On the Optimality of Separation between Caching and Delivery in General Cache Networks

Navid Naderializadeh*, Mohammad Ali Maddah-Ali†, and A. Salman Avestimehr*

*Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA
†Nokia Bell Labs, Holmdel, NJ, USA
E-mails: naderial@usc.edu, mohammad.maddah-ali@nokia.com, avestimehr@ee.usc.edu

Abstract

We consider a system, containing a library of multiple files and a general memoryless communication network through which a server is connected to multiple users, each equipped with a local isolated cache of certain size that can be used to store part of the library. Each user will ask for one of the files in the library, which needs to be delivered by the server through the intermediate communication network. The objective is to design the cache placement (without prior knowledge of users’ future requests) and the delivery phase in order to minimize the (normalized) delivery delay. We assume that the delivery phase consists of two steps: (1) generation of a set of multicast messages at the server, one for each subset of users, and (2) delivery of the multicast messages to the users.

In this setting, we show that there exists a universal scheme for cache placement and multicast message generation, which is independent of the underlying communication network between the server and the users, and achieves the optimal delivery delay to within a constant factor for all memoryless networks. We prove this result, even though the capacity region of the underlying communication network is not known, even approximately. This result demonstrates that in the aforementioned setting, a separation between caching and multicast message generation on one hand, and delivering the multicast messages to the users on the other hand is approximately optimal. This result has the important practical implication that the prefetching can be done independent of network structure in the upcoming delivery phase.

I. INTRODUCTION

We consider a system with a server and multiple users, where the server has access to a library of files (e.g., movies). Each user is equipped with an isolated cache of certain size which can be used to store parts of the library. The server is connected to the users via an arbitrary memoryless communication network that could be consisting of multiple relays (wireline or wireless). This represents a general cache network which encompasses the network configurations studied in several prior works, including bottleneck [1], multiserver [2], combinatorial [3], [4], and tree [5] cache networks. The system operates in two phases. In the first phase, called the prefetching phase, each cache is populated up to its limited size from the contents of the library. This phase is followed by a delivery phase, where each user reveals its request for a file in the library. Afterwards, the delivery strategy consists of generating a set of multicast messages at the server, one for each subset of users, and then delivering the generated multicast messages to the users through the intermediate communication network.
Our main result is to demonstrate that there exists a universal scheme for cache placement at the users and multicast message generation at the server, which is independent of the underlying communication network between the server and the users, and is approximately optimal for all cache network configurations (in terms of the delivery delay). In particular, we demonstrate that the prefetching and multicast message generation which takes place assuming that there exists a bottleneck link (see [5]) is approximately optimal for all network configurations. Note that even though we do not know how to use the communication network between the server and the users optimally in order to deliver the multicast messages, as opposed to the special case of shared bottleneck link, we still show that the corresponding prefetching and delivery is approximately optimal.

The importance of our result lies in the fact that in our proposed universal scheme, caching and multicast message generation take place without the knowledge of the structure of the communication network between the server and the users, yet they are approximately optimal. This is particularly important since most cache networks are time-varying and dynamic. Hence, it is critical to rely on a scheme which is oblivious to these variations. Our result suggests that the proposed universal scheme performs almost as well as any other scheme with a priori knowledge of all network dynamics. Moreover, our result essentially demonstrates that a separation between caching and multicast message generation on one hand, and delivering the multicast messages to the users on the other hand is approximately optimal. As mentioned before, this separation is practically important since it demonstrates that the design of caching and multicast message generation can be done without knowledge of the underlying communication network.

In order to prove the main result, we first propose a universal scheme for cache placement and multicast message generation, inspired by the decentralized coded caching scheme presented in [5] for the case of bottleneck link networks, and characterize its achievable normalized delivery delay (as a function of the multicast capacity region of the underlying network). We then provide an outer bound on the normalized delivery delay achievable by any scheme with full knowledge of the communication network between the server and the users, where we first bound the normalized delivery delay for any fixed set of user demands and then relax worst-case to average demands to obtain a lower bound on the optimal normalized delivery delay. Finally, through optimizing the rates over the multicast capacity region of the network, we show that the normalized delivery delay achievable by our proposed universal scheme is within a constant factor of the outer bound, hence establishing our main result.

The rest of the paper is organized as follows. We describe the system model and provide two motivating examples in Section [II]. We formulate the problem and state our main result in Section [III]. We prove our main result in Section [IV]. Finally, we conclude the paper in Section [V].

II. SYSTEM MODEL AND MOTIVATING EXAMPLES

A. System Model

Consider a system with a server $S$ and $K$ users $\{D_i\}_{i=1}^K$, where the server has access to a library of $N$ files $\{W_1, W_2, ..., W_N\}$, each containing $F$ bits. More precisely, each file is uniformly selected from $[2^F]$, where for any integer $n \in \mathbb{N}$, $[n]$ is used to denote the set $\{1, 2, ..., n\}$. Also, we assume that each user is equipped with a cache memory of size $MF$ bits which can be used to cache parts of the library at the users. The server is connected to the
Fig. 1. System model for the case of $K = 3$ users and $m$ relays, with a library of $N$ files, each containing $F$ bits, and an isolated cache of size $MF$ bits at each user. The communication network is modeled by the probability distribution $p(y_{H_1}, \ldots, y_{H_m}, y_{D_1}, \ldots, y_{D_K} | x_S, x_{H_1}, \ldots, x_{H_m})$, where $x_S$ and $(x_{H_1}, \ldots, x_{H_m})$ respectively denote the transmit signal by the server and the relays, and $(y_{H_1}, \ldots, y_{H_m})$ and $(y_{D_1}, \ldots, y_{D_K})$ respectively denote the received signal of the relays and the users. Moreover, for any subset $D \subseteq [3]$, $V_D$ denotes the message that the server multicasts to the users $\{D_i\}_{i \in D}$.

users through an arbitrary memoryless communication network consisting of $m$ relays $\{H_i\}_{i=1}^m$, with the channel model given by an arbitrary transition probability $p(y_{H_1}, \ldots, y_{H_m}, y_{D_1}, \ldots, y_{D_K} | x_S, x_{H_1}, \ldots, x_{H_m})$, where for each node $j$, $X_j$ represents the input alphabet and $Y_j$ represents the output alphabet, $x_S \in X_S$ and $(x_{H_1}, \ldots, x_{H_m}) \in X_{H_1} \times \ldots \times X_{H_m}$ respectively denote the channel input by the server and the relays, and $(y_{H_1}, \ldots, y_{H_m}) \in Y_{H_1} \times \ldots \times Y_{H_m}$ and $(y_{D_1}, \ldots, y_{D_K}) \in Y_{D_1} \times \ldots \times Y_{D_K}$ respectively denote the channel output to the relays and the users. This represents a general cache network model where the caches are located at the edge of the network and it includes bottleneck [1], multiserver [2], combinatorial [3], [4], and tree [5] cache networks. An example of such a system model with $K = 3$ users is illustrated in Figure 1.

The system operates in two phases:

**Prefetching Phase:** In this phase, each user $D_i$, $i \in [K]$, caches a subset of bits from the library, denoted by $Q_i$, such that $|Q_i| \leq MF$. We restrict our attention to uncoded prefetching in which each user is allowed to cache any subset of the bits of the files in the library in an uncoded manner.

**Delivery Phase:** In this phase, each user $D_i$, $i \in [K]$, will reveal its request for a file $W_{d_i}$ from the library. After the demand vector $d = [d_1 \cdots d_K]^T$ is revealed, the delivery strategy consists of the following:

- **Multicast message generation:** Generating $2^K - 1$ multicast messages $\{V_D\}_{D \subseteq [K]}$ at the server, and
- **Multicast message delivery:** Implementing an encoder at the server and decoders at the users, and a relaying strategy for each relay, so that for each subset $D \subseteq [K]$, the message $V_D$ is multicast to the users $\{D_i\}_{i \in D}$.

The multicast messages are generated such that if they are successfully delivered, each user $D_i$, $i \in [K]$, will be able to decode its desired file $W_{d_i}$ based on its cache contents $Q_i$ and its decoded multicast messages $\{V_D\}_{i \in D}$.

---

1 For two sets $A$ and $B$, $A \subseteq B$ is equivalent to $A \subseteq B, A \neq \emptyset$. 
In order to deliver the multicast messages, the server selects a multicast rate tuple \((R_D)_{D \subseteq [K]}\) from the multicast capacity region \(C\), which is defined as follows.

**Definition 1.** Consider a communication network represented as

\[
(\mathcal{X}_S \times \mathcal{X}_{H_1} \times ... \times \mathcal{X}_{H_m}, \mathcal{Y}_{H_1} \times ... \times \mathcal{Y}_{H_m} \times \mathcal{Y}_{D_1} \times ... \times \mathcal{Y}_{D_K}, p(y_{H_1}, ..., y_{H_m}, y_{D_1}, ..., y_{D_K} | x_S, x_{H_1}, ..., x_{H_m})).
\]

A multicast rate tuple \((R_D)_{D \subseteq [K]}\) is said to be achievable if for each \(D \subseteq [K]\), the server can multicast an independent message \(V_D\) to the users \(\{D_i\}_{i \in D}\) at rate \(R_D\) with vanishing error probability as the blocklength increases. The multicast capacity region \(C\) is defined as the closure of the set of all achievable multicast rate tuples.

The goal is, given a communication network between the server and the users and the cache capacity of each user, to design the cache placement and the delivery strategy in order to minimize the delivery delay.

**B. Motivating Examples**

In this section, we first present two examples to motivate a universal scheme for cache networks. In particular, we consider a system with \(N = 3\) files \(W_1 = A, W_2 = B,\) and \(W_3 = C\) and \(K = 3\) users, each with a cache of size equal to \(M = 1\) file. As for the communication network between the server and the users, we consider two different scenarios in the following examples, for each of which we present prefetching and delivery schemes tailored to the communication network between the server and the users. We will then present a universal scheme which is applicable to both network configurations and achieves the optimal normalized delivery delay in both of the examples.

**Example 1 (Shared bottleneck link).** Consider the communication network illustrated in Figure 2a, in which all the three users receive the same signal which is transmitted by the server through a shared bottleneck link. This is the same model considered in [1], which suggests that the prefetching and delivery phases can be done as follows:

![Diagram](image_url)

Fig. 2. (a) The “shared bottleneck link” communication network, where all the users receive the same signal through a shared bottleneck link, and (b) The “orthogonal links” communication network, where each user receives a distinct signal through its dedicated orthogonal link.

Example 1 (Shared bottleneck link). Consider the communication network illustrated in Figure 2a, in which all the three users receive the same signal which is transmitted by the server through a shared bottleneck link. This is the same model considered in [1], which suggests that the prefetching and delivery phases can be done as follows:
• Prefetching phase: We break each file $W_n, n \in \{1, 2, 3\}$, to three disjoint subfiles $\{W_{n,1}, W_{n,2}, W_{n,3}\}$ each of size $\frac{F}{3}$ bits. Then, each user $D_i, i \in \{1, 2, 3\}$, caches

$$Q_i = \{W_{1,i}, W_{2,i}, W_{3,i}\} = \{A_i, B_i, C_i\}.$$  (1)

• Delivery phase: Without loss of generality, suppose that users $D_1, D_2$ and $D_3$ request files $A$, $B$ and $C$, respectively. The server multicasts the following messages over the shared bottleneck link.

$$V_{\{1,2\}} = \{A_2 \oplus B_1\}$$  (2)
$$V_{\{1,3\}} = \{A_3 \oplus C_1\}$$  (3)
$$V_{\{2,3\}} = \{B_3 \oplus C_2\},$$  (4)

where $\oplus$ denotes the bitwise XOR operator.

Assuming a multicast capacity of 1 for the shared bottleneck link, this scheme achieves a normalized delivery delay of 1. Moreover, as demonstrated in [6], the aforementioned prefetching and delivery phases lead to the optimal normalized delivery delay in this communication network. \(\triangle\)

**Example 2** (Orthogonal links). Consider the communication network illustrated in Figure 2-b, in which the server can send distinct signals to each of the users through separate dedicated orthogonal links. The prefetching and delivery phases can be done as follows:

• Prefetching phase: Each user caches the first $\frac{F}{3}$ bits of each file in the library. This suggests that $Q_1 = Q_2 = Q_3$.

• Delivery phase: Once the demands $[d_1, d_2, d_3]$ are revealed, for any user $D_i, i \in \{1, 2, 3\}$, the server sends the last $\frac{2F}{3}$ bits of file $W_{d_i}$ to user $D_i$ through its dedicated link. Therefore, we will have

$$|V_{\{1\}}| = |V_{\{2\}}| = |V_{\{3\}}| = \frac{2F}{3}$$  (5)
$$|V_D| = 0, \forall D \subseteq \{1, 2, 3\}, |D| \neq 1.$$  (6)

Assuming a capacity of 1 for each of the orthogonal links, this scheme achieves a normalized delivery delay of $\frac{2}{3}$. Moreover, in this communication network, a simple cut-set outer bound would show that the aforementioned prefetching and delivery phases lead to the optimal normalized delivery delay. \(\triangle\)

The schemes that we presented in Examples 1 and 2 were each tailored to the structure of the corresponding communication network. However, quite interestingly, we can use the same prefetching and multicast message generation that we presented in Example 1 for the case of orthogonal links communication network in Example 2 as well. Namely, the prefetching can be done similarly to (1) and in the delivery phase, we send useful coded messages over each of the orthogonal links. In particular, we generate the multicast messages as

$$V_{\{1,2\}} = \{A_2 \oplus B_1\}$$  (7)
$$V_{\{1,3\}} = \{A_3 \oplus C_1\}$$  (8)
\[ V_{\{2,3\}} = \{ B_3 \oplus C_2 \}. \] (9)

The total size of the message delivered to each user is \( \frac{2K}{3} \), hence this scheme also achieves the optimal normalized delivery delay of \( \frac{2}{3} \) in Example 2.

Motivated by the above examples, the question that we attempt to answer in this paper is whether the scheme that is optimal under the assumption of existence of a bottleneck link is also universally optimal for all network configurations in terms of the normalized delivery delay. Our main result, which we state in the next section, is to provide an affirmative answer to this question.

III. PROBLEM FORMULATION AND MAIN RESULT

In this section, we formally define the problem and we will then state our main result and its implications. For a given communication network between the server and the users with multicast capacity region \( \mathcal{C} \), user cache contents \( \{ Q_i \}_{i=1}^K \) and demand vector \( \mathbf{d} \), the optimal delivery strategy is to optimally generate the multicast messages \( \{ V_D \}_{D \subseteq \emptyset[K]} \) and select the multicast rate tuple \( (R_D)_{D \subseteq \emptyset[K]} \in \mathcal{C} \), in order to minimize the delivery delay. It is clear that for sufficiently large \( F \), delivering the multicast message \( V_D \) to the users \( \{ D_i \}_{i \in D} \) requires a delay of \( \frac{|V_D|}{R_D} \). Hence, for any choice of \( (\mathcal{C}, \{ Q_i \}_{i=1}^K, \mathbf{d}) \), the minimum delivery delay is given by

\[
T(\mathcal{C}, \{ Q_i \}_{i=1}^K, \mathbf{d}) = \inf_{(R_D)_{D \subseteq \emptyset[K]} \in \mathcal{C}} \inf_{(V_D)_{D \subseteq \emptyset[K]} \in \mathcal{C}} \max_{D \subseteq \emptyset[K]} \frac{|V_D|}{R_D}. \tag{10}
\]

Therefore, the optimal normalized delivery delay, denoted by \( T^* \), is achieved by the prefetching and delivery strategies which minimize the maximum delivery delay across all possible demands for sufficiently large \( F \); i.e.,

\[
T^* = \lim_{F \to \infty} \frac{1}{F} \inf_{\{ Q_i \}_{i=1}^K} \sup_{\mathbf{d}} T(\mathcal{C}, \{ Q_i \}_{i=1}^K, \mathbf{d}) \tag{11}
\]

subject to \( |Q_i| \leq MF, \quad \forall i \in [K] \). \tag{12}

Moreover, motivated by Examples 1 and 2, we define a *universal scheme* as follows.

**Definition 2.** A universal scheme is a cache placement and multicast message generation scheme which is only a function of the number of files \( N \), the number of users \( K \) and the cache size at each user \( M \), and is independent of the communication network between the server and the users.

As mentioned in Section II-B our main result is to show that there exists a universal scheme, based on the assumption of existence of a shared bottleneck link, which is approximately optimal for all cache network configurations (i.e., \( K, N, M \)) in terms of the normalized delivery delay. We state this result formally in the following theorem.

**Theorem 1.** There exists a universal scheme with an achievable normalized delivery delay denoted by \( T_{\text{UNIVERSAL}} \) which satisfies

\[
\frac{1}{24} \leq \frac{T^*}{T_{\text{UNIVERSAL}}} \leq 1, \tag{13}
\]
implying that the universal scheme is within a factor of 24 optimal in terms of the normalized delivery delay.

Remark 1. In order to prove Theorem 1, we demonstrate that the prefetching and multicast message generation which takes place assuming that there exists a bottleneck link is approximately optimal for all network configurations; i.e., its achievable normalized delivery delay is always within a factor of 24 of the optimal normalized delivery delay $T^*$. The description of the scheme and its achievable delay will be presented in Section IV-A. Note that even though we do not know how to use the communication network between the server and the users optimally in order to deliver the multicast messages, as opposed to the special case of shared bottleneck link, we still show that the corresponding prefetching and delivery is approximately optimal.

Remark 2. The importance of Theorem 1 lies in the fact that in our proposed universal scheme, caching and multicast message generation can take place without the knowledge of the structure of the communication network between the server and the users, yet it is approximately optimal. This is particularly important since most cache networks are time-varying and dynamic. Hence, it is critical to rely on a scheme which is oblivious to these variations. Theorem 1 suggests that our proposed universal scheme performs almost as well as any other scheme with a priori knowledge of all network dynamics.

Remark 3. Theorem 1 essentially demonstrates that a separation between caching and multicast message generation on one hand, and delivering the multicast messages to the users on the other hand is approximately optimal. As mentioned before, this separation is practically important since it demonstrates that the design of caching and multicast message generation can be done without knowledge of the underlying communication network. Such a scheme admits a layered solution, where the caching and multicast message generation are done in, for example, the application layer and delivering the generated multicast messages is done in the network and link layers. Our result shows that such a layered solution does not violate the optimality.

Remark 4. There has been a surge of recent works on cache networks considering various topologies for the communication network between the server and the users, such as bottleneck [1], multiserver [2], combinatorial [3], [4], and tree [5] cache networks. Theorem 1 implies that under an uncoded prefetching and two-phase delivery assumption (multicast message generation at the server and delivery to the users), there exists a universal scheme which is approximately optimal regardless of the structure of the communication network between the server and the users, hence providing an approximately optimal result for all the aforementioned network configurations.

IV. Proof of Theorem 1

In order to prove Theorem 1, our goal is to show that the universal scheme that is based on the assumption of existence of a shared bottleneck link is approximately optimal for all network configurations. To that end, we first describe the universal scheme and derive its achievable normalized delivery delay. We will then derive an outer bound on the optimal normalized delivery delay and we will show that the normalized delivery delay achieved by our universal scheme is within a factor of 24 of the outer bound.
A. Description of the Universal Scheme

In this section, we describe our universal scheme based on the decentralized algorithm in [5]. In particular, the details of the prefetching and delivery phases are as follows.

- Prefetching phase: In this phase, each user caches $\frac{M}{N}$ bits of each file in the library uniformly at random. This clearly satisfies the cache size constraint at each user.

- Delivery phase: The aforementioned prefetching phase partitions each file $W_n, n \in [N]$ in the library to $2^K$ subfiles $(W_n,D)_{D \subseteq [K]}$, where for any subset $D \subseteq [K]$, $W_n,D$ denotes the bits of file $W_n$ which are exclusively cached by the users $\{D_i\}_{i \in D}$. The law of large numbers implies that for large enough file size $F$, size of each subfile $W_n,D$ is approximately equal to

$$|W_n,D| \approx \left( \frac{M}{N} \right)^{|D|} \left( 1 - \frac{M}{N} \right)^{K-|D|} F.$$ (14)

This suggests that for any $n \in [N]$, the size of $W_n,D$ only depends on the size of $D$. Based on this observation, once the demands $[d_1, d_2, \ldots, d_K]$ are revealed, the server generates the multicast messages as follows.

$$V_D = \oplus_{i \in D} W_{d_i}, D \setminus \{i\}, \forall D \subseteq [K].$$ (15)

As shown in [5], for any subset $D \subseteq [K]$, the size of the corresponding generated multicast message $V_D$ is equal to

$$|V_D| = |D| \cdot (1 - M/N) \cdot \frac{N}{|D|M} \left( 1 - (1 - M/N)^{|D|} \right).$$ (16)

This implies that the normalized delivery delay achieved by our universal scheme, denoted by $T_{\text{UNIVERSAL}}$, can be written as

$$T_{\text{UNIVERSAL}} = \inf_{(R_S)_{S \subseteq [K]} \in \mathcal{C}} \max_{D \subseteq \emptyset [K]} \left\{ \frac{|D| \cdot (1 - M/N) \cdot \frac{N}{|D|M} \left( 1 - (1 - M/N)^{|D|} \right)}{\sum_{S' \subseteq [K]; S' \cap D \neq \emptyset} R_{S'}} \right\}. $$ (17)

B. Approximate Optimality of the Universal Scheme

In this section, we first provide an outer bound on the optimal normalized delivery delay through two steps, where we first bound the normalized delivery delay for any fixed user demands and then we relax the worst-case demands to average demands in order to derive an outer bound on the optimal normalized delivery delay. Afterwards, via optimizing the rates over the multicast capacity region, we demonstrate that the normalized delivery delay achieved by our proposed universal scheme is within a factor of 24 of the outer bound.

1) Bounding the Normalized Delivery Delay for Each Demand Vector: We start the proof of the outer bound by the following lemma which provides lower bounds on the normalized delivery delay for any set of user demands.

Lemma 1. Consider a demand vector $d = [d_1, \ldots, d_K]^T$ and a corresponding operating multicast rate tuple $(R_S)_{S \subseteq [K]} \in \mathcal{C}$. Then we have

$$T \geq \frac{\sum_{R \subseteq [K]; i \in R} \left| W_{d_i,R} \right|}{F \sum_{S \subseteq [K]; i \in S} R_S}, \quad \forall i \in [K]$$ (18)
\[
T \geq \frac{\sum_{R \subseteq [K]: \mathbb{P}[S \cap R] = 0} \frac{1}{|S|} \sum_{i \in D \setminus R} |W_{d, R}|}{F \sum_{S \subseteq [K]: S \cap \emptyset \neq \emptyset} R_{S}^{d}}, \quad \forall D \subseteq [K].
\]  

(19)

\textbf{Proof.} For any nonempty subset of the users \( S \subseteq [K] \), we can write

\[
H(V_S) \leq FR_{S}^{d}T, \quad \forall S \subseteq [K].
\]  

(20)

Now, for any \( n \in [N], R \subseteq [K] \), let \( W_{n, R} \) denote the subfile of file \( W_n \) exclusively cached at the users in \( R \). Then the decodability condition at user \( i \in [K] \) implies

\[
H(W_{d, i}|Q_i, \{V_S\}_{i \in S}) = 0
\]

(21)

\[
\Rightarrow H(\{W_{d, i} \}_i \notin R|\{W_{d, j} \}_j \in [K], i \in R, \{V_S\}_{i \in S}) = 0.
\]  

(22)

We can then write

\[
I(\{V_S\}_{i \in S}: \{W_{d, i} \}_i \notin R|\{W_{d, j} \}_j \in [K], i \in R)
\]

\[
= \begin{cases} 
H(\{V_S\}_{i \in S}|\{W_{d, j} \}_j \in [K], i \in R) - H(\{V_S\}_{i \in S}|W_{d, i}, \{W_{d, j} \}_j \in [K]\{i\}, i \in R) \\
H(\{W_{d, i} \}_i \notin R),
\end{cases}
\]  

(23)

where the second term follows from (22). We therefore have

\[
F \sum_{S \subseteq [K]: i \in S} R_{S}^{d}T \geq H(\{V_S\}_{i \in S})
\]

(24)

\[
\geq H(\{V_S\}_{i \in S}|\{W_{d, j} \}_j \in [K], i \in R)
\]

(25)

\[
= H(\{W_{d, i} \}_i \notin R) + H(\{V_S\}_{i \in S}|W_{d, i}, \{W_{d, j} \}_j \in [K]\{i\}, i \in R),
\]  

(26)

which immediately implies the single-user lower bounds in (18).

Now, note that

\[
H(\{V_S\}_{i \in S}|W_{d, i}, \{W_{d, j} \}_j \in [K]\{i\}, i \in R) = 0,
\]  

(27)

which implies

\[
H(\{V_S\}_{i \in S}|W_{d, i}, \{W_{d, j} \}_j \in [K]\{i\}, i \in R)
\]

\[
= I(\{V_S\}_{i \in S}: \{W_{d, i} \}_i \notin R|W_{d, i}, \{W_{d, j} \}_j \in [K]\{i\}, i \in R)
\]

(28)

\[
= H(\{W_{d, i} \}_i \notin R)
\]

(29)

\[
- H(\{W_{d, j} \}_j \in [K]\{i\}, i \notin R|\{V_S\}_{i \in S}, W_{d, i}, \{W_{d, j} \}_j \in [K]\{i\}, i \in R)
\]

(30)
Now, we have the following lemma which bounds the second term in (34).

**Lemma 2.** For any \( i \in [K] \),

\[
H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) \geq H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) - H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R \cup \{ i \}}) \geq 0.
\]

(31)

\[
= H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) - H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R \cup \{ i \}}) \geq H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) - H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) \geq 0.
\]

(32)

\[
H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R \cup \{ i \}}) \geq H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) - H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) \geq 0.
\]

(33)

Now, we have the following lemma which bounds the second term in (34).

**Lemma 2.** For any \( i \in [K] \),

\[
H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R \cup \{ i \}}) \geq H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) - H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) \geq 0.
\]

(34)

Now, we have the following lemma which bounds the second term in (34).

**Lemma 2.** For any \( i \in [K] \),

\[
H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R \cup \{ i \}}) \geq H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) - H(\{W_{d_j, R} \}_{j \in [K] \setminus \{ i \}, i \notin R, j \notin R}) \geq 0.
\]

(35)

**Proof.** See Appendix A. \qed

Lemma 2 together with (26) and (34), implies

\[
F \sum_{S \subseteq [K] \setminus \{ i \}} R_{S} T \geq \sum_{R \subseteq [K] \setminus \{ i \}} |W_{d_i, R}| \sum_{j \notin R, j \notin R} |W_{d_j, R}| \sum_{j \notin R, j \notin R} \frac{|R|}{1 + |R|} H(W_{d_j, R}), \quad \forall i \in [K].
\]

(36)

Summing the above bound over all \( i \in [K] \) yields

\[
KF \sum_{S \subseteq [K]} R_{S} T \geq \sum_{i=1}^{K} \left[ \sum_{R \subseteq [K] \setminus \{ i \}} |W_{d_i, R}| + \sum_{j \notin R \cup \{ i \}} |W_{d_j, R}| \right] \sum_{j \notin R \cup \{ i \}} \frac{|R|}{1 + |R|}
\]

(37)

\[
= \sum_{R \subseteq [K]} \left[ \sum_{i \notin R} |W_{d_i, R}| + \sum_{j \notin R \cup \{ i \}} |W_{d_j, R}| \right]
\]

(38)

\[
= \sum_{R \subseteq [K]} \frac{K}{1 + |R|} \sum_{i \notin R} |W_{d_i, R}|
\]

(39)

hence implying that

\[
\sum_{S \subseteq [K]} R_{S} T \geq \frac{1}{F} \sum_{R \subseteq [K]} \sum_{i \notin R} |W_{d_i, R}| \frac{|R|}{1 + |R|},
\]

(40)
which is equivalent to the bound in (19) for the case of \( D = [K] \). Through very similar steps, the general bound in (19) can be proven.

2) Relaxing Worst-Case Demands to Average Demands: In this section, we analyze the bounds developed in Lemma 1 for the case of average demands in order to derive a lower bound on the optimal normalized delivery delay. We present this bound in the following lemma.

**Lemma 3.** The optimal normalized delivery delay is lower-bounded by

\[
T^* \geq \max \left\{ \frac{1}{\pi(N,K)} \sum_{d \in \mathcal{P}_{N,K}} \sum_{S \subseteq [K]: |i| \in S} R_d^S \max_{D \subseteq [K]: |D| \geq 2} \frac{1}{\pi(N,K)} \sum_{d \in \mathcal{P}_{N,K}} \sum_{S \subseteq [K]: S \cap D \neq \emptyset} R_d^S \right\}.
\]

**Proof.** Since the bounds in (18) and (19) should hold for any user demand vector, summing them over the set of all distinct demand vectors, denoted by \( \mathcal{P}_{N,K} \), yields

\[
T \sum_{d \in \mathcal{P}_{N,K}} \sum_{S \subseteq [K]: |i| \in S} R_d^S \geq \frac{1}{F} \sum_{d \in \mathcal{P}_{N,K}} \sum_{R \subseteq [K]: i \not\in R} |W_{d,R}|, \quad \forall i \in [K]
\]

(42)

\[
T \sum_{d \in \mathcal{P}_{N,K}} \sum_{S \subseteq [K]: S \cap D \neq \emptyset} R_d^S \geq \frac{1}{F} \sum_{d \in \mathcal{P}_{N,K}} \sum_{R \subseteq [K]: D \subseteq R} 1 + \frac{1}{|D \cap R|} \sum_{i \in D \setminus R} |W_{d,R}|, \quad \forall D \subseteq [K], |D| \geq 2.
\]

(43)

Now, we can bound the RHS of (42) as

\[
\frac{1}{F} \sum_{d \in \mathcal{P}_{N,K}} \sum_{R \subseteq [K]: i \not\in R} |W_{d,R}| = \frac{\pi(N - 1, K - 1)}{F} \sum_{n=1}^{N} \sum_{R \subseteq [K]: i \not\in R} |W_{n,R}|
\]

(44)

\[
= \frac{\pi(N - 1, K - 1)}{F} \left[ \sum_{n=1}^{N} \sum_{R \subseteq [K]} |W_{n,R}| - \sum_{n=1}^{N} \sum_{R \subseteq [K]: i \in R} |W_{n,R}| \right]
\]

(45)

\[
\geq \frac{\pi(N - 1, K - 1)}{F} [NF - MF]
\]

(46)

\[
= \pi(N, K) \left( 1 - \frac{M}{N} \right),
\]

(47)

where the inequality is due to the cache size constraint at user \( D_i \). Furthermore, we can bound the RHS of (43) as

\[
\frac{1}{F} \sum_{d \in \mathcal{P}_{N,K}} \sum_{R \subseteq [K]: D \subseteq R} \frac{1}{1 + |D \cap R|} \sum_{i \in D \setminus R} |W_{d,R}| = \frac{\pi(N - 1, K - 1)}{F} \sum_{n=1}^{N} \sum_{R \subseteq [K]: D \subseteq R} \frac{1}{1 + |D \cap R|} \sum_{i \in D \setminus R} |W_{n,R}|
\]

(48)

\[
= \frac{\pi(N - 1, K - 1)}{F} \sum_{R \subseteq [K]: D \subseteq R} \frac{|D| - |D \cap R|}{1 + |D \cap R|} a_R,
\]

where \( a_R \triangleq \sum_{n=1}^{N} |W_{n,R}| \) is the total number of bits of the library exclusively cached at the users in \( R \). Now, the cache size constraints at the users in \( D \) imply that

\[
|D|MF \geq \sum_{i \in D} \sum_{R \subseteq [K]: i \in R} a_R
\]

(49)
\[ \sum_{R \subseteq \mathcal{K}} |D \cap R| a_R. \]  

(50)

Moreover, the file size constraints imply that

\[ NF = \sum_{R \subseteq \mathcal{K}} a_R \geq \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} a_R, \]  

(51)

which together with (50) yields

\[ (N + |D|)F \geq \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} (1 + |D \cap R|)a_R. \]  

(52)

Furthermore, we have

\[ |D|(N - M)F \leq \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} (|D| - |D \cap R|)a_R. \]  

(53)

Using the Cauchy-Schwartz inequality implies that

\[ \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} (|D| - |D \cap R|)^2 \frac{1}{1 + |D \cap R|} a_R \geq \frac{\sum_{R \subseteq \mathcal{K} : D \not\subseteq R} (|D| - |D \cap R|)a_R}{\sum_{R \subseteq \mathcal{K} : D \not\subseteq R} (1 + |D \cap R|)a_R} \]  

\[ \geq \frac{(|D|(N - M)F)^2}{(N + |D|)F}. \]  

(54)

(55)

Hence, continuing (48), we can write

\[ \frac{1}{F} \sum_{d \in P_{N,K}} \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} \frac{1}{1 + |D \cap R|} \sum_{i \in D \setminus R} |W_{d,i,R}| \geq \pi(N - 1, K - 1) \frac{|D|}{F} \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} \frac{(|D| - |D \cap R|)^2}{1 + |D \cap R|} a_R \]  

\[ \geq \pi(N - 1, K - 1) \frac{|D|^2(N - M)^2}{|D|} \frac{N + |D|M}{N} \]  

\[ \geq \pi(N, K) \frac{|D|(1 - \frac{M}{N})^2}{1 + \frac{|D|M}{N}}. \]  

(56)

(57)

(58)

On the other hand, we have from (48) that

\[ \frac{1}{F} \sum_{d \in P_{N,K}} \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} \frac{1}{1 + |D \cap R|} \sum_{i \in D \setminus R} |W_{d,i,R}| \geq \pi(N - 1, K - 1) \frac{|D|}{F} \sum_{R \subseteq \mathcal{K} : D \not\subseteq R} (|D| - |D \cap R|)a_R \]  

\[ \geq \pi(N - 1, K - 1) \frac{|D|}{F} (|D|(N - M)F) \]  

\[ = \pi(N, K) \left( 1 - \frac{M}{N} \right), \]  

(59)

(60)

(61)

where the inequality in (60) is due to (53). Plugging in (47) into (42), and (58) and (61) into (43) will complete the proof.

3) Rate Optimization over the Multicast Capacity Region: Having Lemma 3, now we can optimize the delivery rates over the multicast capacity region of the network in order to parametrically derive the tightest lower bound on the optimal normalized delivery delay for any multicast capacity region. In particular, denoting the RHS of (41)
by \(T_{lb}\left(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\right)\), the optimal normalized delivery delay can be lower bounded as

\[
T^* \geq \inf_{\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}} T_{lb}\left(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\right).
\]  

(62)

Finally, to prove that our universal scheme is within a constant factor of the outer bound, we have the following lemma, which completes the proof of Theorem 1.

Lemma 4.

\[
\frac{T_{universal}}{\inf_{\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}} T_{lb}\left(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\right)} \leq 24.
\]  

(63)

Proof. Let \(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\) denote the set of rate tuples that minimize \(T_{lb}\left(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\right)\); i.e.,

\[
\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}} = \arg \inf_{\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}} T_{lb}\left(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\right).
\]  

(64)

Then, define

\[
R^*_S \triangleq \frac{1}{\pi(N,K)} \sum_{d \in P_{N,K}} R^d_S, \quad \forall S \subseteq [K].
\]

(65)

Note that since the multicast capacity region \(C\) is convex, the rate tuple \(\{ R^*_S \}_{S \subseteq [K]}\) will belong to \(C\) as well. Using this rate tuple for our universal scheme, it follows from (17) that

\[
T_{universal} \leq \max_{D \subseteq [K]} \left\{ \frac{|D| \cdot (1 - M/N) \cdot \frac{N}{|D|^M} \cdot (1 - (1 - M/N)^{|D|})}{\sum_{S \subseteq [K], S \cap D \neq \emptyset} R^*_S} \right\}
\]

\[
\leq \max_{i \in [K]} \left\{ \frac{1 - M}{N} \sum_{S \subseteq [K], i \in S} R^*_S \cdot \max_{D \subseteq [K], |D| \geq 2} \frac{12 |D|(1 - M/|D|)}{1 + |D|^M} \sum_{S \subseteq [K], S \cap D \neq \emptyset} R^*_S \right\},
\]

(66)

(67)

where the second inequality follows from Corollary 3 in [5]. Now, consider two cases:

- \(\frac{M}{N} \geq \frac{1}{2}\): In this case, we have

\[
12 \frac{|D|(1 - M/|D|)}{1 + |D|^M} \leq 24 \left(1 - \frac{M}{N}\right).
\]  

(68)

- \(\frac{M}{N} < \frac{1}{2}\): In this case, we have

\[
\frac{|D|(1 - M/|D|)^2}{1 + |D|^M} \geq \frac{1}{2} \frac{|D|(1 - M/|D|)}{1 + |D|^M}.
\]  

(69)

Therefore, combining (41) with (67), (68) and (69) implies that

\[
T_{universal} \leq 24 \cdot T_{lb}\left(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\right)
\]

\[
= 24 \inf_{\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}} T_{lb}\left(\{ (R^d_S)_{S \subseteq [K]} \}_{d \in P_{N,K}}\right).
\]

(70)

(71)

Hence, the proof is complete. ☐
V. CONCLUSION AND FUTURE DIRECTIONS

In this paper, we considered a cache network consisting of a server which has access to a library of multiple files and a communication network through which a server is connected to multiple users, each equipped with a local cache of size of certain size. In such a setting, we showed that a separation between caching and multicast message generation on one hand, and delivering the multicast messages to the users on the other hand is approximately optimal. To prove our result, we presented a universal scheme which had been designed for the case of shared bottleneck link networks and we derived its achievable normalized delivery delay. We then developed an outer bound on the normalized delivery delay, and then showed that our universal scheme achieves a normalized delivery delay which is within a constant factor of 24 of the outer bound for all system parameters.

Note that in this work, we relied on two assumptions, namely uncoded prefetching at the users and a two-stage delivery in which the generation of the multicast messages at the server was separate from their delivery to the users through the communication network. Relaxing or removing any of these assumptions may invalidate the approximate optimality of our universal scheme. Therefore, an interesting future direction would be to devise a universal scheme which is (approximately) optimal in the fully-general setting.

APPENDIX A

PROOF OF LEMMA 2

Given a user index \( i \in [K] \), there exist \((K-1)!\) permutations of the users in \([K] \setminus \{i\}\), denoted by \( \Pi_{[K]\setminus\{i\}} \). Consider one such permutation \( \pi = (\pi_1, ..., \pi_{K-1}) \in \Pi_{[K]\setminus\{i\}} \). Now, using the chain rule and the fact that conditioning does not increase entropy, we can expand the LHS of (35) as

\[
H(\{W_{d_j, R}\}_{j \in [K] \setminus \{i\}}, i \notin R, j \notin R | \{V_S \}_{S \subseteq \emptyset[K]}, W_d, \{W_{d_j, R}\}_{j \in [K] \setminus \{i\}, i \in R \text{ or } j \in R})
\]

\[
= \sum_{m=1}^{K-1} H(\{W_{d_j, R}\}_{j \in [K] \setminus \{i, \pi_1, ..., \pi_m\}, R \subseteq [K] \setminus \{i, j, \pi_1, ..., \pi_{m-1}\}, : \pi_m \in R})
\]

\[
+ H(\{W_{d_j, R}\}_{R \subseteq [K] \setminus \{i, \pi_1, ..., \pi_{K-1}\}, \{V_S \}_{S \subseteq \emptyset[K]}, W_d, \{W_{d_j, R}\}_{j \in [K] \setminus \{i, \pi_m\}, : \pi_m \in R})
\]

\[
= \sum_{m=1}^{K-1} H(\{W_{d_j, R}\}_{j \in [K] \setminus \{i, \pi_1, ..., \pi_m\}, R \subseteq [K] \setminus \{i, j, \pi_1, ..., \pi_{m-1}\}, : \pi_m \in R})
\]

Now, averaging the above bound over all the permutations \( \pi \in \Pi_{[K]\setminus\{i\}} \), we will have

\[
H(\{W_{d_j, R}\}_{j \in [K] \setminus \{i\}, i \notin R, j \notin R | \{V_S \}_{S \subseteq \emptyset[K]}, W_d, \{W_{d_j, R}\}_{j \in [K] \setminus \{i\}, i \in R \text{ or } j \in R})
\]

\[
= \frac{1}{(K-1)!} \sum_{\pi \in \Pi_{[K]\setminus\{i\}}} \sum_{m=1}^{K-1} H(\{W_{d_j, R}\}_{j \in [K] \setminus \{i, \pi_1, ..., \pi_m\}, R \subseteq [K] \setminus \{i, j, \pi_1, ..., \pi_{m-1}\}, : \pi_m \in R})
\]

Now, the entropy of each subfile \( W_{d_j, R} \) appears in the RHS of (76) in the summation for all the permutations \( \pi \in \Pi_{[K]\setminus\{i\}} \) which satisfy at least one of the following conditions:

- \( \pi_1 \in R \).
- \( \pi_2 \in R \text{ and } j \neq \pi_1 \).
- \( \pi_3 \in R \text{ and } j \neq \pi_i, \forall i \in [2] \).
\[ \pi_{K - |R| - 1} \in R \text{ and } j \neq \pi_i, \forall i \in [K - |R| - 2]. \]

It is not hard to verify that the number of permutations that \( \pi \in \Pi_{[K]} \setminus \{i\} \) satisfying at least one of the aforementioned conditions is equal to

\[ \sum_{l=0}^{K - |R| - 2} |R| \Pi(K - |R| - 2, l)(K - l - 2)! = |R|(K - |R| - 2)! \sum_{l=0}^{K - |R| - 2} \frac{(K - l - 2)!}{(K - l - |R| - 2)!} \]  
(77)

\[ = |R|(K - |R| - 2)! |R|! \sum_{l=0}^{K - |R| - 2} \binom{K - l - 2}{|R|} \]  
(78)

\[ = |R|(K - |R| - 2)! |R|! \binom{K - 1}{1 + |R|} \]  
(79)

\[ = (K - 1)! \frac{|R|}{1 + |R|}, \]  
(80)

where (79) is due to the column-sum property of the Pascal triangle. Hence, we can continue (76) as

\[ H(\{W_{d_j, R}\}_{j \in [K] \setminus \{i\}, i \notin R}, \{V_S\}_{S \subseteq [K]}, W_{d_i}, \{W_{d_j, R}\}_{j \in [K] \setminus \{i\}, i \in R \text{ or } j \in R}) = \frac{1}{(K - 1)!} \sum_{j \in [K] \setminus \{i\}} \sum_{R \subseteq [K] \setminus \{i, j\}} (K - 1)! \frac{|R|}{1 + |R|} H(W_{d_j, R}) \]  
(81)

\[ = \sum_{j \in [K] \setminus \{i\}} \sum_{R \subseteq [K] \setminus \{i, j\}} H(W_{d_j, R}). \]  
(82)

**References**

1. M. A. Maddah-Ali and U. Niesen, “Fundamental limits of caching,” *Information Theory, IEEE Transactions on*, vol. 60, no. 5, pp. 2856–2867, 2014.
2. S. P. Shariatpanahi, S. A. Motahari, and B. H. Khalaj, “Multi-server coded caching,” *arXiv preprint arXiv:1503.00265*, 2015.
3. M. Ji, M. F. Wong, A. M. Tulino, J. Llorca, G. Caire, M. Effros, and M. Langberg, “On the fundamental limits of caching in combination networks,” in *IEEE 16th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 695-699, June 2015.
4. L. Tang and A. Ramamoorthy, “Coded caching for networks with the resolvability property,” *arXiv preprint arXiv:1605.01424*, 2016.
5. M. A. Maddah-Ali and U. Niesen, “Decentralized coded caching attains order-optimal memory-rate tradeoff,” *IEEE/ACM Trans. on Networking*, vol. 23, no. 4, pp. 1029-1040, Aug. 2015.
6. Q. Yu, M. A. Maddah-Ali, A. S. Avestimehr, “The Exact Rate-Memory Tradeoff for Caching with Uncoded Prefetching,” *arXiv preprint arXiv:1609.07817*, 2016.