Fundamental Limits of Caching in D2D Networks With Secure Delivery

Zohaib Hassan Awan and Aydin Sezgin
Institute of Digital Communication Systems,
Ruhr-Universität Bochum, 44780 Bochum, Germany.
Email: {zohaib.awan, aydin.sezgin}@rub.de

***Abstract*—We study the problem of secure transmission over a caching D2D network. In this model, end users can prefetch a part of popular contents in their local cache. Users make arbitrary requests from the library of available files and interact with each other to deliver requested contents from the local cache to jointly satisfy their demands. The transmission between the users is wiretapped by an external eavesdropper from whom the communication needs to be kept secret. For this model, by exploiting the flexibility offered by the local cache storage, we establish a coding scheme that not only conforms to the demands of all users but also delivers the contents securely. In comparison to the insecure caching schemes, the coding scheme that we develop in this work illustrates that for large number of files and users, the loss incurred due to the imposed secrecy constraints is insignificant. We illustrate our result with the help of some examples.**

I. Introduction

High data demand in peak hours emanates an important issue of congestion in communication networks. A key technique to avoid congestion is through caching. In caching, popular contents are prefetched in the local storage in off-peak hours. At the time of high traffic, the users requests are partially served through the information stored in the local cache; and, so it reduces network load. The caching problem comprises of two phases, i.e., content placement or storage phase and delivery or decoding phase. These two aspects of caching are *separately* studied in literature, for instance in [1]–[3]. In [1], [2], the authors studied a model in which the delivery phase is constrained to fixed unicast or multicast transmissions. With this choice of communication the content placement in the cache is then optimized. As opposed to the model in [1], recently in [3] the authors studied the delivery phase of caching problem with fixed cache contents. In all these works, the two facets of caching are studied independently. Maddah-Ali *et al.* in [4] introduced an information-theoretic formulation in which the two phases of caching are studied *jointly*. In [4], by taking collective storage size of the network into account, the authors propose a new coding scheme that jointly constructs the cache placement and delivery phases. In this scheme, the files are distributed over the network in a non-trivial manner such that in the delivery phase with the help of multicast coded transmissions from the central server, all requests are served completely. The scheme in [4] is extended to study a variety of models, for example, D2D networks in [5], [6], in the context of decentralized storage in [7] and for secure communication in [8].

In this work, we study a caching D2D network as shown in Figure 1. In the cache placement phase end users prefetch the part of files in their local cache from the set of files available in the library of the central server. Users make arbitrary file requests. In the delivery phase, there is no connection between the users and server. Nodes interact with each other through multicast coded transmissions to deliver contents and, in doing so, satisfy the demands of each other. The interaction between the nodes is wiretapped by an external eavesdropper form whom the transmission needs to be kept secret. The performance of this model is measured by the notion of memory-rate trade-off, that captures the secure transmission rate of the network as a function of cache memory size. For this model, we establish a coding scheme that not only conforms to the demands of all users but also delivers them securely. The secrecy in the system is obtained in the spirit of Shannon's one-time-pad scheme [9], where a set of keys are disseminated across the users and stored in end users cache, such that in each interaction a unique key is used. It is interesting to note that in comparison to the insecure caching scheme [10], the coding scheme that we construct in this work illustrates that for large number of files and users, the loss incurred due to the imposed secrecy constraints is insignificant. For a specific example, we also provide a sketch of the lower bound on the D2D caching model with secure delivery. The general lower bound follows along similar lines and due to space limitations is omitted for brevity.

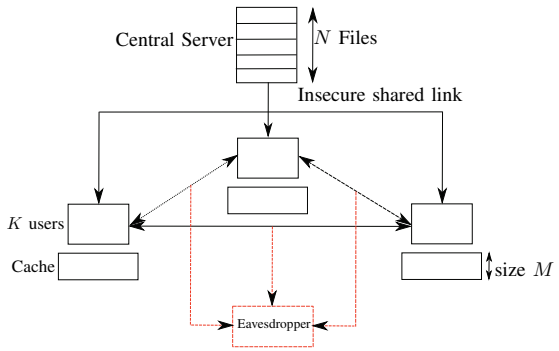

Fig. 1. System model for caching in D2D network with security constraints. In the above figure $K = 3$, $N = 5$ and $M = 1$.

II. System Model and Definitions

We consider a network which consists of a central server containing a database of N files, K users that are connected to the central server over a public noiseless link and an eavesdropper as shown in Figure 1. We denote the files available in the database of server as $(W_1, \ldots, W_N)$. In addition to this, the k-th user contains a cache memory Z_k with size M where $M \in [1, N]$. This communication model operates in two phases. In the first phase, which we refer to as cache placement phase,

This work is supported by the German Research Foundation, Deutsche Forschungsgemeinschaft (DFG), Germany, under grant SE 1697/11.

978-1-4673-6305-1/15/$31.00 ©2015 IEEE

all nodes have complete access to the database of the central server and can fill in their respective cache from the available files $(W_1, \ldots, W_N)$. In this phase, the bottle neck is the cache memory size. After the cache placement phase, each user places an arbitrary request from the set of available files in the server. The demand vector $\mathbf{d} := (d_1, \ldots, d_K)$ captures the request by all user, where $d_k \in \{W_1, \ldots, W_N\}$ for $k \in [K]$. The request made by each user is furnished in the second phase, which we refer to as delivery phase. In this phase, users have no link to the server and they communicate with each other such that with the help of cached data stored locally and the transmission from other users, the k-th user is able to construct the requested file, for $k \in \{1, \ldots, K\}$. We assume that the transmission by each user is heard by all elements in the network. We build the caching model, along the lines of [10] by allowing *asynchronous content reuse*, so as to avoid any form of uncoded multicasting gain [4]. In particular, we assume that 1) each file is composed of L packets, i.e., $W_n := (W_n^1, \ldots, W_n^L)$, and 2) each user is allowed to cache an arbitrary segment of length L' of the requested file W_n, for $L' \neq L$ for $n = 1, \ldots, N$. Subsequently, for an arbitrarily large L, and finite L' and N, it is always possible that each user requests a unique segment of the same file. The demand by the k-th user can then be concisely written as $d_k \in \{d_k^1, \ldots, d_k^L\}$, where d_k^l denotes that the l-th subfile is required by the k-th user. The communication between the users is wiretapped by an external passive eavesdropper.

We now formally define the system model as follows. Let $\{W_n^l\}_{l=1}^L$ be independent random variables uniformly drawn over $\{1, \ldots, 2^F\}$, for some $F \in \mathbb{N}$. The size of each packet, W_n^l, is F bits.

Definition 1 (Cache Placement): In the cache placement phase, the users have full access to the database of the server and map the files onto the cache memory. In particular, the k-th random caching function

$$\phi_k \; : \; [2^F]^{NL} \to [2^F]^{ML}$$

maps the files $\{W_1, \ldots, W_N\}$ into the cache content

$$Z_k = \phi_k(W_1^l, \ldots, W_N^l, l = 1, \ldots, L) \text{ for } k \in \{1, \ldots, K\}. \quad (1)$$

Thus, Z_k can store a total of MFL bits.

At the end of cache placement phase, each user places an arbitrary request of file from the library of files available in the database of server.

Definition 2 (Delivery): The delivery phase consists of a set of encoding and decoding functions at each user. The random encoder

$$\psi_k \; : \; [2^F]^{ML} \times [K] \to [2^F]^{R_k L'}$$

maps the locally stored cached information and demand vector to the input signal as

$$\{X_{k,(d_1,\ldots,d_K)} = \psi_k(Z_k, d_1, \ldots, d_K)\}_{k=1}^K \quad (2)$$

for $(d_1, \ldots, d_K) \in [N]^K$, where R_k denotes the transmission rate of user k. The decoder

$$\mu_{k,(d_1,\ldots,d_K)} : [2^F]^{ML} \times [K] \times \{[2^F]^{R_i L'}\}_{i=\{1,\ldots,K\}\setminus k} \to [2^F]^L$$

maps the input signals from the other users and with the help of cached information at the k-th user and demand vector, estimates the requested file as

$$\hat{W}_{k,(d_1,\ldots,d_K)} = \mu_{k,(d_1,\ldots,d_K)}(Z_k, \{X_{i,(d_1,\ldots,d_K)}\}_{i=\{1,\ldots,K\}\setminus k}, d_1, \ldots, d_K) \quad (3)$$

for W_{d_k} of user $k \in [K]$.

Definition 3: The probability of error is defined as

$$\max_{(d_1,\ldots,d_K)\in[N]^K} \max_{k\in[K]} \Pr\{\hat{W}_{k,(d_1,\ldots,d_K)} \neq W_{d_k}\}. \quad (4)$$

For convenience, we define $R_s = \sum_{k=1}^K R_k$.

Definition 4: The memory-rate pair (M, R_s) is said to be securely achievable if $\forall\ \epsilon > 0$ and $F \to \infty$, there exists a secure caching scheme satisfying following

1) Reliability condition:

$$\max_{(d_1,\ldots,d_K)\in[N]^K} \max_{k\in[K]} \Pr\{\hat{W}_{k,(d_1,\ldots,d_K)} \neq W_{d_k}\} \leq \epsilon \quad (5)$$

2) Perfect secrecy condition:

$$\max_{(d_1,\ldots,d_K)\in[N]^K} I(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)}; W_1, \ldots, W_N) \leq \epsilon. \quad (6)$$

Definition 5: We define the secure memory-rate trade-off as

$$R_s^\star(M) \triangleq \inf\{R_s : (M, R_s) \text{ is securely achievable}\}. \quad (7)$$

III. Main Results and Discussion

The following theorem gives an achievable secure caching scheme that yields an upper bound on the memory-rate trade-off $R_s^\star(M)$ for the caching D2D network shown in Figure 1.

Theorem 1: For N files and K users, each with a cache size of $M \in \frac{(N-1)t}{K} + \frac{1}{t} - \frac{1}{K} + 1$, for $t \in \{1, \ldots, K\}$, the following rate is securely achievable

$$R^\star(M) \leq \frac{2K(N-1)}{K(M-1)+1+\sqrt{(K(M-1)+1)^2 - 4K(N-1)}} - 1. \quad (8)$$

Moreover, the lower convex envelope of these points is also achievable.

Proof: The proof of Theorem 1 appears in Appendix A. □

Remark 1: The coding scheme in Theorem 1 follows by jointly designing the cache placement and delivery phases by taking security constraints into account. The cache of each user is appropriately filled, so that by coded multicast transmissions each user can construct the requested file. Secrecy is obtained through Shannon's one-time-pad scheme [9] by generating secret keys in the system. As opposed to a related model in [8], where authors study a centralized secure system that transmits input signals from the central server and end users only act as receivers, each node in the model that we study plays two different roles, 1) it sends input signals to satisfy the requests of other users and in addition to this 2) it also receives signals from other users to construct its own desired file. We generate two sets of keys, such that, one set is used to encode the transmission to other users and the other is used to decode the input signals received from other users that helps to construct the requested files. Thus, by using multiple sets of keys during transmission and reception of signals, we make the system more robust to failures.

Example: We now illustrate the main elements of the coding scheme with the help of a simple example. We consider the

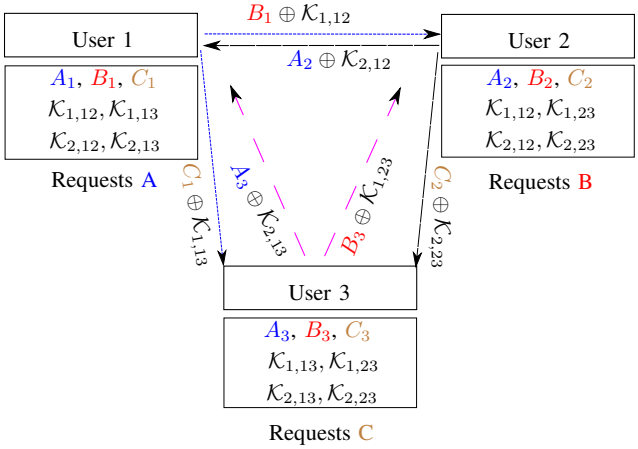

Fig. 2. Secure caching scheme with $N = K = 3$ and $M = \frac{7}{3}$.

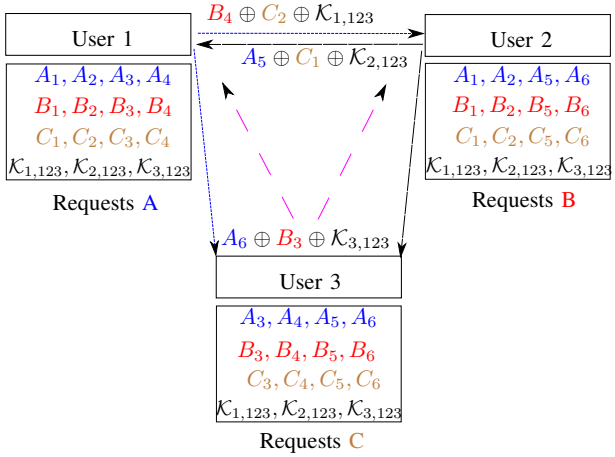

Fig. 3. Secure caching scheme with $N = K = 3$ and $M = \frac{5}{2}$.

three user case in which $K = 3$. For simplicity, we assume that there are three files present in the central server, i.e., $N = 3$ and each file is composed of only one packet, i.e., $L = 1$. For convenience, we name these files as A, B, and C. For this system configuration the allowed cache sizes at each user are $M \in \{\frac{7}{3}, \frac{5}{2}, 3\}$. First consider the extreme case $M = N$. When $M = N$ each user can apriori cache all the files available in the central server. Thus, no transmission is required in the delivery phase and (M, R_s) pair $(3, 0)$ is securely achievable. We now focus our attention to the remaining two points when $M = \frac{7}{3}$ and $\frac{5}{2}$.

Consider the case when $M = \frac{7}{3}$. The coding scheme in this case is illustrated in Figure 2. Each file is first partitioned into 3 subfiles each of size $F/3$ such that $A = (A_1, A_2, A_3)$, $B = (B_1, B_2, B_3)$ and $C = (C_1, C_2, C_3)$. In addition to this, we generate 6 unique keys $\{\mathcal{K}_{j,12}, \mathcal{K}_{j,13}, \mathcal{K}_{j,23}\}$ for $j = 1, 2$. The size of each key is $F/3$. In the cache placement phase $\mathcal{K}_{j,mn}$ refers to the key being placed in the memory of both users m and n, for all j. Thus, the cached information at each user is given by

$$\begin{aligned} Z_1 &= \{A_1, B_1, C_1, \mathcal{K}_{1,12}, \mathcal{K}_{1,13}, \mathcal{K}_{2,12}, \mathcal{K}_{2,13}\} \\ Z_2 &= \{A_2, B_2, C_2, \mathcal{K}_{1,12}, \mathcal{K}_{1,23}, \mathcal{K}_{2,12}, \mathcal{K}_{2,23}\} \\ Z_3 &= \{A_3, B_3, C_3, \mathcal{K}_{1,13}, \mathcal{K}_{1,23}, \mathcal{K}_{2,13}, \mathcal{K}_{2,23}\}. \end{aligned}$$

Notice that each user's cache contains three subfiles of total size $M_D = 3 \times \frac{1}{3} = 1$ and four keys of total size $M_K = \frac{4}{3}$, such that $M = M_D + M_K = 1 + 4/3 = 7/3$. This satisfies the memory size constraint at each user. Next, we consider the worst case in which each user requests a different file, i.e., $(d_1, d_2, d_3) = (A, B, C)$. In this phase, the users communicate with each other to satisfy the requests of each other by transmitting

$$\begin{aligned} X_{1,(A,B,C)} &= \{B_1 \oplus \mathcal{K}_{1,12}, C_1 \oplus \mathcal{K}_{1,13}\} \\ X_{2,(A,B,C)} &= \{A_2 \oplus \mathcal{K}_{2,12}, C_2 \oplus \mathcal{K}_{2,23}\} \\ X_{3,(A,B,C)} &= \{A_3 \oplus \mathcal{K}_{2,13}, B_3 \oplus \mathcal{K}_{1,23}\}. \end{aligned} \tag{9}$$

At the end of transmission, with the help of side information which in this case is the secret key stored locally, each user can easily decode the desired subfiles. This yields a transmission rate of $R_s = 6 \times \frac{1}{3} = 2$. The information leaked to the eavesdropper is bounded by $I(X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}; A, B, C) = 0$. Thus, the rate pair $(M, R_s) = (\frac{7}{3}, 2)$ is securely achievable.

Next, we consider the case when $M = \frac{5}{2}$. The coding scheme in this case is given in Figure 3. We first partition each file into 6 subfiles of size $F/6$ each as $A = (A_1, \ldots, A_6)$, $B = (B_1, \ldots, B_6)$ and $C = (C_1, \ldots, C_6)$. Also, generate 3 unique keys $\{\mathcal{K}_{j,123}\}$ for $j = 1, 2, 3$, where the size of each key is $F/6$ and is shared between all users. At the end of cache placement phase, the cache of each user contains

$$\begin{aligned} Z_1 &= \{A_t, B_t, C_t, \mathcal{K}_{1,123}, \mathcal{K}_{2,123}, \mathcal{K}_{3,123}\} \text{ for } t = 1, 2, 3, 4 \\ Z_2 &= \{A_t, B_t, C_t, \mathcal{K}_{1,123}, \mathcal{K}_{2,123}, \mathcal{K}_{3,123}\} \text{ for } t = 1, 2, 5, 6 \\ Z_3 &= \{A_t, B_t, C_t, \mathcal{K}_{1,123}, \mathcal{K}_{2,123}, \mathcal{K}_{3,123}\} \text{ for } t = 3, 4, 5, 6. \end{aligned}$$

Each user's cache contains 12 subfiles of total size $M_D = 12 \times \frac{1}{6} = 2$ and 3 keys of total size $M_K = \frac{3}{6}$, such that $M = M_D + M_K = 2 + 1/2 = 5/2$. In the delivery phase the transmission by each user is given by

$$\begin{aligned} X_{1,(A,B,C)} &= B_4 \oplus C_2 \oplus \mathcal{K}_{1,123} \\ X_{2,(A,B,C)} &= A_5 \oplus C_1 \oplus \mathcal{K}_{2,123} \\ X_{3,(A,B,C)} &= A_6 \oplus B_3 \oplus \mathcal{K}_{3,123}. \end{aligned} \tag{10}$$

At the end of transmission, with the help of side information available in each user's cache and the signals from other users, each user can easily construct the required file. This yields a transmission rate of $R_s = 3 \times \frac{1}{6} = \frac{1}{2}$. It can be readily shown that the information leaked to the eavesdropper is bounded by $I(X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}; A, B, C) = 0$, that yields the memory-rate pair $(M, R_s) = (\frac{5}{2}, \frac{1}{2})$.

We now provide the sketch of the proof, which gives a lower bound on the memory-rate trade-off for the D2D network with security constraints. The lower bound generalize the one developed in [10] in the context of a similar network by taking secrecy constraints into account. Let $R_{k,\mathbf{d}}$ denote the transmission rate by the k-th user that satisfies the demand vector $\mathbf{d} := (d_1, d_2, d_3)$. Since each user can request for an arbitrary file from the set of available files, the lower bound needs to exhaust all possible set of demands. In this example, the possible requests are $(d_1, d_2, d_3) = (A, B, C)$, $(d_1, d_2, d_3) = (B, C, A)$ and $(d_1, d_2, d_3) = (C, A, B)$. Figure 4, concisely illustrates the compound extension of these requests. First consider the demand vector $(d_1, d_2, d_3) = (A, B, C)$. With the help of input signals $(X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)})$ and cached information (Z_1, Z_2, Z_3), all files can be recovered with

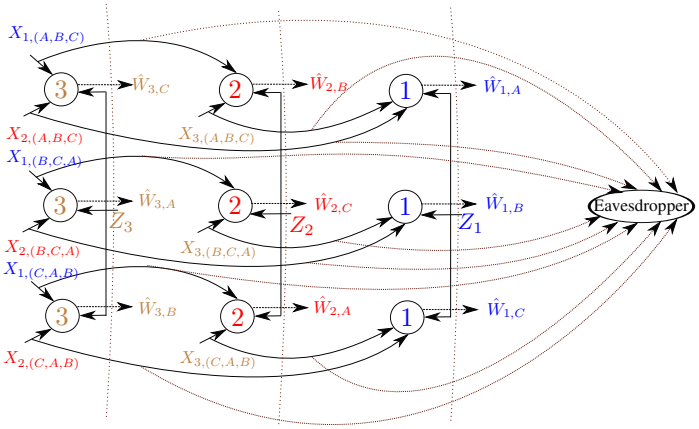

Fig. 4. Compound extension of network with request vectors, (A, B, C), (B, C, A) and (C, A, B), with $K = N = 3$.

arbitrarily small error probability. This implies that

$$H(A, B, C| \\ X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}, Z_1, Z_2, Z_3) \leq \epsilon. \quad (11)$$

Due to the broadcast nature of the communication, the input signals are wiretapped by the external eavesdropper. As mentioned before, for secure transmission the perfect secrecy constraint (6) implies that

$$I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}) \leq \epsilon. \quad (12)$$

We begin the proof as follows.

$$\begin{aligned}
3F &\leq H(A, B, C) \\
&= I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}, \\
&\quad Z_1, Z_2, Z_3) + H(A, B, C| \\
&\quad X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}, Z_1, Z_2, Z_3) \\
&\overset{(a)}{\leq} I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}, \\
&\quad Z_1, Z_2, Z_3) + \epsilon \\
&= I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}) \\
&\quad + I(A, B, C; Z_1, Z_2, Z_3| \\
&\qquad X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}) + \epsilon \\
&\overset{(b)}{\leq} I(A, B, C; Z_1, Z_2, Z_3| \\
&\quad X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}) + \epsilon' \\
&\leq H(Z_1, Z_2, Z_3|X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}) + \epsilon' \\
&\leq H(Z_1, Z_2, Z_3) + \epsilon' \\
&\leq 3MF + \epsilon' \qquad (13)
\end{aligned}$$

where (a) follows from (11) and (b) follows from (12). Continuing from (13), we get

$$M \geq 1 - \frac{\epsilon'}{3F}. \quad (14)$$

Then, by taking $\lim \epsilon' \to 0$ in (14), we get $M \geq 1$.

Next, we focus our attention on user 3 and consider the cut that separates $(X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{1,(B,C,A)}, X_{2,(B,C,A)}, X_{1,(C,A,B)}, X_{2,(C,A,B)}, Z_3)$ and $(\hat{W}_{3,C}, \hat{W}_{3,A}, \hat{W}_{3,B})$. This implies that

$$H(A, B, C|X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{1,(B,C,A)}, X_{2,(B,C,A)}, \\ X_{1,(C,A,B)}, X_{2,(C,A,B)}, Z_3) \leq \epsilon. \quad (15)$$

In addition to this, the following constraint

$$I(A, B, C; X_{1,(C,A,B)}, X_{2,(C,A,B)}, X_{3,(C,A,B)}) \leq \epsilon \quad (16)$$

needs to be satisfied for secure communication.
We proceed as follows.

$$\begin{aligned}
3F &\leq H(A, B, C) \\
&= I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{1,(B,C,A)}, \\
&\quad X_{2,(B,C,A)}, X_{1,(C,A,B)}, X_{2,(C,A,B)}, Z_3) + H(A, B, C| \\
&\quad X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{1,(B,C,A)}, X_{2,(B,C,A)}, \\
&\quad X_{1,(C,A,B)}, X_{2,(C,A,B)}, Z_3) \\
&\overset{(a)}{\leq} I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{1,(B,C,A)}, \\
&\quad X_{2,(B,C,A)}, X_{1,(C,A,B)}, X_{2,(C,A,B)}, Z_3) + \epsilon \\
&= I(A, B, C; X_{1,(C,A,B)}, X_{2,(C,A,B)}) \\
&\quad + I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, \\
&\quad X_{1,(B,C,A)}, X_{2,(B,C,A)}, Z_3|X_{1,(C,A,B)}, X_{2,(C,A,B)}) + \epsilon \\
&\overset{(b)}{\leq} I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, \\
&\quad X_{1,(B,C,A)}, X_{2,(B,C,A)}, Z_3|X_{1,(C,A,B)}, X_{2,(C,A,B)}) \\
&\quad \underbrace{- I(A, B, C; X_{3,(C,A,B)}|X_{1,(C,A,B)}, X_{2,(C,A,B)})}_{\geq 0} + \epsilon' \\
&\overset{(c)}{\leq} H(X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{1,(B,C,A)}, X_{2,(B,C,A)}, Z_3| \\
&\quad X_{1,(C,A,B)}, X_{2,(C,A,B)}) + \epsilon' \\
&\leq H(X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{1,(B,C,A)}, X_{2,(B,C,A)}, Z_3) \\
&\quad + \epsilon' \\
&\leq (R_{1,(A,B,C)} + R_{2,(A,B,C)} + R_{1,(B,C,A)} + R_{2,(B,C,A)})F \\
&\quad + MF + \epsilon' \qquad (17)
\end{aligned}$$

where (a) follows from (15), (b) follows from (16) and (c) follows due to non-negativity of mutual information and the fact that conditioning reduces entropy.

Similarly, for user 2 we consider the cut that separates $(X_{1,(A,B,C)}, X_{3,(A,B,C)}, X_{1,(B,C,A)}, X_{3,(B,C,A)}, X_{1,(C,A,B)}, X_{3,(C,A,B)}, Z_2)$ and $(\hat{W}_{2,B}, \hat{W}_{2,C}, \hat{W}_{2,A})$; and, following reliability and secrecy constraints

$$\begin{aligned}
&H(A, B, C|X_{1,(A,B,C)}, X_{3,(A,B,C)}, X_{1,(B,C,A)}, X_{3,(B,C,A)}, \\
&\qquad X_{1,(C,A,B)}, X_{3,(C,A,B)}, Z_2) \leq \epsilon \\
&I(A, B, C; X_{1,(B,C,A)}, X_{2,(B,C,A)}, X_{3,(B,C,A)}) \leq \epsilon. \qquad (18)
\end{aligned}$$

Then, following steps similar to in (17), we get

$$\begin{aligned}
3F \leq &(R_{1,(A,B,C)} + R_{3,(A,B,C)} + R_{1,(C,A,B)} + R_{3,(C,A,B)})F \\
&+ MF + \epsilon' \qquad (19)
\end{aligned}$$

Finally for user 1, by considering the cut that separates $(X_{2,(A,B,C)}, X_{3,(A,B,C)}, X_{2,(B,C,A)}, X_{3,(B,C,A)}, X_{2,(C,A,B)}, X_{3,(C,A,B)}, Z_1)$ and $(\hat{W}_{1,A}, \hat{W}_{1,B}, \hat{W}_{1,C})$; and, following reliability and secrecy constraints

$$\begin{aligned}
&H(A, B, C|X_{2,(A,B,C)}, X_{3,(A,B,C)}, X_{2,(B,C,A)}, X_{3,(B,C,A)}, \\
&\qquad X_{2,(C,A,B)}, X_{3,(C,A,B)}, Z_1) \leq \epsilon \\
&I(A, B, C; X_{1,(A,B,C)}, X_{2,(A,B,C)}, X_{3,(A,B,C)}) \leq \epsilon, \qquad (20)
\end{aligned}$$

we get

$$\begin{aligned}
3F \leq &(R_{2,(B,C,A)} + R_{3,(B,C,A)} + R_{2,(C,A,B)} + R_{3,(C,A,B)})F \\
&+ MF + \epsilon'. \qquad (21)
\end{aligned}$$

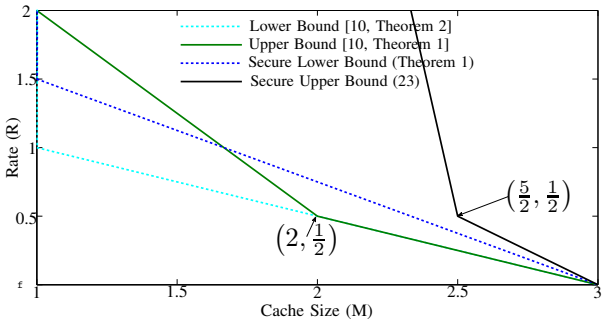

Fig. 5. Secure and insecure bounds for caching in D2D network with $N = K = 3$.

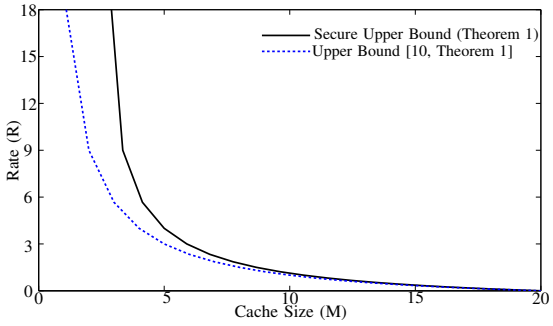

Fig. 6. Secure and insecure upper bounds for caching in D2D network with $N = K = 20$.

Then, by combining (17), (19) and (21) and noticing that 1) the worst case sum rate $R = R_{1,\mathbf{d}} + R_{2,\mathbf{d}} + R_{3,\mathbf{d}}$ must yields the same rate, independent from any demand vector $(\mathbf{d})$ and 2) due to the symmetry of transmission rates for all requests at each user, we get

$$\begin{aligned}&(\underbrace{R_{1,(A,B,C)} + R_{2,(A,B,C)} + R_{3,(A,B,C)}}_{R})F + \\ &(\underbrace{R_{1,(B,C,A)} + R_{2,(B,C,A)} + R_{3,(B,C,A)}}_{R})F + \\ &(\underbrace{R_{1,(C,A,B)} + R_{2,(C,A,B)} + R_{3,(C,A,B)}}_{R})F + \\ &(R_{1,(A,B,C)} + \underbrace{R_{2,(B,C,A)}}_{R_{2,(A,B,C)}} + \underbrace{R_{3,(C,A,B)}}_{R_{3,(A,B,C)}})F + 3MF \\ &\qquad\qquad \geq 9F + 3\epsilon'. \quad (22)\end{aligned}$$

Hence $4RF \geq 9F - 3MF + 3\epsilon'$, which yields

$$R^\star(M) \geq \frac{3}{4}\big(3 - M\big) + \frac{3\epsilon'}{4F}. \quad (23)$$

Then by taking $\epsilon' \to 0$, we obtain $R^\star(M) \geq \frac{3}{4}\big(3 - M\big)$ for $1 \leq M \leq N$.

IV. Numerical Examples

In this section, we illustrate our results with the help of some numerical examples. Figure 5 shows the lower and upper bounds for caching in D2D network with security constraints as a function of cache size (M) with $N = K = 3$. For comparison reasons, we also plot the lower and upper bounds for a similar bound without secrecy constraints in [10]. The visible gap between the two regions shows the loss incurred due to the imposed secrecy constraints. Next, in Figure 6, we compare the upper bounds with and without security constraints for $N = K = 20$. It can be easily seen from the Figure 6 that as the cache size increases the loss incurred with secure coding scheme decreases. With a very large value of M the secure and insecure bounds eventually coincide. This result shows that there is negligible performance loss in terms of secrecy constraints and secrecy for the considered system is available almost *free*.

Appendix A
Proof of Theorem 1

We now provide the coding scheme which gives the upper bound in Theorem 1. Let $M \in \frac{(N-1)t}{K} + \frac{1}{t} - \frac{1}{K} + 1$, for $t \in \{1, \ldots, K\}$ and $M \leq N$. The cache memory at each user can then be written as

$$M = \underbrace{\frac{Nt}{K}}_{M_D} + \underbrace{\left(1 + \frac{1}{t}\right)\left(1 - \frac{t}{K}\right)}_{M_K} \quad (24)$$

where M_D corresponds to the part of memory which contains cached data and M_K is designated to store keys. Through straightforward algebra, from (24), we obtain

$$t = \frac{K(M-1) + 1 + \sqrt{(K(M-1)+1)^2 - 4K(N-1)}}{2(N-1)}. \quad (25)$$

Without the loss of generality, in the rest of proof we assume that each file in the server is composed of only one packet, i.e., $L = 1$ and consider the case such that each user demands a unique file. The scheme which we construct below, extends readily to any value of L.

1) Cache Placement: In the cache placement phase, each file W_n for $n \in \{1, \ldots, N\}$ is split into $t\binom{K}{t}$ non-overlapping subfiles of equal size as

$$W_n = (W_{n,(j,\tau)} \ : \ \tau \subseteq \{1, \ldots, K\}, |\tau| = t, j = \{1, \ldots, t\})$$

where the size of each subfile is $F/t\binom{K}{t}$ bits. For each n, the subfile $W_{n,(j,\tau)}$ is placed in the cache of user k if $k \in \tau$, for $j = 1, \ldots, t$. For a given j and $k \in \tau$, there are $t-1$ out of remaining $K-1$ possible users with whom the k-th user shares the subfile $W_{n,(j,\tau)}$. Thus, each user caches a total of $Nt\binom{K-1}{t-1}$ subfiles. Next, we generate a set of keys as

$$(\mathcal{K}_{(j_k,\tau_k)} : \tau_k \subseteq \{1, \ldots, K\}, |\tau_k| = t+1, j_k = \{1, \ldots, t+1\})$$

where the size of each key is $F/t\binom{K}{t}$ bits. All keys are independent from each other and are uniformly drawn as

$$\mathcal{K}_{(j_k,\tau_k)} \sim \text{unif}\{1, 2, \ldots, 2^{F/t\binom{K}{t}}\}.$$

For a given j_k and $k \in \tau_k$, there are t out of remaining $K-1$ possible users with whom the k-th user shares the secret key $\mathcal{K}_{(j_k,\tau_k)}$. Consequently, each user caches a total of $(t+1)\binom{K-1}{t}$ keys. The cache memory size required by each user is then given by

$$\begin{aligned}&Nt\binom{K-1}{t-1}\frac{F}{t\binom{K}{t}} + (t+1)\binom{K-1}{t}\frac{F}{t\binom{K}{t}} \\ &= F\left(\frac{Nt}{K} + \left(1 + \frac{1}{t}\right)\left(1 - \frac{t}{K}\right)\right) = FM \text{ bits} \quad (26)\end{aligned}$$

that satisfies the memory constraint in (24).

Example 1: For completeness, we now express the example in main section in terms of general notations described above. In particular, we consider the case when $K = N = 3$ and $M = 5/2$, this yields $t = 2$. Next, we partition

the files into $t\binom{K}{t} = 6$ subfiles and index them as $W_n = (W_{n,(1,12)}, W_{n,(1,13)}, W_{n,(1,23)}, W_{n,(2,12)}, W_{n,(2,13)}, W_{n,(2,23)})$ for $n = 1, 2, 3$. Afterwards, we generate a set of keys and index them as $\{\mathcal{K}_{1,123}, \mathcal{K}_{2,123}, \mathcal{K}_{3,123}\}$. The information stored in each cache is then given by

$$Z_1=(\{W_{n,(1,12)}, W_{n,(1,13)}, W_{n,(2,12)}, W_{n,(2,13)}\}_{n=1}^{3}, \{\mathcal{K}_{j_k,123}\}_{j_k=1}^{3})$$
$$Z_2=(\{W_{n,(1,12)}, W_{n,(1,23)}, W_{n,(2,12)}, W_{n,(2,23)}\}_{n=1}^{3}, \{\mathcal{K}_{j_k,123}\}_{j_k=1}^{3})$$
$$Z_3=(\{W_{n,(1,13)}, W_{n,(1,23)}, W_{n,(2,13)}, W_{n,(2,23)}\}_{n=1}^{3}, \{\mathcal{K}_{j_k,123}\}_{j_k=1}^{3}).$$

2) Delivery and Decoding: Consider $\mathcal{S} \subset \{1, \ldots, K\}$, where $|\mathcal{S}| = t + 1$ users. We focus our attention on a subset $\mathcal{S}$. At the end of cache placement, any t users in $\mathcal{S}$ share t subfiles which are required by the $t + 1$-th user. For a given $s \in \mathcal{S}$, such that $|\mathcal{S} \setminus \{s\}| = t$, note that 1) $W_{d_s,(j,\mathcal{S}\setminus\{s\})}$ is the subfile requested by the s-th user, which is a part of file W_{d_s}, 2) this subfile is available in the cache of remaining users, i.e., $\mathcal{S} \setminus \{s\}$, for $j = 1, \ldots, t$. In the D2D network that we study, since each node communicates with each other, the s-th user multicasts

$$\mathcal{K}_{(j_k,\mathcal{S})} \oplus_{k\in\mathcal{S}\setminus s} W_{d_k,(j,\mathcal{S}\setminus\{k\})}$$

of size $F/t\binom{K}{t}$ bits, where the parameter j_k is appropriately chosen such that a unique key is used in each transmission. Any subset of $t+1$ users require $t+1$ transmissions; and, there are in total of $\binom{K}{t+1}$ possible subsets of $\mathcal{S}$. As a result of the cache placement scheme described before, there are $(t+1)\binom{K}{t+1}$ unique keys available in the system. Thus, each transmission signal is appended with a key which is chosen from the set of keys available in the local cache of the user. Each transmission requires $F/t\binom{K}{t}$ bits. Hence, in total

$$RF=(t+1)\binom{K}{t+1} \times \frac{F}{t\binom{K}{t}} = \left(\frac{K}{t} - 1\right)F \qquad (27)$$

bits are transmitted. Then, finally replacing (25) in (27) and dividing both sides by F, we get

$$R^\star(M) \leq \frac{2K(N-1)}{K(M-1)+1+\sqrt{(K(M-1)+1)^2 - 4K(N-1)}} - 1. \qquad (28)$$

Example 2: Continuing from Example 1, we now explain the delivery phase as follows. Let $(d_1, d_2, d_3) = (W_1, W_2, W_3)$ be the demand vector. In this case since $|\mathcal{S}| = 3$, we have a single subset. Now, observe that user 1 requires subfiles $W_{1,(1,23)}$ and $W_{1,(2,23)}$ to construct W_1. These files are available in the cache of users 2 and 3, respectively. Similarly, the subfiles required by user 2 and 3 are available in the cache of the other users, respectively. In the transmission phase, first each user XOR's the subfiles required by other users along with a unique key and multicasts them as

$$X_{1,(d_1,d_2,d_3)}=W_{3,(1,12)} \oplus W_{2,(1,13)} \oplus \mathcal{K}_{1,123}$$
$$X_{2,(d_1,d_2,d_3)}=W_{1,(1,23)} \oplus W_{3,(2,12)} \oplus \mathcal{K}_{2,123}$$
$$X_{3,(d_1,d_2,d_3)}=W_{1,(2,23)} \oplus W_{2,(2,13)} \oplus \mathcal{K}_{3,123}.$$

With the help of side information stored locally, each user can readily recover the required subfiles.

3) Equivocation Analysis: In what follows, we show that no information is leaked to the eavesdropper. In particular, we show that

$$I(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)}; W_1, \ldots, W_N) = 0. \qquad (29)$$

Continuing from LHS in (29) we get

$$\begin{aligned} &I(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)}; W_1, \ldots, W_N) \\ &= \underbrace{H(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)})}_{\eta_1} \\ &- \underbrace{H(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)}|W_1, \ldots, W_N)}_{\eta_2}. \end{aligned} \qquad (30)$$

We first bound η_1 as follows.

$$\begin{aligned} \eta_1 &= H(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)}) \\ &= H(\{\mathcal{K}_{(j_k,\mathcal{S})} \oplus_{k\in\mathcal{S}\setminus s} W_{d_k,(j,\mathcal{S}\setminus\{k\})} : |\mathcal{S}| = t+1, \\ &\qquad j = 1, \ldots, t, j_k = 1, \ldots, t+1, s \in [K]\}) \\ &\leq \sum_{i=1}^{\binom{K}{t+1}} \sum_{j_k=1}^{t+1} H(\{\mathcal{K}_{(j_k,\mathcal{S}_i)} \oplus_{k\in\mathcal{S}_i\setminus s} W_{d_k,(j,\mathcal{S}_i\setminus\{k\})} : \\ &\qquad\qquad |\mathcal{S}_i| = t+1, j = 1, \ldots, t, s \in [K]\}) \\ &= (t+1)\binom{K}{t+1}\frac{F}{t\binom{K}{t}}. \end{aligned} \qquad (31)$$

Next, we bound η_2 as follows.

$$\begin{aligned} \eta_2 &= H(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)}|W_1, \ldots, W_N) \\ &\overset{(a)}{=} H(\{\mathcal{K}_{(j_k,\mathcal{S})} : |\mathcal{S}| = t+1, j_k = 1, \ldots, t+1\}) \\ &\overset{(b)}{=} \sum_{i=1}^{\binom{K}{t+1}} \sum_{j_k=1}^{t+1} H(\{\mathcal{K}_{(j_k,\mathcal{S}_i)} : |\mathcal{S}_i| = t+1\}) \\ &= (t+1)\binom{K}{t+1}\frac{F}{t\binom{K}{t}} \end{aligned} \qquad (32)$$

where (a) follows due to the independence of keys and $(W_1, \ldots, W_N)$ and the fact that they are uniformly distributed and (b) follows due to the independence of the keys.

Substituting (31) and (32) in (30) we get

$$I(X_{1,(d_1,\ldots,d_K)}, \ldots, X_{K,(d_1,\ldots,d_K)}; W_1, \ldots, W_N) = 0.$$

This concludes the proof.

REFERENCES

[1] L. W. Dowdy and D. V. Foster, "Comparative models of the file assignment problem," *ACM Comput. Surv.*, vol. 14, no. 2, pp. 287–313, Jun. 1982.

[2] K. Almeroth and M. Ammar, "The use of multicast delivery to provide a scalable and interactive video-on-demand service," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 6, pp. 1110–1122, 1996.

[3] Z. Bar-Yossef, Y. Birk, T. Jayram, and T. Kol, "Index coding with side information," *IEEE Trans. Inf. Theory*, vol. 57, no. 3, pp. 1479–1494, 2011.

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

[5] N. Golrezaei, A. G. Dimakis, and A. F. Molisch, "Wireless device-to-device communications with distributed caching," 2012. [Online]. Available: http://arxiv.org/abs/1205.7044

[6] M. Ji, G. Caire, and A. F. Molisch, "The throughput-outage tradeoff of wireless one-hop caching networks," 2013. [Online]. Available: http://arxiv.org/abs/1312.2637

[7] M. Maddah-Ali and U. Niesen, "Decentralized coded caching attains order-optimal memory-rate tradeoff," *IEEE/ACM Trans. Netw.*, to appear 2014.

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

[9] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, pp. 656–715, Oct. 1949.

[10] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of caching in wireless D2D networks," 2014. [Online]. Available: http://arxiv.org/abs/1405.5336