

GaoNing

1st Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address

2nd Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address

3rd Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address

4th Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address

5th Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address

6th Given Name Surname

dept. name of organization (of Aff.)

name of organization (of Aff.)

City, Country

email address

Abstract—This document is a model and instructions for L^AT_EX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.

Index Terms—component, formatting, style, styling, insert

I. INTRODUCTION

With the development of mobile Internet, a large number of new network services and applications are widely applied, such as AR/VR, mobile high definition video, online game etc., which require that the mobile network provides the higher data transmission rate and lower network delay. Moreover, it is predicted that by year 2020, the mobile data traffic will reach 30.6 exabytes/month globally [1]. To handle the severe challenge, densely deployed small cell network (SCN) have been widely introduced by network operators to enhance the network capacity. However, the costly and heavily-loaded backhaul link between SCN and the core network is becoming the bottleneck.

Caching on the SCN has been widely recognized as an effective and economical solution to tackle the aforementioned problem of network densification. By storing popular files in the SBS cache in advance [2], [3], more requests can be satisfied at SBSs instead of retrieving duplicated file over the backhaul links. Besides, downloading directly from SBSs reduces download delay substantially due to the short transmission distance and thus high rate. However, the storage capacity on SBSs is generally limited, and thus caching strategies should be carefully designed to make the best use of the SBS cache.

There have been some recent works focusing on designing file caching strategies in the scenario of network densification. Reference [2] present a wireless distributed caching strategy with a low-bandwidth backhaul link but high storage capacity where user can access multiple SBSs. Finding the optimal cache placement to maximize cache hit ratio is proved to be NP complete. The authors of [2] study the optimization issue

for cache content placement in caching enabled SCN with heterogeneous file and cache sizes, and adopt multicast transmission to minimize the average backhaul rate. In [Energy-efficient Cooperative Coded Caching for Heterogeneous Small Cell Networks], the authors propose an cooperative caching strategy with a low-bandwidth backhaul link but high storage capacity where user can access multiple SBSs. Finding the optimal cache placement to maximize cache hit ratio is proved to be NP complete. The authors of [cluster hetnet coded] study the optimization issue for cache content placement in caching enabled SCN with heterogeneous file and cache sizes, and adopt multicast transmission to minimize the average backhaul rate. In [4], the authors propose an cooperative caching strategy where SBS can get desired content from neighbouring SBSs so as to enhance the content delivery efficiency.

Although these recent works have provided insight into caching strategy in cellular network, users are assumed to be static and the association between users and SBSs remains unchanged. Obviously, such assumption is unreasonable. Users are mobile and the association to SBSs can change during the download of a file, which can be more frequently with densely deployed SBSs. Therefore the impact of user mobility can not be neglect when designing file caching strategies [5]. Reference [6] assumes that the user movement obeys a discrete Markov model and the amount of data downloaded by users depends on their location and the cache placement in each time slot. The author of [7] uses a discrete random jump model to describe the mobility pattern and derive the expression of throughput. Due to the complexity of the problem, two heuristic algorithms are provided to obtain the optimal solution. Reference [8] model user mobility as Markov chain and put forward a distributed caching paradigm in a two-tier heterogeneous network with the aim of minimizing the content fetching from MBS.

Although these works have taken user mobility into consideration, they all consider a static global file popularity among all the SBSs. The effect of mobility on content popularity at

different SBSs are not well investigated. In a real system, SBSs covering different areas generally have different file popularity, considering the global popularity is not suitable. Moreover, file popularity of neighboring SBS will affect each other, because moving user may request the same file again in another SBS if he fails to complete the download in the current SBS.

In this paper, we propose a mobility-aware coordinate proactive caching strategy. Taking advantages of heterogeneous small cell network architecture, both MBS and SBS are equipped with cache and files can be delivered between them. In order to measure the data rate and transmission delay of the user request file together, we adopt the effective capacity criterion. Effective capacity is proposed to form channel model in link layer to calculate the data rate for a specific user with the impact of transmission delay.

The contributions of this paper include:

- Allowing multiple V-UEs share the same RBs to explore the potential to accommodate more V-UEs.
- We construct the conflict graph based on the worst-case Signal-to-Interference Ratio(SINR).
- We model the allocation problem as *Graph Coloring Problem*, which is NP-Complete.
- A Resource Sharing and Admission Control(RSAC) scheme is proposed to solve the problem.

The remainder of this paper is organized as follows. Section II describes the system model, including network model, file popularity model, mobility model and transmission delay model. In Section III, we introduce the proposed mobility-aware coordinate proactive caching strategy and formulate the cache placement problem as an optimization problem. In Section IV, a genetic algorithm is adopted to solve the problem. In Section V, numerical results are illustrated with performance comparisons between MPC. Finally, conclusions are summarized in Section VI.

II. SYSTEM MODEL

A. Network Model

In this paper, we [4] consider a heterogeneous network comprising a single MBS and K densely deployed SBS denoted as $\mathbf{S} \triangleq \{1, 2, 3, \dots, K\}$, where S_k represents the MBS. The MBS and SBSs are connected by a fronthaul link with limited capacity of r_{FH} and the MBS is connected to the EPC network by a backhaul link with limited capacity of r_{BH} . The set of active users is $\mathbf{U} \triangleq \{1, 2, 3, \dots, U\}$ and each user only requests one file at a time while moving from one cell to another from time to time. We consider that each user u only connects to and receives data from the nearest BS (in terms of signal strength), which we later refer to as the user u 's home BS. And the users set associate with SBS S_k is denoted as U_k .

We assume that the MBS has access to a library of N files denoted as $\mathbf{F} \triangleq \{1, 2, 3, \dots, N\}$, and each file has the same size denoted by $B[\text{MB}]$. Note that the assumption of equal size files is justifiable in practice since files can be divided into blocks of the same size.

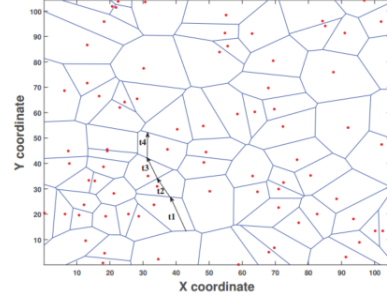


Fig. 1. System Model.

We assume that SBS S_k are equipped with a cache to store files with storage capacity of $C_i[\text{MB}]$, and the MBS is equipped with a large cache with storage capacity of $C_0[\text{MB}]$. Each user requesting for files in \mathbf{F} is initially served by the SBS he contacts. If the requested file is not present in the cache, other SBSs nearby or the MBS will send the file to the user as shown in Fig.1. To describe the cache placement decision, we define the cache placement matrix as:

$$\mathbf{X} = \begin{Bmatrix} x_{0,1} & x_{0,2} & \cdots & x_{0,N} \\ x_{1,1} & x_{1,2} & \cdots & x_{1,N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{K,1} & x_{K,2} & \cdots & x_{K,N} \end{Bmatrix}$$

where $x(i, j)$ denotes caching the file f_j in BS_i . The cache storage capacity constraints can be expressed as:

$$\sum_{j=1}^N B \cdot x_{i,j}, \quad \forall S_i \in \mathbf{S} \quad (1)$$

B. Content Popularity

We define the popularity distribution of files in cell i as $\mathbf{P} \triangleq \{p_{i,1}, p_{i,2}, p_{i,3}, \dots, p_{i,N}\}$, in which $p_{i,j}$ is the probability of the file f_j being requested from a user in cell i , and $\sum_{j=1}^N p_{i,j} = 1, \forall S_i \in \mathbf{S}$. It should be noted that the content popularity here is inferred with historical data and can not reflect the short-term variation due to user mobility.

C. Mobility Model

User mobility is modeled by discrete random jumps, with the corresponding intensity characterized by average sojourn time. In terms of the distribution of sojourn time, it is reasonable to model it by an exponential function [MEC]. Therefore, the probability density function (PDF) of sojourn time in cell i can be written as follows:

$$p(\tau, u_i) = \frac{1}{\mu_i} e^{-\tau/\mu_i}, \quad \forall S_i \in \mathbf{S} \quad (2)$$

where μ_i is the average sojourn time. Different cells have different μ_i because the coverage area of different cells is different.

In addition, in order to describe the trajectory of moving users, we adopt the stationary Markov model. Let $T_{l,i} \geq 0$

denote the probability of transition from cell l to cell i , where $\sum_{i \in S}$. When a user has incomplete download at one cell and enters the new cell, the content is requested at the new cell thus affecting the static content popularity $p(i, j)$. Therefore, the popularity $p(i, j)$ of content j in BS i may not reflect the actual request probability of content j in BS i . If the content requested by user is not cached in the new cell, it has to be fetched from EPC network via high latency link. In order to provide a better QoS service, we need to dynamically adjust the distribution of file popularity based on the user's mobile pattern, thereby reducing the huge latency caused by cache misses.

D. Delay and EC QoS Model

$$\alpha(\theta) = - \lim_{t \rightarrow \infty} \frac{1}{\theta t} \log E e^{-\theta S(t)} \quad (3)$$

where $S(t) = \int_0^t r(t) dt$ represents the data throughput accumulated on the time domain. Considering the scenario that channel coefficients keep constant over the frame duration T and vary independently for each frame, the formula of effective capacity can be rewritten as follow:

$$\alpha(\theta) = - \lim_{t \rightarrow \infty} \frac{1}{\theta t} \log E e^{-\theta T R[i]} \quad (4)$$

where $R[i]$ indicates the instantaneous channel capacity during the i th frame.

Due to the time-varying and randomness of wireless channels, deterministic delay guarantees for specific links are unreasonable and impractical. So, the effective capacity denotes θ as the statistical QoS guarantee parameter. As [9] mentions, the delay violation possibility can be written as

$$Pr\{D(\infty) > D_{max}\} \approx e^{-\theta D_{max}} \quad (5)$$

where D_{max} indicates the delay bound and a larger θ represents a better link quality or a tighter QoS constraints. Note that when θ approaches to zero, the effective capacity converges to the ergodic capability [10].

The place where user can get the requested file affects the effective capacity of the wireless link, as the value of the QoS guarantee factor changes according to $x_{i,j}$. According to the queuing theory, the queue delay $D(\infty)$ of the packet satisfies the following relation [31].

$$\theta = - \lim_{D_{max} \rightarrow \infty} \frac{\log(Pr\{D(\infty) > D_{max}\})}{D_{max} - \sum_i d_i} \quad (6)$$

where d_i is the transmission delay caused by the i -hop in data packet transmission.

In the scenario of this paper, the transmission delay can be different according to the cache placement as shown in Fig.2

home SBS cache hits do not need transmission delay. The transmission delay of MBS transferring file to home SBS is $d_0 = \frac{B}{r_{FH}}$.

SBS l share file to home SBS i will cause different transmission delay depending on the hops number h , which can be denoted by $d_{l,i} = h \frac{B}{r_{FH}}$.

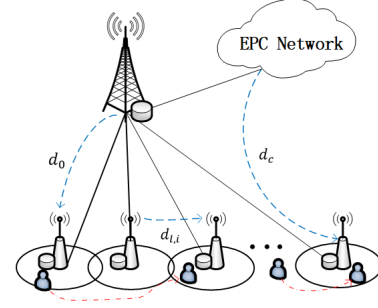


Fig. 2. Delay Model.

Retrieving files from the EPC network through the backhaul link generally results in a large transmission delay $d_c = \frac{B}{r_{BH}}$.

So, given a cache placement matrix \mathbf{X} , we can get the average transmission delay of user u in cell i requesting file j as,

$$D_{u,i,j} = (1 - x_{i,j})(x_{o,j}d_0 + \frac{1 - x_{o,j}}{K - 2} \sum_{l \in S \setminus \{o,i\}} x_{l,j}d_{l,i}) + c_{i,j}d_c \quad (7)$$

where $c_{i,j} = \prod_{l \in S} 1 - x_{l,j}$, which means file j is not stored in RAN.

The QoS guarantee parameter can be expressed as,

$$\theta_{u,i,j} = - \lim_{D_{max} \rightarrow \infty} \frac{\log(Pr\{D(\infty) > D_{max}\})}{D_{max} - \sum_i d_i} \quad (8)$$

III. MOBILITY-AWARE CACHING STRATEGY WITH EFFECTIVE CAPACITY

Here, we first explore the impact of user mobility on file popularity of different base stations. Then, we cast the cache placement optimization problem, which is aimed at maximizing the system effective capacity. Mobility-aware content request probability: When a user with an incomplete download session of content j arrives at cell k , part of the file, $B_{k,j}$, has been downloaded already; hence the user only needs to download the rest of the content j . For simplicity, we assume the maximum bandwidth of G [MB/s] of each BS is shared equally among the associated users of that BS. Thus, the bandwidth allocated to each user connecting to BS l is given by $r_l = \frac{G}{U_l}$.

Since each content file has B MB, the time needed to completely download one content at cell l is $t_l = B/r_l$. However, the sojourn time that user stays in cell l is subject to exponential distribution, so the probability of the user completing the download when leaving the cell l can be inferred using the Cumulative Distribution Function $F(\tau, \mu_l)$ as,

$$q_{l,j} = Pr(\tau < t_l - \frac{B_{l,j}}{r_l}) = AAAAAAAAAA \quad (9)$$

Then, the file requested probability $p(i, j)$ for content j at cell i is,

$$P_{i,j} = \begin{cases} p_{i,j} + \lambda \sum_{l \neq i} q_{l,j} p_{l,j}, & i = 0 \\ p_{i,j} + (1 - \lambda) \sum_{l \neq i} q_{l,j} p_{l,j} T_{l,j}, & i \in S \setminus 0 \end{cases} \quad (10)$$

Where $p(i, j)$ is the prior popularity of file j at cell i and $T(l, i)$ is the transition probability from cell l to i . And $\lambda \in (0, 1)$ is constant determined empirically, controlling the tendency of the file cache hierarchy. A larger λ factor will make files with higher mobile strength tend to be cached in the MBS cache, voiding cache misses caused by frequent handovers among SBS. Conversely, a smaller λ factor will make the file tend to be cached in the SBS cache closer to the user, minimizing the transmission delay.

IV. PROBLEM FORMULATION

Based on the system model and effective capacity theory previously mentioned, the effective capacity achieved by user u in cell i can be expressed by the following formulation:

$$EC_{u,i} = \sum_j \mathbf{p}_{i,j} \left(-\frac{1}{\theta_{u,i,j} T} \log E e^{-\theta_{u,i,j} T r_i} \right) \quad (11)$$

The problem can be formulated as follows with the purpose of maximizing the sum effective capacity of the whole system.

$$\begin{aligned} \max \quad & \sum_u \sum_i EC_{u,i} \\ \text{s.t.} \quad & \sum B_{x_{i,j}} < C_i \\ & x_{i,j} \in \{0, 1\} \end{aligned} \quad (12)$$

Obviously, the optimal problem is a Mixed-Integer Nonlinear Programming (MINLP) and it is highly complicated to find the global optimization solution. In next section, the Genetic Algorithm (GA) is adopted to get a sub-optimal solution.

V. THE SOLUTION OF OPTIMAL PROBLEM

Genetic Algorithm [5] is adaptive heuristic search algorithm premised on the evolutionary ideas of natural selection and genetic. The basic concept of GA is designed to simulate processes in natural system necessary for evolution, specifically those that follow the principles first laid down by Charles Darwin of survival of the fittest. As such they represent an intelligent exploitation of a random search within a defined search space to solve a problem. The processes of GA include coding, colony initialization, initialization, crossover and mutation.

The GA maintains a population of possible cache placement solutions. A solution corresponds to a chromosome which is an encoded representation of the cache placement of all the SBS and MBS.

A. Coding

We choose the linear structure coding method, in which a chromosome represents a feasible solution of the cache allocation problem. There are $N(K + 1)$ elements in each chromosome as shown in fig3; each element records the allocation information in the corresponding cache placement matrix \mathbf{X} .

B. Colony initialization

We generate Z_g chromosomes with $\frac{\sum_{i=0}^K C_i}{NB}$ number of 1 randomly distributed so as to satisfy the cache capacity constrain.

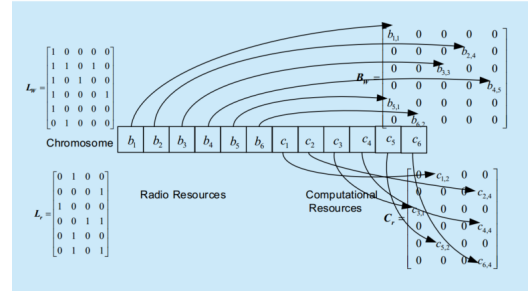


Fig. 3. Coding of Genetic Algorithm.

C. Fitness function

Fitness function is the system optimal function defined in equation (1).

D. Selection

The chromosomes in the colony are ranked, then the first $x\%$ (Elite selection ratio) is selected and the remains are selected at random. This strategy can guarantee that the genetic algorithm converges to the global optimal solution.

E. Crossover

For two chromosomes to be mated which are chosen with crossover probability P_c , exchange the each corresponding element with probability P_s (crossover ratio).

F. Mutation

For each chromosome chosen with mutation probability P_m , replace each element with a random value with probability P_b (mutate ratio).

G. Algorithm process

- step 1 Initialize colony.
- step 2 Evaluate the individual in colony using fitness function, if the evolutionary generation reaches to the terminal generation then jump to step 4, otherwise continue.
- step 3 Generate the new colony using selection, crossover and mutate operator, then jump to step 2.
- step 4 Get the cache placement matrix through the best individual generated by genetic algorithm.
- step 5 Deliver files to each BS according to the cache placement matrix. Cache placement in this cycle is over.

VI. PERFORMANCE EVALUATION

In this section, we use computer simulation methods to evaluate the performance of the mobility-aware proactive caching placement scheme for heterogeneous ultra density networks. We simulated a heterogeneous ultra density network depicted in Fig1, where the MBS is located in the center and 16 SBSs are uniformly deployed in the network. Moreover, we divided the macrocell into 4 clusters, and each cluster consists of 4 SBSs. Additionally, we consider a wrap-around network layout such that when a user moves out of the network on one side, it comes back in on the opposite side w.r.t. the origin. There several users in the network and the initial location of the users

is random, following a uniform distribution. We assume that the files users request can be modeled as the Zipf distribution. The request probability of file f_j is

$$p_j = \frac{1/j^\gamma}{\sum_{i=1}^N 1/i^\gamma}, \quad \forall j \in F$$

where $\gamma \in \{0.3, 1.3\}$ is the skewness reflecting the concentration of the popularity distribution among users. Moreover, we consider a library of $N = 200$ files, each of size $B = 50\text{MB}$. The popularities of the files at each BS are generated randomly simulating different file preferences in different regions. The backhaul rate is set to $r = 1000\text{Mbps}$ and the fronthaul rate $r_{FH} = 600\text{Mbps}$.

The parameter of genetic algorithm are set as Table I shown.

TABLE I
SIMULATION PARAMETERS

| Parameters | Value |
|-----------------------------|-------|
| Colony size Z_g | 50 |
| Terminal generation number | 100 |
| Crossover probability P_c | 0.75 |
| Aberrance probability P_m | 0.02 |
| Reserve ratio x% | 25% |
| Crossover ratio P_s | 0.5 |
| Mutate ratio P_b | 0.05 |

We compare the performance of our optimal caching scheme with MPC (most popular caching) schemes, which uses global static content popularity and each BS stores the most popular files until its cache is full.

A. Capacity Based System Throughput Performance

Fig.4 illustrate the performance of our proposed scheme where we set $\gamma = 0.7$ and $\mu = 4$. Fig.4 shows the effective capacity based system throughput when the num of users varies in the network. Obviously we see that as the number of user increases, the system throughput improves and the mobility-aware caching strategie always outperforms the MPC strategy. This is because that files cached in the SBS and MBS changes according to users location and preference. Hence, it can match the user request better and reduce the transmission delay.

Fig.5 investigate the relationship between cache size and system throughput. As shown in Fig.5, with the total cache size increases, the effective capacity based system throughput gains at the same time. It demonstrates the fact that expense of storage can alleviate the shortage of backhaul bandwidth and improve the data rate that a user achieves. The more contents BSs cache, the more chances users can get the contents in RAN so that they can be served immediately.

B. Cache Hit Ratio Performance

Fig.6 display the impact of file populatity distribution parameter γ on the cache hit ratio. In this simulation, user number is set to 70. As the figure shows, when γ is small, both two caching strategy have low cache hit ratio. Because in this

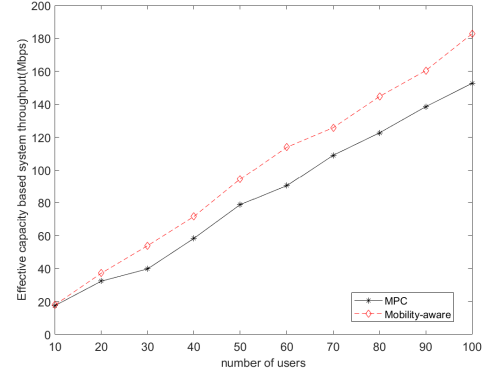


Fig. 4. Fig4.

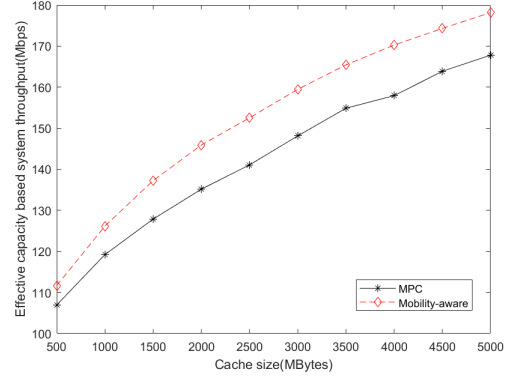


Fig. 5. Fig5.

case, all files are nearly equally likely to be requested but the storage capacity of BS is limited. But mobility-aware caching strategy still achieves superior performance than MPC. When γ becomes larger, the requests are more concentrated among some popular files. Therefore, these files are more likely to be cached in most of the SBS resulting in hitting more user requests. Hence the caching hit ratio of mobility-aware and MPC both improve notably.

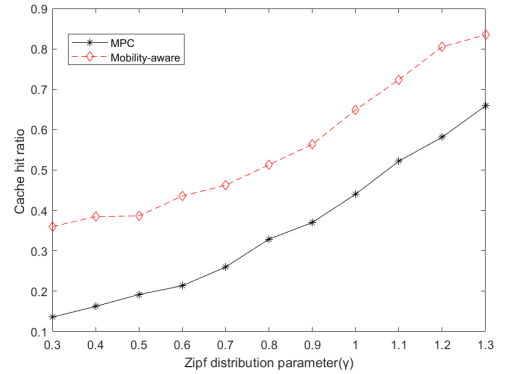


Fig. 6. Fig6.

VII. CONCLUSION

Algorithm 1 Admission Control.

Input:

Admission control vector \mathbf{A} with $a_m = 1, \forall m \in \mathbf{M}$.

Output:

Admission control vector \mathbf{A} .

```

1: for all  $C_q \in \mathbf{C}$  do
2:   while  $|C_q| > N$  do
3:     if  $\mathbf{L} \cap C_q \neq \emptyset$  then
4:        $v_m = \arg \max_{v_m \in \mathbf{L} \cap C_q} |C_q|$ ;
5:     else
6:        $v_m = \arg \max_{v_m \in \mathbf{H} \cap C_q} |C_q|$ ;
7:     end if
8:      $a_m = 0$ ;
9:      $C_q \leftarrow C_q \setminus v_m, \forall C_q \in \mathbf{C}^m$ 
10:     $|C_q| \leftarrow$ 
11:  end while
12:  Employ greedy algorithm to allocate RBs.
13: end for
14: Remove added edges and regain some RBs for the corresponding vertices.
```

$\sum_{l=1}^L a_{l,n}^{(c)} > M_N$ In this paper, we proposed a new mobility-aware proactive caching strategy for heterogeneous ultra-dense network and use the effective capacity to evaluate the effect of transmission delay. We use random jump model and stationary Markov model to describe the mobile pattern of user and amend the popularity at BSs with it. Then, we formulated the optimal content placements problem and solved it by genetic algorithm. Finally, simulation results show that the proposed mobility-aware proactive caching strategy achieves higher throughput and cache hit ratio than MPC strategy while users are moving. This indicates that our proposed caching strategy is a promising way to address the challenge of network densification.

REFERENCES

- [1] C. V. Mobile, "Cisco visual networking index: Global mobile data traffic forecast update, 2011-2016," 2014.
- [2] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 8402–8413, Dec 2013.
- [3] E. Ba?tu?, M. Bennis, and M. Debbah, "Cache-enabled small cell networks: Modeling and tradeoffs," in *2014 11th International Symposium on Wireless Communications Systems (ISWCS)*, Aug 2014, pp. 649–653.
- [4] Q. Jia, R. Xie, T. Huang, J. Liu, and Y. Liu, "Energy-efficient cooperative coded caching for heterogeneous small cell networks," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, May 2017, pp. 468–473.
- [5]
- [6] K. Poularakis and L. Tassiulas, "Exploiting user mobility for wireless content delivery," in *2013 IEEE International Symposium on Information Theory*, July 2013, pp. 1017–1021.
- [7] X. Liu, J. Zhang, X. Zhang, and W. Wang, "Mobility-aware coded probabilistic caching scheme for mec-enabled small cell networks," *IEEE Access*, vol. 5, pp. 17824–17833, 2017.

- [8] K. Poularakis and L. Tassiulas, "Code, cache and deliver on the move: A novel caching paradigm in hyper-dense small-cell networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 675–687, March 2017.