Packet Distribution based Tuning of RTS Threshold in IEEE 802.11

S.M. Rifat Ahsan, Mohammad Saiful Islam, Naeemul Hassan, Ashikur Rahman,
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh, 1000
Email: rifat_3n@yahoo.com, msislam04@csebuet.org, naeemulhassan@gmail.com, ashikur@cse.buet.ac.bd

***Abstract*— IEEE 802.11 Medium Access Control (MAC) protocol employs two techniques for packet transmission; the basic access scheme and the RTS/CTS-based reservation scheme. A parameter called *RTS Threshold* determines which scheme to use. If packet size is smaller than the RTS Threshold then the basic scheme is used otherwise the reservation scheme is used. With current standard, the RTS Threshold is fixed. In this paper, we, first point out the advantages and disadvantages of RTS/CTS based scheme. Then we state the problems of having a fixed RTS Threshold. Next, we present a numerical way to fix the RTS Threshold adaptively based on network traffic. The proposed adaptive scheme creates a balance between the basic scheme and the RTS/CTS based scheme and optimizes the network throughput. Considering multi-hop networks with hidden node problems we have validated our proposal through simulation.**

***keywords:* Ad-hoc Network; Medium Access Control; RTS Threshold; Adaptive; Cumulative Distribution Function; Packet Distribution.**

I. Introduction

In recent years, mobile and wireless networks has become popular due to its convenience. Among various types of wireless networks, there has been several areas where Adhoc network has proved its applicability due to its seamless intigration capability, flexibility and convenience. The performance of the Adhoc network highly depends on efficient channel sharing among the nodes. Several protocols has been defined for channel sharing. Among these, IEEE 802.11 Medium Access Control protocol is most widely accepted and used both in research areas and industry.

The standard for Wireless LAN's IEEE 802.11 specifies two medium access control mechanisms, DCF (Distributed Coordination Function), and PCF (Point Coordination Function). DCF defines two access mechanisms to employ packet transmission; the default, two-way handshaking technique called basic access and the optional four-way handshaking called RTS/CTS-based reservation scheme. Later one is used to reduce the possibility of collisions. Data transmission performance changes with different *RTS Threshold(RT)* values.

Our proposition is to evaluate the value of RTS Threshold dynamically based on network condition. To do so, we assume that the transmission range and the interference range are equal thus considering that hidden node problem prevails in the system. We also assume that network contains packets of different sizes. A Cumulative Distribution Function (CDF) is then used to determine the value of RTS Threshold. This idea is elaborately organized in the remnants of this paper as follows: in Section 2 we provide description of DCF; Section 3 talks about related works; Section 4 presents motivation of our work; in section 5 we present our proposal. In Section 6, we validate the proposal through simulation and in Section 7, we conclude the paper. Our approach relies only on packet distribution, irrespective of network size. The adaptive adjustment of RTS Threshold assures balance between higher collision penalty and better channel utilization.

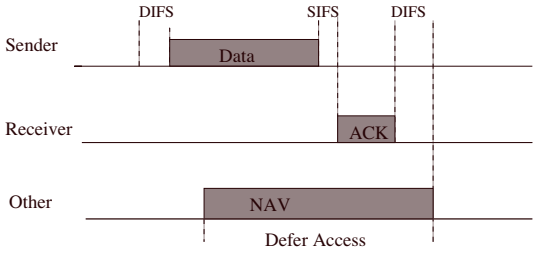


Fig. 1: Basic CSMA/CA Access Mechanism

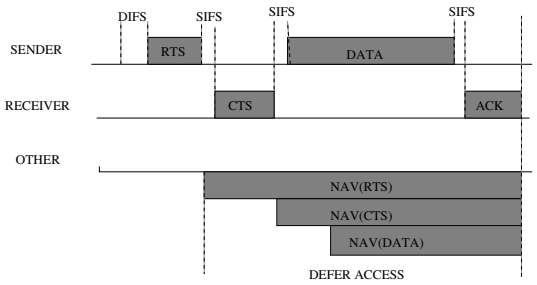


Fig. 2: CSMA/CA With RTS/CTS Mechanism

II. Description of DCF

The basic scheme and RTS/CTS based scheme of DCF are based on CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). In basic scheme, carrier sensing is done only by *physical sensing*, whereas an additional *virtual sensing* is used in the RTS/CTS-based scheme. Both sensing mechanisms are used to determine the state of the medium. For physical carrier sensing traditional CSMA/CA is used. It requires the nodes to first sense the channel to check whether it is idle for a DCF Inter-frame Space (DIFS) interval,

978-1-4244-7755-5/10/$26.00 ©2010 IEEE

and then attempt packet transmission. On the other hand, in *virtual carrier sensing*, RTS/CTS handshake and Network Allocation Vector (NAV) are used as shown in Fig. 2. The virtual carrier sensing employs RTS/CTS packets exchange for channel reservation. The sender at first transmits a Request-To-Send (RTS) frame to it's receiver. The receiver after receiving the RTS, sends a Clear-To-Send (CTS) frame in response. All other entity receiving a RTS or CTS or both mark the channel as busy by updating their NAV with prescribed duration of the talk time proposed in sender's RTS and/or receiver's CTS. Successfull receipt of CTS by sender ensures channel reservation for the conversation. After reservation, the sender transmits the DATA frame. The receiver after successfully receiving the DATA, send acknowledgement(ACK) frame. The conversation ends after the sender successfully receives the ACK. RTS, CTS, DATA and ACK packets are separated by a time interval called SIFS (Short Inter Frame Space) duration.

III. Related Works

During the past few years wireless network has evolved a lot and many works have been done to increase its performance. Many research papers have been published both for and against the use of RTS Threshold. Some of them have also pointed out various ways to adaptively tune the value of RTS Threshold. Almost every research has been done considering simple markov chain models with linear or some fixed network topology. Many have not considered the hidden node problem which is one of the most imporatnt design factor of any wireless networks with shared channel. Authors in [1] proved the superiority of RTS/CTS in highly loaded networks. His work is based on a 2-D Markov chain model. The authors in [2], have evaluated the dependency of the RTS/CTS scheme on network size, however, without providing any general expression for the RTS/CTS Threshold. But works in [3], [4] have pointed out that the RTS/CTS handshake does not work well as expected in theory.

Approaches to fix the value of RT can be clustered broadly into two categories: Static and Dynamic. Authors in [5] have performed analysis to determine RT values for maximum performance and proposed static value [RT = 0] for all nodes. However, they have considered only single hop environment. On the other hand, dynamic approaches are discussed in [6], [7], [5], [8], [9], [10]. In [6], authors proposed to set RT based on number of stations. Others, such as [7], [8] have emphasized on packet delivery ratio or transmission probability. The common practice in the literature [6], [7], [9], [10] is not to consider hidden node problem.

IV. Motivation

In the basic transmission scheme due to the fact that it only employs physical carrier sensing, the probability of collisions is highly increased. The problem of a station not being able to detect a potential competitor for the medium because the competitor is too far away (i.e. outside of the carrier sensing range) is called the hidden node problem. Depending on the geographical positioning of the nodes, hidden node problem can occur, because in wireless networks interference is location based. Resolving the hidden node problem becomes one of the major design consideration for MAC protocols of wireless networks. To alleviate the hidden node problem, Karn [11] proposed two way handshaking protocol known as the RTS/CTS handshaking mechanism. Bharghaven et al. [12] proposed an improved four-way protocol which employs a RTS-CTS-DATA-ACK handshaking mechanism.

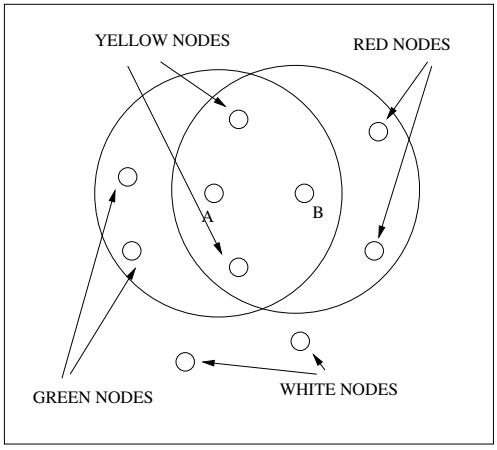


Fig. 3: Different area of perception from a transmission from A to B

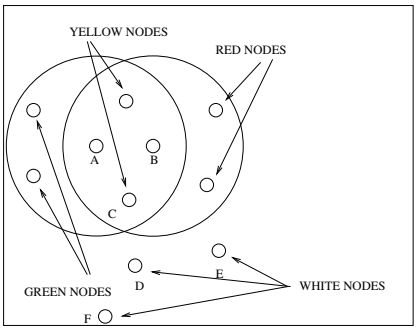
Though the RTS/CTS mechanism is able to solve the hidden node problem and reduce packet collision probability, it has several disadvantages. Authors in [13] have discussed several of these disadvantages like inhibiting non-interfering parallel transmission, false blocking and virtual jamming. To consider these problems lets consider a scenerio described in Fig. 3. Here, A is the sender and B is the receiver and they are using RTS/CTS mechanism for the data transmission. Let R_A and R_B denote the transmission range of A and B respectively. Again P_A and P_B denote the respective transmission areas. Let we divide all the possible nodes in four subsets denoted by four different colors. A node v is green, if $v \in P_A - P_B$; red, if $v \in P_B - P_A$; yellow, if $v \in P_A \cap P_B$ and white, if $v \in (P_A \cup P_B)'$.

A. Inhibiting non-interfering parallel transmission

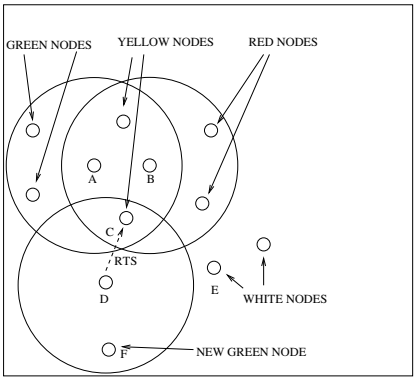
Suppose that A is the sender and B is the receiver (Fig. 3). If a green node, being outside the range of B, decides to transmit to a white node, it will cause no problem to ongoing conversation between A and B. Similarly a red node, located outside the range of A, is technically able to receive from white nodes while B is receiving from A. Only yellow nodes are in a truely restricted zone, not being able to transmit or receive while A is sending data to B.

So, the RTS/CTS mechanism blocks some transmissions that could be carried out in parallel. Thus, while reducing the probability of collisions, this mechanism is actually reducing the throughput.

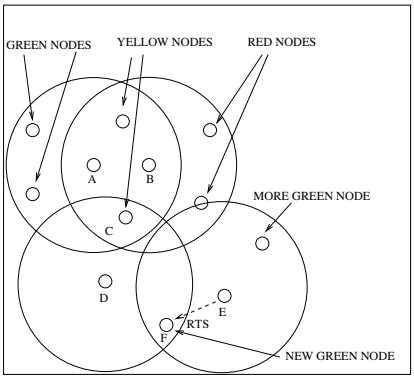
978-1-4244-7755-5/10/$26.00 ©2010 IEEE



(a)



(b)



(c)

Fig. 4: False Blocking

B. False Blocking

Lets consider the scenario in Fig. 4 (a). While A is transmitting to B, all red, green and yellow nodes are blocked. But, white nodes are free to transmit and receive. Let a white node D is trying to transmit a data to a yellow node C. For this to happen, D will send RTS packet, as in Fig. 4 (b) to C. But C, being blocked can not respond by sending a CTS to D. So D will initiate backoff. But D's initial RTS will paint all the nodes in it's transmission range green, so they will be remain blocked during the entire duration stated in the RTS though no effective conversation will take place. Interestingly, this false blocking will propagate if some other white nodes will try to send data to the new green nodes. This is illustrated in Fig. 4T (c), where node E is trying to communicate with node F by sending RTS. Thus a huge number of nodes can remain blocked in this way.

C. Virtual Jamming

Virtual Jamming is a kind of Denial of Service attack that uses false blocking. A malicious node can deliberately send short RTS packets at some interval announcing long transmissions which will never occure. By using this method, the node can jam a large portion of network by using relatively low power. Though a single malicious node can create a chain of blocking events, carefully positioned nodes can cripple the whole network with ease.

From the above discussion, we can realize that the RTS/CTS-based reservation scheme trades some problems (like the hidden node problem) for others (inhibition of parallel transmissions and exposure to virtual jamming attacks). While elimination of the interference caused by hidden nodes does have a positive impact on the network performance, the problems introduced by the RTS/CTS mechanism will tend to counterbalance those benefits.

Therefore we need to continue with the both schemes (i.e. the basic scheme and the RTS/CTS-based scheme) in a balanced way to reduce the probability of collision and at the same time to avoid the problems of RTS/CTS mechanism. There are some proposals that try to fight the problems after they occur. But in our consideration this is a wrong way to confront the problem. According to our view it is better to organize the system so that the network faces the problems with relatively low frequency. By using a scheme which generates the minimal number of RTS packets in the first place we can lower the problem occurance frequency. In this scheme, the balance can be achieved if we can avoid using the RTS/CTS mechanism for a certain $\eta \times 100$ percent of the packets and use RTS/CTS mechanism for the rest $(1-\eta)\times 100$ percent. η can be tuned to achieve best performance. Our simulation result shows that the best value of η lies between $0.6 - 0.8$. Moreover the smaller η percent packets should be transmitted using basic schemes without the RTS/CTS protection because the collision probability is less for the small sized packets.

V. The Proposal

When the payload is large, the probability of collision is high, so it is beneficial to use RTS/CTS conversation. On the other hand, if the payload is small, the probability of collision is comparatively low and it is better to go with the basic scheme. Again, using a small value causes RTS packets to be sent more often, consuming more of the available bandwidth, therefore reducing the apparent throughput of the network packet. However, the more RTS packets that are sent, the quicker the system can recover from interference or collisions - as would be the case in a heavily loaded network, or a wireless network with much electromagnetic interference.

RTS Threshold is not specied by IEEE 802.11 standard and has to be managed separately by each node. Traditionally RTS

978-1-4244-7755-5/10/$26.00 ©2010 IEEE

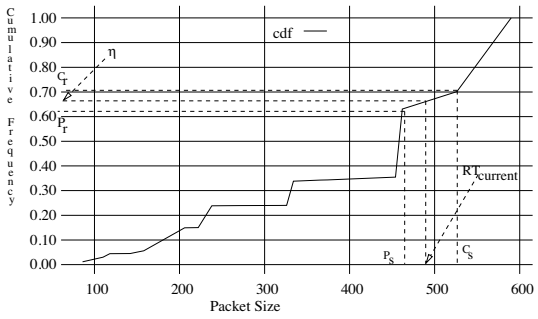


Fig. 5: RTS Threshold Calculation Using CDF

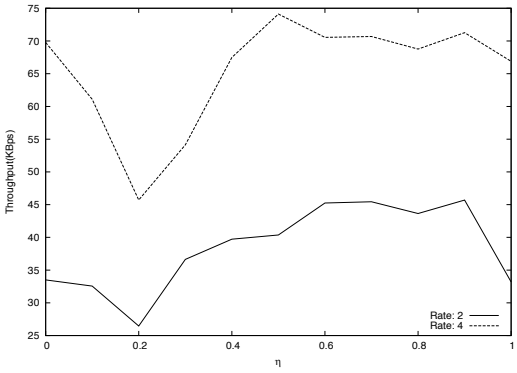


Fig. 6: η vs Throughput, number of nodes 50 and packet sizes 8-512 Byte

Threshold is set to a *fixed* small value. But setting to a fixed small value is not optimal for all network situations and it can not effectively inter-mix the two schemes over all the packets flowing through the network. The problem of having fixed RTS Threshold can be described as follows. As the packet size of a network is random and not known before, with a fixed RTS Threshold it may be happen that all the packets in the network are having sizes larger than that fixed RTS Threshold value. Consequently, all the packets will be transmitted using RTS/CTS mechanism. Also the other way around may happen, for example all the packets may have sizes smaller than the fixed value of RTS Threshold, causing all of them to be transmitted using the basic scheme. In both the cases we can not use the η percent rule, therefore can not intelligently inter-mix both schemes.

On the contrary, if we can adaptively set the value of RTS Threshold based on network traffic then intermixing these two schemes can be easily achieved. Our main proposal is too use basic scheme for *relatively* small sized packets and use RTS/CTS mechanism for *relatively* large size packets. To incorporate this idea, the value of RTS Threshold needs to be intelligently set to a value such that $\eta \times 100$ percent of packet's size fall below that value. Mathematically it can be described as follows: suppose the sizes of packets flowing through a node are $s_1, s_2, s_3, \ldots, s_n$ (in ascending sorted order) with probability $p_1, p_2, p_3, \ldots, p_n$. Then the value of RTS Threshold is set to a value such that:

$$Pr\{S \leq RTS\ Threshold\} = \eta$$

where S is a random variable denoting packet size.

A node at first learns the sizes of the packets it is generating or forwarding as an intermediate node for a certain time interval. Then it sets the value of RTS Threshold for the next interval using the above equation. It also continues its learning process in the subsequent intervals and adjusts the RTS Threshold dynamically from one interval to another. The details of the algorithm is as follows.

We equip each node with a *traffic observer* which runs in the background. Having received a packet p with size s_i, the node increments the frequency count f_i for the packet size s_i. Every δ seconds the traffic observer wakes up and calculates a new value of RTS Threshold based on the packet distribution statistics collected within the last δ interval. To calculate new RT value, the observer at first rearranges the frequency count of packet sizes in increasing order of packet size. Suppose the total number of different packet sizes is n and S is a random variable denoting packet size. Let us denote P_i be the probability that a packet's size is less than or equal to s_i. Then, mathematically:

$$P_i = Pr\{S \leq s_i\} = \left(\frac{\sum_{j=1}^{i} f_j}{\sum_{k=1}^{n} f_k}\right)$$

Note that using the above equation $P_n = 1$. Actually P_i is the *cumulative distribution function* (CDF) for the different packet sizes which is depicted in Fig. 5.

Using this CDF, calculation of new RTS Threshold is pretty simple. Let, P_r is the greatest probability less then η, P_s is the packet size at P_r, C_r is the least probability greater then η, and C_s is the packet size at C_r. Using linear interpolation the traffic observer calculates the current RTS Threshold using the equation below (see Fig. 5):

$$RT_{current} = \left\lfloor P_s + \frac{(\eta - P_r) * (C_s - P_s)}{(C_r - P_r)} \right\rfloor$$

The average RTS Threshold is updated as

$$RT_{average} = \lfloor \alpha * RT_{prev} + (1 - \alpha) * RT_{current} \rfloor$$

where, RT_{prev}=previous RTS Threshold and α controls the relative weight of recent and past history of RTS Threshold calculation. The value of α lies between 0 to 1.

VI. Evaluation by Simulation

We use NS-2 simulator to validate our proposal. For simulation a rectangular area of 1000*1000 square unit is considered. Each experiment is 1000 time units long. For each experiment we have created 5 network topologies and have taken the average of the 5 results. All the experiments are done using CBR traffic sources. The environment we have considered is static, multi-hop with the presence of hidden nodes. For traffic generation we have used a uniform packet size generator with min and max size specified for each experiment. We experiment under three network conditions; light density (50 nodes), medium density (75 nodes) and high density (100

978-1-4244-7755-5/10/$26.00 ©2010 IEEE

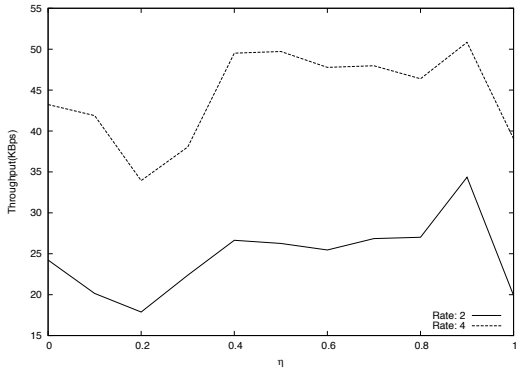


Fig. 7: η vs Throughput, number of nodes 75 and packet sizes 8-512 Byte

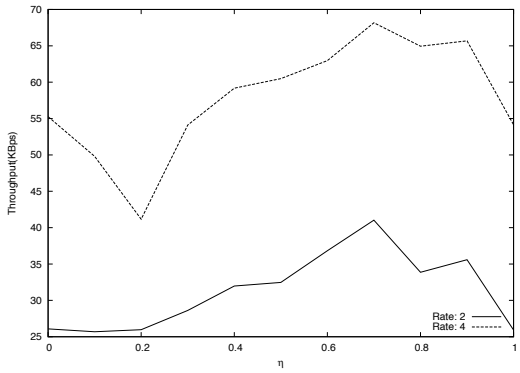


Fig. 8: η vs Throughput, number of nodes 100 and packet sizes 8-512 Byte

nodes). For performance measurement we have calculated the overall (aggregate) throughput of all the nodes. We compare throughput at different η. Our objective of the simulation is to compare the troughput in Basic Scheme (with all packets using RTS/CTS mechanism), Scheme where no packet uses the RTS/CTS mechanism and schemes with different percentages of packetes using RTS/CTS mechanism.

In static network , if nodes have similar distribution of payload, RTS Threshold will converge to a fixed value eventually from its default value 0. Moreover if payload is distributed within a range, it is also reflected in RTS Threshold value. In Fig. 6, 7, and 8 overall throughput for different η values are shown. When the value of η is set to 0 then all packets are transmitted with RTS/CTS conversation. $\eta = 1$ indicates that all packets are transmitted with basic scheme without any RTS/CTS. The current IEEE 802.11 standard transmits all packets with RTS/CTS dialogue. So the value of the curve with $\eta = 0$, actually represents the performance of the current standard. It can be observed that the throughput is high for $0 \leq \eta \leq 1$ and in most of the experiments the optimum value is found for $0.6 \leq \eta \leq 0.8$.

Our next set of experiments show the adaptiveness of the algorithm. It is shown with these experiments how the algorithm successfully tunes the value of the RT, when the

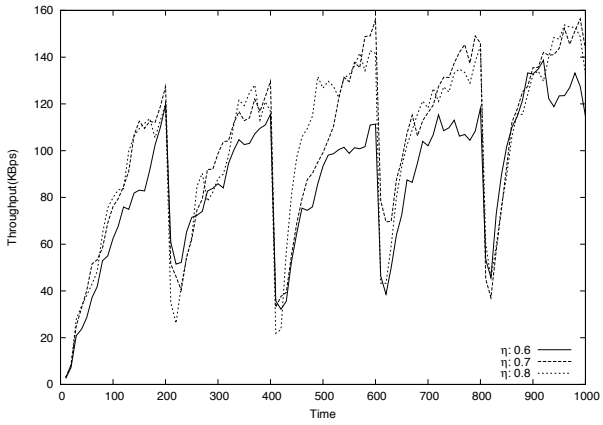


Fig. 9: Instantaneous Throughput: number of nodes 50, window size $\delta = 50$

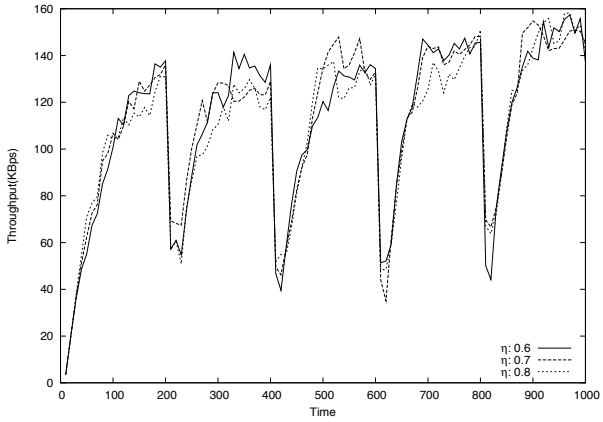


Fig. 10: Instantaneous Throughput: number of nodes 75, window size $\delta = 50$

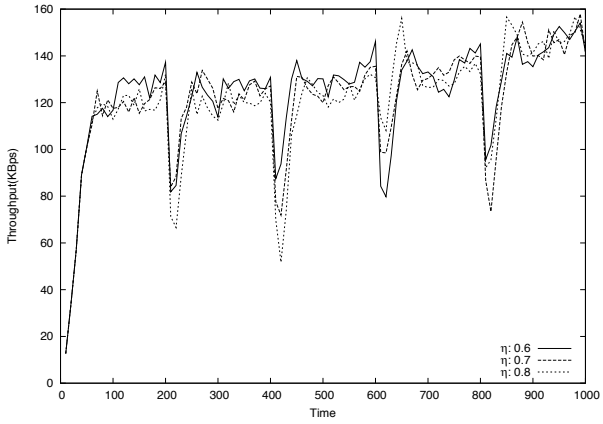


Fig. 11: Instantaneous Throughput: number of nodes 100, window size $\delta = 50$

978-1-4244-7755-5/10/$26.00 ©2010 IEEE

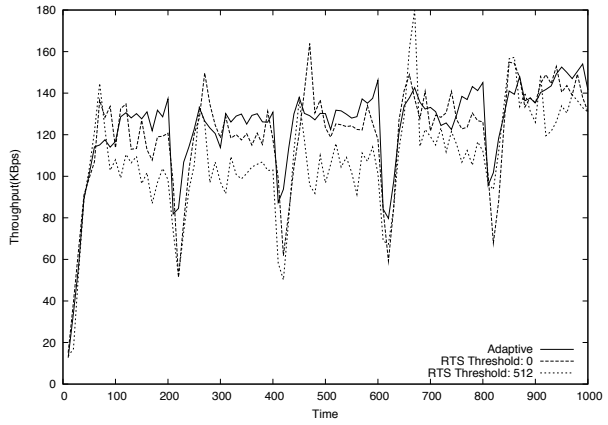


Fig. 12: Comparision between Adaptive and Fixed RTS Threshold Scheme

packet distribution of network (sensed by each node) changes, so that always the optimum value can be used by the nodes. The algorithm always reflects the current packet distribution of the network. In the graph of Fig. 9, 10, and 11 we see the effect of change in throughput for the change in packet size distribution. Here we have carefully devised the CBR generators to create 5 non overlapping time intervals of 200 time unit each. In each of the intervals, nodes generate packets of different sizes. In first interval, packet size is uniformly distributed between $8-512$ byte. In second interval it is from $16-512$ byte and so on. This type of packet generation reflects the uncertain nature of network about the packet distribution. We have turned off all the generators at the end of each interval and a new set of generators are started uniformly in the next interval. This has created a *sea-saw* effect on the performance curve. There is a sharp fall in throughput at the end of each interval which rises again in the next new interval as more and more traffic sources are started.

In Fig. 12 our proposed adaptive scheme is compared with the scheme that uses *fixed* RTS Threshold value of 0 and 512 in an environment with 100 nodes and different packet distributions. It is clear from the experiment that the proposed algorithm can better cope with change of packet distribution and can give better performance compare to fixed RTS Threshold value. It can be observed that our proposed algorithm has two way benefits than the fixed values. When the packet distribution changes drastically, as shown in the end of each interval of 200 time units, there is big drop of throughput. But as shown in the figure, while using our algorthim the drop is much lower than the fixed values. The second benefit comes from the fact that, our algorithm being able to cope with the change quickly can show better throughput when the condition is stabled, which is shown in the middle of each interval. This two way process makes our algorithm to perform better.

VII. Conclusion

In this paper we propose an adaptive scheme to effectively use RTS/CTS handshaking in IEEE 802.11. We proposed a dynamic way to adjust the RTS Threshold based on current packet distribution of the network. Through simulation we validate our proposal. Evaluated results in NS-2 showed that the proposed adaptive scheme achieves better result than the current IEEE 802.11 scheme. We further intend to experiment on mobile networks, where mobility would cause the increase of hidden node problems and packet distribution may change more drastically.

References

[1] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal of Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.

[2] A. B. P. Chatzimisios and V. Vitsas, "Packet delay analysis of the IEEE 802.11 MAC protocol," *IEEE Electronic Letters*, vol. 39, no. 18, pp. 1358,1359, 2003.

[3] S. Xu and T. Saadawi, "Revealing the problems with 802.11 medium access control protocol in multi-hop wireless ad hoc networks," *Computer Networks*, vol. 38, no. 4, pp. 531–548, 2002.

[4] M. G. K. Xu and S. Bae, "Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks," *Ad Hoc Networks Journal*, vol. 1, no. 1, pp. 107–123, 2003.

[5] J.-H. C. F. S.-T. Sheu, T. Chen, "The impact of rts threshold on IEEE 802.11 mac protocol," in *IEEE International Conference on Parallel Distributed Systems (ICPADS)*, 2002, pp. 267–272.

[6] S. W.-W. G. Shaohu Yan, Yongning Zhuo, "Adaptive RTS threshold for maximum network throughput in IEEE 802.11 DCF," vol. 5284, 2004, pp. 332–343.

[7] B. l. X. F. H. Jun Liu, Wei Guo, "RTS threshold adjustment algorithm for IEEE 802.11 DCF," in *6th International Conference on ITS Telecommunications*, 2006, pp. 654–658.

[8] N. N. K. K. A. T. N. S. Mostafa Mjidi, Debasish Chakraborty, "A new dynamic scheme for effcient RTS thresholdhandling in wireless networks," in *22nd International Conference on Advanced Information Networking and Applications*, 2008, pp. 734–740.

[9] P. Chatzimisios and A. C. Boucouvalas, "Improving performance through optimization of the RTS/CTS mechanism in IEEE 802.11 wireless lans," in *International Conference on Communication Systems, Networks and Digital Processing (CSNDSP 2004)*, 2004, pp. 375–378.

[10] Y.-C. K. Huei-Jiun Ju, Izhak Rubin, "An adaptive RTS/CTS control mechanism for IEEE 802.11 MAC protocol," *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, vol. 2, pp. 1469– 1473, 2003.

[11] P. Karn, "MACA-a new channel access method for packet radio," in *9th Computer Networking Conference on ARRL/CRRL Ameture Radio*, 1990, pp. 134–140.

[12] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless lan's," in *ACM SIGCOMM '94*, 1994, pp. 212–225.

[13] A. Rahman and P. Gburzynski, "Hidden problems with the hidden node problem," in *23rd Biennial Symposium on Communications- QBSC 2006*, 2006, pp. 270–273.

978-1-4244-7755-5/10/$26.00 ©2010 IEEE