cloaking 영역의 크기를 비교한 것이다. 모든 기법이 k 값이 증가함에 따라 Cloaking 영역을 크게 설정한다. k가 10인 경우 KAA는 0.00045, GCC는 0.00055, DGCC는 0.00051의 Cloaking 영역을 설정한다. DGCC가 GCC에 비해 작은 영역을 설정하는 이유는, 사용자간의 암호화된 메시지를 주고받기 위해 좌표 정보를 힐버트 값으로 변환하면서, 사용자의 위치정보가 한 점으로 사상(mapping)되는 현상이 발생하기 때문이다. 이로 인해, DGCC의 경우 셀 확장 시 보다 많은 사용자를 검색할 수 있게 됨에 따라 GCC에 비해 작은 영역을 설정한다.

<그림 2>는 k 변화에 따른 Cloaking 영역 설정 시간을 비교한 것이다. 모두 k값이 증가함에 따라 Cloaking 영역 설정 시간이 증가하였다. k가 10인 경우, KAA는 0.2718, GCC는 0.01451, DGCC는 0.00814초가 소요된다. GCC와 DGCC의 경우 그리드 기반의 셀 탐색을 통해 빠르게 Cloaking 영역을 설정하기 때문에, 임시 Cloaking 영역 내의 모든 후보 집합을 고려하는 KAA에 비해 Cloaking 영역 설정 시간이 크게 증가하지 않음을 알 수 있다. 아울러, DGCC가 GCC보다 좋은 성능을 보이는 이유는, Cloaking 영역 설정에 참여하는 사용자와 영역 설정 작업을 분산적으로 처리하기 때문이다.

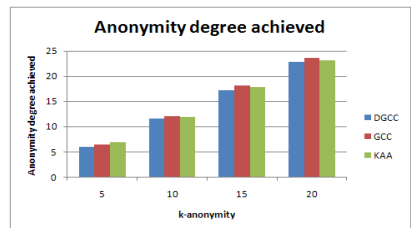
<그림 3>은 k 변화에 따른 Cloaking 영역 내 사용자의 수를 측정한 것이다. 모든 기법이 사용자가 요구한 k 값과 거의 유사한 수의 사용자를 포함하는 Cloaking 영역을 설정하는 것을 볼 수 있다. 이를 통해, 제안하는 DGCC가 사용자가 요구한 K-anonymity를 보장하면서 효율적으로 Cloaking 영역을 설정함을 알 수 있다.



<그림 1> Cloaking 영역 크기



<그림 2> Cloaking 영역 설정 시간



<그림 3> Cloaking 영역 내 사용자 수

참고 문헌

[1] Toby Xu and Ying Cai, "Location Anonymity in Continuous Location-based Services", ACMGIS, pp. 221-238, 2007.
 [2] 이아름, 김형일, 장재우, "연속적인 위치기반 서비스를 지원하는 그리드 기반 Cloaking 영역 설정 기법", 한국공간정보시스템 학회, 제11권, 3호, 2009.10.
 [3] C. Y. Chow, M. F. Mokbel, and X. Liu. A, "Peer-to-Peer Spatial Cloaking Algorithm for Anonymous Location-based Services," In Proc. of the ACM-GIS, pp. 171-178, 2006.
 [4] T. Brinkhoff, "A Framework for Generating Network-Based Moving Objects, GeoInformatica, Vol.6 No.2, pp.153-180, 2002.