Spatially Correlated Placement Policies for Wireless Content Caching Networks

Shukai Chen*, Derya Malak†, and Alhussein A. Abouzeid*
*Rensselaer Polytechnic Institute, Troy, NY, {chens37@rpi.edu, abouzeid@ecse.rpi.edu}
†EURECOM, Biot, Sophia Antipolis, France, derya.malak@eurecom.fr

***Abstract*—We propose a geographic content placement policy for wireless caching networks. This policy, named Joint Caching Policy (JCP), jointly determines the caching strategy across the set of base stations (BSs) to improve the hit probability of an arbitrarily located user in the network versus caching policies where placement is independent and identically distributed across the BSs. To that end, JCP divides the BSs into groups, and executes a joint caching policy for each group, while content placement is independent across the groups. Existing joint caching policies require knowledge of user location to optimize content placement. On the other hand, JCP does not require any information on user location, and it provides a content placement policy that outperforms the one given by Independent Caching Policy (ICP), proposed by Błaszczyszyn and Giovanidis in 2015, under any user location distribution. We prove that the hit probability under JCP is lower bounded by that of ICP. We further propose an extension of JCP, named JCP-OPT, which improves the hit probability over JCP by solving a concave maximization problem, provided that there is side information about the user locations. We validate the performance of JCP and JCP-OPT via numerical evaluations and demonstrate that they can provide up to a 30% gain in hit probability over ICP.**

***Index Terms*—Edge caching, Poisson network, content popularity, hit probability, optimization.**

I. Introduction

With the growth of mobile data traffic by a factor of 2 between 2020 and 2022 [1], the network infrastructure requires costly upgrades to support the demand. Edge caching technology has been used to offload the traffic of wireless networks and alleviate backhaul. Another promising technique is caching at the base stations (BSs) [2], using the readily available infrastructure. Because of limited cache capacities, the BSs can only store part of the entire catalog content [3]. The placement of these content items determines how much gain can be achieved by caching at the BSs.

A. Related Work

There has been a significant amount of work on caching close to the edge. We next review the most related work, on node-based cache replacement policies, followed by algorithms for caching in networks and coded caching.

This material is based upon work supported by the National Science Foundation under Award No. 2008639.

Co-funded by the European Union (ERC, SENSIBILITÉ, 101077361). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

Cache replacement algorithms. The conventional approach to caching, the Most Popular Content (MPC) policy, fills each cache with the most popular items [4]. In the case when the popularity of content is unknown, policies, such as Least Recently Used (LRU) [5], Least Frequently Used (LFU) [6] and Time-To-Live (TTL) [7], have been designed to achieve a placement that performs closely to the one of MPC. Otherwise, if the popularity is known or predicted, there are policies [8]–[17] that outperform MPC in networks.

Caching in networks and distributed algorithms. Building on the replacement policies based on various eviction rules, there have been efforts to optimize the cache hit performance in wireless networks. For instance, the authors of [8], [9], and [10] proposed Independent Caching Policies (ICP), which achieve higher cache hit probabilities compared to MPC. However, these ICP policies are still suboptimal, because they set each BS to cache independently from the other BSs. The authors of [11], [12], [14] proposed joint caching policies that alleviate redundantly storing identical content in nearby caches. There have also been game theory-based [15], Gibbs sampling-based [16], and learning-based algorithms [17] to optimize uncoded content placement to maximize the cache hit probability, which has been proved to be NP-hard [18].

Note that joint content placement across caches outperforms independent placement, as demonstrated by e.g., [11], [12], [14], [19]. However, existing works on joint placement, such as [11], [12], [14]–[17], [20], [21], require knowledge of user location to optimize content placement. In practice, user locations can be unpredictable. *Joint caching without information on user locations is a key novelty of our proposed framework.*

Coded caching in networks. Maddah-Ali and Niesen considered content placement in a network, where users with finite memory share a link to the source server containing all content [22]. The authors discussed the tradeoff between memory and link rate, which led to the idea of coded caching. In coded content caching, a cache divides each content item into small chunks and only stores a subset of these chunks, instead of storing the item in its entirety. Hence, coded caching leverages the fine-grained granularity of joint caching, even without knowledge of user locations. A given user needs to collect a sufficient number of chunks to obtain a complete item. To that end, the authors of [23] considered linear network coding-based content placement in a stochastic network model, but only for a single content item. The authors of [9] further used dynamic programming to optimize coded content placement for multiple items in a stochastic network. In theory, coded caching can significantly improve the memory-bandwidth tradeoff. However, in practice, achieving significant gains via coding requires devising subpacketization mechanisms at fine-grained granularity, which may not be possible. In this paper, we restrict our focus to uncoded content caching policies that always outperform ICP without any knowledge of user locations. This idea can be reinforced using coded caching arguments, which is left as a future direction.

B. Contributions

In this paper, we consider geographic content placement for a wireless caching network for a downlink user that requests content items, where we model the locations of the BSs using a Poisson Point Process (PPP) model where each BS has a finite caching capacity. We divide the BSs into groups and propose low-complexity joint caching policies for each group exploiting the demand distribution of the user and the BS groups. Our main contributions can be summarized as follows:

- We propose a novel caching policy, named Joint Caching Policy (JCP), which divides the BSs into groups and sets each group of the BSs to cache jointly. The achievable hit probability by JCP varies relative to the groupings.
- For our proposed caching network model, we provide a general Upper Bound for Hit probability (UBH) under any caching policy, which is not practically achievable.
- We prove that the hit probability of JCP is lower-bounded by that of ICP, as proposed in [8], [9], and [10].
- We conjecture an upper bound JCP-UB for the hit probability under JCP, which is achieved when the BSs covering the user are always grouped together. We prove that JCP-UB reaches UBH if the items are equally popular.
- We propose an extension, named JCP-OPT, to improve the hit probability over JCP, given that the BSs groups, as well as the joint placement policy per group, are known.
- Our numerical evaluations show that JCP and JCP-OPT can provide up to 30% gains in hit probability over ICP.

C. Organization

The organization of the paper is as follows. In Section II, we introduce the caching network model and describe the cache hit probability for a typical user. In Section III, we recast prior work ICP [8], and provide a general Upper Bound for Hit probability (UBH) under any possible policy. We also propose the JCP policy with its hit probability upper bound JCP-UB. In Section IV, we offer an extension of the JCP policy, namely JCP-OPT, which achieves the upper bound OPT-UB. In Section V, we compare our policies with the state-of-the-art through numerical simulations. In Section VI, we conclude the paper and discuss some possible avenues for future work.

II. Wireless System Model

In this section, we describe our caching network model. We detail the coverage model in Section II-A, and define the cache hit probability for our model in Section II-B.

A. Caching Network Model

We consider a wireless downlink where BSs share the same frequency. The positions of BSs with index $i = 1, 2, 3, ...$ are modeled by a uniform Poisson Point Process (PPP) with intensity λ on a finite 2D plane. Let Φ denote the set of all BSs. Content items of identical size with indices $j \in [n] = \{1, 2, ..., n\}$ are cached by the set of BSs. Each BS can store up to C items. Let binary indicator $X_{i,j} = 1$ denote that BS i caches item j, and $X_{i,j} = 0$ denote that BS i does not cache item j. The variable $X_{i,j}$ obeys the BS cache capacity constraint:

$$\sum_{j \in [n]} X_{i,j} \le C, \quad \forall i \in \Phi \ . \tag{1}$$

In this paper, we consider one typical user at the origin point o. A subset Φ_T of Φ BSs covers the user. The user requests one item in the item catalog. The probability of an item being selected obeys Zipf's law. Let p_j denote the probability that the user requests item $j \in [n]$. With normalization coefficient c_1 and Zipf's exponent ξ, p_j is given by [24]

$$p_j = c_1^{-1} j^{-\xi}, \quad j = 1, 2, ..., n \ . \tag{2}$$

If the set of BSs Φ_T caches the requested item, a 'hit' will be generated, otherwise, there will be a 'miss'.

According to the Slivnyak-Mecke theorem, the reduced Palm distribution of a PPP is identical at all points [25], which means the properties of a PPP are independent of the observation position. Therefore, we can apply the results of the typical user to any user on the plane.

We use the Boolean model, see e.g., [25], to model the coverage region of the user, which sets the maximum coverage distance to be a constant R. The coverage area is the circle $O(o, R)$ with a radius R for a user that is centered at o. We denote by Φ_T the set of BSs inside $O(o, R)$, and the set of BSs outside $O(o, R)$ are in $\Phi \backslash \Phi_T$. The BS coverage number $|\Phi_T|$ follows a Poisson distribution with density $\Lambda = \lambda \pi R^2$ [25]. Denoting by $\mathbb{P}[Y = y]$ the probability of Y taking the value y, the probability that $|\Phi_T| = m$ is

$$\mathbb{P}[|\Phi_T| = m] = a_m = \frac{e^{-\Lambda} \Lambda^m}{m!}, \quad m = 0, 1, 2... \tag{3}$$

B. Cache Hit Probability

The cache hit probability is the key metric for measuring network performance. Let $X_{\Phi_T, j}$ denote vector $[X_{i,j}]_{i \in \Phi_T}$. The probability of a 'miss' for item j is

$$\mathbb{P}[X_{i,j} = 0, \forall i \in \Phi_T] = \mathbb{P}[X_{\Phi_T, j} = 0] \ . \tag{4}$$

Let J be a random variable that denotes the index of the requested item, which satisfies $\mathbb{P}[J = j] = p_j, \quad \forall j \in [n]$. The probability of 'cache hit' averaged over Φ_T and the requested item J is

$$\begin{aligned} P_{hit} &= 1 - \mathbb{E}_{\Phi_T, J}[\mathbb{P}[X_{\Phi_T, J} = 0]] \\ &= 1 - \sum_{m=0}^{\infty} a_m \sum_{j \in [n]} p_j \mathbb{P}[X_{\Phi_T, j} = 0 \mid |\Phi_T| = m] \end{aligned} \tag{5}$$

where the probabilities $\mathbb{P}[X_{\Phi_T,j} = 0 \mid |\Phi_T| = m]$ are average weights in deterministic network models. We list the important notations of Section II in Table I. To improve the hit probability expressed in (5), various caching policies are proposed, which we discuss next.

III. Caching Policies

In this section, we introduce different content placement policies. We first recast the prior work [8] utilizing our notation, and summarize their main result. Then, we propose a general Upper Bound for Hit probability (UBH) in Section III-B under any caching policy. We propose Joint Caching Policy (JCP) in Section III-C, give its conjectured upper bound JCP-UB in Section III-D, and prove that JCP outperforms ICP in Section III-E.

A. Independent Caching Policy

Under Independent Caching Policy (ICP) [8], each BS caches item $j \in [n]$ with a probability q_j. The caching probabilities $\{q_j\}_{j\in[n]}$ satisfy a probabilistic capacity constraint

$$\sum_{j\in[n]} q_j \le C \ . \tag{6}$$

The authors of [8] showed that when (6) is satisfied, the original capacity constraint in (1) is also satisfied.

Under ICP, each BS caches independently, so the miss probability for item $j \in [n]$ and coverage number $m \in [0, \infty)$ is $(1-q_j)^m$. The hit probability under ICP is defined as [8]

$$P_{ICP} = 1 - \sum_{m=0}^{\infty} a_m \sum_{j\in[n]} p_j (1-q_j)^m \ . \tag{7}$$

The authors of [8] proved that P_{ICP} is a concave function of caching probability vector $\{q_j\}_{j\in[n]}$, and give the hit probability under ICP by solving a concave maximization problem with constraint (6)

$$\text{ICP: } \underset{\{q_j\}_{j\in[n]}}{\text{Maximize:}} \quad P_{ICP}(\{q_j\}_{j\in[n]}) \tag{8a}$$

$$\text{s.t.} \quad \sum_{j\in[n]} q_j \le C \ . \tag{8b}$$

We note that ICP always achieves a higher hit probability than MPC, as proven in [8].

Notation	Description
λ, Φ, Φ_T	Intensity of PPP modeling the BS positions, set of all BSs and set of BSs covering the user
a_m	Probability that BS coverage is m
n	Item catalogue size
i, j	BS index $i \in \mathbb{N}^+$, item index $j \in [n]$
R	Coverage distance in Boolean model
p_j	Popularity of item j
ξ	Exponent in Zipf's law
J	Random variable satisfying $\mathbb{P}[J = j] = p_j$
q_j	probability that item j is cached in a BS
C	Capacity constraint per BS
$X_{i,j}$	Binary indicator of item j being cached by BS i
$X_{\Phi_T,j}$	Vector of $X_{i,j}, i \in \Phi_T$. $X_{\Phi_T,j} = [X_{i,j}]_{i\in\Phi_T}$

Table I: Notation.

B. General Upper Bound for Hit Probability

We derive a general Upper Bound for Hit Probability (UBH) in our network model for any possible caching policy. Hit probability is the probability of 'cache hit' averaged over different coverage numbers. When there are m BSs covering the user, to maximize the probability of 'hit', we can regard these m BSs as a joint large cache with capacity $m \cdot C$, and fill this large cache with the most popular items. Assuming that for any coverage number m, BSs in Φ_T are always filled with the most popular items jointly, we obtain UBH

$$P_{UBH} = \sum_{m=0}^{\infty} a_m \sum_{j\in[min\{mC,n\}]} p_j \ . \tag{9}$$

We next prove that the hit probability defined in (5) is upper bounded by UBH.

Proposition 1. *UBH gives an upper bound on the cache hit probability as*

$$P_{hit} \le P_{UBH} \ . \tag{10}$$

Proof. See proof of Proposition 1 in our report [26]. □

In our network model, the coverage set Φ_T is randomly generated, so without further restrictions, no policy can guarantee that Φ_T is jointly filled with the most popular items, and therefore UBH is not achievable in practice.

C. Joint Caching Policy

Under Joint Caching Policy (JCP), to obtain a gain from the joint placement, we divide the Φ BSs into groups and set the BSs in the same group cache jointly. The caching policy for the BSs in different groups is independent. We do not restrict which BSs to be grouped together.

In each group, the joint placement is done according to caching probabilities $\{q_j^I\}_{j\in[n]}$, obtained as the solution of ICP optimization problem in (8), where I represents ICP. In this setting, when we group l BSs together, we regard these l BSs as a large unified cache with capacity lC. In this unified cache, items can be repeatedly stored. Our goal is to first determine the number of item copies to be stored in this unified cache. To that end, we first draw lC empty bars with length 1, which is shown in Fig. 1. Then, we fill n colored bars with lengths from lq_1^I to lq_n^I into the empty bars one by one. If a colored bar exceeds the empty bar, we put the rest of the colored bar into the next empty bar. Because $\{q_j^I\}_{j\in[n]}$ satisfies probabilistic

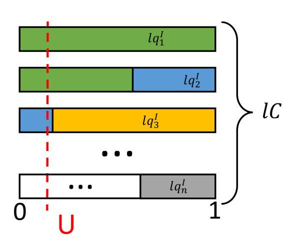


Figure 1: Illustration for describing caching policy under JCP, where $U \sim \text{Unif}[0, 1]$, l is the BS group size, and C is the cache capacity.

capacity constraint (6), all colored bars are filled into empty bars without overflow. After this, we uniformly generate a random number U between 0 and 1, and draw a line at U. Every time the line crosses a colored bar, the unified cache stores one more copy of the corresponding item. As a result, there will be at most lC copies of items to be stored.

After the number of copies for each content item has been determined, we distribute the item copies to the l BSs that compose this unified cache. Denote the copy amount of item j by h_j. We detail this algorithm below (Algorithm 1).

Algorithm 1 Joint placement algorithm for each BS group

```
Randomly index the BSs with i = 1, ..., l.
for j = 1, ..., n do
    Set indicator h = 0
    for i = 1, ..., l do
        while h < h_j do
            if BS i has spare capacity ∑_{j∈[n]} X_{i,j} < C then
                BS i ← a copy of item j
                h = h + 1
            end if
        end while
    end for
end for
```

We next define the hit probability under JCP as a function of the caching probabilities. We first set

$$Q_{l,q_j^I} = \lfloor l q_j^I \rfloor, \quad Q \in \mathbb{N} \ , \tag{11}$$

$$x_{l,q_j^I} = Q_{l,q_j^I} + 1 - l q_j^I, \quad x \in (0,1] \ . \tag{12}$$

Under JCP, for a BS group with size l, Q_{l,q_j^I} BSs cache item j with a probability of x_{l,q_j^I}, and $(Q_{l,q_j^I}+1)$ BSs cache item j with a probability of $(1-x_{l,q_j^I})$.

If we randomly pick $k \in [l]$ BSs from the l BSs in the group, the probability that these k BSs miss item j is a function of k, l, and q_j^I, which we determine next.

Proposition 2. *Under JCP, for k BSs from a group with size l, the probability of missing item j is*

$$\begin{aligned} &P_{miss}(k,l,q_j^I) \\ =&x_{l,q_j^I} \cdot \prod_{r=0}^{k-1} max\{1-\frac{Q_{l,q_j^I}}{l-r},0\} \\ &+(1-x_{l,q_j^I}) \cdot \prod_{r=0}^{k-1} max\{1-\frac{Q_{l,q_j^I}+1}{l-r},0\} \ . \end{aligned} \tag{13}$$

Proof. See proof of Proposition 2 in [26]. □

Because Φ_T is randomly generated from Φ in our network model, the BSs in Φ_T may come from different groups, e.g., blue or yellow, as shown in Fig. 2-(c). In set Φ_T, for a subset of all BSs from a same group, e.g., the set of blue or the set of yellow BSs, we denote the size of the subset by k, e.g., $k=1$ for each group, i.e., $|\Phi_T|=2$, and the size of the group by l, e.g., $l=3$ for each group. The union of the groups gives the set of all BSs Φ. We denote by $b_{k,l}$ the number of the subsets with identical (k,l) pair. We use a matrix $\mathbf{B}=[b_{k,l}]_{k,l\in[l']}$ to denote how BSs in Φ_T are grouped, where l' is the largest group size. Because Φ_T is random, $\mathbf{B}$ is a random matrix. When $|\Phi_T|=m$, the possible realizations of matrix $\mathbf{B}$ satisfy

$$\sum_{k,l\in[l']} k \cdot b_{k,l} = m \ . \tag{14}$$

We denote the set of all realizations of $\mathbf{B}$ that satisfy (14) by $\mathcal{B}_m$. Let $P_G(m,B')$ denote the probability of realization $B'=[b'_{k,l}]_{k,l\in[l']}$ in $\mathcal{B}_m$, which is given by

$$P_G(m,B') = \mathbb{P}[\mathbf{B}=B' \mid B' \in \mathcal{B}_m] \ . \tag{15}$$

Therefore, the probability that the set Φ_T with size m misses item j under JCP is given as

$$\begin{aligned} &\mathbb{P}[X_{\Phi_T,j}=0 \mid |\Phi_T|=m] \\ =&\sum_{B'\in\mathcal{B}_m} P_G(m,B') \prod_{k,l\in[l']} P_{miss}(k,l,q_j^I)^{b'_{k,l}} \ . \end{aligned} \tag{16}$$

Substituting (16) into (5) gives the hit probability under JCP:

$$\begin{aligned} \text{JCP: } &P_{JCP}(\{q_j^I\}_{j\in[n]}) \\ =&1-\sum_{m=0}^{\infty} a_m \sum_{j\in[n]} p_j \sum_{B'\in\mathcal{B}_m} P_G(m,B') \\ &\prod_{k,l\in[l']} P_{miss}(k,l,q_j^I)^{b'_{k,l}} \ . \end{aligned} \tag{17}$$

D. Joint Caching Policy Upper Bound

As shown in (17), the hit probability under JCP depends on how BSs in Φ_T are grouped. We conjecture an upper bound for the hit probability under JCP, named JCP-UB, by setting that BSs in Φ_T always compose a group. Then, for a given coverage number m, the probability that the set Φ_T misses item j under JCP-UB is

$$\mathbb{P}[X_{\Phi_T,j}=0 \mid |\Phi_T|=m] = max\{1-m\cdot q_j^I, 0\} \ . \tag{18}$$

Substituting (18) into (5) gives JCP-UB

$$\begin{aligned} &P_{JCP-UB}(\{q_j^I\}_{j\in[n]}) \\ =&1-\sum_{m=0}^{\infty} a_m \sum_{j\in[n]} p_j max\{1-m\cdot q_j^I, 0\} \ . \end{aligned} \tag{19}$$

We conjecture that the hit probability under JCP is upper bounded by JCP-UB.

Conjecture 1. *The hit probability under JCP is upper bounded by JCP-UB as follows:*

$$P_{JCP}(\{q_j^I\}_{j\in[n]}) \le P_{JCP-UB}(\{q_j^I\}_{j\in[n]}) \ . \tag{20}$$

We also find that JCP-UB equals UBH when items are equally popular, which is given next.

Corollary 1.1. *When the item popularity is uniform, i.e., $p_j = \frac{1}{n}$, $\forall j \in [n]$, JCP-UB equals UBH*

$$P_{JCP-UB}(\{q_j^I\}_{j\in[n]}) = P_{UBH} \ , \tag{21}$$

where the right-hand side is given in (9).

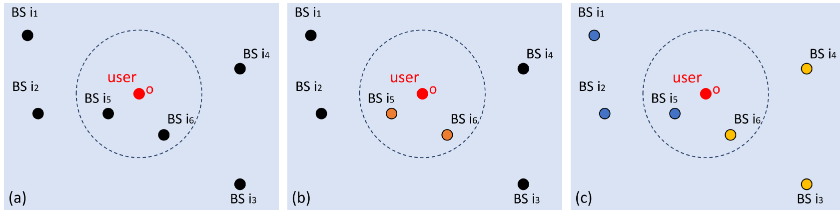


Figure 2: Under Independent Caching Policy (ICP) [8] as shown in (a), the BSs cache independently. Under JCP-UB as shown in (b), the BSs in Φ_T cache jointly. Under JCP as shown in (c), the BSs are grouped. BSs i_1,i_2, and i_5 compose a group, while BSs i_3, i_4, and i_6 form another group. Only the BSs in the same group cache jointly.

Proof. See proof of Corollary 0.1 in [26]. □

E. Joint Caching Policy Lower Bound

We prove that JCP always outperforms ICP.

Theorem 1. *The hit probability under JCP is lower bounded by that of ICP as*

$$P_{JCP}(\{q_j^I\}_{j\in[n]}) \geq P_{ICP}(\{q_j^I\}_{j\in[n]}) \ . \tag{22}$$

Proof. See proof of Theorem 1 in [26]. □

IV. Extension of JCP

In this section, we propose an extension of the JCP policy, named JCP-OPT, and give its conjectured upper bound OPT-UB. Under JCP, we do not have knowledge about the distribution of random matrix $\mathbf{B}$. However, there can be situations in which we obtain some information on $\mathbf{B}$.

Under the assumption that the function P_G is known, we propose JCP-OPT, which improves JCP by solving an optimization problem with a probabilistic capacity constraint, as given next.

$$\text{JCP-OPT: } \underset{\{q_j\}_{j\in[n]}}{\text{Maximize:}} \quad P_{JCP}(\{q_j\}_{j\in[n]}) \tag{23a}$$

$$\text{s.t.} \quad \sum_{j\in[n]} q_j \leq C \ . \tag{23b}$$

We prove that the optimization problem (23) is a concave maximization problem.

Theorem 2. *The JCP-OPT optimization problem (23) is a concave maximization problem.*

Proof. See proofs of Proposition 3, Proposition 4, and Theorem 2 in [26]. □

The upper bound for JCP-OPT can be found by solving a similar concave maximization problem:

$$\text{OPT-UB: } \underset{\{q_j\}_{j\in[n]}}{\text{Maximize:}} \quad P_{JCP-UB}(\{q_j\}_{j\in[n]}) \tag{24a}$$

$$\text{s.t.} \quad \sum_{j\in[n]} q_j \leq C \ . \tag{24b}$$

We denote the maximal hit probabilities under JCP-OPT and OPT-UB by P^*_{JCP} and P^*_{JCP-UB}, respectively. Similar to Conjecture 1, we conjecture that JCP-OPT is upper bounded by OPT-UB.

Conjecture 2. *JCP-OPT is upper bounded by OPT-UB as*

$$P^*_{JCP} \leq P^*_{JCP-UB} \ . \tag{25}$$

OPT-UB equals UBH when items are equally popular.

Corollary 2.1. *When the item popularity is uniform, i.e.,* $p_j = \frac{1}{n}$, $\forall j \in [n]$, *OPT-UB equals UBH:*

$$P^*_{JCP-UB} = P_{UBH} \ . \tag{26}$$

Proof. See proof of Corollary 2.1 in [26]. □

Fig. 2 briefly shows the differences between ICP, JCP and JCP-UB.

V. Numerical Results

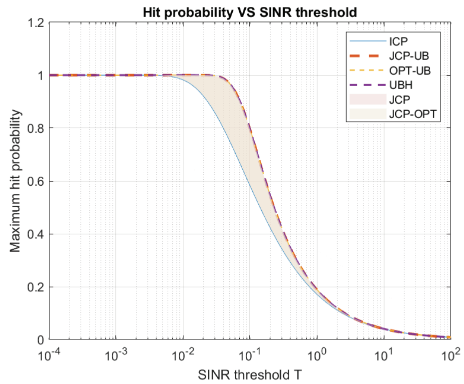
In this section, we compare the hit probabilities achieved by ICP, JCP-UB, JCP-OPT, and UBH, under equal and unequal item popularity conditions.

We first consider the case that all items are equally popular. We set $\xi = 0$, $\lambda = 0.5$, $c_2 = 1$, $\alpha = 3$, $n = 25$, and $C = 3$. We show the hit probabilities achieved under different policies versus SINR threshold T in Fig. 3-(a). As proved in Corollaries 1.1 and 2.1, both JCP-UB and OPT-UB are equal to UBH. The hit probability of JCP is located in the shaded region between ICP and JCP-UB, and the hit probability of JCP-OPT (the shaded region) lies between the curves corresponding to ICP and OPT-UB. Both JCP and JCP-OPT provide up to a 30% gain in hit probability over ICP.

We next consider the case that the items are not equally popular. We set $\xi = 0.56$, and keep all other parameters the same as in the previous paragraph. We illustrate the hit probabilities achieved under different policies versus the SINR threshold T in Fig. 3-(b). As shown in the figure, JCP-UB and OPT-UB perform worse than UBH, but always better than ICP. The hit probability of JCP (the shaded region) lies between the curves of JCP-UB and ICP, and the hit probability of JCP-OPT is located in the shaded region between OPT-UB and ICP. Both JCP and JCP-OPT provide up to a 25% gain in hit probability over ICP.

VI. Conclusions

In this paper, we considered a hit probability maximization problem in a PPP-modeled cellular network. Drawing on previous work ICP [8], we proposed a new caching policy, JCP, which increases the hit probability regardless of user location.



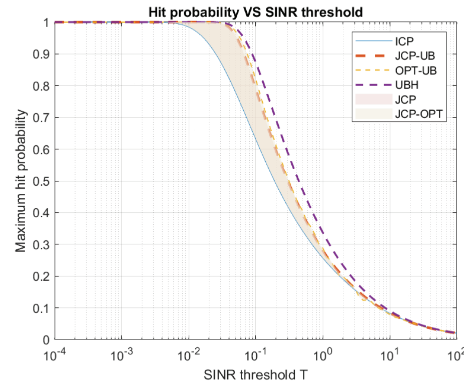


Figure 3: (Left) Cache hit probability versus SINR threshold when content items have equal popularity. (Right) Cache hit probability versus SINR threshold, where Zipf's exponent is given as $\xi = 0.56$.

We further proposed the JCP-OPT policy, which improves JCP by solving a concave maximization problem, yet requires extra information on how the BSs are grouped. We also provided UBH, which is a general upper bound to the hit probability under any caching policy. We conjectured upper bounds for JCP and JCP-OPT and proved that the cache hit performances of both policies are lower bounded by that provided by ICP. We also proved that the upper bounds for the two policies reach UBH when the content items are equally popular.

The potential future directions include the followings. First, the hit probabilities under JCP and JCP-OPT are impacted by how the BSs are grouped. The BS grouping policy is an interesting topic. Second, item popularity is dynamic in practice, and a policy is required to efficiently count the item popularity. Considering the hit probability achieved with expired cache items is also part of potential future work. Third, we only considered the case of a single user in this paper, but the multi-user situation is more practical. Finally, because the network has a distributed nature, a distributed algorithm needs to be proposed to implement joint content placement in each group under JCP and JCP-OPT.

References

[1] Ericsson, "Ericsson mobility report," 2022. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/mobility-report

[2] G. Paschos, G. Iosifidis, G. Caire *et al.*, "Cache optimization models and algorithms," *Foundations and Trends® in Communications and Information Theory*, vol. 16, no. 3–4, pp. 156–345, Aug. 2020.

[3] G. Paschos, E. Bastug, I. Land, G. Caire, and M. Debbah, "Wireless caching: Technical misconceptions and business barriers," *IEEE Communications Magazine*, vol. 54, no. 8, pp. 16–22, Aug. 2016.

[4] A. Bura, D. Rengarajan, D. Kalathil, S. Shakkottai, and J.-F. Chamberland, "Learning to cache and caching to learn: Regret analysis of caching algorithms," *IEEE/ACM Transactions on Networking*, vol. 30, no. 1, pp. 18–31, Aug. 2021.

[5] N. Gast and B. Van Houdt, "Asymptotically exact TTL-approximations of the cache replacement algorithms LRU (m) and h-LRU," in *Proc., International Teletraffic Congress*, vol. 1, Würzburg, Germany, Sep. 2016, pp. 157–165.

[6] G. Einziger, R. Friedman, and B. Manes, "Tinylfu: A highly efficient cache admission policy," *ACM Transactions on Storage*, vol. 13, no. 4, pp. 1–31, Nov. 2017.

[7] S. Basu, A. Sundarrajan, J. Ghaderi, S. Shakkottai, and R. Sitaraman, "Adaptive TTL-based caching for content delivery," *IEEE/ACM Transactions on Networking*, vol. 26, no. 3, pp. 1063–1077, Apr 2018.

[8] B. Blaszczyszyn and A. Giovanidis, "Optimal geographic caching in cellular networks," in *Proc., IEEE International Conference on Communications*, London, UK, Jun. 2015, pp. 3358–3363.

[9] K. Avrachenkov, X. Bai, and J. Goseling, "Optimization of caching devices with geometric constraints," *Performance Evaluation*, vol. 113, pp. 68–82, May 2017.

[10] X. Lin, J. Xia, and Z. Wang, "Probabilistic caching placement in UAV-assisted heterogeneous wireless networks," *Physical Communication*, vol. 33, pp. 54–61, Jan. 2019.

[11] A. Giovanidis and A. Avranas, "Spatial multi-LRU caching for wireless networks with coverage overlaps," *ACM SIGMETRICS Performance Evaluation Review*, vol. 44, no. 1, pp. 403–405, Jun. 2016.

[12] D. Malak, M. Al-Shalash, and J. G. Andrews, "Spatially correlated content caching for device-to-device communications," *IEEE Transactions on Wireless Communications*, vol. 17, no. 1, pp. 56–70, Oct. 2017.

[13] S. Shukla, O. Bhardwaj, A. A. Abouzeid, T. Salonidis, and T. He, "Proactive retention-aware caching with multi-path routing for wireless edge networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 6, pp. 1286–1299, Jun. 2018.

[14] D. Malak, M. Médard, and J. G. Andrews, "Spatial concentration of caching in wireless heterogeneous networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3397–3414, Jan. 2021.

[15] K. Avrachenkov, J. Goseling, and B. Serbetci, "A low-complexity approach to distributed cooperative caching with geographic constraints," *Proc., ACM on Measurement and Analysis of Computing Systems*, vol. 1, no. 1, pp. 1–25, Jun. 2017.

[16] A. Chattopadhyay, B. Błaszczyszyn, and H. P. Keeler, "Gibbsian on-line distributed content caching strategy for cellular networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 2, pp. 969–981, Nov. 2017.

[17] X. Lin, Y. Tang, X. Lei, J. Xia, Q. Zhou, H. Wu, and L. Fan, "MARL-based distributed cache placement for wireless networks," *IEEE Access*, vol. 7, pp. 62 606–62 615, May 2019.

[18] 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, Sep. 2013.

[19] M. A. Maddah-Ali and U. Niesen, "Cache-aided interference channels," *IEEE Transactions on Information Theory*, vol. 65, no. 3, pp. 1714–1724, Feb. 2019.

[20] T. Zhou, S. Yang, L. Wang, J. Yao, and G. Gui, "Improved cross-label suppression dictionary learning for face recognition," *IEEE Access*, vol. 6, pp. 48 716–48 725, Aug. 2018.

[21] Y. Li, X. Cheng, and G. Gui, "Co-robust-ADMM-net: Joint ADMM framework and DNN for robust sparse composite regularization," *IEEE Access*, vol. 6, pp. 47 943–47 952, Aug. 2018.

[22] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 2856–2867, Mar. 2014.

[23] E. Altman, K. Avrachenkov, and J. Goseling, "Distributed storage in the plane," in *Proc., IFIP Networking Conference*, Trondheim, Norway, Jun. 2014, pp. 1–9.

[24] W. Li, "Zipf's law everywhere." *Glottometrics*, vol. 5, pp. 14–21, 2002.

[25] F. Baccelli and B. Błaszczyszyn, *Stochastic Geometry and Wireless Networks*. Now Publishers Inc, 2009, vol. 1.

[26] S. Chen, D. Malak, and A. Abouzeid, "Spatially correlated placement policies for wireless content caching networks," 2022. [Online]. Available: bit.ly/3xjZkvH