Circularity-Based Medium Access Control in Mobile Ad Hoc Networks

Mohammad Z. Ahmad[1], Damla Turgut[1], and R. Bhakthavathsalam[2]

[1] School of Electrical Engineering and Computer Science
University of Central Florida, Orlando, Florida 32816
{zubair, turgut}@cs.ucf.edu
[2] Supercomputer Education and Research Center
Indian Institute of Science, Bangalore 560 012, India
bhaktha@serc.iisc.ernet.in

Abstract. The RTS/CTS access scheme, designed to reduce the number of collisions in a IEEE 802.11 network, is known to exhibit problems due to masked nodes, the imbalance between the interference range and the communication range of the nodes, and scenarios in which nodes are unnecessarily silenced, thus preventing parallel transmissions to take place. We present an approach for enhancing the performance of the IEEE 802.11 MAC protocol by selectively discarding or delaying specifically marked RTS and CTS packets. By dropping the circularity-satisfied RTS, we allow certain parallel transmissions to proceed, even if there is a non-zero risk of collision. By delaying the circularity-satisfied CTS, we allow a neighboring parallel transmission to continue. One important feature of the circularity approach is that it is fully compatible with the IEEE 802.11 standard. We implemented the circularity approach in ns-2 simulator. Through a series of experiments, we show that the circularity approach provides a significant improvement in the throughput and end-to-end delay of the network, and contributes to a reduction of the number of collisions in most scenarios.

1 Introduction

A mobile ad hoc network (MANET) is a robust, self-managing and autonomous system of cooperating mobile nodes connected by unreliable wireless links. Devices in communication range can communicate directly with each other while intermediate nodes serve as routers to forward packets from the source to the destination.

Since node communication in a dense network happens at the same frequency band, the problem of packet loss due to collision becomes an area of focus. The IEEE 802.11 standard [1] has been primarily designed for wireless LANs and is responsible for scheduling medium access for multiple stations which are contending for the common channel. It uses a medium access scheme based on the Carrier Sense Multiple Access (CSMA) [2] protocol, where a node transmits only if it finds the medium to be idle for a pre-defined Inter Frame Space (IFS).

It also uses a Request-To-Send (RTS) and Clear-To-Send (CTS) control packets to coordinate channel access [3] and minimize costly packet collisions in hidden node scenarios. In multi-hop networks, some nodes may not hear control packets from other nodes within the network. This leads to an RTS/CTS exchange with reduced chances of success and increased possibility of packet collisions. The problem is further complicated with packet loss arising due to some transmissions being masked by other on-going transmissions in their neighborhood [4]. The masked node problem is an example of a shortcoming of the IEEE 802.11 when it is used in MANETs.

In [5] the authors propose the selective disengagement of the RTS-CTS handshake in IEEE 802.11. They point out that the 802.11 standard was developed keeping in mind that the carrier sensing range is equal to the transmission range whereas it is 1.78 times of the latter [6]. This means that any node can actually hear transmissions going on two hops away thereby resolving the hidden node problem to some extent. This however leads to another problem: due to the basic CSMA/CA scheme, a sender will not transmit neither control nor data packets if it senses the channel is busy. By disengaging the RTS-CTS handshake based on the number of "CTS timeouts", they enable greater fairness and higher network throughput both due to lesser control overhead and enabling parallel transmissions to take place.

In this paper, we use a similar technique of turning off the RTS-CTS handshake for particular instances along with delaying the transmission of the CTS packet to enable parallel transmissions to be completed. These changes are made to the IEEE 802.11 MAC protocol and simulations are carried out to compare performance differences. However, revamping the entire MAC standard or the transport layer protocol for use in MANETs is absolutely infeasible due to wide adoption of both. Some modifications need to be incorporated to make the proposed and the legacy protocols inter-operable with each other to ensure fairness and optimal resource utilization. Effective changes should be incorporated in an "non-invasive" manner, i.e. the changes should be based on the underlying mechanisms of the current standard, with as few major changes as possible. The modifications proposed in this paper are novice but effective enhancements to the current standard keeping the underlying principles and the workings of the IEEE 802.11 standard almost untouched.

2   Related Work

The IEEE 802.11 standard uses the Distributed Coordination Function (DCF) which includes Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) as the fundamental access technique. There are two primary access methods in IEEE 802.11: the *basic access* and the *RTS/CTS access* method. The basic access scheme involves only a reliable transfer of the data packets from the source to the destination by using ACK packets. In the RTS/CTS access scheme, the RTS and CTS control packets are first exchanged and the channel is reserved exclusively between the source and destination, which is followed by

the DATA/ACK packet transmission. This RTS/CTS dialog helps in the implementation of the virtual carrier sensing mechanism which is also accompanied by physical carrier sensing in IEEE 802.11 DCF. The RTS/CTS frames contain a duration field which is used by the neighbors to set a specific Network Allocation Vector (NAV) during which the nodes are sent to "silenced" state during which the packet exchanges are being carried out between the sender and receiver nodes. On the other hand, physical carrier sensing is implemented using interframe spaces. After a channel is sensed idle for a DCF Interframe Space (DIFS) time interval, the back-off procedure is invoked by the station which has to send the data. A Short Interframe Space (SIFS) is used to separate transmissions belonging to a single session (CTS, DATA and ACK packets). Extensive work has been carried out on the IEEE 802.11 DCF and the usage of the RTS/CTS mechanism [7], [8].

In [6], the authors present mathematical proof that the interference range is typically 1.78 times the communication range. Even though a node may not be within the transmission range to successfully receive a CTS packet, it may still be the cause of interference at the sender. As a simple solution to the above problem, it is suggested that a node should only reply with a CTS when the received RTS is above a certain receiving power threshold, i.e., it is sufficiently close to the transmitter and hence avoid perceptible interference from other nodes. [9] shows that an optimal carrier sensing range along with an appropriate transmission range and an interference model significantly increases network throughput. Another related work [5] tunes the RTS/CTS exchange by selectively disengaging it when there are occurrences of the CTS not being returned.

In [10], the author states with increasing network complexity and/or node mobility, a node which has not heard of a RTS or CTS packet may migrate into the footprint of a receiver and destroy a DATA packet by initiating its own transmission, oblivious to its surroundings. [4] points out another type of nodes in the same class as that of hidden nodes which are termed "masked" nodes. The authors show that the RTS/CTS exchange is not enough under perfect operating conditions since neighbor nodes are masked by other on-going transmissions nearby. Masked nodes cannot decode the RTS/CTS packets correctly and may end up causing Data/ACK packet collisions later on.

Other widely used approaches to resolving medium access include splitting the available channel into separate control and data subchannels. [11] proposes two schemes that attempt to pipeline *contention resolution* with *data transmission* to reduce the idle waiting time and decrease overall delay. For wireless environments, it also proposes a partial pipeline approach to overcome the shortcomings of the total pipelining scheme. The authors show that with proper channel division, a net throughput increase can be obtained. The authors of [12] propose, the Bi-directional Multi-Channel MAC protocol, where the bandwidth is divided into one control channel and several data channels. It is bi-directional because the receiver may also send his own data packet (if any) to the sender using any of the other available channels thereby eliminating the need of another RTS/CTS handshake.

3   Circularity-Based Medium Access Control

3.1   Motivation

The implicit goal of any MAC protocol is to minimize packet collisions and unnecessary retransmissions. The IEEE 802.11 protocol uses a suitable acknowledgement timeout interval to infer a possible collision. The collision detection time is dependent on the packet sizes and the net bandwidth of the channel. Packet collisions may be categorized as follows:

- RTS packets colliding with other RTS packets when two stations start transmission simultaneously.
- DATA packets colliding with RTS/CTS packets from masked/deaf nodes.
- ACK packets colliding with RTS/CTS packets from masked/deaf nodes.

In this work, we define a channel-access scheme for better channel utilization while encouraging parallel transmissions from other non-interfering nodes.

3.2   Circularity

Circularity is defined as a number which enables the identification of specific groups of control packets sent from each node. The total number of packets in each group is equal to its circularity value and the last packet in the group is termed as the *circularity-satisfied* packet. Each node in the network is assigned a circularity value which may be unique to itself or the entire network may have the same value for each of its nodes. For example, if the circularity value is defined as four, then we divide the RTS/CTS packets being created in each node into groups of four and the first such packet in a group is the circularity-satisfied packet. Hence, every fourth packet being created by the node (i.e. every multiple of four) is circularity-satisfied.

Mathematically, a packet is circularity-satisfied if:

$$N \ modulo \ c = 0$$

where N is the current count of the number of packets generated (RTS/CTS) and c is the circularity value for the particular node.

Essential characteristics of applying circularity to RTS and CTS packets include the following:

- By identifying certain RTS and CTS packets as circularity-satisfied, we induce these packets to **behave** differently from the rest. The structure of the packet (size, headers etc) remains the same, so there are no explicit changes which have to be made to the standard IEEE 802.11 protocol.
- There is no absolute grouping of packets taking place to identify the circularity-satisfied packets. The packets are identified through the simple mathematical formula above.
- There are different RTS and CTS circularity values. For simplicity, in this paper, we have considered both of them to be the same. Thus, for the above mentioned example, every fourth RTS packet and fourth CTS packet emanating from one particular node will be the circularity-satisfied packet.

– The RTS-CTS packets identified as circularity-satisfied are independent of the flow for which they are created. Source nodes and any intermediary node mechanism for all the traffic routed through it.
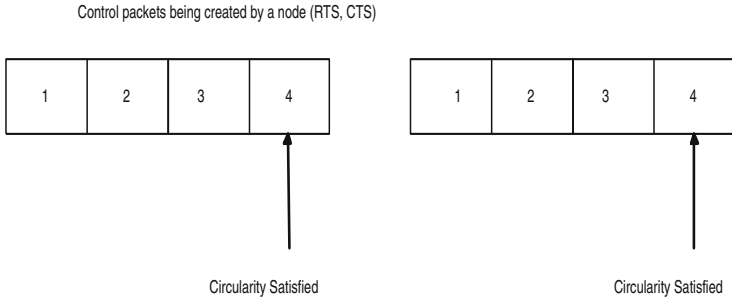
Control packets being created by a node (RTS, CTS)

Fig. 1. Identifying Circularity-satisfied control packets for a circularity value of 4. In effect, every fourth packet becomes a Circularity-Satisfied packet.

Essentially, circularity is just a scheme to identify certain packets which behave differently than the rest. Thus, if the RTS circularity value is considered to be four, then a node which sends out a total of 20 RTS packets during the network lifetime will in effect have identified *five* (20/4) of them as circularity-satisfied. Fig 1 refers to such a case where packets are divided into groups of four and the last packet of each group is tagged as circularity-satisfied. Similarly, considering CTS circularity as five, then a node sending out 20 CTS packets would have identified *four* (20/5) of them as circularity-satisfied.

In our scheme, each node is assigned a specific circularity value for both their RTS and CTS packets. These value pairs are used to **drop** circularity-satisfied RTS packets, irrespective of the existing scenario. Similarly, the circularity-satisfied CTS packets are **delayed** to allow parallel DATA or ACK transmissions in the immediate neighborhood to terminate. In our experimental setup we have restricted the delay of the CTS packets to one SIFS time interval.

3.3 RTS Packet Dropping with Circularity

Dropping RTS packets is a technique of selectively disengaging the RTS/CTS dialogue for a particular transmission session. Let us consider the scenario in Figure 2. In this figure we depict the circles as the sensing range (which is roughly 1.78 times the transmission range). Looking at the timeline sequence in the figure, it can be observed that D initiates a transmission to E with an RTS packet. This RTS packet is transmitted to E and C with E sending a corresponding CTS. C then gets blocked from transmitting. This is followed by D starting its data packet transfer to E, but this can also be sensed by B as it is within carrier sensing range of D. At this point, B's neighbor A (who cannot hear beyond C) initiates a transmission by sending an RTS packet to
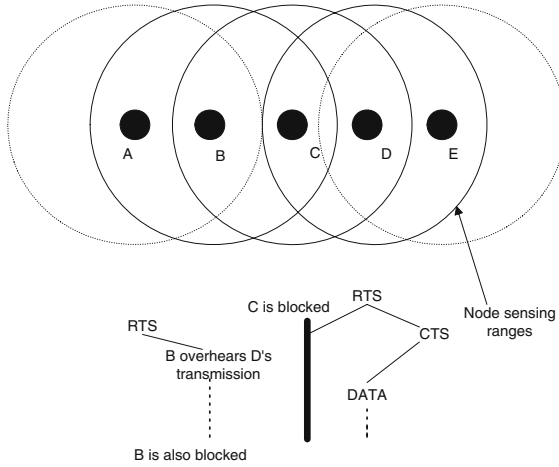
Fig. 2. Hidden Node Scenario with sensing ranges being shown

B. However, as B can sense D sending a long data packet, it will back-off and not reply with a CTS to A. Hence, a parallel transmission is prevented from taking place resulting in decreased network throughput. Another point to be noted is that the flow D to E may capture the channel for a long period of time which results in A giving up re-transmitting RTS packets after the retry count is exceeded and hence reporting a route failure to the routing layer. This will in turn lead to a new process of route discovery and increase network overhead considerably. Disabling the RTS-CTS exchange and simply transmitting the data packet would be of greater effect.

We implement this selective RTS-CTS disengagement by identifying certain RTS packets based on their RTS circularity value and dropping them. We discuss posssible techniques of setting the circularity values in a later section. The authors in [5] take a more conservative approach by waiting for a number of CTS timeouts to occur before sending the data packet directly. We we propose a more aggressive approach of scheduling these packet drops based on the circularity value. The tradeoff is that there may be a higher number of packet collisions occuring but a higher overall network throughput could be obtained due to the parallel transmissions.

3.4   CTS Packet Delay with Circularity

The concept of delaying the CTS packet is also aimed at making possible the occurence of parallel transmission within the network. By delaying the transmission of a CTS packet by a small time interval (one SIFS), we aim to help a neighboring transmission to either continue or complete. Let us consider a sample scenario.

In the Figure 3, we assume that D and C have had a succesful RTS-CTS exchange and D has sent its data packet to C. During the course of this
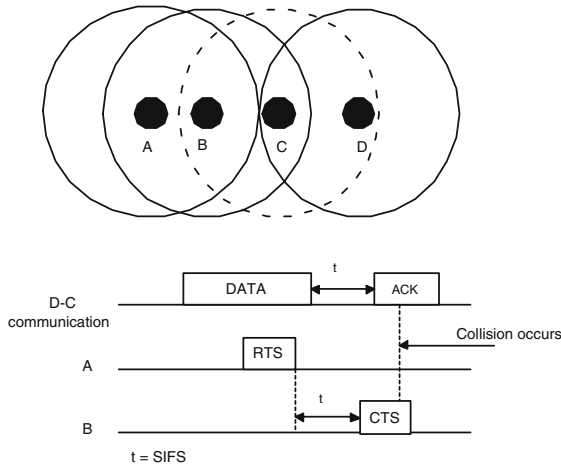
Fig. 3. Scenario depicting CTS collision with ACK packet. The circles denote the transmission range of each node.

transmission, due to the nodes being mobile or if it was earlier a masked node, B comes into transmission range of C. At this point of time, A sends data to B by initiating an RTS-CTS handshake with an RTS packet. Since B is unaware of an ongoing transmission nearby, it responds with its CTS packet which collides with the ACK packet from C. However, if the CTS packet from B was delayed by a small interval, then the ACK packet would be correctly received by D. As only the circularity-satisfied CTS packets are delayed, the impact on the overall delays in the system is reduced. Also it does not guarantee that all CTS-ACK collisions will be avoided but the primary aim is to reduce the probability of these collisions as much as possible. The timeline diagram of CTS delay with circularity is shown in Figure 4.
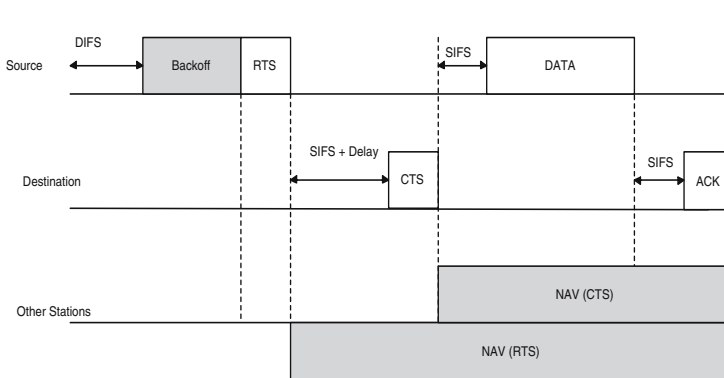

Fig. 4. Timeline Diagram for CTS Delay

3.5   Selecting the Circularity Values

Setting the appropriate pair of circularity values is critical for real-world deployment. In a sparsely populated network, the use of circularity would yield no benefits. In a densely populated networks, we might need to use lower circularity values (which corresponds to a more aggressive dropping and delaying of the packets). This value could be set during the initial network setup time for a device or when the device is joining an existing network. For example, by default the RTS-CTS dialog is turned off in current WLAN cards and can be easily turned on by setting a lower RTS-CTS threshold value. In a similar fashion, the circularity value could be set by the user during or anytime after network bootup. The mechanism could be turned off anytime by setting a very high circularity value.

4   Simulation Study

4.1   Simulation Environment

We carried out simulations using *ns-2* [13] with wireless extensions from the CMU Monarch Group. Changes to the MAC source files were carried out to implement the circularity concept and enable the dropping/delaying of circularity-satisfied RTS-CTS packets.

Node movement is modeled by the random waypoint mobility model [14] with nodes moving at a speed between 10 and 100 m/s and a pause time of 20s. Each data point plotted is the average of 10 different scenarios with different initial network configurations.

Each run in our simulation is executed for 900 seconds of simulation time unless otherwise specified, and the nodes move in a 500m x 500m area. Each node in the network generates traffic of fixed size packets of 512 bytes. The propagation model is the two ray ground model and the maximum number of flows in the network is based on the network size. The data rate is 1 Mb/s and FTP is the application generating packets over TCP as the transport layer protocol. AODV is used as the routing layer protocol for all our simulations as it is quite widely used.

Table 1 lists the parameter values used throughout the simulations. The simulation metrics are net throughput, packet delay and packet loss ratio. We define net throughput as the aggregated throughput over all the flows in the network as the foremost performance metric. Along with this, end to end packet delay and packet drop ratio are also considered as reliable performance metrics. The metrics defined above are simulated with different circularity values.

4.2   Simulation Results

In our simulations, the values of the circularity was varied from 10 to 200. Traffic is generated by ns-2's random traffic generator with different random number seeds. For simplicity, we set the circularity of RTS and CTS to the same value.

Table 1. Simulation Parameters

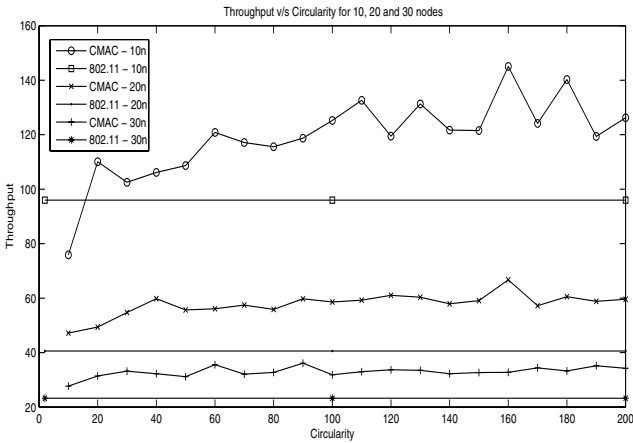| Data Rate	1 Mbps
RTS Size	20 Byte
CTS Size	14 Byte
Data Packet Size	512 byte
ACK Size	14 byte
SIFS	$10\mu s$
DIFS	$50\mu s$
CW Min	31
CW Max	1023
LongRetryLimit	7
Routing Protocol	AODV
Mobility Model	Random waypoint
Propagation Model	Two ray ground

Fig. 5. Throughput versus Circularity for 10, 20 and 30 nodes. Note that for the standard IEEE 802.11, throughput remains constant with increase in circularity.

Figure 5 shows the throughput obtained with respect to circularity for increasing network. We denote our enhanced MAC protocol as Circularity-MAC (C-MAC) and plot the throughput obtained for 10, 20 and 30 node scenario, while the standard IEEE 802.11 protocol naturally remains constant. The graph shows that C-MAC is is obtaining a 15-20% improvement in throughput compared to the 802.11 MAC. Intuitively, this is due to the lower number of packet collisions, which in turn leads to a lower end-to-end delay between source and destination nodes.

Figure 6 confirms this point as it can be observed that with C-MAC, the number of collisions are significantly lower. Packet drop count is an important metric in determining the effectiveness of any MAC protocol, and Figure 6
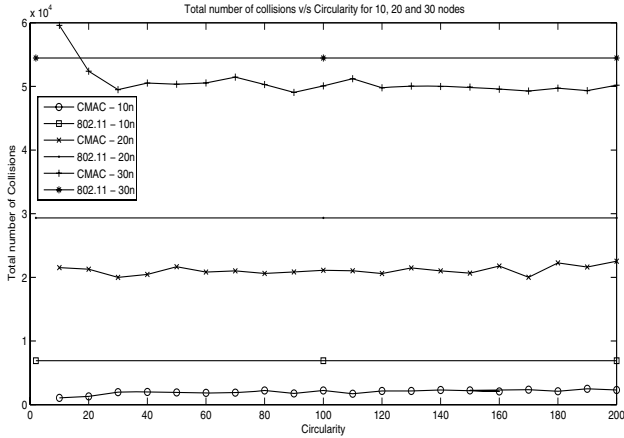
Fig. 6. Total number of collisions versus Circularity for 10, 20 and 30 nodes
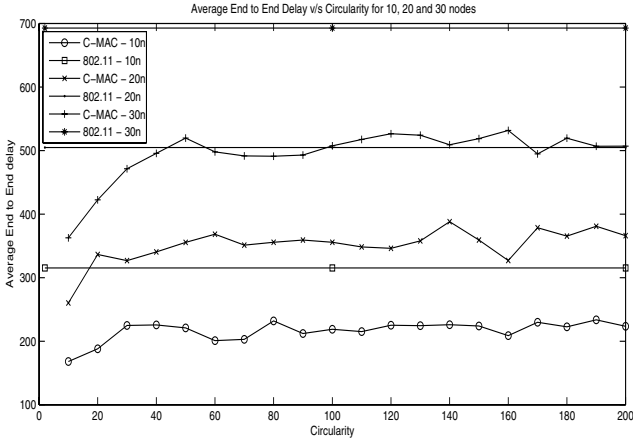

Fig. 7. Average end-to-end delay versus Circularity for 10, 20 and 30 nodes

substantiates the earlier statement that dropping and delaying the circularity-satisfied packets indeed reduces packet collisions.

For both 10 and 20 nodes scenarios, it can be seen that the reduction is almost the tune of 40 percent. Drop in packet collisions leads to lower average end-to-end delay as the network is saved from making successive recoveries from these collisions. The reduced collisions are leading to greater throughput and significantly lower end to end delay (Fig 7). The simulation results show that the improvement can be obtained at all network sizes.

The simulations show that any circularity value greater than 50 results in enhanced performance for the network. Hence, in a real world scenario, any device can have its circularity value set by its user to an arbitrary value within

the range of 50-200. Such a value may not result in ideal network performance, but will certainly be an improvement over the IEEE 802.11 standard.

5 Conclusion

In this paper, we have designed and evaluated enhancements to the IEEE 802.11 mechanism for MANETs, using a novel concept of circularity. This concept has been embedded in the existing RTS-CTS handshake and shows better performance, specially in situations where the contention for the medium is high. Simulation results quantify the advantages in terms of both throughput and the number of packet collisions. As a future work, we propose to investigate the effect of greater delays of circularity-satisfied CTS packets, increase traffic loads by increasing packet sizes for arbitrary circularity values. The goal is to identify standard pairs of circularity values for different scenarios, so that they can be effectively deployed according to the various existing network conditions.

References

1. IEEE Std 802.11b-1999. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Standard 802.11 (1999)
2. Kleinrock, L., Tobagi, F.: Packet switching in radio channels: Part I - carrier sense multiple-access modes and their throughput-delay characteristics. IEEE Transactions on Communications **COM-23** (1975) 1400–1416
3. Bharghavan, V., Demers, A., Shenker, S., Zhang, L.: MACAW: A media access protocol for wireless LAN's. In: Proceedings of ACM SIGCOMM '94. (1994) 221–225
4. Ray, S., Carruthers, J.B., Starobinski, D.: Evaluation of the masked node problem in ad-hoc wireless lans. IEEE Transactions on Mobile Computing **4** (2005) 430–442
5. Ju, H., Rubin, I., Kuan, Y.: An adaptive RTS/CTS control mechanism for IEEE 802.11 MAC protocol. In: Proceedings of IEEE Vehicular Technology Conference. Volume 2. (2003) 1469 – 1473
6. Xu, K., Gerla, M., Bae, S.: Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks. Ad Hoc Networks **1** (2003) 107–123
7. Crow, B.P., Widjaja, I., Kim, J.G., Sakai, P.T.: IEEE 802.11 Wireless local area networks. IEEE Communications Magazine **35** (1997) 116–126
8. Bianchi, G.: Performance analysis of the IEEE 802.11 Distributed Coordination Function. IEEE Journal on Selected Areas in Communications **18** (2000) 535–547
9. Deng, J., Liang, B., Varshney, P.: Tuning the carrier sensing range of IEEE 802.11 MAC. In: Proceedings of IEEE Global Telecommunications Conference. Volume 5. (2004) 2987–2991
10. Haas, Z.: On the performance of a medium access control scheme for the reconfigurable wireless networks. In: Proceedings of the IEEE MILCOM. Volume 3. (1997) 1558–1564
11. Yang, X., Vaidya, N., Ravichandran, P.: Split-channel pipelined packet scheduling for wireless networks. IEEE Transactions on Mobile Computing **5** (2006) 240–257
12. Kuang, T., Williamson, C.: A bidirectional multi-channel MAC protocol for improving TCP performance on multihop wireless ad hoc networks. In: Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems. (2004) 301–310

13. VINT: The UCB/LBNL/VINT network simulator-ns (version 2). (URL `http://www.isi.edu/nsnam/ns`)
14. Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y., Jetcheva, J.: A performance comparison of multi-hop wireless ad hoc network routing protocols. In: Proceedings of Mobile Computing and Networking. (1998) 85–97