Capacity and Delay Tradeoffs for Ad-Hoc Mobile Networks

Michael J. Neely
University of Southern California
http://www-rcf.usc.edu/∼mjneely

Eytan Modiano
Massachusetts Inst. of Technology
http://web.mit.edu/modiano/www/

***Abstract*— We consider the throughput/delay tradeoffs for scheduling data transmissions in a mobile ad-hoc network. To reduce delays in the network, each user sends redundant packets along multiple paths to the destination. Assuming the network has a cell partitioned structure and users move according to a simplified iid mobility model, we compute the exact network capacity and the exact end-to-end queueing delay when no redundancy is used. The capacity achieving algorithm is a modified version of the Grossglauser-Tse 2-hop relay algorithm and provides $O(N)$ delay (where N is the number of users). We then show that redundancy cannot increase capacity, but can significantly improve delay. The following necessary tradeoff is established: $delay/rate \geq O(N)$. Two protocols which use redundancy and operate near the boundary of this curve are developed, with delays of $O(\sqrt{N})$ and $O(\log(N))$, respectively. Networks with non-iid mobility are also considered and shown through simulation to closely match the performance of iid systems in the $O(\sqrt{N})$ delay regime.**

I. Introduction

We consider the effects of transmitting redundant packets along independent paths of an ad-hoc wireless network with mobility. Such redundancy improves delay at the cost of increasing overall network congestion. We show that redundancy cannot increase network capacity, but can significantly improve delay performance, yielding delay reductions by several orders of magnitude when data rates are sufficiently less than capacity.

The first part of this paper closely follows our previous work in [2], and several statements from [2] are repeated here for completeness. We use the following *cell partitioned* network model: The network is partitioned into C non-overlapping cells of equal size (see Fig. 1). There are N mobile users independently roaming from cell to cell over the network, and time is slotted so that users remain in their current cells for a timeslot, and potentially move to a new cell at the end of the slot. If two users are within the same cell during a timeslot, one can transfer a single packet to the other. Each cell can support exactly one packet transfer per timeslot, and users within different cells cannot communicate during the slot. Multi-hop packet transfer proceeds as users change cells and exchange data. The cell partitioning reduces scheduling complexity and facilitates analysis. Similar cell partitioning has recently been considered by Cruz et. al in [3].

We consider the following simplified mobility model: Every timeslot, users choose a new cell location independently and identically distributed over all cells in the network. Such a mobility model is of course an over-simplification. Indeed,

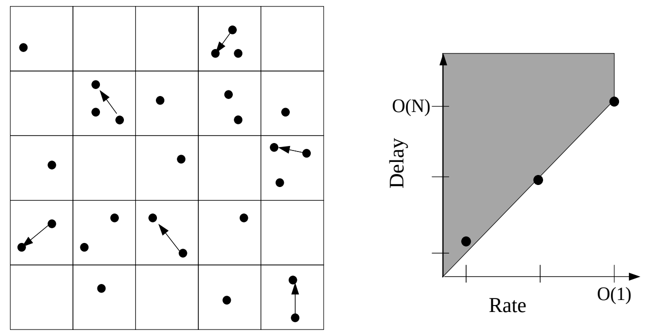

Fig. 1. A cell-partitioned ad-hoc wireless network with C cells and N mobile users.

actual mobility is better described by Markovian dynamics, where users choose new locations every timeslot from the set of cells adjacent to their current cell. However, analysis under the simplified iid mobility model provides a meaningful bound on performance in the limit of *infinite mobility*. With this assumption, the network topology dramatically changes every timeslot, so that network behavior cannot be predicted and fixed routing algorithms cannot be used. Rather, because information about the current and future locations of users is unknown, one must rely on robust scheduling algorithms. Furthermore, it is shown in [1], [4] that the network capacity under an iid mobility model is identical to the capacity region of a network with non-iid mobility with the same steady state distribution. Delay analysis for non-iid mobility is also discussed.

We compute an exact expression for the per-user transmission capacity of the network (for any number of users $N \geq 3$), and show that this capacity cannot be increased by using redundant packet transfers. When no redundancy is used, a modified version of the Grossglauser-Tse 2-hop relay algorithm in [5] is presented and shown to achieve capacity. The queueing delay in the network is explicitly computed and shown to be $O(N)/(\mu - \lambda_i)$ (where μ is the per-user network capacity, and λ_i is the rate at which user i transfers packets intended for its destination). Furthermore, it is shown that no scheduling algorithm can improve upon $O(N)$ delay performance unless redundancy is used. We then consider modifying the 2-hop relay algorithm to allow redundant packet transmissions. It is shown that no scheme which restricts packets to two hops can achieve a better delay than $O(\sqrt{N})$. A scheduling protocol that employs redundant packet transmissions is developed and shown to achieve this delay bound when all users communicate at a reduced data rate of $O(1/\sqrt{N})$. A multi-hop protocol is then developed to achieve $O(\log(N))$ delay by further sacrificing throughput. The necessary condition $delay/rate \geq O(N)$ is established for any routing and scheduling algorithm, and the 2-hop relay algorithms are shown to meet this bound with equality while the multi-hop algorithm deviates from optimality by no more than a logarithmic factor.

Previous work on the capacity of ad-hoc wireless networks is found in [1-11], [13]. Gupta and Kumar present asymptotic results for static networks in [6], [7], where it is shown that per-user network capacity is $O(1/\sqrt{N})$, and hence vanishes as the number of users N increases. The effect of mobility on the capacity of ad-hoc wireless networks was first explicitly developed in [5], where a 2-hop relay algorithm was developed and shown to support constant per-user throughput which does not vanish as the size of the network grows. These works do not consider the associated network *delay*, and analysis of the fundamental queueing delay bounds for general networks remains an important open question.

In [8] it is shown that for a network with a mixture of stationary users and mobile relay nodes, delay can be improved by exploiting velocity information and relaying packets to nodes moving in the direction of their destination. Routing for fully mobile networks using table updates is considered in [9]. Schemes for improving delay via diversity coding and multi-path routing are considered in [10], [11], although this work does not consider delays due to path sharing, queueing, or stochastic arrivals. Delay improvement via redundant packet transfers is considered in [2]. This idea is related to the notion of *content replication* considered for static peer-to-peer systems in [12] and for mobile networks in [13]. Our i.i.d. mobility model is similar to that used in [13], where *mobile infostations* are used to store content for users requesting file access. Throughput and delay tradeoffs were perhaps first considered in [14], where delay of multi-hop routing is reduced by increasing the coverage radius of each transmission, at the expense of reducing the number of simultaneous transmissions the network can support. Similar radial scaling techniques have recently appeared in [15] [16] [17]. While our work was developed prior to the work in [15] [16] [17] and does not directly consider radial scaling, for completeness we include a detailed comparison with these approaches at the end of Section VI.

In this paper, we consider the delay performance offered by cell partitioned wireless networks, extending our previous work in [2] by proving the delay/throughput statements given there. The contributions are threefold: First, we demonstrate network capacity and delay analysis which considers the full effects of queueing, and show that delay grows as $O(N)$ when no redundancy is used. Second, we establish a fundamental delay/rate tradeoff curve that bounds performance of any routing and scheduling algorithm. Third, we develop three different protocols which achieve optimal or near optimal performance in different rate regimes. Protocols for three different delay/rate regimes are developed and shown to operate on or near the boundary of this curve.

Throughout this paper, we assume that the number of cells is of the same order as the number of users, so that the user/cell density d is constrained to be $O(1)$ (independent of N). This is a necessary constraint in cases when the network area is increased while maintaining the same average number of users per unit area and the same transmission power (and hence, transmission radius) for each user. In the opposite case when the network area is fixed but the number of users N grows large (increasing the number of users per unit area), it is possible to consider cell densities that increase with N, although the $d = O(1)$ constraint can still be imposed by appropriately scaling the cell size. Note that in this case, it would be possible to maintain a cell size for which the user/cell density increases to infinity with N. However, this would require the coordination of an increasingly large number of users in each cell, and it would necessarily shrink network capacity to zero with growing N (as shown in the next section). It could, however, provide an alternate means of improving network delay, as described in [15] [16] [17]. Indeed, in the extreme case where there is only one cell containing all nodes, it is clear that any user could reach any other user in just a single hop. A detailed comparison of our results with those of [15] [16] [17] is given in Section V.

In the next section, we establish the capacity of the cell partitioned network and analyze the delay of the capacity achieving relay algorithm. In Section III we develop delay bounds for transmission schemes with redundancy, and in Section IV we provide scheduling protocols which achieve these bounds. In Section V we prove necessity of $delay/rate \geq O(N)$, and show that the given protocols operate on the boundary of this rate-delay tradeoff curve. Simulations and Markovian mobility models are considered in Sections VI and VII.

II. Capacity, Delay, and the 2-Hop Relay Algorithm

Consider a cell partitioned network such as that of Fig. 1. The shape and layout of cell regions is arbitrary, although we assume that cells have identical area, do not overlap, and completely cover the network area. We define:

- N = Number of Mobile Users
- C = Number of Cells
- $d = N/C$ = User/Cell density

Users move independently according to the *full-mobility model*, where the steady state location of each user is uniform over all cells.

Let λ_i represent the exogenous arrival rate of packets to user i (in units of packets/slot). Packets are assumed to arrive as a Bernoulli process, so that with probability λ_i a single packet arrives during the current slot, and otherwise no packet arrives. Other stochastic inputs with the same time average arrival rate can be treated similarly, and the arrival model does not affect the region of rates the network can support [1].

We assume packets from source i must be delivered to a unique destination j. In particular we assume the number of users N is even and consider the one-to-one pairing: $1 \leftrightarrow 2$, $3 \leftrightarrow 4$, ..., $(N-1) \leftrightarrow N$; so that user 1 communicates with user 2 and user 2 communicates with user 1, user 3 communicates with user 4 and user 4 communicates with user 3, and so on. Other source-destination scenarios can be treated similarly [1].

Packets are transmitted and routed through the network according to some scheduling algorithm. The algorithm chooses which packets to transmit on each timeslot without violating the physical constraints of the cell partitioned network or the following additional *causality constraint*: A user cannot transmit a packet that it has never received. Note that once a packet has been received by a user, it can be stored in memory and transmitted again and again if so desired. We assume that packets are equipped with header information so that they can be individually distinguished for scheduling purposes.

A scheduling algorithm is *stable* if the λ_i rates are satisfied for all users so that queues do not grow to infinity and average delays are bounded. Assuming that all users receive packets at the same data rate (so that $\lambda_i = \lambda$ for all i), the *capacity* of the network is the maximum rate λ that the network can stably support. Note that this is a purely network layer notion of capacity, where optimization is over all possible routing and scheduling protocols. Below we compute the network capacity, assuming users change cells in an iid fashion every timeslot. In [1] it is shown that the capacity region depends only on the steady state user location distribution. Hence, any Markovian model of user mobility which in steady state distributes users independently and uniformly over the network yields the same expression for capacity.

Theorem 1: The capacity of the network is:

$$\mu = \frac{p+q}{2d} \tag{1}$$

where

$$p = 1 - \left(1 - \tfrac{1}{C}\right)^N - \tfrac{N}{C}\left(1 - \tfrac{1}{C}\right)^{N-1} \tag{2}$$

$$q = 1 - \left(1 - \tfrac{1}{C^2}\right)^{N/2} \tag{3}$$

and hence the network can stably support users simultaneously communicating at any rate $\lambda < \mu$.

Note that p represents the probability of finding at least two users in a particular cell, and q represents the probability of finding a source-destination pair within a cell.

Proof: The proof of the above theorem involves proving that $\lambda \leq \mu$ is necessary for network stability, and that $\lambda < \mu$ is sufficient. The necessary condition is proven in [2] [1]. Sufficiency is established in the next subsection, where an algorithm with bounded average delay is provided. □

Taking limits as $N \to \infty$, we find the network capacity tends to the fixed value $(1 - e^{-d} - de^{-d})/(2d)$. Hence, for nonzero capacity, the ratio $d = N/C$ should be fixed as both N and C scale up. The optimal user/cell density d^* and the corresponding capacity μ^* are: $d^* = 1.7933$, $\mu^* = 0.1492$. Thus, large cell partitioned networks cannot support more than 0.1492 packets/slot, but can achieve arbitrarily close to this data rate by scaling the number of cells C with N to maintain a constant user/cell density d^*.

This μ^* capacity value is close to the maximum throughput estimate of 0.14 packets/slot for the $O(1)$ throughput strategy given by Grossglauser and Tse in [5], where the 0.14 number is obtained by a numerical optimization over a transmit probability θ. In the Grossglauser-Tse strategy, transmitting users send to their nearest neighbors to obtain a high signal to interference ratio on each transmission. The proximity of their optimal throughput to the value of μ^* suggests that when the transmit probability is optimized, the nearest-neighbor transmission policy behaves similarly to a cell-partitioned network. The same value μ^* arises when users send independent data to a finite collection of other users according to a *rate matrix* (λ_{ij}). In this case, μ^* represents the maximum sum rate into or out of any user provided that no user sends or receives more than any other [1].

We note that the optimal throughput μ of Theorem 1 cannot be improved even if all users have perfect knowledge of future events (see [1]). Thus, control strategies which utilize redundant packet transfers, enable multiple users to overhear the same transmission, or allow for perfect feedback to all users when a given packet has been successfully received, cannot increase capacity.

A. Delay Analysis and the 2-Hop Relay Algorithm

In this section, we consider a modified version of the Grossglauser-Tse relay algorithm of [5], and show the algorithm is capacity achieving with a bounded average delay. The algorithm restricts packets to 2-hop paths, where on the first hop a packet is transmitted to any available user. This user will act as a 'relay' for the packet. The packet is stored in the buffer of the relay until an opportunity arises for it to be transmitted by the relay to its destination. Note that the notion of relaying is vitally important, as it allows throughput to be limited only by the rate at which a source encounters other users, rather than by the rate at which a source encounters its destination.

Cell Partitioned Relay Algorithm: Every timeslot and for each cell containing at least two users:

1) If there exists a source-destination pair within the cell, randomly choose such a pair (uniformly over all such pairs in the cell). If the source contains a new packet intended for that destination, transmit. Else remain idle.
2) If there is no source-destination pair in the cell, designate a random user within the cell as sender. Independently choose another user as receiver among the remaining users within the cell. With equal probability, randomly choose one of the two options:
    - *Send a Relay packet to its Destination:* If the designated transmitter has a packet destined for the designated receiver, send that packet to the receiver. Else remain idle.

- *Send a New Relay Packet:* If the designated transmitter has a new packet (one that has never before been transmitted), relay that packet to the designated receiver. Else remain idle.

Because packets that have already been relayed are restricted from being transmitted to any user other than their destination, the above algorithm restricts all routes to 2-hop paths. The algorithm schedules packet transfer opportunities without considering queue backlog. Performance can be improved by allowing alternative scheduling opportunities in the case when no packet is available for the chosen transmission. However, the randomized nature of the algorithm admits a nice *decoupling* between sessions (see Fig. 2), where individual users see the network only as a source, destination, and intermediate relays, and transmissions of packets for other sources are reflected simply as random ON/OFF service opportunities.

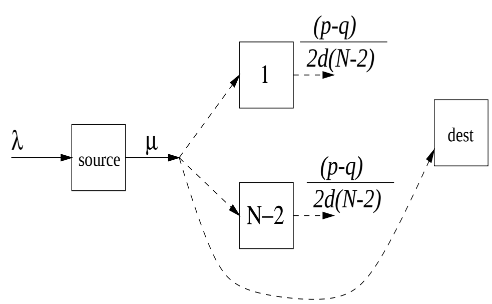

Fig. 2. A decoupled diagram of the network as seen by the packets transmitted from a single user to the corresponding destination. Service opportunities at the first stage are Bernoulli with rate μ. Service at the second stage (relay) queues is Bernoulli with rate $(p-q)/(2d(N-2))$.

Theorem 2: Consider a cell partitioned network (with N users and C cells) under the 2-hop relay algorithm, and assume that users change cells iid and uniformly over each cell every timeslot. If the exogenous input stream to user i is a Bernoulli stream of rate λ_i (where $\lambda_i < \mu$), then the total network delay W_i for user i traffic satisfies:

$$\mathbb{E}\left\{W_i\right\} = \frac{N-1-\lambda_i}{\mu-\lambda_i} \tag{4}$$

where the capacity μ is defined in (1).

Proof: The proof uses reversibility of the first stage queue, and is provided in Appendix A. □

Note that the decoupling property of the cell partitioned relay algorithm admits a decoupled delay bound, so that the waiting time for user i packets depends only on the rate of the input stream for user i, and does not depend on the rate of other streams—even if the rate of these streams is greater than capacity. It follows that the network is stable with bounded delays whenever all input streams are less than capacity, i.e., when $\lambda_i < \mu$ for all users i. Thus, the relay algorithm achieves the capacity bound given in (1) of Theorem 1. It is perhaps counter-intuitive that the algorithm achieves capacity, as it often forces cells to remain idle even when choosing an alternate sender would allow for a packet to be delivered to its destination. The intuition is that all cases of idleness arise because a queue is empty, an event that becomes increasingly unlikely as load approaches capacity.

The form of the delay expression is worth noting. First note the classic $1/(\mu-\lambda_i)$ behavior, representing the asymptotic growth in delay as data rates are pushed towards the capacity boundary. Second, note that for a fixed loading value $\rho_i = \lambda_i/\mu$, delay is $O(N)$, growing linearly in the size of the network.

The exact delay analysis is enabled by the Bernoulli input assumption. If inputs are assumed to be Poisson, the delay theory in [4] can be used to develop a delay bound, and the bound for Poisson inputs is not considerably different from the exact expression for Bernoulli inputs given in (4). These results can also be extended to the case when the mobility model conforms to a Markovian random walk (see analytical discussion and simulation results in Sections VI and VII).

III. Sending a Single Packet

In the previous subsection we showed that the cell partitioned relay algorithm yields an average delay of $O(N/(\mu-\lambda_i))$. Inspection of (4) shows that this $O(N)$ characteristic cannot be removed by decreasing the data rate λ. The following questions emerge: Can another scheduling algorithm be constructed which improves delay? What is the minimum delay the network can guarantee, and for what data rates is this delay obtainable? More generally, for a given data rate λ (assumed to be less than the system capacity μ), we ask: What is the optimal delay bound, and what algorithm achieves this? In this section we present several fundamental bounds on delay performance, which establishes initial steps towards addressing these general questions.

A. Scheduling Without Redundancy

Suppose that no redundancy is used: that is, packets are not duplicated and are held by at most one user of the network at any given time. Thus, a packet that is transmitted to another user is deleted from the memory storage of the transmitting user. Note that this is the traditional approach to data networking, and that the 2-hop relay algorithm is in this class.

Theorem 3: Algorithms which do not use redundancy cannot achieve an average delay of less than $O(N)$.

Proof: The minimum delay of any packet is computed by considering the situation where the network is empty and user 1 sends a single packet to user 2. It is easy to verify that relaying the packet cannot help, and hence the delay distribution is geometric with mean $C = N/d$. □

Hence, the relay algorithm not only achieves capacity, but achieves the optimal $O(N)$ delay performance among all strategies which do not use redundancy. Other policies which do not use redundancy can perhaps improve upon the delay coefficient, but cannot change the $O(N)$ characteristic.

B. Scheduling With Redundancy

Although redundancy cannot increase capacity, it can considerably improve delay. Clearly, the time required for a packet to reach the destination can be reduced by repeatedly transmitting this packet to many users of the network—improving the chances that some user holding an original or duplicate version of the packet reaches the destination. Consider any network algorithm (which may or may not use redundant packet transfers) that restricts packets to 2-hop paths.

Theorem 4: No algorithm (with or without redundancy) which restricts packets to 2-hop paths can provide an average delay better than $O(\sqrt{N})$.

Again consider the sending of a single packet from its source to its destination. Clearly the optimal scheme is to have the source send duplicate versions of the packet to new relays whenever possible, and for the packet to be relayed to the destination as soon as either the source or a duplicate-carrying relay enters the same cell as the destination.

Let T_N represent the average time required to reach the destination under this optimal policy for sending a single packet. In the following lemma we bound the limiting behavior[1] of $\mathbb{E}\left\{T_N\right\}$, proving Theorem 4.

Lemma 1: $e^{-d} \leq \lim_{N\to\infty} \frac{\mathbb{E}\{T_N\}}{\sqrt{N}} \leq \frac{2}{1-e^{-d}}$

Proof: Lemma 1 (a) *Lower Bound:* To prove the lower bound, note that during timeslots $\{1, 2, \ldots, \sqrt{N}\}$, there are fewer than $\sqrt{N}$ users holding the packet. Hence, $Pr[T_N > \sqrt{N}] \geq (1 - 1/C)^{\sqrt{N}\sqrt{N}}$ (where $(1 - 1/C)^{\sqrt{N}}$ is the probability that nobody within a group of $\sqrt{N}$ particular users enters the cell of the destination during a given timeslot). Recall that the user/cell density d is defined $d \triangleq N/C$. Thus:

$$\begin{aligned} \mathbb{E}\left\{T_N\right\} &\geq \mathbb{E}\left\{T_N | T_N > \sqrt{N}\right\} Pr[T_N > \sqrt{N}] \\ &\geq \sqrt{N}\left(1 - \frac{d}{N}\right)^N \rightarrow e^{-d}\sqrt{N} \end{aligned}$$

(b) *Upper Bound:* To prove the upper bound, note that $\mathbb{E}\left\{T_N\right\} \leq S_1 + S_2$, where S_1 represents the expected number of slots required to send out duplicates of the packet to $\sqrt{N}$ different users, and S_2 represents the expected time until one user within a group of $\sqrt{N}$ users containing the packet reaches the cell of the destination. The probability of the source meeting a new user is at least $1 - (1 - 1/C)^{N-\sqrt{N}}$ for every timeslot where fewer than $\sqrt{N}$ users have packets, and hence the average time to reach a new user is less than or equal to the inverse of this quantity (i.e, the average time of a geometric variable). Hence:

$$S_1 \leq \frac{\sqrt{N}}{1 - (1 - 1/C)^{N-\sqrt{N}}} \rightarrow \frac{\sqrt{N}}{1 - e^{-d}}$$

To compute S_2, note that $P(success)$, the probability that one of the $\sqrt{N}$ users reaches the destination during a slot, is given by the probability there is at least one other user in the same cell as the destination multiplied by the conditional probability that a packet-carrying user is present given there is at least one other user in the cell. The former probability is $1 - (1 - 1/C)^{N-1}$, and the latter is at least $\sqrt{N}/N$:

[1] Using the inequality $e^{\frac{-d^2}{N-d}} e^{-d} \leq \left(1 - \frac{d}{N}\right)^N \leq e^{-d}$, explicit bounds of the form $\alpha\sqrt{N} \leq \mathbb{E}\left\{T_N\right\} \leq \beta\sqrt{N}$ can also be derived.

$$P(success) \geq \frac{1 - (1 - 1/C)^{N-1}}{\sqrt{N}} \rightarrow \frac{1 - e^{-d}}{\sqrt{N}} \tag{5}$$

Hence, $S_2 \leq \frac{\sqrt{N}}{1-e^{-d}}$. Summing S_1 and S_2 proves the result. □

C. Multi-User Reception

To increase the packet replication speed throughout the network, it is useful to allow a transmitted packet to be received by *all other users* in the same cell as the transmitter, not just the single intended recipient. This feature cannot increase capacity, but can considerably improve delay by enabling multiple duplicates to be injected into the network with just a single transmission. However, provided that the user/cell density d is $O(1)$ (independent of N), the $O(\sqrt{N})$ result of Theorem 4 cannot be overcome by introducing multi-user reception (see [1]). For the remainder of this paper, we assume multi-user reception is available.

IV. Scheduling for Delay Improvement

In the previous section an $O(\sqrt{N})$ delay bound was developed for redundant scheduling by considering a single packet for a single destination. Two complications arise when designing a general scheduling protocol using redundancy: (1) All sessions must use the network simultaneously, and (2) Remnant versions of a packet that has already been delivered to its destination create excess congestion and must somehow be removed.

Here we show that the properties of the 2-hop relay algorithm make it naturally suited to treat the multi-user problem. The second complication of excess packets is overcome by the following *in-cell feedback protocol*, in which a receiving node tells its transmitter which packet it is looking for before transmission begins. We assume all packets are labeled with *send numbers* SN, and the in-cell feedback is in the form of a *request number* RN delivered by the destination to the transmitter just before transmission. In the following protocol, each packet is retransmitted $\sqrt{N}$ times to distinct relay users.

In-Cell Feedback Scheme with $\sqrt{N}$ Redundancy: In every cell with at least two users, a random sender and a random receiver are selected, with uniform probability over all users in the cell. With probability $1/2$, the sender is scheduled to operate in either 'source-to-relay' mode, or 'relay-to-destination' mode, described as follows:

1) *Source-to-Relay Mode:* The sender transmits packet SN, and does so upon every transmission opportunity until $\sqrt{N}$ replicas have been delivered to distinct users, or until the sender transmits SN directly to the destination. After such a time, the send number is incremented to $SN + 1$. If the sender does not have a new packet to send, remain idle.

2) *Relay-to-Destination Mode:* When a user is scheduled to transmit a relay packet to its destination, the following handshake is performed:
   - The receiver delivers its current RN number for the packet it desires.
   - The transmitter deletes all packets in its buffer destined for this receiver which have SN numbers lower than RN.
   - The transmitter sends packet RN to the receiver. If the transmitter does not have the requested packet RN, it remains idle for that slot.

Notice that the destination receives all packets in order, and that no packet is ever transmitted twice to its destination.

Theorem 5: The In-Cell Feedback Scheme achieves the $O(\sqrt{N})$ delay bound, with user data rates of $O(1/\sqrt{N})$.

More precisely, if all users receive exogenous data for their destinations according to a Poisson process of rate λ_i, the network can stably support rates $\lambda_i < \tilde{\mu}$, for the reduced network throughput $\tilde{\mu}$ given by:

$$\tilde{\mu} = \frac{\gamma_N \left(1 - e^{-d}\right)}{4(2+d)\sqrt{N}} \tag{6}$$

where γ_N is a sequence that converges to 1 as $N \to \infty$. Furthermore, average end-to-end delay $\mathbb{E}\left\{W_i\right\}$ satisfies:

$$\mathbb{E}\left\{W_i\right\} \leq \frac{1}{2} + \frac{1/\tilde{\mu}}{1-\rho_i}$$

where $\rho_i \triangleq \lambda_i/\tilde{\mu}$.

To prove the result, first note that when a new packet reaches the head of the line at its source queue, the time required for the packet to reach its destination is at most $T_N = S_1 + S_2$, where S_1 represents the time required for the source to send out $\sqrt{N}$ replicas of the packet, and S_2 represents the time required to reach the destination given that $\sqrt{N}$ users have the packet. Bounds on the expectations of S_1 and S_2 which are independent of the initial state of the network can be computed similarly to the proof of Lemma 1. The multi-user environment here simply acts to scale up these expectations by a constant factor due to collisions with other users (compare the upper bound of Lemma 1 with that given in (7) below). This factor does not scale with N because the average number of users in any cell is the finite number d. Indeed, in [1] it is shown that:

$$\mathbb{E}\left\{T_N\right\} \leq \frac{4(2+d)\sqrt{N}}{\gamma_N(1-e^{-d})} \tag{7}$$

where γ_N is a function that converges to 1 as $N \to \infty$.

Note that the random variable T_N satisfies the *sub-memoryless* property: The residual time of T_N given that a fixed number of slots have already passed (without T_N expiring) is stochastically less than the original time T_N.[2] This is because the topology of the network is independent from slot to slot, and hence starting out with several duplicate packets already in the network yields statistically smaller delay than if no such initial duplicates are present.

[2]This is often called the 'New Better than Used' property', see [18].

The RN/SN handshake ensures that newer packets do not interfere with older packets, but that replication of the next packet waiting at the source queue begins on or before completion of the T_N 'service time' for the current packet SN. Packets thus view the network as a single queue to which they arrive and are served sequentially. Although actual service times may not be iid, they are all independently bounded by $\mathbb{E}\left\{T_N\right\}$, as are residual service times seen by a randomly arriving packet. This is sufficient to establish the following lemma, the proof of which is omitted for brevity (see [1]).

Lemma 2: Suppose inputs to a single server queue are Poisson with sub-memoryless service times that are independently bounded by a value $\mathbb{E}\left\{T_N\right\}$. If the arrival rate is λ, where $\lambda < 1/\mathbb{E}\left\{T_N\right\}$, then average delay satisfies:

$$\mathbb{E}\left\{W\right\} \leq \frac{1}{2} + \frac{\mathbb{E}\left\{T_N\right\}}{1-\rho} \tag{8}$$

where $\rho \triangleq \lambda \mathbb{E}\left\{T_N\right\}$. The expression on the right hand side of the above inequality is the standard expression for delay in an M/M/1 queue with iid service times T_N that are restricted to start on slot boundaries.

Defining $\tilde{\mu} \triangleq 1/\mathbb{E}\left\{T_N\right\} = 1/\mathbb{E}\left\{S_1 + S_2\right\}$ proves Thm. 5.

A. Multi-Hop Scheduling

To further improve delay, we can remove the 2-hop restriction and consider schemes which allow for multi-hop paths. Here, a simple flooding protocol is developed and shown to achieve $O(\log(N))$ delay at the expense of further reducing throughput.

To achieve $O(\log(N))$ delay, consider the situation in which a single packet is delivered over an empty network. At first, only the source user contains the packet. The packet is transmitted and received by all other users in the same cell as the source. In the next timeslot, the source as well as all of the new users containing the packet transmit in their respective cells, and so on. If all duplicate-carrying users enter distinct cells every timeslot, and each of these users delivers the packet to exactly one new user, then the number of users containing the packet grows geometrically according to the sequence $\{1, 2, 4, 8, 16, \ldots\}$. The actual growth pattern may deviate from this geometric sequence somewhat, due to multiple users entering the same cell, or to users entering cells that are devoid of other users. However, it can be shown that the *expected* growth is geometric provided that the number of packet-holding users is less than $N/2$.

Define the total time to reach all users as $T_N = S_1 + S_2$, where S_1 and S_2 respectively represent the time required to send the packet to at least $N/2$ users, and the time required to deliver the packet to the remaining users given that at least $N/2$ users initially hold the packet.

Lemma 3: Under the above algorithm of flooding the network with a single packet, for any network size $N \geq 2$, the expected time $\mathbb{E}\left\{T_N\right\}$ for the packet to reach every user satisfies $\mathbb{E}\{T_N\} \leq \mathbb{E}\{S_1\} + \mathbb{E}\{S_2\}$, where:

$$\mathbb{E}\{S_1\} \leq \frac{\log(N)(1+d/2)}{\log(2)(1-e^{-d/2})}$$
$$\mathbb{E}\{S_2\} \leq 1 + \frac{2}{d}(1+\log(N/2)) \quad (9)$$

Proof: The proof of the $\mathbb{E}\{S_2\}$ bound is given in Appendix B. The $\mathbb{E}\{S_1\}$ proof is omitted for brevity (see proof in [1]). □

Thus, $O(\log(N))$ delay is achievable when sending a single packet over an empty network. To enable $O(\log(N))$ delay in the general case where all sessions are active and share the network resources, we construct a flooding protocol in which the oldest packet that has not been delivered to all users is selected to dominate network resources. We assume that packets are sequenced with SN numbers as before. Additionally, packets are stamped with the timeslot t in which they arrived.

Fair Packet Flooding Protocol: Every timeslot and in each cell, users perform the following: Among all packets contained in at least one user of the cell but which have never been received by some other user in the same cell, choose the packet p which arrived earliest (i.e., it has the smallest timestamp t_p). If there are ties, choose the packet from the session i which maximizes $(t_p + i) \mod N$. Transmit this packet to all other users in the cell. If no such packet exists, remain idle.

The above protocol is 'fair' in that in case of ties, session i packets are given top priority every N timeslots. Other schemes for choosing which packet to dominate the network could also be considered. Delay under the above protocol can be understood by comparing the network to a single queue with N input streams of rates $\lambda_1, \lambda_2, \ldots, \lambda_N$ which share a single server with service times T_N. Note that the T_N service time is also sub-memoryless. Thus, from Lemma 2, we have:

Theorem 6: For Poisson inputs with rates λ_i for each source i, the network under the fair flooding protocol is stable whenever $\sum_i \lambda_i < 1/\mathbb{E}\{T_N\}$, with average end-to-end delay satisfying:

$$\mathbb{E}\{W\} \leq \frac{1}{2} + \frac{\mathbb{E}\{T_N\}}{1-\rho} \quad (10)$$

where $\rho \triangleq \sum_i \lambda_i \mathbb{E}\{T_N\}$, and $\mathbb{E}\{T_N\} = \mathbb{E}\{S_1\} + \mathbb{E}\{S_2\}$. Note that $O(\log(N))$ bounds on $\mathbb{E}\{S_1\}$ and $\mathbb{E}\{S_2\}$ are given in Lemma 3. Thus, when all sources have identical input rates λ, stability and logarithmic delay is achieved when $\lambda = O(\frac{1}{N\log(N)})$.

Note that the flooding algorithm easily allows for *multicast sessions*, where data of rate λ is delivered from each source to *all other users*. One might expect that delay can be improved if we only design for unicast. However, it is shown in [1] that logarithmic delay is the best possible for any strategy at any data rate. Hence, communication for unicast or multicast is the same in the logarithmic delay regime. In the next section, we address the following question: Is it possible to increase data rates via some other protocol while maintaining the same average delay guarantees?

V. Fundamental Delay/Rate Tradeoffs

Considering the capacity achieving 2-hop relay algorithm, the 2-hop algorithm with $\sqrt{N}$ redundancy, and the packet flooding protocol, we have the following achievable delay/capacity performance tradeoffs.

scheme	capacity	delay
no redundancy	$O(1)$	$O(N)$
redundancy 2-hop	$O(1/\sqrt{N})$	$O(\sqrt{N})$
redundancy multi-hop	$O(\frac{1}{N\log(N)})$	$O(\log(N))$

A simple observation reveals that $delay/rate \geq O(N)$ for each of these three protocols. In this section, we establish that this is in fact a necessary condition. Thus, performance of each given protocol falls on or near the boundary of a *fundamental rate-delay curve* (see Fig. 1).

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

Theorem 7: The following inequality is necessary for any conceivable routing and scheduling protocol which stabilizes the network with input rates λ while maintaining bounded average end-to-end delay $\overline{W}$:

$$\frac{\overline{W}}{\lambda} \geq \frac{N-d}{4d}(1-\log(2)) \quad (11)$$

where $\log()$ denotes the natural logarithm, and $d = N/C$. In particular, if $d = O(1)$, then $\frac{\overline{W}}{\lambda} \geq O(N)$.

We prove this theorem with a novel technique for probabilistic conditioning.

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

$$\overline{W} = \frac{1}{N}\sum_i \overline{W}_i \quad (12)$$

Let $\overline{R}_i$ represent the average *redundancy* associated with packets from session i. That is, $\overline{R}_i$ is the number of users who receive a copy of an individual packet during the course of the network control operation, averaged over all packets from session i. Note that all packets are eventually received by the destination, so that $\overline{R}_i \geq 1$. Additional redundancy could be introduced by multi-hop routing, or by any packet replication effort that is used to achieve stability and/or improve delay. The average number of successful packet receptions per timeslot is thus given by the quantity $\lambda\sum_{i=1}^N \overline{R}_i$. Because each of the N users can receive at most 1 packet per timeslot, we have:

$$\lambda\sum_{i=1}^{N} \overline{R}_i \leq N \quad (13)$$

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

$$\begin{aligned}
\overline{W}_i &\geq \mathbb{E}\left\{W_i \mid R_i \leq 2\overline{R}_i\right\} Pr[R \leq 2\overline{R}_i] \\
&\geq \mathbb{E}\left\{W_i \mid R_i \leq 2\overline{R}_i\right\} \frac{1}{2} \qquad (14)
\end{aligned}$$

where (14) follows because $Pr[R_i \leq 2\overline{R}_i] \geq \frac{1}{2}$ for any non-negative random variable R_i.

Note that the smallest possible delay for packet p is the time required for one of its carriers to enter the same cell as the destination. Consider now a virtual system in which there are $2\overline{R}_i$ users initially holding packet p, and let Z represent the time required for one of these users to enter the same cell as the destination. Every timeslot the 'success probability' for this system is $\phi \triangleq 1 - (1 - \frac{1}{C})^{2\overline{R}_i}$, so that $\mathbb{E}\{Z\} = 1/\phi$. Although there are more users holding packet p in this system, the expectation of Z does not necessarily bound $\mathbb{E}\left\{W_i \mid R_i \leq 2\overline{R}_i\right\}$ because conditioning on the event $\{R_i \leq 2\overline{R}_i\}$ might skew the probabilities associated with the user mobility process. However, because the event $\{R_i \leq 2\overline{R}_i\}$ occurs with probability at least $1/2$, we obtain the following bound:

$$\mathbb{E}\left\{W_i \mid R_i \leq 2\overline{R}_i\right\} \geq \inf_{\Theta} \mathbb{E}\{Z \mid \Theta\} \qquad (15)$$

where the conditional expectation is minimized over all conceivable events Θ which occur with probability greater than or equal to $1/2$.

We now *stochastically couple* Z to an independent exponential variable $\tilde{Z}$ with rate $\gamma \triangleq \log(1/(1-\phi))$. The variable $\tilde{Z}$ is *stochastically less* than Z because $Pr[\tilde{Z} > \omega] \leq Pr[Z > \omega]$ for all ω. Indeed, because $\tilde{Z}$ is exponential with rate γ, we have $Pr[\tilde{Z} > \omega] = e^{-\gamma\omega} = (1-\phi)^{\omega}$ for any $\omega \geq 0$, while Z is geometric with success probability ϕ, so that:

$$\begin{aligned}
Pr[Z > \omega] = Pr[Z > \lfloor \omega \rfloor] = (1-\phi)^{\lfloor \omega \rfloor} &\geq (1-\phi)^{\omega} \\
&= Pr[\tilde{Z} > \omega]
\end{aligned}$$

The fact that $\tilde{Z}$ is stochastically less than Z leads to the following claim:

Claim 1: *For variables Z and $\tilde{Z}$, we have:*

$$\inf_{\Theta} \mathbb{E}\{Z \mid \Theta\} \geq \inf_{\tilde{\Theta}} \mathbb{E}\left\{\tilde{Z} \mid \tilde{\Theta}\right\} = \frac{1 - \log(2)}{\gamma} \qquad (16)$$

where the first infimum is taken over all events Θ that occur with probability greater than or equal to $1/2$ on the probability space for Z, and the second infimum is taken over all events $\tilde{\Theta}$ that occur with probability greater than or equal to $1/2$ on the probability space for $\tilde{Z}$.

The claim is proven at the end of this subsection. Using (16) and (15) in (14) yields: $\overline{W}_i \geq \frac{1-\log(2)}{2\gamma}$. From the definitions of γ and ϕ, we have $\gamma = \log\left(1/(1-\frac{1}{C})^{2\overline{R}_i}\right) = 2\overline{R}_i \log(1 + \frac{1}{C-1})$. Because $\log(1+x) \leq x$ for any x, we have $\gamma \leq 2\overline{R}_i/(C-1)$. We thus have:

$$\overline{W}_i \geq \frac{1 - \log(2)}{2\gamma} \geq \frac{(C-1)(1-\log(2))}{4\overline{R}_i}$$

Summing this inequality over all i, we have:

$$\begin{aligned}
\overline{W} = \frac{1}{N}\sum_{i=1}^{N} \overline{W}_i &\geq \frac{(C-1)\,(1-\log(2))}{4} \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\overline{R}_i} \\
&\geq \frac{(C-1)\,(1-\log(2))}{4\frac{1}{N}\sum_{i=1}^{N}\overline{R}_i} \qquad (17)
\end{aligned}$$

where (17) follows from Jensen's inequality, noting that the function $1/R$ is convex. Combining (17) and (13), we have:

$$\overline{W} \geq \frac{(C-1)\,(1-\log(2))\,\lambda}{4} = \frac{(N-d)\,(1-\log(2))\,\lambda}{4d}$$

Hence, the delay/rate characteristics necessarily satisfy the inequality $\frac{\overline{W}}{\lambda} \geq O(N)$, proving the theorem. □

We complete the analysis by proving Claim 1:

Proof: (Claim 1) We first compute $\inf_{\tilde{\Theta}} \mathbb{E}\left\{\tilde{Z} \mid \tilde{\Theta}\right\}$. Note that $\tilde{Z}$ is a continuous variable, and so the minimizing event $\tilde{\Theta}$ is clearly the event $\{\tilde{Z} \leq \omega\}$, where ω is the smallest value such that $Pr[\tilde{Z} \leq \omega] \geq \frac{1}{2}$. Because $\tilde{Z}$ is exponential with rate $\gamma = \log(1/(1-\phi))$, we have $Pr[\tilde{Z} > \omega] = e^{-\gamma\omega} = 1/2$, and hence $\omega = \frac{\log(2)}{\gamma}$. Conditioning on this event, we have:

$$\begin{aligned}
\inf_{\tilde{\Theta}} \mathbb{E}\left\{\tilde{Z} \,|\Theta\right\} &= \mathbb{E}\left\{\tilde{Z} \mid \tilde{Z} \leq \omega\right\} \\
&= \frac{\mathbb{E}\left\{\tilde{Z}\right\} - \mathbb{E}\left\{\tilde{Z} \mid \tilde{Z} > \omega\right\} Pr[\tilde{Z} > \omega]}{Pr[\tilde{Z} \leq \omega]} \\
&= \frac{\frac{1}{\gamma} - (\omega + \frac{1}{\gamma})\frac{1}{2}}{1/2} = \frac{1 - \log(2)}{\gamma}
\end{aligned}$$

Now note that $\tilde{Z}$ is *stochastically less* than Z, so that there must exist a *coupling variable* Z' such that variables $\tilde{Z}$ and Z' have the same distribution, and Z' lies on the same probability space as Z and satisfies $Z' \leq Z$ for all instances of Z and Z' (see [18] for a discussion of stochastic coupling). Because Z' is also an exponential with rate γ, it follows that $\inf_{\Theta} \mathbb{E}\{Z' \mid \Theta\} = (1 - \log(2))/\gamma$. However, because $Z' \leq Z$ always, it follows that:

$$\inf_{\Theta} \mathbb{E}\{Z' \mid \Theta\} \leq \inf_{\Theta} \mathbb{E}\{Z \mid \Theta\}$$

proving the claim. □

The fact that $delay/rate \geq O(N)$ establishes a fundamental performance tradeoff, illustrating that no scheduling and routing algorithm can simultaneously yield low delay and high throughput. The $O(N)$ and $O(\sqrt{N})$ scheduling algorithms provided here meet this bound with equality, and the $O(\log(N))$ algorithm lies above the bound by a factor of $O(\log^2(N))$ (see table above).

We note that alternate approaches to the capacity/delay tradeoff problem were recently developed in [16] [15] [17] for networks with different physical characteristics. Specifically, the work in [16] develops a similar $\overline{W}/\lambda \geq O(N)$ curve by assuming the user transmission radius can be increased to include $O(N^{\alpha})$ other users, where α is between 0 and 1 and affects the delay tradeoff. This analysis does not consider the use of redundant packet transfers or multi-user reception. A similar approach by Toumpis and Goldsmith in [15] shows that an improved tradeoff $\overline{W}/\lambda^2 = O(N \log^5(N))$ can be achieved when multi-user reception is used together with transmission radius scaling, but there was no proof of optimality.

In the context of a cell partitioned network as we have defined, an increased transmission radius would correspond to a user/cell density that is a function of N, that is, $d = O(N^{\alpha})$. While our work was developed independently and intended only for the case $d = O(1)$ (independent of N), the necessary condition in Theorem 7 was proven for arbitrary values of the user/cell density d, and hence it can be used to evaluate the performance of the Toumpis-Goldsmith algorithm applied to a cell partitioned network. Indeed, first note that the additional inequality $N\lambda \leq C$ must hold for any policy on a cell partitioned network (as the rate of new packets transmitted by their sources is less than or equal to C, the maximum number of transmissions possible during a slot). Thus, $\lambda \leq 1/d$ is necessary for any protocol, and directly plugging this inequality into (11) yields: $\overline{W}/\lambda^2 \geq \frac{(N-d)(1-\log(2))}{4}$.

Hence, the Toumpis-Goldsmith algorithm is near-optimal over the class of all algorithms that can be implemented on a cell partitioned network that does not impose the constraint $d = O(1)$. We note that a recent preliminary result in [17] suggests that an improved tradeoff $\overline{W}/\lambda^3 \geq O(N)$ is possible if the network has different physical properties that allow for multi-hop transmission during a single slot (so that a bit can be transfered from node 1 to node 2 ... to node K, all during a single slot). Of course, it is not possible to implement such an algorithm on the cell partitioned network that we have defined, because transmission on each successive hop would require a new timeslot.

VI. Non-iid Mobility Models

The analysis developed here for the iid mobility model can be used to bound the performance of a system with a Markovian mobility model. Instead of performing control actions on the network every slot, we decompose the network into a set of K parallel sub-networks. Packets are considered to be of 'type-k' if they arrive during a timeslot t such that $t \mod K = k$. On such timeslots, only control actions on type-k packets take place. The value of K is chosen suitably large to ensure that the user location distribution after K slots is within a constant factor of its steady state value. Specifically, if K is chosen such that, regardless of the initial configuration of users, the probability that two given users are in the same cell after K slots is at least $\frac{1}{2C}$, then delay under the three schemes is bounded by $O(KN), O(K\sqrt{N})$, and $O(K\log(N))$, respectively (see [1]).

However, it is possible that alternative scheduling schemes could yield lower delay. Indeed, in the next section it is shown through simulation that applying the 2-hop relay algorithm and the $\sqrt{N}$ redundancy algorithm exactly as before (without the K-subchannel decomposition) yields similar performance for both iid and non-iid mobility.

VII. Simulation Results

Here we compare the average delay obtained through both analysis and simulation as the network is scaled. We consider a network with cells given by an $M \times M$ grid as shown in Fig. 1. The number of cells C is equal to M^2 (where M is varied between 3 and 15 for simulations), and the number of users N is chosen as the even integer for which N/C most accurately approximates the optimal user/cell density value $d^* = 1.7933$.

In Fig. 3, plots of average end-to-end delay versus the number of users N are provided for the 2-hop relay algorithm and the $O(\sqrt{N})$ redundancy algorithm for both an i.i.d. and a non-i.i.d. mobility model. In the i.i.d. mobility model, users choose new cells uniformly over all cells in the network. In the non-i.i.d. model, each user chooses a new cell every timeslot according to the following Markovian dynamics: With probability $\alpha < 1$ the user stays in the same cell, and else it moves to an adjacent cell to the North, South, East, or West, with each direction equally likely. In the case where a user is on the edge of the network and is selected to move in an infeasible direction, it stays in its place. Using standard random walk theory it is easy to verify that, in steady state, such a Markov model leaves users independently and uniformly distributed over all cells, as the stationary equation for the Markov chain is satisfied when all cell locations have equal probability [19]. In particular, if π_i represents the steady state probability of a particular cell i, we have:

$$\pi_i = \pi_i\alpha + \pi_a\frac{(1-\alpha)}{4} + \pi_b\frac{(1-\alpha)}{4} + \pi_c\frac{(1-\alpha)}{4} + \pi_d\frac{(1-\alpha)}{4}$$

where $\pi_a, \pi_b, \pi_c, \pi_d$ represent steady state probabilities for other cells, possibly including cell i. In the case when cell i is an interior cell, it has four distinct neighbors a, b, c, d. In the case when it is an edge cell with three neighbors a, b, c, we set $d = i$ (so that cell i is its own neighbor). In the case when cell i is a corner cell with 2 neighbors a and b, we set $c = d = i$. Clearly these steady state equations are satisfied when the π_i probabilities are set to $1/C$ for all i. Therefore, the network capacity μ is the same for both the i.i.d. mobility model and the non-i.i.d. mobility model, and is given by $\mu = \frac{p+q}{2d}$ as described in Theorem 1. In the simulation results we set the α parameter of the non-i.i.d. model to $\alpha = 1/2$.

For the capacity achieving 2-hop relay algorithm, the data rate λ into each user is fixed at 80% of the network capacity μ (given in Theorem 1), so that $\rho = \lambda/\mu = 0.8$. The top three curves for average delay in Fig. 3 respectively represent the exact analytical delay for i.i.d. mobility, the simulated performance of the i.i.d. mobility model, and the simulated performance of the Markovian mobility model. Note that the simulation curve for the i.i.d. mobility model is almost indistinguishable from the analytical curve $\mathbb{E}\{W\} = \frac{N-1-\lambda}{\mu-\lambda}$. The curves are plotted on a $\log\log$ scale and have a slope of 1, indicating $O(N)$ delay. The delay curve for Markovian

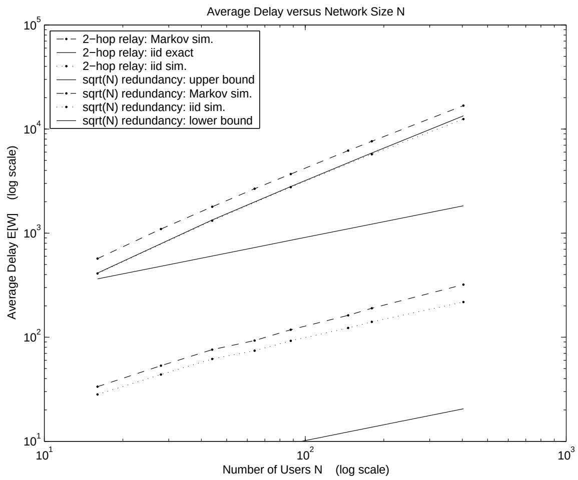

Fig. 3. Average delay versus the number of users N for the 2-hop relay algorithm and the $\sqrt{N}$ redundancy algorithm.

mobility is situated slightly above the curve for i.i.d. mobility, and also has a slope of 1. This suggests that for Markovian mobility, delay is increased by a constant multiplicative factor but remains $O(N)$.

Results for the $\sqrt{N}$ redundancy protocol are also shown in the figure. Data rates λ are set to the value $\lambda = 0.8\tilde{\mu}$, where $\tilde{\mu}$ is given in (6). Note that, unlike the network capacity μ, the throughput $\tilde{\mu}$ decreases as $O(1/\sqrt{N})$. The analytical upper and lower bounds on delay for i.i.d. mobility are shown in the figure, each having a slope of $1/2$ indicating $O(\sqrt{N})$ growth (note that the lower bound represents the delay of sending just a single packet). The simulation performance for i.i.d. mobility is shown in the figure and is situated between the upper and lower bounds. The upper bound is larger than the simulated curve by approximately a factor of 10, suggesting that tighter bounds could be obtained through a more detailed analysis. The slope of the simulation curve varies between $5/8$ and $1/2$. However, because delay is upper and lower bounded by functions of $O(\sqrt{N})$, the average slope would converge to $1/2$ if the graph were extended. Simulation of the Markovian mobility model is also provided, and the curve again lies slightly above the i.i.d. mobility curve. This suggests that delay under the Markovian model is close to $O(\sqrt{N})$.

Experiments to simulate the performance of the $O(\log(N))$ scheme were not performed. However, for this case, we would expect a discrepancy between the i.i.d. mobility model and the non-i.i.d. mobility model. Indeed, although the i.i.d. mobility model yields logarithmic delay, the delay under a Markovian mobility model would likely be closer to $O(\sqrt{N})$ due to the time required for a user to travel from one side of the network to the other.

VIII. Conclusions

This work for the first time presents a multi-hop, multi-user system for which a relatively complete network theory can be developed. Exact expressions for network capacity were derived, and a fundamental rate-delay curve was established, representing performance bounds on throughput and end-to-end network delay for any conceivable routing and scheduling policy.

Delay analysis for the network was facilitated using a simple iid user mobility model. Under this model, an exact expression for end-to-end delay which includes the full effects of queueing was established for the capacity achieving 2-hop relay algorithm. Two other protocols which (necessarily) use redundant packet transfers were provided and shown to improve delay at the expense of reducing throughput. The rate-delay performance of these schemes was shown to lie on the boundary of the fundamental performance curve $delay/rate \geq O(N)$. Analysis of general mobility models can be understood in terms of this iid analysis, where delay bounds can be scaled by the factor K, representing the number of slots required between sampling points for samples of user locations to look nearly iid. Furthermore, simulation results suggest that $O(\sqrt{N})$ delay can be achieved for networks with Markovian mobility, as the delay for such systems closely follows the delay curve for a system with iid mobility.

This inspires a rich set of questions concerning the fundamental limits of data networks. We believe that the condition $delay/rate \geq O(N)$ is necessary for general classes of mobile wireless networks, and that the $(rate, delay) = \Big(O(1/\sqrt{N}), O(\sqrt{N})\Big)$ operating point is always achievable. Such conjectures can perhaps be established using analytical techniques similar to those created here.

Appendix A

Proof of Delay Bound in Theorem 3: The exact end-to-end network delay under the 2-hop relay algorithm with Bernoulli inputs and iid mobility is $\mathbb{E}\left\{W_i\right\} = \frac{N-1-\lambda_i}{\mu-\lambda_i}$.

Proof: A decoupled view of the network as perceived by a single user i is illustrated in Fig. 2. Because of the iid mobility, the source user can be represented as a Bernoulli/Bernoulli queue, where every timeslot a new packet arrives with probability λ, and a service opportunity arises with some fixed probability μ. We first show that $\mu = \frac{p+q}{2d}$. 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. Hence, we have $\mu = r_1 + r_2$, where r_1 represents the rate at which the source is scheduled to transmit directly to the destination, and r_2 represents the rate at which it is scheduled to transmit to one of its relay users. The cell partitioned relay algorithm schedules transmissions into and out of the relay nodes with equal probability, and hence r_2 is also equal to the rate at which the relay nodes are scheduled to transmit to the destination. The total rate of transmission opportunities over the network is thus $N(r_1 + 2r_2)$. A transmission opportunity occurs in any given cell with probability p, and hence:

$$Cp = N(r_1 + 2r_2) \tag{18}$$

Recall that q is the probability that a given cell contains a source-destination pair. Because the cell partitioned relay algorithm schedules the single-hop ‘source-to-destination’ transmissions whenever possible, the rate r_2 satisfies:

$$Cq = Nr_2 \tag{19}$$

It follows from (19) that $r_2 = q/d$, and hence by (18) we infer that $r_1 = \frac{p-q}{2d}$. The total rate of transmissions out of the source node is thus given by $\mu = r_1 + r_2 = \frac{p+q}{2d}$.

The source is thus a Bernoulli/Bernoulli queue with input rate λ and server probability μ, having an expected number of packets given by $\overline{L}_{source} = \frac{\rho(1-\lambda)}{1-\rho}$, where $\rho \triangleq \lambda/\mu$ [20]. This queue is *reversible* ([19], [20]), and so the output process is also a Bernoulli stream of rate λ.

A given packet from this output process is transmitted to the *first relay node* with probability $\frac{r_2}{\mu(N-2)}$ (because with probability r_2/μ the packet is intended for a relay node, and each of the $N-2$ relay nodes are equally likely). Hence, every timeslot this relay independently receives a packet with probability $\tilde{\lambda} = \frac{\lambda r_2}{\mu(N-2)}$. The relay node is scheduled for a potential packet transmission to the destination with probability $\tilde{\mu} = \frac{r_2}{(N-2)}$ (because a 'relay-to-destination' opportunity arises with probability r_2, and arises from exactly one of the $N-2$ relay nodes with equal probability). However, packet arrivals and transmission opportunities are mutually exclusive events in the relay node. It follows that the discrete time Markov Chain for queue occupancy in the relay node can be written as a simple birth-death chain which is identical to the chain of a continuous time M/M/1 queue with input rate $\tilde{\lambda}$ and service rate $\tilde{\mu}$ (where $\tilde{\lambda}/\tilde{\mu} = \rho$). This holds for each relay node, and the resulting occupancy at any relay is thus: $\overline{L}_{relay} = \frac{\rho}{1-\rho}$. From Little's Theorem, the total network delay is $\overline{W}_i = \left[\overline{L}_{source} + (N-2)\overline{L}_{relay}\right]/\lambda$, which proves the theorem. □

Appendix B — Log Delay for Flooding Protocol

Proof: (The $\mathbb{E}\{S_2\}$ Bound of Lemma 3) Let M represent the number of users who do *not* initially have the packet (so that $M \leq N/2$), and label these M users $\{u_1, u_2, \ldots, u_M\}$. Let X_i represent the number of timeslots it takes for the non-packet holding user u_i to reach a cell containing a user who has a packet. Because of the multi-user reception feature, user u_i must receive the packet at this time. The random variable X_i is geometric, in that a 'success' happens on any given timeslot with probability $\psi \geq 1-(1-\frac{1}{C})^{N/2}$. Thus, we have for all N:

$$\psi \geq 1 - e^{-d/2} \tag{20}$$

All times X_i are independent and identically distributed, and hence the random variable S_2 is equal to the maximum value of at most $M = \lfloor N/2 \rfloor$ i.i.d. variables. Hence, $\mathbb{E}\{S_2\} \leq \mathbb{E}\{\max\{X_1, X_2, \ldots, X_M\}\}$. To obtain a simple bound on this time, we consider new random variables $\{Y_1, Y_2, \ldots, Y_M\}$ which are i.i.d. and *exponentially distributed* with rate $\lambda = \log(1/(1-\psi))$. Notice that the random variable $1 + Y_i$ is *stochastically greater* than X_i, because the complementary distribution functions satisfy $Pr[1 + Y_i > t] \geq Pr[X_i > t]$ for all real numbers t (see [18]). It follows that: $\mathbb{E}\{S_2\} \leq \mathbb{E}\{\max\{X_1, X_2, \ldots, X_M\}\} \leq 1 + \mathbb{E}\{\max\{Y_1, Y_2, \ldots, Y_M\}\}$

The expected maximum of M i.i.d. exponential variables of rate λ is equal to the expectation of the sum of intervals $I_1 + I_2 + \ldots + I_M$, where I_i represents the duration of time between the $(i-1)^{th}$ and i^{th} completion time. The interval I_1 is the first completion time of M independently racing exponential variables, and hence I_1 is exponentially distributed with rate $M\lambda$. Furthermore, I_2 is the first completion time of $M-1$ racing exponential variables, I_3 is the first completion time of $M-2$ racing exponentials, and so on. It follows that:

$$\mathbb{E}\{I_1 + I_2 + \ldots + I_M\} = \frac{1}{\lambda}\sum_{m=1}^{M}\frac{1}{m}$$

Hence, $\mathbb{E}\{S_2\} \leq 1 + \frac{1}{\lambda}\sum_{m=1}^{M}\frac{1}{m}$, which is upper bounded by $1 + \frac{1}{\lambda}(1 + \log(M))$. Hence:

$$\mathbb{E}\{S_2\} \leq 1 + \frac{1 + \log(M)}{\log(1/(1-\psi))} \leq 1 + \frac{1 + \log(N/2)}{\log(e^{d/2})} \qquad \square$$

References

[1] M. J. Neely. *Dynamic Power Allocation and Routing for Satellite and Wireless Networks with Time Varying Channels*. PhD thesis, Massachusetts Institute of Technology, LIDS, 2003.

[2] M. J. Neely and E. Modiano. Improving delay in ad-hoc mobile networks via redundant packet transfers. *Proc. of the Conference on Information Sciences and Systems*, Johns Hopkins University: March 2003.

[3] R. Cruz and A. Santhanam. Optimal routing, link scheduling, and power control in multi-hop wireless networks. *IEEE Proceedings of INFOCOM*, April 2003.

[4] M. J. Neely, E. Modiano, and C.E. Rohrs. Dynamic power allocation and routing for time varying wireless networks. *IEEE Proceedings of INFOCOM*, April 2003.

[5] M. Grossglauser and D. Tse. Mobility increases the capacity of ad-hoc wireless networks. *Proceedings of IEEE INFOCOM*, 2001.

[6] P. Gupta and P.R. Kumar. Critical power for asymptotic connectivity in wireless networks. *IEEE Conference on Decision and Control*, 1998.

[7] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, Vol. 46:388–404, March 2000.

[8] N. Bansal and Z. Liu. Capacity, delay and mobility in wireless ad-hoc networks. *IEEE Proceedings of INFOCOM*, April 2003.

[9] M. Grossglauser and M. Vetterli. Locating nodes with ease: Last encounter routing in ad hoc networks through mobility diffusion. *IEEE Proceedings of INFOCOM*, April 2003.

[10] E. Perevalov and R. Blum. Delay limited capacity of ad hoc networks: Asymptotically optimal transmission and relaying strategy. *IEEE Proceedings of INFOCOM*, April 2003.

[11] A. Tsirigos and Z. J. Haas. Multipath routing in the presence of frequent topological changes. *IEEE Communications Magazine*, Nov. 2001.

[12] E. Cohen and S. Shenker. Replication strategies in unstructured peer-to-peer networks. *ACM Proceedings of SIGCOMM*, August 2002.

[13] W. H. Yuen, R. D. Yates, and S-C Mau. Exploiting data diversity and multiuser diversity in noncooperative mobile infostation networks. *IEEE Proceedings of INFOCOM*, April 2003.

[14] L. Kleinrock and J. A. Silvester. Optimum transmission radii in packet radio networks or why six is a magic number. *Proceedings of the National Telecommunications Conference*, Dec. 1978.

[15] S. Toumpis and A. J. Goldsmith. Large wireless networks under fading, mobility, and delay constraints. *IEEE Proceddings of INFOCOM*, 2004.

[16] A. El Gammal, J. Mammen, B. Prabhakar, and D. Shah. Throughput-delay trade-off in wireless networks. *IEEE Proc. of INFOCOM*, 2004.

[17] X. Lin and N. B. Shroff. The fundamental capacity-delay tradeoff in large mobile ad hoc networks. *Purdue University Tech. Report*, 2004.

[18] S. Ross. *Stochastic Processes*. John Wiley & Sons, Inc., New York, 1996.

[19] R. Gallager. *Discrete Stochastic Processes*. Kluwer Academic Publishers, Boston, 1996.

[20] H. Daduna. *Queueing Networks with Discrete Time Scale*. Springer, 2001.