# An Adaptive RTS/CTS Control Mechanism for IEEE 802.11 MAC Protocol

Huei-Jiun Ju, Izhak Rubin, and Yen-Cheng Kuan

Electrical Engineering Department
University of California, Los Angeles
{hju, rubin, yckuan}@ee.ucla.edu

*Abstract--* **In this paper, we study the impact of using or disengaging the RTS/CTS dialogue in IEEE 802.11 DCF MAC protocol under the fact that carrier sensing, transmission and interference ranges are distinctively different. The resulting throughput performance features of a linear topology network configuration are demonstrated when applying Constant Bit Rate (CBR) UDP as well as TCP type traffic flows. Based on these results, we propose a new RTS/CTS control mechanism. Under our scheme, a terminal node decides dynamically and individually whether to use a RTS/CTS dialogue for the transmission of its current data packet. We show that this new mechanism yields distinctive performance improvements.[1]**

## I. INTRODUCTION

In an ad hoc wireless network where mobile units employ omni-directional antennas, each communications channel is shared by closely located mobiles. The sharing of this channel is controlled by the employed Medium Access Control (MAC) protocol. The network's throughput efficiency is determined by the working of this MAC protocol. Contention based random-access multiple access protocols have been commonly used since they are simple to implement. To further increase the efficiency of the operation, carrier sense based MAC algorithms are used, requiring the mobile terminal to first sense the channel to determine that it is idle and only then attempt its packet transmission. The latter attempts can still results in a collision event (when the intended receiver detects multiple transmissions at such power levels that it may not be able to correctly receive any of them). Commonly used such protocols include Carrier Sensing Multiple Access (CSMA) and CSMA with Collision Avoidance (CSMA/CA).

CSMA based MAC protocols can yield an efficient operation (under proper loading levels) when the carrier sensing operation is spatially effective. Unfortunately, stations may be geographically located in a manner that induced blocking, leading to masked terminal scenarios. In this case, two major problems have been identified: hidden terminal and exposed terminal conditions.

To resolve the hidden terminal problem, Karn originally proposed the use of Request-to-send and Clear-to-Send (RTS/CTS) handshaking scheme leading to the Multiple Access Collision Avoidance (MACA) protocol [1]. If a node has a packet to send, it first transmits a RTS packet to request the channel. If available, the receiver replies with a CTS packet. After the sender receives the CTS packet successfully, it proceeds to transmit the actual data packet. Nodes that hear the RTS packet will defer transmission for a sufficiently long period of time to allow the transmitter to receive the CTS packet. Nodes hear the CTS packet will back off for a period of time that is sufficiently long to allow the receiver to receive the entire data packet. A number of extended protocols using this mechanism have been devised, including MACAW[2], FAMA[3], and others. These schemes all employ the basic RTS/CTS scheme described above, while including some modifications aiming at improving net performance.

Under the IEEE 802.11 MAC layer protocol [4], the Distributed Coordination Function (DCF) defines a basic CSMA/CA access method and incorporates the use of the RTS/CTS mechanism. As noted above, the RTS/CTS dialogue is designed to solve the hidden terminal problem often incurred in ad hoc networks. The protocol's operation assumes the carrier sensing range, the transmission range, and the interference range to be all the same. Earlier network simulation tools have also assumed equal values for these three ranges, without incorporating corresponding physical channel measurements, or using physical channel models. As actual protocol implementations and experimentations started to appear, using modules such as the IEEE 802.11 WaveLAN cards, the actual values realized for these ranges have been obtained. For example, for the Orinoco IEEE 802.11 WaveLAN cards, operating at a 2 Mbps transmission rate, the receiver sensitivity is prescribed to be equal to –91dbm, and the power sensitivity is –81dbm. This implies that carrier sensing and transmission ranges assume distinctly different values. Later simulation tools modeled after the WaveLAN card have included these distinctive physical

---

[1] This work was supported by Office of Naval Research under Contract No. N00014-01-C-0016 and by the National Science Foundations under Grant No. ANI-0087148

parameters. We note that under IEEE 802.11 protocol, the RTS Threshold can be set to that when a terminal uses the RTS-CTS dialog only prior to sending MAC-frames that are longer than this threshold. In our study here, we consider only cases under which the RTS-CTS dialog is always used or always disengaged.

From the recent literatures [5][6], we find that in typically employed IEEE 802.11 radio modules, such as WAVELAN cards, the interference range is usually larger than transmission range and the carrier sensing range is even larger than the interference range (the carrier sensing range is about twice longer than the successful transmission range). However, the RTS/CTS mechanism incorporated into the IEEE 802.11 DCF MAC is designed to solve the hidden terminal problem by assuming equal values for the interference range, carrier sensing range and the transmission range. In contrast, note that if each node can sense a carrier 2-hops away, a hidden terminal problem would be resolved. It means under the basic CSMA/CA scheme, senders that are 2-hops away from each other will generally not transmit data at the same time since they can sense each other's carrier. So, the use of RTS/CTS handshaking would be redundant.

In this paper, we study the impact of using or disengaging the RTS/CTS dialogue in IEEE 802.11 DCF MAC protocol. We demonstrate the performance features of a linear topology network configuration with and without the RTS/CTS dialog. Comparisons are made based on considering the throughput performance realized by different stations situated in different location across the net.

## II. SIMULATION CONFIGURATION

The results reported in this paper are based on simulations using QualNet v 3.1 from Scalable Network Technologies [7]. QualNet is a discrete event parallel simulation environment which provides various detailed wireless protocols in its library.

In our simulation experiments, a basic linear topology is considered with different network loading scenarios, as shown in Fig. 1. The distance between each node is 250m, which can ensure every node can only reach its neighbor node directly. For example, node 2 is in reception range of node 1 and node 3, but not node 4. The channel data rate is set to 2 Mbps. Transmission and propagation delays are modeled, but processing delay is negligible.

Four sets of experiments are examined, as shown in Fig. 1 (a), (b), (c), (d). In each set of experiment, there are two flow of traffic going on at the same time. The arrows represent the direction of data flow. For example, in (a), node 1 is sending data to node 2 while node 3 is sending data to node 4 simultaneously. For each scenario, we evaluate the throughput performance by applying Constant Bit Rate (CBR) UDP type traffic as well as TCP type traffic. In CBR traffic cases, the packets are generated from application layer in a rate higher than the network capacity to force all the packets sent out back to back. On the other hand, for the TCP traffic, it will automatically aggressive to find the best one that will utilize the channel as best as possible. For both types, the packet sizes are fixed at 1460 bytes.

Note that the scenarios (b) {1→2, 4→5} and (d) {1→2, 5→4} will yield very interesting results. If we use the traditional view that the carrier sensing and transmission ranges are the same, these two links are not related to each other since the carrier sensing range is only one hop. However, in the modern view of these ranges, the sensing range is usually more than twice longer than the transmission range. Then, the two connections in pair {1→2, 4→5} can sense each other's carrier though node 2 can not send packets to node 4 directly. We will discuss more about this effect in the following paragraphs.

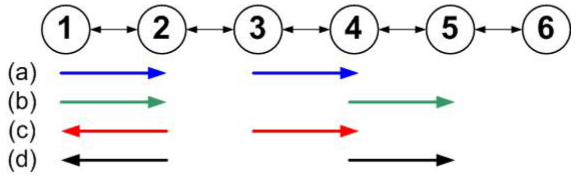


Fig.1 System configuration and network loading scenarios: hidden terminal topologies: (a) {1→2, 3→4}, (b) {1→2, 4→5}, and exposed terminal topologies: (c) {2→1, 3→4}, (d) {2→1, 4→5}

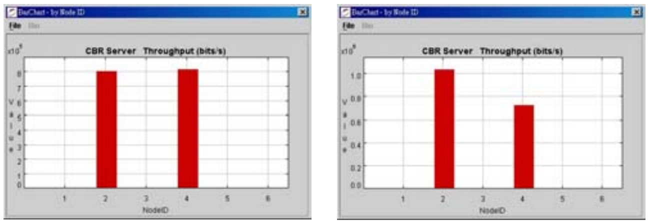
## III. THE EFFECT OF A LONGER CARRIER SENSING RANGE

We carry out extensive simulation based analyses using a simple linear topology to reveal the performance features of the IEEE 802.11 MAC protocol by applying Constant Bit Rate (CBR) UDP type traffic as well as TCP type traffic. Through these simulation results, the effect of using RTS/CTS dialog or not will be discussed.

### *A. Constant Bit Rate traffic with UDP*

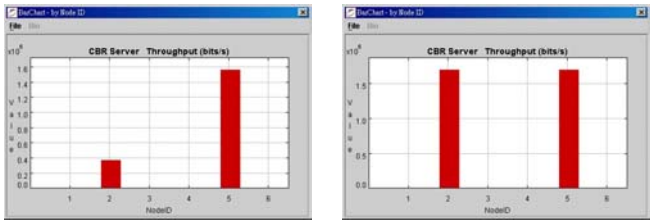
Fig.2 shows the throughput of scenario, {1→2, 3→4}, when using or not using RTS/CTS dialogue. With RTS/CTS, the throughput of link {1→2} and link {3→4} are both about 0.8 Mbps; while without RTS/CTS, the throughput of link {1→2} is higher than link {3→4}, but the average is about 0.9 Mbps, which is close to the case with RTS/CTS. Because the channel capacity is 2 Mbps, it is clear that these two simultaneous transmissions contend with each other to share the total channel capacity. Besides, RTS/CTS dialogue can help to obtain network fairness in terms of throughput.

The throughput performance of scenario, {1→2, 4→5}, is shown in Fig. 3. We observe that, with RTS/CTS dialogue, the throughput of link {1→2} is about 0.4 Mbps and the throughput of link {4→5} is about 1.5 Mbps. The latter almost capture the whole channel. If in old assumption, the transmission range is equal to carrier sensing range, these two links are supposed to be independent to each other. But since the carrier sensing range is about twice of the carrier sensing range, node 2 and node 4 can sense each other's transmission. When node 2 wants to send back CTS packets to node 1, oftentimes, it senses the channel is busy because node 4 is sending data packets to node 5. The RTS/CTS scheme can only make reservation for 1-hop. That means only node 3 will get RTS packets from node 4, but not node 2. So, between node 2 and node 4, RTS/CTS handshaking will not help, but only pure CSMA/CA contention. As we know, in CSMA, the nodes sending large packets usually win the contention. Node 4 is sending data packets while node 2 is sending CTS control packets, which are relatively very small. Thus, link {4→5} will capture most of the channel capacity. On the other hand, when disengaging RTS/CTS dialogue, both links achieve high throughput of around 1.7 Mbps, like two independent links. We can conclude that though node 4 can sense a carrier when node 2 is sending, the transmission of node 4 will not interfere with the reception of node 2 from node 1. In this case, the benefit of disengaging RTS/CTS is very obvious.



(a) with RTS/CTS (b) without RTS/CTS

Fig. 2 UDP throughput of {1→2, 3→4}

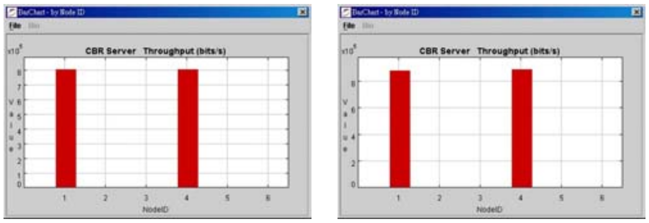


(a) with RTS/CTS (b) without RTS/CTS

Fig. 3 UDP throughput of {1→2, 4→5}

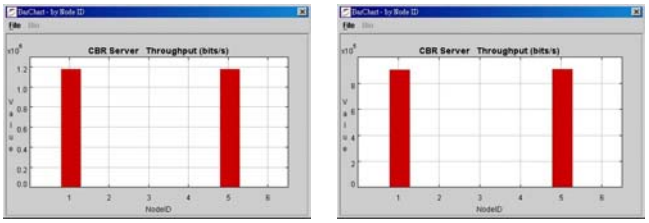
Fig. 4 and Fig. 5 show the throughput for exposed terminal topologies, connection pair, {2→1, 3→4}, and {2→1, 4→5}. In the case of {2→1, 3→4}, because node 2 and node 3 can hear each other's transmission, the channel capacity will just be shared by the two links no matter with or without applying RTS/CTS dialogue. When disengaging RTS/CTS dialogue, we can get 0.9 Mbps for both links, which is slightly higher than the 0.8 Mbps while using RTS/CTS. This is because the control packets will cause some overhead to the networks. For example, the CTS packets sent by node 1 can make the data transmission of node 3 to back off or node 1 will have difficulty in sending back CTS packets to node 2 because of sensing the transmission of node 3.

As for the scenario of {2→1, 4→5}, using RTS/CTS dialogue can achieve higher throughput of close to 1.2 Mbps instead. Note that between node 2 and node 4, only pure CSMA/CA contention exists. The control packets (RTS, CTS) are not going to help for contention, but they can help smooth out the data flow of each link independently.



(a) with RTS/CTS (b) without RTS/CTS

Fig. 4 UDP throughput of {2→1, 3→4}

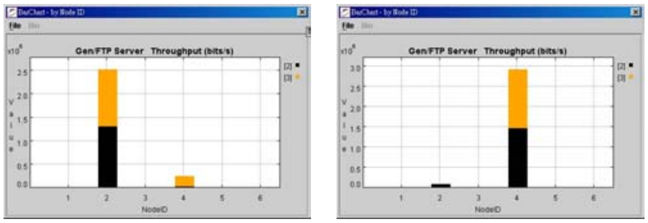


(a) with RTS/CTS (b) without RTS/CTS

Fig. 5 UDP throughput of {2→1, 4→5}

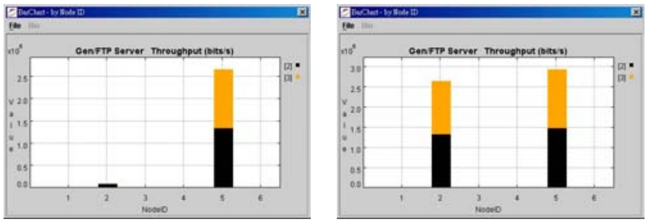
### *B. File transfer with TCP*

First, consider hidden terminal scenario, {1→2, 3→4}, where node 1 is transmitting TCP type traffic to node 2 while node 3 is transmitting TCP type traffic to node 4 at the same time. In Fig. 6, the black bars represent the throughput obtained when link {1 → 2} starts first and the yellow bars represent the throughput attained when link {3 → 4} starts first, with the use of the RTS/CTS dialogue in (a) and without it in (b). We observe see that when using RTS/CTS handshaking, the channel is captured by link {1 → 2} no matter which flow starts first. This is explained as follows: when node 4 wants to send back the TCP ACK packet to node 3, node 3 has difficulty to reply with the CTS packet since its transmission is always deferred by the other RTS or CTS packets from node 1, node 2, and node 4. In contrast, if we turn off the RTS/CTS dialogue, the channel is captured by the connection {3 → 4} instead. Node 2 now has difficulty in sending out the TCP ACK packets to node 1 since it keeps sensing the channel busy once connection {3 → 4} starts. We conclude that resolution to the TCP unfairness problem can be achieved through the removal or inclusion of the RTS/CTS dialog depending upon the underlying spatial configuration of the temporally-simultaneous connection pattern.



(a) with RTS/CTS (b) without RTS/CTS

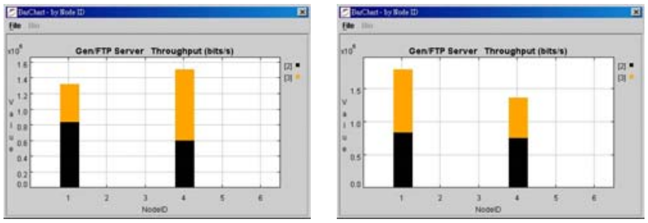
Fig. 6 TCP throughput of {1→2, 3→4}



(a) with RTS/CTS (b) without RTS/CTS

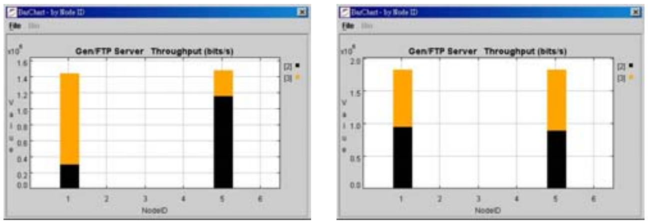
Fig. 7 TCP throughput of {1→2, 4→5}

For {1→2, 4→5} case shown in Fig. 7, the TCP connection {4→5} almost captures the whole channel capacity no matter which link starts first. The main reason is that node 2 has difficulty to reply CTS packets back to node 1. In this scenario, node 2 needs to send out several kinds of small packets, CTS packets, RTS packets and TCP ACK packets. But node 2 will sense the carrier of node 4, which is sending relatively very large data packets. Once the TCP connection of {4→5} starts, node 2 only has very few chances to jump in and capture the channel. If turning off the RTS/CTS dialogue, these two links can work well simultaneous (both can get about 1.5 Mbps throughput) while link {1→2} still gets less throughput because the TCP ACK packets sent from node 2 will need to contend with the data packet transmission of node 4 as well.



(a) with RTS/CTS (b) without RTS/CTS

Fig. 8 TCP throughput of {2→1, 3→4}



(a) with RTS/CTS (b) without RTS/CTS

Fig. 9 TCP throughput of {2→1, 4→5}

When it comes to exposed terminal scenarios, {2→1, 3→4} in Fig. 8 and {2→1, 4→5} in Fig. 9, situations become less complicated because of the symmetry natural. In the case of {2→1, 3→4}, the throughput performance has no much difference between using and not using RTS/CTS dialogue. Each link will roughly obtain 0.7 Mbps in some random fashion. As for applying RTS/CTS handshaking in the case of {2→1, 4→5}, the throughput each link can get depends on who jumps in the channel first. The link starts latter can achieve higher throughput. On the other hand, if removing RTS/CTS control dialogue, the two links can achieve similar throughput performance and also higher than with RTS/CTS in total because of less controlling overhead.

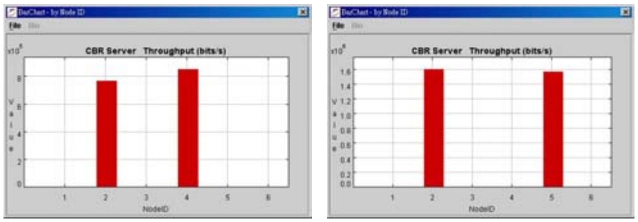
## IV. ADAPTIVE RTS/CTS EMPLOYMENT SCHEME

Through careful simulation study of many activity scenarios for the linear topology, we conclude that when using RTS/CTS handshaking, the problems generally occur when the receiver is unable to send back CTS control packets. In contract, if the RTS/CTS dialogue is disengaged, some packet collisions may happen and pure CSMA contentions may also cause some unfairness.

Therefore, we propose a new RTS/CTS control mechanism, which allows the system to derive the benefits observed for each case. Under our algorithm, each terminal keeps counting the number of "Waiting for CTS timeout" events. This value is updated in a sliding window fashion; thus, it is accumulated only over a limited recent period of time (the window). When a node attempts to send out a data packet, it first checks this count value. If this value exceeds a threshold, we turn off the RTS/CTS dialogue. This is a distributed control algorithm, so that every node decides on its own whether to use RTS/CTS or not dynamically according to its current situation.

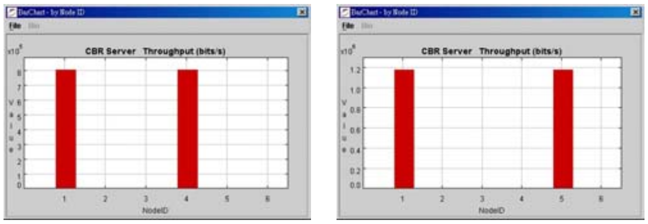
### *A. UDP Throughput When Applying Our Adaptive RTS/CTS Employment Scheme*

In the cases of {1→2, 3→4} when applying UDP traffic, the use of RTS/CTS dialogue leads to a more fair sharing of the channel capacity. Comparing Fig. 10 (a) and Fig. 2, we observe that our algorithm activates in this case the RTS/CTS dialogue to achieve a more fair behavior. Comparing Fig. 10 (b) and Fig. 3, we note that our algorithm will automatically choose to disengage RTS/CTS to obtain high throughput for both links. As for the exposed terminal scenarios, {2→1, 3→4} and {2→1, 4→5} shown in Fig 4, Fig.5, and Fig. 11, though there is not much performance difference realized by engaging or disengaging RTS/CTS, our algorithm still tends to make the network attain higher throughput performance.



(a) {1→2, 3→4} (b) {1→2, 4→5}

Fig. 10 UDP throughput for hidden terminal topologies when applying our RTS/CTS control algorithm

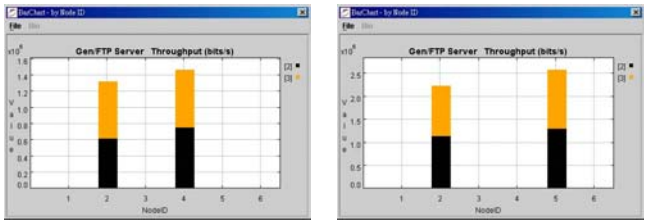


(a) {2→1, 3→4} (b) {2→1, 4→5}

Fig. 11 UDP throughput for exposed terminal topologies when applying our RTS/CTS control algorithm

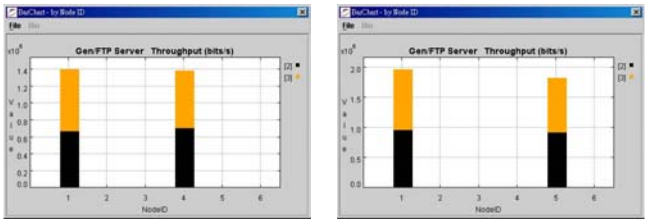
### *B. TCP Throughput When Applying Our Adaptive RTS/CTS Employment Scheme*

Applying our algorithm, in the case of {1→2, 3→4} with TCP traffic, as shown in Fig. 12 (a), the channel is utilized by these 2 connections almost equally compared to the throughput performance shown in Fig. 6. In the case of {1→2, 4→5}, our algorithm makes the network tend to not use RTS/CTS to achieve good throughput for both links (see Fig. 7 and Fig. 12 (b)). Moreover, for exposed terminal cases, our algorithm can perform very well in terms of throughput fairness. These examples demonstrate clearly the key performance advantages offered by our new scheme in improving both UDP and TCP throughput performance behavior.



(a) {1→2, 3→4} (b) {1→2, 4→5}

Fig. 12 TCP throughput for hidden terminal topologies when applying our RTS/CTS control algorithm



(a) {2→1, 3→4} (b) {2→1, 4→5}

Fig. 13 TCP throughput for exposed terminal topologies when applying our RTS/CTS control algorithm

## V. CONCLUSIONS

We study the impact of using or disengaging the RTS/CTS dialogue in IEEE 802.11 DCF MAC protocol, and demonstrate the performance features of a linear topology network configuration with and without the RTS/CTS dialog. The new mechanism proposed in this paper can improve throughput performance for CBR traffic flows and provides for better throughput fairness for TCP traffic flows.

In relation to the throughput unfairness problem induced by the IEEE 802.11 DCF MAC with RTS/CTS option, we note that in [6] the authors propose to resolve the TCP unfairness problem by slightly modifying the TCP mechanism. In turn, our new scheme makes a slight modification at the MAC layer. In this manner, our new mechanism can achieve performance improvement for UDP type and TCP type traffic.

We have carried out investigations in studying the impact on system throughput, while using the new RTS/CTS employment protocol introduced here, for different threshold levels. The selection of this level impacts the throughput and fairness behavior of the system. We have also studied grid based packet radio topologies to demonstrate the characteristics of the new protocol in contributing to fair throughput performance, under certain topological and local scenarios. These results will be presented in future papers.

## REFERENCE

[1] Phil Karn, MACA – A New Channel Access Method for Packet Radio, ARRL/CRRL Amateur Radio 9th Computer Networking Conference 1990.

[2] Bharghavan et al, MACAW: A Media Access Protocol for Wireless LAN's, SIGCOM 1994

[3] Fullmer et al, Floor Acquisition Multiple Access (FAMA) Packet-Radio Networks, SIGCOM 1995

[4] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE 802.11, 1999

[5] J. L. Sobrinho and A. S. Krishnakumar, Quality-of-Service in Ad Hoc Carrier Sense Multiple Access Wireless Networks, IEEE JSEC in communications, Aug. 1999

[6] Xu et al, Revealing the Problems with 802.11 Medium Access Protocol in multi-hop wireless ad hoc networks, Computer Networks, 2002.

[7] QualNet simulator, http://www.qualnet.com/