TCP WELCOME

TCP variant for Wireless Environment, Link losses, and COngestion packet loss ModEls

Alaa SEDDIK-GHALEB *(Author)*, Yacine GHAMRI-DOUDANE
Networks & Multimedia Systems Research Group (LRSM), ENSIIE.
Evry- France.
{seddik, ghamri}@ensiie.fr

Sidi-Mohammed SENOUCI
France Telecom R&D.
Lannion, France
sidimohammed.senouci@orange-ftgroup.com

Abstract— **The problem of TCP and all its existing variations within MANETs resides in its inability to distinguish between different data packet loss causes. Thus, TCP has not always the optimum behaviour in front of packet losses which might cause network performance degradation and resources waste. Multiple Loss Differentiation Algorithms (LDAs) have been designed to improve TCP performances. They had been optimized for data networks where only the last link is a wireless link. In these LDAs the most common packet losses that are handled are those due to wireless channel errors or congestion. We show in this paper that a third common packet loss cause, link failure, has to be handled in multi-hop wireless networks such as mobile ad hoc networks (MANETs). In order to handle these three packet loss causes, we propose TCP-WELCOME. This latter follows a two-step process: (i) first, distinguish between most common packet loss causes, and (ii) then, triggers the most appropriate packet loss recovery according to the identified loss cause. The performance evaluation shows that TCP-WELCOME optimizes both energy consumption and throughput. Also, TCP-WELCOME does not change the standard as its loss differentiation and recovery algorithms can operate with the already existing TCP variants.**

Keywords*- TCP, Light-infrastructure networks, MANET, LDA, LRA, Energy consumption.*

I. INTRODUCTION

TCP is considered as the most popular reliable transport protocol today. It is compatible with almost all other Internet protocols and applications. However, TCP as it exists nowadays may not well fit in wireless ad hoc networks since it was designed for wired networks where network congestion is the primary cause of data packet losses. On the contrary of wired links, wireless radio channels are affected by many factors that may lead to different levels of channel errors. Wireless channel behavior cannot be predictable, but in many cases, such channels have high channel errors that cannot be neglected when studying light-infrastructure networks such as wireless ad hoc networks. Furthermore, in addition to wireless channel behavior, there are many other factors that could affect TCP performance within this kind of networks. Link failures and network partitioning due to nodes' mobility or battery depletion may have a negative effect on the performances of TCP connections. Hence, TCP does not have the capability to recognize whether the packet loss is due to network congestion, channel errors, or link failure. It reacts to all packet losses as if it was due to congestion. To overcome this problem, there is a need of:

1- Finding a fine grained classification of data packet loss causes for TCP within a mobile ad hoc network. This is known as Loss Differentiation Algorithm (LDA).

2- Developing an appropriate loss recovery algorithm according to the identified packet loss model. This is known as Loss Recovery Algorithm (LRA).

We present, in this paper, a new TCP variant for Wireless Environment, Link losses, and COngestion packet loss ModEls (WELCOME) in MANETs. Our first goal is to define the most appropriate LDA that leads to better data packet loss classification. The proposed LDA should be able to recognize the three above mentioned models of data packet loss within MANETs (i.e. link failure, wireless channel errors, congestion). The second goal is to define a suitable LRA that makes TCP able to react accordingly with the most adequate loss recovery strategy for each classified data packet loss.

The remainder of this paper is organized as follows: after presenting the related work and our motivations behind this study in Sections 2 and 3, Section 4 introduces TCP-WELCOME and describes its algorithms. Section 5 discusses the simulation scenarios used to evaluate TCP-WELCOME and the obtained results. Finally, we summarize the main results and give some ideas for future work in Section 6.

II. RELATED WORK

Loss differentiation and classification algorithms can be categorized into two classes according to their characteristics:

implicit and explicit [1]. The major contributions within these two classes and their limitations in MANETs are discussed in the following.

A. Implicit Loss Classification in the TCP Congestion Control Mechanism

The main TCP variant within this class is TCP Westwood [2] [3] [4]. TCP Westwood is a sender-side modification of TCP New Reno [5] that estimates the connection bandwidth based on the rate of the received acknowledgements. TCP Westwood uses the estimated bandwidth to adjust and set its congestion window and slow-start threshold parameters. This is in contrast to traditional TCP congestion control implementation, where both congestion window size and slow-start are updated using simple rules (i.e. additive increase/multiplicative-decrease) when an ACK is received or is missing [6]. This alternative bandwidth estimation algorithm enhances the performance of TCP Westwood in front of random, sporadic data packet losses which are mainly due to wireless channel errors. In [7], the authors show that TCP Westwood performances (throughput and energy consumption) decrease drastically with the increase of the Bit Error Rates (BER). Indeed, when the BER increases over the wireless channels, the returned ACKs might be lost or corrupted. These lost or corrupted ACKs could lead to mistaken estimated bandwidth calculations which degrade the performances of TCP Westwood. Furthermore, in front of link losses, still TCP Westwood reacts as the former TCP variants do, i.e. it recognizes the packet loss with the Retransmission TimeOut (RTO) expiration. Thus, it reacts as in the case of strong congestion by decreasing its throughput to the minimum which leads to an important unnecessary drop in performance. So, one of the major drawbacks to use TCP Westwood in MANETs is its inability to handle link losses situations where burst packet losses occur and an ad hoc re-routing is required.

B. Explicit Loss Identification through Estimation Techniques

Many proposals were trying to classify explicitly data packet losses within the network using Loss Differentiation Algorithms (LDA) to react in an appropriate manner. In [8] the authors suggest that, instead of applying solutions at the transport layer, the problems of TCP within wireless networks can be circumvented by using physical and link-layer solutions such as forward error correction (FEC) and/or link level retransmissions. They state that the loss differentiation is often performed at the receiver side and the congestion control at the sender side. Thus, they propose a simple checksum based approach for loss differentiation together with two loss notification schemes in the context of TCP. This solution is beyond the scope of our suggestions as we do not want to use notifications mechanism. Indeed, when used in MANETs such notification mechanisms will imply all the crossed nodes and will hence increase the overhead of TCP execution. This overhead has a negative impact on both network resource usage and energy consumption. In [9] the authors propose a cross-layer solution based on the parallel use of two LDA schemes in order to classify the loss origin on an 802.11 link and then to react accordingly. The objective of these two LDA schemes is to adapt both TCP and 802.11 MAC layer in order to reach optimized network resource usage and connection's throughput. This approach is similar to the previous one. Indeed, to be used in MANETs, it needs to interact with all wireless nodes on the path which is costly in terms of resources.

In [10] a sender-side method of end-to-end loss differentiation and adaptive segmentation (Robin) is proposed, for enhancing TCP performance in data networks. This loss differentiation algorithm enables the TCP sender to distinguish congestion losses from wireless imperfections-related losses. The objective of the segmentation algorithm is to improve the error recovery phase during a non-congestive period. Indeed, the proposed adaptive segmentation technique enables the TCP sender, if a non-congestive packet loss is detected, to retransmit smaller packets having aggregate payload equal to the payload of the lost packet. Decreasing segment size reduces the Packet Error Rate (PER) [11]. We have to note here that the proposed approach is designed to cope with networks where only the last hop is a wireless link. In spite of the good performance improvements it brings, it is obvious that this approach is hardly adaptable to MANETs where all communication links are wireless channels.

Many other approaches propose alternatives to improve TCP behavior in front of non-congestive wireless-related packet losses. However, none of these are adaptable to MANETs for almost the same reasons presented above. Among these other approaches, in [12] the authors tried to augment the basic LIMD (Linear Increase/Multiplicative Decrease) congestion control with additional mechanisms to predict the cause of packet losses and react accordingly. In [13], the authors propose a new algorithm that uses packet inter-arrival time at the receiver-side end to differentiate losses. Using simulation, they show that it works very well in a network where the last hop is wireless and is the bottleneck link. In [14] the Spike Scheme, at the receiver end, measures the one-way delays and switches accordingly between congestion state and wireless state. More precisely, if the measured delay exceeds a certain threshold, it is a congestion state; otherwise, it is a wireless state. The ZigZag Scheme presented in [15], extends the Spike scheme to include both the mean and standard deviation values of the measured one-way delays as well as the number of packet losses when computing the delay threshold. According to this calculation, the higher the number of packet losses, the wireless errors state becomes more likely the cause of data packet losses over the network. The author in [16] proposes a Non-Congestion Packet Loss Detection (NCPLD) algorithm in order to differentiate between congestion and non-congestion data packet losses. NCPLD is based on the "Knee Point" concept of the throughput-load graph at which the network reaches its optimum performance. Before the knee point, no congestion is present. On the other hand, after the knee point, queuing delay at the routers results in RTT delay increase. All these approaches suffer from the same limitation when used in MANETs: they are not designed to cope with the losses due to link losses and ad hoc route breaking. In this case, they all classify the burst packet losses as due to strong congestion and thus trigger inappropriately an important reduction of the throughput.

III. DISCUSSION

From the above approaches, we can also see that most of the work done in this domain addressing the problems of TCP within wireless environment deals with both congestion and wireless link errors. This is valid since these approaches were designed for situations where the last link is the only wireless channel connection through the communication path. In MANETs, however the LDA must also take into consideration the multi-hop nature of the wireless path between the communicating end points and address the link failure problem. Indeed, the fact that the communication path in MANETs is composed by multiple wireless links compared to networks where only the last link is wireless, commensurate the effect of link failures on the communication performances. This is especially true for TCP and all its existing variants. Link failure within MANETs introduces burst data packet losses and requires a specific reaction from TCP in order to recover from losses. Although that burst losses could result from a network congestion scenario, the reaction of TCP in front of packet losses due to link failure assuming that it's due to network congestion leads to an aggressive reaction of TCP (i.e. an important unnecessary throughput drop). Thus, our argument here is that within MANETs, we have to consider three different data packet loss causes to differentiate (not only two as discussed in previous researches):

- Wireless channel errors
- Link failure within the network
- Network congestion

Furthermore, an ideal solution will be to make such differentiation without introducing an additional overhead (i.e. notification mechanism between network nodes).

To resume, the current implementation of TCP loss recovery algorithm is congestion oriented, as indicates the name congestion control algorithm. This congestion orientation makes TCP not capable to well manage other data packet loss models. Then, TCP problem resides in its inability to recognize the main cause of data packet losses within the network. Thus, the proposed solution must cope with this problem by finding how TCP could be more intelligent to differentiate between the most common data packet loss causes within MANETs. In the mean time, TCP must be able to react accordingly using the most appropriate recovery strategy.

IV. TCP-WELCOME

TCP-WELCOME is an end-to-end, implicit, loss differentiation and recovery algorithm solution. We do not require any support from intermediate nodes within the network. End-to-end solutions differ in the measures they use to differentiate the cause of data packet losses. These measurements could be estimated at the sender side without any co-operation from the receiver side (e.g. round trip delay, RTT), while others require support from the receiver side host (e.g. one-way delay or delay variance). Our proposed algorithm will be based on RTT measures at the sender side host; this has the additional advantage to not require synchronizing clocks at both sending and receiving ends. Furthermore, in order to decrease the overhead of TCP algorithms execution and the interaction from the intermediate nodes within the network: our LDA and LRA algorithms are end-to-end sender-side modifications. Figure 1 summarizes the behavior of TCP-WELCOME that is detailed in the following sub-sections.

A. Loss Differentiation Algorithm (LDA) Rules

With respect to all the above concerns and suggestions, we need to have an adapted LDA algorithm that enables TCP to correctly distinguish the right data packet loss model within wireless ad hoc network environments. This algorithm should differentiate between the above predefined data loss models. In order to achieve our goal, our proposed solution will be based on the following idea: letting TCP make its decision at the sender-side depending on the evolution of RTT samples of sent packets. We will see in the following how RTT samples evolution can be used to help identify different causes of data packet loss.

TCP-WELCOME realizes its loss differentiation by observing: (i) the history of RTT samples evolution over the connection and (ii) the data packet loss triggers (3 Duplicate Acknowledgments, or Retransmission Time Out - RTO). The communication between MANET nodes is established via wireless channels. This is considered as an unreliable communication media type. It is affected by the weather conditions (e.g. rain), obstacles within the route between source and destination, interferences from other nodes, and many other factors. All mentioned factors may lead to unreliable data transmission over such wireless communication channels. To differentiate between wireless-related packet losses and congestion related packet loss, the idea is to look at the evolution of RTT samples. Indeed, congestion is always preceded by an increase in the RTT values. This one is mainly due to an increase in the queuing delay at intermediate nodes. So, if RTT values are stable, the packet loss is not congestion related. Furthermore, wireless-related packet losses are not bursty in nature so it is often recognized through the reception of duplicate acknowledgments and not from RTOs. On the other hand, losing a route between two communicating end points or even an intermediate link within the route between them is detected through RTO. Indeed, loosing the route leads to burst packet losses, which is similar to the case of strong congestion. In this case also it is the evolution of RTT samples that allows us to differentiate between congestion-related and link-failure-related losses. In the former case there is an increase in RTT values while in the latter they remain stable. According to this, Figure 1 summarizes the main idea of our Loss Differentiation Algorithm. It is followed by a detailed description of our Loss Differentiation Rules.

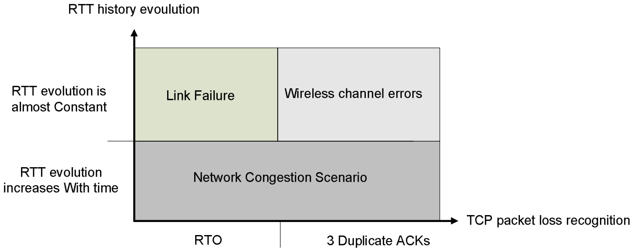

Figure 1 TCP-WELCOME Loss Differentiation Algorithm

1) *Classifying Wireless Channel related Losses*

If RTT samples evolution within the network is not highly fluctuating and stays around an average value and the data packet loss is recognized through three duplicate acknowledgements; then the data packet loss is due to wireless channel errors. Indeed, in case that there is one or multiple lossy links within the communication route between the nodes, there will be no change in both propagation and queuing delay over the connection. According to the above statement, the RTT samples over the connection should not experience high fluctuations with time. Thus, the RTT samples will stay around an average value. In addition, when there is a valid route between the source and the destination, even that there might be link errors, the source can always receive acknowledgements from the destination. The possibility of acknowledgement corruption is smaller as the size of these packets is small compared to data segments.

2) *Classifying Link Failure related Losses*

If the history of RTT samples evolution is relatively constant, and TCP recognizes data packet loss through RTO expiration; then, it is more likely that these data packet losses correspond to link failure within the route towards the destination.

We have also to precise, that in reality two situations may happen while looking to the time required by the ad hoc routing protocol in order to recover from the lost link or to find an alternative one:

- This time is less than TCP's RTO and could be unnoticeable by the sender side. In this case TCP might recognize data packet loss through duplicate ACKs. When TCP sender side checks the evolution of RTT samples over the connection, it finds that it is relatively constant. Accordingly, TCP will classify this type of loss as a wireless channel error and will react accordingly. This mistaken classification has a very limited impact. Indeed, as we will see later, this will lead to less aggressive (more appropriate) reaction compared to the one that undertook by traditional TCP while classifying the packet loss as being network congestion related.
- If the time is longer than TCP's RTO, then the sender side will recognize a packet loss through RTO. In this case, when the sender checks the history of RTT samples and finds that the evolution is almost constant, TCP will classify the data packet loss as due to link failure (which is correct) and reacts accordingly.

3) *Classifying Network Congestion related Losses*

If the history of RTT samples at the sender side is increasing gradually. Then, the loss is due to network congestion, regardless of data packet loss recognition by TCP (RTO or 3 Duplicate ACKs). In this case, queuing delay increases gradually because that the network nodes' buffers are filled with time.

B. *Loss Recovery Algorithm (LRA)*

After identifying the cause of a data packet loss using the previously proposed LDA, TCP-WELCOME have to react accordingly using the most appropriate actions in order to optimize network resources usage. These reactions concern both RTO calculations and TCP data transmission rate. Indeed, TCP-WELCOME has to adjust and update these two parameters after each data loss appropriately. One should also note that in all cases, the lost packets should be retransmitted immediately (Fast Retransmit/Fast Recovery) after identifying the packet loss.

1) *RTO Calculation Algorithm*

TCP should be able to adjust its RTO value when needed. This is realized according to the identified loss model within the network. First of all, let us note that when congestion is detected within the network, the RTO estimation is not changed and remains similar to the one used by TCP New Reno [17][1]. Alternatively, in the case of wireless channel errors, no RTO calculation or adjustment is necessary as the network conditions are supposed to be unvaried.

In the case of link failure, the RTO value has to be modified based on the characteristics (length, load, and link qualities) of the new route discovered by the routing protocol. So, after link loss recovery by the ad hoc routing protocol, we may observe that both the propagation and queuing delays change suddenly. As RTT is one of the most direct TCP connection characterization parameter that reflects network links conditions, our estimation algorithm will be depending on it. It is obvious that the number of hops as well as the load

---

[1] TCP New Reno is the best-performing TCP variant in front of packet losses related to congestions.

of the route between the TCP sender and receiver affects the RTT value over that connection. Thus, the characteristics of the new discovered route could be represented by RTT values over that route. Thus, the RTO value would be updated as follows:

$$RTO_{new} = (RTT_{new} / RTT_{old})\ RTO_{old} \quad (1)$$

where RTT_{new} is the round trip time over the new discovered route, and RTT_{old} the round trip time over the lost route before link failure. Of course these RTT values are average values calculated upon a set of segments. One should note that the old version of the RTO value will be kept until a sufficient number of RTT samples are collected to make an accurate estimation of the new RTO value.

2) *TCP Data Transmission Rate*

Estimating TCP data transmission rate is dependant on the path capacity and the queuing or buffering conditions within network nodes. In the following, we will detail how the proposed TCP-WELCOME loss recovery algorithm should adjust its data transmission rate according to the data loss model (identified by LDA). Hence, in the case of losses related to wireless-errors, there will be no data transmission rate modification of TCP as this is not needed. Conversely, in the case of a link failure within the route between the source and the destination, TCP should adjust its data transmission rate according to the recovered link characteristics. In this case, we have different opinions of how TCP could change the connection data transmission rate. First, TCP could keep the actual data transmission rate before the loss episode, and we let TCP adjust it according to its congestion control algorithm, if necessary. Second solution, TCP might decrease its data transmission rate automatically after the data loss episode by a fixed factor. For instance, we may propose to half the data transmission rate reached before loss as commonly used after a light-congestion-related packet loss. This could be considered as a conservative action of TCP. This way, we avoid having congestion over the new discovered route, in case it is more charged than the lost one. The third solution, that we envisage, is to adjust the TCP data transmission rate according to the new discovered route characteristics compared to those of the lost one. Here also the only available performance metric is the RTT value over the connection. It is thus the only metric that can be used as a reference to reflect the communication link capabilities within the network. In this case, we follow also a conservative strategy in order to prevent a network congestion episode on the links composing the new route. The congestion window (CWND) is then updated as follows:

$$CWND_{new} = (RTT_{old} / RTT_{new})\ CWND_{old} \quad (2)$$

We will show later that this third solution optimizes TCP performance in terms of both energy consumption and throughput.

Obviously, when there is a network-congestion-related packet loss over the TCP connection, TCP-WELCOME will use its normal congestion control mechanism as defined by TCP New Reno.

V. SIMULATIONS AND RESULTS

A. *Simulation Scenarios*

In order to have a wide range of results that help better understanding the behavior of TCP-WELCOME in front of different packet loss models, we compared it to different TCP variants using different simulation scenarios that describe multiple data packet loss cases. More precisely, our study uses the following data packet loss scenarios that are related to wireless ad hoc networks: (i) network congestion, (ii) data packets interferences, (iii) link failure, and (iv) signal loss. Our simulations use the Ad-hoc On-demand Distance Vector (AODV) as an ad hoc routing protocol. AODV is a reactive ad hoc routing protocol: It triggers the route discovery process only when the source has data to be transmitted toward the destination. This leads to low routing messages overhead. Due to the lack of space, the effect of different ad hoc routing protocols (both reactive and proactive) is not presented here. It will be the subject of a future publication. The scenarios are defined to be run using NS-2 [18] as a Network Simulator tool. In our simulations, all nodes communicate through identical wireless radio settings using the standard MAC 802.11 having a bandwidth of 2Mbps and a radio propagation range of 250 meters. The idea of our simulations is to study the effect of the different loss scenarios (link failure, congestion, signal loss, or interference) one by one. In this work, we preferred to study each case separately, in order to get the exact performance of TCP-WELCOME in each case and to eliminate the possibility of having multiple situations effect that may mistake or complicate the interpretation of the obtained results. The data loss scenarios defined in NS-2 are described as follows:

1) Network Congestion Scenario: In this scenario, we create a congested node at the middle of a five-node topology by generating three TCP data traffic flows that must pass by this intermediate node to reach the other communicating end. One should also note that, different levels of data congestion can be generated by controlling the number of TCP data flows crossing this particular network node at a certain time.

2) Interference between Neighboring Nodes Scenario: In this case, two TCP connections are on-going in parallel. The main TCP connection is disturbed by the interferences generated by the second TCP connection. Indeed, the node acting as forwarder for the main TCP connection is placed within the interference range of the second TCP connection sender. So, this situation creates interference and thus data packet drops.

3) Link Loss and Communication Route Changes Scenario: In this model we force TCP to change its communication path by shutting down one intermediate node between the communicating end points. In addition, we imply routes with different number of hops. Thus, each time TCP changes the communication route, the characteristics of the path between the communicating nodes change. It is obvious that the choice and the establishment delay of the new route will be dependant on the implemented ad hoc routing protocol. Packet losses and delay changes will also be implied by the link loss and the new chosen route.

We notice that the effect of such networks nodes' mobility can be represented by the link failure scenario described above as it is the most direct consequence of mobility.

4) Signal Loss Scenario: This scenario illustrates the situation where the wireless signal is not stable. The communicating nodes loose the connection due to signal loss and resume the communication when the signal comes back. Signal losses are generated by moving one of the intermediate nodes out of the radio range of its connection neighbor for a certain time (few seconds) and then moving it back.

B. Simulation Results

In this section, we describe the results obtained by TCP-WELCOME evaluation tests. It is compared to four other TCP variants (New-Reno, SACK, Vegas, and Westwood). The terms of performance evaluation are the following: the average throughput which is computed as the TCP data bytes transmitted over the total connection time, and the average energy consumption which is computed as the average transmission and reception energy consumption over the connection during the communication session and taking into consideration total data bytes received.

Table I summarizes the different TCP-WELCOME threshold variables used to evaluate its performance.

Table I. Summary of the threshold values used

Variable description	value
`RTT_SAMPLE_COUNT: RTT sample counts to take into account during the classification decision.`	10
`RTT_TRESHOLD: RTT value excess threshold (%) over average values. The values within that range can be considered within average.`	10
`RTT_G_THRESHOLD: RTT growth threshold (%) beyond which RTT values are considered as growing.`	5
`RTT_G_COUNT_THRESHOLD: Number of consecutive RTT growing values needed to trigger real congestion scenario.`	5

1) Network Congestion Scenario Results

The results depicted in Figure 3 demonstrate that, in front of network congestions, TCP-WELCOME has almost the same performance compared to the other studied variants, in terms of energy consumption (Figure 3). This is expected as TCP-WELCOME reacts in the same manner in front of congestion (as in TCP New Reno). On the other hand, this confirms that TCP-WELCOME is able to classify correctly the packet losses due to network congestions and takes the right actions to recover from the loss. Regarding the average throughput, we notice, from Figure 2, that TCP-WELCOME has a similar performance compared to best performing variants (New-Reno, SACK, and Westwood).

2) Interference Scenario Results

Figures 4 and 5, show clearly that in front of interferences, TCP-WELCOME outperforms all the other variants in terms of average throughput and energy consumption. The ability of TCP-WELCOME to classify the cause of data loss, as due to wireless channel imperfections, and not decreasing the data transmission rate improves its performances compared to other TCP variants. Indeed, the reaction of the other variants leads to decreasing their data transmission rate (halving data transmission rate in most cases) in front of some wireless-related packet losses. We notice also that TCP-WELCOME outperforms TCP Westwood, which was developed for wireless networks, and has the ability to differentiate between wireless-channel and congestion induced losses, in both terms of throughput and energy consumption. The fact that TCP-WELCOME does not decrease its data transmission rate or modify it as in TCP Westwood is the main difference between these two variants leading to our results.

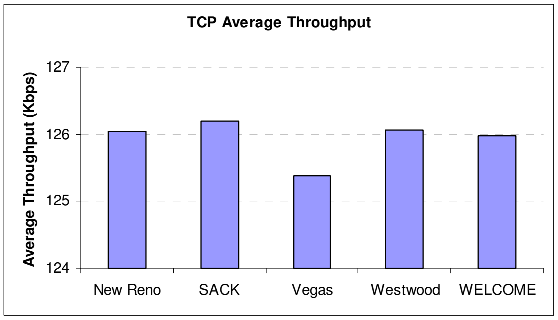

Figure 2 Average Throughput (network congestion)

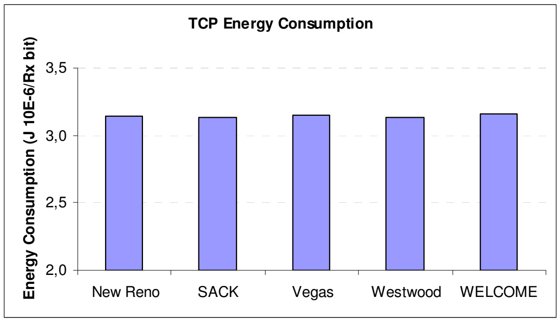

Figure 3 Energy Consumption (network congestion)

3) Link Failure Scenario Results

In MANETs, it is obvious that the nodes might have broken communication paths between the communicating end points (due to mobility or depletion of nodes' batteries). The effect of the ad hoc routing protocol chosen to be implemented within the network can be considered from two points of view:

(i) its robustness to recover from a link failure, (ii) the overhead and frequency of its routing information update messages which might result in congestion or traffic interference over the network links. For example, the overhead of ad hoc routing protocol update messages could aggravate the congestion situation over the TCP connection. This leads to more congestion control actions taken to recover from the packet loss. In our case, using AODV as an ad hoc routing protocol does not provoke any routing messages overhead problems. Also, it was found that, in this scenario, AODV was robust to recover from link failures.

Figures 6 and 7, show that the average throughput of TCP-WELCOME and its energy consumption are improved significantly compared to those of other TCP variants. The ability of TCP-WELCOME to detect that the packet losses are due to link failure and to react with the most appropriate action leads to better performance compared to all other TCP variants which react assuming that losses are due to strong congestions and decrease data transmission rate to minimum. Thus, consuming more energy and leading to low throughput (i.e. network resource usage un-efficiency). One should also note that in TCP-WELCOME, adjusting data transmission rate according to the new discovered route's characteristics helps conserving node's energy and maximizing the average throughput.

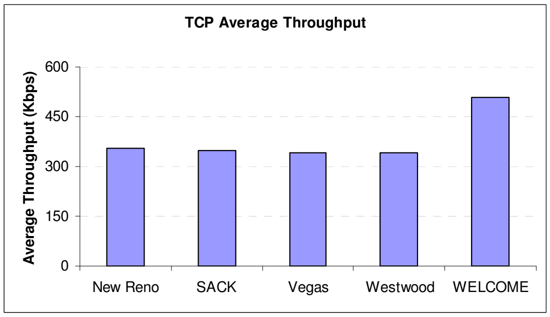

Figure 4. Average Throughput (interference)

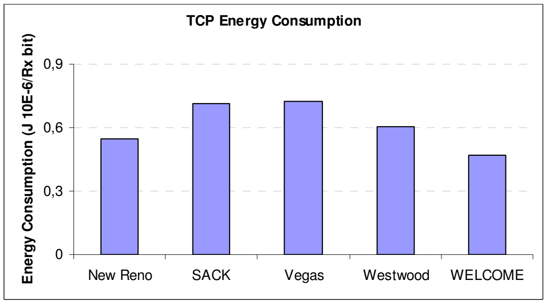

Figure 5. Energy Consumption (interference)

4) Signal Loss Scenario Results

Loosing the radio wireless signal might be considered as another reason to get disconnected from the other communicating end side. While experiencing a link loss, both nodes (sender and receiver) would search for another route to complete the session. In the case of signal loss, we consider that such alternative route is not available. We also consider that a signal loss last for a predefined period of time after which the signal is restored. After signal loss recovery, in almost all TCP variants, the sender will restart the communication session from the beginning, commencing from the Slow Start phase again. This will be the case, each time the communicating nodes get disconnected in absence of wireless signal. Inversely, TCP-WELCOME recognizes this data packet loss as link failure and reacts accordingly. While, the energy consumption of most variants are almost the same (Figure 9) except for TCP Vegas (this latter has very bad performance in this case). TCP-WELCOME outperforms the others in term of average throughput (Figure 8). Depending on the duration of the signal loss, the packet loss is detected through RTO or through 3 duplicated ACKs. In both cases, TCP-WELCOME does not decrease its data transmission rate after data packet loss (as in TCP New Reno) leading to the noticed throughput gain and to better usage of wireless channel bandwidth resources. Finally, let us notice that TCP Vegas has the least energy consumption among the others; its performance in term of average throughput is bad.

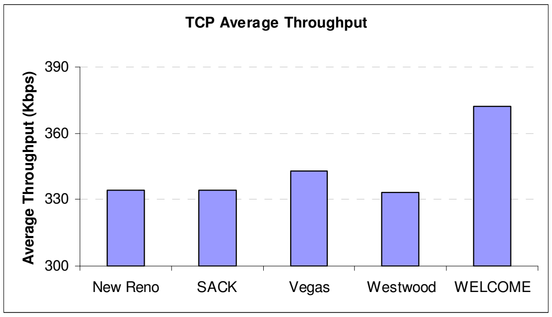

Figure 6. Average Throughput (link failure)

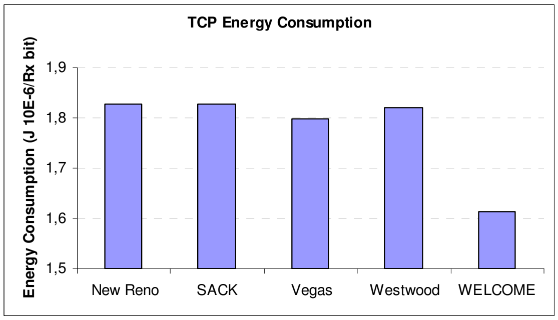

Figure 7. Energy Consumption (link failure)

VI. CONCLUSION

In this paper we proposed TCP-WELCOME, a new TCP variant that is suitable for mobile ad hoc networks. Differently to other TCP variants, it uses a Loss Differentiation Algorithm (LDA) that is able to recognize efficiently the three common packet loss causes within such network: network congestion, wireless channel errors, and link losses. In order to show the performance improvement of TCP-WELCOME we compared it to different TCP variants under different data packet loss scenarios (congestion, interference, link failure, and signal loss). This comparative study showed that both TCP average throughput and energy consumption are improved significantly. We also showed that TCP-WELCOME outperforms other TCP variants in most cases thanks to its ability to clearly classify data packet loss and takes the most appropriate actions to recover from packet losses (Loss Recovery Algorithm).

In our future work, we intend to study the performance of TCP-WELCOME and improve its behavior according to the used ad hoc routing protocol (i.e. reactive or proactive). More precisely, this will be done according to the effect of different routing protocol algorithms on TCP-WELCOME performance. In addition, analyzing the performance of TCP-WELCOME using more complex and mixed scenarios is viewed as future work.

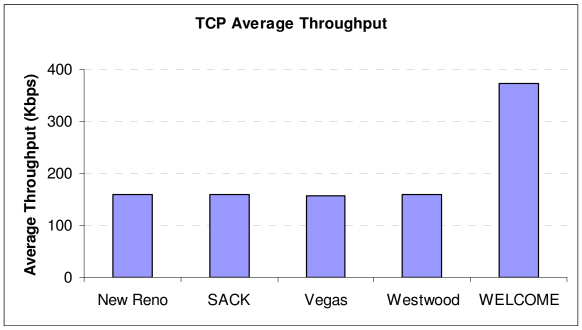

Figure 8. Average Throughput (signal loss)

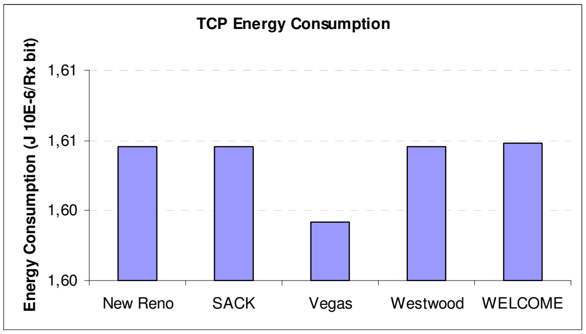

Figure 9. Energy Consumption (signal loss)

REFERENCES

[1] D. Barman and I. Matta, "Effectiveness of Loss Labeling in Improving TCP Performance in Wired/Wireless Networks"*, in Proc. 10th IEEE Int. Conf. on Network Protocols.* Boston, MA, pp. 2–11, Nov. 2002.

[2] L. A. Grieco and S. Mascolo, "TCP Westwood and Easy RED to Improve Fairness in High-Speed Networks", *In Seventh International Workshop on Protocols For High-Speed Networks (PfHSN'02).* Berlin, Germany, vol.2334, pp.130–146, April 2002.

[3] S. Morris and C. Casetti, "TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links", *In the Seventh Annual International Conference on Mobile Computing and Networking (MobiCom'01).* Rome, Italy, pp. 287-297,July 2001.

[4] R. Wang, M. Valla, M. Y. Sanadidi, B. K. F. Ng, and M. Gerla, "Efficiency/Friendliness Tradeoffs in TCP Westwood", *In Seventh IEEE Symposium on Computers and Communications (ISCC'02).* Taormina, Italy, p.304, July 01-04July 2002.

[5] S. Floyd, T. Henderson and A. Gurtov, "RFC 3782: The NewReno Modification to TCP's Fast Recovery Algorithm", Request For Comments 3782, Proposed Standard, April 2004.

[6] V. Jacobson, "Congestion Avoidance and Control", *In ACM SIGCOMM Conference*. Stanford, CA, pp. 314-329, August 1988.

[7] A. Seddik Ghaleb, Y. M. Ghamri Doudane et S. M. Senouci, "A Performance Study of TCP variants in terms of Energy Consumption and Average Goodput within a Static Ad Hoc Environment", *2nd ACM International Wireless Communications and Mobile Computing Conference (IWCMC'06).* Vancouver, Canada, pp. 503 - 508, July 3 - 6, 2006.

[8] J. Garcia, A. Brunstrom, "Transport Layer Loss Differentiation and Loss Notification", *Proceedings of First Swedish National Computer Networking Workshop (SNCNW'03).* Arladastad, September 2003.

[9] L. Stéphane, Y. Ghamri Doudane and G. Pujolle, "Cross-Layer Loss Differentiation Algorithms to improve TCP Performances in WLANs", *11th IFIP International Conference on Personal Wireless Communications (PWC'06).* Albacete, Spain, pp. 427-433, September, 2006.

[10] M. Mancuso, «A Novel Scheme of Loss Differentiation and Adaptive Segmentation to Enhance TCP Performance over Wireless Networks», Politecnico di Milano, Dept. of Electronics and Information, P.zza L. Da

[11] G. Xylomenos, G.C: Polyzos, P. Mahonen, M. Saaranen, "TCP Performance Issues over Wireless Links", IEEE Communications Magazine, Vol. 39, No. 4, pp. 52-58, April 2001.Vinci 32, 20133 Milano, ITALY.

[12] T. Kim, S. Lu and V. Bharghavan, "Improving Congestion Control Performance Through Loss Differentiation", *International Conference on Computers and Communications Networks (ICCCN'99).* Boston, MA, pp. 412-418 ,October 1999.

[13] S. Biaz and N. H. Vaidya, "Discriminating Congestion Losses from Wireless Losses using Inter-Arrival Times at the Receiver", *In IEEE Symposium on Application-specific Systems and Software Engineering & Technology (ASSET'99).* Richardson, TX, March, pp. 10–17 1999.

[14] Y. Tobe, H. Aida, Y. Tamoura, and H. Tokuda, "Detection of Congestion Signals from Relative One-Way Delay", *IPSJ Journal*, 42 (12), pp. 3222-3230, 2001.

[15] S. Cen, P. C. Cosman, and G. M. Voelker, "End-to-End Differentiation of Congestion and Wireless Losses", *In SPIE Multimedia Computing and Networking (MMCN'02).* San Jose, CA, January 2002.

[16] N.K.G. Samaraweera, "Non-Congestion Packet Loss Detection for TCP Error Recovery using Wireless Links", *In IEE Proceedings of Communications, volume 146 (4)*, pages 222-230, August 1999.

[17] J. Hoe., "Start-up Dynamics of TCP's Congestion Control and Avoidance Scheme". *Master's thesis*, MIT, June 1995.

[18] Network Simulator, NS-2. Available at www.isi.edu/nsnam/ns/