Multicast Performance with Hierarchical Cooperation

Xinbing Wang, Luoyi Fu, Chenhui Hu
Dept. of Electronic Engineering
Shanghai Jiao Tong University, China
Email: {xwang8,fly hch}@sjtu.edu.cn

***Abstract*—It has been shown in [1] that hierarchical cooperation achieves a linear throughput scaling for unicast traffic, which is due to the advantage of long range concurrent transmissions and the technique of distributed MIMO. In this paper[1], we investigate the scaling law for multicast traffic with hierarchical cooperation, where each of the n nodes communicates with k randomly chosen destination nodes. Specifically, we propose a new class of scheduling policies for multicast traffic. By utilizing the hierarchical cooperative MIMO transmission, our new policies can obtain an aggregate throughput of $\Omega\big((\frac{n}{k})^{1-\epsilon}\big)$ for any $\epsilon > 0$. This achieves a gain of nearly $\sqrt{\frac{n}{k}}$ compared with the non-cooperative scheme in [26]. Among all four cooperative strategies proposed in our paper, one is superior to in terms of the 3 performance metrics: throughput, delay and energy consumption. Two factors contribute to the optimal performance: multi-hop MIMO transmission and converge-based scheduling. Compared with the single-hop MIMO transmission strategy, the multi-hop strategy achieves a throughput gain of $(\frac{n}{k})^{\frac{h-1}{h(2h-1)}}$ and meanwhile reduces the energy consumption by $k^{\frac{\alpha-2}{2}}$ times approximately, where $h > 1$ is the number of the hierarchical layers, and $\alpha > 2$ is the path loss exponent. Moreover, to schedule the traffic with the converge multicast instead of the pure multicast strategy, we can dramatically reduce the delay by a factor of about $(\frac{n}{k})^{\frac{h}{2}}$. Our optimal cooperative strategy achieves an approximate delay-throughput tradeoff $D(n,k)/T(n,k) = \Theta(k)$ when $h \to \infty$. This tradeoff ratio is identical to that of non-cooperative scheme, while the throughput is greatly improved.**

I. Introduction

Capacity of wireless ad hoc networks is constrained by interference between concurrent transmissions. Observing this, Gupta and Kumar adopt Protocol and Physical Model to define a successful transmission, and study the capacity scaling, i.e., the asymptotically achievable throughput of the network in their seminal work [3]. Assume there are n nodes in a unit disk area, they show that the per-node throughput capacity scales as $\Theta\big(\frac{1}{\sqrt{n\log n}}\big)$ for random networks, and the per-node transport capacity for arbitrary networks scales as $\Theta\big(\frac{1}{\sqrt{n}}\big)$, respectively.

The results on network capacity provide us both a theoretical bound and insights in the protocol design and architecture of wireless networks. Thus, great efforts are devoted to understand the scaling laws in wireless ad hoc networks. One important stream of work is improving unicast capacity. With percolation theory, Franceschetti et al. [4] show that a rate $\Theta\big(\frac{1}{\sqrt{n}}\big)$ is attainable in random ad hoc networks under Generalized Physical Model. However, it is still vanishing when we have infinite number of nodes. To achieve linear capacity scaling, Grossglauser et al. [5] exploit nodes' mobility to increase network throughput while at a cost of induced delay. Tradeoff between capacity and delay is studied in literatures [10] – [12]. An alternative way is adding infrastructure to the network. It is shown in [13] – [17] that when the number of base stations grows linearly as that of the nodes (implying a huge investment), capacity will scale linearly. Moreover, instead of letting nodes perform traditional operations such as storage, replication and forwarding, [18] and [19] introduce coding into the network. This also brings about the gain on throughput.

Recently, Aeron et al. [6] introduce a multiple-input multiple-output (MIMO) collaborative strategy achieving a throughput of $\Omega(n^{-1/3})$. Different from the Gupta and Kumar's results, they use a cooperative scheme to obtain capacity gain by turning mutually interfering signals into useful ones. Later, Özgür et al. [1] [2] utilize hierarchical schemes relying on distributed MIMO communications to achieve linear capacity scaling. The optimal number of hierarchical stages is studied in [7], while multi-hop and arbitrary networks are investigated in [8] and [9], respectively.

Another line of research deals with more generalized traffic patterns. In [20], Toumpis develops asymptotic capacity bounds for non-uniform traffic networks. In [21], broadcast capacity is discussed. Then, a unified perspective on the capacity of networks subject to a general form of information dissemination is proposed in [22]. As a more efficient way for one-to-many data distribution than multiple unicast, multicast is well fit for the applications such as group communications and multi-media services. Thus, it raises great interests to the research community and has been studied by different manners in [23] – [30]. Specifically, in [24], the authors derive the asymptotic upper and lower bounds for multicast capacity by focusing on data copies and area argument in the routing tree established in the paper; In [25], multicast capacity is studied under a more realistic channel model, physical layer model instead of simplified protocol model assumed in many previous literatures; In [26], through mathematical derivations and simulations, the authors demonstrate that multicast achieves a gain compared to unicast when information is disseminated to n destinations in mobile ad hoc networks; In [27], a

[1]*An earlier version of this paper appeared in the Proceedings of IEEE Infocom 2010 [33].*.

comb-based architecture is proposed instead of routing tree for multicast and this is shown to achieve an order-optimal multicast capacity in static networks; In [28], Wang *et. al.* prove that network coding cannot necessarily bring about gain in multicast capcity, which is a counter-intuitive result. Very lately, Niesen et al. [31] characterize the multicast capacity region in an extended network. And capacity-delay tradeoff for mobile multicast is inquired in [32].

However, the capacity of all the work above is largely restricted by adjacent interference which is caused by the concurrently transmitting nodes nearby. This is the bottleneck for the capacity existing in the traditional ad-hoc networks. This motivates us to focus on multicast scaling laws with hierarchical MIMO in this paper. We jointly consider the effect of traffic patterns and cooperative strategies on the asymptotic performance of networks, aiming to break the bottleneck. Moreover, there lacks a former work following into this kind. Thus, the next questions are still open.

- How to hierarchically schedule multicast traffic to optimize the achievable multicast throughput?
- Is there a strategy with good delay performance and is energy-efficient when achieving optimal throughput?
- What is the delay-throughput tradeoff in our hierarchical cooperative multicast strategies?

To answer the above questions, we propose a class of hierarchical cooperative scheduling strategies to solve the multicast problem. Specifically, we divide the network into clusters; nodes in the same cluster cooperate to transmit data for each other. In this way, all transmissions in the network consist of two parts: inter-cluster communication and intra-cluster communication.

Inter-cluster communication: The transmissions between clusters are conducted by distributed MIMO. When a cluster acts as a sender, all nodes in the cluster transmit a *distinct* bit at the same time. Then each node in the receiving cluster can observe a signal containing information of all transmitted bits.

We propose two kinds of transmission: direct and multi-hop MIMO transmission, which is more general than that in [33]. For the communication between clusters, the direct manner uses MIMO transmission only once from the source cluster to all destination clusters, while the multi-hop manner conducts MIMO transmissions for many hops, and each time a cluster only transmits to the neighboring cluster. After analysis, we find multi-hop MIMO transmission can increase the throughput and reduce the energy consumption due to better spatial reuse and power management.

Intra-cluster communication: To decode MIMO transmissions, the destination nodes in each destination cluster must collect observation results from all nodes in the same cluster. Since each cluster may act as a destination cluster of multiple source clusters, there are several sets of destination nodes in it. For each set, every node in the cluster sends one *identical* bit to all nodes in the set. This traffic can be seen as multicast, but considering the "converge" nature of the data flows, it can also be regarded as *converge multicast*. Hence, we propose two kinds of strategies: multicast-based strategy and converge-based strategy.

Comparing these two kinds of strategies, there are no differences on throughput and energy consumption. However, the converge-based strategy can dramatically reduce the delay by approximately $\Theta\Big((\frac{n}{k})^{\frac{h}{2}}\Big)$, where $h > 1$ is the number of hierarchical layers in the network. We further divide clusters into "sub-clusters", and still use distributed MIMO to communicate between them. When using multicast-based strategy, for each source node it must distribute data within its sub-cluster, which accounts for the major part of the delay. On the other hand, utilizing the converge nature of the traffic, converge-based strategy omits the distribution procedure and significantly reduces the delay.

Our main contributions are as follows.

- We propose a class of hierarchical cooperative scheduling policies for multicast traffic, which can nearly achieve the throughput information-theoretic upper bound.
- We reschedule the traffic of our cooperative transmission and dramatically reduce the delay.
- We achieve an identical delay-throughput tradeoff to non-cooperative multicast scheme, while the throughput is greatly improved. The multicast tradeoff even outperforms that of unicast in some special cases.

Our main results are presented below.[2]

- We achieve a throughput of $\widetilde{\Theta}\Big((\frac{n}{k})^{\frac{2h-2}{2h-1}}\Big)$, which has a gain of nearly $\sqrt{\frac{n}{k}}$ compared with non-cooperative scheme.
- The delay of our optimal strategy is $\widetilde{\Theta}\Big(n^{\frac{2h-4}{2h-1}}k^{\frac{3}{2h-1}}\Big)$, which achieves a delay-throughput tradeoff ratio $\widetilde{\Theta}\Big(k(\frac{k}{n})^{\frac{2}{2h-1}}\Big)$.
- The energy-per-bit consumption is $O\Big(n^{\frac{1-\alpha}{2h-1}}k^{-\frac{2h\alpha-3\alpha+2}{4h-2}}\Big)$.

The rest of the paper is organized as follows. In Section II, we give our network models and definitions of terms. In Section III, we outline the multicast hierarchical cooperative scheme. Then, the analysis of throughput, delay and energy consumption are presented in Section IV, V-A and V-B, respectively. All the results are discussed in detail in Section VI. Finally, we conclude the paper in Section VII.

II. Network Models and Definitions

A. Network Models

We consider a set of n nodes $V = \{v_1, v_2, \ldots, v_n\}$ uniformly and independently distributed in a unit square Ω. Each node v_i acts as a source node of a multicast session.

Multicast Traffic: For a source node v_i, we randomly and independently choose a set of k nodes $U_i = \{u_{i,j} | 1 \le j \le k\}$ other than v_i in the deployment square as its destination nodes. We define a multicast *session* as the collection of transmissions from one source node to k destination nodes, and use MP(n, k) to denote a n-session multicast problem with each node acting as a source node for a session.

We then define another traffic that helps in our analysis.

[2]We use Knuth's notation in this paper. Also we use $f(n) = \widetilde{\Theta}(g(n))$ to indicate $f(n) = O(n^{\epsilon}g(n))$ and $f(n) = \Omega(n^{-\epsilon}g(n))$, for any $\epsilon > 0$. Intuitively, this means $f(n) = \Theta(g(n))$ with logarithmic terms ignored.

Converge Multicast Traffic: We randomly and independently choose a set of k nodes $U_i = \{u_{i,j} | 1 \le j \le k\}$ as destinations. Each of n nodes in the network acts as a source node and sends one identical bit to all nodes in U_i. This is a "converge" transmission because the overall data flow is from all n nodes to the set of k nodes. See Figure 1-(c) for illustration. And we define it as a converge multicast *frame*. Use CMP(n, m, k) to denote a m-frame converge multicast problem, for each frame we choose a set of k destination nodes.

Wireless Channel Model: We assume that communication takes place over a channel of limited bandwidth W. Each node has a power budget of P. For the transmission from v_j to v_i, the channel gain between them at time t is given by:

$$g_{ij}[t] = \sqrt{G} d_{ij}^{-\alpha/2} e^{j\theta_{ij}[t]} \tag{1}$$

where d_{ij} is the distance between v_i and v_j, $\theta_{ij}[t]$ is the random phase at time t, uniformly distributed in $[0, 2\pi)$. $\{\theta_{ij}[t] | 1 \le i, j \le n\}$ is a collection of independent and identically distributed (i.i.d.) random processes. The parameters G and $\alpha > 2$ are assumed to be constants; α is called the path-loss exponent. Then, the signal received by node v_i at time t can be expressed as

$$Y_i[t] = \sum_{j \in \mathbb{T}[t]} g_{ij}[t] X_j[t] + Z_i[t] + I_i[t] \tag{2}$$

where $Y_i[t]$ is the signal received by node v_i at time t, $\mathbb{T}[t]$ represents the set of active senders, which can be added constructively, $Z_i[t]$ is the Gaussian noise at node v_i of variance N_0 per symbol, and $I_i[t]$ is the interference from the nodes which are destructive to the reception of node v_i.

When conducting cooperative transmission, we assume that full channel state information (CSI) is available at each node[3]. Also we assume the far-field condition holds for all nodes, i.e. the minimum distance between any two nodes is larger than the wavelength of the carrier frequency[4].

In this paper, we only consider *dense network*, which means the network area is a unit square. Our hierarchical cooperative scheme can also be applied to *extended network*, with a $\sqrt{n} \times \sqrt{n}$ square network area.

B. Definition of Performance Metrics

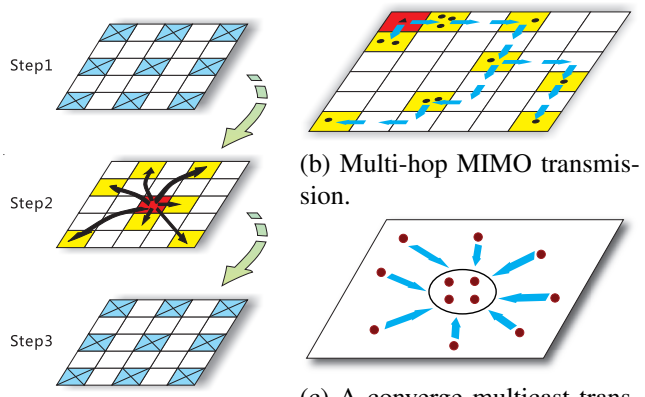
Definition of Throughput: A per node throughput of $\lambda(n, k)$ bit/s is feasible if there is a spatial and temporal transmission scheme, such that every node can send $\lambda(n, k)$ bit/s on average to its k randomly chosen destination nodes. The aggregate multicast throughput of the system is $T(n, k) = n\lambda(n, k)$. When $k = 1$, it becomes aggregate unicast throughput.

Definition of Delay: The delay $D(n, k)$ of a communication scheme for the network is defined as the average time it takes for a bit to reach its k destination nodes after leaving its source node. The averaging is over all bits transmitted in the network.

Definition of Energy-Per-Bit: Define energy-per-bit $E(n, k)$ as the average energy required to carry one bit from a source node to one of its k destination nodes.

[3]This assumption is also made in reference paper [1].

[4]The assumption is proved to be reasonable on page 3, the first paragraph in [1].



(a) Three-step structure.

(b) Multi-hop MIMO transmission.

(c) A converge multicast transmission frame.

Fig. 1. Transmission strategy of hierarchical cooperation.

III. Transmission Strategy

A. General Multicast Structure

The key idea of our multicast structure is dividing the network into *clusters* with equal number of nodes, then the traffic can be transformed into intra- and inter-cluster transmissions. In this way, we divide the network into two *layers*: the clusters and the whole network. We call the prior *lower layer*, and the later *upper layer*. In our two-layer scheme, let n_1 and n_2 be the number of nodes in the lower and upper layer, respectively.

In each multicast session, there is a source node and k randomly chosen destination nodes. Let k_1 be the number of destination nodes in a cluster, and $k_2 = k$ be that in the network. We also call the cluster containing the source node *source cluster*, and clusters containing at least one destination node *destination clusters*. Each multicast session is realized by a three-step structure (see Figure 1-(a)).

1) Step 1: Source node distributes n_1 bits among n_1 nodes in the cluster, one bit for each node. The traffics in this step are unicasts from the source node to $n_1 - 1$ other nodes in the same cluster.
2) Step 2: The nodes in the source cluster transmit simultaneously by implementing *distributed MIMO transmission* to convey data to the destination clusters. There are two ways of MIMO transmissions:
   - **Multi-hop MIMO transmission:** Each source cluster uses MIMO to transmit to a neighboring cluster, which is called *relay cluster*. After each node in the relay cluster receives a MIMO observation, it amplifies the received signal to a desirable power and retransmits it to the following relay cluster in the next chance according to the routing protocol. This process is repeated until all the destination clusters receive MIMO observations. See Figure 1-(b) for illustration.
   - **Direct MIMO transmission:** The nodes in the source cluster *broadcast* the data in the network simultaneously. Then all nodes in the destination clusters can receive a MIMO observation.
3) Step 3: After each destination cluster receives the MIMO transmissions, each node in the cluster holds an obser-

vation. The k_1 destination nodes in the cluster must collect all n_1 observations to decode the transmitted n_1 bits. Thus, the traffics in this step are n_1 multicast sessions, with each node in the cluster acting as a source node. Also, the k_1 destination nodes are identical for all n_1 sessions. Hence, the traffic can also be treated as a *converge multicast problem*, which means all source nodes "converge"[5] their data to a set of destination nodes.

Now consider a network with more layers. The hierarchical recursion of the whole system is shown in Fig. 2. In this way, we have built a hierarchical scheme to achieve the desired throughput. At the lowest layer of the hierarchy, we use simple TDMA protocol to exchange bits for setting up cooperation among small clusters. Combining this with multi-hop MIMO transmissions, we get a higher throughput scheme for cooperation among nodes in larger clusters at the next layer of the hierarchy. Finally, at the top layer of the hierarchy, the size of the cooperation clusters are maximum and the MIMO transmissions are almost over the global scale to meet the desired traffic demands.

B. Four Strategies for cooperative multicast

Following the three-step multicast structure, there are four strategies that can realize the steps. All of them involve a *multi-layer solution*.

- Multi-hop MIMO multicast (MMM): treat the traffic in step 3 as multicast problem, with multi-hop MIMO transmissions. The multicast problem in step 3 can also be solved using the same three-step structure. Implementing the three-step structure recursively we can get a hierarchical solution to multicast problem.
- Direct MIMO multicast (DMM): treat the traffic in step 3 as multicast problem, with direct MIMO transmissions.
- Converge based multi-hop MIMO multicast (CMMM): treat the traffic in step 3 as converge multicast problem, with multi-hop MIMO transmissions. The converge multicast problem can also be solved in a multi-layer manner.
- Converge based direct MIMO multicast (CDMM): treat the traffic in step 3 as converge multicast problem, with direct MIMO transmissions.

For the hierarchical schemes with multiple number of layers, we give the following more detailed definition of converge multicast frame introduced in CMMM and CDMM schemes.

Converge multicast: Consider the cooperative hierarchical scheme with the number of layers to be 2. At layer i, for any destination cluster, there are n_{i-1} nodes in that cluster, with k_{i-1} of them being destinations. The convergecast multicast frame here refers to the traffic pattern where all the n_{i-1} nodes in this destination cluster transmit their data to those k_{i-1} destinations. Here there are n_1 multicast sessions, with each node in the cluster acting as a source node.

[5]Note that the traffic mode is similar to converge-cast in step 3, our multicast analysis can well cover converge-cast case, where sources transmit information to the destination with distinctive data rates.

C. Notations

We use the following notations throughout this paper. First let h be the number of layers which is independent of n and k. Then we give every layer a unique number $1 \leq i \leq h$, indicating the ith layer from the bottom to the top.

Given a layer i, let n_i be the number of nodes and k_i be that of destination nodes for each source node. Apparently, $n_h = n$ and $k_h = k$. Use $n_{c_i} = n_i/n_{i-1}$ to denote the number of clusters, and k_{c_i} to denote that of destination clusters at layer i.

When analyzing strategies, we use m_i to denote the number of multicast sessions at layer i when considering MMM/DMM, or the number of converge multicast frames at layer i when considering CMMM/CDMM.

IV. Analysis of Multicast Throughput

In this section, we first present the information-theoretic upper bound of the multicast throughput. Then we provide strategies that can nearly achieve the upper bound by utilizing cooperation in the network. When analyzing the throughput, we use a "assume-and-verify" method, i.e. we first make some assumptions on the network; after we obtain the results, we verify these assumptions. Using this method, we make our analysis both strict and easy to follow.

A. Upper Bound of Multicast Throughput

To prove the upper bound, we need lower-bound the mutual distance between nodes, which is provided in the following lemma.

Lemma 4.1: In a network with n nodes randomly and uniformly distributed on a unit-square, the minimum distance between any two nodes is $\frac{1}{n^{1+\delta}}$ *whp*[6], for any $\delta > 0$.

Theorem 4.1: In the network with n nodes and each sending packets to k randomly chosen destination nodes, the aggregate multicast throughput is *whp* bounded by

$$T(n,k) \leq p_1 \frac{n \log n}{k}$$

where $p_1 > 0$ is a constant independent of n and k.

Proof: For each source node in the network, we have randomly assigned k destination nodes to it. If the sets of destination nodes for each source node do not intersect with each other, nk nodes will act as destination nodes in total. However, there are only n nodes in the whole network. Thus, by considering the source-destination paring from a reverse view, for each node d, there are on average k nodes $s_1, s_2, \ldots, s_k$ that choose d as one of its destination nodes. Assume each source node transmits data to d at a same rate $\lambda(n,k)$. The total rate $k\lambda(n,k)$ from source nodes $s_i (1 \leq i \leq k)$ to the destination node d is upper-bounded by the capacity of a multiple-input single-output (MISO) channel between d and the rest of the network. Using a standard formula for this

[6]In this paper, *whp* stands for with high probability, which means the probability tends to 1 as $n \to \infty$.

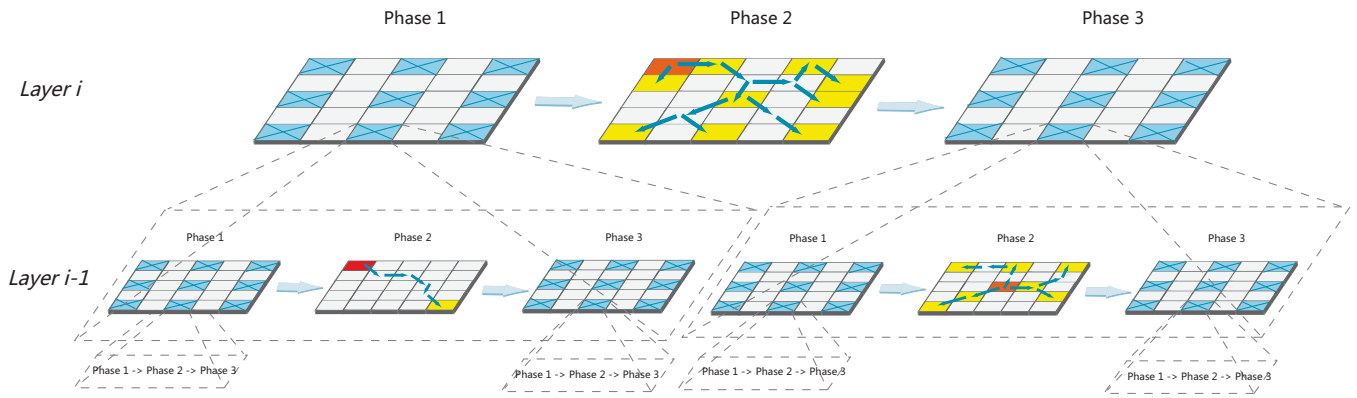


Fig. 2. A whole view of the hierarchical cooperative multicast scheme.

channel, we get

$$k\lambda(n,k) \leq \log\left(1+\frac{P}{N_0}\sum_{\substack{i=1\\ s_i\neq d}}^{n}|g_{s_id}|^2\right) = \log\left(1+\frac{P}{N_0}\sum_{\substack{i=1\\ s_i\neq d}}^{n}\frac{G}{d_{s_id}^{\alpha}}\right).$$

According to Lemma 4.1, the distance between $s_i(1\leq i\leq k)$ and d is larger than $\frac{1}{n^{1+\delta}}$ *whp*. Using this fact, we obtain *whp*

$$\lambda(n,k)\leq\frac{1}{k}\log\left(1+\frac{GP}{N_0}n^{\alpha(1+\delta)+1}\right)\leq\frac{p_1\log n}{k}$$

for some constant p_1 independent of n and k. The theorem then follows. ■

B. Throughput Analysis with MMM

To ensure successful MIMO transmissions, there must be same number of nodes in each cluster. The following lemma ensures the number of nodes in each cluster at layer $2\leq i\leq h$ has the same order. For simplicity, we consider the number of nodes in each cluster is exactly n_{i-1}.

Lemma 4.2: Consider n_i nodes uniformly distributed in the network area. Divide the network into n_{c_i} identical square-shaped clusters. Then the number of nodes in each cluster is $n_{i-1}=\frac{n_i}{n_{c_i}}$ *whp* when **Assumption 1: $n_i=\Omega(n_{c_i}\log n_{c_i})$** is satisfied.

Remark 4.1: Note that the purpose of Lemma 4.2 is to show the relationship between the number of nodes at layer i, denoted by n_i, and the number of cells at layer i, namely, n_{c_i}. It does not aim to show how n_{c_i} depends on n. Actually, how n_{c_i} varies at each layer not only depends on n, but on the number of total layers h and the property of the cooperative scheme adopted as well. Under different ways of hierarchical division at each layer will result to different throughput results. In our following MMM, CMMM, DMM and CDMM schemes, the detailed dependency of n_{c_i} on n can be revealed during the analysis on throughput and delay.

As mentioned, to solve the MP(n,k) in the network area, we divide it into three steps. Since the problems in step 1 and 3 are also multicast problems[7], we can apply the three steps recursively and build a h-layer solution.

1) Solution to Multicast Problem: We consider the ith layer in the network $(2\leq i\leq h)$ and follow the three steps.

Step 1. Preparing for Cooperation: Given the total number of multicast sessions m_i at layer i, each node holds $\frac{m_i}{n_i}$ bits that need to multicast. In this step, each node must distribute all its data to other nodes in the same cluster, $\frac{m_i}{n_in_{i-1}}$ bits for each one. Considering n_{i-1} source nodes in each cluster, the traffic load are $\Theta\left(\frac{m_in_{i-1}}{n_i}\right)$ bits. Since the data exchanges only involve intra-cluster communication, they can work according to the 9-TDMA scheme. We divide the time into slots; at each time slot, let the neighboring eight clusters keep silent when the centric cluster is exchanging data. According to the channel model (2), we assume the received interference signal $I_r(t)$ is a collection of uncorrelated zero-mean stationary and ergodic random processes with power upper-bounded by a constant.[8] This assumption is also made in the proof of Lemma 3.1 [2]. Thus, the power of destructive interference is bounded, enabling clusters operate simultaneously in 9-TDMA manner. This is ensured by Lemma 4.3.

Lemma 4.3: By 9-TDMA scheme, when $\alpha>2$, one node in each cluster has a chance to operate data exchanges at a constant transmission rate. Also when $\alpha>2$, the interfering power received by a node from the simultaneously operating clusters is upper-bounded by a constant.

Assume an aggregate unicast throughput of $\Theta(n_{i-1}^a), 0\leq a\leq 1$ can be achieved for every possible source-destination paring at layer $(i-1)$. Given a traffic load of $\Theta\left(\frac{m_in_{i-1}}{n_i}\right)$ bits, this step can be completed in $\Theta\left(\frac{m_in_{i-1}^{1-a}}{n_i}\right)$ time slots.

Step 2. Multi-hop MIMO Transmissions: In this step, each source cluster starts a series of MIMO transmissions to reach all its corresponding destination clusters in multi-hop manner. To achieve the asymptotically optimal multicast throughput, we construct a multicast tree (MT) by adopting Algorithm 1 in [26], spanning from a source cluster S_i and its

[7]We view unicast as a special case of multicast problem.

[8]This assumption is also needed in other strategies. We will not repeat. Also note that negligible channel interference is one of the basic catches that make both our work and analysis go through. Without the guarantee of constant bounded interference, we cannot ensure the high decoding probability at the receiving nodes.

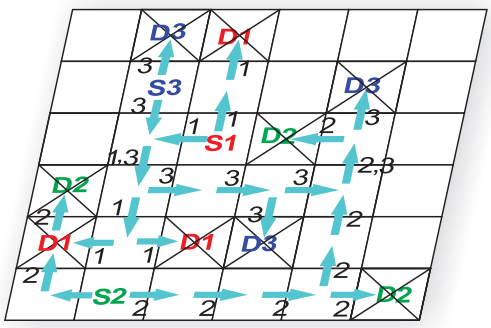


Fig. 3. An example of three MTs in multi-hop MIMO transmission. S_i denotes a source cluster and D_i is one of its destination clusters. The number on the arrow indicates which MT it serves. For each pair of neighboring clusters, the communication between them may involve data from different sources.

corresponding destination clusters D_{ij}, where $1 \le j \le k_{c_i}$. Let $P_i = \{S_i, D_{ij}, 1 \le j \le k_{c_i}\}$.

For simplicity, we do not present the detailed algorithm of how to constructing an MT here since it is not our major contribution. Briefly speaking, the algorithm is conducted through the following way:

For a set of nodes P_i containing a super source node and its super destination nodes, we first build an Euclidean spanning tree, denoted as EST (P_i), to connect them. For each link uv in EST (P_i), we decompose it into a Manhattan path connecting u and v to form Manhattan routing tree $MRT(P_i)$. Then, for each edge uw in $MRT(P_i)$, we connect super nodes crossed by uw in sequence. The final tree is called multicast tree MT.

The constructed MT possesses properties below and hence we can acquire Lemma 4.4.

- *The maximum length of each hop at layer i is* $\Theta\Big(\sqrt{\frac{n_{i-1}}{n}}\Big)$.
- *The total length of MT(P_i) is at most* $O\Big(\sqrt{k_{c_i}} \times \sqrt{\frac{n_i}{n}}\Big)$.

Lemma 4.4: The number of hops in MT is $O\Big(\sqrt{\frac{n_i k_{c_i}}{n_{i-1}}}\Big)$.

Accounting all m_i multicast sessions, at layer i there are $\frac{m_i}{n_{i-1}}$ MTs, and the total number of hops is $O\Big(\frac{m_i}{n_{i-1}}\sqrt{\frac{n_i k_{c_i}}{n_{i-1}}}\Big)$. Using the 9-TDMA scheduling, each cluster is allowed to take MIMO transmission in every nine time slots. If a cluster serves as a relay cluster for multiple multicast sessions, it will deliver the packets of different sessions including its own packets with equal probability. See Figure 3 for illustration. Hence, according to our protocol, at each time slot $\Theta\Big(\frac{n_{c_i}}{9}\Big)$ clusters can transmit simultaneously. The total amount of time to accomplish all m_i sessions' MIMO transmissions is no more than $O\Big(m_i\sqrt{\frac{k_{c_i}}{n_i n_{i-1}}}\Big)$.

Step 3. Cooperative Decoding: Now that each MT has k_{c_i} destination clusters, after step 2, every cluster receives $\Theta\Big(\frac{m_i k_{c_i}}{n_i}\Big)$ MIMO transmissions[9]. For each MIMO transmission, every node in a destination cluster obtains an observation of the n_{i-1} bits transmitted from the source node. To decode the original n_{i-1} bits, all nodes in the destination cluster must first quantify each observation into Q bits, where Q is a constant. Then each node conveys the Q bits to all k_{i-1} destination nodes in the cluster. Clearly, this procedure is a MP(n_{i-1}, k_{i-1}). After all observation results reach the destination nodes, they can decode the transmitted n_{i-1} bits.

Assume an aggregate multicast throughput $\widetilde{\Theta}(n_{i-1}^a k_{i-1}^b)$ is achievable at layer $(i-1)$ *whp*, where $0 \le a \le 1, -1 \le b \le 0$, and $a + b \le 0$. Then MP(n_{i-1}, k_{i-1}) can be solved within $\widetilde{\Theta}\Big(\frac{Q n_{i-1}}{n_{i-1}^a k_{i-1}^b}\Big)$ time slots. Note each cluster receives $\Theta\Big(\frac{m_i k_{c_i}}{n_i}\Big)$ MIMO transmissions, and needs to perform this decoding process for each transmission. By utilizing the 9-TDMA scheme, we can finish all $m_{i-1} = m_i k_{c_i}$ multicast sessions in $\Theta\Big(\frac{m_i k_{c_i}}{n_i}\Big)$ rounds. Thus, step 3 costs $\widetilde{\Theta}\Big(\frac{m_i n_{i-1}^{1-a} k_{c_i}}{n_i k_{i-1}^b}\Big)$ time slots.

For the last part of our solution, we specify the transmission at the bottom layer. In each session, every node broadcasts its data. Then each time, all destination nodes can receive one bit. Thus a multicast session can be completed in one time slot.

2) The Division of Network: By minimizing the total time cost during the three steps at layer i, we present the throughput-optimal division of the network. First, we have

Lemma 4.5: Given k_i independently and uniformly distributed destination nodes in the network at layer i. The number of destination clusters k_{c_i} is given by

$$k_{c_i} = \begin{cases} \Theta(k_i), & \text{when } k_i = O(n_{c_i}), \\ \Theta(\frac{n_i}{n_{i-1}}), & \text{when } k_i = \Omega(n_{c_i}). \end{cases}$$

Lemma 4.6: When **Assumption 2:** $\boldsymbol{m_h = O\big((n_{c_i})^{p_2}\big)}$ holds for all $2 \le i \le h$ with a constant $p_2 > 0$:

(a) if $k_i = \Omega(n_{c_i} \log n_{c_i})$, then $k_{i-1} = \Theta\Big(\frac{k_i}{n_{c_i}}\Big)$ *whp*;

(b) if $k_i = O(n_{c_i} \log n_{c_i})$, then $k_{i-1} = O(\log n_{c_i})$ *whp*.

In the following Lemma 4.7, we use l_i to denote the number of destination sets in each cluster. More specifically, let each source node choose a set of destination nodes in the network, and l_i is the number of source nodes that choose at least one destination node in a layer i network. Thus, for MMM/DMM, in which m_i is the number of multicast sessions, we acquire $l_i = m_i / \prod_{j=i+1}^{h} n_{c_j}$; for CMMM/CDMM, in which m_i is the number of converge multicast frames, we acquire $l_i = m_i / \prod_{j=i+1}^{h-1} n_{c_j}$.

Lemma 4.7: When $k_i = o(n_{c_i})$, the number of destination sets at the $(i-1)$th layer l_{i-1} is

(a) when **Assumption 3:** $\boldsymbol{l_i = \Omega\Big(\frac{n_{c_i}}{k_i} \log \frac{n_{c_i}}{k_i}\Big)}$ is satisfied, then *whp* $l_{i-1} = \Theta\Big(\frac{l_i k_i}{n_{c_i}}\Big)$;

(b) when $l_i = O\Big(\frac{n_{c_i}}{k_i} \log \frac{n_{c_i}}{k_i}\Big)$, then *whp* $l_{i-1} = O\Big(\log \frac{n_{c_i}}{k_i}\Big)$.

Now we are ready to present our network division scheme.

Lemma 4.8: When $k = O(n^{1-\epsilon})$ for a small $\epsilon > 0$, the number of nodes at each layer to achieve optimal throughput in MMM strategy is given by

$$n_i = \begin{cases} (\frac{n}{k})^{\frac{2i-1}{2h-1}}, & i < h, \\ n, & i = h. \end{cases} \tag{3}$$

[9]This is valid under assumption 3 in Lemma 4.7, which we present later.

Proof: Still we consider the three steps at layer i. When assumptions 1 and 3 are satisfied, combining the three steps, the total time to complete m_i multicast sessions is

$$\Theta\left(\frac{m_i n_{i-1}^{1-a}}{n_i}\right)+O\left(m_i\sqrt{\frac{k_{c_i}}{n_i n_{i-1}}}\right)+\widetilde{\Theta}\left(\frac{m_i n_{i-1}^{1-a}k_{c_i}}{n_i k_{i-1}^{b}}\right) \quad (4)$$

Since the time cost on step 3 is always longer than that on step 1 in the order sense, the throughput at layer i is given by

$$\begin{aligned} T(n_i,k_i) &= \frac{m_i}{\Theta\left(\frac{m_i n_{i-1}^{1-a}}{n_i}\right)+O\left(m_i\sqrt{\frac{k_{c_i}}{n_i n_{i-1}}}\right)+\widetilde{\Theta}\left(\frac{m_i n_{i-1}^{1-a}k_{c_i}}{n_i k_{i-1}^{b}}\right)} \\ &= \widetilde{\Theta}\left(\frac{n_i n_{i-1}}{\sqrt{n_i n_{i-1}k_{c_i}}+n_{i-1}^{2-a}k_{i-1}^{-b}k_{c_i}}\right) \end{aligned} \quad (5)$$

To optimize the network division at layer i, we consider two cases: $n_{c_i}=O(k_i)$ and $n_{c_i}=\Omega(k_i)$[10]. Note we suppose the assumption 2 is satisfied. According to Lemmas 4.5 and 4.6, the properties of two cases are summarized below.

- Case 1: When $n_{c_i}=O(k_i)$, then $k_{c_i}=\Theta(n_{c_i})$, $k_{i-1}=\widetilde{\Theta}\left(\frac{k_i}{n_{c_i}}\right)$;
- Case 2: When $n_{c_i}=\Omega(k_i)$, then $k_{c_i}=\Theta(k_i)$, $k_{i-1}=O(\log n_{c_i})=\widetilde{\Theta}(1)$.

In case 1, the throughput in (5) can be written as

$$T(n,k)=\widetilde{\Theta}\left(\frac{n_i n_{i-1}}{n_i+n_{i-1}^{1-a-b}k_i^{-b}n_i^{1+b}}\right) \quad (6)$$

The result is optimized when $n_{i-1}=(\frac{n_i}{k_i})^{\frac{b}{1-a-b}}$. However, since case 1 requires that $n_{c_i}=O(k_i)$, or $n_{i-1}=\Omega(\frac{n_i}{k_i})$, the optimal result cannot be achieved. Thus the maximum achievable throughput in case 1 is $\widetilde{\Theta}\left(\frac{n_i}{k_i+n_i^{1-a}k_i^a}\right)$ when choosing $n_{i-1}=n_i/k_i$, which is not superior to the throughput at the $(i-1)$th layer.

In case 2, the throughput in (5) can be written as

$$T(n,k)=\widetilde{\Theta}\left(\frac{n_i n_{i-1}}{\sqrt{n_i k_i/n_{i-1}}+n_{i-1}^{2-a}k_i}\right) \quad (7)$$

The result is optimized when $n_{i-1}=(\frac{n_i}{k_i})^{\frac{1}{3-2a}}$. Since the inequality $(\frac{n_i}{k_i})^{\frac{1}{3-2a}}<\frac{n_i}{k_i}$ holds, we can achieve a throughput of $\widetilde{\Theta}\left((\frac{n_i}{k_i})^{\frac{2-a}{3-2a}}\right)$, which is better than the throughput at the $(i-1)$th layer as $0\le a<1$. Therefore, we can improve the throughput by adopting case 2.

At bottom layer, the aggregate multicast throughput is $T(n_1,k_1)=1$. When dividing the network in the optimal way at each layer, the relationship of n_i, k_i and throughput in each layer is as follows

$$\begin{aligned} n_h &= k_h n_{h-1}^{\frac{2h-1}{2h-3}}, \quad (\frac{n_h}{k_h})^{\frac{2h-2}{2h-1}} \\ &\vdots \\ n_3 &= n_2^{\frac{5}{3}}, \quad (\frac{n_3}{k_3})^{4/5} \\ n_2 &= n_1^3, \quad (\frac{n_2}{k_2})^{2/3} \end{aligned} \quad (8)$$

[10]The network division is equivalent to power control. By optimal network division, a node does not need to transmit with full power. This can well solve the problem of limited power.

Note $n_h=n, k_h=k$. Substituting this into (8), it yields (3). This finishes the proof. ■

Remark 4.2: Now the number of sessions at each layer is $m_i=n\prod_{j=i+1}^{h}k_j=\widetilde{\Theta}(nk)$. Under this condition, when (3) is satisfied, the time spent at each layer is identical in the order sense, i.e. it takes the same amount of time on the broadcast transmission at bottom layer and multi-hop MIMO transmission at every other layer. However, when $m_i=\Theta(nk)$ do not exactly holds, the throughput of the network is determined by the layer with the maximum number of sessions $\max_{1\le i\le h-1}\{m_i\}$. Note this conclusion also holds for CMMM strategy, with m_i denoting the number of frames at each layer. To get the precise throughput result of MMM strategy, we must further calculate the number of multicast sessions at each layer.

3) The Verification of Assumptions: To calculate the accurate throughput result, there are three conditions need justification. We now consider these factors under (3).

a) First we consider assumptions 1 and 2. According to our multicast traffic in the network, the number of multicast sessions at the top layer is $m_h=n$, which is smaller than $n_{c_i}^h$ for $2\le i\le h$. Thus, assumption 2 holds. As for assumption 1, obviously $k_i=O(\log n_{c_{i+1}})=O(\frac{n_i}{\log n_{c_i}})$ for $1\le i\le h-1$. Considering the top layer, $k=O(\frac{n}{n_{c_h}})$ satisfies when $k=O(n/\log^{\frac{2h-1}{2h-3}}n)$. Since we only consider the case $k=O(n^{1-\epsilon})$ for a small $\epsilon>0$, assumption 1 is also satisfied.

b) Then we consider the number of destination nodes at each layer. By Lemma 4.6

$$k_i=\begin{cases}O\Big(\log(\frac{n}{k})^{\frac{2}{2h-1}}\Big), & 1\le i\le h-2,\\ O\Big(\log k(\frac{n}{k})^{\frac{2}{2h-1}}\Big), & i=h-1.\end{cases}$$

This will change the number of sessions to

$$\begin{aligned} m_h &= n \\ m_{h-1} &= nk \\ m_{h-2} &= nk\log(\frac{n}{k}) \\ &\vdots \\ m_1 &= nk\log^{h-2}(\frac{n}{k}) \end{aligned} \quad (9)$$

c) In our scheme, Lemma 4.7-(a) must be applied recursively. Each time, we have to ensure assumption 3 is satisfied. Recall the number of destination sets is given by

$$l_i=\frac{m_i}{\prod_{j=i+1}^{h}n_{c_j}}=\frac{m_i}{k(n/k)^{\frac{2h-2i}{2h-1}}}$$

Combining (9), we obtain $l_i=\Omega\Big((\frac{n}{k})^{\frac{2}{2h-1}}\Big)$. Note in our network division $\frac{n_{c_i}}{\log n_{c_i}}\log\frac{n_{c_i}}{\log n_{c_i}}=\Theta\Big((\frac{n}{k})^{\frac{2}{2h-1}}\Big)$ for $2\le i\le h-1$. Thus $l_i=\Omega\Big(\frac{n_{c_i}}{k_i}\log\frac{n_{c_i}}{k_i}\Big)$, and assumption 3 is satisfied for all layers.

4) The Calculation of Throughput: From the analysis above, plus the conclusion of Remark 4.2, the throughput is determined by the number of sessions at the bottom layer

because $m_1 = \max_{1\le i\le h-1}\{m_i\} = nk\log^{h-2}(\frac{n}{k})$. Followed by (5), the throughput is

$$T(n,k) = \Theta\Big(\Big(\frac{n}{k}\Big)^{\frac{2h-2}{2h-1}}\log^{-(h-2)}\frac{n}{k}\Big). \tag{10}$$

Then the following theorem naturally holds.

Theorem 4.2: By using MMM strategy, we can achieve an aggregate throughput of

$$T(n,k) = \Theta\Big(\Big(\frac{n}{k}\Big)^{\frac{2h-2}{2h-1}}\log^{-(h-2)}\frac{n}{k}\Big). \tag{11}$$

C. Throughput Analysis with CMMM

Consider three top layers h, $h-1$ and $h-2$, and call layer $h-1$ and $h-2$ as "clusters" and "sub-clusters" respectively. We organize $\frac{n_{h-1}}{n_{h-2}}$ *rounds* of transmission and for each round, choose a sub-cluster in every cluster ($\frac{n_h n_{h-2}}{n_{h-1}}$ source nodes per round). At each round, only nodes in the chosen sub-clusters serve as source nodes. We divide a round into three steps.

Step 1. Preparing for Cooperation: Each source node in the chosen sub-clusters must deliver n_{h-1} bits to nodes in the same cluster for cooperation, one bit for each node. This includes two sub-steps:

- **Sub-Step 1. MIMO Transmissions:** In a specific cluster, each node acts as a destination node. For each destination node d, the chosen sub-cluster uses direct MIMO transmission[11] to communicate with the sub-cluster where d locates. This takes n_{h-1} time slots to accomplish.
- **Sub-Step 2. Cooperate Decoding:** All sub-clusters in the network work in parallel to decode. This sub-step is a CMP$(n_{h-2}, n_{h-2}, 1)$.

Step 2. Multi-hop MIMO Transmission: After step 1, all source nodes in the chosen sub-cluster have distributed their n_{h-1} bits among the nodes in the same cluster. To use multi-hop MIMO transmission, we must build $\frac{n_h n_{h-2}}{n_{h-1}}$ MTs, each corresponding to a source node. According to Lemma 4.4 and the 9-TDMA scheme, step 2 can be completed in $\Theta\Big(n_{h-2}\sqrt{\frac{n_h k_{c_h}}{n_{h-1}}}\Big)$ time slots.

Step 3. Cooperative Decoding: Each destination cluster works in parallel and decodes the original n_{h-2} bits from MIMO observations. The decoding process can be treated as an CMP$(n_{h-1}, m_{h-1}, k_{h-1})$, with $m_{h-1} = n_{h-2}k_{c_h}$. This conclusion is based on assumption 3.

1) Solution to Converge Multicast Problem: We start by studying a two-layer network. Given a CMP(n_2, m_2, k_2), we divide the network into clusters of n_1 nodes. A frame of transmission includes the following steps.

Step 1: After the division of clusters, there are k_{c_2} destination clusters. Since all n_2 nodes must send one bit to k_2 destination nodes, all n_{c_2} clusters must act as source clusters and transmit to k_{c_2} destination clusters using MIMO.

For each of the n_{c_2} source clusters, build a MT connecting the source and destination clusters. By Lemma 4.4, we can finish all the transmissions on MTs in $O\Big(\sqrt{\frac{n_2 k_{c_2}}{n_1}}\Big)$ slots. Considering m_2 frames, the time cost in step 1 is $O\Big(m_2\sqrt{\frac{n_2 k_{c_2}}{n_1}}\Big)$.

Step 2: After a destination cluster receives a MIMO transmission, all n_1 nodes must quantify the observation and converge them to the destination nodes in the cluster. This is a converge multicast problem. When assumption 3 is satisfied, there are $m_1 = \Theta\Big(\frac{m_2 k_{c_2}}{n_{c_2}}\Big)$ frames that choose a cluster as destination cluster. Thus, there is a CMP(n_1, m_1, k_1) in each cluster.

Since the problem in step 2 is also a converge multicast problem, our two-step scheme can be applied recursively to construct a hierarchical solution. In our CMMM strategy, we build a $(h-1)$-layer strategy for step 3. Plus the top layer, there is a total of h layers.

At last, we specify the transmission of the bottom layer. For each frame, every node broadcasts its data and all destination nodes can receive one bit per time slot. Then a frame can be completed in n_1 time slots.

2) The Division of Network: Similar to MMM strategy, we first present the throughput-optimal network division.

Lemma 4.9: When $k = O(n^{1-\epsilon})$ for a small $\epsilon > 0$, the number of nodes at each layer to achieve optimal throughput in CMMM strategy is given by

$$n_i = \begin{cases}(\frac{n}{k})^{\frac{2i-1}{2h-1}}, & i < h,\\ n, & i = h.\end{cases} \tag{12}$$

Proof: Consider two layers i and $i-1$, with $2 \le i \le h-1$. Assume at the $(i-1)$th layer, CMP$(n_{i-1}, m_{i-1}, k_{i-1})$ can be solved in $\widetilde{\Theta}(m_{i-1}n_{i-1}^a k_{i-1}^b)$ time slots. Similar to the analysis of MMM strategy, we assume that assumptions 1, 2 and 3 are satisfied. Then we have $m_{i-1} = \Theta(m_i k_{c_i})$ Still, we consider two cases: $n_{c_i} = O(k_i)$ and $n_{c_i} = \Omega(k_i)$, with the properties still hold.

- Case 1: When $n_{c_i} = O(k_i)$, then $k_{c_i} = \Theta(n_{c_i})$, $k_{i-1} = \Theta\Big(\frac{k_i}{n_{c_i}}\Big)$;
- Case 2: When $n_{c_i} = \Omega(k_i)$, then $k_{c_i} = \Theta(k_i)$, $k_{i-1} = O(\log n_{c_i}) = \widetilde{\Theta}(1)$.

In case 1, the CMP(n_i, m_i, k_i) can be solved in

$$m_i\sqrt{\frac{n_i k_{c_i}}{n_{i-1}}} + m_{i-1}n_{i-1}^a k_{i-1}^b = \frac{m_i n_i}{n_{i-1}} + m_i n_i^{1-b}k_i^b n_{i-1}^{a+b-1} \tag{13}$$

time slots. The result is optimized by choosing $n_{i-1} = (\frac{n_i}{k_i})^{\frac{b}{a+b}}$. However, since $(\frac{n_i}{k_i})^{\frac{b}{a+b}} < \frac{n_i}{k_i}$, which contradicts with the requirement $n_{i-1} = \Omega(\frac{n_i}{k_i})$ of case 1. Thus the minimum time to solve the CMP(n_i, m_i, k_i) is $m_i n_i^a k_i^{1-a}$, which is achieved when $n_{i-1} = n_i/k_i$. This is not superior to the solving time at the $(i-1)$th layer.

In case 2, the CMP(n_i, m_i, k_i) can be solved in

$$m_i\sqrt{\frac{n_i k_{c_i}}{n_{i-1}}} + m_{i-1}n_{i-1}^a k_{i-1}^b = m_i\sqrt{\frac{n_i k_i}{n_{i-1}}} + m_i k_i n_{i-1}^a \tag{14}$$

time slots. The result is optimized by choosing $n_{i-1} = (\frac{n_i}{k_i})^{\frac{1}{2a+1}}$. Since the equation $(\frac{n_i}{k_i})^{\frac{1}{2a+1}} < \frac{n_i}{k_i}$ holds, CMP(n_i, m_i, k_i) can be solved in $m n_{i+1}^{\frac{a}{2a+1}} k_{i+1}^{\frac{a+1}{2a+1}}$ time slots,

[11] Because the time cost in step 1 is not the dominating factor on throughput, this will not affect the result. The reason we do not use multi-hop is that the traffic is not uniformly distributed and is hard to schedule by TDMA scheme.

which is better than the solving time at ith layer. Therefore, we can shorten the solving time by adopting case 2.

At bottom layer, a frame can be finished in n_1 time slots. When we divide the network in the optimal way at each layer, the relationship of n_i, k_i and solving time in each layer from 1 to $h-1$ is shown as follows

$$\begin{aligned} n_{h-1} &= k_{h-1} n_{h-2}^{\frac{2h-3}{2h-5}}, \quad n_{h-1}^{\frac{1}{2h-3}} k_{h-1}^{\frac{2h-4}{2h-3}} \\ &\vdots \\ n_3 &= n_2^{\frac{5}{3}}, \qquad n_3^{1/5} k_3^{4/5} \\ n_2 &= n_1^3, \qquad n_2^{1/3} k_2^{2/3} \end{aligned} \tag{15}$$

Thus, the minimum solving time of CMP$(n_{h-1}, m_{h-1}, k_{h-1})$ is $\widetilde{\Theta}\Big(m_{h-1} n_{h-1}^{\frac{1}{2h-3}} k_{h-1}^{\frac{2h-4}{2h-3}}\Big)$.

Accounting all procedures together, at every round of transmission, we deliver $n_{h-1} \times n_{h-2} \times \frac{n_h}{n_{h-1}}$ bits to their destination nodes in

$$\Big(n_{h-1} + n_{h-2}^{\frac{2h-6}{2h-5}}\Big) + n_{h-2}\sqrt{\frac{n_h k_{c_h}}{n_{h-1}}} + n_{h-2} k_h n_{h-1}^{\frac{1}{2h-3}} k_{n-1}^{\frac{2h-4}{2h-3}} \tag{16}$$

time slots. So the aggregate throughput is given by

$$\frac{n_{h-1} \times n_{h-2} \times \frac{n_h}{n_{h-1}}}{\Big(n_{h-1} + n_{h-2}^{\frac{2h-6}{2h-5}}\Big) + n_{h-2}\sqrt{\frac{n_h k_{c_h}}{n_{h-1}}} + n_{h-2} k_h n_{h-1}^{\frac{1}{2h-3}} k_{n-1}^{\frac{2h-4}{2h-3}}} \tag{17}$$

We optimize the result by choosing $n_{h-1} = (\frac{n_h}{k_h})^{\frac{2h-3}{2h-1}}$. Combining with (15), we obtain (12). This finishes the proof. ∎

3) The Verification of Assumptions: Before presenting the throughput result, the three conditions in section IV-B3 also need justification.

a) To begin with, the verification procedure of assumptions 1 and 2 is identical to MMM strategy, and the assumptions are satisfied. For simplicity we omit the details.

b) Then we consider the number of destination nodes at each layer. Accounting $m_{h-1} = k(\frac{n}{k})^{\frac{2h-5}{2h-1}}$ and $k_{i-1} = \log n_{c_i}$ for $2 \le i \le h$:

$$\begin{aligned} m_{h-1} &= k\Big(\frac{n}{k}\Big)^{\frac{2h-5}{2h-1}} \\ m_{h-2} &= k\Big(\frac{n}{k}\Big)^{\frac{2h-5}{2h-1}} \log\Big(\frac{n}{k}\Big) \\ &\vdots \\ m_1 &= k\Big(\frac{n}{k}\Big)^{\frac{2h-5}{2h-1}} \log^{h-2}\Big(\frac{n}{k}\Big) \end{aligned} \tag{18}$$

c) In our scheme, Lemma 4.7-(a) must be applied recursively. Each time, we need to ensure that assumption 3 is satisfied. Recall the number of destination sets is given by

$$l_i = \frac{m_i}{\prod_{j=i+1}^{h-1} n_{c_j}} = \frac{m_i}{(n/k)^{\frac{2h-2i-2}{2h-1}}}$$

The equation holds under the network division (12). Combining (18), we obtain $l_i = \Omega\Big((\frac{n}{k})^{\frac{2}{2h-1}}\Big) = \Omega\Big(\frac{n_{c_i}}{\log n_{c_i}} \log \frac{n_{c_i}}{\log n_{c_i}}\Big)$ for $3 \le i \le h-1$, and assumption 3 is satisfied. However, when $i = 2$,

$$l_2 = k\Big(\frac{n}{k}\Big)^{\frac{1}{2h-1}} \log^{h-3}\Big(\frac{n}{k}\Big).$$

Comparing l_2 with $(\frac{n}{k})^{\frac{2}{2h-1}}$, there exist a threshold[12]

$$k_{th} = \Theta\Big(n^{\frac{1}{2h}} \log^{\frac{(h-3)(2h-1)}{2h}} n\Big) = \widetilde{\Theta}\Big(n^{\frac{1}{2h}}\Big). \tag{19}$$

When $k = \Omega(k_{th})$, assumption 3 holds for layer 2, otherwise it does not. Thus the number of frames at bottom layer is thus given by

$$m_1 = \begin{cases} (\frac{n}{k})^{\frac{2h-4}{2h-1}} \log(\frac{n}{k}), & \text{when } k = O(k_{th}), \\ k(\frac{n}{k})^{\frac{2h-5}{2h-1}} \log^{h-2}(\frac{n}{k}), & \text{when } k = \Omega(k_{th}). \end{cases} \tag{20}$$

4) The Calculation of Throughput: From the above analysis, plus the conclusion of Remark 4.2, the throughput is determined by the number of frames at the bottom layer because $m_1 = \max_{1 \le i \le h-1}\{m_i\}$. Thus, followed by (17) and (20), the throughput is given by

$$T(n,k) = \begin{cases} \Theta\Big(n^{\frac{2h-3}{2h-1}} k^{\frac{2}{2h-1}} \log^{-1}\frac{n}{k}\Big), & \text{when } k = O(k_{th}), \\ \Theta\Big((\frac{n}{k})^{\frac{2h-2}{2h-1}} \log^{-(h-2)}\frac{n}{k}\Big), & \text{when } k = \Omega(k_{th}). \end{cases} \tag{21}$$

Then, we have the following theorem.

Theorem 4.3: By using CMMM strategy, we can achieve an aggregate throughput of

$$T(n,k) = \begin{cases} \Theta\Big(n^{\frac{2h-3}{2h-1}} k^{\frac{2}{2h-1}} \log^{-1}\frac{n}{k}\Big), & \text{when } k = O(n^{\frac{1}{2h}}), \\ \Theta\Big((\frac{n}{k})^{\frac{2h-2}{2h-1}} \log^{-(h-2)}\frac{n}{k}\Big), & \text{when } k = \Omega(n^{\frac{1}{2h}}). \end{cases} \tag{22}$$

D. Broadcast Case

So far we have only proved the throughput result when $k = O(n^{1-\epsilon})$ for an arbitrarily small $\epsilon > 0$. Another case is $k = \widetilde{\Theta}(n)$, which we refer to as *broadcast case*.

According to Theorem 4.2, the network cannot be divided into more than $\widetilde{\Theta}(n_i)$ clusters at layer i. Thus for broadcast case, we can only divide the network as $n_{c_i} = O(k_i)$. This division has been discussed in the proof of Lemma 4.8 and 4.9 (see case 1), and the throughput performance does not increase as the number of layer becomes higher. Thus, there is no gain on the throughput when utilizing our cooperative scheme in the broadcast case, and the throughput results in Theorems 4.2 and 4.3 still hold.

In the rest of this paper, we do not distinguish $k = O(n^{1-\epsilon})$ and $k = \widetilde{\Theta}(n)$, because the conclusions hold for both cases.

E. Throughput Analysis Using Direct MIMO Transmission

DMM and CDMM operate in similar manners to MMM and CMMM, respectively. The only difference is that we use direct MIMO transmission in these two strategies. Because the similarity, we only present some important conclusions and results.

[12]We will discuss the influence of it in Section VI-C

In DMM and CDMM, we perform direct MIMO transmissions at each layer, which takes one time slot for each source clusters. This difference leads to another optimized network division, which is identical for both DMM and CDMM.

$$n_i = \begin{cases} (\frac{n}{k})^{\frac{i}{h}}, & i < h, \\ n, & i = h. \end{cases} \tag{23}$$

Under this division, the throughput results are given by the following theorem.

Theorem 4.4: By using either DMM or CDMM strategy, we can achieve an aggregate throughput

$$T(n,k) = \Theta\left(\left(\frac{n}{k}\right)^{\frac{h-1}{h}} log^{-(h-1)}\frac{n}{k}\right). \tag{24}$$

V. Delay and Energy Consumption Analysis

A. Delay Analysis

1) Delay Analysis with MMM: As mentioned in the previous section, delay performance of MMM is poor. Intuitively, at the ith layer, a source node must divide the data into n_{i-1} parts of the same size and distribute to other nodes for cooperation. This division is repeated at each layer. Since the smallest part of data at the bottom later is one bit, the minimum size of data packets at layer i is $B_i = \prod_{j=1}^{i-1} n_j$ bits.

For the ith layer, let $D(n_i, k_i)$ be the average time to accomplish a multicast session for each of n_i nodes. To analyze the delay, we consider the three steps separately.

1) For step 1, each source node distributes B_i bits to other nodes within the same cluster. Because in this step, all traffic is unicast, the distribution process takes $D(n_{i-1}, 1)$ time slots. We ignore the time spent in step 1 since it is smaller than that in step 3.
2) For step 2, to transmit B_i bits for all n_i source node, there are $n_i B_i / n_{i-1}$ MTs at layer i. The number of hops on each MT at layer i is $\Theta\Big(\sqrt{\frac{n_i k_i}{n_{i-1}}}\Big)$. Using 9-TDMA scheme, we can accomplish $\Theta(\frac{n_i}{n_{i-1}})$ hops per time slot. Thus, we can complete the second step in $\Theta\Big(B_i\sqrt{\frac{n_i k_i}{n_{i-1}}}\Big)$ time slots.
3) For step 3, the traffic load are $n_{i-1}k_i$ multicast sessions in every cluster. Recall we use $D(n_{i-1}, k_{i-1})$ to denote the amount of time to finish the transmission of n_{i-1} multicast sessions at layer $i-1$. Thus, step 3 takes $k_i D(n_{i-1}, k_{i-1})$ time slots.

These three steps cost $D(n_i, k_i)$ time slots. Thus

$$D(n_i,k_i) = \Theta\Big(B_i\sqrt{\frac{n_i k_i}{n_{i-1}}}\Big) + k_i D(n_{i-1}, k_{i-1}) \tag{25}$$

where $B_i = (\frac{n}{k})^{\frac{(i-1)^2}{2i-1}}$ for $1 \le i \le h$. Also by the bottom layer transmission scheme, $D(n_1, k_1) = n_1 = (\frac{n}{k})^{\frac{2}{2h-1}}$. Substituting these into (25) and iterating the equation for $i = 1, 2, \ldots, h$, we then obtain the final result

$$D(n,k) = \Theta\Big(n^{\frac{h^2-2h+2}{2h-1}} k^{-\frac{h^2-4h+3}{2h-1}}\Big) \tag{26}$$

Remark 5.1: Observing the result, the delay is determined by the number of nodes at each layer. And the transmission time at the top layer is the dominating factor on delay. This implies that we can just calculate the time cost at the top layer.

Combining (11) with (26), the delay-throughput tradeoff is

$$D(n,k)/T(n,k) = \Theta\Big(n^{\frac{h^2-4h+3}{2h-1}} k^{-\frac{h^2-6h+4}{2h-1}} \log^{h-2}\frac{n}{k}\Big). \tag{27}$$

2) Delay Analysis with CMMM: In our CMMM strategy, delay is the amount of time that a transmission round spends, and it is calculated when analyzing the throughput. The time cost to finish each round is given by (16). By Lemma 4.9, substituting all parameters by n and k in (16), we obtain the delay

$$D(n,k) = \begin{cases} \Theta\Big((\frac{n}{k})^{\frac{2h-3}{2h-1}} \log\frac{n}{k}\Big), & \text{when } k = O(k_{th}), \\ \Theta\Big(n^{\frac{2h-4}{2h-1}} k^{\frac{3}{2h-1}} \log^{h-2}\frac{n}{k}\Big), & \text{when } k = \Omega(k_{th}), \end{cases} \tag{28}$$

which is simplified as

$$D(n,k) = \begin{cases} \widetilde{\Theta}\Big((\frac{n}{k})^{\frac{2h-3}{2h-1}}\Big), & \text{when } k = O(n^{\frac{1}{2h}}), \\ \widetilde{\Theta}\Big(n^{\frac{2h-4}{2h-1}} k^{\frac{3}{2h-1}}\Big), & \text{when } k = \Omega(n^{\frac{1}{2h}}). \end{cases} \tag{29}$$

Combining with (22), the delay-throughput tradeoff is

$$\frac{D(n,k)}{T(n,k)} = \begin{cases} \Theta\left(k^{-1}\log\frac{n}{k}\right), & \text{when } k = O(n^{\frac{1}{2h}}), \\ \Theta\Big(k(\frac{n}{k})^{-\frac{2}{2h-1}} \log^{h-2}\frac{n}{k}\Big), & \text{when } k = \Omega(n^{\frac{1}{2h}}). \end{cases} \tag{30}$$

3) Delay Analysis with DMM: The delay analyzing procedure of DMM is similar to that of MMM. Thus, we can easily obtain the delay result by the conclusion of Remark 5.1.

For DMM, each time a source node must transmit $B_h = (\frac{n}{k})^{\frac{h-1}{2}}$ bits. And the transmission rate at the top layer is $n^{\frac{1}{h}} k^{\frac{h-1}{h}}$ bit/s using MIMO. Then we derive the delay as

$$D(n,k) = \Theta\Big(n^{\frac{h^2-h+2}{2h}} k^{\frac{h^2-3h+2}{2h}}\Big). \tag{31}$$

Combining with (24), the delay-throughput tradeoff is

$$D(n,k)/T(n,k) = \Theta\Big(n^{\frac{h^2-3h+4}{2h}} k^{-\frac{h^2-5h+4}{2h}} \log^{h-1}\frac{n}{k}\Big). \tag{32}$$

4) Delay Analysis with CDMM: The way we obtain the delay of CDMM is similar to that of CMMM. The result is

$$D(n,k) = \Theta\Big(n^{\frac{h-1}{h}} k^{\frac{1}{h}} \log^{h-1}\frac{n}{k}\Big). \tag{33}$$

Comparing with (26), CMMM strategy reduces the delay dramatically by a factor nearly $(\frac{n}{k})^{\frac{h}{2}}$. Combining with (24), the delay-throughput tradeoff is

$$D(n,k)/T(n,k) = \Theta\left(k\log^{2h-2}\frac{n}{k}\right). \tag{34}$$

B. Energy Consumption Analysis

Suppose the energy consumption for each transmission is proportional to d^α, where d is the distance between the sender and the receiver and $\alpha > 2$ is the path loss exponent. Recall that we define $E(n,k)$ as the energy cost to carry one bit from a source node to one of its k destination nodes. We focus our attention on the energy consumption of MMM strategy. For the rest three strategies, we only present the results, which can be obtained in similar manner.

1) Energy Consumption of MMM: In the MMM strategy, a multicast session is divided into three steps. We consider the three steps respectively. For the ith layer, we use $E(n_i, k_i)$ to denote the energy consumption.

1) For step 1, each source node distributes packets among the network. The amount of traffic load is less than that in step 3 in the order sense. Thus, we need not consider the power spent in this step.
2) For step 2, the number of hops on each MT are $\Theta\Big(\sqrt{\frac{n_i k_i}{n_{i-1}}}\Big)$. For each hop, all n_{i-1} nodes in the sending cluster must transmit to a distance of $\sqrt{\frac{n_{i-1}}{n}}$, which is the side length of a cluster at the ith layer. Thus the energy spent to finish the transmissions on each MT is
$$O\Big(n_{i-1}\sqrt{\frac{n_i k_i}{n_{i-1}}}(\frac{n_{i-1}}{n})^{\frac{\alpha}{2}}\Big)$$
3) For step 3, we will perform $\Theta(n_{i-1}k_i)$ sessions of multicast at layer $(i-1)$, each transmitting Qk_{i-1} bits. Hence, the energy consumption in this step is
$$O\Big(n_{i-1}k_i E(n_{i-1}, k_{i-1})\Big)$$

In these three steps, a total of $n_{i-1}k_i$ bits are transmitted. Combining the above analysis

$$n_{i-1}k_i E(n_i, k_i) = n_{i-1}\sqrt{\frac{n_i k_i}{n_{i-1}}}\Big(\frac{n_{i-1}}{n}\Big)^{\frac{\alpha}{2}} + n_{i-1}k_i E(n_{i-1}, k_{i-1})$$

holds in the order sense. Equivalently we have

$$E(n_i, k_i) = \sqrt{\frac{n_i}{n_{i-1}k_i}}\Big(\frac{n_{i-1}}{n}\Big)^{\frac{\alpha}{2}} + E(n_{i-1}, k_{i-1}) \quad (35)$$

Considering the network division (12) and the factor $k_i = \Omega(1)$ for all layer, we obtain

$$E(n_i, k_i) = n^{\frac{(i-h-1)\alpha+1}{2h-1}} k^{-\frac{2+2i\alpha-3\alpha}{4h-2}} + E(n_{i-1}, k_{i-1}) \quad (36)$$

For $1 \le i \le h-1$, summing (36) up, it yields

$$E(n_{h-1}, k_{h-1}) = \sum_{i=2}^{h-1} n^{\frac{(i-h-1)\alpha+1}{2h-1}} k^{-\frac{2+2i\alpha-3\alpha}{4h-2}} + E(n_1, k_1),$$

where $E(n_1, k_1) = \Big(\sqrt{\frac{n_1}{n}}\Big)^{\alpha} = n^{\frac{(1-h)\alpha}{2h-1}} k^{-\frac{\alpha}{4h-2}}$, which is smaller than the first term on the right-hand-side in the order sense. Thus, the power spent at the $(h-1)$th layer is

$$E(n_{h-1}, k_{h-1}) = O\Big(n^{\frac{-2\alpha+1}{2h-1}} k^{-\frac{2h\alpha-5\alpha+2}{4h-2}}\Big) \quad (37)$$

For $i = h$ in (36), substitute $E(n_{h-1}, k_{h-1})$ with (37), we can obtain the final result

$$E(n, k) = O\Big(n^{\frac{1-\alpha}{2h-1}} k^{-\frac{2h\alpha-3\alpha+2}{4h-2}}\Big) \quad (38)$$

Remark 5.2: Observing the result, we find the energy consumption is determined by the amount of energy spent on the transmissions at the top layer. This is similar in other transmission strategies.

2) Energy Consumption of CMMM: Our CMMM strategy consumes the same amount of energy to transmit a bit as that of MMM strategy, i.e. the equation (38) also holds for CMMM. Through a deeper investigation, two reasons lead to this.

- The network division is identical in two strategies.
- In two strategies, we all build MTs. The number of MTs is the same at each layer, leading to a same amount of power to transmit one bit.

3) Energy Consumption of DMM and CDMM: Intuitively, DMM and CDMM use direct MIMO transmission, which is less energy-efficient than multi-hop MIMO transmission. Using the conclusion of Remark 5.2, we only consider the transmissions at the top layer. At the top layer, to transmit n_{h-1} bits to all $k_{c_h} = \Theta(k)$ destination clusters, nodes in the source cluster broadcast data among the whole network. Thus, the energy to transmit one bit to all $\Theta(k)$ destination clusters is $O(1)$ on average. The result is identical in two strategies, which is presented below.

$$E(n, k) = O\Big(\frac{1}{k}\Big) \quad (39)$$

VI. Discussion

Until now, we have derived all the performance matrices for the four strategies. In this section we will further discuss these results.

A. The Advantage of Cooperation

In our cooperative multicast scheme, we assume that the nodes nearby help each other on transmitting and receiving. Moreover, the hierarchical scheme we propose can bring about great improvements on the throughput only when h is sufficiently large. When setting h to 1, we cannot obtain a good capacity result since the cooperation is not fully utilized in this case. Because the cooperation between nodes becomes stronger as h increases. In such case, we get a $\Theta\Big(\sqrt{\frac{n}{k}}\Big)$ gain on the achievable throughput compared with [26]. And there is also a gain on throughput if compared with the results in [25]. The reason of the improvement is that when using distributed MIMO transmission, we exploit interference cancelation and could transmit many bits simultaneously. This method reduces the average interference level caused by each multicast session, which is the bottleneck of the achievable throughput.

B. The Effect of Different Network Division

Although we use cooperative schemes, there are still cases when throughput cannot be improved. An obvious example is broadcast. In the broadcast case, the number of clusters at each layer is smaller than that of the destination nodes, i.e. $n_{c_i} = O(k_i)$ for $2 \le i \le h$. Moreover, even when $k = O(n^{1-\epsilon})$ for a small $\epsilon > 0$, under this kind of network division we still cannot achieve a gain on the throughput.

Assume at the ith layer, we partition the network as $n_{c_i} = O(k_i)$. Then it follows that $k_{c_i} = \Theta(n_{c_i})$. The reason that we cannot improve the throughput lies on the number of multicast sessions m_i (or converge multicast frames). Since

Strategy		Delay	Throughput	Delay/Throughput	Energy
Multi-hop MIMO Transmission	MMM	$n^{\frac{h^2-2h+2}{2h-1}}k^{-\frac{h^2-4h+3}{2h-1}}$	$(\frac{n}{k})^{\frac{2h-2}{2h-1}}\log^{-(h-2)}\frac{n}{k}$	$n^{\frac{h^2-4h+3}{2h-1}}k^{-\frac{h^2-6h+4}{2h-1}}\log^{h-2}\frac{n}{k}$	$n^{\frac{1-\alpha}{2h-1}}k^{-\frac{2h\alpha-3\alpha+2}{4h-2}}$
	CMMM	$n^{\frac{2h-4}{2h-1}}k^{\frac{3}{2h-1}}\log^{h-2}\frac{n}{k}$		$k(\frac{n}{k})^{-\frac{2}{2h-1}}\log^{h-2}\frac{n}{k}$	
Direct MIMO Transmission	DMM	$n^{\frac{h^2-h+2}{2h}}k^{\frac{h^2-3h+2}{2h}}$	$(\frac{n}{k})^{\frac{h-1}{h}}\log^{-(h-2)}\frac{n}{k}$	$n^{\frac{h^2-3h+4}{2h}}k^{-\frac{h^2-5h+4}{2h}}\log^{h-1}\frac{n}{k}$	$\frac{1}{k}$
	CDMM	$n^{\frac{h-1}{h}}k^{\frac{1}{h}}\log^{h-1}\frac{n}{k}$		$k\log^{2h-2}\frac{n}{k}$	

TABLE I
THE SUMMARIZATION OF PERFORMANCE FOR THE FOUR STRATEGIES. DUE TO SPACE LIMITS, THE EXPRESSIONS IN THE ENERGY COLUMN HAVE OMITTED PREFIX $O(\cdot)$, WHILE OTHER EXPRESSIONS ARE IN THE SENSE OF $\Theta(\cdot)$. NOTE WE ASSUME $k=\Omega(k_{th})$ WHEN CONSIDERING CMMM STRATEGY.

$m_{i-1}=\Theta(m_i k_{c_i})$, we conclude $m_{i-1}=\Theta\big(\frac{m_i n_i}{n_{i-1}}\big)$, which is greater than m_i in the order sense. This means that the transmission scale grows as the layer becomes lower, which cancels the advantage of parallel communications at lower layers, and results in no gain on the achievable throughput.

Besides, in MMM and DMM strategies, the delay decreases as k increases. When performing multicast, we need to transmit $B_h=\prod_{i=1}^{h-1}n_i$ bits to other cooperative nodes to prepare for distributed MIMO, which is also decided by the network division. The time cost on distributing B_h bits is the deterministic factor of delay, and gets smaller when k grows.

C. Delay-Throughput Tradeoff

First of all, we discuss how the number of destination nodes k affect the delay-throughput tradeoff. The delay-throughput tradeoff $D(n,k)/T(n,k)$ under multicast traffic is approximately $D/T=\widetilde{\Theta}(k)$, which is identical to that of non-cooperative schemes. As Figure 4 shows, when k grows, the tradeoff curves of CMMM/CDMM move leftwards, indicating D/T increases. The reason is obvious: when k increases, each source node has to deliver more copies of data among the network. Thus the time to complete a multicast session gets longer, and D/T become larger.

However, exceptions exist: when $k=1$ and $k=n^{0.2}$, the CMMM curves intersect, which means for certain h, multicast D/T may be better than that of unicast. The reason is the existence of k_{th} in (19). In our CMMM strategy, when $k<k_{th}=\widetilde{\Theta}(n^{\frac{1}{2h}})$, the assumption 3 cannot be ensured at the 2nd layer layer, i.e. $l_2=k(\frac{n}{k})^{\frac{1}{2h-1}}\log^{h-3}(\frac{n}{k})=O\Big((\frac{n}{k})^{\frac{2}{2h-1}}\Big)$. Thus l_1 can only be derived by Lemma 4.7-(b): $l_1=O(\log(\frac{n}{k}))$. However, when $k>k_{th}$ and assumption 3 holds, l_1 can be given as $l_1=\Theta\Big(n^{-\frac{1}{2h}}k^{\frac{2h}{2h-1}}\log^{h-2}(\frac{n}{k})\Big)$. Combining the relationship between l_i and m_i, the number of transmission frames at the bottom layer is given in (20), which we repeat it here

$$m_1=\begin{cases}(\frac{n}{k})^{\frac{2h-4}{2h-1}}\log(\frac{n}{k}), & \text{when } k=O(k_{th}),\\ k(\frac{n}{k})^{\frac{2h-5}{2h-1}}\log^{h-2}(\frac{n}{k}), & \text{when } k=\Omega(k_{th}).\end{cases}$$

In unicast, $k=1$ is always below the threshold k_{th}. Thus the number of frames at the bottom layer can only upper bounded by $m_1=\widetilde{\Theta}\big(n^{\frac{2h-4}{2h-1}}\big)$. However, $k=n^{0.2}>k_{th}$ when the number of layer $h>2.5$, and therefore we can bound m_1 by $m_1=\widetilde{\Theta}\Big(k(\frac{n}{k})^{\frac{2h-5}{2h-1}}\log^{h-2}(\frac{n}{k})\Big)$. If we choose $h=3$, then $m_1=\widetilde{\Theta}(n^{0.4})$ when $k=1$; and $m_i=\widetilde{\Theta}(n^{0.36})$ when $k=n^{0.2}$. Hence, in this case, the number of frames at bottom layer of multicast is smaller than that of unicast. By the conclusion of Remark 4.2, the number of frames at the bottom layer will determine the transmission time of each round, which results in a larger D/T of unicast case.

The effect of k_{th} is also embodied in Figure 4. Since the number of destination nodes k is smaller than the threshold $k_{th}=\widetilde{\Theta}(n^{\frac{1}{2h}})$ only when k and h are both small, an typical example is $k=n^{0.1}$, see the solid red line in Figure 4. The lower-left part of it is a straight line, indicating $h\le 5$ and $k<k_{th}$. In this case, the delay-throughput ratio D/T can only be lower-bounded by $\Theta(k^{-1})$. But when $h\ge 5$, $k>k_{th}$ is satisfied and D/T is bounded by $\Theta\Big(k(\frac{n}{k})^{-\frac{2}{2h-1}}\Big)$. This is indicated by the upper-right part of the curve. As for other CMMM curves, the number of destination nodes k is never below the threshold since $h>2$ in our CMMM strategy. Thus, the threshold has no effect on them.

Second, when considering the tradeoff $D(n,k)/T(n,k)$, CMMM has a better performance. However, this tradeoff becomes worse as the number of layers h grows. See Figure 4. Actually in CMMM, the delay is the time to complete a round. For each round, only a number of $n\times\frac{n_{h-2}}{n_{h-1}}$ nodes act as source nodes. When it increases, the time to finish a round will also increase. However, this does not affect the multicast throughput, since the number of bits transmitted in a round is linear with the time cost of a round. Hence, the tradeoff

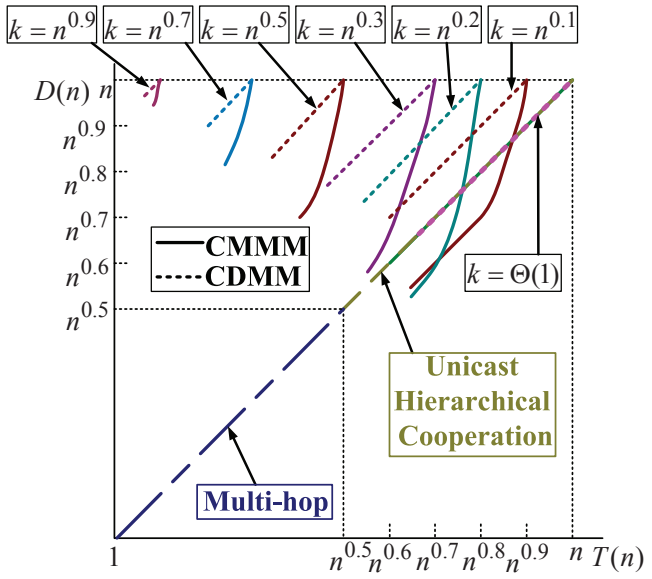


Fig. 4. Throughput-delay tradeoff for CMMM and CDMM compared with known results, the upper-right part of curves is achieved when choosing larger k. When $k = \Theta(1)$, CMMM line covers CDMM one, while the beginning points are different. For two curves with the same k, we use a common color.

ratio D/T increases when the transmission scale of each round grows. Particularly, if all n nodes would act as source nodes in a round, the tradeoff $D/T = k$, which is independent of h. While in our scheme, there are $n \times (\frac{n}{k})^{\frac{2}{2h-1}}$ active nodes each round. The transmission scale grows as h increases, which results in the phenomena above.

At last, another interesting phenomena appears to us. As the results in Section V-A show, the delay-throughput tradeoff results are poor in MMM and DMM strategies. Surprisingly, the tradeoff ratio D/T is identical to that of CMMM and CDMM when $k = \Theta(n)$. Namely, in the broadcast case, the tradeoff results of the four strategies unify to $D/T = n$. To explain this, we explore the common features of the four strategies in broadcast case: (1) The network division is the same. (in broadcast, we only divide each layer into clusters of a constant number) (2) We schedule the transmissions at the bottom layer in a same method. The direct consequences of these features are (1) the size of packets that need to distribute in step 1 is the same ($\Theta(n)$ bits); (2) the time spent on MIMO transmission at each layer is $\widetilde{\Theta}(1)$ for each source cluster; and (3) the identical transmission strategy at the bottom layer result in the same amount of transmission time. Thus, for the four strategies, the throughput and delay are both identical in the broadcast case, leading to the unification of tradeoff ratio. We summarize all the main results (throughput, delay, delay-throughput tradeoff and energy consumption) obtained under our four strategies in TABLE I.

D. Multi-hop vs. Direct MIMO Transmission

For a given h, the throughput and delay of MMM/CMMM are both better than that of DMM/CDMM. Two factors contribute to the less delay. (1) Parallel MIMO transmissions (The average time to complete the transmission of a MT at layer i is $O(\sqrt{n_{i-1}k_{c_i}/n_i})$, which is smaller than that of direct transmission, namely one slot.) (2) Less bits transmitted at each round in CMMM. By reducing the transmission time, multi-hop scheme also improves the throughput. Comparing (11) and (24), the throughput gain is $(\frac{n}{k})^{\frac{h-1}{h(2h-1)}}$. Thus, the delay-throughput tradeoff of CMMM is better than that of CDMM, which is shown in Fig. 4.

As for the energy consumption, multi-hop is approximately $k^{\frac{\alpha-2}{2}}$ times smaller than that of direct MIMO transmission. Intuitively, multi-hop performs several short distance communications, which is more energy efficient than direct manner.

E. MMM vs. Existing Approach in Other Published Papers

Now we consider comparing our MMM scheme with some other existing schemes published by other papers. We compare MMM with several cooperative schemes proposed in [22], [24], [25] and [32], respectively. In [22], [24] and [25], the authors study multicast capacity in protocol model in static networks. In [22] and [24], the authors establish the routing by constructing an Euclidean tree for multicast. Information is then transmitted from source to the destinations through the constructed tree. In [32], the authors study throughput and delay for multicast under mobile networks. They propose several approaches for multicasting transmission such as 2-hop relay without redundancy, 2-hop relay with redundancy and multi-hop relay with redundancy. In [25], the authors consider multicast capacity in a more realistic and less pessimistic channel models. They propose a multicast routing and time scheduling scheme to achieve the computed asymptotic bound over all channel models except the simple Protocol Model.

The throughput comparison is listed in Table II. From the table, we can see that MMM achieves much larger throughput than the scheme proposed in [22], [24] and [25]. In [22], [24] and [25], a large number of extra transmission are wasted for redundancy in the routing process. Moreover, all the adjacent transmission has to be treated as interference while it is efficiently canceled in our MMM scheme. These two factors causes the poor throughput performance in [22], [24] and [25]. Compared with the three relay schemes proposed in [32], our MMM scheme also can guarantee a good aggregate throughput, which is close to the upper bound with a difference of only $\log n$ factor. This is almost the same as the result achieved in 2-hop relay without redundancy in [32], $\Theta\left(\frac{n}{k}\right)$. Moreover, the authors also study multicast capacity in mobile networks under more realistic channel model in [25]. And they achieve the same capacity result $\Theta\left(\frac{n}{k}\right)$, which is also the result in our MMM scheme, when h goes to infinity.

To better demonstrate the gain achieved in our MMM scheme, we also illustrate the throughput performance in Fig. 5, compared with other known results. It can be seen that for any $\epsilon > 0$, our cooperative scheme obtains a throughput of $\Omega\left((\frac{n}{k})^{1-\epsilon}\right)$, with h large enough. However, the delay performance of MMM strategy is poor. Intuitively, this is because each node must transmit a large amount of bits a time to achieve this throughput. Hence, if concerning delay performance, our MMM scheme is not the appropriate choice.

scheme (static)	aggregate throughput
MMM	$\Theta\left(\left(\frac{n}{k}\right)^{\frac{2h-2}{2h-1}}\log^{-(h-2)}\frac{n}{k}\right)$
multi-hop relay in [22]	$\Theta\left(\frac{n}{\sqrt{nk\log n}}\right)$
spanning routing tree in [24]	$\Theta\left(\frac{n}{k\log n}\right)$ $(k=O\left(\frac{n}{\log n}\right))$ $\Theta(1)$ $(k=\Omega\left(\frac{n}{\log n}\right))$
Multi-hop scheme in [25]	$\Theta\left(\sqrt{\frac{n}{k}}\right)$ $(k\le\frac{n}{\log^3 n})$ $\Omega\left(\frac{n}{k\sqrt{\log^3 n}}\right)$ $(\frac{n}{\log^3 n}\le k\le\frac{n}{\log^2 n})$ $\Omega\left(\sqrt{\frac{n}{k\log n}}\right)$ $(\frac{n}{\log^2 n}\le k\le\frac{n}{\log n})$ $\Theta(1)$ $(k\ge\frac{n}{\log n})$

scheme (mobile)	aggregate throughput
routing scheme in [25]	$\Theta\left(\frac{n}{k}\right)$
2-hop relay in [30]	$\Theta\left(\frac{n}{k}\right)$
2-hop relay in [30]	$\Omega\left(\frac{n}{\sqrt{n\log k}}\right)$
multi-hop relay in [30]	$\Omega\left(\frac{1}{\log n}\right)$

TABLE II
COMPARISON ON THROUGHPUT BETWEEN OUR MMM SCHEME AND SOME APPROACHES PUBLISHED BY OTHER PAPERS.

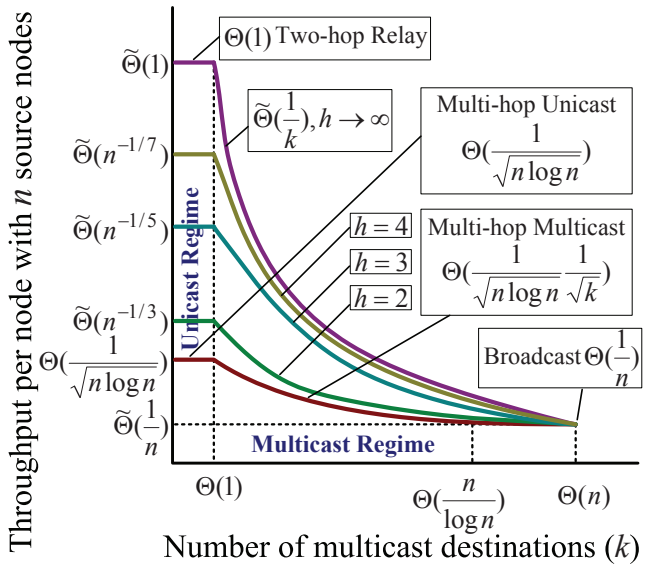


Fig. 5. We compare the known throughput results in static and mobile networks with that of our MMM strategy when $h=2,3,4$. It shows MMM strategy can achieve a higher throughput than that of non-cooperative schemes, and can also achieve the information-theoretic upper bound up to a logarithmic term when $h\to\infty$.

VII. CONCLUSION

In this paper, we develop a class of hierarchical cooperative schemes achieving an aggregate throughput of $\Omega\left((\frac{n}{k})^{1-\epsilon}\right)$ for any $\epsilon>0$, which is arbitrarily close to the upper bound. Our proposed schemes rely on MIMO transmissions, and consist of three steps. To maximize the aggregate throughput, in step 1 and step 3, we use multi-layer solutions to communicate within the clusters. We analyze the delay and energy consumption in each strategy. We find that converge-based multi-hop scheme performs better on both throughput and delay. Moreover, our CMMM strategy achieves the delay-throughput tradeoff identical to that of non-cooperative schemes when $h\to\infty$. While for certain k and h, the tradeoff ratio can be even smaller than that of unicast.

There are still many aspects for us to investigate in the future. For example, it remains an interesting problem to study multicast throughput with cooperative MIMO scheme in extended network, where the network area scales linearly with the number of nodes n on it.

APPENDIX A: PROOF OF LEMMA 4.1

Proof: Consider a specific node v_i. To prove the distance between v_i and all other nodes is larger than $\frac{1}{n^{1+\delta}}$, it is equivalent to prove that there are no other nodes inside a circle of area $\frac{\pi}{n^{2+2\delta}}$ around v_i. The probability of such an event is $\left(1-\frac{\pi}{n^{2+2\delta}}\right)^{n-1}$. The minimum distance between any two nodes in the network is larger than $\frac{1}{n^{1+\delta}}$ only if this condition is satisfied for all nodes in the network. Thus, by union bound we have

$$P\Big[d_{ij}\le\frac{1}{n^{1+\delta}},\text{ for all } i,j \text{ and } i\ne j\Big]\le n\Big(1-(1-\frac{\pi}{n^{2+2\delta}})^{n-1}\Big)$$

which diminishes to zero when n tends to infinity. ■

APPENDIX B: PROOF OF LEMMA 4.2

Proof: The number of nodes in a cluster at layer i is a sum of i.i.d. Bernoulli random variables X_j, such that $P[X_j=1]=1/n_{c_i}$. Using Chernoff bounds

$$P\Big[\sum_{j=1}^{n_i}X_j\ge(1+\delta)\frac{n_i}{n_{c_i}}\Big]<e^{-f(\delta)\frac{n_i}{n_{c_i}}}$$

where $f(\delta)=(1+\delta)\log(1+\delta)-\delta$, and

$$P\Big[\sum_{j=1}^{n_i}X_j\le(1-\delta)\frac{n_i}{n_{c_i}}\Big]<e^{-\frac{1}{2}\delta^2\frac{n_i}{n_{c_i}}}$$

When $n_i=\Omega(n_{c_i}\log n_{c_i})$

$$P\Big[\Big|\sum_{j=1}^{n_i}X_j-\frac{n_i}{n_{c_i}}\Big|\ge\delta\frac{n_i}{n_{c_i}}\Big]<e^{-\frac{n_i}{n_{c_i}}\theta}\to 0$$

when $n\to+\infty$. Here $\theta>0$ is a constant depend only on δ. Thus $n_{i-1}=\sum_{j=1}^{n_i}X_j=\Theta(\frac{n_i}{n_{c_i}})$ *whp*. ■

APPENDIX C: PROOF OF LEMMA 4.3

Proof: We divide the network into groups, each of which contains nine sub-squares. The nine squares in each group are numbered from 1 to 9 in the same way. We further divide time into sequences of successive slots, denoted by t ($t=0,1,2,3,\ldots$). During a particular slot t, one node in sub-squares that are numbered $(t \mod 9)+1$ are allowed to transmit packets.

Consider a slot when a node inside sub-square s_i is allowed to transmit to another node inside s_i. Then, those nodes that may interfere with the current transmission are located along the perimeters of concentric sub-squares centered at s_i. They can be grouped based on their distance to s_i such that the j-th group contains $8j$ interfering nodes or less (near the boundary of the network) and the shortest distance from the receiver

in s_i is $(3j-1)\sqrt{A}$, where A is the area of the sub-square. Assume all nodes use the same transmission power $P(n,k)$. Thus, with the power propagation model in (1), the cumulative interference at sub-square s_i, denoted by I_{s_i}, can be bounded by

$$\begin{aligned} I_{s_i} &\leq \sum_{j=1}^{n/M} 8j \times \frac{GP(n,k)}{[(3j-1)\sqrt{A}]^\alpha} \\ &\leq \frac{8GP(n,k)}{A^{\frac{\alpha}{2}}}\Big[1+\sum_{j=2}^{n/M}(3j-1)^{1-\alpha}\Big] \\ &< \frac{8GP(n,k)}{A^{\frac{\alpha}{2}}}\Big[1+\int_{j=0}^{\infty}(3j+2)^{(1-\alpha)}dj\Big] \\ &< \frac{8GP(n,k)}{A^{\frac{\alpha}{2}}}\Big[1+\frac{1}{3(\alpha-2)}\Big] \\ &= \frac{8GP(n,k)}{A^{\frac{\alpha}{2}}}\cdot\frac{3\alpha-5}{3\alpha-6} \end{aligned} \tag{40}$$

If we choose the transmission power $P(n,k)=\Theta(A^{\frac{\alpha}{2}})$, then interfering power will be upper-bounded by a constant independent of n. Besides, since the maximum distance for a transmitter to a receiver is $\sqrt{2A}$, the reception power can be lower-bounded by

$$R_{s_i} \geq \frac{GP(n,k)}{(\sqrt{2A})^\alpha} \tag{41}$$

As a result, the *SINR* for the transmission in s_i, denoted by SINR_{s_i}, is

$$\begin{aligned} \mathit{SINR}_{s_i} &= \frac{R_{s_i}}{N_0+I_{s_i}} \\ &\geq \frac{\frac{GP(n,k)}{(\sqrt{2A})^\alpha}}{N_0+\frac{8GP(n,k)}{A^{\frac{\alpha}{2}}}\cdot\frac{3\alpha-5}{3\alpha-6}} \end{aligned} \tag{42}$$

Note that $P(n,k)=\Theta(A^{\frac{\alpha}{2}})$, the *SINR* is a constant irrespective to n and k. Thus, according to the Shannon's channel capacity formula, i.e., $R(n,k)=W\log(1+\mathit{SINR})$ where $R(n,k)$ is the feasible rate, and W is the channel bandwidth, a fixed transmission rate independent of n and k can be achieved. ∎

Appendix D: Proof of Lemma 4.5

Proof: Let X_j be a random variable:

$$X_j=\begin{cases}1, & \text{if cluster } j \text{ contains at least one destination node;}\\ 0, & \text{else.}\end{cases}$$

Then $k_{c_i}=\sum_{j=1}^{n_{c_i}}X_j$. Since the k_i destination nodes at layer i are uniformly and independently distributed in n_{c_i} clusters, the probability that a destination node is in cluster j is $1/n_{c_i}$. The probability that none of the k_i destination nodes is in cluster j is $\left(1-\frac{1}{n_{c_i}}\right)^{k_i}$. Thus,

$$E[X_j]=1-\left(1-\frac{1}{n_{c_i}}\right)^{k_i}$$

Since $\{X_j\}_1^{n_{c_i}}$ is a sequence of i.i.d. random variables, using the law of large numbers, we obtain *whp*:

$$\frac{k_{c_i}}{n_{c_i}}=\frac{1}{n_{c_i}}\sum_{j=1}^{n_{c_i}}X_j\to 1-\left(1-\frac{1}{n_{c_i}}\right)^{k_i} \text{ when } n_{c_i}\to\infty \tag{43}$$

Consequently, the number of clusters which contain at least one destination node is $k_{c_i}=n_{c_i}\Big(1-(1-\frac{1}{n_{c_i}})^{k_i}\Big)$. When $k_i=O(n_{c_i})$, $k_{c_i}=n_{c_i}\Big(1-(1-\frac{1}{n_{c_i}})^{k_i}\Big)=\Theta(k_i)$ *whp*; when $k_i=\Omega(n_{c_i})$, $k_{c_i}=n_{c_i}\Big(1-(1-\frac{1}{n_{c_i}})^{k_i}\Big)=\Theta(n_{c_i})$ *whp*. ∎

Appendix E: Proof of Lemma 4.6

Proof: There are m_h sessions (frames) at layer m_h, or equivalently there are at most m_h sets of destination nodes distributed in the network. When consider a specific session (frame), let X_j be the number of destination nodes in the jth cluster at layer i. Obviously $E[X_j]=\frac{k_i}{n_{c_i}}$.

(a) If $k_i=\Omega(n_{c_i}\log n_{c_i})$, using Chernoff bound we obtain

$$P\Big[\Big|X_j-\frac{k_i}{n_{c_i}}\Big|\geq\delta\frac{k_i}{n_{c_i}}\Big]<e^{-\frac{k_i}{n_{c_i}}\theta}$$

At layer i, the number of destination sets is at most $m_h=O((n_{c_i})^{p_2})$. Thus

$$P\Big[\Big|X_j-\frac{k_i}{n_{c_i}}\Big|\leq\delta\frac{k_i}{n_{c_i}}\text{ for all sets}\Big]>\left(1-e^{-\frac{k_i}{n_{c_i}}\theta}\right)^{(n_{c_i})^{p_2}}\to 1$$

Therefore, $k_{i-1}=X_j=\Theta(\frac{k_i}{n_{c_i}})$ *whp*.

(b) If $k_i=O(n_{c_i}\log n_{c_i})$, Chernoff's inequality implies for all $s>0$

$$P[X_j>(p_2+3)\log n_{c_i}]\leq e^{-(p_2+3)s\log n_{c_i}}E[e^{sX}] \tag{44}$$

with $E[e^{sX}]=\exp((e^s-1)k_i/n_{c_i})$. From $k_i=O(n_{c_i}\log n_{c_i})$, the following inequality holds

$$(e-1)k_i/n_{c_i}-(p_2+3)\log n_{c_i}\leq(e-p_2-4)\log n_{c_i}$$

Let $s=1$ and we can get from (44)

$$P[X_j>(p_2+3)\log n_{c_i}]\leq n_{c_i}^{e-p_2-4}$$

Considering there are n_{c_i} clusters and at most m_h destination sets, we get

$$\begin{aligned} &P[X_j\leq(p_2+3)\log n_{c_i}\text{, for all } j \text{ and all sets}] \\ \geq\ &\left(1-n_{c_i}^{e-p_2-4}\right)^{n_{c_i}m_h} \\ \geq\ &\left(1-n_{c_1}^{e-p_2-4}\right)^{n_{c_i}^{p_2+1}}\to 1 \end{aligned}$$

Therefore, $k_{i_1}=X_j=O(\log n_{c_i})$ *whp*. ∎

Appendix F: Proof of Lemma 4.7

Proof: According to Lemma 4.5, when $k_i=o(n_{c_i})$, the k_i destination nodes are distributed into at least $\Theta(k_i)$ clusters *whp*.

(a) We use X_j to denote that nodes from jth destination set exist in a cluster at layer i. Then from Lemma 4.5

$$\frac{p_3k_i}{n_{c_i}}\leq P[X_j]\leq\frac{k_i}{n_{c_i}}$$

where p_3 is a constant. Let $X = \sum_{j=1}^{l_i} X_j$, we can conclude that $E[X] = \frac{p_4 l_i k_i}{n_{c_i}}$, where $p_3 \leq p_4 \leq 1$. Using Chernoff bounds we obtain for any $0 < \delta < 1$,

$$P[|X - E[X]| > \delta E[X]] < e^{-\theta E[X]}$$

where $\theta > 0$. Thus

$$P\Big[\Big|X - \frac{p_4 l_i k_i}{n_{c_i}}\Big| > \delta \frac{p_4 l_i k_i}{n_{c_i}}\Big] < e^{-\theta \frac{p_4 l_i k_i}{n_{c_i}}}$$

Since $l_i = \Omega\Big(\frac{n_{c_i}}{k_i} \log \frac{n_{c_i}}{k_i}\Big)$ and $k_i = o(n_{c_i})$, the right-hand-side of the above equation tends to 0. Therefore, $l_{i-1} = \Theta\Big(\frac{l_i k_i}{n_{c_i}}\Big)$ *whp*.

(b) Let X_j denote the number of destination sets in the jth cluster at layer i. Using Chernoff's inequality, for all $s > 0$

$$P\Big[X_j > (2p_4+1) \log \frac{n_{c_i}}{k_i}\Big] \leq e^{-(2p_4+1)s \log \frac{n_{c_i}}{k_i}} E[e^{sX}] \quad (45)$$

Let $s = 1$ we can get $E[e^{sX}] = \exp((e-1)p_4 l_i k_i / n_{c_i})$ and

$$P\Big[X_j > (2p_4+1) \log \frac{n_{c_i}}{k_i}\Big] \leq \Big(\frac{n_{c_i}}{k_i}\Big)^{p_4(e-3)-1}$$

Considering all n_{c_i} clusters, we can get

$$\begin{aligned} & P\Big[X_j \leq (2p_4+1) \log \frac{n_{c_i}}{k_i}, \text{ for all } j\Big] \\ \geq \quad & \Big(1 - \Big(\frac{n_{c_i}}{k_i}\Big)^{p_4(e-3)-1}\Big)^{n_{c_i}} \to 1 \end{aligned}$$

Therefore, $l_{i-1} = O\Big(\log \frac{n_{c_i}}{k_i}\Big)$ *whp*. ■

REFERENCES

[1] A. Özgür, O. Lévêque and D. Tse, "Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3549-3572, Oct. 2007.

[2] A. Özgür and O. Lévêque, "Throughput-delay trade-off for hierarchical cooperation in ad hoc wireless networks," in *Proc. Int. Conf. Telecom.*, Jun. 2008.

[3] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388-404, Mar. 2000.

[4] M. Franceschetti, O. Dousse, D. Tse and P. Thiran, "Closing the gap in the capacity of wireless networks via percolation theory," *IEEE Trans. Inf. Theory*, vol. 53, no. 3, pp. 1009-1018, Mar. 2007.

[5] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. on Netw.*, vol. 10, no. 4, pp. 477-486, Aug. 2002.

[6] S. Aeron and V. Saligrama, "Wireless ad hoc networks: strategies and scaling laws for the fixed snr regime," *IEEE Trans. Inf. Theory*, vol. 53, no. 6, pp. 2044-2059, Jun. 2007.

[7] J. Ghaderi, L. Xie and X. Shen, "Throughput optimization for hierarchical cooperation in ad hoc networks," in *Proc. ICC*, May 2008.

[8] S. Vakil and B. Liang, "Effect of joint cooperation and multi-hopping on the capacity of wireless networks," in *Proc. IEEE SECON*, Jun. 2008.

[9] U. Niesen, P. Gupta and D. Shah, "On capacity scaling in arbitrary wireless networks," accepted for publication in *IEEE Trans. Inf. Theory*, March 2009. Available online at http://arxiv.org/abs/0711.2745.

[10] M. J. Neely, and E. Modiano, "Capaicty and delay tradeoffs for ad hoc mobile networks," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 1917-1937, Jun. 2005.

[11] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah,"Throughput-delay trade-off in wireless networks," in *Proc. IEEE INFOCOM*, Mar. 2004.

[12] X. Lin and N. B. Shroff, "The fundamental capacity-delay tradeoff in large mobile wireless networks", Technical Report, 2004. Available at http://cobweb.ecn.purdue.edu/~linx/papers.html

[13] A. Agarwal and P. Kumar, "Capacity bounds for ad hoc hybrid wireless networks," *ACM SIGCOMM Computer Commun. Rev.*, vol. 34, no. 3, pp. 71-81, Jul. 2004.

[14] U. Kozat and L. Tassiulas, "Throughput capacity of random ad hoc networks with infrastructure support," in *Proc. ACM MobiCom*, Jun. 2003.

[15] B. Liu, Z. Liu and D. Towsley, "On the Capacity of Hybrid Wireless Networks", in *IEEE Infocom*, 2003.

[16] B. Liu, P. Thiran and D. Towsley, "Capacity of a wireless ad hoc network with infrastructure," in *Proc. ACM MobiHoc*, Sept. 2007.

[17] P. Li, C. Zhang and Y. Fang, "Capacity and delay of hybrid wireless broadband access networks", in *IEEE J. Sel. Areas Commun.*, vol. 27, No. 2, pp. 117-125, Feb. 2009.

[18] C. Zhang, Y. Fang, X. Zhu, "Throughput-Delay Tradeoffs in Large-Scale MANETs with Network Coding", in *Proc. IEEE Infocom 2009*, Rio de Janeiro, Brazil, Apri. 2009.

[19] L. Ying, S. Yang and R. Srikant. "Optimal Delay-Throughput Trade-Offs in Mobile Ad Hoc Networks", in *IEEE Trans. Inform. Theory*, Vol. 54, No. 9, pp. 4119-4143, Sept. 2008.

[20] S. Toumpis, "Asymptotic capacity bounds for wireless networks with non-uniform traffic patterns, " *IEEE Trans. Inf. Theory*, vol. 7, no. 6, pp. 2231-2242, Jun. 2008.

[21] A. Keshavarz-Haddad, V. Ribeiro, and R. Riedi, "Broadcast capacity in multihop wireless networks," in *Proc. ACM MobiCom*, Sept. 2006.

[22] Z. Wang , H. R. Sadjadpour and J. J. Garcia-Luna-Aceves, "A unifying perspective on the capacity of wireless ad hoc networks," in *Proc. IEEE INFOCOM*, Apr. 2008.

[23] X. Li, "Multicast Capacity of Wireless Ad Hoc Networks", in IEEE/ACM Trans. Networking, Jan., 2008.

[24] B. Liu, D. Towsley and A. Swami, "Data Gathering Capacity of Large Scale Multihop Wireless Networks", in *IEEE MASS*, 2008.

[25] S. Li, Y. Liu, X.Li, "Capacity of Large Scale Wireless Networks Under Gaussian Channel Model", in *ACM MobiCom*, 2008

[26] X. Li, S. Tang and O. Frieder, "Multicast capacity for large scale wireless ad hoc networks," in *Proc. ACM MobiCom*, Sept. 2007.

[27] A. Keshavarz-Haddad and R. Riedi, "Multicast capacity of large homogeneous multihop wireless networks," in *Proc. WiOPT*, Apr. 2008.

[28] P. Jacquet and G. Rodolakis, "Multicast scaling properties in massively dense ad hoc networks," in *Proc. ICPADS*, July 2005.

[29] S. Shakkottai, X. Liu and R. Srikant, "The multicast capacity of large multihop wireless networks," in *Proc. ACM MobiHoc*, Sept. 2007.

[30] Z. Wang, S. Karande, H. R. Sadjadpour and J. J. Garcia-Luna-Aceves, "On the capacity improvement of multicast traffic with network coding," in *Proc. MILCOM*, Sept. 2008.

[31] U. Niesen, P. Gupta and D. Shah, "The multicast capacity region of large wireless networks", in *Proc. IEEE INFOCOM*, Apr. 2009.

[32] C. Hu, X. Wang and F. Wu, "MotionCast: on the capacity and delay tradeoffs", in *Proc. ACM MobiHoc*, May 2009.

[33] C. Hu, X. Wang, D. Nie and J. Zhao, "Multicast scaling laws with hierarchical cooperation", in *Proc. IEEE INFOCOM*, San Diego, US, Mar. 2010.