Joint Optimization of File Placement and Delivery in Cache-Assisted Wireless Networks

Bojie Lv*, Rui Wang*, Ying Cui‡ and Haisheng Tan‡
*Southern University of Science and Technology
‡Shanghai Jiao Tong University
‡University of Science and Technology of China

Abstract—In this paper, the downlink file transmission in one cell with the assistance of cache nodes is studied. Specifically, the base station (BS) reactively delivers files to cache nodes and a requesting user in a multicast manner. Therefore, one file transmission may lead to the cache status update, which further affects the future file transmissions. We consider the joint optimization of file placement and delivery. In particular, we first formulate the optimization of transmission power and time in one finite file lifetime as a Markov Decision Process (MDP) with a random number of stages, where the objective is to minimize the transmission resource at the BS. It is shown that the optimal solution can be obtained via a revised Bellman’s equation. Due to the curse of dimensionality, a novel approximation approach is proposed, where the value functions of the Bellman’s equation can be calculated from analytical expressions. Hence, iterative algorithms, which appear in the general approximate MDP solutions, can be avoided. Moreover, an bound on the approximation error is also provided.

I. INTRODUCTION

Recently, caching in cellular networks has attracted tremendous attention. By offloading traffics from BSs to cache nodes, the total energy consumption can be reduced or the overall spectral efficiency can be improved. For example in [1], the performance of wireless heterogeneous networks was evaluated, where the small BSs (cache nodes) store the most popular files. The authors in [2] showed that caching files randomly with optimized probabilities is better than storing the most popular files when each user can be served by multiple cache nodes. Moreover, there are also a number of works on coded caching schemes for exploring coded multicast opportunities [3],[4]. In all the above works, the cost of file placement is not taken into consideration, as it is assumed to be completed before the phase of file delivery.

There are also some works on the joint caching and downlink file transmission design. For example, a content caching and distribution framework for heterogeneous OFDM networks was investigated in [5], where file requests of users can be served either by the BS directly or by cache nodes. In [6], an optimal caching and user association policy was proposed to minimize the latency in a cached-enabled heterogeneous network with wireless backhaul. In the above works, the files are delivered to small BSs via dedicated backhaul links, without resource competition with the downlink transmission. Hence the scheduling is optimized in each individual file transmission. In practice, it is possible that there is no dedicated link or period for file placement. The file placement and delivery are in a multicast manner [7]. This raises an trade-off between the transmission resource consumption and the file placement. For example, if more resource is spent in downlink multicast, files will be cached in more devices, which may save the downlink resource in future transmissions. As a result, a joint optimization of file placement and delivery with the consideration of total transmission resource consumption becomes necessary. To our best knowledge, this is still an open issue. In our previous work [8], this resource allocation issue was addressed via dynamic programming, however without considering the cache size constraint.

In this paper, we consider the downlink transmission in a cell with the assistance of multiple cache nodes during a finite file lifetime, where users appear randomly in each frame to request files from a database. All the cache nodes are empty at the very beginning of the lifetime. When one file is requested, the BS multicasts it to the requesting user as well as some chosen cache nodes according to the channel condition and cache status. The current file transmission may change the cache status, which affects the future file transmissions. We
formulate the transmission resource optimization as an MDP with a random number of stages, where the objective is to minimize the average total downlink resource consumption at the BS. We first show that the optimal solution can be obtained via a revised Bellman’s equation. Note that there is a common phenomenon namely curse of dimensionality for a general MDP problem, which refers to the prohibitive computation complexity. We further propose a novel approach to approximate the value functions of the Bellman’s equation by analytical expressions. Thus, the computation complexity is significantly reduced. The approximated value functions are upper-bounds of the true ones, and an upper-bound on the approximation error is also provided. It is shown by simulations that the proposed dynamic programming approach could effectively exploit the coupling of multiple file transmissions and suppress the total resource consumption at the BS.

II. SYSTEM MODEL

A. Network Model

As illustrated in Fig. 1, we consider the downlink file transmission in a cell with one BS and $N_C$ single-antenna cache nodes. There are $N_P$ antennas at the BS. Let $C \subseteq \mathbb{R}^2$ be the service area of the cell, $C_c (\forall c = 1, 2, ..., N_C)$ be the service region of the $c$-th cache node and $C_0 := C - C^*$ be the region not served by any cache node, where $C^* := C_1 \cup C_2 \cup ... \cup C_{N_C}$. In this paper, we consider cache node deployment without overlapping, thus assuming $C_c \cap C_j = \emptyset$ for $\forall i \neq j$. A database with $M_F$ files is accessible to the BS. Each file consists of $R_F$ information bits, and each cache node can store at most $M_C$ files ($M_C < M_F$). In this paper, we consider the downlink transmission of these files within a finite common lifetime, which captures the practical scenario where the popularity of a number of content drops down quickly after a certain period (e.g. breaking news). Within the lifetime, users randomly request files from the BS. The locations of the requesting users are independent and identically distributed (i.i.d.) in the cell according to certain spatial distribution.

Suppose there are $T$ frames in the common file lifetime $\mathcal{L}$, i.e. $\mathcal{L} \triangleq \{1, 2, ..., T\}$. Let $\mathcal{F} \triangleq \{1, 2, ..., M_F\}$ be the set of files. In each frame, it is assumed that there is at most one file request, and the $f$-th file is requested by one user with probability $\beta p_f$, where $\beta \in [0, 1]$ is the probability that there is one file request in one frame$^1$, and $p_f$ is the probability that the $f$-th file is selected among $\mathcal{F}$. Without loss of generality, we assume $p_1 \geq p_2 \geq ... \geq p_{M_F}$. In this paper, we do not have any constraint on the distribution of $\{p_f|\forall f\}$. For example, they may follow the Zipf distribution as in [9]. For elaboration convenience, we refer to $\mathcal{F}^H \triangleq \{1, 2, ..., M_C\}$ and $\mathcal{F}^L \triangleq \{M_C + 1, ..., M_F\}$ as the high-popularity and low-popularity file sets, respectively.

At the beginning of the lifetime, all the caches at the cache nodes are empty. The BS reactively multicasts the requested file to a user and some cache nodes, if the requesting user cannot be served by the cache node nearby. Hence there are two types of file transmissions in the network, namely reactive multicast and device-to-device (D2D) communications. The former refers to the downlink file delivery from the BS, and the latter is the data transmission from the cache nodes to the users. It is assumed that the D2D links can use Wi-Fi, bluetooth, or other air interfaces, which are not in the same spectrum as the downlink transmission [10] [11]. In this paper, we shall minimize the transmission resource consumption, including the transmission energy and time, at the BS by offloading traffics to the cache nodes.

Remark 1. We consider the network deployment where the transmission from the BS to the cache nodes shares the downlink resource with cellular users. Moreover, there is no dedicated time period for file placement before lifetime. Therefore, one file transmission may lead to the cache status update, which further affects the following transmissions of all files. Since new requesting users appear randomly, it is impossible to determine the transmission parameters for all requests at the very beginning of the lifetime. We shall address this complicated multi-transmission scheduler design problem via a finite-horizon MDP.

B. Downlink Physical Layer Model

In downlink, the space-time block code (STBC) with full diversity is used at the BS for two reasons: (1) there is no requirement on the channel state information at the transmitter (CSIT); (2) the diversity gain can be achieved at all the receivers.

For elaboration convenience, we refer to the user, who sends the $\tau$-th request, as the $\tau$-th user. Since the transmission time of one file is much larger than the channel coherent time, it is assumed that the ergodic channel capacity span all possible small-scale channel fading can be achieved during one file transmission. Let $\rho(\tau)$ and $p_{\tau}$ be the pathloss from the BS to the $\tau$-th user and the $c$-th cache node respectively, and $\eta(\tau)$ and $\eta'(\tau)$ be the corresponding shadowing coefficients during the $\tau$-th file transmission. $\eta(\tau)$ and $\eta'(\tau)$ are quasi-static within one file transmission and change independently and identically for different transmissions. Moreover, let $P_{\tau}$ be the downlink transmission power for the $\tau$-th user, $N_r$ be the number of downlink transmission symbols scheduled for the $\tau$-th file transmission, the throughput achieved by the $\tau$-th user is given by

$$R(\tau) = N_r \mathbb{E}_{h_{\tau}} \left[ \alpha \log_2 \left( 1 + \frac{|h_{\tau}|^2 P_{\tau}}{N_T \sigma_z^2} \right) \right],$$

where $\alpha$ is the transmission rate of the adopted full-diversity STBC, $\sigma_z^2$ is the power of noise as well as inter-cell interference, $h_{\tau}$ is the i.i.d. channel vector from the BS to the requesting user. Each element of $h_{\tau}$ is complex Gaussian distributed with zero mean and variance $\rho(\tau)\eta(\tau)$.

The $\tau$-th user can decode the file only when $R(\tau) \geq R_F$. 

$^1$Note that as the transmission of one file in practice usually consumes a large number of frames, it is assumed that the probability that there is more than one new request per frame is negligible. Otherwise, the network will be overloaded.
Moreover, the achievable throughput of the \( c \)-th cache node is given by
\[
R^c(\tau) = N_r \mathbb{E}_h \left[ \alpha \log_2 \left( 1 + \frac{||h^c \|_2^2 P_r}{N_T \sigma^2} \right) \right],
\]
where \( h^c \) is the i.i.d. channel vector from the BS to the \( c \)-th cache node. Each element of \( h^c(\tau) \) is complex Gaussian distributed with zero mean and variance \( \rho_c \eta^2(\tau) \). The \( c \)-th cache node can decode the file only when \( R^c(\tau) \geq R_F \).

\section{System State and Scheduling Policy}

When a requesting user (say the \( \tau \)-th user) cannot be served by its nearby cache node, the BS should determine the downlink transmission power \( P_r \) and transmission time \( N_r \) for multicast. In order to formulate the downlink scheduler design as an MDP, we first define the system state and scheduling policy as follows.

\begin{definition}[System State]
At the \( \tau \)-th request, the system status \( S_\tau \) is uniquely specified by:
\begin{itemize}
  \item Index of the requested file: \( A_\tau \in \mathcal{F} \).
  \item Cache state information: \( \{B^c_f(\tau)\}_{f \in \mathcal{F}, c = 1, \ldots, N_C} \) where \( B^c_f(\tau) = 1 \) means that the \( f \)-th file has been stored by the \( c \)-th cache node at the \( \tau \)-th request, and \( B^c_f(\tau) = 0 \) means otherwise.
  \item Statistical channel state information (SCSI): the pathloss and shadowing coefficients from the BS to all cache nodes and the \( \tau \)-th requesting user, \( [\eta(\tau), \eta^c(\tau), \rho(\tau)]_{\forall c = 1, \ldots, N_C} \).
\end{itemize}
\end{definition}

\begin{definition}[Multicast Scheduling Policy]
At the \( \tau \)-th request (\( \forall \tau \)), the scheduling policy \( \Omega_\tau \) is a mapping from the system state \( S_\tau \) and the number of frames in the remaining lifetime \( T_\tau \) to the following scheduling actions:
\begin{itemize}
  \item BS transmission parameters: \( \{P_r, N_r\} \).
  \item Cache update: \( \{\Delta B^c_f(\tau)\}_{f \in \mathcal{F}, c = 1, \ldots, N_C} \) where \( \Delta B^c_f(\tau) \in \{-1, 0, 1\} \) and \( B^c_f(\tau + 1) = B^c_f(\tau) + \Delta B^c_f(\tau) \).
\end{itemize}
Meanwhile, the following constraints should be satisfied:
\begin{itemize}
  \item Successful decoding at the requesting user:
  \[
  R(\tau) \geq R_F. \tag{3}
  \]
  \item Cache size constraint:
  \[
  \sum_f B^c_f(\tau) \leq M_C, \forall c. \tag{4}
  \]
  \item Cache update constraint:
  \[
  \Delta B^c_{A_\tau}(\tau) \in \{0, 1\} \mathbb{I}[R^c(\tau) \geq R_F], \forall c,
  \]
  \[
  \Delta B^c_f(\tau) \in \{0, -1\}, \forall c, f \neq A_\tau, \tag{5}
  \]
\end{itemize}
where \( \mathbb{I}(\cdot) \) is the indicator function, (3) is the decision on whether to store the decoded file, and (5) is the decision on whether to remove one file from a cache.

\section{Problem Formulation}

In practice, the BS should deliver both popular and unpopular file in downlink. The former delivery can be assisted by cache nodes, and the latter has to be served by the BS. In this paper, we shall minimize the transmission resource consumption at the BS by offloading traffic to the cache nodes, so that more downlink transmission resource can be spared for unpopular file delivery.

Specifically, let \( C_{f,\tau} = \bigcup_{c \in \mathcal{B}^c_f(\tau)=1} \mathcal{C}_c \) be the area where the users are able to receive the \( f \)-th file from one of the cache nodes, and \( I_\tau \) be the location of the \( \tau \)-th user. The resource consumption at the \( \tau \)-th file transmission, measuring the weighted sum of transmission energy and opportunities, is given by
\[
g_\tau = (P_r N_r + w_\tau N_r) I[I_\tau \notin \mathcal{C}_{A_\tau,\tau}],
\]
where \( w_\tau \) is the weight on transmission opportunities. The average total cost of the BS in the whole lifetime is given by
\[
\mathcal{C}(\Omega_1, \Omega_2, \ldots) = \mathbb{E}_{A, \eta, \rho, N_R} \left\{ \sum_{\tau=1}^{N_R} g_\tau \right\},
\]
where the expectation is taken over all possible large-scale channel fading (including the shadowing effect \( \eta \) and pathloss \( \rho \)), requested files (\( A = \{A_\tau|\forall \tau\} \)) and the total request number in the lifetime \( N_R \). The probability mass function (PMF) of \( N_R \) is given by
\[
\Pr(N_R = n) = \binom{T}{n} \beta^n (1 - \beta)^{T-n}, \forall n = 0, 1, \ldots, T.
\]
As a result, the transmission design in this paper can be formulated via the following dynamic programming problem.

\begin{problem}[Optimization with Random Number of Requests]
\[
\min_{\Omega_1, \Omega_2, \ldots} \mathcal{C}(\Omega_1, \Omega_2, \ldots) = \mathbb{E}_{A, \eta, \rho, N_R} \left\{ \sum_{\tau=1}^{N_R} g_\tau \right\}
\]
\quad s.t. Constraints in (3) – (6).
\end{problem}

\section{Optimal Scheduling Policy}

Problem 1 cannot be solved directly by the standard approach for finite-horizon MDPs in [12], which only applies to an MDP with a deterministic number of stages. In order to solve Problem 1, we first introduce the following intermediate MDP problem with fixed \( M_R \) stages, which is similar to Problem 1 except for the stage number.

\begin{problem}[MDP with Fixed Stage Number]
\[
\min_{\Omega_1, \Omega_2, \ldots, \Omega_{M_R}} \mathbb{E}_{A, \eta, \rho} \left\{ \sum_{\tau=1}^{M_R} g_\tau \right\}
\]
\quad s.t. Constraints in (3) – (6),
\end{problem}
where \( M_R \) is a sufficiently large integer such that the probability \( \Pr(M_R < N_R) \) is negligible.
The Bellman’s equation of Problem 2 is
\[
V_{M_R - 	au + 1}(S_T) = \min_{\Omega_T(S_T)} \left\{ g_{\tau} \left[ S_T, \Omega_T(S_T) \right] + \sum_{S_{T+1}} V_{M_R - \tau}(S_{T+1}) \Pr \left[ S_{T+1} \mid S_T, \Omega_T(S_T) \right] \right\}, \forall S_T
\]
s.t.: Constraints in (3) – (6).

In the above equation, \( V_{M_R - \tau + 1}(\cdot) \) is referred to as the value function of the \( \tau \)-th stage, which measures the average remaining cost from the \( \tau \)-th requests to the last \( (M_R-\tau) \)-th request given the current system state. Note that the large-scale fading and requested file index are i.i.d. in each request, we can further simplify the above Bellman’s equation as follows.

**Lemma 1** (Bellman’s Equations with Reduced Space). The optimal scheduling policy of Problem 2 is the solution of the Bellman’s equation with reduced state space in (9), where \( \tilde{V}_{M_R - \tau + 1}(\tilde{S}_T) = E_{A_T, \eta_T, c_T} \left[ V_{M_R - \tau + 1}(S_T) \right] \), \( \tilde{S}_T = \{ S_T \mid \forall \eta(\cdot), \eta^c(\cdot), \rho(\cdot), c \} \) and \( \Omega_T(\tilde{S}_T) = \{ \Omega_T(S_T) \mid \forall A_T, \rho(\cdot), \eta_T(\cdot), \eta^c(\cdot), c \} \).

**Proof.** The proof is similar to that of Lemma 1 in [8].

The standard value iteration introduced in [12] can be used to solve the Bellman’s equations in (9), and obtain the value functions \( \tilde{V}_{M_R - \tau + 1}(\tilde{S}_T) \) \( \forall \tau, \tilde{S}_T \). In the following lemma, we show that Problem 1 can also be solved via the above value functions.

**Lemma 2** (Optimal Control Policy of Problem 1). With the value functions \( \tilde{V}_{M_R - \tau + 1}(\tilde{S}_T) \) \( \forall \tau, \tilde{S}_T \), the optimal control policy for Problem 1, denoted as \( \Omega_T^*(S_T, T_T) \), can be calculated from the revised Bellman’s equations in (10), where \( \tilde{S}_{T+1} \) denotes the cache state at the \( (\tau + 1) \)-th request.

**Proof.** The proof is similar to that of Lemma 2 in [8].

It can be observed from (10) that the optimal scheduling policy at the \( \tau \)-th stage can be obtained by minimizing the total cost of the current stage \( g_{\tau} [ S_T, \Omega_T(S_T) ] \) and the future stages \( \sum_{N, \tilde{S}_{T+1}} N \beta^N(1 - \beta)^{T_T - N} \tilde{V}_N(\tilde{S}_{T+1}) \Pr[\tilde{S}_{T+1} \mid S_T, \Omega_T(S_T)] \), where the latter depends on the value functions \( \tilde{V}_N(\tilde{S}_{T+1}) \) \( \forall N, \tilde{S}_{T+1} \). In fact, the state space of \( \tilde{S}_{T+1} \) is huge, which grows exponentially with respect to \( N_C \) and \( M_F \). The accurate evaluation of the value functions, which is via an iterative algorithm named value iteration, is computationally prohibitive. In the following section, we shall propose an analytical approximation of the value functions for the Bellman’s equations, such that the iterative algorithm can be avoided and the complexity can be essentially reduced.

**V. LOW-COMPLEXITY POLICY DESIGN**

In this section, we shall propose a novel approximation approach for the finite-horizon MDP, where the approximation error can be bounded. According to the definition of the value functions in the Bellman’s equations, \( \tilde{V}_{M_R - \tau + 1}(\tilde{S}_T) \) is the average transmission cost from the \( \tau \)-th transmission to the last given the current cache state information \( \tilde{S}_T \), which can be written as
\[
\tilde{V}_{M_R - \tau + 1}(\tilde{S}_T) = E_{A_T, \eta_T, c_T} \left( \sum_{k=\tau}^{M_R} g_k(\tilde{S}_k, \Omega^*_k) \right)
\]
where \( \Omega^*_k \) is the optimal scheduling policy for the \( k \)-th stage, and the constraints in (3)–(6) should be satisfied. \( \tilde{V}_{M_R - \tau + 1, f}(\tilde{S}_T) \) is the per-file value function, measuring the average cost spent on the \( f \)-th file since the \( \tau \)-th stage. The above decomposition motivates us to approximate \( \tilde{V}_{M_R - \tau + 1}(\tilde{S}_T) \) as
\[
\tilde{V}_{M_R - \tau + 1}(\tilde{S}_T) \approx J_{M_R - \tau + 1}(\tilde{S}_T) = \sum_{f=1}^{M_F} J_{M_R - \tau + 1, f}(\tilde{S}_T),
\]
where \( J_{M_R - \tau + 1} \) and \( J_{M_R - \tau + 1, f} \) are the approximations of \( \tilde{V}_{M_R - \tau + 1} \) and \( \tilde{V}_{M_R - \tau + 1, f} \), respectively. The expressions of \( J_{M_R - \tau + 1, f} \) for \( f \in \mathcal{F}^H \) and \( f \in \mathcal{F}^L \) are elaborated in the following.

**A. Approximation of Per-File Value Functions**

We define \( b_f \) as the cache state that all the cache nodes have successfully decoded and stored the \( f \)-th file, i.e., \( b_f = \{ b_f^i = 1 \mid \forall i \} \). Moreover, \( b_f = \{ b_f^i = 0 \} \cup \{ b_f^i = 1 \mid \forall i \neq f \} \). The approximation for high-popularity files (say the \( H \) files) is given below.

- **Given the cache state \( b_f \),** the per-file value function can be written as
\[
J_{M_R - \tau + 1, f}(b_f) = \tilde{V}_{M_R - \tau + 1, f}(b_f) = (M_R - \tau + 1) \rho_f \Pr[I_f \in C_0] \times E_{A, \eta, c} \left( \frac{w_1 \ln(2) R_E}{\alpha W(\frac{R_E}{\nu_c})} \right) I_f \in C_0,
\]
where \( \theta = E_{h_f} \left[ \log_2 \left( \frac{\| h_c \|^2}{\lambda R_E} \right) \right] \), and \( W(x) \) is the Lambert-W function [13].
- **Given the cache state \( b_f \),** the per-file value function is approximated as
\[
J_{M_R - \tau + 1, f}(b_f) = \min_{k=\tau}^{M_R} \sum_{k=\tau}^{M_R} g_k(\tilde{S}_k, \Omega^*_k) \frac{M_F}{\mathcal{F}^L}
\]
s.t. Constraints in (3) – (6).

Hence, it can be derived that \( J_{M_R - \tau + 1, f}(b_f) \) is given by (11), where \( Q^H, Q_{su}, Q_c \) are given by (12) and \( \theta^c = E_{h_c} \left[ \log_2 \left( \frac{\| h_c \|^2}{\lambda R_E} \right) \right] \).

For any other \( S_T \), \( J_{M_R - \tau + 1, f}(\tilde{S}_T) \) is given by (13). Moreover, in order to approximate \( J_{M_R - \tau + 1, f} \) for \( f \in \mathcal{F}^L \), we first define the following notations.
\[
\bar{V}_{M_{R}-r+1}(\bar{S}_r) = \min_{\Omega_*(\bar{S}_r)} \mathbb{E}_{A_\eta, \eta_0} \left\{ g_{\tau} \left[ S_{\tau}, \Omega_r(\bar{S}_r) \right] + \sum_{S_{r+1}} \bar{V}_{M_{R}-r}(\bar{S}_{r+1}) \Pr \left[ \bar{S}_{r+1} | S_{\tau}, \Omega_r(\bar{S}_r) \right] \right\}, \forall \bar{S}_r.
\]

\[
\Omega^*_r(S_r, T_r) = \arg \min_{\Omega_*} \left\{ g_{\tau} \left[ S_{\tau}, \Omega_r(S_r) \right] + \sum_{N, S_{r+1}} \left( \frac{T_r}{N} \right)^{\beta N(1-\beta)T_r-N} \bar{V}_N(\bar{S}_{r+1}) \Pr(\bar{S}_{r+1} | S_{\tau}, \Omega_r(S_r)) \right\}, \forall S_r, T_r.
\]

\[
J_{M_{R}-r+1, f}(b^*_f) = (1 - p_f)J_{M_{R}-r-1, f}(b^*_f) + p_f \left\{ J_{M_{R}-r-1, f}(b^*_f) \Pr \left[ l_r \in C_{f, r}(b^*_f) \right] + \mathbb{E}_{\eta, \rho} \left[ Q^* \left[ R(\tau) \leq R^c(\tau) \right] \Pr \left[ R(\tau) \leq R^c(\tau) \right] + \mathbb{E}_{\eta, \rho} \left[ \min \{ Q_u, Q_c \} \right] \Pr \left[ R(\tau) > R^c(\tau) \right] \Pr \left[ l_r \notin C_{f, r}(b^*_f) \right] \right\} \right\}
\]

\[
Q^* = \frac{w_1 \ln(2)}{\alpha W(\frac{2^b w_b}{e})} + J_{M_{R}-r-1, f}(b^*_f), Q_u = \frac{w_1 \ln(2)}{\alpha W(\frac{2^b w_b}{e})} + J_{M_{R}-r-1, f}(b^*_f), Q_c = \frac{w_1 \ln(2)}{\alpha W(\frac{2^b w_b}{e})} + J_{M_{R}-r-1, f}(b^*_f)
\]

\[
J_{M_{R}-r+1, f}(b^*_f) = J_{M_{R}-r+1, f}(b^*_f) + \sum_{\{ c \in B^c_f(S_r) = 0 \}} \left\{ J_{M_{R}-r+1, f}(b^*_f) - J_{M_{R}-r+1, f}(b^*_f) \right\}
\]

\[
\bar{V}_{M_{R}-r+1}(\bar{S}_r) \geq \sum_{f \in \mathcal{F}} \left\{ J_{M_{R}-r+1, f}(b^*_f) + \sum_{\{ c \in B^c_f(S_r) = 0 \}} \left\{ J_{1, f}(b^*_f) - J_{1, f}(b^*_f) \right\} \right\} \triangleq L_{M_{R}-r+1}(\bar{S}_r)
\]

\[
J_{M_{R}-r+1}(\bar{S}_r) = \sum_{f \in \mathcal{F}^H} \left\{ J_{M_{R}-r+1, f}(b^*_f) + \sum_{\{ c \in B^c_f(S_r) = 0 \}} \left\{ J_{M_{R}-r+1, f}(b^*_f) - J_{M_{R}-r+1, f}(b^*_f) \right\} \right\}
\]

\[
\sum_{f \in \mathcal{F}^L} \left\{ \sum_{n, \{ \forall \{ B^c_f(S_r) \} = 1 \}} n \bar{g}_f^0 \bar{g}_{f, r, n}(S_r) \right\} = \sum_{f \in \mathcal{F}^H} J_{M_{R}-r+1, f}(b^*_f) + \sum_{f \in \mathcal{F}^L} \bar{G}_f^0
\]

\[
+ \sum_{f \in \mathcal{F}^L} \left\{ \sum_{n, \{ \forall \{ B^c_f(S_r) \} = 1 \}} \left[ J_{M_{R}-r+1, f}(b^*_f) - J_{M_{R}-r+1, f}(b^*_f) \right] I(B^c_f = 0) \right\} - \sum_{n} \sum_{f \in \mathcal{F}^L} \left[ n \bar{g}_f^0 \bar{g}_{f, r, n}(S_r) I(B^c_f = 1) \right]
\]

\[
J_{M_{R}-r+1}(\bar{S}_r)
\]

**Proof:** Please refer to Appendix A.

\[
\bar{G}_f^0 = \min_{A_\eta, \eta_0} \sum_{k=1}^{M_B} g_k I(A_k = f) \nonumber
\]

s.t. (3); $B^c_f = 0$, $\forall c$; $\Delta B^c_f = 0$.

- $\bar{G}_f^0$ is the average transmission cost spent on the $f$-th file and the region $C_c$ in one file request, if the request is on low-popularity files and the requested file is not cached at the $c$-th cache node.

- $\bar{G}_f^0$ is the probability that there are $M_C - \sum_{k=1}^{f-1} B^c_k - 1$ requests on the high-popularity files from the $\tau$-th request to the $(\tau + n - 1)$-th request, and the $(\tau + n)$-th request is also for high-popularity files, given the current cache state $S_r$.

Hence for $f \in \mathcal{F}^L$, the approximation of value function is given by

\[
J_{M_{R}-r+1, f}(\bar{S}_r) = \bar{G}_f^0 - \sum_{\{ \forall \{ B^c_f(S_r) \} = 1 \}} (n - M_C + \sum_{k=1}^{f-1} B^c_k + 1) \bar{g}_f^0 \bar{g}_{f, r, n}(\bar{S}_r).
\]

The expressions of $\bar{G}_f^0$, $\bar{g}_f^0$ and $\bar{G}_f^0 \bar{g}_{f, r, n}(\bar{S}_r)$ can be easily derived, which are omitted here due to page limitation.

**B. Bound on Approximation Error**

We have the following two conclusions on upper-bound and lower-bound of the true value function $\bar{V}_{M_{R}-r+1}(\bar{S}_r)$.

**Lemma 3** (Upper-Bound of Value Function). The approximation of value function $J_{M_{R}-r+1, f}(\bar{S}_r)$ is an upper-bound of the true value function $\bar{V}_{M_{R}-r+1}(\bar{S}_r)$, i.e. $\bar{V}_{M_{R}-r+1}(\bar{S}_r) \leq J_{M_{R}-r+1, f}(\bar{S}_r)$, $\forall \bar{S}_r, \bar{S}_r$.

**Proof.** Please refer to Appendix A.
Lemma 4 (Lower-Bound of Value Function). A lower-bound of the value function \( \hat{V}_{M_{R-\tau+1}}(S_\tau) \) is given by (14).

Proof. This cost lower-bound is obtained by assuming the cache size at the cache nodes is sufficiently large for all the files in \( \mathcal{F} \). The proof is similar to that of Lemma 5 in [8]. \( \square \)

Note that \( J_{M_{R-\tau+1}}(47) \) is the proposed approximation of value functions, and its approximation error is given by

\[
J_{M_{R-\tau+1}}(\tilde{S}_\tau) - \tilde{V}_{M_{R-\tau+1}}(\tilde{S}_\tau) \\
\leq J_{M_{R-\tau+1}}(\tilde{S}_\tau) - L_{M_{R-\tau+1}}(\tilde{S}_\tau), \forall \tau, \tilde{S}_\tau,
\]

where \( L_{M_{R-\tau+1}} \) is defined in (14).

C. Scheduling Policy with Approximated Value Functions

The expression of the approximated value function \( J_{M_{R-\tau+1}}(S_\tau) \) is summarized in (15), where \( \Xi_{\mathcal{C}} = \{B^c_f(\tau)\} \forall f \) is state of the \( c \)-th cache node at the \( \tau \)-th request. Moreover, \( J_{M_{R-\tau+1}}(\tilde{S}_\tau) \) in (15) is referred to as the per-cache cost for the cache state \( \tilde{S}_\tau \). Hence the scheduling for arbitrary system state \( S_\tau \), denoted as \( \Omega_\tau(S_\tau) \), can be obtained via the following optimization problem.

**Problem 3 (Policy Iteration at the \( \tau \)-th Stage).**

\[
\Omega_\tau^*(S_\tau) = \arg\min_{\Omega_\tau} \left\{ \sum_{N,c} \left( \frac{T_\tau}{N} \right) \beta^N(1-\beta)^{T_\tau-N}J_N^{c}(\tilde{S}_\tau^{c}+1) \right\}
\]

s.t. \( \text{Constraints in (3-6)} \)

where \( T_\tau \) is the number of frames in the remaining lifetime.

Note that this is an integrated continuous and discrete optimization problem, its solution algorithm is given below.

**Algorithm 1.** Let \( d_1, d_2, \ldots, d_{N_{\mathcal{C}}} \) be the indexes of cache nodes, whose large-scale fading coefficients satisfy \( \rho_{d_1} \eta^{d_1}(\tau) \leq \rho_{d_2} \eta^{d_2}(\tau) \leq \ldots \leq \rho_{d_{N_{\mathcal{C}}}} \eta^{d_{N_{\mathcal{C}}}}(\tau) \). Without loss of generality, it is assumed that \( \rho_{d_m} \eta^{d_m}(\tau) \leq \rho(\tau) \eta(\tau) \leq \rho_{d_{m+1}} \eta^{d_{m+1}}(\tau) \). Thus the large-scale fading coefficient of the requesting user is between the \( m \)-th and \((m+1)\)-th cache nodes. The optimal solution of Problem 3 can be obtained as follows.

- **If** \( f \in C_{\mathcal{A},\tau} \), the user can be served by the nearby cache node. Thus \( P_\tau = 0, N_\tau = 0 \) and \( \triangle B^c_f(\tau) = 0 \), \( \forall c, f \).
- **Otherwise,** for each \( i = 1, 2, \ldots, m + 1 \), calculate

\[
Q^*_{i} = \min \left\{ P_\tau N_\tau + w_i N_\tau \right\}
\]

\[
+ \sum_{c=d_i}^{d_{N_{\mathcal{C}}}} \sum_{N} \left( \frac{T_\tau}{N} \right) \beta^N(1-\beta)^{T_\tau-N}J_N^{c}(\tilde{S}_\tau^{c}+1) \right\}
\]

\[
+ \sum_{c=d_i}^{d_{N_{\mathcal{C}}}} \sum_{N} \left( \frac{T_\tau}{N} \right) \beta^N(1-\beta)^{T_\tau-N}J_N^{c}(\tilde{S}_\tau^{c}+1) \right\}
\]

\[
\text{s.t. } \text{Constraints in (3-6)}
\]

In the above equation, \( \tilde{S}_\tau^{c} \) represents the cache state of the \( c \)-th cache node after decoding the \( A_\tau \)-th file. Thus the \( A_\tau \)-th file will be stored if there is spare memory or files with lower popularity (than \( A_\tau \)) at the \( c \)-th cache node with state \( \tilde{S}_\tau^{c} \). The optimal BS scheduling parameters for the above problem are denoted as \( [P_\tau^{d_i}, N_\tau^{d_i}] = \left[ \frac{w_i}{W(2^{\beta\tau_{\mathcal{A}}})}, \frac{R_F \ln(2)}{\alpha W(2^{\beta\tau_{\mathcal{A}}}) + 1} \right] \), and \( \theta^{d_i} = E_{h^c_{\mathcal{A}}} \left[ \log_2 \left( \frac{\|h^c_{\mathcal{A}}\|^2}{\theta^{d_i}} \right) \right] \).

- Let \( d^* = \arg\min_{d_i} Q^* \), the solution of **Problem 3** is then given by \( \Omega^*_\tau(S_\tau) = [P_\tau^{d^*}, N_\tau^{d^*}] \).

VI. SIMULATION

In the simulation, the radius of one cell is 500 meters, cache nodes are randomly deployed on the cell-edge region with a service radius of 90 meters. The number of antennas at the BS is 8, and the number of cache node is 20. The downlink pathloss exponent is 3.5, and the transmission bandwidth is 20MHz. The weights on transmission time is \( w_i = 100 \). The number of popular files is \( M_F = 20 \). It is assumed that \( \{p_f\} \) follows a Zipf distribution [9] with skewness factor \( 2\gamma \). In addition to the proposed algorithm, the performance of the following two baseline schemes is also compared.

**Baseline 1.** The BS only ensures the file delivery to the requesting user in each transmission. The cache nodes with better channel conditions to the BS can also decode the file. The decoded file will be stored at the cache node if there

\[ 2\gamma \] is the factor characterizing the Zipf distribution in which \( \gamma \to 0 \) makes distribution steeper, whereas \( \gamma \to 0 \) makes distribution more uniform.
is spare memory or files with lower popularity at the cache node.

**Baseline 2.** The BS ensures that all the cache nodes can decode each file in its first transmission. Files with lowest popularity will be replaced at cache nodes if the caches are fully occupied.

The average transmission cost versus the average number of file requests is illustrated in Fig. 2 (a). It is shown that the proposed algorithm consumes less transmission resource than both baselines. Moreover, the performance gain tends to be a constant when $\beta T$ is large. This is because all the three schemes have the same performance if all the high-popularity files have been stored in cache nodes. In other words, the gain of the proposed scheme lies in the phase of caching.

In Fig. 2 (b), the performance of the three schemes are compared for different Zipf distributions. It can be observed that the gain of the proposed scheme over Baseline 2 is more significant for uniform popularities of files. This is because the transmission coupling between the high-popularity and low-popularity files are better exploited in the proposed scheme.

In Fig. 2 (c), the effect of the cache size is evaluated. It can be observed that when increasing the cache size, more traffic can be offloaded to cache nodes, which leads to less transmission resource consumption at the BS in all three schemes. Moreover, the proposed scheme has better utilization of the caches than Baseline 1 especially for large $M_C$, as the performance gain increases with respect to $M_C$.

**VII. CONCLUSION**

In this paper, we consider the downlink file transmission with the assistance of cache nodes in a finite lifetime. The BS multicasts files to the requesting users and the selected cache nodes, and the cache nodes with decoded files can help to offload the traffic upon the successive requests via other air interfaces. We formulate the optimization of file placement and delivery as an MDP with a random number of stages, and propose a revised Bellman’s equation to obtain the optimal control policy. In order to avoid the curse of dimensionality, we also introduce a low-complexity sub-optimal solution based on linear approximation of the value functions, which can be calculated analytically. It is shown by simulations that the proposed low-complexity algorithm can significantly reduce the resource consumption at the BS.

**APPENDIX A: PROOF OF LEMMA 3**

We first introduce the following heuristic scheduling policy.

**Policy 1 (Heuristic Scheduling Policy $\{\Omega^1_s|\forall \tau\}$).** For arbitrary cache state $S_\tau$, the scheduling policy from the $\tau$-th request to the $M_R$-th request is determined as follows.

- **Cache nodes will not decode any low-popularity files.** The requesting users are the only receivers of low-popularity files in downlink. Thus, if the $k$-th request ($\forall k = \tau, ..., M_R$) is on the low-popularity file, the transmission parameters are determined as

$$\{P_k, N_k\} = \arg\min P_k N_k + w_c N_k$$

s.t. (3); $\Delta B_{A_k}^{\tau}(\tau) = 0, \forall c$.

- The transmission of high-popularity files is optimized such that the total average cost on these files is minimized. Thus, let $\Omega^{1, H}_s$ be the control policy at the $\tau$-th stage on high-popularity file, we have

$$\Omega^{1, H}_s = \arg\min \mathbb{E}_{A_s, \rho} \sum_{k=\tau}^{M_R} g_k I(A_r \in F^H)$$

s.t. Constraints of (3, 4, 5, 6).

With the Policy 1, let $U_{M_R-\tau+1,f}(S_\tau)$ be the average cost from the $\tau$-th stage to the $M_R$-th stage, and $U_{M_R-\tau+1,f}(S_\tau)$ be the per-file cost on the $f$-th file, we have

$$\sum_{f=1}^{S_\tau} U_{M_R-\tau+1,f}(S_\tau) = U_{M_R-\tau+1}(S_\tau) \geq U_{M_R-\tau+1}(S_\tau)$$

It is easy to see that $U_{M_R-\tau+1,f}(S_\tau) \leq J_{M_R-\tau+1,f}(S_\tau)$, $\forall f, \tau, S_\tau$. This finishes the proof.

**REFERENCES**

[1] C. Yang, Y. Yao, Z. Chen, and B. Xia, “Analysis on cache-enabled wireless heterogeneous networks,” IEEE Transactions on Wireless Communications, vol. 15, no. 1, pp. 131–145, Jan 2016.

[2] S. H. Chae and W. Choi, “Caching placement in stochastic wireless caching helper networks: Channel selection diversity via caching,” IEEE Transactions on Wireless Communications, vol. 15, no. 10, pp. 6620–6637, Oct 2016.

[3] M. A. Maddah-Ali and U. Niesen, “Fundamental limits of caching,” IEEE Transactions on Information Theory, vol. 60, no. 5, pp. 2856–2867, May 2014.

[4] X. Xu and M. Tao, “Modeling, analysis, and optimization of coded caching in small-cell networks,” IEEE Transactions on Communications, vol. 65, no. 8, pp. 3415–3428, Aug 2017.

[5] X. Huang and N. Ansari, “Content caching and user scheduling in heterogeneous wireless networks,” in 2016 IEEE Global Communications Conference (GLOBECOM), Dec 2016, pp. 1–6.

[6] Y. Cui, F. Lai, S. Hanly, and P. Whiting, “Optimal caching and user association in cache-enabled heterogeneous wireless networks,” in 2016 IEEE Global Communications Conference (GLOBECOM), Dec 2016, pp. 1–6.

[7] Y. Cui and D. Jiang, “Analysis and optimization of caching and multicasting in large-scale cache-enabled heterogeneous wireless networks,” IEEE Transactions on Wireless Communications, vol. 16, no. 1, pp. 250–264, Jan 2017.

[8] B. Lv, L. Huang, and R. Wang, “Cellular offloading via downlink cache placement,” in 2018 IEEE International Conference on Communications (ICC), 2018.

[9] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, “Web caching and zipf-like distributions: evidence and implications,” in INFOCOM ’99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 1, Mar 1999, pp. 126–134 vol.1.

[10] K. Poularakis, G. Isisofidis, I. Pefkianakis, L. Tassiulas, and M. May, “Mobile data offloading through caching in residential 802.11 wireless networks,” IEEE Transactions on Network and Service Management, vol. 13, no. 1, pp. 71–84, March 2016.

[11] M. Ji, G. Caire, and A. F. Molisch, “Wireless device-to-device caching networks: Basic principles and system performance,” IEEE Journal on Selected Areas in Communications, vol. 34, no. 1, pp. 176–189, Jan 2016.

[12] D. P. Bertsekas, Dynamic Programming and Optimal Control. Athena Scientific, 2000.

[13] U. Tamir, “Some reflections about the Lambert W function as inverse of $x\log(x)$,” in 2014 Information Theory and Applications Workshop (ITA), Feb 2014, pp. 1–4.