International Journal on New Computer Architectures and Their Applications (IJNCAA) 1(1): 149-161
The Society of Digital Information and Wireless Communications, 2011 (ISSN: 2220-9085)

Improving Transport Protocol Performance in MANET by exploiting the Backoff Algorithm of MAC Protocol

Sofiane Hamrioui [1], Mustapha Lalam [2]
[1] Department of Computer Science, University of Sciences and Technologies Houari Boumedienne, Algiers, Algeria.
[2] Department of Computer Science, University of Mouloud Mammeri, Tizi Ouzou, Algeria.E-mail
s.hamrioui@gmail.com, lalamustapha@yahoo.fr

ABSTRACT

In this paper, we propose an improvement to the transport protocol performance in Mobile Ad Hoc Network (MANET) by exploiting the backoff algorithm of Medium Access Control (MAC) protocol. We are especially interested in the TCP (Transmission Control Protocol) performance parameters like the throughput and end-to-end delay. This improvement is IB-MAC (Improvement of Backoff algorithm of MAC protocol) which proposes a new backoff algorithm based on a dynamic adaptation of its maximal limit according to the number of nodes and their mobility. The evaluation of our IB-MAC solution and the study of its incidences on TCP performance are done with AODV as like routing protocol, TCP New Reno as like transport protocol and varied network conditions such as load and mobility. The results are satisfactory and showed that our algorithm can outperform not only MAC standard, but also similar techniques that have been proposed in the literature like MAC-LDA and MAC-WCCP.

KEYWORDS

MANET, Transport Protocol, MAC Protocol, Backoff Algorithm, Performance Improvement.

1 INTRODUCTION

Mobile Ad Hoc Networks (MANET) [1] are complex distributed systems that consist of wireless mobile nodes. In such network, the MAC protocol [2], [3], [4] must provide access to the wireless medium efficiently and reduce interference. Important examples of these protocols include CSMA with collision avoidance that uses a random back-off even after the carrier is sensed idle [5]; and a virtual carrier sensing mechanism using request-to-send/clear-to-send (RTS/CTS) control packets [6]. Both techniques are used in IEEE 802.11 MAC protocol [5] which is a current standard for wireless networks.

Transmission Control Protocol (TCP) [7], [8] is the transport protocol used in the most IP networks [9] and recently in ad hoc networks like MANET [10]. It is important to understand the TCP behavior when coupled with IEEE 802.11 MAC protocol in an ad hoc network. When the interactions between the MAC and TCP protocols are not taken into account, this may degrades MANET performance notably TCP performance parameters (throughput and the end-to-end delay) [11], [12], [13].

In [15], we presented a study of interactions between the MAC and TCP protocols. We have shown that the TCP parameters performance (notably throughput) degrades while the nodes number increase in a MANET using IEEE 802.11 MAC as access control protocol. In [16], we have proposed solutions to the problem posed in [15], but we have just limited to a chain topology and also to the influence of the nodes number on the TCP performance. In [17] we studied the validity of the proposed solutions in [16] with several routing and transport protocols and with different static topologies. We found that the proposed solutions not only are not influenced by the change of the topology of the network but also by the routing and the transport protocols used.

Our contribution in this paper is the following of those done in [15], [16] and [17]. Another parameter which is the mobility of nodes is taken into account in addition to the number nodes. Also in this present work we compared our solution with other solutions proposed in the same context. After a short presentation of MAC and TCP protocols, we will present our IB-MAC (Improvement of the Backoff algorithm of MAC protocol) and study its incidences on TCP performance parameters (throughput and end-to-end delay). IB-MAC proposes a dynamic adaptation of the maximal limit of the MAC backoff algorithm. This adaptation is as function of the nodes number in the network and their mobility.

Our paper is structured in five sections: the section two gives a short presentation of MAC and TCP protocols, in section three we present the IB-MAC improvement to better TCP performance in MANET, in section four we study the incidences of these improvements on TCP performance parameters and we finish with section five which consists on conclusion and perspectives.

2 INTERACTIONS BETWEEN MAC AND TCP PROTOCOLS

2.1 MAC IEEE 802.11 and TCP protocols in MANET

IEEE 802.11 MAC protocol defines two different access methods; a distributed coordination function (DCF) and polling based point coordination function (PCF). In MANET, the DCF feature is used. The DCF access is basically a carrier sense multiple access with collision avoidance (CSMA/CA) mechanism. In order to avoid collision due to the hidden terminal problem [18], [19] the node first transmits a Request To Send (RTS) control frame. The destination node responds with a Clear To Send (CTS) control frame. Once a successful RTS-CTS frame exchange takes place, the data frame (DATA) is transmitted. The receiving node checks the received data frame, and upon correct receipt, sends an acknowledgement (ACK) frame. Although the introduction of RTS-CTS-DATA-ACK frame format makes the transmission more reliable, there is still the possibility of transmission failure.

It has been shown that TCP does not work well in a wireless network [7], [20]. TCP associates the packet loss to the congestion, and then it starts its congestion control mechanism. Therefore, transmission failures at the MAC layer lead to the congestion control activation by TCP protocol then the number of packets is reduced. Several mechanisms have been proposed to address this problem [21], [22], [23], but most of them focus on the cellular architecture. The problem is more complex in MANET where there is no base station and each node can act as a router [24], [25].

The TCP Performance parameters (like the throughput and the end-to-end delay) have been the subject of several evaluations. It has been shown that these parameters degrade when the interactions between MAC and TCP are not taken into account [7], [18]. In our previous work [15], we confirmed these results by studying the effect of the MAC layer when the number of nodes increases. The major source of these effects is the problem of hidden and exposed nodes [18], [19]. The most important solution which has been proposed to the hidden node problem is the use of RTS and CTS frames [26], [27]. Although the use of RTS/CTS frames is considered as a solution to the hidden node problem, it was shown in [15], [18], [28] that it also leads to further degrade the TCP flow by creating more collisions and introduce an additional overhead. Then these two constraints decrease the TCP performance.

2.2 Related work

In [29], [30], [31], [32], [33], many analyses of TCP protocol performance are done and several solutions on how to improve this performance are proposed. In this paper we present the most important of these solutions.

Yuki and al. [34] have proposed a technique that combines data and ACK packets, and have shown through simulation that this technique can make radio channel utilization more efficient.

Altman and Jimenez [35], proposed an improvement for TCP performance by delaying 3-4 ACK packets.

Kherani and Shorey [36], suggest significant improvement in TCP performance as the delayed acknowledgement parameter d increases to the TCP window size W.

Allman [37], conducted an extensive evaluation on Delayed Acknowledgment (DA) strategies, and they presented a variety of mechanisms to improve TCP performance in presence of side-effect of delayed ACKs.

Chandran [38] proposed TCP-feedback, with this solution, when an intermediate node detects the disruption of a route; it explicitly sends a Route Failure Notification (RFN) to the TCP sender.

Holland and Vaidya [39] proposed a similar approach based on ELFN (Explicit Link Failure Notification), when the TCP sender is informed of a link failure, it freezes its state. Liu and Singh [40] proposed the ATCP protocol; it tries to deal with the problem of high Bit Error Rate (BER) and route failures.

Fu et al. [41] investigated TCP improvements by using multiple end-to-end metrics instead of a single metric. They claim that a single metric may not provide accurate results in all conditions.

Biaz and Vaidya [42] evaluated three schemes for predicting the reason for packet losses inside wireless networks. They applied simple statistics on observed Round-trip Time (RTT) and/or observed throughput of a TCP connection for deciding whether to increase or decrease the TCP congestion window.

Liu et al. [43] proposed an end-to-end technique for distinguishing between packet losses due to congestion from packet loss by a wireless medium. They designed a Hidden Markov Model (HMM) algorithm to perform the mentioned discrimination taking RTT measurements over the end-to-end channel.

Kim and al. [44], [45] proposed the TCP-BuS (TCP Buffering capability and Sequence information), like previous proposals, uses the network feedback in order to detect route failure events and to take convenient reaction to this event.

Oliveira and Braun [46] propose a dynamic adaptive strategy for minimizing the number of ACK packets in transit and mitigating spurious retransmissions.

Hamadani and Rakocevic [47] propose a cross layer algorithm called TCP Contention Control that it adjusts the amount of outstanding data in the network based on the level of contention experienced by packets as well as the throughput achieved by connections.

Zhai et *al*. [48] propose a systematic solution named Wireless Congestion Control Protocol (WCCP) which uses channel busyness ratio to allocate the shared resource and accordingly adjusts the sender's rate so that the channel capacity can be fully utilized and fairness is improved.

Lohier et *al*. [49] proposes to adapt one of the MAC parameters, the *Retry Limit* (*RL*), to reduce the drop in performance due to the inappropriate triggering of TCP congestion control mechanisms. Starting from this, a MAC-layer LDA (Loss Differentiation Algorithm) is proposed.

The approaches just presented suggest improvements to TCP performance based on MAC and TCP protocols. In our work, we propose to study the interactions between these two protocols and improve them. In what follows, we examine the interactions between MAC and TCP before proceeding to the presentation of our solutions.

3 IB-MAC (IMPROVEMENT OF THE BACKOFF ALGORITHM OF MAC PROTOCOL)

The MAC protocol is based on the backoff algorithm that allows it to determine which will access to the wireless medium in order to avoid collisions. The time backoff is calculated as follows:

$$\text{BackoffTime} = \text{BackoffCounter} * \text{aSlotTime} \quad (1)$$

In (1), aSlotTime is a time constant and BackoffCounter is an integer from uniform distribution in the interval [0, CW] and CW is the contention window who's minimum and maximum limits are (CW_{min}, CW_{max}) and are defined in advance.

The CW value is increased in the case of non availability of the channel using the following formula:

$$\begin{gathered} m \leftarrow m + 1 \\ CW(m) = (CW_{min} + 1)*2^m - 1 \\ CW_{min} <= CW(m) <= CW_{max} \end{gathered} \quad (2)$$

m: the number of retransmissions.

The first parameter used by our IB-MAC solution is the number of nodes in the network As we have seen through the simulations presented in the [15] and [16], when the number of nodes in the network increases, the performance of TCP deteriorates. The cause of this degradation is the frequent occurrence of collisions between nodes. These collisions become more frequent with a small backoff interval because the probability to have two or more nodes choose the same value in a small interval is greater than the probability that these nodes choose the same value in a larger interval.

Note by I this interval, S_I its size, and Pr(i,x) is the probability that the node i chooses the x value in the I interval. The problem then is how to ensure that for any two nodes i and j in the network with i != j, we will have:

$$| Pr(i,x) - Pr(j,x) | = y \quad (3)$$
$$y \,!= 0$$

For an important number of nodes in the network, and for a high probability that the formula **(3)** will be verified, we must have a larger S_I. To do this we wanted to make the size of S_I adaptable to the number of nodes in the network, then we intervene on one of the limits of this interval, we then propose the limit CW_{max}.

Note by n the number of nodes in the network.

Then the first part of the expression of CW_{max} will be:

$$F(n) = \log(n) \quad (4)$$

Log () is used here because we found in [15] and [16] that the effects of the large values of the nodes number on the TCP performance are almost the same.

Our IB-MAC also takes into account the mobility of nodes because it participates in the degradation of TCP performance. In fact, node mobility often leads to the breakdown of connectivity between nodes, resulting in loss of TCP packets and then the degradation of the TCP performance. At the MAC protocol, when the packets losses are detected, they are associated to the collisions problem, which is not the case here. Then, more the mobility increases, more the backoff interval increases, something that should not happen because these packets are lose due to the rupture of the connectivity and no to the collisions. Therefore, we will try to find a compromise between the effect of mobility and the size of the backoff interval.

Mobility is generally characterized by its speed and angle of movement, two factors that determine the degree of the impact of mobility on packets loss. Consider a node i, in communication with another node j, then we note by:

α: the angle between the line (i, j) and the movement direction of node i,

W: the speed of mobile node i.

To consider the impact of mobility on the loss of packets is necessary to study the effects of mobility parameters (W and α). For the effect of speed W, as in the case of number of nodes, we use a logarithmic function because for large values of speed mobility the results converge. So this is expressed as follows:

$$H(W) = \begin{cases} 1 & \text{if } W=0 \text{ (Without mobility)} \\ \mathrm{Log}(W) & \text{else} \end{cases} \quad (5)$$

Also, the direction of the node movement determines the degree of the influence of mobility on packets loss, it is given by *G(W, α)*:

$$G(W, \alpha) = \begin{cases} 1 & \text{if } -\Pi/4 <= \alpha <= \Pi/4 \\ \sqrt{W} & \text{else} \end{cases} \quad (6)$$

Note that *G(W, α)* = 1 when *W* = 0 (without mobility).

We know that when the W, the packets loss increase too, it increases more when the node is moving in the opposite direction of communication.

This increasing of packets losses has a negative impact on backoff interval because they can be associated to the collisions, but is not the case here (as explained above). To make this impact positive, we must use the inverse, as like follows:

$$M(W, \alpha) = 1 / (G(W, \alpha) * H(W)) \qquad (7)$$

The equation (7) decreases with the increasing of $G(W, \alpha)$ and $H(W)$ (when W increases), it decreases more when the node is moving in the opposite direction of communication.

$M(W,\alpha)$ expresses the impact of the mobility on the packets losses, it is the probability that the cause of these losses is the mobility.

With (7), we can guarantee that when the mobility of nodes is significant, the adaptation of the backoff algorithm is not important because this mobility is more probable to be the cause of many losses packets. But with weak mobility the same equation makes it possible to get a significant adaptation to the backoff algorithm because in this case the collisions between frames are more probable to be the cause of the losses packets.

We give now the new expression of CW_{max} as follows:

$$CW_{max}(n, W, \alpha) = CW_{max0} + (F(n) * M(W,\alpha)) \qquad (8)$$

From (4), (5) and (7), we will have:

$$CW_{max}(n, W, \alpha) = CW_{max0} + \mathrm{Log}(n) * (1 / ((G(W, \alpha) * H(W))) \qquad (9)$$

CW_{max0}: initial value of CW_{max} defined by the MAC protocol (with the 802.11 version, it is equal to 1024);

Our approach is fully distributed within the MANET; each node may determine alone the values of n, W and α so it can then calculate the value of CW_{max} according to the formula given in (9). The value of n is variable; it is updated always when there is a new arrived node to the network or a leaved node from the network. So, our solution also contains an agent let updating the value of n as follows:

```
Begin
……
Variable N ← 0;
……
Node_i ← NEW (Node_Class);
Add (Node_i)
N ← N+1
……
Free (Node_j)
N ← N-1;
……
End;
```

After having made the values of CW_{max} adaptive to the number of nodes used and their mobility, the IB-MAC (improved version of that given by the formula (2)) becomes:

$$\begin{gathered} m \leftarrow m + 1 \\ CW(m) = (CW_{min}(n) + 1)*2^{m} - 1 \\ CW_{min} <= CW(m) <= CW_{max}(n,W, \alpha) \\ CW_{max}(n, W, \alpha) = CW_{max0} + (1 / (G(W, \alpha) * H(W)))*\log(n) \end{gathered} \qquad (10)$$

m: the number of retransmissions, n**:** the number of the nodes used.
α: the angle between the line formed by the mobile node and its corresponding node and the movement direction of this mobile node.
W: the speed of mobile node.
$H(W)$: see the expression (5).
$G(W, \alpha)$: see the expression (6),
CW_{max0}: initial value of CW_{max}.

4 INCIDENCES OF IB-MAC ON TCP PERFORMANCE

4.1 Simulation Environment

The evaluation is performed through the simulation environment NS-2 (version 2.34) [50], [51]. MAC level use the 802.11b with DCF (Distributed Coordination Function) and the values of its basic parameters are listed in the in **Table 1**.

Table 1: Parameters for IEEE 802.11 MAC

Parameters	Values
Preamble length (bit)	144
RTS length (bit)	160
CTS/ACK length (bit)	112
MAC header (bit)	224
IP header (bit)	160
SIFS (μs)	10
DIFS (μs)	50
Slot time (μs)	20
Contention window	31
Retry limit	7

For our simulations, the effective transmission range is of 250 meters and an interference range of 550 meters. Each node has a queue buffer link layer of 50 packets managed with a mode drop-tail [52]. The scheduling packet transmissions technique is the First in First out (FIFO) type. The propagation model used is two-ray ground model [53].

Our simulations are done with reactive routing protocol AODV [54]. We used TCP NewReno [55] which is a reactive variant, derived and widely deployed, and whose performances were evaluated under conditions similar to those conducted here. This choice is because the previous works [17] have shown almost similar results for different routing protocols and TCP versions used.

The values, such as the duration of simulation, the speed of nodes, and the number of connections have been established in order to obtain interpretable results compared to those published in the literature. The simulations are performed for 1000 seconds, this choice in order to analyze the full spectrum of TCP throughput.

We considered two cases: without and with mobility. In the first case, chain topology is studied in which always the node 1 send for the node *n* (where n is the length of the chain). We just limited to the chain topology because our in our previous work [17] we have studied different topologies and we have found that it influence the communication environment in the same way. The distance between two neighbouring nodes is 200 meters and each node can communicate only with its nearest neighbour. The interference range of a node is about two times higher than its transmission range.

In the case with mobility, we study a random topology with two cases: weak and strong mobility. In both cases, it is only for the node 1 that to send for the node n. The mobility model uses the random waypoint model [56], we justify our choice by the fact that the network is not designed for mobility and that this particular model is widely used in the literature. In this model the node mobility is typically random and all nodes are uniformly distributed in space simulation. The nodes move in 2200m*600m area, each one starts its movement from a random location to a random destination. Once the destination is reached, another random destination is targeted after a pause time.

4.2 Parameters Evaluation

We have simulated several scenarios with different numbers of nodes n, topologies, routing and TCP protocols, and mobility. We are interested in each scenario into two parameters. The first is the throughput which is given by the ratio of the received data on all data sent. The second parameter is the end-to-end delay which is given by time for receipt of data - the data transmission time / number of data packets received.

4.3 Simulation and Results

In these scenarios, we compare our solution (IB-MAC) with MAC standard and two other solutions proposed in the literature. As like our IB-MAC improvement, these solutions use the MAC layer to improve TCP performance in the MANET. The first solution is Wireless Congestion Control Protocol (WCCP) [48] and the second one is MAC-layer LDA (Loss Differentiation Algorithm) [49]. The principle of each solution is given in the section of related work. Two cases are also considered, with and without mobility. In the case without mobility, we just use the chain topology since our in our previous work [17] we have studied different topologies and we have found that it influence the communication environment in the same way. In both cases (with and without mobility), we used TCP New Reno version and AODV routing protocol. This choice is because the previous works [17] have shown almost similar results for different routing protocols and TCP versions used.

Scenario 1: Without mobility (Chain Topology), TCP New Reno, AODV.

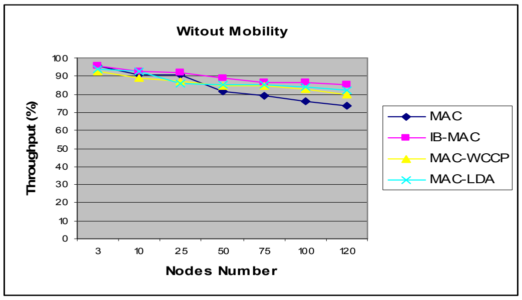

Figure 1. Throughput variation without Mobility (chain topology).

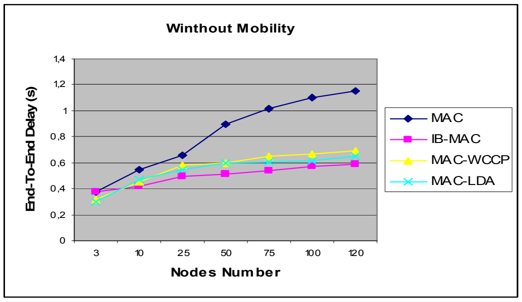

Figure 2. End-To-End Delay variation without mobility (chain topology).

We see, through **Figure 1**, with MAC protocol, more the nodes number participating in the network increases, more the throughput decreases. This degradation at a given time (from n=100 nodes) begins to take stability for the three protocols. This degradation is due to TCP packet loss occurred, and that becomes more important with increasing size of the network. With the analysis of the trace files for these graphs, we found that RTS and CTS frames, handled at the MAC level, are sensitive to the network size, more the nodes number increases, the lose of these two frames increases too. It has been shown previously that such frames losses in such conditions of simulations are mainly due to the consequences of hidden and exposed nodes, a result that has already been achieved in our past work [15] [16].

But when the IB-MAC is used as MAC protocol we see that the throughput is better. There is an important improvement of this parameter, even if there is a slight decrease when the number of nodes increases but this decrease is much smaller compared to the first case when the MAC protocol is used.

This improvement is due to the use of the adaptive nature of our solution IB-MAC to the nodes number in the network.

Figure 2 shows the evolution of the second parameter studied which is the end-to-end delay when the nodes number increases. With MAC protocol, we find that this parameter significantly increases with the increase of the used nodes number. The increase of the end-to-end delay is essentially due to the detection of frequent loss of TCP packets in the network more the number of nodes increases. These losses will be the cause for the frequent start of the congestion avoidance mechanism by the TCP protocol, so that will result in delaying the transmission of TCP packets and the increase in delay. This increase in delay begins to stabilize from n = 110 nodes and that below t = 1.2 s approximately.

When the IB-MAC is used as MAC protocol we see that the end-to-end delay is better. There is an important improvement of this parameter, even if there is a slight increase when the number of nodes increases but this decrease is smaller compared to the first case when the MAC protocol is used.

Figure 1 and **Figure 2** have been shown that our IB-MAC outperform not only MAC standard, but also similar techniques that have been proposed in the literature. The results of the variation of the throughput and the end-to-end delay parameters are better than those of MAC-LDA, MAC-WCCP and MAC standard.

The improvement of the throughput and end-to-end delay parameters is due to the dynamic nature of our new IB-MAC algorithm which makes the size of the backoff interval adjustable to the nodes number in the network. This adjustment reduces the probability of collisions between nodes, thus the number of lost packages is reduced while the throughput and delay are improved.

Scenario 2: Weak mobility (W = 5 m/s), TCP New Reno, AODV.

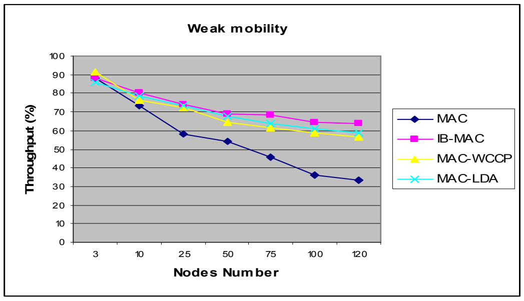

Figure 3. Throughput variation with weak mobility (speed W=5 m/s).

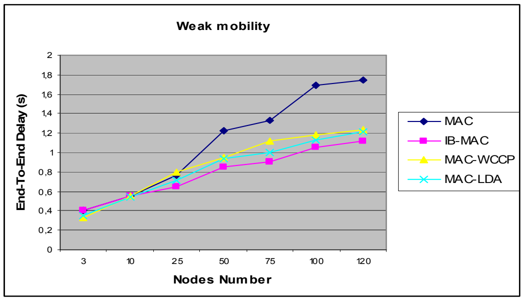

Figure 4. End-To-End Delay variation with weak mobility (speed W = 5 m/s).

For the weak mobility (**Figure 3** and **Figure 4**), when the MAC protocol is used, we found an important degradation of the throughput and end-to-end delay parameters in comparison to the first case (without mobility). To explain this degradation, we analyzed the obtained trace files and we found:

i) The increase of RTS/CTS frames losses with the increase of nodes number in the network (same to the first case without mobility);

ii) There are TCP packets losses even if there are successful RTS/CTS frames transmissions. In this case, these losses are caused by the unavailability route due the nodes mobility (the used route is outdated, denoted by "NRTE" in the trace file).

We deduce through i) and ii) that the mobility of nodes, although it is weak (here speed W = 5 m/s), participates to the degradation of the throughput and end-to-end delay parameters.

With our IB-MAC solution, always with weak mobility, we found an important improvement of the throughput and end-to-end delay parameters in comparison to the first case when the MAC protocol is used. Our IB-MAC algorithm makes the size of the backoff interval adjustable to the nodes number in the network and their mobility. For this reason, even for the case where the nodes are mobiles, the probability of collisions between nodes is reduced, and then throughput and the end-to-end delay parameters are improved.

Figure 3 and **Figure 4** shows also that IB-MAC outperform the others protocols used (MAC-LDA and MAC-WCCP). The results of the variation of the throughput and the end-to-end delay parameters are better than those of the others. Although there is a slight difference in performance but our strategy remains the best to the other three used here.

Scenario 3: Strong mobility (W = 25 m/s), TCP New Reno, AODV.

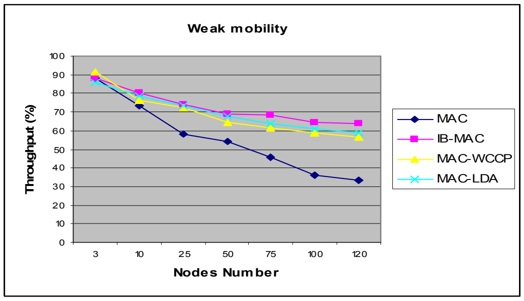

Figure 5. Throughput variation with strong mobility (speed W=25 m/s).

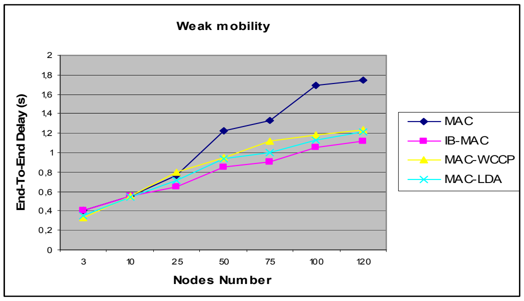

Figure 6. End-To-End Delay variation with strong mobility (speed W = 25 m/s).

For strong mobility (**Figure 5** and **Figure 6**), we see that there is also a degradation of the throughput and end-to-end parameters when the MAC protocol is used, more important than the case with weak mobility because here the breaks connectivity increases then the links stability becomes more important. We have done the same analysis as above to know the reasons of this degradation; we found that the causes of this degradation are also related to those discussed in i) and ii) in weak mobility case.

In fact, when the network has a weak mobility (nodes with low speeds), it presents a rather high stability; then links failure are less frequent than the case of a high mobility. Consequently, the fraction of data loss is smaller when for the case where nodes move at low speeds (strong mobility), and grows with the increase in their mobility.

In this case too (strong mobility), with our solution IB-MAC, we found an important improvement of the throughput and end-to-end delay parameters in comparison to the first case when the MAC protocol is used.

We found also a better improvement of the throughput and end-to-end delay parameters in comparison to the others protocols used (MAC-LDA, MAC-WCCP).

From these results, we can say that even in the case of a random topology where nodes are mobile (a feature specific to MANET networks) the IB-MAC solution improves the performance of TCP.

5 CONCLUSION

Improving TCP performance over 802.11 MAC protocol in multi-hop ad-hoc networks is truly a problem on interaction between two layers. In this paper, we proposed an improvement of TCP protocol performance (throughput and end-to-end delay) in MANET. Our solution is IB-MAC which is a new Backoff algorithm making dynamic the CW_{max} terminal in depending on the number of nodes used in the network and their mobility. This adaptation is to reduce the number of collisions between nodes produced after having learned the same values of the interval Backoff algorithm.

We studied the effects of IB-MAC on TCP performance, we limited our studies on very important parameters in such networks which are throughput and end-to-end delay because they have great effects on the performance of TCP protocol and that of the total network. The results are satisfactory and showed that our algorithm can outperform not only MAC standard, but also similar techniques that have been proposed in the literature like MAC-LDA and MAC-WCCP.

We do not claim that our IB-MAC solution is the optimal backoff algorithm to better TCP performance, but the achieved results are indeed encouraging, justifying further investigation on this direction. As perspectives, we have to test for how many nodes our solutions remains valid. The continuation of our work will consist in looking for a complete cross-layer IB-MAC in order to adapt dynamically and in a coordinated way the MAC and the TCP parameters.

6 REFERENCES

1. Basagni, S., Conti, M., Giordano, S., and Stojmenovic, I.: Mobile Ad hoc Networking. Wiley-IEEE Press. USA, ISBN: 0-471-37313-3. (2004).
2. Karn, P.: MACA - A New Channel Access Method for Packet Radio. Proc. 9th ARRL/CRRL Amateur Radio Computer, Networking Conference. (1990).
3. Bhargavan, V., Demers, A., Shenker, S., Zhang, L.: MACAW, A Media Access Protocol for Wireless LANs. Proc. ACM SIGCOMM. (1994).
4. Parsa, C., Garcia-Luna-Aceves, J.: TULIP - A Link-Level Protocol for Improving TCP over Wireless Links. Proc. IEEE WCNC. (1999).
5. IEEE Std. 802.11. Wireless LAN Media Access Control (MAC) and Physical Layer (PHY) Specifications. (1999).
6. Mjeku, M., Gomes, N. J.: Analysis of the Request to Send/Clear to Send Exchange in WLAN Over Fiber Networks, Journal of lightware technology, issn: 0733-8724, vol. 26, n°13-16, pp. 2531-2539. (2008).
7. Holland, G., Vaidya, N.: Analysis of TCP performance not over mobile ad hoc networks. In Proc. ACM Mobicom. (1999).
8. Hanbali, A., Altman, E., Nain, P.: A Survey of TCP over Ad Hoc Networks. In IEEE Communications Surveys & Tutorials. Vol 7, no 3, pp 22-36. (2005).
9. Kurose, J., Ross, K.: Computer Networking: A top-down approach featuring the Internet. Addison Wesley. (2005).
10. Kawadia, V., Kumar, P.: Experimental investigations into TCP performance over wireless multihop networks. In SIGCOMM

Workshop on experimental approaches to wireless network design and analysis (E-WIND). (2005).

11. Jiang, R., Gupta, V., Ravishankar, C.: Interactions Between TCP and the IEEE 802.11 MAC Protocol. DARPA Information Survivability Conference and Exposition. (2003).
12. Nahm, K., Helmy, A., Kuo, C.-C. J.: On Interactions Between MAC and Transport Layers in 802.11 Ad-hoc Networks. SPIE ITCOM'04. Philadelphia. (2004).
13. Papanastasiou, S., Mackenzie, L., Ould-Khaoua, M., Charissis, V.: On the interaction of TCP and Routing Protocols in MANETs. In Proc. of AICT/ICIW. (2006).
14. Li, J.: Quality of Service (QoS) Provisioning in Multihop Ad Hoc Networks. Doctorate of Philosophy, Computer Science in the Office of Graduate Studies. California, (2006).
15. Hamrioui, S., Bouamra, S., Lalam, M. : Interactions entre le Protocole MAC et le Protocole de Transport TCP pour l'Optimisation des MANET. In Proc. of the 1st International Workshop on Mobile Computing & Applications (NOTERE'2007). Morocco, (2007).
16. Hamrioui, S., Lalam, M.: Incidence of the Improvement of the Transport – MAC Protocols Interactions on MANET Performance. 8th Annual International Conference on New Technologies of Distributed Systems (NOTERE 2008). Lyon, France, (2008).
17. Hamrioui, S., Lalam, M.: Incidences of the Improvement of the MAC-Transport and MAC–Routing Interactions on MANET Performance. International Conference on Next Generation Networks and Services, MARRAKESH, Morocco. (2010).
18. Jayasuriya, A., Perreau, S., Dadej, A., Gordon, S.: Hidden vs. Exposed Terminal Problem in Ad hoc Networks. In Proc of the Australian Telecommunication Networks and Applications Conference, Sydney, Australia. (2004).
19. Altman, E., Jimenez, T.: Novel Delayed ACK Techniques for Improving TCP Performance in Multihop Wireless Networks. IFIP International Federation for Information Processing. PWC 2003, LNCS 2775, pp. 237–250, (2003).
20. Kuang, T., Xiao, F., Williamson, C.: Diagnosing wireless TCP performance problems: A case study. In Proc. of SPECTS. (2003).
21. Bakre, B., Badrinath, R.: I-TCP: Indirect TCP for mobile hosts. In Proc. 15th Int. Conf. Distributed Computing Systems. (1995).
22. Brown, K., Singh, S.: M-TCP: TCP for mobile cellular networks. ACM Compueer Communication Review. 27(5). (1997).
23. Bensaou, B., Wang, Y., Ko, C. C.: .Fair Media Access in 802.11 Based Wireless Ad-hoc Networks. In Proc. Mobihoc. (2000).
24. Gerla, M., Tang, K., Bagrodia, R.: TCP Performance in Wireless Multihop Networks. *IEEE* WMCSA. (1999).
25. Gupta, A., Wormsbecker, C.: Experimental evaluation of TCP performance in multi-hop wireless ad hoc networks. In Proc. of MASCOTS. (2004).
26. Jain, A., Dubey, K., Upadhyay, R., Charhate, S.V.: Performance Evaluation of Wireless Network in Presence of Hidden Node: A Queuing Theory Approach. Second Asia International Conference on Modelling and Simulation. (2008).
27. Marina, M. K., Das, S. R.: Impact of caching and MAC overheads on routing performance in ad hoc networks. Computer Communications. (2004).
28. Ng, P. C., Liew, S. C., Sha K. C., To, W. T.: Experimental Study of Hidden-node Problem in IEEE802.11 Wireless Networks. ACM SIGCOMM'05. USA. (2005).
29. Bakre, A., Badrinath, B.: I-tcp: Indirect tcp for mobile hosts. IEEE ICDCS'95. Pages 136–143. Vancouver, Canada. 1995.
30. Balakrishnan, H., Seshan, S., Amir, E., Katz, R.: Improving tcp/ip performance over wireless networks. 1st ACM Mobicom. Vancouver, Canada. (1995).
31. Brown, K., Singh, S.: M-tcp: Tcp for mobile cellular networks. ACM Computer Communications Review. Volume 27, pages 19.43. (1997).
32. Tsaoussidis, V., Badr. H.: Tcp-probing: Towards an error control schema with energy and throughput performance gains. 8th IEEE Conference on Network Protocols. Japan. (2000).
33. Zhang, C., Tsaoussidis, V.: Tcp-probing: Towards an error control schema with energy and throughput performance gains. 11th IEEE/ACM NOSSDAV. New York, June. (2001).
34. Yuki, T., Yamamoto, T., Sugano, M., Murata, M., Miyahara, H., Hatauchi, T.: Performance improvement of tcp over an ad hoc network by combining of data and ack packets. IEICE Transactions on Communications. (2004).

35. Jimenez, T., Altman, E.: Novel delayed ack techniques for improving tcp performance in multihop wireless networks. Personal Wireless Communications (PWC'03). Italy. (2003).
36. Kherani, A., Shorey, R.: Throughput analysis of tcp in multi-hop wireless networks with ieee 802.11 mac. IEEE WCNC'04. Atlanta, USA. (2004).
37. Allman, M.: On the generation and use of tcp acknowledgements. ACM Computer Communication Review. Volume 28, pages 1114.1118. (1998).
38. Chandran, K.: A feedback based scheme for improving TCP performance in ad-hoc wireless networks. In Proc of International Conference on Distributed Computing Systems. (1998).
39. Holland, G., Vaidya, N. H.: Analysis of tcp performance over mobile ad hoc networks. Mobicom'99. Seattle. (1999).
40. Liu, J., Singh, S.: ATCP: TCP for mobile ad hoc networks. IEEE JSAC. vol. 19, no. 7, pp. 1300-1315. (2001).
41. Fu, Z., Greenstein, B., Meng, X., Lu, S.: Design and implementation of a tcp-friendly transport protocol for ad hoc wireless networks. 10th IEEE International Conference on Network Protocols (ICNP'02). (2002).
42. Biaz, S., Vaidya, N. H.: Distinguishing congestion losses from wireless transmission losses: a negative result. IEEE 7th Int. Conf. on Computer Communications and Networks. New Orleans, USA. (1998).
43. Liu, J., Matta, I., Crovella, M.: End-to-end inference of loss nature in a hybrid wired/wireless environment. WiOpt'03. INRIA Sophia-Antipolis, France. (2003).
44. Kim, D., Toh, C., Choi, Y.: TCP-BuS: Improving TCP performance in wireless ad hoc networks. Journal of Communications and Networks. vol. 3, no. 2, pp. 175–186. (2001).
45. Toh, C-K.: A Novel Distributed Routing Protocol to support Ad-Hoc Mobile Computing. In Proc. Of IEEE 15th Annual Int'l Phoenix Conf. Comp. and Commun. (1996).
46. Oliveira, R., Braun, T.: A Dynamic Adaptive Acknowledgment Strategy for TCP over Multihop Wireless Networks. Proc. Of IEEE INFOCOM. (2005).
47. Hamadani, E., Rakocevic, A Cross Layer Solution to Address TCP Intra-flow Performance Degradation in Multihop Ad hoc Networks, JOURNAL OF INTERNET ENGINEERING. VOL. 2, NO. 1. (2008).
48. Zhai, H., Chen, X., Fang, Y.: Improving Transport Layer Performance in Multihop Ad Hoc Networks by Exploiting MAC Layer Information. IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. VOL. 6, NO. 5. (2007).
49. Lohier, S., Doudane, Y. G., Pujolle, G.: MAC-layer Adaptation to Improve TCP Flow Performance in 802.11 Wireless Networks. WiMob'06, IEEE Xplore. Canada. (2006).
50. NS2. Network simulator. http://www.isi.edu/nsnam.
51. Fall, K., Varadhan, K., Notes and documentation. *LBNL*, http://www.mash.cs.berkeley.edu/ns. (1998).
52. Floyd, S., Jacobson, V.: Random Early Detection Gateways for Congestion Avoidance. IEEE/ACM Transactions on Networking. Vol. 1, pp. 397-413. (1993).
53. Bullington, K.: Radio Propagation Fundamentals. The Bell System Technical Journal. Vol. 36, No.3. (1957).
54. Perkins, C. E., Royer, E. M., Das, S. R.: Ad Hoc On-Demand Distance-Vector (AODV) Routing. *IETF Internet draft* (draft-ietf-manet-aodv-o6.txt).
55. Floyd, S., Henderson, T.: New Reno Modification to TCP's Fast Recovery. RFC 2582. (1999).
56. Hyytiä, E., Virtamo, J.: Randomwaypoint model in n-dimensional space. Operations Research Letters 33. (2005).