IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 22 January 2024, accepted 13 February 2024, date of publication 19 February 2024, date of current version 27 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3367013

RESEARCH ARTICLE

Content Caching Strategies With On-Device Recommendation Systems in Wireless Caching Systems

MINJOONG RIM
Department of Information and Communication Engineering, Dongguk University, Seoul 04620, South Korea
e-mail: minjoong@dongguk.edu

This work was supported by the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (MSIT), Korean Government, under Grant NRF-2022R1F1A1062987.

ABSTRACT Wireless network traffic is exploding, mainly due to the growth of video streaming services. To cope with the increase of mobile traffic in wireless networks, techniques for installing content caches in base stations or devices are being investigated. Although the total amount of content is huge, the capacity of a cache installed in base stations or devices is limited, so efficient caching methods are needed to improve the hit ratio. To this end, we can consider content recommendation systems on devices. Since many users tend to select and watch videos from recommended content, a cache can improve its hit ratio by storing content that is more likely to be recommended on each device. This paper discusses how caching systems differ when devices recommend content versus when they do not. It also discusses how caching systems differ when recommendations are made based on cached content versus when they are not. Content with high average preferences should be cached without recommendations, while the caching system should take personal preferences into account when recommending content personally preferred by each device. This is especially true for cache-independent recommendation systems, since each device will recommend personal favorites regardless of cached content. If the cache size is very large compared to the number of recommended contents, the consideration of cached content in recommendation systems may become less important, since much personal content should be cached anyway. The simulation results show that caching schemes with recommendation systems can differ significantly from those without.

INDEX TERMS Wireless caching, mobile caching, D2D caching, recommendation, content preference, data offloading, hit ratio.

I. INTRODUCTION

Wireless network traffic is exploding, largely due to the growth of video streaming services [1], [2], [3]. In order to reduce the load on wired and wireless networks, techniques have been employed to distribute content caches across multiple geographic locations, and are evolving to install caches at the edge of the network, closer to the users [4], [5], [6]. More recently, techniques for placing content caches on base stations or devices have been explored to cope with the growth of mobile traffic in wireless networks [7], [8], [9], [10], [11]. Although the total number of contents is huge, the capacity of a cache installed on a base station or a device is limited. Therefore, it is crucial to improve the efficiency of caches [12], [13], [14], [15].

To increase cache efficiency, methods for linking caches with content recommendation systems have been studied [16], [17], [18], [19], [20]. Since many users tend to select and watch videos from recommended content, the probability of using content can be greatly increased if it is recommended [21], [22], [23], [24]. When a device enters an area covered by a cache, the hit ratio of the cache can be significantly increased by recommending content that the user may prefer from the cached content. A cache can also

The associate editor coordinating the review of this manuscript and approving it for publication was Francisco Rafael Marques Lima.

store content with a high probability of being recommended to each device. However, a cache in a base station or a device generally cannot know in advance which other devices will enter the coverage area during peak hours. This makes it difficult to account for content that is preferred by specific devices.

The size of a cache expands as storage becomes cheaper, but the total amount of content also expands rapidly, so a cache will continue to hold only a minimal fraction of the total content. Compared to increasing the total number of contents or the size of a cache, the number of recommended contents on a device is difficult to scale over time because it is related to the ability of humans to view or consider the list of recommendations at once. Therefore, the size of a cache is expected to gradually increase in the future compared to the number of recommended contents on a device. Assuming that the size of a cache is large compared to the number of recommended contents on a device, the caching system can store the contents that are likely to be recommended on devices to increase the hit ratio of the cache. In an environment where recommendations can significantly increase content preferences, traditional caching schemes that do not consider recommendations may not be appropriate for devices with recommendation systems, and it is necessary for a caching scheme to consider the characteristics of the recommendation systems.

Many literatures have considered and optimized caching and recommendation systems simultaneously, and have shown that the performance improvement is very large when caching and recommendations are combined or jointly optimized [16], [17], [18], [19], [20], [21], [22], [23], [24]. In [16], the concept of a soft hit was introduced, and [17] studied multiple Internet content providers. In [18], the dynamic behavior of users was considered, and [19], [20], [21] studied the social relationships among users. In [22], users' delay requirements, incentives, and protection mechanisms are considered. In [23], [24], and [25], deep reinforcement learning was used to simultaneously optimize caching and recommendation systems. In [26], [27], [28], and [29], the focus was not only on increasing the hit ratio, but also on improving the quality of experience to increase user satisfaction. In [30], instead of recommending a single item, a set of items is recommended. In [31], the freshness of information is considered, and in [32], the long- and short-term interests of users are considered. Reference [33] reduces the cost of the network in the long run while maintaining the quality of recommendations. Reference [34] makes recommendations based on resource availability. Reference [35] considers the diversification of recommendations to overcome the degradation of recommendation quality.

Most of these works focus on formulating and solving the joint optimization problems of caching and recommendation systems. However, by considering a minimal number of contents to solve the complex optimization problems, most of them fail to show the trends as the cache size increases. In real-world systems, the number of contents to be considered for caching or recommendation is huge, and the cache size is also increasing. Therefore, it is necessary to understand the characteristics of caching systems as the cache size increases.

In this paper, we consider a caching system and a recommendation system, where the caching system reduces the load on wireless networks, and the recommendation system increases user satisfaction and encourages video viewing by taking into account the user characteristics and viewing history. Since a caching system and a recommendation system are located in different places and serve different purposes, it is not easy to optimize them simultaneously, and it is necessary to consider them to operate independently with minimal information exchange. Most of the existing literature has not explained how caching methods differ when a device's recommendation system takes the caching system into account and when it does not.

The contributions of this paper are as follows.

(1) While much of the literature assumes that caching systems provide recommendations, or that caching and recommendation systems can be combined into one system, typical recommendation systems on devices are separate from caching systems. In this paper, we consider recommendation systems that are not part of caching systems and that serve for a different purpose.
(2) This paper shows that the caching scheme with recommendation systems on devices should be significantly different from the caching scheme without recommendation systems.
(3) While the total number of contents or the size of a cache increases continuously, the number of recommended contents is unlikely to increase indefinitely. This paper describes how caching methods change as the size of a cache increases relative to the number of recommended contents on a device.
(4) Some content may be preferred by only a small subset of users. This study examines how the degree of personalization affects caching methods with and without recommendations.
(5) Recommendation systems on devices are unlikely to be fully controlled by caching systems in base stations or other devices. This paper examines the differences between caching methods when recommendations are made based on cached content and when they are not.

This paper is organized as follows. Section II describes the system model used in this paper, including content preferences, caching systems, and recommendation systems. Section III discusses caching methods when no recommendations are made and when recommendations are made. When a recommendation system is present on a device, we also distinguish when recommendation systems take the caching system into account and when they do not. Section IV presents simulation results to show how caching methods differ when recommendations are made compared to when no recommendations are made. Finally, Section V draws conclusions.

II. SYSTEM MODEL

A. CONTENT PREFERENCE

For the sake of simplicity, let us assume that the contents are all the same size. The total number of contents, denoted by K_{total}, can be huge, but the number of contents considered for caching or recommendation will be limited. The content under consideration is divided into two types: *common content*, which is preferred by all users, and *personal content*, which is preferred by only some users. For simplicity, we assume that personal content is further divided into N_{group} groups, which can represent special genres or categories, and that each device can belong to only one of N_{group} groups, i.e., a device in group g prefers personal content in group g as well as common content.

The amount of common content is K_{common}, and the amount of personal content is $K_{personal}$ per group. The number of contents considered for caching or recommendation is less than or equal to the total number of contents, so $K_{common} + N_{group}K_{personal} \leq K_{total}$. Let $C_k^{common}(k = 1, \ldots, K_{common})$ denote the k-th piece of common content, and let $C_{g,k}^{personal}(g = 1, \ldots, N_{group}, k = 1, \ldots, K_{personal})$ denote the k-th piece of personal content in group g.

When a device in group g requests content, the *preference* of a piece of content is defined as the probability that the piece is the requested content, written as follows:

$$\mathcal{P}_g\left(C_k^{common}\right) = P_k^{common} \qquad (k = 1, \ldots, K_{common}), \tag{1}$$

$$\mathcal{P}_g\left(C_{f,k}^{personal}\right) = P_k^{personal} \qquad if\ f = g\ \left(k = 1, \ldots, K_{personal}\right). \tag{2}$$

For simplicity, assume that the probability of a device belonging to group g is $1/N_{group}$, and that the distribution of personal content preferences is the same regardless of group. If $f \neq g$ in Equation (2), the content preference is assumed to be small not enough to affect caching or recommendation. Assume that the content in each group is sorted in descending order of preference,

$$P_k^{common} \geq P_{k+1}^{common} \qquad (k = 1, \ldots, K_{common} - 1), \tag{3}$$

$$P_k^{personal} \geq P_{k+1}^{personal} \qquad \left(k = 1, \ldots, K_{personal} - 1\right). \tag{4}$$

Since not all content is considered for caching or recommendation, the sum of the preferences for the considered content is less than or equal to one, in other words, $P_{sum} \equiv P_{sum}^{common} + P_{sum}^{personal} \leq 1$, where $P_{sum}^{common} \equiv \sum_{k=1}^{K_{common}} P_k^{common}$ is the sum of common content preferences and $P_{sum}^{personal} \equiv \sum_{k=1}^{K_{personal}} P_k^{personal}$ is the sum of personal content preferences. Let the ratio of $P_{sum}^{personal}$ to P_{sum}^{common} be α, in other words, $\alpha \equiv P_{sum}^{personal}/P_{sum}^{common}$. α represents how much larger the sum of personal content preferences considered by each device is compared to the sum of common content preferences. A large α indicates a large difference in preferences between different groups of devices.

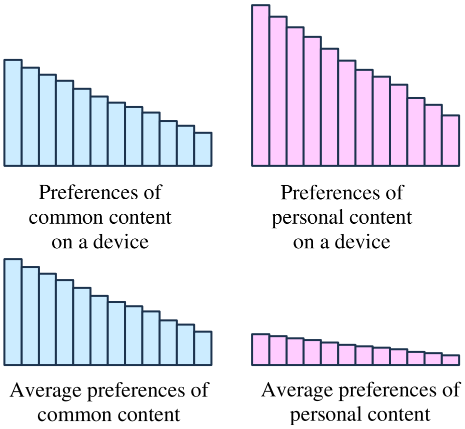

FIGURE 1. Content preference.

The average preferences across all groups are as follows:

$$\mathcal{P}_{average}\left(C_k^{common}\right) = P_k^{common} \qquad (k = 1, \ldots, K_{common}), \tag{5}$$

$$\mathcal{P}_{average}\left(C_{g,k}^{personal}\right) = \frac{1}{N_{group}} P_k^{personal} \qquad \left(k = 1, \ldots, K_{personal}\right). \tag{6}$$

The number of groups, N_{group}, represents the degree of personalization, which is the ratio of the individual preference to the average preference for personal content. Assuming $1 \leq \alpha \ll N_{group}$, each device has a high preference value for personal content, but the average preference value for personal content may be small compared to the common content. Let the k-th piece of personal content, sorted in descending order of average preference, be $C_{average,k}^{personal}$ and its average preference be $P_{average,k}^{personal}$, expressed as:

$$C_{average,k}^{personal} = C_{G(k),I(k)}^{personal} \qquad \left(k = 1, \ldots, N_{group}K_{personal}\right), \tag{7}$$

$$P_{average,k}^{personal} = \frac{1}{N_{group}} P_{I(k)}^{personal} \qquad \left(k = 1, \ldots, N_{group}K_{personal}\right), \tag{8}$$

where

$$G(k) \equiv k - N_{group}\left(\left\lceil \frac{k}{N_{group}} \right\rceil - 1\right) \qquad \left(k = 1, \ldots, N_{group}K_{personal}\right), \tag{9}$$

$$I(k) \equiv \left\lceil \frac{k}{N_{group}} \right\rceil \qquad \left(k = 1, \ldots, N_{group}K_{personal}\right), \tag{10}$$

and $\lceil\rceil$ is a ceiling (rounded up) operation. Figure 1 shows an example of preferences, where even if personal content preferences are large on each device, the average across all groups may be small.

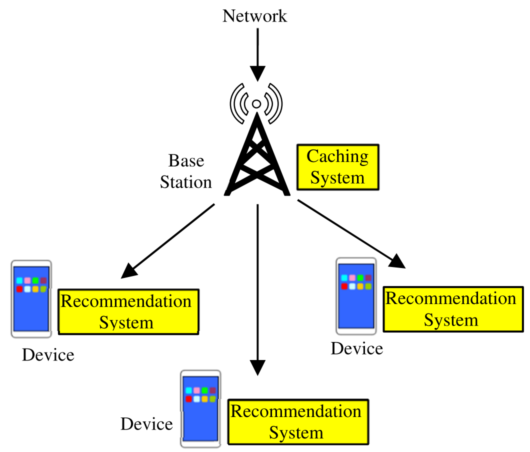

FIGURE 2. Caching and recommendation systems.

B. CACHING SYSTEM

A cache can be installed in a base station or in a dedicated device, as shown in Figure 2, and content can be stored before peak traffic congestion occurs, as shown in Figure 3 [36], [37], [38]. Assuming that it is not known in advance which devices will be within the coverage area of a cache during peak hours, it is not desirable to store content that is preferred by some specific devices. In this paper, we assume that caching is performed considering the average case of all devices. If a device within the coverage area of a cache requests content and the content is in the cache, it is delivered using cellular communication without going through the backhaul if the cache is installed at a base station, or using device-to-device (D2D) communication if the cache is installed in a device, resulting in data offloading, as shown in Figure 3 [36], [37], [38]. If the content is not in the cache, it is retrieved over the wireless network. To avoid unnecessary confusion, this paper does not consider self-offloading [15], but rather D2D offloading when a cache is installed in a device.

Assuming that a cache can store K_{cache} pieces of content, this paper investigates how many of the K_{cache} pieces stored in the cache are personal content and how many are common content. Let us express the caching scheme of storing $k_0 (0 \le k_0 \le K_{cache})$ pieces of personal content $C^{personal}_{average,k}$ and $K_{cache} - k_0$ pieces of common content C^{common}_{k} in the cache as follows:

$$S^{cache}(k_0) = \left\{ C^{common}_{k_c}, C^{personal}_{average,k_p} \right\} \\ \left(k_c = 1, \ldots, K_{cache} - k_0, k_p = 1, \ldots, k_0\right). \tag{11}$$

If a device in the cache's coverage area requests content and the content is stored in the cache, offloading can be achieved by receiving content from the cache without requesting the content over the network. Assuming that it is not possible to predict which devices will come into the cache's coverage area during peak hours, the cache has no choice

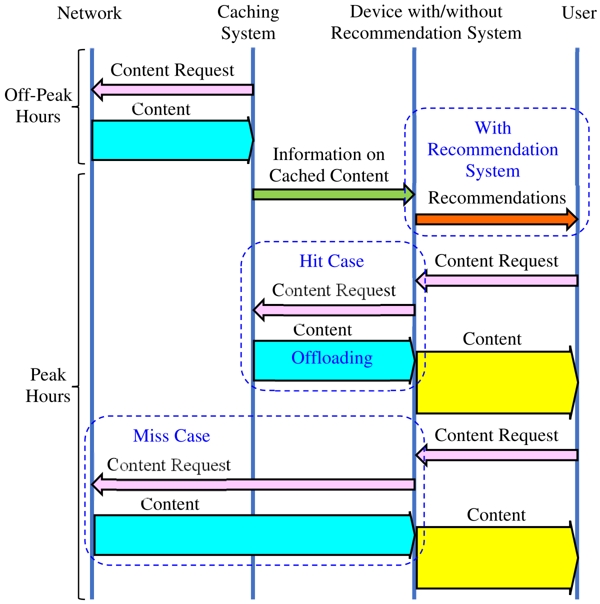

FIGURE 3. Caching and offloading.

but to consider average cases. Let the *hit ratio* $H(k_0)$ be the probability that, when a device requests content, the caching scheme $S^{cache}(k_0)$ includes the requested content in the cache and thus data offloading can be achieved. The optimal value of k_0 can be determined as the value that maximizes the hit ratio:

$$k_{optimal} \equiv argmax_{k_0} H(k_0), \tag{12}$$

$$H_{optimal} \equiv H\left(k_{optimal}\right). \tag{13}$$

C. RECOMMENDATION SYSTEM

If each device has its own recommendation system, as shown in Figure 2, it is necessary for a cache to store content taking into account the recommendation systems. When recommendations are made, users may tend to choose from the recommended content, so the preferences for the recommended content may increase compared to the case where no recommendations are made [16], [17], [18], [19], [20], [21], [22], [23], [24]. A recommendation system may consider the cached content if information about the cached content is passed from the caching system to the recommendation system. Alternatively, the cached content may not be considered for recommendations if the recommendation system operates independently of the caching system. In this paper, for a recommendation system that does not consider cached content, we assume that it recommends the $K_{recommend}$ most preferred content on each device, regardless of the cached content. For a recommendation system that considers cached content, we assume that it recommends the $K_{recommend}$ most preferred content on each device from the content stored in the cache.

If there are recommendation systems on devices, we assume that each device recommends $K_{recommend}$ pieces of content, and the preferences of the content increase by $\beta(\gg 1)$ times, and the preferences of the remaining content decrease accordingly. It is assumed that the number of recommended contents is very small compared to the total number of contents, so even if the preferences of the recommended contents increase by a factor of β, the sum of the preferences does not exceed one. The content preferences of a device in group g can be written as follows:

$$\mathcal{P}_g\left(C_k^{common}\right) = P_k^{common} \qquad (k = 1, \ldots, K_{common}), \tag{14}$$

$$\mathcal{P}_g\left(C_{average,k}^{personal}\right) = N_{group} P_{average,k}^{personal} \quad \text{if } G(k) = g \quad \left(k = 1, \ldots, K_{personal}\right). \tag{15}$$

If $G(k) \neq g$ in the content $C_{average,k}^{personal}$, it is assumed that the content does not have a meaningfully large preference value to be recommended by a device in group g. When the common content $C_k^{common}(k = 1, \ldots, K_{common})$ and the averaged personal content $C_{average,k}^{personal}(k = 1, \ldots, K_{personal})$ are combined into a single set on a device of group g, the content is called $C_{g,k}^{recommend}$ and its preference on the device is called $P_k^{recommend}$. Each device recommends $K_{recommend}$ pieces of content by some recommendation method, and the set of content recommended by a device in group g is called $S_g^{recommend}$. We assume that the recommendation changes the content preferences as:

$$\tilde{P}_{g,k}^{recommend} = \begin{cases} \beta P_k^{recommend} & \text{if } k \in S_g^{recommend} \\ \gamma_g P_k^{recommend} & \text{otherwise}, \end{cases} \tag{16}$$

where

$$\gamma_g = \frac{1 - \beta \sum_{k \in S_g^{recommend}} P_k^{recommend}}{1 - \sum_{k \in S_g^{recommend}} P_k^{recommend}}, \tag{17}$$

provided that

$$\sum_{k \in S_g^{recommend}} P_k^{recommend} \leq \frac{1}{\beta} \tag{18}$$

is satisfied.

III. CACHING SCHEMES

A. WHEN NO RECOMMENDATIONS ARE MADE

Consider caching methods in cases where devices do not recommend content. Using the caching method $S^{cache}(k_0)$, the hit ratio $H^{non-recommend}(k_0)$ is written as

$$H^{non-recommend}(k_0) = \sum_{k=1}^{K_{cache}-k_0} P_k^{common} + \sum_{k=1}^{k_0} P_{average,k}^{personal}, \tag{19}$$

and the optimal value of k_0 can be determined as the value that maximizes the hit ratio:

$$k_{optimal}^{non-recommend} \equiv argmax_{k_0} H^{non-recommend}(k_0), \tag{20}$$

$$H_{optimal}^{non-recommend} \equiv H^{non-recommend}\left(k_{optimal}^{non-recommend}\right). \tag{21}$$

First, consider the caching method $S^{cache}(0)$, which stores only common content. The hit ratio in this case is written as follows:

$$H^{non-recommend}(0) = \sum_{k=1}^{K_{cache}} P_k^{common}. \tag{22}$$

When the number of groups is large and the cache size is small, it may be sufficient to store only common content, but as the cache size grows, it may be necessary to store some personal content as well.

Consider the caching method $S^{cache}(K_{cache})$, which stores only personal content. The hit ratio in this case is:

$$H^{non-recommend}(K_{cache}) = \sum_{k=1}^{K_{cache}} P_{average,k}^{personal} = \frac{1}{N_{group}} \sum_{k=1}^{K_{cache}} P_{I(k)}^{personal}. \tag{23}$$

Because the preferences of personal content are averaged, the magnitudes of the preferences become smaller, especially when the number of groups is large, so storing only personal content cannot achieve excellent performance unless the preferences of personal content are very large. A cache may store content with high average preferences. Given the hypothetical probability $P_0^{common} = 1$ and $P_{average,0}^{personal} = 1$, let us consider an integer $K_1(0 \leq K_1 \leq K_{cache})$ that satisfies:

$$\begin{aligned} P_{K_{cache}-K_1+1}^{common} &\leq P_{average,K_1}^{personal}, \\ P_{average,K_1+1}^{personal} &\leq P_{K_{cache}-K_1}^{common}, \end{aligned} \tag{24}$$

or the rewritten equations:

$$\begin{aligned} P_{K_{cache}-K_1+1}^{common} &\leq \frac{1}{N_{group}} P_{I(K_1)}^{personal}, \\ \frac{1}{N_{group}} P_{I(K_1+1)}^{personal} &\leq P_{K_{cache}-K_1}^{common}. \end{aligned} \tag{25}$$

Consider the caching scheme $S^{cache}(K_1)$, which stores K_1 pieces of personal content and $K_{cache} - K_1$ pieces of common content. The hit ratio in this case is expressed as follows:

$$H^{non-recommend}(K_1) = \sum_{k=1}^{K_{cache}-K_1} P_k^{common} + \sum_{k=1}^{K_1} P_{average,k}^{personal}. \tag{26}$$

For a caching method $S^{cache}(k_0)$, where $k_0 < K_1$,

$$\begin{aligned} &H^{non-recommend}(K_1) - H^{non-recommend}(k_0) \\ &= \sum_{k=k_0+1}^{K_1} P_{average,k}^{personal} - \sum_{k=K_{cache}-K_1+1}^{K_{cache}-k_0} P_k^{common} \\ &\geq (K_1 - k_0)\left(P_{average,K_1}^{personal} - P_{K_{cache}-K_1+1}^{common}\right) \\ &\geq 0, \end{aligned} \tag{27}$$

and the hit ratio is less than or equal to that of $S^{cache}(K_1)$. For a caching method $S^{cache}(k_0)$, where $k_0 > K_1$,

$$\begin{aligned} &H^{non-recommend}(K_1) - H^{non-recommend}(k_0) \\ &= \sum_{k=K_{cache}-k_0+1}^{K_{cache}-K_1} P_k^{common} - \sum_{k=K_1+1}^{k_0} P_{average,k}^{personal} \\ &\geq (k_0 - K_1)\left(P_{K_{cache}-K_1}^{common} - P_{average,K_1+1}^{personal}\right) \end{aligned}$$

$$\geq 0, \tag{28}$$

and again, the hit ratio is less than or equal to that of $S^{cache}(K_1)$. Therefore, $S^{cache}(K_1)$ can maximize the hit ratio.

B. WHEN RECOMMENDATIONS ARE MADE WITHOUT CONSIDERING CACHED CONTENT

While the purpose of a caching system is to reduce the load on the wireless network, the purpose of a recommendation system on a device is to encourage video viewing and increase user satisfaction, so it can operate independently of the caching system. Consider a recommendation system that does not take the caching system into account when recommending content, assuming that each device recommends $K_{recommend}$ pieces of content from the total content with high individual preferences for each device, regardless of the content stored in the cache.

On a device in group g, common content $C_k^{common}(k = 1, \ldots, K_{common})$ and averaged personal content $C_{average,k}^{personal}(k = 1, \ldots, K_{personal})$ are combined into a single set for recommendations. The content sorted in descending order of the preference that is individually preferred by the device is called $C_{g,k}^{independent}$, and its corresponding preference is called $P_k^{independent}$. We assume that each device recommends $K_{recommend}$ pieces of content with high preferences from the set of $C_{g,k}^{independent}$, which is written as follows:

$$S_g^{independent} = \left\{ C_{g,k}^{independent} \right\} \ (k = 1, \ldots, K_{recommend}). \tag{29}$$

When recommendations are made, the preferences change as:

$$\tilde{P}_{g,k}^{independent} = \begin{cases} \beta P_k^{independent} & \textit{if } k \leq K_{recommend} \\ \gamma P_k^{independent} & \textit{otherwise}, \end{cases} \tag{30}$$

where

$$\gamma = \frac{1 - \beta \sum_{k=1}^{K_{recommend}} P_k^{independent}}{1 - \sum_{k=1}^{K_{recommend}} P_k^{independent}}, \tag{31}$$

provided that

$$\sum_{k=1}^{K_{recommend}} P_k^{independent} \leq \frac{1}{\beta} \tag{32}$$

is satisfied.

Given hypothetical probabilities $P_0^{common} = 1$ and $P_0^{personal} = 1$, let us consider an integer $K_2 (0 \leq K_2 \leq K_{recommend})$ that satisfies the following equations:

$$\begin{aligned} P_{K_{recommend}-K_2+1}^{common} &\leq P_{K_2}^{personal}, \\ P_{K_2+1}^{personal} &\leq P_{K_{recommend}-K_2}^{common}. \end{aligned} \tag{33}$$

While Equation (24) compares common content to averaged personal content, Equation (33) compares common content to personal content for each device. Consider the case where each device recommends K_2 pieces of personal content and $K_{recommend} - K_2$ pieces of common content. With recommendations, the preference changes as:

$$\tilde{P}_k^{common} = \begin{cases} \beta P_k^{common} & \textit{if } k \leq K_{recommend} - K_2 \\ \gamma P_k^{common} & \textit{otherwise}, \end{cases} \tag{34}$$

$$\tilde{P}_k^{personal} = \begin{cases} \beta P_k^{personal} & \textit{if } k \leq K_2 \\ \gamma P_k^{personal} & \textit{otherwise}, \end{cases} \tag{35}$$

$$\tilde{P}_{average,k}^{personal} = \begin{cases} \beta P_{average,k}^{personal} & \textit{if } k \leq K_2 N_{group} \\ \gamma P_{average,k}^{personal} & \textit{otherwise}, \end{cases} \tag{36}$$

where

$$\gamma = \frac{1 - \beta \sum_{k=1}^{K_{recommend}-K_2} P_k^{common} - \beta \sum_{k=1}^{K_2} P_k^{personal}}{1 - \sum_{k=1}^{K_{recommend}-K_2} P_k^{common} - \sum_{k=1}^{K_2} P_k^{personal}}, \tag{37}$$

provided that

$$\sum_{k=1}^{K_{recommend}-K_2} P_k^{common} + \sum_{k=1}^{K_2} P_k^{perosnal} \leq \frac{1}{\beta} \tag{38}$$

is satisfied.

Consider the caching scheme $S^{cache}(k_0)$, which stores k_0 pieces of personal content and $K_{cache} - k_0$ pieces of common content. The hit ratio can be written as:

$$\begin{aligned} &H^{independent}(k_0) \\ &\quad = \beta \sum_{k=1}^{\min(K_{recommend}-K_2, K_{cache}-k_0)} P_k^{common} \\ &\quad + \gamma \sum_{k=\min(K_{recommend}-K_2, K_{cache}-k_0)+1}^{K_{cache}-k_0} P_k^{common} \\ &\quad + \beta \sum_{k=1}^{\min(N_{group}K_2, k_0)} P_{average,k}^{personal} \\ &\quad + \gamma \sum_{k=\min(N_{group}K_2, k_0)+1}^{k_0} P_{average,k}^{personal}, \end{aligned} \tag{39}$$

and the optimal value of k_0 can be determined as the value that maximizes the hit ratio:

$$k_{optimal}^{independent} \equiv argmax_{k_0} H^{indepedent}(k_0), \tag{40}$$

$$H_{optimal}^{independent} \equiv H^{independent}\left(k_{optimal}^{independent}\right). \tag{41}$$

If the cache size is very small compared to the total number of contents, β is significant, and thus γ is small, then the content stored in the cache but not recommended may not contribute much to the hit ratio. In this case, the hit ratio can be approximated by considering only recommended content as follows:

$$\begin{aligned} H^{independent}(k_0) \approx \beta &\sum_{k=1}^{\min(K_{recommend}-K_2, K_{cache}-k_0)} P_k^{common} \\ &+ \beta \sum_{k=1}^{min(N_{group}K_2, k_0)} P_{average,k}^{personal}. \end{aligned} \tag{42}$$

Considering the caching method $S^{cache}(0)$, which stores only common content, the hit ratio can be approximated as follows:

$$H^{independent}(0) \approx \beta \sum_{k=1}^{K_{recommend}-K_2} P_k^{common}. \tag{43}$$

Even if more than $K_{recommend} - K_2$ pieces of common content are stored, the excess amount of common content is not recommended, and the performance improvement from storing a large amount of common content may not be significant, assuming that the contribution of non-recommended content to the hit ratio is negligible. Since each device makes recommendations based on each device's individual content preferences rather than the averaged content preferences, it may be necessary to store more personal content when considering recommendations.

Let us consider the caching method $S^{cache}(K_{cache})$, which stores only personal content in the cache. The hit ratio can be written as follows:

$$H^{independent}\left(K_{cache}\right) \approx \beta \sum\nolimits_{k=1}^{min(N_{group}K_2, K_{cache})} P_{average,k}^{personal}. \tag{44}$$

If only common content is stored in a cache, the same $K_{recommend} - K_2$ pieces of common content are recommended for each device, and the performance may not be significantly improved by storing more than $K_{recommend} - K_2$ pieces of common content. On the other hand, if only personal content is stored, each device recommends different personal content according to the device's group, and the hit ratio increases until the cache size reaches $N_{group}K_2$. Therefore, if the cache size is very large compared to the number of recommended contents, it may be necessary to store a larger amount of personal content.

Consider $S^{cache}(k_0)$, which stores k_0 pieces of personal content and $K_{cache} - k_0$ pieces of common content. The optimal value of k_0 can be obtained by computing Equation (40), but let's take a very rough look to see what the trend is. In order to store K_2 personal content and $K_{recommend} - K_2$ common content recommended by devices in all N_{group} groups, the equation

$$K_{cache} \geq N_{group}K_2 + K_{recommend} - K_2 \tag{45}$$

or the rewritten equation

$$K_2 \leq \left(K_{cache} - K_{recommend}\right) / \left(N_{group} - 1\right) \tag{46}$$

must be satisfied.

If the cache size is large enough and Equation (45) is satisfied, the caching scheme $S^{cache}(N_{group}K_2)$, which stores $N_{group}K_2$ pieces of personal content and $K_{cache} - N_{group}K_2$ pieces of common content, can be considered. Caching personal content can help improve the hit ratio by a factor of β if it is recommended on the device, but storing non-recommended personal content may not help much due to averaging over all groups, so we consider storing only personal content that can be recommended. The hit ratio in this case is approximated as follows:

$$H^{independent}\left(N_{group}K_2\right) \approx \beta \sum\nolimits_{k=1}^{K_{recommend}-K_2} P_k^{common} + \beta \sum\nolimits_{k=1}^{K_2} P_k^{personal}. \tag{47}$$

Suppose Equation (45) is not satisfied. In this case, it is necessary to reduce the amount of personal content stored in the cache to match the cache size. Storing more than $K_{recommend} - K_2$ pieces of common content does not contribute much to improving the hit ratio if γ is small, so k_0 can be taken as $K_{cache} - K_{recommend} + K_2$. Consider $S^{cache}(K_3)$, where

$$K_3 \equiv K_{cache} - K_{recommend} + K_2, \tag{48}$$

and the hit ratio in this case is approximated as follows:

$$H^{independent}\left(K_3\right) \approx \beta \sum\nolimits_{k=1}^{K_{recommend}-K_2} P_k^{common} + \beta \sum\nolimits_{k=1}^{K_{cache}-K_{recommend}+K_2} P_{avearge,k}^{personal}. \tag{49}$$

When the cache size does not satisfy Equation (45), as the cache size increases, a larger amount of personal content must be stored in the cache to increase the hit ratio. Combining the cases when Equation (45) is satisfied and when it is not, K_3 can be rewritten as follows:

$$K_3 \equiv \begin{cases} N_{group}K_2 & if\ K_2 \leq \dfrac{K_{cache} - K_{recommend}}{N_{group} - 1} \\ K_{cache} - K_{recommend} + K_2 & otherwise. \end{cases} \tag{50}$$

A cache can contain a small amount of personal content if no recommendations are made. However, especially for a large cache size, the amount of personal content in the cache must increase when recommendations are made.

In particular, if the cache size K_{cache} is equal to the number of recommended contents $K_{recommend}$, then $K_3 = K_2$ and the hit ratio in this case is as follows:

$$H_{smallcache}^{independent}\left(K_2\right) \approx \beta \sum\nolimits_{k=1}^{K_{cache}-K_2} P_k^{common} + \beta \sum\nolimits_{k=1}^{K_2} P_{avearge,k}^{personal}. \tag{51}$$

C. WHEN RECOMMENDATIONS ARE MADE WITH CONSIDERING CACHED CONTENT

Since the purpose of a recommendation system on a device is to increase user satisfaction and encourage video viewing, it can operate independently of the corresponding caching system. However, in cases where the wireless traffic is congested and the communication quality is very poor, the user experience may be degraded when the content is delivered over the congested wireless network, and recommendations based on cached content could improve the user experience.

Consider a recommendation system that can improve the hit ratio of the corresponding cache by taking into account the content stored in the cache. Each device can receive information about the content stored in the cache, or it can infer what is cached based on the information about the cache size and caching method. Suppose each device recommends $K_{recommend}$ pieces of cached content with high preferences. If K_{cache} is less than $K_{recommend}$, then only K_{cache} pieces of content are assumed to be recommended, and $K_{recommend}$ becomes K_{cache}.

Consider $S^{cache}(k_0)$, which stores k_0 pieces of personal content and $K_{cache} - k_0$ pieces of common content. When a device in group g combines the common content $C_k^{common}(k = 1, \ldots, K_{cache} - k_0)$ and the averaged personal content $C_{average,k}^{personal}(k = 1, \ldots, k_0)$ stored in the cache into a single set and sorts them in descending order of the device's individual preference, the content is called $C_{g,k}^{cache-aware}(k_0)$ and its corresponding preference is called $P_{g,k}^{cache-aware}(k_0)$. The device selects and recommends $K_{recommend}$ pieces with high preferences from the content stored in the cache:

$$S_g^{cache-aware}(k_0) = \left\{C_{g,k}^{cache-aware}(k_0)\right\} \quad (k = 1, \ldots, K_{recommend}). \tag{52}$$

When recommendations are made, the preferences change as

$$\tilde{P}_{g,k}^{cache-aware}(k_0) = \begin{cases} \beta P_{g,k}^{cache-aware}(k_0) & if\ k \leq K_{recommend} \\ \gamma_g(k_0) P_{g,k}^{cache-aware}(k_0) & otherwise, \end{cases} \tag{53}$$

where

$$\gamma_g(k_0) = \frac{1 - \beta \sum_{k=1}^{K_{recommend}} P_{g,k}^{cache-aware}(k_0)}{1 - \sum_{k=1}^{K_{recommend}} P_{g,k}^{cache-aware}(k_0)}, \tag{54}$$

provided that

$$\sum_{k=1}^{K_{recommend}} P_{g,k}^{cache-aware}(k_0) \leq \frac{1}{\beta} \tag{55}$$

is satisfied.

The hit ratio on a device in group g is written as:

$$\begin{aligned} H_g^{cache-aware}(k_0) &= \beta \sum_{k=1}^{K_{recommend}} P_{g,k}^{cache-aware}(k_0) \\ &+ \gamma_g(k_0) \sum_{k=1}^{K_{cache}-K_{recommend}} P_{g,k}^{cache-aware}(k_0). \end{aligned} \tag{56}$$

and the hit ratio is obtained by averaging over all groups, written as follows:

$$H^{cache-aware}(k_0) = \frac{1}{N_{group}} \sum_{g=1}^{N_{group}} H_g^{cache-aware}(k_0). \tag{57}$$

The optimal value of k_0 can be determined as the value that maximizes the hit ratio:

$$k_{optimal}^{cache-aware} \equiv argmax_{k_0} H^{cache-aware}(k_0), \tag{58}$$

$$H_{optimal}^{cache-aware} \equiv H^{cache-aware}\left(k_{optimal}^{cache-aware}\right). \tag{59}$$

If the cache is small and $K_{recommend} = K_{cache}$, i.e. all content stored in the cache can be recommended, the hit ratio is expressed as follows:

$$\begin{aligned} H_{smallcache}^{cache-aware}(k_0) &= \frac{\beta}{N_{group}} \sum_{g=1}^{N_{group}} \sum_{k=1}^{K_{recommend}} P_{g,k}^{cache-aware}(k_0) \\ &= \beta \sum_{k=1}^{K_{cache}-k_0} P_k^{common} + \beta \sum_{k=1}^{k_0} P_{average,k}^{personal} \\ &= \beta H^{non-recommend}(k_0). \end{aligned} \tag{60}$$

If all stored content can be recommended, the performance can be greatly improved by recommendations. In this case, as in the case where no recommendations are made, the caching scheme $S^{cache}(K_1)$ can be used, and the hit ratio can be written as follows:

$$\begin{aligned} H_{smallcache}^{cache-aware}(K_1) &= \beta \sum_{k=1}^{K_{cache}-K_1} P_k^{common} + \beta \sum_{k=1}^{K_1} P_{average,k}^{personal}. \end{aligned} \tag{61}$$

Compared to the hit ratio equation in (51), where recommendations are made independent of cached content, better performance can be achieved by recommending content with higher average preferences. Especially when the cache size is small, cache-aware recommendations can significantly improve performance.

The total amount of content increases, and cache sizes grow as storage prices decrease. However, the number of recommended contents is related to a person's ability to view or consider the list of recommendations at one time and cannot grow indefinitely. Therefore, we need to consider the case where the cache size is larger than the number of recommended contents. The optimal value of k_0 can be found using Equation (58) and may vary depending on the distribution of content preferences. Nevertheless, let us take a very rough look at how the value of k_0 affects the hit ratio differently when no recommendations are made, or when recommendations are made with or without considering cached content.

Suppose the cache size is minimal compared to the total number of contents, β is significant, and thus $\gamma_g(k_0)$ is small, then the contents stored in the cache but not recommended may not contribute much to the hit ratio. In this case, the hit ratio can be approximated by considering only recommended content as follows:

$$H^{cache-aware}(k_0) \approx \frac{1}{N_{group}} \sum_{g=1}^{N_{group}} \sum_{k=1}^{K_{recommend}} \beta P_{g,k}^{cache-aware}(k_0). \tag{62}$$

First, consider the caching method $S_{cache}(0)$, which stores only common content. Since only common content is stored in the cache, the recommended content is also common content. The hit ratio is approximated as follows:

$$H^{cache-aware}(0) \approx \beta \sum_{k=1}^{K_{recommend}} P_k^{common}. \tag{63}$$

Even if more than $K_{recommend}$ pieces of common content are stored, the excess amount of common content is not recommended, and the performance improvement from storing more than $K_{recommend}$ pieces of common content may not be significant, assuming that the contribution of non-recommended content to the hit ratio is negligible. Compared to Equation (43), where recommendations are made independently of cached content, performance can be further improved by not recommending unstored personal content.

Consider $S^{cache}(K_{cache})$, which stores only personal content. The hit ratio can be approximated as follows:

$$H^{cache-aware}(K_{cache})$$

$$\approx \beta \sum\nolimits_{k=1}^{min(N_{group}K_{recommend},K_{cache})} P^{personal}_{avearge,k}. \tag{64}$$

If only common content is cached, all devices will recommend the same $K_{recommend}$ pieces of common content. However, if only personal content is stored, each device will recommend different pieces of personal content, and the hit ratio will increase until the cache size reaches $N_{group}K_{recommend}$. As the cache size increases compared to the number of recommended contents, more pieces of personal content must be stored to improve the hit ratio. Consider $S^{cache}(k_0)$, which stores k_0 pieces of personal content and $K_{cache} - k_0$ pieces of common content. If the equations

$$\begin{aligned} K_{cache} - k_0 &\geq K_{recommend} - \lfloor k_0/N_{group} \rfloor, \\ P^{common}_{K_{recommend}-I(k_0)+1} &\leq P^{personal}_{I(k_0)} \end{aligned} \tag{65}$$

are satisfied, approximately $[k_0/N_{group}]$ pieces can be recommended from personal content and $K_{recommend} - [k_0/N_{group}]$ pieces can be recommended from common content, where $\lfloor\rfloor$ is a floor (rounding down) and [] is a rounding operation. The hit ratio in this case can be approximated as follows:

$$\begin{aligned} &H^{cache-aware}(k_0) \\ &\approx \beta \sum\nolimits_{k=1}^{K_{recommend}-[k_0/N_{group}]} P^{common}_k \\ &\quad + \beta \sum\nolimits_{k=1}^{[k_0/N_{group}]} P^{personal}_k. \end{aligned} \tag{66}$$

The caching scheme when no recommendations are made, shown in Equation (19), selects K_{cache} pieces commonly preferred by all devices. On the other hand, the caching scheme when cache-aware recommendations are made, shown in Equation (66), supports each device to select $K_{recommend}$ pieces personally preferred by the device from the content stored in the cache. When no recommendations are made, the average preferences of personal content have small values, especially when the number of groups is large, and personal content is given little consideration. On the contrary, when making recommendations, the personally preferred content of each device is selected and recommended without averaging, so it is necessary to store more pieces of personal content, especially if the cache size is huge compared to the number of recommended contents.

The optimal value of k_0 when recommendations are made can be determined by calculating Equation (58), but we can deduce a very rough value to compare with the scheme without recommendations. If Equation (45) is satisfied, the caching scheme $S^{cache}(N_{group}K_2)$ can be used. When personal content stored in the cache is recommended to a device, its preference increases by a factor of β, which can significantly improve the hit ratio. However, storing non-recommended personal content in the cache may not help much, so we do not consider storing non-recommended personal content. The approximate hit ratio of the caching scheme $S^{cache}(N_{group}K_2)$ is written as follows:

$$\begin{aligned} &H^{cache-aware}(N_{group}K_2) \\ &\approx \beta \sum\nolimits_{k=1}^{K_{recommend}-K_2} P^{common}_k + \beta \sum\nolimits_{k=1}^{K_2} P^{personal}_k. \end{aligned} \tag{67}$$

The above equation is equivalent to Equation (47). When the cache size is small compared to the number of recommended contents, it is beneficial to consider cached content in recommendation systems in terms of the hit ratio, but when the cache size is huge compared to the number of recommended contents, the performance improvement using cache-aware recommendations may not be significant. Suppose Equation (45) is not satisfied. In this case, it is necessary to reduce the amount of personal content stored in the cache to match the cache size. If k_0 pieces of personal content are stored in the cache, approximately $\lfloor k_0/N_{group} \rfloor$ pieces of personal content can be recommended on each device, so in order to recommend $K_{recommend}$ pieces of content on each device, it is necessary to store at least $K_{recommend} - \lfloor k_0/N_{group} \rfloor$ pieces of common content. In order to recommend $K_{recommend}$ pieces of content with meaningfully large preferences, the number of personal contents stored in the cache is as follows:

$$0 \leq k_0 \leq \frac{N_{group}(K_{cache} - K_{recommend})}{N_{group} - 1}. \tag{68}$$

If Equation (45) is not satisfied, consider the caching method $S^{cache}(K_4)$, where

$$\begin{aligned} K_4 \equiv &\left\lfloor \frac{N_{group}(K_{cache} - K_{recommend})}{N_{group} - 1} \right\rfloor \\ &\quad if\ K_2 > (K_{cahe} - K_{recommend}) / \left(N_{group} - 1\right). \end{aligned} \tag{69}$$

Let us define the integer K_5 as follows:

$$K_5 \equiv \left[(K_{cache} - K_{recommend}) / \left(N_{group} - 1\right)\right]. \tag{70}$$

Using the caching method $S^{cache}(K_4)$, approximately K_5 pieces of personal content and $K_{recommend} - K_5$ pieces of common content can be selected on each device. In this case, the hit ratio is approximated as follows:

$$\begin{aligned} &H^{cache-aware}(K_4) \\ &\approx \beta \sum\nolimits_{k=1}^{K_{recommend}-K_5} P^{common}_k + \beta \sum\nolimits_{k=1}^{K_5} P^{personal}_k. \end{aligned} \tag{71}$$

Since $K_5 \leq K_2$, from Equation (33), the following equation can hold:

$$P^{common}_{K_{recommend}-K_5+1} \leq P^{personal}_{K_5}. \tag{72}$$

If more than K_4 pieces of personal content are stored in the cache, a device may not be able to recommend $K_{recommend}$ pieces of content with meaningful preference values. Consider the caching scheme $S^{cache}(k_0)$, which stores $k_0(< K_4)$ pieces of personal content in the cache. On average, $[k_0/N_{group}] (\leq K_4)$ pieces of personal content and $K_{recommend} - [k_0/N_{group}]$ pieces of common content can be recommended. Since

$$\begin{aligned} &H^{cache-aware}(K_4) - H^{cache-aware}(k_0) \\ &\approx \beta \sum\nolimits_{k=\left[\frac{k_0}{N_{group}}\right]+1}^{K_5} P^{personal}_k \\ &\quad - \beta \sum\nolimits_{k=K_{recommend}-K_5+1}^{K_{recommend}-\left[\frac{k_0}{N_{group}}\right]} P^{common}_k \end{aligned}$$

$$\geq \beta\left(K_5 - \left[\frac{k_0}{N_{group}}\right]\right)\left(P^{personal}_{K_5} - P^{common}_{K_{recommend}-K_5+1}\right)$$
$$\geq 0, \tag{73}$$

it may not be advantageous to store less than K_4 pieces of personal content in the cache if the contribution of non-recommended content to the hit ratio is negligible.

In practice, when considering non-recommended content that is omitted from the approximated equations, it may be desirable to store less than K_4 pieces of personal content, because the common content may have much larger average preference values than the personal content. Combining the cases when Equation (45) is satisfied and when it is not, K_4 can be rewritten as follows:

$$K_4 \equiv \begin{cases} N_{group}K_2 & \text{if } K_2 \leq \dfrac{K_{cache}-K_{recommend}}{N_{group}-1} \\ \left\lfloor \dfrac{N_{group}(K_{cache} - K_{recommend})}{N_{group} - 1} \right\rfloor & \text{otherwise.} \end{cases} \tag{74}$$

Comparing the above equation with Equation (50) for the case of recommendations independent of cached content, the values grow similarly with increasing cache size. In particular, when recommendations are made independent of cached content, it is necessary to store more personal content in the cache because each device will recommend personally preferred content.

The caching methods are very different when recommendations are provided than when they are not. Without recommendations, average preferences are taken into account, making personal content less important for caching. With recommendations, however, individual preferences on each device are taken into account, making personal content more important for caching, especially if the cache size is large enough to store personal content from multiple groups.

IV. SIMULATION RESULTS

In the simulation, the hit ratio was calculated considering the average case, assuming that devices from each group enter the coverage area of the cache with equal probability. The amount of personal content at a given cache size was increased from zero to the cache size, and an exhaustive search was performed to calculate the value for all cases and find the best amount of personal content.

Suppose the content in each group follows a Zipf distribution with Zipf coefficient $\lambda_{common} = 0.8$ for common content and Zipf coefficient $\lambda_{personal} = 0.8$ for personal content. The amount of common content $K_{common} = 10000$, the amount of personal content considered in each group $K_{personal} = 500$, the number of groups $N_{group} = 50$, and the number of recommended contents per device $K_{recommend} = 25$. The cache size K_{cache} varies from 0 to 500 to measure the hit ratio of the cache as a function of the cache size. The sum of the considered content preferences $P_{sum} = 0.6$, the ratio of the sum of personal content preferences to the sum of common content preferences $\alpha = 1$, and the increase in preferences due to recommendations $\beta = 5$. The simulation parameters are summarized in Table 1.

TABLE 1. Simulation parameters.

Notation	Parameter	Value
λ_{common}	Zipf coefficient for common content	0.8
$\lambda_{personal}$	Zipf coefficient for personal content	0.8
K_{common}	The amount of common content	10000
$K_{personal}$	The amount of personal content in each group	500
N_{group}	The number of groups	50
$K_{recommend}$	The number of recommended contents per device	25
K_{cache}	The cache size	0~500
P_{sum}	The sum of the considered content preferences	0.6
α	The ratio of the sum of personal content preferences to the sum of common content preferences	1
β	The increase in preferences due to recommendations	5

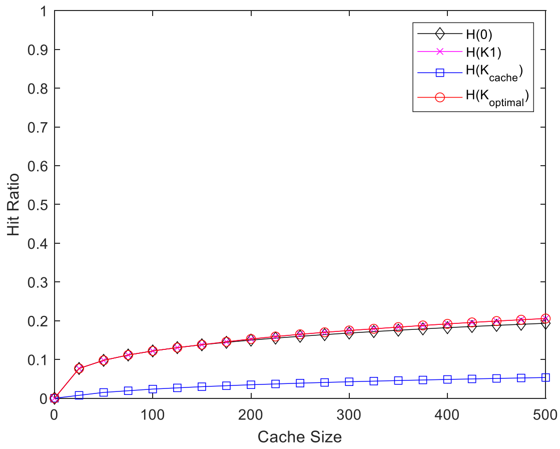

FIGURE 4. Hit ratio when not making recommendations.

Figure 4 shows the hit ratio as a function of the cache size when recommendations are not made. Storing a large amount of personal content can lead to unsatisfactory results, while storing only common content does not show much difference from the optimal results.

Figure 5 shows the hit ratio as a function of the cache size when using recommendation systems independent of cached content. Even when cached content is not considered, there is a significant performance improvement over the case without recommendations. Especially when the cache size is large, storing only common content or storing only personal content will result in a poor hit ratio, and it is necessary to store a reasonable amount of personal content. If only common content is cached, the hit ratio does not improve significantly as the case size increases, whereas if a moderate amount of personal content is stored, the performance improves as the cache size increases.

Figure 6 shows the hit ratio as a function of cache size when using cache-aware recommendations. The performance is better than when using independent recommendations, but

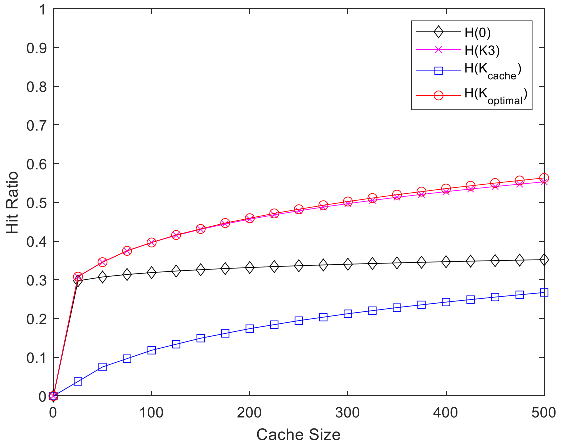

FIGURE 5. Hit ratio when making recommendations independent of caching.

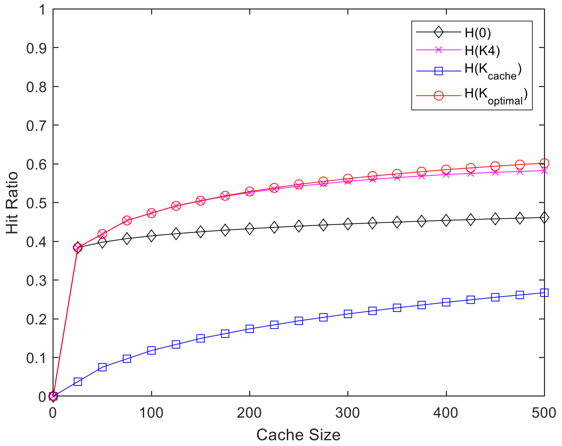

FIGURE 6. Hit ratio when making cache-aware recommendations.

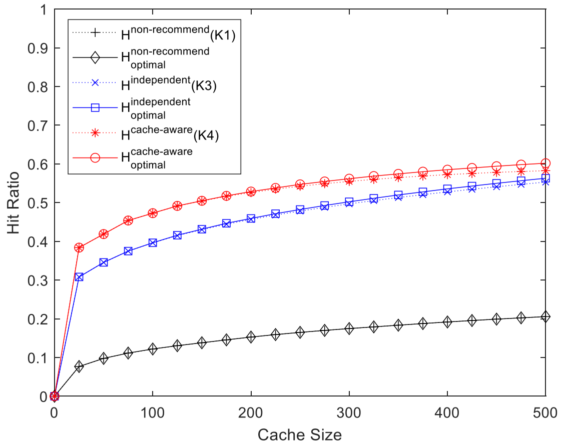

FIGURE 7. Hit ratio according to cache size.

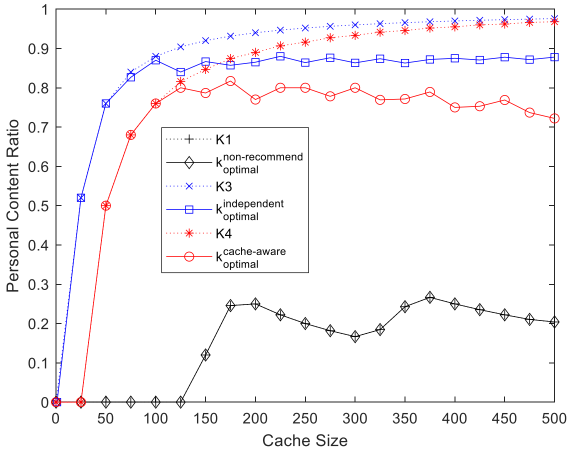

FIGURE 8. Ratio of personal content according to cache size.

the improvement is not very significant. Again, storing only common content or only personal content does not yield satisfactory results.

Figure 7 combines some parts of Figures 4, 5, and 6 into a single figure to compare the cases with and without recommendations. Content recommendations significantly improve performance, and cache-aware recommendations can improve performance even further. The additional performance improvement of cache-aware recommendations over independent recommendations diminishes as the cache size increases. For a small cache size, cache awareness may be essential for recommendation systems, but it becomes less important as the cache size increases.

Figure 8 shows the proportion of personal content stored in the cache. Without recommendations, the cache stores mostly common content, while with recommendations, it is necessary to store a more significant amount of personal content, especially when the cache size is large. The proportion of personal content among the stored content may increase as the cache size increases. When the recommendation systems do not consider cached content, it is necessary to store a more significant amount of personal content in the cache compared to the cache-aware recommendation case because each device makes recommendations based on its personalized preferences regardless of cached content. While the value of K_1 is the same as the optimal amount of personal content when no recommendations are made, the value of K_3 for independent recommendations or the value of K_4 for cache-aware recommendations is somewhat different from the corresponding optimal value. However, as shown in Figure 7, the hit ratio of K_3 or K_4 is not significantly different from the corresponding optimal hit ratio.

In Figures 9 and 10, the other simulation parameters are the same as in Table 1, except that $\alpha = 3$ to increase the favorability of personal content on each device. Figure 9 shows the hit ratio as a function of cache size, and Figure 10 shows the proportion of personal content stored in the cache. As the favorability of personal content on each device increases, more personal content must be cached, not only when recommendations are made, but also when no recommendations are made.

In Figures 11 and 12, the other simulation parameters are the same as in Table 1, except that $\alpha = 0.3$ to reduce

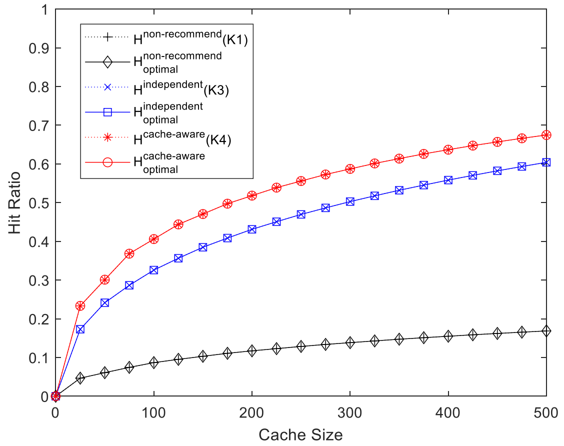

FIGURE 9. Hit ratio when personal content has large preferences.

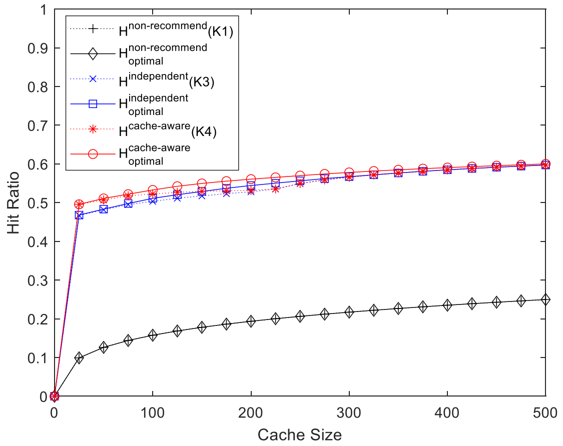

FIGURE 11. Hit ratio when personal content has small preferences.

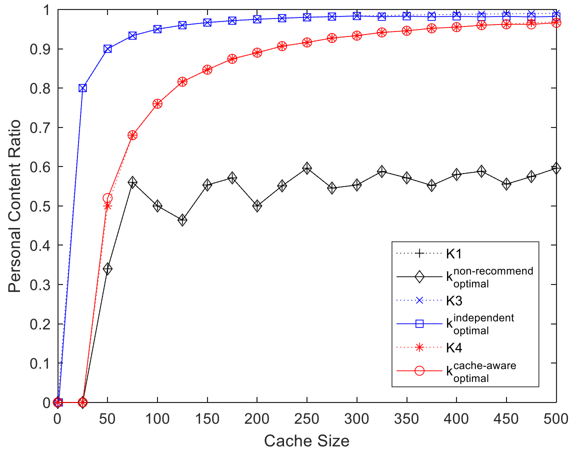

FIGURE 10. Ratio of personal content when personal content has large preferences.

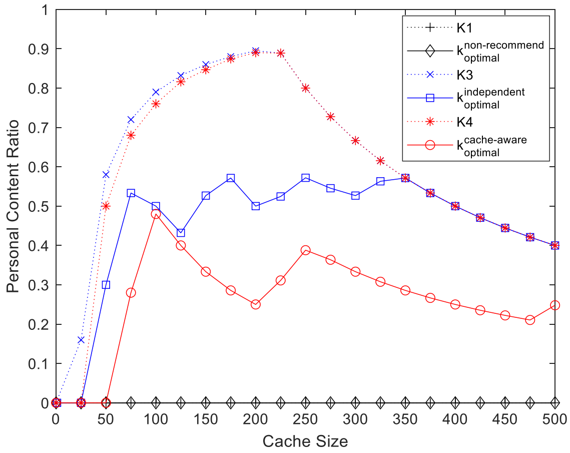

FIGURE 12. Hit ratio when preference increase due to recommendations is large.

the favorability of personal content on a device. Figure 11 shows the hit ratio as a function of cache size, and Figure 12 shows the proportion of personal content stored in the cache. As the favorability of personal content on a device decreases, the proportion of personal content stored in the cache also decreases. In Figure 12, when no recommendations are made, no personal content is cached at all. However, when recommendations are made, it is necessary to store a large amount of personal content, especially when the cache size is large. In Figure 12, in contrast to Figure 10, the value of K_3 for independent recommendations or the value of K_4 for cache-aware recommendations has a large difference from the corresponding optimal value. However, Figure 11 shows that the hit ratio is not significantly different from the optimal one.

This time, the other simulation parameters are the same as in Table 1, except that $\beta = 8$ in order to get a more significant increase in preferences with recommendations. Figure 13 shows the hit ratio as a function of cache size, and Figure 14 shows the proportion of personal content stored in the cache. When β is large, the hit ratio improves significantly when recommendations are made. As the effectiveness of recommended content increases, the content personally preferred by each device needs to be cached, and it may be necessary to store a larger amount of personal content with recommendations.

In Figures 15 and 16, the other simulation parameters are the same as in Table 1, except that $N_{group} = 200$ to increase the degree of personalization. The more granular the genre or category, the larger the number of groups can be. Figure 15 shows the hit ratio as a function of cache size, and Figure 16 shows the proportion of personal content stored in the cache. The number of groups indicates the degree of personalization of personal content, and when this value is significant, the average preference for personal content decreases. It may be necessary to cache less personal content, especially if no recommendations are made. Figure 16 shows that without recommendations, since personal content has minimal average preferences, maximizing the hit ratio can be achieved by storing only common content. On the other hand, with recommendations, it is still necessary to store a large amount

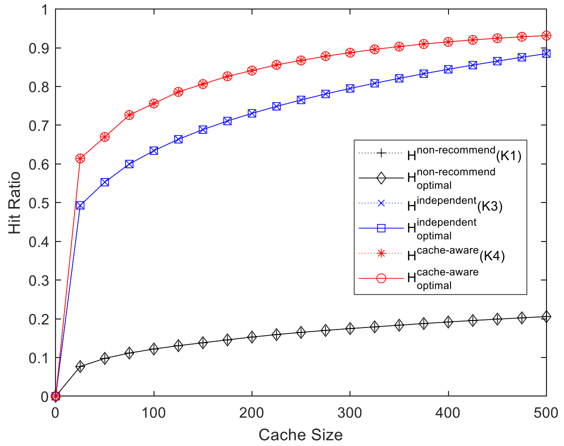

FIGURE 13. Ratio of personal content when personal content has small preferences.

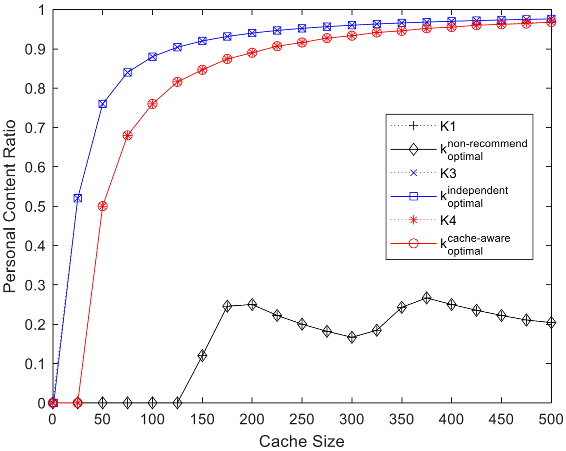

FIGURE 14. Ratio of personal content when preference increase due to recommendations is large.

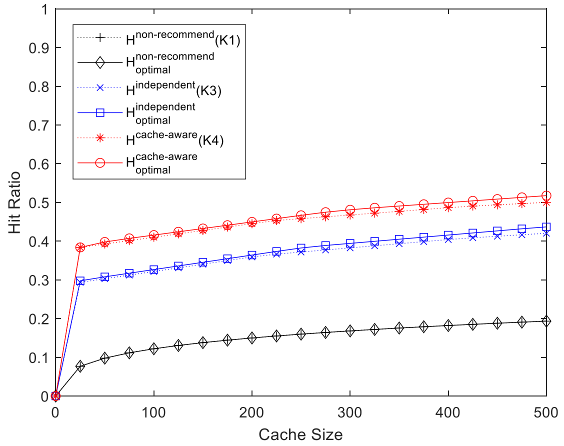

FIGURE 15. Hit ratio when the degree of personalization is high.

of personal content because the preferences of each device must be considered.

In Figures 17 and 18, the other simulation parameters are the same as in Table 1, except that $N_{group} = 25$ to reduce the

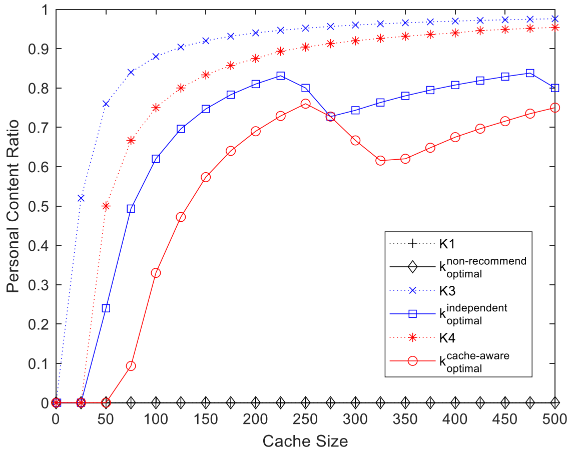

FIGURE 16. Ratio of personal content when the degree of personalization is high.

degree of personalization. Groups represent genres or categories, so a device or a piece of personal content can belong to multiple groups. Making such generalizations would require many assumptions and complex equations, so we leave this as future work in this paper. In the absence of recommendations from a device, average preferences are considered, whereas in the presence of recommendations on a device, the individual preferences of each device become important, and caching is necessary to account for this. The ratio of individual preferences to average preferences for personal content is called the degree of personalization in this paper, and is simply denoted by N_{group}. If a device is interested in multiple genres, or if a single piece of personal content belongs to multiple categories, the degree of personalization may decrease, depending on the assumptions. Also, if we can predict the movement patterns of devices so that we can predict in advance which devices will come into the coverage area of the cache, we can consider their personal content and the degree of personalization may decrease. Figure 17 shows the hit ratio as a function of cache size, and Figure 18 shows the proportion of personal content stored in the cache as N_{group} becomes smaller. We can see that the proportion of personal content in the cache increases.

In Figures 19 and 20, the other simulation parameters are the same as in Table 1, except that $K_{recommend} = 50$. Increasing the number of recommended contents may decrease the preference increase due to recommendations, β, but this is not reflected in this simulation for simple comparisons with other figures. Figure 19 shows the hit ratio as a function of the cache size. In the figure, the hit ratio increases as the number of recommended contents increases while β remains unchanged. In practice, as the number of recommended contents increases, the value of β will become smaller, so the hit ratio may increase or decrease depending on how much β decreases. If we can obtain a function of β according to $K_{recommend}$, we could obtain the optimal value of $K_{recommend}$ to maximize the hit ratio. Figure 20 shows the proportion of

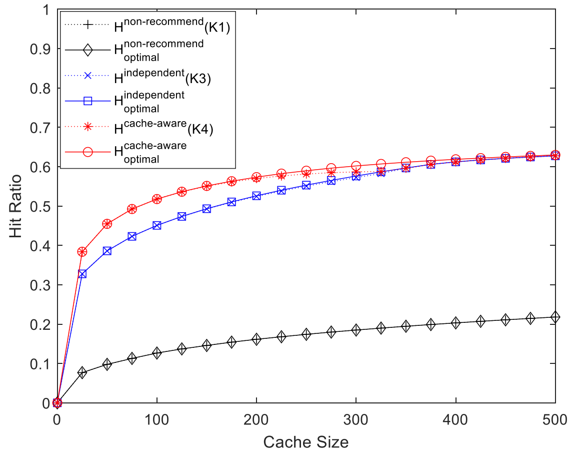

FIGURE 17. Hit ratio when the degree of personalization is low.

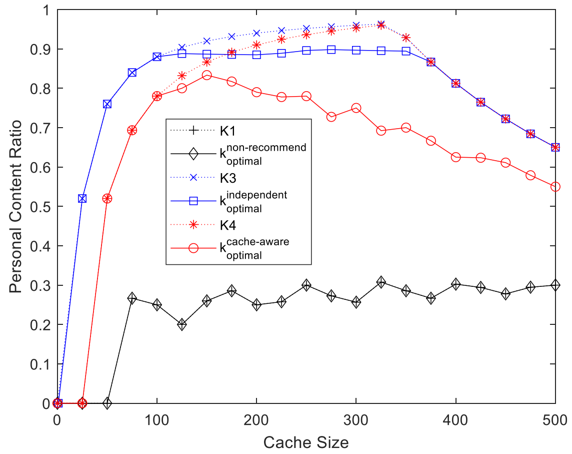

FIGURE 18. Ratio of personal content when the degree of personalization is low.

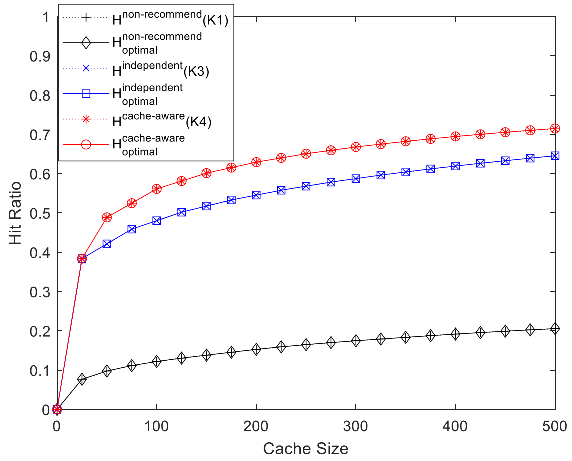

FIGURE 19. Hit ratio when the number of recommended contents is large.

personal content stored in the cache. As more content is being recommended while β remains unchanged, the contribution of non-recommended content to the hit ratio becomes smaller and the approximated equations fit more accurately.

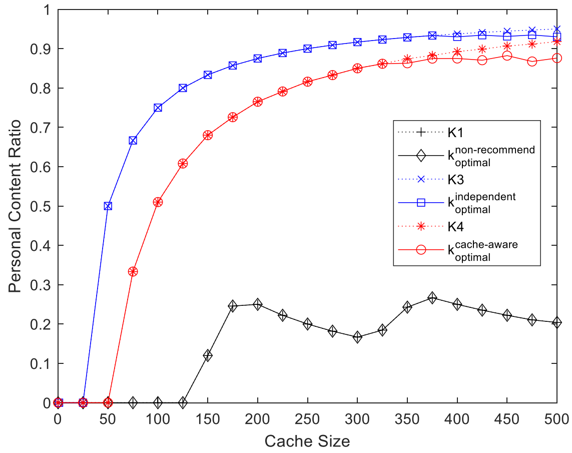

FIGURE 20. Ratio of personal content when the number of recommended contents is large.

V. CONCLUSION

In this paper, we discussed which types of content should be stored in a cache when a content recommendation system is present on a device. Caching schemes can be significantly different in the presence of recommendation systems than in the absence of recommendation systems. When no recommendations are made on a device, content with high average preferences should be cached, whereas when a recommendation system is present on a device and content that is personally preferred by each device is recommended, the caching system should take into account individual preferences for each device. When the degree of personalization is high, and thus the average preference for personal content is low, it is necessary to store mostly common content when no recommendations are made. On the other hand, with recommendations, it is necessary to store a larger amount of personal content, especially when the cache size is huge compared to the number of recommended contents. When recommendation systems are independent of the caching system, it is necessary to store a larger amount of personal content in the cache than for cache-aware recommendations, because each device will recommend personally preferred content regardless of the cached content. If the cache size is very large compared to the number of recommended contents, the consideration of cached content in recommendation systems may become less important, since a large amount of personal content should be stored in the cache anyway.

For the sake of simplicity, this paper assumes that a device or a piece of personal content belongs to only one group, but we can extend the discussion to allow a device or a piece of personal content to belong to multiple groups. In addition, content is divided into two types of groups: common content and personal content, but there will be many other types in between, and the discussion can be extended to account for this. This paper also assumes that we do not know which devices will enter the coverage area of a cache, but it will be possible to extend the study to the cases where we can predict the mobility patterns of devices or the social relationships between devices.

REFERENCES

[1] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[2] M. Zhang, H. Luo, and H. Zhang, "A survey of caching mechanisms in information-centric networking," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1473–1499, 3rd Quart., 2015.

[3] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.

[4] E. Zeydan, E. Bastug, M. Bennis, M. A. Kader, I. A. Karatepe, A. S. Er, and M. Debbah, "Big data caching for networking: Moving from cloud to edge," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 36–42, Sep. 2016.

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

[6] J. Yao, T. Han, and N. Ansari, "On mobile edge caching," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2525–2553, 3rd Quart., 2019.

[7] L. Li, G. Zhao, and R. S. Blum, "A survey of caching techniques in cellular networks: Research issues and challenges in content placement and delivery strategies," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1710–1732, 3rd Quart., 2018.

[8] J. Song and W. Choi, "Mobility-aware content placement for device-to-device caching systems," *IEEE Trans. Wireless Commun.*, vol. 18, no. 7, pp. 3658–3668, Jul. 2019.

[9] D. Liu and C. Yang, "Caching at base stations with heterogeneous user demands and spatial locality," *IEEE Trans. Commun.*, vol. 67, no. 2, pp. 1554–1569, Feb. 2019.

[10] Y. Fan, B. Yang, D. Hu, X. Yuan, and X. Xu, "Social- and content-aware prediction for video content delivery," *IEEE Access*, vol. 8, pp. 29219–29227, 2020.

[11] J. Lee, S. H. Lee, M. Rim, and C. G. Kang, "System-level spatiotemporal offloading with inter-cell mobility model for device-to-device (D2D) communication-based mobile caching in cellular network," *IEEE Access*, vol. 8, pp. 51570–51581, 2020.

[12] R. Zhang, S. Jia, Y. Ma, and C. Xu, "Social-aware D2D video delivery method based on mobility similarity measurement in 5G ultra-dense network," *IEEE Access*, vol. 8, pp. 52413–52427, 2020.

[13] M. Rim and C. G. Kang, "Content prefetching of mobile caching devices in cooperative D2D communication systems," *IEEE Access*, vol. 8, pp. 141331–141341, 2020.

[14] M. Rim and C. G. Kang, "Peak-hour caching schemes of mobile devices for overload cells in wireless caching systems," *IEEE Access*, vol. 8, pp. 195274–195289, 2020.

[15] M. Rim and C. G. Kang, "Cache partitioning and caching strategies for device-to-device caching systems," *IEEE Access*, vol. 9, pp. 8192–8211, 2021.

[16] P. Sermpezis, T. Giannakas, T. Spyropoulos, and L. Vigneri, "Soft cache hits: Improving performance through recommendation and delivery of related content," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1300–1313, Jun. 2018.

[17] D. Zheng, Y. Chen, M. Yin, and B. Jiao, "Cooperative cache-aware recommendation system for multiple internet content providers," *IEEE Wireless Commun. Lett.*, vol. 9, no. 12, pp. 2112–2115, Dec. 2020.

[18] G. Yu, Z. Chen, and J. Wu, "Content-aware personalized sharing based on cooperative user selection and attention in mobile Internet of Things," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 1, pp. 521–532, Mar. 2023.

[19] M. Lee and Y. P. Hong, "Socially-aware joint recommendation and caching policy design in wireless D2D networks," in *Proc. ICC*, 2021, pp. 1–6.

[20] Y. Fu, L. Salan, X. Yang, W. Wen, and T. Q. S. Quek, "Caching efficiency maximization for device-to-device communication networks: A recommend to cache approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 10, pp. 6580–6594, Oct. 2021.

[21] Y. Hua, Y. Fu, and Q. Zhu, "On cost minimization for cache-enabled D2D networks with recommendation," *China Commun.*, vol. 19, no. 11, pp. 257–267, Nov. 2022.

[22] M. Song, H. Shan, Y. Fu, H. H. Yang, F. Hou, W. Wang, and T. Q. S. Quek, "Joint user-side recommendation and D2D-assisted offloading for cache-enabled cellular networks with mobility consideration," *IEEE Trans. Wireless Commun.*, vol. 22, no. 11, pp. 8080–8095, Nov. 2023.

[23] Z. Zhao, H. Gao, W. Hong, X. Duan, and M. Peng, "Joint design of content delivery and recommendation in wireless caching networks," *China Commun.*, vol. 18, no. 11, pp. 61–75, Nov. 2021.

[24] D. Yu, T. Wu, C. Liu, and D. Wang, "Joint content caching and recommendation in opportunistic mobile networks through deep reinforcement learning and broad learning," *IEEE Trans. Services Comput.*, vol. 16, no. 4, pp. 2727–2741, Jul./Aug. 2023.

[25] C. Sun, X. Li, J. Wen, X. Wang, Z. Han, and V. C. M. Leung, "Federated deep reinforcement learning for recommendation-enabled edge caching in mobile edge-cloud computing networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 3, pp. 690–705, Mar. 2023.

[26] M. Vitoropoulou, K. Tsitseklis, V. Karyotis, and S. Papavassiliou, "Caching, recommendations and opportunistic offloading at the network edge," in *Proc. 17th Int. Conf. Mobility, Sens. Netw. (MSN)*, Dec. 2021, pp. 112–119.

[27] B. Zhu and W. Chen, "Coded caching with moderate recommendation: Balancing delivery rate and quality of experience," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1456–1459, Oct. 2019.

[28] D. Wei and S. Han, "An experimental study of recommendation for wireless edge caching," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Aug. 2022, pp. 731–736.

[29] Y. Li, L. Chen, H. Shi, X. Hong, and J. Shi, "Joint content recommendation and delivery in mobile wireless networks with outage management," *Entropy*, vol. 20, no. 1, pp. 1–24, Jan. 2018.

[30] Y. Fu, Y. Zhang, A. Wong, and T. Q. S. Quek, "Revenue maximization: The interplay between personalized bundle recommendation and wireless content caching," *IEEE Trans. Mobile Comput.*, vol. 22, no. 7, pp. 4253–4265, Jul. 2023.

[31] G. Ahani and D. Yuan, "Optimal content caching and recommendation with age of information," *IEEE Trans. Mobile Comput.*, vol. 23, no. 1, pp. 689–704, Jan. 2024.

[32] M. Si, Q. Liu, Z. Zhou, and W. Yang, "Edge caching strategy based on user's long and short term interests," in *Proc. AICIT*, 2023, pp. 1–5.

[33] T. Giannakas, P. Sermpezis, and T. Spyropoulos, "Network friendly recommendations: Optimizing for long viewing sessions," *IEEE Trans. Mobile Comput.*, vol. 22, no. 3, pp. 1633–1645, Mar. 2023.

[34] A. C. B. L. Monç ao, S. L. Correa, A. C. Viana, and K. V. Cardoso, "Combining resource-aware recommendation and caching in the era of MEC for improving the experience of video streaming users," *IEEE Trans. Services Comput.*, vol. 16, no. 3, pp. 1698–1712, 2023.

[35] K. Shi, Y. Fu, and K. Hung, "A diversified recommendation scheme for wireless content caching networks," *IEEE Internet Things J.*, early access, Dec. 15, 2023, doi: 10.1109/JIOT.2023.3343364.

[36] M. Rim, S. Chae, and C. G. Kang, "Interference mitigation and D2D parameter estimation for distributed-control D2D underlay systems," *Trans. Emerg. Telecommun. Technol.*, vol. 28, no. 1, pp. 1–10, Jan. 2017.

[37] M. Rim and C. G. Kang, "Carrier sensing for OFDMA-based D2D groupcasting systems," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2301–2310, Mar. 2017.

[38] S. Zhang, J. Liu, H. Guo, M. Qi, and N. Kato, "Envisioning device-to-device communications in 6G," *IEEE Netw.*, vol. 34, no. 3, pp. 86–91, May/Jun. 2020.

MINJOONG RIM received the B.S. degree in electronics engineering from Seoul National University, Seoul, South Korea, in 1987, and the Ph.D. degree in electrical and computer engineering from the University of Wisconsin–Madison, Madison, WI, USA, in 1993. From 1993 to 2000, he was with Samsung Electronics. He is currently a Professor with the Department of Information and Communication Engineering, Dongguk University, Seoul. His research interests include mobile communication and wireless communication.