MotionCast: On the Capacity and Delay Tradeoffs

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

Feng Wu
Microsoft Research Asia
Beijing, China
fengwu@microsoft.com

ABSTRACT

In this paper, we define multicast for ad hoc network through nodes' mobility as *MotionCast*, and study the capacity and delay 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 capacity and delay 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 capacity and delay 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 [3] 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 is smaller than that of directly extending the fundamental tradeoff for unicast established in [1] to multicast, i.e., $O(nk^2)$.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communications*

General Terms

Theory and Algorithms

Keywords

Multicast, Capacity, Delay, Scaling Law

1. INTRODUCTION

Multicast in MANETs is predominant in many practical situations. For example, group communications in military networks and disaster alarming in sensor networks. Another example is the current mobile multimedia services. It is likely that there is a large number of mobile users accessing the services, while they favor different programmes provided by a variety of service stations. These stations are then required to multicast their data stream to certain groups of users.

Since certain links may be shared by several destinations, one potential benefit of multicast is that it can reduce the total bandwidth required to communicate with all the destinations. Thus, compared with multiple unicast capacity gain can be obtained by using multicast. Li et al. in [3] study the capacity of a static random wireless ad hoc network for multicast where each node sends packets to $k-1$ destinations. Under protocol interference model, they show that the per-node multicast capacity is $\Theta(\frac{1}{\sqrt{n \log n}\sqrt{k}})$ when $k = O(\frac{n}{\log n})$; the per-node multicast capacity is $\Theta(\frac{1}{n})$ when $k = \Omega(\frac{n}{\log n})$. These results generalize the previous capacity bounds on unicast by Gupta and Kumar [4] and broadcast [5]. Other works falling into this class can be seen in [6] and [7]. Jacquet et al. [6] 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. [7] propose a comb-based architecture for multicast routing which achieves the upper bound for capacity in an order sense.

While the above studies are all based on static networks, the effect of mobility on the capacity of wireless ad hoc networks has been first explicitly developed in [8], where Grossglauser and Tse demonstrate that per-node unicast capacity does not vanish as the size of the network grows by implementing a 2-hop relay algorithm. Note that the price of this improving capacity is the increased delay. It has been shown in [1] [9] that the 2-hop relay algorithm in [8] yields a tremendous average delay of $\Omega(n)$.

The relationships between capacity and delay are further investigated in the literature of [1] [2] [10] [11]. In the work by Neely and Modiano [1], the authors present a strategy utilizing redundant packets transmissions along multiple paths in a cell partitioned MANET to reduce delay with a sacrifice on the capacity. They establish the following necessary tradeoff: $delay/capacity \geq O(n)$, and develop schemes that can achieve $\Theta(1), \Theta(1/\sqrt{n})$ and $\Theta(1/(n \log n))$ per-node capacity, when the delay constraint is on the order

MobiHoc'09, May 18–21, 2009, New Orleans, Louisiana, USA.

of $\Theta(n), \Theta(\sqrt{n})$ and $\Theta(\log n)$, respectively. In [11], Toumpis and Goldsimth construct a 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 $O(n^d)$, which turns out to be better than that in [1]. Afterwards, Lin and Shroff [2] search the optimal capacity-delay tradeoff and identify the limiting factors of the existing scheduling schemes in MANETs. Recently, Ying et al. [10] develop joint coding-scheduling algorithms to achieve the optimal delay-throughput tradeoffs.

A key feature of multicast in MANETs is that packets can be delivered via nodes' mobility, thus we refer it as MotionCast in this paper. Intuitively, capacity and delay tradeoffs still exist for MotionCast, but being more complicated than the situations for an unicast scenario. Packets can also be delivered through the mobility of the relay nodes, thus a higher per-node multicast capacity compared with that of a static wireless ad hoc network is imaginable. However, the scheduling design becomes harder because of the permanent change of the network topology and now that more destinations need obtain packets from the source, it will take a longer time to complete a multicast process, which suggests a larger delay. Hence, some challenging questions raised naturally in this context are as follows:

- What is the maximum per-node MotionCast capacity?
- How long will be the induced delay to achieve this capacity and what is the minimum delay?
- How does the capacity and delay tradeoffs emerge for MotionCast?

Answering these questions is helpful for us to evaluate the performance and better understand the fundamental tradeoffs for multicast in large-scale ad hoc networks with mobility.

In our work, we conduct a study on the scaling behaviors in a cell partitioned MANET under multicast traffic pattern. To start with, we propose a 2-hop relay algorithm without redundancy. This algorithm is a generalized version of the algorithm presented in [1], which corresponds to a decoupled queuing model. The variation is that $k-1$ more destinations are associated with a source, and delay for a packet will be determined by the time when it is delivered to all the destinations. As for a specific packet, we clearly divide nodes other than the source into relays and destinations (referred to as an *non-cooperative mode*) first. 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 do the job itself. The expression of MotionCast capacity and delay are calculated under this model, and it turns out that capacity will degenerate when k is large.

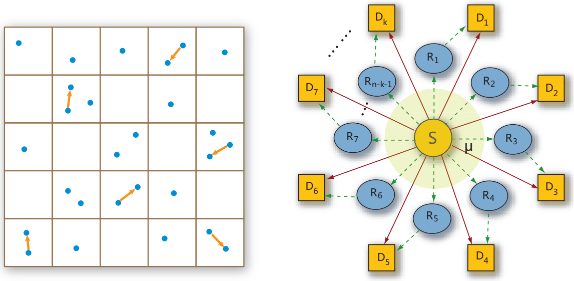
Then, we loose the constraints within our initial model by permitting information spread among destinations (called a *cooperative mode*). At this moment, we do not discriminate destinations and the remnant nodes except the source rigorously. We define the first node a source meets as the "*designated 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 relates to a special packet in the 2-hop relay algorithm, thus after a relay is designated current destinations will merely act as receivers for the packet and do not help transmit the packet to other destinations.

Next, we employ redundant packets transmissions to reduce the delay. In a 2-hop relay strategy with redundancy, a source sends a packet to many distinct 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 expect delay in the network is no less than $\Omega(\sqrt{n \log k})$. Besides, delay of $O(\sqrt{n \log k})$ is attainable in a proposed scheme 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 the average delay of $\Theta(\sqrt{n \log k})$. Notice that the per-node capacity is better than the results of static multicast scenario in [3] as long as k is strictly less than n in an order sense, i.e., $k = O(n^\epsilon)$ $(0 \le \epsilon < 1)$. 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, capacity and delay 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.

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. Finally, we conclude in Section VI.

2. NETWORK MODEL



(a) Network model. (b) Traffic pattern

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

Cell Partitioned Network Model: The system model is based on the cell partitioned network model exploited in [1] and [13]. 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 are within a same cell (to locate the nodes, please refer to [12] 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 $\Theta(1)$.

Mobility Model: Dividing time into constant duration slots, we adopt the following ideal i.i.d. mobility model. 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. We do not account the time a node moves from the existing cell to a new one, hence this model captures the characteristic of the infinite 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. Numbering all the nodes from 1 to n, we assume each node i is a source node associated with $k = k(n)$ 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

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

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..

3. CAPACITY AND DELAY IN THE 2-HOP RELAY ALGORITHM WITHOUT REDUNDANCY

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

3.1 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.

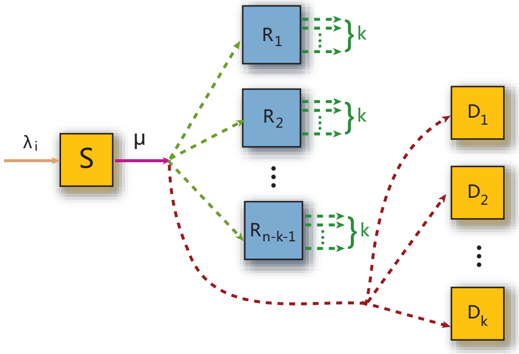


Figure 2: A decoupled queuing model of the network as seen by the packets transmitted from a single source to multiple destinations.

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[2], 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:

    - 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.

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

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

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

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

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

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 $k = O(n^{\epsilon})$ $(0 \le \epsilon < 1)$, $q \to 0$; else if $k = \Theta(n)$, $q \to 1 - (d+1)e^{-d}$. 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 $k = O(n^{\epsilon})$ $(0 \le \epsilon < 1)$, 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. In every timeslot, a new packet arrives with probability λ_i at source i, and with probability μ 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.

Denote r_1 for the rate at which the source is scheduled to transmit directly to one of the destinations, and r_2 for the rate at which it is scheduled to transmit to one of its relays. 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

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

It follows from (5) 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, the source can be viewed as a Bernoulli/Bernoulli queue with input rate λ_i and service rate μ , having an expected number of occupancy packets given by $\bar{L}_s = \frac{\rho(1-\lambda_i)}{1-\rho}$, where $\rho \overset{\Delta}{=} \frac{\lambda_i}{\mu}$. 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.

A given packet from this output process is transmitted to the first relay node with probability $\frac{r_2}{\mu(n-k-1)}$ (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). Hence, every timeslot, this relay independently receives a packet with probability $\lambda_r = \frac{\lambda_i r_2}{\mu(n-k-1)}$. On the other hand, the relay node is scheduled for a potential packet transmission to a destination node with probability $\mu_r = \frac{r_2}{n-2}$ (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. However, different from unicast, each relay node is in charge of sending a same packet to k distinct destinations in the multicast scenario abided by the algorithm.

From a more delicate point of view, we model a relay node as $n-2$ parallel sub-queues (each of them buffers the packets intended for a certain destination), shown in Figure 3. Then, when a new relay packet arrives at the relay, it will "copy" this packet into k "virtual-duplicates" and add them into respective sub-queues associated with the k destinations. Hence for unicast, the incoming rate of each sub-queue is λ_r, while for multicast it is k times of that quantity. It follows that the discrete time Markov chain for queue occupancy in each sub-queue 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 $E\{W_{rd}^i\} = 1/(\mu_r - k\lambda_r)$.

The resulting waiting time W_{rd} for multicast is determined by the maximum value among all the waiting times $W_{rd}^1, W_{rd}^2, ..., W_{rd}^k$ of the above queues. Observing that these waiting times are i.i.d exponential variables, by Lemma 2 (see the proof in Appendix II), we obtain that $E\{W_{rd}\} = \log k/(\mu_r - k\lambda_r)$. 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 group of k parallel M/M/1 sub-queues corresponding to its k destinations similarly. The source will "copy" this packet into $k-1$ "virtual-duplicates" and add them into respective sub-queues associated with the remnant $k-1$ destinations. Since the probability that the source need send packets directly to destinations is $\frac{r_1}{\mu}$, the incoming data rate is thus $\frac{(k-1)\lambda_i r_1}{\mu}$ for each sub-queue. Mean-

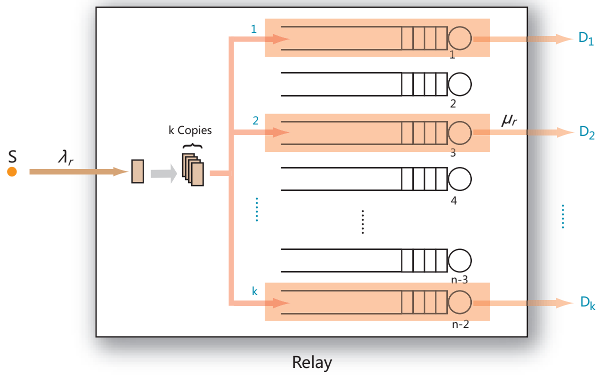


Figure 3: A more delicate view of a relay. Each of a relay node can be modeled as $n-2$ parallel sub-queues buffering packets intended for different destinations. In special, k sub-queues associated with k destinations of the current source are depicted in red in the figure.

while, the service rate at each sub-queue equals to the transmission rate $\frac{r_1}{k}$ between a source-destination pair. Hence, the expectation of the waiting time at each sub-queue is $1/(\frac{r_1}{k} - \frac{(k-1)\lambda_i r_1}{\mu})$. And by Lemma 2, we have the expect waiting time for the packet to reach all $k-1$ remnant destinations is $E\{W_{sd}\} = \log(k-1)/(\frac{r_1}{k} - \frac{(k-1)\lambda_i r_1}{\mu})$.

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

$$\begin{aligned} E\{W_i\} &= \frac{r_1}{\mu}\big(E\{W_s\}+E\{W_{sd}\}\big)+\frac{r_2}{\mu}\big(E\{W_s\}+E\{W_{rd}\}\big) \\ &= \frac{r_2}{\mu}\left(\frac{1-\lambda_i}{\mu-\lambda_i}+\frac{\log k}{\frac{r_2}{n-2}-\frac{k\lambda_i r_2}{\mu(n-k-1)}}\right) \\ &\quad +\frac{r_1}{\mu}\left(\frac{1-\lambda_i}{\mu-\lambda_i}+\frac{\log(k-1)}{\frac{r_1}{k}-\frac{(k-1)\lambda_i r_1}{\mu}}\right) \qquad (7) \end{aligned}$$

Looking upon the asymptotic behaviors of the network delay when $k, n \to \infty$, we have

- If $k = O(n^\epsilon)$ $(0 \leq \epsilon < 1)$, then it follows $r_1 \to 0$ and $r_2, \mu \to \frac{1-(d+1)e^{-d}}{2d}$, which means almost all the traffic is carried along the path of "source-relay-destinations". To ensure the stability of the network, the incoming rate should be less than the service rate at any stage of the network. Thus,

$$\begin{cases} \mu - \lambda_i > 0 \\ \frac{r_2}{n-2} - \frac{k\lambda_i r_2}{\mu(n-k-1)} > 0 \end{cases}$$

  i.e., $\lambda_i < \frac{(n-k-1)\mu}{(n-2)k} \to \mu/k$ $(n \to \infty)$. Besides, the total network delay is governed by the first term of (7), which is on the order of $O(n \log k)$ for a fixed traffic loading value $\rho_r = \frac{k\lambda_i(n-2)}{\mu(n-k-1)}$ at each relay.

- If $k = \Theta(n)$, then it follows $r_1, \mu \to \frac{1-(d+1)e^{-d}}{d}$ and $r_2 \to 0$, which means nearly all the packets are delivered directly from source to destinations. Likewise, the network capacity is limited by $\frac{r_1}{k} - \frac{(k-1)\lambda_i r_1}{\mu} > 0$, i.e., $\lambda_i < \frac{\mu}{k(k-1)}$. Besides, the total network delay is governed by the second term of (7), which scales as $O(k \log k) = O(n \log k)$ for a fixed traffic loading value $\rho_s = \frac{k(k-1)\lambda_i}{\mu}$ at the source.

From the first case of the above discussion, we conclude the theorem. □

3.2 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 $O(n \log k)$, when $k = O(n^\epsilon)(0 \leq \epsilon < 1)$. However, if $k = \Theta(n)$ we find that with the same amount of delay in an order sense, the per-node capacity decreases to $\Omega(1/k^2)$. To avoid this degeneration, 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 per-node capacity $\Omega(1/k)$ with delay $O(n \log k)$ for any $k \leq n$, and 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". 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 the performance in an order sense, we omit this difference between these two cases. Thus, following the same analytical steps as Theorem 1 when k is strictly less than n in an order sense, 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 $k \leq n$.*

Since the second algorithm is better 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.

3.3 Maximum capacity and minimum delay

Although we have constructed the achievable capacity and delay 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 $O(1/k)$ if only a pair of sender and receiver is active in each cell per timeslot.*

Proof. We use hop argument to prove this result. Consider the minimum number of hops h_{min} that a source can send a packet to all k destinations. Although nodes are mobile, we can treat the process of the transmissions as a dynamic graph. Specifically, we connect an edge along each two nodes if the packet is transmitted between them. To form a connected graph among the source and its destinations, the minimum number of edges is equal to $(k+1)-1=k$, which means that the graph is a tree. Thus, we get $h_{min}=k$.

Denote the input rate at each source by λ, then the number of bits arriving at these n nodes in an interval $[0,T]$ is λTn. Thus, the total number of transmissions of all bits to their destinations is at least λTnh_{min}. On the other hand, the total number of possible transmissions at any timeslot is upper bounded by that of the cells containing at least two users, which is no more than c. Hence,

$$\lambda Tnk \le Tc \tag{8}$$

i.e., $\lambda \le \frac{1}{dk}$. Notice that $d=\Theta(1)$, thus we have $\lambda = O(1/k)$. □

Theorem 4. *Algorithm permitting at most one transmission in a cell at each timeslot, which do not use redundancy cannot achieve an average delay of $O(n\log k)$.*

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. 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

$$\begin{aligned} P\{W'_{min}=i\} \;=\; & kp'\Big[(1-p')^{i-1}-\binom{k-1}{1}(1-2p')^{i-1} \\ & +\binom{k-2}{2}(1-3p')^{i-1}-\cdots\Big] \end{aligned} \tag{9}$$

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

$$\begin{aligned} & E\{W'_{min}\} \\ &= kp'\sum_{i=1}^{+\infty} i\Big[(1-p')^{i-1}-\binom{k-1}{1}(1-2p')^{i-1} \\ &\quad +\binom{k-2}{2}(1-3p')^{i-1}-\cdots\Big] \\ &= kp'\Big[\sum_{i=1}^{+\infty} i(1-p')^{i-1} \\ &\quad -\binom{k-1}{1}\sum_{i=1}^{+\infty} i(1-2p')^{i-1} \\ &\quad +\binom{k-1}{2}\sum_{i=1}^{+\infty} i(1-3p')^{i-1}-\cdots\Big] \\ &= kp'\Big[\frac{1}{p'^2}-\binom{k-1}{1}\frac{1}{(2p')^2}+\binom{k-1}{2}\frac{1}{(3p')^2}-\cdots\Big] \\ &= \frac{k}{p'}\Big[1-\frac{1}{2^2}\binom{k-1}{1}+\frac{1}{3^2}\binom{k-1}{2}-\cdots\Big] \\ &= \frac{\log k}{p'} \end{aligned} \tag{10}$$

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

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

Finally, notice that $1/p'=\Theta(n)$, we obtain that $E\{W'_{min}\} = \Theta(n\log k)$. 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 capacity and delay achieved by the 2-hop relay algorithm without redundancy, we find the exact order of the capacity and delay are $\Theta(1/k)$ and $\Theta(n\log k)$, respectively.

4. CAPACITY AND DELAY 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 [14]. 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.

4.1 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 proof 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* ([1]) 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 [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 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 intermedia relays (including the source) at the beginning of slot t. We have that every timeslot the increase of the relays is merely due to the source sends the packet to a new relay. Thus, we have for all $t \geq 1$:

$$K_t \leq t \tag{11}$$

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 $1-(1-\frac{1}{c})^{tK_t}$. We thus have

$$\begin{aligned} P\{T'_N > t\} &\geq 1-\big[1-(1-\frac{1}{c})^{tK_t}\big]^k \\ &\geq 1-\big[1-(1-\frac{d}{n})^{t^2}\big]^k \\ &= 1-(1-e^{-\frac{d}{n}t^2})^k \quad (n \to \infty) \end{aligned} \tag{12}$$

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

$$\begin{aligned} P\{T'_N > t\} &\geq 1-(1-e^{-\log k})^k \\ &= 1-(1-\frac{1}{k})^k \\ &= 1-e^{-1} \end{aligned} \tag{13}$$

Thus:

$$\begin{aligned} E\{T_N\} \geq E\{T'_N\} &\geq E\{T'_N \mid T'_N > t\}P\{T'_N > t\} \\ &\geq (1-e^{-1})\sqrt{n \log k/d} \end{aligned} \tag{14}$$

as $k, n \to \infty$. From (14), we prove the theorem. □

4.2 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.

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 ([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 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, and hence every timeslot the probability that at least one of these nodes visits the cell of the source 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: 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 [1], 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 $E\{T_2\}$, note that every timeslot in which there are at least $\sqrt{n\log k}$ nodes possess the duplicates of the packet, 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: 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 [1], we bound the total network delay $E\{W\} = O(\sqrt{n\log k})$, and obtain the achievable per-node capacity under this algorithm is $\Omega(1/(k\sqrt{n\log k}))$ (Note that a new relay packet arriving at a relay will occupy k sub-queues in the model of Fig. 3 until it reaches all k destinations, thus the capacity should be divided by a factor k in the expression.). □

5. DISCUSSION

In Section 3 and Section 4, we present algorithms both without and with redundancy to fulfill the task of MotionCast. In this section, we draw a comparison of the capacity and delay with the former results and discuss the capacity and delay tradeoffs obtained in this paper.

The capacity and delay tradeoffs between the 2-hop relay algorithm without and with redundancy can be summarized in the following table.

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

Compared with the multicast capacity of static networks developed in [3], we find that capacity of the 2-hop relay algorithm without redundancy is better when $k = O(n^{\epsilon})$ $(0 \leq \epsilon < 1)$; otherwise, capacity remains the same as that of static networks, i.e., mobility cannot increase capacity. However, capacity of the 2-hop relay algorithm with redundancy is no better than that of static networks if $k\log k = \Omega(\log n)$ due to the redundant packets transmissions. Moreover, compared with 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 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. 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 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 [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.

6. CONCLUSION AND FUTURE WORK

In this paper, we study capacity and delay 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 $k = O(n^{\epsilon})$ $(0 \leq \epsilon < 1)$. And our tradeoff is better than that of directly extending the tradeoff for unicast to multicast. 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.

7. ACKNOWLEDGMENT

The authors wish to thank Jianzhou Feng, Wentao Huang and other group members for the helpful discussions. This work is supported by NSF China (No. 60702046, 60832005); China Ministry of Education (No. 20070248095); Shanghai Jiaotong University Young Faculty Funding (No. 06ZBX800050); Qualcomm Research Grant; China International Science and Technology Cooperation Programm (No. 2008DFA11630); PUJIANG Talents (08PJ14067); Shanghai Innovation Key Project (08511500400).

8. REFERENCES

[1] 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.

[2] 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

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

[4] 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.

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

[6] 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.

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

[8] 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.

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

[10] 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.

[11] S. Toumpis and A. J. Goldsmith,"Large wireless networks under fading, mobility, and delay constraints," in *Proceedings of IEEE INFOCOM*, Mar. 2004.

[12] M. Li and Y. Liu, "Rendered path: range-free localization in anisotropic sensor networks with holes," in *Proceedings of ACM MobiCom*, Sept. 2007.

[13] 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.

[14] 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.

[15] S. M. Ross, *Stochastic processes.* New York: John Wiley & Sons, 1996.

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 directly calculating the probability can hardly obtain an integrate expression, we temporarily adopt the following assumptions here. Suppose the number of nodes n is divisible by $k+1$. For simplicity, 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. 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. $\sum\limits_{i=1}^{k} \frac{(-1)^{i-1}}{i}\binom{k}{i} = \ln(k+1) + r$, *where* $k \geq 1$ *and* r *is Euler constant.*

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

$$\begin{aligned} A(k) - A(k-1) &= \sum_{i=1}^{k} \frac{(-1)^{i-1}}{i}\binom{k-1}{i-1} \\ &= \frac{1}{k}\sum_{i=1}^{k}(-1)^{i-1}\binom{k}{i} \qquad (15) \end{aligned}$$

Recall that $(1-1)^k = \sum\limits_{i=0}^{k}(-1)^i\binom{k}{i} = 0$, hence we obtain $\sum\limits_{i=1}^{k}(-1)^{i-1}\binom{k}{i} = -\sum\limits_{i=1}^{k}(-1)^i\binom{k}{i} = -\big[\sum\limits_{i=0}^{k}(-1)^i\binom{k}{i} - 1\big] = 1$. Combining with (15), we get $A(k) - A(k-1) = \frac{1}{k}$, then

$$\begin{aligned} A(k) &= A(1) + \sum_{i=2}^{k}\big[A(k) - A(k-1)\big] \\ &= 1 + \sum_{i=2}^{k}\frac{1}{k} = \sum_{i=1}^{k}\frac{1}{k} \qquad (16) \end{aligned}$$

Since the right-hand-side of (16) 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},

$$F_{X_{max}}(t) = P\{X_{max} \leq t\} = (1-e^{-at})^k \qquad (17)$$

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

$$f_{X_{max}}(t) = \frac{dF_{X_{max}}(t)}{dt} = k(1-e^{-at})^{k-1} \cdot ae^{-at} \tag{18}$$

Then, we obtain

$$\begin{aligned} E\{X_{max}\} &= \int_0^\infty k(1-e^{-at})^{k-1} ae^{-at} \cdot tdt \\ &= ka \int_0^\infty \sum_{i=0}^{k-1} \binom{k-1}{i} (-1)^i e^{-a(i+1)t} \cdot tdt \\ &= \sum_{i=0}^{k-1} ka \binom{k-1}{i} (-1)^i \frac{1}{\big[a(i+1)\big]^2} \\ &= \sum_{i=1}^{k} ka \binom{k-1}{i-1} (-1)^{i-1} \frac{1}{a^2 i^2} \\ &= \frac{k}{a} \sum_{i=1}^{k} \frac{(-1)^{i-1}}{i^2} \binom{k-1}{i-1} \\ &= \frac{1}{a} \sum_{i=1}^{k} \frac{(-1)^{i-1}}{i} \binom{k}{i} \end{aligned} \tag{19}$$

According to Lemma 1, we conclude this lemma. □