

# A Study on Mobility-Aware Edge Caching and User Association Algorithm

TaeYoon Lee<sup>†</sup> · SuKyoung Lee<sup>\*\*</sup>

## ABSTRACT

Mobile Edge Computing(MEC) is considered as a promising technology to effectively support the explosively increasing traffic demands. It can provide low-latency services and reduce network traffic by caching contents at the edge of networks such as Base Station(BS). Although users may associate with the nearest BSs, it is more beneficial to associate users to the BS where the requested content is cached to reduce content download latency. Therefore, in this paper, we propose a mobility-aware joint caching and user association algorithm to improve the cache hit ratio. In particular, the proposed algorithm performs caching and user association based on sojourn time and content preferences. Simulation results show that the proposed scheme improves the performance in terms of cache hit ratio and latency as compared with existing schemes.

Keywords : Edge Caching, User Association, User Mobility, Latency

## 이동성 기반의 엣지 캐싱 및 사용자 연결 알고리즘 연구

이 태 윤<sup>†</sup> · 이 수 경<sup>\*\*</sup>

### 요 약

최근 스마트 디바이스 및 스트리밍 서비스의 수요 증가에 따른 네트워크 트래픽을 효과적으로 관리하기 위한 방법으로 Mobile Edge Computing (MEC)기술이 주목받고 있다. MEC는 Base Station(BS)과 같은 네트워크 엣지에 캐시를 설치함으로써 사용자에게 보다 가까운 곳에서 서비스를 제공하므로 낮은 지연시간을 제공하고, 네트워크 부하를 감소시킬 수 있다. 또한, 엣지 네트워크에서 사용자는 가장 가까운 BS와 연결되는 것보다 요청된 콘텐츠가 캐싱되어 있는 BS와 연결하는 것이 서비스 지연시간 감소에 유리하다. 따라서 본 논문에서는 캐시 적중률 향상을 위한 이동성 기반 캐싱 및 사용자 연결(user association)알고리즘을 제안한다. 제안 알고리즘은 체류시간과 콘텐츠 요청 유사도를 토대로 사용자 연결을 결정하고 콘텐츠를 캐싱한다. 시뮬레이션을 통해 기존 연구 대비 제안 알고리즘의 향상된 캐시 적중률과 감소된 지연시간을 확인한다.

키워드 : 엣지 캐싱, 사용자 연결, 사용자 이동성, 지연시간

### 1. 서 론

최근 스마트 디바이스 및 스트리밍 서비스의 수요 증가에 따른 네트워크 트래픽을 효과적으로 관리하기 위한 방법으로 Mobile Edge Computing(MEC)기술이 주목받고 있다. MEC는 Base Station(BS)과 같은 네트워크 엣지에 캐시를 설치함으로써 사용자에게 보다 가까운 곳에서 서비스를 제공하므로 낮은 지연시간을 제공하고, 네트워크 부하를 감소시킬 수 있다. 이러한 환경에서 모바일 사용자는 시간에 따라 위치가 변경되므로 캐싱되어 있지 않은 서버로부터의 콘텐츠 제공을 유발하고, 이는 응답시간 지연으로 이어진다. 따라서, 이동성(mobility)으로 인한 캐시 성능 저하를 줄이기 위해 체류시간을 활용한 엣지 캐싱 연구가 진행되었다[1-4]. MoPC [1]은 체

류시간을 지수분포로 가정하였으나, [2-4]은 이동성을 Markov Renewal Process로 모델링하여 체류시간을 예측하였다. 예측한 체류시간을 토대로 콘텐츠를 캐싱하여 캐시 적중률 향상을 보였다.

한편, 엣지 네트워크에서 사용자는 가장 가까운 BS와 연결되는 것보다 요청된 콘텐츠가 캐싱되어 있는 BS와 연결되는 것이 서비스 지연시간 감소에 유리하다. 따라서 캐시 성능 향상을 위해 사용자 연결(user association) 문제와 콘텐츠 캐싱 문제를 함께 다룬 연구가 진행되었다[6-10]. [6]은 콘텐츠 요청 선호도에 따른 유사도(similarity)를 계산하고, 유사도에 따라 캐싱과 사용자 연결을 결정하였다. [7]은 기존의 추천 시스템을 활용한 캐싱 연구[5]에 사용자 연결을 함께 고려하여 캐시 적중률을 향상시켰다. [8]은 무선 네트워크에서 페이딩(fading)과 콘텐츠 요청 선호도를 고려하여 캐싱과 사용자 연결 문제를 Lyapunov 최적화 기법을 활용하여 해결하였다. [9]는 예측한 인기도를 기반으로 캐싱하고 지연시간 감소를 목표로 사용자 연결을 결정하였다. [10]은 에너지 효율을 고려하여, 엣지 네트워크에서 콘텐츠 전송에 소모되는 전력을 최소화하기 위한 사용자 연결 및 캐싱 알고리즘을 제안

※ 이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구결과임(No. 2022R1A2B5B01001683).

† 비 회 원 : 연세대학교 컴퓨터과학과 석사과정

\*\* 정 회 원 : 연세대학교 컴퓨터과학과 교수

Manuscript Received : October 7, 2022

Accepted : November 3, 2022

\* Corresponding Author : SuKyoung Lee(sklee@yonsei.ac.kr)

하였다. 하지만 기존 연구[6-10]에서는 이동성이 고려되지 않아 연결된 BS를 통해 요청 콘텐츠를 제공받을 수 없는 경우가 발생하여 서비스 지연시간이 증가한다.

그러므로 본 논문에서는 콘텐츠 캐싱과 사용자 연결문제 (Joint content Caching and user Association:JCA)를 공식화하고, JCA문제는 NP-hard이므로 (i)사용자 연결과 (ii)캐싱의 하위 문제(sub-problem)로 나누어 해결한다. 제안 알고리즘은 요청 선호도에 따른 유사도와 체류시간을 토대로 사용자 연결을 결정하고 콘텐츠를 캐싱한다. 시뮬레이션을 통해 기존 연구 대비 제안 알고리즘의 향상된 캐시 적중률과 감소한 지연시간을 확인한다.

## 2. 시스템 모델 및 문제 공식화

### 2.1 시스템 모델

시스템 모델은 [11]과 같이 엣지 컨트롤러(Edge Controller: EC)와  $K$ 개의 BS가 밀집하게 배치된 Small Cell Networks (SCN) 환경을 고려한다. EC는 각 BS와 연결되어 있으며, 사용자는 하나의 BS와 연결된다. 본 논문에서 사용자 이동은 Markov Renewal Process (MRP)[2-4]로 모델링하고, 사용자( $u \in U$ )는 연결된 BS에 콘텐츠( $f \in F$ )를 요청한다. 그리고 사용자가  $n$ 번째로 이동한 BS( $b_n$ )와 이동 발생 시점( $t_n$ )은 BS를 통해 EC에 전달된다.

**EC(Edge Controller):** 사용자의 이동성을 모니터링하고, 사용자가 이동한 BS집합  $B = \{b_1, \dots, b_n, \dots\}$ 과 이동 발생 시점을 나타내는 집합  $T = \{t_1, \dots, t_n, \dots\}$ 을 사용자 이동성 테이블 (User Mobility Table:UMT)에 저장한다. UMT에 저장된 정보를 토대로 EC가 관리하는 지역에 사용자  $u$ 가 머무는 체류시간( $r_u$ )을 예측한다. EC에서 예측한 체류시간은 사용자 연결에 활용된다.

**BS(Base Station):** BS는 연결된 사용자  $u$ 가 요청한 콘텐츠의 인기도 리스트  $q_u = [q_{u,1}, \dots, q_{u,f}, \dots]$ 를 사용자 요청 테이블(User Request Table:URT)에 저장한다. URT에 저장되는  $q_{u,f}$ 는  $q_{u,f} \in [0,1]$ ,  $\sum_{f \in F} q_{u,f} = 1$ 로 정의한다. 또한 URT에는 BS에 요청된 콘텐츠의 인기도  $q_i = [q_{i,1}, \dots, q_{i,f}, \dots]$ 가 포함된다. 또한  $q_{i,f}$ 는  $q_{i,f} \in [0,1]$ ,  $\sum_{f \in F} q_{i,f} = 1$ 로 정의한다. BS는 URT에 저장된 정보를 토대로 사이즈가  $C$ 인 캐시에 저장할 콘텐츠를 결정한다.

**인기도(Popularity):** 사용자가 요청한 콘텐츠의 인기도는 Zipf 분포를 따른다고 가정한다[2, 3, 6].

$$Zipf(f) = \frac{f^{-\delta}}{\sum_{j=1}^F j^{-\delta}} \quad (1)$$

이때,  $Zipf(f)$ 의  $f$ 는  $f$ 번째로 인기있는 콘텐츠를 의미하고,  $\delta$ 는 분포 특성을 나타낸다. 그리고 콘텐츠  $f$ 의 인기도  $q_f$ 는

$$\frac{\sum_{u \in U} q_{u,f}}{\sum_{u \in U} \sum_{f \in F} q_{u,f}}$$
로 나타낸다.

**인기 유사도:** '인기 유사도'는 두 사용자 또는 사용자와 BS 간의 요청 콘텐츠에 대한 유사도를 의미한다. 먼저 사용자 간의 유사도는 URT에 저장된 인기도 리스트  $q_u$ 를 사용한다. 인기도 리스트  $q_u$ 는 사용자가 요청한 콘텐츠의 인기도를 의미하므로 사용자  $u$ 와  $v$ 의 유사도는  $q_u$ 와  $q_v$ 를 입력으로 코사인 유사도(cosine similarity)연산에 의해 Equation (2)와 같이 계산된다.

$$\Phi_{u,v} = \frac{\sum_{f \in F} q_{u,f}}{\left( \sqrt{\sum_{f \in F} (q_{u,f})^2} \right) \left( \sqrt{\sum_{f \in F} (q_{v,f})^2} \right)} \quad (2)$$

마찬가지로, 사용자와 BS 간의 유사도는  $q_u$ 와  $q_i$ 를 입력으로 Equation (2)를 통해 계산할 수 있다.

### 2.2 문제 공식화

본 논문의 목표는 사용자 연결과 콘텐츠 캐싱을 결합하여 체류시간이 긴 사용자의 요청을 BS에서 최대한 제공하는 것이다.  $x_{u,i}$ 와  $y_{f,i}$ 는 각각 사용자 연결과 캐싱의 결정 변수(decision variable)이다.  $x_{u,i}$ 는 사용자  $u$ 와 BS  $i$ 가 연결된 경우 1이고, 그렇지 않으면 0이다. 또한,  $y_{f,i}$ 가 1인 경우 BS  $i$ 의 캐시에 콘텐츠  $f$ 가 캐싱되었음을 의미하고, 그렇지 않은 경우 0으로 나타낸다. 체류시간이 긴 사용자의 요청을 BS에서 제공하기 위한 JCA문제는 다음과 같이 표현할 수 있다.

$$\max \sum_{u \in U} \sum_{i \in B} \sum_{f \in F} x_{u,i} y_{f,i} q_{u,f} \quad (3a)$$

$$\text{s.t.} \sum_{f \in F} y_{f,i} L_f \leq C, \forall i \in B \quad (3b)$$

$$r_{u,i} \geq r_{th}, \forall u \in U, i \in B \quad (3c)$$

$$\sum_{i \in B} x_{u,i} = 1 \quad (3d)$$

여기서  $L_f$ 가 콘텐츠의 크기일 때, 조건 (3b)는 캐시의 크기에 대한 제약을 의미하고,  $r_{th}$ 가 미리 정해진 체류시간 임계 값일 때, 조건 (3c)는 체류시간에 대한 제약을 나타낸다. 조건 (3d)에 의해 사용자는 하나의 BS에 연결된다.

## 3. 제안 알고리즘

JCA문제는 이전 연구[6-8]에서 NP-hard임이 증명되었다. 그러므로 본 논문에서는 (i)사용자 연결과 (ii)콘텐츠 캐싱 결정문제로 나누어 해결하는 알고리즘을 제안한다. 제안 알고리즘은 다음과 같은 순서로 수행된다.

1) 먼저, EC는 관리하는 지역에 사용자가 머무르는 체류시간을 예측한다.

- 2) 체류시간 예측 이후, 콘텐츠 인기도를 토대로 사용자 간의 유사도를 계산한다. 유사도와 사용자 간의 거리를 고려하여 사용자를 클러스터링한다.
- 3) 그다음으로 클러스터에 포함된 사용자와 BS 간의 유사도를 계산하고, 체류시간을 고려하여 사용자를 BS에 연결한다.
- 4) BS는 연결된 사용자들이 요청한 콘텐츠를 인기도에 따라 캐싱한다. 사용자는 연결된 BS를 통해 요청 콘텐츠를 제공받으며, 캐시에 존재하지 않는 경우 original server로부터 제공받는다.

### 3.1 체류시간 예측

MRP로 모델링한 이동성은 BS에 머무는 시간의 분포가 현재 머무는 BS와 다음 이동할 BS에 의해 결정되고, BS간 이동 확률은 Markov 성질을 가진다는 것을 의미한다. 각 BS는 MRP의 상태(state)를 나타내고, 사용자의 BS이동은 상태전이(state transition)로 고려된다.  $b_n$ 과  $t_n$ 이 각각  $n$ 번째로 이동한 BS와 이동이 발생한 시점을 나타낼 때, 사용자  $u$ 가 BS  $i$ 에서 BS  $j$ 로 시간  $t$ 동안 머무른 이후 이동할 확률  $Q_{i,j}^{(u)}(t)$ 을 수식으로 나타내면 Equation (4)와 같다.

$$Q_{i,j}^{(u)}(t) = \Pr \{b_{n+1} = j, t_{j+1} - t_j \leq t, b_n = i\}, \quad (4)$$

$$= p_{i,j}^{(u)} h_{i,j}^{(u)}(t)$$

Equation (4)의  $p_{i,j}^{(u)}$ 는 Equation (5)와 같이 정의하고,

$$p_{i,j}^{(u)} = \lim_{t \rightarrow \infty} Q_{i,j}^{(u)}(t) \quad (5)$$

$$= \Pr(b_{n+1} = j \mid b_n = i), p_{i,j}^{(u)} \in P^{(u)}$$

Equation (4)의  $h_{i,j}^{(u)}(t)$ 는 Equation (6)과 같이 정의한다.

$$h_{i,j}^{(u)}(t) = \Pr(t_{n+1} - t_n \leq t \mid b_{n+1} = j, b_n = i) \quad (6)$$

$$, h_{i,j}^{(u)}(t) \in H^{(u)}(t)$$

Equation (5)의  $p_{i,j}^{(u)}$ 는 사용자가 BS  $i$ 에서 BS  $j$ 로 이동할 확률을 나타내고, 전체 BS의 수가  $K$ 일 때,  $P^{(u)}$ 는 내재 Markov chain의 확률전이행렬(probability transition matrix)으로써, Equation (7)과 같이 주어진다.

$$P^{(u)} = \begin{bmatrix} p_{1,1}^{(u)} & \dots & p_{1,K}^{(u)} \\ \dots & \ddots & \vdots \\ p_{K,1}^{(u)} & \dots & p_{K,K}^{(u)} \end{bmatrix} \quad (7)$$

Equation (6)의  $h_{i,j}^{(u)}(t)$ 는 BS  $i$ 에서 BS  $j$ 로 이동하여 체류하는 시간 분포를 나타낸다. 본 논문에서 BS  $i$ 에서 BS  $i$ 로 이동하는 경우는 고려하지 않으므로  $P^{(u)}$ 의 대각 원소(diagonal elements)는 모두 0이다.  $Q_{i,j}^{(u)}(t)$ 를 사용하여 이전 위치와 상관없이 사용자가 BS  $i$ 에서 체류하는 시간은 다음과 같이 정의할 수 있다.

$$r_{u,i}(t) = \Pr(t_{n+1} - t_n \leq t \mid b_n = i) \quad (8)$$

$$= \sum_{j \in B} Q_{i,j}^{(u)}(t)$$

따라서 EC가 관리하는 지역에 사용자  $u$ 의 체류시간은 Equation (9)와 같다.

$$r_u = \sum_{i \in B} r_{u,i} \quad (9)$$

EC는 사용자 이동성 테이블(UMT)에 저장된 기록을 활용하여 확률전이행렬  $P^{(u)}$ 과 체류시간 분포행렬  $H^{(u)}(t)$ 을 초기화한다. BS  $i$ 에서 BS  $j$ 로 이동할 확률  $p_{i,j}^{(u)}$ 은 BS  $i$ 에서 BS  $j$ 로 이동한 횟수  $n_{i,j}$ 와 BS  $i$ 에서 이동한 횟수  $n_i$ 를 사용하여

$$p_{i,j}^{(u)} = \frac{n_{i,j}}{n_i}$$

로 초기화되고,  $H^{(u)}(t)$ 는 특정 시간  $\gamma$  ( $\gamma < r_{th}$ )보다 오래 머무른 횟수  $n_{i,j,\gamma}$ 를 사용하여  $h_{i,j}^{(u)}(\gamma) = \frac{n_{i,j,\gamma}}{n_{i,j}}$ 와 같이 초기화된다[2-4, 12]. 사용자  $u$ 가 BS  $i$ 에서 BS  $j$ 로 이동할 때마다  $p_{i,j}^{(u)}$ 와  $H_{i,j}^{(u)}(\gamma)$ 가 업데이트되면,  $Q_{i,j}^{(u)}(t)$ 가 계산된다. Equation (9)에 의해 예측된 체류시간은 UMT에 저장한다. EC는 체류시간을 예측한 이후, 인기 유사도를 토대로 사용자를 클러스터링한다.

### 3.2 사용자 클러스터링

EC는 유사한 사용자들을 같은 BS에 연결하기 위해 Equation (2)을 통해 모든 사용자 간의 인기 유사도  $\phi_{u,v}$ 를 계산한다. 인기 유사도를 토대로 클러스터링을 진행하기 전에, 거리가 먼 사용자들이 같은 BS에 연결되지 않도록 유사도를 조정한다 [6]. 본 논문에서는 BS의 커버리지 지름을 사용자 간의 거리에 대한 기준으로 정하고, BS의 반지름을 구하기 위해 사용자와 BS의 거리  $\rho_u$ 를 Friis' transmission equation[13]에 의해 Equation (10)과 같이 계산한다.

$$\rho_u = \frac{\lambda}{4\pi} \sqrt{\frac{P_b}{P_u G_b G_u}} \quad (10)$$

여기서  $\lambda$ 는 파장(frequency),  $P_b$ 와  $P_u$ 는 각각 BS와 사용자의 전력(power)을 의미하고,  $G_b$ 와  $G_u$ 는 각각 BS, 사용자의 안테나 이득(antenna gain)을 의미한다. 계산된 사용자와 BS의 최대 거리  $\max \{\rho_u\}$ 를 BS의 반지름  $\rho$ 로 고려할 때,  $2\rho$ 보다 사용자 간의 거리가 먼 경우 두 사용자는 같은 클러스터에 할당하지 않는다. 따라서, 사용자  $u$ 의 현재 위치를 좌표  $(x,y)_u$ 로 나타낼 때, 사용자  $u, v$ 의 거리가  $\sqrt{(x_u - x_v)^2 + (y_u - y_v)^2} > 2\rho$ 인 경우 유사도  $\phi_{u,v}$ 를 0으로 조정한다.

사용자 클러스터링은 사용자 유사도  $\phi_{u,v}$ 와 클러스터 개수  $K$ 를 입력으로 클러스터를 결정한다. 클러스터는 각 BS에 할당되므로, 클러스터의 개수는 BS의 수  $K$ 와 동일하게 지정한다. 사용자 클러스터는 다음과 같은 순서로 진행된다.

- 1) 먼저, 사용자 클러스터  $k$ 의 중심(centroid)을  $u_k$ 로 나타낼 때,  $u_k$ 를 선택하기 위해 Equation (10)을 통해 사용자  $u$ 와 다른 사용자들 사이의 평균 유사도를 계산한다.

$$\Phi_u = \frac{\sum_{v \in U \setminus u} \Phi_{u,v}}{L_u - 1} \quad (11)$$

Equation (11)의  $L_u$ 는 전체 사용자 수를 나타낸다.

- 2) 모든 사용자와 유사도가 높은 사용자  $u_1$ 를 첫 번째 클러스터의 중심으로 설정하고, 나머지  $K-1$ 개의 클러스터의 중심은 Equation (11)을 만족하는 사용자로 선택한다.

$$\operatorname{argmax}_u [\Phi_u - \frac{1}{K-1} \sum_{k=2}^{K-1} \Phi_{u,u_k}] \quad (12)$$

Equation (12)을 통해 사용자들과 가장 유사하면서 이미 정해진 클러스터 중심의 인기도와는 다른 사용자가 클러스터 중심으로 선택된다.

- 3) 각 클러스터의 중심이 정해진 다음으로 나머지 사용자들은  $\operatorname{argmax} \Phi_{u,k}$ 를 만족하는 가장 유사한 클러스터에 할당되고, 클러스터  $k$ 에 할당된 사용자는  $U_k$ 로 나타낸다.

### 3.3 사용자 연결

본 논문에서는 Equation (13)과 같이 BS와 사용자 간의 유사도에 대한 총합이 최대화되도록 사용자 연결을 결정한다.

$$\max_x \sum_{u \in U} \sum_{i \in B} x_{u,i} \Phi_{u,i} \quad (13)$$

*s.t.* (3c), (3d)

각 BS에 유사한 콘텐츠 요청을 전송하는 사용자들을 연결함으로써 캐시에 캐싱할 콘텐츠를 특징지을 수 있으며, 이를 통해 간접적으로 캐싱의 효과를 높일 수 있다. Equation (13)을 해결하기 위해, BS와 개별 사용자  $u$ 의 유사도  $\Phi_{u,i}$  대신 BS와 사용자 클러스터링을 통해 지정된 클러스터  $k$ 의 유사도  $\Phi_{k,i}$ 를 계산한다.  $\Phi_{k,i}$ 는 클러스터 중심인  $u_k$ 의 콘텐츠 인기도  $q_k$ 와 BS에 요청된 콘텐츠의 인기도  $q_i$ 를 사용하여 Equation (2)를 통해 유사도를 계산한다. 한편, 사용자와 BS의 연결 이익(benefit)은 유사도  $\Phi_{k,i} \in (0,1]$ 와 체류시간  $r_{k,i} \in (0,1]$ 에 의해 Equation (14)와 같이 정의된다.

$$v_{k,i} = \Phi_{k,i} + r_{k,i} \quad (14)$$

클러스터  $k$ 에 포함된 사용자 수가  $L_k$ 일 때, Equation (14)의  $r_{k,i}$ 는 클러스터  $k$ 에 포함된 사용자 체류시간의 평균  $\frac{1}{L_k} \sum_{u \in U_k} r_u$ 을 나타내며, 최소-최대 정규화[14]를 통해 0과 1 사이의 값으로 나타낸다.

EC는 클러스터를 유사도와 체류시간에 따른 이익  $v_{k,i}$  값이 큰 BS에 연결하기 위해 사용자 연결(User Association:UA) 알고리즘을 수행한다. UA 알고리즘은 BS에 연결되지 않은 클러스터에 대해 반복적으로 수행한다. 먼저 Equation (14)와 같이 사용자 클러스터의 이익  $V_k$ 을 계산한다(line4). 이때 조건 (3c)에 의해 미리 정의된 체류시간 임계 값  $r_{th}$ 보다 적게 머무르는 사용자는 클러스터  $U_k$ 에서 제외한다(line5). 클러스터의 이익  $V_k$ 은 내림차순 정렬하고, 가장 큰 이익  $v_{k,i}$ 을 가지는 BS  $i$ 를

Table 1. User Association Algorithm

#### Algorithm 1. User Association Algorithm

```

Input:  $v_{k,i}, K' = [1, \dots, k, \dots, K]$ 
Output:  $X = [x_{1,i}, \dots, x_{k,i}, \dots, x_{K,i}], i \in B$ 
1: while  $K' \neq \emptyset$  do
2:    $z^* = -\infty$ 
3:   for each cluster  $k$  do
4:      $V_k = \{v_{k,i} : i \in B\}$ 
5:      $U_k = U_k \setminus \{u : r_u < r_{th}\}$ 
6:     Sort  $V_k$  in descending order
7:     Find BS  $i'$  with  $v_{k,i'} = \max\{v_{k,i}\}$ 
8:     if  $V_k \setminus \{i'\} = \emptyset$  then  $z = +\infty$ 
9:     else  $z = v_{k,i'} - \max\{v_{k,i} : i \neq i'\}$ 
10:    if  $z > z^*$  then  $z^* = z, i^* = i', k^* = k$ 
11:   end for
12:    $x_{k^*,i^*} = 1, B' = B' \setminus \{i^*\}, K' = K' \setminus \{k^*\}$ 
13: end while

```

계산한다(line6-7). 만약 클러스터가 최대 이익을 가지는 BS  $i'$  외에 연결할 BS가 없는 경우,  $i'$ 에 먼저 연결한다(line8). 그렇지 않은 경우, 첫 번째와 두 번째로 큰 이익의 차이  $z$ 를 계산한다(line9).  $z$ 값이 클수록 클러스터  $k$ 가 BS  $i'$ 에 연결하였을 때 이익이 커진다는 것을 의미하므로, BS  $i'$ 에 대해  $z$ 값이 가장 큰 클러스터  $k$ 와 연결한다(line10-12). UA는 모든 클러스터가 BS에 연결될 때까지 반복된다.

### 3.4 콘텐츠 캐싱

UA 알고리즘을 통해 사용자가 BS에 연결된 이후, BS는 Equation (3)의 sub-problem인 Equation (15)를 최대화하기 위해 캐시에 저장할 콘텐츠를 결정한다.

$$\max_y \sum_{i \in B} \sum_{u \in U_i} \sum_{f \in F} y_{f,i} q_{u,f} \quad (15)$$

*s.t.* (3b)

Equation (15)를 최대화하기 위해 각 BS는 인기도  $q_f$ 가 높은 콘텐츠를 캐싱한다. BS에 할당된 사용자( $u \in U_i$ )의 콘텐츠  $f$ 에 대한 인기도  $q_f = \sum_{u \in U_i} q_{u,f}$ 를 사용하여 캐싱을 결정한다. 콘텐츠는 인기도를 기준으로 내림차순 정렬하고, 인기도가 높은 콘텐츠부터  $C$ 가 full 상태일 때까지 차례대로 캐싱한다.

## 4. 시뮬레이션

본 장에서는 시뮬레이션을 통하여 이동성 기반의 캐싱 및 사용자 연결 알고리즘의 성능을 기존 알고리즘과 비교한다.

### 4.1 시뮬레이션 설정

체류시간을 고려한 사용자 연결과 콘텐츠 캐싱(JCA) 알고리즘의 성능을 평가하기 위해 Python 기반의 시뮬레이터를 구성하여 활용하였다. 시뮬레이션 파라미터는 Table 2와 같이

Table 2. Simulation Parameters

Parameter	Value
Distance between user and BS, $\rho_u$	[10, 40] m
Sojourn time threshold, $r_{th}$	90 min
Mean sojourn time, $r$	60, 90 min
Number of BSs, $K$	4
Number of users, $L_u$	[100, 200]
Content size, $L_f$	4 MB
Cache size, $C$	600 MB
Number of contents	500
Zipf parameter, $\delta$	0.8
Average latency from BS to user	50 ms
Average latency from original server to user	500 ms

EC의 서비스 범위 내 100명의 사용자가 4개의 BS에 골고루 분포된 상황을 가정하였으며, 이동성은 Random Waypoint Model을 활용하여 발생시켰다[3]. 사용자 요청은 Zipf 분포의 파라미터  $\delta$ 를 0.8로 설정하여 발생시켰으며, 각 사용자는 인기도 리스트  $q_u$ 를 가진다.

제안 알고리즘의 성능 평가를 위해 동일 환경에서 체류시간을 고려하지 않는 JCA-w/o mobility(Joint content Caching and user Association without mobility) 알고리즘[6, 8]과 사용자 연결을 포함하지 않으며 이동성과 인기도에 따라 콘텐츠를 캐싱하는 Greedy 알고리즘[2-4]의 성능을 비교한다. 성능은 각 BS에 요청된 콘텐츠 중에서 BS를 통해 제공되는 요청 비율의 평균을 의미하는 캐시 적중률과 사용자가 요청 콘텐츠를 제공받는데 소요되는 지연시간을 비교한다.

4.2 시뮬레이션 결과

Fig. 1과 Fig. 2는 각각 평균 체류시간( $r$ )은 60(min), BS의 개수( $K$ )가 4일 때 사용자 수에 따른 캐시 적중률과 평균 지연시간을 나타낸다. Fig. 1에서 캐시 크기는 고정된 데 반해, 사용자 수가 증가함에 따라 요청 콘텐츠가 증가하여 캐시 적중률이 감소함을 확인할 수 있다. 제안 알고리즘은 체류시간과 요청 선호도를 고려하였으므로, 비교 알고리즘보다 상대적으로 적은 감소 폭을 보였으며 JCA-w/o mobility와 Greedy 알고리즘보다 캐시 적중률이 각각 평균 9%, 28% 향상됨을 확인할 수 있다. Fig. 2는 Fig. 1에서 확인한 바와 같이, 캐시 적중률이 향상됨에 따라 지연시간이 JCA-w/o mobility와 Greedy 알고리즘보다 각각 최대 13%, 38% 감소를 확인할 수 있다.

Fig. 3과 Fig. 4는 Fig. 1, Fig. 2의 시뮬레이션 환경에서 사용자의 평균 체류시간을 90(min)으로 변경하였을 때, 캐시 적중률과 지연시간을 나타낸다. Fig. 1과 Fig. 3의 비교를 통하여 제안 알고리즘은 EC가 관리하는 지역에 사용자가 머무르는 평균 시간이 증가할수록 성능이 향상됨을 확인할 수 있다. Fig. 4는 제안 알고리즘의 지연시간이 체류시간을 고려하지 않는 JCA-w/o mobility와 사용자 연결을 고려하지 않는 Greedy 알고리즘보다 각각 평균 16%, 29% 감소함을 보인다. 이는 체류시간과 요청 선호도를 고려하는 제안 알고리즘이 기존 알고리즘에 비해 효율적임을 증명한다.

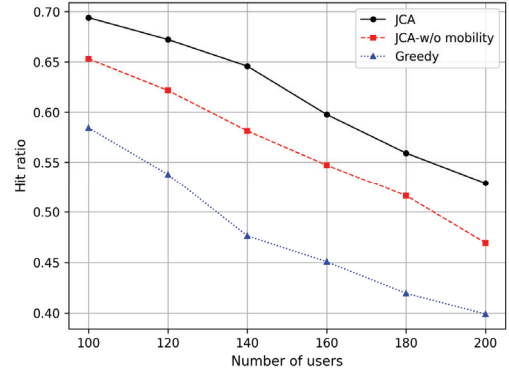


Fig. 1. Hit Ratio with Number of Users When  $r = 60$

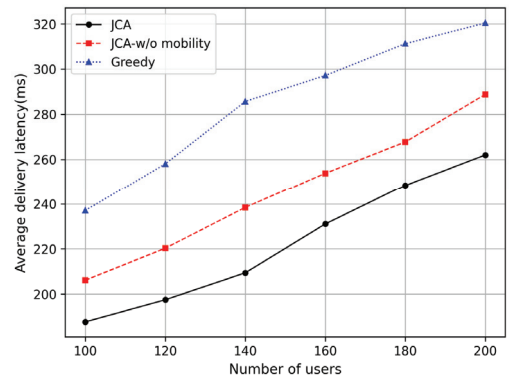


Fig. 2. Average Delivery Latency with Number of Users When  $r = 60$

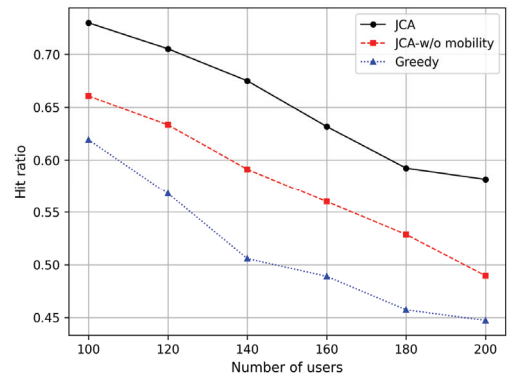


Fig. 3. Hit Ratio with Number of Users When  $r = 90$

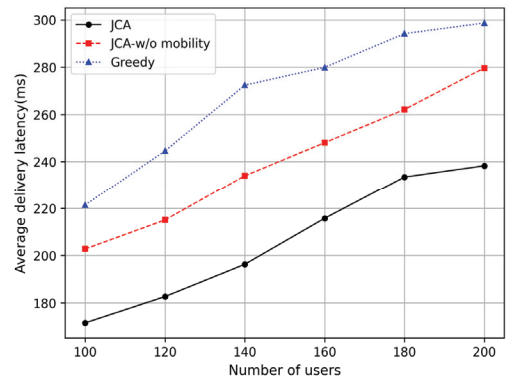


Fig. 4. Average Delivery Latency with Number of Users When  $r = 90$

## 5. 결 론

본 논문에서는 모바일 엣지 네트워크에서 이동성 기반의 캐싱 및 사용자 연결 알고리즘을 제안하였다. 캐시 적중률을 위한 사용자 연결 및 캐싱 문제를 공식화하고, 사용자 연결과 캐싱 문제로 나누어 해결하였다. 제안 알고리즘은 요청 신호도에 따른 유사도와 체류시간을 토대로 사용자 연결을 결정하고, 콘텐츠를 캐싱하였다. 시뮬레이션을 통해 체류시간 또는 사용자 연결을 고려하지 않는 기존 알고리즘에 비해 제안 알고리즘의 향상된 캐시 적중률과 이에 따라 감소한 지연시간을 증명하였다. 본 논문의 연구 결과를 바탕으로 향후 사용자 이동성 특징을 구체화하여 캐싱 및 사용자 연결 알고리즘을 연구할 수 있을 것으로 기대한다.

## References

- [1] N. Gao, X. Xu, Y. Hou, and L. Gao, "A mobility-aware proactive caching strategy in heterogeneous ultra-dense networks," *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications(PIMRC)*, pp.1-7, 2019.
- [2] M. K. Somesula, R. R. Rout, and D. V. L. N. Somayajulu, "Contact duration-aware cooperative cache placement using genetic algorithm for mobile edge networks," in *Computer Networks*, Vol.193, Article 108062, 2021.
- [3] Y. Ye, M. Xiao, and M. Skoglund, "Mobility-aware content preference learning in decentralized caching networks," *IEEE Transactions on Cognitive Communications and Networking*, Vol.6, No.1, pp.62-73, 2020.
- [4] Y. Ye, M. Xiao, Z. Zhang, and Z. Ma, "Performance analysis of mobility prediction based proactive wireless caching," *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, pp.1-6, 2018.
- [5] L. E. Chatzieftheriou, M. Karaliopoulos, and I. Koutsopoulos, "Jointly optimizing content caching and recommendations in small cell networks," *IEEE Transactions on Mobile Computing*, Vol.18, No.1, pp.125-138, 2019.
- [6] G. Darzanos, L. E. Chatzieftheriou, M. Karaliopoulos, and I. Koutsopoulos, "Content preference-aware user association and caching in cellular networks," *2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT)*, pp.1-8, 2020.
- [7] L. E. Chatzieftheriou, G. Darzanos, M. Karaliopoulos, and I. Koutsopoulos, "Joint user association, content caching and recommendations in wireless edge networks," *ACM SIGMETRICS Performance Evaluation Review*, Vol.46, No.3, pp.12-17, 2019.
- [8] K. Ntougias, C. Psomas, E. Demarchou, I. Krikidis, and I. Koutsopoulos, "Joint dynamic wireless edge caching and user association: A stochastic optimization approach," in *2022 IEEE 23rd International Workshop on Signal Processing Advances in Wireless Communication (SPAWC)*, pp.1-5, 2022.
- [9] Y. Li, H. Ma, L. Wang, S. Mao, and G. Wang, "Optimized content caching and user association for edge computing in densely deployed heterogeneous networks," *IEEE Transactions on Mobile Computing*, Vol.21, No.6, pp. 2130-2142, June 2022.
- [10] X. Liu, H. Zhang, K. Long, A. Nallanathan, and V. C. M. Leung, "Energy efficient user association, resource allocation and caching deployment in fog radio access networks," *IEEE Transactions on Vehicular Technology*, Vol.71, No.2, pp.1846-1856, 2022.
- [11] M. Polese, R. Jana, V. Kounev, K. Zhang, S. Deb, and M. Zorzi, "Machine learning at the edge: A data-driven architecture with applications to 5G cellular networks," *IEEE Transactions on Mobile Computing*, Vol.20, No.12, pp.3367-3382, 2021.
- [12] H. Farooq and A. Imran, "Spatiotemporal mobility prediction in proactive self-organizing cellular networks," *IEEE Communications Letters*, Vol.21, No.2, pp.370-373, 2017.
- [13] H. T. Friis, "A note on a simple transmission formula," *Proceedings of the IRE*, Vol.34, No.5, pp.254-256, 1946.
- [14] J. Han, J. Pei, and M. Kamber, "Data mining: Concepts and techniques. amsterdam," The Netherlands: Elsevier, 2011.



이 태 윤

<https://orcid.org/0000-0002-9757-8882>

e-mail : tylee814@yonsei.ac.kr

2019년 한성대학교 전자정보공학과(학사)

2020년 ~ 현재 연세대학교 컴퓨터과학과 석사과정

관심분야 : 엣지 컴퓨팅, 지능형 시스템



이 수 경

<https://orcid.org/0000-0002-3497-3295>

e-mail : sklee@yonsei.ac.kr

1992년 연세대학교 컴퓨터과학과(학사)

1995년 연세대학교 컴퓨터과학과(석사)

2000년 연세대학교 컴퓨터과학과(박사)

2012년 ~ 현재 연세대학교 컴퓨터과학과 교수

관심분야 : 엣지 컴퓨팅, 지능형 IoT 서비스