Delay and Capacity Tradeoff Analysis for MotionCast

***Abstract*—In this paper, we define multicast for ad hoc network through nodes' mobility as *MotionCast*, and study the delay and capacity tradeoffs for it. Assuming nodes move according to an independently and identically distributed (i.i.d.) pattern and each desires to send packets to k distinctive destinations, we compare the delay and capacity in two transmission protocols: one uses 2-hop relay algorithm without redundancy, the other adopts the scheme of redundant packets transmissions to improve delay while at the expense of the capacity. In addition, we obtain the maximum capacity and the minimum delay under certain constraints. We find that the per-node delay and capacity for 2-hop algorithm *without redundancy* are $\Theta(1/k)$ and $\Theta(n \log k)$, respectively; and for 2-hop algorithm *with redundancy* they are $\Omega(1/k\sqrt{n \log k}$ and $\Theta(\sqrt{n \log k})$, respectively. The capacity of the 2-hop relay algorithm without redundancy is better than the multicast capacity of static networks developed in [22] as long as k is strictly less than n in an order sense; while when $k = \Theta(n)$, mobility does not increase capacity anymore. The ratio between delay and capacity satisfies $delay/rate \geq O(nk \log k)$ for these two protocols, which are both smaller than that of directly extending the fundamental tradeoff for unicast established in [20] to multicast, i.e., $delay/rate \geq O(nk^2)$. More importantly, we have proved that the fundamental delay-capacity tradeoff ratio for multicast is $delay/rate \geq O(n \log k)$, which would guide us to design better routing schemes for multicast.**

***Index Terms*—Multicast, Capacity, Delay, Scaling Law**

I. INTRODUCTION

A mobile ad hoc network (MANET) consists of a collection of wireless mobile nodes dynamically forming a temporary network without the support of any network infrastructure or centralized control. In these networks, nodes often operate not only as sources, but also as relays, forwarding packets for other mobile nodes. With the fast progress of computing and wireless networking technologies, there are increasing interests and use of MANETs. Examples where they may be employed are the establishment of connections among handheld devices or between vehicles.

Multicast is a fundamental service for supporting information communication and collaborative task completion among a group of users and enabling cluster-based system design in a distributed environment [2]. Different from in the wired networks, multicast in MANETs is faced with a more challenging environment. In particular, one needs to deal with node mobility and thus frequent and possible drastic topology changes [1]. Numerous protocols have then been proposed for multicast in MANETs. They include traditional tree- or mesh-based protocols [3], [4], [5], [6], stateless protocols [7], [8], flooding-based protocols [9], location-based protocols [10] and hybrid protocols [11]. Some of them have already pointed out that because links can be shared by several destinations, multicast is beneficial to improve performance comparing to multiple unicast.

However, the feasible performance gains, in terms of both throughput capacity and delay, that can be achieved by exploiting multicast, as well as the resulting scaling laws in a network with increasing number of nodes have not been investigated so far. In this paper, we bridge the theoretical analysis of fundamental scaling laws in multicast mobile ad hoc networks with the insights already gained through practical protocol development. By so doing, we provide a theoretical foundation to the design of intelligent communication schemes which exploit multicast, analytically showing the potential of such schemes in terms of capacity delay tradeoffs.

The theoretical analysis of scaling laws in wireless networks is initiated by the seminal work of Gupta and Kumar [23]. Several interesting studies have later emerged aiming at establishing the fundamental scaling laws for networks with multicast traffic. Li *et al*. [22] study the capacity of a static random wireless ad hoc network for multicast where each node sends packets to $k-1$ destinations. They show that the per-node multicast capacity is $\Theta(\frac{1}{\sqrt{n \log n}\sqrt{k}})$ when $k = O(\frac{n}{\log n})$; and is $\Theta(\frac{1}{n})$ when $k = \Omega(\frac{n}{\log n})$. Their results generalize previous capacity bounds on unicast [23] and broadcast [24]. Under a more general Guassian channel model, multicast capacity is investigated in [25] using percolation theory. Jacquet *et al*. [26] consider multicast capacity by accounting the ratio of the total number of hops for multicast and the average number of hops for unicast. Shakkottai *et al*. [27] propose a comb-based architecture for multicast routing which achieves the upper bound for capacity in an order sense.

In contrast to the above discussed static networks, Grossglauser and Tse [28] for the first time have shown that a constant unicast per-node capacity can be achieved in mobile ad hoc networks by exploiting the store-carry-forward communication paradigm, i.e., by allowing nodes to store the packets and physically carry them while moving around the network. Although this communication scheme incurs a tremendous average delay of $\Omega(n)$ [20], [29], it has laid the foundation of an entire new area of research, usually referred to as delay tolerant or disruption tolerant networks (DTNs), which has recently attracted a lot of attention. A typical DTN consists of a set of fixed or mobile nodes, and is characterized by intermittent connectivity and frequent network partitioning, such that node mobility is essential to ensure end-to-end communication. Many interesting applications of DTN have been already envisioned and experimented upon, such as vehicular networks based on WiFi [12], [13], [14], [15], networks based on human mobility [16], disaster-relief networks [18] and Internet access to remote villages [18].

The asymptotic capacity delay tradeoff in MANETs exploiting store-carry-forward schemes has attracted significant attentions and is studied by many authors under various mobility models. The mostly studied model is arguably the i.i.d. mobility model, where all nodes are reshuffled in a new time slot, due to its mathematical tractability. With this assumption, Neely and Modiano [20] present a strategy utilizing redundant packets transmissions along multiple paths to reduce delay at the cost of capacity. They establish the necessary tradeoff of $delay/capacity \geq O(n)$, and propose schemes to achieve $\Theta(1), \Theta(1/\sqrt{n})$ and $\Theta(1/(n \log n))$ per-node capacity, when the delay constraint is $\Theta(n), \Theta(\sqrt{n})$ and $\Theta(\log n)$, respectively. In [35], Toumpis and Goldsimth construct a better scheme that can achieve a per-node capacity of $\Theta(n^{(d-1)/2}/\log^{5/2} n)$ under fading channels when the delay is bounded by $O(n^d)$. Lin and Shroff [21] later study the fundamental capacity-delay tradeoff and identify the limiting factors of the existing scheduling schemes in MANETs. Recently, Ying *et al*. [34] propose joint coding-scheduling algorithms to improve capacity-delay tradeoffs while Garetto and Leonardi [19] show that under a restricted i.i.d. mobility model, it is possible to exploit node heterogeneity to achieve both constant capacity and constant delay.

To the best of our knowledge, this is the first work to study capacity and delay tadeoffs in MANETs with multicast traffics. Because a key feature of multicast in MANETs is that packets can be delivered via nodes' mobility, we refer it as MotionCast. Intuitively, delay and capacity tradeoffs still exist for MotionCast, but are more complicated than unicast scenarios. Since packets can be delivered through the mobility of relay nodes, a higher per-node multicast capacity than in static networks is expected. However, the scheduling design becomes more difficult because of the permanent change of the network topology as well as the fact that multiple destinations for a packet will imply a larger delay. Hence, some challenging issues raised naturally in this context are:

- What is the maximum per-node MotionCast capacity?
- What is the delay for maximal capacity achieving schemes and and what is the minimum possible delay?
- What is the delay and capacity tradeoff for MotionCast?

Answering these questions would provide helpful fundamental insights on the understanding and design of large scale multicast MANETs.

In this paper, we study the scaling laws in a cell partitioned MANET with multicast traffic. To start with, we propose a 2-hop relay algorithm without redundancy. This algorithm is a generalized version of the algorithm presented in [20], and corresponds to a decoupled queuing model. Because k destinations are associated with a source, the delay for a packet is defined as the total time needed to deliver it to all destinations. For a specific packet, we first divide nodes other than the source into relays and destinations (referred to as *non-cooperative mode*). In this case, the packet may be carried to the destinations either through the relays or via the source, but will not be passed from one destination to another. Once a packet is sent to a relay, the relay will be in charge of delivering it to all its destinations. Otherwise, if the source encounters a destination before a relay, it will take full responsibility of the rest multicast session. The MotionCast delay and capacity are calculated under this model.

Then, we loose the constraints of our initial model by permitting information dissemination among destinations (*co-operative mode*). In this scheme, we do not discriminate destinations against the remnant nodes except the source. We define the first node that a source meets as the "*designated relay*", which in fact may possible be an intended destination. Likewise, the designated relay should carry the packet from the source until it delivers this packet to all the destinations that have not received the message. Notice that only one relay is associated to a specific packet in the 2-hop relay algorithm, and therefore after a relay is designated other destinations will merely act as receivers for the packet and do not help transmit the packet to other nodes. Quite counter-intuitively, we find that there would be no gain in performance for the cooperative scheme comparing to the non-cooperative one from an order sense.

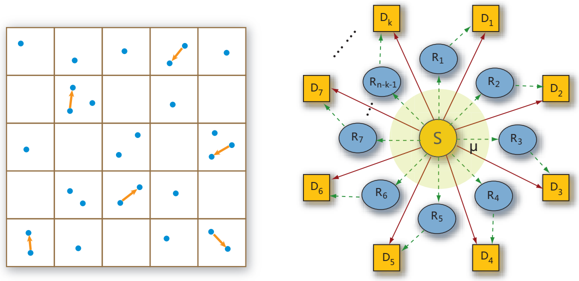
Next, we employ redundant packets transmissions to reduce the delay. In a 2-hop relay strategy with redundancy, a source sends a packet to multiple relays before all the destinations receive the packet, which increases the chance that a destination meets some of the relays at the expense of reduced capacity. If each timeslot only one transmission from a sender to a receiver is permitted in a cell, we show that the expected delay in the network is no less than $\Omega(\sqrt{n \log k})$. Besides, delay of $O(\sqrt{n \log k})$ is achievable with per-node capacity of $\Omega(1/k\sqrt{n \log k})$.

The main results of this paper are summarized as follows. For 2-hop relay algorithm without redundancy, the capacity for MotionCast is $\Theta(1/k)$ with average delay of $\Theta(n \log k)$. Notice that the per-node capacity is better than the results of static multicast scenario in [22] as long as k is strictly less than n in an order sense, i.e., $k = o(n)$. For 2-hop relay algorithm with redundancy, the capacity is $\Omega(1/k\sqrt{n \log k})$ with the delay scaling as $\Theta(\sqrt{n \log k})$. Thus, delay and capacity tradeoffs emerge between these two algorithms, i.e., we can utilize redundant packets transmissions to reduce delay but the capacity will also decrease. The tradeoff obtained by us is better than that of directly extending the tradeoff for unicast to multicast. We have also studied the fundamental delay-capacity tradeoff for MotionCast and shown that the fundamental tradeoff ratio is $delay/capacity \geq \Omega(n \log k)$.

The rest of the paper is organized as follows. In Section II, we describe the network model. In Section III, we introduce the 2-hop relay algorithm without redundancy. In Section IV, the 2-hop relay algorithm with redundancy is presented. In Section V, we discuss the results and figure out the fundamental tradeoff for multicast. Finally, we conclude in Section VI.

II. Network Model

Cell Partitioned Network Model: The system model is based on the cell partitioned network model exploited in [20] and [37]. Suppose the network is an unit square and there are n mobile nodes in it. Then, we divide it into c non-overlapping cells with equal size as depicted in Figure 1. We assume nodes can communicate with each other only when they



(a) Network model. (b) Traffic pattern

Fig. 1. A cell partitioned MANET model with c cells and n mobile nodes under multicast traffic pattern.

are within a same cell (to locate the nodes, please refer to [36] and the references therein), and to avoid interference different frequencies are employed among the neighboring cells.[1] Additionally, to bound the interference inside each cell, we assume that the number of the cells is on the same order as that of the nodes throughout this paper. Thus, node per cell density $d = n/c$ scales as $\Theta(1)$[2].

Mobility Model: Dividing time into constant duration slots, we adopt the following ideal i.i.d. mobility to model the sometimes drastic topology changes in MANETs and investigate their impact. The initial position of each node is equally likely to be any of the c cells independent of others. And at the beginning of each time slot, nodes randomly choose and move to a new cell i.i.d. over all cells in the network. Although the ideal i.i.d. mobility model may appear to be a oversimplification, it has been widely adopted in the literature because of its mathematical tractability, which could provide meaningful bounds on performance. Note that the i.i.d. model also characterizes the maximum degree of mobility. With the help of mobility, packets can be carried by the nodes until they reach the destinations.

Traffic Pattern: We first define the source-destination relationships before the transmissions start. In particular, we assume the number of users n is divisible by $k+1$ and number all the nodes from 1 to n. We uniformly and randomly divide the network into different groups with each of them having $k+1$ nodes. And assume packets from each node i in a specific group must be delivered to all the other nodes within the group. And nodes not belonging to the group can serve as relays. Hence each node i is a source node associated with k randomly and independently chosen destination nodes $D_1, D_2, \ldots, D_k$ over all the other nodes in the network. The relationships do not change as nodes move around. Then, the sources will communicate data to their k destinations respectively through a common wireless channel.

Definition of Capacity: First, we define stability of the network. Packets are assumed to arrive at node i with probability λ_i during each slot, i.e. as a Bernoulli process of arrival rate λ_i packets/slot. For the fixed λ_i rates, the network is *stable* if there exists a scheduling algorithm so that the queue in each node does not increase to infinity as time goes to infinity. Thus, the *per-node capacity* of the network is the maximum rate λ that the network can stably support. Note that sometimes the per-node capacity is called capacity for brief.

Definition of Delay: The delay for a packet is defined as the time it takes the packet to reach all its k destinations after it arrives at the source. The *total network delay* is the expectation of the average delay over all packets and all random network configurations in the long term.

Definition of Redundancy: At each timeslot, if more than one nodes are performing as relays for a packet, we say there is redundancy in the network. Furthermore, we say the corresponding scheduling scheme is with redundancy or redundant. Otherwise, it is without redundancy.

Definition of Cooperative: We adopt the term "cooperative" here to refer a destination can relay a packet from the source to other destinations. Otherwise, the destinations merely accept packets destined for them, but do not forward to others, which is called non-cooperative mode.

Notations: In our work, we adopt the following widely used order notations in a sense of probability. We say that an event occurs with high probability (w.h.p.), if its probability tends to 1 as n goes to infinity. Given two functions $f(n)$ and $g(n)$, we say that $f(n) = O(g(n))$ w.h.p., if there exist a constant c such that

$$\lim_{n\to\infty} P(f(n) \le cg(n)) = 1. \tag{1}$$

If the above sign of inequality is strict, we denote $f(n) = o(g(n))$. Moreover, we say that $f(n) = \Omega(g(n))$ w.h.p., if $g(n) = O(f(n))$ w.h.p.. If both $f(n) = \Omega(g(n))$ and $f(n) = O(g(n))$ w.h.p., then we say that $f(n) = \Theta(g(n))$ w.h.p..

III. Delay and Capacity in the 2-hop Relay Algorithm Without Redundancy

In this section, we propose 2-hop relay algorithms without redundancy and compute the achievable delay and capacity both under non-cooperative mode and cooperative mode. Then, we explore the maximum capacity and the minimum delay in these situations.

A. Under non-cooperative mode

In this subsection, we describe a 2-hop relay algorithm without redundancy. Usually, a source sends a packet to one of the relays, then the relay will distribute the packet to all its destinations. While as an initial step, we consider the non-cooperative mode, which means a destination can not be a relay.

2-hop Relay Algorithm Without Redundancy I: During a timeslot, for a cell with at least two nodes:

1) If there exists a source-destination pair within the cell, randomly select such a pair uniformly over all possible pairs in the cell. If the source has a new packet in the buffer intended for the destination, transmit. If all its destinations have received this packet,[3] then it will delete the packet from the buffer. Otherwise, stay idle.

[1] It is clear that only four frequencies are enough for the whole network.

[2] Theorem 3 and Theorem 4 will show this assumption does not vitiate our result and can lead us to design a more simple and practical scheduling algorithm with the purpose to achieve a good tradeoff between throughput and delay.

[3] We assume that nodes can aware this from the control information passed over a reserved bandwidth channel.

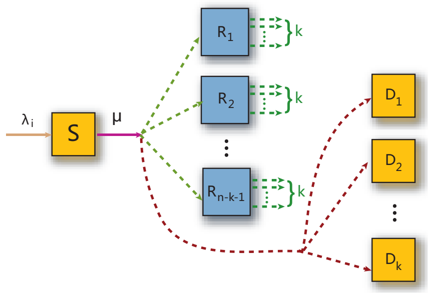


Fig. 2. A decoupled queuing model of the network as seen by the packets transmitted from a single source to multiple destinations.

2) If there is no such pair, randomly assign a node as sender and independently choose another node in the cell as receiver. With equal probability, choose from the following two options[4]:
   - Source-to-Relay Transmission: If the sender has a new packet one that has never been transmitted before, send the packet to the receiver and delete it from the buffer. Otherwise, stay idle.
   - Relay-to-Destination Transmission: If the sender has a new packet from other node destined for the receiver, transmit. If all the destinations who want to get this packet have received it, it will be dropped from the buffer in the sender. Otherwise, stay idle.

Intuitively, since there are no redundant transmissions and the cell partition with constant density scheme guarantees maximal spatial reuse, the algorithm could achieve maximal throughput. The only reason that a constant throughput cannot be achieved is that a single packet needs to be transmitted repetitively for about k times to different destinations, and therefore a $\Theta(1/k)$ throughput is feasible. Considering delay, it is intuitive for us to loosely model the network as a queueing system such that every source destination pair corresponds to a M/M/1 queue. The service time for a single packet, which follows exponential distribution, has an expectation of $\Theta(n)$, i.e., the average waiting time that two specific nodes meet. Then the total delay for a complete multicast session will roughly equals the maximum of k such i.i.d. random delays, and turns out to be $\Theta(n\log k)$. In below we formally derive the performance of the above algorithm.

The algorithm has an advanced decoupling feature between all n multicast sessions, as illustrated in Figure 2, where nodes are divided into destinations and relays for the packets from a single source, and the packets transmissions for other sources are modeled just as random ON/OFF service opportunities.

Let p represent the probability of finding at least two nodes in a particular cell, and q represent the probability of finding a source-destination pair within a cell. From Appendix I, we obtain that

$$p = 1 - \big(1-\frac{1}{c}\big)^n - \frac{n}{c}\big(1-\frac{1}{c}\big)^{n-1} \tag{2}$$

[4]Note that because of the traffic pattern we assume and the probabilities of source-destination and source-relay (or relay-destination) transmissions we calculate, source-destination transmission does not have priority over non-source-destination transmission, i.e., they happen independently.

$$q = 1 - \left[\frac{k+1}{c}\big(1-\frac{1}{c}\big)^k + \big(1-\frac{1}{c}\big)^{k+1}\right]^{\frac{n}{k+1}} \tag{3}$$

When n tends to infinity, it follows $p \to 1-(d+1)e^{-d}$ and $q \to 1-e^{-\frac{k}{k+1}d}(1+\frac{k}{c})^{\frac{n}{k+1}}$. Thus, if $k=o(n)$, $q\to 0$;[5] else if $k=\Theta(n)$, $q \to 1-(d+1)e^{-d}$. Intuitively, when k approaches the same order as n, the multicast will reduces to a broadcast and the events corresponding to p and q will gradually become identical. Then, we have the following theorem.

Theorem 1: Consider a cell-partitioned network (with n nodes and c cells) under the 2-hop relay algorithm without redundancy I, and assume that nodes change cells i.i.d. and uniformly over each cell every timeslot. If the exogenous input stream to node i which makes the network stable is a Bernoulli stream of rate $\lambda_i = O(\mu/k)$ and $k=o(n)$, then the average delay W_i for the traffic of node i satisfies

$$E\{W_i\} = O(n\log k) \tag{4}$$

where $\mu = \frac{p+q}{2d}$.

Proof: A decoupled view of the network as seen by a single source i is shown in Figure 2. Due to the i.i.d. mobility model, the source user can be represented as a Bernoulli/Bernoulli queue, where in every timeslot a new packet arrives with probability λ_i, and a service opportunity arises with some fixed probability μ when the packet is handed over a relay or transmitted to a destination. We first show that the expression $\mu = \frac{p+q}{2d}$ still holds.

The Bernoulli nature of the server process implies that the transmission probability μ is equal to the time average rate of transmission opportunities of source i.[6] Let r_1 represent the rate at which the source is scheduled to transmit directly to one of the destinations, and r_2 represent the rate at which it is scheduled to transmit to one of its relays. The same as μ, r_1 equals to the probability that the source is scheduled to transmit directly to the destination and r_2 equals to the probability that the source is scheduled to transmit to one of its relay users. Then, we have $\mu = r_1 + r_2$. Since the relay algorithm schedules transmissions into and out of the relay nodes with equal probability, hence r_2 is also equal to the rate at which the relay nodes are scheduled to transmit to the destinations. Every timeslot, the total rate of transmission opportunities over the network is thus $n(r_1+2r_2)$. Meanwhile, a transmission opportunity occurs in any given cell with probability p, hence,

$$cp = n(r_1 + 2r_2) \tag{5}$$

Recall that q is the probability that a given cell contains a source-destination pair. Since the algorithm schedules the single-hop source-to-destination transmissions whenever possible, the rate r_1 satisfies

$$cq = nr_1 \tag{6}$$

[5]because when $n\to\infty$ and $k=o(n)$, $q \to 1-e^{-\frac{k}{k+1}d}(1+\frac{k}{c})^{\frac{n}{k+1}} \to 1-e^{-d}e^{d}=0$

[6]A transmission opportunity arises when user is selected to transmit to another user, and corresponds to a service opportunity in the Bernoulli/Bernoulli queue. Such opportunities arise with probability μ every timeslot, independent of whether or not there is a packet waiting in the queue.

It follows from (8) and (6) that $r_1 = \frac{q}{d}$, $r_2 = \frac{p-q}{2d}$. The total rate of transmissions out of the source node is thus given by $\mu = r_1 + r_2 = \frac{p+q}{2d}$.

Next, we compute the average delay for the traffic of node i. There are two possible routings from a source to its destinations: one is the 2-hop path along "source-relay-destinations", the other is the single-hop path from source to destinations directly. As for the first routing, packet delay is composed of the waiting time at source and relay. In this case, since the source can be viewed as a Bernoulli/Bernoulli queue with input rate λ_i and service rate μ , it has an expected number of occupancy packets given by $\bar{L}_s = \frac{\rho(1-\lambda_i)}{1-\rho}$, where $\rho \triangleq \frac{\lambda_i}{\mu}$. From Little's theorem, the average waiting time in the source is $E\{W_s\} = \frac{\bar{L}_s}{\lambda_i} = \frac{1-\lambda_i}{\mu-\lambda_i}$. Besides, this queue is reversible, so the output process is also a Bernoulli stream of rate λ_i.

Notice that our traffic pattern has defined every disjoint $k+1$ nodes as a group and every node in this group is the source for the other k nodes. From a more delicate point of view, a packet delivered from a source to a relay contains not only necessary payload, but also redundant data in its head which tells the relay which k destinations this packet should be transmitted to, shown in Figure 3-(a). Based on this information, the relay can make k similar copies, each of which contains less redundant data in its head just indicating its own corresponding destination. And also since a node can act as a relay to transmit packets to other $n-k-1$ destinations, we model a relay as a node that has $n-k-1$ parallel subqueues (each of them buffers the packets intended for a certain destination), shown in Figure 3-(b). Next, we will compute the input rate and output rate of a subqueue.

A given packet from a source is transmitted to the first relay node with probability $p_i = \frac{r_2}{\mu(n-k-1)}$ and rate $\lambda_r = \lambda_i p_i$ (because with probability $\frac{r_2}{\mu}$ the packet is delivered to a relay, and each of the $n-k-1$ relay nodes are equally likely). Since there are k sources for each subqueue, every timeslot, a subqueue in this relay receives a packet with probability $1-(1-p_i)^k$ which can be expressed as $kp_i + o(kp_i)$. The latter one won't influence our results, so we omit it. Hence, the input rate of a subqueue is $k\lambda_r$. On the other hand, the subqueue in the relay node is scheduled for a potential packet transmission to a destination node with probability $\mu_r = \frac{r_2}{n-k-1}$ (because when it acts as a relay, it can transmit packets to $n-k-1$ destinations except the source of the given packet and itself with equal probability). Notice that packet arrivals and transmission opportunities in a subqueue of the relay node are mutually exclusive events. It follows that the discrete time Markov chain for queue occupancy in the relay node can be written as a simple birth-death chain which is identical to a continuous time M/M/1 queue with input rate $k\lambda_r$ and service rate μ_r. Each destination i $(1 \le i \le k)$ obtains the packet from the relay though such a queue, thus the waiting time for it is an exponential distributed variable with expectation of $E\{W^i_{rd}\} = 1/(\mu_r - k\lambda_r)$.

The resulting waiting time W_{rd} for multicast is determined by the maximum value among all the waiting times $W^1_{rd}, W^2_{rd}, ..., W^k_{rd}$ of these k destinations. Due to the interference constraint that a relay node can communicate with only one destination in one time slot, $\{W^k_{rd}\}$ are correlated over k. However, we can construct a set of dual random variables $\{W'^i_{rd}\}$ such that they are i.i.d.. They provide a slightly alternated view of the queueing system depicted in Fig. 3, with *multi-destination reception* enabled, i.e., if a source encounters more than one destinations, the packet will be transmitted to all of them. In the following we can show $E\{W^k_{rd}\} = \Theta(E\{W'^k_{rd}\})$.

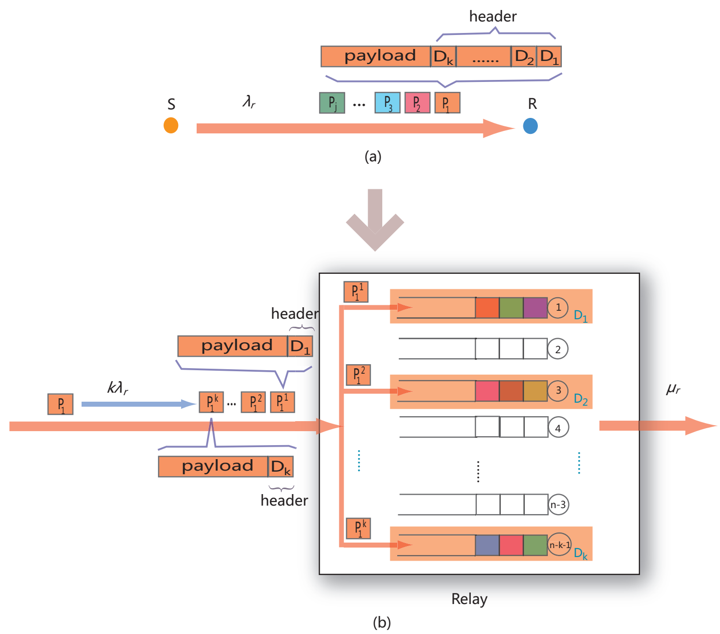


Fig. 3. A more delicate view of a relay-destinations transmission. (a) A subqueue in a relay receives packets from a source with rate λ_r. Here different packets are in different colors and each has a similar form as the extended red one. (b) Each of a relay can make a packet into k similar copies, and can be modeled as a node having $n-k-1$ parallel subqueues buffering packets intended for different destinations. In special, k subqueues associated with k destinations of the current source are depicted in red in the figure.

Conditioning on the event that the relay encounters one or more destinations, and denotes φ_1, $\varphi_{>1}$ as the probability that exactly one or more than one destinations are reached, respectively. It is clear that $\varphi_1 = \omega(\varphi_{>1})$ if $k = o(n)$ and $\varphi_1 = \Theta(\varphi_{>1})$ if $k = \Theta(n)$. Because of the nature of M/M/1 queue, the waiting time W^i_{rd} or W'^i_{rd} for each subqueue only depends on the input rate and the service rate. Note that the service rate for these two cases without or with multi-destination reception are only determined by φ_1 and $\varphi_1 + \varphi_{>1}$, respectively. Since $\varphi_1 = \omega(\varphi_{>1})$ if $k = o(n)$ and $\varphi_1 = \Theta(\varphi_{>1})$ if $k = \Theta(n)$, $\mu_r = \Theta(\mu'_r)$. Also note that the input rate for a subqueue in these two cases are the same $k\lambda_r$, it is clear $E\{W^k_{rd}\} = \Theta(E\{W'^k_{rd}\})$, which also indicates from the viewpoint of the order of n, the waiting time W_{rd} (The maximum value among all the waiting times $W^1_{rd}, W^2_{rd}, ..., W^k_{rd}$) for multicast without multi-destination reception achieves the same order with that of multi-destination reception that enables these waiting time $\{W'^k_{rd}\}$ are i.i.d. exponential variables. By Lemma 2 (see the proof in Appendix II), we obtain that $E\{W_{rd}\} = E\{W'_{rd}\} = \log k/(\mu_r - k\lambda_r)$. Thus, if the packet is delivered through the path "source-relay-destinations", the average delay is $E\{W_s\}+E\{W_{rd}\}$.

While if the packet is directly sent to the destinations by the source, it will wait at the source for a time W_s first, then the source distributes this packet to the remnant $k-1$ destinations. At this time, the source can be treated as a node having k parallel M/M/1 sub-buffers corresponding to its k destinations similarly. The source will copy this packet into $k-1$ similar duplicates and add them into respective sub-buffers associated with the remnant $k-1$ destinations. Also, at this time, the source can be treated as a continuous time M/M/1 queue. Since the probability that the source need send packets directly to destinations is $\frac{r_1}{\mu}$, the incoming data rate is thus $\frac{\lambda_i r_1}{\mu}$ for such a queue. Meanwhile, the service rate at each equals to the transmission rate $\frac{r_1}{k}$ between a source-destination pair. Hence, the expectation of the waiting time for each one of the $k-1$ destinations through such a queue is $1/(\frac{r_1}{k}-\frac{\lambda_i r_1}{\mu})$. And by Lemma 2 and the same method in the above calculation of the delay through 2-hop route, we have the expect waiting time for the packet to reach all $k-1$ remnant destinations is $E\{W_{sd}\}=\log(k-1)/(\frac{r_1}{k}-\frac{\lambda_i r_1}{\mu})$.

Finally, by weighting the delay occurs in both two routings, we achieve the total network delay is

$$\begin{aligned} E\{W_i\} &= \frac{r_1}{\mu}\big(E\{W_s\}+E\{W_{sd}\}\big)+\frac{r_2}{\mu}\big(E\{W_s\}+E\{W_{rd}\}\big) \\ &= \frac{r_2}{\mu}\left(\frac{1-\lambda_i}{\mu-\lambda_i}+\frac{\log k}{\frac{r_2}{n-k-1}-\frac{k\lambda_i r_2}{\mu(n-k-1)}}\right) \\ &\quad +\frac{r_1}{\mu}\left(\frac{1-\lambda_i}{\mu-\lambda_i}+\frac{\log(k-1)}{\frac{r_1}{k}-\frac{\lambda_i r_1}{\mu}}\right) \\ &= \frac{1-\lambda_i}{\mu-\lambda_i}+\frac{(n-k-1)\log k}{\mu-k\lambda_i}+\frac{k\log(k-1)}{\mu-k\lambda_i} \end{aligned} \quad (7)$$

To ensure the stability of the network, the incoming rate should be less than the service rate at any stage of the network. Thus,

$$\begin{cases} \mu-\lambda_i>0 \\ \frac{r_1}{k}-\frac{\lambda_i r_1}{\mu}>0 \\ \frac{r_2}{n-k-1}-\frac{k\lambda_i r_2}{\mu(n-k-1)}>0 \end{cases}$$

i.e., $\lambda_i<\mu/k$. Besides, the total network delay is in the order of $O(n\log k)$ for a fixed traffic loading value $\rho_r=\frac{k\lambda_i}{\mu}$ at each relay and source.

From the above discussion, we conclude the theorem. ■

B. Under cooperative mode

In the above subsection, we propose a 2-hop relay algorithm without redundancy obtaining per-node capacity $\Omega(1/k)$ with delay $O(n\log k)$. In this subsection we bring forward a more general algorithm which does not discriminate destinations and the nodes other than the source, i.e., under cooperative mode. This algorithm achieves the same performance as the first one. Since the second algorithm is more simple than the first one, we adopt this algorithm and refer it as *2-hop relay algorithm without redundancy* for brief in the rest of the paper. It is described as follows.

2-hop Relay Algorithm Without Redundancy II: For each cell with at least two nodes in a timeslot, a random sender and a random receiver are picked with uniform probability over all nodes in the cell. With equal probability, the sender is scheduled to operate in the two options below:

1) Source-to-Relay Transmission: If the sender has a new packet one that has never been transmitted before, send the packet to the receiver and delete it from the buffer. Otherwise, stay idle.
2) Relay-to-Destination Transmission: If the sender has packets received from other nodes which are destined for the receiver and have not been transmitted to the receiver yet, then choose the latest one, transmit. If all the destinations who want to get this packet have received it, it will be dropped from the buffer in the sender. Otherwise, stay idle.

The algorithm simply designates the first node a source meets as the relay, no matter if it is a destination. Thus according to the scheduling scheme, all the packets will be delivered along a 2-hop path "source-relay-destinations". Then, we summarize the next theorem.

Theorem 2: Consider the same assumptions for the network as Theorem 1, under the 2-hop relay algorithm without redundancy II. The resulting per-node capacity and the average delay are $\Omega(1/k)$ and $O(n\log k)$, respectively, for all $k\le n$.

Proof: Since, all the packets will be delivered along a 2-hop "source-relay-destinations" path, by using the same analytical method, we can know $r_1=0$ and $\mu=r_2$. Meanwhile, a transmission opportunity occurs in any given cell with probability p, hence,

$$cp=2n\mu \quad (8)$$

It follows from (8) that $\mu=r_2=\frac{p}{2d}$.

Thus following the same analytical steps as Theorem 1 when k is strictly less than n in an order sense, we can know that packet delay is composed of the waiting time at source and relay. And since the source can be viewed as a Bernoulli/Bernoulli queue with input rate λ_i and service rate μ, the average waiting time in the source is $E\{W_s\}=\frac{1-\lambda_i}{\mu-\lambda_i}$. Besides, this queue is reversible, so the output process is also a Bernoulli stream of rate λ_i.

A given packet from this output process is transmitted to the first relay node with probability $\frac{1}{n-1}$. Hence, every timeslot, this relay independently receives a packet with probability $\lambda_r=\frac{\lambda_i}{n-1}$. On the other hand, the relay node is scheduled for a potential packet transmission to a destination node with probability $\mu_r=\frac{\mu}{n-2}$ (because when it acts as a relay, it can transmit packets to $n-2$ destinations except the source of the given packet and itself with equal probability). Notice that packet arrivals and transmission opportunities are mutually exclusive events in the relay node.

When taking the 2-hop algorithm without redundancy II, the first node a source meets as the relay, no matter if it is a destination. The difference is that if the relay is a destination node, it need only relay the packet to the rest $k-1$ destinations; otherwise, it need relay the packet to all k destinations. Since we focus on the performance in an order sense, we omit this difference between these two cases and assume a relay is responsible for delivering a new packet to its corresponding k destinations for simplicity.

Thus at this time, when receiving a new packet from the source, the relay node will make it into k similar duplicates. Thus, a relay can be viewed as a M/M/1 queue with input rate $k\lambda_r$ and service rate μ_r. Hence the expectation of the waiting time of each destination is $1/(\frac{\mu}{n-2} - \frac{k\lambda_i}{n-1})$. And by lemma 2, we have the expect waiting time for the packet to reach all k destinations is $E\{W_{sd}\} = \log(k)/(\frac{\mu}{n-2} - \frac{k\lambda_i}{n-1})$.

Finally, we achieve the total network delay is

$$\begin{aligned} E\{W_i\} &= E\{W_s\} + E\{W_{sd}\} \\ &= \frac{1-\lambda_i}{\mu - \lambda_i} + \frac{\log k}{\frac{\mu}{n-2} - \frac{k\lambda_i}{(n-1)}} \end{aligned} \quad (9)$$

Looking upon the asymptotic behaviors of the network delay when $k, n \to \infty$, we have $\mu = r_2 \to \frac{1-(d+1)e^{-d}}{2d}$, To ensure the stability of the network, the incoming rate should be less than the service rate at any stage of the network. Thus,

$$\begin{cases} \mu - \lambda_i > 0 \\ \frac{\mu}{n-2} - \frac{k\lambda_i}{n-1} > 0 \end{cases}$$

i.e., $\lambda_i < \frac{(n-1)\mu}{(n-2)k} \to \mu/k \ (n \to \infty)$. Besides, the total network delay is governed by (9), which is on the order of $O(n \log k)$ for a fixed traffic loading value $\rho_r = \frac{\lambda_i(n-2)}{n-1}$ at each relay.

From the above discussion, we conclude the theorem. ■

C. Maximum capacity and minimum delay

Although we have constructed the achievable delay and capacity if no redundancy is used, open questions still leave for the maximum capacity and the minimum delay of this network. We address these problems here by presenting the following theorems.

Theorem 3: The multicast capacity of a cell partitioned network is $O(\frac{1}{dk})$ if only a pair of sender and receiver is active in each cell per timeslot. In particular, if $d = \Theta(1)$, the multicast capacity is $O(1/k)$.

Proof: We use hop argument to prove this result. Since for any interval [0,T], the less hops the source need send a packet to its k destinations, the more capacity it can achieve. So we assume a packet is delivered directly from a source to one of its destinations via the 1-hop route "Source-destination". Let $X_s(T)$ represent the total number of packets transferred over the network from sources to destinations via 1-hop route during the interval [0,T]. Fix $\epsilon > 0$. For network stability, there must be arbitrarily large values T such that the sum output rate is within ϵ of the total input rate

$$\frac{X_s(T)}{T} \geq nk\lambda - \epsilon \quad (10)$$

If this were not the case, the total number of packets in the network would grow to infinity and hence the network would be unstable. Since every transmission just needs one hop, the total number of packet transmissions in the network during the first T slots is also $X_s(T)$. This value must be less than or equal to the total number of transmission opportunities $Y(T)$, and hence,

$$X_s(T) \leq Y(T) \quad (11)$$

where $Y(T)$ represents the total number of cells containing at least two users in a particular timeslot, summed over all timeslots $1, 2, \ldots, T$. By the law of large numbers, it is clear that $\frac{1}{T}Y(T) \to cp$ as $T \to \infty$, where p is the steady-state probability that there are two or more users within a particular cell, and is given by (2).

From (10) and (11), it follows that

$$\lambda \leq \frac{\frac{Y(T)}{T} + \epsilon}{kn} = \frac{p}{kd} + \frac{\epsilon}{kn} \quad (12)$$

Notice that k=$O(n)$, thus we have λ=$O(\frac{1}{dk})$. And if d=$\Theta(1)$, λ=$O(1/k)$. ■

Theorem 4: Algorithms permitting at most one transmission in a cell at each timeslot, which do not use redundancy cannot achieve an average delay of $O(\frac{n \log k}{d})$. In particular, if $d = \Theta(1)$, $E\{W'_{min}\} = \Theta(n \log k)$.

Proof: The minimum delay of any packet is calculated by considering the situation where the network is empty and node 1 sends a single packet to k destinations.[7] Since relaying the packet can not help reduce delay, it can be treated as having no relay at all. Denote p' and W'_{min} as the chance that node 1 *meets* (i.e., two nodes move into a same cell) one of the destinations in a timeslot and the minimum amount of time it takes the source to meet all the destinations, respectively. We have that $p' = 1/c$. Since $W'_{min} = i$ means that at the $(i-1)$th timeslot the source has met $k-1$ destinations and at the ith timeslot it meets the last one, thus the probability $W'_{min} = i$ can be written as

$$\begin{aligned} P\{W'_{min} = i\} \;=\; & kp'\Bigg[(1-p')^{i-1} - \binom{k-1}{1}(1-2p')^{i-1} \\ & + \binom{k-2}{2}(1-3p')^{i-1} - \cdots\Bigg] \end{aligned} \quad (13)$$

Therein the factor kp' denotes that the last destination D'_k meets by the source can be any one of the k destinations. The first term in the latter factor infers that D'_k has not been met in the former $i-1$ timeslots. Because the first term also includes the probability that the source has not met D'_k and any one of the other nodes from D'_1 to D'_{k-1}, this value should be subtracted from the first term, so the second term attached and similarly we have the following terms. Hence, the expectation

[7] By saying the network is empty we mean only node 1 has packets to send and other nodes have no packet and stay idle.

of $E\{W'_{min}\}$ is

$$
\begin{aligned}
& E\{W'_{min}\} \\
&= kp'\sum_{i=1}^{+\infty} i\Big[(1-p')^{i-1} - \binom{k-1}{1}(1-2p')^{i-1} \\
&\quad + \binom{k-2}{2}(1-3p')^{i-1} - \cdots\Big] \\
&= kp'\Big[\sum_{i=1}^{+\infty} i(1-p')^{i-1} \\
&\quad - \binom{k-1}{1}\sum_{i=1}^{+\infty} i(1-2p')^{i-1} \\
&\quad + \binom{k-1}{2}\sum_{i=1}^{+\infty} i(1-3p')^{i-1} - \cdots\Big] \\
&= kp'\Big[\frac{1}{p'^2} - \binom{k-1}{1}\frac{1}{(2p')^2} + \binom{k-1}{2}\frac{1}{(3p')^2} - \cdots\Big] \\
&= \frac{k}{p'}\Big[1 - \frac{1}{2^2}\binom{k-1}{1} + \frac{1}{3^2}\binom{k-1}{2} - \cdots\Big] = \frac{\log k}{p'} \qquad (14)
\end{aligned}
$$

wherein Lemma 1 and the following identical relation for any $|x| < 1$ are exploited

$$\sum_{i=1}^{+\infty} ix^{i-1} = (\sum_{i=1}^{+\infty} x^i)' = \frac{1}{(1-x)^2}.$$

Finally, notice that $1/p' = c = \frac{n}{d}$, we obtain that $E\{W'_{min}\} = \Theta(\frac{n\log k}{d})$. In particular, if $d = \Theta(1)$, $E\{W'_{min}\} = \Theta(n\log k)$. Since at any timeslot, if there are more than one destinations in a same cell as the source, only one destination could be selected as the receiver, the actual delay $E\{W_{min}\}$ for the packet to be delivered to all the destinations will be larger or equal than $E\{W'_{min}\}$, which points out the theorem. ∎

Combining these results with the delay and capacity achieved by the 2-hop relay algorithm without redundancy, we find the exact order of the delay and capacity are $\Theta(1/k)$ and $\Theta(n\log k)$, respectively.

IV. Delay and Capacity in the 2-hop Relay Algorithm With Redundancy

In this section, we adopt redundancy to improve delay. The idea originates from a basic notion that if we send a particular packet to many nodes of the network, the chances that some node holding the packet reaches a destination will increase. This approach is also implemented in [20] and [38]. We first consider the minimum delay of 2-hop relay algorithms with redundancy. Then, we design a protocol using redundancy to achieve the minimum delay.

A. Lower bound of delay

In this subsection, we obtain lower bound of delay if only one transmission from a sender to a receiver is permitted in a cell in the below Theorem.

Theorem 5: There is no 2-hop algorithm with redundancy can provide an average delay lower than $O(\sqrt{n\log k})$, if only one transmission from a sender to a receiver is permitted in a cell.

Proof: To prove this result, we consider an ideal situation where the network is empty and only node 1 sends a single packet to k destinations. Clearly the optimal scheme for the source is to send duplicate versions of the packet to new relays whenever possible, and if there is a destination within the same cell as the source, it will choose a destination as relay. And for a duplicate-carrying relay, it sends the packet to be relayed to the destinations as soon as it enters the same cell as a destination. Denote T_N as the time required to reach the k destinations under this optimal strategy for sending a single packet.

In order to avoid the interdependency of the probability that different destinations obtain a packet from the source or the relay nodes, we additionally assume that all the destinations within a same cell as the source or a relay node can obtain the packet during the transmission, which is referred to as a *multi-destination reception* style. Note that our assumption differs from the *multi-user reception* ([20]) in that usually each cell is permitted to have a single reception except there are more than one intended destinations within a the cell, while [20] allows a transmitted packet to be received by all other users in the same cell as the transmitter. Denote T'_N as the time to reach the k destinations when we add the multi-destination reception assumption. It is easy to see that $E\{T_N\} \geq E\{T'_N\}$.

Then, let K_t represent the total number of nodes that act as intermediate relays (including the source) at the beginning of slot t. Because the limitation of 2 hop transmission, a new relay can only be generated by the source. Hence, every timeslot at most one node can be a new relay. Thus, we have for all $t \geq 1$:

$$K_t \leq t \qquad (15)$$

Observe that during slots $\{1, 2, \ldots, t\}$ there are at most K_t nodes holding the packet and willing to help forward it to the destinations. Hence, during this period, the probability that a destination meets at least a relay is at most $1-(1-\frac{1}{c})^{tK_t}$. Moreover, note that since we take the multi-destination reception style, the events that every timeslot different destinations meet the source or a relay node are independent. Thus, the probability that all the k destinations meet at least a relay during this period $\{1, 2, \ldots, t\}$ is at most $\big[1-(1-\frac{1}{c})^{tK_t}\big]^k$. We thus have

$$
\begin{aligned}
P\{T'_N > t\} &\geq 1 - \big[1-(1-\frac{1}{c})^{tK_t}\big]^k \\
&\geq 1 - \big[1-(1-\frac{d}{n})^{t^2}\big]^k \\
&= 1-(1-e^{-\frac{d}{n}t^2})^k \quad (n\to\infty) \qquad (16)
\end{aligned}
$$

Choosing $t = \sqrt{n\log k/d}$ and letting $k \to \infty$, it yields that

$$
\begin{aligned}
P\{T'_N > t\} &\geq 1-(1-e^{-\log k})^k \\
&= 1-(1-\frac{1}{k})^k \\
&= 1-e^{-1} \qquad (17)
\end{aligned}
$$

Thus:

$$E\{T_N\} \geq E\{T'_N\} \geq E\{T'_N \mid T'_N > t\}P\{T'_N > t\} \geq (1-e^{-1})\sqrt{n\log k/d} \quad (18)$$

as $k, n \to \infty$. From (18), we prove the theorem. ∎

B. Scheduling scheme

In the above subsection, we consider the minimum delay of the network if we implement redundant packets transmissions. In this subsection, for acquiring the upper bound of the delay, we propose a 2-hop relay algorithm with redundancy to achieve the minimum delay.

Assume each packet is labeled with a Sender Number *SN*, and a request number *RN* is delivered by the destination to the transmitter just before transmission. In the following algorithm, we let each packet be retransmitted $\sqrt{n\log k}$ times to distinct relay nodes.

Denoting redundancy as A, to better understand the reason we let $A = \Theta(\sqrt{n\log k})$, it is intuitive to simplify a multicast session into two phases: duplication of relays and delivery to destinations, and assume they happen in sequence. Clearly the duration of the first phase is $\Theta(A)$. Consider the duration of the second phase, again it is convenient for us to loosely model the network as a queueing system such that every source destination pair corresponds to a M/M/1 queue, where the exponentially distributed service time has the average $\Theta(n/A)$, i.e., the expected time that a generic relay meets a specific destination. The overall delay for a multicast session would then be $\Theta(n\log k/A)$. To minimize delay, clearly we should let $\Theta(A) = \Theta(n\log k/A)$, which yields $A = \Theta(\sqrt{n\log k})$. Interestingly, this is exactly the lower bound of delay established in Theorem 5.

2-hop Relay Algorithm With Redundancy: In every cell with at least two nodes, randomly select a sender and a receiver with uniform probability over all nodes in the cell. With equal probability, the sender is scheduled to operated in either "source-to-relay" transmission, or "relay-to-destination" transmission, as described below:

1) Source-to-Relay Transmission: The sender transmits packet SN, and does so upon every transmission opportunity until $\sqrt{n\log k}$ duplicates have been delivered to distinct relay nodes (possible be some of the destinations), or until the k destinations have entirely obtained SN. After such a time, the sender number is incremented to $SN+1$. If the sender does not have a new packet to send, stay idle.
2) Relay-to-Destination Transmission: When a node is scheduled to transmit a relay packet to its destinations, the following handshake is taken place:
   - The receiver delivers its current RN number for the packet it desires.
   - The transmitter sends packet RN to the receiver. If the transmitter does not have the requested packet RN, it stays idle for that slot.
   - If all k destinations have already received RN, the transmitter will delete the packet which has SN number equal to RN in its buffer.

Next, we present the performance of this algorithm.

Theorem 6: The 2-hop relay algorithm with redundancy achieves the $O(\sqrt{n\log k})$ delay bound, with a per-node capacity of $\Omega(1/(k\sqrt{n\log k}))$.

Proof: For the purpose of proving this theorem, we consider an extreme case of the packets transmissions. Note that when a new packet arrives at the head of its source queue, the time required for the packet to reach its k destinations is at most $T_N = T_1 + T_2$, where T_1 represents the time required for the source to distribute $\sqrt{n\log k}$ duplicates of the packet, and T_2 represents the time required to reach all the k destinations given that $\sqrt{n\log k}$ relay nodes hold the packet. The reason behind this claim is the *sub-memoryless* property of the random variable T_N ([20]), which means the residual time of T_N given that a certain number of slots have already passed before it expires is stochastically less than the original time T_N.

Now we bound the expectations of T_1 and T_2[8] by taking into account the collisions among the multiple sessions.

The $E\{T_1\}$ bound: For the duration of T_1, there are at least $n - \sqrt{n\log k}$ nodes who do not have the packet. Let G represent the event that every timeslot at least one of these nodes visits the cell of the source. Hence the probability of event G is at least $1-(1-\frac{1}{c})^{n-\sqrt{n\log k}}$. Given this event, the probability that the source is chosen by the 2-hop relay algorithm with redundancy to transmit is expressed by the product $\alpha_1\alpha_2$, representing probabilities for the following conditionally independent events given event G: under the condition that at last one of these $n-\sqrt{n\log k}$ nodes visits the cell of the source, α_1 is the probability that the source is selected from all other nodes in the cell to be the transmitter, and α_2 represents the probability that this source is chosen to operate in "source-to-relay" transmission. From Lemma 6 in [20], we have $\alpha_1 \geq 1/(2+d)$.

The probability α_2 that the source operates in "source-to-relay" transmission is $1/2$. Thus, every timeslot during the interval T_1, the source delivers a duplicate packet to a new node with probability of at least ϕ, where

$$\phi \geq (1-(1-\frac{1}{c})^{n-\sqrt{n\log k}})\frac{1}{2(2+d)} \to \frac{1-e^{-d}}{4+2d}$$

The average time until a duplicate is transmitted to a new node is thus a geometric variable with mean less than or equal to $1/\phi$. It is possible that two or more duplicates are delivered in a single timeslot, if we enable multi-user reception. However, in the worst case, $\sqrt{n\log k}$ of these times are required, so that the average time $E\{T_1\}$ is upper bounded by $\sqrt{n\log k}/\phi$.

The $E\{T_2\}$ bound: To prove the bound on $E\{T_2\}$, let H represent the event that every timeslot in which there are at least $\sqrt{n\log k}$ nodes possess the duplicates of the packet, and note that event H is already a certainty with a probability of

[8]Note that the bounds on $E\{T_1\}$ and $E\{T_2\}$ are computed under suitably large n values.

1. The probability that one of these nodes transmits the packet to one of the destinations is given by the chain of probabilities $\theta_0\theta_1\theta_2\theta_3$. The θ_i values represent probabilities for the following conditionally independent events given event H: Under the condition that there are at least $\sqrt{n\log k}$ nodes possess the duplicates of the packet in every timeslot, θ_0 represents the probability that there is at least one other node in the same cell as the destination ($\theta_0 = 1-(1-\frac{1}{c})^{n-1} \to 1-e^{-d}$), θ_1 represents the probability that the destination is selected as the receiver (similar to α_1, we have $\theta_1 \geq 1/(2+d)$), θ_2 represents the probability that the sender is operates in "relay-to-destination" transmission ($\theta_2 = 1/2$), and θ_3 represents the probability that the sender is one of the $\sqrt{n\log k}$ nodes who possess a duplicate of the packet intended for the destination (where $\theta_3 = \sqrt{n\log k}/(n-1) \geq \sqrt{\log k/n}$). Thus, every timeslot, the probability that each destination receives a desired packet is at least $\frac{1-e^{-d}}{4+2d}\sqrt{\log k/n}$. Similar to Theorem 4, since T_2 completes when all k destinations receive the packet, the value of $E\{T_2\}$ is thus less than or equal to the $\log k$ times of the inverse of that quantity. Hence, we have $E\{T_2\} \leq \frac{4+2d}{1-e^{-d}}\sqrt{n\log k}$.

Finally according to Lemma 2 in [20], we bound the total network delay $E\{W\} = O(\sqrt{n\log k})$, and obtain the achievable per-node capacity under this algorithm is $\Omega(1/k\sqrt{n\log k})$. ■

V. Fundamental delay and capacity Tradeoff

In Section III and Section IV, we present algorithms both without and with redundancy to fulfill the task of MotionCast. In this section, we first draw a comparison of the delay and capacity with the former results. Then we derive the fundamental delay and capacity tradeoff for multicast.

A. Results comparison

Recall that the multi-hop algorithm in [20] is based on flooding the message among the network. It could also serve for multicast. The delay and capacity tradeoffs in the 2-hop relay algorithm without and with redundancy, together with multi-hop relay algorithm with redundancy can be summarized as the following table.

TABLE I
Delay and capacity tradeoffs in different algorithms

scheme	*capacity*	*delay*
2-hop relay w.o. redund	$\Theta(\frac{1}{k})$	$\Theta(n\log k)$
2-hop relay w. redund	$\Omega(\frac{1}{k\sqrt{n\log k}})$	$\Theta(\sqrt{n\log k})$
multi-hop relay w. redund	$\Omega(\frac{1}{n\log n})$	$\Theta(\log n)$

Compared with the multicast capacity of static networks developed in [22], we find that capacity of the 2-hop relay algorithm without redundancy is better when $k = o(n)$; otherwise, capacity remains the same as that of static networks, i.e., mobility cannot increase capacity. Moreover, compared with the results of unicast in [20], we find that capacity diminishes by a factor of $1/k$ and $1/k\sqrt{\log k}$ for the 2-hop relay algorithm without and with redundancy, respectively; delay increases by a factor of $\log k$ and $\sqrt{\log k}$ for the 2-hop relay algorithm without and with redundancy, respectively. This is because we need distribute a packet to k destinations during MotionCast. Particularly, if $k = \Theta(1)$ we find the results of unicast is a special case of our paper.

Furthermore, we see that delay of the 2-hop algorithm with redundancy is better than that of the 2-hop algorithm without redundancy, but its capacity is also smaller than that of the no redundancy algorithm when $k = o(\sqrt{n})$. This suggests that redundant packets transmissions can reduce delay at an expense of the capacity. The ratio between delay and capacity satisfies $delay/rate \geq O(nk\log k)$ for both of these two protocols. However, if we fulfill the job of MotionCast by multiple unicast from the source to each of the k destinations, we find that capacity will diminish by a factor of $1/k$ and delay will increase by a factor of k for both algorithms without and with redundancy, which infers the fundamental tradeoff for unicast established in [20] becomes $delay/rate \geq O(nk^2)$ in MotionCast. Thus, it turns out our tradeoff is better than that of directly extending the tradeoff for unicast to multicast.

B. Fundamental delay and capacity tradeoff for multicast

Observing Table 1 we see that the delay-capacity ratio under these three schemes are $\Theta(nk\log k)$, $O(nk\log k)$, $O(n(\log n)^2)$ respectively, which lead us to suppose the general relationship between delay and capacity is that their ratio is larger than $O(n\log k)$.

Consider a network with n users, and suppose all users receive packets at the same rate λ. A control protocol which makes decisions about scheduling, routing, and packet retransmissions is used to stabilize the network and deliver all packets to their respective k destinations while maintaining an average delay less than some threshold $\overline{W}$. We have the following theorem.

Theorem 7: A necessary condition for any conceivable routing and scheduling protocol with k destinations for transmitting that stabilizes the network with input rates λ while maintaining bounded average delay $\overline{W}$ is given by:

$$\overline{W} \geq \Theta(n\log k)\frac{\lambda}{1-k\lambda} \tag{19}$$

which equals to this following expression,

$$\begin{cases} \lambda = O(1/k), & \overline{W}/\lambda \geq \Theta(n\log k) \\ \lambda = \omega(1/k), & \overline{W} \geq \Theta(n\log k/k) \end{cases} \tag{20}$$

Proof: Suppose the input rate of each of the n sessions is λ, and there exists some stabilizing scheduling strategy which ensures a delay of $\overline{W}$. In general, the delay of packets from individual sessions could be different, and we define W_i as the resulting average delay of packets from session i. We thus have:

$$\overline{W} = \frac{1}{n}\sum_i \overline{W_i} \tag{21}$$

Now we count the number of transmission times for session i. Every time-slot, if this packet or its copies has been transmitted to M different non-destination receivers, the count will be added by M. We define $\overline{R_i}$ as the *non-destination redundancy* which represents the final number of counting when the packet finally reaches the k^{th} destination and end its task, averaged over all packets from session i. That is, $\overline{R_i}$ is average number of non-destination transmissions for a packet from session i. Note that all packets are eventually received by the k destinations, so that $\overline{R_i}+k$ is the actual number of transmissions for packets from session i, and then the average number of successful packet receptions per time-slot is thus given by the quantity $\lambda\sum_{i=1}^{n}(\overline{R_i}+k)$. Since each of the n users can receive at most 1 packet per time-slot, we have:

$$\lambda\sum_{i=1}^{n}(\overline{R_i}+k)\leq n \tag{22}$$

Now consider a single packet P which enters the network from session i. This packet has an average delay of $\overline{W_i}$ and an average non-destination redundancy of $\overline{R_i}$. Let random variables W_i and R_i represent the actual delay and non-destination redundancy for this packet. We have:

$$\begin{aligned}\overline{W_i} &= \mathbb{E}\{W_i|R_i\leq 2\overline{R_i}\}Pr[R_i\leq 2\overline{R_i}]+\\ &\quad \mathbb{E}\{W_i|R_i\geq 2\overline{R_i}\}Pr[R_i\geq 2\overline{R_i}]\\ &\geq \mathbb{E}\{W_i|R_i\leq 2\overline{R_i}\}Pr[R_i\leq 2\overline{R_i}]\\ &\geq \mathbb{E}\{W_i|R_i\leq 2\overline{R_i}\}\frac{1}{2}\end{aligned} \tag{23}$$

where the last inequality follows because $Pr[R_i\leq 2\overline{R_i}]\geq\frac{1}{2}$ for any nonnegative random variable R_i.

Consider now a virtual system in which there are $2\overline{R_i}$ users initially holding packet P, and let Z_m represent the time required for one of these users to enter the same cell as the m^{th} destination. Then let Z represent the time required for these users to enter all the k destinations, so we have $Z=\max\{Z_1,Z_2,\ldots,Z_k\}$. Note that the distribution of each Z_m is the same as $Pr[Z_m>w]=(1-\phi)^{[w]}$, in which $\phi=1-(1-\frac{1}{c})^{2\overline{R_i}}$. And thus $\mathbb{E}\{Z_m\}=\frac{1}{\phi}$.

In order to connect this variable Z to our interest W_i, we develop another parameter W_i^{rest}, which represents the corresponding delay under the *restricted scheduling policy* that schedules packets as before until either the packet is successfully delivered to all k destinations, or the redundancy increases to $2\overline{R_i}$(where no more redundant transmissions are allowed). Since this modified policy restricts redundancy to at most $2\overline{R_i}$, the delay W_i^{rest} is stochastically greater than the variable Z, representing the delay in a virtual system with only one packet that is initially held by $2\overline{R_i}$ users. In addition, as the restricted policy is identical to the original policy whenever $R_i\leq 2\overline{R_i}$, hence $\mathbb{E}\{W_i|R_i\leq 2\overline{R_i}\}=\mathbb{E}\{W_i^{rest}|R_i\leq 2\overline{R_i}\}$.

Finally, we introduce the last much easier calculated continuous variable $\widetilde{Z}$, which is also the maximum of several ones - $\widetilde{Z}=\max\{\widetilde{Z}_1,\widetilde{Z}_2,\ldots,\widetilde{Z}_k\}$. For each of them has the same distribution as $Pr[\widetilde{Z}_m>w]=e^{-\gamma w}=(1-\phi)^w\leq(1-\phi)^{[w]}=Pr[Z_m>w]$, where $\gamma=\log\frac{1}{1-\phi}$.

So now we put the relationship among these three variables clearly as follows:

$$W_i^{rest}\succeq Z\succeq\widetilde{Z}^{\,9} \tag{24}$$

[9]Because $Pr[Z>w]=1-Pr[Z\leq w]=1-\prod_{m=1}^{k}Pr[Z_m\leq w]\geq 1-\prod_{m=1}^{k}Pr[\widetilde{Z}_m\leq w]=Pr[\widetilde{Z}]$, and according to the definition in [39], we have that Z is stochastically greater than $\widetilde{Z}$.

Further, although Z and $\widetilde{Z}$ defined in our paper are a little different from those defined in [1], i.e., $Z=max\{Z_1,Z_2,...,Z_K\}$ and $\widetilde{Z}=max\{\widetilde{Z_1},\widetilde{Z_2},...,\widetilde{Z_k}\}$, they also follow *claim 1* and *claim 2* in [1]. Thus, we have this useful inequality:

$$\mathbb{E}\{W_i|R_i\leq 2\overline{R}_i\}\geq\inf_{\Theta}\mathbb{E}\{Z|\Theta\}\geq\inf_{\widetilde{\Theta}}\mathbb{E}\{\widetilde{Z}|\widetilde{\Theta}\} \tag{25}$$

where the conditional expectation is minimized over all conceivable events Θ(for Z, while $\widetilde{\Theta}$ for $\widetilde{Z}$) which occur with probability greater than or equal to $1/2$.

Until now we have to calculate the last value $\inf_{\widetilde{\Theta}}\mathbb{E}\{\widetilde{Z}|\widetilde{\Theta}\}$. From Lemma 8 in [20], whose result been put as follows:

For any nonnegative random variable X, we have:

$$\begin{aligned}\inf_{\{\Theta|Pr[\Theta]\geq\frac{1}{2}\}}\mathbb{E}\{X|\Theta\}&=\mathbb{E}\{X|X<w\}2Pr[X<w]+\\ &\quad w(1-2Pr[X<w])\end{aligned} \tag{26}$$

Where w is the unique real number such that $Pr[X<w]\leq\frac{1}{2}$ and $Pr[X\leq w]\geq\frac{1}{2}$.

Note that in the special case when $P(x)$ is continuous at $x=w$, then $Pr[X<w]=Pr[X\leq w]=\frac{1}{2}$ and hence we get the simpler expression:

$$\inf_{\widetilde{\Theta}}\mathbb{E}\{\widetilde{Z}|\widetilde{\Theta}\}=\mathbb{E}\{\widetilde{Z}|\widetilde{Z}\leq w\} \tag{27}$$

Now recall the distribution expression of $\widetilde{Z}$:

$$Pr[\widetilde{Z}\leq z]=\prod_{m=1}^{k}Pr[\widetilde{Z}_m\leq z]=(1-e^{-z\gamma})^k. \tag{28}$$

Then we get the value of w: $w=-\frac{1}{\gamma}\ln[1-(\frac{1}{2})^{\frac{1}{k}}]$.

Return to our question, we do the calculation as follows:

$$\begin{aligned}\mathbb{E}\{\widetilde{Z}|\widetilde{Z}\leq w\} &= \frac{\int_0^w x[(1-e^{-\gamma x})^k]_x' dx}{Pr[\widetilde{Z}\leq w]}\\ &= 2x(1-e^{-x\gamma})^k|_0^w-2\int_0^w(1-e^{-x\gamma})^k dx\\ &= w-2\int_0^w\sum_{i=0}^{k}\binom{k}{i}(-1)^i e^{-ix\gamma}dx\\ &= w+2\sum_{i=0}^{k}\binom{k}{i}(-1)^{i+1}\int_0^w e^{-ix\gamma}dx\\ &= -w+2\sum_{i=1}^{k}\frac{1}{i\gamma}\binom{k}{i}(-1)^i[e^{-iw\gamma}-1]\\ &\triangleq -w+2X(k)\end{aligned} \tag{29}$$

Note that $\binom{k}{i}=\binom{k-1}{i-1}+\binom{k-1}{i}$, we can calculate $X(k)$ by gradually reducing the variable k, as $X(1)$ can be easily obtained as $X(1)=-\frac{1}{\gamma}[e^{-w\gamma}-1]$.

$$
\begin{aligned}
X(k)-X(k-1) &= \sum_{i=1}^{k}\frac{1}{i\gamma}\binom{k-1}{i-1}(-1)^i[e^{-iw\gamma}-1] \\
&= \frac{1}{k\gamma}\sum_{i=0}^{k}\binom{k}{i}(-1)^i[e^{-iw\gamma}-1] \\
&= \frac{1}{k\gamma}[(1-e^{-w\gamma})^k-(1-1)^k] \\
&= \frac{1}{2k\gamma} \qquad (30)
\end{aligned}
$$

Wherein $\frac{1}{i}\binom{k-1}{i-1}=\frac{1}{k}\binom{k}{i}$. And continue this recursion, we have that:

$$
\begin{aligned}
X(k) &= X(1)+\frac{1}{2\gamma}(\frac{1}{2}+\frac{1}{3}+\ldots+\frac{1}{k}) \\
&= -\frac{1}{\gamma}[e^{-w\gamma}-1]+\frac{1}{2\gamma}(\frac{1}{2}+\frac{1}{3}+\ldots+\frac{1}{k}) \\
&\triangleq -\frac{1}{\gamma}[e^{-w\gamma}-1]+\frac{1}{2\gamma}f(k) \qquad (31)
\end{aligned}
$$

Wherein $f(k)=\Theta(\log k)$. Connecting (27),(29) and (31) can we have that:

$$
\begin{aligned}
\inf_{\widetilde{\Theta}}\mathbb{E}\{\widetilde{Z}|\widetilde{\Theta}\} &= \mathbb{E}\{\widetilde{Z}|\widetilde{Z}\le w\} \\
&= -w+2X(k) \\
&= -\frac{1}{\gamma}\ln[1-(\frac{1}{2})^{\frac{1}{k}}]-\frac{2}{\gamma}(\frac{1}{2})^{\frac{1}{k}}+\frac{1}{\gamma}f(k) \\
&\triangleq \frac{1}{\gamma}g(k) \qquad (32)
\end{aligned}
$$

Wherein $g(k)=\Theta(\log k)$. From the definitions of γ and ϕ, we have $\gamma=\log(1/(1-\frac{1}{c})^{2\overline{R}_i})=2\overline{R}_i\log(1+\frac{1}{c-1})$. Since $\log(1+x)\le x$ for any x, we have $\gamma\le 2\overline{R}_i/(c-1)$. Then using (23), (25) and (32) in (21) yields:

$$
\begin{aligned}
\overline{W}=\frac{1}{n}\sum_{i=1}^{n}\overline{W}_i &\ge \frac{1}{2n}\sum_{i=1}^{n}\frac{1}{\gamma}g(k) \\
&\ge \frac{g(k)}{2n}\sum_{i=1}^{n}\frac{c-1}{2\overline{R}_i} \\
&\ge g(k)(c-1)\frac{1}{4n}\sum_{i=1}^{n}\frac{1}{\overline{R}_i} \\
&\ge g(k)\frac{c-1}{4}\frac{1}{\frac{1}{n}\sum_{i=1}^{n}\overline{R}_i} \qquad (33)
\end{aligned}
$$

Where (33) follows from Jensen's inequality, noting that the function $f(R)=\frac{1}{R}$ is convex, and hence $\frac{1}{n}\sum_{i=1}^{n}f(\overline{R}_i)\ge f(\frac{1}{n}\sum_{i=1}^{n}\overline{R}_i)$. Combining (33) and (22), we have:

$$
\overline{W}\ge g(k)\frac{c-1}{4}\frac{\lambda}{1-k\lambda}=\Theta(n\log k)\frac{\lambda}{1-k\lambda} \qquad (34)
$$

wherein c has the same order as n. Proving the theorem. ■

We notice that $\overline{R_i}>=0$ in inequality (22), thus $\lambda<\frac{1}{k}$. Divide λ on both sides of formula (1), we get $\frac{\overline{W}}{\lambda}\ge\Theta(n\log k)\frac{1}{1-k\lambda}$. Since $\lambda=o(1/k)$, i.e. $1-k\lambda$ remain a constant as n and k growing into infinity, we get

$$
\frac{\overline{W}}{\lambda}\ge\Theta(n\log k) \qquad (35)
$$

Finally, we have the following corollary.

Corollary 1: For any scheduling algorithm in the network with n nodes moving according to i.i.d. pattern, and each desire to send its data to k distinct destination nodes, the achievable capacity λ and delay $\overline{W}$ satisfy the fundamental relationship: $\frac{\overline{W}}{\lambda}=\Omega(n\log k)$.

VI. Conclusion and Future Work

In this paper, we study delay and capacity tradeoffs for MotionCast. We utilize redundant packets transmissions to realize the tradeoff, and present the performance of the 2-hop relay algorithm without and with redundancy respectively. We find that the capacity of the 2-hop relay algorithm without redundancy is better than that of static networks when $k=o(n)$. And our tradeoff is better than that of directly extending the tradeoff for unicast to multicast. Moreover, we prove that the fundamental delay-capacity tradeoff ratio for multicast is $\Omega(n\log k)$. We have not taken into account the multi-hop transmission schemes and the effect of different mobility patterns yet, which could be a future work.

References

[1] M. Mauve, A. Widmer and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," *IEEE Network*, vol. 2, no. 6, pp. 30-39, 2001.

[2] C.M. Cordeiro, H. Gossain and D. Agrawal, "Multicast over wireless mobile ad hoc networks: Present and future directions," *IEEE Network*, vol. 17, pp. 52-59 2003.

[3] E.M. Royer and C.E. Perkins, "Multicast operation of the ad-hoc on-demand distance vector routing protocol," in *Proc. ACM Mobicom*, 1999.

[4] Y. Wu , P.A. Chou and S. Kung, "Minimum-energy multicast in mobile ad hoc networks using network coding," *IEEE Transactions on Communications*, vol. 53, no. 11, pp. 1906-1918, Nov. 2005.

[5] K. Viswanath , K. Obraczka and G. Tsudik, "Exploring mesh and tree-based multicast routing protocols for manets," *IEEE Transactions on Mobile Computing*, vol. 5, no. 1, pp. 28-42 , Jan. 2006.

[6] P. Hsiu and T. Kuo, "A Maximum-Residual Multicast Protocol for Large-Scale Mobile Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 11, pp. 1441-1453 , Nov. 2009.

[7] C. Gui and P. Mohapatra, "Scalable multicasting in mobile ad hoc networks," in *Proc. IEEE INFOCOM*, 2004.

[8] X. Xiang, X. Wang and Y. Yang, "Stateless Multicasting in Mobile Ad Hoc Networks," *IEEE Transactions on Computers*, vol. 59, no. 8, pp. 1076-1090 , Aug. 2009.

[9] B. Adamson et al., "Simplified Multicast Forwarding," *Internet Draft*, Sept. 2010

[10] S.M. Das, H. Pucha and Y.C. Hu, "Distributed Hashing for Scalable Multicast in Wireless Ad Hoc Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 3, pp. 347-362 , Mar. 2008.

[11] A.B. Mnaouer , L. Chen , C.H. Foh and J.W. Tantra, "OPHMR: An Optimized Polymorphic Hybrid Multicast Routing Protocol for MANET," *IEEE Transactions on Mobile Computing*, vol. 6, no. 5, pp. 551-562, May 2007.

[12] J. Burgess, B. Gallagher, D. Jensen, and B.N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," in *Proc. IEEE INFOCOM*, 2006.

[13] N. Banerjee, M.D. Corner, D. Towsley, and B.N. Levine, "Relays, Base Stations, and Meshes: Enhancing Mobile Networks with Infrastructure," in *Proc. ACM Mobicom*, 2008.

[14] A. Balasubramanian, R. Mahajan, A. Venkataramani, B.N. Levine, and J. Zahorjan, "Interactive WiFi Connectivity for Moving Vehicles," in *Proc. ACM Sigcomm*, 2008.

[15] A. Balasubramanian, B.N. Levine and A. Venkataramani, "Enhancing Interactive Web Applications in Hybrid Networks," in *Proc. ACM Mobicom*, 2008.

[16] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass and J. Scott, "Impact of Human Mobility on the Design of Opportunistic Forwarding Algorithms," in *Proc. IEEE INFOCOM*, 2006.

[17] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *Proc. ACM MobiHoc*, 2004.
[18] A. Pentland, R. Fletcher, A. Hasson, "Daknet: rethinking connectivity in developing nations," *IEEE Computer*, 2004.
[19] M. Garetto and E. Leonardi, "Restricted Mobility Improves Delay-Throughput Tradeoffs in Mobile Ad Hoc Networks," *IEEE Transactions on Information Theory*, vol. 56, no. 10, pp 5016 - 5029, Oct. 2010.
[20] M. J. Neely, and E. Modiano, "Capaicty and delay tradeoffs for ad hoc mobile networks," *IEEE Transactions on Information Theory*, vol. 51, no. 6, pp. 1917-1937, Jun. 2005.
[21] 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
[22] X. Li, "Multicast capacity of Wireless Ad Hoc Networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 3, pp. 950-961, 2009.
[23] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388-404, Mar. 2000.
[24] A. Keshavarz-Haddad, V. Ribeiro, and R. Riedi, "Broadcast capacity in multihop wireless networks," in *Proceedings of ACM MobiCom*, Sept. 2006.
[25] X. Li, Y. Liu, S. Li, and S. Tang, "Multicast Capacity of Wireless Ad Hoc Networks Under Gaussian Channel Model," to appear in *IEEE/ACM Transactions on Networking*, 2010.
[26] P. Jacquet and G. Rodolakis, "Multicast scaling properties in massively dense ad hoc networks," in *Proceedings of International Conference on Parallel and Distributed Systems*, July 2005.
[27] S. Shakkottai, X. Liu and R. Srikant, "The multicast capacity of large multihop wireless networks," to appear in *IEEE/ACM Transactions on Networking*, 2010.
[28] M. Grossglauser and D. N. C. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, pp. 477-486, Aug. 2002.
[29] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah,"Throughput-delay trade-off in wireless networks," in *Proceedings of IEEE INFOCOM*, Mar. 2004.
[30] M. Ibrahim, A. A. Hanbali, and P. Nain, "Delay and resource analysis in MANETs in prsence of throwboxes," in *Proceedings of International Symposium on Computer Performance, Modeling, Measurements, and Evaluation*, 2007.
[31] N. Banerjee, M. Corner, D. Towsley, and B. Levine, "Relays, Base Stations, and Meshes: Enhancing mobile networks with infrastructure," in *Proceedings of ACM MobiCom*, Sept. 2008.
[32] M. Garetto, P. Giaccone, E. Leonardi, "Capacity scaling in ad hoc networks with heterogeneous mobile nodes: The super-critical regime," in *IEEE/ACM Transactions on Networking*, vol. 17, no. 5, pp. 1522-1535, Oct. 2009.
[33] M. Garetto, P. Giaccone, E. Leonardi, "Capacity scaling in ad hoc networks with heterogeneous mobile nodes: The subcritical regime," in *IEEE/ACM Transactions on Networking*, vol. 17, no. 6, pp. 1888-1901, Dec. 2009.
[34] Lei Ying, Sichao Yang and R. Srikant, "Optimal delay-throughput trade-offs in mobile ad hoc Networks," *IEEE Transactions on Information Theory*, vol. 54, no. 9, pp. 4119-4143, Sept. 2008.
[35] S. Toumpis and A. J. Goldsmith,"Large wireless networks under fading, mobility, and delay constraints," in *Proceedings of IEEE INFOCOM*, Mar. 2004.
[36] M. Li and Y. Liu, "Rendered path: range-free localization in anisotropic sensor networks with holes," in *Proceedings of ACM MobiCom*, Sept. 2007.
[37] R. L. Cruz and A. V. Santhanam, "Hierarchical link scheduling and power control in multihop wireless networks," in Proceedings of the Annual Allerton Conference on communication, Control and Computing, Oct. 2002.
[38] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multi-copy case," *IEEE/ACM Transaction on Networking*, vol. 16, no. 1, pp. 77-90, Feb. 2008.
[39] S. M. Ross, *Stochastic processes*. New York: John Wiley & Sons, 1996.
[40] C. Hu, X. Wang and F. Wu,"MotionCast: On the Capacity and Delay Tradeoffs," in *ACM MobiHoc 09*, New Orleans, May 2009.

APPENDIX I – THE DERIVATION OF p AND q

Since p represents the probability of finding at least two nodes in a particular cell, the opposite event of it is there is no node (and this happens with a probability of $(1-\frac{1}{c})^n$) or only one node in the cell (this occurs with a probability of $\frac{n}{c}(1-\frac{1}{c})^{n-1}$, where n infers that the node in the cell can be any one among all n nodes of the network). Thus, we have the expression of (2).

As for q, it represents the probability of finding a source-destination pair within a cell. Note that in our traffic pattern, we suppose the number of nodes n is divisible by $k+1$ and uniformly and randomly divide the network into different groups with each of them having $k+1$ nodes. Also assume packets from each node i in a specific group must be delivered to all the other nodes within the group. Thus, any two nodes within a same group is a pair of source-destination. The probability that there is not any source-destination pair belonging to any group within a particular cell is $\frac{k+1}{c}(1-\frac{1}{c})^k+(1-\frac{1}{c})^{k+1}$. Since each group is independent with others, the probability that there is not any source-destination pair in the cell is thus $\frac{n}{k+1}$th power of the above quantity. Hence, the probability of the inverse event q is given by (3).

APPENDIX II – USEFUL LEMMAS

Here we present useful lemmas in this paper.

Lemma 1: $\sum_{i=1}^{k}\frac{(-1)^{i-1}}{i}\binom{k}{i}=\ln(k+1)+r$, where $k\geq 1$ and r is Euler constant.

Proof: Denote the left-hand-side of the equation by $A(k)$, then we have $A(k-1)=\sum_{i=1}^{k}\frac{(-1)^{i-1}}{i}\binom{k-1}{i}$. Notice that $\binom{k}{i}=\binom{k-1}{i}+\binom{k-1}{i-1}$, it follows

$$A(k)-A(k-1) = \sum_{i=1}^{k}\frac{(-1)^{i-1}}{i}\binom{k-1}{i-1} = \frac{1}{k}\sum_{i=1}^{k}(-1)^{i-1}\binom{k}{i} \quad (36)$$

Recall that $(1-1)^k=\sum_{i=0}^{k}(-1)^i\binom{k}{i}=0$, hence we obtain $\sum_{i=1}^{k}(-1)^{i-1}\binom{k}{i}=-\sum_{i=1}^{k}(-1)^i\binom{k}{i}=-\left[\sum_{i=0}^{k}(-1)^i\binom{k}{i}-1\right]=1$. Combining with (36), we get $A(k)-A(k-1)=\frac{1}{k}$, then

$$A(k) = A(1)+\sum_{i=2}^{k}\left[A(k)-A(k-1)\right] = 1+\sum_{i=2}^{k}\frac{1}{k}=\sum_{i=1}^{k}\frac{1}{k} \quad (37)$$

Since the right-hand-side of (37) is the harmonic series, this lemma holds. ■

Lemma 2: Suppose $X_1, X_2, ..., X_k$ are continuous i.i.d. exponential variables with expectation of $1/a$, and denote $X_{max}=max\{X_1,X_2,...,X_k\}$, then $E\{X_{max}\}=\Theta(\log k/a)$ (for simplicity, we can treat $E\{X_{max}\}$ just as $\log k/a$), where $k\geq 1$.

Proof: Consider the cdf of X_{max},

$$F_{X_{max}}(t)=P\{X_{max}\leq t\}=(1-e^{-at})^k \quad (38)$$

Thus, the pdf of X_{max} can be expressed as

$$f_{X_{max}}(t) = \frac{dF_{X_{max}}(t)}{dt} = k(1-e^{-at})^{k-1} \cdot ae^{-at} \quad (39)$$

Then, we obtain

$$\begin{aligned} E\{X_{max}\} &= \int_0^\infty k(1-e^{-at})^{k-1} ae^{-at} \cdot tdt \\ &= ka \int_0^\infty \sum_{i=0}^{k-1} \binom{k-1}{i} (-1)^i e^{-a(i+1)t} \cdot tdt \\ &= \sum_{i=0}^{k-1} ka \binom{k-1}{i} (-1)^i \frac{1}{\big[a(i+1)\big]^2} \\ &= \sum_{i=1}^{k} ka \binom{k-1}{i-1} (-1)^{i-1} \frac{1}{a^2 i^2} \\ &= \frac{k}{a} \sum_{i=1}^{k} \frac{(-1)^{i-1}}{i^2} \binom{k-1}{i-1} \\ &= \frac{1}{a} \sum_{i=1}^{k} \frac{(-1)^{i-1}}{i} \binom{k}{i} \end{aligned} \quad (40)$$

According to Lemma 1, we conclude this lemma. ■