The RRMS Protocol: Fair Medium Access in Ad Hoc Networks

Jacob Eshet and Ben Liang
Edward S. Rogers Department of Electrical and Computer Engineering
University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada
e-mail: {eshet,liang}@comm.utoronto.ca

***Abstract*— MAC level protocols in ad hoc networks must overcome the decentralized nature of the network and provide adequate throughput and fairness. This paper presents a new MAC protocol named Randomly Ranked Mini Slots (RRMS), which provides throughput at near channel capacity and good fairness as defined by a common sense criterion we develop. The protocol achieves this by utilizing random (pseudo noise) sequences, a receiver busy tone, and by sharing a minimal amount of information between two hop neighbouring nodes. The performance of our protocol is compared to the Dual Busy Tone Multiple Access (DBTMA) protocol in terms of throughput and fairness.**

I. Introduction

Wireless ad hoc networks offer a simple alternative to current network designs. However, the lack of central coordination and dynamic network topology present in ad hoc networks result in detrimental effects to network performance that are difficult to overcome. These effects are particularly troublesome at the MAC layer, where a lack of coordination results in frequent collisions and a dramatic degradation of throughput. Furthermore, the shared medium suffers from the famous "Hidden Terminal Problem", which degrades throughput even more, and creates an unfair allocation of throughput among flows. Many protocols have been proposed to overcome these issues (see for example [1][2]) but none have managed to achieve a good balance between fairness and throughput so far.

In this paper we propose a new MAC protocol named Randomly Ranked Mini Slots (RRMS). RRMS utilizes a number of features, such as a busy tone, synchronized time slots, and some information broadcasting in small neighbourhoods of nodes to achieve excellent throughput and fairness performance. We will demonstrate the performance of RRMS via simulation results at the end of this paper.

To operate optimally, RRMS requires the use of a busy tone. We therefore compare the performance of RRMS to the Dual Busy Tone Multiple Access (DBTMA) protocol [3], since it is the latest and best performing protocol that uses busy tones. The remainder of this section summarizes the development of busy tones and explains the operation of DBTMA.

A busy tone is an out of band signal with a narrow bandwidth. Busy tones were first proposed by Tobagi and Kleinrock in [4], where they proposed a MAC protocol for a centralized network that used a busy tone to reserve the channel. This protocol was extended by Wu and Li for use in ad hoc networks in [5]. This extended protocol, named Receiver Initiated Busy Tone Multiple Access (RI-BTMA), used a receiver busy tone and time slotted operation. Haas and Deng expanded the concept further by using two busy tones in a protocol they named DBTMA. The two busy tones allowed the use of an unslotted system and offered more protection against collisions.

The DBTMA mechanism uses an RTS-DATA exchange together with two busy tones. A transaction takes place as follows. The sender transmits an RTS packet and turns on the transmitter busy tone (BT_t). When the RTS is sent, BT_t is turned off. When the receiver receives the RTS, it turns on the receiver busy tone (BT_r). The sender detects the BT_r and sends the DATA packet. The receiver turns off BT_r once the DATA packet is received, and the exchange is complete. A neighbouring node will not begin to send an RTS if either a BT_t or BT_r is detected. If a node wants to send an RTS and a busy tone is detected, the node will backoff. The backoff mechanism is not specified by the authors. We will use the Multiple Increase Linear Decrease (MILD) algorithm in our simulations. Note that 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 protocol.

II. The RRMS Protocol

The RRMS protocol described in this paper strives to divide throughput fairly among the different flows, while taking advantage of spatial reuse to maximize throughput. At the heart of this protocol is the Pseudo Noise (PN) sequence unique to each node. Each PN sequence is generated by a random number generator such as a shift register (see [6] for more details on how this is done). Each node uses a random seed to initialize its random number generator, ensuring that each PN sequence is unique. Nodes must occasionally broadcast their seed and the seed of all their one hop neighbours, so that every node can keep track of all the PN sequences in its two hop neighbourhood. Time is divided into slots that are synchronized between neighbours. Each number in the PN sequence is associated with a time slot. The number in the PN sequence associated with the current time slot is called the *rank*. Transmitters only send an RTS if their rank is higher than the ranks of all the transmitters of interfering flows. Since the probability that the rank of a given transmitter is higher than all interfering transmitters is a function of the number of interfering transmitters alone $\left(p = \frac{1}{(1+n_I)}\right.$, where n_I is the number of interfering transmitters$\left.\right)$, the fraction of channel capacity that any transmitter is allocated depends only on the number of interfering flows and not on the specific topology. This means that the system is immune to hidden terminals or any other topology configurations which could cause a transmitter to achieve an unfair share of throughput. Note that the randomness introduced by the ranks eliminates the need for wasteful backoff periods.

Unfortunately, the protocol suggested so far does not take full advantage of spatial reuse. For example, consider a topology with three transmitters, A, B, and C, where A interferes with B, and B interferes with C, but A and C do not interfere with each other. In this topology there is a probability of 1/3 that B will capture the channel. There is a probability of 1/3 that A and C will capture the channel simultaneously, but there is also a probability of 1/3 that A or C (but not both) will capture the channel, and hence bandwidth will be wasted. To solve this problem we introduce the concept of a mini slot. Whereas traditionally a slot is the length of time required to transmit a complete exchange, our exchange takes place over multiple slots. Using the parameters of the 802.11b standard and a 1 Mb/s channel, we set the mini slot length to 500μs, which is long enough to contain the RTS packet plus overhead. Returning to the above scenario, if A (or C) captures the channel and C (or A) doesn't, there is a probability of 1/2 that C (or A) will capture the channel in the second mini slot. This probability increases to 3/4 in the third mini slot, 7/8 in the fourth mini slot, etc. Thus C (or A) will wait idly for several mini slots (at 500μs per mini slot), rather than a full slot (approximately 10ms).

To achieve the highest possible throughput, an RTS-DATA exchange is used together with a receiver busy tone. A transaction begins with the sender sending an RTS packet. The receiver on hearing the packet transmits a busy tone. The sender detects the busy tone and sends the DATA packet. Once the DATA packet is received, the receiver terminates the busy tone. Since the receiver never sends packets, this exchange mechanism solves the exposed terminal problem. A transmitter may send an RTS in the current time slot if a busy tone is not detected, and its rank is higher than all transmitters that are one hop away from its receiver and its rank is higher than all transmitters whose receiver is one hop away from itself (see Figure 1). This ensures two interfering transmitters will not transmit simultaneously.

While the mini slot concept utilizes spatial reuse well and improves throughput, it also degrades fairness considerably. To see this, consider the topology mentioned above and assume an exchange lasts for N mini slots. Suppose that in the first mini slot A defers to B and B defers to C. Node A can capture the channel in the 2^{nd} to N^{th} mini slot, while C is still transmitting. In fact A will capture the channel while C is transmitting with a probability of $1 - (\frac{1}{2})^N$ (e.g. 99.9% for $N = 10$). When C completes its exchange, B can not

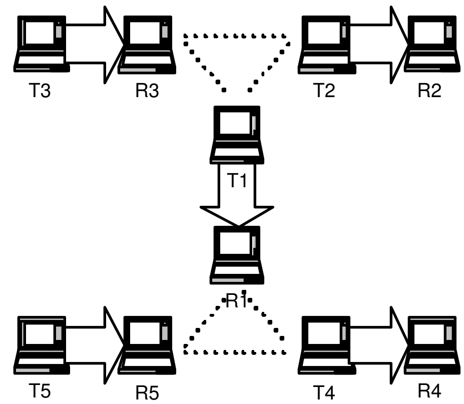

Fig. 1: T1 must consider T4's and T3's ranks when sending an RTS. The other nodes do not interfere with T1's transmission.

transmit since A is transmitting. Thus C has a better chance of capturing the channel than B. B's share of throughput in this topology is very small. We solve this problem by introducing *rank attenuation*. After a node completes a transaction, it sets its own rank to 0 for the next N mini slots. After N mini slots, the rank returns to the value it would have had if rank attenuation was not used. This ensures that every node, even one that is in a congested neighbourhood, will receive a fair share of throughput.

III. Performance Evaluation

We evaluate the performance of RRMS in terms of throughput, long term fairness and short term fairness. Throughput is simply the total number of successful exchanges of all nodes in the network. Long term fairness is a qualitative observation of the fraction of throughput allocated to each flow. To evaluate short term fairness we calculate the ideal delivery times of packets in a fifo sequence with spatial reuse (the sequence that would be used if there was a central coordinator). The root mean square error (fifo RMSE) between actual (simulated) delivery times and ideal delivery times is then computed in seconds.

We simulate the performance of DBTMA and RRMS over three topologies. The first topology, shown in Figure 2, simulates the protocols' performance when a hidden terminal is present. This topology demonstrates that fairness is an important consideration even for very simple topologies. The second topology is the one shown in Figure 1. This topology tests the performance of each protocol in a highly congested locality. The third topology consists of a network of 100 nodes randomly distributed over a square area with edge wrapping, with an average of six neighbours per node. Each scenario was simulated for 100 seconds with either the DBTMA protocol or the RRMS protocol. DBTMA simulations use the MILD backoff algorithm with contention window sizes as defined by the 802.11b standard [7]. The length of the RTS packet is 352 bits (as in the 802.11b standard), and the length of a DATA packet is 8000 bits. A 1 Mb/s channel is used. Data packets are always available at transmitter nodes (constant

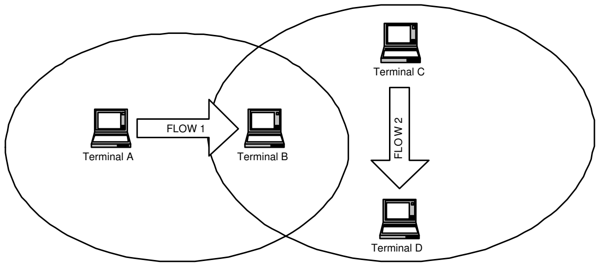

Fig. 2: Scenario 1 Topology

backlog) to simulate performance under high load. The number of successful exchanges of each flow is recorded together with a detailed log of each successful exchange's time of delivery. Delivery times are then used to calculate the fifo RMSE.

IV. Simulation Results

Scenario 1

The number of successful exchanges per flow for the first scenario is shown in the following chart:

	DBTMA (MILD)	RRMS
Flow 1	55	5750
Flow 2	11518	5748
Total	11573	11498

While DBTMA achieves slightly better throughput, the difference in fairness is striking. RRMS achieves perfect long term fairness. A further comparison of fifo RMSE (short term fairness) is unnecessary since DBTMA does not provide long term fairness.

Scenario 2

The results for the second scenario are shown in the following table:

	DBTMA (MILD)	RRMS
Flow 1	0	271
Flow 2	1153	524
Flow 3	2	426
Flow 4	1152	438
Flow 5	6	507
Total	2313	2168

Again, DBTMA achieves slightly better throughput, but at an exorbitant cost to fairness. As expected, we see that flows 2 and 5 perform the best (since they have interference from only one flow). Flows 3 and 4 do slightly worse since they have interference from 2 neighbouring flows. Flow 1 does the worst since it suffers from two interfering flows (3 and 4) that do not interfere with each other. There is no need to calculate the fifo RMSE for this scenario since DBTMA exhibits no fairness at all according to our definition.

Scenario 3

A Simulation result for the third scenario, yielded 16 flows, and the throughput shown in the following chart:

Throughput as Percentage of Capacity	Number of Flows		
	DBTMA (MILD)	RRMS	Fifo (Ideal)
0% - 10 %	9	0	0
10% - 30 %	0	3	0
30% - 70 %	2	9	14
70% - 90 %	0	0	0
90% - 100 %	5	4	2
Total Throughput	126681	100872	95790
Fifo RMSE(s)	6.26	0.16	0

Here DBTMA yielded more throughput than RRMS but again, at an unacceptable cost to fairness. DBTMA let 9 of the 16 flows transmit barely or not at all, while RRMS divided throughput based on local congestion. The low fifo RMSE corroborates this conclusion.

V. Conclusion

While the decentralized nature of ad hoc networks is hard to manage, our simulation results show that fairness can be achieved with little compromise in throughput. The additional cost of the RRMS protocol includes the busy tone and information sharing in 2 hop neighbourhoods. However, performance improvements are dramatic. An attractive feature of the RRMS protocol is that backoffs are not necessary. This eliminates time wasted by waiting idly. Also, The random rankings guarantee a fair division of throughput between flows. Simulation results from scenarios 1 and 2 show that these features indeed yield results that make ad hoc topologies immune to hidden terminals and other topology related degradations. Scenario 3 simulation results show that these results also apply in realistic topologies.

References

[1] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs," in *Proc. ACM SIGCOMM*, 1994, pp. 212–225.
[2] J. J. Garcia-Luna-Aceves and C. L. Fullmer, "Floor acquisition multiple access (FAMA) in single-channel wireless networks," *Mobile Networks and Applications*, vol. 4, pp. 157–174, 1999.
[3] Z. Haas and J. Deng, "Dual busy tone multiple access (DBTMA)-a multiple access control scheme for ad hoc networks," *IEEE Trans. on Commun.*, vol. 50, no. 6, Jun 2002.
[4] 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 Trans. on Commun.*, vol. COM-23, pp. 1417–1433, 1975.
[5] 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*. ACM Press, 1988, pp. 336–342.
[6] R. Rozovsky and P. R. Kumar, "SEEDEX: a MAC protocol for ad hoc networks," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing (MOBIHOC)*. ACM Press, 2001, pp. 67–75.
[7] *IEEE Standard, ANSI/IEEE Std 802.11, 1999 Edition, Part 11*, 1999.