2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)

Edge Caching via Content Offloading in Heterogeneous Mobile Opportunistic Networks

Chenyang Wang[†‡], Wenkai Li[†‡], Ding Li[†‡], Mingyang Song[§], Chen Dong[¶], and Xiaofei Wang[†‡]
[†]School of Computer Science and Technology, Tianjin University, Tianjin, China
[‡]Tianjin Key Laboratory of Advanced Networking, Tianjin, China
[§]School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China
[¶]HUAWEI Technologies CO., LTD , Shenzhen, China
chenyangwang@tju.edu.cn, liwenkai_1994@126.com, liding_cs@tju.edu.cn,
hsdyjssmy@126.com, cd2g09@ecs.soton.ac.uk, xiaofeiwang@tju.edu.cn

***Abstract*—Content transmission over Mobile Opportunistic Networks is in the manner of "store-carry-forward" due to opportunistic contacts between node pairs. Most of the existing works focus on the "forward" process rather than how to "store" content in the network. Moving contents to the edge of network can significantly offload network traffic while satisfying content requests from mobile users locally. In this paper, considering the preference of nodes for different content items, we propose Fetcher Selection Greedy Algorithm (FSGA), a novel strategy to select a subset of nodes for cooperative caching scheme by evaluating each node utility in the network. To improve the cooperative caching opportunity and reduce the traffic pressure of core network, we introduce the Heterogeneous Mobile Opportunistic Networks (HMONs) architecture by deploying an Access Point (AP) which is acting as a bridge to connect the core network and mobile nodes. The mobile nodes in HMONs are segmented into two kinds, one is called $Fetchers$ which are responsible for caching content from AP then forwarding them to the other ones, and the other mobile nodes are $Regulars$. We formulate and analyze the average content transmission delay in different transmission conditions. Finally, our experiment results indicate that the cooperative caching scheme improves the network performance.**

***Index Terms*—edge caching, content offloading, heterogeneous mobile opportunistic networks**

I. Introduction

Along with the mobile applications environment moving to maturity progressively, wherein more powerful sensing and computing functions have been integrated into the wireless portable devices, making these smart devices enable short range communications by the technologies such as Wi-Fi, Bluetooth or Zigbee, which giving rise to the emergence of mobile opportunistic networks [1]. In this network diagram, only intermittent communication links existing due to the sparse node density and the constantly changeable node mobility, and the difficulty of maintaining the node pairs connectivity makes it necessary to employ "store-carry-forward" way for content transmission. Such networks are suitable for the applications in the fields of information sensing [2], urban computing [3], etc.

However, the unpredictable node mobility may cause the high delay when content is transmitted along the time-varying end-to-end path, resulting in the over consumption of network resource. Selecting "which" mobile nodes as relays is the key problem in the aforementioned application scenarios. Compare to the traditional networks, it is especially important to deploy effective content caching management policy in mobile opportunistic networks. Most of the previous work focus on drawing the "forward" strategy (e.g., Spray-Wait [4], SMART [5] and Hotent [6]), only limited researches on "store" strategy [8].

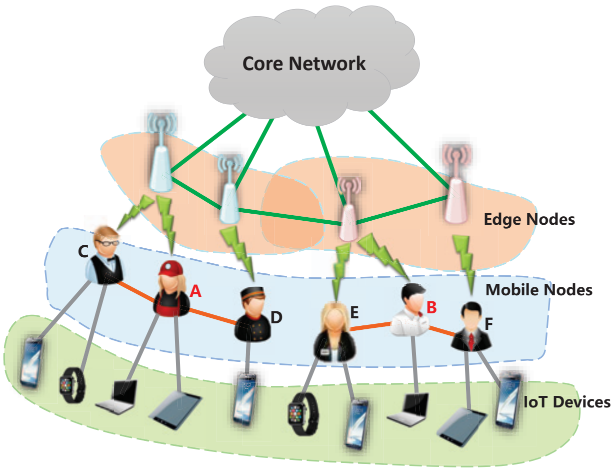


Fig. 1: Demonstration of Heterogeneous Mobile Opportunistic Networks Architecture

The architecture of Heterogeneous Mobile Opportunistic Networks (HMONs) is demonstrated in Fig. 1, where APs are considered as edge nodes, mobile nodes (e.g. users) can connect to each other with various of smart IoT devices. Mobile nodes are connected with each other opportunistically, and any other node pairs can cache content items and forward them under the traditional MONs. In this paper, we artificially endow the different duties between the $Fetchers$ and the $Regulars$, assuming that only $Fetchers$ can cache content item, e.g. nodes A and B, these two $Fetchers$ are responsible for caching content from the edge node(AP) and the $Regulars$, e.g. C,...,H. The caching and content exchange process occurred among $Fetchers$, $Fetchers$ and $Regulars$ as well as $Fetchers$ and AP, but such process will be

978-1-5386-7308-9/18/$31.00 ©2018 IEEE
DOI 10.1109/ICPADS.2018.00107

suffocated among the $Regluars$ themselves.

In our scheme, edge node AP is acting as a bridge which contacts the core network and mobile nodes. We assume that all the mobile nodes visit AP in the same probability. When content is generated by a resource (no matter what the resource is), it will be pushed into the $Fetchers$, one copy of content is cached in all the $Fetcher$ nodes. Due to the limited buffer occupation of the $Fetchers$, we propose a dynamic content replacement policy to ensure that more popular content can be maintained in the network based on the content query history. The main contributions are demonstrated as following:

- We model the node utility by the probability of opportunistic encounter with each other, and propose Fetcher Selection Greedy Algorithm (FSGA) to select a subset of nodes in the network with high utilities and the submodularity of FSGA is proved.
- We borrow the "divide-and-conquer" strategy to present a cooperative caching scheme to allocate the more popular content to $Fetchers$ and AP while the contents with less popularity are cached in the $Regulars$.
- We use Markov Chain to formulate and analyze the average content request and transmission delay within "two-hop" transmission. The transmission process is divided into two categories, the one without AP assisted is called $Device-to-Device\ Communcation\ Phase$ (D2D), the other is $Device-to-AP\ Communication\ Phase$ (D2A). We obtain the content request delay as $\frac{1}{N_m}(\frac{1}{\beta+\lambda})$ and $\frac{1}{N_{ap}}(\frac{1}{\beta+\lambda})$, respectively for D2D and D2A communication. And the expectation delay of transmission of D2D and D2A communication are $\frac{1}{\beta}\sqrt{\frac{\pi}{2}}\frac{1}{\sqrt{N_M-1}}$ and $\frac{1}{\lambda}\sqrt{\frac{\pi}{2}}\frac{1}{\sqrt{N_{AP}-1}}$, respectively.

The rest of this paper are organized as follows. Section II reviews related works and section III introduces the overview of HMONs and the system model. Then, we formulate the optimization problem and propose the FSGA to select $Fetchers$. Section V presents caching scheme. In Section VI, we discuss the average content transmission delay in different conditions and simulation results are demonstrated in section VII. We conclude our paper and discuss some future research areas in Section VIII.

II. RELATED WORK

Recall that only fews work focus on the "store" process during the past several years, however, some remarkable efforts have been made to improve the content accessibility or message delivery ratio in terms of optimizing some internal factors (e.g. the node utility, message utility or both) or integrating the external factors (e.g. node GPS locations, auxiliary infrastructures, etc.).

A. Internal-factor-based Caching Strategy

The authors in [8] employed the node utility that the probability of content forwarding to determine a set of nodes in the network central locations (NCLs), caching the content in NCLs and their neighbours. However, in order to improve the content accessibility and reduce the content access delay, thus cooperative caching among multiple nodes may high caching overhead of the nodes in the network central locations.

C. Barakat et al. proposed a message utility based caching content management strategy [9]. They introduced the per-message utility, aiming to maximize the network throughput, by evaluating the request rate and delivery ratio of message based on the global knowledge of network. However, one barrier to deploy the caching protocol into opportunistic networks is lack of global knowledge, which makes it unapplicable.

Wang et al. [10] introduced a hierarchical cooperative caching scheme, which obtains both the node inter-contact time and message request rate, nodes are portioned into $friends$ and $strangers$ and the caching buffer space is divided into three components: self, friends, and strangers equally. However, when the buffer is full, the new most popular message will replace the lowest one in the local community, which leads to low content accessibility of other users.

B. External-factor-based Caching Strategy

Recent studies focus on the metrics such as wireless communication technologies (e.g. GPS, Wifi, etc.) [11] [12], and auxiliary infrastructures [13] [14] etc., wherein, these external factors always play an important role of caching strategies made. For example, in [12], Park et al. proposed a location-based cache maintenance strategy, which enables the mobile nodes to pre-fetch message that they interested in, and caches them at a location close to the node's location in the wireless broadcast environment. Yaping Sun et al. [13] proposed a cooperative content caching scheme, and cached the content items in the base stations (BSs), where users can request the content from the local BS, the other cooperative BSs or the core network. In the similar work [14], only the popularity content can be obtained from the BSs. The authors of both aforementioned work aim to minimize the average delay in terms of caching content at BSs, however, cooperative caching without nodes participation may lead to low content delivery ratio.

III. OVERVIEW AND MODELING

In this section, firstly we demonstrate an overview of HMONs-based cooperative caching scheme, then discuss the network model and finally we introduce the preference of nodes for different content items.

A. Overview

The HMONs-based edge caching via content offloading scheme is depicted in Fig. 2. Considering in the single cell case, here we treat the edge node AP as the spring pool, $Fetchers$ are responsible for caching contents from it and deliveries them to the $Regulars$. For example, the content request for c_2 of $Regular$ node R_2 is satisfied by the $Fetcher$ node F_2, F_2 moves to R_2 with the contact rate λ and transmits the content c_2 to R_2 with the probability of P_{F_2,c_2} in Fig. 2 (the parameters will be defined in the next Section). However, as for the way that how these $Fetchers$ receive

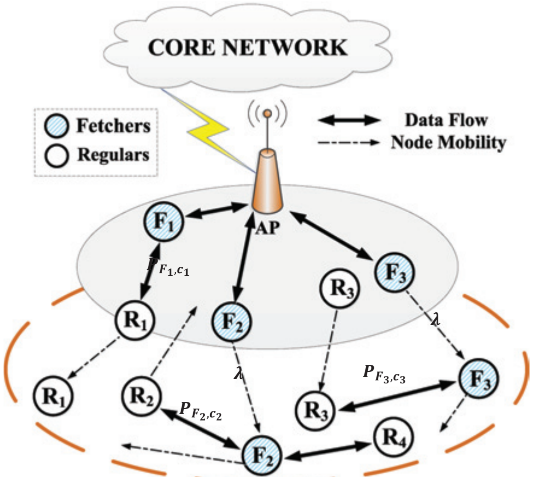


Fig. 2: Single Cell Case: Edge Content Offloading under HMONs

content or forward them to the $Regulars$ in some routing protocols, we don't focus on the full analysis in our paper. Initially, some mobile nodes are chosen as the $Fetchers$ according to the cumulative contact ratio between node pairs and node interest. The $Fetchers$ cache one copy of each top popular content (so as the caching strategy of AP, but the buffer of AP is much larger) until the buffer is full and the $Regulars$ keep the different content randomly. However, we only consider the two-hop query model in the query process, that is, if a $Regular$ cannot receive content after querying all the $Fetchers$, it will request the content from AP directly.

B. Network Modeling

In our work, we utilize the network contact graph $G = (V,\ E)$ to describe the heterogeneous MONs, since the mobile nodes are divided into two types, here we assuming that there are $F + R$ mobile nodes in our Network, denoted as $V_i \in \{v_1, v_2, ..., F+R\}$ and the opportunistic contact process between a node pair $i, j \in V$ is modeled as an opportunistic edge $e_{ij} \in E$. Wherein, the Poisson distributed contacts are widely used to model opportunistic contact process [17] [18] [19], hence we obtain that the Inter-Contact Time (ICT) of the pairwise node follows exponential distribution, which giving raise to a Poisson process with the contact rate β_{v_i,v_j} between mobile nodes v_i and v_j. In our abstract network topology, it is obviously unpractical to employ part of the nodes to participate the content caching and forwarding process willingly. Therefore, the $Fecthers$ would be selected carefully from a God's perspective based on the cumulative contact ratio of the node pair v_i and v_j, which remains that we can obtain the contact rate β_{v_i,v_j} as a relative constant at real-time.

C. Node Preference Modeling

In the real world, some contents may be interested in many nodes, while they are not so popular to some other nodes. So it is practical to consider the different node preference in our system. Similar to previous work [8], we assume that the popularity of content item follows Poisson Distribution. In this way, we model the node preference for different content items.

It is well known that if we interest in some contents, we will request that in the Internet. Note that the preference of one node for a certain content item can be reflected from the number of requests for the item intuitively. Specially, in this paper, we formulate the node preference in terms of the content request perspective. For all kinds of content items in the network system, denoted as $\mathcal{C}$ and w_c is the popularity of content item c which describes the importance of that content item. To model the node preference, we define $r_{v_i,c}$ as the request number of node $v_i \in V$ for the content item c. In this way, the preference degree of node $v_i \in V$ for content item $c \in \mathcal{C}$, denoted by $p_{v_i,c}$, can be obtained as $p_{v_i,c} = r_{v_i,c} w_c$.

IV. PROBLEM STATEMENT

In this section, we formally formulate the node utility, and demonstrate the $Fetchers$ selection algorithm and its mathematical analysis.

A. Node Utility

We will formulate the node utility and the edge content offloading optimization problem in the single cell case. Firstly, the utility of mobile nodes is discussed and then we can obtain the content offloading optimization by considering the edge node (AP) assisted.

According to the aforementioned analysis of node preference and network modeling, we can obtain each node utility, then leading to a greedy algorithm to select $Fetchers$. Denote the indicator function $B = (b_{v_i,c})$, $v_i \in \mathcal{F}$ and $c \in \mathcal{C}$, as the caching state that wether the $Fetcher$ node f stores content item c, in which $b_{v_i,c} \in \{0, 1\}$ and $b_{v_i,c} = 1$ indicates that the $Fetcher$ v_i obtains a copy of content item c in its buffer before the item expires time T_e, 0 is the otherwise. Further we define F_c is as the set of $Fetchers$ which keep the copies of content c in there buffers. Formulated as follow:

$$\mathcal{F}_c = \{v_i \in \mathcal{F} | b_{v_i,c} = 1\} \tag{1}$$

If a $Regular$ does not receive the content item that it requests from the $Fetchers$ after the expire time T_e, it will request the content item from the AP directly. Therefore, our objective is to maximize the node utility to obtain as many contents as possible.

$$U(\mathcal{F}) = \sum_{c \in \mathcal{C}} l_c \sum_{v_i \in \mathcal{F}} P_{v_i,c}, \tag{2}$$

where l_c is the size of content item c and $P_{v_i,c}$ is the probability that a $Fetcher$ node v_i obtains content item c before the lifetime T_e. Since the content item lifetime T_e is assigned, we consider the opportunistic contact as the Poisson process with node contact rate β_{v_i,v_j} and the contact event is independent of the node preference event. Thus, integrating the two events, we can model the content item obtaining process by one combing Poisson process with the metric $\beta_{v_i,v_j} p_{v_i,c}$. Next, taking the node caching state B into consideration, for all the $v_i \in \mathcal{F}$, we can derive the probability that a $Regular$ node v_j gets the content item c from the $Fetcher$ node v_i before the lifetime T_e as follows:

$$\begin{aligned} P_{v_i,c} &= \int_0^{\mathrm{T}_e} \beta_{v_i,v_j} p_{v_i,c} e^{-b_{v_i,c}\beta_{v_i,v_j}p_{v_i,c}y} dy \\ &= 1 - e^{-p_{v_i,c}T_e \sum\limits_{v_i \in \mathcal{F}} \beta_{v_i,v_j} b_{v_i,c}} \end{aligned} \tag{3}$$

In order to simplify our formulation, we just consider $b_{v_i,c} = 1$. By substituting $P_{v_i,c}$ into (3), the node utility $U(\mathcal{F})$, can be written as:

$$U(\mathcal{F}) = \sum_{c \in \mathcal{C}} l_c \sum_{v_i \in \mathcal{F}} (1 - e^{-p_{v_i,c}T_e \sum\limits_{v_i \in \mathcal{F}} \beta_{v_i,v_j} b_{v_i,c}}) \tag{4}$$

Specifically, the Fetchers selection problem can be obtained as choosing the nodes with high utility.

Furthermore, the edge node AP can be treat as one of the mobility node with velocity is 0, we assume that the contact probability of mobile nodes v_i and the edge nodes v_e follows Poisson distribution with the density parameter of λ_{v_i,v_e}. Similarly, the probability of content offloading to the edge node AP can be formulated as:

$$P_{v_i,c}^{v_e} = 1 - e^{-p_{v_i,c}T_e \sum\limits_{v_i \in \mathcal{F}} \lambda_{v_i,v_e} m_{v_e,c}}, \tag{5}$$

where $m_{v_e,c} = \{0,1\}$ is the indicator function of the caching state of the edge node AP.

In this paper, the request process of the mobile node-pairs will occur firstly, then is that of "mobile nodes-AP", so the size of edge content offloading problem can be obtained as

$$O(E) = \sum_{c \in \mathcal{C}} l_c \sum_{v \in V} P_O, \tag{6}$$

Then the optimization of edge caching via content offloading can be formulated as:

$$\begin{aligned} &\max O(E) \\ &s.t. \quad b_{v_i,c} \in \{0,1\}, \forall v_i \in \mathcal{F}, \forall c \in \mathcal{C} \\ &\qquad m_{v_e,c} \in \{0,1\}, \forall v \in V, \forall c \in \mathcal{C} \\ &\qquad \sum_{c \in \mathcal{C}} b_{v_i,c} l_c \le L_{v_i}, \forall v_i \in \mathcal{F} \\ &\qquad \sum_{c \in \mathcal{C}} m_{v_e,c} l_c \le L_{v_e}, \forall v \in V \end{aligned} \quad , \tag{7}$$

where $\sum\limits_{c \in \mathcal{C}} m_{v_e,c} l_c \le L_{v_e}, \forall v \in V$ is the edge node buffer size constraint.

B. Fecthers Selection

In the optimization problem (6), all of the constraints are linear and the values of $b_{v_i,c}$ takes 0 or 1. For simplicity, we only consider $b_{v_i,c} = 1$ situation for the analysis.

The main purpose of our work is to determine the optimal subset $\mathcal{F} \le \mathcal{N}$. There exits a greedy algorithm to construct the target set $\mathcal{F}$ if we can prove the optimization problem (6) is submodularity, in which achieves the closed optimum solution with a probability at least $1 - 1/e$. Let $\mathcal{G}$ denote the function mapped from the $Fetcher$ nodes set $\mathcal{F}$ to a nonnegative set. We have the following definition.

Definition 1 (Submodularity): The function $\mathcal{G}$ defined on the universe Ω, $\mathcal{G} : 2^{\Omega} \to R$. We call $\mathcal{G}$ is submodular if and only if $\mathcal{G}(S \cup \{x\}) - \mathcal{G}(S) \ge \mathcal{G}(S' \cup \{x\}) - \mathcal{G}(S')$ for $\forall S \subseteq S' \subseteq \Omega$ and $\forall x \in \Omega \setminus S'$.

Lemma 1: The node utility $U(\mathcal{F}) : 2^{\mathcal{F}} \to R$ is submodular.

Proof: For $S \subseteq S' \subseteq 2^{\mathcal{F}}$ and $v_j \in 2^{\mathcal{F}} \setminus S'$, we get:

$$\begin{aligned} &(U(S \cup \{v_j\}) - U(S)) - (U(S' \cup \{v_j\}) - U(S')) \\ &= \sum_{c \in C} l_c \sum_{v_i \in F} (1 - e^{-p_{v_i,c}T_e b_{v_j,c}\beta_{v_i,v_j}}) \\ &(e^{-p_{v_i,c}T_e \sum\limits_{v_i \in S} \beta_{v_i,v_j}} - e^{-p_{v_i,c}T_e \sum\limits_{v_i \in S'} \beta_{v_i,v_j}}) \\ &= \sum_{c \in C} l_c \sum_{v_i \in F} e^{-p_{v_i,c}T_e \sum\limits_{v_i \in S} \beta_{v_i,v_j}} (1 - e^{-p_{v_i,c}T_e \sum\limits_{v_i \in S' \setminus S} \beta_{v_i,v_j}}) \\ &(1 - e^{-p_{v_i,c}T_e b_{v_j,c}\beta_{v_i,v_j}}) \ge 0 \end{aligned} \tag{8}$$

The submodular of node utility $U(\mathcal{F})$ is proved.

■

C. $Fetcher$ Selection Greedy Algorithm(FSGA)

According to the aforementioned demonstration, we prove the submodular of node utility $U(\mathcal{F})$ which gives raise a greedy algorithm, achieving the optimal solution with a $(1 - 1/e)$-approximation. Here we present the Fetcher Selection Greedy Algorithm(FSGA) to select the $Fetchers$ from $\mathcal{N}$. The core idea of FSGA is to choose the mobile nodes with high utility as $Fetchers$ to aid content item caching.

Algorithm 1 Fetcher Selection Greedy Algorithm(FSGA)

```
Input: A heterogeneous MON $G = (V, E)$
Output: the $Fetcher$ nodes set $\mathcal{F}$
while $k$=0 and $v_k \in \mathcal{N} \setminus S_{k-1}$ do
    $k = k + 1$
    $v_k \leftarrow arg \max_{v_k \in \mathcal{N} \setminus S_{k-1}} (U(S_{k-1} \cup \{v_k\}) - U(S_{k-1}))$
    if $U(S_k) - U(S_{k-1} > 0)$ then
        $S_k = S_{k-1} \cup \{v_k\}$
    else
        Return $S_{k-1}$
    end if
end while
$\mathcal{F} = S_k$
Return $\mathcal{F}$
```

V. CACHING SCHEME

In this section, we present our cooperative caching scheme under HMONs. As the previous introduction, we use the "divide-and-conquer" strategy to allocate content item copies where the more popular content is cached in the $Fetchers$ and AP while less popular content is cached in $Regulars$. The cooperative edge caching workflow is shown as Fig. 3 and the basic idea is proposed as follows:

- Without considering the new content item generation, according to the different content popularity, we push one copy of the top popular content items to $Fetchers$ and the less popular content items are randomly cached in the $Regulars$ in terms of the node buffer constraint. However, the AP's buffer is much larger that mobile nodes', it will cache more content items in the same caching rule.

- When a new content item is generated, because it has never been requested, so the popularity is 0 and its copy will be pushed to $Regulars$ and AP. In general, for AP, content copy allocation process satisfies the time sequence order that it caches prior to the first-coming copy between the new content item and popular content item.
- Content item request will occur between a requesting node and the $Fetchers$ firstly, if the requesting node cannot receive any content item from any $Fetcher$ a certain period time which is related to the content lifetime T_e, it will ask for the content item from AP directly.
- Caching replacement is conducted when two nodes encountered due to the popularity of content item they carried. In order to cache more popular content item in the network, the two encounter nodes will compare the least popularity content item they cached at first. The less popular content item in AP and $Fetchers$ will be replaced by the higher one in $Regulars$.

In this paper, we deploy the popularity of content item as a metric to make caching replacement policy. Due to the content items are placed in the $Fetchers$ and AP uniformly, in the very beginning of the caching process, each $Fetcher$ stores the same content item with the same popularity distribution, so we take following scenarios into account. When a $Regular$ meets a $Fetcher$, they swap the content items information firstly, if a content item with the biggest popularity in $Regular$ is bigger than the smallest one in $Fetcher$, they swap the content item in order to make sure that the popular content items are cached in $Fetchers$. When two $Regulars$ encounter, they don't swap any content item because the content items are randomly cached among $Regulars$. If one $Fetcher$ has no any content items after it forwards all the content to others(this situation indeed exits), when this $Fetcher$ meets AP, it will dump content items in constraints of the buffer size. However, in general, if one $Regular$ has no copies of any content, AP doesn't send any content to it in terms of reducing the traffic pressure of the core network.

VI. DELAY DISCUSSION

In this section, we model the average content transmission delay in the networks and analyze the best and worst condition. We assume that the content delivery process follows "two-hop" transmission, that is, when the requested content is satisfied, the content will be transmitted at most two hops between the resource node and destination node.

In this paper, we consider the process "resource node→mobile relay→destination node" as $Device-to-Device\ Communication\ Phase$ $(D2D)$ and the process "resource node→AP Access Point→destination node" as $Device-to-AP\ Communication\ Phase$ $(D2A)$.

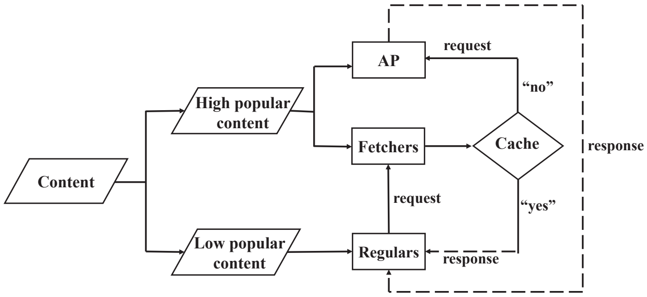


Fig. 3: Cooperative Edge Caching Workflow

Assume that content query process is i.i.d. for each f, $\mathcal{F}=\{f_1,f_2,...,f_M\}$. The distribution of content popularity follows Zipf and each node can only ask for one content at each time during the content query process. Let r_i be the request probability for $i-th$ content, $1\leq i\leq M$ and $\sum_{i=1}^{M} r_i=1$. Moreover, the contact rate of mobile nodes is assumed as β which follows the Possoin distribution. The contact rate between mobile nodes and AP access point is λ. Let T_i be the content access delay of node n for $i-th$ content. The content access delay T_i consists of two parts, namely, content request delay T_{req} and content transmission delay T_{trans}. We have

$$T_i = T_{req} + T_{trans} \tag{9}$$

A. Content Request Delay

In order to model the content request delay, we simply consider that the size of content is the same in the network. Each node is i.i.d. for content and follows the Zipf distribution, let p_i be the probability of content i that cached in the neighbour nodes or AP of the requested node, we have:

$$p_i = \frac{1/i\gamma_c}{\sum_{j=1}^{M} 1/j\gamma_c} \tag{10}$$

Assume that the number of neighbour is N_n, the number of AP is N_{ap},

$$\begin{cases} N_{ap} = \pi r^2 \lambda_1 \\ N_m = \pi r^2 \lambda_2 \end{cases}, \tag{11}$$

where r is the communication range of mobile nodes, λ_1 and λ_2 are the density parameters of spatial poisson distribution of AP access point and mobile nodes. Thus, the number of mobile nodes which cache content i is $N_m P_i$, similarly, the number of AP which has content i is $N_{ap} p_i$. Assuming that $r_i = p_i$, we can calculate the content request delay in D2D communication phase:

$$E_{D2D}[T_{req}] = r_i \frac{1}{\beta N_m p_i}(\frac{\beta}{\beta+\lambda}) = \frac{1}{N_m}(\frac{1}{\beta+\lambda}) \tag{12}$$

D2A communication phase is established based on the failure of D2D communication phase. Similarly, the expectation of content request delay can be expressed as:

$$E_{D2A}[T_{req}] = r_i \frac{1}{\lambda N_{ap} p_i}(\frac{\lambda}{\beta+\lambda}) = \frac{1}{N_{ap}}(\frac{1}{\beta+\lambda}) \tag{13}$$

In conclusion, the expectation of content request delay can be obtained as:

$$\begin{aligned} E[T_{req}] &= E_{D2D}[T_{req}] + E_{D2A}[T_{req}] \\ &= \frac{1}{\beta+\lambda}(\frac{1}{N_{ap}} + \frac{1}{N_m}) \end{aligned} \tag{14}$$

B. Content Transmission Delay

In this paper, we just analyze content transmission delay within "two-hop" delivery. We discuss the transmission delay under D2D communication phase, when the request is satisfied, the cache node forwards the content to some relay, and the relay will not transmit the content to others until it meets the destination node. Similar with the existing work [20], we use Markov chain to analyze the "two-hop" epidemic transmission process.

Assuming that $n_{I(t)}$ is the infected nodes within time t, N is the total number of nodes in network, β is the contact rate of mobile nodes, the transmit probability $r_{N(nI)}$ from state n_I to $n_I + 1$ is:

$$r_N(n_I) = \beta n_I(N - n_I) \tag{15}$$

Let $\gamma = N\beta$, the Eq. (15) can be rewritten in the density dependent form:

$$r_N(n_I) = N\gamma(n_I/N)(1 - n_I/N) \tag{16}$$

According to [21], (n_I/N) is asymptotic convergence when N increases and the solution is:

$$i'(t) = \gamma i(t)(1 - i(t)), t \geq 0 \tag{17}$$

The initial condition of Eq. (15) is $i'(t) = \gamma i(t)(1 - i(t)), t \geq 0$, when the initial condition is $I(0) = Ni(0)$, the number of average infected nodes is $I(t) = Ni(t)$. The Eq. (15) is rewrote as:

$$I'(t) = \beta I(N - I) \tag{18}$$

Let T_d^1 be the transmission delay in that condition, its cumulative distribution function is $P(t) = Prob(T_d^1 < t)$, according to Eq. (17), we have $P(t) : P'(t) = \gamma i(t)(1-P(t))$, where $i(t)$ is the solution of Eq. (17).

Let $P_N(t)$ be the distribution function of $N + 1$ condition, we have:

$$\begin{aligned} & P_N(t + dt) - P_N(t) \\ &= Prob\{t \leq T_d^1 < t + dt\} \\ &= E[Prob\{P_I(t) \in [t, t + dt]|n_I(t)]\} \times (1 - P_N(t))] \\ &= E[\beta n_I(t)](1 - P_N(t))dt \\ &= \gamma E[\frac{n_I(t)}{N}](1 - P_N(t))dt \end{aligned} \tag{19}$$

and

$$\frac{dP_N(t)}{dt} = \gamma E[\frac{n_I(t)}{N}](1 - P_N(t)) \tag{20}$$

As for the fixed number of nodes N, we can obtain that:

$$P'(t) = \beta I(t)(1 - P(t)) \tag{21}$$

When the initial condition is $I(0) = 1$, $P(0 = 0)$, we have:

$$I(t) = \frac{N}{1 + (N - 1)e^{-\beta N i}} \tag{22}$$

$$P(t) = 1 - \frac{N}{N - 1 + e^{\beta N i}} \tag{23}$$

According to Eq. (21), we can obtain that the average expectation transmission delay of T_d^1 is

$$E_{D2D}[T_d^1] = \int\limits_0^\infty (1 - P(t))dt = {}^{\ln N}\!/_{\beta(N - 1)} \tag{24}$$

However, there is one cache node can infect other relay nodes under the "two-hop" transmission model. Thus, the Eq. (18) can be rewritten as

$$I'(t) = \beta(N_M - I), \tag{25}$$

where N_M is the number of infected nodes.

We have the following two equations to model the "two-hop" transmission,

$$I(t) = N_M - (N_M - 1)e^{-\beta t} \tag{26}$$

Similarly, we have

$$P(t) = 1 \text{ - } e^{-\beta N t}e^{-\beta t} \tag{27}$$

In this condition, the solution is determined by infected nodes $I(t)$ and cumulative distribution function $P(t) = \text{Prob}(T_d^2 < t) = 1 - Q(t)$, that is

$$E_{D2D}[T_d^m] = \int_0^\infty Q(t)dt \tag{28}$$

Namely, we can obtain the average expectation "two-hop" transmission delay at the D2D communication phase.

$$E_{D2D}[T_d^m] \approx \frac{1}{\beta}\int_0^\infty e^{-(N_M-1)t^2/2}dt = \frac{1}{\beta}\sqrt{\frac{\pi}{2}}\frac{1}{\sqrt{N_M - 1}} \tag{29}$$

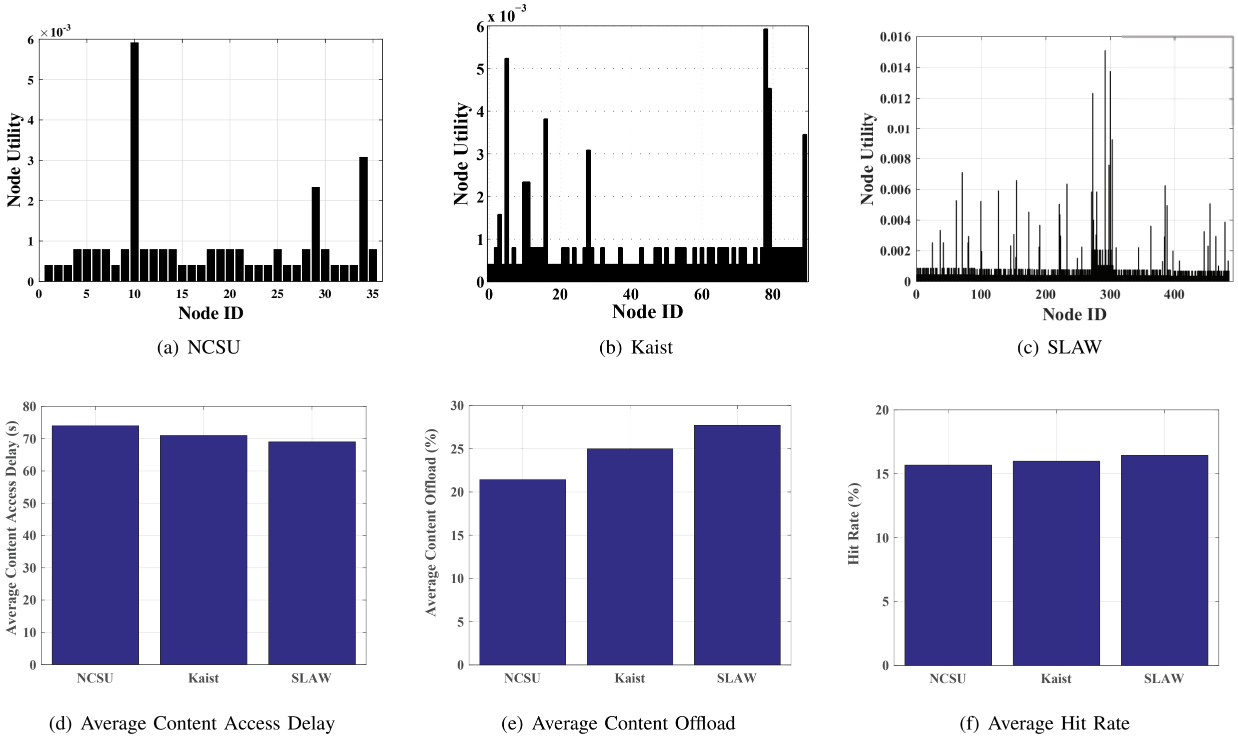
In the D2A communication phase, similar with Eq. (26) and Eq. (27), because the contact rate of mobile nodes and AP is λ, it gets

$$I(t) = N_{AP} - (N_{AP} - 1)e^{-\lambda t} \tag{30}$$

$$P(t) = 1 \text{ - } e^{-\lambda N t}e^{-\lambda t} \tag{31}$$

Then the average content transmission delay of D2A communication phase can be expressed as

$$\begin{aligned} E_{D2A}[T_d^{AP}] &\approx \frac{1}{\lambda}\int_0^\infty e^{-(N_{AP}-1)t^2/2}dt \\ &= \frac{1}{\lambda}\sqrt{\frac{\pi}{2}}\frac{1}{\sqrt{N_{AP} - 1}} \end{aligned} \tag{32}$$



(a) NCSU

(b) Kaist

(c) SLAW

(d) Average Content Access Delay

(e) Average Content Offload

(f) Average Hit Rate

Fig. 4: Node Utility Evaluation (4(a)-4(c)) and Average Content Access Delay, Average Content Offload and Hit Rate for NCSU, Kaist and SLAW (4(d)-4(f)).

In conclusion, the average expectation delay of "two-hop" transmission is

$$E[T_{trans}] = \begin{cases} E_{D2D}[T_d^m] = \dfrac{1}{\beta}\sqrt{\dfrac{\pi}{2}}\dfrac{1}{\sqrt{N_M-1}} \\ E_{D2A}[T_d^{AP}] = \dfrac{1}{\lambda}\sqrt{\dfrac{\pi}{2}}\dfrac{1}{\sqrt{N_{AP}-1}} \end{cases} \quad (33)$$

Finally, the average expectation of content access delay can be summarised as

$$\begin{aligned} &E[T_i] = E[T_{req}] + E[T_{trans}] \\ &= \begin{cases} \dfrac{1}{\beta+\lambda}(\dfrac{1}{N_{ap}} + \dfrac{1}{N_m}) + \dfrac{1}{\beta}\sqrt{\dfrac{\pi}{2}}\dfrac{1}{\sqrt{N_M-1}} \\ \dfrac{1}{\beta+\lambda}(\dfrac{1}{N_{ap}} + \dfrac{1}{N_m}) + \dfrac{1}{\lambda}\sqrt{\dfrac{\pi}{2}}\dfrac{1}{\sqrt{N_{AP}-1}} \end{cases} \quad (34) \end{aligned}$$

VII. PERFORMANCE EVALUATION

In this section, we evaluate the performance of proposed cooperative caching scheme of the NCSU, Kaist [15] and SLAW [16] real content traces. NCSU data set records the daily activities of 2 or 3 students with mobile devices which are randomly selected at each week and collects 35 trajectories from the State University of North Carolina. Kaist records the daily traces of human in the campus, 34 volunteers carried the smart devices with GPS for 92 days. SLAW includes 226 traces collected from 101 volunteers with GPS mobile devices from five different locations for five five hours.

The simulation environment is conducted by .Net platform, the simulation time is $15,000$s for NCSU/Kaist and $18,000$s for SLAW, the simulation field size is $600 \times 600\ m^2$, the node number is set as $35,\ 90$ and 500 for NCSU, Kaist and SLAW, respectively. The transmission range is $25m$, the storage size of node is limited to $20MB$ and the content storage size is $[0.5, 1]MB$, TTL of the message is $300s$ and the results are the average values from 100 times simulations.

Fig. 4 shows the $Fetchers$ selection distribution of NCSU, Kaist and SLAW as well as in terms of average content delay, offload and hit rate. From Fig. 4(a)-4(c), it is obvious to see that only small nodes are with higher node utility than the others and these nodes support the feasibility of our proposed $FSGA$ algorithm. In this way, we choose 3, 10 and 50 nodes as the $Fetchers$ for NCSU, Kaist and SLAW, respectively.

We calculate the average content access delay for all the satisfied requests, the results are demonstrated in Fig. 4(d). For example, the average content access delay of NCSU is around 70 with regard to the $15,000s$ simulation time and $300s$ TTL. From Fig. 4(e), we can observe that the average content offload are 20%, 25% and 27.4% for NCSU, Kaist and SLAW, respectively. The proposed edge caching strategy achieved the better performance of the average hit rate of the three real data traces, we can see that the average hit rate are all more than 15% in Fig 4(f).

In order to make a comparison, we set up four different caching scenarios for Kaist data set. Shown in Fig. 5 and Fig. 6, we evaluate the average content access delay for Kaist, the proposed caching scheme achieves the better network performance. For example, compared with the AP Caching Only scenario, the average delay of the proposed caching scheme is lower from Fig. 5. It is mainly because that the proposed scheme shortens the request transmission time from mobile users to edge node AP by considering cooperative caching among mobile nodes.

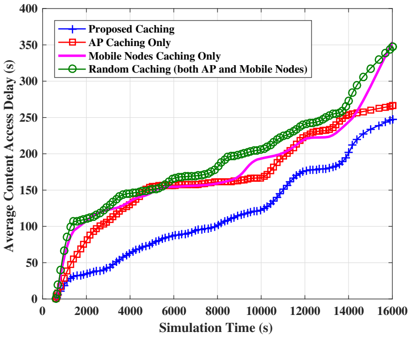


Fig. 5: Average Content Access Delay for Kaist

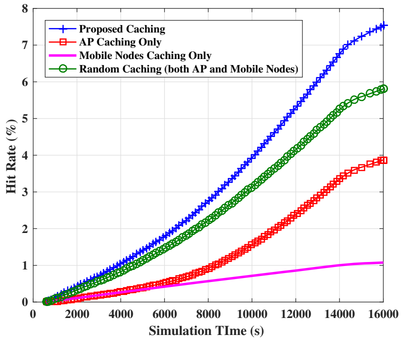


Fig. 6: Average Hit Rate for Kaist

From Fig. 6, the caching success ratio performance of our proposed caching scheme is $7x$ than that of the random caching only among mobile nodes, it also achieves almost $3.5x$ improvement over AP Caching Only scenario as well as $1.6x$ over the random caching scenario (considering both AP and mobile nodes). Specifically, due to the node mobility limitation and random query distribution, the latter three caching scenarios are inefficient.

VIII. Conclusion

In this paper, we study the cooperative caching scheme under heterogeneous mobile opportunistic networks. We model the node preference by considering the content items requests and content popularity, then the node utility is evaluated with Poisson distribution. The average content transmission delay is discussed under different conditions. Our simulation results demonstrate the $Fetchers$ selection distribution and the efficiency of proposed cooperative caching scheme. In the future, we will extend the work in the large scale scenario by considering the cooperative caching among edge nodes (APs), and exploit the caching strategy joint with routing protocol.

Acknowledgment

This work is partially supported by the National Key R&D Program of China (2018YFC0809803), China NSFC (Youth) through grant 61702364, China NSFC GD Joint fund U1701263.

References

[1] Yuan P and Liu P, “Content Fusion Prolongs the Lifetime of Mobile Sensing Networks,” *Journal of Network and Computer Applications,* vol. 49, pp. 51-59, March 2015.

[2] Ma H, Zhao D and Yuan P, “Opportunities in mobile crowd sensing,” *IEEE Communications Magazine*, vol. 52, no. 8, pp. 29-35, Aug. 2014.

[3] Eriksson J., Girod L., Hull B., Newton R., Madden S., and Balakrishnan H., “The Pothole Patrol: Using a Mobile Sensor Network for Road Surface Monitoring”, Proc. ACM Sixth Ann. Intl Conf. Mobile Systems, Applications and Services (MobiSys), 2008.

[4] Spyropoulos T and Psounis K et al, “Efficient Routing in Intermittently Connected Mobile Networks: The Multiple-Copy Case,” *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 77-90, Feb. 2008.

[5] Chen K and Shen H Y, “SMART: Lightweight distributed social map based routing in delay tolerant networks,” In *Proc. IEEE ICNP*, 2012, pp. 1-10.

[6] Yuan P, Ma H and Fu H, “Hotspot-entropy based content forwarding in opportunistic social networks,” *Pervasive and Mobile Computing*, vol. 16, pp. 136-154, Jan. 2015.

[7] Kayastha N, Niyato D, Wang P, et al. “Applications, Architectures, and Protocol Design Issues for Mobile Social Networks: A Survey,” Proceedings of the IEEE, 2011, 99(12):2125-2129.

[8] Gao W, Cao G, Iyengar A, et al. “Cooperative Caching for Efficient content Access in Disruption Tolerant Networks,” IEEE Transactions on Mobile Computing, 2014, 13(3):611-625.

[9] Francisco De Meneses Neves Ramos Dos Santos, Barakat C, Spyropoulos T, et al. “Content Management in Mobile Wireless Networks,” HAL-LIRMM, 2012.

[10] Wang Y, Wu J, Xiao M. “Hierarchical cooperative caching in mobile opportunistic social networks,” Global Communications Conference. IEEE, 2015:411-416.

[11] Zhou H, Wang H, Li X, et al. “A Survey on Mobile Data Offloading Technologies,” IEEE Access, 2018, 6(99):5101-5111.

[12] Park K, Jeong Y S. “A caching strategy for spatial queries in mobile networks,” Journal of Information Science and Engineering, 2014, 30(4):1187-1207.

[13] Sun Y, Chen Z, Liu H. “Delay Analysis and Optimization in Cache-enabled Multi-Cell Cooperative Networks,” arXiv. 2016.

[14] Wang X, Li X, Leung V C M, et al. “A Framework of Cooperative Cell Caching for the Future Mobile Networks,” Hawaii International Conference on System Sciences. IEEE, 2015:5404-5413.

[15] Rhee I, Shin M, Hong S, et al. “On the levy-walk nature of human mobility,” IEEE/ACM transactions on networking, 2011, 19(3): 630-643.

[16] Lee K, Hong S, Kim S J, Rhee I, Chong S. “SLAW: Self-Similar Least-Action HumanWalk,” IEEE/ACM Trans. Netw. 2012, 20, 515-529.

[17] Gao W. and G. Cao, “User-centric data dissemination in disruption tolerant networks,” in Proc. 30th IEEE INFOCOM, Shanghai, China, Apr. 2011, pp. 1-9.

[18] Lee K. et al., “Max-contribution: On optimal resource allocation in delay tolerant networks,” in Proc. 29th IEEE INFOCOM, San Diego, CA, USA, Mar. 2010, pp. 1-9.

[19] Li Y., D. Jin, P. Hui, L. Su, and L. Zeng, “Revealing contact interval patterns in large scale urban vehicular ad hoc networks,” ACM SIGCOMM Comput. Commun. Rev., vol. 42, no. 4, pp. 299-300, 2012.

[20] Groenevelt R, Nain P, Koole G. “The message delay in mobile ad hoc networks,” Performance Evaluation, 2005, 62(1):210-228.

[21] Kurtz T G. “Solutions of ordinary differential equations as limits of pure jump Markov processes,” Journal of applied Probability, 1970, 7(1): 49-58.