Individual Preference Aware Caching Policy Design in Wireless D2D Networks

Ming-Chun Lee, *Student Member, IEEE*, and Andreas F. Molisch, *Fellow, IEEE*

Ming Hsieh Department of Electrical and Computer Engineering
University of Southern California
Los Angeles, CA, USA
Email: mingchul@usc.edu, molisch@usc.edu

Abstract

Cache-aided wireless device-to-device (D2D) networks allow significant throughput increase, depending on the concentration of the popularity distribution of files. While many previous investigations assumed that all users have the same preference, this may not be true in practice. This work investigates whether and how the information about *individual* preferences can benefit such cache-aided D2D networks. Considering a clustered network, we derive a network utility that considers both the user distribution and channel fading effects, and formulate a utility optimization problem. This formulation can be used to optimize several important quantities, including throughput, energy efficiency (EE), and hit-rate, and solve different tradeoff problems. We provide designs of caching policies that solve the utility maximization problem with and without coordination between the devices, and prove that the coordinated design can obtain the stationary point under a mild assumption. Using simulations of practical setups, we show that by properly exploiting the diversity entailed by the individual preferences, performance can be improved significantly. Besides, different types of tradeoffs exist between different performance metrics, and they can be managed by means of proper caching policy and cooperation distance design.

I. INTRODUCTION

Wireless networks have been strained by the rapid increase of wireless data traffic, which is predicted to continue over the next several years [2]. Among all the wireless applications, on-demand
video accounts for the largest portion of this traffic. Thus, finding an efficient approach to support this application is a paramount issue for modern wireless communication systems. Conventional approaches for increasing throughput, such as, cell densification, large-scale antenna systems, and millimeter-wave communications [3], are deemed insufficient because of the potential high cost when obtaining more physical resources. Different from those approaches that tend to improve wireless networks without regard to the type of data that are to be transmitted, video caching at the wireless edge exploits the unique video accessing behavior of typical consumers and the cheap storage resources to trade memory for bandwidth. In essence, different users cache different popular video files on their devices; a file request can then either be satisfied from a user's own cache, or through D2D communication with a nearby user that has stored the requested file. The potential of D2D-based video caching renders it widely discussed in recent years [4]–[11], and existing papers have demonstrated that, using either theoretical or empirical approaches, the wireless video caching can improve the throughput by orders of magnitude [8]–[11].

A. Literature Review and Motivations

Cache-aided D2D has demonstrated the ability to significantly improve network performance without the need for newly installed infrastructure and complicated coding. Thus, numerous papers have been published on this topic, aiming to understand and improve the hit-rate (file outage) [9], [16], throughput [9], [17], [18], energy efficiency (EE) [18], [19], and latency of the networks [7], [34]. For example, the theoretical throughput-outage tradeoff was investigated in [9], [10]. The difference between the hit-rate and throughput was discussed in [16]. By considering clustering networks, caching policy and cluster size are optimized for throughput in [17]. In [18], the optimizations of throughput, EE, and their tradeoff were investigated. Battery life was taken into account for reducing energy consumption in [19]. In [7], caching policies aiming to optimize the network delay were proposed. Not only different design goals, but also different techniques and scenarios were considered for cache-aided D2D. Stochastic geometry was exploited to analyze networks and propose caching policy designs in [20], [21]. Considering MIMO systems, scaling laws of throughput were discussed in [22] and [23]. In [24], mobility was leveraged to enhance network performance. To accommodate the environmental dynamics,

¹Concentrated popularity distribution of video files can also be exploited in other ways, e.g., femtocaching [12]–[14] and coded multicasting [10], [15]; those approaches are outside the scope of this paper.
a cache replacement approach was proposed in [25]. We note that the field of cache-aided D2D has been of great interest and several hundred related papers have been published. Hence the above literature review by necessity cites only a sample of papers and topics.

Most of the existing papers for cache-aided D2D networks consider a homogeneous preference model, which assumes all users have the same file preference - in other words, each user requests files independently and randomly according to the same popularity distribution. However, this model is at best an approximation, because different users indeed have different taste and preferences. Such heterogeneity of preferences of users has been observed in [26] and modeled in recent works [27], [28], [32]. Furthermore, based on real-world data, results in [26] have shown that leveraging the individual (user) preferences indeed can improve network performance. Thus, based on the above observations, it becomes clear that by replacing homogeneous modeling with heterogeneous modeling, cache-aided D2D networks could be further improved since the homogeneous preference model can only be an approximation of the true user behavior, and designs based on this model are restricted by their lack of considering individual user preferences.

Consequently, recently investigations started to consider individual preferences and showed the possibility of using this information to improve wireless caching networks [26], [29]–[34]. In [26], individual preferences were studied, and a machine learning approach was used to learn user preferences and decide which video to preload onto a local device cache purely based on preference of the particular user. While this kind of approach (also known as the “Netflix challenge”) is very important for recommendation systems and preloading on individual devices, it is not applicable to cache-aided D2D networks. When files are cached at BSs, an individual preference aware weighted sum utility of users was formulated and optimized in [29]. In [30], by considering users in different groups to have different preferences for files, a caching policy was designed to maximize the successful file discovery probability of different groups without considering the possible interference. In [31], a content push strategy was designed to maximize the D2D offloading gain for a particular demand realization by jointly considering the influences of the user preference and sharing willingness. In [32], user preferences were used to maximize the offloading probability without considering details of the physical layer. Using individual preference and user similarity, [33] proposed a caching content allocation approach to maximize

---

Note that most of the paper discussed below were published in parallel with, or after, the conference version of our paper [1].
a specifically defined utility. While the focus is on estimating the individual preferences using a learning-based algorithm, in [34], a caching policy exploiting the estimated preferences is provided to minimize the average delay of D2D networks. Despite this progress, understanding of how to use individual preference to improve the cache-aided D2D networks is still far from complete. Especially, it is unclear whether considering individual preferences can significantly improve the network in practice and how different performance metrics, such as throughput, EE, and hit-rate, interplay and trade off with one another. Most importantly, existing papers lack of providing sufficient evaluations based on real-world data. Our paper aims to resolve these issues.

B. Contributions

We consider BS-assisted cache-aided D2D networks, where users can obtain the desired files from the BS, caches of neighboring users via D2D links, and their local caches. Using different approaches to obtain the files thus leads to different utilities of the networks. Allowing different users to have different random caching policies, our goal is to maximize the utility by appropriately designing the individual preference aware caching policies. Based on clustering and random-push scheduling [18], we analyze the network and propose a utility maximization problem formulation that is non-convex. Observing that this utility maximization problem can be cast as different practical problems, such as, the throughput, hit-rate, and EE optimization problems, and several tradeoff problems, we provide discussions to reveal their relationship. Besides, we also discuss how the proposed utility maximization problem can accommodate scenarios with different fading and user distributions.

To solve the utility maximization problem, we first propose a coordinated caching policy, in which users need coordination when designing the policy. A solution approach that iteratively updates the caching policies of users is proposed. We prove that this approach can improve at each iteration and converge to a stationary point under a mild assumption. We also propose a non-coordinated approach by assuming all users to adopt the same caching policy while knowing the individual preferences. We evaluate the proposed caching policies in networks considering realistic setups; especially, we adopt a practical individual preference generator proposed in [27] based on extensive real-world data. The results show that by properly exploiting the information of individual preferences, network performance can be significantly improved. We also compare

3Note that this is different from designing a caching policy considering a homogeneous user preference.
between designs optimizing throughput, EE, and hit-rate, and investigate the influences of the cooperation range of the D2D network. The results indicate that tradeoffs exist between these important metrics, and we can manage the tradeoffs by properly exploiting our designs. Finally, we show how the proposed systems can be used as good reference systems for obtaining effective designs in networks with complicated scheduling. We emphasize that this is the first work to validate the benefits of exploiting the individual preferences of users and obtain insights by simulations based on extensive real-world data. To sum up, our paper has the following contributions:

- Considering the individual preferences, we analyze a cache-aided network and formulate a network utility maximization problem.
- Based on the utility maximization problem, we provide designs optimizing the network utility and several practically important metrics, e.g., throughput, EE, and hit-rate.
- Considering realistic setups based on extensive real-world data, comprehensive simulations are conducted to show the benefits of exploiting individual preferences and provide insights.
- We use simulations to demonstrate that tradeoff exists between different metrics, and we could obtain efficient tradeoffs by appropriately designing the caching policy and cluster size. We also demonstrate how the results in this paper can help in designing caching policies in the networks without tractable problem formulations.

C. Organization of This Paper

The remainder of the paper is organized as follows. In Sec. II, the network and individual preference models are presented. We provide analysis of the networks, and formulate the utility maximization problem in Sec. III. Also in Sec. III, we relate the proposed problem to different practical problems and show the effects of fading. The coordinated and non-coordinated caching policy designs are proposed in Sec. IV. Extensive simulation results are provided in Sec. V. We conclude the paper in Sec. VI. Proofs are relegated to Appendices.

II. NETWORK AND INDIVIDUAL PREFERENCE MODELS

We consider a BS-assisted cache-aided wireless D2D network with a single BS. Users can obtain desired files by retrieval from their own caches, D2D communications, or BS links. The file library consists of $M$ files, and we assume all files have the same size for simplicity. Each user is able to cache $S$ files in the device storage. Users can be active or inactive: an active user
is a user who places a request that needs to be satisfied and participates in the D2D cooperation (i.e., sends files to other users that request them); an inactive user is a user who does not place a request of its own but still participates in the D2D cooperation.

Following a widely used model [9], [17], [18], [22], [23], we consider a clustering network, in which a cell is divided into equal-sized square clusters with side length $D$. Users are allowed to cooperate via D2D communications only with users in the same cluster. We assume no interference between users of different clusters; this can be achieved by letting different clusters use different time/frequency resources with “spatial reuse”. The “communication radius” or “cluster size” we henceforth reference thus corresponds to the dimension $D$ of such a cluster, not the cell radius of the macro BS. We thus focus on a single cluster. We denote the number of active users in a cluster as $K_A$; the number of inactive users as $K_I$. The total number of users is then $K = K_A + K_I$.

We consider serving users via “random-push” scheduling [18], which functions as follows. For a cluster, the BS first randomly selects an active user without knowing whether this request can be satisfied by the user’s own (local) cache. If the selected user can obtain the desired file from the local cache, i.e., the desired file is actually cached by the selected user, the user request is satisfied immediately. Otherwise, the user checks whether other users in the D2D network store the desired content and whether the channel quality between the user and the other users storing the desired file is sufficiently good for D2D communications. If so, a D2D link is used to transmit the desired content; otherwise, the user needs to use the BS link to access the content. It is assumed that the BS has an unlimited bandwidth backhaul to repositories that store all files in the library, which guarantees that any request from a selected user can always be satisfied, though potentially at high cost. After the scheduling for the selected user, the remaining active users check whether their requests can be satisfied by files in their local caches. If yes, their requests are satisfied. Clearly, such scheduling can guarantee that at least one user is served and all users are scheduled fairly.

That being said, our results can be easily extended to multi-cluster scenarios provided that the intra-cluster interference of D2D links are appropriately handled, e.g., [18].

Of course, there exists more effective scheduling, e.g., the priority-push scheduling [18], though at the price of higher complexity. Most importantly, it is very challenging to obtain tractable formulations for such advanced scheduling scheme [18], [35]. On the other hand, the random-push scheduling leads to tractable expressions for different critical metrics and is easy to implement; thus serving as a good reference system.
Individual preferences for requesting video files are considered and represented in the form of probabilities. We denote the request probability of user $k$ for file $m$, i.e., probability that user $k$ wants file $m$ in the future as $a_{km}^k$, in which $0 \leq a_{km}^k \leq 1$, $\forall m, k$, and $\sum_{m=1}^{M} a_{km}^k = 1$, $\forall k$. Different users can have different caching policies. Denoting $b_{km}^k$ as the probability for user $k$ to cache file $m$, the caching policy of user $k$ is described by $\{b_{km}^k\}_M^1$, where $0 \leq b_{km}^k \leq 1$, $\forall m, k$, and $\sum_{m=1}^{M} b_{km}^k \leq S$. In the limiting case that $b_{km}^k$ becomes 1 or 0, the caching policy becomes deterministic. This is meaningful in the situation where the central controller knows a priori which users are going to be in a cluster, so that the caching policy can avoid detrimental file overlap (compare [7]). Such a deterministic predictability of user location occurs, e.g., in an office scenario, where the same people are in geographical proximity every day.

In this work, users could access the desired files from their local caches, caches of other users, and BS. We consider different utilities when different types of approaches are used. We denote the utility of accessing a file via a BS link as $U_B$, the utility of accessing a file via a D2D link as $U_D$, and the utility of accessing a file via the users’ own cache as $U_S$. We generally assume $U_B \leq U_D \leq U_S$. The utility can be used for different practical goals, such as throughput, cost, and etc.. We will discuss this in Sec. III.C.

III. CACHING POLICY DESIGN PROBLEM

Our goal in this work is to design caching policies that optimize the network utility by exploiting the knowledge of individual preferences. In this section, firstly the access probabilities of different accessing approaches for a user are derived. Based on the results, we formulate a caching policy design problem aiming to maximize the network utility.

A. Fundamental Access Probability

Consider the system model in Sec. II. We denote $\mathcal{U}_A$ as the index set of active users and $\mathcal{U}_I$ as the index set of inactive users, and $\mathcal{U} = \mathcal{U}_A \cup \mathcal{U}_I$. We denote the channel between user $k$ and

---

6An implementation of this caching policy could be found in [13].

7This implies that using self-access is superior to using a D2D link, and using a D2D link is superior to using a BS link. Besides, although we consider all users to have the same utility, the extension to the case that different users have different utilities is straightforward.
and user \( l \) as \( h_{k,l} \) and the corresponding signal-to-noise ratio (SNR) as \( \text{SNR}_{k,l} \). When user \( k \) is selected, the probability user \( k \) accesses the desired file through a BS link is

\[
P_B^k = \sum_{m=1}^{M} a^k_m \left[ \prod_{l \in \mathcal{U}} \left( 1 - b^l_m 1 \{h_{k,l},C\} \right) \right],
\]

where \( 1 \{h_{k,l},C\} = 1 \) if \( \log_2(1 + \text{SNR}_{k,l}) > C \); otherwise \( 1 \{h_{k,l},C\} = 0 \). \( \prod_{l \in \mathcal{U}} \left( 1 - b^l_m 1 \{h_{k,l},C\} \right) \) is the probability that file \( m \) is not in the caches of any user device, and \( a^k_m \prod_{l \in \mathcal{U}} \left( 1 - b^l_m 1 \{h_{k,l},C\} \right) \) is the probability that the user wants file \( m \) but file \( m \) is not in the caches of user devices. We define the self-access probability, i.e., the probability that user \( k \) can obtain the desired file from the local cache as

\[
P_S^k = \sum_{m=1}^{M} a^k_m b^k_m.
\]

By using \( P_B^k \) and \( P_S^k \), the probability that user \( k \) reaches the desired file via a D2D link is

\[
P_D^k = 1 - P_S^k - P_B^k = 1 - \sum_{m=1}^{M} a^k_m \left[ \prod_{l \in \mathcal{U}} \left( 1 - b^l_m 1 \{h_{k,l},C\} \right) \right] - \sum_{m=1}^{M} a^k_m b^k_m.
\]

**B. Utility Maximization Problem Formulation**

Now, we derive the expected utility of the network. We assume that, for any user \( k \), the channel gains of all the possibly associated D2D links are independent, i.e., \( 1 \{h_{k,l},C\}; \forall l \), are independent (see use cases in Sec. III.D). By using the results in Sec. III.A, when the user \( k \) is selected by the BS, the expected utility of the selected user \( k \) is

\[
U_k = \mathbb{E} \left[ U_D \cdot P_D^k + U_B \cdot P_B^k + U_S \cdot P_S^k \right].
\]

Given some weights \( w_1, w_2, ..., w_{K_A} \), the weighted sum expected utility of the selected users is

\[
U = \sum_{k \in \mathcal{U}_A} w_k \mathbb{E} \{ U_D \cdot P_D^k + U_B \cdot P_B^k + U_S \cdot P_S^k \} = \sum_{k \in \mathcal{U}_A} \frac{w_k}{K_A} \left[ U_D \cdot P_D^k + U_B \cdot P_B^k + U_S \cdot P_S^k \right]
\]

Since users not being selected by the BS can still check whether their desired files are cached in their local caches, we obtain additional local gains coming from the ability of users to satisfy their own requests. Thus, the expected utility of the network is

\[
U_{\text{net}} = U + U_{\text{local}} = U + U_S \sum_{k \in \mathcal{U}_A} \sum_{l \in \mathcal{U}_A, l \neq k} \sum_{m=1}^{M} w_l a^l_m b^l_m \frac{1}{K_A} - (U_B - U_D) \sum_{m=1}^{M} S_m + (K_A U_S - U_D) \sum_{m=1}^{M} \sum_{k \in \mathcal{U}_A} \frac{w_k a^k_m b^k_m}{K_A},
\]
where \( S_m = \sum_{k \in \mathcal{U}_A} \frac{w_k a_{m,k}}{K_A} \prod_{l \in \mathcal{U}} (1 - b_{m,l}^l L_{k,l}) \) and \( L_{k,l} = \Pr[\log_2(1 + \text{SNR}_{k,l}) > C] \).

**Proof.** The derivation of (6) is provided in appendix A.

Using (6), the caching policy design problem maximizing the network utility is

\[
\max_{b_{m,k}^k} U_{\text{net}} \quad \text{subject to} \quad \sum_{m=1}^{M} b_{m,k}^k \leq S, \forall k,
\]

\[
0 \leq b_{m,k}^k \leq 1, \forall k, m. \tag{7}
\]

We then have the following Proposition.

**Proposition 1:** The optimal solution of (7) must be tight at the equality of the sum constraint, i.e., for the optimal solution \((b_{m,k}^k)^*\), \(\forall k, m\), we have

\[
\sum_{m=1}^{M} (b_{m,k}^k)^* = S, \forall k. \tag{8}
\]

**Proof.** By (6), the first order partial derivatives of \(U_{\text{net}}\) is:

\[
\frac{\partial U_{\text{net}}}{\partial b_{m,k}^k} = -(U_B - U_D) \sum_{k \in \mathcal{U}_A} \frac{w_k a_{m,k}^k}{K_A} L_{k,j} \prod_{l \in \mathcal{U}_d, l \neq j} (1 - b_{m,l}^l L_{k,l}) + 1_{\{j \in \mathcal{U}_A\}} (K_A U_S - U_D) \frac{w_j a_{m,j}^j}{K_A}, \forall j, m, \tag{9}
\]

where \(1_{\{j \in \mathcal{U}_A\}} = 1\) when \(j \in \mathcal{U}_A\); otherwise \(1_{\{j \in \mathcal{U}_A\}} = 0\). Then since \(U_B \leq U_D \leq U_S, 0 \leq L_{k,l} \leq 1, \forall k, l, \) and \(0 \leq b_{m,k}^k \leq 1, \forall k, m\), we have \(\frac{\partial U_{\text{net}}}{\partial b_{m,k}^k} \geq 0, \forall k, m\). Therefore \(U_{\text{net}}\) is non-decreasing with respect to \(b_{m,k}^k, \forall k, m\). This indicates the optimal solution of (7) must be tight at the equality of the sum constraint. \(\square\)

### C. Interpretations of the Utility Maximization Problem and its relationship to Practice

The utility maximization problem can be specialized to various practically important problem, as elaborated in the following. Note that in this subsection, we consider the equal-weight case, i.e., \(w_1 = w_2 = \ldots = w_{K_A} = 1\), for notational simplicity, and the extensions to other weights are straightforward.

1) **Throughput Maximization Problem:** When letting \(U_B = T_B, U_D = T_D, U_S = T_S\), and \(T_B \leq T_D \leq T_S\), where \(T_B\) is the throughput of a BS link, \(T_D\) is the throughput of a D2D link, and \(T_S\) is the throughput of self-access, the utility maximization problem becomes the throughput maximization problem, in which the expected throughput is

\[
T_{\text{net}} = T_D + (T_B - T_D) \sum_{m=1}^{M} S_m + (K_A T_S - T_D) \sum_{m=1}^{M} \sum_{k \in \mathcal{U}_A} \frac{a_{m,k}^k b_{m,k}^k}{K_A}. \tag{10}
\]
2) Cost Minimization Problem: Then when letting $U_B = -C_B$, $U_D = -C_D$, $U_S = -C_S$, and $C_B \geq C_D \geq C_S$, where $C_B$ is the cost of a BS link, $C_D$ is the cost of a D2D link, and $C_S$ is the cost of self-access, the problem can be cast as the cost minimization problem, expressed as

$$\min_{b_{k, m}} C_{\text{net}} = C_D + (C_B - C_D) \sum_{m=1}^{M} S_m + (K_A C_S - C_D) \sum_{m=1}^{M} \sum_{k \in U_A} \frac{a_k b_k m}{K_A}$$

subject to

$$\sum_{m=1}^{M} b_{k, m} \leq S, \forall k, 0 \leq b_{k, m} \leq 1, \forall k, m.$$ 

(11)

It should be noted that when the cost is equal to energy consumption, the problem is the energy minimization problem.

3) Hit-Rate Maximization Problem: When letting $U_B = 0$, $U_D = 1$, and $U_S = \frac{1}{K_A}$, the problem is to maximize

$$H_{\text{net}} = \sum_{k=1}^{K} w_k \mathbb{E} \left[ \frac{P_k D}{P_k S} \right] = 1 - \sum_{m=1}^{M} S_m,$$

(12)

which is to maximize the file hit-rate of the network.

4) Throughput–Cost Weighted Sum Problem: To trade off between different metrics in a desired manner, a common approach is to maximize the weighted sum/difference of different metrics \[36\]. For example, considering the tradeoff between throughput and cost, we can consider maximizing

$$w_T T_{\text{net}} - w_C C_{\text{net}},$$

(13)

where $w_T \geq 0$ and $w_C \geq 0$. Such a weighted sum/difference problem is equivalent to the utility maximization problem as we let $U_B = w_T T_B - w_C C_B$, $U_D = w_T T_D - w_C C_D$, and $U_S = w_T T_S - w_C C_S$. Note that the similar concept can also be used for the throughput–hit-rate tradeoff.

5) Cost Efficiency Problem: In some situations, we also want to optimize the cost efficiency, e.g., we want to maximize EE (bits/Joule). Such efficiency maximization problem can be considered as a special case of the weighted sum problem as shown in the following.

Suppose we want to maximize EE. The problem is:

$$\max_{b_{k, m}, \forall k, m} EE = \frac{T_{\text{net}}}{C_{\text{net}}} = \frac{T_B + (T_B - T_B) \sum_{m=1}^{M} S_m + (K_A T_S - T_B) \sum_{m=1}^{M} \sum_{k \in U_A} \frac{a_k b_k m}{K_A}}{C_B + (C_B - C_B) \sum_{m=1}^{M} S_m + (K_A C_S - C_D) \sum_{m=1}^{M} \sum_{k \in U_A} \frac{a_k b_k m}{K_A}}$$

subject to

$$\sum_{m=1}^{M} b_{k, m} \leq S, \forall k, 0 \leq b_{k, m} \leq 1, \forall k, m.$$ 

(14)

This problem is then equivalent to

$$\max_{t, b_{k, m}, \forall k, m} t$$

subject to

$$\frac{T_{\text{net}}}{C_{\text{net}}} \geq t,$$

$$\sum_{m=1}^{M} b_{k, m} \leq S, \forall k, 0 \leq b_{k, m} \leq 1, \forall k, m.$$ 

(15)

\[8\]The same concept can also be applied to trade off between more than two objectives.
Assuming that the optimal \( t^* \) is known, the problem in (15) is equivalent to finding the optimal policy in

\[
\max_{b^m_k, \forall k, m} T_{\text{net}} - t^* C_{\text{net}}
\]

subject to

\[
\sum_{m=1}^{M} b^k_m \leq S, \forall k, 0 \leq b^k_m \leq 1, \forall k, m.
\]

(16)

Observing (16), it is clear that we have a weighted difference problem, in which \( w_T = 1 \) and \( w_C = -t^* \). Thus, it can be cast into the utility maximization framework. Also, the optimal policy should result in \( T_{\text{net}} - t^* C_{\text{net}} = 0 \).

The remaining issue is to find \( t^* \). Assume that we can find the global optimum of the utility maximization problem. Considering the same problem in (16) but replacing \( t^* \) with \( t \), we know that if the solution results in a positive number, i.e., \( T_{\text{net}} - t C_{\text{net}} > 0 \), our solution can provide an EE larger than \( t \). On the other hand, if the solution gives \( T_{\text{net}} - t C_{\text{net}} < 0 \), we know that such \( t \) is not achievable. By adjusting \( t \) according to the results, we can then find the \( t \) such that it is arbitrarily close to \( t^* \). We note that this concept is identical to the approach for solving a quasi-convex problem [37]. Although the above approach is based on the assumption that the global optimum of the utility maximization problem is attainable, the concept can be applied even if the globally optimal solution might not be obtained [38]. This is because whenever a solution gives \( T_{\text{net}} - t C_{\text{net}} > 0 \), we know that such solution can achieve EE = \( t \). Then by carefully adjusting \( t \) and solving the utility maximization problem efficiently, we could still obtain an effective design.

D. Effects of the Statistics of Wireless Channels and User Distributions

By (6), it can be observed that the expected utility is influenced by the channel quality via \( L_{k,l} \). Thus understanding the general expression of \( L_{k,l} \) and its relationship to channel physics is important. In this section, we provide several useful expressions for \( L_{k,l} \) and discuss its relationship to the possible scenarios. It should be noted that if \( k = l, L_{k,l} = L_{k,k} = 1 \). Therefore in the following, we consider \( k \neq l \).

Let \( d_{k,l} \) be the distance between user \( k \) and user \( l \). We consider the input–output relationship between users \( k \) and \( l \) to follow the general expression:

\[
y_l = \sqrt{\text{PG}(d_{k,l}) s_{k,l} h_{k,l} x_k} + n_l,
\]

(17)

where \( y_l \) is the received signal at user \( l \), \( x_k \) is the transmit signal from user \( k \), \( \text{PG}(d_{k,l}) \) is the path gain effect (channel (power) gain averaged over small-scale and large-scale fading), \( s_{k,l} \) is

\[9\text{Since the utility maximization problem is non-convex, this is the case we might encounter.}\]
the shadowing power gain, $h_{k,l}$ is the small-scale fading amplitude, and $n_l$ is the Gaussian noise with power $\sigma_n^2$. Denote $E_D$ as the transmission power of the D2D link. By \((17)\), the received SNR is $\text{SNR}_{k,l} = \frac{E_D|h_{k,l}|^2 s_{k,l} \text{PG}(d_{k,l})}{\sigma_n^2}$, and therefore

$$L_{k,l} = \text{Pr} \left[ |h_{k,l}|^2 s_{k,l} \text{PG}(d_{k,l}) > \frac{\sigma_n^2(2^C - 1)}{E_D} \right]. \quad (18)$$

In the following, we show some practical examples and demonstrate the computations of \((18)\) using user fading distributions. The extensions to other models are feasible by leveraging the existing results of fading \[38\] and distance distributions \[39\].

1) **Case 1: Systems with effective quality control:** In modern wireless communication systems, approaches such as adaptive power control and frequency-and-antenna-diversity, are used to combat fading effects in wireless channels. Thus, in systems with effective quality control, we can assume that D2D channels between users in an area can be guaranteed, leading to $L_{k,l} = 1, \forall k, l$.

2) **Case 2: Systems with deterministic path-loss and shadow fading:** When users are in low-mobility or no-mobility modes, the joint effect of path-loss and shadow fading between users is deterministic. In this case, we focus on characterizing the small-scale fading. Thus,

$$L_{k,l} = \text{Pr} \left[ |h_{k,l}|^2 s_{k,l} \text{PG}(d_{k,l}) > \frac{\sigma_n^2(2^C - 1)}{E_D} \right], \quad (19)$$

where the closed-form expressions are attainable for commonly used fading distribution. For example, considering normalized Rayleigh fading whose average power is 1, we have

$$L_{k,l} = \exp \left[ -\frac{\sigma_n^2(2^C - 1)}{E_D s_{k,l} \text{PG}(d_{k,l})} \right], \quad (20)$$

3) **Case 3: K users uniformly distributed in a square with side length D and with shadowing and small-scale fading:** Here we consider using lognormal shadowing and normalized Rayleigh fading as an example. According to results in \[40\], the distance $d$ between two users independently and uniformly distributed over a square area with unit side length is described by the probability density function:

$$f_{sq}(d) = \begin{cases} 
2d(\pi + d^2 - 4d), & 0 \leq d \leq 1, \\
2d(-2 - d^2 + 4\sqrt{d^2 - 1} + 2\sin^{-1} \frac{2 - d^2}{d^2}), & 1 < d \leq \sqrt{2}. 
\end{cases} \quad (21)$$

Then when fixing the shadowing $s_{k,l}$, by using the property of Rayleigh fading again, we have

$$L_{k,l}(s_{k,l}) = \int_{0}^{\sqrt{2D}} \exp \left[ -\frac{\sigma_n^2(2^C - 1)}{E_D s_{k,l} \text{PG}(x)} \right] f[d = x])dx = \int_{0}^{\sqrt{2}} \exp \left[ -\frac{\sigma_n^2(2^C - 1)}{E_D s_{k,l} \text{PG}(dx)} \right] f_{sq}(x))dx. \quad (22)$$
Assume that the shadowing and small-scale fading effects of different links between different users are independent. We can then generalize (22) as

\[
L_{k,l} = \int_0^{\sqrt{2}} \left[ \int_0^{\infty} \exp \left( -\frac{\sigma_n^2(2^C - 1)}{E_D s \text{PG}(D_x)} \right) f_{s_{k,l}}(s) \, ds \right] f_{sq}(x) \, dx,
\]

where \( f_{s_{k,l}}(s) \) is the pdf of the shadowing effect for the channel link between user \( k \) and \( l \). Consider the mean and standard deviation of the lognormal distribution to be \( u_{dB} \) and \( \sigma_F \), respectively. We then obtain

\[
L_{k,l} = \int_0^{\sqrt{2}} \left[ \int_0^{\infty} \exp \left( -\sigma_n^2(2^C - 1) \right) \frac{20 \log(10)}{s \sigma_F \sqrt{2\pi}} \exp \left( \frac{-(20 \log_{10}(s) - u_{dB})^2}{2\sigma_F^2} \right) ds \right] f_{sq}(x) \, dx.
\]

It should be noted that the inner integral of (24) is the complement of the channel outage when the joint effect of the fading is the Suzuki distribution \([38]\).

IV. PROPOSED CACHING POLICY DESIGNS

A. Proposed Coordinated Caching Policy Design

To solve (7), we propose an approach that iteratively optimizes the caching policies of users. Specifically, denoting \( b_{k'} = [b_{k'}^1, ..., b_{k'}^M]^T \), for user \( k' \), we iteratively solve the following subproblem for different \( k' \):

\[
\begin{align*}
\max_{b_{k'}} U_{lp}^{k'} & = U_{\text{net}}(b_1, ..., b_{k'}, ..., b_K) \\
\text{subject to} & \sum_{m=1}^{M} b_{k'}^m = S, \\
& 0 \leq b_{m} \leq 1, \forall m,
\end{align*}
\]

in which, when \( k' \in U_A \), we obtain

\[
U_{lp}^{k'} = \sum_{k \in U_A} \frac{w_k U_D}{K_A} + (U_B - U_D) \sum_{m=1}^{M} \sum_{k \in U_A} \frac{w_k a_{m,k}}{K_A} \left[ \prod_{l \in U, l \neq k'} (1 - b_{m}^l L_{k,l}) \right] \\
+ (K_A U_S - U_D) \left[ \sum_{m=1}^{M} \sum_{k \in U_A, k \neq k'} \frac{w_k a_{m,k} b_{m}^k}{K_A} \right] \\
- \sum_{m=1}^{M} b_{m}^{k'} \left( U_B - U_D \sum_{k \in U_A} \frac{w_k a_{m,k}}{K_A} L_{k,k'} \left[ \prod_{l \in U, l \neq k'} (1 - b_{m}^l L_{k,l}) \right] + (U_D - K_A U_S) \frac{w_{k'} a_{m}^{k'}}{K_A} \right); \tag{26}
\]
when $k' \in \mathcal{U}_I$, we obtain

$$U_{LP}^{k'} = \sum_{k \in \mathcal{U}_A} \frac{w_k U_D}{K_A} + \left( U_B - U_D \right) \sum_{m=1}^M \sum_{k \in \mathcal{U}_A} \frac{w_k a_{m,k}}{K_A} \left[ \prod_{l \in \mathcal{U}, l \neq k'} (1 - b_{m,l}^{k'}) \right] \left( U_B - U_D \right) \sum_{k \in \mathcal{U}_A} \frac{w_{k}a_{m,k}}{K_A} L_{k,k'} \left[ \prod_{l \in \mathcal{U}, l \neq k'} (1 - b_{m,l}^{k'}) \right] \right).$$

We remark that (26) and (27) are simply reformulations of (6), in which we isolate the terms containing variables to be optimized. Observing (26) and (27), it is then clear that (25) is a linear program. Note that we consider only equality constraints in (25b) because of Proposition 1.

To solve (25), general linear program solvers can be applied. However, we herein provide a more insightful and efficient solution approach via using the analytical closed-form expressions in (26) and (27). By letting

$$U_{LP S}^{k',m} = \left\{ \begin{array}{ll}
(U_D - U_B) \sum_{k \in \mathcal{U}_A} \frac{w_k a_{m,k}}{K_A} L_{k,k'} \left[ \prod_{l \in \mathcal{U}, l \neq k'} (1 - b_{m,l}^{k'}) \right] \left( K_A U_S - U_D \right) \frac{w_{k}a_{m,k}}{K_A} & , k' \in \mathcal{U}_A, \\
(U_D - U_B) \sum_{k \in \mathcal{U}_A} \frac{w_k a_{m,k}}{K_A} L_{k,k'} \left[ \prod_{l \in \mathcal{U}, l \neq k'} (1 - b_{m,l}^{k'}) \right] & , k' \in \mathcal{U}_I,
\end{array} \right.$$

we notice that maximizing $U_{LP}^{k'}$ is equivalent to maximizing

$$\sum_{m=1}^M b_{m,k'}^{k'} U_{LP S}^{k',m}.$$  \hfill (29)

Then by observing that the optimal solution of (29) subject to constraints (25b) and (25c) can be obtained by allocating the cache space to the terms offering larger payoffs, the optimal solution of (25) is expressed as

$$(b_{m,k'}^{k'})^* = \left\{ \begin{array}{ll}
1, & m \in \Phi_{k'}, \\
0, & \text{otherwise},
\end{array} \right.$$  \hfill (30)

where $\Phi_{k'} = \{m : U_{LP S}^{k',m} \text{ is among the } S \text{ largest of all } U_{LP S}^{k',m}\}$. By iteratively solving (25) via using (30) for different $k'$ until convergence, the caching policy design problem in (7) can be effectively solved. Denote $\mathcal{B}_k = \{(b_{m,k}^1, ..., b_{m,k}^M)^T : \sum_{m=1}^M b_{m,k}^M = S; 0 \leq b_{m,k}^M \leq 1, \forall m\}$. The solution approach is summarized in Alg. 1. We note that since (30) suggests that the probability for a user to cache file $m$ is either 1 or 0, we actually eliminate the probabilistic interpretation and attain the deterministic policies of users. To characterize the performance of the proposed solution approach, we provide the following theorem:
Theorem 1: Alg. 1 is monotonically non-decreasing at each iteration and can converge to a stationary point if each iteration provides an unique maximizer.

Proof. See Appendix C. □

Algorithm 1: Iterative User-Based Caching Policy Design

At iteration $r$, choose an user $k'$ and update

$$b_{k'}^{r+1} = \arg \max_{b_{k'} \in B_{k'}} U(b_1^r, ..., b_{k'}^{r-1}, b_{k'}^r, b_{k'}^{r+1}, ..., b_K^r)$$

$$b_k^{r+1} = b_k^r, \forall k \neq k'$$

B. Proposed Non-Coordinated Caching Policy Design

The previous design needs coordination between users. Here we propose an intuitive design without coordination. By considering that users adopt the same caching policy $\{b_m\}_{1}^{M}$ but are aware of the individual preference of the demanding user set, the utility function $U_{\text{net}}$ is lower bounded as

$$U_{\text{net}} = \sum_{k \in \mathcal{U}} \frac{w_k U_D}{K_A} + (U_B - U_D) \sum_{m=1}^{M} S_m + (K_A U_S - U_D) \sum_{m=1}^{M} \sum_{k \in \mathcal{U}} \frac{a_{m,k}^k b_m}{K_A} \geq \sum_{k \in \mathcal{U}} \frac{w_k U_D}{K_A}$$

$$+ (U_B - U_D) \sum_{m=1}^{M} \sum_{k \in \mathcal{U}} \frac{w_k a_{m,k}^k (1 - b_m)(1 - b_m L_k^{\min})^{K-1} + (K_A U_S - U_D) \sum_{m=1}^{M} \sum_{k \in \mathcal{U}} \frac{w_k a_{m,k}^k b_m}{K_A},}$$

(31)

where $L_k^{\min} = \min_{l \in \mathcal{U},l \neq k} L_{k,l}$. We should note that the lower bound is tight at the equality if all the users encounter the same fading and are independently and randomly distributed in a D2D area. Denote the lower bound in (31) as $U_{\text{nc}}$, the non-coordinated caching policy design problem is equivalent to

$$\max_{b_m, \forall m=1,..,M} U_{\text{nc}}$$

subject to \( \sum_{m=1}^{M} b_m \leq S, 0 \leq b_m \leq 1, \forall m. \)

Since every user designs the caching policy by independently solving the same problem in (32), the design can be conducted without coordination. Note that to leverage the user preferences, users still need to exchange information regarding their preferences. The following proposition then indicates how we can solve (32):

If there does not exist a unique maximization, we will encounter a tie between different $U_{\text{LPS}}^{k',m}$, which is generally unlikely when users have different preferences on different files. Thus such unique maximizer assumption is considered mild.
Proposition 2: The problem in (32) is a standard concave optimization problem. Besides, the optimal caching policy must be tight at the equality of the sum constraint.

Proof. Trivial when looking at the gradient and Hessian of $U_{nc}$. □

V. NUMERICAL RESULTS

In this section, simulation results are provided to validate the analysis, evaluate the proposed designs, compare between different designs, and provide insights.

A. Simulation Setup

In the simulations, we evaluate the performance of a cluster whose coverage is a square with side length $D$. Users are uniformly distributed within the cluster. We assume the random-push scheduling for users unless otherwise indicated. We assume equal-weight for users, i.e., $w_k = 1, \forall k \in U_A$. We consider a practical channel model, consisting of the path-loss, shadowing, Rayleigh fading, and Gaussian noise, for D2D links. The noise power spectral density is $N_0 = -174$ dBm/Hz. The path-loss model of the D2D link between users $k$ and $l$ is described as [8], [38]

$$20 \log_{10} \frac{4\pi d_0}{\lambda_c} + 10\alpha \log_{10} \frac{d_{k,l}}{d_0},$$

(33)

where $d_0 = 10$ m is the breakpoint distance, $\lambda_c = \frac{3 \times 10^8}{f_c}$ m, where $f_c = 2$ GHz is the carrier frequency, $\alpha = 3.68$ is the path-loss exponent, and $d_{k,l}$ is the distance between users $k$ and $l$. The shadowing is modeled by a log-normal distribution with mean $\mu_{dB} = 0$ dB and standard deviation $\sigma_F = 8$ dB, and the small-scale fading is Rayleigh fading. We denote $E_D$ as the transmission power of the device, and $\text{SNR}_{\text{min}} = 5$ dB as the minimum SNR requirement for a successful transmission of a D2D link. Thus, $R_{\text{min}} = \log_2(1 + 3.16)$ is the minimum transmission rate of a D2D link. We then let $T_D = B_D R_{\text{min}}$ be the throughput of a D2D link, where $B_D = 20$ MHz is the bandwidth of a D2D link. We assume a BS link always exists when a user is scheduled to use it. Since the BS must supply users in many clusters, we assume that a BS link can only share $\frac{1}{100}$ of the BS resources. The transmission power for a BS link is $E_B = 26$ dBm, which is $\frac{1}{100}$ of the total 46 dBm of the BS power. Similar, the bandwidth of a BS link is $B_B = 200$ kHz, which is $\frac{1}{100}$ of the total 20 MHz bandwidth. We thus let $T_B = B_B R_{\text{min}}$. We assume no cost when users can obtain the desired file from their local caches, and consider $T_S = 2T_D$ to
indicate the slightly better quality of the video when self-access is possible. For simplicity, we assume here that energy cost is purely determined by RF energy required for transmission; access to storage and coding/decoding is assumed to be negligible in comparison. Thus, based on the above setup, we obtain $C_B = E_B$, $C_B = E_D$, and $C_S = 0$; the network EE is thus given as $EE = \frac{T_{\text{net}}}{C_{\text{net}}}$ by definition in Sec. III.C.5.

To generate the practical individual preference probabilities for users, the generation approach and parameters in [27] are adopted. Individual preference probabilities in [27] are modeled by a hierarchical structure in which the preference probability of a user for a file is modeled as the probability that a user wants a certain video genre, and then the conditional probability a user wants a file within the genre. Therefore, each file in the model can be categorized into a genre, and we have $M = \sum_{g=1}^{G} M_g$, where $M_g$ is the number of files in a genre and $G$ is the total number of genres in the library.

In the following, we want to show the benefits of exploiting the individual preferences. To provide a fair comparisons, we observe that both the coordinated and non-coordinated designs can be implemented either by using the knowledge of individual preference probabilities as in Sec. IV or simply by using the system-wide popularity distribution. When implementing using system popularity distribution, the individual preference probabilities of all users in (14) are replaced by the global (system) popularity distribution, i.e., all users assume the same preference probabilities described by the global popularity distribution. To generate the system popularity of the simulations, we average the individual preference probabilities of 10000 users generated by the same generation model, meaning that the global popularity distribution is constructed by averaging 10000 individuals. According to our experiences, such construction is relatively stable and only minor differences are observed between different constructions.

**B. Effects of the Individual Preferences and Network Parameters**

In this subsection, we validate the analytical results provided in Sec. III and show the results with different network parameters. For all simulations in this subsection, we adopt $D = 80$ and $E_D = 20$ dBm. In the figures, the results of the designs adopting the individual preferences are

\footnote{Note that although we can immediately obtain the file when this file is in the local cache, the throughput is bounded by the rate that the user watches the file. Also, mathematically, we should not let $T_S$ go to infinity if we want meaningful results.}

\footnote{Please refer to [28] for the detailed recipe of the generator.
Fig. 1: Evaluations of different designs in terms of throughput.

labeled by “+ Individual”; the designs adopting the global popularity distribution are by labeled “+ Global”.

We first verify our analytical results and investigate the effectiveness of the coordinated and non-coordinated caching policies. In Fig. 1 we consider both $S = 5$ and $S = 20$ and no inactive users ($K_1 = 0$), and evaluate different designs in terms of the throughput. The curves labeled with “Analytical” are directly computed from expressions in Sec. III; the curves with “Simulations” are results of Monte-Carlo simulations. We observe that the analytical results match the simulations very well, validating our derivations in Sec. III. Besides, we see that the coordinated design exploiting the individual preferences can significantly outperform the corresponding design without using the individual preferences. On the other hand, we notice that the non-coordinated design provides almost no improvement even if the information of the individual preferences is exploited. This indicates that simply knowing individual preferences without actually letting different users have different caching policies such that the diversity of the user preferences is leveraged is not effective.

In Fig. 2 we consider both $S = 5$ and $S = 20$ and no inactive users ($K_1 = 0$), and evaluate different designs in terms of the EE. We observe again that our analytical results match the simulation results. Besides, the coordinated design exploiting the individual preferences once again show the benefits of using the individual preferences, while the non-coordinated design still cannot induce obvious improvement. Therefore, a better non-coordinated design that can smartly provide the sufficiently large degree of the heterogeneity between users in terms of the EE.

$^{13}$There are some cases that the non-coordinated design with individual preferences can provide better improvement than the results in Fig. 1 (as well as in Fig. 2), e.g., when the design goal is to minimize cost or maximize hit-rate. However, the improvement is still less than the coordinated design.
caching policy without much imposed coordination between users is necessary. This is one of our important future directions.

We next show the impact of inactive users. In Fig. 3 we consider $S = 20$, and compare results between networks with two different numbers of inactive users, i.e., $K_I = 0$ and $K_I = 25$. We show only the results of the coordinated designs. From the figure, we see that the benefits of the inactive users are more significant when the number of active users in a cluster is small. Compared to the results with no active users, when $K_A = 3$, the improvement of having 25 inactive users is 96%; when $K_A = 53$, the improvement of having 25 inactive users is 3.0%. This is clearly because although the inactive users can improve the hit-rate, such hit-rate improvement becomes insignificant for the throughput when there are too many users (in the same cluster) to share a single D2D band and the number of active users is enough for bringing good hit-rate without the aid of inactive users. This implies that when the number of inactive users is large, we might want to have multiple D2D links to benefits more from the inactive users or adjust the number of users in a cluster by reducing the cluster size.\textsuperscript{14}

\textbf{C. Tradeoff Behaviors between Different Performance Metrics}

In this subsection, we compare different designs and show the tradeoffs between throughput, EE, and hit-rate. Specifically, in all the following figures, we compare between different coordinated designs in pursuit of different goals, i.e., throughput, EE, hit-rate, and the throughput–hit-rate tradeoff, in terms of all these performance metrics. For the throughput–hit-rate tradeoff

\textsuperscript{14}Of course, either approach should be subject to careful considerations between different aspects, such as interference management, power control, reduction of hit-rate, and etc..
design, we design the caching policies by maximizing $T_{\text{net}} + \zeta T_D K_A H_{\text{net}}$, i.e., considering $U_B = T_B$, $U_D = T_D + \zeta K_A T_D$, and $U_S = T_S + \zeta T_D$. Such tradeoff design is interpreted as a weighted sum of throughput and hit-rate, in which the throughput is rendered the weight 1 and the hit-rate rendered the weight $\zeta K_A T_D$. Note that the term $T_D$ in the weight of the hit-rate is basically to calibrate between different units. This tradeoff design is then labeled by “TH-HIT Tradeoff - $\zeta$” in the figures, where $\zeta$ might be different to indicate different tradeoff behaviors. We also compare with the baseline selfish design, in which each user selfishly caches the files according to their own preferences without considering other users. Such design can be considered as an extreme as opposed to the maximum hit-rate design which maximizes the cooperation between users.

Considering $S = 20$, $M = 934$, $R = 80$, $E_D = 13$ dBm, and $K_I = 0$, we compare different designs in Fig. 3. Unsurprisingly, the throughput-based, EE-based, and hit-rate-based designs provide the best throughput, EE, and hit-rate, respectively. Besides, the selfish design is well-performing in terms of throughput, while it is very poor in terms of EE and hit-rate. This is because, when all users are active, the network throughput can be effectively enhanced by having large local gains. On the other hand, such selfish design inherently provides very poor hit-rate, leading to poor EE because of the frequent use of BS links. The hit-rate-based design provides poor throughput because it does not consider the local gains possibly brought by letting users to cache their desired files. In contrast, the throughput-based design is not effective in terms of hit-rate due to emphasis on obtaining the local gains. To strike a balanced viewpoint between them, the appropriate throughput–hit-rate tradeoff designs can efficiently trade throughput for hit-rate, resulting in significant improvement of the hit-rate with little degradation on throughput. By
adjusting ζ, we can effectively adjust the tradeoff behavior. Finally, we observe that to design a caching policy that is effective in terms of EE, we shall balance between throughput and hit-rate\footnote{Although not shown here for brevity, we see in some cases that the throughput–hit-rate tradeoff design can be near-optimal in terms of EE.}. Also, it is worthwhile noting that when compared to Figs.\footnote{The channel outage rate increases by only 0.012.} 1 and 2, the $E_D$ is significantly lower, resulting in significantly better EE. However, such transmission power reduction only slightly increases the channel outage\footnote{The channel outage rate increases by only 0.012.} so that the throughput (also the hit-rate) is almost identical to those in Figs. 2(b). This implies the usefulness of a good power control policy of the network.

In Figs. 5 and 6, we evaluate the proposed designs with respect to the cluster size $D$. Since changing the cluster size should accompany a suitable transmission power control of D2D links in order to appropriately manage the average SNR of the received signal and the interference between clusters, the power control policy proposed in [18] is adopted:

$$E_D = \left[ (\sqrt{K} - 1) \frac{d}{d_0} \right]^\alpha \cdot \left( \frac{4\pi d_0}{\lambda_c} \right)^2 \cdot \nu,$$

(34)

Fig. 4: Comparisons between different designs in terms of throughput, EE, and hit-rate.
where $K = 16$ is the reuse factor and $\nu = 2^{\frac{K}{2}} N_0 B_D$ is the maximal allowable interference between clusters\(^\text{17}\). Such power control policy can adjust the transmission power of devices such that the average SNR of received signal and inter-cluster interference are almost invariant when changing the cluster size. Since D2D links are expected to exist only for short-distance transmissions, we consider $D \leq 90$ m, resulting in $E_D \leq 20$ dBm when using (34) to adjust the power. To model the number of active and inactive users for a cluster size $D$, we consider Poisson point processes with $\lambda_A$ and $\lambda_I$ to represent the densities of active and inactive users, respectively. Thus, the number of active and inactive users are random variables described by the Poisson distributions with parameter $\lambda_A D^2$ and $\lambda_I D^2$, respectively. Also, to fairly accommodate the fact that a cluster has different number of users when $D$ is different, instead of directly looking at the throughput, we evaluate using the throughput per area (Bits/s/m\(^2\)). Similar to Fig. 4, we compare between different coordinated designs. In addition, we compare to another two reference curves: coordinated designs using global popularity distribution, labeled by “Global”, and the coordinated design with homogeneous modeling, labeled by “Homogo Model”. The former one is the same as in Figs. 1-3 that we design the policy using global popularity distribution while the users actually have different preferences; the latter one is that we design the policy using global popularity distribution while the users indeed have the same preferences following the global popularity distribution. This reference curve represents the performance of systems designing and evaluating with the homogeneous modeling as employed in previous papers - we want to see the influences on the performance of cache-aided D2D networks when changing from a homogeneous modeling to a more practical heterogeneous modeling.

In Fig. 5 we consider $S = 20$, $M = 934$, $\lambda_A = 0.01$, and $\lambda_I = 0$, i.e., no inactive users. We see that due to the interplay of the hit-rate, area efficiency, and the degree of diversity of preferences of users, the throughput of the throughput-based design fluctuates when $D$ ranges at $10 - 50$ m. Then it becomes relatively flat when $D$ is large because the contribution of the D2D transmission becomes minor as too many users share the same D2D band in a cluster. We can also see that the selfish design is relatively effective again because all users are active. Interestingly, we observe that the area throughput of the EE-based design first decreases with respect to $D$, and then increase as $D \geq 50$ m. This is because when $D$ is small, the EE-based

\(^{17}\)The value of $\nu$ is at the level of noise power so that we can ignore the inter-cluster interference in the simulations for brevity. $\nu$ is only used to compute the $E_D$.\n
design strives to increase hit-rate for optimal EE (thus sacrificing the local gains), and then when the hit-rate is sufficiently large, it gradually improves the local gains of the users, leading to a bounce back of the area throughput. As expected, hit-rate-based design provides the best hit-rate while the area throughput of the hit-rate-based design continuously decreases with respect to $D$ since it strives to improve hit-rate without considering the local gains. In contrast, the throughput-based design is again not effective in terms of hit-rate. In terms of EE, the EE-based design outperforms others significantly. Besides, it is observed that we can trade off between the area throughput and EE not only by means of using different caching policies but by means of using different cluster sizes. Thus, a network designer should take both caching policy and cluster size into consideration. Finally, we note that exploiting individual preferences can bring benefits as expected. Furthermore, we see that the proposed design can outperform the design operating with users with homogeneous distribution. Such result implies, rather than being detrimental, the diverse preferences of users on files can actually be used to further improve the network.

In Fig. 6, we conduct a similar evaluation as in Fig. 5 while we here adopt $\lambda_A = 0.005$ and $\lambda_I = 0.005$, i.e., there are some inactive users. We can observe that most of the phenomena
Fig. 6: Comparisons between different designs in terms of throughput, EE, and hit-rate with respect to cluster size with $\lambda_A = 0.005$ and $\lambda_I = 0.005$.

observed in Fig. 5 can be observed again. Besides, since we now have inactive users, to obtain the optimal throughput, while the active users are still fairly selfish, the inactive users should be cooperative. This thus smoothes the area throughput of the throughput-based design; distinguishes the selfish design from the proposed designs; and renders the throughput-based design well-performing in terms of EE and hit-rate. We can actually observe that the performance differences between the throughput- and EE-based designs is smaller. However, the tradeoff between throughput and EE is still significant as we change the cluster size. Finally, we see that the proposed designs outperform the design with pure homogeneous modeling, again validating our points that users having preference diversity is beneficial.

D. Evaluations with Different Schedulers

Finally, we evaluate the proposed designs in the clustering networks with two different schedulers to show how the proposed designs can help designs for a network that has a very complicated scheduler. Specifically, in addition to evaluating using the random-push scheduler, we
evaluate (under the same caching policy) the “priority-push scheduler” [18], which functions as follows: every user first checks whether their requests can be satisfied by files in their local caches. If yes, the requests are satisfied; otherwise, they send requests to the BS. The BS then checks whether there exists users that can be satisfied by using D2D links. If yes, the BS randomly selects one to be served by the D2D link; otherwise, the BS randomly selects one user from those sending the requests to serve via a BS link. Obviously, such scheduler maximizes the usage of the D2D communications, and thus is expected to have better network performance in terms of throughput and EE as compared to the random-push scheduler. On the other hand, it might be unfair to those whose preferences are not similar to the mainstream - they might be less likely to be selected to serve. More importantly, such complicated scheduler results in an intractable expression for designing caching policies. We demonstrate how to exploit the proposed designs in this work along with some numerical results to provide guidance for obtaining the effective designs for it.

In Fig. 7 we evaluate the proposed coordinated designs in both networks with random-
push and priority-push schedulers, labeled as “Random” (dash line) and “Priority” (solid line), respectively. We observe that the priority-push network generally outperforms the random-push network in terms of the area throughput and EE. Besides, we observe that, in terms of the area throughput, the proposed designs can be fairly representative as the best cluster size of both random-push and priority-push networks and the orders of the throughput results of different designs are the same. The results for EE show some more subtle effects. We can see that the optimal cluster size for the priority-push network is smaller. Besides, to obtain the best EE in the priority-push network, it is unnecessary to have high hit-rate. The reason is that the priority-push scheduler would schedule a D2D link as long as there exists one, implying higher rate for scheduling D2D links than simply the hit-rate - the probability for at least one user to find the desired file in the D2D network is higher than a particular user to find his/her desired file. Thus, to obtain an effective design in the priority-push network in terms of EE, we might choose a design with the smaller cluster size and lower hit-rate as compared to the optimal EE design in the random-push network. Overall, we observe that to obtain an effective design in the priority-push network, on the basis of the results of the random-push networks, we should reduce the cluster size and consider various throughput–hit-rate tradeoff designs. Since our proposed tradeoff designs can efficiently evaluate the throughput and hit-rate, such try-and-error procedure might not be challenging.

VI. CONCLUSIONS

In this work, we improved cache-aided D2D networks by considering the individual preferences of users. Using an individual preference probability model, we derived the network utility of a clustering network that fairly serves users in a cluster, and proposed a utility optimization problem. Such problem can be specialized to different important practical problems, such as throughput, EE, hit-rate optimization, and different tradeoff problems. Two solution approaches that solve the utility optimization problem coordinatedly and non-coordinatedly were proposed. Comprehensive numerical evaluations were conducted with the practical individual preference and network setups. The results show that when appropriately exploiting the information of individual preferences, the cache-aided D2D network can be significantly improved thanks to the preference diversity of users. The results also show that the throughput and hit-rate significantly conflict with each other, and such conflict can be resolved through a suitable tradeoff design. To obtain an effective EE design, in addition to directly optimizing EE, we can solve a properly
designed throughput–hit-rate tradeoff design, offering another perspective for EE optimization. Besides optimizing the network by means of optimizing caching policy, the network performance can be optimized by changing the cooperation range of the D2D; a tradeoff again exists in this regard. Finally, we demonstrate how to use results in our work to serve as a foundation for designing effective caching policies in networks with more involved scheduling policies.

APPENDIX A
DERIVATIONS OF THE EXPECTED UTILITY

We first derive the expression of $U$. Using (1), (2), and (3), we obtain

$$U = \sum_{k \in \mathcal{U}} \frac{w_k}{K_A} \left\{ U_D \left( 1 - \sum_{m=1}^{M} a_m^k \left[ \prod_{l \in \mathcal{U}} (1 - b_m^l \mathbb{1}_{\{h_m,c\}}) \right] - \sum_{m=1}^{M} \sum_{n=1}^{K} \alpha_m^k \right) \right\} + \sum_{m=1}^{M} a_m^k + \sum_{m=1}^{M} \sum_{n=1}^{K} \alpha_m^k \left( U_B - U_D \right) K_A.$$

By using (35), we thus obtain

$$U_{\text{net}} = \sum_{k \in \mathcal{U}} \frac{w_k U_D}{K_A} + (U_B - U_D) \sum_{m=1}^{M} \frac{w_k}{K_A} \sum_{n=1}^{K} \alpha_m^k \left( U_B - U_D \right) K_A.$$

$$= \sum_{k \in \mathcal{U}} \frac{w_k U_D}{K_A} + (U_B - U_D) \sum_{m=1}^{M} \frac{w_k}{K_A} \sum_{n=1}^{K} \alpha_m^k \left( U_B - U_D \right) K_A.$$

$$= \sum_{k \in \mathcal{U}} \frac{w_k U_D}{K_A} + (U_B - U_D) \sum_{m=1}^{M} \frac{w_k}{K_A} \sum_{n=1}^{K} \alpha_m^k \left( U_B - U_D \right) K_A.$$
APPENDIX B

PROOF OF THEOREM 1

To prove the Theorem, we first note that the problem in (7) satisfies the block separable structure as follows:

\[
\max_{b_1,\ldots,b_K} U(b_1, b_2, \ldots, b_K) \\
\text{s.t.} \quad b_k \in B_k, \forall k.
\]

Eq. (36) indicates that the constrains on different blocks are separable. Denote \( u(b_k'; B^r) = U(b^r_1, \ldots, b^r_{k'-1}, b^r_k, b^r_{k'+1}, \ldots, b^r_K) \) for brevity. From Alg. 1, we notice that since constraints are block separable and \( b^r_{k'+1} = \arg\max_{b_k' \in B_k} u(b_k'; B^r) \) at each iteration, we have

\[
u(b^r_{k'+1}; B^r) \geq u(b^r_k; B^r).
\]

Thus, we know the algorithm is monotonically non-decreasing. Then since the optimal objective function of (7) should not be infinity, the algorithm must converge.

To prove that Alg. 1 converges to a stationary point if every iteration has an unique maximization, we note that the proposed algorithm can be analyzed by the framework of block coordinate descent methods in [41]. Then by directly applying the Proposition 2.7.1 in [41], the proof is complete.

REFERENCES

[1] M.-C. Lee and A. F. Molisch, "Individual preference aware caching policy design for energy-efficient wireless D2D communications," in Proc. IEEE GLOBECOM, Dec. 2017.
[2] "Cisco Virtual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021," San Jose, CA, USA.
[3] J. G. Andrews, S. Buzzi, W. Choi, and et. al., "What will 5G be?,” IEEE J. Sel. Areas Commun., vol. 32, no. 6, pp. 1065-1082, Jun. 2014.
[4] A. F. Molisch, G. Caire, D. Ott, J. R. Foerster, D. Bethanabhotla, and M. Ji, Caching eliminates the wireless bottleneck in video aware wireless networks, Adv. Elect. Eng., vol. 2014, Nov. 2014, Art. ID 261390.
[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] 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.
[7] N. Golrezaei, P. Mansourifard, A. F. Molisch, and A. G. Dimakis, "Base-Station assisted device-to-device communications for high-throughput wireless video networks," IEEE Trans. Wireless Commun., vol. 13, no. 7, pp. 3665-3676, Jul. 2014.
[8] M. Ji, G. Caire, and A. F. Molisch "Wireless device-to-device caching networks: Basic principles and system performance,” IEEE J. Sel. Area Commun., vol. 34, no. 1, pp. 176-189, Jan. 2016.
[9] M. Ji, G. Caire, and A. F. Molisch, "The throughput-outage tradeoff of wireless one-hop caching networks," IEEE Trans. Inf. Theory, vol. 61, no. 12, pp. 6833-6859, Dec. 2015.
[10] M. Ji, G. Caire, and A. F. Molisch, “Fundamental limits of caching in wireless D2D networks,” *IEEE Trans. Inf. Theory*, vol. 62, no. 2, pp. 849-864, Feb. 2016.

[11] M.-C. Lee, M. Ji, A. F. Molisch, and N. Sastry, “Throughput-Outage analysis and evaluation of cache-aided D2D networks with measured popularity distributions,” *arXiv:1902.04563*, Feb. 2019.

[12] K. Shanmugam, N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, “FemtoCaching: Wireless content delivery through distributed caching helpers,” *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402-8413, Dec. 2013.

[13] B. Blaszczyszyn and A. Giovanidis, “Optimal geographic caching in cellular networks,” *IEEE ICC*, Jun. 2015.

[14] Z. Chen, J. Lee, T. Q. S. Quek, and M. Kountouris, “Cooperative caching and transmission design in cluster-centric small cell networks,” *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3401-3415, May 2017.

[15] U. Niessen and M. A. Maddah-Ali, “Coded caching with nonuniform demands” *IEEE Trans. Inf. Theory*, vol. 63, no. 2, pp. 1146-1158, Feb. 2017.

[16] Z. Chen, N. Pappas, and M. Kountouris, “Probabilistic caching in wireless D2D networks: cache hit optimal vs. throughput optimal,” *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 584-587, Mar. 2017.

[17] B. Chen, C. Yang, G. Wang, "High-Throughput opportunistic cooperative device-to-device communications with caching,” *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 7527-7539, Aug. 2017.

[18] M.-C. Lee and A. F. Molisch, "Caching policy and cooperation distance design for base station assisted wireless D2D caching networks: Throughput and energy efficiency optimization and trade-off," *IEEE Trans. Wireless Commun.*, vol. 17, no. 11, pp. 7500-7514, Nov. 2018.

[19] B. Chen, C. Yang, and A. F. Molisch, "Cache-enabled device-to-device communications: Offloading gain and energy cost," *IEEE Trans. Wireless Commun.*, vol. 17, no. 7, pp. 4519-4536, Jul. 2017.

[20] D. Malak, M. Al-Shalash, and J. G. Andrews, "Optimizing content caching to maximize the density of successful receptions in device-to-device networking, " *IEEE Commun.*, vol. 65, no. 10, pp. 4365-4380, Oct. 2016.

[21] M. Naslcheraghi, M. Afshang, H. S. Dhillon, "Modeling and performance analysis of full-duplex communications in cache-enabled D2D networks,” in *Proc. IEEE ICC*, May 2018.

[22] J. Guo, J. Yuan, and J. Zhang, “An achievable throughput scaling law of Wireless device-to-device caching networks with distributed MIMO and hierarchical Cooperations,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 492-505, Jan. 2018.

[23] A. Liu, V. K. N. Lau, and G. Caire, “Cache-induced hierarchical cooperation in wireless device-to-device caching networks,” *IEEE Trans. Inf. Theory*, vol. 62, no. 6, pp. 4629-4652, Jun. 2018.

[24] R. Wang, J. Zhang, S. H. Song, and K. B. Letaief, “Mobility-Aware Caching in D2D Networks,” *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 5001-5015, Aug. 2017.

[25] M.-C. Lee, H. Feng, and A. F. Molisch, “Design of dynamic caching content replacement in base station assisted wireless D2D caching networks,” *IEEE ICC, in Press*, May 2019.

[26] D. Karamshuk, N. Sastry, M. Al-Bassam, A. Secker, and J. Chandaria, "Take-Away TV: Recharging wok commutes with predictive preloading of catch-up TV content,” *IEEE J. Sel Commun.*, vol. 34, no. 8, pp. 2091-2101, Aug. 2016.

[27] M.-C. Lee, A. F. Molisch, N. Sastry, and A. Raman, "Individual preference probability modeling for video content in wireless caching networks,” in *Proc. IEEE GLOBECOM*, Dec. 2017.

[28] M.-C. Lee, A. F. Molisch, N. Sastry, and A. Raman, "Individual preference probability modeling and parameterization for video content in wireless caching networks," *IEEE/ACM Trans. Netw., In press*, 2019.

[29] D. Liu and C. Yang, “Optimizing caching policy at base stations by exploiting user preference and spatial locality,” *IEEE Trans. Commun.*, 2018.

[30] Y. Guo, L. Duan, and R. Zhang, “Cooperative local caching under heterogeneous file preferences,” *IEEE Trans. Commun.*, vol. 65, no. 1, pp. 444-457, Jan. 2017.
[31] Y. Pan, C. Pan, H. Zhu, and et. al., "On consideration of content preference and sharing willingness in D2D assisted offloading," arXiv preprint, arXiv:1702.00209, Feb. 2017.

[32] B. Chen and C. Yang, “Caching policy optimization for D2D communications by learning user preference,” in Proc. IEEE VTC-Spring, Jun. 2017.

[33] T. Zhang, H. Fan, J. Loo, and D. Liu, “User preference aware caching deployment for device-to-device caching networks,” IEEE System J., Dec. 2017.

[34] Y. Li, C. Zhong, M. C. Gursory, and S. Velipasalar, “Learning-Based delay-aware caching in Wireless D2D caching networks,” IEEE Access, vol. 6, pp. 77250-77264, Nov. 2018.

[35] L. Zhang, M. Xiao, G. Wu, and S. Li, "Efficient scheduling and power allocation for D2D-assisted wireless caching networks," IEEE Trans. Commun., vol. 64, no. 6, pp. 2438-2452, June 2016.

[36] M. Ehrgott, Multicriteria Optimization. Springer-Verlag New York, 2005.

[37] S. Boyd and L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[38] A. F. Molisch, Wireless Communications, IEEE Press-Wiley, 2nd ed., 2012.

[39] A. M. Mathai, An introduction to geometrical probability: Distributional aspects with applications. Boca Raton, FL, USA: CRC, 1999.

[40] C.-C. Tseng, H.-T. Chen, and K.-C. Chen, “On The Distance Distributions of The Wireless Ad Hoc Networks,” in Proc. IEEE VTC-Spring, 2006.

[41] D.P. Bertsekas, “Nonlinear Programming,” Belmont: Athena scientific, 1999.