International Journal of Research in Computer Science
eISSN 2249-8265 Volume 3 Issue 5 (2013) pp. 1-9
www.ijorcs.org, A Unit of White Globe Publications
doi: 10.7815/ijorcs. 35.2013.069

CHANNEL AWARE MAC PROTOCOL FOR MAXIMIZING THROUGHPUT AND FAIRNESS

[1]G.Suseendran, [2]E.Chandrasekaran

[1]Ph.D. Research Scholar, Department of Mathematics
Presidency College (Autonomous), Chennai, INDIA
Email: suseendranphd@gmail.com

[2]Associate Professor, Department of Mathematics
Presidency College (Autonomous), Chennai, INDIA
Email: e_chandrasekaran@yahoo.com

Abstract: *The proper channel utilization and the queue length aware routing protocol is a challenging task in MANET. To overcome this drawback we are extending the previous work by improving the MAC protocol to maximize the Throughput and Fairness. In this work we are estimating the channel condition and Contention for a channel aware packet scheduling and the queue length is also calculated for the routing protocol which is aware of the queue length. The channel is scheduled based on the channel condition and the routing is carried out by considering the queue length. This queue length will provide a measurement of traffic load at the mobile node itself. Depending upon this load the node with the lesser load will be selected for the routing; this will effectively balance the load and improve the throughput of the ad hoc network.*

Keywords: *Mobile ad hoc networks (MANETs), MAC protocol, Throughput and Fairness.*

I. INTRODUCTION

A. MANET

Mobile ad hoc networks (MANETs) are self-organizing networks that provide an efficient solution when centralized control is infeasible like emergency and rescue operations, disaster-relief efforts, etc. A key design objective in MANETs is to achieve high network throughput while maintaining energy-efficient wireless communications for mobile terminals. To achieve this objective, efficient design of the MAC layer is necessary in order to resolve channel-contention and reduce packet collisions. Several important issues like energy efficiency, fairness, or quality of service (QoS) provision need to be carefully considered when designing MAC protocols for MANET. For the system's quality of service (QoS) requirements, a MAC protocol for ad hoc networks shares the medium and the available resources in a distributed manner, and allows for efficient interference management. Despite all the research done on MAC protocols thus far, only a limited number of works have considered an interference channel that is both stochastic and continuous in time [1][2][3].

B. Fairness Issues in 802.11

The IEEE 802.11 defines two MAC protocols, i.e., Point Coordination Function (PCF) and Distributed Coordination Function (DCF), but only DCF is used in MANETs since PCF requires base stations.

DCF is a CSMA/CA based protocol, this protocol has two main components: Carrier Sensing (CS) and Collision Avoidance (CA). Since CS and CA work in a distributed manner without having precise information, they cause unfairness. The CA algorithms and imprecise EIFS problem, the hidden-terminal problem and signal capture, which are the two most important issues in wireless networks, also affect the MAC fairness. Unfairness could result from different opportunities of channel access.

There are two major sources for unequal channel access opportunities: the backoff mechanism and location. While the backoff mechanism is broadly used in MAC protocols for MANETs to reduce collisions and achieve high channel efficiency, it always favors the node that just successfully seized the channel. As a result, different nodes may use different backoff window, leading to different transmission probabilities and consequently short-term unfairness as well as long-term unfairness. To achieve fairness among all nodes, the network's aggregate throughput, namely, efficiency often has to be sacrificed. The MAC protocol and the routing the overall performance of protocol interact with each other, which in turn affect a MANET, it is very important to characterize the interaction between the MAC fairness and the routing mechanism. Fairness in wireless ad hoc networks has been studied under various network scenarios. Many algorithms has been proposed to achieve fairness among single-hop flows, but they do not consider multi-hops flows, which reflect the reality in wireless ad hoc networks [3][4][5].

C. Problem Identification

In our previous paper, we have proposed an interference reduction technique in MANET using hidden markov model (HMM). Initially, node that receives the RREQ packet calculates its received signal power and compares with predefined threshold values namely, P_{min} and P_{max}. Based on these comparison initial transmission values are set. During transmission, RTS message is sent using initial transmission power. RTS includes interference value predicted by the source. Nodes use hidden markov model (HMM) to predict their interference value. On receiving RTS, the destination calculates its interference and sends it along with CTS to the source with the power level P_{ini}. While receiving the CTS, the source calculates the minimum power required for transmitting data using the interference value of the destination and transmits the data with that power value. Finally, the destination uses the interference value of the source for transmitting the ACK message.

Drawbacks

- The proposed technique does not deal with the proper channel utilization.

Estimation of queue length is not considered which results in the throughput and fairness of the MAC protocol.

II. RELATED WORK

Fan Wang et al., in paper [1] have proposed a distributed, single-channel MAC protocol (GMAC) that is inspired by game theory. In GMAC, each transmitter computes a utility function that maximizes the link's achievable throughput. The utility function includes a pricing factor that accounts for energy consumption. GMAC allows multiple potential transmitters to contend through an admission phase that enables them to determine the transmission powers that achieve the Nash equilibrium (NE). The advantage of this approach is that GMAC significantly improves the network throughput over the 802.11 scheme and over another single-channel power-controlled MAC protocol (POWMAC). These gains are achieved at no extra energy cost.

P. Priakanth et al., in paper [4] have proposed a topology-aware MAC protocol which attains fairness across multi-hop flows and minimizes the energy. The proposed protocol estimates the feasible bandwidth and channel condition of each wireless link by monitoring its traffic and calculates a combined score. The score is sent along with the data packets of the flows using any routing protocol. Then transmission is allowed only for those nodes with high scores. Nodes attempting to access the wireless medium with a low score will be allowed to transmit again when their score becomes high.

Zhifei Li et al., in paper [5] by using static routing they have first studied the MAC fairness by considering three factors including the hidden terminal, capture, and imprecise EIFS, which are all related to the signal attenuation property of the wireless networks. Since the MAC protocol and the routing protocol interact with each other, we then study the interaction between MAC fairness and routing mechanism by replacing static routing with AODV and DSDV. The main lessons learnt from this studies include: (i) should distinguish between the MAC fairness and the overall system fairness, (ii) in studying the MAC fairness, should isolate the factors (e.g., routing protocol) that are not directly related to the MAC layer, and (iii) to study the overall system fairness, in addition to the study of MAC fairness, we should also consider the effects of many other factors such as the routing mechanism, traffic pattern, mobility pattern etc.

Xuemei Gao et al., in paper [6] have proposed a load-aware routing protocol using two load metrics for route selection, which include MAC layer channel contention information, and the number of packets in the interface queue. MAC layer contention information provides an accurate estimation of neighbor nodes' state, and queue length provides a measurement of traffic load at the mobile node itself. This load-aware routing protocol can effectively balance the load and improve the performance of the ad hoc network. Impacts of these load metrics on the routing performance are studied.

Mun Choon Chan et al., in paper [7] they present, CaSMA, a scheduling mechanism for mobile ad hoc networks (MANETs) that takes into account both the congestion state and end-to-end path duration. This scheduling mechanism is termed Channel aware scheduling for Mobile Ad hoc networks (CaSMA), where the term channel-aware is used to indicate both the congestion state and the end-to-end path duration. CaSMA is complimentary to packet scheduling scheme that utilizes only local channel information, and can be added to these schemes.

III. PROPOSED WORK

A. Overview

We are extending the previous work by improving the MAC protocol to maximize the Throughput and Fairness. In this work we are estimating the channel condition and Contention for a channel aware packet scheduling and the queue length is also calculated for the routing protocol which is aware of the queue length. The channel is scheduled based on the channel condition and the routing is carried out by considering the queue length. This queue length will provide a measurement of traffic load at the mobile node itself. Depending upon this load the node with the lesser load will be selected for the routing; this will effectively balance the load and improve the throughput of the ad hoc network.

B. System Design

In our previous paper, we have proposed an interference reduction technique in MANET using hidden markov model (HMM). Initially, node that receives the RREQ packet calculates its received signal power and compares with predefined threshold values namely, Pmin and Pmax. Based on these comparison initial transmission values are set. During transmission, RTS message is sent using initial transmission power. RTS includes interference value predicted by the source. Nodes use hidden markov model (HMM) to predict their interference value. On receiving RTS, the destination calculates its interference and sends it along with CTS to the source with the power level P_{ini}. While receiving the CTS, the source calculates the minimum power required for transmitting data using the interference value of the destination and transmits the data with that power value. Finally, the destination uses the interference value of the source for transmitting the ACK message.

We are extending this work by improving the MAC protocol to maximize the Throughput and Fairness. In this extension we are estimating the channel condition and Contention for a channel aware packet scheduling and the queue length is also calculated for the routing protocol which is aware of the queue length.

1. Channel Condition: In this approach the end to end channel quality is represented in the form of path lifetimes. The channel state keeps changing continuously hence the end to end path will be valid for a temporary period of time. The term path lifetime is used to define the time interval for which the path associated for a flow is valid or exists. Suppose the lifetime of each and every link of path P from node i to node j is estimated as $l_1, l_2, \ldots, l_n$, then the path lifetime is given by;

$$P_j = min(l_1, l_2, \ldots, l_n)$$

2. Global Ideal Scheduler: Now let us consider a simple model with multiple flows over a single bottleneck link where we have a single scheduler. After the single shared link (with infinite lifetime), these flows use different (non-shared) links with different lifetime. Let S_g be the global scheduler, which will schedule the flows (q flows). Let us consider a single continuous period *"p"* of *"q"* flows, with arrivals within this continuous period, and no further arrivals.

Use of *QS/RLT* to Approximate Ideal Scheduler

The scheduling approach is considered where queue with maximum value of $\frac{QS}{RLT}$ is chosen, where QS is queue size and RLT is the residual life time. The $\frac{QS}{RLT}$ acts as a request rate, hence serving queue which has higher $\frac{QS}{RLT}$ values first will result in providing higher priority to flows which takes short-lived paths.

Considering $\frac{QS}{RLT}$ is not sufficient to provide equal proportion of service, so we are considering a simple model, which is a single snap-shot in time where we have *"n"* flows with each flow *"I"* having workload (number of packets) as W_i . Let the service time for all packets be 1 time unit. Now all the flows have T (minimum packet inter-arrival time) and C (maximum packet transmission time) set to *1*, the $P_i = W_i$ i.e. RLT for each flow will be the same as W_i . Let the maximum number of packets the scheduler serves in the given time duration (or the maximum duration of time snapshot) be maximum of W_i values, termed as W_{max} . Let the of packets served for flow i be X_i and use R_{sum} to represent sum of all R_{is} .

We know that QS either decreases or remain the same, and RLT strictly decreases. Therefore request rate $\left(\frac{QS}{RLT}\right)$ can either remain the same or increase. Using the above model and notations we can rewrite request rate as

$$\frac{W_i - X_i}{W_i - \sum_n X_j} \qquad (1)$$

Let *α (0 < α < 1)* be the proportion (percentage) of of W_i service that any flow *"i"* receives at any given time within the considered time duration. The important point to note here is that, there is no one-to-one mapping between the request rate considered and proportion of service received *(α)*. That is, if a flow *i* has greater request rate than the other flow *j*, then it may not mean that amount of the service (proportionately, α) received by the flow *i* is lesser than *j*. When the $P_i^{'}$ s varies to a larger extent, the proportionate amount of services received by flows can also vary to a larger extent.

For a special case where $P_i^{'}$ s are same, if a flow has received lesser proportion of service than the other flow, then its request rate will always be higher than the other flow. Under these conditions, it can be shown that serving by $\frac{QS}{RLT}$, results in fair distribution of service.

If the P_i values vary then we should avoid the condition where short-lived flows can receive proportionately greater service. This is achieved by having an additional parameter termed as eligible − service, for each flow. This eligible − service for any flow $\underline{i}$ is equivalent to, $\frac{W_{\max}}{W_{sum}}$ and is computed by considering the $R_i s$, which is given as follows:

$$\frac{\left(W_i * \frac{C_i}{T_i}\right)}{\sum_{j=1}^{n} W_j * \frac{C_i}{T_i}} * \left(W_{\max} * \frac{C_{\max}}{T_{\max}}\right) \qquad (2)$$

C_{max} and T_{max} indicate the maximum possible values of C and T, respectively. The first term indicates the ratio of the work to be performed for a flow i and the total amount of work considering all flows. Whereas, the second term indicates the maximum work that can be done, and this term, in practice, is related to the maximum wireless link rate. We update the eligible − service parameter only when new flows arrive or existing flows leave. The priority is given to flows by considering both the request rate and eligible service. Higher priority is given to flows whose request rate is high, and which has not yet received its eligible-service. This parameter will ensure that flows do not receive greater service (in proportion) at the cost of other flows.

3. Schedulability: Now we will enhance the approximation of ideal scheduler by considering end-to-end packet scheduling. A set of flows Γ is said to be "schedulable" (S) if none of the flows has packets queued in the intermediate nodes at the end of their respective continuous periods. Any set of flows at a node that are schedulable over a link is termed as "schedulable set".

For the flow schedulability the following two cases are considered.

Case 1: We have to consider that given a set of n flows *Γ = (T*$_i$*, C*$_i$*, o*$_i$*), i = 1, 2, . . .n, how* many of them (m, m ≤ n) are schedulable over a link? (Schedulable set).

First, let us begin with the schedulable set (ζ). A schedulable set is derived as given below. Let us assume that a node has n flows, of which it has to choose m flows to form a schedulable set. The necessary condition for a set of flows to be schedulable over a link is given as

$$\sum_{i=1}^{m}\left(\frac{C_i}{T_i}\right) \leq 1 \qquad (3)$$

The above necessary condition in rewritten interms of the packets scheduled. Since the minimum packet interarrival time of a flow i is T_i, there are at most (o_i) Ti packets arrived over channel i during the interval, and which need at most (o_i) T_i C_i units of time to transmit. Now the summation of this time for all the m flows should be less than the r_{max}, which is written as

$$\sum_{i=1}^{m}\left(\frac{P_i}{T_i}\right) \leq W_{\max} \qquad (4)$$

As there are different combinations that are possible in choosing m flows out of n flows (Cm n). The value of m is dependent on the C_i and T_i values. For example, value of m becomes smaller for smaller values of T_i. Hence, we have to decide on a specific way to choose m flows out of n flows. We choose the m flows considering the residual lifetime values of the flows. Scheduling based on residual lifetime is similar to earliest deadline scheduling (EDF). We sort all the n flows in terms of the increasing residual lifetime, and from this sorted set we choose the first m flows. These m flows from our schedulable set ζ.

Case 2: Suppose there are n flows *Γ = (T*$_i$*, C*$_i$*, o*$_i$*), i = 1, 2, . . .n,* of which *m* flows form a schedulable set *ζ*. Now, given a new flow *j*, what is the maximum value of its continuous period (o_j), such that the new flow will be subset of the schedulable set (may result in preemption of a flow existing in the current schedulable set).

If a node has a set of flows *Γ* passing through it, we define a schedulable set *ζ* $(\zeta \subseteq \Gamma)$ where *ζ* is the set of flows which are schedulable at that particular node. Let the maximum continuous period in the set *ζ* be o_j of some flow *j*. The schedulable set ζ also satisfies the necessary condition provided above. Now the maximum value of continuous period for a new flow, say k to be schedulable is to be lesser than o_j , and the arrival rate is lesser than or equal to j´s arrival rate. That is, a new flow k with continuous period ok will be schedulable, iff $o_k < o_j$ and C_k $T_k \leq C_j$ T_j . This is because; the schedulable set is built considering two conditions - residual lifetime and the necessary condition as equation 3.

If the continuous period of the new flow *(k)* is lesser than the continuous period of a flow *(j),* where flow j is both a member of the existing schedulable set and has a maximum continuous period in the set, then the new flow *(k)* will be added into the schedulable set at the expense of this existing flow *(j,* which had maximum continuous period will be preempted). In addition, the second condition ($C_k T_k \leq C_j T_j$) is important to make sure that the new schedulable set does not violate the condition given by the equation 3. Therefore, for a flow to become eligible as a member of the existing schedulable set is that its continuous period be lesser than the maximum continuous period in the existing schedulable set.

The solution to the second case leads to the notion of a flow *i* being "schedulable" (S) at node l. This notion provides an important parameter in our analysis, as it is used in two ways: (1) An end-to-end measure of this value during the path set-up helps the source to decide on initiating the traffic (2) Intermediate nodes make their scheduling decision based on these values, which can be updated by the downlink neighbors whenever value changes. We know that if a flow is schedulable at all the intermediate nodes, then it is schedulable over the path. The idea is analogous to the series of traffic lights. It is useful to turn the first light green when all the remaining lights will turn green within some acceptable duration. This technique helps in increasing the merit of a scheduler, as priorities are given to packets which will be "completely served". The notion of schedulability takes on only binary values (TRUE/FALSE). When we use this parameter in the algorithm, the mechanism just makes the decision for given values and existing conditions. This decision process is used to build the schedulable-list message, as described below, in the following example.

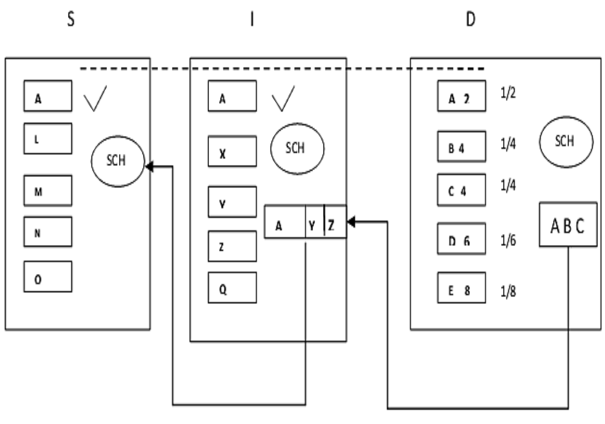

Figure 1: Example for Scheduling Process

Let us consider the example in which three nodes S, I, and D as shown in Figure 1. We will focus on a single flow 'a' starting at node 'S', with intermediate node 'I' and terminating at node 'D'. Let {a, b, c, d} be the flows at 'D'. Let {2, 4, 4, 6} and {1/2, 1/4, 1/4, 1/6} be their continuous periods and rates (C T), respectively. Node 'D' chooses flows {a,b,c} as schedulable following the condition given by equation 3, and creates a schedulability list message, which is transmitted to the upstream neighboring nodes. When 'I' receives this message, marks flow 'a' as schedulable at downstream (sets the schedulability value to TRUE), and builds its own schedulable-list (let it be {a, y, z}) and transmits it to its upstream neighbors. In this manner, the schedulable-list message flows upstream until it reaches source node 'S', which upon receiving will mark flow 'a' as schedulable. If either the destination node or any of the intermediate nodes does not include flow 'a' in their schedulable list message, then the source node will not set flow 'a' as schedulable.

C. Algorithm

```
while(a set of high-priority real-time queues)
{
    if (q.schedulability = TRUE)
        {
          select the set of queues, such that for every
         queue q,
        }
else
{
   select all the queues.
}
  if (value QS(q) RLT(q) is the maximum and not
  yet received eligible-service)
{
  select queue q
}
  if (tie)
{
  select flow that has received least throughput
 }
```

D. Contention and Queue Length Aware Routing

Initially the node *"i"* stays in IDLE state. When packet arrives to node *i*, it will enter Packet-Arrival state. In this state, if node i senses medium busy in SIFS period, it identifies the channel is busy and enters the Back-off state right away, and then change the value of contention window. The station calculates a random value called back-off time. The random number resulting from the binary exponential back-off algorithm is uniformly distributed in a range, called the contention window, the size of this doubles each time the attempt to transmit is deferred, until a maximum size is reached for the range. If the medium is idle for more than Distributed Inter Frame Space

(DIFS) then it will enter the Attempt state and delay a random back-off time interval before the transmission. The back-off timer is periodically decremented by one for every time slot the medium remains idle after the channel has been detected idle for a period greater than DIFS. As soon as the back-off timer expires, the station can access the medium. In the Attempt state the node i will first issue the RTS control packet and then waits for the CTS packet to make sure the transmission is successful. If no CTS is detected within a slot-time, node i will return Back-off state immediately and double the value of CW else it will transmit the data and wait for ACK. If no ACK is detected within a slot-time, node i will return to Back-off state immediately. Once the ACK is detected then it means a transmission is successfully transmitted, CW is reduced to its minimum value for next transmission. The CW indicates that the medium is busy and can be considered as a useful metric for contention and traffic situation around this node. This can replicate the contention of the channel roundly and effectively. To mitigate the effect of traffic bursts, CW is used to calculate the traffic load over a long period. The calculation of the CW of a node is performed every T seconds. The CW is calculated by the below equation

$$CW = a * CW_{old} + CW_{sample} \quad (5)$$

We set α to 0.3 to better reflect the current condition of a node, which grants a higher priority to the current sample. The number of packets in the queue is a metric reflecting the traffic load of this very mobile node. A mobile node with more traffic flows passing through it usually has more packets in its interface queue. Average queue size can indicate this node's traffic load in a long term. The calculation of the average queue size is updated every T seconds according to the following formula,

$$qlength = \beta * qlength_{old} + (1-\beta) * qlength_{sample} \quad (6)$$

Where $qlength$ denotes the average queue length and $qlength_{sample}$ denotes the current queue length, β is constant. It is noteworthy that α and β can be any number selected from the range [0, 1]. Based on the CW and queue length, the local load of node i can be calculated as follow,

$$L_i = \gamma * \frac{CW}{CW_{\max}} + (1-\gamma) * \frac{qlength}{qlength_{\max}} \quad (7)$$

The selection of constant γ is to balance the effects of the two factors. The small qlength means the low load, the small *CW* reflects the benign channel and these cause lower L_i. Therefore, the higher value of the Li means the larger load.

1. Route discovery and maintenance: This routing scheme is based on Dynamic Source Routing (DSR) protocol with the following modifications:

(1) During the route discovery process, the intermediate nodes are not allowed to send back route replies even if they have routes to the destination in their caches. The purpose is to get the up-to-date load information along the whole route for each route discovery.

(2) Suppose if there are multiple routes available, route selection is based on the comprehensive route load value defined in the previous section. When a source node has packet to send and there is no available route in its route cache, a route request packet is initiated and flooded through the network. Each node receiving this request will process and forward it until it reaches the final destination. When the intermediate node processes the request, it adds its load value to the request message. Once the request packet reaches the destination, the destination generates a route reply packet and sends it back to the source node. And the path is chosen with smaller load. When some link on an active route is broken, the source node is notified by a route error packet. The source updates its route cache by removing any route using this broken link, and then initiates another route discovery process to find a new least load route if necessary.

E. Advantages

- The fairness mechanism is added in the MAC protocol to the interference reduction technique.
- This technique will increase the throughput further and achieve fair channel utilization.
- The proposed technique consists of estimating channel condition and queue level based on which the channel is fairly allocated to all the nodes.

IV. SIMULATION RESULTS

A. Simulation Parameters

We evaluate our Channel Aware MAC protocol (CAMAC) through NS-2 [8]. We use a bounded region of 1000 x 1000 sqm, in which we place nodes using a uniform distribution. The number of nodes is 100. We assign the power levels of the nodes such that the transmission range of the nodes is 250 meters. In our simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. We use the distributed coordination function (DCF) of IEEE 802.11 for wireless LANs as the MAC layer protocol. The simulated traffic is Constant Bit Rate (CBR). The simulation topology is given in Figure 2.

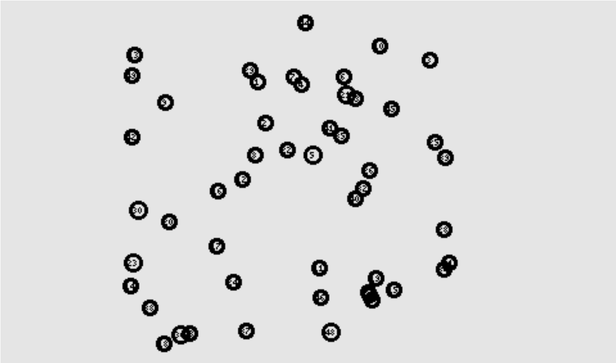

Figure 2: Simulation Topology

The following table summarizes the simulation parameters used

Table 1. Simulation Parameters

No. of Nodes	100.
Area Size	1000 X 1000
Mac	802.11
Simulation Time	50 sec
Traffic Source	CBR
Packet Size	500
Transmit Power	0.660 w
Receiving Power	0.395 w
Idle Power	0.035 w
Initial Energy	10.3 J
Transmission Range	250m
Routing Protocol	AODV
Flows	2, 4, 6 and 8.
Error Rate	0.01 to 0.05

B. Performance Metrics

We compare the performance of our proposed CAMAC with Channel-aware Scheduling mechanism for MANETS (CaSMA) technique [7]. We evaluate mainly the performance according to the following metrics:

Received Bandwidth: It is the number of bits transmitted to the destination through the channel.

Packet Lost: It is the number of packets dropped during the transmission.

Delay: It is the amount of time taken by the packets to reach the destination.

C. Results

1. Based on Flows

In our experiment we vary the number of CBR traffic flows as 2, 4, 6 and 8 with 50 nodes.

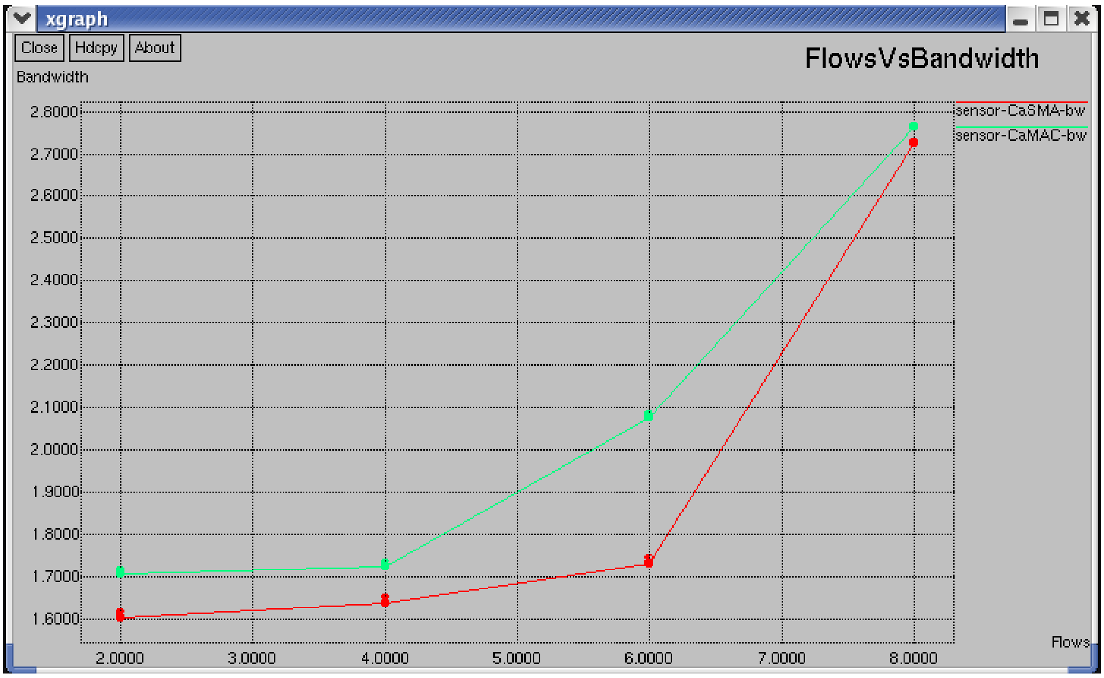

Figure 3: Flows vs Received Bandwidth

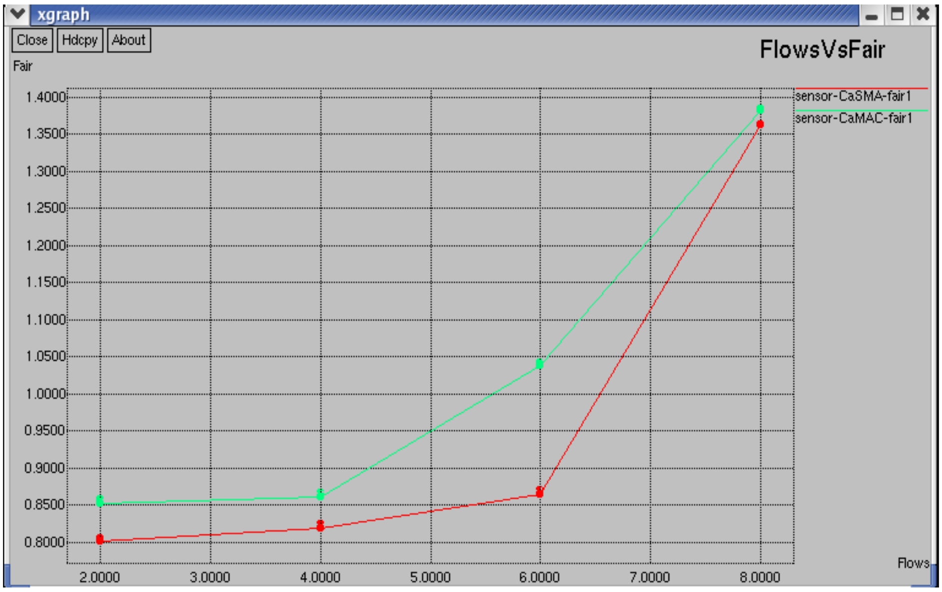

Figure 4: Flows vs Fairness

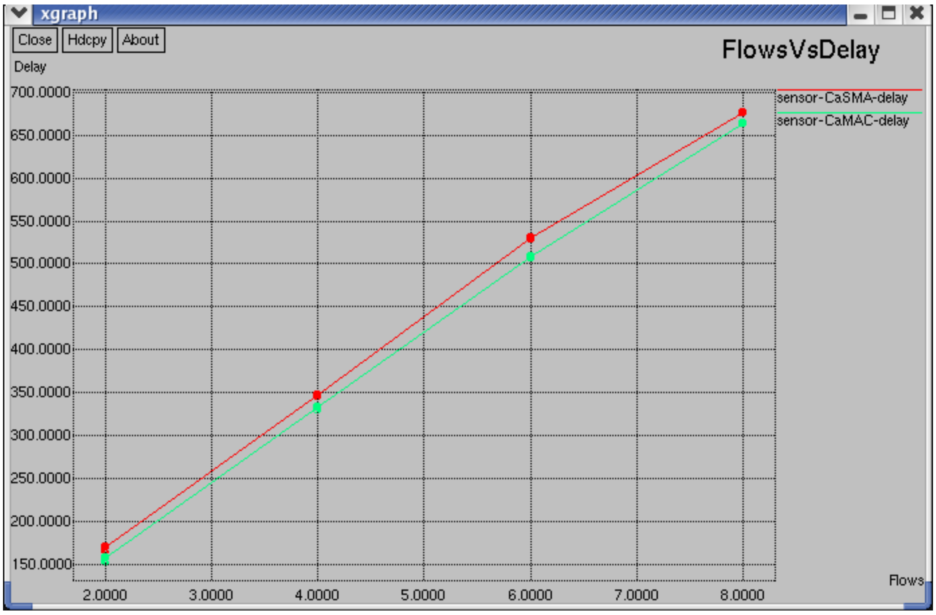

Figure 5: Flows vs Delay

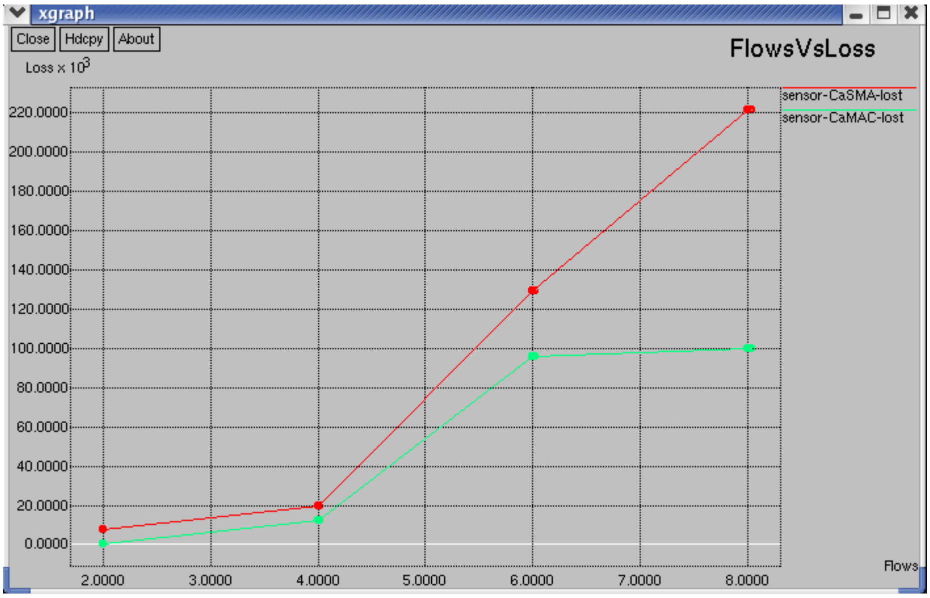

Figure 6: Flows vs packet Lost

From figure 3, we can see that the received bandwidth of our proposed CAMAC is higher than the existing CASMA technique.

From figure 4, we can see that the fairness of our proposed CAMAC is higher than the existing CASMA technique.

From figure 5, we can see that the delay of our proposed CAMAC is less than the existing CASMA technique.

Form figure 6, we can see that the Packet lost ratio of our proposed CAMAC is less than the existing CASMA technique.

2. Based on Error Rate

In our second experiment we vary the error rate as 0.01, 0.02, 0.03, 0.04 and 0.05 for 100 nodes.

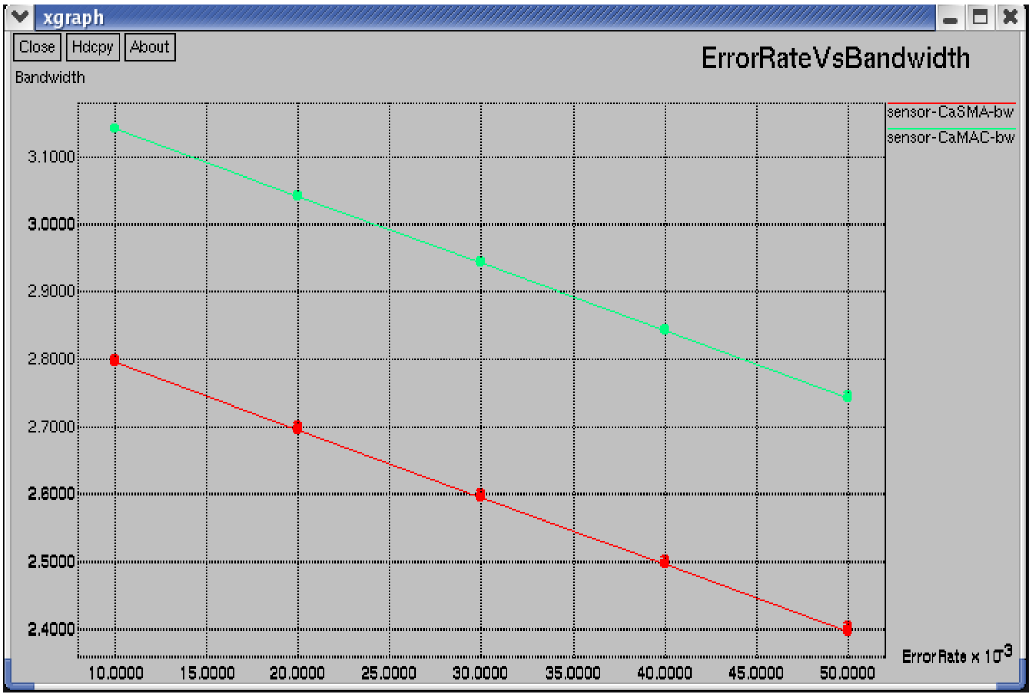

Figure 7: Error Rate vs Received Bandwidth

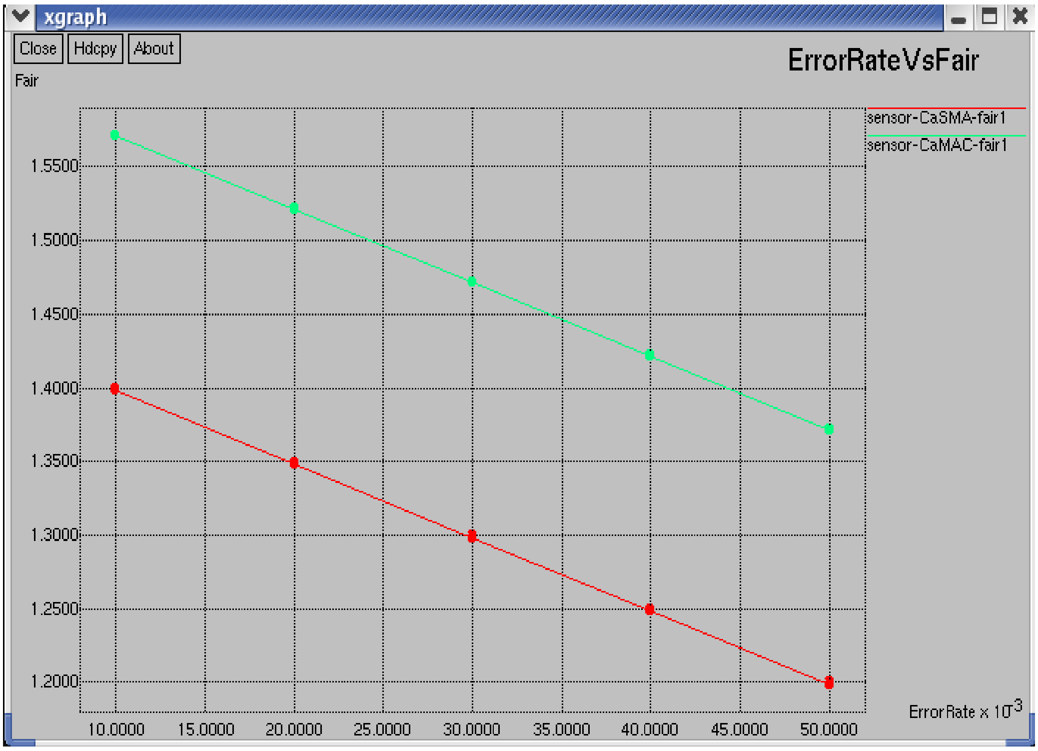

Figure 8: Error Rate vs Fairness

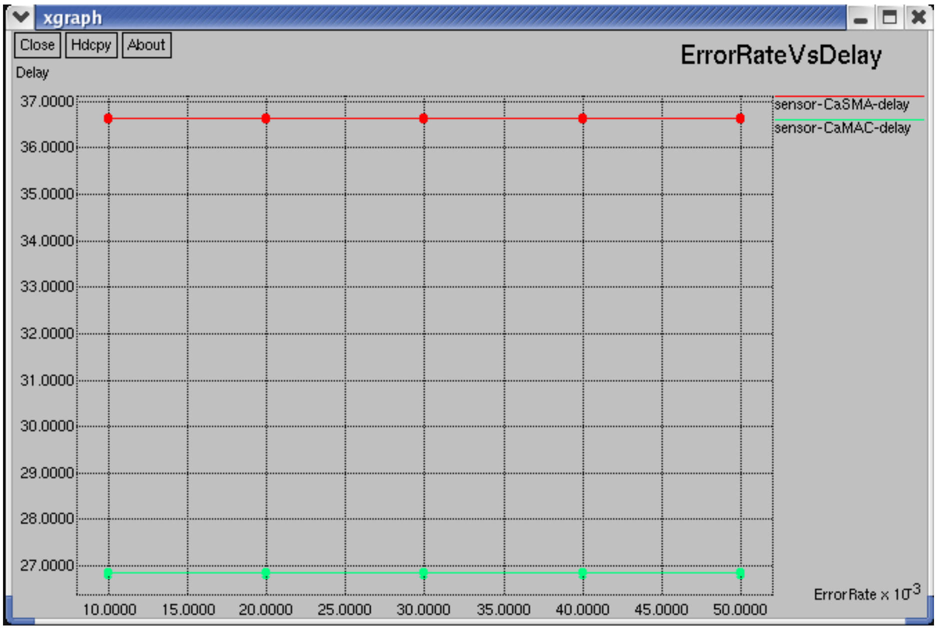

Figure 9: Error Rate vs Delay

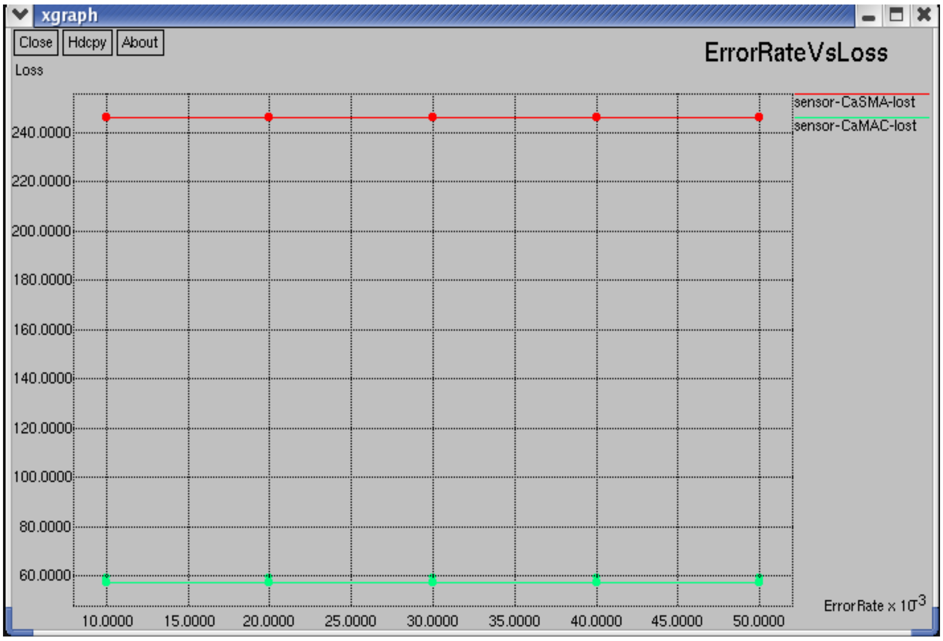

Figure 10: Error Rate vs Packet Lost

From figure 7, we can see that the received bandwidth of our proposed CAMAC is higher than the existing CASMA technique.

From figure 8, we can see that the fairness of our proposed CAMAC is higher than the existing CASMA technique.

From figure 9, we can see that the delay of our proposed CAMAC is less than the existing CASMA technique.

Form figure 10, we can see that the Packet lost ratio of our proposed CAMAC is less than the existing CASMA technique.

V. CONCLUSION

In this paper we are extending the previous work by improving the MAC protocol to maximize the Throughput and Fairness. In this work we are estimating the channel condition and Contention for a channel aware packet scheduling and the queue length is also calculated for the routing protocol which is aware of the queue length. The channel is scheduled based on the channel condition and the routing is carried out by considering the queue length. This queue length will provide a measurement of traffic load at the mobile node itself. Depending upon this load the node with the lesser load will be selected for the routing; this will effectively balance the load and improve the throughput of the ad hoc network. The advantage of this work is that, the fairness mechanism is added in the MAC protocol to the interference reduction technique. Future work concentrates on developing a rate adaptation algorithm based on the channel condition.

VI. REFERENCES

[1] Fan Wang, Ossama Younis and Marwan Krunz, “Throughput-oriented MAC for mobile ad hoc networks: A game-theoretic approach”, 2008 Published by Elsevier B.V. doi: 10.1016/j.adhoc.2007.12.002

[2] Mariam Kaynia, Nihar Jindal and Geir E. Øien, “Improving the Performance of Wireless Ad Hoc Networks Through MAC Layer Design”, IEEE Transactions on Wireless Communications, vol. 10, no. 1, January 2011, doi: 10.1109/TWC.2010.110310.100316

[3] Hongqiang Zhai, Jianfeng Wang, Xiang Chen, and Yuguang Fang, “Medium Access Control in Mobile Ad Hoc Networks: Challenges and Solutions”, 2006. doi: 10.1002/wcm.376

[4] P.Priakanth, Dr.P.Thangaraj, “Energy and Channel Aware MAC Protocol to Achieve Fairness in Multi-Hop Mobile Adhoc Networks”, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.4, April 2008.

[5] Zhifei Li, Sukumar Nandi, and Anil K. Gupta, “Study of IEEE 802.11 Fairness and its Interaction with Routing Mechanism”, In the Fifth IFIP International Conference on Mobile and Wireless Communications Networks (MWCN), Singapore, Oct. 27-29, 2003.

[6] Xuemei Gao, Xinming Zhang, Dong Shi, Fengfu Zou, Wenbo Zhu, “Contention and Queue-aware Routing Protocol for Mobile Ad h oc Networks”, IEEE International Conference on Wireless Communications, Metwork and Mobile Computing, 2007. doi: 10.1109/WICOM.2007.410

[7] Sridhar K N, Mun Choon Chan, “Channel-aware Packet Scheduling for MANETs”, IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, 2008. doi: 10.1109/WOWMOM.2008.4594864

[8] Network Simulator: http: ///www.isi.edu/nsnam/ns

How to cite

G.Suseendran, E.Chandrasekaran, " Channel Aware Mac Protocol for Maximizing Throughput and Fairness ". *International Journal of Research in Computer Science*, 3 (5): pp. 1-11, September 2013. doi: 10.7815/ijorcs. 35.2013.069