Secure Caching and Delivery for Combination Networks with Asymmetric Connectivity

Ahmed A. Zewail and Aylin Yener

Wireless Communications and Networking Laboratory (WCAN)
The School of Electrical Engineering and Computer Science
The Pennsylvania State University, University Park, PA 16802.
zewail@psu.edu *yener@engr.psu.edu*

***Abstract*—We consider information theoretic security in a two-hop combination network where there are groups of end users with distinct degrees of connectivity served by a layer of relays. The model represents a network set up with users having access to asymmetric resources, here the number of relays that they are connected to, yet demand security guarantees uniformly. We study two security constraints separately and simultaneously: secure delivery where the information must be kept confidential from an external entity that wiretaps the delivery phase; and secure caching where each cache-aided end-user can retrieve the file it requests and cannot obtain any information on files it does not. The achievable schemes we construct are multi-stage where each stage completes requests by a class of users.**

I. Introduction

Coded caching is a recent advance in alleviating network congestion and improving delivery performance. In [1], it is shown that utilizing users' cache memories, network traffic is partially shifted to off-peak hours; and designing cache contents amenable to utilizing multicast signaling during delivery brings in gains in addition to those provided by local content storage. There has been extensive work in coded caching systems in various network settings to date. References [2]–[5] have studied cache-aided two-hop networks, where a layer of relays connects the server to its end users. In particular, reference [2] has investigated a single-server symmetric layered network, known as a *combination network*, where the end users are equipped with cache memories. In such networks, the server is connected to a set of h relay nodes, which communicate to $\binom{h}{r}$ users, such that each user is connected to a distinct set of r relay nodes. In references, [3], [4], we have considered this network model, adding cache memories at the relay nodes as well as the ones at the end users. We have proposed a coded caching scheme that decomposes the network into h virtual sub-networks such that the delivery load per relay node is optimal with respect to the cut-set bound.

The main application space for coded caching is content delivery services. In many of these, information security is important, not only from the perspective of the end user, but also from the perspective of the provider. Streaming services require paid subscribers for access to their database contents. This calls for cache-aided systems that not only reduce the delivery load but also keep the content secret from unauthorized parties. In this paper, we focus on this aspect and study information theoretic security in coded-caching. The model that we consider is that of a combination network as in [4]. However different from [3], [4], we will study combination networks where end-users are heterogeneous in their access to network resources. The notion of heterogeneity in user resources have been studied in different coded-caching settings, see for example, references [6], [7] which consider distinct cache sizes and link qualities at the end users.

In recent reference [8], we have modeled heterogeneity in connectivity of combination networks by considering different classes of end users, each of which is connected to a different number of relay nodes. More specifically, we have considered two classes of end users such that each user from class 1 is connected to r_1 relay nodes while each user from class 2 is connected to r_2 relay nodes, $r_1 > r_2$, as shown in Fig. 1. We have developed a centralized coded caching scheme that utilizes maximum distance separable (MDS) codes and jointly optimizes the cache placement and delivery phases. In this work, we consider this exact setting when information security constraints must be imposed on the system design. First, we consider the scenario where the database must be kept secret from any external eavesdropper that overhears the delivery phase, i.e., *secure delivery* [9], [10]. Next, we consider the scenario where end users must not be able gain any information about the files that they did not request, i.e., *secure caching* [11] [12]. Last, we impose both requirements simultaneously. Utilizing the approach in [8], we jointly optimize the cache placement and delivery phases using one-time padding [13] and secret sharing schemes [14]. We observe that the cost of imposing secure delivery becomes negligible as the library size increases, while secure caching incurs an inherent cost. We note that, while in this paper, we focus on networks with two classes of end users, the ideas are extendable to networks with more than two classes of end users, by translating the two-stage schemes to multi-stage schemes.

Notation: $\oplus$ refers to bitwise XOR operation, $|W|$ denotes size of W, and $[K] \triangleq \{1, \dots, K\}$.

II. System Model

A. Network Connectivity

We consider a two-hop network, where the server, S, is connected to K end users via a set of h relay nodes. The end users are classified into two classes. K_1 users belong to class 1; each of these users is connected to a distinct set of r_1 relay nodes, i.e., $K_1 = \binom{h}{r_1}$. The remaining $K_2 = K - K_1$ users

978-1-5386-6900-6/19/$31.00 ©2019 IEEE

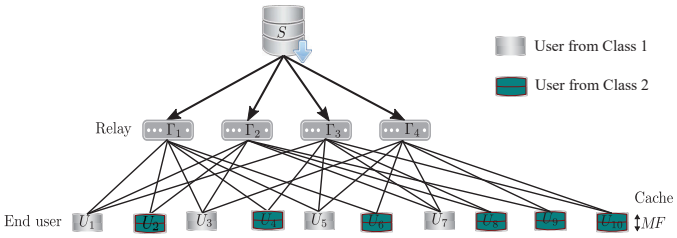

Fig. 1: An asymmetric combination network with two classes of end users where $K=10$, $h=4$, $r_1=3$ and $r_2=2$.

belong to class 2; each of them is connected to r_2 relay nodes, i.e., $K_2=\binom{h}{r_2}$. Thus, each relay node is connected to $L_1=\binom{h-1}{r_1-1}=\frac{r_1K_1}{h}$ and $L_2=\binom{h-1}{r_2-1}=\frac{r_2K_2}{h}$ users from class 1 and 2, respectively. Similar to [2], all network links are assumed to be noiseless and unicast. We define $\mathcal{R}=\{\Gamma_1,..,\Gamma_h\}$ as the set of relay nodes, and $\mathcal{U}=\{U_1,..,U_K\}$ as the set of all end users in the network. We denote the set of end users connected to Γ_j by $\mathcal{N}(\Gamma_j)$, i.e., $|\mathcal{N}(\Gamma_j)|=L_1+L_2$ for $j=1,..,h$, and the set relay nodes connected to user k from class i by $\mathcal{N}(U_k)$, i.e., $|\mathcal{N}(U_k)|=r_i$, $i=1,2$. Without loss of generality, we assume that $r_1\geq r_2$. We define the following function which returns the relative order of user k with respect to the neighbors of relay node Γ_j. The function $Index(,):(j,k)\rightarrow\{1,..,L_1+L_2\}$, where $j\in\{1,..,h\}$ and $k\in\mathcal{N}(\Gamma_j)$, is defined as a function that orders the end users connected to each relay in ascending order. For example, in Fig. 1, we have $\mathcal{N}(\Gamma_2)=\{1,2,3,7,8,9\}$, $\mathcal{N}(\Gamma_4)=\{3,5,6,7,9,10\}$ and

$$Index(2,1)=1,\ Index(2,2)=2,\ Index(2,9)=6.$$

B. Caching Model

The server S has a database of N files, $W_1,..,W_N$, each with size F symbols over the field $\mathbb{F}_{2^q}$ and $N\geq K$. Each end user has a cache memory of size MF symbols, i.e., M represents the normalized memory size. The system operates over two phases.

1) Cache Placement Phase: The server allocates functions of its database in the end users' cache memories. These are designed, without the knowledge of the actual demands in the delivery phase, subject to the memory capacity constraints.

Definition 1. *(Cache Placement): The contents of the cache memory at user k are given by*

$$Z_k=\phi_k(W_1,W_2,..,W_N),\qquad(1)$$

where $\phi_k:[2^F]^N\rightarrow[2^F]^M$, such that $H(Z_k)\leq MF$. ■

2) Delivery Phase: Each user requests a randomly selected file [1]. We define d_k to denote the index of the requested file by user k, i.e., $d_k\in\{1,2,..,N\}$, and $\boldsymbol{d}$ to represent the demand vector of all users. The server responds to the users' requests by transmitting signals to each of the relay nodes. Then, each relay node forwards its received signal to the set of intended end users. From its received signals and Z_k, user k should be able to reconstruct its requested file W_{d_k}.

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

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

where $\psi_i:[2^F]^N\times[N]^K\rightarrow[2^F]^{R_1}$, 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

$$Y_{j,\boldsymbol{d},k}=\varphi_k(X_{j,\boldsymbol{d}},\boldsymbol{d}),\qquad(3)$$

where $\varphi_k:[2^F]^{R_1}\times[N]^K\rightarrow[2^F]^{R_{2,i}}$, and $R_{2,i}$ is the normalized rate of the transmitted signal from the relay node to a connected end user from class i. In addition, user k, from class i, has a decoding function to recover its requested file, given by

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

where $\mu_k:[2^F]^{M_2}\times[N]^K\times[2^F]^{r_iR_{2,i}}\rightarrow[2^F]$, and $i=1,2$. ■

Each end user must be able to recover its requested file *reliably*, i.e., for any $\epsilon>0$, $\max_{\boldsymbol{d},k}P(\hat{W}_{d_k}\neq W_{d_k})<\epsilon$.

III. Networks with Secure Delivery

In this section, 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 $\delta>0$

$$I(\mathcal{X},\mathcal{Y};W_1,\ldots,W_N)<\delta,\qquad(5)$$

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

To guarantee (5), we place keys in the network caches during placement phase. These keys are used to encrypt, i.e., one-time pad [13], the transmitted signals during the delivery phase as in [9] and [15]. As a first step, the server divides each file into r_1 equal-size subfiles. Then, it encodes them using an $(h+r_1-r_2,r_1)$ maximum distance separable (MDS) code [16]. We denote by f_n^j the resulting encoded symbols, where n is the file index and $j=1,2,\ldots,h+r_1-r_2$. The size of each encoded symbol, f_n^j, is F/r_1 symbols, and any r_1 encoded symbols are sufficient to reconstruct the whole file.

A. First Stage

1) Cache Placement Phase: For $M=1+\frac{t(N-1)}{L_1+L_2}$, and $t\in\{0,1,\ldots,L_1+L_2\}$, each encoded symbol is divided into $\binom{L_1+L_2}{t}$ disjoint pieces each of which is denoted by $f^j_{n,\mathcal{T}}$, where $\mathcal{T}\subseteq[L_1+L_2]$, and $|\mathcal{T}|=t$. The size of each piece is $\frac{F}{r_1\binom{L_1+L_2}{t}}$ symbols. The server allocates the pieces $f^j_{n,\mathcal{T}}$, $\forall n$ 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 $h\binom{L_1+L_2}{t+1}$ independent keys. Each key is uniformly distributed with length $\frac{F}{r_1\binom{L_1+L_2}{t}}$ symbols. We denote each key by $K^u_{\mathcal{T}_K}$, where $u=1,\ldots,h$, and $\mathcal{T}_K\subseteq[\hat{K}],|\mathcal{T}_K|=t+1$. User k stores the keys $K^u_{\mathcal{T}_K}$, $\forall u\in\mathcal{N}(U_k)$, whenever $Index(u,k)\in\mathcal{T}_K$. Therefore, the cache contents at the end users are given by

$$Z_k=\left\{f^j_{n,\mathcal{T}},K^j_{\mathcal{T}_K}:\forall n,\forall j\in\mathcal{N}(U_k),Index(j,k)\in\mathcal{T},\mathcal{T}_K\right\}.\quad(6)$$

At the end of the cache placement phase of the first stage, each user from class 1 stores $r_1\binom{L_1+L_2-1}{t-1}$ pieces each of size $\frac{F}{r_1\binom{L_1+L_2}{t}}$ symbols, in addition to $r_1\binom{L_1+L_2-1}{t}$ keys with the same size. Therefore, the accumulated number of symbols in its cache memory is given by

$$r_1 N\binom{L_1+L_2-1}{t-1}\frac{F}{r_1\binom{L_1+L_2}{t}}+r_1\binom{L_1+L_2-1}{t}\frac{F}{r_1\binom{L_1+L_2}{t}} = 1+\frac{t(N-1)}{L_1+L_2}F = MF \text{ symbols}, \quad (7)$$

i.e., the memory capacity constraint is satisfied. Each user from class 2 at this stage have stored $r_2 N\binom{L_1+L_2-1}{t-1}$ pieces each of size $\frac{F}{r_1\binom{L_1+L_2}{t}}$ symbols and $r_2\binom{L_1+L_2-1}{t}$ keys. We define M_f to be normalized memory size of the users from class 2 at the end of the first stage which is equal to $\frac{r_1-r_2}{r_1}M$.

2) Coded Delivery Phase: For each relay Γ_j, at each transmission instance, we consider $\mathcal{S}\subseteq[L_1+L_2]$, where $|\mathcal{S}|=t+1$. For each choice of $\mathcal{S}$, the server transmits to the relay node Γ_j, the signal

$$X^{\mathcal{S},1}_{j,\boldsymbol{d}} = K^j_S \oplus_{\{k:k\in\mathcal{N}(\Gamma_j),Index(j,k)\in\mathcal{S}\}} f^j_{d_k,\mathcal{S}\setminus\{Index(j,k)\}}. \quad (8)$$

In total, the server transmits to Γ_j, the following signal

$$X^1_{j,\boldsymbol{d}} = \bigcup_{\mathcal{S}\subseteq[L_1+L_2]:|\mathcal{S}|=t+1}\{X^{\mathcal{S},1}_{j,\boldsymbol{d}}\}. \quad (9)$$

Then, Γ_j forwards $X^{\mathcal{S}}_{j,\boldsymbol{d}}$ to user k if $Index(j,k)\in\mathcal{S}$, i.e.,

$$Y^1_{j,\boldsymbol{d},k} = \bigcup_{\mathcal{S}\subseteq[L_1+L_2]:|\mathcal{S}|=t+1,Index(j,k)\in\mathcal{S}}\{X^{\mathcal{S},1}_{j,\boldsymbol{d}}\}. \quad (10)$$

After decrypting its received signals, user k can recover the following set of pieces, utilizing its cached contents

$$\left\{f^j_{d_i,\mathcal{T}}:\mathcal{T}\subseteq[L_1+L_2]\setminus\{Index(j,i)\},|\mathcal{T}|=t\right\}.$$

Adding these pieces to the cached ones, i.e., $f^j_{d_k,\mathcal{T}}$ with $Index(j,k)\in\mathcal{T}$, user k can recover the encoded symbol $f^j_{d_k}$. If user k belongs to class 1, i.e., it receives signals from r_1 different relay nodes, it obtains the encoded symbols $f^j_{d_k}$, $\forall j\in\mathcal{N}(U_k)$, thus user k is able to reconstruct W_{d_k}. In contrast, if user k belongs to class 2 it obtains only r_2 encoded symbols from its requested file.

B. Second Stage

In the second stage, we focus on delivering the missing r_1-r_2 encoded symbols of the requested files by the users in class 2. After the first stage, we have a reduced network, where the server has a library of N files, each of them if formed by the concatenation of the encoded symbols $f^{h+1}_n,\dots,f^{h+r_1-r_2}_n$, i.e., the size of each reduced file is $\frac{r_1-r_2}{r_1}F$ symbols. To describe our achievability scheme, we first define $t_1=\lceil\frac{M-1}{N-1}L_2\rceil$, $t_2=\lfloor\frac{M-1}{N-1}L_2\rfloor$, and α is chosen such that $\frac{M-1}{N-1}L_2=\alpha t_1+(1-\alpha)t_2$, for some $\alpha\in[0,1]$. The scheme is described given the memory parameters t_1 and t_2 as follows. The concatenation of $f^{h+1}_n,\dots,f^{h+r_1-r_2}_n$ is divided into two parts, $\hat{W}^1_n$ and $\hat{W}^2_n$, of sizes $\alpha\frac{r_1-r_2}{r_1}F$ symbols and $(1-\alpha)\frac{r_1-r_2}{r_1}F$ symbols, respectively.

1) Cache Placement Phase: The first part, $\hat{W}^1_n$, is divided into r_2 equal-size subfiles. Then, it encodes them using an (h,r_2) maximum distance separable (MDS) code [16]. We denote by $g^{1,j}_n$ the resulting encoded symbols, where n is the file index and $j=1,2,\dots,h$. The size of each encoded symbol, $g^{1,j}_n$, is $\alpha\frac{r_1-r_2}{r_2r_1}F$ symbols, and any r_2 encoded symbols are sufficient to reconstruct $\hat{W}^1_n$.

Each encoded symbol is divided into $\binom{L_2}{t_1}$ disjoint pieces each of which is denoted by $g^{1,j}_{n,\mathcal{T}_1}$, where $\mathcal{T}_1\subseteq[L_2]$, and $|\mathcal{T}_1|=t_1$. The size of each piece is $\alpha\frac{r_1-r_2}{r_2r_1\binom{L_2}{t_1}}F$ symbols. In addition, the server generates $\binom{L_2}{t_1+1}$ keys, $K^{1,j}_{\mathcal{T}_{K,1}}$, and $\mathcal{T}_{K,1}\subset[L_1]$ and $|\mathcal{T}_K|=t_1+1$. each of them with same length as $g^{1,j}_{n,\mathcal{T}_1}$. The server allocates the pieces $g^{1,j}_{n,\mathcal{T}_1}$, $\forall n$ and the keys $K^{1,j}_{\mathcal{T}_{K,1}}$ in the cache memory of user k from class 2 if $k\in\mathcal{N}(\Gamma_j)$ and $Index(j,k)\in\mathcal{T},\mathcal{T}_{K,1}$.

A similar allocation scheme with key generation will be applied the second part $\hat{W}^2_n$ with parameter t_2 instead of t_1. Therefore, by the end of the cache placement phase, the cached contents at user k from class 2 is given by

$$Z_k=\Big\{f^j_{n,\mathcal{T}},K^j_{\mathcal{T}_K},g^{1,j}_{n,\mathcal{T}_1},K^{1,j}_{\mathcal{T}_{K,1}},g^{2,j}_{n,\mathcal{T}_2},K^{2,j}_{\mathcal{T}_{K,2}}:k\in\mathcal{N}(\Gamma_j), Index(j,k)\in\mathcal{T},\mathcal{T}_1,\mathcal{T}_2,\mathcal{T}_K,\mathcal{T}_{K,1},\mathcal{T}_{K,2}\ \forall n\Big\}. \quad (11)$$

The memory capacity constraint can be verified to be satisfied.

2) Coded Delivery Phase: For each relay Γ_j, we consider $\mathcal{S}_i\subseteq[L_2]$, where $|\mathcal{S}|=t_i+1$, and $i=1,2$. For each choice of $\mathcal{S}_i$, the server transmits to Γ_j, the signal

$$K^{i,j}_{\mathcal{S}_i}\oplus_{\{k:k\in\mathcal{N}(\Gamma_j),\ Index(j,k)\in\mathcal{S}_i\}}g^{i,j}_{d_k,\mathcal{S}_i}.$$

Then, Γ_j forwards its received signal to user k from class 2 if $Index(j,k)\in\mathcal{S}_i$. At the end of the second stage, user k from class 2 can recover the following set of pieces from the signals received from Γ_j, utilizing its cached contents

$$\left\{g^{i,j}_{d_k,\mathcal{T}}:\mathcal{T}_i\subseteq[L_2]\setminus\{Index(j,k)\},|\mathcal{T}_i|=t_i,\ i=1,2\right\}.$$

Note that user k had cached $g^{i,j}_{d_k,\mathcal{T}_i}$ with $Index(j,k)\in\mathcal{T}_i$, thus user k can recover the encoded symbol $g^{i,j}_{d_k}$. Since, user k from class 2 receives signals from r_2 different relay nodes, it obtains the encoded symbols $g^{i,j}_{d_k}$, $\forall j\in\mathcal{N}(U_k)$, thus user k can reconstruct $f^{h+1}_{d_k},\dots,f^{h+r_1-r_2}_{d_k}$. Therefore, at the end of the delivery phase, user k from class 2 can decode its requested file from r_1 of its encoded symbols.

Remark 1. *Note that each of the transmitted signals by the server is encrypted using a one-time pad that has length equal to the length of each subfile ensuring prefect secrecy [13]. Observing any of the transmitted signals without knowing the encryption key will not reveal any information about the database files [13]. The same applies for the messages transmitted by the relays. Thus, (5) is satisfied.* ■

C. Secure Delivery Rates

Denote the secure delivery rates in the first and second hop with R^s_1 and $R^s_{2,i}$, respectively.

1) First Stage: Each relay node is responsible for $\binom{L_1+L_2}{t+1}$ transmissions, each of length $\frac{F}{r_1\binom{L_1+L_2}{t}}$, thus

$$R_1^{s,1}F = \frac{\binom{L_1+L_2}{t+1}}{r_1\binom{L_1+L_2}{t}}F = \frac{L_1+L_2-t}{r_1(t+1)}F. \quad (12)$$

In addition, each relay node forwards $\binom{L_1+L_2-1}{t}$ from its received signals to each of its connected end users, thus

$$R_{2,i}^{s,1}F = \frac{\binom{L_1+L_2-1}{t}}{r_1\binom{L_1+L_2}{t}}F = \frac{L_1+L_2-t}{r_1(L_1+L_2)}F, \quad (13)$$

2) Second Stage: Each relay node is responsible for $\binom{L_2}{t_1+1}$ transmissions, each of length $\alpha\frac{r_1-r_2}{r_2r_1\binom{L_2}{t_1}}F$, and $\binom{L_2}{t_2+1}$ transmissions, each of length $(1-\alpha)\frac{r_1-r_2}{r_2r_1\binom{L_2}{t_2}}F$, thus we have

$$\begin{aligned}R_1^{s,2}F &= \alpha\frac{(r_1-r_2)\binom{L_2}{t_1+1}}{r_2r_1\binom{L_2}{t_1}}F + (1-\alpha)\frac{(r_1-r_2)\binom{L_2}{t_2+1}}{r_2r_1\binom{L_2}{t_2}}F \\ &= \frac{r_1-r_2}{r_2r_1}\left(\alpha\frac{L_2-t_1}{t_1+1} + (1-\alpha)\frac{L_2-t_2}{t_2+1}\right)F. \quad (14)\end{aligned}$$

Then, each relay forwards $\binom{L_2-1}{t_1}$ and $\binom{L_2-1}{t_2}$ from its received signals to each of its connected end users from class 2, each of length equal to $\alpha\frac{r_1-r_2}{r_2r_1\binom{L_2}{t_1}}F$ and $(1-\alpha)\frac{r_1-r_2}{r_2r_1\binom{L_2}{t_2}}F$, respectively, thus

$$\begin{aligned}R_{2,2}^{s,2}F &= \alpha\frac{(r_1-r_2)\binom{L_2-1}{t_1}}{r_2r_1\binom{L_2}{t_1}}F + (1-\alpha)\frac{(r_1-r_2)\binom{L_2-1}{t_2}}{r_2r_1\binom{L_2}{t_2}}F \\ &= \frac{r_1-r_2}{r_2r_1L_2}\left(\alpha(L_2-t_1)+(1-\alpha)(L_2-t_2)\right)F. \quad (15)\end{aligned}$$

In total, $R_1^s = R_1^{s,1} + R_1^{s,2}$, $R_{2,1}^s = R_{2,i}^{s,1}$ and $R_{2,2}^s = R_{2,2}^{s,1} + R_{2,2}^{s,2}$. Therefore, we obtain the upper bound on the normalized secure delivery rates as stated in the following theorem.

Theorem 1. *The normalized transmission rates with secure delivery, $M = 1+\frac{t(N-1)}{L_1+L_2}$, and $t \in \{0,1,\ldots,L_1+L_2\}$, are upper bounded by*

$$\begin{aligned}R_1^s \leq\ & \frac{L_1+L_2-t}{r_1(t+1)} \\ &+ \frac{r_1-r_2}{r_1r_2}\left(\frac{\alpha(L_2-t_1)}{t_1+1} + \frac{(1-\alpha)(L_2-t_2)}{t_2+1}\right), \quad (16)\end{aligned}$$

$$R_{2,i}^s \leq \frac{1}{r_i}\left(1-\frac{M-1}{N-1}\right), \qquad i=1,2, \quad (17)$$

where $t_1 = \lceil\frac{M-1}{N-1}L_2\rceil$, $t_2 = \lfloor\frac{M-1}{N-1}L_2\rfloor$ and α is chosen such that $\frac{M-1}{N-1}L_2 = \alpha t_1 + (1-\alpha)t_2$. In addition, the convex envelope of these points is achievable by memory sharing.

IV. 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 $\delta>0$

$$\max_{\boldsymbol{d},\mathcal{V}} I(\boldsymbol{W}_{-d_k}; \{Y_{j,\boldsymbol{d},k} : j \in \mathcal{N}(U_k)\}, Z_k) < \delta, \quad (18)$$

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

In our achievability, we utilize secret sharing schemes [14] 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. *[14] [17] For a secret W with size F symbols, an (m,n) non-perfect secret sharing scheme generates n shares, $S_1, S_2, \ldots, S_n$, such that accessing any m shares does not reveal any information about the file W, i.e.,*

$$I(W;\mathcal{S}) = 0, \quad \forall \mathcal{S} \subseteq \{S_1,S_2,\ldots,S_n\}, |\mathcal{S}| \leq m. \quad (19)$$

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

$$H(W|S_1,S_2,\ldots,S_n)=0. \quad (20)$$

■

For large enough F, an (m,n) secret sharing scheme exists with shares of size equal to $\frac{F}{n-m}$ symbols [14], [17].

1) First Stage: For $M = \frac{tN}{L_1+L_2-t}$, and $t \in [L_1+L_2-1]$, during the first stage, each of the first h encoded subfiles is encoded using $\left(\binom{L_1+L_2-1}{t-1}, \binom{L_1+L_2}{t}\right)$ secret sharing scheme. The resulting shares are denoted by $S_{n,\mathcal{T}}^j$, where n is the file index i.e., $n \in [N]$, j is the index of the encoded symbol, i.e., $j=1,\ldots,h$, and $\mathcal{T} \subseteq [L_1+L_2], |\mathcal{T}| = t$. Each share has size

$$F_s = \frac{F/r_1}{\binom{L_1+L_2}{t} - \binom{L_1+L_2-1}{t-1}} = \frac{tF}{r_1(L_1+L_2-t)\binom{L_1+L_2-1}{t-1}}.$$

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 $Index(j,k) \in \mathcal{T}$. The delivery phase is performed as follows. 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

$$X_{j,\boldsymbol{d}}^{\mathcal{S}} = \bigoplus_{\{i: i\in\mathcal{N}(\Gamma_j),\ Index(j,i)\in\mathcal{S}\}} S_{d_i,\mathcal{S}\backslash\{Index(j,i)\}}^j. \quad (21)$$

In total, the server transmits to Γ_j, the signal $X_{j,\boldsymbol{d}} = \bigcup_{\mathcal{S}\subseteq[\hat{K}]:|\mathcal{S}|=t+1}\{X_{j,\boldsymbol{d}}^{\mathcal{S}}\}$. Then, Γ_j forwards the signal $X_{j,\boldsymbol{d}}^{\mathcal{S}}$ to user i whenever $Index(j,i) \in \mathcal{S}$, i.e., we have

$$Y_{j,\boldsymbol{d},i} = \bigcup_{\mathcal{S}\subseteq[\hat{K}]:|\mathcal{S}|=t+1, Index(j,i)\in\mathcal{S}} \{X_{j,\boldsymbol{d}}^{\mathcal{S}}\}. \quad (22)$$

User i can recover $\{S_{d_i,\mathcal{T}}^j : \mathcal{T} \subseteq [\hat{K}]\backslash\{Index(j,i)\}, |\mathcal{T}| = t\}$ from the signals received from Γ_j, utilizing its cache's contents. Adding these shares to the ones in its cache, i.e.,

$S^j_{d_i,\mathcal{T}}$ with $Index(j,i) \in \mathcal{T}$, user i can decode the encoded symbol $f^j_{d_i}$ from its $\binom{L_1+L_2}{t}$ shares. Since, user i from class 1 receives signals from r different relay nodes, it obtains the encoded symbols $f^j_{d_i}$, $\forall j \in \mathcal{N}(U_i)$, and can reconstruct W_{d_i}.

2) Second Stage: The server should serve the users from class 2 with the encoded subfiles $f_n^{h+1},\ldots, f_n^{h+r_1-r_2}$. We note that the fraction of free memory at the users from class 2 is given by $\frac{r_1-r_2}{r_1}M$. The same strategy from [4] can be applied on the reduced network, knowing that the effective file size is $\frac{r_1-r_2}{r_1}F$. Therefore, we can get the following theorem.

Theorem 2. *The normalized transmission rates with secure caching, $M=\frac{tN}{L_1+L_2}$, and $t\in\{0,1,\ldots,L_1+L_2\}$, are upper bounded by*

$$R_1^c \leq \frac{L_1+L_2}{r_1(t+1)}+\frac{(r_1-r_2)L_2}{r_1r_2}\left(\frac{\alpha}{t_1+1}+\frac{1-\alpha}{t_2+1}\right), \quad (23)$$

$$R_{2,i}^c \leq \frac{1}{r_i}, \qquad i=1,2, \quad (24)$$

where $t_1=\lceil\frac{ML_2}{N+M}\rceil$, $t_2=\lfloor\frac{ML_2}{N+M}\rfloor$ and α is chosen such that $\frac{ML_2}{N+M}=\alpha t_1+(1-\alpha)t_2$. In addition, the convex envelope of these points is achievable by memory sharing.

V. Networks with Secure Caching and Delivery

Here, we consider *secure caching* and *secure delivery* to be satisfied, simultaneously. The achievability scheme utilizes both secret sharing and one-time padding. Thus, we can get the following theorem.

Theorem 3. *The normalized transmission rates with secure caching and delivery, $M=1+\frac{tN}{L_1+L_2}$, and $t\in\{0,1,\ldots,L_1+L_2\}$, are upper bounded by*

$$R_1^{sc} \leq \frac{L_1+L_2}{r_1(t+1)}+\frac{(r_1-r_2)L_2}{r_1r_2}\left(\frac{\alpha}{t_1+1}+\frac{1-\alpha}{t_2+1}\right), \quad (25)$$

$$R_{2,i}^{sc} \leq \frac{1}{r_i}, \qquad i=1,2, \quad (26)$$

where $t_1=\lceil\frac{(M-1)L_2}{N+M-1}\rceil$, $t_2=\lfloor\frac{(M-1)L_2}{N+M-1}\rfloor$ and α is chosen such that $\frac{(M-1)L_2}{N+M-1}=\alpha t_1+(1-\alpha)t_2$. In addition, the convex envelope of these points is achievable by memory sharing.

VI. Numerical Results

In Fig. 2, we compare the achievable delivery load of our proposed schemes with different secrecy requirements. It is evident that the cost of imposing the secure delivery decreases as the memory size increases. In addition, whenever the end user's cache is sufficient to store the entire library, the delivery load is equal to zero in the case of secure delivery. In contrast, with secure caching the normalized delivery load is lower bounded by $\frac{1}{r_2}$.

VII. Conclusions

In this work, we have investigated combination networks with caches at both relay nodes and end users under secure delivery constraints, secure caching constraints, as well as both secure delivery and secure caching constraints. We have provided achievability schemes, for each of these requirements, by jointly optimizing the cache placement and delivery phases, utilizing one-time padding and secret sharing schemes. We have illustrated the impact of the network structure and relaying on the system performance after imposing different secrecy constraints.

We conclude by remarking that the idea behind our proposed caching schemes can be extended to networks with more than two classes of end users as illustrated in [8].

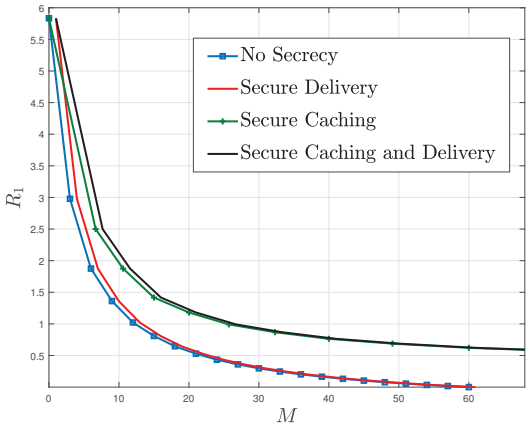

Fig. 2: The delivery load for $N=60$, $h=6$, $r_1=4$ and $r_2=3$.

References

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

[2] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Caching in combination networks," in *Proc. IEEE Asilomar*, 2015.

[3] A. A. Zewail and A. Yener, "Coded caching for combination networks with cache-aided relays," in *Proc. IEEE ISIT*, 2017.

[4] ——, "Combination networks with or without secrecy constraints: The impact of caching relays," *IEEE Journ. Sel. Areas in Commun.*, vol. 36, no. 6, pp. 1140–1152, 2018.

[5] K. Wan, M. Ji, P. Piantanida, and D. Tuninetti, "Novel inner bounds with uncoded cache placement for combination networks with end-user-caches," in *Proc. IEEE Allerton*, 2017.

[6] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Coded caching for heterogeneous systems: An optimization prespective," *accepted at IEEE Trans. Commun., arXiv:1810.08187*, 2019.

[7] S. S. Bidokhti, M. Wigger, and A. Yener, "Benefits of cache assignment on degraded broadcast channels," *arXiv:1702.08044*, 2017.

[8] A. A. Zewail and A. Yener, "Cache-aided combination networks with asymmetric end users," in *Proc. IEEE SPAWC*, 2019.

[9] 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.

[10] A. A. Zewail and A. Yener, "The wiretap channel with a cache," in *Proc. IEEE ISIT*, 2018.

[11] N. K. V. Ravindrakumar, P. Panda and V. Prabhakaran, "Fundametal limits of secretive coded caching," in *Proc. IEEE ISIT*, 2016.

[12] A. A. Zewail and A. Yener, "Fundamental limits of secure device-to-device coded caching," in *Proc. IEEE Asilomar*, 2016.

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

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

[15] Z. H. Awan and A. Sezgin, "Fundamental limits of caching in D2D networks with secure delivery," in *Proc. IEEE ICCW*, 2015.

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

[17] 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.