Middle-East Journal of Scientific Research 21 (10): 1787-1801, 2014
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2014.21.10.21743

The Introduction of a Cross-Layer Approach to Increase the Efficiency of TCP in Mobile AD HOC Networks by the Physical Cross-Layer Interactions and Network Layer and its Simulation by NS-2 Network Simulator

[1]Saman Afrasiabi, [2]Mehdi Jafari Shahbaz Zadeh, [3]Hossein Dorosti and [4]Ramin Tavakoli

[1]Department of Computer Engineering, Islamic Azad University, Kerman Branch, Iran
[2]Department of Electronic Engineering, Islamic Azad University, Kerman Branch, Iran
[3]Department of Software Computer, Islamic Azad University, Kerman Branch, Iran
[4]Department of Hardware Computer, Islamic Azad University, Kerman Branch, Iran

Abstract: Congestion control is one of the classic issues in network, various strategies are presented until now and one of the main duties of Transport Control Protocol (TCP) is dealing with this issue. Today, with the advancement of computer networks of wired models to wireless fields, the functions are various protocols of the network are more consistent with these environments. One of the networks is Mobile Ad hoc network (MANET). The current study aimed to used cross-layer interactions and by the collaboration of physical cross-layer and network layer attempts to create an environment on which TCP can work well.

Key words: Computer network · Network simulator NS-2 · Transport Control Protocol · Physical Layer · Network Layer · Mobile Ad hoc network (MANET)

INTRODUCTION

Wireless networks were developed due to presenting various applications and good services. These networks are increasing rapidly and their services are improved. An infrastructure less network or mobile network is a model of wireless networks in which there is no fixed structure and it is only consisting of mobile nodes and no wired connection. Each mobile node acts as both host and mediating node in a wireless connection. In these networks, the nodes are mobile and they have a dynamic topology and some phenomena as route failure and packet losses are occurred more.

Nodes mobility and energy constraint of the nodes are the issues creating some limitations in the networks. TCP is created based on wired model and most of the new characteristics of wireless models such as nodes mobility, high error rate in the environments, signal attenuation, noise, etc are not considered. Thus, working on these networks, we are faced with many problems including misinterpretation of packet losses and unsuitable use of congestion control mechanism. It is required to present the mechanisms to adapt TCP with the Mobile Ad hoc network that the protocol with the view of the network can have acceptable function. In recent years, many studies are carried out on this model of the networks and wireless networks. Some of the researches considered TCP itself and by improving it attempted to cope with the problems of the networks. Some others dealt with the lower layers and by taking cross-layer mechanisms and lower layers information had better view to TCP that this protocol can act in more consistent way based on the model applied in the lower layers of the network. TCP considers any error or losses due to the congestion in the network and attempts to apply congestion control mechanism. In Mobile Ad hoc network, other reasons except congestion can cause error and packets losses and a good behavior should be applied to deal with each special error and calling the congestion control blindly can drop the efficiency of the network [1].

Corresponding Author: Saman Afrasiabi, Department of Computer Engineering, Islamic Azad University, Kerman Branch, Iran.

The method in this paper is such that by defining new cross-layer collaboration among the existing network layers, some of the problems of Mobile Ad hoc network are managed and a suitable environment is provided for TCP. In this method, the network layer applies the existing information in physical layer and the information of signal power to manage these problems [2-9].

The current study is organized as following: The second section deals with a general review of Mobile Ad hoc network and TCP and congestion control mechanism. The third section presents a complete review of the presented methods about congestion control and improving the efficiency of TCP in Mobile Ad hoc network. The proposed method to improve the performance of TCP and adaptation of this protocol in Mobile Ad hoc network is elaborated in the section 4 of the current study. Section 5 deals with the simulation of the proposed approach by NS2 software. All the issues of simulation including the applied scenario, the parameters are explained and the section 6 is dedicated to conclusion[5].

A Review of Mobile AD HOC Network and the Nature of TCP

Mobile AD HOC Networks: As the name reveals, the difference between wired and wireless networks is such that the nodes can be connected without wired infrastructure. Today, various technologies including Radio media, infrared and laser can be widely applied in wireless communication. For example, Radio media is used in Mobile Ad hoc network because they are able to be connected without being informed of the place of other nodes [10-15].

Wireless networks are consisting of many models ranging from wireless LAN (Local area network) to cell and Mobile Ad hoc networks. The classifications are done based on the fact that the nodes are fixed in the network or they are mobile or with the existence and the lack of fixed infrastructures. The model in the current study is Mobile Ad hoc network.

A MANET consists of mobile platforms as a router with multiple hosts and wireless communications devices herein simply referred to as "nodes “which are free to move about arbitrarily. The nodes maybe located in or on airplanes, ships, trucks, cars, perhaps even on people or very small devices. A MANET is an autonomous system of mobile nodes. The system may operate alone, or may have gateways to and interface with a fixed network.

MANETs have several main characteristics [16]:

- Dynamic topologies
- Bandwidth-constrained
- Energy-constrained operation
- Limited physical security

In addition, some special networks (e.g. mobile military networks or highway networks) may be relatively large (e.g. tens or hundreds of nodes per routing area). The need for scalability is evident in MANETS. However, in light of the preceding characteristics, the mechanisms required to achieve scalability likely are [17].

A Review of Transport Control Protocol (TCP): Today, TCP is one of the most applied transport layer protocol. TCP is located in the fourth reference layer of OSI (Open layering system interaction). This protocol is designed reliable to deliver end-to-end data. OSI can be independent from infrastructure. Generally, the main services of TCP are as following:

- Connection oriented: Establishing a logical connection between the users, maintaining the connection is to the end of data transport. The connection should be made between the source and destination before sending each data.
- Full-duplex: It means the data can be flowing in each direction independently of the other direction)
- Stream-oriented: Transport unit of TCP manages the currents and acts as a media between IP layer and applied processes. A TCP connection transfers a flow of bites not a sequence of the messages. It means that the boundary of the messaged exchanged between two points is not maintained.
- Reliability: Two factors of Sequence Number and ACK number cause a reliable connection in TCP.
- Supporting the concept of end to end: It means that a connection is made between the final source and destination to send data.
- Using flow control mechanism: It is related to flow control of point to point traffic between the sender and receiver. In flow control the direct reports and feedbacks are sent from the sender to the receiver to show how to do the works.
- Using congestion control mechanism: Congestion is a public case and is resulting from the behavior of all host machines.

TCP and its Performance Are Explained in the Following: TCP is a connection-oriented service; it means that a connection should be made between the source and destination before sending any data. To guarantee the reliable transport of the data, TCP should send an Acknowledgement (ACK) for the received segments. There is a unique Sequence Number for each bite of data and the receiver after send an ACK. TCP ACKs are cumulative; it means that each ACK confirms the receiving of all the bites to the given Sequence Number. ACKs are sent only in response to the received packets and in regular interval. If no packet is sent, no ACK is sent. The cumulative nature of the ACKs is effective because we are not forced to send an ACK for each packet received. Although these kinds of ACKs are ambiguous, because they can not explicitly inform the sender of the lost or damaged packets. The sender can not conclude based on acknowledged bites that whether all the sent data are received or not? ACKs of TCP can be a separate packet or piggyback on a data packet. At normal mode, TCP doesn't send ACK for each received segment immediately and waits a little and if a data segment was sent, ACK piggyback on it. The receiver can wait until another data segment enters and then send ACK both received segments together. TCP shouldn't delay ACKs more than 0.5s and it should send a ACK for each new received segment [18].

Congestion Control Mechanisms TCP: When total received data exceeds the network capacity, congestion is occurred. In congestion, the queues in router are filled and the packets passing from the routers inflict man delays. If the congestion continues again, the size of the queue gets greater than the buffer size and buffer overflow is occurred. The packets can not be located in a buffer, are discarded by router and they should be re-transmitted by the source. In many cases, it is seen that the packets passing successfully from the congested router are re-sent by the source because the source assumed that the packets are deleted due to congestion. Re-transmission of the packets wastes precious network bandwidth [19]. In addition, the packets reaching the destination by re-transmission are delayed. Congestion leads into considerable drop in operation power and increase of delay in packets transport. It is very bad and it should be avoided [20].

Figure 1 shows delay changes and operation power against the increase of the network load. When the load is low, any increase improves the operation power without increasing the delays considerably. When the network load reaches its capacity, the more the load, the more the delay but the operation power doesn't change. The point on which the load equals to the capacity is called the knee. The point where the throughput falls is called the cliff.

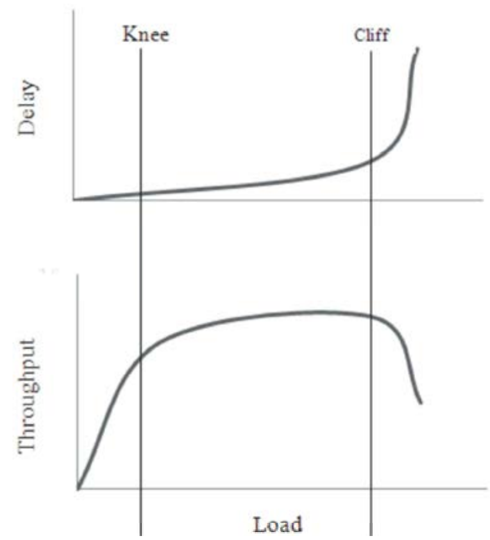

Fig. 1: The network load and its effect on operation power and delays

Finding the network capacity is not easy. Protocol strategy is such that immediately after the congestion, it is controlled. It prevents the congestion in the first place. Indeed, TCP increases the network load repeatedly to find the critical pint in which congestion is occurred. Then it backs off from this point. In other words, to find the existing bandwidth by TCP, some packets should be lost. This strategy is called congestion control. Another method is called congestion avoidance. This method predicts the congestion occurrence time and then by reducing the transport rate, before the packets are discarded avoids the congestion [21-25].

The aim of congestion avoidance is to keep system performance at point Knee. In this point, the delays are minimum and there is an idealized operation power in the system. While congestion control aim is to keep the system performance in the left side of cliff. TCP congestion control is including some mechanisms to provide the sender to estimate the network capacity immediately and reach steady state and identify congestion in the network and respond to it and finally recover the errors of congestion. These mechanisms are elaborated in the following. Each send creates two windows: The first window is the advertised window that the receiver is ready to accept, the number of the packets their ACK is received. The second window is congestion window specifies the amount of data a sender can transmit to a receiver without receiving any ACK. Congestion window is determined by the sender and based on feedback of the network. Each of the windows defines the number of the bites the sender can send. Thus, the effective size of the window is the minimum amount of what the sender thinks is correct and what the receiver thinks it should be. For example, if the receiver states to send 8 Kbyte but the sender knows that sending more than 4 Kbyte make the blockage, it sends 4 Kbyte. However, if the receiver state to send 8 Kbyte and sender knows that sending to 34 Kbyte is allowable, only the requested 8 Kbyte is sent. When there is a connection, TCP sender enters slow start phase and the sender sends a packet to the receiver. If ACK of the segment is received before the time out, then the congestion window size is increased with each received ACK. Thus, the congestion window power is increased due to the double size of the window in each RTT. The numbers of unacknowledged send packets determine the size of congestion window until the window size reaches the threshold of slow start [22-30].

Then we enters congestion avoidance phase and in this phase the window growth is linear. The sender probes to obtain more bandwidth linearly until a packet lost is observed. Then the sender infers that congestion is occurred and the size of congestion window is reduced. The reduction of congestion window size is related to packet loss [3]. It means that:

- If packet loss is said by three duplicate Ack, there is a considerable reduction for the window. Thus, the window reduces to its current size and the connection enters congestion avoidance phase.
- If the packet loss is advertised again due to time out, the window size is reduced as one and the connection enters slow start phase again.

Challenges for TCP in Wireless AD HOC Networks: Despite the fact that wireless networks have many capabilities including the movement for users, service quality of the current wireless networks is not idealized. Interruption is possible when the mobile users move between the cells. Wireless connections suffer from high bit rate, low bandwidth and long delays. Most of the local wireless networks reach a part of the maximum bandwidth. To find the source of the problem, we should define the differences of wired and wireless media. In addition to the various forms of wireless networks, wireless media is completely different from the wired media. Generally, the characteristic of wireless media is more sensitive than wired media.

First, the limited bandwidth of the wireless media is shared among many users and overall Throughput is much less than wired media.

Second, shared media of wireless network. The competition for shared media causes the contention among the rivals. If this is not done appropriately, the contention can lead into low Throughput and unpredicted delays in access to the channel. The third issue is such that wireless media is susceptible to transport errors. Varied conditions in surrounding environment, multiple path reflections, noise, wave interference and fluctuations of channel all affect the transport signal and lead into high bit error [31-33].

According to the problems above, the existing challenges in the networks leading into the reduction of performance are:

- Lossy channels
- Hidden and exposed stations
- Path asymmetry
- Network partitions
- Route failures
- Power constraints

As it was said TCP in mobile ad hoc networks showed bad behavior model and lack the strategies for energy economy to reach high good put. The lost piece of TCP led into the negative effects on wireless environments and it is detection of error. TCP can not detect the error nature and it states only the error occurrence. In other words, TCP says that a packet is lost. Thus, the mechanism of error recovery is not efficient namely when the error pattern is changed regularly. Because losing the packet in according to TCP is due to congestion. For example when random, rare or serious or short errors are occurred, the sender backs off and increases the size of the window that is reduced. During slow start of the window, it has many opportunities for error-free transfer and it is lost and data exchange time is increased. In other words, in case of sudden and temporary errors, the strategy of back off of TCP instead of partial re-transmission, the size of sending window and timeout are changed and this leads into considerable drop in goodput and increase of total connection time. When the error is occurred and TCP backs off, it tries to do transport operation via the shrink window. Regarding the continuous errors like channel fluctuation and repetitious errors of this behavior is not useful in energy saving. It improves good put against the high cost for the energy of the transport.

The main problem is the inability of TCP in correct recognition of error nature and it can not respond the error well [9, 10]. In addition, this protocol can not monitor the various conditions of the network effectively to regulate the speed, window size with these conditions. On the other hand, TCP to identify the congestion applies losing or discarding the packets during the transport time. Thus, a part of time and energy is dedicated to re-transmission of the packets and this leads into the considerable drop of efficiency. The traditional strategy to control congestion was such that the lower adjustment of the congestion window was used during re-transmission being modeled of standard TCP and this method is not accountable in mobile and wireless networks. Complementing operation besides regulating the congestion window is to control time out. Increase of time out causes that TCP can not detect the error-free conditions immediately and rapid recovery is not made. In other words, AIMD [11] when the packets losses is due to the errors except the low bandwidth to meet the demands of flow, it is not useful for fairness or stability.

As TCP is not designed for a special network or applied program, we can make some modifications in the mechanisms to increase its efficiency. TCP behavior in wired networks when the major cause of being damaged or packets loss was congestion, it was investigated for the first time by Jacobson [12].

Recently, TCP behavior is considered in wireless and satellite networks. The researches show that TCP transport rate is reduced during random/serious errors and long delays. Thus, some of the researchers attempted to propose the structures that besides improving the protocol performance in the special networks have the least change in the protocol itself. Thus, most of the propositions attempted to improve the network efficiency or control congestion via dealing with the network tools important in protocol activities.

The Introduction of the Presented Approaches to Control Congestion in Mobile AD HOC Networks: In this section, some of the presented methods for congestion control and improving the efficiency of TCP in mobile ad hoc networks are investigated. These methods in high level can be classified into layer methods and cross-layer methods. The layer methods were the most common method in networks architecture since 1950s. This attitude was considered by most of the researchers and engineers about telecommunication and computer networks. In recent years, with the development of mobile and wireless networks, layer strategies lost their efficiency on these networks. To do this, cross-layer methods were applied. In recent years, considerable studies were performed on cross-layer methods in academic and research centers.

Cross-layer methods were taken into attention in 1960s by introducing TCP/IP network model of layer architecture [8]. In next year, due to development of networks application cases, OSI networking model was developed in 1980s. By the development of local networks, layer methods were considered more.

Approximately layer models were applied for some decades. Wan Newman architecture for computer systems or layer architecture for internet, etc is good examples of layer methods. Cross-layer methods, using wireless mobile networks were developed. In considerable applications including using internet wireless and mobile devices were applied. By the development of wireless networks, the researchers found that using layer models in these networks is not good because in layer models, it is assumed that each problem is arising from a special layer. On the other hand, many researchers are carried out providing that in wireless and mobile networks considerable reduction of efficiency due to Multi hops, nodes mobility and media is not resulting from a special layer and two or more layers reduce the efficiency in these networks.

Thus, the interactions of two or more layers are considered to improve the general efficiency of wireless and mobile networks. Even in recent years, the researchers for applying security and saving in energy, applied cross-layer methods.

Layer architecture is powerful in terms of design but as it doesn't have high flexibility, it doesn't achieve good efficiency in dynamic environments. Thus, cross-layer architecture in wireless networks and mobile ad hock networks has better efficiency.

In wired networks, a problem is arising from one or two reasons. On the other h and, the existing problems in wireless and mobile networks is not arising from a special factor and some reasons create some problems in wireless and mobile networks. In these networks, there are many problems compared to wired networks.

In wired networks, the main reason of packet loss in TCP is congestion. Thus, using congestion control mechanism during packet losses improves the performance of TCP. While in wireless networks, various reasons including handoff and channel fading, route failures and disconnections cause packet loss. We can not conclude that during the problem in TCP, using congestion control mechanism increases the efficiency of wireless and mobile networks [6, 7, 8]. In this section some of the existing methods to improve efficiency of TCP in mobile ad hoc networks are presented. Only the main characteristics of the methods are dealt and the full details are not mentioned. The methods of improving TCP performance on ad hoc networks is classified into two groups:

- Layered methods
- Cross-layer methods

In cross-layer methods, some layers work beside each other. But in layered methods, one layer alone improves the condition. On the other hand, we can say that the reason of reduction of TCP efficiency in mobile ad hoc networks is as following:

- The lack of ability of distinguishing the packet losses due to congestion and due to other factors as route failure and nodes mobility.
- Reduction of the efficiency of TCP due to route failures, nodes mobility rate, channel fading and signal attenuation, etc.

Generally, cross-layer methods are classified into four groups and layer methods are classified into three groups. Layered methods act by changing one of the following layers:

- TCP layer
- IP layer
- Link layer

On the other hand, cross-layer methods achieve the improve of TCP performance of the interaction between the following layers with each other:

- TCP and network
- TCP and link
- TCP and physical
- Network and physical

A brief explanation of the proposed methods is given in the following and we try to present a full detail of each one.

TCP and Network Interaction Based Methods

TCP-F (TCP Feedback): TCP-F [13] is feedback-based strategy to manage route failures in mobile ad hoc networks. This strategy provides the sender of TCP to distinguish between the losses due to route failure and losses due to network congestion. When the node routing factor detects a route failure, it implicitly sends a Route Failure Notification (RFN) packet to the source. On receiving the RFN, the source goes into a snooze state. The sender of TCP in snooze state completely stops sending further packets and freezes all of its variables including timer and congestion window size.

The sender TCP remains in this snooze state until it is notified of the restoration of the route through a Route Re-establishment Notification (RRN) packet. As soon as the source receives the RRN, it changes to an active state from the snooze state and resumes the transmission based on the stored sender window and timeout values. After receiving RRN, the sender tries to leave the snooze state and resume the sending based on old values of sender window and time out. We can say that the sender to avoid blockage in snooze state after receiving RFN, starts rout failure time. When the time is ended, the congestion control algorithm works normally. The author's report that by F-TCP improvement is observed in TCP. The simulated scenario is general and is not based on ad hoc networks.

ELFN (Explicit Link Failure Notification): ELFN [14] is similar to TCP-F. Despite TCP-F the evaluation of the proposition is based on a real interaction between routing protocol and TCP. This interaction helps during route failure to the knowledge of TCP factor about route failures. The authors implement the ELFN scheme that is modified to piggy-back the route failure message is sent to routing protocol to the sender. ELFN is similar to a 'host unreachable' ICMP (internet control message protocol) message for notification. The ELFN message contains the sender and receiver addresses and port numbers as well as the TCP segment's sequence number. When the TCP sender receives an ELFN message, it enters a 'standby' mode by disabling its retransmission timers. To gain information about the route re-establishment in the ELFN scheme, the sender sends a probe packet periodically in 'standby' mode. On arrival of

ACK for the probe packets, the sender breaks out of the 'standby' mode restoring its timers and continues normal operation. In this method, TCP is interacting with routing protocol to detect the route errors and show the good reaction to cope with it. This method is evaluated only under DSR routing protocol as the problem of invalid routes can decrease the efficiency of this method. In addition, the length of interval between the packets that are sent for probing the route reconstruction and the type of the packets are evaluated and the interval is of great importance and if the length of interval depends upon RTT, is better compared to the time fixed value is considered.

AD HOC TCP (ATCP): Despite the methods that are explained until now, ATCP [15] doesn't try to change in standard TCP and a layer is inserted between the network layer and transport layers to show its efficiency well and interact with the machines not using ATCP using TCP. This method is based on ICMP, ECN protocol to detect the disconnection and congestion problems. This layer acts as when there is no need to use congestion control of TCP, this is avoided and it is done as following.

This method applies network layer feedback. In addition in route failures, ACTP is dealt with high bit error rate.

TCP sender can enter persist, congestion control or re-transmission mode. A layer is between TCP, IP in source nodes of TCP and it is called ATCP. ATCP monitors the network state information provided by (internet control message protocol) message for notification 'Destination unreachable' messages and ECN messages. Then, ATCP puts TCP in a good condition. On receiving a third duplicate ACK and it shows lossy channel and puts TCP into 'persist' mode.

TCP by receiving 'destination unreachable' message TCP into 'persist' mode. TCP factor is fixed in this condition and when a new route is found by probing the network, no packet is sent. ECN is used as a explicit sender notification about network congestion during the application of the route. By receiving ECN, applies congestion control normally without waiting for TCP time out. This model was implemented as pilot and was evaluated under some conditions as congestion, loss channel, partitioning and unarranged packets.

TCP-Buffering Capability and Sequence Information (TCP-BUS): TCP-buffering capability and sequence information (TCP-BuS) [16] applies network feedback about route failure as the previous strategies and do the appropriate reaction to it. The new method is the introduction of buffering capability in mobile nodes.

Associatively Based Routing protocol (ABR) [17] is applied. In this method some new techniques to develop TCP are as following:

Explicit route notification: To inform the source about route failure, an explicit route disconnection notification (ERDN) message is generated and the second is explicit route successful notification (ERSN). Receiving ERDN, the node detecting the route failure (pivoting node) stops the send source. Similarly after the re-establishment of the route, by local Query (LQ), pivoting node sends ERSN message.

Extending timeout values: During the route recovery process RRC, the packets are buffered along the path from the source to the pivoting node Therefore, it is necessary to increase transmission timeout values to avoid timeout events. For ease of implementation, the proposed scheme just doubles the timeout values. This study has a major role in buffering techniques and reliable sending of control messages.

Split TCP: The methods mentioned were mostly general methods not dedicated to a special network model. The methods proposed in this section [18] are considered for developing TCP in mobile ad hoc networks. We know that the main reason of route failure in mobile ad hock networks is mobility. In addition, by the increased of hopes on a path, route failure probability is increased. This increases packet losses. Thus, TCP connections being composed of many hopes suffer from considerable route error being created by nodes mobility. This method is based on dividing long TCP connections into small segments to reduce the number of route failure frequencies. In other words, to improve the efficiency of the connections and to solve the problem of unfairness, the separated TCP connection method is proposed to divide the great TCP connections to smaller local segments. The mediating node between two local segments is called proxy. Proxy takes TCP packets and buffers them and send Ack message to the sender or previous proxy and this is called lack. In addition, proxy is responsible to deliver the packets with a good rate to the next local segment. By receiving lack, the proxy removes the packets from its buffer. To provide the reliability of source to destination, sending the message

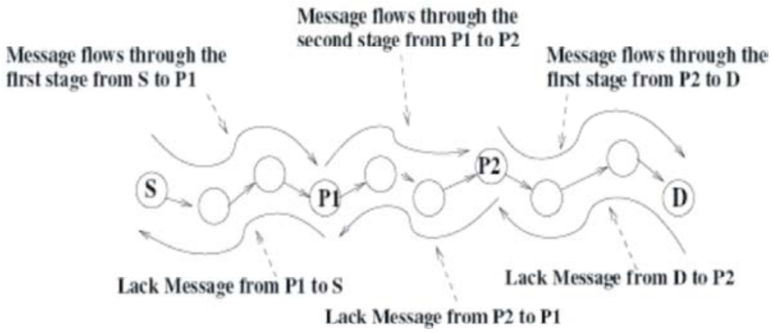

Fig. 2: TCP split

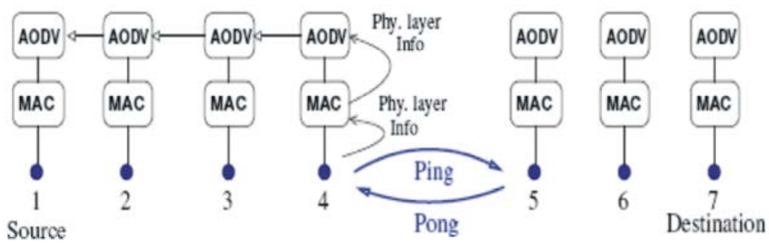

Fig. 3: Network and physic among cross-layer

to destination and sending Ack from destination to source is like standard TCP. Indeed, this method divides transport layer functionality to end to end transport capability and congestion control and this is done by two transport windows in source and one is transport window and another one is end to end window. Congestion window is a sub window of end to end window. While congestion window changes based on the rate of lack messages received of next proxy, the end to end window is changed based on lack message rate sent by the destination. In each proxy, there is a congestion window controlling the message sending rate between the proxies.

The simulations showed that the distance 3 to 5 between the proxies can have good effect on efficiency and fairness. By separate TCPs, the improvement is obtained equal to 30% in total efficiency.

The methods based on physical layer and network layer interaction The cross-layer propositions between the physical and network layers to improve the performance of TCP are classified into two groups:

- Pre-emptive routing [19]
- Signal strength-based link management [20]
- Each of the methods is explained briefly.

Pre-Emptive Routing: It is possible that TCP due to continuous route failure suffers from long idle periods. This method attempts to solve this problem to reduce the frequency of route failure and reduction of re-establishment of the route time by switching to a new route when we expect that a current route link is broken in future. This technique is combined with AODV [30], DSR [21].

This failure detection mechanism is based on energy. When a middle node along the route detects that energy signal of a packet sent from upstream node is less than a threshold level called pre-emptive threshold level. Then, the middle node detects a route failure. For example, in Figure 3, when node 4 feels that energy signal of a packet received of node 5 is less than a pre-emptive threshold and then node 4 detects a routing failure event. After the detection of the event, node 4 notifies the source. The source routing factor proactively searches a new route. If the new route was existing, then the routing factor is switched to the new route. The pre-emptive threshold value is very important.

Signal Strength-Based Link Management: This algorithm is similar to the previous method with the difference that in this algorithm, each node keeps a record of signal strength received of mono-hope neighbouring nodes. By these records, routing protocol predicts link failure event in the near future. This prediction is called proactive link management. After the detection of this event, source routing factor notifies by going down message. In Figure 3, by receiving this message, the source routing factor stops sending the packets and gives initial values to route detection procedure. This mechanism can increase sending capability to re-establish a broken link. Recently, link management as reactively is proposed. Proactive and reactive mechanisms can be combined with each other. Node routing factor by predicting that a link is out of order informs the source to stop sending. Then, this node increases its sending power to management the passing packets of this link.

Improvement Methods Based on Tcp Layer: As it was said before, in addition to the cross-layer methods, some of the methods by focusing on a special layer try to improve the performance of networks function in various models. These methods are called layered methods. To improve the performance of TCP, various studies are performed on the layer itself to adapt with wireless and mobile environment. Some of the methods are investigated as following.

Fixed RTO: Fixed RTO technique [22] is sender-based not relying on network feedback. Indeed, the authors applied an innovative method to detect route failures and congestion. In standard TCP, for the unacknowledged packet, power backward algorithm is applied. In Fixed RTO for unacknowledged packets, retransmission is done. But RTO is not doubled (constant) until the route is re-established and the created packet gets Ack. In [22], the authors reported that when Fixed RTO with on-demand routing protocols is used, considerable improvements are made. This method is based on the fact that route errors should be recovered by a rapid method by routing algorithm. Thus, any disconnection doesn't lead into using standard back ward mechanism as using this mechanism can create more unnecessary delay in recovery.

In this method, when two consecutive transmissions are faced with time out, this is considered as route error and TCP sender is allowed to do re-transmission in regular interval rather than at a time with exponential increase rate. Indeed, the TCP sender doubles RTO time only once. If the lost packet doesn't arrive before the end of the second time out, then the packet is re-transmitted repeatedly but RTO will not be increased and it will be fixed until the route is reconstructed and Ask is sent by receiving the packet.

TCP Door (Tcp Detection of Out-of-Order and Response): TCP DOOR [23] is an end-to-end approach. This approach doesn't need the collaboration of middle nodes and it is based on out of order delivery event. Out of order event (OOO) is interpreted as a sign of route failure. There are two mechanisms of TCP DOOR:

- At the sender ends: the non-reducing properties of ACK sequence number are used. Two duplicate ACKs have the same contents. Thus, the send needs the additional information to detect OOO event. The additional information adding a one byte TCP option to the duplicate ACKs called ACK duplication sequence number (ADSN). While sending a duplicate ACK for the same sequence number will trigger an increment in the ADSN number.
- At the receiver end, we need a two-byte TCP option to detect OOO called TCP packet sequence number (TPSN). If the initial value of the TPSN is zero, then it will be incremented with each data packet sent, including retransmitted data packets. If OOO is detected by the receiver, then the receiver can notify by a special option bit (OOO bit) in ACK packet header to the receiver. If TCP receiver detects OOO event, the following reactions are done:

- The sender disables the congestion control temporarily
- Instant recovery during congestion avoidance

The authors propose to use receive based mechanism as it doesn't need send to receiver warning.

It should be said that a layered approach in the network layer is presented called Backup path routing [24] to improve the performance of TCP in mobile environments. This helps the improvement of path availability in TCP by multi-path routing. The authors found that the main multi-path routing for the following reasons worsen the performance of TCP:

- The incorrect size of RTT mean
- Delivery of out of order packets due to multiple paths

Thus, they presented a new version of multiple- path called Backup path routing. In this routing, some paths are kept from source to destination. But only one path is used in each moment. When in current failure path we can switch to an alternative path.

On the other hand, to solve the interference problem in wireless channel, the followings are proposed:

- TCP layer-based solution: The solution called Dynamic delayed Ack [25] is proposed.
- Network layer-based solution: The solution in this layer is called Contention-based Path Selection (COPAS) [26].
- Link layer based solution: There are two solutions in this layer called Link RED [27], Adaptive pacing in [27, 28].

Presenting the Proposed Approach: By the advancement of the mobile networks, this technology was rapidly applied in various applications of computer. This led into the development of new models as mobile calculations, comprehensive computation, etc. As the nature of wireless networks is different form wired networks we should do some optimization in the protocols of these networks. For example, we know that about 80% of the traffic of the network is dedicated to TCP protocol and most of the upper layers protocols apply TCP as a good tool to transfer data. The users using wireless networks expect to receive the services similar to wired network services. Fulfilling the expectations created some challenges in the model of networks and this is arising from the nature of wireless networks. In theoretical case, TCP is designed as it acts independent from the lower layer but when we work with this protocol, we can find that the technology of network infrastructure is very important and some of the main functions are based on the assumptions that this protocol have from the lower layers and this protocol is created based on the existing assumptions in wired networks.

Attenuation and fluctuation of signal strength is one of the factors degrading the performance of TCP in mobile ad hoc networks. The nodes mobility and channel conditions cause signal strength fluctuations. This problem leads into packets losses in mobile networks. As we know, to cope with the problem of packets loss, TCP applies congestion control mechanism in wired networks. As TCP is independent from infrastructure, thus in mobile ad hoc networks can use congestion control mechanism to avoid more packets losses. Due to inherent differences in wireless media to wired media, at this time congestion control mechanism is not involved. Because it is possible that the incoming signal attenuation condition returns to initial state due to the mobility of neighboring nodes and there is no need to use congestion control mechanism. Thus, during signal attenuation, we should treat well because unduly uses of congestion control mechanism have considerable effect on reduction of TCP efficiency in mobile ad hoc networks. Unduly use of congestion control leads into considerable reduction of TCP performance.

The proposed method in this report is similar to the approach presented in [19] benefits from the interactions between network layer and physical layer and based on evaluations on incoming signal strength in the nodes can keep the path or create new path in case of interrupting the previous path.

In this approach, to improve network performance and based on the mobile nature of mobile ad hoc networks and energy resources are limited we use the point that it is possible that the node that is in the path is out of order based on the network dynamics or its energy is consumed and this leads into route failure. We consider two mechanisms, at first we define a threshold value for incoming signals and when the incoming signal is less than the threshold value, route failure is possible. As it is possible that one of the existing nodes is out of order due to mobility or its signal node is attenuated due to low energy level. To do this we define a threshold limit in the nodes and when a node received a signal less than threshold value, one of above options is possible. Thus, the physical layer immediately sends a message regarding the attenuation of signal to network layer. The network layer by receiving the first message of signal attenuation asks for its counter with a data value and physical layer to double the transmission strength because it is possible that signal attenuation is temporary and it is removed after a while. To be more ensured, the network layer creates a new path on the network layer and stores it in the buffer. If for the next time, the network layer receives a message regarding the signal attenuation and by checking the counter found that signal attenuation is occurred for more than once. The network layer replaces the backup path with the current path. Because based on the existing conditions, path disconnection is possible.

In another case, to understand that a node can finish its battery, a minimum value can be defined for energy level in the nodes. When a node in the path is less than threshold limit, it is possible to be turned off and route failure is occurred. Thus, by notification of the threshold level when the node found that its battery is empty, it can notify the network layer immediately and the network layer understands that the current path is instable and it will be disconnected. Thus, immediately a new path is found without considering the node being turned off and replaces it with the existing path. This avoids the sudden interruption of path and packets losses.

Table 1: General parameters of simulation

Parameter	Range
The number of nodes	5 to 15
Bandwidth	Mbps 2
Transport region	200 m
Packet size	Bytes 1460
Queue model	IFQ
Link layer protocol	ieee 82.11
Network layer protocol	AODV

The network and physical layers by presenting an interactive approach provide the conditions on which most of the errors and problems in lower layers of the network are hidden from the view of TCP and this protocol is faced with the environment as wired environments. It can be said that in network layer, we can use on-demand routing protocols as AODV [30] and DSR [21].

The Evaluation of the Efficiency of the Proposed Approach: In this section, we calculated the efficiency of the proposed approach and compared it with the famous models of TCP as TCP Newreno [31] and TCP Vegas [32]. For this operation, we can use various models of network design. Simulation operation was done by NS2 simulator [33]. For transport layer protocol in the proposed model we applied TCP Reno [12] and in the network layer we apply the on-demand protocol of AODV [30]. Other general parameters of the problem are shown in Table 1.

Simulation 1- the Investigation of the Changes in Congestion Window Size: In this section we deal with the changes in congestion window size (CWND) in proposed approach and we consider the cases in which there are 5, 10 and 15 nodes in our network. All these cases have definite random movement and the nodes moves in the network region with a definite model. The results of the simulations with two other models TCP Newreno, TCP Vegas being applied in simialr conditions are compared.

The results of Figures 4, 5, 6 are shown. Figure 4 shows a state in which 5 nodes are in the network and in this case the function of our method is equal to TCP Newreno but it has better function compared to TCP Vegas. In Figure 5, this operation is repeated with the presence of 10 nodes and it is observed that congestion window size changes were more. In Figure 6, the scenario was performed by the presence of 15 nodes and after 30s the simulation of a node is out of the path and our proposed method detected the mobility immediately and responded to it. Also, it had better performance compared to two other methods.

Simulation 2- the Comparison Between Re-transmission and Throughput: In this section we compare the Throughput and re-transmission of the proposed model with two other models of TCP. The network model and the parameters were similar to the applied items in the previous simulation. Figures 7, 8 showed the results of simulation.

According to Figure 7, we can say that TCP Vegas at first had high Throughput but with the increase of the number of nodes in the path and as TCP Vegas keeps the sending window size small, its size is decreased and the proposed model has moderate performance.

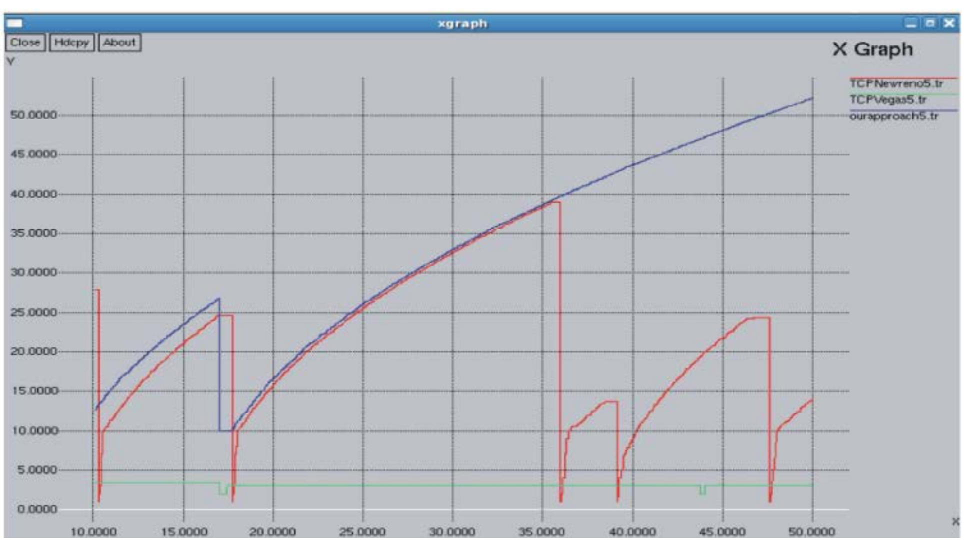

Fig. 4: The changes in congestion window size with 5 nodes

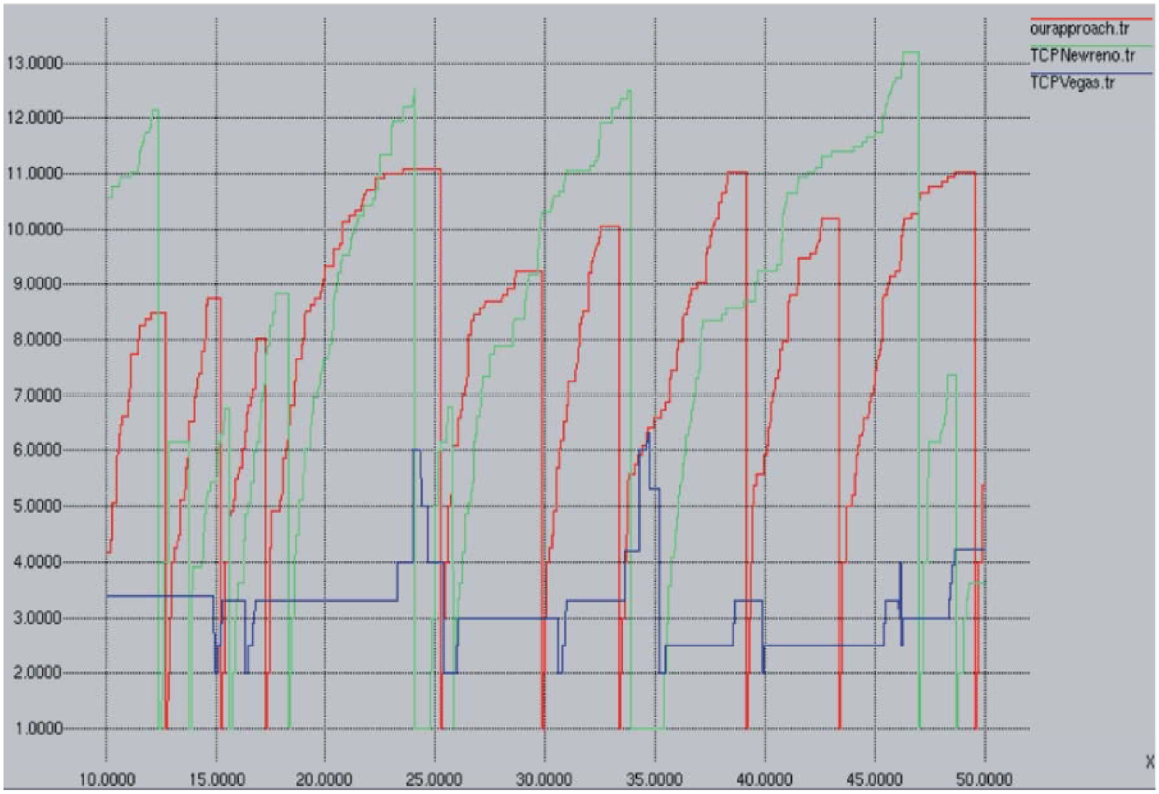

Fig. 5: The changes in congestion window size with 10 nodes

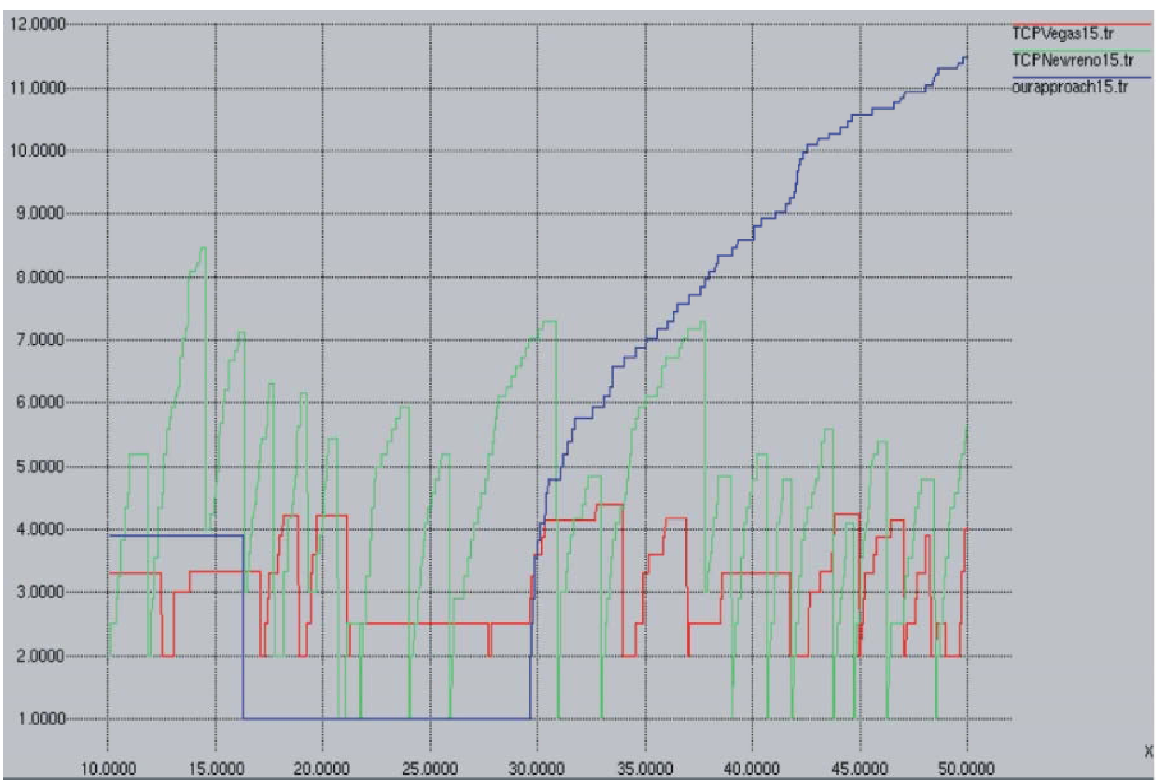

Fig. 6: The changes in congestion window size with 15 nodes

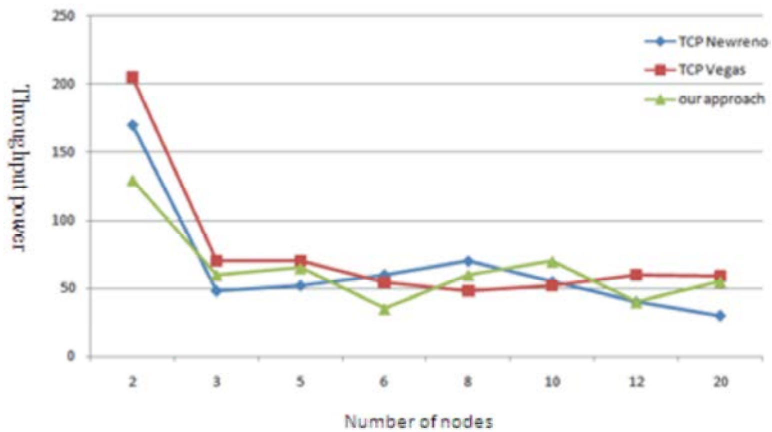

Fig. 7: Throughput power

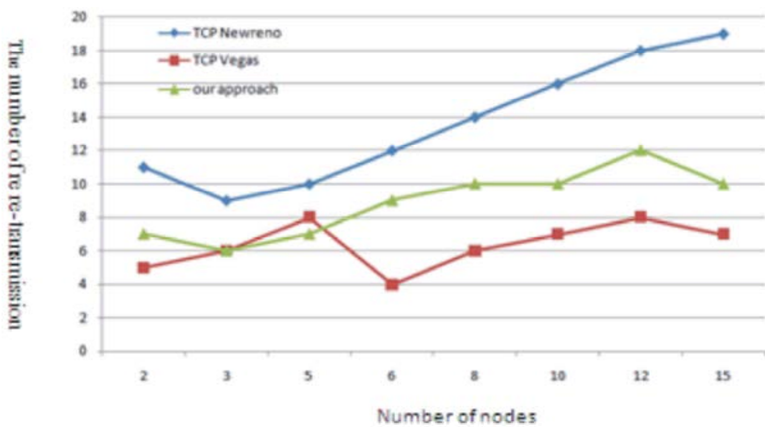

Fig. 8: The comparison of the number of re-transmission

Figure 8 showed that TCP Vegas keeps its sending window small and it had low re-transmission compared to other methods. But in TCP Newreno, the re-transmissions are more. The proposed method had a moderate limit between these two values for the number of re-transmissions [33].

CONCLUSION

As it was said before, TCP is one of the most common protocols in the network but the nature and design method of the protocol is not suitable for mobile ad hoc networks and it is problematic. One of the problems is the lack of coordination in congestion control mechanism not consisting with the structure of the network and it reduces the efficiency of the network. Thus, we require a series of optimizations to make the coordination. In the current paper, we investigated this issue. At first, TCP protocol was investigated and the shortcomings were studied and the methods and approaches were mentioned to solve the problems in mobile ad hoc networks. Then with the proposed view, a new cross-layer approach was introduced by which we can provide an environment that TCP can be faced with a few problems and the network is reliable in lower layers. This method applies the interaction between physical layer and network layer and avoids sudden disconnection of the route that is due to the mobility of the nodes and being out of the path or turn off due to the empty batteries. Finally, by NS2 simulator, the method was implemented and the efficiency of this method was compared with two models of TCP and the results were analyzed.

REFERENCES

1. Locher Christian, Bjorn Scheuermann and Martin Mauve, 2007. A Survey on Congestion Control for Mobile Ad-Hoc Networks, Wiley Wireless Communications and Mobile Computing, 7(5): 655-676.

2. Anantharaman, V., S.J. Park, K. Sundaresan and R. Sivakumar, 2004. TCP performance over mobile ad hoc networks: A quantitative study, Journal of Wireless Communications and Mobile Computing, 4(2): 203-222.
3. Allman, M., V. Paxson and W. Stevens, 1999. TCP Congestion Control. RFC 2581 (Proposed Standard), Apr. 1999. Updated by RFC 3390.
4. Akyol, Umut, Matthew Andrews and Piyush Gupta, 2008. Joint Scheduling and Congestion Control in Mobile Ad-Hoc Networks, IEEE INFOCOM.
5. Kliazovich, D. and F. Granelli, 2006. Cross-layer congestion control in ad hoc wireless networks. Ad Hoc Networks, 4(6): 687-708.
6. Conti, M., G. Maselli and G. Turi, 2004. Cross-layering in a mobile ad hoc network design, IEEE Comp. Soc., Commercial Implementation, pp: 48.
7. Liljana Gavrilovska, 2006. Cross-Layering Approaches in Wireless Ad Hoc Networks, Springer 2006, Wireless Personal Communications, 37: 271-290.
8. Kawadia, V. and P. Kumar, 2005. A cautionary perspective on cross layer design," IEEE Wireless Communication Magazine, 12(1): 3-11.
9. Tsaoussidis, V. and H. Badr, 2000. TCP-probing: towards an error control schema with energy and throughput performance gains, In Proceedings of the 8th IEEE International Conference on Networks Protocols.
10. Balakrishnan, H., 1997. A Comparison of Mechanisms for improving TCP Performance over Wireless Links", IEEE/ACM Trans. Net.
11. Braden, R.T., 1989. RFC 1122: Requirements for Internet hosts -- communication layers.
12. Jacobson, V., 1988. Congestion avoidance and control, ACM Computer Communications Review, Proceedings of the Sigcomm 88 Symposium in Stanford, CA, 18(4): 314-329.
13. Chandran, K., S. Raghunathan, S. Venkatesan and R. Prakash, 1998. A feedback based scheme for improving TCP performance in Ad-Hoc wireless networks, in Proc. of the International Conference on Distributed Computing Systems (ICDCS'98), Amsterdam, Netherlands.
14. Holland, G. and N. Vaidya, 2002. Analysis of TCP performance over mobile ad hoc networks, ACM Wireless Networks, 8(2): 275-288.
15. Liu J. and S. Singh, 2001. ATCP: TCP for mobile ad hoc networks, IEEE JSAC, 19(7): 1300-1315.
16. Kim, D., C. Toh and Y. Choi, 2001. TCP-BuS: Improving TCP performance in wireless ad hoc networks, Journal of Communications and Networks, 3(2): 175-186.
17. Toh, C., 1997. Associativity-based routing for ad hoc mobile networks, Journal of Wireless Personal Communications, 4(2): 103-139.
18. Kopparty, S., S. Krishnamurthy, M. Faloutous and S. Tripathi, 2002. Split TCP for mobile ad hoc networks, in Proc. of IEEE GLOBECOM, Taipei, Taiwan, Nov.
19. Goff, T., N. Abu-Ghazaleh, D. Phatak and R. Kahvecioglu, 2001. Preemptive routing in ad hoc networks, in Proc. of ACM MOBICOM, Rome, Italy, pp: 43-52.
20. Klemm, F., S. Krishnamurthy and S. Tripathi, 2003. Alleviating effects of mobility on tcp performance in ad hoc networks using signalstrength based link management, in Proc. of the Personal Wireless Communications, Venice, Italy, pp: 611-624.
21. Johnson, D., D. Maltz and Y. Hu, 2003. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR), Internet draft.
22. Dyer, T. and R. Boppana, 2001. A comparison of TCP performance over three routing protocols for mobile ad hoc networks," in Proc. Of ACM MOBIHOC, Long Beach, CA, USA, pp: 56-66.
23. Wang. F. and Y. Zhang, 2002. Improving TCP performance over mobile ad hoc networks with out-of-order detection and response," in Proc.of ACM MOBIHOC, Lausanne, Switzerland, pp: 217-225.
24. Lim, H., K. Xu and M. Gerla, 2003. TCP performance over multipath routing in mobile ad hoc networks, in Proc. of IEEE ICC, Anchorage, Alaska.
25. Altman E. and T. Jim´enez, 2003. Novel delayed ACK techniques for improving TCP performance in multi-hop wireless networks," in Proc.of the Personal Wireless Communications, Venice, Italy, 237–253.
26. Cordeiro, C., S. Das and D. Agrawal, 2003. COPAS: Dynamic contention-balancing to enhance the performance of tcp over multi-hop wireless networks, in Proc. of IC3N, Miami, USA, pp: 382-387.
27. Fu, Z., P. Zerfos, H. Luo, S. Lu, L. Zhang and M. Gerla, 2003.
28. Ramakrishnan, K., S. Floyd and D. Black, 2001. The addition of explicit congestion notification (ECN) to IP, RFC 3168, Category: Standards Track.
29. Macker, J., 1999. RFC 2501: Mobile Ad hoc Networking (MANET), IETF.

30. Perkins, C., E. Belding-Royer and S. Das, 2003. Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561, Category: Experimental.
31. Floyd, S., 1999. The New Reno Modification to TCP's Fast Recovery Algorithm. RFC, 2582.
32. Brakmo, L.S., S.W. O'Malley and L.L. Peterson, 1994. TCP Vegas: New techniques for congestion detection and avoidance,In Proceedings, 1994 SIGCOMM Conference, London, UK, pp: 24-35.
33. The Network Simulator - ns-2, http://www.isi.edu/nsnam/ns/.