This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

Combination Networks with or without Secrecy Constraints: The Impact of Caching Relays

Ahmed A. Zewail, *Student Member, IEEE,* and Aylin Yener, *Fellow, IEEE*

***Abstract*—This paper considers a two-hop network architecture known as a combination network, where a layer of relay nodes connects a server to a set of end users. In particular, a new model is investigated where the intermediate relays employ caches in addition to the end users. First, a new centralized coded caching scheme is developed that utilizes maximum distance separable (MDS) coding, jointly optimizes cache placement and delivery phase, and enables decomposing the combination network into a set virtual multicast sub-networks. It is shown that if the sum of the memory of an end user and its connected relay nodes is sufficient to store the database, then the server can disengage in the delivery phase and all the end users' requests can be satisfied by the caches in the network. Lower bounds on the normalized delivery load using genie-aided cut-set arguments are presented along with second hop optimality. Next recognizing the information security concerns of coded caching, this new model is studied under three different secrecy settings: 1) secure delivery where we require an external entity must not gain any information about the database files by observing the transmitted signals over the network links, 2) secure caching, where we impose the constraint that end users must not be able to obtain any information about files that they did not request, and 3) both secure delivery and secure caching, simultaneously. We demonstrate how network topology affects the system performance under these secrecy requirements. Finally, we provide numerical results demonstrating the system performance in each of the settings considered.**

***Index Terms*—Combination networks with caching relays, coded caching, maximum distance separable (MDS) codes, secure delivery, secure caching.**

I. Introduction

Caching is foreseen as a promising avenue to provide content based delivery services for 5G systems and beyond [1], [2]. Caching enables shifting the network load from peak to off-peak hours leading to a significant improvement in overall network performance. During off-peak hours, in the *cache placement phase*, the network is likely to have a considerable amount of under-utilized wireless bandwidth which is exploited to place *functions* of data contents in the cache memories of the network nodes. This phase takes place prior to the end users' content requests, and thus content needs to be placed in the caches without knowing what specific content each user will request. The cached contents help reduce the required transmission load when the end users actually request the contents, during peak traffic time, known as the *delivery phase*, not only by alleviating the need to download the entire requested data, but also by facilitating multicast transmissions that benefit multiple end users [3]. As long as the storage capabilities increase, the required transmission load during peak traffic can be decreased, leading to the rate-memory trade-off [3], [4].

Various network topologies with caching capabilities have been investigated to date, see for example [5]–[13]. References [5], [8]–[11] have studied two-hop cache-aided networks. Reference [5] has studied hierarchical networks, where the server is connected to a set of relay nodes via a shared multicast link and the end users are divided into equal-size groups such that each group is connected to only one relay node via a multicast link. Thus, one relay needs to be shared by multiple users. We will not consider this model.

A fundamentally different model is investigated in references [8] and [9] where multiple overlapping relays serve each user. In this symmetric layered network, known as a *combination network* [14], the server is connected to a set of h relay nodes, and each end user is connected to exactly r relay nodes, thus each relay serves $\binom{h-1}{r-1}$ end nodes. In these references, end users randomly cache a fraction of bits from each file subject to the memory capacity constraint. Two delivery strategies have been proposed: one relies on routing the requested bits via the network links and the other is based on coded multicasting and combination network coding techniques [15]. More recently, reference [10] has considered a class of networks which satisfies the resolvability property, which includes combination networks where r divides h [16]. A centralized coded caching scheme has been proposed and shown to outperform, analytically and numerically, those in [8] and [9]. The cache allocation of [10] explicitly utilizes resolvability property, so that one can design the cache contents that make each relay node see the same set of cache allocations. In all of these references studying combination networks -resolvable or not-, only the end users are equipped with cache memories.

In this paper, we boost the caching capabilities of combination networks by introducing caches at the relay nodes. In particular, we consider a general combination network equipped with caches at *both the relay nodes and the end users*. The model in effect enables cooperation between caches from different layers to aid the server. We develop a new centralized coded caching scheme, by utilizing maximum distance separable (MDS) codes [17] and jointly optimizing the cache placement and delivery phases. This proposed construction enables *decomposing* the coded caching in combination networks

Manuscript received December 10, 2017; revised April 7, 2018; accepted April 18, 2018; date of current version April 20, 2018. This work was supported in part by the National Science Foundation Grants CNS 13-14719 and CCF 17-49665. This paper was presented in part at the IEEE International Symposium of Information Theory (ISIT) 2017 and the IEEE Conference on Information Sciences and Systems (CISS) 2018.

A. A. Zewail and A. Yener are with the School of Electrical Engineering and Computer Science, Pennsylvania State University, University Park, PA 16802 USA (e-mail: zewail@psu.edu; yener@ee.psu.edu).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

into sub-problems in the form of the classical setup studied in [3]. We show that if the sum of the memory size of a user and its connected relay nodes is large enough to store the library, then the server can disengage during the delivery phase altogether and all users' requests can be satisfied utilizing the cache memories of the relay nodes and end users. Genie-aided cut-set lower bounds on the transmission rates are provided. Additionally, for the special case, where there are no caches at the relays, we show that our scheme achieves the same performance of the scheme in [10] without requiring resolvability.

In many practical scenarios, reliability is not the only consideration. Confidentiality, especially in file sharing systems, is also of paramount importance. Thus in the latter part of the paper, for the same model, we address the all important concerns of information security. Specifically, we consider combination networks with caches at the relays and end users, under three different scenarios. In the first scenario, we consider that the database files must be kept secret from any external eavesdropper that overhears the delivery phase, i.e., *secure delivery* [18] [19]. In the second scenario, we consider that each user must only be able to decode its requested file and should not be able gain any information about the contents of the remaining files, i.e., *secure caching* [20] [21]. Last, we consider both secure delivery and secure caching, simultaneously. We note that, in security for cache-aided combination networks, the only previous work consists of our recent effort [22], where the schemes are limited to resolvable combination networks with no caching relays.

For all the considered scenarios, our proposed schemes based on the decomposition turn out to be optimal with respect to the total transmission load per relay, i.e., we achieve the cut set bound. Our study demonstrates the impact of cache memories at the relay nodes (in addition to the end users) in reducing the transmission load of the server. In effect, these caches can cooperatively replace the server during the delivery phase under sufficient total memory. Furthermore, we demonstrate the impact of the network topology on the system performance under secrecy requirements. In particular, we demonstrate that satisfying the *secure caching* requirement does not require encryption keys and is feasible even with memory size less than the file size, unlike the case in references [20] and [21]. In addition, we observe that the cost due the *secure delivery* is almost negligible in combination networks, similar to the cases in references [18] and [19] for other network topologies.

The remainder of the paper is organized as follows. Section II describes the system model. In Section III, we propose a new centralized coded caching scheme that is applicable to any cache-aided combination network. In Sections IV, V and VI, we detail the achievability techniques for the three secrecy scenarios. In Section VII, we provide the numerical results and discuss the insights learned from them. Section VIII concludes the paper.

II. System Model

A. Network Model

Consider a combination network, where the server, S, is connected to K end users via a set of h relay nodes. More specifically, each end user is connected to a distinct set of r relay nodes, $r < h$, with $K = \binom{h}{r}$. Each relay node is connected to $\hat{K} = \binom{h-1}{r-1} = \frac{rK}{h}$ end users. Similar to references [8]–[10], all network links are *unicast*. In addition, similar to references [3]–[5], [8]–[10], [18]–[21], all network links are assumed to be noiseless. Let $\mathcal{R} = \{\Gamma_1, .., \Gamma_h\}$ denote the set of relay nodes, and $\mathcal{U} = \{U_1, .., U_K\}$ the set of all end users. We denote the set of end users connected to Γ_j by $\mathcal{N}(\Gamma_j)$, $|\mathcal{N}(\Gamma_j)| = \hat{K}$ for $j = 1, .., h$, and the set relay nodes connected to user k by $\mathcal{N}(U_k)$, $|\mathcal{N}(U_k)| = r$. The function $Index(,) : (j, k) \rightarrow \{1, .., \hat{K}\}$, where $j \in \{1, .., h\}$ and $k \in \mathcal{N}(\Gamma_j)$, is defined as a function that orders the end users connected to relay node Γ_j in an ascending manner. For example, for the network in Fig. 1, $\mathcal{N}(\Gamma_2) = \{1, 5, 6, 7\}$, $\mathcal{N}(\Gamma_4) = \{3, 6, 8, 10\}$, $Index(2, 1) = 1$, $Index(2, 5) = 2$, $Index(2, 6) = 3$, $Index(2, 7) = 4$, $Index(4, 3) = 1$, $Index(4, 6) = 2$, $Index(4, 8) = 3$, and $Index(4, 10) = 4$. For a positive integer, L, we will use the notation $[L] \triangleq \{1, .., L\}$.

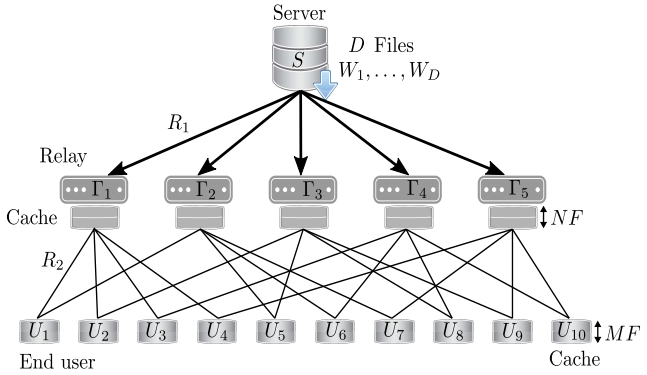

Fig. 1: A combination network with $K = 10$, $h = 5$, $r = 2$, and caches at both relays and end users.

B. Caching Model

Server S has D files, $W_1, .., W_D$, each with size F bits. We treat the case where the number of users is less than or equal to the number of files, i.e., $K \leq D$. Each end user is equipped with a cache memory of size MF bits while each relay node has cache memory of size NF bits, i.e., M and N denote the normalized cache memory sizes at the end users and relay nodes, respectively. The server has the complete knowledge of the network topology. The system operates in two phases.

1) Cache Placement Phase: In this phase, the server allocates functions of its database files in the relay nodes and end users caches. The allocation is done ahead of and without the knowledge of the demand of the individual users.

Definition 1. *(Cache Placement): The content of the cache memories at relay node j and user k, respectively are given by*

$$V_j = \nu_j(W_1, W_2, .., W_D), \qquad Z_k = \phi_k(W_1, W_2, .., W_D), \quad (1)$$

where $\nu_j : [2^F]^D \rightarrow [2^F]^N$ and $\phi_k : [2^F]^D \rightarrow [2^F]^M$, i.e., $H(V_j) \leq NF$ and $H(Z_k) \leq MF$. ■

2) Delivery Phase: Each user requests a file independently and randomly [3]. Let d_k denote the index of the requested file by user k, i.e., $d_k \in \{1, 2, .., D\}$; $\boldsymbol{d}$ represents the demand

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

vector of all users. The server responds to users' requests by transmitting signals to the relay nodes. Then, each relay transmits *unicast* signals to its connected end users. From the r received signals and Z_k, user k must be able to reconstruct its requested file W_{d_k}.

Definition 2. *(Coded Delivery): The mapping from the database files, $\{W_1, .., W_D\}$, and the demand vector $\boldsymbol{d}$ into the transmitted signal by the server to Γ_j is given by the encoding function*

$$X_{j,\boldsymbol{d}} = \psi_j(W_1, .., W_D, \boldsymbol{d}), \qquad i = 1, 2, .., h, \tag{2}$$

where $\psi_j : [2^F]^D \times [D]^K \rightarrow [2^F]^{R_1}$, and R_1 is the rate, normalized by the file size, F, of the transmitted signal from the server to each relay node. The transmitted signal from Γ_j to user $k \in \mathcal{N}(\Gamma_j)$, is given by the encoding function

$$Y_{j,\boldsymbol{d},k} = \varphi_k(X_j, \boldsymbol{d}, V_j, \boldsymbol{d}), \tag{3}$$

where $\varphi_k : [2^F]^{R_1} \times [2^F]^N \times [D]^K \rightarrow [2^F]^{R_2}$, and R_2 is the normalized rate of the transmitted signal from a relay node to a connected end user. User k recovers its requested file by

$$\hat{W}_k = \mu_k(Z_k, \boldsymbol{d}, \{Y_{i,\boldsymbol{d},k} : i \in \mathcal{N}(U_k)\}), \tag{4}$$

where $\mu_k : [2^F]^M \times [D]^K \times [2^F]^{rR_2} \rightarrow [2^F]$ is the decoding function. ■

We require that each end user k recover its requested file reliably, i.e., for any $\epsilon > 0$,

$$\max_{\boldsymbol{d},k} P(\hat{W}_{d_k} \neq W_{d_k}) < \epsilon. \tag{5}$$

Our goal is to develop caching schemes that minimize the worst case delivery load over the two hops. We will characterize the achievable rates over the two hops, R_1 and R_2, under the worst case demand, by jointly designing the cache placement and delivery, i.e., $\{Z_k\}_{k=1}^K, \{V_j\}_{j=1}^h, \{X_{j,\boldsymbol{d}}\}_{j=1}^h$, and $\{Y_{j,\boldsymbol{d},k}\}_{j=1,k\in\mathcal{N}(\Gamma_j)}^h$, subject to the memory constraints, $|Z_k| \leq MF, \forall k$, and $|V_j| \leq NF, \forall j$, while ensuring that each user is able to decode its requested file reliably. In Sections IV-VI, we will require the system to satisfy the secrecy constraints in addition to the reliability constraint. Note that characterizing the achievable rates over both hops results in improving the normalized total network load, $hR_1 + rKR_2$, as discussed in subsection VII-B.

III. A NEW CODED CACHING SCHEME FOR COMBINATION NETWORKS

We develop a new caching scheme for general cache-aided combination networks. In addition, we show that the upper bound derived in [10] for resolvable combination networks, is in fact achievable for all combination networks.

The main idea behind our proposed scheme is that each file is encoded using an (h, r) maximum distance separable (MDS) code [17], [23]. Then, each relay node acts as a server for one of the resulting encoded symbols. Since each end user is connected to r different relay nodes, by the end of the delivery phase, it will be able to obtain r different encoded symbols that can be used to recover its requested file.

A. Cache Placement Phase

As a first step, the server divides each file into r equal-size subfiles. Then, it encodes them using an (h, r) maximum distance separable (MDS) code [17], [23]. We denote by f_n^j the resulting encoded symbol, where n is the file index and $j = 1, 2, .., h$. The size of each encoded symbol, f_n^j, is F/r bits, and any r encoded symbols are sufficient to reconstruct the file n. The server divides each encoded symbol into two parts, $f_n^{j,1}$ and $f_n^{j,2}$, such that the size of $f_n^{j,1}$ is $\frac{NF}{D}$ bits, and the size of $f_n^{j,2}$ is $(\frac{1}{r} - \frac{N}{D})F$ bits.

We describe the achievability for $M = \frac{(t_1-t_2)Nr}{\hat{K}} + \frac{t_2 D}{\hat{K}}$, and $t_1, t_2 \in \{0, 1, .., \hat{K}\}$, noting that the convex envelope of these points is achievable by memory sharing as was shown in reference [3]. First, the server places $f_n^{j,1}, \forall n$ in the cache memory of relay node Γ_j. Then, user k, with $k \in \mathcal{N}(\Gamma_j)$, caches a random fraction of $\frac{t_1}{\hat{K}}$ bits from $f_n^{j,1}, \forall n$, which we denote by $f_{n,k}^{j,1}$. Thus, t_1 represents the fraction cached by each user from the contents stored at the cache of each of its connected relay nodes. On the other hand, $f_n^{j,2}$ is divided into $\binom{\hat{K}}{t_2}$ disjoint pieces each of which is denoted by $f_{n,\mathcal{T}}^{j,2}$, where n is the file index, i.e., $n \in [D]$, j is the index of the encoded symbol, $j = 1, .., h$, and $\mathcal{T} \subseteq [\hat{K}], |\mathcal{T}| = t_2$. The size of each piece is $\frac{(\frac{1}{r} - \frac{N}{D})}{\binom{\hat{K}}{t_2}} F$ bits. Note that the parameter t_2 represents the number of users that shares the same piece of the encoded symbol. The set $\mathcal{T}$ determines the allocation scheme as follows. The server allocates the pieces $f_{n,\mathcal{T}}^{j,2}, \forall n$ in the cache memory of user k if $k \in \mathcal{N}(\Gamma_j)$ and $Index(j, k) \in \mathcal{T}$. Therefore, the cache contents at the relay nodes and end users are given by

$$V_j = \left\{ f_n^{j,1} : \ \forall n \right\}, \tag{6}$$

$$Z_k = \left\{ f_{n,k}^{j,1}, f_{n,\mathcal{T}}^{j,2} : j \in \mathcal{N}(U_k), Index(j, k) \in \mathcal{T}, \forall n \right\}. \tag{7}$$

Clearly, this satisfies the memory constraint at each relay node. Each user caches a fraction of size $\frac{t_1}{\hat{K}}$ from each part of the encoded symbols cached by the connected relays in addition to $r\binom{\hat{K}-1}{t_2-1}$ pieces of size $|f_{n,\mathcal{T}}^{j,2}|$ bits. Thus, the number of the accumulated bits at the cache memory of each end user is given by

$$\begin{aligned} &Dr|f_{n,k}^{j,1}| + Dr\binom{\hat{K}-1}{t_2-1}|f_{n,\mathcal{T}}^{j,2}| \\ &\quad = Dr\frac{N}{D}\frac{t_1}{\hat{K}}F + Dr\frac{(\frac{1}{r} - \frac{N}{D})}{\binom{\hat{K}}{t_2}}F\binom{\hat{K}-1}{t_2-1} \\ &\quad = \frac{Nt_1 r}{\hat{K}}F + \frac{(D - Nr)t_2}{\hat{K}}F = MF, \end{aligned} \tag{8}$$

which satisfies the memory constraint. We summarize the cache placement procedure in Algorithm 1.

B. Coded Delivery Phase

Note that whenever $t_2 = \hat{K}$, each end user and connected relays are capable of caching the entire database file, and there is no transmission needed from the server during the delivery phase. After announcing the demand vector to the network, the server and the relays start to serve the end users' requests. For each relay Γ_j, at each transmission instance, we consider

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

Algorithm 1 Cache placement procedure

Input: $\{W_1, \ldots, W_D\}$
Output: $Z_k, k \in [K]$

for $n \in [D]$ **do**
  Encode each file using an (h, r) MDS code $\rightarrow f_n^j$, $j = 1, .., h$.
  for $j \in [h]$ **do**
    Divide f_n^j into $f_n^{j,1}$ with size $\frac{NF}{D}$ bits and $f_n^{j,2}$ with size $(\frac{1}{r} - \frac{N}{D})F$ bits.
    $V_j \leftarrow f_n^j$
    Partition $f_n^{j,2}$ into equal-size pieces $f_{n,\mathcal{T}}^{j,2}$, $\mathcal{T} \subseteq [\hat{K}]$ and $|\mathcal{T}| = t_2$.
  end for
end for
for $k \in [K]$ **do**
  User k caches a random fraction $\frac{t_1}{\hat{K}}$ bits from $f_n^{j,1}$, $\forall n \rightarrow f_{n,k}^{j,1}$
  $Z_k \leftarrow \bigcup_{j \in \mathcal{N}(U_k)} \bigcup_{n \in [N]} \left\{ f_{n,\mathcal{T}}^{j,2} : Index(j,k) \in \mathcal{T} \right\} \bigcup f_{n,k}^{j,1}$
end for

$\mathcal{S} \subseteq [\hat{K}]$, where $|\mathcal{S}| = t_2 + 1$. For each choice of $\mathcal{S}$, the server transmits to the relay node Γ_j, the signal

$$X_{j,\boldsymbol{d}}^{\mathcal{S}} = \bigoplus_{\{k:k \in \mathcal{N}(\Gamma_j),\ Index(j,k) \in \mathcal{S}\}} f_{d_k, \mathcal{S} \setminus \{Index(j,k)\}}^{j,2}. \quad (9)$$

In total, the server transmits to Γ_j the signal

$$X_{j,\boldsymbol{d}} = \bigcup_{\mathcal{S} \subseteq [\hat{K}]: |\mathcal{S}| = t_2 + 1} \left\{ X_{j,\boldsymbol{d}}^{\mathcal{S}} \right\}. \quad (10)$$

Γ_j forwards the signal $X_{j,\boldsymbol{d}}^{\mathcal{S}}$ to user k whenever $Index(j,k) \in \mathcal{S}$. In addition, Γ_j transmits the missing bits from $f_{d_k}^{j,1}$ to user k, $k \in \mathcal{N}(\Gamma_j)$. The transmitted signal from Γ_j to user k is

$$Y_{j,\boldsymbol{d},k} = \left\{ f_{d_k}^{j,1} \setminus f_{d_k,k}^{j,1} \right\} \bigcup_{\mathcal{S} \subseteq [\hat{K}]: |\mathcal{S}| = t_2 + 1, Index(j,k) \in \mathcal{S}} \{X_{j,\boldsymbol{d}}^{\mathcal{S}}\}. \quad (11)$$

User k can recover $\left\{ f_{d_k,\mathcal{T}}^{j,2} : \mathcal{T} \subseteq [\hat{K}] \setminus \{Index(j,k)\} \right\}$ from the signals received from Γ_j, utilizing its cache's contents. XORing these pieces to the ones already in its cache, i.e., $f_{d_k,\mathcal{T}}^{j,2}$ with $Index(j,k) \in \mathcal{T}$, user k can recover the encoded symbol $f_{d_k}^{j,2}$. Additionally, from its received signal, user k directly gets $f_{d_k}^{j,1}$. Therefore, it can obtain $f_{d_k}^{j}$. Since, user k receives signals from r different relay nodes, it can obtain the encoded symbols $f_{d_k}^{j}$, $\forall j \in \mathcal{N}(U_k)$, and is able to successfully reconstruct its requested file W_{d_k}. The delivery procedure is summarized in Algorithm 2.

C. Rates

First, observe that the server transmits $\binom{\hat{K}}{t_2+1}$ sub-signals to each relay node, each of which has length $\frac{(\frac{1}{r} - \frac{N}{D})}{\binom{\hat{K}}{t_2}} F$ bits, thus the transmission rate in bits from the server to each relay node is

$$R_1 F = \binom{\hat{K}}{t_2+1} \frac{(\frac{1}{r} - \frac{N}{D})}{\binom{\hat{K}}{t_2}} F = \frac{(\hat{K} - t_2)(\frac{1}{r} - \frac{N}{D})}{t_2 + 1} F. \quad (12)$$

Algorithm 2 Delivery procedure

Input: $\boldsymbol{d}$
Output: $X_{j,\boldsymbol{d}}, Y_{j,\boldsymbol{d},k}, j \in [h], k \in [K]$

for $j \in [h]$ **do**
  for $\mathcal{S} \in [\hat{K}], |\mathcal{S}| = t_2 + 1$ **do**
    $X_{j,\boldsymbol{d}}^{\mathcal{S}} \leftarrow \bigoplus_{\{k:k \in \mathcal{N}(\Gamma_j),\ Index(j,k) \in \mathcal{S}\}} f_{d_k, \mathcal{S} \setminus \{Index(j,k)\}}^{j,2}$
  end for
  $X_{j,\boldsymbol{d}} \leftarrow \bigcup_{\mathcal{S} \subseteq [\hat{K}]} \{X_{j,\boldsymbol{d}}^{\mathcal{S}}\}$
  for $k \in \mathcal{N}(\Gamma_j)$ **do**
    $Y_{j,\boldsymbol{d},k} \leftarrow \left\{ f_{d_k}^{j,1} \setminus f_{d_k,k}^{j,1} \right\} \bigcup_{\mathcal{S} \subseteq [\hat{K}]: Index(j,k) \in \mathcal{S}} \{X_{j,\boldsymbol{d}}^{\mathcal{S}}\}$
  end for
end for

During the second hop, each relay node forwards $\binom{\hat{K}-1}{t_2}$ from its received sub-signals to each of its connected end users. Additionally, it sends $(1 - \frac{t_1}{\hat{K}})\frac{N}{D}F$ bits, from its cache memory to each of its connected end users. Therefore, we have

$$\begin{aligned} R_2 F &= \binom{\hat{K}-1}{t_2} \frac{(\frac{1}{r} - \frac{N}{D})}{\binom{\hat{K}}{t_2}} F + (1 - \frac{t_1}{\hat{K}})\frac{N}{D}F \\ &= \frac{(\hat{K} - t_2)(\frac{1}{r} - \frac{N}{D})}{\hat{K}} F + \frac{(\hat{K} - t_1)N}{D\hat{K}} F \\ &= \frac{1}{r}\Big(1 - \frac{t_2}{\hat{K}} - \frac{(t_1 - t_2)Nr}{D\hat{K}}\Big)F = \frac{1}{r}\Big(1 - \frac{M}{D}\Big)F. \end{aligned} \quad (13)$$

Finally, we have the following theorem of the achievable delivery load.

Theorem 1. *The normalized transmission rates, for* $0 \le N \le \frac{D}{r}$, $M = \frac{(t_1 - t_2)Nr}{\hat{K}} + \frac{t_2 D}{\hat{K}}$, $t_1 \in \{0, 1, .., \min(\hat{K}, \lfloor \frac{\hat{K}N}{D} \rfloor\}$, *and* $t_2 \in \{0, 1, .., \hat{K}\}$, *are upper bounded by*

$$R_1 \le \frac{\hat{K} - t_2}{r(t_2 + 1)} \left(1 - \frac{Nr}{D}\right), \quad R_2 \le \frac{1}{r}\left(1 - \frac{M}{D}\right). \quad (14)$$

Furthermore, the convex envelope of these points is achievable. ■

If M is not in the form of $M = \frac{(t_1 - t_2)Nr}{\hat{K}} + \frac{t_2 D}{\hat{K}}$, we use memory sharing as in [3], [5].

Remark 1. *Observe that the caches at the relays help decrease the transmission load only during the first hop,* R_1. *The transmission load over the second hop,* R_2, *depends only on the size of end users' cache memories,* M, *as it is always equal to the complement of the local caching gain divided by the number of relay nodes connected to each end users.* ■

Remark 2. *It can be seen from (12) that when* $t_2 = \hat{K}$, *i.e.,* $M \ge D - Nr$, *we can achieve* $R_1 = 0$. *In other words, whenever* $M + Nr \ge D$, *i.e., the total memory at each end user and its connected relay nodes is sufficient to store the whole file library, the server is not required to transmit during the delivery phase.* ■

When there are no caches at the relays [8], [9], i.e., setting $N = 0$, we obtain the following result.

Corollary 1. *The normalized transmission rates, for* $N = 0$,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

$M = \frac{tD}{\hat{K}}$, and $t \in \{0, 1, .., \hat{K}\}$, are upper bounded by

$$R_1 \leq \frac{\hat{K}}{r}\left(1 - \frac{M}{D}\right)\frac{1}{1 + \frac{\hat{K}M}{D}}, \qquad R_2 \leq \frac{1}{r}\left(1 - \frac{M}{D}\right). \quad (15)$$

In addition, the convex envelope of these points is achievable. ■

Remark 3. *The achievable rates in (15) are the same as the ones in [10] which have been shown to be achievable for a special class of combination networks where r divides h, i.e., resolvable networks. By our scheme, we have just demonstrated that the resolvability property is not necessary to achieve these rates. Furthermore, it has been shown in [10] that, for resolvable networks, these rates outperform the ones in [8] and [9]. Thus, our proposed scheme outperforms the ones in [8] and [9].* ■

Remark 4. *One can see from (15) that the upper bound on R_1 is formed by the product of three terms. The first term $\frac{\hat{K}}{r}$ is due the fact that each relay node is connected to $\hat{K}$ end users, each of which is connected to r relay nodes. Thus, each relay node is responsible for $\frac{1}{r}$ of the load on a server that is connected to $\hat{K}$ end users. The second term $(1 - \frac{M}{D})$ represents the local caching gain at each end user. The term $\frac{1}{1+\frac{\hat{K}M}{D}}$ represents the global caching gain of the proposed scheme.* ■

The merit our proposed scheme is that it allows us to virtually decompose the combination network into a set of sub-networks, each of which in the form of the multicast network [3]. In particular, for the case where $N = 0$, each relay node acts as a virtual server with library of D files each of size F/r bits, while each connected end user dedicates $1/r$ from its memory to this library. Therefore, any scheme developed for the classical multicast setup [3] which achieves rate $R_{\text{Multicast}}(MF/r, D, \hat{K}, F/r)$ can be utilized in the context of combination networks and achieves rate $R_1 = R_{\text{Multicast}}$. In other words, for large enough F, schemes developed for the cases where the users' demands are non-uniform [24], the number of user is greater than the number of files, [25], for small values of the end users memories [26], utilizing coded prefetching [27], can be adopted in a combination network after the decomposition step via MDS coding.

In addition, by applying the proposed decomposition, we can utilize any scheme that is developed for combination networks with no relay caches, $N = 0$, in the case where the relays are equipped with cache memories, i.e., $0 < N \leq \frac{D}{r}$, as indicated in the following proposition.

Proposition 1. *Suppose that the rate pair $R_1^{N=0}(MF, D, K, F)$ and $R_2^{N=0}(MF, D, K, F)$ is achievable in a combination network with no relay caches. Then, for a combination network with relay cache of size NF bits, the rate pair $R_1 = R_1^{N=0}(M_1F, D, K, (1-\frac{Nr}{D})F)$ and $R_2 = R_2^{N=0}(M_1F, D, K, (1-\frac{Nr}{D})F) + (Nr - M_2)\frac{F}{rD}$ is achievable for any choice of $M_1, M_2 \geq 0$ and $M_1 + M_2 \leq M$.* ■

Proof. Split each file of the database, W_n into two subfiles W_n^1 of size $\frac{Nr}{D}F$ bits and W_n^2 of size $(1 - \frac{Nr}{D})F$ bits. Encode each of subfiles $\{W_n^1, \forall n\}$ using an (h, r) MDS code. Each encoded symbol is cached by one of the relays. Divide the cache of each end user into two partitions of sizes M_1F and M_2F such that $M_1 + M_2 \leq M$. The partition of M_1F bits is dedicated to the library formed by the subfiles $\{W_n^2, \forall n\}$, for which we apply any caching scheme that is known for a combination networks with no relay caches. The second partition of size M_2F is filled by bits from the memories of relays connected to the end user as explained in subsection III-A, leading to the achievable pair in the proposition. □

User k	Z_k
1	$\{f^1_{n,123}, f^1_{n,124}, f^1_{n,134}, f^2_{n,123}, f^2_{n,124}, f^2_{n,134} : \forall n\}$
2	$\{f^1_{n,123}, f^1_{n,124}, f^1_{n,234}, f^3_{n,123}, f^3_{n,124}, f^3_{n,134} : \forall n\}$
3	$\{f^1_{n,123}, f^1_{n,134}, f^1_{n,234}, f^4_{n,123}, f^4_{n,124}, f^4_{n,134} : \forall n\}$
4	$\{f^1_{n,124}, f^1_{n,134}, f^1_{n,234}, f^5_{n,123}, f^5_{n,124}, f^5_{n,134} : \forall n\}$
5	$\{f^2_{n,123}, f^2_{n,124}, f^2_{n,234}, f^3_{n,123}, f^3_{n,124}, f^3_{n,234} : \forall n\}$
6	$\{f^2_{n,123}, f^2_{n,134}, f^2_{n,234}, f^4_{n,123}, f^4_{n,124}, f^4_{n,234} : \forall n\}$
7	$\{f^2_{n,124}, f^2_{n,134}, f^2_{n,234}, f^5_{n,123}, f^5_{n,124}, f^5_{n,234} : \forall n\}$
8	$\{f^3_{n,123}, f^3_{n,134}, f^3_{n,234}, f^4_{n,123}, f^4_{n,134}, f^4_{n,234} : \forall n\}$
9	$\{f^3_{n,124}, f^3_{n,134}, f^3_{n,234}, f^5_{n,123}, f^5_{n,134}, f^5_{n,234} : \forall n\}$
10	$\{f^4_{n,124}, f^4_{n,134}, f^4_{n,234}, f^5_{n,124}, f^5_{n,134}, f^5_{n,234} : \forall n\}$

Table I: The cache contents at the end users for $N = K = 10$ and $M = \frac{15}{2}$.

Remark 5. *From Proposition 1, we can observe that caches at the relay nodes help in reducing the delivery load over the first hop. To see this, let $M_1 = M$, the delivery load over the first hop is then scaled by a factor $1 - \frac{Nr}{D}$.* ■

Lastly, we note that if the objective of the system is to minimize the maximum load over the two hops, $\max(R_1, R_2)$, as in [11]–[13], one can optimize over the end user's cache partitioning, M_1 and M_2, in order to minimize the maximum rate over the two hops.

D. An Illustrative Example

We illustrate our proposed scheme by an example. Consider the network depicted in Fig. 1, where $D = 10$, $N = 0$ and $M = \frac{15}{2}$, i.e., $t = 3$. This network is not resolvable.

1) Cache Placement Phase: Each file, W_n, is divided into 2 subfiles. Then, the server encodes them using an $(5, 2)$ MDS code. We denote the resulting encoded symbols by f_n^j, where n is the file index, i.e., $n = 1, .., 10$, and $j = 1, .., 5$. Furthermore, we divide each encoded symbol into 4 pieces each of size $\frac{F}{8}$ bits, and denoted by $f^j_{n,\mathcal{T}}$, where $\mathcal{T} \subseteq [4]$ and $|\mathcal{T}| = 3$. The contents of the cache memories at the end users are given in Table I. Observe that each user stores 6 pieces of the encoded symbols of each file, i.e., $\frac{3}{4}F$ bits, which satisfies the memory constraint.

2) Coded Delivery Phase: Assume that user k requests the file W_k, and $k = 1, .., 10$. The server transmits the following signals

$$X_{1,\boldsymbol{d}} = f^1_{4,123} \oplus f^1_{3,124} \oplus f^1_{2,134} \oplus f^1_{1,234},$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

$$X_{2,\boldsymbol{d}} = f^2_{7,123} \oplus f^2_{6,124} \oplus f^2_{5,134} \oplus f^2_{1,234},$$
$$X_{3,\boldsymbol{d}} = f^3_{9,123} \oplus f^3_{8,124} \oplus f^3_{5,134} \oplus f^3_{2,234},$$
$$X_{4,\boldsymbol{d}} = f^4_{10,123} \oplus f^4_{8,124} \oplus f^4_{6,134} \oplus f^4_{4,234},$$
$$X_{5,\boldsymbol{d}} = f^5_{10,123} \oplus f^5_{9,124} \oplus f^5_{7,134} \oplus f^5_{4,234}.$$

Then, each relay node forwards its received signal to the set of connected users, i.e., $Y_{i,\boldsymbol{d},k} = X_{i,\boldsymbol{d}}$, $\forall k \in \mathcal{N}(\Gamma_i)$. The size of each transmitted signal is equal to the size of a piece of the encoded symbols, i.e., $\frac{1}{8}F$. Thus, $R_1 = R_2 = \frac{1}{8}$. Now, utilizing its memory, user 1 can extract the pieces $f^1_{1,234}$ and $f^2_{1,234}$ from the signals received from relay nodes Γ_1 and Γ_2, respectively. Therefore, user 1 reconstructs f^1_1 and f^2_1, and decodes its requested file W_1. Similarly, user 2 reconstructs f^1_2 and f^3_2, then decodes W_2, and so on for the remaining users.

E. Lower Bounds

Next, we derive genie-aided lower bounds on the delivery load.

1) Lower bound on R_1: Consider a cut that contains l relay nodes, $l \in \{r, .., h\}$, and s end users from the $\binom{l}{r}$ end users who are connected exclusively to these l relay nodes. The remaining end users are served by a genie. Suppose at the first request instance, these s users request the files W_1 to W_s. Then, at the second request instance, they request the files W_{s+1} to W_{2s}, and so on till the request instance $\lfloor D/s \rfloor$. In order to satisfy all users' requests, the total transmission load from the server and the total memory inside the cut must satisfy

$$H(W_1, .., W_{s\lfloor D/s \rfloor}) = s\lfloor D/s \rfloor F \leq \lfloor D/s \rfloor l R_1 F + sMF + lNF. \tag{16}$$

Therefore, we can get

$$R_1 \geq \frac{1}{l}\left(s - \frac{sM + lN}{\lfloor D/s \rfloor}\right). \tag{17}$$

Similar to [8, Appendix B-A], the smallest number of relay nodes serving a set of x users equals to $u = \min(x + r - 1, h)$. Therefore, by the cut set argument, we can get

$$R_1 \geq \frac{1}{u}\left(x - \frac{xM + uN}{\lfloor D/x \rfloor}\right). \tag{18}$$

2) Lower bound on R_2: Consider the cut that contains user k only. Assume D request instances such that at instance i, user k requests the file W_i. Then, we have the following constraint in order to satisfy the user's requests

$$H(W_1, .., W_D) = DF \leq DrR_2 + MF. \tag{19}$$

Therefore, we can get the following bound on R_2

$$R_2 \geq \frac{1}{r}\left(1 - \frac{M}{D}\right). \tag{20}$$

Now, taking into account all possible cuts, we have the following theorem.

Theorem 2. *The normalized transmission rates for $0 < M + rN \leq D$ are lower bounded by*

$$R_1 \geq \max\left(\max_{l \in \{r,..,h\}} \max_{s \in \{1,..,\min(D, \binom{l}{r})\}} \frac{1}{l}\left(s - \frac{sM + lN}{\lfloor D/s \rfloor}\right), \right.$$
$$\left. \max_{x \in \{1,..,\min(D,K)\}} \frac{1}{u}\left(x - \frac{xM + uN}{\lfloor D/x \rfloor}\right)\right) \tag{21}$$

where $u = \min(x + r - 1, h)$, and

$$R_2 \geq \frac{1}{r}\left(1 - \frac{M}{D}\right). \tag{22}$$

■

In the following three sections, we investigate the cache-aided combination network under three different secrecy requirements.

IV. Coded caching with secure delivery

First, we examine the system with *secure delivery*. That is, we require that any external eavesdropper that observes the transmitted signals during the delivery phase, must not gain any information about the files, i.e., for any $\delta > 0$

$$I(\mathcal{X}, \mathcal{Y}; W_1, .., W_D) < \delta, \tag{23}$$

where $\mathcal{X}, \mathcal{Y}$ are the sets of transmitted signals by the server and the relay nodes, respectively.

In order to satisfy (23), we place keys in the network caches during the placement phase. These keys are used to encrypt, i.e., one-time pad [28], the transmitted signals during the delivery phase as in [18] and [19].

A. Cache Placement Phase

We start by providing a scheme for $M = 1 + \frac{t_2(D-1)}{\hat{K}} + \frac{(t_1 - t_2)r(D-1)N}{\hat{K}(D+\hat{K}-t_1)}$, and $t_1, t_2 \in \{0, 1, .., \hat{K}\}$. Other values of M are achievable by memory sharing. First, the server encodes each file using an MDS code to obtain the encoded symbols $\{f^j_n \in [h]\}$. Then, the servers divides each encoded symbol into two parts, $f^{j,1}_n$ with size $\frac{NF}{D+\hat{K}-t_1}$ bits and $f^{j,2}_n$ with size $\frac{F}{r} - \frac{NF}{D+\hat{K}-t_1}$ bits. Second, the server places $f^{j,1}_n$, $\forall n$ in the cache memory of relay node Γ_j. Then, user k, with $k \in \mathcal{N}(\Gamma_j)$, caches a random fraction of $\frac{t_1}{\hat{K}}$ bits from $f^{j,1}_n$, $\forall n$, which we denote by $f^{j,1}_{n,k}$. On the other hand, $f^{j,2}_n$ is divided into $\binom{\hat{K}}{t_2}$ disjoint pieces each of which is denoted by $f^{j,2}_{n,\mathcal{T}}$, where n is the file index, i.e., $n \in [D]$, j is the index of the encoded symbol, $j = 1, .., h$, and $\mathcal{T} \subseteq [\hat{K}], |\mathcal{T}| = t_2$. The size of each piece is $\frac{\frac{F}{r} - \frac{NF}{D+\hat{K}-t_1}}{\binom{\hat{K}}{t_2}} F$ bits. The server allocates the pieces $f^{j,2}_{n,\mathcal{T}}$, $\forall n$ in the cache memory of user k if $k \in \mathcal{N}(\Gamma_j)$ and $Index(j,k) \in \mathcal{T}$.

In addition, the server generates $h\binom{\hat{K}}{t_2+1}$ independent keys. Each key is uniformly distributed with length $\frac{\frac{F}{r} - \frac{NF}{D+\hat{K}-t_1}}{\binom{\hat{K}}{t_2}} F$ bits. We denote each key by $K^j_{\mathcal{T}_K}$, where $j = 1, .., h$, and $\mathcal{T}_K \subseteq [\hat{K}], |\mathcal{T}_K| = t_2 + 1$. User k stores the keys $K^j_{\mathcal{T}_K}$, $\forall j \in \mathcal{N}(U_k)$, whenever $Index(j,k) \in \mathcal{T}_K$. Also, the server generates the random keys K^j_l each of length $\frac{NF(\hat{K}-t_1)}{(D+\hat{K}-t_1)\hat{K}}$ bits, for $j = 1, .., h$ and $l = 1, .., \hat{K}$. K^j_l will be cached by relay j and user k with $Index(j,k) = l$. Therefore, the cache contents at the relay nodes and end users are given by

$$V_j = \left\{f^{j,1}_n, K^j_l : \forall n, l\right\}, \tag{24}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

$$Z_k = \Big\{ f^{j,1}_{n,k}, K^j_l, f^{j,2}_{n,\mathcal{T}}, K^j_{\mathcal{T}_K} : \forall n, \forall j \in \mathcal{N}(U_k), \\ Index(j,k) \in \mathcal{T}, \mathcal{T}_K, Index(j,k) = l \Big\}. \quad (25)$$

The accumulated number of bits cached by each relay is given by

$$\frac{DNF}{D+\hat{K}-t_1} + \frac{\hat{K}NF(\hat{K}-t_1)}{(D+\hat{K}-t_1)\hat{K}} = \frac{DNF+NF\hat{K}-NFt_1}{D+\hat{K}-t_1} = NF. \quad (26)$$

The accumulated of bits at each end user is given by

$$\frac{Dr\binom{\hat{K}-1}{t_2-1}|f^{i,2}_n|}{\binom{\hat{K}}{t_2}} + \frac{r\binom{\hat{K}-1}{t_2}|f^{i,2}_n|}{\binom{\hat{K}}{t_2}} + \frac{Drt_1}{\hat{K}}|f^{i,1}_n| + \frac{r(\hat{K}-t_1)}{\hat{K}}|f^{i,1}_n| \\ = \frac{Drt_2|f^{i,2}_n|}{\hat{K}} + \frac{r(\hat{K}-t_2)|f^{i,2}_n|}{\hat{K}} + \frac{Drt_1|f^{i,1}_n|}{\hat{K}} + \frac{r(\hat{K}-t_1)|f^{i,1}_n|}{\hat{K}} \\ = F + \frac{t_2(D-1)F}{\hat{K}} + \frac{(t_1-t_2)r(D-1)NF}{\hat{K}(D+\hat{K}-t_1)} = MF, \quad (27)$$

thus satisfying the memory constraints on all caches.

B. Coded Delivery Phase

At the beginning of the delivery phase, the demand vector $\boldsymbol{d}$ is announced in the network. For each relay node Γ_j, at each transmission instance, we consider $\mathcal{S} \subseteq [\hat{K}]$, where $|\mathcal{S}| = t_2 + 1$. For each $\mathcal{S}$, the server sends to the relay node Γ_j, the signal

$$X^{\mathcal{S}}_{j,\boldsymbol{d}} = K^j_{\mathcal{S}} \bigoplus_{\{k:k\in\mathcal{N}(\Gamma_j),\ Index(j,k)\in\mathcal{S}\}} f^j_{d_k,\mathcal{S}\setminus\{Index(j,k)\}}. \quad (28)$$

In total, the server transmits to Γ_j, the signal $X_{j,\boldsymbol{d}} = \cup_{\mathcal{S}\subseteq[\hat{K}]:|\mathcal{S}|=t_2+1}\{X^{\mathcal{S}}_{j,\boldsymbol{d}}\}$.

Then, Γ_j forwards the signal $X^{\mathcal{S}}_{j,\boldsymbol{d}}$ to user k whenever $Index(j,k) \in \mathcal{S}$. In addition, the relay Γ_j sends $f^{j,1}_{d_k} \setminus f^{j,1}_{d_k,k}$ to user k encrypted by the key K^j_l such that $Index(j,k) = l$, i.e., we have

$$Y_{j,\boldsymbol{d},k} = \Big\{K^j_l \oplus \{f^{j,1}_{d_k} \setminus f^{j,1}_{d_k,k}\}\Big\} \bigcup_{\mathcal{S}\subseteq[\hat{K}]:|\mathcal{S}|=t_2+1, Index(j,k)\in\mathcal{S}} \{X^{\mathcal{S}}_{j,\boldsymbol{d}}\}. \quad (29)$$

First, user k can decrypt its received signals using the cached keys. Then, it can recover the pieces $\Big\{f^{j,2}_{d_k,\mathcal{T}} : \mathcal{T} \subseteq [\hat{K}] \setminus \{Index(j,k)\}\Big\}$ from the signals received from Γ_j, utilizing its cache's contents. With its cached contents, user k can recover $f^{j,2}_{d_k}$. In addition, user k directly gets $f^{j,1}_{d_k}$ from its the signal transmitted by relay j. Thus, it can obtain $f^j_{d_k}$. Since, user k receives signals from r different relay nodes, it can obtain the encoded symbols $f^j_{d_k}$, $\forall j \in \mathcal{N}(U_k)$, and is able to successfully reconstruct its requested file W_{d_k}.

Remark 6. *In total, the server sends $h\binom{\hat{K}}{t_2+1}$ signals, each of which is encrypted using a one-time pad that has length equal to the length of each subfile ensuring prefect secrecy [28]. Observing any of the transmitted signals without knowing the encryption key will not reveal any information about the database files [28]. The same applies for the messages transmitted by the relays. Thus, (23) is satisfied.* ■

C. Secure Delivery Rates

Denote the secure delivery rates in the first and second hop with R^s_1 and R^s_2, respectively. Each relay node is responsible for $\binom{\hat{K}}{t_2+1}$ transmissions, each of length $\frac{|f^{i,2}_n|}{\binom{\hat{K}}{t_2}}$, thus the transmission rate in bits from the server to each relay node is

$$R^s_1 F = \frac{\binom{\hat{K}}{t_2+1}}{\binom{\hat{K}}{t_2}}|f^{i,2}_n| = \frac{\hat{K}-t_2}{(t_2+1)}\left(\frac{F}{r} - \frac{NF}{D+\hat{K}-t_1}\right). \quad (30)$$

Γ_j forwards $\binom{\hat{K}-1}{t_2}$ from its received signals to each connected end users. In addition, it transmits a message of size $\frac{NF(\hat{K}-t_1)}{(D+\hat{K}-t_1)\hat{K}}$ bits from its cached contents to each user, thus we have

$$R^s_2 F = \frac{\binom{\hat{K}-1}{t_2}}{\binom{\hat{K}}{t_2}}|f^{i,2}_n| + \frac{NF(\hat{K}-t_1)}{(D+\hat{K}-t_1)\hat{K}} \\ = \left(1-\frac{t_2}{\hat{K}}\right)\left(\frac{F}{r} - \frac{NF}{D+\hat{K}-t_1}\right) + \frac{NF(\hat{K}-t_1)}{(D+\hat{K}-t_1)\hat{K}} \\ = \frac{F}{r}\left(1-\frac{M-1}{N-1}\right). \quad (31)$$

Finally, we can express our results in following theorem.

Theorem 3. *The normalized transmission rates with secure delivery, for $N \geq 0$, $M = 1 + \frac{t_2(D-1)}{\hat{K}} + \frac{(t_1-t_2)r(D-1)N}{\hat{K}(D+\hat{K}-t_1)}$, $t_1, t_2 \in \{0,1,..,\hat{K}\}$ and $\frac{t_1}{\hat{K}} \leq \frac{N}{D+\hat{K}-t_1}$, are upper bounded by*

$$R^s_1 \leq \frac{\hat{K}-t_2}{r(t_2+1)}\left(1-\frac{Nr}{D+\hat{K}-t_1}\right), \quad R^s_2 \leq \frac{1}{r}\left(1-\frac{M-1}{D-1}\right). \quad (32)$$

In addition, the convex envelope of these points is achievable by memory sharing. ■

For the special case of no caches at the relays, i.e., $N = 0$, we obtain the following upper bound on the secure delivery rates.

Corollary 2. *The normalized transmission rates with secure delivery, for $N = 0$, $M = 1 + \frac{t(D-1)}{\hat{K}}$, and $t \in \{0,1,..,\hat{K}\}$, are upper bounded by*

$$R^s_1 \leq \frac{\hat{K}\left(1-\frac{M-1}{D-1}\right)}{r\left(\hat{K}\frac{M-1}{D-1}+1\right)}, \qquad R^s_2 \leq \frac{1}{r}\left(1-\frac{M-1}{D-1}\right). \quad (33)$$

In addition, the convex envelope of these points is achievable by memory sharing. ■

Remark 7. *Under secure delivery, we place keys in the memories of both end user and relays, i.e., we divide the cache between storing data and keys. Observe that the rate of the second hop is the complement of the data caching gain of end user and is determined by M only. In addition, whenever $M \geq D$, each user can cache the entire library and there is no need for caching keys as $R^s_1 = R^s_2 = 0$. On the other hand, the rate of the first hop R^s_1 is affected by both*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

M and N. We achieve zero rate over the first hop whenever $M \geq D - \frac{(D-1)Nr}{\hat{K}+D}$. ■

V. Combination networks with secure caching

Next, we consider *secure caching*, i.e., an end user must be able to recover its requested file, and must *not* be able to obtain any information about the remaining files, i.e., for $\delta > 0$

$$\max_{\boldsymbol{d},k} I(\boldsymbol{W}_{-d_k}; \{Y_{j,\boldsymbol{d},k} : j \in \mathcal{N}(U_k)\}, Z_k) < \delta, \tag{34}$$

where $\boldsymbol{W}_{-d_k} = \{W_1, .., W_N\} \backslash \{W_{d_k}\}$, i.e., the set of all files except the one requested by user k.

In our achievability, we utilize secret sharing schemes [29] to ensure that no user is able to obtain information about the files from its cached contents. The basic idea of the secret sharing schemes is to encode the secret in such a way that accessing a subset of shares does not suffice to reduce the uncertainty about the secret. For instance, if the secret is encoded into the scaling coefficient of a line equation, the knowledge of one point on the line does not reveal any information about the secret as there remain infinite number of possibilities to describe the line. One can learn the secret only if two points on the line are provided.

In particular, we use a class of secret sharing scheme known as *non-perfect secret sharing schemes*, defined as follows.

Definition 3. *[29] [30] For a secret W with size F bits, an* (m, n) *non-perfect secret sharing scheme generates n shares,* $S_1, S_2, ..S_n$*, such that accessing any m shares does not reveal any information about the file W, i.e.,*

$$I(W; \mathcal{S}) = 0, \quad \forall \mathcal{S} \subseteq \{S_1, S_2, ..S_n\}, |\mathcal{S}| \leq m. \tag{35}$$

Furthermore, W can be losslessly reconstructed from the n shares, i.e.,

$$H(W|S_1, S_2, .., S_n) = 0. \tag{36}$$

■

For large enough F, an (m, n) secret sharing scheme exists with shares of size equal to $\frac{F}{n-m}$ bits [29], [30].

A. Cache Placement Phase

Again, as a first step, the server divides each file into r equal-size subfiles. Then, it encodes them using an (h, r) maximum distance separable (MDS) code. We denote by f_n^j the resulting encoded symbol, where n is the file index and $j = 1, 2, .., h$. For $M = \frac{tD}{\hat{K}-t}(1 - \frac{Nr}{D})$ and $t \in \{0, 1, .., \hat{K}-1\}$, we divide each encoded symbol into two parts, $f_n^{j,1}$ with size $\frac{NF}{D}$ bits and $f_n^{j,2}$ with size $\frac{F}{r} - \frac{NF}{D}$ bits. The parts $\{f_n^{j,1} : \forall n\}$ will be cached in the memory of relay Γ_j and will not be cached by any user.

The parts to be cached by the end users are encoded using an (m, n) non-perfect secret sharing scheme. We choose the parameters of the scheme as follows. Parameter n is the total number of the resulting shares for each encoded symbol and is chosen to ensure the ability of each end user to reconstruct its requested encoded symbol from its n shares by the end of the delivery phase. Parameter m is chosen to be the number of shares of each encoded symbol to be cached by the user. This choice ensures that no user is able to obtain any information about the database files from its cache contents only which is essential to satisfy the secure caching constraint. In particular, we encode each of the symbols $f_n^{j,2}$ using $\left(\binom{\hat{K}-1}{t-1}, \binom{\hat{K}}{t}\right)$ secret sharing scheme from [29], [30]. The resulting shares are denoted by $S^j_{n,\mathcal{T}}$, where n is the file index i.e., $n \in \{1, .., N\}$, j is the index of the encoded symbol, i.e., $j = 1, .., h$, and $\mathcal{T} \subseteq [\hat{K}], |\mathcal{T}| = t$. Each share has size

$$F_s = \frac{\frac{F}{r} - \frac{NF}{D}}{\binom{\hat{K}}{t} - \binom{\hat{K}-1}{t-1}} = \frac{t\left(1 - \frac{Nr}{D}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}} F \text{ bits.} \tag{37}$$

The server allocates the shares $S^j_{n,\mathcal{T}}, \forall n$ in the cache of user k whenever $j \in \mathcal{N}(U_k)$ and $Index(j,k) \in \mathcal{T}$. Therefore, at the end of cache placement phase, the contents of the cache memory at relay j and user k are given by

$$V_j = \left\{ f_n^{j,1} : \forall n \right\}, \tag{38}$$

$$Z_k = \left\{ S^j_{n,\mathcal{T}} : k \in \mathcal{N}(\Gamma_j),\ Index(j,k) \in \mathcal{T},\ \forall n \right\}. \tag{39}$$

Remark 8. *Each user stores* $Dr\binom{\hat{K}-1}{t-1}$ *shares, thus the accumulated number of bits stored in each cache memory is*

$$Dr\binom{\hat{K}-1}{t-1} \frac{t\left(1 - \frac{Nr}{D}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}} F = \frac{tD}{\hat{K}-t}\left(1 - \frac{Nr}{D}\right) F = MF. \tag{40}$$

Clearly, the proposed scheme satisfies the cache capacity constraint at both relays and end users. Furthermore, from (40), we can get $t = \frac{\hat{K}M}{D+M-Nr}$. ■

B. Coded Delivery Phase

At the beginning of the delivery phase, each user requests a file from the server. First, we focus on the transmissions from the server to Γ_j. At each transmission instance, we consider $\mathcal{S} \subseteq [\hat{K}]$, where $|\mathcal{S}| = t + 1$. For each $\mathcal{S}$, the server transmits the following signal to Γ_j

$$X^{\mathcal{S}}_{j,\boldsymbol{d}} = \bigoplus_{\{k : k \in \mathcal{N}(\Gamma_j),\ Index(j,k) \in \mathcal{S}\}} S^j_{d_k, \mathcal{S} \backslash \{Index(j,k)\}}. \tag{41}$$

In total, the server transmits to Γ_j, the signal $X_{j,\boldsymbol{d}} = \bigcup_{\mathcal{S} \subseteq [\hat{K}] : |\mathcal{S}| = t+1} \{X^{\mathcal{S}}_{j,\boldsymbol{d}}\}$. Then, Γ_j forwards the signal $X^{\mathcal{S}}_{j,\boldsymbol{d}}$ to user k whenever $Index(j,k) \in \mathcal{S}$. In addition, Γ_j sends directly $f^{j,1}_{d_k}$ to user k. Therefore, we have

$$Y_{j,\boldsymbol{d},k} = \{f^{j,1}_{d_k}\} \bigcup_{\mathcal{S} \subseteq [\hat{K}] : |\mathcal{S}| = t+1, Index(j,k) \in \mathcal{S}} \{X^{\mathcal{S}}_{j,\boldsymbol{d}}\}. \tag{42}$$

User k can recover $\{S^j_{d_k,\mathcal{T}} : \mathcal{T} \subseteq [\hat{K}] \setminus \{Index(j,k)\}, |\mathcal{T}| = t\}$ from the signals received from Γ_j, utilizing its cache's contents. Adding these shares to the ones in its cache, i.e., $S^j_{d_k,\mathcal{T}}$ with $Index(j,k) \in \mathcal{T}$, user k can decode the encoded symbol $f^{j,2}_{d_k}$ from its $\binom{\hat{K}}{t}$ shares. Since, user k receives signals from r different relay nodes, it obtains the encoded symbols $f^j_{d_k}, \forall j \in \mathcal{N}(U_k)$, and can reconstruct W_{d_k}.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

C. Secure Caching Rates

Under secure caching requirement, we denote the first and second hop rates as R_1^c and R_2^c, respectively. Since, each relay node is responsible for $\binom{\hat{K}}{t+1}$ transmissions, each of length F_s, the transmission rate, in bits, from the server to each relay node is

$$R_1^c F = \frac{t\binom{\hat{K}}{t+1}\left(1-\frac{Nr}{D}\right)F}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}} = \frac{\hat{K}\left(1-\frac{Nr}{D}\right)F}{r(t+1)} = \frac{\hat{K}(D+M-rN)}{r\left((\hat{K}+1)M+D-rN\right)}\left(1-\frac{Nr}{D}\right)F. \quad (43)$$

Then, each relay forwards $\binom{\hat{K}-1}{t}$ from these signals to each of its connected end users. In addition, each relay forwards $\frac{NF}{D}$ bits from its cache to each of these users, therefore

$$R_2^c F = \binom{\hat{K}-1}{t}\frac{t\left(1-\frac{Nr}{D}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}}F + \frac{NF}{D} = \frac{1}{r}F. \quad (44)$$

Consequently, we have the following theorem.

Theorem 4. *The normalized rates with secure caching, for $0 \le N \le \frac{D}{r}$, $M=\frac{tD}{\hat{K}-t}(1-\frac{Nr}{D})$, and $t \in \{0,1,..,\hat{K}-1\}$, are upper bounded by*

$$R_1^c \le \frac{\hat{K}(D+M-rN)}{r\left((\hat{K}+1)M+D-rN\right)}\left(1-\frac{Nr}{D}\right), \qquad R_2^c \le \frac{1}{r}. \quad (45)$$

The convex envelope of these points is achievable by memory sharing. ■

Remark 9. *Secret sharing encoding guarantees that no user is able to reconstruct any file from its cache contents only, as the cached shares are not sufficient to reveal any information about any file. In addition, the only new information in the received signals by any end user is the shares related to its requested file. Thus, (34) is satisfied.* ■

For the special case of no relay caches, we obtain the following corollary.

Corollary 3. *The normalized rates with secure caching, for $N=0$, $M=\frac{tD}{\hat{K}-t}$, and $t \in \{0,1,..,\hat{K}-1\}$, are upper bounded by*

$$R_1^c \le \frac{\hat{K}(D+M)}{r\left((\hat{K}+1)M+D\right)}, \qquad R_2^c \le \frac{1}{r}. \quad (46)$$

The convex envelope of these points is achievable by memory sharing. ■

Remark 10. *R_2^c is optimal, as it coincides with the cut-set bound.* ■

To see why Remark 10 holds, consider two request instances where user 1 requests the files W_1 and W_2 respectively. Let Z_1 be the cached contents by user 1 and $\mathcal{Y}_i$ be the transmitted signals by the connected relays to user 1 at request instance i. We have the following

$$F = H(W_2) = I(W_2;\mathcal{Y}_1,\mathcal{Y}_2,Z_1) + H(W_2|\mathcal{Y}_1,\mathcal{Y}_2,Z_1) \quad (47)$$
$$\le I(W_2;\mathcal{Y}_1,\mathcal{Y}_2,Z_1) + \epsilon \quad (48)$$
$$= I(W_2;\mathcal{Y}_1,Z_1) + I(W_2;\mathcal{Y}_2|\mathcal{Y}_1,Z_1) + \epsilon \quad (49)$$
$$\le I(W_2;\mathcal{Y}_2|\mathcal{Y}_1,Z_1) + \epsilon + \delta \quad (50)$$
$$\le H(\mathcal{Y}_2) + \epsilon + \delta \quad (51)$$
$$= rR_2^c F + \epsilon + \delta. \quad (52)$$

(48) and (50) follow from the decodability and secure caching constraints, respectively. By taking ϵ and δ arbitrary close to zero, we get $R_2^c \ge \frac{1}{r}$. Similar spirited results can be found in [20] and [21] for multicast and device-to-device networks, respectively.

VI. Combination networks with secure caching and secure delivery

Now, we investigate the network under the requirements studied in Sections IV and V, simultaneously. The achievability scheme utilizes both one-time pads and secret sharing.

A. Cache Placement Phase

For $M = 1 + \frac{tD}{\hat{K}-t}(1-\frac{rN}{D+\hat{K}})$, and $t \in \{0,1,..,\hat{K}-1\}$, after encoding each file using an (h,r) MDS code, we divide each encoded symbol into two parts, $f_n^{j,1}$ with size $\frac{NF}{D+\hat{K}}$ bits and $f_n^{j,2}$ with size $\frac{F}{r} - \frac{NF}{D+\hat{K}}$ bits. Only Γ_j caches the parts $\{f_n^{j,1} : \forall n\}$.

Each of the symbols $f_n^{j,2}$ is encoded using a $\left(\binom{\hat{K}-1}{t-1},\binom{\hat{K}}{t}\right)$ secret sharing scheme from [29], [30]. The resulting shares are denoted by $S_{n,\mathcal{T}}^j$, where n is the file index i.e., $n \in \{1,..,N\}$, j is the index of the encoded symbol, i.e., $j = 1,..,h$, and $\mathcal{T} \subseteq [\hat{K}], |\mathcal{T}| = t$. Each share has size

$$F_s = \frac{\frac{F}{r} - \frac{NF}{D+\hat{K}}}{\binom{\hat{K}}{t} - \binom{\hat{K}-1}{t-1}} = \frac{t\left(1-\frac{Nr}{D+\hat{K}}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}}F \text{ bits.} \quad (53)$$

The server allocates the shares $S_{n,\mathcal{T}}^j$, $\forall n$ in the cache of user k whenever $j \in \mathcal{N}(U_k)$ and $Index(j,k) \in \mathcal{T}$.

Furthermore, the server generates $h\binom{\hat{K}}{t+1}$ independent keys. Each key is uniformly distributed with length F_s bits. We denote each key by $K_{\mathcal{T}_K}^j$, where $j = 1,..,h$, and $\mathcal{T}_K \subseteq [\hat{K}], |\mathcal{T}_K| = t+1$. User k stores the keys $K_{\mathcal{T}_K}^j$, $\forall j \in \mathcal{N}(U_k)$, whenever $Index(j,k) \in \mathcal{T}_K$. Also, the server generates the random keys K_l^j each of length $\frac{NF}{D+\hat{K}}$ bits, for $j = 1,..,h$ and $l = 1,..,\hat{K}$, which will be cached by relay j and user k with $Index(j,k) = l$.

Therefore, at the end of cache placement phase, the contents of the cache memory at relay j and user k are given by

$$V_j = \left\{f_n^{j,1}, K_l^j : \forall n, l\right\}, \quad (54)$$

$$Z_k = \Big\{S_{n,\mathcal{T}}^j, K_{\mathcal{T}_K}^j, K_l^j : \forall n, \forall j \in \mathcal{N}(U_k), Index(j,k) \in \mathcal{T}, \mathcal{T}_K, Index(j,k) = l\Big\}. \quad (55)$$

Remark 11. *In addition to the keys, each user stores $Dr\binom{\hat{K}-1}{t-1}$ shares, thus the accumulated number of bits stored in each cache memory is*

$$\frac{Dr\binom{\hat{K}-1}{t-1}t\left(1-\frac{Nr}{D+\hat{K}}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}}F + \frac{r\binom{\hat{K}-1}{t}t\left(1-\frac{Nr}{D+\hat{K}}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}}F + \frac{rNF}{D+\hat{K}}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

$$= \frac{Dt\left(1-\frac{Nr}{D+\hat{K}}\right)}{r(\hat{K}-t)}F + \left(1-\frac{Nr}{D+\hat{K}}\right)F + \frac{rNF}{D+\hat{K}} = MF. \quad (56)$$

Thus, the scheme satisfies the memory constraints, and we get $t = \frac{\hat{K}(M-1)(D+\hat{K})}{(D+\hat{K})(M+D-1)+rND}$. ■

B. Coded Delivery Phase

The delivery phase begins with announcing the demand vector to all network nodes. For Γ_j, at each transmission instance, we consider a $\mathcal{S} \subseteq [\hat{K}]$, where $|\mathcal{S}| = t+1$. For each $\mathcal{S}$, the server transmits to Γ_j, the following signal

$$X^{\mathcal{S}}_{j,\boldsymbol{d}} = K^j_{\mathcal{S}} \bigoplus_{\{k:k\in\mathcal{N}(\Gamma_j),\ Index(j,k)\in\mathcal{S}\}} S^j_{d_k,\mathcal{S}\setminus\{Index(j,k)\}}, \quad (57)$$

i.e., the server transmits to Γ_j, the signal $X_{j,\boldsymbol{d}} = \bigcup_{\mathcal{S}\subseteq[\hat{K}]:|\mathcal{S}|=t+1}\{X^{\mathcal{S}}_{j,\boldsymbol{d}}\}$. Then, Γ_j forwards the signal $X^{\mathcal{S}}_{j,\boldsymbol{d}}$ to user k whenever $Index(j,k) \in \mathcal{S}$. In addition, Γ_j sends $f^{j,1}_{d_k}$ encrypted by K^j_l to user k such that $Index(j,k) = l$. After decrypting the received signals, user k get $f^{j,1}_{d_k}$ and can extract the set of shares $\{S^j_{d_k,\mathcal{T}} : \mathcal{T} \subseteq [\hat{K}] \setminus \{Index(j,k)\}, |\mathcal{T}| = t\}$ from the signals received from Γ_j. These shares in addition to the ones in its cache, i.e., $S^j_{d_k,\mathcal{T}}$ with $Index(j,k) \in \mathcal{T}$, allow user k to decode $f^{j,2}_{d_k}$ from its $\binom{\hat{K}}{t}$ shares. Since, user k receives signals from r different relay nodes, it obtains $\{f^j_{d_k}, \forall j \in \mathcal{N}(U_k)\}$, then decodes W_{d_k}.

C. Secure Caching and Secure Delivery Rates

We refer to the first and second hop rates as R^{sc}_1 and R^{sc}_2, respectively. Each relay node sends $\binom{\hat{K}}{t+1}$ signals, each of length F_s, thus we have

$$R^{sc}_1 F = \binom{\hat{K}}{t+1}\frac{t\left(1-\frac{Nr}{D+\hat{K}}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}}F = \frac{\hat{K}}{r(t+1)}\left(1-\frac{Nr}{D+\hat{K}}\right)F$$
$$= \frac{\hat{K}\left(rND+(D+\hat{K})(M+D-1)\right)\left(1-\frac{Nr}{D+\hat{K}}\right)F}{r\left(rND+(D+\hat{K})[D+(M-1)(\hat{K}+1)]\right)}. \quad (58)$$

In the second hop, each relay node is responsible for forwarding $\binom{\hat{K}-1}{t}$ from its received signals to each of its connected end users, in addition, it transmits $\frac{NF}{D+\hat{K}}$ bits from its cache, thus

$$R^{sc}_2 F = \binom{\hat{K}-1}{t}\frac{t\left(1-\frac{Nr}{D+\hat{K}}\right)}{r(\hat{K}-t)\binom{\hat{K}-1}{t-1}}F + \frac{NF}{D+\hat{K}} = \frac{1}{r}F. \quad (59)$$

Therefore, we can obtain the following theorem.

Theorem 5. *Under secure delivery and secure caching requirements, for* $0 \le N \le \frac{D+\hat{K}}{r}$, $M = 1 + \frac{tD}{\hat{K}-t}(1-\frac{rN}{D+\hat{K}})$, *and* $t \in \{0,1,..,\hat{K}-1\}$, *the transmission rates are upper bounded by*

$$R^{sc}_1 \le \frac{\hat{K}\left(rND+(D+\hat{K})(M+D-1)\right)\left(1-\frac{Nr}{D+\hat{K}}\right)}{r\left(rND+(D+\hat{K})[D+(M-1)(\hat{K}+1)]\right)}, \quad (60)$$

$$R^{sc}_2 \le \frac{1}{r}. \quad (61)$$

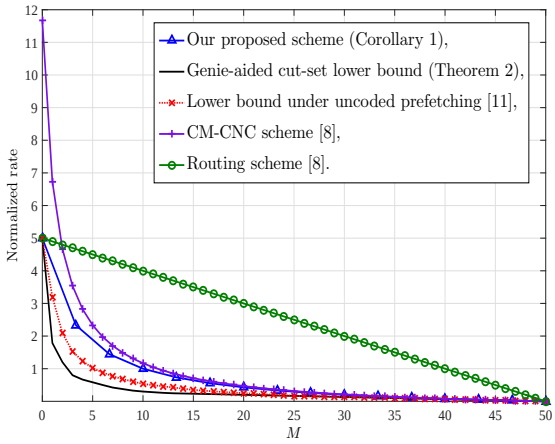

Fig. 2: Lower and upper bounds for $K = 35$, $N = 0$, $D = 50$, $h = 7$ and $r = 3$.

In addition, the convex envelope of these points is achievable by memory sharing. ■

For the case where there is no caches at the relays, we have

Corollary 4. *Under secure delivery and secure caching requirements, for* $N = 0$, $M = \frac{tD}{\hat{K}-t}+1$, *and* $t \in \{0,1,..,\hat{K}-1\}$, *the transmission rates are upper bounded by*

$$R^{sc}_1 \le \frac{\hat{K}(D+M-1)}{r\left((\hat{K}+1)(M-1)+D\right)}, \qquad R^{sc}_2 \le \frac{1}{r}. \quad (62)$$

In addition, the convex envelope of these points is achievable by memory sharing. ■

VII. Numerical Results and Discussion

In this section, we discuss the insights gained from our study and demonstrate the performance of our proposed techniques. We focus on the achievable rates over the links of each hop of communication.

A. Achievable Rates over the First Hop

Fig. 2 shows the comparison between the achievable normalized rate of our proposed scheme in Corollary 1 (the special case with no caching at the relays), lower bound in Theorem 2, lower bound under uncoded prefetching [11], the coded multicasting and combination network coding (CM-CNC) scheme [8], and the routing scheme from [8]. We can see that our proposed scheme outperforms the ones in [8]. We remark that in this special case of no caching relays, the lower bounds in subsection III-E1 reduce to the ones in [8]. Therefore, the same order optimality as in [8, Theorem 4] applies.

In Fig. 3, we plot the normalized rate for different relay cache sizes. It can be observed that the normalized rates are decreasing functions of the memory capacities and whenever $M + rN \ge 60$, $R_1 = 0$, while $R_2 = 0$ if $M \ge 60$. This shows how the cache memories at the relay nodes as well as the ones at the end users can completely replace the main server during the delivery phase. We note that the gap between lower and upper bound decreases as M increases.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

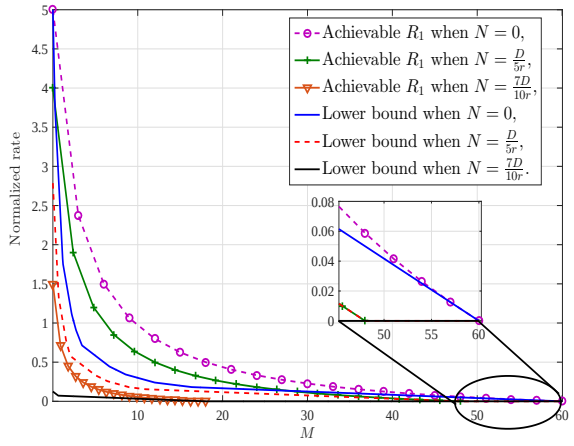

Fig. 3: Lower and upper bounds for $K=35$, $D=60$, $h=7$ and $r=4$.

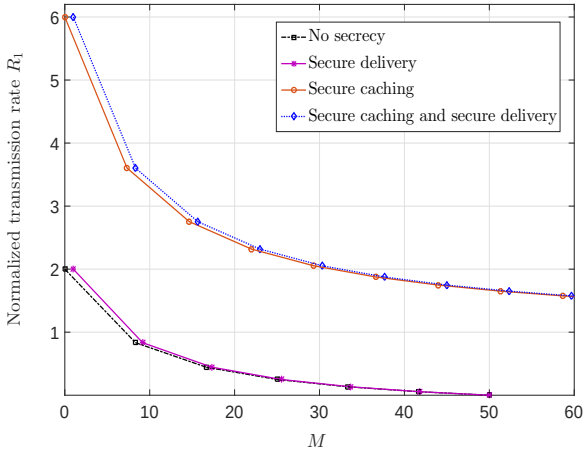

Fig. 4: Rates over the first hop under different system requirements for $N=0$, $D=50$, $K=15$, $h=5$ and $r=3$.

B. Optimality over the Second Hop

From the cut set bound in Theorem 2, we see that our achievable rate over the second hop is optimal. Thus, the total delivery load per relay is minimized.

For the network with no caches at the relays, it has been shown in [11] that the proposed schemes achieve lower rates over the first hop compared with the scheme in [10], i.e., achieves lower R_1 than our scheme in Section III, for the case where $M=N/K$. Additionally, it has been shown that the schemes in [11] achieve the optimal rate under uncoded prefetching for $r=h-1$. Note that the scheme based on interference alignment in [11] for the case where $M=D/K$ achieves lower rates over the first hop, however the achievable rate during the second hop is not optimal. As an example consider the network with $D=K=6$, $h=4$, $r=2$ and $M=1$, the normalized optimal delivery load during the second hop is $\frac{5}{12}$ and it is achievable by our scheme. The scheme in [11, Section IV-B] achieves normalized delivery load of $\frac{7}{12}$. On the other hand, in this example, the scheme in [11, Section IV-B] achieves $R_1=\frac{2}{3}$, while our scheme achieves 1. Therefore, the total normalized network load, i.e., hR_1+rKR_2, under the scheme in [11, Section IV-B] is $\frac{29}{3}$, while our scheme achieves 9. This example demonstrates to the importance of ensuring the optimality over the second hop in order to reduce the overall network load.

C. Performance with Secrecy Requirements

In Figs. 4 and 5, we compare the achievable rates under different secrecy scenarios. From these figures, we observe that the cost of imposing secure delivery is *negligible* for realistic system parameters. The gap between the achievable rates of the system without secrecy and the system with secure delivery vanishes as M increases. Same observation holds for the gap between the rates with secure caching and those with secure caching and secure delivery.

In addition, achievable rates over the second hop is optimal, i.e., achieves the cut set bound. In particular, under secure delivery, each user caches a fraction $\frac{M-1}{N-1}$ of each file, and the total data received by any end user under secure delivery equals $(1-\frac{M-1}{N-1})F$, which is the minimum number of bits required to reconstruct the requested file. Similarly, in the two remaining scenarios, we know from the result in reference [20] that the minimum number of bits required by each user to be able to recover its requested file is F, and our achievable schemes achieve this lower bound. Another observation is that under secure caching requirement only (Section V), we do not need to use keys in order to ensure the secure caching requirement, in contrast with the general schemes in references [20] and [21]. This follows from the network structure, as the relay nodes unicast the signals to each of the end users. In particular, the received signals by user k are formed by combinations of the shares in its memory and "fresh" shares of the requested file. Thus, at the end of communications, it has $r\binom{\hat{K}}{t}$ shares of the file W_{d_k}, and only $r\binom{\hat{K}-1}{t-1}$ shares of the remaining files, i.e., the secure caching requirement is satisfied, without the need to encrypt. In addition, for the case where $M=0$, i.e., no cache memory at the end users, secure caching is possible via routing, unlike the case in [20], where M must be at least 1.

Remark 12. *Corollaries 2-4 generalize our previous results that were limited to resolvable networks [22], i.e., we show the achievability of the rates in [22] for any combination network.* ■

VIII. Conclusion

In this work, we have investigated the fundamental limits two-hop cache-aided combination networks with caches at the relays and the end users, with and without security requirements. We have proposed a new coded caching scheme, by utilizing MDS coding and jointly optimizing the cache placement and delivery phases. We have shown that whenever the sum of the end user cache and the ones of its connected relays is sufficient to store the database, then there is no need for the server transmission over the first hop. We have developed genie-aided cut-set lower bounds on the rates and shown order optimality for the first hop and optimality for the second.

We have next investigated combination networks with caching relays under secure delivery constraints, secure caching constraints, as well as both secure delivery and secure

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2018.2844941, IEEE Journal on Selected Areas in Communications

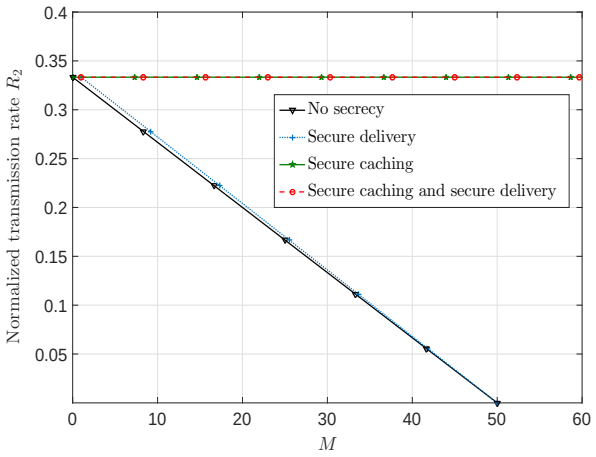

Fig. 5: Rates over the second hop under different system requirements for $N = 0$, $D = 50$, $K = 15$, $h = 5$ and $r = 3$.

caching constraints. The achievability schemes, for each of these requirements, jointly optimize the cache placement and delivery phases, and utilize one-time padding and secret sharing. We have illustrated the impact of the network structure and relaying on the system performance after imposing different secrecy constraints.

The decomposition philosophy using MDS codes we have utilized in this work allows adopting the ideas developed for the classical coded caching setup to cache-aided combination networks. Future directions in combination networks include caching with untrusted relays and considering the physical layer impairments in the delivery phase.

REFERENCES

[1] K. C. Almeroth and M. H. Ammar, "The use of multicast delivery to provide a scalable and interactive video-on-demand service," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 6, pp. 1110–1122, 1996.

[2] M. R. Korupolu, C. G. Plaxton, and R. Rajaraman, "Placement algorithms for hierarchical cooperative caching," *Journal of Algorithms*, vol. 38, no. 1, pp. 260–302, 2001.

[3] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Info. Theory*, vol. 60, no. 5, pp. 2856–2867, 2014.

[4] M. Ji, G. Caire, and A. Molisch, "Fundamental limits of caching in wireless D2D networks," *IEEE Trans. Info. Theory*, vol. 62, no. 2, pp. 849–869, 2016.

[5] N. Karamchandani, U. Niesen, M. A. Maddah-Ali, and S. N. Diggavi, "Hierarchical coded caching," *IEEE Trans. Info. Theory*, vol. 62, no. 6, pp. 3212–3229, 2016.

[6] M. A. Maddah-Ali and U. Niesen, "Cache-aided interference channels," in *IEEE International Symposium on Info. Theory (ISIT)*, 2015, pp. 809–813.

[7] S. P. Shariatpanahi, S. A. Motahari, and B. H. Khalaj, "Multi-server coded caching," *IEEE Trans. Info. Theory*, vol. 62, no. 12, pp. 7253–7271, 2016.

[8] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Caching in combination networks," in *49th Asilomar Conference on Signals, Systems and Computers*, 2015, pp. 1269–1273.

[9] 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 *16th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2015, pp. 695–699.

[10] L. Tang and A. Ramamoorthy, "Coded caching for networks with the resolvability property," in *IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 420–424.

[11] K. Wan, M. Ji, D. Tuninetti, and P. Piantanida, "Combination networks with caches: Novel inner and outer bounds with uncoded cache placement," in *arXiv:1701.06884*, 2017.

[12] K. Wan, M. Ji, P. Piantanida, and D. Tuninetti, "Novel inner bounds with uncoded cache placement for combination networks with end-user-caches," in *55th IEEE Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2017, pp. 361–368.

[13] ——, "Caching in combination networks: Novel multicast message generation and delivery by leveraging the network topology," *arXiv:1710.06752*, 2017.

[14] C. K. Ngai and R. W. Yeung, "Network coding gain of combination networks," in *IEEE Information Theory Workshop (ITW)*, 2004, pp. 283–287.

[15] M. Xiao, M. Médard, and T. Aulin, "A binary coding approach for combination networks and general erasure networks," in *IEEE International Symposium on Information Theory (ISIT)*, 2007, pp. 786–790.

[16] Z. Baranyai, "On the factorization of the complete uniform hypergraph," in *Infinite and finite sets (Colloq., Keszthely, 1973; dedicated to P. Erdos on his 60th birthday)*, vol. 1, 1975, pp. 91–108.

[17] S. Lin and D. J. Costello, *Error control coding*. Pearson Education India, 2004.

[18] A. Sengupta, R. Tandon, and T. C. Clancy, "Fundamental limits of caching with secure delivery," *IEEE Trans. on Info. Forensics and Security*, vol. 10, no. 2, pp. 355–370, 2015.

[19] Z. H. Awan and A. Sezgin, "Fundamental limits of caching in D2D networks with secure delivery," in *International Conference on Communication Workshop (ICCW)*, 2015, pp. 464–469.

[20] V. Ravindrakumar, P. Panda, N. Karamchandani, and V. Prabhakaran, "Fundamental limits of secretive coded caching," in *IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 425–429.

[21] A. A. Zewail and A. Yener, "Fundamental limits of secure device-to-device coded caching," in *50th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2016, pp. 1414–1418.

[22] A. A. Zewail and A. Yener, "Coded caching for resolvable networks with security requirements," in *the 3rd Workshop on Physical-Layer Methods for Wireless Security, CNS*, 2016.

[23] R. W. Yeung, *Information theory and network coding*. Springer Science & Business Media, 2008.

[24] U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands," *IEEE Trans. Info. Theory*, vol. 63, no. 2, pp. 1146–1158, 2016.

[25] K. Wan, D. Tuninetti, and P. Piantanida, "On caching with more users than files," in *IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 135–139.

[26] Z. Chen, P. Fan, and K. B. Letaief, "Fundamental limits of caching: Improved bounds for small buffer users," *arXiv:1407.1935*, 2014.

[27] C. Tian and J. Chen, "Caching and delivery via interference elimination," in *IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 830–834.

[28] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, 1949.

[29] R. Cramer, I. B. Damgard, and J. B. Nielsen, *Secure Multiparty Computation and Secret Sharing*. Cambridge University Press, 2015.

[30] G. R. Blakley and C. Meadows, "Security of ramp schemes," in *Workshop on the Theory and Application of Cryptographic Techniques*. Springer, 1984, pp. 242–268.

Ahmed A. Zewail (S'07) received the B.Sc. degree in electrical engineering from Alexandria University, Alexandria, Egypt, in 2011, and the M.Sc. degree in wireless communications from Nile University, Giza, Egypt, in 2013. Since 2013, he has been pursuing the Ph.D. degree and has been a graduate research assistant with the School of Electrical Engineering and Computer Science, Pennsylvania State University, University Park, PA, USA. His research interests include wireless communication, network information theory, information theoretic security, and cache-aided networks.

Aylin Yener (S'91–M'01–SM'14–F'15) received the B.Sc. degree in electrical and electronics engineering and the B.Sc. degree in physics from Bogazici University, Istanbul, Turkey, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Wireless Information Network Laboratory (WINLAB), Rutgers University, New Brunswick, NJ, USA. She is a Professor of Electrical Engineering at The Pennsylvania State University, University Park, PA, USA, since 2010, where she joined the faculty as an assistant professor in 2002, and was an associate professor 2006-2010. Since 2017, she is a Dean's Fellow in the College of Engineering at The Pennsylvania State University. She was a visiting professor of Electrical Engineering at Stanford University in 2016-2018 and a visiting associate professor in the same department in 2008-2009. Her current research interests are in caching systems, information security, green communications, and more generally in the fields of communication theory, information theory and network science. She received the NSF CAREER Award in 2003, the Best Paper Award in Communication Theory from the IEEE International Conference on Communications in 2010, the Penn State Engineering Alumni Society (PSEAS) Outstanding Research Award in 2010, the IEEE Marconi Prize Paper Award in 2014, the PSEAS Premier Research Award in 2014, and the Leonard A. Doggett Award for Outstanding Writing in Electrical Engineering at Penn State in 2014. She is a fellow of the IEEE, and a distinguished lecturer for the IEEE Communications Society and the IEEE Vehicular Technology Society.

Dr. Yener is a member of the Board of Governors of the IEEE Information Theory Society (2015-2020), where she was previously the Treasurer from 2012 to 2014. She served as the Student Committee Chair for the IEEE Information Theory Society from 2007 to 2011, and was the co-Founder of the Annual School of Information Theory in North America in 2008. She was a Technical (Co)-Chair for various symposia/tracks at the IEEE ICC, PIMRC, VTC, WCNC, and Asilomar in 2005, 2008-2014 and 2018. She served as an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS from 2009 to 2012, an Editor and an Editorial Advisory Board Member for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2001 to 2012, and a Guest Editor for the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY in 2011, and the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS in 2015. Currently, she serves on the Editorial Board of the IEEE TRANSACTIONS ON MOBILE COMPUTING and as a Senior Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS.