User Scheduling and Power Allocation for Content Delivery in Caching Helper Networks

Minseok Choi[†*], Andreas F. Molisch[†], and Joongheon Kim[*]
[†]Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA
[*]School of Electrical Engineering, Korea University, Seoul, Korea
E-mails: {choimins, molisch}@usc.edu, joongheon@korea.ac.kr

***Abstract*—This paper proposes a user scheduling and power allocation method for content delivery in wireless caching helper networks without any stringent constraint on the interference model. For supporting delay-sensitive and time-varying user demands, the actual delivery quantity of the requested content should be dynamically controlled by advanced scheduling and power allocation. In addition, it is difficult for a central unit to control the content delivery due to a lack of knowledge of the entire time-varying network; therefore, a belief-propagation (BP)-based algorithm that facilitates distributed decisions on user scheduling and power allocation at every caching helper is presented. The proposed delivery scheme maximizes power efficiency while limiting the average delay of user request satisfactions by managing interference among users well. Simulation results show that the proposed scheme provides almost the same delay performance as the exhaustively found optimal one at the expense of little power consumption.**

I. Introduction

Increasingly exploding data traffic is caused by a small number of popular contents being requested at multiple times and at ultra high rates [1]. Wireless caching has been considered as a promising technique for supporting the overlapped user demands by storing popular contents on independent entities, such as on femto-base stations (BSs) [2], or on user devices [3] during off-peak hours. Since these entities are close to end users, transmission delay and redundant backhaul costs stemming from overlapped requests can be reduced. However, storage sizes of helpers are limited, caching and delivery of popular contents are critical issues in this field. The content caching distributions have been shown to be relatively robust, i.e., deviations of the actual caching distribution from the optimum one results in small performance losses [3]. We thus assume in this paper that the caching distribution is given.

When caching is already conducted, content delivery in wireless caching networks is fundamentally different from conventional donwlink communications where a transmitter is already determined for a given receiver [4]. On the other hand, in caching networks, users are sufficient to receive contents from any helper and to be scheduled sporadically if their requests are being provided well within the delay threshold. Thus, helper association for the content-requesting user is important in the delivery phase, which is selection of the best source node. The traditional method for helper association is to choose the one whose channel condition is the strongest [5]. Meanwhile, when the identical contents but different qualities and sizes are stored on different helpers, the authors of [6], [7] proposed a helper association scheme to maximize the average quality. However, interference management for multiple active users is not considered in [5]–[7].

The content delivery policy for supporting multiple active users at the same time in BS- or helper-assisted caching networks has been researched; however, hard constraints are still applied to interference models. Ref. [8] exploits orthogonal resources for multiple active users in the same picocell. Downlink scheduling in HetNets was studied in [9], with the assumption that interfering BSs transmit peak power. Meanwhile, link scheduling schemes in BS-assisted and device-to-device (D2D)-assisted caching networks were proposed in [10] and [11] respectively by managing interference without any strict constraint on interference models; however, adaptive power allocations are not considered. For device caching, D2D link scheduling with the power control in the presence of interference was investigated in [12]. The scheme proposed in [12] is a centralized decision process, and it requires all information of the entire network and channel gains.

Thus, this paper jointly optimizes user scheduling and power allocation depending on stochastic network states in a distributed manner. The main contributions are as follows:

- This paper proposes user scheduling and power allocation policy for content delivery in wireless helper networks, without any stringent assumption restricting the interference model. Different from most of the existing works, it does not require any clustering with different bandwidth allocations in order to avoid interference.
- The proposed BP-based algorithm facilitates distributed decisions on user scheduling and power allocation at every helper without full knowledge of all channel information. Therefore, the proposed delivery scheme is much applicable to the practical scenario where gathering the exact information of the time-varying network is difficult.
- An adaptive control of user scheduling and power allocation is designed based on the Lyapunov optimization theory for satisfying time-varying and delay-sensitive user demands.
- We perform simulations to verify the proposed link scheduling and power allocation policy. It is shown that the proposed scheme provides very similar limits on the averaged queueing delay as that obtained with the optimal centralized decision mechanism, at the expense of power consumption, which is increased by 70%.

The rest of the paper is organized as follows. The wireless

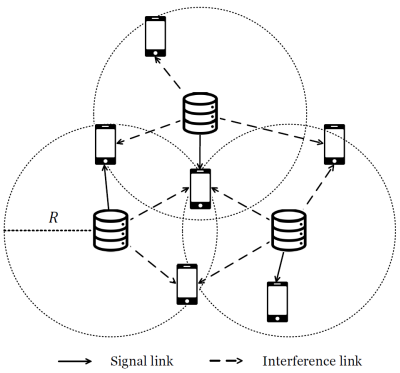

Fig. 1: Caching helper network model

caching network model and user queue model are described in Section II. The joint optimization problem of user scheduling and power allocation is formulated in Section III. The BP-based content delivery policy is proposed in Sections IV. Simulation results are presented in Section V, and Section VI concludes this paper.

II. System Model

A. Wireless Caching Network

A wireless caching helper network is considered, where there are K helpers and N users, as shown in Fig. 1. Denote each helper and user by $h_k \in \mathcal{H}$ and $x_n \in \mathcal{X}$, respectively, where $k \in \{1, \cdots, K\}$, $n \in \{1, \cdots, N\}$, $\mathcal{H}$ and $\mathcal{X}$ represent the helper set and the user set, respectively. Each helper has its own coverage, and users are distributed based on a homogeneous Poisson point process (PPP) with intensity λ. Suppose that popular contents are already pushed by the central BS into helpers having the finite storage size. Each user requests one of the contents in a library $\mathcal{F}$ according to the popularity distribution, e.g., a Zipf distribution, and helpers can deliver cached contents to users if following two conditions are satisfied: 1) a content-requesting user is in the coverage region of given helper, and 2) the content requested by the user is already cached in given helper. The scheduling indicator between h_k and x_n is denoted by $z_{kn} \in \{0, 1\}$, and $z_{kn} = 1$ if h_k schedules x_n for content delivery. We assume $K < N$, and users whose desired contents are cached in their associated helpers are considered only; therefore, every helper is determined to deliver the content to one of the active users in its coverage at every time. Denote the user set whose members can receive the content from h_k by $\mathcal{V}_k$, and the helper set whose members can deliver the content to x_n by $\mathcal{J}_n$. In addition, even though the delivery link can be constructed within the distance R, interference power among different BSs' coverage regions could not be ignored. Therefore, we define the interference distance d_i so that when the distance between h_k and x_n is smaller than d_i, h_k can interfere with x_n. Then, denote the user set whose members are interfered from h_k by $\mathcal{X}_k$, and the helper set whose members interfere with x_n by $\mathcal{H}_n$. In this paper, a pair of h_k and x_n is called as "neighboring" when h_k can generate the signal link or the interference link with x_n.

As shown in Fig. 1, coverage regions of helpers are partially overlapped; therefore, users should choose one of the helpers that (i) store the requested content and (ii) whose channel conditions are sufficiently strong for successful content delivery. Therefore, helper association becomes a user scheduling problem. The key difference to the standard, well-explored, link scheduling problem is that each receiver has *multiple* possible transmitters from which it can obtain the content. Here, this paper does not allow broadcasting the content to multiple users and cooperation among helpers, because user demands are generated asynchronously in general. Thus, one-to-one link scheduling is considered only. We consider discretized time slots, i.e., $t \in \{1, 2 \cdots, \}$, and user scheduling and power allocation are updated for every slot. Suppose that every helper has the same power budget of $P_{\max}$, and denote transmit power of h_k by q_k, satisfying $0 \leq q_k \leq P_{\max}$.

The Rayleigh fading channel model is assumed, and the channel gain between h_k and x_n is described by the (frequency-flat) transfer function whose amplitude gain is described by $g_{kn}(t) = \sqrt{D_{kn}}u(t)$, where $D_{kn} = 1/d_{kn}^{\alpha}$ denotes path gain (the inverse of the path loss). In addition, d_{kn} and α are the distance between h_k and x_n and the path loss exponent, respectively. $u(t)$ is the fast fading component at slot t having a complex Gaussian distribution, $u(t) \sim \mathcal{CN}(0, 1)$. Then, the link rate between h_k and x_n can be written by

$$R_{kn}(t) = \mathcal{B}\log_2\left(1 + \frac{\sum_{h_k \in \mathcal{J}_n} |g_{kn}|^2 \cdot z_{kn} q_k}{\sum_{h_i \in \mathcal{H}_n} |g_{in}|^2 \sum_{\substack{m \in \mathcal{V}_i \\ m \neq n}} z_{ik} q_i + \sigma^2}\right), \tag{1}$$

where $\mathcal{B}$ is the bandwidth, and σ^2 is the noise variance. Assume that every scheduled user accesses the same bandwidth. The data rate of x_n is obtained as $R_n(\mathbf{z}_{\mathcal{H}_n}, \mathbf{q}_{\mathcal{H}_n}, t) = \sum_{h_k \in \mathcal{H}_n} R_{kn}(t)$, where $\mathbf{z}_{\mathcal{H}_n} = [\mathbf{z}_k : h_k \in \mathcal{H}_n]$ and $\mathbf{q}_{\mathcal{H}_n} = [q_k : h_k \in \mathcal{H}_n]$ are the scheduling indicator and the power allocation vectors of helpers neighboring to x_n. Here, $\mathbf{z}_k = [z_{ki} : x_i \in \mathcal{X}_k]$. Since this paper considers one-to-one scheduling only, $\sum_{h_k \in \mathcal{J}_n} z_{kn} \leq 1$ and $\sum_{x_n \in \mathcal{V}_k} z_{kn} = 1$ should be satisfied for all $x_n \in \mathcal{X}$ and $h_k \in \mathcal{H}$. Note that users may not be scheduled, i.e., $\sum_{h_k \in \mathcal{J}_n} z_{kn} = 0$ is possible.

B. User queue model

Suppose that each content is divided into many chunks, and x_n consecutively receives desired chunks from one helper in $\mathcal{J}_n$ in each slot. Then, user demands, i.e., the number of chunks requested but not delivered yet, are accumulated in the user queue. The queue dynamics of x_n in each slot $t \in \{0, 1, \cdots\}$ can be represented as $Q_n(t+1) = Q_n(t) - \mu_n(t) + a_n(t)$, where $Q_n(t)$, $a_n(t)$, and $\mu_n(t)$ stand for the queue backlog, numbers of newly requested and delivered chunks, respectively. Here, $\mu_n(t) = \min\{\tilde{\mu}_n(t), Q_n(t)\}$ which is the maximal number of chunks that can be delivered at slot t. Assume that $a_n(t)$ is an i.i.d. uniform random variable, i.e., $a_n \sim \mathcal{U}(0, a_{\max})$. The interval of each slot is denoted by τ_c; we further assume block fading channel, and channel gains are static for each slot. Therefore, the departure $\mu_n(t)$ is given by

$$\mu_n(t) = \min\left\{\left\lfloor \frac{\tau_c R_n(\mathbf{z}_{\mathcal{H}_n}, \mathbf{q}_{\mathcal{H}_n}, t)}{S} \right\rfloor, Q_n(t)\right\}, \tag{2}$$

where S is the size of a chunk. Since partial chunk transmission is meaningless, a flooring operation is used in (2).

III. Problem formulation for user scheduling and power allocation in caching helper networks

The optimization problem that maximizes the long-term time-averaged power efficiency, while limiting the time-averaged service delay, can be formulated as follows:

$$\{\mathbf{z},\mathbf{q}\} = \underset{z_{kn},q_k,\forall h_k,\forall x_n}{\arg\min} \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\Big[\sum_{h_k\in\mathcal{H}} q_k \sum_{x_n\in\mathcal{X}_k} z_{kn}\Big] \tag{3}$$

$$\text{s.t.} \quad \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\Big[\sum_{x_n\in\mathcal{X}} Q_n(t)\Big] < \infty \tag{4}$$

$$\sum_{h_k\in\mathcal{J}_n} z_{kn} \le 1,\ \forall x_n \in \mathcal{X} \tag{5}$$

$$\sum_{x_n\in\mathcal{V}_k} z_{kn} = 1,\ \forall h_k \in \mathcal{H}, \tag{6}$$

$$0 \le q_k \le P_{\max},\ \forall h_k \in \mathcal{H} \tag{7}$$

$$z_{kn} \in \{0,1\},\ \forall h_k \in \mathcal{H},\ \forall x_n \in \mathcal{X} \tag{8}$$

where $\mathbf{z} = [z_{kn} : h_k \in \mathcal{H}, x_n \in \mathcal{X}_m]$, and $\mathbf{q} = [q_k : h_k \in \mathcal{H}]$. Specifically, expectations of both (3) and (4) are with respect to random channel realizations. The constraint (4) pursues strong stability of the user queueing system, and the one-to-one link scheduling is guaranteed by (5) and (6).

According to Little's theorem [13], the averaged queueing delay is proportional to the average queue length. Based on the Lyapunov optimization theory, the time-averaged queue length can be limited; finally, the solution of the problem (3)–(6) averts excessive accumulation of user demands by achieving queue stability in (4). In this respect, many delay-constrained transmission policies which limit the queueing delay by pursuing the queue stability have been proposed in [6], [7]. In this paper, simulation results in Section V show that the queueing delay can be reduced by ensuring (4).

Let $\mathbf{Q}(t) = [Q_n(t) : x_n \in \mathcal{X}]$ be the queue backlog vector at slot t, and define the quadratic Lyapunov function as $L(\mathbf{Q}(t)) = \frac{1}{2}\sum_{x_n\in\mathcal{X}}(Q_n(t))^2$. Then, let $\Delta(\mathbf{Q}(t)) = \mathbb{E}[L(\mathbf{Q}(t+1)) - L(\mathbf{Q}(t))|\mathbf{Q}(t)]$ be a conditional quadratic Lyapunov function which is the drift on t, and it can be upper bounded by

$$\Delta(\mathbf{Q}(t)) \le C - \mathbb{E}\Big[\sum_{x_n\in\mathcal{X}} Q_n(t)(\mu_n(t) - a_n(t))\Big|\mathbf{Q}(t)\Big], \tag{9}$$

where

$$\frac{1}{2}\mathbb{E}\Big[\sum_{x_n\in\mathcal{X}}\Big(\mu_n(t)^2 + a_n(t)^2\Big)\Big] \le C, \tag{10}$$

which assumes that arrival and departure process rates are upper bounded. In order to achieve queue stability in (4), the dynamic policy is designed by minimizing a upper bound on *drift-plus-penalty* [14]:

$$\Delta(\mathbf{Q}(t)) + V\mathbb{E}\Big[\sum_{h_k\in\mathcal{H}} q_k \sum_{x_n\in\mathcal{X}_k} z_{kn}\Big], \tag{11}$$

where V is an importance weight for power efficiency. Then, according to (9), *min-drift-plus-penalty* algorithm minimizes a upper bound on drift, i.e.,

$$V\mathbb{E}\Big[\sum_{h_k\in\mathcal{H}} q_k \sum_{x_n\in\mathcal{X}_k} z_{kn}\Big] - \mathbb{E}\Big[\sum_{x_n\in\mathcal{X}} Q_n(t)\mu_n(t)\Big|\mathbf{Q}(t)\Big], \tag{12}$$

because C is a constant and $a_n(t)$ for all $u_n \in \mathcal{U}$ is not controllable. Here, the concept of opportunistically minimizing the expectations is used; therefore, (12) is minimized by the algorithm that observes the current queue state $\mathbf{Q}(t)$, and determines $\mathbf{z}$ and $\mathbf{q}$ to minimize

$$V\sum_{h_k\mathcal{H}} q_k \sum_{x_n\in\mathcal{X}_k} z_{kn} - \sum_{x_n\in\mathcal{X}} Q_n(t)\mu_n(t). \tag{13}$$

According to (13), the utility function representing the negative sign of upper bound on drift-plus-penalty can be written by

$$F(\mathbf{z},\mathbf{q},t)$$

$$= \sum_{x_n\in\mathcal{X}} Q_n(t)\mu_n(\mathbf{z}_{\mathcal{H}_n},\mathbf{q}_{\mathcal{H}_n},t) - V\cdot\sum_{h_k\in\mathcal{H}} q_k \sum_{x_n\in\mathcal{X}_k} z_{kn} \tag{14}$$

$$= \sum_{x_n\in\mathcal{X}}\Big(Q_n(t)\mu_n(\mathbf{z}_{\mathcal{H}_n},\mathbf{q}_{\mathcal{H}_n},t) - V\cdot\sum_{h_k\in\mathcal{H}_n} q_k z_{kn}\Big) \tag{15}$$

$$= \sum_{x_n\in\mathcal{X}} \tilde{f}_n(\mathbf{z}_{\mathcal{H}_n},\mathbf{q}_{\mathcal{H}_n},t). \tag{16}$$

Then, according to Lyapunov optimization theory [14], the problem of (3)–(6) can be converted into the *min-drift-plus-penalty* problem as follows:

$$\{\mathbf{z}^\star,\mathbf{q}^\star\} = \underset{\mathbf{z},\mathbf{q}}{\arg\max} \sum_{x_n\in\mathcal{X}} \tilde{f}_n(\mathbf{z}_{\mathcal{H}_n},\mathbf{q}_{\mathcal{H}_n},t) \tag{17}$$

$$\text{s.t.} \quad (5)-(8). \tag{18}$$

$\mathbf{z}^\star$ and $\mathbf{q}^\star$ are the optimal user scheduling and power allocation vectors. The objective function of (17) is not separable because of interference effects. Therefore, distributed decisions on $\mathbf{z}^\star$ and $\mathbf{q}^\star$ are difficult to make in this formulation. For simplicity, notations for the dependency of all parameters on t are omitted in the remaining sections because user scheduling and power allocation are determined in every different slot.

Solutions of the problem of (17)–(18) can be obtained by relaxing the constraints of (5) and (6). First, suppose that transmit power should be one of L discrete levels, i.e., $q_k \in \{P_1,\cdots,P_L\}$, and $P_l > 0$ for all $l \in \{1,\cdots,L\}$. This is reasonable because practical power control uses discrete levels. Second, in order to suggest a distributed delivery scheme, we consider the probabilistic policy for each h_k to determine $\mathbf{z}_k$ and q_k. In other words, given the probability distribution of $\mathbf{z}_k$ and q_k, the most probable decisions on user scheduling power allocation are made by

$$\{\mathbf{z}_k^\star, q_k^\star\} = \underset{x_n\in\mathcal{X}_k, l\in\{1,\cdots,L\}}{\arg\max} \Pr\{\mathcal{E}_{knl}\}, \tag{19}$$

where $\mathcal{E}_{knl}$ represents the event that h_k schedules x_n with

power level P_l, as defined by

$$\mathcal{E}_{knl} \triangleq \{z_{kn} = 1, \sum_{\substack{x_i \in \mathcal{V}_k \\ i \neq n}} z_{ki} = 0, \ q_m = P_l\}. \quad (20)$$

Note that only one user x_n can be scheduled by h_k, i.e., $z_{kn} = 1$, among users in $\mathcal{V}_k$; therefore, $\cup_{x_n \in \mathcal{V}_k} \cup_{l=1}^{L} \mathcal{E}_{knl}$ is the entire region of possible decision parameter sets at h_k, and there are $L \cdot |\mathcal{V}_m|$ possible decisions. Thus, the goal of the content delivery scheme in this paper becomes to find probability distributions of user scheduling and power allocation at all caching helpers, i.e., $p(\mathbf{z}_k, q_k)$ for all $h_k \in \mathcal{H}$.

With the above decision process, every helper can schedule only one user; therefore, the constraint (6) can be removed. In addition, the constraint (5) can be combined with the objective function (17) by using the indicator function as given by $f_n(\mathbf{z}_{\mathcal{H}_n}, \mathbf{q}_{\mathcal{H}_n}) = \tilde{f}_n(\mathbf{z}_{\mathcal{H}_n}, \mathbf{q}_{\mathcal{H}_n}) \cdot \mathcal{I}\Big(\sum_{h_k \in \mathcal{H}_n} z_{kn} \leq 1 \Big)$. Finally, the problem of (17)–(18) can be re-written by

$$p(\mathbf{z}, \mathbf{q}) = \arg\max_{p(\mathbf{z}_k, q_k)} \sum_{x_n \in \mathcal{X}} f_n(\mathbf{z}_{\mathcal{H}_n}, \mathbf{q}_{\mathcal{H}_n}) \quad (21)$$

$$\text{s.t.} \ \ q_k \in \{P_1, \cdots, P_L\}, \ \forall h_k \in \mathcal{H} \quad (22)$$

$$z_{kn} \in \{0, 1\}, \ \forall h_k \in \mathcal{H}, \ \forall x_n \in \mathcal{X}_k \quad (23)$$

$$(19). \quad (24)$$

The goal of the above problem becomes to find the marginal probabilities $p(\mathbf{z}_k, q_k)$ for all helpers $h_k \in \mathcal{H}$, and this goal can be achieved by using the BP algorithm.

IV. The Belief Propagation Algorithm

This section explains how the optimization problem of (21)–(24) can be solved by using the BP algorithm. The probability distributions of all possible $\mathbf{z}_k$ and q_k for all $h_k \in \mathcal{H}$ and $x_n \in \mathcal{X}$ can be defined with a constant $\delta > 0$ as follows:

$$p(\mathbf{z}, \mathbf{q}) = \frac{1}{Z} \exp\Big(\delta F(\mathbf{z}, \mathbf{q})\Big) = \frac{1}{Z} \prod_{x_n \in \mathcal{X}} \exp\Big(\delta f_n(\mathbf{z}_{\mathcal{H}_n}, \mathbf{q}_{\mathcal{H}_n})\Big), \quad (25)$$

where Z is a normalization factor called the partition function of δ. The goal is to find $p(\mathbf{z}_k, q_k)$ for all $h_k \in \mathcal{H}$ to decide user scheduling and power allocation at every helper in a distributed manner. The marginal distribution of $p(\mathbf{z}, \mathbf{q})$ with respect to the decision variables $\mathbf{z}_k$ and q_k, i.e., $p(\mathbf{z}_k, q_k)$, can be estimated by the BP algorithm. According to a standard result of large deviations [15], the optimal decisions to maximize the utility function $F(\mathbf{z}, \mathbf{q})$ as $\delta \to \infty$, are as follows:

$$\lim_{\delta \to \infty} \{\hat{\mathbf{z}}, \hat{\mathbf{q}}\} = \arg\max_{\mathbf{z}, \mathbf{q}} F(\mathbf{z}, \mathbf{q}). \quad (26)$$

Therefore, we can estimate the marginal expectations of the probability distribution $p(\mathbf{z}, \mathbf{q})$ for large δ, and h_k can make decisions on user scheduling and power allocation based on the marginalized $p(\mathbf{z}_k, q_k)$.

A bipartite graph $G = (V, E)$ called the factor graph is constructed to represent the network topology, where the vertex set V consists of K helpers and N users as shown in Fig. 2. Caching helpers are variable nodes and users are factor

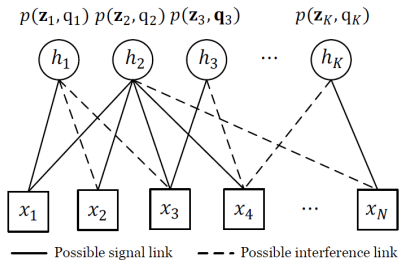

Fig. 2: Factor graph consisting of caching helpers and users

nodes in the factor graph. An edge $(h_k, x_n) \in E$ is constructed if $h_k \in \mathcal{H}_n$ and $x_n \in \mathcal{X}_k$. Therefore, an edge $(h_k, x_n) \in E$ implies that h_k and x_n are neighboring to each other, i.e., h_k can interfere with x_n. In the factor graph, only helpers that store at least one or more contents requested by neighboring users are considered, as well as only users who can find at least one or more neighboring helpers storing the requested content are considered.

In the BP algorithm, variable nodes and factor nodes iteratively exchange the belief messages along the edges of the factor graph. The belief messages of the variable node representing h_k transmitted to and received from neighboring factor nodes deliver estimates of $p(\mathbf{z}_k, q_k)$. Denote the belief message delivered from x_n to h_k at iteration i by $p^i_{n \to k}(\mathbf{z}_k, q_k)$ and and the reverse message that h_k passes to x_n is denoted by $p^i_{n \leftarrow k}(\mathbf{z}_k, q_k)$. After exchanging the messages between users and helpers for some fixed number of iterations, the final decision is made at each node h_k to compute the marginalized distribution $p(\mathbf{z}_k, q_k)$. The BP algorithm steps are as follows:

1) Initialization: Before beginning iterations, the initial values of $p^1_{n \leftarrow k}(\mathbf{z}_k, q_k)$ should be given for all $h_k \in \mathcal{H}$ and $x_n \in \mathcal{X}$. Suppose that every possible decision on user scheduling and power allocation at each helper follows a uniform distribution at $i = 1$; therefore, $p^1_{n \leftarrow k}(\mathcal{E}_{knl}) = \frac{1}{L \cdot |\mathcal{V}_k|}$, for all $h_k \in \mathcal{H}$, $x_n \in \mathcal{U}_k$, and $l \in \{1, \cdots, L\}$.
2) Factor node update: x_n (i.e., factor node n) updates the message $p^i_{n \to k}(\mathbf{z}_k, q_k)$ and send it to h_k (i.e., variable node k), where $h_k \in \mathcal{H}_n$. x_n computes $p^i_{n \to k}(\mathbf{z}_k, q_k)$ based on the messages received from helpers $h_m \in \mathcal{H}_n \setminus \{h_k\}$, i.e., $p^i_{n \leftarrow m}(\mathbf{z}_m, q_m)$, as given by

$$p^i_{n \to k}(\mathbf{z}_k, q_k) = \mathbb{E}\Big[\exp\Big(\delta f_n(\mathbf{z}_{\mathcal{H}_n}, \mathbf{q}_{\mathcal{H}_n})\Big) \Big| \mathbf{z}_k, q_k \Big]. \quad (27)$$

The expectation of (27) is with respect to $p^i_{n \leftarrow m}(\mathbf{z}_m, q_m)$ for all $h_m \in \mathcal{H}_n \setminus \{h_k\}$. When computing (27), there are three different cases to be considered: 1) h_k delivers the content to x_n 2) h_k interferes with x_n and x_n receives the content from another neighboring helper $h_j \neq h_k$, and 3) x_n cannot receive any content from neighboring helpers. Note that the third case includes both situations where $\sum_{h_s \in \mathcal{H}_n} z_{sn} = 0$ and $\mathcal{I}\Big(\sum_{h_k \in \mathcal{H}_n} z_{kn} \leq 1 \Big) = 0$, representing when x_n is not scheduled by any neighboring helper and when two or more helpers

schedule x_n at the same time, respectively.
The event $\mathcal{E}_{knl}$ represents the first case where h_k schedules x_n with the transmit power P_l, and $p^i_{n\to k}(\mathcal{E}_{knl})$ is updated by averaging data rates given by

$$\mathcal{B}\log_2\left(1+\frac{|g_{kn}|^2\cdot q_k}{\sum_{\substack{h_s\in\mathcal{H}_n\\ s\neq k}}|g_{sn}|^2\sum_{\substack{x_v\in\mathcal{V}_s\\ v\neq n}}z_{sv}q_s+\sigma^2}\right), \tag{28}$$

with respect to $\mathbf{x}_s$ and q_s for all $c_s\in\mathcal{H}_n\setminus\{c_m\}$.
Meanwhile, the event $\mathcal{E}_{kml}$ for all $x_m\in\mathcal{V}_k\setminus\{x_n\}$ represents the second case where h_k schedules x_m with transmit power P_l and interferes with x_n. In this case, the signal link between x_n and another neighboring helper h_j is generated, and $p^i_{n\to k}(\mathcal{E}_{kml})$ for all $x_m\in\mathcal{V}_k$ can be computed by averaging data rates given by

$$\mathcal{B}\log_2\left(1+\frac{\sum_{\substack{h_j\in\mathcal{J}_n\\ j\neq k}}|g_{jn}|^2q_j\cdot\mathcal{I}\Big(\sum_{\substack{h_j\in\mathcal{J}_n\\ j\neq k}}z_{jn}\leq 1\Big)}{|g_{kn}|^2q_k+\sum_{\substack{h_s\in\mathcal{H}_n\\ s\neq m,j}}|g_{sn}|^2\sum_{\substack{x_v\in\mathcal{V}_s\\ v\neq n}}z_{sv}q_s+\sigma^2}\right) \tag{29}$$

with respect to $\{\mathbf{z}_j,q_j\}$ and $\{\mathbf{z}_s,q_s\}$ for all $h_j\in\mathcal{J}_n$, $j\neq m$, and $h_s\in\mathcal{H}_n$, $s\neq m,j$.
Lastly, in the third case in which x_n cannot receive any content, zero throughput is achieved at x_n.

3) Variable node update: In every iteration, each helper sends updated messages to its neighboring users after receiving belief messages from neighboring users.
h_k updates the belief message $p^{i+1}_{n\leftarrow k}(\mathbf{z}_k,q_k)$ by using received messages $p^i_{j\to k}(\mathbf{z}_k,q_k)$ for all $x_j\in\mathcal{X}_k$, $j\neq n$, and sends it to factor node $x_n\in\mathcal{X}_k$ as follows:

$$p^{i+1}_{n\leftarrow k}(\mathbf{z}_k,q_k)=\frac{1}{Z}\prod_{\substack{x_j\in\mathcal{X}_k\\ j\neq n}}p^i_{j\to k}(\mathbf{z}_k,q_k). \tag{30}$$

The updates of belief messages at every factor node and variable node are iteratively performed.

4) Final solution: After the predetermined I iterations, the final decisions at every helper $h_k\in\mathcal{H}$ can be made based on received messages from neighboring users, as given by

$$p^I_k(\mathbf{z}_k,q_k)=\frac{1}{Z}\prod_{x_j\in\mathcal{X}_k}p^I_{j\to k}(\mathbf{z}_k,q_k). \tag{31}$$

Based on $p^I_k(\mathbf{z}_k,q_k)$ for all $h_k\in\mathcal{H}$, each helper can schedule one of neighboring users and determine an appropriate power level that gives the largest probability. Note that the complexity of the BP algorithm is dominated by the computations required for the expectation step in (27) of factor node updates and the iteration number I of the BP algorithm. For each factor node, the BP algorithm has to average the belief message in (27) for all different power allocations of interfering caching nodes h_k to x_n, i.e., $h_k\in\mathcal{H}_n$. Therefore, it requires $|\mathcal{H}_n|(L+1)^{|\mathcal{H}_n|}$ computations; therefore, total $I\cdot N\cdot|\mathcal{H}_n|(L+1)^{|\mathcal{H}_n|}$ computations are required If the slot duration is determined by the channel coherence time and longer than the operation time of the BP algorithm depending on its computational complexity, the proposed scheme can be performed in a real-time manner. Note that caching nodes and users exchange their belief messages via wireless communications, and its signaling overhead is assumed to be small compared to the effort in transmitting the payload.

TABLE I: System Parameters

Max. power budget ($P_{\max}$)	2 W
Max. value of random user demands ($a_{\max}$)	5
Bandwidth ($\mathcal{B}$)	10 MHz
Coherence time (τ_c)	10 ms
Path loss exponent (α)	3
Noise variance (σ^2)	10^{-8}
File size (S)	20 kbits

V. Numerical results

This section shows how proposed user scheduling and power allocation scheme achieves limited service delay as well as high power efficiency by numerical results. In Fig. 1, there are three caching helpers each having a coverage region with radius $R=100$, the locations of the helpers are $[(0,0),(\frac{5}{3}R,0),(\frac{5}{6}R,\frac{5\sqrt{3}}{6}R)]$, so that their coverage regions are partially overlapped. Users are randomly distributed according to a homogeneous PPP with the intensity of $\lambda=0.025\times10^{-2}$. In addition, $d_i=2R$; therefore, the caching helper can interfere with users outside of its coverage. Other simulation parameters are listed in Table I. Since we focus on the delivery phase, assume that content placements in helpers are already completed. In this paper, the probabilistic caching policy proposed in [5] is used.

The performance of the proposed user scheduling and power allocation policy is verified by comparing with the optimal scheme that is obtained by exhaustive search. This comparison technique exhaustively finds user scheduling and power allocation that minimizes the upper bound on the drift-plus-penalty term in (13). It can be considered as the numerically optimal scheme, and a centralized decision strategy.

The average queue length and the average power consumption versus time are shown in Figs. 3 and 4, respectively. We can see that the average queue length and power consumption of the proposed scheme and 'Exhaustive search' are clearly upper bounded, in accordance with Lyapunov theory [14]. In fact, tightness of these upper bounds is not explicitly ensured by the min-drift-plus-penalty algorithm; nevertheless, the average queue length of the proposed scheme is almost the same as that of 'Exhaustive search'. Therefore, when $V=1$, the proposed scheme can provide an average queueing delay that is almost the same as the optimal one, at the expense of 40% increased transmit power.

In Figs. 5 and 6, the impact of the system parameter V on power consumption and the time-average queue length respectively, is observed. As mentioned in (13), V is an importance factor for power-efficiency; therefore, as V increases, power consumption decreases and the average queue length grows. While the average queue lengths do not vary significantly, power consumption is saved especially for the proposed scheme. Consequently, the proposed scheme using

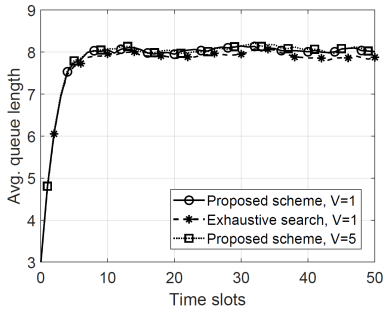

Fig. 3: Average and worst queue lengths

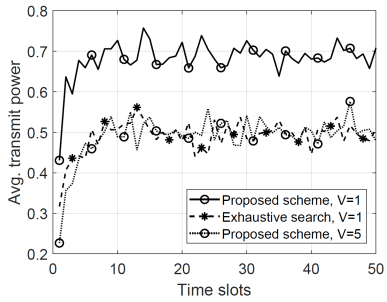

Fig. 4: Average power sum

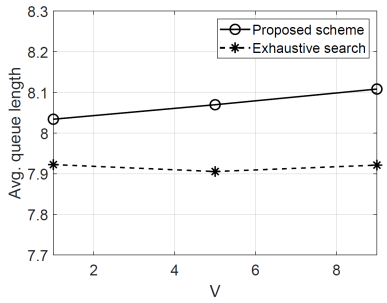

Fig. 5: The effect of V on avg. queue length

$V = 5$ can consume almost the same power as 'Exhaustive search' while limiting the queueing delay on a similar scale of 'Exhaustive search'. Note that V is a system parameter, whose appropriate value has to be found by experiments as well as which is free to be adjusted by the system designer.

VI. Concluding Remarks

This paper proposes dynamic and distributed content delivery in wireless caching helper networks. The joint optimization problem for user scheduling and power allocation is formulated. Then, the distributed decision process at each helper depending on the probability distributions of user scheduling and power allocation is presented. These probability distributions are obtained by constructing the factor graph representing the caching helper network and using the BP-based user scheduling and power allocation scheme. The numerical results show that the proposed scheme enables users to pursue high power efficiency while limiting the average queueing delay.

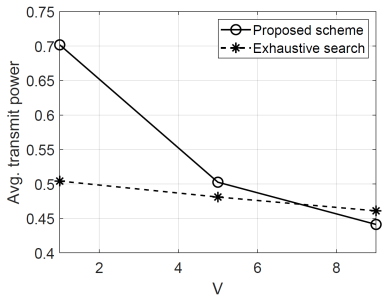

Fig. 6: The effect of V on avg. transmit power

Acknowledgment

This work was supported by NSF under projects NSF CCF-1423140 and NSF CNS-1816699, and Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2018-0-00170, Virtual Presence in Moving Objects through 5G). Joongheon Kim is a corresponding author.

References

[1] X. Cheng, J. Liu, and C. Dale, "Understanding the characteristics of Internet short video sharing: A YouTube-based measurement study," *IEEE Trans. Multimedia*, vol. 15, no. 5, pp. 1184–1194, Aug. 2013.

[2] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless video content delivery through distributed caching helpers," in *Proc. of IEEE INFOCOM*, Orlando, FL, USA, 2012.

[3] M. Ji, G. Caire and A. F. Molisch, "Wireless Device-to-Device Caching Networks: Basic Principles and System Performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176-189, Jan. 2016.

[4] S. Rangan and R. Madan, "Belief Propagation Methods for Intercell Interference Coordination in Femtocell Networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 631-640, April 2012.

[5] S. H. Chae and W. Choi, "Caching placement in stochastic wireless caching helper networks: Channel selection diversity via caching," *IEEE Trans. Wireless Commun.*, 15(10):6626–6637, Oct. 2016.

[6] M. Choi, J. Kim and J. Moon, "Wireless Video Caching and Dynamic Streaming Under Differentiated Quality Requirements," *IEEE J. Sel. Areas Commun.,* vol. 36, no. 6, pp. 1245-1257, June 2018.

[7] M. Choi, A. No, M. Ji and J. Kim, "Markov Decision Policies for Dynamic Video Delivery in Wireless Caching Networks", *IEEE Trans. Wireless Commun.*, Early Access, Sep. 2019.

[8] Z. Yang et al., "Cache Placement in Two-Tier HetNets With Limited Storage Capacity: Cache or Buffer?," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5415-5429, Nov. 2018.

[9] B. Lv et al., "Joint Downlink Scheduling for File Placement and Delivery in Cache-Assisted Wireless Networks With Finite File Lifetime," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4177-4192, June 2019.

[10] J. Kwak, L. B. Le, H. Kim and X. Wang, "Two Time-Scale Edge Caching and BS Association for Power-Delay Tradeoff in Multi-Cell Networks," *IEEE Trans. Commun.*, Early Access, 2019.

[11] J. Chuan, L. Wang and J. Wu, "Belief Propagation Based Distributed Content Delivery Scheme in Caching-Enabled D2D Networks", *IEEE Int. Conf. on Commun. (ICC)*, Shanghai, May 2019.

[12] L. Zhang, M. Xiao, G. Wu and S. Li, "Efficient Scheduling and Power Allocation for D2D-Assisted Wireless Caching Networks," *IEEE Trans. on Commun.*, vol. 64, no. 6, pp. 2438-2452, June 2016.

[13] D. Bertsekas and R. Gallager, *Data Networks*, 2nd Ed., Prentice, 1992.

[14] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*, Morgan & Claypool, 2010.

[15] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*. New York: Springer, 1998.

[16] Y. Weiss, "Correctness of Local Probability Propagation in Graphical Models with Loops," *Neural Computation*, vol. 12, no. 1, pp. 1-41, 1 Jan. 2000.