Randomly Ranked Mini Slots for Fair and Efficient Medium Access Control in Ad Hoc Networks

Jacob Eshet and Ben Liang, *Senior Member, IEEE*

Abstract— Ad hoc networks offer infrastructure free operation, where no entity can provide reliable coordination among nodes. Medium Access Control (MAC) protocols in such a network must overcome the inherent unreliability of the network and provide high throughput and adequate fairness to the different flows of traffic. In this paper we propose a MAC protocol that can achieve an excellent balance between throughput and fairness. Our protocol has two versions. Randomly Ranked Mini Slots (RRMS) utilizes control-message handshakes similar to IEEE 802.11. Randomly Ranked Mini Slots with Busy Tone (RRMS-BT) is the better performer of the two but requires a receiver busy tone. The protocol makes use of granule time slots and sequences of pseudo random numbers to maximize spatial reuse and divide the throughput fairly among nodes. We demonstrate the performance of this protocol using simulation with fixed and random topologies, and show that these results are robust to difficult network configurations and unsynchronized clocks. We further develop novel metrics of long-term and short-term fairness for rigorous performance evaluation. Our simulation results include detailed comparison between the proposed protocol and existing protocols that have been shown to excel in terms of throughput or fairness.

Index Terms— distributed multihop wireless networks, ad hoc networking, medium access control, random ranks, mini slots, busy tone, aggregate throughput, long-term fairness, short-term fairness.

I. INTRODUCTION

As wireless technologies advance and become more popular, wireless protocols must evolve to meet the higher demands of these technologies. Ad hoc networks are an exciting approach to network design [1], [2]. Not relying on complex and expensive infrastructure, which is required by their traditional counterparts, ad hoc networks can operate in scenarios where traditional networks fail, such as disaster relief and military applications. Ad hoc networks are also particularly well suited for sensor networks and small scale temporary solutions (e.g. conferences). Much research is also taking place in an effort to extend ad hoc networks to more general-purpose applications such as wireless local area networks (WLANs) [3], [4], [5] and as extensions to centralized networks, such as future generation cellular networks [6], [7].

However, the benefits of an ad hoc network come at a significant cost. The wireless medium is shared among many nodes and is prone to collisions, and the lack of infrastructure requires protocols to be distributed. For example, the hidden

Ben Liang is with the Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, Ontario, M5S 3G4, Canada. Jacob Eshet was a Master of Applied Science student in the same department when most of this work was performed. E-mail: {eshet,liang}@comm.utoronto.ca. This work has been supported in part by a grant from the Natural Sciences and Engineering Research Council of Canada.

terminal problem [8] is a well-known consequence of this phenomenon. In general, the unpredictable and dynamic topology of ad hoc networks leads to each node possessing only partial knowledge of the network topology. This often results in an unfair allocation of throughput between different senders.

A. Fairness in Multihop Medium Access

The IEEE 802.11 standard specifies a Distributed Foundation Wireless Medium Access Control (DFWMAC) protocol [9] for wireless channel contention. DFWMAC uses the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism. It utilizes RTS-CTS handshakes to reserve the channel before data transmission. This nearly eliminates the hidden terminal problem. DFWMAC also makes use of a binary exponential backoff (BEB) algorithm to control network congestion. Nodes must backoff after the completion of a successful exchange or when an exchange fails. The duration of the backoff is random and uniformly distributed between zero and a backoff window, which is doubled whenever an exchange fails or set to a predefined minimum value when an exchange succeeds. Many previous research efforts have shown that DFWMAC can perform poorly in the multihop environment [10], and various improvements have been proposed.

The first well-known effort to deal with the fairness problem at the MAC layer was the MACAW protocol [11]. MACAW incorporated several innovations to address the issues of fairness and better collision avoidance, including in particular a Multiplicative Increase Linear Decrease (MILD) backoff algorithm, which resulted in more subtle adjustments to the backoff window size. This work provided a starting point for a considerable amount of research into fairness in ad hoc networks. Ozugur et al. in [12] proposed that nodes should exchange information on the number of connections they have to other nodes, or the average time they have to wait before sending an RTS. They show how this information can be used to calculate the transmission probability for each flow. Bononi et al. proposed that each node should monitor the channel to determine what fraction of time the channel is in use [13]. They use this parameter together with the number of retry attempts to compute the transmission probability of each flow. In [14], Haas et al. further extended these results and studied how the backoff window size should be dynamically adjusted.

The *fair queuing* technique is another approach to achieve fairness in ad hoc networks [15], [16], [17]. It requires the labelling of data packets with start tags and finish tags to keep track of the priority of each packet. Nodes must then estimate

when to send a packet of a given priority. This estimate can be based on an approximation, or can be calculated by monitoring the traffic sent by other nodes [18]. Fairness can also be achieved by directly exchanging between nodes the priorities of the packets waiting to be transmitted at each node. One protocol that uses this approach is DWOP [19], [20], proposed by Kanodia et al. This protocol achieves First-In-First-Out (FIFO) fairness by maintaining a schedule of packets that are waiting to be sent or received by all neighbors together with their priorities at every node. A node would only send its own packet if this packet has a higher priority than all other packets in its schedule. To update neighbors' schedules on newly arriving packets, information on the current packet sent and the next packet waiting in the queue is inserted into all packets of an exchange. Specifically, information on the next packet (it's id and priority) is inserted into the RTS and CTS, so that neighbors can add an entry to their schedule, and information on the current exchange is inserted into the DATA and ACK packets so that nodes can delete this entry from their schedule.

Another class of contention resolution schemes are based on pseudo random priorities. In the Neighborhood-aware Contention Resolution (NCR) algorithm by Bao et al. [21], each node maintains a sequentially updated pseudo random rank in each data transmission time slot. The ranks of all contenders (one-hop neighborhood for node-based NCE or two-hop neighborhood for link-based NCR) are recorded. A node or a link, depending on the particular channel access flavor of NCR, is activated and allowed to participate in data transmission, if it has the highest priority among its contenders. In the SEEDEX protocol by Rozovsky et al. [22], each node generates a pseudo random number sequence using a random seed. The time axis is slotted. Each slot is labelled 'L' for Listen or 'PT' for Possibly Transmit based on the random sequence. Periodically, nodes must broadcast their own seed and the seed of all their one hop neighbors. This allows every node to keep track of the schedules of every other node in its two-hop neighborhood. In order to send a packet, the sender must wait for a slot that is labelled 'PT' for itself and 'L' for its receiver. In this slot, a number of the receiver's neighbors will also be in the 'PT' state. The authors calculated the packet transmission probability in order to minimize the probability that more than one neighbor of the receiver will transmit in this time slot.

B. Main Contributions of this Work

Toward the goal of fair and efficient spectrum sharing in distributed multihop wireless networks, we introduce a new medium access control protocol termed Randomly Ranked Mini Slots (RRMS). Borrowing from [21] and [22], RRMS utilizes a pseudo random sequence of *ranks* specific to each node. Similar to [19], [20], [21], [22], the ranking information is made known to all of the node's contending neighbors via periodic broadcasting or piggy-backing in control and data packets. However, unlike the previously proposed schemes, RRMS includes new random access mechanisms where the times slots for the rank sequence, termed *mini slots*, can have a scale much smaller than the data packet transmission duration.

We show that using mini slots can lead to significant improvement in the spatial-reuse of the wireless medium, while maintaining *long-term* and *short-term* fairness in spectrum sharing. However, it can also lead to unexpected complications in terms of unfairness due to a *dual capture* phenomenon, as well as due to time synchronization inaccuracy. We then demonstrate that, with proper handshakes and a *rank attenuation* scheme, under moderate time synchronization inaccuracy, granule time-scale random ranking can achieve an excellent balance between throughput and fairness. We propose two variants of this protocol, an in-band version simply termed RRMS, which utilizes control-message handshakes similar to IEEE 802.11, and an improved version termed RRMS-BT, which uses an out-of-band receiver busy tone similar to [23], [24]. As far as we are aware, this work is the first to propose a viable, fair medium access mechanism based on node co-ordination with random ranks over granule time slots and an optional busy tone.

The rest of this paper is organized as follows. In Sections II and III, we provide detailed descriptions of RRMS and RRMS-BT. In Section IV, we discuss the performance measurements in terms of throughput and fairness. In Section V, we present extensive simulation results, to show that RRMS and RRMS-BT can achieve excellent throughput and fairness in both long and short terms. We further provide guidelines on how the granularity of the mini slots should be tuned, given the accuracy of time synchronization. Finally, conclusions are given in Section VI.

II. RANDOMLY RANKED MINI SLOTS

The lack of coordination in ad hoc networks leads to frequent collisions between nodes. This problem can only be overcome by introducing randomness into the MAC protocol. Many protocols, including those based on CSMA-CA, use random backoff periods. This solution has two obvious disadvantages. First, there is no optimal method to calculate the bounds of the random backoff period for multi hop ad hoc networks, which leads to inefficient operation and in some cases, extreme unfairness. Second, backoff periods are, by their very nature, time wasted by waiting idly. The proposed RRMS protocol introduces new mechanisms to bring randomness into collision avoidance by utilizing pseudo random number sequences over granule time slots.

A. RRMS Operational Principle

There are several ways to generate a pseudo random sequence [21], [22]. In RRMS, we use a fixed finite state machine $\mathcal{S}$ for all nodes based on a shift register sequence [25], which ensures low correlation between the generated number sequences. Each node generates a new term in the random sequence for each new time slot. The number generated in the current slot is called the *rank*. More precisely, let Ti denote a potentially transmitting node, and $R_{Ti}(t)$ be its rank at time slot t. Then $R_{Ti}(t)$ is computed by the finite state machine using Ti's rank at some time t_{i0} in the past as

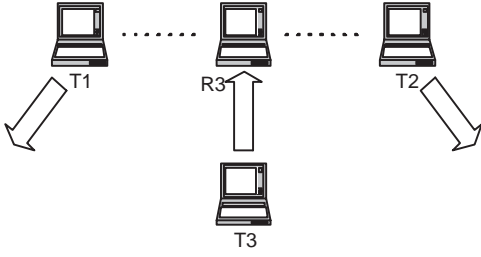$$R_{Ti}(t) = \mathcal{S}(R_{Ti}(t_{i0}), t) \ .$$

Fig. 1. RRMS and spatial reuse. Ti and Ri denote potential transmitters and receivers respectively.

Fig. 2. RRMS exchange mechanism.

We call $R_{Ti}(t_{i0})$ the *seed* of Ti at time t_{i0}. If the random sequence has a sufficiently large state space, and each node selects a hash function of its own id as the seed at network initiation, then the probability of two interfering nodes selecting the same rank in the same time slot can be made negligible.

Note that t_{i0} can be any arbitrary time in the past. Therefore, each node requires only the seed of its contenders in order to keep track of their entire rank sequence. In Section II-D we will discuss the contender set based on interfering flows specific to RRMS, and in Section II-E we will discuss how seeds can be exchanged between nodes up to two hops away. For now, we assume that the identities of the contenders and their seeds are known to each node. Then, a node will begin to send in the current time slot if the channel is idle and its own rank is higher than the ranks of all the potential senders that interfere with it.

One main feature that distinguishes RRMS from NCR or SEEDEX [21], [22] is that the rank-sequence time slots in RRMS are much smaller than the data transmission duration. Therefore, a single data exchange is sent over multiple time slots. We refer to these granule time slots as *mini slots*. The disadvantage of having time slot size equal to the data transmission duration is best explained in a sample topology as shown in Figure 1. This figure shows a common scenario where three flows line up. Given independent rank sequences, it is clear that the event $E_{monotone} = \{R_{T1} < R_{T3} < R_{T2}\} \cup \{R_{T2} < R_{T3} < R_{T1}\}$ occurs in any time slot with probability

$$P(E_{monoton}) = \frac{2}{3!} = \frac{1}{3} .$$

However, this event is highly undesirable if the rank-sequence time slot size equals the data transmission duration, since in the first case, only $T2$ is allowed to send in the current slot, and in the second case, only $T1$ is allowed to send in the current slot. Ideally, we would like $T1$ and $T2$ to send simultaneously (i.e., to take advantage of spatial reuse). In Section II-B, we further quantify the advantage of using mini slots that are much shorter than the data transmission duration, such that $T1$ and $T2$ are allowed to send nearly simultaneously.

Since the data exchange takes place over multiple random-rank time slots, the RTS-CTS handshake are needed to reserve the channel. Without channel reservation, one node may begin to send in one slot because it has the highest rank in this slot, and an interfering node may begin to send in the following slot because it has the highest rank in that slot. The packets
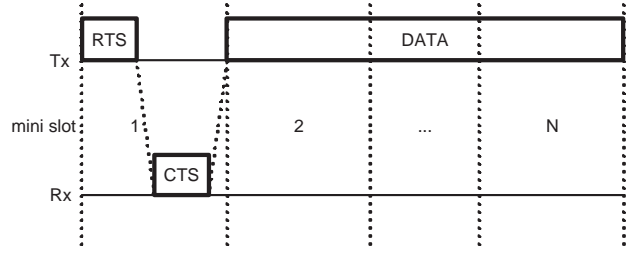
would then collide. The exchange mechanism of RRMS as shown in Figure 2 solves this problem. The RTS packet also provides added protection from data collision in the case where interfering nodes have erroneous ranking information of each other.

The exchange begins with the sender sending an RTS in the first slot. The receiver, if it is ready to receive, will reply with a CTS in the same slot. The sender will then begin to send a DATA packet in the second slot for as many slots as are needed. The RTS and CTS serve the same purpose as they do in DFWMAC [9], namely, they announce to neighboring nodes that an exchange of a stated duration is about to take place and that all interferers should defer their transmissions. Neighbors that hear the RTS or CTS set up a Network Allocation Vector (NAV) for the duration of the exchange, which prevents them from transmitting during this time. A sender only begins an exchange in a slot in which it has a higher rank than all of its interfering senders and if it does not have its NAV set.

Note that there is no need for backoff. If the sender of an RTS does not receive a CTS from the receiver by the end of the mini slot, the transmission is aborted. Then, the sender just needs to wait for the next slot in which it has the highest rank. If the CTS is received in time, the DATA packet is sent at the beginning of the next mini slot and continues for as many slots as needed.

The following pseudo code summarizes the general operation of RRMS in each time slot for a node attempting to transmit:

1. *if* NAV is set
2.   Wait for next slot
3. *else if* self rank is not maximum among contenders
4.   Wait for next slot
5. *else*
6.   Send RTS
7.   Wait for CTS from receiver
8.   *if* CTS is not received
9.   Wait for next slot
10.   *else*
11.   Send DATA

B. Mini Slots and Spatial Reuse

Next, we further quantify the spatial-reuse efficiency of RRMS through random-rank mini slots. We again use the common scenario as shown in Figure 1. Without loss of generality, we consider one of the outcomes in the event

$E_{monotone}$, $R_{T1} < R_{T3} < R_{T2}$, at the current mini slot. Then $T2$ alone will send in this mini slot. In the next mini slot, $T1$ will begin to send if $R_{T3} < R_{T1}$, which will occur with probability $\frac{1}{2}$. On the other hand, if $T3$ has the highest rank in the second mini slot, then nothing will happen, since $T3$ will send an RTS but $R3$ won't receive it. Then, in the third mini slot, $T1$ will again begin to send with probability $\frac{1}{2}$, and so on.

Clearly, the smaller is the mini slots, the sooner will $T1$ be allowed to send. More precisely, the slot in which $T1$ will send is a geometrically distributed random variable with parameter $\frac{1}{2}$, whose mean wait time is 2 mini slots. This means that we can expect that on average only 2 mini slots will be wasted by $T1$. This illustration clearly demonstrates the spacial-reuse advantages of mini slots. Therefore, in general we should try to minimize the duration of a mini slot.

However, the minimum allowable size of the mini slot is limited to the duration of an RTS packet and a CTS packet plus overhead (propagation time, processing time, etc.), because the highest-ranked sender must send RTS and receive CTS in the same mini slot, to reserve the channel before an interferer begins to send. If the size of the RTS and CTS packets could be made smaller, less bandwidth would be wasted. Ideally, the control packets' size to data packet size ratio should approach zero, and then spatial reuse would be perfect; that is, all nodes that can send simultaneously without interfering with each other would do so.

In Section V we will present quantitative results to show that throughput increases with smaller mini slots in more complicated random topologies as well, even when time synchronization is imperfect.

C. Dual Capture and Rank Attenuation

Although using mini slots can achieve spatial reuse efficiency, in some scenarios it can lead to unfairness. In particular, we discuss a *dual capture* phenomenon and present a method to alleviate its effect.

The simplest example of this can be shown with the topology of Figure 1. Let N be the number of mini slots in one complete data transmission duration. Suppose all nodes are backlogged and $R_{T1} < R_{T3} < R_{T2}$ in a given mini slot. $T2$ will begin transmitting in this mini slot if the channel is available. As explained previously, $T3$ cannot send while $T2$ is sending. Then, it can be shown that $T1$ will likely begin to transmit while $T2$ is sending. Since the the waiting time for $T1$ is geometrically distributed with parameter $\frac{1}{2}$, it will transmit within the time from the 2nd mini slots to the Nth mini slot with probability $1 - \left(\frac{1}{2}\right)^{N-1}$. For example, if we assume a nominal DATA packet length of 8000 bits and the mini slot size corresponding to 800 bits[1], then $N = 10$. Hence, the probability that $T1$ will begin to transmit while $T2$ is sending is 99.8%. Thus, $T1$ is virtually guaranteed to capture the channel from $T3$ while $T2$ is sending.

When $T2$ has completed sending its data packet, $T3$ remains quiet because $T1$ is still sending. Hence $T2$ is likely to send
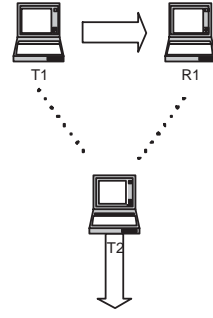

Fig. 3. Because $R1$ is a neighbor of $T2$, $T1$ would benefit from knowing when $T2$'s rank is attenuated.

again before $T3$ does. This cycle may go on as long as both $T1$ and $T2$ are backlogged. More sophisticated topologies leads to various levels of the dual capture effect. In general, $T3$ will receive a much lower share of the medium than $T1$ or $T2$.

We solve this problem with *rank attenuation*. After any sender completes a successful exchange, it attenuates its rank to r_A for N mini slots. The exact value of r_A depends on the network traffic load. In this work, we assume prevalent contention and set $r_A = 0$. After N mini slots, the rank returns to the value it would have had if rank attenuation did not take place. Note that, the interfering senders do not have to be made aware of the rank attenuation taking place. In the previous example, as long as $T1$ does not capture the channel, $T3$ will begin transmitting in the first time slot where its rank is higher than the un-attenuated rand of $T1$. However, potentially interfering senders can benefit from the rank attenuation information by not deferring to senders whose rank has been attenuated.

To make interfering senders aware of rank attenuation, a Notice of Rank Attenuation (NRA) is added to the RTS and CTS control packets. Senders neighboring the receiver benefit from NRA in the CTS, since they are interfering senders. Neighbors of the sender do not usually benefit from the NRA in the RTS. If they are receivers then the NRA is not forwarded to their senders.[2] If the neighbor of the sender is also a sender, then the two senders do not interfere with each other in most cases. The only exception is when a sender and its receiver both neighbor a second sender. In this situation the two senders interfere with each other and the first sender benefits from the NRA in the RTS of the second sender. This is illustrated in Figure 3.

Returning to the topology in Figure 1 and assuming that $R_{T1} < R_{T3} < R_{T2}$, suppose now that rank attenuation is used. $T2$ begins an exchange in the first mini slot. The mean time for $T1$ to begin its own exchange is the third mini slot. Then, in the $(N+1)^{th}$ mini slot, $T2$'s rank will be attenuated. In the $(N+3)^{th}$ mini slot $T1$'s rank will be attenuated. If $T3$ hears the NRAs from $T1$ and $T2$, then it will send in the $(N+3)^{th}$ mini slot. If it does not hear either NRA, possibly due to packet collision, its expected time to send is the $(N+5)^{th}$

---

[1]This will fit a complete exchange of RTS at 352 bits and CTS at 304 bits with inter-frame spacing [9].

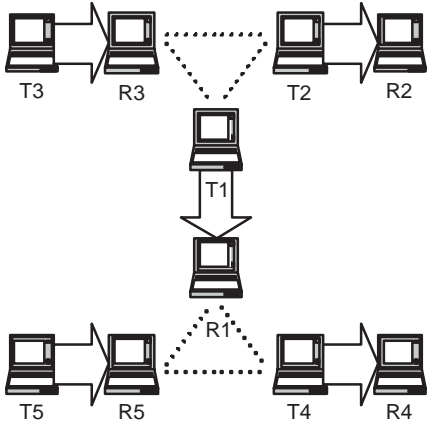[2]This protocol could potentially be expanded by having receivers notify their senders of other senders' NRAs.

Fig. 5. $T1$ must consider $T3$'s and $T4$'s ranks before sending an RTS. The other senders do not interfere with $T1$'s transmission.

mini slot.[3]

In effect, the duration of a complete exchange of $T1$ and $T2$, who would ideally send simultaneously, is stretched by only 2 mini slot durations. Again we see that smaller mini slots improve throughput by taking better advantage of spatial reuse. Note that thanks to rank attenuation each sender achieves the same amount of throughput. We compare the ideal spatial-reuse transmission sequence and the RRMS transmission sequence for the above example in Figure 4.

D. Interfering Flows in RRMS

The exchange mechanism of RRMS is designed to allow the most spatial reuse that a system without busy tones can provide. This means that the exchange mechanism should minimize the sources of interference that it is vulnerable to. Two flows, A and B, will always interfere with each other if A's sender is a neighbor of B's receiver or if B's sender is a neighbor of A's receiver. Other potential sources of interference, however, can be minimized or avoided altogether.

Figure 5 demonstrates all the possible sources of interference that a flow, denoted $T1$-$R1$, might encounter. Flow $T1$-$R1$ is surrounded by four other flows. Flow $T3$-$R3$ is clearly an interfering flow because $T1$ and $R3$ are neighbors. Flow $T4$-$R4$ is also an interfering flow because $R1$ and $T4$ are neighbors. For this reason, $T1$ must have a higher rank than $T3$ and $T4$ to begins an exchange in any given mini slot (assuming it does not have a NAV in place).

On the other hand, $T1$-$R1$ and $T2$-$R2$ may send simultaneously if they both begin their exchange in the same mini slot, because the control packets will not collide (see Figure 2). We can expect that $T1$ will not send an RTS while $T2$ is sending data because $T1$ will have a NAV in place (and vice versa). Even if $T1$ misses $T2$'s RTS and does not set up a NAV, it will not interfere with flow $T2$-$R2$ by sending an RTS. Rather, it will not be able to continue its exchange because it won't hear the CTS from $R1$ (since $T2$ is sending DATA). We can

[3]Denote the mini slot when $T3$ will begin its exchange $N + 2 + S_3$. For large N, S_3 is a random variable with geometric distribution $P[S_3 = i] \approx \frac{1}{3}\left(\frac{2}{3}\right)^{i-1}$, for $i \geq 1$. It has mean 3.

exploit this situation fully by not requiring $T1$ to consider the rank of $T2$ when sending an RTS. This allows $T1$ and $T2$ to send an RTS in the same mini slot. As explained before, there is no danger of them interfering with each other.

The interference between flows $T1$-$R1$ and $T5$-$R5$ is similar to the case with flows $T1$-$R1$ and $T2$-$R2$. Here too, $T1$ and $T5$ can send at the same time only if they both send the RTS in the same mini slot. If $T1$ sends an RTS to $R1$ while $R5$ receives DATA from $T5$, $R1$ should not respond because it should have a NAV in place. If $R1$ did not set up a NAV and did respond with a CTS, the DATA packet being received by $R5$ would be lost. To allow $T1$ and $T5$ to send an RTS in the same mini slot, we again do not require $T1$ and $T5$ to consider each other's ranks when sending.

To summarize, a sender, Ti, needs to consider the rank of another sender, Tj, only if Tj is a neighbor of Ti's receiver, or Tj's receiver is a neighbor of Ti. This also implies that two interfering senders may be at most two hops apart. Rank seeds, therefore, never need to be broadcast over more than two hops.

In realistic scenarios nodes are not usually aware of which node is a sender and which node is its receiver. Furthermore, this information often changes over time due to traffic arrival patterns and mobility. If the topology is fixed, or if this information changes slowly over time, there may be significant advantage in broadcasting this information along with the seed broadcasts. If, however, it is unknown whether some node is a sender or receiver, a sender must consider that node's rank when sending an RTS. If that node is not an interfering sender, then some bandwidth is wasted in considering its rank. However, the amount of wasted bandwidth decreases by using smaller mini slots.

E. Seed Broadcasting

As explained previously, a node needs to know the ranks of all interfering senders, which may be one or two hops away. The two-hop neighborhood information could be propagated via each node broadcasting the list of its neighbors to its neighbors. As proposed in [15]-[22], such broadcasts can be performed periodically or piggy-backed in control and data packets. Similarly, the seed information could be broadcast periodically, or could be piggy-backed in existing exchange packets. In order to disseminate seed information to two hop neighbors, every node must broadcast its own seed information together with all of its one hop neighbors' seed information.

Because of the nature of the finite state machine in RRMS, ideally once a node is made aware of its neighbor's rank, it can track its random sequence indefinitely. Realistically, however, periodic updates of seed information are required to allow for unsynchronized clocks and changing topologies due to mobility or other reasons. However, as long as the level of mobility is not too high, seed information could be sent infrequently. In particular, this also implies that the operation of RRMS does not require constantly backlogged traffic.

III. RANDOMLY RANKED MINI SLOTS WITH BUSY TONE

The Randomly Ranked Mini Slots with Busy Tone (RRMS-BT) protocol is a variant of RRMS that makes use of a receiver
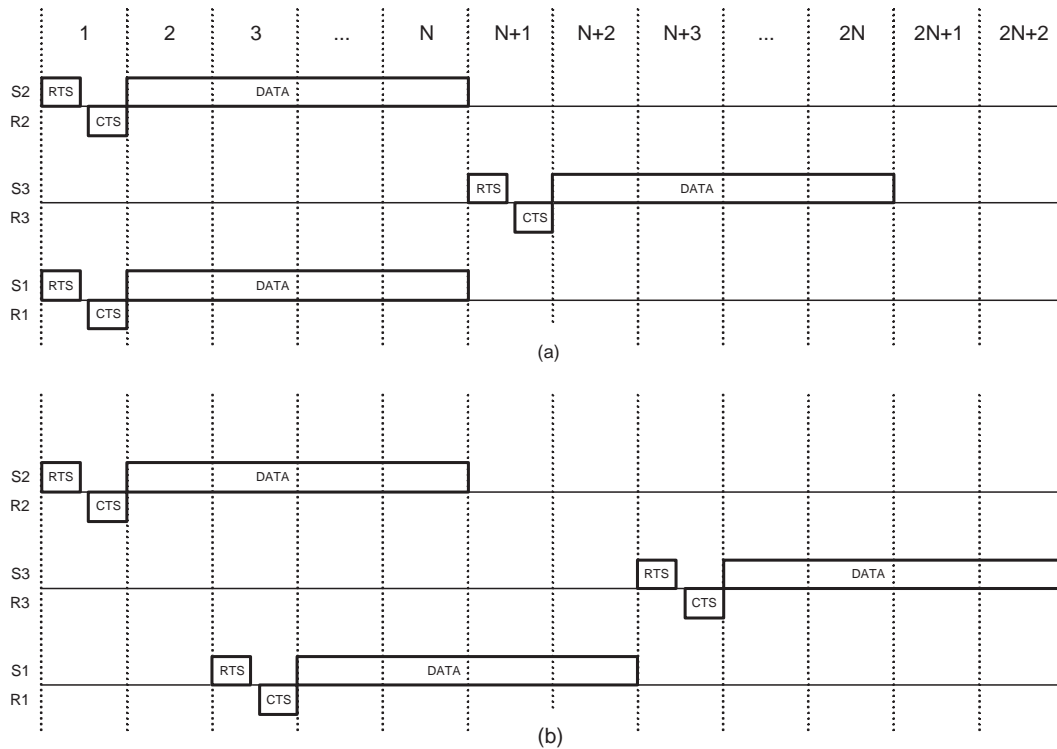
Fig. 4. Transmission sequences: (a) ideal scenario for spatial reuse (b) RRMS scenario.

busy tone. The busy tone provides better protection for the DATA packet and a complete solution to the exposed terminal problem [23], [24]. This can result in significant improvements in performance, especially in topologies where the exposed terminal problem is pronounced. In this section we explain the differences between RRMS and RRMS-BT.

A. Busy Tones in Wireless Medium Access Control

Out-of-band busy tones has been shown previously to drastically improve wireless MAC efficiency. The busy tones provide better collision protection than the RTS-CTS collision avoidance mechanism used by DFWMAC. They were first proposed by Tobagi and Kleinrock in [8], where a MAC protocol was proposed for a centralized network that used a busy tone to reserve the channel. Wu and Li extended this approach for ad hoc networks in [23], proposing a Receiver Initiated Busy Tone Multiple Access (RI-BTMA) protocol, which used a receiver busy tone and time slotted operation. Haas and Deng expanded the protocol further in [24] by using two busy tones in a protocol named Dual Busy Tone Multiple Access (DBTMA). The two busy tones allowed the use of an unslotted system and offered more protection against collisions.

Unlike the RTS and CTS packets, busy tones do not collide, and therefore it is impossible for a neighbor not to hear the busy tone. The busy tone also stays on for the entire duration of a packet, unlike RTS and CTS packets, which need to be transmitted before the DATA packet. Since the receiver never sends any packets to the sender, the exposed terminal problem as well as the hidden terminal problem are solved by this approach.
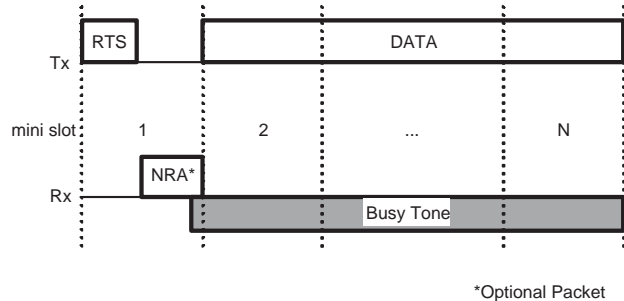

*Optional Packet

Fig. 6. RRMS-BT exchange mechanism.

B. RRMS with a Receiver Busy Tone

The proposed exchange mechanism of RRMS-BT is shown in Figure 6. The exchange proceeds as follows. The potential sender (with the highest rank) sends an RTS in the first mini slot. If the receiver is ready to receive, it monitors the channel for busy tones. If no busy tone is detected, the receiver sends an NRA packet immediately, in the same mini slot, and transmits a busy tone. If a busy tone is detected, the receiver transmits a busy tone only, without the NRA packet so as not to interfere with neighboring nodes that are currently receiving. The sender does not need to receive the NRA packet to proceed. After it detects the busy tone, the sender begins to send the DATA in the second mini slot and continues for as many mini slots as necessary. The receiver terminates the busy tone when the DATA packet is received completely.

Thus, a sender may send an RTS in a given mini slot only if its rank is higher than the ranks of all interfering senders

and it does not detect a busy tone. Furthermore, the mini slot length must be large enough to contain the RTS and NRA packets, and permit the sender to conclude that the receiver is not ready to receive if a busy tone has not been sensed by the end of the first mini slot. The busy tone eliminates the need for NAVs and provides better collision protection since it is never missed due to collisions.

Since RRMS-BT does not require a CTS packet, NRA information is sent in an NRA packet, which is designated specifically for this purpose. This packet is optional since NRA information is beneficial but not necessary. Making the NRA packet optional effectively solves the exposed terminal problem, which can increase throughput significantly in some topologies.

In Section II-D we have analyzed the sources of interference that affect each flow when the RRMS protocol is used. If we repeat this analysis with the RRMS-BT protocol, we will see that the busy tone solution minimizes interference between flows greatly. In Figure 5 flow $T1$-$R1$ was shown in a highly congested neighborhood, surrounded by four other flows. Flows $T3$-$R3$ and $T4$-$R4$ interfere with $T1$-$R1$ because they expose a sender from one flow to a receiver from the other flow. $T1$ must therefore have a higher rank than $T3$ and $T4$ to send an RTS. Flows $T2$-$R2$ and $T5$-$R5$ however, do not interfere with $T1$-$R1$ in any way and consequently $T1$ does not need to have a rank higher than $T2$ or $T5$ in order to send. Because $T1$ only needs to consider the ranks of two other transmitters when sending, we can expect that flow $T1$-$R1$'s throughput will be approximately $\frac{1}{3}$ of the channel capacity, since $T1$'s rank will be higher than the ranks of $T3$ and $T4$ in $\frac{1}{3}$ of the time slots.

In general, the share of throughput that a flow will receive depends only on the number of flows it is interfering with, and not on the specific topology of these flows. Specifically, a flow that has n interfering flows will receive $\frac{1}{n+1}$ of the channel capacity in throughput. This renders RRMS-BT (and RRMS to a lesser degree) immune to hidden terminals or any specific topology configuration.

IV. PERFORMANCE EVALUATION

We evaluate the performance of RRMS and RRMS-BT by comparing them with DFWMAC (802.11), DWOP, and DBTMA. Comparisons are made in three categories: throughput, long-term fairness, and short-term fairness.

A. Throughput

Throughput comparison is based on straight forward examination of the total throughput of all flows in the network (aggregate throughput). The reader should keep in mind, however, that throughput and fairness are a trade-off, and therefore the highest possible aggregate throughput for any given scenario is usually extremely unfair and therefore undesirable.

For example, in the three flow topology shown in Figure 1, maximum aggregate throughput will be achieved if flows $T1$-$R1$ and $T2$-$R2$ transmit continuously and flow $T3$-$R3$ does not transmit at all. This will result in a throughput of two times the channel capacity. This situation is clearly unacceptable.

A more fair solution would be to alternate transmissions by letting flows $T1$-$R1$ and $T2$-$R2$ transmit, then flow $T3$-$R3$, and then flows $T1$-$R1$ and $T2$-$R2$ again. This would result in an aggregate throughput of only $\frac{3}{2}$ times the channel capacity, but each flow would receive an equal share of throughput.

One could also argue that flow $T3$-$R3$ is in a denser part of the network since it is competing with 2 other flows, whereas flows $T1$-$R1$ and $T2$-$R2$ are competing with only one flow, and therefore flow $T3$-$R3$ should receive a smaller share of throughput. In any case, we emphasize that throughput results must be considered in light of fairness results and not as an independent evaluation.

B. Long-term Fairness

We also compare the simulation results in terms of long-term fairness. We first present the following example as a guideline for qualitative comparison. In Figure 1, one could argue that flow $T3$-$R3$ should receive $\frac{1}{3}$ of the channel capacity because it is competing with 2 other flows, and flows $T1$-$R1$ and $T2$-$R2$ should receive $\frac{1}{2}$ of the channel capacity, since each of them is competing with only one flow. However, the throughput values ($\frac{1}{2}$, $\frac{1}{3}$, $\frac{1}{2}$) do not take full advantage of spatial reuse. For example, if flow $T3$-$R3$ receives $\frac{1}{3}$ of the channel capacity then flows $T1$-$R1$ and $T2$-$R2$ could each receive $\frac{2}{3}$ of the channel capacity. Clearly a protocol should not be penalized for giving 2 flows an extra $\frac{1}{6}$ of channel capacity in throughput. Alternatively, the extra $\frac{1}{6}$ of channel capacity could be given to flow $T3$-$R3$, resulting in each of the 3 flows getting $\frac{1}{2}$ of channel capacity in throughput. It is impossible to say which result is more fair. We may therefore state that a protocol will be considered fair in the long-term if every flow that has n interfering flows receives a minimum share of $\frac{1}{n+1}$ of channel capacity in throughput. Given that such *basic fairness* is satisfied, a larger throughput value is, of course, more desirable.

We also note that qualitative judgements cannot be made when the topology is very large since it is impossible to keep track of the performance of each flow. For these topologies we use a quantitative measure of long-term fairness, which we denote by $Flow\ RMSE$. The $Flow\ RMSE$ is found by carrying out a long-term fairness comparison between the resultant transmission sequence of a given MAC protocol and that of the *ideal transmission sequence*, which is the transmission sequence that would result if a central coordinator was available and would have instructed nodes to transmit in FIFO order while utilizing spatial reuse. Formally, we form the ideal sequence, S_I, with the following algorithm, where a transmission slot is the duration required to send a complete exchange:

STEP 1: TRANS SLOT = 1
STEP 2: Node with the earliest packet time of arrival transmits in this time slot.
STEP 3: Node with the earliest packet time of arrival that can send in this time slot, without interfering with other transmissions currently taking place, transmits.
STEP 4: Repeat STEP 3 until no more nodes can transmit in this time slot.

STEP 5: TRANS SLOT = TRANS SLOT + 1
STEP 6: Go to STEP 2 until the end of experiment.

The resulting sequence contains entries of the form $S_I(i) = (id_i, td_i)$, where id_i is the id of the ith packet, and td_i is the time of delivery of the ith packet. The time of delivery is simply the time of the end of the time slot in which the packet was sent.

Note that for a fair comparison, different ideal sequences are required for protocols that solve the exposed terminal problem and for protocols that do not (i.e., Step 3 would produce different sequences). We assume implicitly when presenting our results, that DFWMAC, DWOP and RRMS are compared to the ideal sequence that does not solve the exposed terminal problem and that DBTMA and RRMS-BT are compared to the ideal sequence that solves the exposed terminal problem.

Once S_I is obtained, we find the transmission sequence (id's and delivery times), S_{Sim}, for each protocol via simulation. A comparison between S_I and S_{Sim} must be made to evaluate how closely the simulated sequence resembles the ideal sequence. Suppose a given scenario has m flows, labelled 1 to m. Let T_I^i and T_{Sim}^i represent the throughput achieved by the ith flow in the ideal sequence and simulated sequence, respectively. Let T_I and T_{Sim} represent the total throughput of the ideal and simulated sequences, respectively. We define the measure of long-term fairness as

$$Flow\ RMSE = \sqrt{\sum_{i=1}^{m} \left(\frac{T_I^i}{T_I} - \frac{T_{Sim}^i}{T_{Sim}} \right)^2} \ .$$

Since the $Flow\ RMSE$ represents the root mean square error between the normalized flow rates of the ideal sequence and the actual sequence, a low $Flow\ RMSE$ indicates the protocol's results are close to the ideal results. It is important to note that the throughput of each flow in this formula is normalized by the total throughput of the relevant sequence. This is because we wish to compare the fraction of total throughput achieved by each flow and not the absolute throughput value. If we did not do so, then protocols that achieve high throughput might achieve lower $Flow\ RMSE$, since the ideal sequence achieves relatively high throughput also. By normalizing each flow's throughput by the total throughput, we ensure that $Flow\ RMSE$ is a measure of long-term fairness alone and is independent of throughput performance.

C. Short-term Fairness

In order to evaluate the short-term fairness of a MAC protocol, we again compare its resultant transmission sequence to the ideal transmission sequence. The comparison is made difficult by the fact that the simulated sequence is unfair to some degree, and therefore entries for some packet id's will be missing. For example, in a two flow topology, where packets with odd id's arrive at flow 1 and packets with even id's arrive at flow 2, an unfair protocol might yield a simulated sequence that consists of very few even id's[4]. We cannot ignore the entries for the even id's that are not present, since they reflect

the unfairness of the protocol. We resolve this problem by comparing only the first N_U packets of the ideal sequence to their corresponding entries in the simulated sequence[5], where N_U is the largest value for which the first N_U packets of the ideal sequence have a corresponding entry in the simulated sequence.

We define U as the set of id's of the first M packets of the ideal sequence where $1 \leq M \leq N_U$. We find the difference between the ideal delivery time td_{id} and the simulated delivery time td_{id}^{sim} for each packet id as follows.

$$DT(id) = \left| \frac{td_{id} - td_{id}^{sim}}{td_{id}} \right|, \ id \in U \ .$$

Finally, we can compare the simulated sequence to the ideal sequence with the $Fifo\ Deviation$ measure, defined as

$$Fifo\ Deviation(M) = \frac{\sum_{i \subset U}[DT(i)]}{M} \ .$$

Usually, we will assume that $M = N_U$ and write $Fifo\ Deviation(N_U)$ simply as $Fifo\ Deviation$.

The Fifo Deviation figure gives a good indication of short-term fairness. The lower the value of the Fifo Deviation, the better the protocol approximates the ideal sequence that would be generated by a centralized system.

V. SIMULATION RESULTS

We begin by presenting simulation results for a fixed-topology scenario with five protocols: RRMS, RRMS-BT, DFWMAC, DWOP, and DBTMA[6]. The fixed-topology scenario shows the effect of the topology configuration on the performance of the five protocols. We then simulate the same five protocols over multiple instances of a large random topology for various Poisson traffic arrival rates to show that our results extend to realistic and complex scenarios. After that, we investigate the effects of different mini slot sizes on throughput and fairness, to examine the relationship between the mini slot size and spatial reuse. Finally, we inquire into the effects of unsynchronized clocks on throughput and fairness in large topologies.

We used a packet-level discrete event simulator that was built using the C++ programming language[7]. The simulator assumed a "perfect" physical layer, that is, if two nodes are within each other's range the transmission is received successfully (assuming there are no interfering flows). Furthermore, the simulator assumes a packet collision model, such that all concurrently arriving packets at a receiver are lost.

RRMS and RRMS-BT belong in a class of scheduling protocols that operate independently from the underlying signal propagation model, given the neighborhood topology. Similar protocols also include [19], [20], [21], [22], among others. Clearly the neighborhood topology itself depends on the signal propagation model. These protocols would perform as specified if they are operated faster than the channel state variation. Therefore, slow fading will have similar effect as

---

[4]In other words, the time of delivery of the missing packets with even id's is past the end of the simulation.

[5]A corresponding entry is an entry for the same packet id.

[6]DBTMA simulations use the MILD backoff algorithm, with contention window sizes as defined by the 802.11 standard [9].

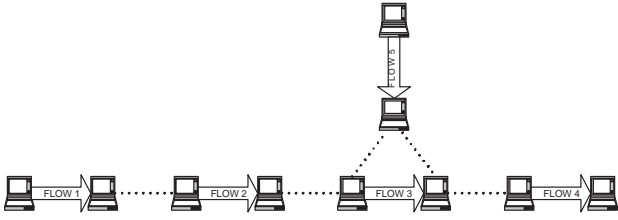[7]The simulator is available online or by email request.

Fig. 7. Fixed topology example.

Protocol	Data Transmitted (10^3 bits)					
	Flow 1	Flow 2	Flow 3	Flow 4	Flow 5	Total
DFWMAC	568	16408	8	17112	16904	51000
DWOP	6432	5248	5176	5176	6904	28936
RRMS	7840	7400	7552	7896	7512	38200
DBTMA	112	18392	0	18504	18480	55488
RRMS-BT	7792	7720	7616	7808	7808	38744
FIFO	9584	9576	9576	9576	9576	43888

TABLE I

FIXED TOPOLOGY EXAMPLE THROUGHPUT RESULTS FOR EACH FLOW.

node mobility. However, the effect of fast fading is more difficult to quantify. In general, we assume that the physical layer is sufficiently competent to alleviate the effects of fast fading.

All of our simulations used a 1 Mbps channel with packet lengths listed in the following table, except where noted otherwise. The RTS, CTS, and ACK parameters are taken directly from the IEEE 802.11 standard [9]. The DATA length is a nominal value within the bounds of the standard specification.

Packet	RTS	CTS/NRA	DATA	ACK
Length (bits)	352	304	8000	304

A. Fixed Topology Example

In this section we present simulation results over a fixed topology to illustrate how each protocol handles the difficulties that can result from the topology. The network under consideration is shown in Figure 7. The simulation was carried out for 20 seconds, and backlogged traffic arrival was assumed at every sender. RRMS and RRMS-BT used a mini slot duration of $800\mu s$. This duration is long enough to contain the RTS and CTS (or NRA) packets and overhead.

The per-flow throughput results for this scenario are shown in Table I. This table demonstrates both the throughput performance and long-term fairness of the protocols. DFWMAC and DBTMA achieved significantly higher total throughput than the other protocols (and DBTMA outperformed DFWMAC because of the busy tones and shorter exchange mechanism). However, the cost in terms of fairness is great. Both DFWMAC and DBTMA choke flows 1 and 3 almost completely. In effect, by not using an effective fairness mechanism, they allow only the maximal independent set of flows (2, 4, and 5) to transmit, and choke all other flows. DWOP achieves better results than DFWMAC and DBTMA in terms of long-term fairness. However, it performs suboptimally in terms of throughput, since its

packet priority ranking strategy does not take full advantage of spatial reuse. In contrast, the performance advantage of RRMS and RRMS-BT stems from random ranking over granule time slots. Table I shows that they yield superior results in both long-term fairness and throughput. This observation confirms our discussion in Section II-B, where we quantify the spatial-reuse efficiency of RRMS and RRMS-BT.

Another performance metric of interest is the short-term fairness. Short-term fairness comparison could not be made for DFWMAC and DBTMA, since they choked two of the five flows, and consequently their value of N_U was very small. The $Fifo\ Deviation$ parameter for the other protocols converged to 0.84 for DWOP, 0.27 for RRMS, and 0.26 for RRMS-BT.

Further discussions on the performance of RRMS and RRMS-BT in larger networks with random topology and imperfect time synchronization are presented in the next subsections.

B. Random Network Topology

To evaluate the performance of RRMS and RRMS-BT in more realistic scenarios, we simulated the five protocols over many instances of a random topology. The topology consisted of 100 nodes randomly placed in a square area with edge wrapping. Each node had on average six neighbors, and the probability of each node being a sender was $\frac{1}{6}$. A receiver was randomly chosen for each sender from one of its neighbors. The simulation was carried out with Poisson traffic with arrival rates $\lambda = \frac{1}{10^i}$ (packets/ms) for $i = 0.6$ to 2.0 in intervals of 0.2, and for a continuously back-logged arrival rate of $\lambda = 1$. Each scenario (for any combination of protocol and arrival rate) was simulated 30 times with each simulation lasting 20 seconds. The two highest and two lowest results were discarded to reduce the variation, and an average was taken over the remaining 26 trials. The confidence interval for these results was also recorded.

RRMS and RRMS-BT used a mini slot duration of $800\mu s$ as in the fixed-topology scenario simulations. The throughput results are shown in Figure 8(a), the $Flow\ RMSE$ results are shown in Figure 8(b), and the $Fifo\ Deviation$ results are shown in Figure 8(c). Figure 8(a) also includes the throughput results achieved by the ideal sequences without and with the exposed terminal problem solved, labelled FIFO (1) and FIFO (2), respectively.

Figure 8 shows that, when the network is not too congested, aggregate throughput increases, and fairness decreases, as the network becomes more heavily loaded, for all protocols. The performance difference between protocols becomes more apparent when the packet arrival rate is higher. Figure 8(a) shows that the throughput of RRMS is near that of DFWMAC, and that the throughput of RRMS-BT is only slightly (less than 20%) lower than that of DBTMA, even when the arrival rates are very high. The reduction of throughput is expected since fair protocols generally achieve somewhat lower throughput than unfair protocols. This is verified by Figures 8(b) and 8(c), which show that RRMS and RRMS-BT yield significant fairness improvements. Again, these improvements are more noticeable at the higher arrival rates.
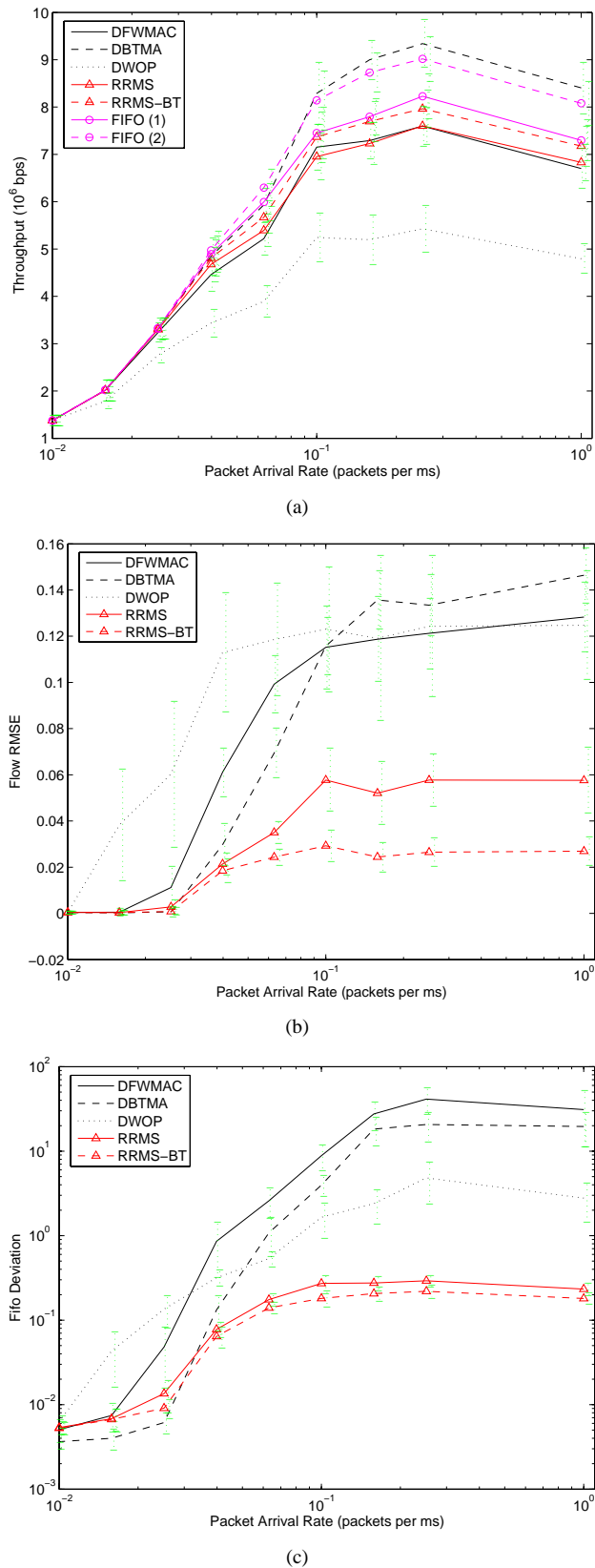
Fig. 8. Throughput and fairness results for DFWMAC, DBTMA, DWOP, RRMS, and RRMS-BT for random topologies and varying packet arrival rate. The interval between each pair of short horizon lines represents the 95% confidence interval with Student's t approximation.

Furthermore, we observe that the throughput and fairness results of RRMS and RRMS-BT are better than those of DWOP. DWOP performs suboptimally in terms of throughput since in many topology instances its packet priority ranking strategy does not take advantage of spatial reuse. On the other hand, DWOP's fairness results are better than DFWMAC and DBTMA when the arrival rate is high, since its fairness mechanism is designed to work in a continuously backlogged network.

C. Varying the Mini Slot Size

To show the effects of mini slot size on throughput and fairness, we carried out simulations of RRMS-BT with different sizes of mini slots over the 100 node random topology described in Section V-B. The simulation was repeated with the ratio between the mini-slot length and the complete-exchange length taken from one of the following: 1/120, 1/60, 1/40, 1/30, 1/20, 1/15, 1/12, 1/10, 1/6, 1/5, 1/4, 1/3, 1/2, and 1/1. In order to allow us to reduce the mini slot size to 1/120 of the total exchange length we used the following parameters:

Packet	RTS	NRA	DATA
Length (bits)	50	50	11900

We ignored all spacing between packets, thus the smallest possible mini slot size was $100\mu s$[8], which is 1/120 the size of the full exchange ($12000\mu s$). The simulations were also carried out for the following rates of Poisson packet arrivals in units of packets per ms: 1/5, 1/12, 1/25, 1/40, and 1/80. Each scenario (mini slots size and rate of arrival) was simulated for 25 trials of 20 seconds each. The top and bottom two results for each scenario were discarded, and the results were averaged over the remaining 21 trials. The standard deviation of these trials was also recorded. Figures 9(a) and 9(b) show the network aggregate throughput and $Fifo\ Deviation$, respectively, versus the mini-slot to full-exchange ratio for various packet arrival rates.

Figure 9(a) demonstrates a consistent increase in throughput with smaller sized mini slots (assuming a fixed exchange length), showing that smaller mini slots can significantly increase spatial reuse. In particular, we note that, at high network traffic load, using a mini slot size of $100\mu s$ can improve the throughput from using full-exchange time slots by more than 50%.

An exception to the rule exists when the mini slot size is increased from one half to full exchange duration. Here we see that throughput does not decrease as sharply, or may even increase. The reason for this is the following. Using smaller mini slots improves throughput because spatial reuse increases. However, the increase in spatial reuse results in an increased likelihood of nodes missing their neighbors' NRAs, because more neighbors of any given node will transmit simultaneously more often. The effect of not hearing NRAs translates into a decrease in throughput, since deferrals are made to nodes whose ranks have been attenuated, and fairness, since nodes in less congested neighborhood are more likely to capture the

---

[8]Recall that the minimum mini slot length duration is limited to the duration of the RTS and NRA packets plus overhead.
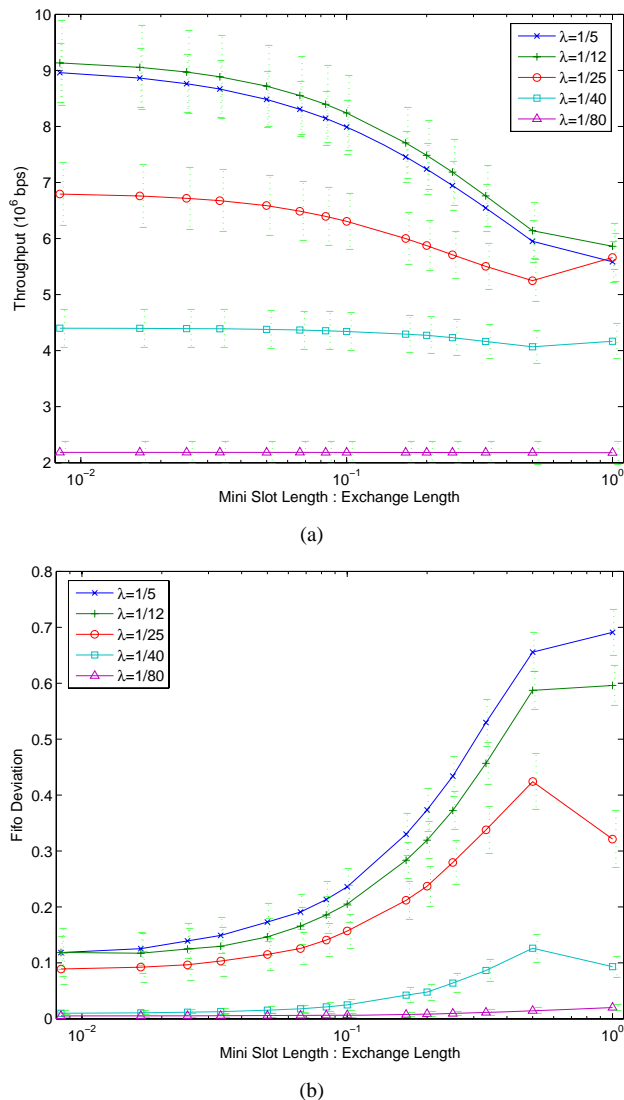
(a)

(b)

Fig. 9. The effects of varying mini slot sizes on throughput and fairness. The interval between each pair of short horizon lines represents the 95% confidence interval with Student's t approximation.

channel (see Sections II-C and II-D). These results suggest that, as the mini slot sizes are decreased from a full exchange length to half an exchange length, the benefits of more spatial reuse are smaller than the detriments of not hearing the neighbors' NRAs.

Figure 9(b) shows that fairness also increases (i.e., *Fifo Deviation* decreases) with smaller sized mini slots. This is not a surprising result since smaller mini slots take more advantage of spatial reuse and the ideal sequence also takes advantage of spatial reuse to achieve the best possible performance. Finally, we note that for both throughput and fairness, smaller mini slots are more advantageous when the packet arrival rate is higher.

D. Clock Synchronization

The operation of RRMS and RRMS-BT relies on having accurate clocks at each node that can keep mini slots synchronized between nodes. Distributed clock synchronization

protocols for multihop wireless networks have been proposed in the past [26], [27], but due to the connection instability in such networks, ideal synchrony at the physical clock's time resolution is generally impossible. We have shown in Section V-C that the performance of RRMS and RRMS-BT improves greatly when the mini slot size is reduced, but so far we have ignored the performance degradation that would result from the difficulty of synchronizing shorter time slots. As the slot sizes become smaller, the effects of unsynchronized clocks would become more noticeable.

If nodes became unsynchronized, any node might perceive the wrong rank for a neighbor with whom it is out of sync, and this could cause interfering senders to transmit simultaneously. We performed simulation to investigate the effects of unsynchronized clocks with RRMS-BT over the same 100 node random topology and with the same packet lengths and mini slot sizes described in Section V-C. The traffic arrival pattern at each sender was Poisson with an arrival rate of 1 packet every $10ms$. We define a parameter, T_{Dev}, for the maximal time deviation between the clocks of the different nodes. At the beginning of the simulation, every node's clock is set ahead of the reference simulation time by an amount that is uniformly distributed between 0 and T_{Dev}. Each scenario (for any combination of mini slot size and T_{Dev}) was simulated 100 times, and each simulation lasted for 20 seconds. Due to the large variation in data, we discarded the five lowest and five highest results, and averaged the remaining 90 results. The throughput results of these simulations are shown in Figure 10(a) and the fairness results are shown in Figure 10(c). The corresponding standard deviations are shown in Figures 10(b) and 10(d).

Figure 10(a) shows that throughput increases when smaller mini slots are used, even when the clocks are extremely unsynchronized. This is partly due to the fact that spatial reuse increases with smaller mini slots. Furthermore, the probability of DATA packet collision does not depend on time synchronization. This is because DATA packets are protected by busy tones (or the NAV, for the case of RRMS without a busy tone), which provide protection against interferers whether they are synchronized or not. The only packets that can collide due to poor synchronization are RTS packets, and the only result of an RTS collision is that one mini slot is wasted. Therefore smaller mini slots provide less wastage due to unsynchronized clocks.

Unfortunately, fairness can degrade with smaller mini slots as can be seen in Figure 10(c). The figure shows that as T_{Dev} is increased, smaller mini slots provide an ever smaller improvement in fairness and, in some cases, offer no fairness advantage at all. However, we emphasize that for moderate variation in clock synchronization (e.g., for $T_{Dev} < 3ms$), RRMS still provides significant short-term fairness improvements over the existing protocols, while maintaining a near optimal level of throughput.

We further note that RRMS requires time synchronization only within a two-hop neighborhood. Recent research results have indicated that the synchronization accuracy in ad hoc networks of up to two hops can be maintained well below 1 ms [27]. Finally, in practice, better clock synchronization
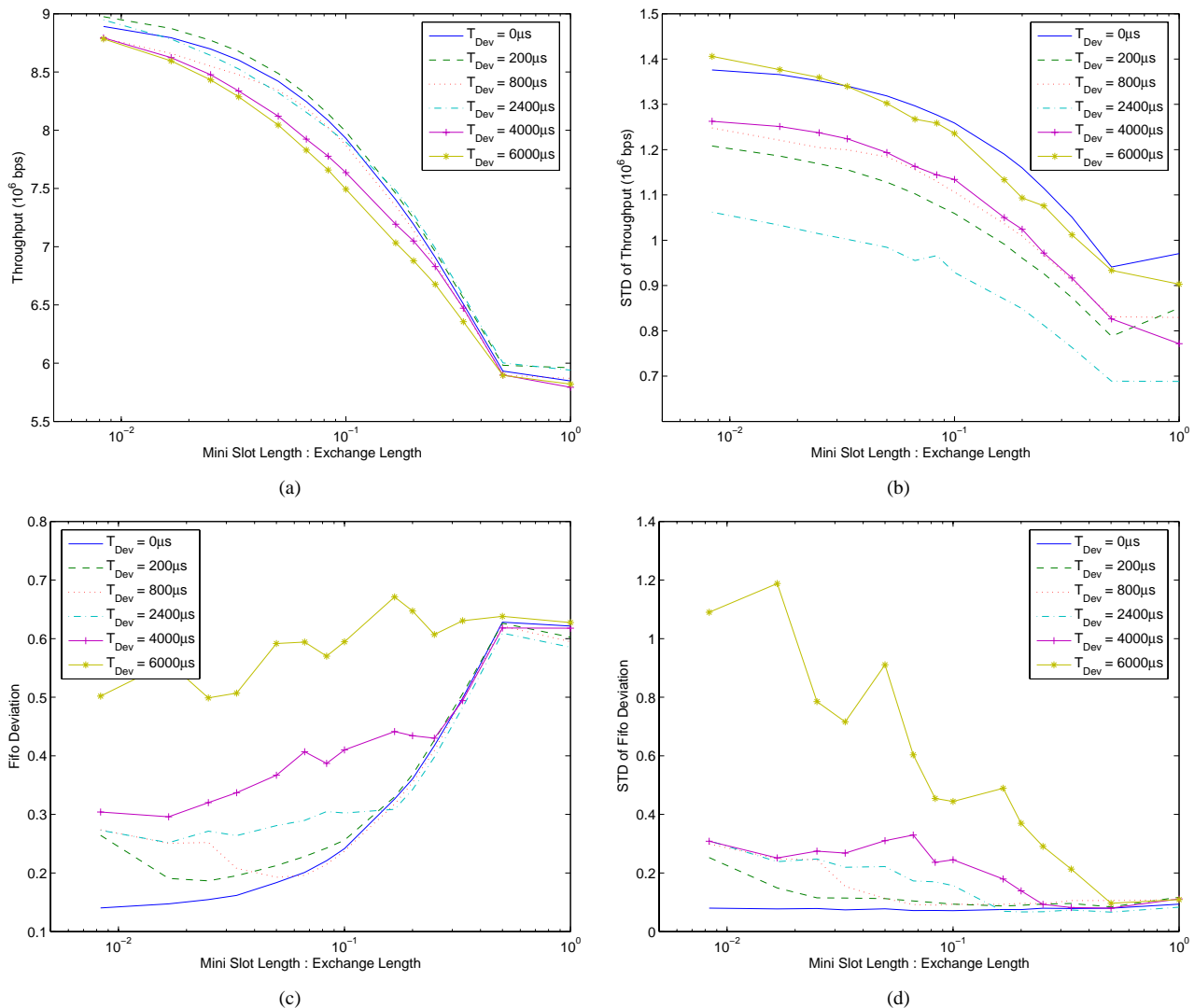
Fig. 10. The effects of unsynchronized nodes with varying mini slot sizes. Both the mean and standard deviation values are shown. For 90 trials, the 95% confidence interval equals approximately 0.42 standard deviation.

inaccuracy can be alleviated by more frequent seed broadcasts.

VI. CONCLUSIONS

The MAC layer of an ad hoc network is an uncoordinated dynamic environment, which presents severe difficulty for reliable communication. The problem of fairness particularly, is a real hurdle, since the shared medium is prone to collisions. The future success of ad hoc networks relies on finding fair, efficient, and robust MAC layer algorithms and protocols.

In this paper we have presented the RRMS and RRMS-BT protocols and provided detailed evaluation of their throughput and fairness performance. RRMS utilizes control-message handshakes similar to IEEE 802.11, which allows it to be implemented easily. RRMS-BT makes use of a busy tone that offers extra performance advantages. We showed that our protocols are, to a large extent, immune to difficult topology configurations, such as the hidden terminal problem, and allocate throughput to flows based on the number of flows they interfere with. We developed a simulator in C++ to provide

fair comparisons between RRMS/RRMS-BT and other leading protocols. These simulations provided throughput values and fairness benchmarks that were well defined. The simulations were carried out over a wide range of scenarios including fixed and random topologies and different rates of packet arrival. The simulation results clearly demonstrated the advantages of RRMS and RRMS-BT in terms of throughput and fairness. Throughput values were lower in some cases as expected, since fairness and aggregate throughput are a trade-off. However, RRMS and RRMS-BT yielded balanced throughput and fairness results that were close to the ideal results that can only be achieved by a centralized (coordinated) network.

We have identified a dual capture phenomenon and flow interference behavior unique to the random contention based on mini slots, and have proposed handshaking and rank attenuation mechanisms to alleviate their effects. We have also investigated the performance of the proposed protocol with varying mini slot sizes and various degrees of unsynchronized clocks, to better understand its capabilities and demonstrate

its practicality. Our results showed that both, throughput and fairness, can be improved with smaller sized mini slots. Furthermore, the improvement was greater when the rate of traffic arrival was higher. Our synchronization experiments showed that even for highly unsynchronized clocks, the network aggregate throughput consistently increases with smaller sized mini slots. Fairness does suffer, to various degrees, when the clocks are unsynchronized and mini slot sizes are made too small. Our results allow a system designer to select the mini slot size that optimizes fairness if the time deviation between clocks is known.

Finally, we remark on two extensions of RRMS for assigning weighted fairness to different flows. Appropriate weighted fairness at the MAC layer would ensure efficient multimedia communication in multihop wireless networks [28] and would also allow priority access in hierarchical ad hoc networks [29]. The first option for extending RRMS is by weighing the rank of each node according to some function $f(R_i, w_{ij})$, where R_i is the rank of node i and w_{ij} is the weight of the flow from node i to node j. The parameter w_{ij} would then have to be broadcast along with the rank information. Another way to achieve weighted fairness is by modifying the length of the data packet according to the weight of the flow, where a flow with a larger weight would send longer data packets and vice versa. Since RRMS and RRMS-BT fully support variable data packet lengths, this method would be easier to implement. The detailed design, optimization, and comparative performance study of these schemes are possible directions for future research.

REFERENCES

[1] C. Perkins, Ed., *Ad Hoc Networking*. Addison Wesley Longman, 2001.

[2] Z. J. Haas, J. Deng, B. Liang, P. Papadimitratos, and S. Sajama, "Wireless ad hoc networks," in *Wiley Encyclopedia of Telecommunications*, J. Proakis, Ed. John Wiley & Sons, 2002.

[3] S. Hsu, L. Qian, and M. Ilyas, "An analytic study of two probabilistic models for establishing ad hoc WLANs," *Information Technology and Management*, vol. 4, no. 1, pp. 55–67, 2003.

[4] A. Readhead and S. Trill, "The role of ad hoc networks in mobility," *BT Technology Journal*, vol. 21, no. 3, pp. 74–80, 2003.

[5] A. So and B. Liang, "A lagrangian approach for the optimal placement of wireless relay nodes in wireless local area networks," in *Proceedings of the IFIP Networking Conference*, Coimbra, Portugal, May 2006, pp. 160 – 172, in *Springer-Verlag Lecture Notes in Computer Science*, vol. 3976.

[6] Y. Lin and Y. Hsu, "Multihop cellular: a new architecture for wireless communications," in *Proceedings of IEEE INFOCOM*, vol. 3, no. 30, March 2000, pp. 1273–1282.

[7] M. Buddhikot, G. Chandranmenon, S. Han, Y. Lee, S. Miller, and L. Salgarelli, "Design and implementation of a WLAN/cdma2000 interworking architecture," *IEEE Communications Society Magazine*, vol. 41, no. 11, pp. 90–100, November 2003.

[8] F. A. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part II the hidden terminal problem in carrier sense multiple-access and the busy-tone solution," *IEEE Transactions on Communications*, vol. COM-23, pp. 1417–1433, 1975.

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

[10] J. Eshet and B. Liang, "Fairness and scheduling in ad hoc networks," in *Security and Routing in Wireless Networks*, Y. Xiao, J. Li, and Y. Pan, Eds. Nova Science, 2005.

[11] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LANs," in *Proceedings of ACM SIGCOMM*, 1994, pp. 212–225.

[12] T. Ozugur, M. Naghshineh, P. Kermani, and J. A. Copeland, "Fair media access for wireless LANs," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, 1999, pp. 570–579.

[13] L. Bononi, M. Conti, and E. Gregori, "Design and performance evaluation of an asymptotically optimal backoff algorithm for IEEE 802.11 wireless LANs," in *Proceedings of the 33rd Hawaii International Conference on System Sciences*, vol. 8, 2000.

[14] Z. Haas and J. Deng, "On optimizing the backoff interval for random access schemes," *IEEE Transactions on Communications*, vol. 51, no. 12, pp. 2081–2090, Dec 2003.

[15] H. Luo, S. Lu, and V. Bharghavan, "A new model for packet scheduling in multihop wireless networks," in *Proceedings of the 6th ACM Annual International Conference on Mobile Computing and Networking (MOBICOM)*, 2000, pp. 76–86.

[16] H. Luo, P. Medvedev, J. Cheng, and S. Lu, "A self-coordinating approach to distributed fair queueing in ad hoc wireless networks," in *Proceedings of IEEE INFOCOM*, vol. 3, April 2001, pp. 1370–1379.

[17] N. Vaidya, P. Bahl, and S. Gupta, "Distributed fair scheduling in a wireless lan," in *Proceedings of the 6th ACM Annual International Conference on Mobile Computing and Networking (MOBICOM)*, 2000, pp. 167–178.

[18] B. Bensaou, Y. Wang, and C. Ko, "Fair medium access in 802.11 based wireless ad-hoc networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2000, pp. 99–106.

[19] V. Kanodia, A. Sabharwal, B. Sadeghi, and E. Knightly, "Ordered packet scheduling in wireless ad hoc networks: mechanisms and performance analysis," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2002, pp. 58–70.

[20] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly, "Distributed multi-hop scheduling and medium access with delay and throughput constraints," in *Proceedings of the 7th ACM Annual International Conference on Mobile Computing and Networking (MOBICOM)*, 2001, pp. 200–209.

[21] L. Bao and J. Garcia-Luna-Aceves, "A new approach to channel access scheduling for ad hoc networks," in *Proceedings of the 7th ACM Annual International Conference on Mobile Computing and Networking (MOBICOM)*, July 2001, pp. 210 – 221.

[22] R. Rozovsky and P. Kumar, "SEEDEX: a MAC protocol for ad hoc networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2001, pp. 67–75.

[23] C. Wu and V. Li, "Receiver-initiated busy-tone multiple access in packet radio networks," in *Proceedings of the ACM workshop on Frontiers in computer communications technology*, 1988, pp. 336–342.

[24] Z. Haas and J. Deng, "Dual busy tone multiple access (DBTMA)-a multiple access control scheme for ad hoc networks," *IEEE Transactions on Communications*, vol. 50, no. 6, Jun 2002.

[25] S. Golomb, *Shift Register Sequences*. Aegean Park Press, 1981.

[26] J. Elson and D. Estrin, "Time synchronization for wireless sensor networks," in *Proceedings of the 15th International Parallel and Distributed Processing Symposium (IPDPS)*, April 2001, pp. 186–191.

[27] K. Romer, "Time synchronization in ad hoc networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2001, pp. 173–182.

[28] B. Liang and M. Dong, "Balancing distance and lifetime in delay constrained ad hoc networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, Florence, Italy, May 2006, pp. 97 – 107.

[29] B. Liang and Z. J. Haas, "Hybrid routing in ad hoc networks with a dynamic virtual backbone," *IEEE Transactions on Wireless Communications*, vol. 5, no. 6, pp. 1392 – 1405, June 2006.

Jacob Eshet received the Bachelor of Applied Science degree in engineering science (electrical option) from the University of Toronto in 2001, with a bachelor thesis on model based parameter estimation techniques for position determination in cellular wireless systems. He joined the graduate program in the Department of Electrical and Computer Engineering at the University of Toronto in 2002 and received the Master of Applied Science degree in 2004. His master thesis research was on fair medium access in distributed multihop wireless networks.

Ben Liang (S'94-M'01-SM'06) received honors simultaneous B.Sc. (valedictorian) and M.Sc. degrees in electrical engineering from Polytechnic University in Brooklyn, New York, in 1997 and the Ph.D. degree in electrical engineering with computer science minor from Cornell University in Ithaca, New York, in 2001. In the 2001 - 2002 academic year, he was a visiting lecturer and post-doctoral research associate at Cornell University. He joined the Department of Electrical and Computer Engineering at the University of Toronto as an Assistant Professor in 2002. His current research interests are in mobile networking and multimedia systems. He received an Intel Foundation Graduate Fellowship in 2000 toward the completion of his Ph.D. dissertation, the Best Paper Award at the IFIP Networking conference in 2005, and the Runner-up Best Paper Award at the International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks in 2006. He is a senior member of IEEE and a member of ACM and Tau Beta Pi. He serves on the organizational and technical committees of a number of major conferences each year.