Cooperative Content Caching and Distribution in Dense Networks

Asif Kabir[1,2]*
[1]College of Communication Engineering, Chongqing University, Chongqing, 400044, P.R China
[2]University of Kotli Azad Jammu & Kashmir
[e-mail: asifkabir, yunjian@cqu.edu.cn]
*Corresponding author : Asif Kabir

Received March 22, 2018; revised May 6, 2018; accepted May 31, 2018; published November 30, 2018

Abstract

Mobile applications and social networks tend to enhance the need for high-quality content access. To address the rapid growing demand for data services in mobile networks, it is necessary to develop efficient content caching and distribution techniques, aiming at significantly reduction of redundant content transmission and thus improve content delivery efficiency. In this article, we develop optimal cooperative content cache and distribution policy, where a geographical cluster model is designed for content retrieval across the collaborative small cell base stations (SBSs) and replacement of cache framework. Furthermore, we divide the SBS storage space into two equal parts: the first is local, the other is global content cache. We propose an algorithm to minimize the content caching delay, transmission cost and backhaul bottleneck at the edge of networks. Simulation results indicates that the proposed neighbor SBSs cooperative caching scheme brings a substantial improvement regarding content availability and cache storage capacity at the edge of networks in comparison with the current conventional cache placement approaches.

Keywords: cooperative caching, small cells, zone base cluster, 5G.

1. Introduction

While the field of mobile communication is evolving, significant progress has been achieved in this field over the past few years although facing many challenges. A demand for high capacity mobile networks is felt because of the rapid growth of mobile data traffic which introduces significant concern for the researchers. Global mobile data traffic is expected to grow seven-fold from 2016 to 2021. During these years, the mobile data traffic will grow at an annual growth rate of 47 percent, which will surpass 49 exabyte's per month by 2021. In addition, the video on demand streaming will generate 78 percent of the mobile data traffic by 2021 [1]. To deal with such bulk of data traffic in the cellular network, three main factors namely: increase in wireless access points, wireless link performance improvements and radio spectrum expansion are proposed [2]. To meet the emerging multimedia stringent quality of services (QoS) in the future fifth generation (5G) cellular network leads to the introduction of a new decentralized architecture, based on the concepts of dense SBSs deployment, like Femtocells, Picocells, or Microcells. SBSs are dedicated to deploy low operating costs with high QoS [3].

However, to gain these benefits in 5G cellular networks, we must determine distinct technical challenges, especially in the areas of load balancing, data storage, data transference and limited backhaul capacity. By integrating offloading techniques and data storage units at the edge of the network (mobile devices and base stations), a state of the art cache enables SBS architecture has been proposed to overcome the burden and bottleneck on backhaul networks [4]. To enhance data capacity at the edge, the content distributed network first introduces caching during the off-peak periods resulting in balancing network traffic at strategic nodes. However, the number of BSs increases with the explosion of network traffic and backhaul of radio access network (RAN), which will also become congested, further motivating the use of caching, stores reusable content at the BS.

Along these lines, through SBSs, users requested contents that are locally sufficient without any bottleneck on the backhaul network or by employing the MBSs. The idea of using edge caching to support mobile users in a cellular wireless network has been established in [5]. In RAN, local caching at the edge of the network has emerged as a promising technique for enhancing the QoS of user equipment's (UEs). Cache capacity of local BS is a new type of resource besides time, frequency, and space. Moreover, cache reduces duplicate content transmission, in addition, it improves the energy efficiency (EE) and spectrum efficiency (SE).

To exploit the full advantage of edge caching, sophisticated caching placement strategies will be required. That acknowledges the limitation of SBSs storage capacity, user degree of attention and popularity of content which varies over time. Additionally, for a mobile network operator (MNO), the cache enabled cellular network has also triggered new challenges such as what, where and how to cache [6]. The number of hits and size at each SBS is much smaller than content delivery network or core network (CN). So, the internet reactive cache design is not operative for the caching of SBSs. Designing a proactive caching policy for each SBS independently may result in insufficient utilization of caches. One way to address this problem is to enable SBS to share the cached contents, that is, cooperative caching. If the requested content is not available in the cache of local SBS then retrieval request for contents from caches of neighboring cooperative SBSs will be done instead of the content provider. In this way, the transmission cost and latency can be reduced and overall cache hit probability will be improved.

To achieve spectral efficiency gain over the conventional caching scheme, a PHY-caching scheme for 5G wireless networks is proposed in [7]. The edge cache content placement, delivery and key differences between wireless and wired caching are discussed in [8]. The impact of backhaul delays on caching placement for a wireless network is studied in [9] where base stations via backhaul links are connected to the central controller.

Due to the openness of wireless channels, the coverage of SBSs often overlaps. It indicates that a user can get contents from multiple caches and hence the equivalent cache size seen from the user perspective has increased. Based on this observation, caching policies for adjacent SBSs can be jointly optimized to improve cache hit probability without data sharing over backhaul links, which is referred to as distributed cooperative caching that will improve the utilization of limited resources in 5G networks.

1.1. Related Work

Due to proliferation and uncontrolled installation of small cells, several issues occur in networks like; wastege of energy resources, and interference which implies that the coordination of small cells is needed. Numerous coordination schemes are proposed in previous studies both centralized and decentralized, focusing on the formation of clusters for interference mitigation, load balancing, knowledge distribution, emergency communication links to backhaul and so many others. Small cell cooperation, jointly optimizes the caching strategy, distributed caching and PHY layer cooperative transmission are studied in [10]. Clustering as an alternative scheme is proposed in [11], which focus on a small cell user having different preferences over different content types. Where the interaction of user association and caching at the wireless edge are considered. In particular, on the basis of user's content request similarity, they are clustered together in an associated small cell. A cluster-centric SCN with the combined design of cooperative caching and coordinated multipoint transmission technique is proposed in [12], used to deliver content to the user. A performance aware community-based VOD stream over VANETs is studied in [13]. Also, social aware mobile multimedia streaming services are discussed in [14] which focuses on peer to peer communication for the community. Similarly, the authors in [15] studied a user aware edge cache scheme which focus on user interaction on social networks. To achieve the spectral and energy efficiency, reference [16] proposed an intelligence and collaboration for D2D content delivery over ultra-dense networks. In reference [17], the authors proposed an energy efficient content delivery system via device to device communication for smart cities, where they explore the relationship among the coding, storage, and transmission. Moreover, devices used the only local information to make decisions and implement its scheme individually.

Decentralized distributed cache placement and limited collaboration are proposed in reference [18]. Clustering and replication in hierarchical cache network has been studied in [19], on the basis of content popularity which consists of both core and edge networks. Distributed edge caching and file redundancy ratio that minimizes total transmission cost in the system is proposed in [20]. The authors in [21] enlighten video caching policy in RAN and have proposed caching policies based on the user preference profile. In [22], the authors have developed cache scheme that performs globally in an adaptive way and periodically estimates cluster numbers.

Cooperative caching such as in-network joint routing and caching with the goal of minimizing the content access delay were studied in [23]. Reference [24] and [25] focus on the joint cache placement and delay minimization, for a given anticipated content demand, it determines which content should be placed in each cache, so as to reduce the average content

delivery delay for all requests. A joint routing and caching that maximize locally content requests by the deployed SBSs were proposed in [26].

Various studies about clustering and cooperative caching have been carried out. These studies mainly focus on the popularity of the contents within proximity to the users. However, most of the existing works assume similar popularity of the content, largest content diversity (LDC) of content and user different interest on the content. Furthermore, less attention is given to cooperative caching and distributed cache placement within the zone. Regarding the issue of redundant content transmission, designing a caching policy is a challenging and a well-known NP-hard problem. Motivated by the above issue, we focus on how to organize the efficient placement of edge cache in dense networks in which the clustering is performed locally to improve QoS and minimize transmission cost. Furthermore, we examine how cache placement effects hit ratio and load perceived.

We introduce a scheme that fully exploits the gain of cache, reduces users perceived latency, alleviates the backhaul burden, increases SBS cache capacity at the edge of the networks and improves overall networks performance. Zonal based clustering is proposed, whereas satisfying users content requirement locally according to the user's preferences. So, a collaborative scheme is developed that brings the popular and local content according to user's demand. The motivation behind such a system is to obtain contents from a cache that is near to users; which is likely to experience a shorter delay from one that is farther because a smaller number of hops have to be traversed. Caching contents and serving users from neighboring base stations alleviate both the traffic load and the latency in the network for the end users. Our contribution can be summarized as follows:

1.2. Contribution

- Our proposed caching scheme considers the trade-off between the diversity and redundancy cache content that minimizes the transmission delay and backhaul congestion. Cache replacement strategy has been investigated, that maximizes content requests served by the deployment of local SBS. Additionally, a zone-based joint clustering and cooperative caching approach are proposed.
- We intend to incorporate policies among the base station within a region and formulate the corresponding optimization problem for devising the globally optimal caching policy using SBS cooperation and cache storage space partitioning technique based on local and global content popularity distribution.
- We have evaluated the proposed system model theoretically, systematic simulations are carried out to achieve minimum transmission cost, and performance of the proposed algorithm. Furthermore, the efficient utilization of available cache resources and balancing the distribution of content among SBS at the edge of the network are carried out.

The rest of the paper is organized as follows: Section II presents proposed system model and problem formulation. Section III presents proposed zone-base and cache placement algorithms whereas Section IV shows simulation results and Section V concludes the paper.

2. Model Description and Problem Formulation

2.1 System Model

In this section, we examine the system illustrated in **Fig. 1**, a zone-based cooperative SBSs has been considered. Here we study a wireless network with the single macro base station (MBS) and a set of SBSs is denoted by $B=\{b_1,...,b_i,b_j,...,b_n\}$ which serves the users $U=\{u_1,u_2,...,u_k\}$. Each SBS is equipped with a fixed data storage capacity and has a library of N files that is denoted by $C=\{f_1,f_2,...,f_n\}$, $f_s \geq 0$. We will use the terms file and content interchangeable throughout this paper. Assume that user u_k served by the small base station b_i, the SBSs are connected with MBS and neighbor SBS via backhaul (β) and virtual link respectively. The content popularity of a particular content can be varied depending upon the cell location. So, there is a need to cluster the cells according to their zone. In zone, users request a content, each user is served by designated SBS depending upon the content location and transmission scheme. Within a cluster, there is a cooperative transmission. Each SBS storage space is divided into two parts; the first part is the top popular content and second part is on the basis of local user's requests. In other words, each SBS should serve a group of users in the same location according to their requirement with optimizing cache placement strategy.

2.2 Content Probability Distribution

In the proposed scenario, content popularity is important. To exploit the full potential of wireless edge cache, popularity prediction and user preferences are critical [8]. The popularity of content is first zone based and then particular SBS are calculated in proposed scenario. Their popularity determines request rates for content at each SBS. We approximate content popularity

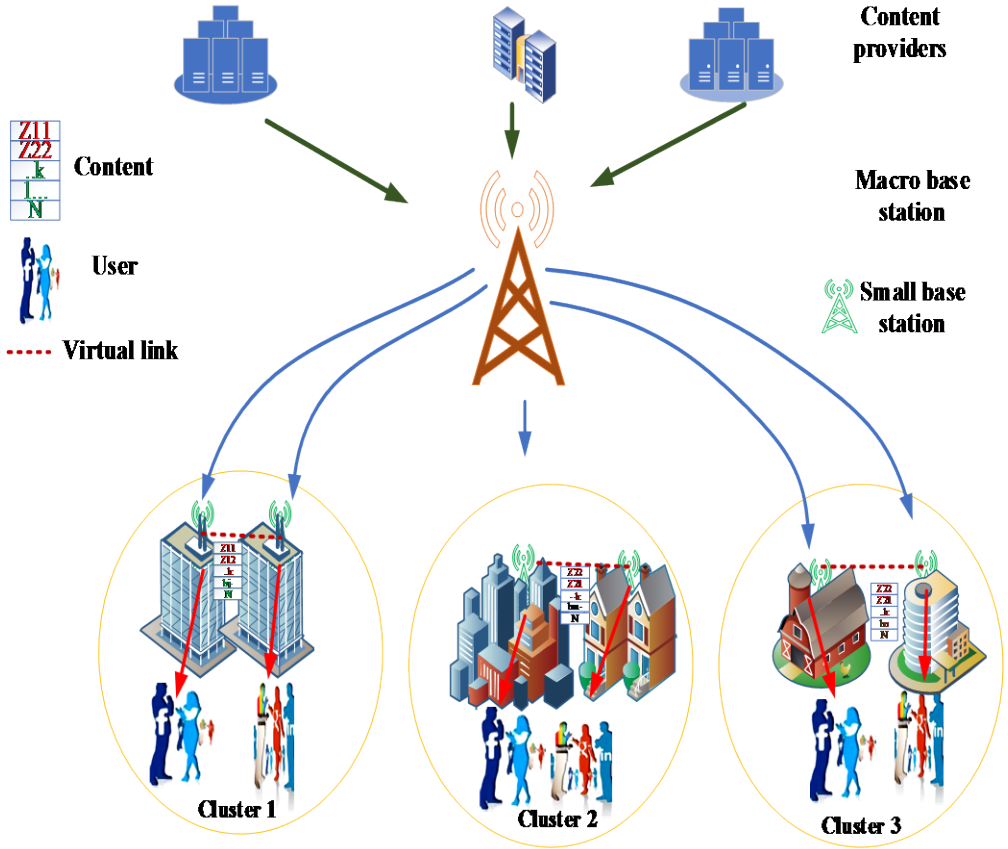

Fig. 1. Illustration of considered network

by a Zipf law distribution [27]. In general, due to local user interests, each content popularity may differ from place to place, where request pattern aggregates are different across various locations, it is captured through a localized request generation model. All over the globe, some popular content might only watch in a confined region, authors in [28] investigated the geographic popularity of videos around the world which concluded that different contents will be requested in different regions.

There are various regions and all these regions are characterized by the same value for the Zipf distribution exponent which captures the local popularity of content. The popularity of content may be different in each location and being assumed that different contents are cached in each of the SBS.

Assume that $p_{i,f}$ denotes the popularity of content f at b_i, and the files are sorted according to popularity, considering the overall network, total popularity for each content denoted as $p_f = \sum p_{i,f}$ obeys a Zipf distribution, we have

$$p_f = \frac{k^{-r}}{\sum_{f=1}^{n} k^{-r}} \tag{1}$$

Here, exponent r denotes the rank of popular content and $r \geq 0$, characterizing the skewness of popularity distribution. The global multimedia platform monitored that caching entity global content popularity is not even replicated by the local content popularity [28]. Therefore, for proactive cache content placement, caching entities should learn local content popularity. Secondly, different users have different content preferences. As a result, the popularity of the local content can be changed according to different preferences of mobile users in the vicinity of a caching entity. Thus, according to diversity in local user content popularity, the proactive cache content placement should be taken into consideration that maximizes the number of cache hits.

2.3 Problem Formulation

The main component of delay is backhaul and wireless transmission delay. Assume that content C consists of a file size f_s, the user u requests a certain file i.e. $c \in C$ from the serving SBS b. The approximate wireless transmission delay between user and SBS to retrieve the requested content is calculated as follows:

$$d_{u,c}^{b} = \frac{f_s}{dl} \tag{2}$$

dl is the download link transmission bandwidth from the base station b for the user u and dl is given as;

$$dl = W_{u,b} \log_2(1 + \Gamma_{u,b}^{Dl}) \tag{3}$$

Where bandwidth allocation to user u is represented by $W_{u,b}$. The total bandwidth of the base station b is divided to the number of users' requests, and $\Gamma_{u,b}^{Dl}$ is the downlink signal to interference plus noise ratio (SINR). We further define the total service delay to retrieve requested content that can be expressed as follows:

$$\sum_{u\in U}\sum_{c\in C} d_{u,c}^{b} \tag{4}$$

Another component of delay is backhaul delay. The backhaul delay of SBS is related to the average link distance, the average traffic load and the number of SBSs are connected to the single gateway. It can be modeled to be an exponentially distributed random variable with a mean value of CBH [29]. Let variable xf be a binary variable indicating whether file f_n is cached at SBS b. The backhaul delay is evaluated as

$$CBH = (1 - xf)d_{u,c}^{b} \tag{5}$$

Primary ambition is to find an optimal cache strategy that minimizes the total delay. Each time when an end user makes a request for a file, the associated SBS can transmit file to the user, if file is in its cache. However, if a requested file is not available in the cache of local serving SBS, then BS will fetch the file from the content server via backhaul link and transmit it to the user. This process will increase the delay to retrieve the content from remote servers. To alleviate or minimize this delay, an alternative method is to make a group of SBSs that allows SBS to retrieve the file from the cache of neighboring inter-linked SBS in the same network domain hence reducing delay and the backhaul conjunction.

Many SBSs can cooperate to form a group called coalitions and jointly decide how to transmit files to users. SBSs coalitions increase the hit ratio and exploit cache diversification. Based on file popularity distribution and caching popularity at each SBS, it is possible to determine the probability $\mathrm{p}_{u,i,f}$ that a file f requested by user u is available at SBS location i [30]. On the basis of specific region, a cluster and cooperative cache are made. Suppose Z is a vector that can represent the zone, so, $b_i \in Z$ which is serving the user. Therefore, the total delay can be calculated as

$$D = d_{u,c}^{b} + CBh \tag{6}$$

Let $xf = 1$ indicate that file is stored in the buffer of base station b_i otherwise $xf = 0$. So, cache strategy can be expressed as matrix X consists of variables $\{xf\}(f_n \in C), \mathrm{b}_i \in B$. Cache placement can overcome the overall delay of all the users to access the content.

Let $p_{f,u}$ describe the requested probability for the user u_k to make a request for a file f_n from the serving base station b_i. The user's file preferences are normalized such that $\sum_{f=f_1}^{f_n} p_{f,u} = 1$. BS decides how to transmit file based on the caching strategy, i.e. $\sum_{u=1}^{u_k}\sum_{b=1}^{b_n}\sum_{f=1}^{f_n} p_{f,u}.D.\mathrm{X}$. Transmit file to the user directly, if the file is cached, otherwise, the connected SBS decided how to fulfill the user requested requirement. Generally, the delivery latencies are distinctive for different hit scenarios and routing path. The average delay of small cell can be calculated as

$$Avg.D = \frac{1}{U}\sum_{u=1}^{u_k}\sum_{b=1}^{b_n}\sum_{f=1}^{f_n} p_{f,u}.D \tag{7}$$

Thus, the cache delay optimization problem can be formulated as follows.

$$\min \sum_{b\in B}\sum_{b\in Z}\sum_{f\in C} Avg.D \tag{8}$$

Subject to

$$0 \le p_f \le 1 \tag{8a}$$

$$\forall f_n \in C, \forall u_k \in b_i, Z = \forall b_i \in B \tag{8b}$$

$$\sum f_s \le C \tag{8c}$$

Where (8a) is the probability constraint, (8b) is an association of content, SBSs and users, and (8c) is the storage limitation at SBSs.

Content requested by the users that are not available in the cache of its serving SBS, in traditional cellular networks are downloaded from a content provider via the backhaul.

Apparently, this operation entails various challenges, such as delay, data rate constraints, which are generally processed by some packet gateways whose estimation is difficult, each one introducing a transmission delay. Thus, a caching approach which maximizes the hit ratio cannot ensure high quality of services delivery and it can compromise the QoS of the uncached content request. To address aforementioned dilemma, an alternative is to allow the SBSs to get content from the neighbor SBSs which are significantly affected by the backhaul traffic congestion. A lot of existing works have been proposed the cooperation among SBSs when designing caching policies. In [20], authors investigate the cooperation among caches in the RAN and obtain an optimal redundancy ratio of content cached in each base station. A collaborative framework within same network domain has studied in [31] which yields a significant gain in term of the content availability. Similarly, to decrease the transmission delay and improve the content requests served locally a joint routing and caching is proposed in [32].

Furthermore, data storage at edge based on cache policy and file popularity distribution at each base station. It is possible to determine the probability of the file f requested by the user at a specific location. Moreover, the following factors also affect the performance, therefore, we address in our scheme:

- If user requested file is available at the local cache of SBS $b_i, p_{f u_k}^{b_i}\{f \in b_i\}$, the cost to acquire f is equal to zero.
- So, the probability that a user u_k requests an un-cached file is $f \notin b_i$, consequently in that case $p_f = 1 - p_{f u_k}^{b_i}$.
- There is another situation where we have collaborating cache scenario; the file may occur within a cluster in the adjacent SBS. Suppose it is in b_j. So, the probability at that case can be $p_{f u_k}^{b_i}\{f \in$ cache b_i, f user requested, $\{b_i, b_j\} \in Z$ cluster$\}$.
- In this case, file travels one hop. In zone base cluster cache there is the third scenario, if user requested content is unavailable in the cluster, then it can be found in MBS, the probability distribution of user request file in this case is $p_{f u_k}^{MB} = 1 - p_{f u_k}^{b_i} - p_{f u_k}^{b_j}$.

Delay Probability

$$\begin{cases} 0: & \text{local SBS prob.} p_{f u_k}^{b_i} \\ 1: & \text{cooperative prob.}(1\text{-}p_{f,u_k}^{b_j}) \\ \text{k}, 1 \le k \le \tau: & \text{MB prob.}(1-(1-p_{f u_k}^{b_i} - p_{f u_k}^{b_j})) \\ \tau_i: & \text{gobal access prob.(1-MB) Prob.} \end{cases} \tag{9}$$

(10)
(11)
(12)

Finally, when the file is unavailable in both designated SBS and cluster, it is retrieved from the content provider. The following summary (9-12) respectively expresses the probability and latency of a content that is requested by the user, served by the local cache and cooperative SBS or unavailable. A file should be obtained from the local cache, neighboring SBS cache, MBS cache, or remote server.

2.4 Small Cell Clustering

Cluster-based on content popularity faces many challenges as the future popularity of content cache is not available at the caching decision time. The popularity of content changes with time, hence using estimated caching is another challenge. In general, the descriptions are already enlightened before. However, we need a specific description of Algorithm 1 in **Fig. 2.** We consider a general network architecture where a set of users are together in the same geographical region. It is considered that there is one MBS and a set of the SBS as shown in **Fig. 3** in each zone. Clustering Algorithm1 describes the proposed clustering method that groups the users on the basis of their zone. However, by grouping, more popular content will be cached closer to the user. Due to cooperation fewer number of requests being served by the MBS and backhaul networks, consequentially minimizing the total service delays.

Algorithm 1. Proposed cluster algorithm

```
Begin
1.  Make cluster size 100 ×100 m
2.  Specify the total number of SBSs (n); // n=27
3.  Fix the total rounds (r); //r=3000
4.  e_init(s)=e0; // s=1,2,3 …, n
Initialization:
5.  do {// r rounds repetition
6.  r=rand [0,1] ;//SBS randomly deployment
7.  XY= [xm/3, ym/3]; // Split X & Y axis in 9 equal parts.
8.  Clusters (Z) = 9; //Z= [Z1:Z9] (cluster/zone)
9.  for i=1: Z
10.      plot_MBS=((xZj)/2, (yZj)/2); //Plotting main base stations.
11.   for j=1: j
12.      MBS is the centralized reference point in each zone &   j=1, 2…9;
13.         deploy SBS nodes r = fix (0:2);
14.  end for
15.      end for
16. } // one round is completed
    End
```

Fig. 2. Cluster Algorithm

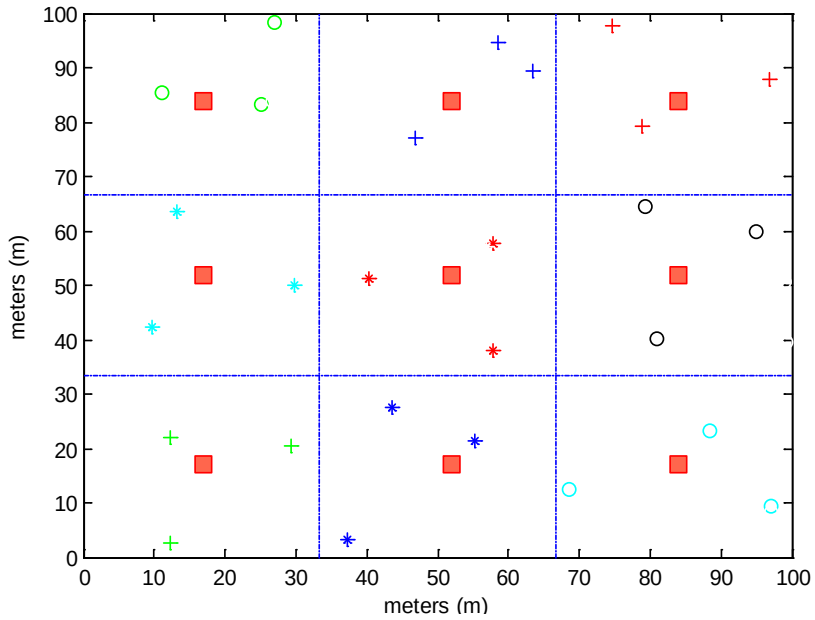

Fig. 3. Zone base cluster

2.5 Distributed Cache Placement and Cache Partition

Our next step is to propose a decentralized caching scheme that reduces the delay incurred to deliver the requested file to the user. Here, each SBS is interested in following a caching policy that minimizes the service delay for its associated group. The proposed caching scheme is decentralized and can perform in each SBS using local caching strategies. In particular, the cache replacement policy depends upon the number of hits and content with lower access probability should be chosen for replacement. The access probability p_f of each content f_i is initially set to be zero. Using un-coded cache scheme, T_i is set to the content requested time and then p_f is updated as content is requested by the host BS. Where T_c is the current time, T_i is the last access time and α is a constant factor to weight the importance of the most recent access. Consider that there is C file library on each SBS, $S = \{s_1, s_2, ..., s_z\}$ file size and assume that all files have the same size. On SBS total cache size can be expressed as M bytes, the files storage procedure can continue until the storage capacity of SBSs is exhausted.

$$M \leq \sum_{f \in C} S_z \tag{13}$$

Usually, placement of file is represented by x_p so, the size of file is multiplied by x_p then total cache constraint is given as:

$$\sum_{f=1}^{F} S_z x_p f \leq M \tag{14}$$

To minimize the average download delay within cluster, we assume that it depends on the number of the users' request. The users requested content query latency is given as follows.

$$\sum_{b\in z}\sum_{b\in B}\sum_{c\in C}\lambda(pf + pf_{u_k}^{b_i} + pf_{u_k}^{b_j} + pf_{u_k}^{b_{MB}}) \tag{15}$$

The latency function can be rewritten as;

$$\sum_{b\in z}\sum_{b\in B}\sum_{c\in C}\lambda(pf_{u_k}^{b_i} + (1 - pf_{u_k}^{b_j}) + (1 - (pf_{u_k}^{b_i} - pf_{u_k}^{b_j} pf_{u_k}^{b_i}) + (1 - (1 - (pf_{u_k}^{b_i} - pf_{u_k}^{b_j} pf_{u_k}^{b_i})) \tag{16}$$

In a cluster, base stations are connected to each other so that, they can share their cached files. They are also connected to the MBS which is connected to the core network via the backhaul link. In proposed distributed cache replacement strategy, each SBS obtains the cache information, manages it's in a distributed way, and no centralized controller is needed for caching. The most popular files in the network are cached in SBS and requests of these files are directly sent to the user from the cache of each BS. If the user of b_i requests for a file which is not available in the cache of b_i then SBS b_i requests to the neighbor cooperative SBS b_j and then serve to the user. If the requested files are not in a local cluster, then it requests to the MBS or fetch from the core network. In this way, it reduces the transmission cost and backhaul bottleneck.

Caching the popular content at the edge of the network can offload the backhaul traffic and increase the internal traffic. The optimal solution is to logically divide the content storage space into two equal parts as mentioned in **Fig. 4(b)**, where top files are redundant files and other files corresponding to their local base station. In this way, more diverse files are stored at the edge of wireless networks and reduces transmission between SBS and backhaul.

$$S_z b_i = \sum_{b\in z}\sum_{b\in B}\sum_{c\in C} pf + \sum_{b\in z}\sum_{b\in B}\sum_{c\in C} pf_{u_k}^{b_i} \tag{17}$$

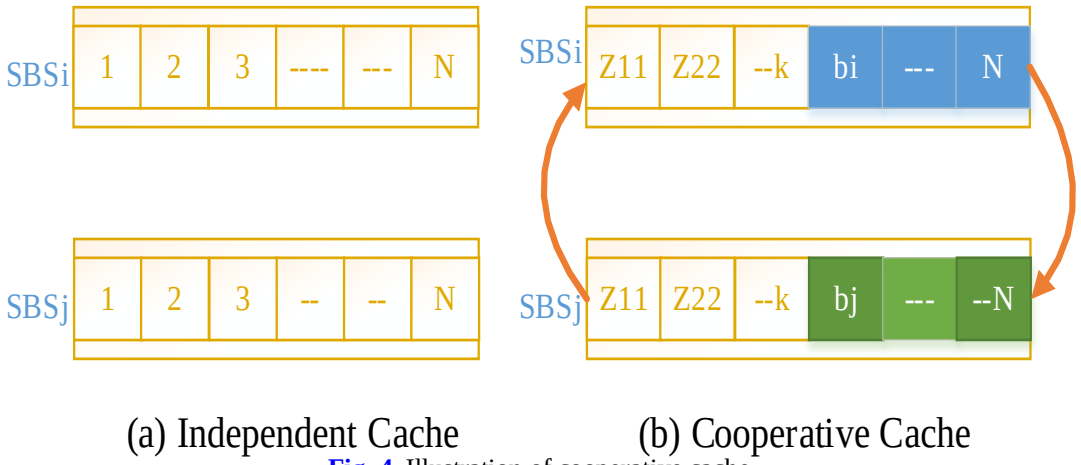

Fig. 4. Illustration of cooperative cache

The benefits of cooperative caching can be easily understood from **Fig. 4**, which improves the quality and existence of cache in a specific region. Hence, for content placement, intelligent algorithms are required. Caching content placement has many challenges concerning content popularity such as determining whether an optimal cache placement primarily depends on content popularity distribution or not. When and which content in future will be requested is unclear.

Algorithm 2 shows the overall delay minimization using cooperative exchange cache content technique which will reduce the cost and redundancy of cache at the edge of the networks. The above optimum formulation is executed after clustering initiation presented in the algorithm 1. The idea is to minimize the delay for cache within the cluster. So, the total delay minimization can be expressed as

$$T.D_i = p_{f_{uk}}^{bi} \times 0 + [1-(1-p_{f_{uk}}^{bj})] \times 1 + \sum_{k=1}^{\tau-k}(1-p_{f_{uk}}^{bj}) \\ [1-(1-p_{f_{uk}}^{bj})] \times k + (1-p_{f_{uk}}^{bj}) p_{f_{uk}}^{bj} \times \tau_i \quad (18)$$

To simplify this model, we assume that P is the probability that a data is cached. The average delay can be expressed as,

$$D_i.avg = 1-(1-p) + \sum_{k=1}^{\tau}(1-p) \\ [1-(1-p) \times k + (1-p) p \times \tau_i] \quad (19)$$

This equation is an approximation of $Di.avg$ since, in practice, probability (p) for the different local SBSs may be different. If there is no cooperation in SBSs within the cluster, the average delay becomes:

$$D_i.avg.cd = \sum_{k=1}^{\tau}(1-p) p \times \tau \quad (20)$$

Latency is given as follow:

$$D_i.Q = p_{f_{uk}}^{bi} \times q_L + [1-(1-p_{f_{uk}}^{b_j})] \times [q_{Cluster} + q_{se}] + \\ \sum_{k=1}^{\tau-k}(1-p_{f_{uk}}^{b_j})[1-(1-p_{f_{uk}}^{b_j})] \times [q_{Cluster} + q_{se}] + (1-p_{f_{uk}}^{b_j}) p_{f_{uk}}^{b_j} \times \tau_i \quad (21)$$

$$D_i.avg = p \times q_L + [1-(1-p)] \times [q_{Cluster} + q_{se}] + \sum_{k=1}^{\tau-k}(1-p) \\ [1-(1-p)] \times [q_{Cluster} + q_{se}] + (1-p) p \times \tau_i \quad (22)$$

3. Proposed Algorithm and Cache Placement

In the proposed algorithm, cooperation cache can be expressed as minimizing the delay, redundant content, increased storage capabilities at edge and improved the backhaul capacity at each SBS. In order to solve the problem presented in (8), we divide the whole communication network area into multiple zones (9 zones). Each zone represents a cluster. This zonal division helps to manage the network traffic efficiently and reducing the network load. The base stations are deployed in such a way that each zone contains three SBSs and one MBS. The SBSs communicate with each other and also send and retrieve the information from MBS. We propose algorithm 2 which works on behalf of algorithm1.

Algorithm 2. Propose cooperation and cache placement

Notations:

MB-Macro base station $b_1, b_2, ..., b_i$ - SBS

$u_1, ..., u_k$ - Users $f_1, ..., f_n$ - Content

N - library of files Z - Cluster

S_z - Fixed data storage capacity b_i connect to b_j – neighbor

1. **BEGIN**
2. **PHASE I:** Cluster Phase (Algorithm 1)
3. **PHASE II:** Ranking and cache placement
4. $p_f(z)$ //Calculate the cluster base cache popularity of the content
5. $p_{fu_k}^{b_i}$, $p_{fu_k}^{bj}$ //Calculate the request popularity on cell base.
6. if $R(u_k) > 0$
7. for each $c \in C$, $b_i \in Z$ sorted according to pf set, S_{Zi}
8. end if
9. end for
10. $S_z b_i = C_i + (c - s_i)$ //logical division
11. **PHASE III:** Cooperative Cache Management
12. for $u_k \in b_i$
13. if file $\leftarrow b_i$
14. else if file $\neq b_i$ then $b_j \in Z$ //local cluster finds with in cluster/neighbor's SBS
15. else search in MBS
16. else from Backhaul
17. end if
18. end for
19. PHASE IV: Cache replacement phase
20. If f_n, p_f file is low ∩ Recent LLR that has to be evicted form cache.
21. if $\sum_{f=1}^{F} S_z x_p f >> M$
22. if $\sum p_f + p_{fu_k}^{b_i} - \sum p(\text{LLR}) > 0$
23. S_z =Cache +C-{LLR}
24. Update Cache=cache + file newly requested
25. end if
26. end if
27. end if

Fig. 5. Delay minimization and cache placement algorithm

Algorithm 2 has four main phases, first, to generate a cluster according to the regional basis and then to find the probability of content, first globally then locally according to the SBS within cluster. The storage space on SBS can be divided logically according to the popularity of content as mentioned in the above "distribution cache placement and cache partition" section. After that within the region, each SBS sorts its neighbor SBSs, where they establish cooperation relation for exchange data via a virtual link. In the cooperative cache management phase, a user requests data from serving SBS, if the requested content found then serving SBS transfer the requested content directly. Otherwise serving SBS has to request the file to neighbor cooperative SBSs within the cluster. If the requested content is not found within a zone, then serving SBS retrieves content from MBS or fetches it from the backhaul. After that, the cache replacement is performed which depends upon content probability distribution and Least Likely Requested (LLR).

Proposition for Algorithm 2. The worst-case performance $_{wz}$ of algorithm 2 with respect to the optimal solution w^* of the problem in (8) is such that $\frac{w^*}{z} \leq wz$ at each base station maximize iteration and can be represented with $_z$ in each SBS.

Proof. Please see Appendix A

Proposition 2. The complexity of algorithm 2 is a polynomial and scales in order of O(z.C.uk)

Proof. Please see Appendix B

4. Performance Evaluation and Discussion

In this section, numerical results of the proposed algorithms are presented. We compare the performance of proposed scheme and evaluate it using MATLAB. We compare the performance of caching policies, as well as validate the effectiveness of the proposed technique with conventional cache strategies. **Table 1** shows the parameters used for our simulation results. To ensure the simulation, we restrict the total number of content and storage space at each SBS, for the content selection following Zipf popularity distribution. Users will request contents randomly and popularity of contents first generate globally within cluster then according to a specific SBS popularity is generated.

Table 1. Simulation parameters used in numerical results

Parameters	Value
Number of MBS	01
Number of SBSs in cluster	03
System bandwidth	10 MHz
Zipf's law exponent	0.8
Cache size of each MBS	500GB
Cache size of each SBS	250GB
Small cell radius	30
Number of users	20
Total number of area in a local zone	100x100m
Content size	1MB
MBS transmission power	46 dBm
SBS transmission power	30 dBm

In the proposed scheme, we have considered a region-based clustering. We assume three-layer content caches, if the required content directory access by the host SBS, then there is zero latency, within a cluster, there is minimum latency and we set to the average latency of the user request content. Whereas t_0 denotes the average latency of cooperative cache within cluster and t_1 denotes the average latency of serving request from MBS. Finally, t_2 represents the average latency of fetching contents from the original CDN server, where all contents are stored. In simulation, we assume that $t \in \{20,30,60\}$ ms depending upon where the content exists. We assume that each SBS has 10 MB/s backhaul bandwidth and fixed content storage.

In proposed cluster cache, first, we make a cluster on the basis of region. The total cache capacity in the cluster is considered as an entity and cache placement is performed in distributed ways. We choose a region of 100 x 100m. After that, we divide the whole area into two equal parts and find a central point of the cluster that is on (50,50). As mentioned in **Fig. 3**. The zone is divided in such a way that the traffic is efficiently managed. Now dividing the total deployment area into equally distributed zones to balance network traffic and reduce traffic load. The number of users varies in each cluster in each subdivided zones (clusters). A subzone is considered as one cluster. A number of users vary in each zone and content popularity also varies in each area. So, in each small subzone, there are social similarities. Request rates for content items at each SBS are determined randomly by their popularity. In general, the popularity of content is estimated using Zipf law distribution, it varies from place to place, and content to content.

In each region, specific SBS content ranking of items within the region may be different. The proposed scheme will improve the cache size at the edge of wireless networks as the storage capacity is limited. Therefore, there is a need for a distributed cache replacement policy that minimizes the average transmission delay. We find the popularity of content as cluster base and cell base. If the requested content exists in local SBS then latency of access the content and delay time also goes to zero.

Next, we study the effectiveness of the proposed cache policy improving cache hit rate, delay and increase storage capacity. We explore the impact of varying bandwidth between cache and average content delivery delay. We compare the proposed cache policy with random cache and MPV. In [21] the author investigated the effectiveness of proactive caching most popular content(videos) (MPV) policy, which works on the popularity distribution nationwide.

Fig. 6 shows the performance and gain of various cache policies regarding cache hit ratio when comparing the proposed technique with popular cache replacement schemes. It is evident that proposed cache scheme is significantly better than conventional cache policies for all cache size. Proposed cache scheme gets significant results as compared to other popular cache schemes. It is noticed that the proposed cache scheme hit popularity is high due to the fact that it is not an efficient approach that places a set of all popular files in all SBSs. In the proposed method, the hit probability mainly increases because most popular and different files are cached in all SBSs in a cluster within a region.

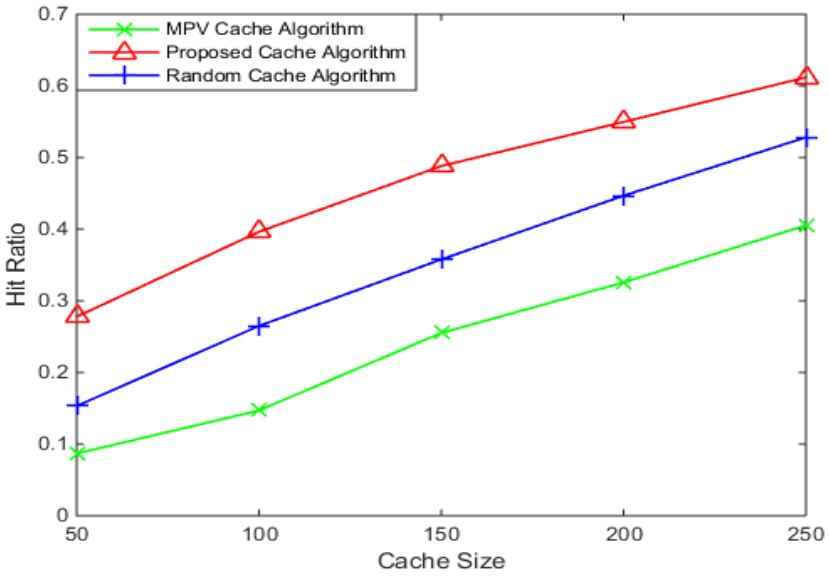

Fig. 6. Hit probability with different cache sizes

Increasing SBS density brings these files closer to the users, hence, improving hit probability. Obviously, it is a practical approach that can increase cache size and also system performance. In **Fig. 7**, while comparing with other cache schemes, we can see that the proposed scheme performs better than MPV and random cache. MPV works better globally and it cannot determine local cache, so its performance is much diverted. For example, when the cache size is 250 GB, the proposed cache achieves hit ratio of 0.66 percent while random and MPV policies achieves cache hit ratios of 0.5 and 3.5 respectively. When a cache hit ratio decreases, the backhaul bandwidth requirement increases. The goal of the proposed scheme is scheduling and supporting as much as possible. Collaborating cache at edge minimizes the delay. In the proposed scheme, most user requests are served within a local cluster. User requests to be connected BS, if the user request is not found, then search within the cluster, then macro and finally backhaul. If there is no local cache at the edge, the backhaul bandwidth is needed to bring all data from backhaul.

However, this simple approach would deprive the client of lower initial delay and save the backhaul bandwidth. **Fig. 7**, shows the delay saving ratio in the contents of cache size. As we compare with MPV, it demonstrates that proposed algorithm can reduce the backhaul delay about 45%. Also proposed algorithm compared with random cache algorithm, the proposed algorithm accesses 250 GB data, it save average download time about 10 %. It is clearly depicted from the above results that the proposed zone base cluster approach is superior to random and MPV.

We analyze the impact content delivery time and cache size in **Fig. 8**. In all algorithms, as expected increasing cache sizes, reduces transmission delay, all requests are satisfied locally. When cache size is 50 GB, then there is up to 44% difference, since later when cache size goes up, it can reduce likewise, on cache size 100 GB it goes down up to 20 %. Though the randomized strategy is simplest to implement, randomized cache strategy uses the random decisions to find the object and replace it. This strategy reduces the complexity of the process without sacrificing the quality. Its performance is worst due to its instability.

The proposed cache replacement strategy has a greater performance gain when the scale of

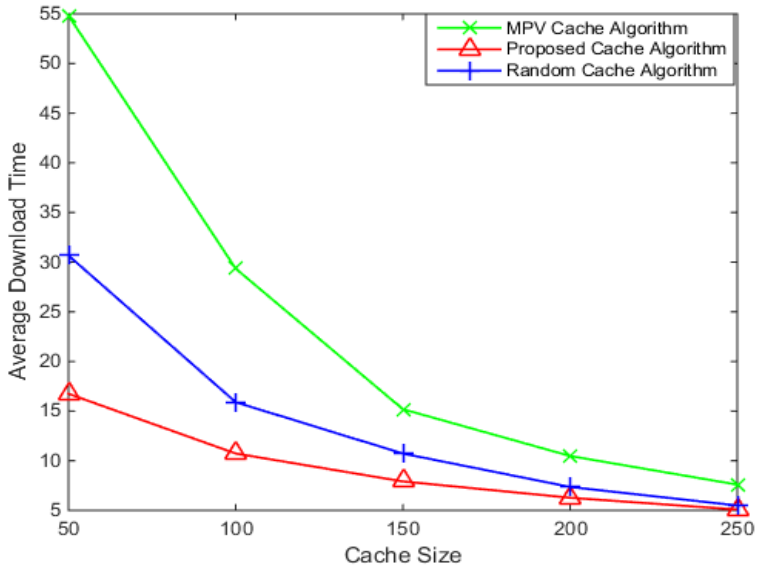

Fig. 7. Delay saving ratio with different cache sizes

the cellular system is significant. Since the caching information of other BSs are more important for efficient caching in that case. From the performance comparison in **Fig. 6, 7** and **8**, we can conclude that the proposed caching replacement strategy works much better than the conventional strategies, especially in large networks.

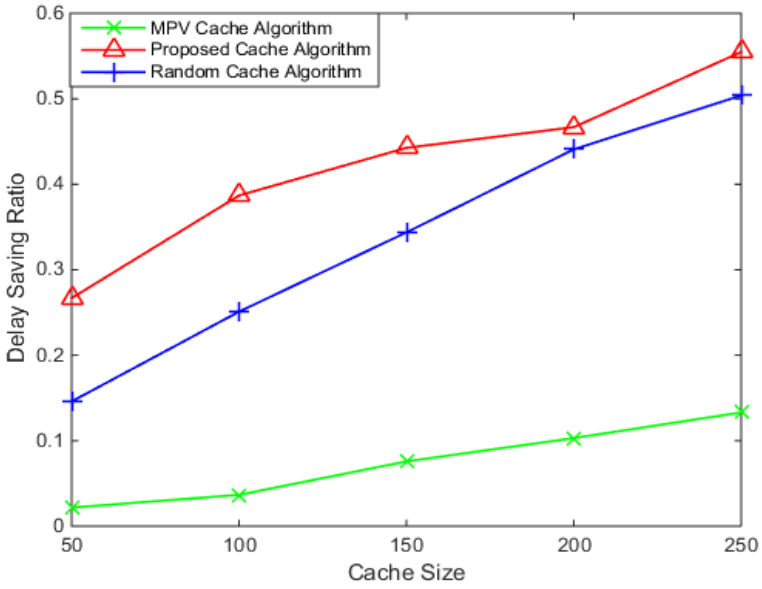

Fig. 8. Average delivery delay

5. Conclusion

Caching at the edge of wireless networks has been widely adopted to reduce the content delivery delay, alleviate the backhaul traffic between BSs and core network. This work proposes zone based collaborative content cache and novel logical cache partition approach which accomplished cache diversity, reduce the content redundancy and transmission cost at the edge of cellular networks. We have studied the popularity of content in the network and exploits the correlation between user demands. Finally, simulation results show that the proposed cooperative caching among neighbor SBSs provides significant gains over the non-cooperative scenario, it also overcomes congested backhaul problem and yields a significant gain regarding file delivery. The proposed framework is further extended to distributed algorithms for content placement and routing among different operators.

Appendix A

Let us consider that $\breve{D}_i$ and $\breve{w}_s$ are obtained through our proposed algorithm. According to the proposed equation (8), all files $f_n \in C$ and SBS $\in$ Z that shares the required content according to popularity and location, so we can

$$wz = \sum_{b\in z}\sum_{b\in B}\sum_{f_n\in C}(\tau_i - C_1) \tag{23}$$

According to xf we have

$$wz = \sum_{b\in z}\sum_{b\in B}\sum_{fn\in C}\{(\tau_i - C_1).\min(1,\sum_{fn}^{c} xf)\} \tag{24}$$

Now let user consider that collaboration maximizes the delay saving. We have a logical cache, that is, $Ic = pf_g \cup pf_l$ according to the user request. To achieve the optimal saving, we use xf^* for the file which is in SBS or not for the worst case the cost we need to $\frac{w}{wz} \le z$

Where, $z = |b|$ which have maximum intercommunication to achieve the files which have not in their cache. By $\min(1,\sum_{x=1}^{N}\alpha_x \ge \frac{1}{N}\sum_{x=1}^{N}\min(1,\alpha_x))$ for general representation, we can write the above equations as,

$$wz \ge \frac{1}{z}\sum_{b\in z}\sum_{uk\in b}\sum_{fn\in C}\{(\tau_i - C_1).\min(1,xf)\} \tag{25}$$

Since we have xf^* is cache for maximizing the proposed solution, so

$$\sum_{b\in z}\sum_{uk\in u}\sum_{fn\in C}\{(\tau_i - C_1).\min(1,xf)\} \ge \sum_{b\in z}\sum_{uk\in b}\sum_{fn\in C}\{(\tau_i - C_1).\min(1,xf^*)\} \tag{26}$$

Combine these we can get the

$$wz \ge \frac{1}{z}\sum_{b\in z}\sum_{u\in b}\{(\tau_i - C_1).\min(1,\sum_{j\in z}^{c} xf^*)\} \tag{27}$$

$$= \frac{w}{wz} \tag{28}$$

Appendix B

The number of iterations of the algorithm depends upon the collection of SBS, and the number of user request, also cache file within each SBS and within the cluster.

References

[1] T. J. Barnett, A. Sumits, S. Jain, and U. Andra, "Cisco Visual Networking Index (VNI) Update Global Mobile Data Traffic Forecast," *Vni*, pp. 2015–2020, 2015. Article (CrossRef Link).

[2] N. Bhushan et al., "Network densification: The dominant theme for wireless evolution into 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 82–89, 2014. Article (CrossRef Link).

[3] J. G. Andrews, H. Claussen, M. Dohler, S. Rangan, and M. C. Reed, "Femtocells : Past , Present , and Future," *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, 2012, vol. 30, no. 3, pp. 497–508. Article (CrossRef Link).

[4] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and Device-to-Device Collaboration:A New Architecture for Wireless Video Distribution," *IEEE Comm. Mag.*, no. i, pp. 1–7, 2011. Article (CrossRef Link).

[5] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402–8413, 2013. Article (CrossRef Link).

[6] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, 2014. Article (CrossRef Link).

[7] W. Han, A. Liu, and V. K. N. Lau, "PHY-caching in 5G wireless networks: Design and analysis," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 30–36, 2016. Article (CrossRef Link).

[8] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: Design aspects, challenges, and future directions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 22–28, 2016. Article (CrossRef Link).

[9] X. Peng, J. C. Shen, J. Zhang, and K. B. Letaief, "Backhaul-aware caching placement for wireless networks," in *Proc. of 2015 IEEE Glob. Commun. Conf. GLOBECOM 2015*, 2015. Article (CrossRef Link).

[10] W. C. Ao and K. Psounis, "Fast Content Delivery via Distributed Caching and Small Cell Cooperation," *IEEE Trans. Mob. Comput.*, pp. 1–1, 2017. Article (CrossRef Link).

[11] M. S. Elbamby, M. Bennis, W. Saad, and M. Latva-Aho, "Content-aware user clustering and caching in wireless small cell networks," in *Proc. of 2014 11th Int. Symp. Wirel. Commun. Syst. ISWCS 2014 - Proc.*, pp. 945–949, 2014. Article (CrossRef Link).

[12] Z. Chen, J. Lee, T. Q. S. Quek, and M. Kountouris, "Cooperative Caching and Transmission Design in Cluster-Centric Small Cell Networks," *IEEE Trans. Wirel. Commun.*, vol. 16, no. 5, pp. 3401–3415, 2017. Article (CrossRef Link).

[13] C. Xu, S. Jia, M. Wang, L. Zhong, H. Zhang, and G. M. Muntean, "Performance-aware mobile community-based VoD streaming over vehicular Ad Hoc networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 3, pp. 1201–1217, 2015. Article (CrossRef Link).

[14] M. Chen, Y. Hao, M. Qiu, J. Song, D. Wu, and I. Humar, "Mobility-aware caching and computation offloading in 5G ultra-dense cellular networks," *Sensors (Switzerland)*, vol. 16, no. 7, 2016.

[15] A. Kabir, M. S. Iqbal, A. Jaffri, and S. A. Rathore, "User Aware Edge Caching in 5G Wireless Networks," *Int. J. Comput. Sci. Netw. Secur.*, vol. 18, no. 1, pp. 25–32, 2018. Article (CrossRef Link).

[16] L. Zhou, D. Wu, Z. Dong, and X. Li, "When Collaboration Hugs Intelligence: Content Delivery over Ultra-Dense Networks," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 91–95, 2017. Article (CrossRef Link).

[17] L. Zhou, D. Wu, J. Chen, and Z. Dong, "Greening the Smart Cities: Energy-Efficient Massive Content Delivery via D2D Communications," *IEEE Trans. Ind. Informatics*, vol. 14, no. 4, pp. 1626–1634, 2018. Article (CrossRef Link).

[18] J. Liu, B. Bai, J. Zhang, and K. B. Letaief, "Content caching at the wireless network edge: A distributed algorithm via belief propagation," in *Proc. of 2016 IEEE Int. Conf. Commun. ICC 2016*, 2016. Article (CrossRef Link).

[19] L. Gkatzikis, V. Sourlas, C. Fischione, I. Koutsopoulos, and G. Dan, "Clustered content replication for hierarchical content delivery networks," in *Proc. of IEEE Int. Conf. Commun.*, vol. 2015–Septe, pp. 5872–5877, 2015. Article (CrossRef Link).

[20] S. Wang, X. Zhang, K. Yang, L. Wang, and W. Wang, "Distributed edge caching scheme considering the tradeoff between the diversity and redundancy of cached content," in *Proc. of 2015 IEEE/CIC Int. Conf. Commun. China*, ICCC 2015, 2016. Article (CrossRef Link).

[21] H. Ahlehagh and S. Dey, "Video-aware scheduling and caching in the radio access network," *IEEE/ACM Trans. Netw.*, vol. 22, no. 5, pp. 1444–1462, 2014. Article (CrossRef Link) .

[22] S. E. Hajri and M. Assaad, "Caching improvement using adaptive user clustering," in *Proc. of 2016 IEEE 17th Int. Work. Signal Process. Adv. Wirel. Commun.*, pp. 1–5, 2016. Article (CrossRef Link).

[23] M. Dehghan et al., "On the complexity of optimal request routing and content caching in heterogeneous cache networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 3, pp. 1635–1648, 2017. Article (CrossRef Link).

[24] Y. Wang, X. Tao, X. Zhang, and G. Mao, "Joint caching placement and user association for minimizing user download delay," *IEEE Access*, vol. 4, pp. 8625–8633, 2016. Article (CrossRef Link).

[25] S. Ren et al., "Design and analysis of collaborative EPC and RAN caching for LTE mobile networks," *Comput. Networks*, vol. 93, pp. 80–95, 2015. Article (CrossRef Link) .

[26] K. . b Poularakis, G. . b Iosifidis, and L. . b Tassiulas, "Approximation algorithms for mobile data caching in small cell networks," *IEEE Trans. Commun.*, vol. 62, no. 10, pp. 3665–3677, 2014. Article (CrossRef Link).

[27] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *Proc. of IEEE INFOCOM*, vol. 1, pp. 126–134, 1999. Article (CrossRef Link).

[28] A. Brodersen, S. Scellato, and M. Wattenhofer, "YouTube Around the World : Geographic Popularity of Videos," in *Proc. of 21st Int. Conf. World Wide Web*, pp. 241–250, 2012. Article (CrossRef Link).

[29] D. C. Chen, S. S. Member, T. Q. S. Quek, S. S. Member, and M. Kountouris, "Backhauling in Heterogeneous Cellular Networks : Modeling and Tradeoffs," *IEEE Trans. Wirel. Commun.*, vol. 1276, no. to appear, pp. 1–29, 2015. Article (CrossRef Link).

[30] E. Bastug, J.-L. Guenego, and M. Debbah, "Proactive small cell networks," *Ict 2013*, pp. 1–5, 2013. Article (CrossRef Link).

[31] F. Pantisano, M. Bennis, W. Saad, and M. Debbah, "In-Network Caching and Content Placement in Cooperative Small Cell Networks," in *Proc. of 1st Int. Conf. 5G Ubiquitous Connect.*, 2014. Article (CrossRef Link).

[32] K. Poularakis, G. Iosifidis, and L. Tassiulas, "Approximation algorithms for mobile data caching in small cell networks," vol. 62, no. 10, pp. 3665–3677, 2014. Article (CrossRef Link).

ASIF KABIR is a Ph.D. candidate in the college of communication engineering at Chongqing University. He received his Master of Sciences degree in the field of Computer Science from the University of Azad Jammu & Kashmir Pakistan. He worked as a "Web Manager" in Information Technology Board Muzaffarabad AJ&K from January 2010 to December 2010. Afterwards, he joined University of Azad Jammu& Kashmir from December 2010 to May 2014 as a "System Engineer/ Network Administrator". From May 2014 to date he is working as a "System Engineer/ Network Administrator" in University of Kotli" Azad Kashmir. His main research areas are in mobile wireless communication networks, wireless sensor networks, and big data.