Delay and Capacity Tradeoff Analysis for MotionCast

Xinbing Wang, *Member, IEEE*, Wentao Huang, Shangxing Wang, Jinbei Zhang, and Chenhui Hu

***Abstract*—In this paper, we define multicast for an 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 the 2-hop algorithm *without redundancy* are $\Theta(1/k)$ and $\Theta(n\log k)$, respectively; for the 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 by Li [*IEEE/ACM Trans. Netw.*, vol. 17, no. 3, pp. 950–961, Jun. 2009] 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 by Neely and Modiano [*IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 1917–1937, Jun. 2005] 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*—Capacity, delay, multicast, 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 uses 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]–[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 compared 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 an 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 doing so, we provide a theoretical foundation to the design of intelligent communication schemes that 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 [4]. Several interesting studies have later emerged aimed at establishing the fundamental scaling laws for networks with multicast traffic. Li *et al.* [3], [22], [23] 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 [4] and broadcast [5]. Under a more general Guassian channel model, multicast capacity is investigated in [6] using percolation theory. Jacquet *et al.* [7] 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.* [8] propose a comb-based architecture for multicast routing that achieves the upper bound for capacity in an order sense.

In contrast to the discussed static networks, Gossglauser and Tse [9] for the first time have shown that a constant unicast

Manuscript received October 09, 2009; revised August 13, 2010 and November 26, 2010; accepted January 16, 2011; approved by IEEE/ACM Transactions on Networking Editor B. N. Levine. Date of publication March 03, 2011; date of current version October 14, 2011. This work was supported by the National Basic Research Program of China (973 Program) under Grant 2011CB302701, NSF China under Grant 60832005, the China Ministry of Education Fok Ying Tung Fund under Grant 122002, and a Qualcomm Research Grant. An earlier version of this paper appeared in the Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), New Orleans, LA, May 18–21, 2009.

X. Wang, W. Huang, S. Wang, and J. Zhang are with the Department of Electric Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: xwang8@sjtu.edu.cn; wht@sjtu.edu.cn; wsx@sjtu.edu.cn; zjb@sjtu.edu.cn).

C. Hu was with the Department of Electric Engineering, Shanghai Jiao Tong University, Shanghai 200240, China. He is now with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA (e-mail: hch@sjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2011.2109042

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 $\Omega(n)$ [1], [10], 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 [13]–[16], networks based on human mobility [17], disaster-relief networks [18], and Internet access to remote villages [19].

The asymptotic capacity delay tradeoff in MANETs exploiting store–carry–forward schemes has attracted significant attention and is studied by many authors under various mobility models. The most studied model is arguably the independently and identically distributed (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 [1] 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 [16], 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 [2] later study the fundamental capacity–delay tradeoff and identify the limiting factors of the existing scheduling schemes in MANETs. Recently, Ying *et al.* [15] propose joint coding-scheduling algorithms to improve capacity–delay tradeoffs, while Garetto and Leonardi [24] show that it is possible to exploit node heterogeneity under a restricted i.i.d. mobility model 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 traffic. 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 the following.

- What is the maximum per-node MotionCast capacity?
- What is the delay for maximal capacity achieving schemes, 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 begin, we propose a 2-hop relay algorithm without redundancy. This algorithm is a generalized version of the algorithm presented in [1] 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 *noncooperative 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 loosen the constraints of our initial model by permitting information dissemination among destinations (*cooperative 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 possibly 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 counterintuitively, we find that there would be no gain in performance for the cooperative scheme compared to the noncooperative 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, in each time slot, 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 $\Omega(\sqrt{n \log k})$. Moreover, 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 the 2-hop relay algorithm without redundancy, the capacity for MotionCast is $\Theta(1/k)$ with an average delay of $\Theta(n \log k)$. Notice that the per-node capacity is better than the results of a static multicast scenario in [3] as long as k is strictly less than n in an order sense, i.e., $k = o(n)$. For the 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

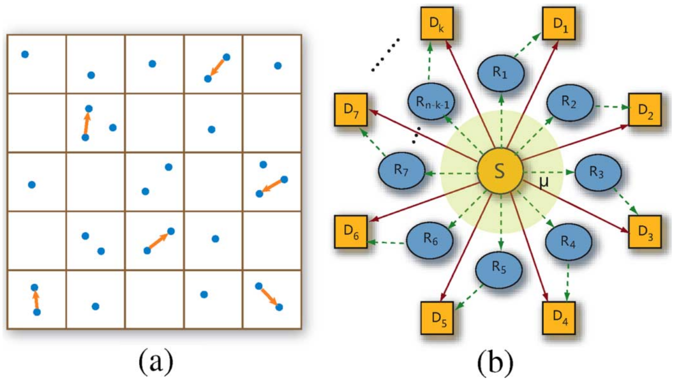


Fig. 1. Cell-partitioned MANET model with c cells and n mobile nodes under multicast traffic pattern. (a) Network model. (b) Traffic pattern.

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 [1] and [18]. Suppose the network is a unit square and there are n mobile nodes in it. Then, we divide it into c nonoverlapping cells with equal size as depicted in Fig. 1. We assume nodes can communicate with each other only when they are within a same cell (to locate the nodes, please refer to [17] 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. 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 an 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. Assume packets from each node i in a specific group must be delivered to all the other nodes within the group. 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 time slot, if more than one node is performing as a relay 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 to a destination that 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 noncooperative mode.

Notations: In our paper, 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 exists a constant c such that

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

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 noncooperative mode and cooperative mode. Then, we explore the maximum capacity and the minimum delay in these situations.

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

[2] Theorems 3 and 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.

A. Under Noncooperative Mode

Here, 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 noncooperative mode, which means a destination cannot be a relay.

2-Hop Relay Algorithm Without Redundancy I: During a time slot, 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.
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 another node destined for the receiver, transmit. If all the destinations that 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 an 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 equal the maximum of k such i.i.d. random delays and turns out to be $\Theta(n \log k)$. We formally derive the performance of the above algorithm.

The algorithm has an advanced decoupling feature between all n multicast sessions, as illustrated in Fig. 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

$$p = 1 - \left(1 - \frac{1}{c}\right)^{n} - \frac{n}{c}\left(1 - \frac{1}{c}\right)^{n-1} \qquad (2)$$

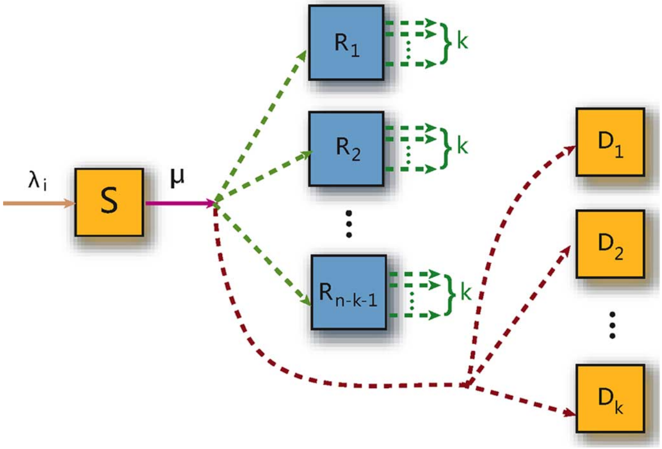


Fig. 2. A decoupled queuing model of the network as seen by the packets transmitted from a single source to multiple destinations.

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

When n tends to infinity, it follows $p \rightarrow 1 - (d+1)e^{-d}$ and $q \rightarrow 1 - e^{-\frac{k}{k+1}d}(1 + \frac{k}{c})^{\frac{n}{k+1}}$. Thus, if $k = o(n)$, $q \rightarrow 0$;[5] else if $k = \Theta(n)$, $q \rightarrow 1 - (d+1)e^{-d}$. Intuitively, when k approaches the same order as n, the multicast will reduce 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 time slot. If the exogenous input stream to node i that 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

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

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

Proof: A decoupled view of the network as seen by a single source i is shown in Fig. 2. Due to the i.i.d. mobility model, the source user can be represented as a Bernoulli/Bernoulli queue, where in every time slot 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 the probability that the source is scheduled to transmit directly to the destination, and r_2 equals the probability that the source is

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

[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.

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

[6] A transmission opportunity arises when a 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 time slot, independent of whether or not there is a packet waiting in the queue.

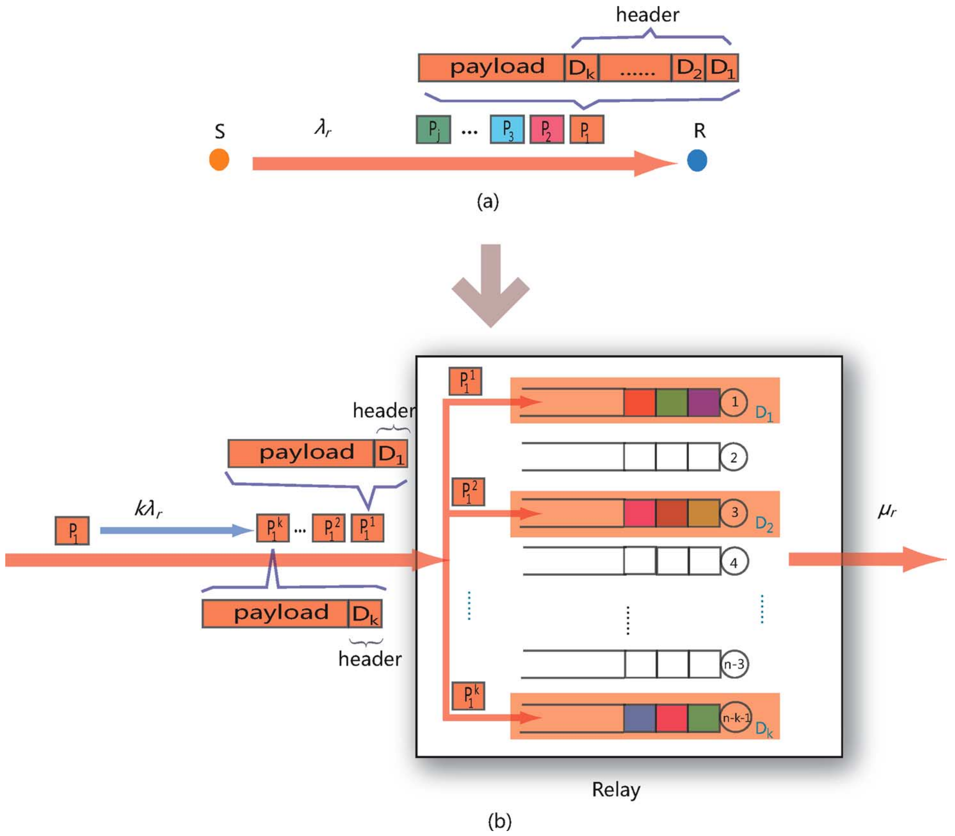


Fig. 3. More delicate view of a relay–destinations transmission. (a) Each packet delivered to a relay node has a similar form that contains its destinations' information in the header. (b) Each relay can make a packet into k similar copies and can be modeled as a node having $n-k-1$ parallel subqueues buffering packets intended for different destinations. k subqueues associated with k destinations of the current source are shaded in the figure.

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 time slot, 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

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

It follows from (6) and (8) 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 $\bar{L}_{\rm s} = \frac{\rho(1-\lambda_i)}{1-\rho}$, where $\rho \triangleq \frac{\lambda_i}{\mu}$. From Little's theorem, the average waiting time in the source is $E\{W_{\rm s}\} = \frac{\bar{L}_{\rm s}}{\lambda_i} = \frac{1-\lambda_i}{\mu-\lambda_i}$. Furthermore, 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 header that tells the relay which k destinations this packet should be transmitted to, shown in Fig. 3(a). Based on this information, the relay can make k similar copies, each of which contains less redundant data in its header just indicating its own corresponding destination. 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 Fig. 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 $\lambda_{\rm r} = \lambda_i p_i$ (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 time slot, 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 will not influence our results, so we omit it. Hence, the input rate of a subqueue is $k\lambda_{\rm r}$. On the other hand, the subqueue in the relay node is scheduled for a potential packet transmission to a destination node with probability $\mu_{\rm r} = \frac{r_2}{n-k-1}$ (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 that is identical to a continuous-time M/M/1 queue with input rate $k\lambda_{\rm r}$ and service rate $\mu_{\rm r}$. Each destination i $(1 \leq i \leq k)$ obtains the packet from the relay through such a queue, thus the waiting time for it is an exponential distributed variable with an expectation of $E\{W_{\rm rd}^i\} = 1/(\mu_{\rm r} - k\lambda_{\rm r})$.

The resulting waiting time $W_{\rm rd}$ for multicast is determined by the maximum value among all the waiting times $W_{\rm rd}^1, W_{\rm rd}^2, \ldots, W_{\rm rd}^k$ of these k destinations. Due to the fact that: 1) all k destinations share the same arrival process, and 2) the interference constraint that a relay node can communicate with only one destination in one time slot, $\{W_{\rm rd}^k\}$ are correlated over k. However, we can construct a set of dual random variables $\{W_{\rm rd}^{\prime i}\}$ such that they are i.i.d. They provide a slightly alternated view of the queueing system depicted in Fig. 3, with *multidestination reception* enabled, i.e., if a source encounters more than one destination, the packet will be transmitted to all of them. Additionally, we hypothesize that the arrival processes of different destinations are independent, each with rate $k\lambda_{\rm r}$. In the following, we shall show $W_{\rm rd}^k = W_{\rm rd}^{\prime k}$ w.h.p.

Condition on the event that the relay encounters one or more destinations, and denote $\varphi_1, \varphi_{>1}$ 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 $k = \Theta(n)$. Therefore, $\mu_{\rm r} = \Theta(\mu_{\rm r}')$, which indicates that multireception does not affect the service process in an order sense. Similarly, also notice that the input process for a subqueue in the two queueing systems, though constructed on independent probability spaces, is the same $k\lambda_{\rm r}$ and does not rely on network scale n. Due to the nature of the M/M/1 queue, the waiting time $W_{\rm rd}^i$ or $W_{\rm rd}^{\prime i}$ only depends on the input and the service process, and it is clear that $W_{\rm rd}^i = \Theta(W_{\rm rd}^{\prime i})$ w.h.p. In other words, there exists constant c_i such that $W_{\rm rd}^i = c_i W_{\rm rd}^{\prime i}$ w.h.p. By Lemma 2 (see the proof in Appendix II), we obtain that $E\{W_{\rm rd}\} = \Theta(E\{W_{\rm rd}'\}) = \Theta(\log k/(\mu_{\rm r} - k\lambda_{\rm r}))$. Thus, if the packet is delivered through the path "source–relay–destinations," the average delay is $E\{W_{\rm s}\} + E\{W_{\rm rd}\}$.

While if the packet is directly sent to the destinations by the source, it will wait at the source for a time $W_{\rm s}$ 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 subbuffers corresponding to its k destinations similarly. The source will copy this packet into $k-1$ similar duplicates and add them into respective subbuffers 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 needs to 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 $1/(\frac{r_1}{k} - \frac{\lambda_i r_1}{\mu})$. By Lemma 2 and the same method in the above calculation of the delay through 2-hop route, we have the expected waiting time for the packet to reach all $k-1$ remnant destinations as $E\{W_{\rm sd}\} = \Theta(\log(k-1)/(\frac{r_1}{k} - \frac{\lambda_i r_1}{\mu}))$.

Finally, by weighting the delay that occurs in both routings, we achieve the total network delay as

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

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

From the above discussion, we conclude the theorem. ■

B. Under Cooperative Mode

In Section III-A, we proposed a 2-hop relay algorithm without redundancy obtaining per-node capacity $\Omega(1/k)$ with delay $O(n\log k)$. Here, we bring forward a more general algorithm that 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 simpler than the first one, we adopt this algorithm and refer to it as the *2-hop relay algorithm without redundancy* for briefness 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 time slot, 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 following two options.

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 that are destined for the receiver and have not been transmitted to the receiver yet, then choose the latest one, transmit. If all the destinations that 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 $k \leq n$.

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

$$cp = 2n\mu. \tag{8}$$

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. 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 $E\{W_{\rm s}\} = \frac{1-\lambda_i}{\mu-\lambda_i}$. Moreover, 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 time slot, this relay independently receives a packet with probability $\lambda_{\rm r} = \frac{\lambda_i}{n-1}$. On the other hand, the relay node is scheduled for a potential packet transmission to a destination node with probability $\mu_{\rm r} = \frac{\mu}{n-2}$ (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 is as the relay, no matter if it is a destination. The difference is that if the relay is a destination node, it needs only to relay the packet to the rest $k-1$ destinations. Otherwise, it needs to 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.

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 an M/M/1 queue with input rate $k\lambda_{\rm r}$ and service rate $\mu_{\rm r}$. Hence, the expectation of the waiting time of each destination is $1/(\frac{\mu}{n-2} - \frac{k\lambda_i}{n-1})$. By Lemma 2, we have that the expected waiting time for the packet to reach all k destinations is $E\{W_{\rm sd}\} = \Theta(\log k/(\frac{\mu}{n-2} - \frac{k\lambda_i}{n-1}))$.

Finally, we achieve the total network delay as

$$\begin{aligned} E\{W_i\} &= E\{W_{\rm s}\} + E\{W_{\rm sd}\} \\ &= \frac{1-\lambda_i}{\mu-\lambda_i} + \Theta\left(\frac{\log k}{\frac{\mu}{n-2} - \frac{k\lambda_i}{(n-1)}}\right). \end{aligned} \tag{9}$$

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

From this 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 are still left 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 a sender and receiver is active in each cell per time slot. 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 needs to send a packet to its k destinations, the more capacity it can achieve. Thus, we assume a packet is delivered directly from a source to one of its destinations via the 1-hop route "Source–destination." Let $X_{\rm s}(T)$ represent the total number of packets transferred over the network from sources to destinations via the 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

$$\frac{X_{\rm s}(T)}{T} \geq nk\lambda - \epsilon. \tag{10}$$

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 1 hop, the total number of packet transmissions in the network during the first T slots is also $X_{\rm s}(T)$. This value must be less than or equal to the total number of transmission opportunities $Y(T)$, and hence

$$X_{\rm s}(T) \leq Y(T) \tag{11}$$

where $Y(T)$ represents the total number of cells containing at least two users in a particular time slot, summed over all time slots $1, 2, \ldots, T$. 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

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

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

Theorem 4: Algorithms permitting at most one transmission in a cell at each time slot that 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 cannot 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 time slot 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 time slot the source has met $k-1$ destinations and at the ith time slot it meets the last one, the probability $W'_{\min} = i$ can thus be written as

$$P\{W'_{\min} = i\} = kp' \left[(1-p')^{i-1} - \binom{k-1}{1} (1-2p')^{i-1} + \binom{k-2}{2} (1-3p')^{i-1} - \cdots \right]. \quad (13)$$

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 $i-1$ time slots. 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 is attached, and similarly we have the following terms. Hence, the expectation of $E\{W'_{\min}\}$ is

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

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

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

Finally, noticing 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 time slot, if there is more than one destination in a same cell as the source, only one destination could be selected as the receiver, and 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 [1] and [19]. 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

Here, we obtain lower bound of delay if only one transmission from a sender to a receiver is permitted in a cell in the following theorem.

Theorem 5: There is no 2-hop algorithm with redundancy that 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. 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 *multidestination reception* style. Note that our assumption differs from the *multiuser reception* ([1]) in that usually each cell is permitted to have a single reception, except there is more than one intended destination within a the cell, while [1] 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 multidestination 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 of the limitation of 2–hop transmission, a new relay can only be generated by the source. Hence, every time

[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.

slot, at most one node can be a new relay. Thus, we have for all $t \geq 1$

$$K_t \leq t. \tag{15}$$

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 multidestination reception style, the events in which different destinations meet the source or a relay node every time slot 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 $[1 - (1 - \frac{1}{c})^{tK_t}]^k$. We thus have

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

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

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

Thus

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

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

B. Scheduling Scheme

In Section IV-A, we considered the minimum delay of the network if we implement redundant packets transmissions. Here, 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 an 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 as follows.

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 (possibly 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 takes 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 that has a 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 *submemoryless* property of the random variable T_N [1], 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 that do not have the packet. Let G represent the event that every time slot 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

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

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 [1, Lemma 6], 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 time slot during the interval T_1, the source delivers a duplicate packet to a new node with probability of at least ϕ, where

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

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 time slot if we enable multiuser reception. However, in the worst case, $\sqrt{n \log k}$ of these times are required, so 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 time slot in which there are at least $\sqrt{n \log k}$ nodes that possess the duplicates of the packet, and note that event H is already a certainty with a probability of 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 that possess the duplicates of the packet in every time slot, θ_0 represents the probability that there is at least one other node in the same cell as the destination ($\theta_0 = 1 - (1 - \frac{1}{c})^{n-1} \rightarrow 1 - e^{-d}$), 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 that 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 time slot, 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 [1, Lemma 2], we bound the total network delay $E\{W\} = O(\sqrt{n \log k})$ and obtain that the achievable per-node capacity under this algorithm is $\Omega(1/k\sqrt{n \log k})$. ■

V. Fundamental Delay and Capacity Tradeoff

In Sections III and IV, we presented 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 multihop algorithm in [1] 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 the multihop relay algorithm with redundancy, can be summarized as in Table I.

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 to the multicast capacity of static networks developed in [3], 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 to the results of unicast in [1], 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 to distribute a packet to k destinations during MotionCast. Particularly, if $k = \Theta(1)$, we find the results of unicast are 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 that the fundamental tradeoff for unicast established in [1] 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 I, we see that the delay–capacity ratio under these three schemes are $\Theta(nk \log k)$, $O(nk \log k)$, and $O(n(\log n)^2)$ respectively, which leads 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 that 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 $\bar{W}$. 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 $\bar{W}$ is given by

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

which equals the following expression:

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

Proof: Suppose the input rate of each of the n sessions is λ, and there exists some stabilizing scheduling strategy that ensures a delay of $\bar{W}$. 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

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

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 nondestination receivers, the count will be added by M. We define $\overline{R_i}$ as the *nondestination redundancy* that represents the final number of counting when the packet finally reaches the kth destination and ends its task, averaged over all packets from session i. That is, $\overline{R_i}$ is the average number of nondestination 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 one 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 that enters the network from session i. This packet has an average delay of $\overline{W_i}$ and an average nondestination redundancy of $\overline{R_i}$. Let random variables W_i and R_i represent the actual delay and nondestination 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 mth 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 $\phi = 1 - (1 - \frac{1}{c})^{2\overline{R_i}}$. 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 $\mathbb{E}\{W_i | R_i \leq 2\overline{R_i}\} = \mathbb{E}\{W_i^{\text{rest}} | R_i \leq 2\overline{R_i}\}$.

Finally, we introduce the last, more easily calculated continuous variable $\widetilde{Z}$, which is also the maximum of several ones $-\widetilde{Z} = \max\{\widetilde{Z}_1, \widetilde{Z}_2, \ldots, \widetilde{Z}_k\}$. 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 $\gamma = \log \frac{1}{1-\phi}$.

Now, we put the relationship among these three variables clearly as follows[9]:

$$W_i^{\text{rest}} \succeq Z \succeq \widetilde{Z}. \tag{24}$$

Furthermore, although Z and $\widetilde{Z}$ defined in our paper are a little different from those defined in [1], i.e., $Z = \max\{Z_1, Z_2, \ldots, Z_K\}$ and $\widetilde{Z} = \max\{\widetilde{Z_1}, \widetilde{Z_2}, \ldots, \widetilde{Z_k}\}$, they also follow *claim 1* and *claim 2* in [1]. Thus, we have the following 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 $\widetilde{Z}$) that occur with probability greater than or equal to $1/2$.

Until now, we have to calculate the last value $\inf_{\widetilde{\Theta}} \mathbb{E}\{\widetilde{Z} | \widetilde{\Theta}\}$. The result of [1, Lemma 8] has been put as follows:

For any nonnegative random variable X, we have

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

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: $w = -\frac{1}{\gamma} \ln[1 - (\frac{1}{2})^{\frac{1}{k}}]$.

[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 [20], we have that Z is stochastically greater than $\widetilde{Z}$.

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

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

Noting 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 $X(1) = -\frac{1}{\gamma}[e^{-w\gamma}-1]$

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

wherein $\frac{1}{i}\binom{k-1}{i-1} = \frac{1}{k}\binom{k}{i}$. Continuing this recursion, we have that

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

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

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

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

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

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 (22) and (33), we have

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

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), and we get $\frac{\bar{W}}{\lambda} \ge \Theta(n\log k)\frac{1}{1-k\lambda}$. Since $\lambda = o(1/k)$, i.e., $1-k\lambda$ remains a constant as n and k grow into infinity, we get

$$\frac{\bar{W}}{\lambda} \ge \Theta(n\log k). \tag{35}$$

Finally, we have the following corollary.

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

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 $k = o(n)$. 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 multihop transmission schemes and the effect of different mobility patterns yet, which could be a future work.

Moreover, the results in this paper are derived theoretically, and it would be an interesting work to validate the results in real experimental or simulation examinations. Also, this is a challenging work since the network is a large-scale one and all the nodes in the network keep on moving. Since there has been some work in the area of DTN trying to implement experiments or simulate situations and some excited results are obtained, it would be very interesting and promising to investigate the capacity and delay under a realistic circumstance. This could be our future work as well.

APPENDIX I
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 source–destination pair. 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 a 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 $\binom{k}{i}=\binom{k-1}{i}+\binom{k-1}{i-1}$, and 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

$$\begin{aligned}A(k)&=A(1)+\sum_{i=2}^{k}[A(k)-A(k-1)]\\&=1+\sum_{i=2}^{k}\frac{1}{k}=\sum_{i=1}^{k}\frac{1}{k}.\end{aligned} \quad (37)$$

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

Lemma 2: Suppose $X_1,X_2,\ldots,X_k$ are continuous i.i.d. exponential variables with expectation of $1/a$, and denote $X_{\max}=\max\{X_1,X_2,\ldots,X_k\}$, 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 cumulative distribution function (cdf) of $X_{\max}$

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

Thus, the probability density function (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

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

According to Lemma 1, we conclude this lemma. ■

ACKNOWLEDGMENT

The authors deeply appreciate the valuable and constructive comments from the anonymous reviewers and the editor.

REFERENCES

[1] 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.

[2] X. Lin and N. B. Shroff, "The fundamental capacity–delay tradeoff in large mobile wireless networks," Tech. Rep., 2004 [Online]. Available: http://cobweb.ecn.purdue.edu/~linx/papers.html

[3] X. Li, "Multicast capacity of wireless ad hoc networks," *IEEE/ACM Trans. Netw.*, vol. 17, no. 3, pp. 950–961, Jun. 2009.

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

[5] A. Keshavarz-Haddad, V. Ribeiro, and R. Riedi, "Broadcast capacity in multihop wireless networks," in *Proc. ACM MobiCom*, Sep. 2006, pp. 239–250.

[6] X. Li, Y. Liu, S. Li, and S. Tang, "Multicast capacity of wireless ad hoc networks under Gaussian channel model," *IEEE/ACM Trans. Netw.*, vol. 18, no. 4, pp. 1145–1157, Aug. 2010.

[7] P. Jacquet and G. Rodolakis, "Multicast scaling properties in massively dense ad hoc networks," in *Proc. Int. Conf. Parallel Distrib. Syst.*, Jul. 2005, pp. 93–99.

[8] S. Shakkottai, X. Liu, and R. Srikant, "The multicast capacity of large multihop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 5, pp. 1691–1700, Oct. 2010.

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

[10] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Throughput-delay trade-off in wireless networks," in *Proc. IEEE INFOCOM*, Mar. 2004, vol. 1, pp. 464–475.

[11] M. Ibrahim, A. A. Hanbali, and P. Nain, "Delay and resource analysis in MANETs in presence of throwboxes," in *Proc. Int. Symp. Comput. Perform., Model., Meas., Eval.*, 2007, pp. 933–947.

[12] N. Banerjee, M. Corner, D. Towsley, and B. Levine, "Relays, base stations, and meshes: Enhancing mobile networks with infrastructure," in *Proc. ACM MobiCom*, Sep. 2008, pp. 81–91.

[13] M. Garetto, P. Giaccone, and E. Leonardi, "Capacity scaling in ad hoc networks with heterogeneous mobile nodes: The super-critical regime," *IEEE/ACM Trans. Netw.*, vol. 17, no. 5, pp. 1522–1535, Oct. 2009.

[14] M. Garetto, P. Giaccone, and E. Leonardi, "Capacity scaling in ad hoc networks with heterogeneous mobile nodes: The subcritical regime," *IEEE/ACM Trans. Netw.*, vol. 17, no. 6, pp. 1888–1901, Dec. 2009.

[15] L. Ying, S. Yang, and R. Srikant, "Optimal delay-throughput trade-offs in mobile ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 4119–4143, Sep. 2008.

[16] S. Toumpis and A. J. Goldsmith, "Large wireless networks under fading, mobility, and delay constraints," in *Proc. IEEE INFOCOM*, Mar. 2004, vol. 1, pp. 609–619.

[17] M. Li and Y. Liu, "Rendered path: Range-free localization in anisotropic sensor networks with holes," in *Proc. ACM MobiCom*, Sep. 2007, pp. 51–62.

[18] R. L. Cruz and A. V. Santhanam, "Hierarchical link scheduling and power control in multihop wireless networks," in *Proc. 40th Annu. Allerton Conf. Commun., Control, Comput.*, Oct. 2002.

[19] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: The multi-copy case," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 77–90, Feb. 2008.

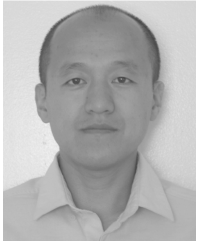
[20] S. M. Ross, *Stochastic Processes*. New York: Wiley, 1996.

[21] C. Hu, X. Wang, and F. Wu, "MotionCast: On the capacity and delay tradeoffs," in *Proc. ACM MobiHoc*, New Orleans, LA, May 2009, pp. 289–298.

[22] X. Li, Y. Liu, S. Li, and S. Tang, "Multicast capacity of wireless ad hoc networks under Gaussian channel model," *IEEE/ACM Trans. Netw.*, vol. 18, no. 4, pp. 1145–1157, Aug. 2010.

[23] C. Wang, X. Li, C. Jiang, S. Tang, and Y. Liu, "Scaling laws on multicast capacity of large scale wireless networks," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 2009, pp. 1863–1871.

[24] M. Garetto and E. Leonardi, "Restricted mobility improves delay-throughput tradeoffs in mobile ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5016–5029, Oct. 2010.



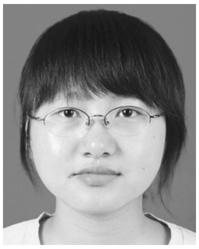
Xinbing Wang (M'06) received the B.S. degree in automation (with honors) from Shanghai Jiao Tong University, Shanghai, China, in 1998, the M.S. degree in computer science and technology from Tsinghua University, Beijing, China, in 2001, and the Ph.D. degree with a major in electrical and computer engineering and minor in mathematics from North Carolina State University, Raleigh, in 2006.

Currently, he is a faculty member with the Department of Electronic Engineering, Shanghai Jiao Tong University. His research interests include resource allocation and management in mobile and wireless networks, TCP asymptotics analysis, wireless capacity, cross-layer call admission control, asymptotics analysis of hybrid systems, and congestion control over wireless ad hoc and sensor networks.

Dr. Wang has been a member of the Technical Program Committee of several conferences including IEEE INFOCOM 2009–2011, IEEE ICC 2007–2011, and IEEE GLOBECOM 2007–2011.

Wentao Huang received the B.S. degree in information science from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2008, and is currently pursuing the M.S. degree in electronic engineering at Shanghai Jiao Tong University, Shanghai, China.

His current research interests include distributed systems, mobile computing, and network security.



Shangxing Wang received the B.S. degree in telecommunication engineering from Xidian University, Xi'an, China, in 2010, and is currently pursuing the Ph.D. degree in electronic engineering, at Shanghai Jiao Tong University, Shanghai, China.

Her current research interests include networking, ad hoc networks, and wireless communication.

Jinbei Zhang received the B.S. degree in electronic engineering from Xidian University, Xi'an, China, in 2010, and is currently pursuing the Ph.D. degree in electronic engineering at Shanghai Jiao Tong University, Shanghai, China.

His current research interests include network capacity, scheduling, and connectivity.

Chenhui Hu received the B.S. and M.S. degrees in electrionic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2007 and 2010, respectively, and is currently pursuing the Ph.D. degree in engineering and applied sciences at Harvard University, Cambridge, MA.

From 2007 to 2010, he was performing research with the Institute of Wireless Communication Technology (IWCT), Shanghai Jiao Tong University. His research interests include wireless capacity and connectivity, asymptotic analysis of mobile ad hoc networks, multicast, distributed MIMO, and percolation theory.