ISSN 2229-5518

Improve Performance of TCP over Mobile Ad-Hoc Network using a cross-layer solution

Mr. Yassine DOUGA, Dr Malika BOURENANE

Abstract-- The Transmission Control Protocol (TCP) is tuned to perform well in traditional wired networks, where packet losses occur frequently because of congestion. Unfortunately, in mobile ad hoc networks TCP is known to suffer from performance degradation since TCP connections are faced with problems such as high bit error rates, frequent route changes, multi-path routing and temporary network partitions. In our paper we present a new approach called Hybrid TCP or H-TCP that tries to adapt TCP for use in Wireless ad-hoc environment. In H-TCP we show that we can obtain better connections on the wireless links, while maintaining the advantages of TCP on the wired networks at the same time. We evaluated our approach via simulation with NS3 and compared the results with standard TCP-Reno scheme and a recent improvement of TCP performed on the basis of the signal strength. This comparison is done in terms of transmission time, interferences and mobility of nodes. The numerical results reveal that H-TCP achieves a significant improvement in the TCP transmission performance over mobile multi-hop wireless.

Index Terms— TCP; TCP Reno; Signal; noise; MANET;AD HOC ; Mobility ; Transport ; Packet ; RTT.

1 INTRODUCTION

Due to advancement of wireless technology and the proliferation of 802.11 based hand-held wireless terminals, recent years have witnessed an ever-increasing popularity of wireless networks, ranging from Wireless Local Area Networks (WLANs) to Mobile Ad hoc Networks (MANETs) [2-4]. In integrated WAN + LAN + 3G cellular systems, data and multimedia communications are carried end to end over the existing Internet infrastructure. In WLANs (Wi- Fi technology using 802.11) mobile hosts communicate with an Access Point (AP) or a Base Station (BS) that is connected to the wired networks. Visibly, only one hop wireless link is needed for communications between a Mobile Host (MH) and a stationary Fixed Host (FH) in wired networks. Most of the data traffic over the WLAN is TCP traffic, including traffic generated by web accesses, e-mails, bulk data transfers, remote terminals, etc. However, TCP/IP needs to meet the challenges introduced by the wireless portion of the network in order to make it compatible with the wired network for providing efficient services.

The ad hoc networks are wireless networks without beforehand defined infrastructure, where each entity (node) communicates directly with its neighbor. To communicate with other entities, it is necessary to rely on other entities in order to allow data to reach their destination. Therefore, it is very important that the entities are well positioned so they can establish links between them. This connection is achieved by the routing protocol. Thus, the operation of an ad hoc network differs itself notably from a GSM network or Wi-Fi networks with access points: while the base stations are necessary for most communications between nodes the network (Infrastructure mode), the ad hoc networks are self-organizing and each entity can play different roles. [4]

- *Mr Yassine DOUGA me is currently pursuingPHD degree program in wireless network engineering in University of Es-Sania Oran (LRIIR), Algeria, PH-00213661234324. E-mail: maximusssse@hotmail.com.*
- *Dr Malika BOURENANE is currently pursuing PHD degree program in wireless network engineering in University of Es-Sania Oran (LRIIR), Algeria. E-mail:mb-regina@yahoo.fr.*

A MANET consists of mobile platforms (e.g., a router with multiple hosts and wireless communications devices)--herein simply referred to as "nodes"--which are free to move about arbitrarily. The nodes may be located in or on airplanes, ships, trucks, cars, perhaps even on people or very small devices, and there may be multiple hosts per router. A MANET is an autonomous system of mobile nodes. The system may operate in isolation, or may have gateways to and interface with a fixed network. In the latter operational mode, it is typically envisioned to operate as a "stub" network connecting to a fixed internetwork.

The multi hop wireless networks share some of packets loss problems with MANETs. In most of these situations, reliable data transfer is required. It is well known that transport control protocol (TCP) has been well tuned to provide such services in traditional wired network environment. Due to its wide use in the Internet, it is desirable that TCP remains in use to provide reliable data delivery for communications within MANETs and for those across MANETs and the Internet.

When TCP packet loss occurs at a congested link, recovery mechanism is triggered by sender TCP either on arrival of Duplicate Acknowledgements (dupacks) or expiration of sender's Retransmission timer. To relieve the link congestion, TCP transmits at a lower rate by shrinking its transmission window or congestion window (cwnd). Thus TCP's loss recovery mechanism is unconditionally coupled with congestion control mechanism. Such TCP behavior works fairly well in

ISSN 2229-5518

wired networks, where packet losses are almost caused by link congestion; and packet loss due to the random Bit Error Rate (BER) is either negligible or not exceeding one packet loss per cwnd. When TCP works over wireless environments, several known problems affect its performance because of the differences in terms of bandwidth, propagation delay, and link reliability. TCP connections encounter other types of delays and packet losses than those are unrelated to congestion. First, packets may be lost due to high BER and channel asymmetries (both in the presence and in the absence of user mobility). Second, communication may pause during hand-off between cells. Third, packets may be lost, while a mobile host is out of reach from other transceivers [24].

Many studies showed that the standard version TCP functions poorly in wireless environment because of its incapacity to distinguish the causes of packets loss caused by the congestion from those caused by transmission errors. As a consequence, appropriate congestion control should be considered to be a key issue for MANETs. There is a large variety of ideas on how to overcome the inherent difficulties to MANETs, most of these suggestions try to solve just one problem as interferences or mobility and others problems or they are designed for cellular networks where the only wireless link exists between the base station and the mobile node. In contrast, ad-hoc networks are multi-hop networks in which data has to be transmitted over several wireless links, from the sender to the receiver. In this paper we provide a proposal to deal with the congestion control problem in mobile multi hop ad-hoc networks.

The rest of this paper is organized as follows: Section 2 describes the related works and the solutions that were used as base to implement the new solution. In Section 3 we provide the proposed solution (HYBRID TCP). In the section 4 we talk about the simulation and the results. At the end (section 5) we finished by the conclusion and the future work.

2 Related Works

A lot of research to mitigate the performance problem of TCP, related to the wireless mobile environment is happening and as a result, several versions have emerged. Earlier attempts to address this problem can be broadly divided into two groups. The first group does not attempt to change or modify the TCP protocol, instead exploits methods such as injecting, removing or delaying TCP packets based on a superior understanding about what is happening at the link layer. These mechanisms provide solutions at the Link Layer (LL), by hiding the deficiencies of a wireless channel from TCP, referred as TCP unaware. On the other hand, the second group endeavors for modifying the behavior of TCP. These mechanisms are referred as TCP aware.

2.1 Loss differentiation algorithms with RTT

Due to the wide use of TCP on the Internet, considerable efforts have been made to improve it. One of the most successful attempts is TCP-Reno. It not only improves the retransmission mechanism, but also the congestion avoidance mechanism. Losses must occur in order for a source to decreases its speed because the congestion control mechanism of TCP increases the congestion window over time and reduce it only when there are losses. In TCP-Reno algorithm many steps are taken to anticipate and avoid the losses. The main idea of TCP-Reno is that: the source can estimate the round trip time (RTT) for a connection by the following formula:

$$RTT\ (i+1) = \alpha\ (RTT\ (i)) + (1-\alpha)\ M \quad (1)$$

Where:

- α is a constant between 0 and 1 which controls how rapidly the RTT adapts to changes.
- M = time taken to return the ACK of a segment
- RTT (i+1) is the value of RTT that TCP-Reno will Estimate.
- RTT (i) is the value of RTT of the last packet sent.

After this, TCP-Reno will compare the value of RTT (i +1) (estimated value) with that of current RTT and take their differences:

$$Diff = RTT\ (i+1) - RTT \quad (2)$$

There are two cases depending on the value of Diff:

- If Diff> = 0 then TCP Reno increases its congestion window linearly during the next RTT.
- If Diff <0 TCP Vegas reduces its congestion window linearly during the next RTT.

TCP-Reno has been widely tested on simulated networks. Experiments show that TCP-Reno gives better performance than the TCP versions that preceded it [1].

The main disadvantage of the solution based on the RTT is the mobility since it is applicable only in a static wireless environment. If the network nodes change position regularly, the time required for transferring a packet in a mobile host cannot be correctly estimated [1].

2.2 Solution using the signal power

The principle of this method is mainly to suppress the degradation in the performance of TCP due to mobility. To reach this goal some mechanisms have been proposed to reduce the number of packet losses. These mechanisms are based on measurements of the signal power at the physical layer. Based on these measures of signal strength, when a node does not communicate with a neighbor, the MAC layer is an estimate to determine if the cause of the failure is due to congestion or the neighbor who has moved out of reach. If the MAC layer considers that the neighbor has just moved out of reach, then, it stimulates the physical layer to increase the transmitting power temporarily and try to keep the active link with the neighbor. It also pushes the network layer to find a new route. The signal strength measurements can also be used to predict possible failures to link to a neighbor who is about to move out of range. Thus, if the measurements indicate that the signal strength decreases and the link is likely to break, a search for a new route may be initiated proactively before the link actually fails. While researching the new route, the network layer should be careful to avoid provisional links high power as well as weaknesses (links that may fail soon). The protocol on demand ad hoc routing of distance vector (AODV) was modified by proceeding to the exclusion of these links and seeking a new route. To cope with failures that are due to congestion, we propose a simple mechanism by which the MAC layer, based on its estimate on the position of neighbor (if still in the margin) persists in its attempt to reaching that neighbor for a longer period [2].

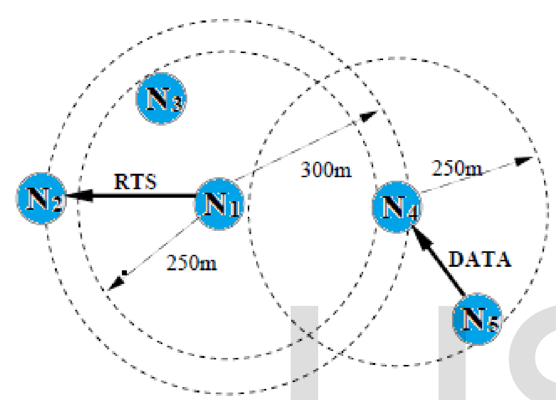


Fig 1. Effect of power asymmetry.

2.3 TCP un-aware Solutions

LL protocol running on top of the physical layer is more adaptable to link characteristics than higher-level protocols and it is much faster as it has immediate knowledge of dropped frames. Alleviating the inefficiencies of the wireless medium at the LL provides the transport layer protocol with a dependable communication channel, similar in characteristics to a wired channel. This is realized using an Automatic Repeat reQuest (ARQ) scheme and/or by means of Forward Error Correction (FEC). ARQ in presence of high error rates can lead to a large volume of retransmissions and can even cause a complete black-out in the connection. On the other hand, FEC is not very well suited for channels with bandwidth constraints common in wireless environments. It also increases power requirement and introduces computation delays for each packet. The 'Delayed Dupacks' scheme looks at the same problem but in the context of a reliable link layer protocol which acknowledges each packet and performs fast retransmission. TULIP (Transport Unaware Link Improvement Layer) also attempts to recover from retransmission losses before TCP coarse-grain timeouts occur. These schemes, in absence of interaction between TCP and MAC protocols lead to futile TCP retransmission, causing wastage of scarce resources and performance problems on end to end basis. In order to overcome limitations of the above mechanisms, TCP has to be made aware of what is happening at link layer for making it adaptable to the link characteristics [24].

2.4 TCP aware Solutions

Here the approach is to divide the end-to-end connection between a FH and a MH into two separate connections at the wired-wireless border (e.g., BS). Major problem with split connection in approaches like I-TCP and M-TCP is hand-off latency due to mobility of MH between BS besides violation of the end-to-end semantic of TCP. Additionally, these approaches are vulnerable to scalability issues due to requirement of per-flow support from the BS or modification in TCP at the FH. These schemes demand for modifications at intermediate nodes; which also create practical issues at the time of deployment in existing networks. Taking into consideration these limitations, the authors have focused on approaches with modifications only at the MH. In Freeze TCP, upon receipt of an indication of disconnection from lower layer, TCP at the MH sends a Zero Window Advertisement (ZWA) to the FH, to pause transmission. Upon reconnection, a fast retransmit is adopted to restart transmission. The main drawback of Freeze TCP is dependency on the network layer to predict future disconnections. The approaches like M-TCP and Freeze TCP may also trigger bursty TCP transmissions and affect the competing traffic. To our knowledge, certain issues as listed below have remained unresolved.

1) The majority of TCP schemes largely rely on signals from end systems or an intermediate node for resuming transmissions after link re-establishment. In presence of network latency, the above schemes would not be able to perform well as the triggering of transmission with negligible delay is unlikely to happen.

2) Futile link layer re-transmissions for loss recovery over a wireless link further delay transmission of subsequent TCP packets from its buffer. This results into false Round Trip Time (RTT) estimation leading to an increased RTO interval. A longer RTO interval subsequently prevents TCP from regaining the available network bandwidth, sooner.

Therefore, there is a need for a solution which can improve TCP performance in wireless environment suffering from frequent disconnections, without sacrificing end to end TCP semantics. In the next section, the impact of link failures on the performance of TCP is examined, in order to define the problem, related to the inherent weaknesses of TCP [24].

2.5 The disadvantages solutions

In this part we present the disadvantages of the solution based on RTT and the one on signal strength:

The main disadvantage of the first solution (RTT) is the mobility since it is applicable only in a static wireless environment where nodes of the network do not move. This is due to the fact that this solution is based on the round trip time RTT a TCP packet. However, this solution is not valid if the network nodes change position regularly because we cannot estimate the time required for transferring a packet in a mobile

ISSN 2229-5518

host [1].

The disadvantage of the second solution (signal strength) is due to the fact that it works only in cellular networks where the only wireless link exists between the base station and the mobile node, and the other disadvantage is that it takes into account only the case where a node is out of range of the wireless network while they may be other nodes in the range of the wireless network but their signal strength is degraded because of other factors such as technical problems due to interference from the external environment [2].

3 THE HYBRID SOLUTION

As shown before, generally the TCP is not really compatible with wireless networks, because TCP consider that all packets losses are due to the congestion, this fact degrades the network performances. Therefore in order to solve this problem of compatibility several approaches have been proposed [1], [2], [3], [8], [9], [10], [11], [24]. Most of these approaches use a cross layer solution which is an escape from the pure waterfall-like concept of the OSI communications model with virtually strict boundaries between layers. The cross layer approach transports feedback dynamically via the layer boundaries to enable the compensation as a method. The purpose of our paper is to provide a cross layer solution based on the RTT formula of V. Jacobson and exploits the force of the wireless signal and the noise value to improve TCP in MANETs. The particularity of the approach is that it takes into account the external environment of the network.

In the proposed solution, the signal strength and the value of noise of the wireless environment are taken as factors to determine the how to react to the packet loss. It is used to calculate the estimated value of round trip time in the normal case (estimated RTT) and compare it with the real RTT to determine the cause of the loss and act according to the analysis.

In order to detect the real cause of the packet loss when it occur on mobile wireless environment and reacts according to it, we integrate into the TCP structure two new variables, the first variable is called LSS (low signal strength) which contains the lowest value of signal strength of all jumps between sender and receiver, the second variable is called HNV (high noise value), represents the highest value of noise of the wireless environment. To integrate these variables (LSS, HNV) we used a cross layer solution, both of the signal strength value and the noise value are physical layer variables and we can't integrate them on the transport layer without using this method.

These two variables are used to detect the real cause of the packet loss. The first variable LSS is used to determine if any used node for transmitting data (sender, receiver or intermediate node) is out of the wireless environment range, by comparing this value with an predefined value of RSS which is necessary to establish communication. By using this comparison TCP will determine if the packet loss is due to mobility or not (when a node has a low value of RSS that mean that it is out the wireless rang). The second variable HNV is used to determine if the value of noise is enough high to stop or interferes the packets transmission, in this case the loss is due to noise or the value is negligible so if a packet loss occurs TCP decides that the cause is not interferences.

When transmitting data normally (without packet) loss between sender and receiver on wireless area using TCP with multi jumps connection, each node when it receives the packet before transferring it to the next node, it updates the value of LSS and HNV by using this way:

- If the value of signal strength of the current node is less than LSS then LSS takes as value the current node value of signal strength , if not than it will do nothing.
- If the noise value of the current node is higher than HNV then HNV takes as value the current node noise value, if not than it will do nothing.

After updating these two values it sends the packet to the next node until the packet reaches its destination.

To decide that a packet loss has occurred, after sending a packet, the sender will waits an RTT value of time; if it does not receive an ACK (acknowledgement packet) form the receiver then it decides that the sent packet was lost.

To estimated how much time the sender has to wait before being sure that the sent packet was lost, TCP use a formula, this formula works well with the wired networks but not with the MANET, because of the fact that they are always mobile this formula does not give the same accurate results as on the wired networks. To adapt this formula with the MANETS we have introduced in it a new factors which are the signal strength and the noise, more value of signal strength is lower and the value of noise is bigger more the waiting time will be more bigger.

The classical formula that is used to estimate the RTT value is (this is a simplified format):

$$RTT\ (i+1) = \alpha * RTT\ (i) + (1-\alpha) * M \qquad (3)$$

Where:

- α Is a constant between 0 and 1 which controls how rapidly the RTT adapts to changes?
- M = time taken to return the ACK of a segment
- RTT (i+1) is the value of RTT that TCP-Reno will Estimate.
- RTT (i) is the value of RTT of the last packet sent.

The new formula that we proposed to solve the mobility problem is (this is a simplified format):

$$RTT\,(i+1) = (\alpha * RTT\,(i) + (1-\alpha) * M) * X \quad (4)$$

$$X = f\,(LSS, HNV) \qquad (5)$$

Where:

- α is a constant between 0 and 1 which controls how rapidly the RTT adapts to changes.
- M = time taken to return the ACK of a segment
- RTT (i+1) is the value of RTT that TCP-Reno will Estimate.
- RTT (i) is the value of RTT of the last packet sent.
- X is a reducing factor calculated by using the function f.
- f is a predefined function that takes as factors the values of LSS and HNV to calculate the reducing factor.

Usually, the congestion window cwnd increases by 1 at each receiving of an ACK until it reaches the ssthresh threshold. TCP uses packet loss to infer the presence of network congestion and reacts accordingly. Indeed, in order to detect the congestion cases TCP compares the current value of RTT with its estimated value. When this current value is less or equal to the estimated value, TCP congestion mechanism will not be activated because this difference reflects the absence of packets loss or congestion. Otherwise, it considers that the packets loss is only due to the congestion, and reduces the cwnd window. Such action degrades TCP throughput and the network performances. This, results from the fact that TCP considers all packet loss as congestion.

However, our approach assumes that in order to differ congestion losses from the other wireless losses.

As we said before when a packet loss occurs on mobile wireless network, using classic TCP, the network performances will dramatically decrease, that is why we create Hybrid TCP, this new version of TCP solves the packet loss problem and keeps the network working with high performances.

Here we explain how HYBRID TCP reacts to packet losses:

The first thing that HYBRID TCP does after detecting that a packet was lost is checking the value of LSS (LOW SIGNAL STRENGTH VALUE) and HNV (HIGHEST NOISE VALUE) one after the other.

First, if HYBIRD TCP finds that the value of LSS is not enough to establish connection, it deduces that the packet loss is due to mobility and there is one or many nodes used to communication which are moved out of the wireless area. To solve this problem, HYBRID TCP tries to find a new path where all used nodes are on the wireless range by using the AODE routing protocol which is the most efficient routing protocol on MANET according to many studies [25], [26]. When it finds a new functional path HYBRID TCP resends the lost packet without doing anything else (without reducing the flow window).

After checking the signal strength value, HYBRID TCP will check now the value of noise by getting back to the value of (HNV). If HNV is too high to interfere with packet transmission, then HYBRID TCP deduces that the loss was caused by interferences. To solve this problem of interferences, HYBIRD TCP Changes the used wireless channel and resends the lost packet into this new channel.

Finally, after checking (LSS and HNV) and if there is no mobility problem or interferences problem then HYBRID accords the packet loss to congestion, in this case and just in this case HYBRID TCP triggers the congestion avoidance mechanism.

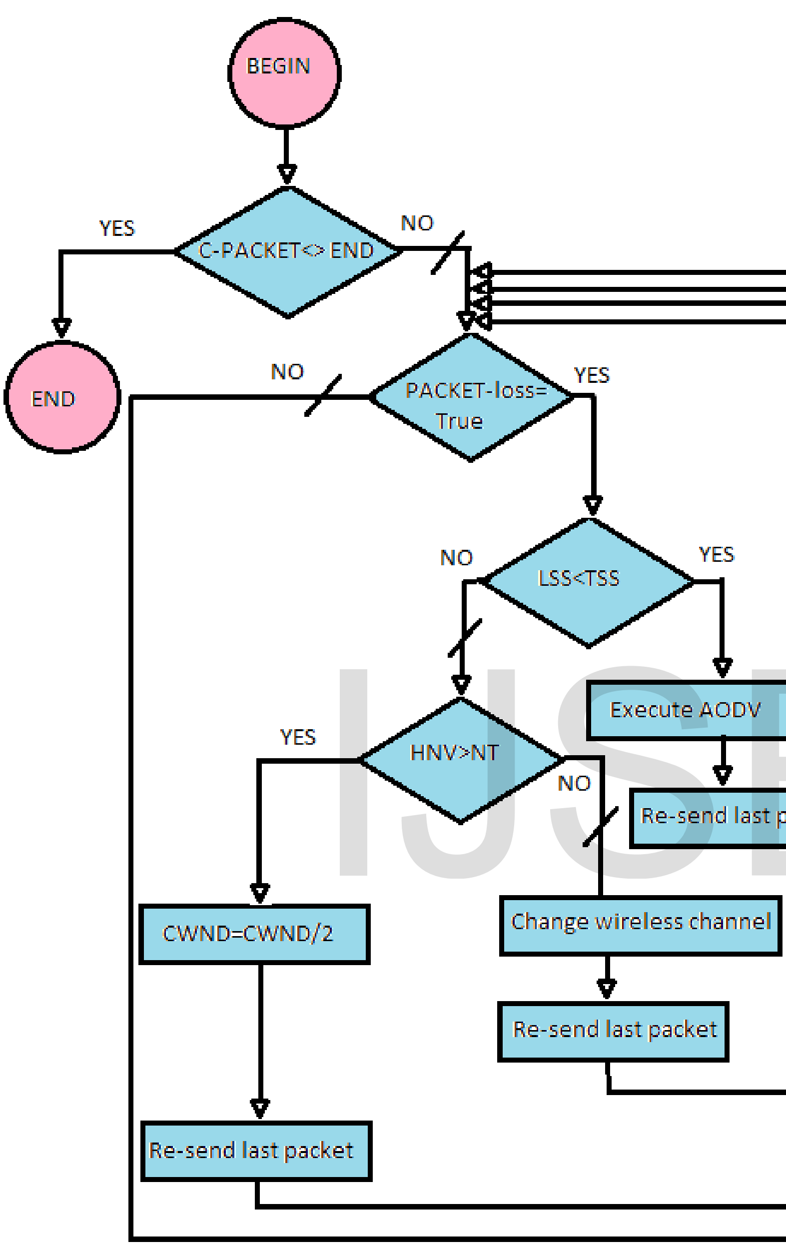
HYBRID TCP algorithm:

When any node receive packet →

```
If (current-node<>sender-node)
{
  CSS = get (signal-strength); //get back the value
                    Signal strength of
                    the current node.

  CN = get (noise); //get back the noise value
              of the current node.
 If (LSS>CSS)
    {
     LSS = CSS; //update LSS
    }
 IF (HNV<CN)
    {
      HNV = CN; //update HNV
    }
}
Else
{
  CSS = get (signal-strength); //get back the value
           Signal strength of the current node.
```

CN = get (noise); //get back the noise value
of the current node



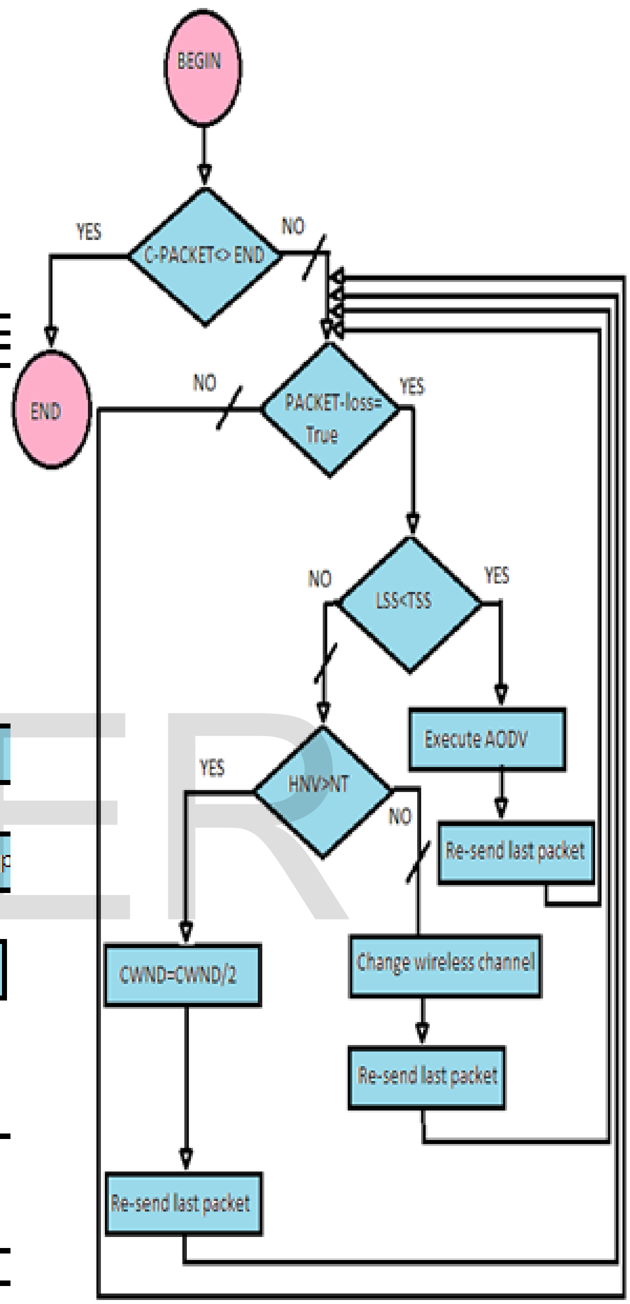


Fig 2. HYBRID TCP organization chart.

4 Simulation and Results

With respect to simulation, we used the network simulator NS3 because of its big library of TCP versions and the fact that it supports the modules of mobility and interferences that helped us in our simulation to represent most of the possible cases. The goal of the ns-3 project is to create an open simulation environment for networking research which is preferred inside the research community.

About the topology we choose, under the effect of interferences, an ad-hoc wireless topology with few numbers of mobile nodes between 10 and 20 by means of the dynamic routing AODV protocol, the value of signal strength is a variable.

The classes that we used to implement our HYBRID TCP and all its functionality are: TOPOLOGIE-TEST.cc/ NODE.cc/ MOBILITY.cc/ TCP-Tahoe.cc/ TCP-SOCKET-BASE.cc/ SET-RTT.cc/ RTT-ESTIMATOR.cc/PACKET.cc/ YANS-WIFI-PHY.cc/ (get signal, get noise).

First we begin by exchanging the packets from end to end through several hops as in the figure 3. The nodes are mobile and under the effect of interferences and the value of the interferences and mobility of nodes are adjustable so that we may change it constantly during the simulation. After receiving any packet, every used node (as jump) will update the value of LSS and HNV using the mechanism that is explained on the chapter "THE HYBRID SOLUTION".

The principle of the simulation is to send many packets over multi-hop and each time we change the nodes mobility and the noise of the wireless environment to see how HYBRID TCP will react to these conditions. We also do the same simulation with the same conditions using the TCP RENO version, to see how much can HYBRID TCP improves the network performances compared to TCP version which was designed for wired networks. We also compare HYBRID TCP to another version of TCP which is a try to improve TCP on the MANETS; this version is "Signal Strength based Link Management" [2].

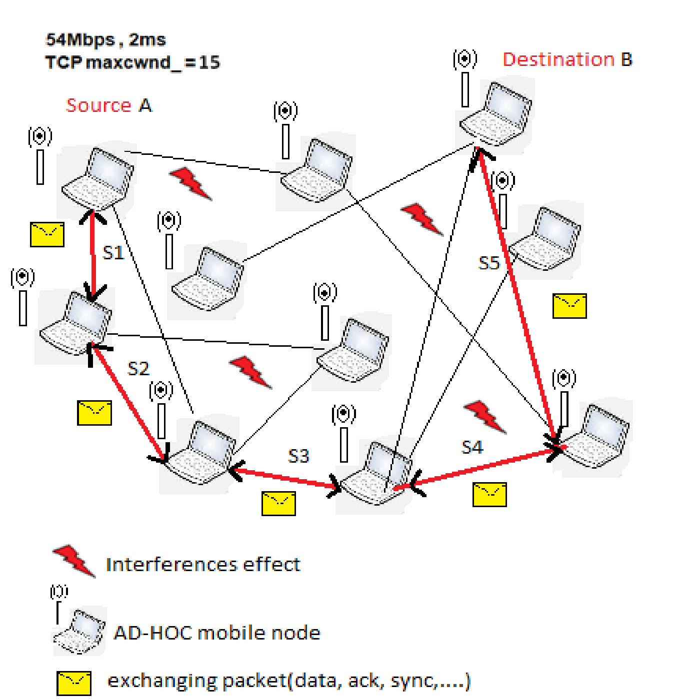


Fig 3. Wireless ad-hoc topology with interferences (Used topology).

As shown in the figure 3, we send packet from host "A" to the destination host "B" by going through S1, S2, S 3, and S4.

After each simulation we create 3 types of graphs to follow the reaction of the HYBRID TCP and its capacity to make a difference between congestion and other problems of packets loss, we compare it with the TCP Reno and "Signal Strength based Link Management", by changing the location of the hosts and the environment interferences. Thus each time, we calculate the size of TCP window evolution (CWND). The figures 4, 5 and 6 represent the corresponding graphs.

ISSN 2229-5518

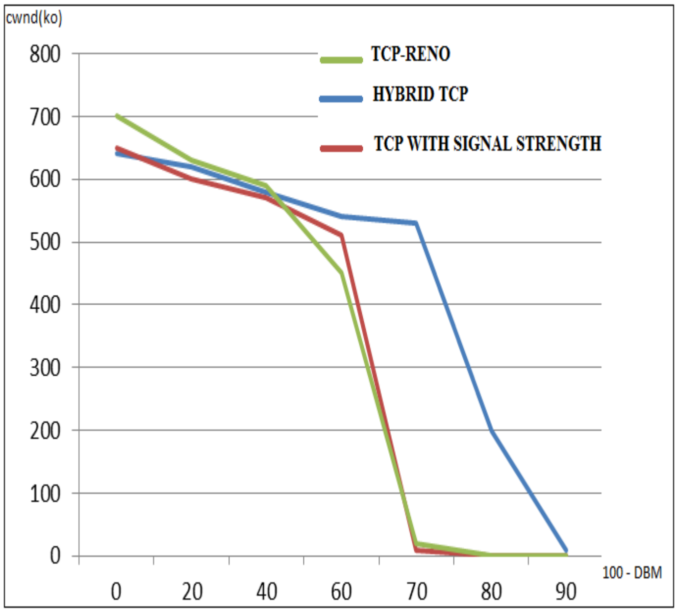


Fig 4. TCP flow windows by considering interferences.

As indicated in figure 4, the results show that when we expose the wireless environment to low value of interferences, HYBRID TCP, TCP RENO and TCP with signal strength, all of them works well and without any problems, but when we increase the noise, we note that HYBRID TCP gives better results and better network performance than TCP RENO and TCP with signal strength, that's because when a packet loss due to interferences occur, the other versions reduce the size of CWND but HYBRID TCP each time try's to change channel and solves the problem of noise and this keeps the network working with good performances.

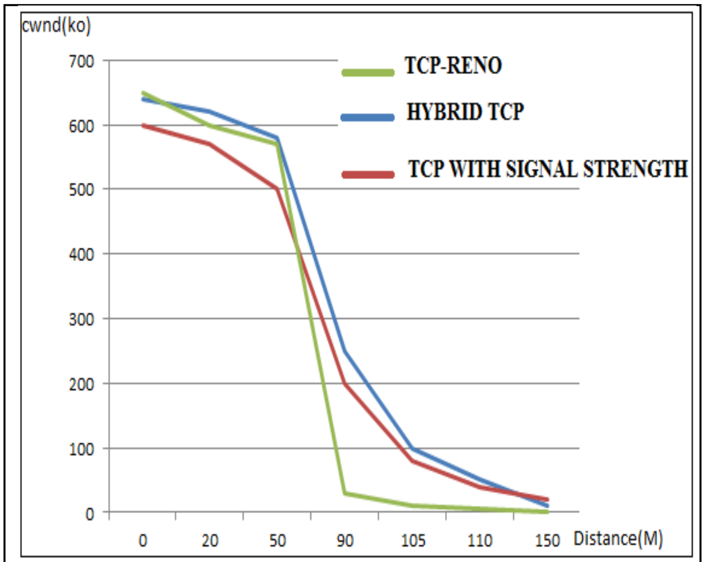
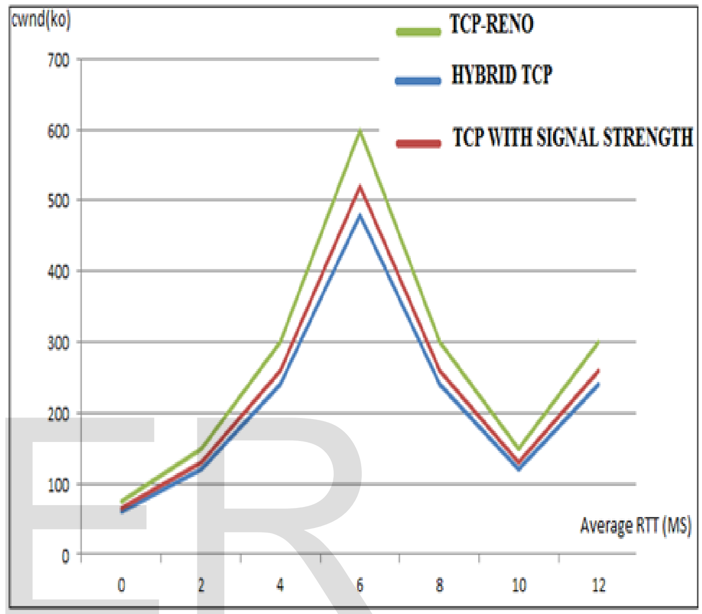


Fig 5. TCP flow windows by considering mobility.

The figure 5, shows the average evolution of the CWND (flow window) which will change the throughput and the transmission time by changing the distance between the used nodes (used to transmit packets), the simulation proves that the HYBRID TCP and TCP with signal strength gives good results compared to TCP RENO, because they takes in consideration the nodes mobility. For HYBRID TCP, each time that one or many nodes goes out of the wireless range , it execute the routing protocol AODV to find a new path to the destination, but TCP RENO considers the mobility as a congestion, this fact degrades the network performances.

Fig 6. TCP flow windows by considering time.



On the figure 6 we note that in a normal condition (no interferences neither no mobility of nodes) all studied versions of TCP works normally and without any problems.

We also add a simulation that compares the studied version of TCP which are: HYBRID TCP, TCP with signal strength and TCP RENO, in term of energy consumption on a wireless environment with mobile nodes and under the effect of interferences, the Figure 7 shows the results of the simulation.

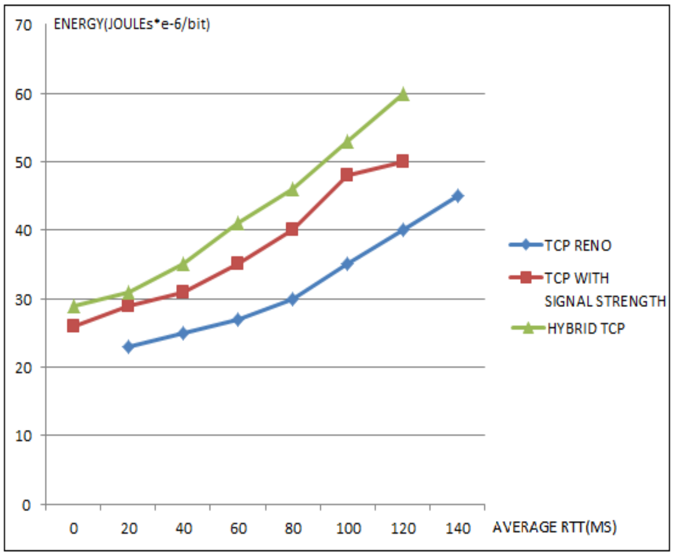


Fig 7. Total Energy E per bit and Goodput for 1%packet loss with no RTS/CTS.

The figure 7 shows that in terms of energy consumption, HYBRID TCP is very greedy compared to TCP RENO and TCP with signal strength solution, that's because of number of operations, that it execute during the loss of packets. In fact we try to optimize the energy consumption of HYBRID TCP to more improve it.

5 Conclusion

In this paper, we presented an essay to improve TCP performances in mobile wireless ad-hoc networks (MANET). Our enhancement is based on enabling TCP to distinguish between different causes of packet loss, as congestion, interferences or mobility of nodes. This approach was implemented as a simulation using NS3. In this article we have represent the first results of the simulation which are obtained by using approximated factors and formula to compare HYBRID TCP, TCP Reno and TCP with signal strength solution to prove that using HYBRID TCP in ad-hoc networks gives the best result in comparison to other studied TCP versions. It is hoped that our future works will try to improve the results of HYBRID TCP by giving the best optimized formula and factors by means of more simulations which will take into consideration the most frequents factors in order to select the best way to detect the causes of packets loss (congestion, mobility or interferences) and also to have a solution which allow us to optimize the energy consumption of HYBRID TCP. Finally why not try to adapt our formula to new versions of TCP.

REFERENCES

[1] G. Marfia, C. Palazzi, G. Pau, M. Gerla, M.Y. Sanadidi and M. Roccetti, (2007), TCP Libra: Exploring RTT-Fairness for TCP, Proceedings of 6th International IFIP-TC6 Networking Conference, Atlanta, GA, USA, Lecture Notes in Computer Science Volume 4479, pp 1005-1013.

[2] F.Klemm, Zhenqiang-Ye, S.V. Krishnamurthy and S.K. Tripathi, (2005) Improving TCP Performance in Ad Hoc Networks using Signal Strength based Link Management, Journal Ad Hoc Networks Volume 3 Issue 2, March, Pages 175-191.

[3] H. Jiang and C. Dovrolis, Passive Estimation of TCP Round Trip Times, (July 2002), Newsletter ACM SIGCOMM Computer Communication Review Volume 32 Issue 3, Pages 75-88.

[4] Wei-Qiang Xe and Tie-Jun Wu, (Jan. 2006), TCP Issues in Mobile Ad hoc Networks: Challenges and Solution, Journal of Computer Sciences & Technology, Vol.21, No.1, pp.72-81.

[5] Chang-hyeon Lim and Ju-wook, Jang, , (March 2007), An Adaptive End-to-End Loss Differentiation Scheme for TCP over Wired/Wireless Networks, IJCSNS International Journal of Computer Science and Network Security, Vol.7 No.3.

[6] L. Yao-Nan, and H. Ho-Cheng, (JUN 2007), A New TCP Congestion Control Mechanism over Wireless Ad Hoc Networks by Router-Assisted Approach, International Conference on Distributed Computing Systems, pp:84-84.

[7] S. Qamar and K. Manoj, (2010), Impact of Random Loss on TCP Performance in Mobile Ad hoc Networks (IEEE 802.11), (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, No. 1.

[8] D. Kliazovich and F. Granelli , (Nov 2006), Cross-layer congestion control in ad hoc wireless networks, Ad Hoc Networks Journal (Elsevier), Vol. 4, Issue 6, pp: 669-792.

[9] A. M. Al-Jubari, M. Othman, B.M. Ali, N.A. Wati and A. Hamid, (2011), TCP performance in multi-hop wireless ad hoc networks: challenges and solution, EURASIP Journal on Wireless Communications and Networking.

[10] Mirhosseini S.M, (2008), Improvement of TCP Performance in Ad Hoc Networks Using Cross Layer Approach, Proceedings of 3rd International Conference on Systems and Networks Communications, ICSNC '08.

[11] M. Gunes and D. Vlahovic, (2002), The Performance of the TCP/RCWE Enhancement for Ad-Hoc Networks, Seventh International Symposium on Computers and Communications (ISCC'02) Proceedings.

[12] S.Anbu Karuppusamy, K.Batri, (2012), Improving the performance of TCP inAd hoc networks based on signal strength and buffering system, Journal of Applied Sciences Research, 8(5): 2554-2563.

[13] Md.M. Ali, A. K. M. S. Alam and Md.S. Sarker, (2011), TCP Performance Enhancement in Wireless Mobile Ad Hoc Networks, International Journal on Internet and Distributed Computing Systems (IJIDCS), Vol: 1 No: 1.

[14] S. Lohier, Y. G. Doudane and G. Pujolle, (Nov 2007), Cross-layer loss differentiation algorithms to improve TCP performance in WLANs, Telecommunication Systems, Volume 36, Issue 1-3, pp 61-72.

[15] P. Dalal, N. Kothari and K. S. Dasgupta, (2011), Improving TCP Performance Over Wireless Networks With Frequent Disconnections, International Journal of Computer Networks & Communications ISSN 0975-2293, Vol. 3, Issue: 6, Start page: 169.

[16] A. Sachan, A. Rajput, (2010), Comparison of TCP Performance on WLAN, Technical report.

[17] T. Mahmoodi, V. Friderikos, O. Holland and A.H. Aghvami, (2007), Cross-Layer Design to Improve Wireless TCP Performance with Link-Layer Adaptation, Proceedings of the 65th IEEE Vehicular Technology Conference (VTC Spring 2007), pp: 1504-1508.

[18] J. Postel (Aug 1980), “RFC 768: User datagram protocol”.

[19] V. Jacobson, Congestion Avoidance and Control, SIGCOMM Symposium no Communication Architecture and protocols.

[20] V. Jacobson (Apr 1990), Modified TCP Congestion Control and Avoidance Alogrithms. Technical Report 30.

[21] S. Lukin, (2010), A Comparison of Round-Trip Time Estimation Algorithms, Loyola University Maryland, UCSC SURF-IT Research.

[22] Ehab Aziz Khalil, (Nov 2012),A Modified Congestion Control Algorithm for Evaluating High BDP Networks, Dept. of Computer Science & Engineering Faculty of Electronic Engineering (MinufiyaUniversity), IJCSNS International Journal of Computer Science and Network Security, VOL.12 No.11.

[23] Ahmad Al Hanbali, Eitan Altman and Philippe, (2005), A survey of TCP over Ad Hoc Networks, *IEEE communications surveys & tutorials*,Third Quarters,Vol**:** 7 pp. 22 - 36.

[24] Purvang Dalal1, Nikhil Kothari1 and K. S. Dasgupta, (Nov 2011), IMPROVING TCP PERFORMANCE OVER WIRELESS NETWORK WITH FREQUENT DISCONNECTIONS, International Journal of Computer Networks & Communications (IJCNC) Vol.3, No.6.

[25] Durgesh Wadbude, Vineet Richariya, (April 2012), An Efficient Secure AODV Routing Protocol in MANET; International Journal of Engineering and Innovative Technology (IJEIT) Volume 1, Issue 4.

[26] Dr. Aditya Goel, Dr. Ajaii Sharma, (2009), Performance Analysis of Mobile Ad-hoc Network Using AODV Protocol, International Journal of Computer Science and Security (IJCSS), Volume (3): Issue (5).