Fundamental Limits of Distributed Caching in Multihop D2D Wireless Networks

Mingyue Ji
ECE Department
University of Utah
Email: mingyue.ji@utah.edu

Rong-Rong Chen
ECE Department
University of Utah
Email: rchen@eng.utah.edu

Giuseppe Caire
EECS Department
Technical University of Berlin
Email: caire@tu-berlin.de

Andreas F. Molisch
EE Department
University of Southern California
Email: molisch@usc.edu

***Abstract*—We consider a wireless Device-to-Device (D2D) caching network, where users make arbitrary requests from a library of files and have pre-fetched (cached) information on their devices, subject to a per-node storage capacity constraint. The network is assumed to obey the "protocol model", widely considered in the wireless network literature. Unlike other related works, which either restrict the communication to single-hop, or assume entire file caching, here we consider both multi-hop transmission and fully general caching strategies, including file subpacketization. We propose a caching strategy based on deterministic assignment of MDS-coded packets of the library files, and a coded multicast delivery strategy where the users send linearly coded messages to each other in order to collectively satisfy their demands. We show that our approach can achieve the information theoretic outer bound within a multiplicative constant factor in practical parameter regimes.**

I. Introduction

Wireless data traffic has grown dramatically over the past few years primarily due to on-demand video streaming [1]. In addition, it is foreseen that wireless and mobile data traffic will increase even more significantly in the next few years due to the development of augmented and virtual reality (AR/VR) over wireless networks [2]. One emerging and promising approach for solving the "wireless data crunch" problem involving improving the spectral efficiency and reducing the latency (delay) is wireless caching and coded multicasting [3], [4]. This idea consists of using storage resources to cache popular content directly at the wireless edge [5], e.g., at small-cell base stations or end user devices, and deliver the user demands via coded multicasting. In [3]–[10], it has been shown that the wireless caching network has the potential to turn the (relatively cheap) memory into (extremely expensive) bandwidth, e.g., if the per-user storage capacity is doubled, then the per-user throughput is doubled.

There is extensive work in the literature on caching and coded multicasting under different physical layer channel models. For example, in the case of idealized channels, normally referred to as *shared link* networks, where all the channels from the source node to users are assumed to have the same channel quality, the seminal paper [4] introduced the information theoretic formulation of the caching problem and proposed a centralized caching placement and coded multicasting scheme, which achieves a total traffic load of $\Theta\left(\min\left\{\frac{m}{M}, m, n\right\}\right)$,[1] where m, n and M denote the library size, the number of users and the per-user storage capacity, respectively. Note that since the channel quality from the source node to all the users is the same, it is straightforward to see that the per-user throughput in the network also scales as $\Theta\left(\max\left\{\frac{M}{m}, \frac{1}{m}, \frac{1}{n}\right\}\right)$. From this result, we can observe that when $M \geq 1$ and $nM \gg m$, the throughput scales as $\Theta\left(\frac{M}{m}\right)$, which is independent of n and linearly proportional to M. Hence, the region of $M \geq 1$ and $nM \gg m$ is the regime of interest in general. Later, in [6], Maddah-Ali and Niesen proposed a decentralized caching placement and coded multicasting scheme which achieves almost the same throughput as [4]. Under the same network setting, different achievable schemes, which are superior to those in [4], [6] in terms of throughput and/or complexity, are presented in [3], [9]. From these results, when the channel is "idealized", the "multiplicative" caching gain can be obtained with manageable complexity, which makes the area of caching highly attractive.

In general, unlike conventional systems, caching networks require the joint design of three important system components, referred to as Caching (storage), Computing and Communication (C^3). In shared link caching networks, due to the oversimplified channel model, only caching and computing, referred to as cache placement and coded multicasting, respectively, are designed jointly. In contrast, a different approach to caching is introduced independently in [11], which considers a one-hop Device-to-Device (D2D) communication network with caching at the user nodes. Unlike the shared link caching networks, D2D caching networks do not have a single source node with access to the whole library. Instead, each of the n user nodes can only access its own memory of M files.[2] Due to the complexity of the network structure, there are more choices for the design of the transmission policy. Hence, D2D caching networks indeed require a joint design of the C^3. Under the simple protocol model of [12] and the worst-case demands, in [13], we proposed a caching

[1]We will use the following standard "order" notation: given two functions f and g, we say that: 1) $f(n) = O(g(n))$ if there exists a constant c and integer N such that $f(n) \leq cg(n)$ for $n > N$. 2) $f(n) = o(g(n))$ if $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$. 3) $f(n) = \Omega(g(n))$ if $g(n) = O(f(n))$. 4) $f(n) = \omega(g(n))$ if $g(n) = o(f(n))$. 5) $f(n) = \Theta(g(n))$ if $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

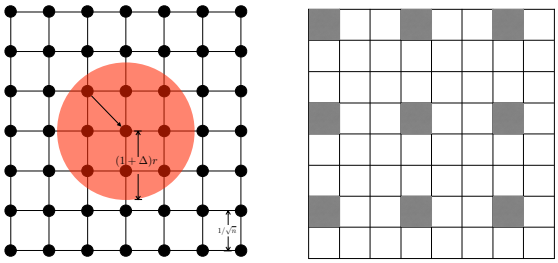
[2]In reality, the the "users" can be femtocell base stations or helpers.

strategy based on deterministic assignment of packets of the library files, and a coded multicasting delivery strategy, where the users send linearly coded messages to each other such that their demands can be satisfied. Further, we also consider the "decentralized" case, where a random caching strategy is designed. Under certain conditions, both approaches can achieve the information theoretic outer bound within a constant multiplicative factor. In particular, under the regime of interest ($M \geq 1$ and $nM \gg m$), the optimal per-user throughput in D2D caching networks scale as $\Theta\left(\frac{M}{m}\right)$, which is surprisingly identical as that in the shared link caching networks in the same regime. In addition, in [13], we also showed that the *spatial reuse gain* of the D2D network is order-equivalent to the *coded multicasting gain*. It is therefore natural to ask whether these two gains are cumulative, i.e., if a D2D network with both local communication (spatial reuse) and coded multicasting can provide an improved scaling law. Somewhat counterintuitively, we showed that these gains do not accumulate (in terms of throughput scaling law). This fact can be explained by noticing that the coded delivery scheme creates messages that are useful to multiple nodes, such that it benefits from broadcasting to as many nodes as possible, while spatial reuse capitalizes on the fact that the communication is local, such that the same time slot can be re-used in space across the network. Unfortunately, these two issues conflict with each other. When the constraint of single-hop communication is relaxed, in [14] and [15], the authors consider the multihop D2D caching networks under protocol model. Interestingly, if the demand distribution is Zipf with parameter less than 1, in the regime of interest, the per-user throughput scales as $\Theta\left(\left(\frac{M}{m}\right)^{\frac{1}{2}+\delta}\right)$ for some arbitrarily small $\delta > 0$, which has an order gain compared to the throughput of the single-hop D2D caching networks. Nevertheless, the limitation of these works is to restrict the caching strategy to caching entire files. Hence, the outer bound characterized in these works is not information theoretic.

In this paper, we extend the results in [13]–[15] to a general D2D caching network under the protocol channel model without any other constraints. In the regime of interest, we characterize the order optimal throughput in this caching networks in the information theoretic sense. Due to the space limit, most of the proofs and details are omitted.

II. Network Model and Problem Definition

We consider a D2D network formed by n user nodes $\mathcal{U} = \{1, \ldots, n\}$, which are placed on a regular grid on the unit square, with minimum distance $1/\sqrt{n}$. (see Fig. 1(a)). Let each user $u \in \mathcal{U}$ make an arbitrary request $f_u \in \mathcal{F} = \{1, \ldots, m\}$, where $\mathcal{F}$ called the file library is a set of m independently generated messages $\{W_1, \ldots, W_m\}$, each of which has entropy F bits. Often, this type of demands is referred to as *worst-case demands*. We assume that the file library is generated once in the beginning, and kept unchanged during the subsequent network operations. The set of all the requests is denoted by $\mathsf{f} = \{f_u, u \in \mathcal{U}\}$.



(a) (b)

Fig. 1. a) Grid network with $n = 49$ nodes (black circles) with minimum separation $\frac{1}{\sqrt{n}}$. The red area is the disk where the protocol model allows no other concurrent transmission. r is the worst case transmission range and Δ is the interference parameter. We assume a common r for all the transmitter-receiver pairs. b) An example of single-cell layout and the interference avoidance spatial reuse scheme. In this figure, each square represents a cluster. The gray squares represent the concurrent transmitting clusters. In this particular example, the reuse factor is $\mathcal{K} = 9$.

Communication between user nodes obeys the *protocol model* [12] as follows: if a node v transmits a packet to node u, then the transmission is successful if and only if: a) The distance between v and u is less than r; b) Any other node k transmitting simultaneously, is at distance $d(k, u) \geq (1 + \Delta)r$ from the receiver u, where $r, \Delta > 0$ are the parameters of the protocol model. In practice, nodes send data at some constant rate C_r bit/s/Hz, where C_r is a non-increasing function of the transmission range r.

A caching scheme consists of information storage, computing and communication, which are referred to as the caching, coded delivery, transmission phases defined as follows.

Definition 1: **(Caching Phase)** The caching phase is a map of the file library $\{W_f : f \in \mathcal{F}\}$ onto the cache of the users $u \in \mathcal{U}$. Each cache has size MF bits (i.e., M files). For $u \in \mathcal{U}$, the function $\phi_u : \mathbb{F}_2^{mF} \to \mathbb{F}_2^{MF}$ generates the cache content $Z_u \triangleq \phi_u(W_f : f \in \mathcal{F})$. The cache messages Z_u are stored in the user caches at the beginning of time, and kept fixed through the subsequent network operations. $\Diamond$

Definition 2: **(Coded Delivery Phase)** The coded delivery phase is defined by two sets of functions: the node encoding functions, denoted by $\{\psi_u : u \in \mathcal{U}\}$, and the node decoding functions, denoted by $\{\lambda_u : u \in \mathcal{U}\}$. Let R_u^{T} denote the number of coded bits transmitted by node u to satisfy the request vector f. The rate (normalized traffic load) of node u is defined by $R_u = \frac{R_u^{\mathrm{T}}}{F}$. The function $\psi_u : \mathbb{F}_2^{MF} \times \mathcal{F}^n \to \mathbb{F}_2^{FR_u}$ generates the transmitted message $X_{u,\mathsf{f}} \triangleq \psi_u(Z_u, \mathsf{f})$ of node u as a function of its cache content Z_u and of the demand vector f. Let $\mathcal{D}_u$ denote the set of users whose transmit messages are received by user u (according to some transmission policy in Definition 3). The function $\lambda_u : \mathbb{F}_2^{F\sum_{v\in\mathcal{D}_u} R_v} \times \mathbb{F}_2^{MF} \times \mathcal{F}^n \to \mathbb{F}_2^F$ decodes the request of user u from the received messages and its own cache, i.e., we have

$$\hat{W}_{u,\mathsf{f}} \triangleq \lambda_u(\{X_{v,\mathsf{f}} : v \in \mathcal{D}_u\}, Z_u, \mathsf{f}). \tag{1}$$

$\Diamond$

Due to the fact that users make arbitrary requests, similar to [4], [13], we focus on the worst-case error probability defined

as

$$P_e = \max_{\mathsf{f}\in\mathcal{F}^n} \max_{u\in\mathcal{U}} \mathbb{P}\left(\hat{W}_{u,\mathsf{f}} \neq W_{f_u}\right). \quad (2)$$

For given number of users n and library size m, letting $R = \sum_{u\in\mathcal{U}} R_u$, we say that the cache-rate pair (M, R) is achievable if $\forall\, \varepsilon > 0$ there exist a sequence indexed by the file size $F \to \infty$ of cache encoding functions $\{\phi_u\}$, delivery functions $\{\psi_u\}$ and decoding functions $\{\lambda_u\}$, with rate $R^{(F)}$ and probability of error $P_e^{(F)}$ such that $\limsup_{F\to\infty} R^{(F)} \leq R$ and $\limsup_{F\to\infty} P_e^{(F)} \leq \varepsilon$. It is clear that RF gives the achievable total traffic load transmitted in the whole network.

Definition 3: **(Transmission Phase)** The transmission policy Π is a rule to activate the D2D links in the network. Let $\mathcal{L}$ denote the set of all directed links. Let $\mathcal{A} \subseteq 2^{\mathcal{L}}$ be the set of all possible feasible subsets of links (this is a subset of the power set of $\mathcal{L}$, formed by all sets of links forming independent sets in the network interference graph induced by the protocol model). Let $\mathsf{A}_t \subset \mathcal{A}$ denote a feasible set of simultaneously active links at time t. A feasible transmission policy Π consists of a sequence of activation sets, i.e., sets of active transmission links, $\{\mathsf{A}_t : t = 1, 2, 3, \cdots\}$, such that at each time t the active links in A_t do not violate the protocol model. $\Diamond$

Suppose that RF is achievable and that there exists a transmission policy that can deliver to each user the coded symbols necessary to decode its requested file in the worst-case demand case in no more than D channel uses. Then, the per-user throughput, measured in useful information bits per channel use, is given by

$$T \triangleq \frac{F}{D}. \quad (3)$$

We say that the pair (M, T) is achievable if RF is achievable and if there exists a transmission policy Π such that the RF encoded bits can be delivered to their destinations in $D \leq F/T$ channel uses. Then, the optimal achievable throughput is defined as

$$T^*(M) \triangleq \sup\{T : (M, T) \text{ is achievable}\}. \quad (4)$$

Notice that in the case of the single bottleneck link network [4] obviously $D = RF$, such that $T = 1/R$. Notice that due to the requirement of the joint consideration of Caching, Computing and Communication (C^3), the transmission rate R is not enough to characterize the system throughput performance in D2D caching networks, i.e., the low transmission rate R does not necessarily lead to a high throughput due to different designs of communication schemes.

In the following we assume that $t \triangleq \frac{Mn}{m} \geq 1$. Notice that this is a necessary condition in order to satisfy any arbitrary demand vector without any outage. If $t < 1$, the the worst-case throughput is trivially $T = 0$ (by requesting the missing file in the library).

III. Main Results

As discussed in Section II, we need to design cache placement, coded delivery and transmission phases. In particular, the achievable scheme designed in this paper consists of a MDS-coded cache placement, a coded multicasting delivery and a randomized Euclidean Minimum Spanning Tree (EMST) based Manhattan multicast routing scheme. The details of the achievable scheme will be presented in Section IV. In this section, we will present the performance of the proposed scheme and information theoretic outer bound for the D2D caching networks.

The following theorem yields the achievable rate obtained by our proposed constructive scheme.

Theorem 1: Let m, n, M be the library size, number of users and the cache size per-user, respectively. For $t = (1-\varepsilon)\frac{Mn}{m} \in \mathbb{Z}^+$ and $\omega\left(\max\left\{\frac{\log\log n}{\log\frac{m}{M}}, 1\right\}\right) = t = O(n)$, as $n \to \infty$, with high probability,[3] the following throughput is achievable:

$$T(M) = \frac{c\sqrt{\frac{M}{m}}}{1-\frac{M}{m}} C_r, \quad (5)$$

where $c > 0$ and $0 < \varepsilon < 1$ are some positive constant, which is independent of n, M, m. Moreover, when t is not an integer, the convex lower envelope of $T(M)$, seen as a function of $M \in [0 : m]$, is achievable. $\square$

A lower bound (converse result) for the achievable rate in this case is given by the following theorem.

Theorem 2: For a given transmission range $r \geq \frac{1}{\sqrt{n}}$, any achievable throughput is upper bounded by

$$T^*(M) \leq \frac{c'\frac{C_r}{\Delta^2}\frac{\sqrt{\frac{m}{nM}}}{\min\{r, \sqrt{\frac{m}{nM}}\}}}{\max_{l\in\{1,2,\cdots,\min\{\frac{m}{2}, \lfloor\frac{m}{2M}\rfloor\}\}}\left(\frac{l}{2} - \frac{l}{\lfloor\frac{m}{2l}\rfloor}M\right)}, \quad (6)$$

where $c' > 0$ is some positive constant, which is independent of n, m, M. $\square$

By using Theorem 2, we can obtain the following corollary.

Corollary 1: Any achievable throughput is upper bounded by

$$T^*(M) \leq \frac{c'\frac{C_r}{\Delta^2}\sqrt{\frac{m}{M}}}{\max_{l\in\{1,2,\cdots,\min\{\frac{m}{2}, \lfloor\frac{m}{2M}\rfloor\}\}}\left(\frac{l}{2} - \frac{l}{\lfloor\frac{m}{2l}\rfloor}M\right)}, \quad (7)$$

where $c' > 0$ is some positive constant, which is not a function of the system parameters n, m, M. $\square$

Proof: Since the righthand side of (6) is an non-increasing function of r, then (7) is obtained by putting $r = \frac{1}{\sqrt{n}}$ into the righthand side of (6). ■

The order optimality of our achievable rate is shown by the following theorem.

Theorem 3: As $n, m \to \infty$, for $\omega\left(\max\left\{\frac{\log\log n}{\log\frac{m}{M}}, 1\right\}\right) = t = O(n)$ and $M \geq \frac{1}{4}$, the ratio between the achievable throughput and the optimal throughput is upper bounded by

$$\frac{T^*(M)}{T(M)} \leq c'', \quad (8)$$

where c'' is a positive constant and is independent of m, M, n. $\square$

[3] A throughput of C is achievable with high probability means that $\lim_{n\to\infty}\mathbb{P}(T \geq C) = 1$, where T is a function of n.

IV. Achievability

A. MDS Coding

Each file is partitioned into K packets of F/K bits each. We let $K = (1-\varepsilon)t\binom{n}{t}$, where $(1-\varepsilon) = (1-\varepsilon')\left(1-e^{-1}\right)$ for some arbitrarily small ε'. These packets represent the elements of the binary extension field $\mathbb{F}_{2^{F/K}}$, and are encoded using a $(K, K/\rho)$-MDS code, where $\rho = (1-\varepsilon)$. Notice that this expands the size of each file from F to F/ρ. Equivalently, each node caches less than M files in terms of size. Note that since we consider the case that $t = \omega(1)$, then we can always guarantee that all the coded files can be cached in the network. The resulting K/ρ encoded packets of F/K bits each will be referred to as "MDS-coded symbols" or "coded packets" in the following. Hence, the total number of MDS coded symbols per file is $K/\rho = (1-\varepsilon)t\binom{n}{t}/(1-\varepsilon) = t\binom{n}{t}$. Since $F \to \infty$, we choose K as a function of F such that both K and F/K go to infinity as F increases. This guarantees that $(K, K/\rho)$-MDS codes exist for any fixed ρ.

B. Coded Cache Placement

The cache placement scheme is closely related to the scheme in [4], [13]. From Section IV-A, we can see that each MDS-coded file is divided into $t\binom{n}{t}$ packets. Letting T denote a specific combination of t out of n elements, each packet is indexed by the superscript (T, j) with $j = 1, \ldots, t$, such that the packets of W_f are denoted by $\{W_f^{\mathsf{T},j}\}$. The cache function Z_u defined in Definition 1 is given by $\{W_f^{\mathsf{T},j} : u \in \mathsf{T}, \forall f = 1, \ldots, m, j = 1, \cdots, t\}$.

C. Coded Multicasting

As a result of the caching scheme described above, any subset of user nodes of size $t+1$ in $\mathcal{U}$ has the property that the nodes of any of its subsets of size t share t MDS-coded symbols (packets) for each file. Consider one of these subsets, and consider any user node in this subset. For any file requested by this node, by construction, there are t coded packets shared by all other t nodes and needed by this node. Hence, each node in every subset of size $t+1$ has t coded packets, each of which is useful for one of the remaining t nodes. Note that such sets of packets are disjoint (empty pairwise intersections). For delivery, for all subsets of $t+1$ nodes, each node computes the XOR of its set of t useful coded packets and multicasts it to all other nodes (see Section IV-D for the multicast transmission scheme). In this way, for every multicast transmission, exactly t nodes will be able to decode a useful packet using "interference cancellation" based on their cache side information. Note that, for successful delivery, every user node only needs K out of the K/ρ coded packets for the requested file.

D. Multicast Routing

For the communication/transmission scheme, the transmission range can be picked arbitrarily in order to allow local D2D communications and spatial reuse. Unlike the single-hop D2D communication networks, where only a scheduling scheme for simultaneous activated links is required, in the case of multihop transmissions, we need to design 1) a routing protocol such that K out of the K/ρ XORed MDS-coded symbols can be delivered to each destination and 2) a scheduling scheme to enable concurrent active links. The proposed policy is based on *clustering*, which means that the network is divided into clusters of equal size g_c, which is a function of the transmission range r and independently of the users' demands. The proposed routing protocol is based on Euclidean Minimum Spanning Tree (EMST) proposed in [16]. In this scheme, for each source node s, and its randomly and uniformly selected destination nodes $d_1, \cdots, d_t$, a EMST is built (see Fig. 2(a)). Then the communication between the two nodes in this EMST is via a Manhattan routing protocol (see Fig. 2(b)). The detailed routing algorithms is shown by Algorithm 1. A simple scheduling policy consists of partitioning the set of clusters into Q subsets, such that the clusters of the same set do not interfere, activate simultaneously one link per cluster in each subset, and use TDMA in order to avoid interference between the clusters. This is a classical time-frequency reuse scheme with reuse factor Q [13], as shown in Fig. 1(b). In particular, we can pick $Q = \left(\left\lceil\sqrt{2}(1+\Delta)\right\rceil + 1\right)^2$.

Algorithm 1 Euclidean Minimum Spanning Tree based Manhattan Routing Protocol

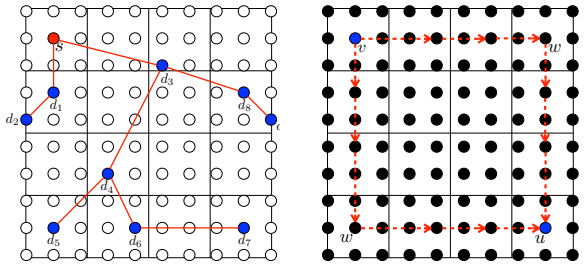
1: We partition the network into clusters, each with side length $r/\sqrt{5}$. Each cluster is denoted by (i, j), where i, j represent the indices of rows and columns respectively.
2: For each user node s and its destinations $d_1, d_2, \cdots, d_t$, a Euclidean Minimum Spanning Tree (EMST) is built as shown in [16].
3: For each link in EMST, the clusters contain v and u are denoted by (i_v, j_v) and (i_u, j_u) respectively. We find a node w in cluster (i_v, j_u) (or cluster (i_u, j_v)) such that vwu is a Manhattan path connecting u and v. This means that we find the shortest path (with minimum Euclidean length) connecting vw and wu, where each hop is within the pre-designed transmission range r. This step is illustrated in Fig. 2.
4: The multicast route is obtained by concatenating all such shortest paths for all links in EMST. Note that if this route is not a tree, we simply remove the cycles that do not contain nodes from user s and its destinations $d_1, \cdots, d_t$.

V. Discussions

The achievable throughput of *Theorem* 1 can be written as the product of four terms as follows.

$$T(M) = \frac{c\sqrt{\frac{M}{m}}}{1-\frac{M}{m}} C_r = \Theta\left(\frac{1}{n} \cdot \frac{1}{1-\frac{M}{m}} \cdot t \cdot \sqrt{\frac{n}{t}}\right), \qquad (9)$$

where $t = \frac{Mn}{m}$ is the ratio between the aggregate memory in the network and the library size. We have the following interpretation for (9). The first three terms $\frac{1}{n}$, $\frac{1}{1-\frac{M}{m}}$ and t are also found in the shared link caching networks [4] and the single-hop D2D caching networks [13]. In particular, the term



(a) (b)

Fig. 2. (a) The D2D caching network is partitioned into square clusters with side length of $\frac{r}{\sqrt{5}}$. The red node s represents the source node and all the blue nodes represent its $t = 9$ multicast destinations $d_1, \cdots, d_9$. The constructed Euclidean Minimum Spanning Tree (EMST) for this multicast session is shown by the red, blue nodes and the red solid lines. (b) A illustration of the Manhattan routing approach from node v (transmitter) to node u (receiver). The red dashed line represent two manhattan routing paths. In this example, the transmission range is assume to be $\frac{3}{\sqrt{n}}$.

$\frac{1}{n}$ is the per-user throughput by using a conventional scheme that serves individual demands without exploiting the demand redundancy; $\frac{1}{1-\frac{M}{m}}$ can be viewed as the *local caching gain*, any user can cache a fraction M/m of each file, hence it needs to receive only the remaining part; t is referred to as *global caching gain*, which is the gain due to the aggregate memory in the network rather than individual user's memory. [4] The new term $\sqrt{\frac{n}{t}}$ is the *multihop transmission gain*, which is the gain of multhop D2D caching networks over the single-hop D2D caching networks and obtained by using multihop. Intuitively, this term can be interpreted as follows. There are at most $\Theta(n)$ concurrent transmissions in the network. Each user has t destinations uniformly selected at random such that the average number of hops per bit is $\Theta(\sqrt{nt})$.[5] Hence, the average number of bits that can be sent in the network per time slot is given by $\Theta\left(\frac{n}{\sqrt{nt}}\right) = \Theta\left(\sqrt{\frac{n}{t}}\right)$.

In [15], a random cache placement based on whole files and local multihop unicasting delivery scheme was proposed. Interestingly, the achievable per-user throughput is $\Theta\left(\left(\frac{M}{m}\right)^{\frac{1}{2}+\delta}\right)$ (for some arbitrarily small δ), which is almost identical as the per-user throughput obtained in this paper.[6] Hence, there is indeed no order gain in terms of throughput by using fractional caching placement and coded multihop multicasting proposed in this paper. In fact, this throughput is order-optimal in the information theoretic sense as shown in *Theorem* 3. In practice, only a constant gain of the scheme proposed in this paper compared to that in [15] may be possible depending on the realistic physical layer channel conditions and the choice of r and Δ in the protocol model. This observation is analogous to the case of single-hop D2D caching networks, where coded multicasting also does not provide order gains in terms of throughput [13]. On the other hand, the proposed scheme may have some other more significant gains. For example, the amount of data traffic N_S generated at each node from its memory is given by

$$N_S = \frac{F}{(1-\varepsilon)t\binom{n}{t}} \cdot \binom{n-1}{t} = \frac{F}{1-\varepsilon}\left(1-\frac{M}{m}\right)\frac{1}{t} = \Theta\left(\left(1-\frac{M}{m}\right)\frac{F}{t}\right). \quad (10)$$

In contrast, due to the constraint of caching the entire files, the data traffic generated at each user from its own memory for the scheme of [15] is given by $F\left(1-\frac{M}{m}\right)$bits. Hence, in terms of the data generated at each node, the proposed scheme in this paper has a gain of t, which is the global caching gain.

[4]Note that the global caching gain for the shared link caching network is $t+1$ [4]. For $nM \gg m$ ($t \geq 1$), these factors are almost identical.

[5]Note that for unicast network, where $t = 1$, the number of hops per bit reduces to $\Theta(\sqrt{n})$ as shown in the seminal paper [12].

[6]Note that precisely speaking, the scenarios studied in this paper and in [15] are not the same. In this paper, we focus on the worst-case user demand and assume users are distributed on a grid. While [15] focuses on the case where user demands follow a heavily tail distribution, where uniform demand distribution is a special case, and users are distributed randomly.

ACKNOWLEDGEMENT

The work of Andreas F. Molisch was partially supported by the NSF.

REFERENCES

[1] Cisco, "The Zettabyte Era-Trends and Analysis," 2013.
[2] Baştuğ E., M. Bennis, M. Médard, and M. Debbah, "Towards interconnected virtual reality: Opportunities, challenges and enablers," *arXiv preprint arXiv:1611.05356*, 2016.
[3] K. Shanmugam, M. Ji, A. M. Tulino, J. Llorca, and A. G. Dimakis., "Finite-length analysis of caching-aided coded multicasting," *IEEE Transactions on Information Theory*, vol. 62, no. 10, pp. 5524–5537, Oct 2016.
[4] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *Information Theory, IEEE Transactions on*, vol. 60, no. 5, pp. 2856–2867, 2014.
[5] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Order-optimal rate of caching and coded multicasting with random demands," *arXiv preprint arXiv:1502.03124*, 2015.
[6] M. A. Maddah-Ali and U. Niesen, "Decentralized coded caching attains order-optimal memory-rate tradeoff," *Networking, IEEE/ACM Transactions on*, vol. 23, no. 4, pp. 1029–1040, Aug 2015.
[7] U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands," *IEEE Transactions on Information Theory*, vol. 63, no. 2, pp. 1146–1158, Feb 2017.
[8] Kai Wan, D. Tuninetti, and P. Piantanida, "On the optimality of uncoded cache placement," in *2016 IEEE Information Theory Workshop (ITW)*, Sept 2016, pp. 161–165.
[9] K. Wan, D. Tuninetti, and P. Piantanida, "On caching with more users than files," in *2016 IEEE International Symposium on Information Theory (ISIT)*, July 2016, pp. 135–139.
[10] A. Liu, V. Lau, and G. Caire, "Cache-induced hierarchical cooperation in wireless device-to-device caching networks," *arXiv preprint arXiv:1612.07417*, 2016.
[11] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 142–149, April 2013.
[12] P. Gupta and P.R. Kumar, "The capacity of wireless networks," *Information Theory, IEEE Trans. on*, vol. 46, no. 2, pp. 388–404, 2000.
[13] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of caching in wireless d2d networks," *IEEE Transactions on Information Theory*, vol. 62, no. 2, pp. 849–869, Feb 2016.
[14] S. Gitzenis, G. Paschos, and L. Tassiulas, "Asymptotic laws for joint content replication and delivery in wireless networks," *Information Theory, IEEE Transactions on*, vol. 59, no. 5, pp. 2760–2776, 2013.
[15] S. W. Jeon, S. N. Hong, M. Ji, G. Caire, and A. F. Molisch, "Wireless multihop device-to-device caching networks," *IEEE Transactions on Information Theory*, vol. 63, no. 3, pp. 1662–1676, March 2017.
[16] X.Y. Li, "Multicast capacity of wireless ad hoc networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 3, pp. 950–961, 2009.