

# MEC 환경에서의 Social Context를 이용한 트래픽 오프로딩 알고리즘

천혜림\*, 이승규\*, 김재현°

## Traffic Offloading Algorithm Using Social Context in MEC Environment

Hye-Rim Cheon\*, Seung-Que Lee\*, Jae-Hyun Kim°

### 요약

트래픽 오프로딩은 폭발적으로 증가하는 모바일 트래픽에 대응하기 위한 유망 솔루션이다. 오프로딩 방법 중, LIPA/SIPTO 오프로딩에서는 애플리케이션의 QoS 요구사항을 만족하면서 트래픽을 오프로딩할 수 있다. 또한, SNS로 인한 많은 트래픽때문에 social context를 이용한 트래픽 오프로딩이 필요하다. 그러므로, 본 논문에서는 social context를 이용하여 트래픽을 오프로딩하는 LIPA/SIPTO 오프로딩 알고리즘을 제안한다. 먼저, 애플리케이션 인기도를 social context로 이용하여 애플리케이션 선택확률을 정의한다. 그 다음, effective data rate 관점에서 소형셀 사용자의 QoS를 최대화하는 최적의 오프로딩 weighting factor를 찾는다. 마지막으로, 애플리케이션 선택확률과 오프로딩 weighting factor를 기반으로 각 애플리케이션의 오프로딩 비율을 정한다. 성능분석 결과, 제안한 알고리즘의 오프로딩 비율이 기존 알고리즘의 약 46%임에도 불구하고, 제안한 알고리즘의 effective data rate achievement ratio 값이 기존 알고리즘과 비슷한 것을 확인하였다.

**Key Words** : Small Cell, Traffic Offloading, Social Context, MEC, LTE-A

### ABSTRACT

Traffic offloading is a promising solution to solve the explosive growth of mobile traffic. One of offloading schemes, in LIPA/SIPTO(Local IP Access and Selected IP Traffic Offload) offloading, we can offload mobile traffic that can satisfy QoS requirement for application. In addition, it is necessary for traffic offloading using social context due to large traffic from SNS. Thus, we propose the LIPA/SIPTO offloading algorithm using social context. We define the application selection probability using social context, the application popularity. Then, we find the optimal offloading weighting factor to maximize the QoS(Quality of Service) of small cell users in term of effective data rate. Finally, we determine the offloading ratio by this application selection probability and optimal offloading weighting factor. By performance analysis, the effective data rate achievement ratio of the proposed algorithm is similar with the conventional one although the total offloading ratio of the proposed algorithm is about 46 percent of the conventional one.

\* 본 연구는 미래창조과학부의 정부출연급사업의 일환으로 수행하였음. [15Z11110, 트래픽 용량 증대를 위한 액세스 네트워크 원천기술 연구]

• First Author : Ajou University Department of Electrical and Computer Engineering, hyerimn1@ajou.ac.kr, 학생회원

° Corresponding Author : Ajou University Department of Electrical and Computer Engineering, jkim@ajou.ac.kr, 종신회원

\* Electronics and Telecommunications Research Institute, sqlee@etri.re.kr, 정회원

논문번호 : KICS2016-12-370, Received December 2, 2016; Revised January 9, 2017; Accepted January 10, 2017

## 1. 서론

이동통신기술의 발전으로 인해 스마트폰, 태블릿 등과 스마트 모바일 기기가 확산되고 있다. 스마트 모바일 기기는 멀티미디어와 컴퓨팅 지원 기능이 있어서 다양한 애플리케이션을 이용할 수 있는데, 이로 인해 스마트 모바일 기기는 더 많은 트래픽을 발생시키고 있다. 시스코 자료에 따르면, 2015년에 스마트 모바일 기기는 일반 모바일 기기에 비해서 평균 14배 더 많은 트래픽을 발생시켰고, 이는 전체 모바일 데이터 트래픽의 급증으로 이어지고 있다<sup>[1]</sup>. 또한, 스마트 모바일 기기에서 사용가능한 애플리케이션 중 페이스북, 인스타그램 등과 같은 SNS(Social Networking Service) 사용자가 급격히 증가하고 있으며, SNS로 인한 트래픽 발생량이 전체 모바일 트래픽의 많은 부분을 차지하고 있다<sup>[2]</sup>. 또한, 최근 한 연구에서는 인기 있는 콘텐츠의 중복 다운로드로 발생하는 트래픽이 모바일 트래픽의 많은 부분을 차지하고 있음을 보여 주고 있다<sup>[3]</sup>. 따라서, 인기도 등과 같은 social context를 활용하여 모바일 데이터 트래픽을 적절히 분산시키는 방안에 대한 연구가 필요하다.

급격하게 증가하고 있는 모바일 데이터 트래픽을 수용하기 위한 방안 중 하나로 트래픽 오프로딩 기술이 주목받고 있다. 시스코 자료에 따르면, 2015년에 전체 모바일 트래픽의 51%가 WiFi나 펌토셀을 통해 오프로딩 되었음을 알 수 있다<sup>[1]</sup>. 또한, 트래픽 오프로딩 관련 연구가 많이 이루어지고 있는데, 그림 1처럼 오프로딩하는 경로에 따라 D2D(Device-to-Device), WiFi, 소형셀, LIPA/SIPTO(Local IP Access and Selected IP Traffic Offload) 오프로딩으로 나뉜다. D2D는 3GPP에서 ProSe(Proximity Services)라는 이름으로 표준화가 진행되고 있는 기술로, 이를 이용한 트래픽 오프로딩은 서비스 제공자가 콘텐츠를 모바일 네트워크를 통해 타겟 모바일 기기에 전송하면 이웃한 모바일 기기는 기지국이 아닌 타겟 모바일 기기를 통해 트래픽을 전송받는 방식으로 이루어진다<sup>[4,5]</sup>. WiFi 오프로딩은 모바일 기기가 무선랜 환경 안에 있을 때 트래픽을 전송하고, 무선랜 환경에서 벗어났을 때는 다시 무선랜 환경 안에 들어갈 때까지 기다렸다가 전송을 마치는 방식이다<sup>[5]</sup>. 일반적인 소형셀 오프로딩은 매크로셀 기지국이 아닌 가까운 소형셀 기지국을 선택하여 이를 통해 트래픽을 전송하는 방식이다<sup>[5]</sup>. LIPA/SIPTO 오프로딩은 3GPP에서 제안한 소형셀 오프로딩 방법 중 하나로 소형셀 백홀망을 이용하는 방식이다. 일반적인 소형셀 오프로딩 방식이 모

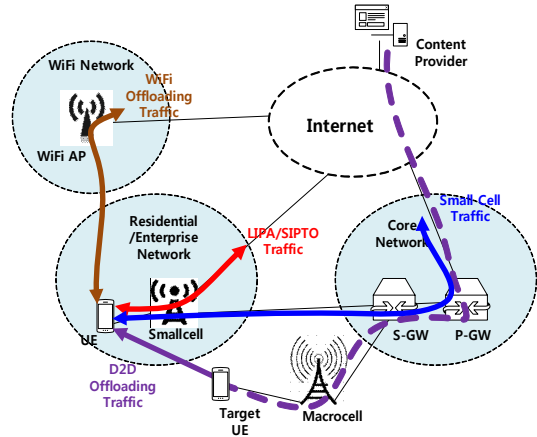


그림 1. 모바일 데이터 오프로딩 경로 종류  
Fig. 1. Mobile data offloading route options

바일 사업자의 코어망을 거치는 것과 달리, LIPA/SIPTO 오프로딩에서는 코어망을 거치지 않고 소형셀 백홀망인 가정/기업 네트워크나 인터넷으로 바로 트래픽을 전송하는 방식이다<sup>[6]</sup>.

D2D와 WiFi는 통신 반경이 크지 않기 때문에 모바일 기기가 D2D나 WiFi 경로로 트래픽을 오프로딩할 때 그 연결이 원활하지 않을 수 있고, 따라서 지연에 민감한 애플리케이션은 QoS(Quality of Service)에 영향이 있을 수 있다. 또한, 기존 소형셀 오프로딩은 단위면적당 시스템 용량은 증가되지만, 모바일 코어망을 거치기 때문에 급증하는 트래픽으로 인한 코어망 부하를 감소시켜 주지는 못한다.

따라서, 본 논문에서는 LIPA/SIPTO 오프로딩을 기반으로 social context를 이용한 트래픽 오프로딩 알고리즘을 제안한다. Social context를 오프로딩 알고리즘에 적용하기 위해서는 복잡한 계산이 필요한데, 이 때문에 알고리즘 계산 주체가 모바일 기기나 소형셀이 아니라 MEC(Mobile-Edge Computing) server인 트래픽 오프로딩 알고리즘을 제안하고자 한다.

본 논문은 다음의 순서로 구성된다. 2장에서는 모바일 트래픽 오프로딩, MEC 관련 연구에 대해 설명하고, 3장에서는 social context를 이용한 LIPA/SIPTO 오프로딩 알고리즘을 제안한다. 4장에서는 제안한 알고리즘의 성능분석 및 기존 알고리즘과의 성능을 비교하고, 5장에서 결론을 맺는다.

## II. 관련연구

### 2.1 모바일 트래픽 오프로딩

앞서 설명한 바와 같이 모바일 트래픽 오프로딩에 관한 연구는 오프로딩 경로에 따라 D2D, WiFi, 소형셀, LIPA/SIPTO의 4가지 관점에서 연구가 이루어지고 있다. 먼저, [7]과 [8]에서는 D2D 오프로딩 관련 연구를 수행하였다. [7]에서는 현재 셀룰러 네트워크에서 D2D의 비전에 대해 설명하고, 시뮬레이션을 통해 D2D 오프로딩을 수행하였을 때의 네트워크 성능 이득을 추정하였다<sup>[7]</sup>. [8]에서는 셀룰러 네트워크의 다운로드 주파수 대역을 재사용하기 위한 D2D 오프로딩 방법을 제안하였다. 이 논문에서 제안한 D2D 오프로딩 방법은 다운로드 주파수 대역 재사용을 위한 채널 이득 추정, 채널할당, D2D pair의 전력제어와 이웃 모바일기기의 콘텐츠 이용가능성 추정 알고리즘을 포함하고 있다<sup>[8]</sup>. 그러나, D2D 오프로딩에서는 좁은 통신반경과 이동성으로 인해 이웃 모바일기기에서 원하는 콘텐츠를 찾을 수 없을 수 있기 때문에, D2D 오프로딩 관련연구에서는 지연에 민감하지 않은 콘텐츠를 오프로딩 한다는 가정을 하고 있다.

[9]와 [10]에서는 WiFi 오프로딩에 관한 연구를 수행하였다. [9]에서는 2주 반 동안 도심지역에서 97명 아이폰 사용자의 WiFi 연결에 대한 통계정보를 수집하는 정량적인 연구를 수행하였다. 이러한 연구 결과를 통해 모바일 데이터 트래픽의 65%가 지연없이 WiFi망으로 오프로딩하고 있고, WiFi 존에 진입할 때까지 데이터 전송을 지연하는 시간이 100초 일 경우는 트래픽 전송이득이 2-3% 미만이지만, 지연시간이 1시간 이상일 경우 트래픽 전송이득은 29% 이상 증가하는 것을 보여주고 있다<sup>[9]</sup>. [10]에서는 도심지역에서의 기지국 전력, 공개 WiFi AP 전력, 모바일 기기 전력 정보를 통합하여 가장 에너지 효율이 좋은 WiFi AP를 선택하는 알고리즘과 이를 수행하는 WiFi 오프로딩 아키텍처를 제안하고 있다. 또한, 실험을 통해 80%이상 에너지를 절약할 수 있음을 증명하고 있다<sup>[10]</sup>. 하지만, WiFi 오프로딩에서는 좁은 WiFi 통신반경 안에서만 오프로딩이 가능하며, 도심지역과 같이 많은 사용자가 동시에 WiFi 망을 이용할 경우 WiFi 품질이 저하되는 한계점이 있다.

[11]에서는 소형셀 오프로딩에 대해 연구하였다. 이 논문에서는 일부 소형셀의 백홀 링크에서 병목현상이 발생하였을 경우, 백홀 용량에 여유가 있는 인접 소형셀로 사용자를 오프로딩하는 알고리즘을 제안하였고, 이를 통해 무선 자원 및 백홀 링크 용량에 맞추

어 더 많은 사용자의 QoS를 지원할 수 있음을 보여주었다<sup>[11]</sup>. 하지만, 일반적인 소형셀 오프로딩은 매크로셀에서 소형셀로 오프로딩하는 방식으로 소형셀의 백홀 링크를 통해 모바일 코어망으로 트래픽이 전송되기 때문에 모바일 사업자 코어망 트래픽이 오프로딩되지는 않는다. 또한, 소형셀로 인해 개선된 단위면적당 시스템 용량으로 인해 더 많은 트래픽이 발생할 수 있고, 이로 인해 모바일 코어망의 부하가 가중될 수 있다.

[12], [13]에서는 LIPA/SIPTO 오프로딩에 대해 연구하고 있다. [12]에서는 베어러 단위로 오프로딩을 수행하는 알고리즘을 제안하고 있다. 이 알고리즘에서는 오프로딩 정책 table을 토대로 각 베어러의 오프로딩 여부를 결정하여 오프로딩 타입 flag을 마킹하면 이를 토대로 RAN(Radio Access Network) node가 오프로딩을 수행한다<sup>[12]</sup>. [13]에서는 DNS(Domain Name System)를 기반으로 한 오프로딩 방법을 제안하고 있다. 제안한 방법은 먼저 UE(User Equipment)가 콘텐츠 서버의 IP address를 요청하는 DNS request를 전송하면 operator DNS 서버가 LIPA/SIPO 오프로딩 여부를 포함한 DNS reply를 L-GW(Local-Gateway)를 통해 UE로 전송하고 L-GW는 DNS reply가 LIPA/SIPO 오프로딩일 경우 그에 해당하는 동작을 적용한다<sup>[13]</sup>. LIPA/SIPTO 오프로딩은 소형셀의 백홀망을 이용하는 오프로딩 알고리즘이기 때문에, D2D, WiFi 오프로딩에 비해 지연에 민감한 트래픽의 QoS 요구사항을 만족할 수 있다.

### 2.2 MEC 관련연구

MEC, 즉, Mobile Edge Computing은 RAN 내 근접한 거리에 있는 모바일 기기에 cloud computing 기능을 제공하는 5G의 핵심기술이다. MEC에서는 RAN edge에 MEC 서버가 위치하여 애플리케이션 개발자와 콘텐츠 사업자에게 저지연, 광대역의 서비스 환경을 제공하며, 실시간 RAN 정보를 획득할 수 있기 때문에 context-related 서비스의 개발이 가능하다<sup>[14]</sup>. 소형셀 포럼에서는 이러한 MEC 기술을 소형셀에 적용하는 방법 및 소형셀 기능의 가상화에 대한 연구를 진행하고 있다<sup>[15]</sup>. [16]에서는 소형셀 환경에서 사용자 context를 이용하여 MEC 서버에 트래픽을 캐싱하는 알고리즘을 제안하였다<sup>[16]</sup>. [17]에서는 5G HetNet(Heterogeneous Network)의 MEC 환경에서 에너지 효율적인 오프로딩 계산 메커니즘을 제안하였다. 제안한 메커니즘은 오프로딩 계산 latency 제약 환경에서 오프로딩 계산과 파일 전송에 사용되는 에너

지를 최소화하기 위한 오프로딩과 radio resource 할당을 동시에 최적화하는 방법이다<sup>17)</sup>. 이와 같이, 오프로딩 computing과 같은 복잡한 기능을 MEC 서버에 두면 오프로딩 계산의 효율이 개선될 수 있다.

### III. 제안하는 트래픽 오프로딩 알고리즘

앞서 언급한 바와 같이, D2D나 WiFi 오프로딩을 수행할 경우 지연에 민감한 애플리케이션의 QoS는 감소될 수 있다. 반면에 소형셀 백홀망을 이용하여 오프로딩을 하는 LIPA/SIPTO 오프로딩은 QoS 요구사항을 만족시키면서도 코어망의 부하를 감소시킬 수 있다. 또한, SNS상의 공유와 인기 콘텐츠의 중복 다운로드 인한 트래픽 증가는 모바일 코어망의 부담이 되기 때문에, social context를 이용하여 오프로딩할 트래픽을 선택하는 것이 효과적일 것이다.

따라서 본 논문에서는 사용자 입장에서 QoS를 최대한으로 만족시키기 위해 social context를 이용하여 소형셀의 백홀망으로 오프로딩할 트래픽을 선택하는 LIPA/SIPTO 트래픽 오프로딩 알고리즘을 제안한다. 제안하는 알고리즘은 total effective data rate achievement ratio가 최대가 되도록 각 애플리케이션의 오프로딩 비율을 결정한다. Effective data rate achievement ratio는 사용자 입장에서 각 애플리케이션의 effective data rate, 즉, QoS를 만족시키기 위해 실제로 필요한 data rate를 만족시키는 정도를 의미한다. 각 애플리케이션의 오프로딩 비율은 애플리케이션 선택 확률에 근거하여 계산되는데, 이때, 애플리케이션 선택확률은 social context를 사용하여 추정된다. 제안하는 알고리즘에서는 social context로 각 애플리케이션의 인기도를 선택하였다. 특히, 소형셀 사용자의 QoS를 만족시키기 위해 애플리케이션 선택확률을 추정할 때 전체 네트워크에서의 인기도뿐만 아니라 소형셀 사용자 내에서의 인기도를 반영하였다.

제안하는 알고리즘은 그림 2와 같이 의사코드로 표현된다. 알고리즘은 크게 5단계로 나뉜다. 1단계에서는 애플리케이션 선택 확률  $P_i$ 를 추정한다. 애플리케이션 선택확률은 사용자가  $P_i$ 의 확률을 가지고 애플리케이션을 선택한다는 것을 의미한다. 애플리케이션 선택확률은 각각 소형셀 사용자 내, 전체 네트워크 내에서의 애플리케이션 인기도를 사용하여 추정하며 다음과 같이 정의한다.

$$P_i = P_{i\_sc} \cdot P_{i\_net} \quad (1)$$

#### Algorithm 1 Social Context Offloading Algorithm

```

1: Phase 1 Estimate the application selection probability
2:   for  $i = 1$  to  $I$  do //  $i$  is application index
3:     
$$P_{i\_sc} = \frac{1}{I} \sum_{k=1}^I \frac{m_k}{n}, \quad \forall k \in I_i$$

4:   end for
5:   for  $i = 1$  to  $I$  do
6:     
$$P_{i\_net} = \frac{\alpha_i}{n} e^{-\frac{\alpha_i}{n}}$$

7:   end for
8:   for  $i = 1$  to  $I$  do
9:     
$$P_i = P_{i\_sc} \cdot P_{i\_net}$$

10:  end for
11: Phase 2 Calculate the offloading weighting factor for each application
12:  for  $h = 1$  to  $H$  do //  $h$  is a case index of the offloading weighting factor combination
13:    
$$D^h = (d_{ij}^h)_{I \times I} = \begin{pmatrix} d_{11}^h & d_{12}^h & \cdots & d_{1I}^h \\ d_{21}^h & d_{22}^h & \cdots & d_{2I}^h \\ \vdots & \vdots & \ddots & \vdots \\ d_{I1}^h & d_{I2}^h & \cdots & d_{II}^h \end{pmatrix},$$

14:    
$$d_{ii}^h = 1, \quad d_{ji}^h = 1 / d_{ij}^h$$

15:    for  $i = 1$  to  $I$  do
16:      
$$w_{OL\_i\_h} = \frac{\sqrt{d_{i1}^h \times d_{i2}^h \times \cdots \times d_{iI}^h}}{\sum_{i=1}^I \sqrt{d_{i1}^h \times d_{i2}^h \times \cdots \times d_{iI}^h}}$$

17:    end for
18:  end for
19: Phase 3 Calculate the offloading probability for each application
20:  for  $i = 1$  to  $I$  do
21:    for  $h = 1$  to  $H$  do
22:      
$$P_{OL\_i\_h} = \frac{w_{OL\_i\_h} \cdot P_i}{\sum_{i=1}^I P_i}$$

23:    end for
24:  end for
25: Phase 4 Find the maximum total effective data rate achievement ratio and offloading weighting factor
26:  for  $h = 1$  to  $H$  do
27:    
$$ERA_h = \sum_{i=1}^I \frac{R_{N\_i\_h}}{R_i}$$

28:  end for
29:   $ERA_{max} = \max(ERA_h)$ 
30:   $w_{OL\_i\_max}$  is the offloading weighting factor for the case of  $ERA_{max}$ 
31: Phase 5 Determine the offloading ratio
32:  
$$r_{OL} = \frac{\sum_{i=1}^I (P_{OL\_i\_max} \cdot TL_i)}{TL_{total}}$$


```

그림 2. 제안한 트래픽 오프로딩 알고리즘  
Fig. 2. Proposed traffic offloading algorithm

$P_{i\_sc}$ 는 소형셀 사용자가 애플리케이션  $i$ 를 선택할 확률이며, 소형셀 사용자 내에서의 인기도를 사용하여 계산된다. 소형셀 사용자 내에서의 애플리케이션의 인기도는 해당 애플리케이션으로 분류되는 콘텐츠의 사용횟수에 의해 추정되며, 이를 이용하여  $P_{i\_sc}$ 는 다음과 같이 정의된다.

$$P_{i\_sc} = \frac{1}{l_i} \sum_{k=1}^{l_i} \frac{m_k}{n}, \forall k \in \mathbf{I}_i \quad (2)$$

여기서  $i$ 는 애플리케이션 index,  $k$ 는 콘텐츠 index,  $l_i$ 는 애플리케이션  $i$ 로 분류되는 콘텐츠의 수,  $m_k$ 는 콘텐츠  $k$ 의 사용횟수,  $n$ 는 소형셀 사용자 수,  $\mathbf{I}_i$ 는 애플리케이션  $i$ 로 분류되는 콘텐츠 집합을 의미한다.

$P_{i\_net}$ 은 전체 네트워크 내에서 애플리케이션  $i$ 를 선택할 확률이며, IBP(Indian Buffet Process) 모델을 사용하여 다음과 같이 정의한다<sup>[18]</sup>.

$$P_{i\_net} = \frac{\alpha_i}{n} \cdot e^{-\frac{\alpha_i}{n}} \quad (3)$$

여기서  $\alpha_i$ 는 Poisson 파라미터로 전체 네트워크에서 애플리케이션  $i$ 의 인기도로 결정된다.

2단계에서는 각 애플리케이션의 오프로딩 weighting factor를 계산한다. 이 때, 특정 기준에 근거하여 각 애플리케이션의 상대적인 중요도에 따라 오프로딩 weighting factor를 결정한다. 이를 위해 여러 기준 요소에 근거한 decision-making 지원 도구인 AHP(Analytic Hierarchy Process)를 사용한다<sup>[19]</sup>. AHP에서는 각 요소별로 pair-wise 비교하여 상대적인 중요도 순위를 매겨 이를 matrix로 만들고, 이를 토대로 weighting factor를 도출한다. 먼저, 특정 기준에 따라 각 애플리케이션의 상대적인 중요도를 결정하고 이를 다음과 같은 matrix로 만든다.

$$\mathbf{D}^h = (d_{ij}^h)_{I \times I} = \begin{bmatrix} d_{11}^h & d_{12}^h & \cdots & d_{1I}^h \\ d_{21}^h & d_{22}^h & \cdots & d_{2I}^h \\ \vdots & \vdots & \ddots & \vdots \\ d_{I1}^h & d_{I2}^h & \cdots & d_{II}^h \end{bmatrix}, d_{ii}^h = 1, d_{ji}^h = 1/d_{ij}^h \quad (4)$$

이때,  $h$ 는 애플리케이션의 상대적인 중요도를 판단할 기준의 종류(data rate, delay 등),  $I$ 는 전체 애플리케이션의 수,  $d_{ij}^h$ 는  $h$ 번째 기준으로 애플리케이션  $i$ 와

애플리케이션  $j$ 의 상대비교로 결정된 값이다. 이러한 matrix를 근거로 각 애플리케이션의 오프로딩 weighting factor를 다음과 같이 정의한다.

$$w_{OL\_i\_h} = \frac{\sqrt[I]{d_{i1}^h \times d_{i2}^h \times \cdots \times d_{iI}^h}}{\sum_{i=1}^I \sqrt[I]{d_{i1}^h \times d_{i2}^h \times \cdots \times d_{iI}^h}} \quad (5)$$

다음 3단계에서는 1, 2단계에서 정의된 값을 토대로 각 애플리케이션의 오프로딩 확률을 계산하며, 다음과 같이 정의한다.

$$P_{OL\_i\_h} = \frac{w_{OL\_i\_h} \cdot P_i}{\sum_{i=1}^I P_i} \quad (6)$$

이때,  $w_{OL\_i\_h}$ 는  $h$ 번째 기준인 경우의 애플리케이션  $i$  오프로딩 weighting factor이다. 오프로딩 weighting factor가 반영된 오프로딩 확률은 4단계에서 total effective data rate achievement ratio를 구하는 데에도 사용된다.

4단계에서는 앞서 언급한 바와 같이 사용자 입장에서 QoS를 최대로 만족시키기 위해 total effective data rate achievement ratio의 최대값과 이 때의 각 애플리케이션의 오프로딩 weighting factor를 찾는다. 먼저, 기준  $h$ 에 따라 정해진 오프로딩 weighting factor 조합 별로 total effective data rate achievement ratio를 구하며 다음과 같이 계산할 수 있다.

$$ERA_h = \sum_{i=1}^I \frac{R_{N\_i\_h}}{R_i} \quad (7)$$

$R_{N\_i\_h}$ 는  $h$ 번째 기준일 때 애플리케이션  $i$ 의 expected data rate이고 다음과 같이 정의된다.

$$R_{N\_i\_h} = P_{OL\_i\_h} \cdot SR_{LN} + (1 - P_{OL\_i\_h}) \cdot SR_{CN} \quad (8)$$

이때,  $SR_{LN}$ 과  $SR_{CN}$ 은 각각 오프로딩 경로 네트워크와 코어 네트워크의 supportable data rate이며, 현재 네트워크 환경조건에서 네트워크가 실제로 지원할 수 있는 data rate를 의미한다.  $SR_{LN}$ 은 다음과 같이 정의된다.

$$SR_{LN} = T_{LN} \cdot b_{LN} = \lambda_{cr_{LN}} \cdot N_{LN} \cdot b_{LN} \quad (9)$$

여기서  $TL_N$ 은 네트워크 용량 초과없이 단위시간당 end-to-end 통신 성공 최대 횟수,  $b_{LN}$ 은 평균 패킷 크기,  $\lambda_{CLN}$ 는 노드의 최대 패킷 generation rate,  $N_{LN}$ 은 오프로딩 경로 네트워크의 전체 노드수를 의미한다<sup>[20,21]</sup>.  $SR_{CN}$ 은  $SR_{LN}$ 과 유사하게 정의된다.

$R_i$ 는 애플리케이션  $i$ 의 effective data rate로 타겟 애플리케이션  $i$ 의 QoS 요구사항을 만족시키기 위해 필요한 data rate를 의미하며 다음과 같이 정의된다.

$$R_i = g_i \cdot s_i \cdot (1 - L_i) \quad (10)$$

여기서  $g_i$ 는 요구 data rate,  $s_i$ 는 패킷을 성공적으로 전송하기 위한 요구 전송횟수,  $L_i$ 는 packet loss ratio bound를 의미하며 애플리케이션 종류에 따라 그 값이 다르게 정의된다<sup>[21]</sup>.

이렇게 오프로딩 weighting factor 조합 별 total effective data rate achievement ratio를 구하고 나면, 이 중 가장 큰 값을 total effective data rate achievement ratio의 최대값  $ERA_{max}$ 로 정하고, 또한, 이 때 해당하는 오프로딩 weighting factor를  $w_{OL\_i\_max}$ 를 정한다.

마지막 5단계에서는 최종적으로 각 애플리케이션 별 오프로딩 비율을 결정하며, 다음과 같이 정의된다.

$$r_{OL\_i} = P_{OL\_i\_max} \cdot TL_i = \frac{w_{OL\_i\_max}}{\sum_{i=1}^I P_i} \cdot TL_i \quad (11)$$

이때,  $P_{OL\_i\_max}$ 는 total effective data rate achievement ratio 값이 최대일 때 오프로딩 확률,  $TL_i$ 는 애플리케이션  $i$ 의 트래픽 볼륨이다. 또한, 전체 오프로딩 비율은 다음과 같이 정의한다.

$$r_{OL} = \frac{\sum_{i=1}^I (P_{OL\_i\_max} \cdot TL_i)}{TL_{total}} \quad (12)$$

여기서  $TL_{total}$ 은 네트워크 전체의 전체 트래픽 볼륨이다.

#### IV. 성능분석

제안한 알고리즘의 성능을 평가하기 위해 MATLAB을 이용하여 시뮬레이션을 수행하였다. 본

논문에서는 제안한 알고리즘과 애플리케이션의 QoS 요구사항을 기준으로 오프로딩하는 기존 알고리즘과의 성능을 비교하였다<sup>[20]</sup>. 성능분석을 위해 소형셀 내에서 애플리케이션 인기도에 따른 3가지 시나리오를 가정하였는데, 첫 번째는 HD video, web, SD video, music/audio, 두 번째는 web, music/audio, SD video, HD video, 마지막 세 번째는 music/audio, web, SD video, HD video 순으로 인기가 높다고 가정하였다. 트래픽 data rate는 4가지 class로 나뉘어 발생시켰고, 4가지 class별 트래픽 발생 비율은 시스코의 자료를 참고하였다<sup>[1,23]</sup>. Delay budget, packet error loss rate 과 같은 QoS 요구사항 파라미터는 3GPP 표준문서의 “Standardized QoS Class Identifier of QCI characteristics”를 참고하였다<sup>[24]</sup>. 각 애플리케이션의 오프로딩 weighting factor는 AHP를 통해 구했는데, 4가지 기준별로 pair-wise matrix를 만든 후, 각각 오프로딩 weighting factor 조합을 구했다. Weighting 1은 모든 애플리케이션의 중요도가 똑같은 경우, weighting 2는 effective data rate가 높은 애플리케이션이 중요한 경우, weighting 3는 delay budget이 큰 애플리케이션이 중요한 경우, weighting 4는 traffic data rate이 큰 애플리케이션이 중요한 경우로 가정하여 그 값을 구했다. 자세한 파라미터는 표 1에 정리되어 있다.

먼저 그림 3은 각 시나리오별로 오프로딩 weighting factor 조합에 따른 total effective data rate achievement ratio를 보여주고 있다. 각 애플리케이션의 오프로딩 weighting factor로 동일하게 설정한 경우(weighting 1), 모든 시나리오에서 오프로딩을 하지 않은 경우와 비슷한 total effective data rate achievement ratio 값을 보여주고 있다. 따라서 total effective data rate achievement ratio를 최대로 만들기 위해서는 최적의 오프로딩 weighting factor 조합을 찾을 필요가 있음을 보여주고 있다. 또한, 시나리오 1에서 traffic data rate이 큰 애플리케이션의 중요도가 높은 오프로딩 weighting 조합(weighting 4)일 때 가장 큰 total effective data rate achievement ratio 값을 보여주고 있는데, 시나리오 1에서 소형셀 사용자가 HD video 애플리케이션을 가장 많이 사용하는 것으로 가정하였고, HD video는 data rate이 높기 때문에 가장 큰 total effective data rate achievement ratio를 얻어낼 수 있다. 그리고 시나리오 2에서는 delay budget이 큰 애플리케이션의 중요도가 높은 오프로딩 weighting 조합(weighting 3)일 때 가장 큰 total effective data rate achievement ratio 값을 가지는데,

표 1. 시뮬레이션 파라미터  
Table 1. Simulation parameters

| Parameter  |             | Value  |
|--|-------------|--|
| Data Rate  | Web         | Uniform(150, 449) kbps   |
|  | Music/Audio | Uniform(450, 749) kbps   |
|  | SD Video    | Uniform(750, 1149) kbps  |
|  | HD Video    | Uniform(1150, 2300) kbps   |
| Delay Budget                                     | Web         | 300 ms   |
|  | Music/Audio | 100 ms   |
|  | SD Video    | 200 ms   |
|  | HD Video    | 150 ms   |
| Packet Error Rate                                | Web         | 10-6   |
|  | Music/Audio | 10-3   |
|  | SD Video    | 10-6   |
|  | HD Video    | 10-3   |
| Traffic Generation Ratio                         | Web         | 14.6 %   |
|  | Music/Audio | 13.4 %   |
|  | SD Video    | 27.7 %   |
|  | HD Video    | 44.3 %   |
| Load Exponent, $\delta$                          |             | 1.8  |
| Load Scaling Constant, $\epsilon$                |             | 0.01   |
| Initial NLN                                      |             | 2  |
| Initial NCN                                      |             | 200  |
| Offloading Weighting Factors of Each Application |             | Weighting1:<br>(0.25 0.25 0.25 0.25)<br>Weighting2:<br>(0.0429 0.6931 0.087 0.177)<br>Weighting3:<br>(0.6931 0.0429 0.177 0.087)<br>Weighting4:<br>(0.0429 0.087 0.177 0.6931) |

이는 시나리오 2에서 소형셀 사용자가 web 애플리케이션을 가장 많이 사용하는 것으로 가정하였고, web의 delay budget이 가장 크기 때문에 이 weighting factor 조합에서 가장 큰 total effective data rate achievement ratio 값을 나타내게 된다.

그림 4는 제안한 알고리즘과 기존의 QoS 요구사항을 기준으로 오프로딩 하는 알고리즘의 total effective data rate achievement ratio 값을 비교하고 있다. 그림에서 보는 바와 같이 제안한 알고리즘과 기존 알고리즘은 유사한 total effective data rate achievement ratio 값을 보이고 있다. 그러나 그림 5에서 보면 제안한 알고리즘의 전체 오프로딩 비율이 기존 알고리즘

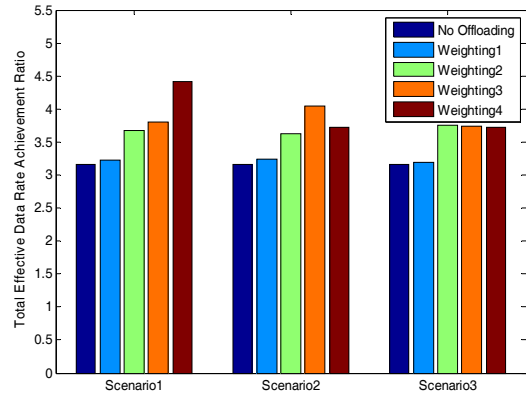


그림 3. 시나리오별 total effective data rate achievement ratio  
Fig. 3. Total effective data rate achievement ratio by scenario

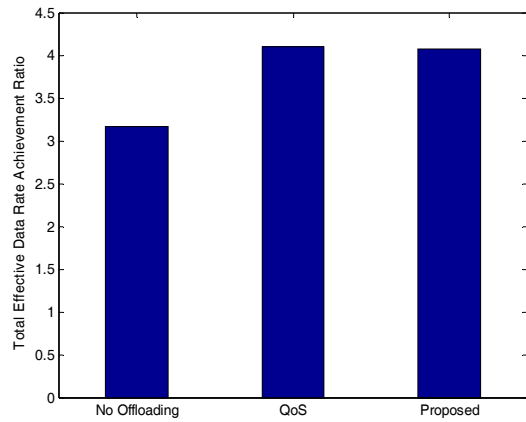


그림 4. 알고리즘별 total effective data rate achievement ratio  
Fig. 4. Total effective data rate achievement ratio by scenario

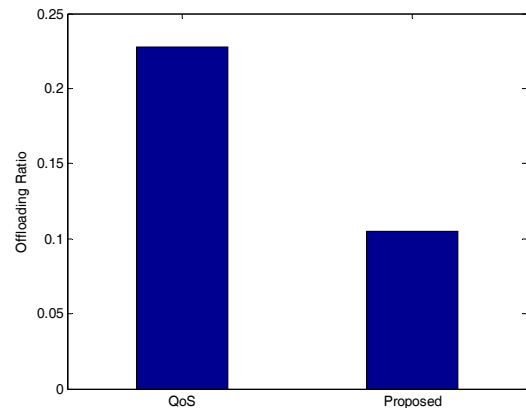


그림 5. 알고리즘별 전체 오프로딩 비율  
Fig. 5. Total offloading ratio by algorithm

의 약 46% 정도임을 확인할 수 있다. 이는 제안한 알고리즘이 social context를 사용하여 애플리케이션별로 적절한 오프로딩 weighting factor 조합을 찾아서, 트래픽을 적게 오프로딩하면서도 total effective data rate achievement ratio 값을 동일하게 최대로 할 수 있음을 의미한다.

## V. 결 론

본 논문에서는 social context를 이용한 새로운 LIPA/SIPTO 트래픽 오프로딩 알고리즘을 제안하였다. 제안한 알고리즘에서는 소형셀 사용자 내외 전체 네트워크에서의 애플리케이션 인기도를 social context로 선택하여 이를 기반으로 애플리케이션 선택 확률을 정의하였다. 또한, 소형셀의 total effective data rate achievement ratio를 최대로 하기 위해서 최적의 오프로딩 weighting factor 조합을 구해낸다. 이렇게 구한 애플리케이션 선택확률과 오프로딩 weighting factor 조합을 이용하여 각 애플리케이션의 오프로딩 비율을 결정한다. 시뮬레이션을 통해 최적 오프로딩 weighting factor 조합 도출의 필요성을 보여주었다. 또한, 제안한 알고리즘이 기존 알고리즘의 약 46%정도만 오프로딩 해도 기존 알고리즘과 비슷한 total effective data rate achievement ratio 값을 얻을 수 있음을 보여주었다.

## References

- [1] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015 - 2020*, Feb. 2016.
- [2] *Mobile Data Traffic Surpasses Voice(2010)*, Retrieved April 13, 2016, from <http://www.cellularnews.com/story/42543.php>,
- [3] M. Cha, H. Kwak, P. Rodriguez, Y. Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: Analyzing the world's largest user generated content video system," in *Proc. 7th ACM SIGCOMM IMC*, pp. 1-14. San Diego, CA, USA, Oct. 2007,
- [4] H. M. Kim, H. N. Lee, and S. K. Kim, "Grouping resource allocation scheme for D2D communications," *J. KICS*, vol. 40, no. 8, pp. 1532-1541, Aug. 2015
- [5] A. Aijaz, H. Aghvami, and M. Amani, "A survey on mobile data offloading: Technical and business perspectives," *IEEE Wirel. Commun.*, vol. 20, no. 2, pp. 104-112, Apr. 2013.
- [6] 3GPP TR 23.829 V10.0.1, *3GPP technical specification group services and system aspects; local IP access and selected IP traffic offload*, Oct. 2011.
- [7] S. Andreev, A. Pyattaev, K. Johnson, O. Galinina, and Y. Koucheryavy, "Cellular traffic offloading onto network-assisted device-to-device connections," *IEEE Commun. Mag.*, vol. 52, no. 4, pp. 20-31, Apr. 2014.
- [8] Z. Wang and V. W. S. Wong, "A novel D2D data offloading scheme for LTE networks," in *Proc. IEEE ICC 2015*, pp. 3107-3112, London, UK, Jun. 2015
- [9] K. Lee, J. Lee, and Y. Yi, "Mobile data offloading: How much can WiFi deliver?," *IEEE/ACM Trans. Networking*, vol. 21, no. 2, pp. 536-550, Apr. 2013.
- [10] A. Y. Ding, B. Han, Y. Xiao, P. Hui, A. Srinivasan, M. Kojo, and S. Tarkoma, "Enabling energy-aware collaborative mobile data offloading for smartphones," in *Proc. SECON 2013*, pp. 487-495, New Orleans, USA, Jun. 2013
- [11] C. S. Yang and C. G. Kang, "QoS-Oriented user association in HetNet with a backhaul constraint," *J. KICS*, vol. 39, no. 10, pp. 654-663, Oct. 2014.
- [12] L. Ma and W. Li, "Traffic offload mechanism in EPC based on bearer type," in *Proc. WiCOM 2011*, Wuhan, China, Sept. 2011.
- [13] K. Samdanis, T. Taleb, and S. Schmid, "Traffic offload enhancements for eUTRAN," *IEEE Commun. Surveys & Tuts.*, vol. 14, no. 3, pp. 884-896, 3rd quarter, 2012.
- [14] ETSI, *Mobile-edge computing—introductory technical white paper*, Sept. 2014.
- [15] Small Cell Forum Release 7.0 Document 154.07.02, *Virtualization in small cell networks*, Jun. 2015.
- [16] H. K. Jung, S. Jung, D. H. Lee, S. Q. Lee, and J. H. Kim, "Wireless caching algorithm



based on user's context in smallcell environments," *J. KICS*, vol. 41, no. 7, pp. 789-798, Jul. 2016

- [17] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896-5907, 2016.
- [18] T. L. Griffiths and Z. Ghahramani, "The indian buffet process: An introduction and review," *J. Mach. Learn. Res.*, vol. 12, no. 4, pp. 1185 - 1224, Apr. 2011.
- [19] H. Tanaka, S. Tsukao, D. Yamashita, T. Niimura, and R. Yokoyama, "Multiple criteria assessment of substation conditions by pair-wise comparison of analytic hierarchy process," *IEEE Trans. Power Delivery*, vol. 25, no. 4, pp. 3017-3023, Oct. 2010.
- [20] H. R. Cheon, S. Q. Lee, and J. H. Kim, "New LIPA/SIPTO offloading algorithm by network condition and application QoS requirement," in *Proc. ICTC 2015*, pp. 191-196, Jeju Island, Korea, Oct. 2015.
- [21] B. Soelistijanto and M. Howarth, "Traffic distribution and network capacity analysis in social opportunistic networks," in *Proc. WiMob 2012*, pp. 823-830, Barcelona, Spain, Oct. 2012.
- [22] S. H. Kang and J. H. Kim, "QoS-aware path selection for multi-homed mobile terminals in heterogeneous wireless networks," in *Proc. CCNC 2010*, pp. 1-2, Las Vegas, Nevada, USA, Jan. 2010.
- [23] Cisco, *Global Internet Speed Test (GIST) for iPhone, BlackBerry and Android*, Retrieved April 13, 2016, from <http://gistdata.ciscovni.com/>
- [24] 3GPP TS 23.203 V12.6.0, *3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Policy and charging control architecture*(Release 12), Sept. 2014.

**천 헤 립 (Hye-Rim Cheon)**



2006년 : 아주대학교 전자공학부 졸업  
 2011년~현재 : 아주대학교 전자공학과 석박사통합과정  
 <관심분야> 무선인터넷 QoS, MAC 프로토콜, Full duplex, OFDMA, 소형셀, LIPA/SIPTO 등

**이 승 규 (Seung-Que Lee)**



1988년 2월 : 중앙대학교 전산학 이학사  
 1990년 2월 : 중앙대학교 전산학 이학석사  
 2007년 8월 : 고려대학교 전산학 이학박사  
 1990년 2월~현재 : 한국전자통신연구원 책임연구원

<관심분야> 5세대 이동통신 시스템, 소형셀 시스템, 서비스 플랫폼, 네트워크 기능 가상화, 모바일 엣지 컴퓨팅 등

**김 재 현 (Jae-Hyun Kim)**



1987년~1996년 : 한양대학교 전산과 학사 및 석/박사  
 1997년~1998년 : 미국 UCLA 전기전자과 박사 후 연수  
 1998년~2003년 : Bell Labs, Performance Modeling and QoS Management Group 연구원

2003년~현재 : 아주대학교 전자공학과 정교수  
 <관심분야> 무선인터넷 QoS, MAC 프로토콜, IEEE 802.11/15, B4G/5G, 국방 기술네트워크, 위성통신 및 시스템 설계 등