An Optimization Framework for Secure Delivery in Heterogeneous Coded Caching Systems

Ahmed A. Zewail, Abdelrahman M. Ibrahim, and Aylin Yener

Wireless Communications and Networking Laboratory (WCAN)
School of Electrical Engineering and Computer Science
The Pennsylvania State University, University Park, PA 16802.
{*zewail,ami137*}*@psu.edu* *yener@engr.psu.edu*

***Abstract*—This paper investigates the performance of cache-aided systems with heterogeneous caches and secure delivery. In particular, we consider users with unequal caches and assume the signals transmitted during the delivery phase to be overheard by an external eavesdropper which must not gain any information about the system's files. We study server-based delivery and device-to-device-based delivery where the server does not participate in the delivery phase. For each scenario, assuming uncoded placement and linear delivery schemes, we provide an optimization framework to minimize the secure delivery load. We show that the secure delivery requirement can be satisfied by modifying the memory capacity constraints in the non-secure framework to take into account the cost of caching keys. In addition, we show that the cost of secure delivery is negligible for caching systems with large number of files.**

I. Introduction

Caching alleviates network congestion during peak-traffic hours, by pushing data into the cache memories at the network edge during off-peak hours. The former is often called the delivery phase and the latter the placement phase. The contents requested by the users, are thus partially available at the users' local cache memories and the remaining pieces need to be delivered. Reference [1] has introduced the fundamental concept of *coded caching*, where the placement and delivery phases are jointly optimized to minimize the delivery load, by creating multicast opportunities. It has shown that there exists a fundamental trade-off between the delivery load and the cache sizes at the end-nodes. Understanding this fundamental trade-off in caching systems has been the focus of several recent studies [2]–[5]. In particular, for coded caching with uncoded placement, i.e., there is no coding over files during the placement phase, the delivery load memory trade-off has been characterized in references [2], [3]. The same trade-off with general (coded) placement has been studied in [4], [5].

End-users connect to the network using a variety of devices with different storage capabilities, e.g., laptops, smartphones, etc., which in turn motivates developing coded caching schemes for systems with heterogeneous cache sizes. References [6]–[9] proposed an optimization framework to minimize the total delivery load under uncoded placement [3]. Another important aspect to take into account while designing cache-aided system is information security, i.e., cache-aided systems should be designed not only to reduce the delivery load but also to keep the contents secret from unauthorized parties [10]–[15].

In this paper, we extend our optimization framework in [7], [9] to systems with secure delivery. In this setup, we wish to prevent an eavesdropper to the delivery phase from gaining any information about the system files [10]–[15]. We propose centralized cache placement and delivery schemes which are optimized in order to fully utilize the distinct cache memory sizes in a heterogeneous caching system while satisfying the secure delivery requirement. We consider two scenarios: server-based secure delivery and device-to-device secure delivery. In the latter, the server does not participate in the delivery.

Our objective is to jointly optimize the cache placement and secure delivery schemes, in order to minimize the worst-case delivery load assuming uniform demand distribution, i.e., we want to minimize the amount of data transmitted during the delivery phase, under the assumptions that the users request different files and the files are equally likely to be requested by any user. We focus on uncoded placement where only subpacketization of the files is utilized, i.e., no coding over the files in the placement phase [3]. We show that our optimization framework can ensure secure delivery by modifying the cache memory constraints in [7], [9] to account for caching secure keys [16] in addition to the data. Our numerical results compare the total delivery load with and without secure delivery. We observe that the cost of secure delivery decreases with the library size and becomes negligible for large library size.

Notation: Vectors are represented by boldface letters, sets of policies are represented by calligraphic letters, e.g., $\mathfrak{A}$, $\oplus$ refers to the binary XOR operation, $(x)^+ \triangleq \max\{0, x\}$, $|W|$ denotes cardinality of W, $[K] \triangleq \{1, \ldots, K\}$, $\mathcal{A} \subset \mathcal{B}$ denotes $\mathcal{A}$ being a subset of or equal to $\mathcal{B}$, $\subsetneq_\phi [K]$ denotes non-empty subsets of $[K]$, and ϕ denotes the empty set.

II. System Model

We consider a system with a server connected to K users [1], [7]. A library $\{W_1, \ldots, W_N\}$ of N files, each with size F bits, is stored at the server. We consider a heterogeneous system, where user k is equipped with a cache memory of size $M_k F$ bits. Without loss of generality, we assume that $M_1 \leq$

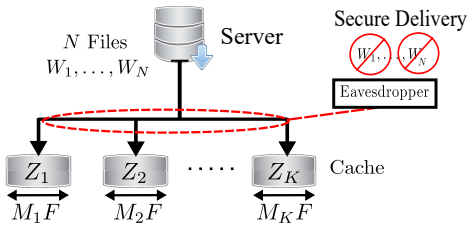

Fig. 1: A server-based secure delivery system with unequal caches.

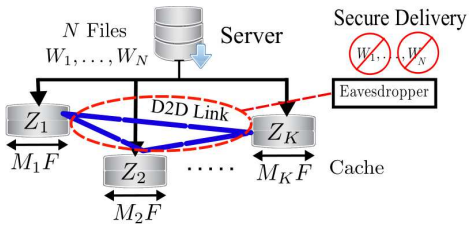

Fig. 2: A D2D-based secure delivery system with unequal caches.

$M_2 \leq \cdots \leq M_K$. Additionally, we define $m_k = M_k/N$ to denote the memory size of user k normalized by the library size NF, i.e., $m_k \in [0,1]$ for $M_k \in [0,N]$. The cache size vector is denoted by $\boldsymbol{M} = [M_1, \ldots, M_K]$ and its normalized version by the library size is denoted by $\boldsymbol{m} = [m_1, \ldots, m_K]$. We focus on the case, where the number of files is larger than or equal to the number of users, i.e., $N \geq K$. The system operates over two phases: placement phase and delivery phase.

A. Cache Placement Phase

In the placement phase, the server populates the users' cache memories without the knowledge of users' demands in the delivery phase. In particular, user k stores a subset Z_k of the library, subject to its cache size constraint. Formally, the users' cache contents are defined as follows.

Definition 1. *(Cache placement function) A cache placement function $\phi_k : [2^F]^N \to [2^F]^{M_k}$ maps the library to the cache of user k, i.e., $Z_k = \phi_k(W_1, W_2, .., W_N)$.* ■

In this work, the cache placement policies set $\mathfrak{A}$ satisfies two assumptions: 1) *Uncoded prefetching* [3], where the server places uncoded data at the users' cache memories, i.e., there is no coding over the files, 2) Under uniform demands, the partition of cache memory at user k, which is dedicated to caching data, is divided equally over the files.

B. Delivery Phase

User k requests a file W_{d_k} from the server. The users' demands are uniform and independent, i.e., the demand vector $\boldsymbol{d} = [d_1, \ldots, d_K]$ consists of identical and independent uniform random variables over the files as in [1], [7]. We consider two scenarios: Sever-Based and D2D-Based delivery.

1) Server-Based Delivery: In order to serve the users' demands, the server transmits a sequence of unicast/multicast signals over a noiseless shared link, see Fig. 1. In particular, it transmits the signals $X_{\mathcal{T},\boldsymbol{d}}$, to the users in the sets $\mathcal{T} \subsetneq_\phi [K]$. Each of the signals $X_{\mathcal{T},\boldsymbol{d}}$ has length $v_{\mathcal{T}}F$ bits, and is defined as follows.

Definition 2. *(Server-Based Encoding) Given a demand vector $\boldsymbol{d}$, an encoding function $\psi_{\mathcal{T},\boldsymbol{d}} : [2^F]^K \to [2^{v_{\mathcal{T}}F}]$ maps the requested files to a signal with length $v_{\mathcal{T}}F$ bits, which is sent to the users in $\mathcal{T}$, i.e., $X_{\mathcal{T},\boldsymbol{d}} = \psi_{\mathcal{T},\boldsymbol{d}}(W_{d_1}, .., W_{d_K})$.* ■

User k must be able to reconstruct W_{d_k} from the transmitted signals $X_{\mathcal{T},\boldsymbol{d}}, \mathcal{T} \subset [K]$ and Z_k, with negligible probability of error. Let $R_S \triangleq \sum_{\mathcal{T} \subsetneq_\phi [K]} v_{\mathcal{T}}$ be the amount of data transmitted by the server, normalized the file size F.

2) D2D-Based Delivery: The requested files must be delivered by utilizing D2D communications only [9], as shown in Fig. 2. User j transmits the sequence of unicast/multicast signals, $X_{j\to\mathcal{T},\boldsymbol{d}}$, to the users in the set $\mathcal{T} \subsetneq_\phi [K] \setminus \{j\}$. Let $|X_{j\to\mathcal{T},\boldsymbol{d}}| = v_{j\to\mathcal{T}}F$ bits, i.e., the *transmission variable* $v_{j\to\mathcal{T}} \in [0,1]$ represents the amount of data delivered to the users in $\mathcal{T}$ by user j as a fraction of the file size F.

Definition 3. *(D2D-Based Encoding) Given demand $\boldsymbol{d}$, an encoding $\psi_{j\to\mathcal{T}} : [2^F]^{M_j} \times [N]^K \to [2^F]^{v_{j\to\mathcal{T}}}$ maps the content cached by user j to a signal sent to the users in $\mathcal{T} \subsetneq_\phi [K]\setminus\{j\}$, i.e., the signal $X_{j\to\mathcal{T},\boldsymbol{d}} = \psi_{j\to\mathcal{T}}(Z_j, \boldsymbol{d})$ and $|X_{j\to\mathcal{T},\boldsymbol{d}}| = v_{j\to\mathcal{T}}F$.* ■

User k must be able to reconstruct W_{d_k} reliably using the received D2D signals $\{X_{j\to\mathcal{T},\boldsymbol{d}}\}_{j\neq k,\mathcal{T}}$ and its cache content Z_k. Let $R_j \triangleq \sum_{\mathcal{T}\subsetneq_\phi [K]\setminus\{j\}} v_{j\to\mathcal{T}}$ be the amount of data transmitted by user j, normalized the file size F, and $R_{D2D} = \sum_{j=1}^K R_j$ be the total normalized D2D delivery load.

C. Secure Delivery

We consider the case where an eavesdropper overhears the signals during the delivery phase. Our objective is to modify our caching scheme in order to prevent the eavesdropper from gaining any information about the library files [10]. This requirement is known as *secure delivery* [10], [14], [15]. For the Server-Based delivery, this requirement is captured by the following mutual information constraint, for any $\epsilon > 0$,

$$I(W_1, ..W_N; \{X_{\mathcal{T},\boldsymbol{d}}\}_{\mathcal{T}\subsetneq_\phi [K]}) \leq \epsilon, \tag{1}$$

while for the D2D-Based delivery, it is captured by

$$I(W_1, ..W_N; \{X_{j\to\mathcal{T},\boldsymbol{d}}\}_{\forall j,\mathcal{T}\subsetneq_\phi [K]\setminus\{j\}}) \leq \epsilon. \tag{2}$$

III. Sever-Based Secure Delivery

A. Placement Phase

Without the secure delivery requirement [7], each file W_l is partitioned into 2^K subfiles, and the set of uncoded prefetching schemes for a given $\boldsymbol{m}$, $\mathfrak{A}(\boldsymbol{m})$, is defined as

$$\left\{ \boldsymbol{a} \in [0,1]^{2^K} \middle| \sum_{\mathcal{S}\in 2^{[K]}} a_{\mathcal{S}} = 1, \sum_{\mathcal{S}\in 2^{[K]} : k\in\mathcal{S}} a_{\mathcal{S}} \leq m_k, \forall k \in [K] \right\}, \tag{3}$$

where $|\tilde{W}_{n\mathcal{S}}| = a_{\mathcal{S}}F$ bits and the allocation vector $\boldsymbol{a}$ represents the collection of allocation variables $a_{\mathcal{S}}$, $\mathcal{S} \subset [K]$.

To ensure the secure delivery, the server generates the keys $K_{\mathcal{T}}, \mathcal{T} \subsetneq_\phi [K]$, where $K_{\mathcal{T}}$ is uniformly distributed over $\{1, .., 2^{v_{\mathcal{T}}F}\}$ [16], i.e., the size of the key $K_{\mathcal{T}}$ is $v_{\mathcal{T}}F$ bits,

which will be specified later. The users' cache memories are divided between the subfiles $\tilde{W}_{n,\mathcal{S}}$ and the keys $K_{\mathcal{T}}$. In particular, user k stores the subfiles $\bigcup_{\mathcal{S}:\, k\in\mathcal{S}} \tilde{W}_{n,\mathcal{S}}$, $\forall\, n \in [N]$ and the keys $\bigcup_{\mathcal{T}:\, k\in\mathcal{T}} K_{\mathcal{T}}$. In turn, we have the following memory capacity constraints.

$$N \sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}} + \sum_{\mathcal{T}\subsetneq_{\phi}[K]\,:\,k\in\mathcal{T}} v_{\mathcal{T}} \leq M_k, \forall\, k \in [K]. \quad (4)$$

The fact that the total amount of bits transmitted to user k during the delivery phase, must be sufficient to complete its requested file, implies

$$\sum_{\mathcal{T}\subsetneq_{\phi}[K]\,:\,k\in\mathcal{T}} v_{\mathcal{T}} \geq 1 - \sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}}. \quad (5)$$

Thus, the memory constraints in (4) can be expressed as

$$\sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}} \leq \frac{(M_k-1)^+}{N-1} \triangleq m_k^s, \forall k \in [K], \quad (6)$$

where m_k^s denotes the normalized local caching gain under secure delivery requirement. Note that the cache memory size has to satisfy $M_k \geq 1$, $\forall k \in [K]$, otherwise secure delivery cannot be guaranteed [10]. Therefore, the set of uncoded prefetching schemes under secure delivery is given by $\mathfrak{A}(\boldsymbol{m}^s)$, where $\boldsymbol{m}^s = [m_1^s, .., m_K^s]$, i.e.,

$$\left\{ \boldsymbol{a} \in [0,1]^{2^K} \middle| \sum_{\mathcal{S}\in 2^{[K]}} a_{\mathcal{S}} = 1, \sum_{\mathcal{S}\in 2^{[K]}\,:\,k\in\mathcal{S}} a_{\mathcal{S}} \leq m_k^s, \forall k \in [K] \right\}. \quad (7)$$

B. Delivery Phase

The delivery procedure is similar to the proposed one in [7], with two additional steps. The first one, at the server, where each data signal intended to the users in $\mathcal{T}$ is encrypted using the key $K_{\mathcal{T}}$. The second step, at the end users, where each of the received signals is decrypted with the knowledge of the cached keys. In particular, a multicast transmission $X_{\mathcal{T},\boldsymbol{d}}$ is constrained by the side information stored at the users in $\mathcal{T}\setminus\{j\}$ and not available at user j, $\forall j \in \mathcal{T}$, represented by

$$\mathcal{B}_j^{\mathcal{T}} \triangleq \left\{ \mathcal{S} \subset [K] : \mathcal{T}\setminus\{j\} \subset \mathcal{S}, j \notin \mathcal{S} \right\}, \forall j \in \mathcal{T}. \quad (8)$$

For example, for $K = 4$, the side information stored at user 2 and not available at user 1 is represented by $\mathcal{B}_1^{\{1,2\}} = \big\{\{2\},\{2,3\},\{2,4\},\{2,3,4\}\big\}$. We denote all storage sets related to $\mathcal{T}$ by $\mathcal{B}^{\mathcal{T}} \triangleq \bigcup_{j\in\mathcal{T}} \mathcal{B}_j^{\mathcal{T}}$. Using the notion of side information sets $\mathcal{B}_j^{\mathcal{T}}$, we get

$$X_{\mathcal{T},\boldsymbol{d}} = K_{\mathcal{T}} \oplus_{j\in\mathcal{T}} W_{d_j}^{\mathcal{T}} = K_{\mathcal{T}} \oplus_{j\in\mathcal{T}} \left(\bigcup_{\mathcal{S}\in\mathcal{B}_j^{\mathcal{T}}} W_{d_j,\mathcal{S}}^{\mathcal{T}} \right). \quad (9)$$

Furthermore, we have

- The structure of the multicast signals $X_{\mathcal{T},\boldsymbol{d}}$ imposes the following constraints

$$\sum_{\mathcal{S}\in\mathcal{B}_j^{\mathcal{T}}} u_{\mathcal{S}}^{\mathcal{T}} = v_{\mathcal{T}},\ \forall\, \mathcal{T} \subsetneq_{\phi} [K],\ \forall\, j \in \mathcal{T}. \quad (10)$$

Algorithm 1 Server-Based Secure Delivery Scheme

Input: $\boldsymbol{d}, \boldsymbol{a}, \boldsymbol{u}, \boldsymbol{v}, \{W_1, \ldots, W_N\}$
Output: $X_{\mathcal{T},\boldsymbol{d}}, \mathcal{T} \subsetneq_{\phi} [K]$

```
for {S ⊂ [K] : 1 ≤ |S| ≤ K − 1} do
    for {j ∈ [K] : j ∉ S} do
        Partition W̃_{d_j,S} into W^T_{d_j,S}, {T : j ∈ T, T ∩ S ≠ φ, T \ {j} ⊂ S}
        and W^φ_{d_j,S}, such that |W^T_{d_j,S}| = u^T_S
        and W^φ_{d_j,S} is the remaining piece.
    end for
end for
for T ⊊_φ [K] do
    if T = {j} then
        X_{{j},d} ← ( W_{d_j} \ ( ⋃_{S:j∈S} W̃_{d_j,S} ⋃ ⋃_{T',S} W^{T'}_{d_j,S} ) ) ⊕ K_j
    else
        X_{T,d} ← ( ⊕_{j∈T} ⋃_{S∈B_j^T} W^T_{d_j,S} ) ⊕ K_T
    end if
end for
```

- To prevent transmitting redundant bits from the subfile $\tilde{W}_{d_j,\mathcal{S}}$ to user j, we need to ensure

$$\sum_{\mathcal{T}\subsetneq_{\phi}[K]\,:\,j\in\mathcal{T},\mathcal{T}\cap\mathcal{S}\neq\phi,\mathcal{T}\setminus\{j\}\subset\mathcal{S}} u_{\mathcal{S}}^{\mathcal{T}} \leq a_{\mathcal{S}},\ \forall\, j \notin \mathcal{S},$$
$$\forall\, \mathcal{S} \in \left\{ \tilde{\mathcal{S}} \subset [K] :\ 2 \leq |\tilde{\mathcal{S}}| \leq K-1 \right\}. \quad (11)$$

  Note that $u_{\mathcal{S}}^{\mathcal{T}}$ is defined only for $|\mathcal{T}| \leq |\mathcal{S}| + 1$.
- The delivery completion constraints are given by

$$\sum_{\mathcal{T}\subsetneq_{\phi}[K]\,:\,k\in\mathcal{T}} v_{\mathcal{T}} \geq 1 - \sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}}, \forall\, k \in [K]. \quad (12)$$

Note that if there are any missing pieces that have not been delivered to user k via any multicast transmission, then the server sends them via a unicast signal, $X_{k,\boldsymbol{d}}$, whose data contents are encrypted with the key K_k. Therefore, for given $\boldsymbol{a}$, the set of feasible delivery schemes that satisfies secure delivery, $\mathfrak{D}(\boldsymbol{a})$, is defined as

$$\mathfrak{D}(\boldsymbol{a}) = \Bigg\{ (\boldsymbol{v},\boldsymbol{u}) \Bigg| \sum_{\mathcal{T}\subsetneq_{\phi}[K]\,:\,k\in\mathcal{T}} v_{\mathcal{T}} \geq 1 - \sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}},\ \forall k \in [K],$$
$$\sum_{\mathcal{S}\in\mathcal{B}_j^{\mathcal{T}}} u_{\mathcal{S}}^{\mathcal{T}} = v_{\mathcal{T}},\ \forall \mathcal{T} \subsetneq_{\phi} [K], \forall j \in \mathcal{T},$$
$$\sum_{\mathcal{T}\in\subsetneq_{\phi}[K]\,:\,j\in\mathcal{T},\mathcal{T}\cap\mathcal{S}\neq\phi,\mathcal{T}\setminus\{j\}\subset\mathcal{S}} u_{\mathcal{S}}^{\mathcal{T}} \leq a_{\mathcal{S}},$$
$$\forall j \notin \mathcal{S}, \forall \mathcal{S} \in \left\{ \tilde{\mathcal{S}} \subset [K] :\ 2 \leq |\tilde{\mathcal{S}}| \leq K-1 \right\},$$
$$0 \leq u_{\mathcal{S}}^{\mathcal{T}} \leq a_{\mathcal{S}},\ \forall \mathcal{T} \subsetneq_{\phi} [K],\ \forall \mathcal{S} \in \mathcal{B}^{\mathcal{T}} \Bigg\}, \quad (13)$$

where the transmission and assignment variables are represented by the vectors $\boldsymbol{v}$ and $\boldsymbol{u}$, respectively. We summarize the placement and delivery procedures under secure delivery in Algorithm 1.

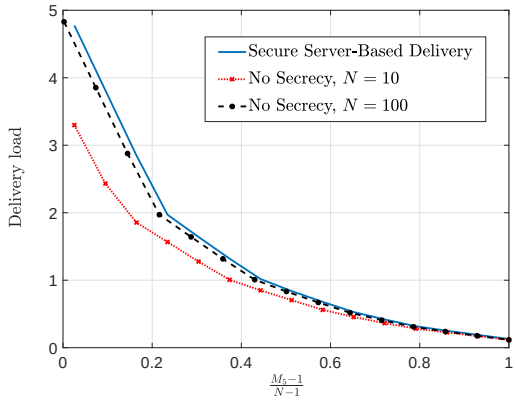

Fig. 3: The delivery loads for $K = 5$, and $m_k = 0.97\, m_{k+1}$.

Remark 1. *The secure delivery constraint is ensured due to the encryption of each signal with a one-time pad key [16] that has the same length as the signal.* ■

C. Optimization

Theorem 1. *The total delivery load under secure delivery can be minimized by solving the following optimization problem.*

$$R_S^*(\boldsymbol{m}^s) = \min_{\boldsymbol{a},\boldsymbol{u},\boldsymbol{v}} \quad \sum_{\mathcal{T}\subsetneq_\phi [K]} v_{\mathcal{T}} \tag{14a}$$

$$\text{subject to} \quad \boldsymbol{a} \in \mathfrak{A}(\boldsymbol{m}^s), \tag{14b}$$

$$(\boldsymbol{u},\boldsymbol{v}) \in \mathfrak{D}(\boldsymbol{a}), \tag{14c}$$

where $\mathfrak{A}(\boldsymbol{m}^s)$ is the set of placement schemes defined in (7) and $\mathfrak{D}(\boldsymbol{a})$ is the set of delivery schemes defined in (13). ■

Furthermore, the following corollary characterizes the achievable worst-case secure delivery load for $\sum_{j=1}^K m_j^s \le 1$.

Corollary 1. *For $\sum_{j=1}^K M_j \le K+N-1$, $N \ge K$, and $1 \le M_1 \le \cdots \le M_K$, the achievable worst-case secure delivery load under uncoded placement and delivery policy in $\mathfrak{D}$, is*

$$R_S^*(\boldsymbol{m}^s) = K - \sum_{j=1}^{K} (K-j+1) m_j^s, \tag{15}$$

where $m_j^s \triangleq \frac{M_j-1}{N-1}$ and the optimal parameters are

$$a_{\{j\}}^* = m_j^s, \quad v_{\{j\}}^* = 1 - \sum_{i=1}^{j-1} m_i^s - (K-j+1) m_j^s, \tag{16}$$

$$v_{\{i,j\}}^* = u_{\{i\}}^{*\{i,j\}} = u_{\{j\}}^{*\{i,j\}} = \min\{a_{\{i\}}^*, a_{\{j\}}^*\}. \tag{17}$$

In Fig. 3, we compare the secure delivery load obtained from optimization problem (14) with the non-secure delivery load [7], for $N = 10$ and $N = 100$. From Fig. 3, we observe that the gap between the secure and non-secure delivery loads decreases as N increases.

IV. Device-to-Device Secure Delivery

In this section, we focus on the scenario where the server must not participate in the delivery process, i.e., the users' requests must be satisfied via device-to-device communications while maintaining the secure delivery requirement.

A. Placement Phase

The main difference from the server-based delivery case, is that, the whole library must be stored at the end users to ensure the disengagement of the server from the delivery phase. In addition, each user not only caches the keys that are needed to decrypt its received signals but also stores the keys needed to encrypt its transmitted signals. In particular, each file W_n is partitioned into 2^K-1 subfiles, $\tilde{W}_{n,\mathcal{S}}, \mathcal{S} \subsetneq_\phi [K]$, such that $\tilde{W}_{n,\mathcal{S}}$ denotes a subset of W_n which is stored exclusively at the users in the set $\mathcal{S}$. The partitioning is symmetric over the files, i.e., $|\tilde{W}_{n,\mathcal{S}}| = a_{\mathcal{S}} F$ bits, $\forall n \in [N]$, where $a_{\mathcal{S}} \in [0,1]$.

In addition, the server generates the set of keys $K_{j\to\mathcal{T}}$ with length equal to $v_{j\to\mathcal{T}} F$ which will be specified later, where $j = 1, ..., K$ and $\mathcal{T} \subsetneq_\phi [K]\setminus\{j\}$. User k caches the keys $K_{k\to\mathcal{T}}$ for all $\mathcal{T}$ and $K_{j\to\mathcal{T}}$ for all $\mathcal{T}$ such that $k \in \mathcal{T}$ and $j \ne k$. More specifically, user k cache content is defined as

$$Z_k = \Big(\bigcup_{n\in[N]} \bigcup_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} \tilde{W}_{n,\mathcal{S}} \Big) \bigcup \Big(\bigcup_{\mathcal{T}} \{K_{k\to\mathcal{T}}\} \Big) \bigcup \Big(\bigcup_{j\in[K]\setminus\{k\},\mathcal{T}:k\in\mathcal{T}} \{K_{j\to\mathcal{T}}\} \Big). \tag{18}$$

The normalized number of bits in each cache is

$$N \sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}} + \sum_{\mathcal{T}\subset[K]\setminus\{k\}} v_{k\to\mathcal{T}} + \sum_{j\ne k} \sum_{\mathcal{T}\subset[K]:k\in\mathcal{T}} v_{j\to\mathcal{T}} = 1 + (N-1) \sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}} + \sum_{j\ne k} \sum_{\mathcal{T}\subset[K]:k\in\mathcal{T}} v_{j\to\mathcal{T}}. \tag{19}$$

The set of feasible uncoded placement schemes under secure delivery, $\mathfrak{A}(\boldsymbol{M},\boldsymbol{v})$, is defined by

$$\mathfrak{A}(\boldsymbol{M},\boldsymbol{v}) = \Bigg\{ \boldsymbol{a} \in [0,1]^{2^K} \Bigg| \sum_{\mathcal{S}\subsetneq_\phi[K]} a_{\mathcal{S}} = 1,\ \forall k \in [K], \ 1 + (N-1) \sum_{\mathcal{S}\subset[K]\,:\,k\in\mathcal{S}} a_{\mathcal{S}} + \sum_{j\ne k} \sum_{\mathcal{T}\subset[K]:k\in\mathcal{T}} v_{j\to\mathcal{T}} \le M_k \Bigg\}, \tag{20}$$

The first constraint follows from the fact the whole library can be reconstructed from the users' cache memories, and the second represents the cache size constraint at user k.

B. Delivery Phase

The delivery scheme is based on the framework provided in [9], with an additional step, i.e., encrypting the transmitted signal. In particular, the unicast signal transmitted by user j to user i should be formed and encrypted as follows

$$X_{j\to\{i\}} = \Big(\tilde{W}_{d_i,\{j\}} \bigcup \bigcup_{\mathcal{S}\subset[K]\setminus\{i\}\,:\,j\in\mathcal{S},|\mathcal{S}|\ge 2} W_{d_i,\mathcal{S}}^{j\to\{i\}} \Big) \oplus K_{j\to\{i\}}, \tag{21}$$

where $W_{d_i,\mathcal{S}}^{j\to\{i\}} \subset \tilde{W}_{d_i,\mathcal{S}}$ such that $|W_{d_i,\mathcal{S}}^{j\to\{i\}}| = u_{\mathcal{S}}^{j\to\{i\}} F$ bits.

User j constructs the multicast signal $X_{j\to\mathcal{T}}$, such that the piece intended for user $i \in \mathcal{T}$, which we denote by $W_{d_i}^{j\to\mathcal{T}}$, is stored at users $\{j\} \cup (\mathcal{T}\setminus\{i\})$. That is, $X_{j\to\mathcal{T}}$ is constructed using the side information at the sets

$$\mathcal{B}_i^{j\to\mathcal{T}} \triangleq \Big\{ \mathcal{S} \subset [K]\setminus\{i\} : \{j\} \cup (\mathcal{T}\setminus\{i\}) \subset \mathcal{S} \Big\}, \tag{22}$$

which represents the subfiles stored at users $\{j\} \cup (\mathcal{T} \setminus \{i\})$ and not available at user $i \in \mathcal{T}$. In turn, we have

$$X_{j\to\mathcal{T}} = \oplus_{i\in\mathcal{T}} \left(\bigcup_{\mathcal{S}\in\mathcal{B}_i^{j\to\mathcal{T}}} W_{d_i,\mathcal{S}}^{j\to\mathcal{T}} \right) \oplus K_{j\to\mathcal{T}}. \tag{23}$$

The set of feasible linear secure delivery schemes [9], $\mathfrak{D}(\boldsymbol{a})$, is defined by

$$\begin{aligned}
\mathfrak{D}(\boldsymbol{a}) = \Bigg\{ (\boldsymbol{v},\boldsymbol{u}) \Bigg| & v_{j\to\{i\}} = a_{\{j\}} + \sum_{\mathcal{S}\subset[K]\setminus\{i\} \,:\, j\in\mathcal{S}, |\mathcal{S}|\geq 2} u_{\mathcal{S}}^{j\to\{i\}}, \\
& \qquad \forall j \in [K],\, \forall\, i \neq j, \\
& v_{j\to\mathcal{T}} = \sum_{\mathcal{S}\in\mathcal{B}_i^{j\to\mathcal{T}}} u_{\mathcal{S}}^{j\to\mathcal{T}},\ \forall j \in [K], \\
& \qquad \forall\, \mathcal{T} \subsetneq_{\phi} [K]\setminus\{j\},\, \forall i \in \mathcal{T}, \\
& \sum_{j\in\mathcal{S}} \sum_{\mathcal{T}\subset\{i\}\cup(\mathcal{S}\setminus\{j\}) \,:\, i\in\mathcal{T}} u_{\mathcal{S}}^{j\to\mathcal{T}} \leq a_{\mathcal{S}}, \\
& \forall\, i \notin \mathcal{S}, \forall\, \mathcal{S} \subset [K], \quad \text{s.t. } 2 \leq |\mathcal{S}| \leq K-1, \\
& \sum_{j\in[K]\setminus\{k\}} \sum_{\mathcal{T}\subset[K]\setminus\{j\} \,:\, k\in\mathcal{T}} v_{j\to\mathcal{T}} \geq 1 - \sum_{\mathcal{S}\subset[K] \,:\, k\in\mathcal{S}} a_{\mathcal{S}}, \forall\, k, \\
& 0 \leq u_{\mathcal{S}}^{j\to\mathcal{T}} \leq a_{\mathcal{S}}, \forall j \in [K], \\
& \qquad \forall\, \mathcal{T} \subsetneq_{\phi} [K] \setminus \{j\}, \forall\, \mathcal{S} \in \mathcal{B}^{j\to\mathcal{T}} \Bigg\},
\end{aligned} \tag{24}$$

where $\mathcal{B}^{j\to\mathcal{T}} \triangleq \bigcup_{i\in\mathcal{T}} \mathcal{B}_i^{j\to\mathcal{T}}$. Note that first constraint follows from the structure of the unicast signals in (21), and the second follows from the structure of the multicast signals in (23).

C. Optimization

We have the following parameterization for the optimum of the class of caching schemes under consideration.

Theorem 2. *Given $N \geq K$, and $\boldsymbol{M}$, the minimum worst-case D2D delivery load assuming uncoded placement and linear secure delivery, is characterized by solving*

$$\min_{\boldsymbol{a},\boldsymbol{u},\boldsymbol{v}} \quad R_{D2D} = \sum_{j=1}^{K} \sum_{\mathcal{T}\subsetneq_{\phi}[K]\setminus\{j\}} v_{j\to\mathcal{T}} \tag{25a}$$

$$\text{subject to} \quad \boldsymbol{a} \in \mathfrak{A}(\boldsymbol{M},\boldsymbol{v}), \tag{25b}$$

$$(\boldsymbol{u},\boldsymbol{v}) \in \mathfrak{D}(\boldsymbol{a}), \tag{25c}$$

where $\mathfrak{A}(\boldsymbol{M},\boldsymbol{v})$ is the set of uncoded placement schemes defined in (20) and $\mathfrak{D}(\boldsymbol{a})$ is the set of feasible linear delivery schemes defined by (24). ■

In Fig. 4, we compare the secure D2D delivery load obtained from optimization problem (25) with the non-secure D2D delivery load [9], for $N = 10$ and $N = 100$. Similarly, we observe that the gap between the secure and non-secure delivery loads decreases with N.

V. Conclusions

In this work, we have provided coded caching schemes which ensure secure delivery of the requested files in systems where the users have heterogeneous cache sizes. We have considered both server-based and device-to-device secure delivery where the users are served via device-to-device communications only. The secrecy is ensured by using one-time padding and the delivery load is minimized by solving a linear program optimizing the parameters of the caching scheme. For server-based secure delivery, the local caching gain becomes $\frac{M_k-1}{N-1}$, while in the case of D2D-based secure delivery the local caching gain is further reduced due to the need of storing keys to encrypt the transmitted signals. Our work shows that the cost of imposing the secure delivery requirement is almost negligible in systems with large library.

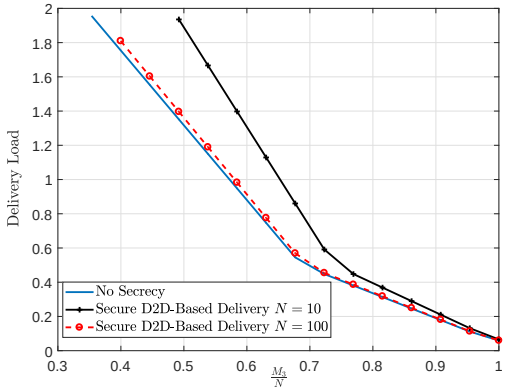

Fig. 4: The D2D delivery loads for $K = 3$, and $m_k = 0.97\, m_{k+1}$.

References

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

[2] K. Wan, D. Tuninetti, and P. Piantanida, "On the optimality of uncoded cache placement," in *IEEE ITW*, September 2016.

[3] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "The exact rate-memory tradeoff for caching with uncoded prefetching," *IEEE Trans. Info. Theory*, vol. 64, no. 2, pp. 1281–1296, 2018.

[4] Z. Chen, "Fundamental limits of caching: Improved bounds for small buffer users," *arXiv:1407.1935*, 2014.

[5] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "Characterizing the rate-memory tradeoff in cache networks within a factor of 2," *IEEE Trans. Info. Theory*, vol. 65, no. 1, pp. 647–663, 2019.

[6] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Centralized coded caching with heterogeneous cache sizes," in *IEEE WCNC*, March 2017.

[7] ——, "Coded caching for heterogeneous systems: An optimization perspective," *accepted at IEEE Trans. Commun., arXiv:1810.08187*, 2018.

[8] ——, "Device-to-device coded caching with heterogeneous cache sizes," in *IEEE ICC*, May 2018.

[9] ——, "Device-to-device coded caching with distinct cache sizes," *submitted to IEEE Trans. Commun., arXiv:1903.08142*, 2019.

[10] 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.

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

[12] A. A. Zewail and A. Yener, "Coded caching for resolvable networks with security requirements," in *IEEE CNS*, October 2016.

[13] M. Bahrami, M. A. Attia, R. Tandon, and B. Vasić, "Towards the exact rate-memory trade-off for uncoded caching with secure delivery," in *IEEE Allerton*, October 2017.

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

[15] ——, "Combination networks with or without secrecy constraints: The impact of caching relays," *IEEE Journ. Sel. Areas in Commun.*, vol. 36, no. 7, pp. 1140–1152, 2018.

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