Balancing Distance and Lifetime in Delay Constrained Ad Hoc Networks

Ben Liang [*]
Dept of Electrical and Computer Engineering
University of Toronto
Toronto, Ontario, Canada
liang@comm.utoronto.ca

Min Dong
Corporate Research and Development
QUALCOMM, Inc.
San Diego, California, USA
mdong@qualcomm.com

ABSTRACT

This paper addresses the problem of optimizing the packet transmission schedule in an ad hoc network with end-to-end delay constraints. The emphasis is to determine the proper relative weights assigned to the remaining distance and the remaining lifetime in order to rank the urgency of a packet. We consider a general class of transmission schemes that represent such relative weights using a single lifetime-distance factor, which includes, as special cases, schedules such as Earliest-Deadline-First and Largest-Distance-First. We propose an analytical framework, based on recursive non-homogeneous Markovian analysis, to study the effect of the lifetime-distance factor on packet loss probability in a general multihop environment, with different configurations of peer-node channel contention. Numerical results are presented to demonstrate how various network parameters affect the optimal lifetime-distance factor. We demonstrate quantitatively how the proper balance between distance and lifetime in a transmission schedule can significantly improve the network performance, even under imperfect schedule implementation.

Categories and Subject Descriptors

C.2.1 [**Computer Communication Networks**]: Network Architecture and Design—*packet-switching networks, wireless communication*; G.3 [**Probability and Statistics**]

General Terms

Algorithms, Design, Performance, Theory

Keywords

Multihop packet scheduling, cross-layer design, delay constraint, stochastic analysis

1. INTRODUCTION

As multimedia applications become more prevalent in wireless networks, it remains a challenging problem to ensure the reliable transmission of multimedia data across multiple hops of wireless links [28, 14]. In particular, an important criterion for the successful decoding of multimedia streams is the end-to-end transmission delay of packets [18]. Typically, if a multimedia streaming receiver has not completely received an audio or video frame by its pre-defined playback deadline, the frame is obsolete. Thus, multimedia communication imposes stringent requirements on the delay threshold of packet transmission.

The efficient operation of ad hoc networks requires flexibility and adaptation across the entire protocol stack. Recent empirical data have shown that the performance of one protocol layer in an ad hoc network is closely related to the time-varying characteristics of the network at the other layers [6, 31]. Therefore, the cross-layer approach to ad hoc network design, which allows the jointly adaptive optimization of the protocol layers, is important to the successful deployment of future large-scale multimedia ad hoc networks.

For multimedia applications in ad hoc networks, a major contribution to the overall transmission delay is at the Medium Access Control (MAC) layer, due to distributed multi-node contention over radio channels. However, the most commonly employed MAC layer protocols for ad hoc networks [14] are not designed based on the multihop end-to-end delay of a packet. Examples of these protocols include RI-BTMA [33], MACA[19], MACAW[7], FAMA [11], IEEE 802.11 DCF [1], and DBTMA [13]. All of them mainly concern the packet transmission within a confined local neighborhood, without considering the accumulated effects over the entire route traversed by a packet.

In this paper, we consider the problem of optimizing the packet transmission schedule of an ad hoc network, through MAC-layer scheduling while using routing and application information from the upper layers. The ultimate goal is to minimize the probability of packet loss due to excessive end-to-end delay. We emphasize on finding the proper assignment of *relative weights* to the remaining distance and the remaining lifetime in ranking the urgency of a packet.

The relative importance of the remaining distance and the remaining lifetime can be summarized in a single *lifetime-distance factor*. We term scheduling scheme that uses the lifetime-distance factor to rank the transmission priority of packets Multihop Latency Aware (MLA) scheduling . It is a general class of schedules that contains several well-known schemes, including for example the Earliest-Deadline-First

(EDF) schedule and the Longest-Distance-First (LDF) schedule as special cases.

Hence, the central question we attempt to answer is *how to determine the optimal lifetime-distance factor in a multihop environment.* Toward this end, we propose an analytical framework, based on recursive non-homogeneous Markovian analysis, to evaluate quantitatively the performance of rank-based scheduling and study the optimization of the lifetime-distance factor under different network parameters. Furthermore, we investigate the effects of imperfect schedule implementation on the optimal balance between distance and lifetime.

The rest of this paper is organized as follows. In Section 2, we summarize prior work on distributed and multihop scheduling. In Section 3, we give detailed description of the network model and lifetime-distance factor-based scheduling. In Section 4, we provide a numerical analysis framework for computing the effect of the lifetime-distance factor on the probability of packet loss. In Section 5, we validate the analysis results and study schedule optimization given various system parameters, including the effects of imperfect schedule implementation. Finally, concluding remarks are given in Section 6.

2. RELATED WORK

Much existing work on distributed scheduling in the ad hoc networking environment focuses on ensuring fair access to the shared medium [5, 26, 24, 25, 32, 16, 4, 30, 17]. However, there is comparatively less work on scheduling to ensure end-to-end delay for ad hoc networks.

Solutions were provided in [5, 17] to implementing distributed multiple-access schemes that approximate any given schedule. These methods do not directly concern the end-to-end delay guarantee. However, they can be employed to implement a rank-based scheduling scheme, such as the MLA schedule.

In [16], a priority index based multihop coordinated scheduling scheme was proposed for ad hoc networks, with three types of index assignments. In particular, as seen in Section 3, its Time To Live assignment scheme can be considered within the general class of MLA schedules that neglects the effect of hop count. Its Uniform Delay Budget scheme is similar to a special case of MLA giving lifetime and hop count the exact same weight, but this scheme considers the *average* lifetime allowance per hop, instead of the *remaining* lifetime.

Multihop scheduling was considered in wireline networks to ensure end-to-end Quality-of-Service bounds [23, 20, 8, 2, 21]. The most commonly studied scheduling schemes are First-In-First-Out (FIFO), where the first packet to enter a transmission queue is given the highest priority, *Global* Earliest-Deadline-First (EDF), where the packet with the least remaining lifetime is given the highest priority, and Longest-Distance-First (LDF), where the packet with the largest remaining hop count is given the highest priority. These schedules are within the general class of MLA scheduling. As will be seen later, under most general conditions, these schedules are out performed by a more balanced schedule with an optimal lifetime-distance factor assigning the proper relative weights to the remaining distance and the remaining lifetime of a packet.

3. NETWORK MODEL AND THE LIFETIME-DISTANCE FACTOR

We consider a cross-layer [6, 31] ad hoc network design where the MAC layer is aware of the routing information and application requirement of contending packets. We assume the scheduling of each packet takes into account the remaining distance in hops, denoted H, from the current location of the packet to its destination, and the remaining lifetime, denoted T, of the packet. Each packet is associated with a ranking function $\gamma(H, T)$, such that a smaller value of $\gamma(H, T)$ denotes higher transmission priority.

Within each node, the values of H and T of all packets can be made available to the MAC layer through cross-layer information exchange. The hop count information is recorded in most of the routing protocols proposed for ad hoc networks [28, 14]. For example, all link-state, distance-vector, and source-routing protocols require that a node stores the exact hop count from itself to all active destination nodes. Furthermore, each packet can be assigned an expiry time based on the delay threshold set by the application [18]. We assume that this information is embedded within each packet, so that a node can compare this value with the current time to compute the remaining lifetime of the packet.[1] Thus, within each node, the packets are queued and ranked based on $\gamma(H, T)$, such that the head-of-line (HoL) packet has the highest transmission priority.

Between nodes, each HoL packet competes for access to the shared medium with all other HoL packets in its *contention area.* For IEEE 802.11-typed MAC, the contention area of a node may be defined as its two-hop neighborhood [28, 14]. However, in general, the contention area depends on many factors, such as the multiple access algorithm, signal modulation scheme, and receiver hardware design. In this work, we consider different types of contention area configurations and propose a general analytical framework for performance evaluation. We assume that the nodes within each other's contention area can exchange the H and T values of the contending HoL packets through short control messages. These control messages may be piggy-backed within the previously transmitted data packets [5], and, in the case of IEEE 802.11 compliant protocols, within the RTS/CTS/ACK control packets [17].

Clearly, the less remaining lifetime a packet has and the more hops it has to traverse, the more urgent the packet is. Therefore, the relative importance of these two quantities can be summarized in a *lifetime-distance factor*, denoted α, as follows:

$$\gamma(H, T) = \frac{T^\alpha}{H} \ , \tag{1}$$

where α can take any non-negative value. We study the optimization of α to minimize the probability of packet loss due to excessive end-to-end delay.

Note that (1) represents a family of ranking functions that also include any function of the form $\frac{T^a}{H^b}$, where $a \geq 0$ and $b > 0$. To see this, we let $\alpha = \frac{a}{b}$. Then, it is clear that,

---

[1]This is easily achievable if all nodes have synchronized clocks; distributed clock synchronization protocols for multihop wireless networks was proposed in [10, 29]. Alternatively, if the remaining lifetime of a packet is embedded within the packet, upon its transmission, the transmitting node can update this value by subtracting from it the queuing delay.

given any (H_1, T_1) and (H_2, T_2) such that $\frac{T_1{}^a}{H_1{}^b} < \frac{T_2{}^a}{H_2{}^b}$, we have $\gamma(H_1, T_1) < \gamma(H_2, T_2)$, and vice versa, i.e., $\gamma(H, T)$ gives the same ordering of packets as $\frac{T^a}{H^b}$ does. Therefore, α can be viewed as a single factor that represents the relative weights of the remaining distance and the remaining lifetime in ranking the urgency of a packet. Furthermore, (1) clearly defines a general class of ranking functions that includes many well-known scheduling schemes. For example, the Uniform Delay Budget scheme in [16] is similar to a special case of MLA with $\alpha = 1$, and the EDF and LDF schemes are special cases of MLA with $\alpha = \infty$ and $\alpha = 0$, respectively[2].

We term the general class of rank-based scheduling schemes using (1) Multihop Latency Aware schedules. Our goal in this paper is to determine the proper value for the lifetime-distance factor, so as to achieve optimal balance between the remaining distance and the remaining lifetime in ranking packets.

4. PERFORMANCE ANALYSIS OF MLA SCHEDULES

In this section, we propose a recursive analytical framework to study the effect of the lifetime-distance factor on the performance of multihop scheduling. We are particularly interested in computing p_{loss}, the probability of packet loss due to insufficient remaining lifetime.

4.1 Analysis Model

We consider a network where the transmission time of each packet is constant. Furthermore, we discretize and normalize all time durations such that the transmission time represents one time unit. Table 1 lists the notations used throughout the rest of this section.

We assume that, in the source node, a multimedia application sends out data packets encoded with packet expiry time. The packet is processed by the network layer, encapsulating it with routing information, including the hop count to its destination node. For a packet entering the transmission queue of the source node, we denote its initial hop count H_0 and initial lifetime T_0, with a general distribution $P_{H_0 T_0}(h, t)$.

The packet is then forwarded successively along the nodes within the predefined route, where its remaining hop count H and remaining lifetime T are updated. At each hop along this route, the packet's ranking function $\gamma(H, T)$, as defined in (1), is compared with the ranks of all packets within the same node. The HoL packet with the highest rank (i.e., smallest $\gamma(H, T)$) within the contention area is scheduled to be transmitted immediately. We assume ties are broken randomly.

A node periodically inspects the remaining life time of all packets within its transmission queue. The node discards any packet that cannot arrive at its destination before its expiry time. This includes all packets that have $T < H$. Note that, since a packet's remaining lifetime decreases as the packet waits in the transmission queue, a packet can be expunged during its waiting time, even though it had adequate residual lifetime when it first arrived. In the analysis below, we assume that the network load is moderate such that the amount of discarded packets is negligible. In

Section 5.1, we study the effect of this assumption on the accuracy of our analysis, and present simulation results for a wide range of network load[3].

We assume that new packets are generated independently at each node with arrival rate λ_s. To allow tractable analysis, we further approximate the network-queue length distribution by assuming that packet arrivals into the contention area of a node form a Poisson stream with arrival rate λ. The value of λ can be estimated from λ_s or measured in practice. The Poisson assumption is inaccurate in general, but its value of approximation can be justified due to random delays incurred under multiple-node contention and the random choice of next-hop node to which a packet is forwarded. In Section 5.1, we compare our analytical results against simulation results obtained without this assumption.

4.2 Overview of Analytical Framework

Let H_e and T_e represent the remaining hop count and the remaining lifetime of a packet when it enters the transmission queue of a source or intermediate node. We denote their joint probability function $P_{H_e T_e}(h, t)$. Let $R(h, t)$ be the amount of delay that a packet experiences at a node before it is transmitted, given that the packet enters the node with $H_e = h$ and $T_e = t$. We denote its probability function $P_{R(h,t)}(r)$. The proposed analytical framework recursively computes the above statistics, which eventually leads to the probability of packet loss. It has three components:

1. A recursive algorithm to compute $P_{R(h,t)}(r)$ given $P_{H_e T_e}(h, t)$.

2. Recursive updating of $P_{H_e T_e}(h, t)$ given $P_{R(h,t)}(r)$.

3. Post convergence absorption analysis to compute p_{loss}.

The above recursions are guaranteed to converge by the ergodicity of non-homogeneous scrambling Markov chains [15, 9]. The details of these components and their convergence are presented in the following subsections.

4.3 Contention Area

The transmission probability of a packet depends on the number of other packets that are contending for the same transmission medium. For a network where packet transmissions between different nodes do not interfere with each other, a packet contends with only the packets in the transmission queue within the same node. In a general multi-hop wireless network, however, multiple nodes that are near each other may contend for the same radio channel.

Figure 1 illustrates examples of different contention area configurations in a network where the nodes are placed on a grid. The dotted lines represent potential communication links. Figure 1(a) shows *overlapping* contention areas, where each node contends with nodes within its two-hop neighborhood. In this example, there are $n_c = 13$ nodes in each contention area. Figure 1(b) shows *non-overlapping* contention areas. In this case, we assume that all nodes within a contention area share the same transmission channel and

---

[2]The ranking function with $\alpha = \infty$ can be implemented as $a = 1$ and $b = 0$.

[3]Note that for highly congested networks, where many packets may be dropped, a selective admission control protocol may be necessary, instead of attempting to transmit all packets. The optimal combination of MLA schedules with an admission control protocol is outside the scope of this paper and remains an open problem for future research.

p_{loss}:	packet loss probability
$\gamma(h,t)$:	ranking function of an (h,t) packet
α, a, b:	MLA ranking parameters, $\alpha = \frac{a}{b}$
λ_s:	per node source packet generation rate
λ:	rate of packets into contention area
n_c:	number of nodes in a contention area
σ:	deviation from perfect scheduling
h_{max}:	maximum initial hop count of a packet
t_{max}:	maximum initial lifetime of a packet
H_0, T_0:	initial hop count and lifetime of a packet
$P_{H_0 T_0}(h,t)$:	distribution of H_0 and T_0
H_e, T_e:	remaining hop count and lifetime of an arriving packet
$P_{H_e T_e}(h,t)$:	distribution of H_e and T_e
$R(h,t)$:	amount of delay until transmission of an arriving packet with $H_e = h$ and $T_e = t$
$P_{R(h,t)}(r)$:	distribution of $R(h,t)$
$p_x(h,t)$:	probability that a packet with (h,t) is transmitted
N:	number of contending packets
$P_N(n)$:	distribution of N
$P_N{}^*(z)$:	probability generating function of N
M:	number of packets with higher rank
$p_g(h,t)$	probability that a contending packet has higher rank than $\gamma(h,t)$
U:	number of packets with equal rank
$p_e(h,t)$	probability that a contending packet has rank equal to $\gamma(h,t)$
H_q, T_q:	remaining hop count and lifetime of a contending packet
$P_{H_q T_q}(h,t)$:	distribution of H_q and T_q
$p_s(h,t)$:	probability that a packet in state (h,t) is eventually transmitted to its destination
$p_a(h,t)$:	probability that a packet in state (h,t) is transmitted to its destination in the next step
$\mathbf{p}_e, \mathbf{p}_q$:	vector versions of $P_{H_e T_e}(h,t)$ and $P_{H_q T_q}(h,t)$
$\mathbf{P}_e, \mathbf{P}_q$:	transition probability matrices of Markov chains to compute $\mathbf{p}_e$ and $\mathbf{p}_q$
$\mathbf{p}_s, \mathbf{p}_a$:	vector versions of $p_s(h,t)$ and $p_a(h,t)$
$\mathbf{P}_s$:	transient part of transition probability matrix of Markov chain to compute $\mathbf{p}_s$

Table 1: Table of nomenclature
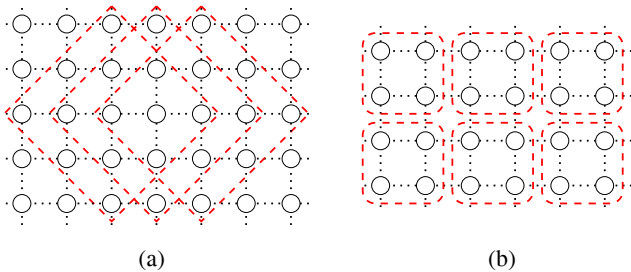

(a)   (b)

Figure 1: Contention area configurations: (a) overlapping, $n_c = 13$; (b) non-overlapping, $n_c = 4$.

the transmission channels of any two contention areas are independent, e.g., via different frequencies or codes [3].

The non-overlapping case can be further divided into two subcases, *distributed access* and *centralized access*. In a distributed access scheme, the nodes within a contention area forward packets in a peer-to-peer manner. In a centralized access scheme, a central controller (e.g., clusterhead) resides in each contention area and coordinates packet forwarding from one contention area to another [22]. Hence, each contention area in the centralized access scheme can be viewed as a super-node, and the definition of a hop can be generalized to the forwarding of a packet from one central controller to another. This is equivalent to the case of $n_c = 1$, with no contention between nodes.

Thus, the proposed analysis framework is applicable to a wide range of contention configurations, from overlapping contention areas to local zone-based medium access. Furthermore, it considers the special case where there is no contention between nodes, which can be used to model multihop wireline networks and mesh networks with directional antennae.

4.4 Number of Contending Packets

Let N be the number of packets waiting to be transmitted within the contention area of a node. Let $P_N(n)$ be its probability function and $P_N{}^*(z)$ be its probability generating function. We present next a method to estimate the statistics of N.

Since each packet is forwarded H_0 hops, the total packet arrival rate per contention area is $\lambda_s n_c + \lambda_s E[H_0] n_c$. However, some of these packet arrivals are from nodes in the same contention area and should be discounted. Let e_{in} be the number of links between nodes within the contention area and e_{out} be the number of links between nodes in the contention area and nodes outside of the contention area. Then the proportion of forwarded packets that come from outside of the contention area is

$$p_{ext} = \frac{e_{out}}{2e_{in} + e_{out}} . \qquad (2)$$

For example, in Figure 1(a), $p_{ext} = \frac{5}{13}$ for $n_c = 1$, and in Figure 1(b), $p_{ext} = \frac{1}{2}$ for distributed access *and* $n_c = 4$, and $p_{ext} = 1$ for centralized access *or* $n_c = 1$. Furthermore, the last-hop packets arriving at their destinations will not be transmitted again and hence do not contribute to the transmission queue of nodes in contention. Hence, the packet arrival rate into the contention area is

$$\lambda = \lambda_s n_c + \lambda_s p_{ext}(E[H_0] - 1)n_c . \qquad (3)$$

To model the packet departure process out of the contention area, we note that, for all contention area configurations discussed above, each node has transmission probability $\frac{1}{n_c}$ at any transmission time slot. Hence, the total rate of packet transmission in the contention area is 1. However, each transmitted packet departs from the contention area only with probability p_{ext}. Hence, the net service rate for packets within the contention area is p_{ext}.

Therefore, to compute the total number of queued packets in the contention area, we use an M/G/1 queue[4] with geometric service time at rate p_{ext}. Then, as a special case

---

[4]Note that the time-varying priorities of packets in this queue does not alter the queue length distribution.

of the Pollaczek-Khintchine formula [12], we have

$$P_N{}^*(z) = \frac{(p_{ext} - \lambda)(1-z)}{p_{ext} + (1-p_{ext})z - ze^{\lambda(1-z)}} \ , \qquad (4)$$

where the time to transmit a packet has been normalized to 1. Here, we have assumed that the amount of discarded packets is small, as explained in Section 4.1.

4.5   Computing $P_{R(h,t)}(r)$ Given $P_{H_e T_e}(h,t)$

To compute $P_{R(h,t)}(r)$, we first need to determine $p_x(h,t)$, the probability that a packet with h remaining hops and t remaining lifetime is transmitted at the current time step.

Given a packet of rank $\gamma(h,t)$, let M and U be the number of packets within the contention area that has greater rank and equal rank, respectively. The packet is transmitted from a node at the current time step if $M = 0$ and at the same time it is chosen among all U packets of the same rank. It is easy to show that, regardless in which nodes these U packets are distributed, the probability that any given one of them wins the random tie breaking is $\frac{1}{U}$. Therefore, the probability that the packet is transmitted at the current time step is

$$p_x(h,t) = \sum_{u=1}^{\infty} Pr\{M = 0, U = u | U \geq 1\}\frac{1}{u}$$
$$= \frac{1}{Pr\{U \geq 1\}} \sum_{u=1}^{\infty} Pr\{M = 0, U = u\}\frac{1}{u} \ . \qquad (5)$$

It can be shown that the z transform of $Pr\{M = 0, U = u\}$ taken over u is

$$Pr\{M = 0, U = u\} \overset{\mathcal{Z}}{\longleftrightarrow} P_N{}^*[1 - p_g(h,t) - p_e(h,t)(1-z)] \ , \qquad (6)$$

where $p_g(h,t)$ and $p_e(h,t)$ are the probabilities that a packet within the contention area has rank higher than $\gamma(h,t)$ and equal to $\gamma(h,t)$, respectively, i.e.,

$$p_g(h,t) = \sum_{\gamma(h',t') < \gamma(h,t)} P_{H_q T_q}(h',t')$$
$$p_e(h,t) = \sum_{\gamma(h',t') = \gamma(h,t)} P_{H_q T_q}(h',t') \ , \qquad (7)$$

where $P_{H_q T_q}(h,t)$ is the steady-state distribution of the remaining hop count and lifetime of a packet waiting to be transmitted within the contention area. Detailed derivation of the above is presented in the Appendix.

Furthermore, we have

$$Pr\{U \geq 1\} = 1 - \sum_{n=0}^{\infty}(1 - p_e(h,t))^n Pr\{N = n\}$$
$$= 1 - P_N{}^*[1 - p_e(h,t)] \ . \qquad (8)$$

Hence, the transmission probability is

$$p_x(h,t) = \frac{1}{1 - P_N{}^*[1 - p_e(h,t)]} \cdot$$
$$\sum_{u=1}^{\infty} \mathcal{Z}^{-1}\{P_N{}^*[1 - p_g(h,t) - p_e(h,t)(1-z)]\}\frac{1}{u} \ , \qquad (9)$$

where $\mathcal{Z}^{-1}$ denotes inverse z-transform. In general, the above inverse z-transform does not have a closed from solution. In this work, we compute its numerical solution by inverse discrete Fourier transform.

Next, we present a recursive algorithm to compute the contending packet state distribution $P_{H_q T_q}(h,t)$. Given the arriving packet state distribution $P_{H_e T_e}(h,t)$ and the probabilities $p_x(h,t)$, we can construct a Markov chain whose states consist all pairs of (h,t), for $h \leq t$, each representing the remaining hop count and remaining lifetime of any given packet as it awaits transmission within a node. Here, the Markovian model provides close approximation because of the random mixing effect of a quotient-form ranking function as h and t are reduced over time.

For a packet in state (h,t) where $t \geq h+1$, with probability $1 - p_x(h,t)$, it will not be transmitted in the current time step, and hence it will transit to new state $(h, t-1)$. With probability $p_x(h,t)$, it will be transmitted. Since, in equilibrium, the rate of packets entering the node equals the rate of packets leaving the node, we may simplify the Markov chain by constructing a model where, with probability $p_x(h,t)$, the packet is transmitted and a packet enters the system with state probability distribution[5] $P_{H_e T_e}(h,t)$. When a packet is in a state where $t = h$, if it is not transmitted immediately, it will be discarded since it has zero probability of arriving at the destination before its lifetime expires. Therefore, such packets always leave the system in the next step, whether it is transmitted or not. Thus, the transition probabilities of this Markov chain can be summarized as follows:

$$Pr\{(h,t) \to (h',t')\} =$$
$$\begin{cases} 1 - p_x(h,t) + p_x(h,t)P_{H_e T_e}(h',t') \ , \\ \qquad\qquad t \geq h+1, (h',t') = (h,t-1) \\ p_x(h,t)P_{H_e T_e}(h',t') \ , \quad t \geq h+1, (h',t') \neq (h,t-1) \\ P_{H_e T_e}(h',t') \ , \qquad\qquad t = h \ . \end{cases} \qquad (10)$$

Let $\mathbf{P}_q$ be the transition probability matrix of this Markov chain. Let $\mathbf{p}_q$ be the vector version of $P_{H_q T_q}(h,t)$. We need to solve $\mathbf{p}_q = \mathbf{p}_q \mathbf{P}_q$ for the steady-state distribution of $P_{H_q T_q}(h,t)$. However, as indicated in (7), $p_x(h,t)$ depends on $\mathbf{p}_q$, and, hence, $\mathbf{P}_q$ is a function of $\mathbf{p}_q$. Therefore, the standard Markov chain steady-state analysis can not be applied. Instead, the follow recursive algorithm on $P_{H_q T_q}{}^{(i)}(h,t)$ can be used:

> *Initialization:* $P_{H_q T_q}{}^{(0)}(h,t) = P_{H_e T_e}(h,t)$
> *Step 1:* Compute $p_x{}^{(i)}(h,t)$ following (7) and (9)
> *Step 2:* Construct $\mathbf{P}_q{}^{(i)}$ following (10)
> *Step 3:* Let $\mathbf{p}_q{}^{(i+1)} = \mathbf{p}_q{}^{(i)}\mathbf{P}_q{}^{(i)}$
> *Repeat from Step 1*

By the Markov chain construction in (10), ignoring the singleton closed sets $\{(h,t)|t < h\}$, the stochastic matrix $\mathbf{P}_q{}^{(i)}$ is *scrambling* for all i, i.e., for any two states j and k, there exists a state l, such that l is reachable from both j and k. Then, it can be shown that the sequence $\{\mathbf{P}_q{}^{(i)}\}$ is ergodic[15, 9]. Hence, the above recursion converges, in the limit to the steady-state distribution $P_{H_q T_q}(h,t)$, given the distribution $P_{H_e T_e}(h,t)$ in the current step.

Also in the above recursion, $p_x{}^{(i)}(h,t)$ converges to $p_x(h,t)$. Then, with the approximating assumption of independence

---

[5]Alternatively, we can create a new *reservoir* state $(0,0)$. All packets leaving the system go into $(0,0)$. While in $(0,0)$, the Markov chain next transits into (h,t) with probability distribution $P_{H_e T_e}(h,t)$. Clearly, this construction yields the same solution.

between transmission decisions on a packet over time, we can compute $P_{R(h,t)}(r)$ with the following recursion:

$$P_{R(h,t)}(1) = p_x(h,t) \ , \qquad (11)$$

and, for $2 \leq r \leq t - h + 1$,

$$P_{R(h,t)}(r) = \left[1 - \sum_{i=1}^{r-1} P_{R(h,t)}(i) \right] p_x(h, t - r + 1) \ . \quad (12)$$

Furthermore, if we allow the notation $P_{R(h,t)}(\infty)$ to represent the probability that the packet is discarded before it can be transmitted, we have

$$P_{R(h,t)}(\infty) = 1 - \sum_{i=1}^{t-h+1} P_{R(h,t)}(i) \ . \qquad (13)$$

4.6 Computing the Steady-State Distribution $P_{H_e T_e}(h,t)$

The previous section presents a method to determine $P_{R(h,t)}(r)$ given $P_{H_e T_e}(h,t)$. Next, we show how this can be used in turn to compute $P_{H_e T_e}(h,t)$.

Given the initial hop-count and lifetime distribution of a packet, denoted $P_{H_0 T_0}(h,t)$, and the delay distribution $P_{R(h,t)}(r)$, we can construct a Markov chain whose states consist all pairs of (h,t), for $h \leq t$, each representing the remaining hop count and remain lifetime of a packet *when it enters a source or intermediate node*. Suppose the state of a packet when it enters the current node is (h,t), where $h > 1$. With probability $P_{R(h,t)}(r)$, for all $1 \leq r \leq t - h + 1$, the packet will be transmitted after r time units and, hence, will enter the next node with state $(h - 1, t - r)$. With probability $P_{R(h,t)}(\infty)$, however, the packet will be discarded. In equilibrium, the rate of packets entering the system equals the rate of packets leaving the system. Therefore, with probability $P_{R(h,t)}(\infty)$, the Markov chain transits to a new state with probability distribution[6] $P_{H_0 T_0}(h,t)$. When a packet enters the node one hop away from its destination node, it is in state $(1,t)$. It is either discarded with probability $P_{R(1,t)}(\infty)$ or successfully transmitted to the destination node with probability $1 - P_{R(1,t)}(\infty)$. In either case, the packet will leave the system in the next state transition. Therefore, the transition probabilities of this Markov chain can be summarized as follows:

$$Pr\{(h,t) \rightarrow (h',t')\} =$$
$$\begin{cases} P_{R(h,t)}(r) + P_{R(h,t)}(\infty)P_{H_0 T_0}(h',t') \ , \\ \quad h > 1, t \geq h, (h',t') = (h-1, t-r), 1 \leq r \leq t-h+1 \\ P_{R(h,t)}(\infty)P_{H_0 T_0}(h',t') \ , \quad\quad h > 1, t \geq h, h' \neq h-1 \\ P_{H_0 T_0}(h',t') \ , \quad\quad\quad\quad\quad\quad\quad h = 1 \ . \end{cases}$$
$$(14)$$

Let $\mathbf{P}_e$ be the transition probability matrix of this Markov chain. Let $\mathbf{p}_e$ be the vector version of $P_{H_e T_e}(h,t)$. We need to solve $\mathbf{p}_e = \mathbf{p}_e \mathbf{P}_e$ for the steady-state distribution of $P_{H_e T_e}(h,t)$. However, from the last section, it is clear that $P_{R(h,t)}(r)$ depends on $\mathbf{p}_e$ and, hence, $\mathbf{P}_e$ is a function of $\mathbf{p}_e$. Therefore, again, the standard Markov chain steady-state analysis can not be applied. Instead, the follow recursive algorithm on $P_{H_e T_e}^{(i)}(h,t)$ can be used:

---

[6]This is equivalent to an alternative construction using a reservoir state. See Footnote 5.

> *Initialization:* $P_{H_e T_e}^{(0)}(h,t) = P_{H_0 T_0}(h,t)$
> *Step 1:* Compute $P_{R(h,t)}(r)$ as in Section 4.5
> *Step 2:* Construct $\mathbf{P}_e^{(i)}$ following (14)
> *Step 3:* Let $\mathbf{p}_e^{(i+1)} = \mathbf{p}_e^{(i)} \mathbf{P}_e^{(i)}$
> *Repeat from Step 1*

Similarly to the computation in Section 4.5, it can be shown that the stochastic matrix $\mathbf{P}_e^{(i)}$ is scrambling, and the above recursion converges. When convergence is reached, we obtain the steady-state distribution $P_{H_e T_e}(h,t)$ and the corresponding $P_{R(h,t)}(r)$.

4.7 Probability of Packet Loss

After the values of $P_{R(h,t)}(r)$ are computed, the probability of packet loss can be determined using a Markov chain similar to (14). In addition to the states (h,t), representing the remaining hop count and remaining lifetime of a packet when it enters a node, we introduce two absorbing states *success* and *loss*, which represent the cases of a packet being successfully transmitted to the destination and being lost due to insufficient remaining lifetime, respectively. When a packet arrives at its destination or is discarded, instead of modelling its re-entry into the system, we let the Markov chain go into one of the absorbing states. Hence, the new Markov chain is expressed by

$$\begin{cases} Pr\{(h,t) \rightarrow (h-1, t-r)\} = P_{R(h,t)}(r) \ , \\ \quad\quad\quad h > 1, t \geq h, 1 \leq r \leq t-h+1 \\ Pr\{(h,t) \rightarrow loss\} = P_{R(h,t)}(\infty) \ , \quad\quad t \geq h \\ Pr\{(1,t) \rightarrow success\} = 1 - P_{R(1,t)}(\infty), \quad t \geq h \ . \end{cases}$$
$$(15)$$

Since the values of $P_{R(h,t)}(r)$ are already given at this stage of the analysis, (15) represents a homogeneous Markov chain. Therefore, standard techniques can be applied to compute its absorption probabilities. In particular, let $\mathbf{P}_s$ be the transient part of the transition probability matrix of the Markov chain. Let $p_s(h,t)$ be the probability that a packet in state (h,t) is eventually successfully transmitted to its destination (i.e., absorbed into the *success* state), and let $\mathbf{p}_s$ be its vector version. Let $p_a(h,t) = Pr\{(h,t) \rightarrow success\}$ be the probability that a packet in state (h,t) is transmitted to its destination in the next step, and let $\mathbf{p}_a$ be its vector version. It can be shown that [27]

$$\mathbf{p}_s = (\mathbf{I} - \mathbf{P}_s)^{-1}\mathbf{p}_a \ , \qquad (16)$$

where $\mathbf{I}$ denotes the identity matrix.

Finally, the overall packet loss probability, given the initial packet hop-count and lifetime distribution $P_{H_0 T_0}(h,t)$, is given by

$$p_{loss} = \sum_{h,t} P_{H_0 T_0}(h,t)[1 - p_s(h,t)] \ . \qquad (17)$$

5. NUMERICAL ANALYSIS AND SIMULATION

In this section, we apply the proposed analysis framework to study how the relative weights assigned to the remaining distance and the remaining lifetime of a packet affect the packet loss performance in the class of MLA schedules as defined in (1). The analysis results are validated by comparison with simulation results. Through numerical analysis, we determine the optimal lifetime-distance factor α given various system parameters. We further demonstrate the
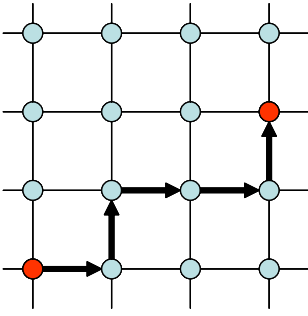
Figure 2: Simulated network on a grid.

performance gain achieved by optimizing α through comparison with systems employing LDF and EDF scheduling. In addition, we also study the effect of imperfect schedule implementation.

5.1 Simulation Model and Comparison

A simulation model has been developed in the OPNET simulator. In this model, an ad hoc network of 100 nodes are place on a square grid as shown in Figure 2, where a node has direct connection only with its four neighbors. The network area is wrapped top-town and left-right to form a border-less network. Data packets are routed randomly following the grid, such that the route length has a uniform distribution between one and $h_{max} = 10$ hops.

Each node generates data packets forming a Poisson stream. Note, however, that the overall data packets arriving at any contention area include both newly created ones and forwarded ones and hence are not Poisson. This contrasts with the queuing model in our analysis. As explained in Section 4 and illustrated in this section, in general the overall packet loss probability predicted by the proposed analysis is a reasonably close approximation for a wide range of system parameter values.

Each data packet is assigned a random lifetime, uniformly distributed between its route length and a maximum value t_{max}. A packet is lost and dropped from the transmission queue if its remaining lifetime becomes shorter than its remaining hop count. Otherwise, we assume an ideal physical layer is employed, so that no packet is lost during transmission.

To provide accurate insight into the effect of scheduling independent of the other system factors, we first assume perfect packet schedule implementation. Hence, a node has exact ranking information of all nodes within its contention area. The effect of imperfect packet scheduling is studied in Section 5.4.

In order to eliminate transient effects, a sequence of data collection windows are used such that on average each source node generates 100 data packets in each collection window. Data collection for each simulation run starts only after the system has stabilized, such that the transmission queues at each node are in steady state.

Figures 3 - 5 compare the simulation results with the analytical results, where we plot the packet loss probability against the contention-area external packet arrival rate λ, for $\alpha = 2$ and various values of t_{max}. Figure 3 is based on centralized non-overlapping contention, with $n_c = 1$, as in ad hoc networks with centralized hierarchical routing, mesh
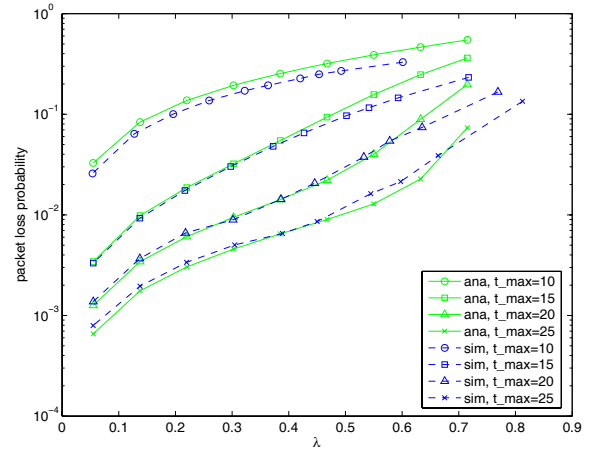

Figure 3: Comparison between analysis and simulation. Centralized non-overlapping contention areas.
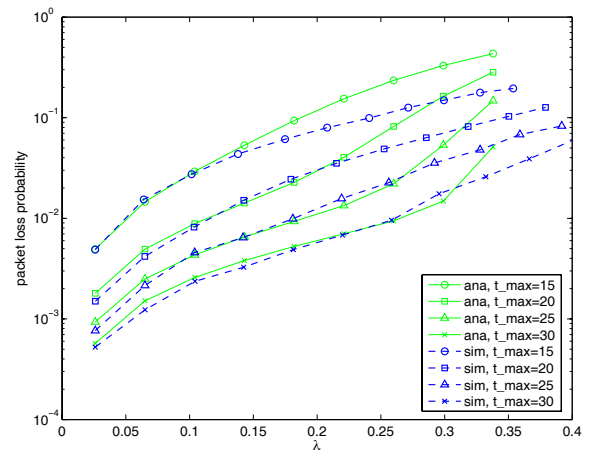

Figure 4: Comparison between analysis and simulation. Distributed non-overlapping contention areas, with $n_c = 4$.
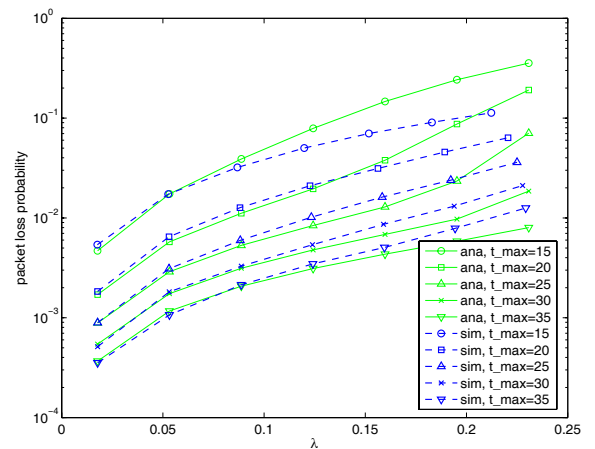

Figure 5: Comparison between analysis and simulation results. Two-hop overlapping contention areas.
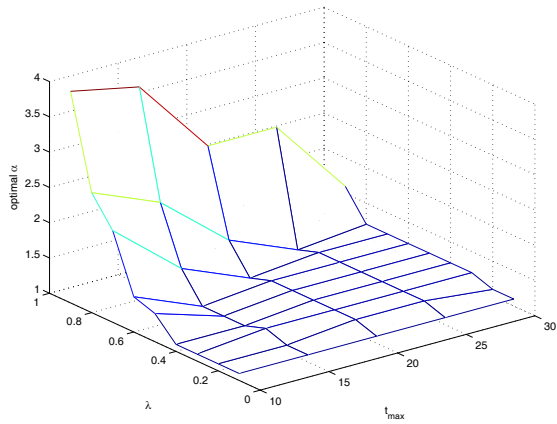
Figure 6: Optimal lifetime-distance factor, for different values of t_{max} and λ.

Figure 7: Probability of packet loss vs. λ, comparing optimal α against LDF and EDF.

networks with direction antennae, and multihop wireline networks. Figure 4 is based on distributed non-overlapping four-node contention as shown in Figure 1(b). Figure 5 is based on two-hop overlapping contention as shown in Figure 1(a).

We note that the analysis generally over estimates the packet loss probability at high traffic load. This is due to the pessimistic assumption in Section 4.5 that the transmission queue length N is not affected by the discarded packets. In reality, when packets are discarded, the transmission queue is shortened, leading to shorter delays and, hence, lower packet loss probability. This mismatch is the most pronounced when the network load is high. In fact, the simulation results suggest that by simply dropping some packets, the queuing system is stable under heavy load, while the analysis results based on a no-dropping policy predicts instability (noting the log-scale on the y-axis). Optimal packet dropping and selective admission control is outside the scope of this paper and remains a topic of future research interest.

For moderate packet dropping rates, the proposed analysis is accurate. Hence, we can use numerical analysis results to study the optimization of MLA schedules.

5.2 Optimal Values for the Lifetime-Distance Factor

We are interested in the probability of packet loss with ranking function (1) for different values of α, using the centralized non-overlapping contention configuration of Figure 3(b) as an example. We set $h_{max} = 10$. Given any source and destination node pair, the initial lifetime of a packet is randomly set such that it has uniform distribution between the hop count and t_{max}. For each set of parameter values for λ and t_{max}, we compute the probability of packet loss for a series of α values, ranging from 0.1 to 4. The optimal values of α, denoted α_{opt}, which minimizes the probability of packet loss, are presented in Figure 6. In these plots, t_{max} ranges from 10 to 30, and λ ranges from 0.1 to 0.9.

Figure 6 indicates that the optimal α increases as the traffic load increases, and it decreases as the initial packet lifetime increases. For example, with $t_{max} = 15$, when $\lambda = 0.4$, we have $\alpha_{opt} = 1.2$, and when the traffic load is increased to $\lambda = 0.8$, we have $\alpha_{opt} = 2.4$. With $\lambda = 0.5$, when $t_{max} = 10$, we have $\alpha_{opt} = 1.5$, and when the initial packet lifetime in-
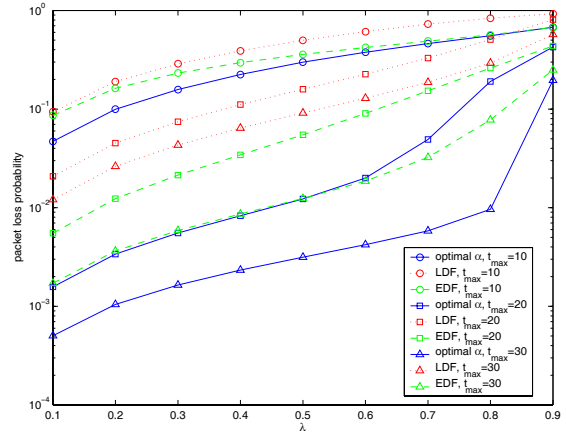
creases such that $t_{max} = 20$, we have $\alpha_{opt} = 1.3$.

The value of α represents the relative weight of the remaining distance and the remaining lifetime in ranking the urgency of packets. The above suggests that the remaining hop count is a more important factor (i.e., α should be small) when the network is lightly loaded and when packets have long lifetimes. Likewise, the remaining lifetime is a more important factor (i.e., α should be large) when the network is heavily loaded and when packets have short lifetimes. In particular, in the extreme case where the networks is highly stressed, $\alpha_{opt} = \infty$ (i.e., the EDF scheduling scheme) should be used. This matches the observed optimality of EDF in previous literature. We further note that, for a wide range of system parameter values, the optimal α is between 1.2 and 1.5.

5.3 Performance of Optimized Schedule

Next, we demonstrate the performance gain of the MLA schedule using optimal α values, against the LDF (i.e., $\alpha = 0$) and EDF (i.e., $\alpha = \infty$) schemes, where either only the remaining hop count or only the remaining lifetime is used in ranking packets[7]. Figures 7 and 8 present the probability of packet loss with the optimal α, and compare it with those of the LDF and EDF schemes, where the same ranges of values for λ and t_{max} as in Section 5.2 are used, but only the data points for $t_{max} = 10, 20, 30$ and $\lambda = 0.2, 0.4$ are shown, respectively.

These figures demonstrate that the optimized MLA schedule always out performs LDF and EDF. In general, the performance gain is consistently high until the network traffic load, represented by λ, becomes nearly full. Also, it decreases slowly as the packet lifetime, represented by t_{max}, decreases. This matches the observations made in Section 5.2.

For a wide range of system parameters, the performance gain is significant. For example, in the cases where $t_{max} = 20$ and $\lambda = 0.5$, the probabilities of packet loss in LDF, EDF, and the optimized MLA, are 0.16, 0.055, and 0.012, respectively. In this case, the amount of loss packets in

---

[7]Previous studies have shown that EDF significantly out performs FIFO in general. Therefore, the comparison results with FIFO is omitted.
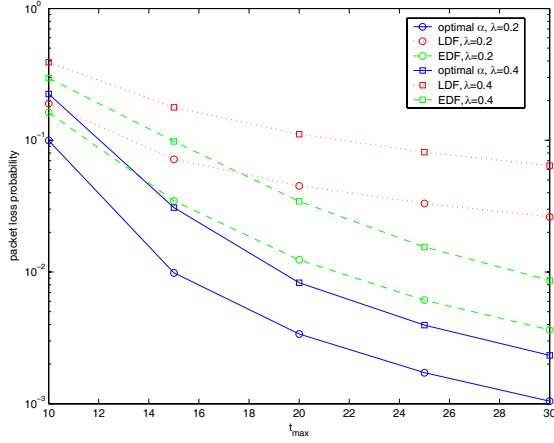
Figure 8: **Probability of packet loss vs. t_{max}, comparing optimal α against LDF and EDF.**

Figure 9: **Probability of packet loss vs. σ, for $h_{max} = 10$, $t_{max} = 20$, and $\lambda = 0.4$.**

optimized MLA are 22% of that in EDF and only 7.5% of that in LDF.

5.4 Imperfect Schedule Implementation

Due to the distributive nature of ad hoc networks, it is usually impossible to achieve perfect scheduling. The reference [17] gives quantitative details on the imperfection of distributed packet scheduling through RTS/CTS/ACK control messages. Instead of repeating those results, in this section, we study the effect of imperfect schedule implementation on the performance and optimization of scheduling. For comparison, we use the same distributions of packet initial hop count and initial lifetime as in Section 5.2.

We capture the level of imperfection in schedule implementation with a schedule deviation parameter σ as follows. Let $R_0(h, t) = r_0$ be the amount of delay of an arriving packet if the schedule implementation were ideal. Then the actual amount of delay $R(h, t)$ is random around r_0, and its distribution depends on the schedule implementation details [5, 17]. Here, for the purpose of illustration, we assume a truncated normal distribution such that

$$P_{R(h,t)}(r) = \frac{e^{-\frac{(r-r_0)^2}{2\sigma^2}}}{\sum_{i=1}^{\infty} e^{-\frac{(i-r_0)^2}{2\sigma^2}}}, \quad r = 1, 2, 3, \dots \quad (18)$$

Figure 9 shows the probability of packet loss as a function of σ, for $h_{max} = 10$, $t_{max} = 20$, and $\lambda = 0.4$, and α is assigned a value among 0, 1, 2, and ∞. Note that the cases of $\alpha = 0$ and $\alpha = \infty$ are equivalent to LDF and EDF, respectively. For reference, we note that the σ values of 0.2, 0.5, and 1 represent the cases where the actual scheduling of a packet is the same as that given in the ideal schedule, for approximately 99%, 80%, and 40% of the time, respectively.

This figure suggests that imperfect schedule implementation can significantly degrade the system performance. For example, when σ increases from 0.2 to 0.6, for all cases of α, the packet loss probability increases faster than exponentially in σ. For large scheduling deviation (e.g., $\sigma > 1$) the performance of all schemes converges, since in this case the packet schedule is essentially random regardless of the intended scheduling scheme.

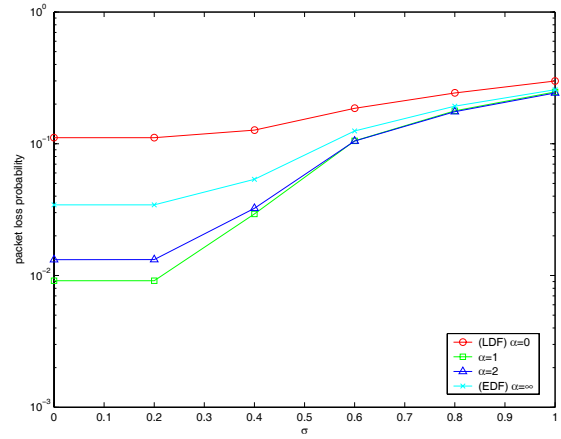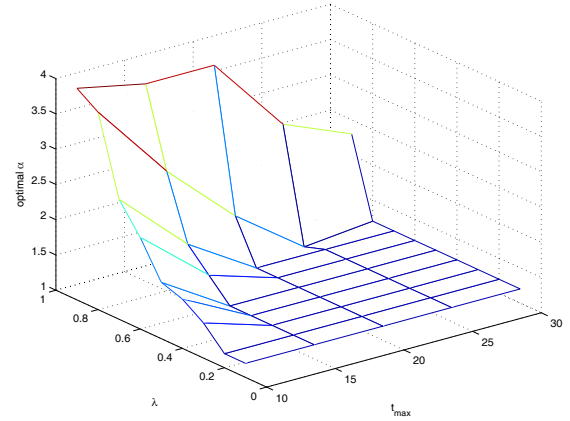We further study the effect of moderate scheduling devi-

Figure 10: **Optimal lifetime-distance factor, for different values of t_{max} and λ, with imperfect schedule implementation ($\sigma = 0.5$).**
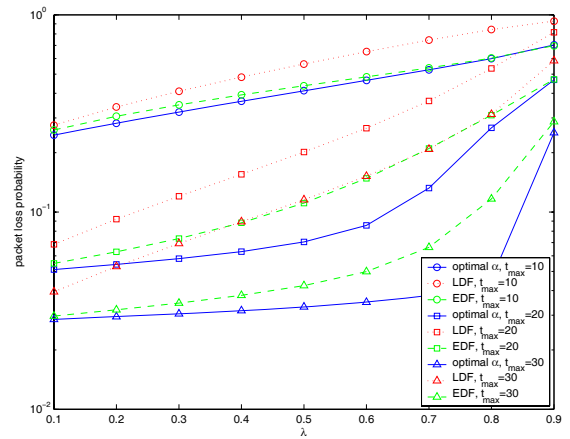

Figure 11: **Probability of packet loss vs. λ, comparing optimal α against LDF and EDF, with imperfect schedule implementation ($\sigma = 0.5$).**
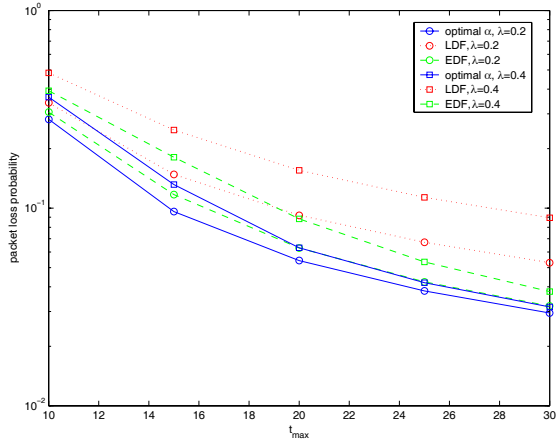
Figure 12: Probability of packet loss vs. t_{max}, comparing optimal α against LDF and EDF, with imperfect schedule implementation ($\sigma = 0.5$).

ation on the optimal value of α. Figures 10-12 correspond to Figures 6-8, with the same system parameters, except $\sigma = 0.5$, i.e., for 20% of the time, the actual scheduling of a packet is not as given by the intended schedule.

Figure 10 shows that, with imperfect schedule implementation, the optimal value of α is slightly larger than that with perfect scheduling. For example, with $t_{max} = 15$ and $\lambda = 0.4$, we have $\alpha_{opt} = 1.3$, compared with $\alpha_{opt} = 1.2$ as shown in Figure 6. We have observed in Section 5.2 that α_{opt} increases as the stress level of the network increases (i.e., having heavier traffic load and shorter packet lifetime). The results shown in Figure 10 agree with this observation, since imperfect schedule implementation increases packet delay, which contributes to the overall stress level of the network.

Figures 11 and 12 demonstrate that scheduling with the optimal α can still significantly outperform LDF and EDF even when the schedule implementation is imperfect. However, as expected from the trends in Figure 9, the performance gain may be much reduced. For comparison, we again take $t_{max} = 20$ and $\lambda = 0.5$. In this case, the probabilities of packet loss in LDF, EDF, and the optimized MLA, are 0.20, 0.11, and 0.071, respectively. Thus, the amount of loss packets in optimized MLA are 65% of that amount in EDF and 36% of that amount in LDF.

Furthermore, the performance gain reduction is the most severe when λ is small or when t_{max} is large. These are also the cases where, under perfect scheduling, the performance gain is the largest. Hence, imperfect schedule implementation, even when the scheduling deviation is moderate (i.e., $\sigma = 0.5$), can effectively undermine the advantage of optimal scheduling. This underlines the significance of designing robust schedule implementation schemes for ad hoc networks.

6. CONCLUSIONS

In this paper, we have considered the optimization of a lifetime-distance factor, α, which represents the relative weights assigned to the remaining distance and the remaining lifetime of a packet to determine its priority in a multihop transmission schedule. The optimal α minimizes, over all MLA schedules, the probability of packet loss due to excessive end-to-end delay. Our investigation is based on a nu-

merical analysis framework utilizing a double recursive construction of scrambling non-homogeneous Markov chains. The numerical evaluation results are shown to provide accurate representation of actual system performance when the packet loss probability is moderate. Furthermore, the proposed analysis framework is applicable to a wide range of overlapping and non-overlapping contention area configurations in ad hoc networks, mesh networks, and multihop wireline networks.

Our numerical results show that the remaining distance is a more important factor (i.e., α should be small) when the network is lightly loaded and when packets have long lifetimes. Likewise, the remaining lifetime is a more important factor (i.e., α should be large) when the network is heavily loaded and when packets have short lifetimes. We also observe that, for a wide range of system parameter values, the optimal α differs from common heuristics.

Finally, our study demonstrates quantitatively that optimizing the lifetime-distance factor can lead to significant performance gain, under low to moderate levels of imperfect schedule implementation. However, scheduling inaccuracy in ad hoc networks can considerably degrade the performance of a scheduling scheme. This suggests the importance of robust schedule implementation in future multihop wireless multimedia networks.

7. ACKNOWLEDGEMENTS

8. REFERENCES

[1] *IEEE Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.* IEEE Standard 802.11, Jun 1999.

[2] M. Andrews and L. Zhang. Minimizing end-to-end delay in high-speed networks with a simple coordinated schedule. In *Proc. IEEE INFOCOM*, 1999.

[3] D. J. Baker, A. Ephremides, and J. A. Flynn. The architectural organization of a mobile radio network with distributed control. *IEEE J. Select. Areas Commun.*, SAC-2:226–237, January 1984.

[4] L. Bao and J. Garcia-Luna-Aceves. A new approach to channel access scheduling for ad hoc networks. In *Proc. of the ACM MOBICOM*, pages 210 – 221, 2001.

[5] C. Barrack and K.-Y. Siu. A distributed scheduling algorithm for quality of service support in multiaccess networks. In *Proc. of IEEE ICNP*, 1999.

[6] C. Barrack, K.-Y. Siu, A. Marathe, and M. V. Marathe. Characterizing the interaction between routing and MAC protocols in ad-hoc networks. In *Proc. ACM MOBIHOC*, pages 92–103, 2002.

[7] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang. MACAW: A media access protocol for wireless LAN's. In *Proc. ACM SIGCOMM*, pages 212–225, 1994.

[8] F. M. Chiussi and V. Sivaraman. Achieving high utilization in guaranteed services networks using early-deadline-first scheduling. In *Proc. IEEE IWQoS*, 1998.

[9] I. Daubechies and J. C. Lagarias. Sets of matrices all infinite products of which converge. *Linear Algebra and Its Applications*, 161:227–263, 1992.

[10] J. Elson and D. Estrin. Time synchronization for wireless sensor networks. In *Proc. of the International Parallel and Distributed Processing Symposium (IPDPS)*, pages 186–191, 2001.

[11] C. Fullmer and J. Garcia-Luna-Aceves. Floor acquisition multiple access (FAMA) for packet-radio networks. In *Proc. ACM SIGCOMM*, 1995.

[12] D. Gross and C. M. Harris. *Fundamentals of Queueing Theory, 3rd ed.* John Wiley& Sons, 1998.

[13] Z. J. Haas and J. Deng. Dual busy tone multiple access (DBTMA) - a multiple access control scheme for ad hoc networks. *IEEE Trans. on Communications*, 50(6):975–985, Jun 2002.

[14] Z. J. Haas and et al. Wireless ad hoc networks. In J. Proakis, editor, *Wiley Encyclopedia of Telecommunications*. John Wiley & Sons, 2002.

[15] J. Hajnal. Weak ergodicity in nonhomogeneous markov chains. In *Proc. Cambridge Philosophical Society*, volume 54, pages 233–246, 1958.

[16] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly. Distributed multi-hop scheduling and medium access with delay and throughput constraints. In *Proc. ACM MOBICOM*, pages 200–209, 2001.

[17] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly. Ordered packet scheduling in wireless ad hoc networks: mechanisms and performance analysis. In *Proc. ACM MOBIHOC*, 2002.

[18] G. Karlsson. Asynchronous transfer of video. *IEEE Communications Magazine*, 34:106–113, Feb 1996.

[19] P. Karn. MACA - a new channel access method for packet radio. In *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, 1990.

[20] S. Keshav. *An Engineering Approach to Computer Networking*. Addison-Wesley, 1997.

[21] C. Li and E. W. Knightly. Coordinated multihop scheduling: a framework for end-to-end services. *IEEE/ACM Trans. Net.*, 10(6), Dec 2002.

[22] B. Liang and Z. J. Haas. Hybrid routing in ad hoc networks with a dynamic virtual backbone. In press, *IEEE Trans. Wireless Comm.*, preliminary version in *Proc. IEEE INFOCOM*, 2000.

[23] J. Liebeherr, D. E. Wrege, and D. Ferrari. Exact admission control for networks with a bounded delay service. *IEEE/ACM Trans. Net.*, 4(6), Dec 1996.

[24] H. Luo, S. Lu, and V. Bharghavan. A new model for packet scheduling in multihop wireless networks. In *Proc. ACM MOBICOM*, 2000.

[25] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan. Achieving MAC layer fairness in wireless packet networks. In *Proc. ACM MOBICOM*, 2000.

[26] T. Ozugur, M. Naghshineh, P. Kermani, and J. Copeland. Fair media access for wireless LANs. In *Proc. IEEE Globecom*, 1999.

[27] A. Papoulis and S. U. Pillai. *Probability, Random Variables, and Stochastic Processes*. Fourth Edition, McGraw-Hill, 2002.

[28] C. E. Perkins, editor. *Ad Hoc Networking*. Addison-Wesley Longman, 2001.

[29] K. Romer. Time synchronization in ad hoc networks. In *Proc. of ACM MOBIHOC*, pages 173–182, 2001.

[30] R. Rozovsky and P. Kumar. SEEDEX: a MAC protocol for ad hoc networks. In *Proc. ACM MOBIHOC*, pages 67–75, 2001.

[31] K. K. Vadde and V. R. Syrotiuk. Factor interaction on service delivery in mobile ad hoc networks. *IEEE J. Selected Areas in Communications*, 22(7):1335–1346, Sep 2004.

[32] N. Vaidya, P. Bahl, and S. Gupta. Distributed fair scheduling in a wireless LAN. In *Proc. ACM MOBICOM*, 2000.

[33] C. Wu and V. O. K. Li. Receiver-initiated busy-tone multiple access in packet radio networks. In *Proc. ACM SIGCOMM*, pages 336–342, 1987.

APPENDIX

Deriving z-Transform of $Pr\{M = 0, U = u\}$

Given $M = 0$, we have

$$
\begin{aligned}
&Pr\{N = n | M = 0\} \\
&= \frac{Pr\{M = 0 | N = n\} Pr\{N = n\}}{Pr\{M = 0\}} \\
&= \frac{[1 - p_g(h, t)]^n P_N(n)}{Pr\{M = 0\}} .
\end{aligned}
\tag{19}
$$

Therefore, the probability generating function of N given $M = 0$ is

$$
\begin{aligned}
&P_{N|M=0}{}^*(z) \\
&= \frac{\sum_{n=0}^{\infty} [1 - p_g(h, t)]^n P_N(n) z^n}{Pr\{M = 0\}} \\
&= \frac{P_N{}^*[(1 - p_g(h, t))z]}{Pr\{M = 0\}} .
\end{aligned}
\tag{20}
$$

Let $p_u(h, t)$ be the probability that a contending packet has rank equal to $\gamma(h, t)$ given $M = 0$. Then

$$
p_u(h, t) = \frac{p_e(h, t)}{1 - p_g(h, t)} .
\tag{21}
$$

We have

$$
\begin{aligned}
&Pr\{U = u | N = n, M = 0\} \\
&= \binom{n}{u} p_u(h, t)^u [1 - p_u(h, t)]^{n-u} .
\end{aligned}
\tag{22}
$$

The z-transform of the above, taken over u, is $[1 - p_u(h, t)(1 - z)]^n$. Thus, we have the probability generating function

$$
\begin{aligned}
&P_{U|M=0}{}^*(z) \\
&= \sum_{n=0}^{\infty} [1 - p_u(h, t)(1 - z)]^n P_{N|M=0}(n) \\
&= P_{N|M=0}{}^*[1 - p_u(h, t)(1 - z)] \\
&= \frac{P_N{}^*[[1 - p_g(h, t)][1 - p_u(h, t)(1 - z)]]}{Pr\{M = 0\}} \\
&= \frac{P_N{}^*[1 - p_g(h, t) - p_e(h, t)(1 - z)]}{Pr\{M = 0\}} .
\end{aligned}
\tag{23}
$$

Finally, the z-transform of $Pr\{M = 0, U = u\}$ is

$$
\begin{aligned}
&P_{U|M=0}{}^*(z) Pr\{M = 0\} \\
&= P_N{}^*[1 - p_g(h, t) - p_e(h, t)(1 - z)] .
\end{aligned}
\tag{24}
$$