

# Multicast Scaling Laws with Hierarchical Cooperation

Chenhui Hu, Xinbing Wang, Ding Nie, Jun Zhao  
Dept. of Electronic Engineering  
Shanghai Jiao Tong University, China  
Email: {hch,xwang8,kirknie,knight4088}@sjtu.edu.cn

**Abstract**—A new class of scheduling policies for multicast traffic are proposed in this paper. By utilizing hierarchical cooperative MIMO transmission, our new policies can obtain an aggregate throughput of  $\Omega\left(\left(\frac{n}{k}\right)^{1-\epsilon}\right)$  for any  $\epsilon > 0$ . This achieves a gain of nearly  $\sqrt{\frac{n}{k}}$  compared with non-cooperative scheme in [19].

Between the two cooperative strategies in our paper, the converge-based one is superior to the other on delay, while the throughput and energy consumption performances are nearly the same. Moreover, to schedule the traffic in a converge multicast manner instead of the simple multicast, we can dramatically reduce the delay by a factor nearly  $\left(\frac{n}{k}\right)^{\frac{h}{2}}$ , where  $h > 1$  is the number of the hierarchical layers. Our optimal cooperative strategy achieves an approximate delay-throughput tradeoff  $D(n, k)/T(n, k) = \Theta(k)$  when  $h \rightarrow \infty$ . This tradeoff ratio is identical to that of non-cooperative scheme, while the throughput performance is greatly improved. Besides, for certain  $k$  and  $h$ , the tradeoff ratio is even better than that of unicast.

## 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  $\Theta\left(\frac{1}{\sqrt{n \log n}}\right)$  for random networks, and the per-node transport capacity for arbitrary networks scales as  $\Theta\left(\frac{1}{\sqrt{n}}\right)$ , 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  $\Theta\left(\frac{1}{\sqrt{n}}\right)$  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] – [15] that when the number of base stations grows linearly as that of the nodes (implying a huge investment), capacity will scale linearly.

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 with cooperation are investigated in [8] and [9], respectively.

Another line of research deals with more generalized traffic patterns. In [16], Toumpis develops asymptotic capacity bounds for non-uniform traffic networks. In [17], broadcast capacity is discussed. Then, a unified perspective on the capacity of networks subject to a general form of information dissemination is proposed in [18]. 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 multimedia services. Thus, it raises great interests to the research community and has been studied by different manners in [19] – [23]. Very lately, Niesen et al. [24] characterize the multicast capacity region in an extended network. And capacity-delay tradeoff for mobile multicast is inquired in [25].

In this paper, we focus on multicast scaling laws with hierarchical MIMO. The motivation is jointly considering the effect of traffic patterns and cooperative strategies on the asymptotic performance of networks. 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 use **multi-hop MIMO transmission** to schedule inter-cluster communication. For the communication between clusters, the multi-hop manner conducts MIMO transmissions for many hops, and each time a cluster only transmits to the neighboring cluster. After analyzing, we find multi-hop MIMO transmission has a good throughput performance and is more energy efficient 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 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  $\Theta\left(\left(\frac{n}{k}\right)^{\frac{h}{2}}\right)$ , 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.<sup>1</sup>**

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

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  $V = \{v_1, v_2, \dots, v_n\}$  uniformly and independently distributed in a unit square  $\Omega$ . 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  $U_i = \{u_{i,j} | 1 \leq j \leq k\}$  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.

**Converge Multicast Traffic:** We randomly and independently choose a set of  $k$  nodes  $U_i = \{u_{i,j} | 1 \leq j \leq k\}$  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 Fig. 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:

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

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

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

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. 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.

<sup>1</sup>We use Knuth’s notation in this paper. Also we use  $f(n) = \tilde{\Theta}(g(n))$  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.

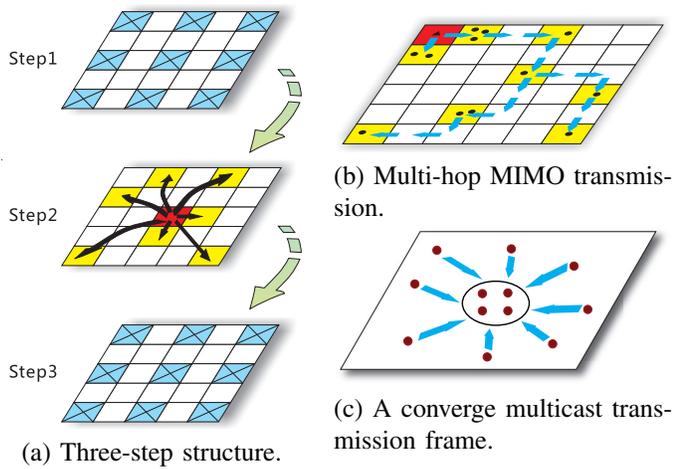


Fig. 1. Transmission strategy of hierarchical cooperation.

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.

## 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 the cluster containing at least one destination node *destination cluster*. Each multicast session is realized by a three-step structure (see Fig. 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. We use **Multi-hop MIMO transmission** to conduct the transmission, where each source cluster uses MIMO to transmit to a neighboring cluster 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 Fig. 1-(b) for illustration.
- 3) Step 3: After each destination cluster receives the MIMO transmissions, each node in the cluster holds an observation. 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” their data to a set of destination nodes.

### B. Two Strategies for cooperative multicast

Following the three-step multicast structure, there are two 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.
- 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.

### 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  $i$ th 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  $k_h = k$ . At layer  $i$ , 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.

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

#### 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

*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  $\frac{1}{n^{1+\delta}} \text{whp}^2$ , 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  $\text{whp}$  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 pairing from a reverse view, for each node  $d$ , there are on average  $k$  nodes  $s_1, s_2, \dots, s_k$  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 channel, we get

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

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

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

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

<sup>2</sup>In this paper,  $\text{whp}$  stands for with high probability, which means the probability tends to 1 as  $n \rightarrow \infty$ .

##### 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  $n_{i-1} = \frac{n_i}{n_{c_i}} \text{whp}$ , when **Assumption 1:**  $n_i = \Omega(n_{c_i} \log n_{c_i})$  is satisfied.

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<sup>3</sup>, we can apply the three steps recursively and build a  $h$ -layer solution.

1) *Solution to Multicast Problem:* We consider the  $i$ th 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,  $\frac{m_i}{n_i n_{i-1}}$  bits for each one. Considering  $n_{i-1}$  source nodes in each cluster, the traffic load are  $\Theta\left(\frac{m_i n_{i-1}}{n_i}\right)$  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.<sup>4</sup> 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 pairing at layer  $(i-1)$ . Given a traffic load of  $\Theta\left(\frac{m_i n_{i-1}}{n_i}\right)$  bits, this step can be completed in  $\Theta\left(\frac{m_i n_{i-1}^{1-a}}{n_i}\right)$  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) that is a good approximation of minimum Euclidean spanning tree, using algorithm provided in [19]. The constructed MT conducts MIMO transmissions between neighboring clusters, and has the following property.

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

<sup>3</sup>We view unicast as a special case of multicast problem.

<sup>4</sup>This assumption is also needed in other strategies. We will not repeat.

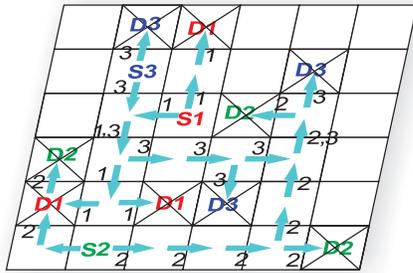


Fig. 2. An example of three MTs in multi-hop MIMO transmission.  $S_i$  denotes a source cluster and  $D_i$  is one of its destination clusters. The number on the arrow indicates which MT it serves. For each pair of neighboring clusters, the communication between them may involve data from different sources.

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  $O\left(\frac{m_i}{n_{i-1}} \sqrt{\frac{n_i k_{c_i}}{n_{i-1}}}\right)$ . 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 Fig. 2 for illustration. Hence, according to our protocol, at each time slot  $\Theta\left(\frac{n_{c_i}}{n_{i-1}}\right)$  clusters can transmit simultaneously. The total amount of time to accomplish all  $m_i$  sessions' MIMO transmissions is no more than  $O\left(m_i \sqrt{\frac{k_{c_i}}{n_i n_{i-1}}}\right)$ .

**Step 3. Cooperative Decoding:** Now that each MT has  $k_{c_i}$  destination clusters, after step 2, every cluster receives  $\Theta\left(\frac{m_i k_{c_i}}{n_i}\right)$  MIMO transmissions<sup>5</sup>. 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  $\tilde{\Theta}(n_{i-1}^{1-a} k_{i-1}^b)$  is achievable at layer  $(i-1)$  *whp*, where  $0 \leq a \leq 1, -1 \leq b \leq 0$ , and  $a + b \leq 0$ . Then  $MP(n_{i-1}, k_{i-1})$  can be solved within  $\tilde{\Theta}\left(\frac{Q n_{i-1}}{n_{i-1}^{1-a} k_{i-1}^b}\right)$  time slots. Note each cluster receives  $\Theta\left(\frac{m_i k_{c_i}}{n_i}\right)$  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  $\Theta\left(\frac{m_i k_{c_i}}{n_i}\right)$  rounds. Thus, step 3 costs  $\tilde{\Theta}\left(\frac{m_i n_{i-1}^{1-a} k_{c_i}}{n_i k_{i-1}^b}\right)$  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.

<sup>5</sup>This is valid under assumption 3 in Lemma 4.7, which we present later.

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

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

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

- (a) if  $k_i = \Omega(n_{c_i} \log n_{c_i})$ , then  $k_{i-1} = \Theta\left(\frac{k_i}{n_{c_i}}\right)$  *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.

**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:**  $l_i = \Omega\left(\frac{n_{c_i}}{k_i} \log \frac{n_{c_i}}{k_i}\right)$  is satisfied, then *whp*  $l_{i-1} = \Theta\left(\frac{l_i k_i}{n_{c_i}}\right)$ ;
- (b) when  $l_i = O\left(\frac{n_{c_i}}{k_i} \log \frac{n_{c_i}}{k_i}\right)$ , then *whp*  $l_{i-1} = O(\log \frac{n_{c_i}}{k_i})$ .

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

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

*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

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

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

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

To optimize the network division at layer  $i$ , we consider two cases:  $n_{c_i} = O(k_i)$  and  $n_{c_i} = \Omega(k_i)$ . 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  $k_{c_i} = \Theta(n_{c_i})$ ,  $k_{i-1} = \tilde{\Theta}\left(\frac{k_i}{n_{c_i}}\right)$ ;
- Case 2: When  $n_{c_i} = \Omega(k_i)$ , then  $k_{c_i} = \Theta(k_i)$ ,  $k_{i-1} = O(\log n_{c_i}) = \tilde{\Theta}(1)$ .

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

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

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

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

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

At the bottom layer, the aggregate multicast throughput is  $T(n_1, k_1) = 1$ . When dividing the network in the optimal way at each layer, we obtain  $n_i = n_{i-1}^{\frac{2i-1}{2i-3}}$  for  $2 \leq i \leq h-1$  and  $n_h = k_h n_{h-1}^{\frac{2h-1}{2h-3}}$ . Note  $n_h = n, k_h = k$ , it yields (3). ■

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

- 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  $2 \leq i \leq h$ . Thus, assumption 2 holds. As for assumption 1, obviously  $k_i = O(\log n_{c_{i+1}}) = O\left(\frac{n_i}{\log n_{c_i}}\right)$  for  $1 \leq i \leq h-1$ . Considering the top layer,  $k = O\left(\frac{n}{n_{c_h}}\right)$  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.
- Then we consider the number of destination nodes at each layer. By Lemma 4.6

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

This will change the number of sessions to

$$m_i = \Theta\left(nk \log^{(h-i-1)}\left(\frac{n}{k}\right)\right) \text{ for } 1 \leq i \leq h-1 \quad (8)$$

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

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

Thus, we have  $l_i = \Omega\left(\left(\frac{n}{k}\right)^{\frac{2}{2h-1}}\right) = \Omega\left(\frac{n_{c_i}}{\log n_{c_i}} \log^{\frac{n_{c_i}}{\log n_{c_i}}}\right)$  for  $2 \leq i \leq h-1$ , and assumption 3 holds for all layers.

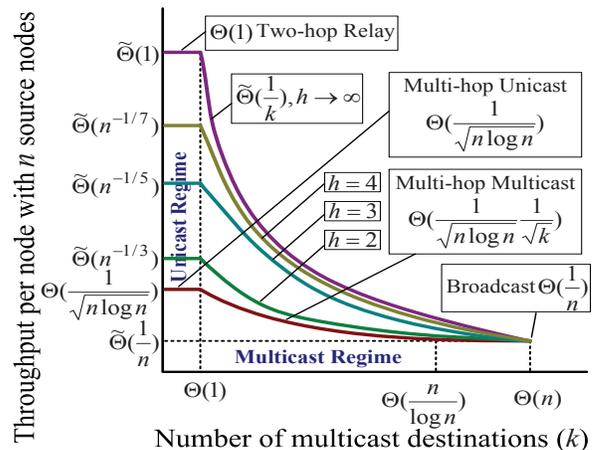


Fig. 3. We compare the known throughput results in static and mobile networks with that of our MMM strategy when  $h = 2, 3, 4$ . It shows MMM strategy can achieve a higher throughput than that of non-cooperative schemes, and can also achieve the information-theoretic upper bound up to a logarithmic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

4) *The Calculation of Throughput:* Followed by (5), the throughput is

$$T(n, k) = \Theta\left(\left(\frac{n}{k}\right)^{\frac{2h-2}{2h-1}} \log^{-(h-2)} \frac{n}{k}\right). \quad (9)$$

For simplicity, we omit the logarithmic order by using  $\tilde{\Theta}(\cdot)$  in the following theorem.

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

$$T(n, k) = \tilde{\Theta}\left(\left(\frac{n}{k}\right)^{\frac{2h-2}{2h-1}}\right). \quad (10)$$

*Remark 4.1:* Note the theorem also holds for broadcast case, which we will specify later. The throughput result is illustrated in Fig. 3, compared with the known results. For any  $\epsilon > 0$ , our cooperative scheme obtains a throughput of  $\Omega\left(\left(\frac{n}{k}\right)^{1-\epsilon}\right)$ , with  $h$  large enough. However, the delay performance of MMM strategy is poor (we will show it in Section V-A). Intuitively, this is because each node must transmit a large amount of bits a time to achieve this throughput. Considering the delay performance, we propose another strategy that dramatically reduces the delay.

### 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. At each round, only nodes in the chosen sub-clusters serve as source nodes ( $\frac{n_h n_{h-2}}{n_{h-1}}$  source nodes per round). 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<sup>6</sup> 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  $\Theta\left(n_{h-2} \sqrt{\frac{n_h k_{c_h}}{n_{h-1}}}\right)$  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  $O\left(\sqrt{\frac{n_2 k_{c_2}}{n_1}}\right)$  slots. Considering  $m_2$  frames, the time cost in step 1 is  $O\left(m_2 \sqrt{\frac{n_2 k_{c_2}}{n_1}}\right)$ .

*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  $m_1 = \Theta\left(\frac{m_2 k_{c_2}}{n_{c_2}}\right)$  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

<sup>6</sup>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.

in CMMM strategy is given by

$$n_i = \begin{cases} \left(\frac{n}{k}\right)^{\frac{2i-1}{2h-1}}, & i < h; \\ n, & i = h. \end{cases} \quad (11)$$

*Proof:* The proof is similar to that of Lemma 4.8. ■

3) *The Calculation of Throughput:* Before presenting the throughput result, the conditions in Section IV-B3 need justification as well. Assumptions 1 and 2 still hold, but assumption 3 is not always satisfied at layer 2, i.e. there exists a threshold<sup>7</sup>

$$k_{th} = \Theta\left(n^{\frac{1}{2h}} \log^{\frac{(h-3)(2h-1)}{2h}} n\right) = \tilde{\Theta}\left(n^{\frac{1}{2h}}\right). \quad (12)$$

When  $k = \Omega(k_{th})$ , assumption 3 holds for layer 2, otherwise it does not. Thus, our throughput result is

$$T(n, k) = \begin{cases} \Theta\left(n^{\frac{2h-3}{2h-1}} k^{\frac{2}{2h-1}} \log^{-1} \frac{n}{k}\right), & \text{when } k = O(k_{th}), \\ \Theta\left(\left(\frac{n}{k}\right)^{\frac{2h-2}{2h-1}} \log^{-(h-2)} \frac{n}{k}\right), & \text{when } k = \Omega(k_{th}). \end{cases} \quad (13)$$

Omitting the logarithmic order, we have the theorem below.

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

$$T(n, k) = \begin{cases} \tilde{\Theta}\left(n^{\frac{2h-3}{2h-1}} k^{\frac{2}{2h-1}}\right), & \text{when } k = O(n^{\frac{1}{2h}}), \\ \tilde{\Theta}\left(\left(\frac{n}{k}\right)^{\frac{2h-2}{2h-1}}\right), & \text{when } k = \Omega(n^{\frac{1}{2h}}). \end{cases} \quad (14)$$

## 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  $i$ th 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  $i$ th 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. 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 nodes, there are  $n_i B_i / n_{i-1}$  MTs at layer  $i$ . The number of hops on each MT is  $\Theta\left(\sqrt{\frac{n_i k_i}{n_{i-1}}}\right)$ . Using 9-TDMA scheme, we can complete step 2 in  $\Theta\left(B_i \sqrt{\frac{n_i k_i}{n_{i-1}}}\right)$  time slots.
- 3) For step 3, the traffic load are  $n_{i-1} k_i$  multicast sessions in every cluster, which take  $k_i D(n_{i-1}, k_{i-1})$  time slots.

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

$$D(n_i, k_i) = \Theta\left(B_i \sqrt{\frac{n_i k_i}{n_{i-1}}}\right) + k_i D(n_{i-1}, k_{i-1}) \quad (15)$$

where  $B_i = \left(\frac{n}{k}\right)^{\frac{(i-1)^2}{2i-1}}$  for  $1 \leq i \leq h$ . Also by the bottom layer transmission scheme,  $D(n_1, k_1) = n_1 = \left(\frac{n}{k}\right)^{\frac{2}{2h-1}}$ . Substituting

<sup>7</sup>We will discuss the influence of it in Section VI-C.

these into (15) and iterating the equation for  $i = 1, 2, \dots, h$ , we then obtain the final result

$$D(n, k) = \Theta\left(n^{\frac{h^2-2h+2}{2h-1}} k^{-\frac{h^2-4h+3}{2h-1}}\right) \quad (16)$$

*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 (10) with (16), the delay-throughput tradeoff is

$$D(n, k)/T(n, k) = \tilde{\Theta}\left(n^{\frac{h^2-4h+3}{2h-1}} k^{-\frac{h^2-6h+4}{2h-1}}\right) \quad (17)$$

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

$$\left(n_{h-1} + n_{\frac{2h-6}{2h-5}}\right) + n_{h-2} \sqrt{\frac{n_h k_{c_h}}{n_{h-1}}} + n_{h-2} k_h n_{\frac{1}{h-1}}^{\frac{2h-4}{2h-3}} k_{\frac{2h-4}{h-1}}^{\frac{2h-4}{2h-3}} \quad (18)$$

By Lemma 4.9, substituting all parameters by  $n$  and  $k$  in (18), we obtain the delay

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

which is simplified as

$$D(n, k) = \begin{cases} \tilde{\Theta}\left(\left(\frac{n}{k}\right)^{\frac{2h-3}{2h-1}}\right), & \text{when } k = O(n^{\frac{1}{2h}}), \\ \tilde{\Theta}\left(n^{\frac{2h-4}{2h-1}} k^{\frac{3}{2h-1}}\right), & \text{when } k = \Omega(n^{\frac{1}{2h}}). \end{cases} \quad (19)$$

Combining with (14), the delay-throughput tradeoff is

$$D(n, k)/T(n, k) = \begin{cases} \tilde{\Theta}(k^{-1}), & \text{when } k = O(n^{\frac{1}{2h}}), \\ \tilde{\Theta}\left(k\left(\frac{n}{k}\right)^{-\frac{2}{2h-1}}\right), & \text{when } k = \Omega(n^{\frac{1}{2h}}). \end{cases} \quad (20)$$

## B. Energy Consumption Analysis

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  $i$ th layer, we use  $E(n_i, k_i)$  to denote the energy consumption. Then

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

holds in the order sense. Considering the network division (11) and the factor  $k_i = \Omega(1)$  for all layer, we obtain

$$E(n_i, k_i) = n^{\frac{(i-h-1)\alpha+1}{2h-1}} k^{-\frac{2+2i\alpha-3\alpha}{4h-2}} + E(n_{i-1}, k_{i-1}) \quad (21)$$

Thus, the power spent at the  $(h-1)$ th layer is

$$E(n_{h-1}, k_{h-1}) = O\left(n^{\frac{-2\alpha+1}{2h-1}} k^{-\frac{2h\alpha-5\alpha+2}{4h-2}}\right) \quad (22)$$

For  $i = h$  in (21), substitute  $E(n_{h-1}, k_{h-1})$  with (22), we can obtain the final result

$$E(n, k) = O\left(n^{\frac{1-\alpha}{2h-1}} k^{-\frac{2h\alpha-3\alpha+2}{4h-2}}\right) \quad (23)$$

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 (23) 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.

## VI. DISCUSSION

### A. The Advantage of Cooperation

In our cooperative multicast scheme, we assume that the nodes nearby help each other on transmitting and receiving. By this assumption, we get a  $\Theta\left(\sqrt{\frac{n}{k}}\right)$  gain on the achievable throughput compared with [19]. 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  $2 \leq i \leq h$ . The reason that we cannot improve the throughput lies on the number of multicast sessions  $m_i$ . When  $n_{c_i} = O(k_i)$ , we obtain  $m_{i-1} = \Theta\left(\frac{m_i n_i}{n_{i-1}}\right)$ , 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. This results in no gain on the achievable throughput.

Besides, in MMM strategy, 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

As shown in (20), when  $h$  is large enough, the delay-throughput tradeoff  $D(n, k)/T(n, k)$  under multicast traffic is approximately  $D/T = \tilde{\Theta}(k)$ , which is identical to that of non-cooperative schemes. When  $k$  grows, the tradeoff ratio  $D/T$  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.

Besides, the tradeoff  $D/T$  of CMMM becomes worse as the number of layers  $h$  grows. Actually in CMMM, the delay is the time to complete a round. For each round, only  $n \times \frac{n_{h-2}}{n_{h-1}}$  nodes act as source nodes. When the number increases, the time to finish a round also increases. However, this does not affect the multicast throughput, since the number of bits transmitted in a round is linear with the time cost of the round. Hence,

the tradeoff 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.

#### D. The Advantage of Multi-hop MIMO Transmission

Using multihop method, we allow parallel MIMO transmissions in the network. Comparing with another **Direct MIMO Transmission** method, which a source cluster *broadcast* among the whole network and every destination cluster receive an observation, the MIMO transmission time from each source cluster is less in multihop method. 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. By reducing the transmission time, multi-hop scheme reduces the delay as well as improves the throughput comparing to direct one.

As for the energy consumption, multi-hop scheme 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.

### VII. CONCLUSION

In this paper, we proposed two kinds of hierarchical cooperative schemes achieving an aggregate throughput of  $\Omega((\frac{n}{k})^{1-\epsilon})$  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  $h \rightarrow \infty$ . While for certain  $k$  and  $h$ , the tradeoff ratio can be even smaller than that of unicast.

### VIII. ACKNOWLEDGE

This work is supported by National Fundamental research grant (2010CB731803, 2006CB303000), NSF China (No. 60702046, 60832005); China Ministry of Education (No. 20070248095); Qualcomm Research Grant; China International Science and Technology Cooperation Programm (No. 2008DFA11630); PUJIANG Talents (08PJ14067); Shanghai Innovation Key Project (08511500400); National High tech grant (2009AA01Z248).

### 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, "Capacity 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, P. Thiran and D. Towsley, "Capacity of a wireless ad hoc network with infrastructure," in *Proc. ACM MobiHoc*, Sept. 2007.
- [16] 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.
- [17] A. Keshavarz-Haddad, V. Ribeiro, and R. Riedi, "Broadcast capacity in multihop wireless networks," in *Proc. ACM MobiCom*, Sept. 2006.
- [18] 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.
- [19] X. Li, S. Tang and O. Frieder, "Multicast capacity for large scale wireless ad hoc networks," in *Proc. ACM MobiCom*, Sept. 2007.
- [20] A. Keshavarz-Haddad and R. Riedi, "Multicast capacity of large homogeneous multihop wireless networks," in *Proc. WiOPT*, Apr. 2008.
- [21] P. Jacquet and G. Rodolakis, "Multicast scaling properties in massively dense ad hoc networks," in *Proc. ICPADS*, July 2005.
- [22] S. Shakkottai, X. Liu and R. Srikant, "The multicast capacity of large multihop wireless networks," in *Proc. ACM MobiHoc*, Sept. 2007.
- [23] 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.
- [24] U. Niesen, P. Gupta and D. Shah, "The multicast capacity region of large wireless networks," in *Proc. IEEE INFOCOM*, Apr. 2009.
- [25] C. Hu, X. Wang and F. Wu, "MotionCast: on the capacity and delay tradeoffs", in *Proc. ACM MobiHoc*, May 2009.