Futuristic Areas in Computer Engineering and Technology (FACET-13)2013
Proceedings published by International Journal of Computer Applications® (IJCA)

A Proposal to use Cross-Layer based TCP Protocol for Congestion Control in Multi hop Mobile Ad hoc Networks

K.Tarun kumar[1], Ch.Satyendra[2], D.Sunitha[3]
Graduate students in CSE[1,2] ,Assist.Professor[3]
Kamala Institute of Technology and Sciences.[123]
Singapuram,Huzurabad,Karimnagar

ABSTRACT
A cross-layer based approach for improving TCP performance in Multihop Mobile Adhoc Networks (MANETs) is proposed in this paper. The proposed congestion triggering mechanism triggers congestion whenever the channel occupied ratio (COR) reaches a maximum threshold value and the received signal strength is less than a minimum threshold value. Following it, the congestion control scheme controls the data sending rate of the sender by determining available bandwidth, delay of its link and COR. Further, a fair resource allocation scheme is put forwarded.

1. INTRODUCTION
1.1 Wireless Networks

Wireless networks are more popular among company and home users worldwide. Users are searching for novel technologies that provides the service to communicate anytime and anywhere using any communication device. Hence, wireless communication play a vital role in upcoming communication systems. The merits of wireless networks over wired counterparts are flexible mobility management, quicker and cheaper operation, and finally simpler preservation and advance procedures.

Users started utilizing the ad hoc mode of 802.11 apart from the infrastructure mode, in which multiple wireless hops are used to join two far-away nodes. Nodes can communicate to each other directly without a central coordinator in the ad hoc mode and convey data to each other in an autonomous manner. This architecture is known as multihop wireless ad hoc networks or multihop wireless networks. The network topology can change quickly and randomly as the mobile nodes change its position or the wireless channel's condition vary. Robust and adaptive communication protocols that can handle the tasks of these multihop networks smoothly are required.

1.2 TCP Based Flow and Congestion Control

One of the window based flow and congestion control protocol is TCP which is used to maintain the data transmission by using sliding window techniques. The main objective of the scheme is to assure that the sender must level its transmission rate to satisfy their own and receiver's needs. Therefore, the TCP sender has a variable denoted window to determine the number of packets it can send into the network prior to receive an ACK. This variable vigorously varies over time to limit the connection's sending rate.

The flow control and the congestion control are the two distinct techniques that regulates the sending rate of a TCP connection. Though these two mechanisms are similar in the case that both try to avoid the connection from sending at an increased rate, they have unique purposes. To avoid a TCP sender from overflowing the receiver's buffer, flow control is implemented. In every ACK transmitted, the receiver advertises a window limit to the sender. This window is called as receiver advertised window (rwin). It alters over the time based on both the traffic conditions and the application speed while reading the receiver's buffer [1] [2] [3].

Congestion control is related with the traffic inside the network contrary to flow control. The main purpose is to avoid failure inside the network when the network is slower than the traffic source in forwarding the data. In addition to that, TCP sender uses a limiting window called congestion window (cwnd). By assuming that the sender is not limited by the receiver, cwnd denotes the amount of data that the sender send before an ACK is received.

By taking both flow control and congestion control into account, the sender has to face two limiting factors for its window size: rwin and the cwnd. The TCP sender regulates its window to the minimum between rwin and cwnd to match with both control schemes. Generally, cwnd is taken as the limiting factor of a TCP sender because the receiver's buffer is frequently large enough not to limit the sender's transmission rate. TCP. To detect congestion inside the network and rapidly react by accurately slowing down, congestion control is developed over the years.

1.3 TCP for Multihop Wireless Networks

For multihop wireless networks, TCP/IP is the best choice because most of the recent applications like HTTP, FTP, SMTP, and Telnet are suitable for this protocol. Also, the use of TCP/IP' makes interoperation with the internet. Nowadays, the exclusive features of 802.11 that is addressed below, needs the alterations in the upper layer protocols used in the Internet. Especially, the trustworthy data delivery provided by the predominant internet transport protocol TCP is compromised in such networks. When the network is large, the degradation is higher. Adjusting TCP to these networks is important in wireless networks as bandwidth is a very limited resource.

Due to the inequality between TCP and the MAC protocol, TCP degradation may happen in Multihop networks. Even if the IEEE 802.11 standard has potential to work on ad hoc mode permitting the setup of a network without infrastructure, it is not optimized for networks with large number of hops. This standard denotes RTS/CTS control frames to confirm that network transmit with maximum three hops are not impacted by the well-known hidden node problem. Contention collisions may take place to degrade the channel quality for more than three hops. The overhead of RTS/CTS combined with the lossy nature of the wireless channel and mobility can cause a TCP connection to a poor performance. This is due to the fact that TCP is designed for wired networks where this condition does not exist. We review the key challenges for TCP over Multihop wireless networks next.

1.4 Challenges of TCP in Multihop Wireless Networks

Wireless networks may have loss due to the lossy nature of its medium and link interruption when the nodes move, unlike wired environments where a dropped packet is always belonged to congestion [1]. In conventional TCP, this may be a problem since it minimizes its transmission rate when a drop is seen irrespective of the loss nature. There is a mechanism at the sender which can distinguish the original cause of a packet drop so that the sender is able to react to each of the factors that is causing the losses. Past work on this problem have severe limitations such as high processing overhead and entire dependence on network explicit signaling.

1.4.1 *High MAC Contention and Collisions:* TCP depends on acknowledgment packets from receiver to sender for establishing a bidirectional flow of data and ACKs so as to ensure reliability [1]. It is an expensive technique in multihop wireless networks. Because of the important MAC overhead associated with an ACK transmission in spite of the smaller ACK size comparative to a data packet. This is because of the random back off procedure that pursues any failed transmission attempt and the RTS/CTS control frames exchanged prior to any packet transmission. So far, inside the network, data and ACK flowing in different directions are highly vulnerable to collide. Under suitable conditions on optimal bandwidth utilization, TCP must avoid sending redundant ACKs.

To solve this issue, earlier techniques have proposed to minimize the number of ACKs entered into the network in a fixed manner. Under certain conditions, it is not possible because the network condition changes and redundant ACKs may be critical to the end-to-end performance.

1.4.2 *Low Energy Resources***:** Multihop wireless networks consists of mobile nodes that are most probably powered by battery [1]. The involved protocols must have a well-saturated compromise between performance and energy utilization.

The autonomous retransmissions due to the lack of interaction between TCP and 802.11 is the primary source of energy wastage in a TCP implementation over multihop networks. Various energy saving mechanisms for link and network layers are arriving, which have not been analyzed on the transport layer.

1.5 Problem Identification and Solution

Daniel Scofield et al. [5] have proposed HxH, a hop-by-hop transport protocol that utilizes credit-based congestion control and reverse ACKs to work out the crisis with TCP. Congestion control algorithms do not use rate-based and pricing based feedback. This is the major limitations of the proposed approach. The proposed technique is not improved with fast variation of the congestion window in [6].A set of techniques in TCP using a cross-layer approach is used to address the challenges are proposed. It also provides better interaction to effectively improve the end-to-end performance among TCP and 802.11 protocols.

The paper is arranged as follows: section-2 contains related works, section-3 contains proposed solution, , Section-4 contains conclusion.

2. RELATED WORKS

Jin Ye et al. [4] proposed an improved TCP with cross layer congestion notification over Wired-wireless hybrid networks. Being different from some existing TCP protocols, it takes account into the congestion of channel competition in MAC layer. What most important is that this model is built on the ECN scheme, which have been proven to be effective on congestion control and widely supported in many situations. So this model is of good feasibility and scalability. In future, the work will be based on other congestion metrics in MAC layer and how explicitly notify TCP sender, by which the proposed TCP model can be applied into other congestion control schemes.

Daniel Scofield et al. [5] designed HxH, a hop-by-hop transport protocol that uses credit-based congestion control and reverse ACKs to solve the problem with TCP. The problems include contention, interference, the hidden and exposed terminal problems, shared queues, half-duplex links, and route changes due to mobility. Credit-based congestion control reacts quickly whenever network conditions change, and can improve fairness among flows competing on the same path. The drawback of the proposed approach is that, congestion control algorithms that use rate-based and pricing based feedback is not used.

Myungjin Lee et al. [6] proposed a path recovery notification (TCP–PRN) mechanism to prevent performance degradation during a handoff. Even though Freeze-TCP achieves a performance increase during handoff, detecting accurate handoff time and the vulnerability of high variation in the round trip time (RTT) become obstacles for deploying Freeze-TCP in a real environment. The proposed protocol, TCP–PRN, quickly recovers lost packets by restoring the congestion window, preventing the congestion window to decrease, or immediately initiating the slow start algorithm. The proposed mechanism is not enhanced with fast adaptation of the congestion window.

Jingyuan Wang et al. [7] demonstrated the performance of a new TCP congestion control algorithm in different network conditions. The analysis show that the new TCP congestion control algorithm, namely, TCP-FIT, out-performs existing algorithms such as Bic/Cubic, Reno, Veno, Westwood, Compound TCP, FastTCP and etc. under challenging network conditions, while maintaining good inter and intra protocol fairness.

Shengming Jiang et al. [8] proposed a semi-TCP using a hop-by-hop congestion control. Due to using hop-by-hop congestion control, the congestion control efficiency of semi-TCP will not rely on the availability of end- to-end connectivity, which makes semi-TCP more suitable than TCP for challenged networks. Besides performance improvement, semi-TCP may further reduce overall system complexity by removing redundant congestion control and using simple congestion control rather than TCP congestion window. The issues of semi-TCP such as its impact to end-to-end behavior of transport layer, effect on high layer networking function and its inter-operability with the original TCP is not studied in the current approach.

Hongqiang Zhai et al. [9] proposed a new wireless congestion control protocol (WCCP) based on the novel use of channel busyness ratio along with characterization of network utilization and congestion status. In this protocol, each forwarding node along a traffic flow exercises the inter-node and intra-node fair resource allocation and determines the MAC layer feedback accordingly. The proposed scheme outperforms traditional TCP in terms of channel utilization, end-to-end delay, and fairness, and solves the starvation problem of TCP flows.

Ruy de Oliveira et al. [10] proposed a smart TCP acknowledgement approach for multihop wireless networks. The crucial challenge faced by TCP to operate smoothly with 802.11 wireless MAC protocol renders TCP acknowledgement transmission quite costly. This paper evaluated a dynamic adaptive strategy for reducing ACK induced overhead and consequently collisions.

Their approach resembles sender side’s congestion control. The receiver is self-adaptive by delaying more ACKs under non-constrained channels and less otherwise. This improves not only throughput but also power consumption. An adaptive receiver mechanism to switch between DAA and DAAp strategies in scenarios susceptible to high bit error rate is not considered in the present approach.

Xuyang Wang et al. [11] proposed a cross-layer hop-by-hop congestion control scheme designed to improve TCP performance in multi hop wireless networks by solving the aforementioned false routing disruption problems and providing the on-demand routing protocol with congestion control capability.

The scheme is based on a hop-by-hop approach that enables the on-demand routing protocol to be congestion-aware and actively participate in congestion control. That is, once a node infers the cause of a packet loss, the response to the loss is coordinated across the MAC, routing, and transport protocols in order to achieve higher overall system performance

Yao-Nan Lien et al. [12] proposed the Hop-by- Hop TCP protocol for sensor networks aiming to accelerate reliable packet delivery. Hop-by-Hop TCP makes every intermediate node in the transmission path execute a light-weight local TCP to guarantee the transmission of each packet on each link. It takes less time in average to deliver a packet in an error-prone environment. In the future, they will use One-Hop TCP to serve all TCP and even UDP so that the number of packets transmitted on the air can be greatly reduced.

3. PROPOSED SOLUTION

3.1 Overview

A cross-layer based approach for improving TCP performance in Multihop Mobile Adhoc Networks (MANETs) is proposed in this paper. The proposed congestion triggering mechanism triggers congestion whenever the channel occupied ratio (COR) reaches a maximum threshold value and the received signal strength is less than a minimum threshold value. Following it, the congestion control scheme controls the data sending rate of the sender by determining available bandwidth, delay of its link and COR. Further, a fair resource allocation scheme is proposed. The resource allocation scheme is incorporated with Additive Increase and Multiplicative Decrease (AIMD) law and it assures fair resource allocation among flows.

3.2 Computation of Metrics

3.2.1 *Channel Occupied Ratio (COR) Evaluation*

In the standard IEEE 802.11 architecture, the channel occupied ratio can be estimated easily. So that IEEE 802.11 is a CSMA-based Mac protocol, which is working on the physical and virtual carrier sensing mechanisms. The wireless communication channel is taken as busy or engaged when it is sending or receiving data. To find whether the channel is busy or idle, the IEEE 802.11 has various functions. Busy channel can be denoted by its network allocation vector (NAV). COR is measured as the top metric for end-to-end congestion control mechanisms since it provides the early signal of network congestion.

The channel occupied ratio (COR) is computed as the ratio of total lengths of busy periods to the total time during a time interval t_n, [9] Let T denotes the total transmission time and T_B denotes the total lengths of busy periods.

The COR can be given as,

$$COR = T_B / T \qquad (1)$$

Taking into account channel utilization factor, we define a threshold value Th_{COR}. It is selected in a way such that,

$$COR \approx U_C \left(COR \leq Th_{COR} \right) \qquad (2)$$

where, U_C denotes channel utilization factor and it is the measure of ratio of channel busyness time for successful transmissions to the total time T.

3.2.2 *Received Signal Strength Estimation*

The received signal strength (R_S) is evaluated as [11],

$$R_S = P_{res} d^{-\beta} \qquad (3)$$

where, P_{res} is the reception power at a reference distance and it is usually one meter. β is the distance-power gradient value that differs with the surrounding terrain conditions.

3.2.3 *Available Bandwidth Assessment*

Consider BW_T as the total bandwidth and BW_A as available bandwidth, then in view of Th_{COR} and COR, the available bandwidth of a node is calculated as,

$$BW_A = \begin{cases} BW_T\,(Th_{COR} - COR)\,Data / AvgT_S\,, when\left(COR < Th_{COR}\right) \\ 0,\ when(COR \geq Th_{COR}) \end{cases} \quad (4)$$

In the above equation, Data represents the average payload size and $AvgT_s$ denotes the average successful transmission time at the MAC layer.

3.2.4 *Delay Calculation*

The time interval between data transmission and reception is referred as the delay of link (L_i). Consider T_D as the time of data transmission and T_R as the time of data reception, then the delay incurred in link L_i can be given as,

$$Delay\ (L_i) = T_D - T_R \quad (5)$$

3.3 Congestion Triggering Mechanisms

The wireless network can bear various types of data losses that happens due to link failures, security attacks etc. The proposed mechanism differentiates packet loss rooted by congestion by considering the two parameters like received signal strength (R_S) and channel occupied ratio (COR). COR is a perfect metric for detecting congestion in the network, as we discussed earlier. To monitor the network so as to fix the congestion in the network, the proposed congestion triggering scheme is used. Every node calculates R_S and COR using equations given in (3) and (1) respectively. Our congestion triggering scheme is incorporated in each node of the network to monitor the network. Consider Th_{COR} as the threshold value of COR and $minTh_{Rs}$ as the threshold of minimum received signal strength value. Assume that every node periodically calculates COR and R_S. The computed values are compared with Th_{COR} and $minTh_{Rs}$ respectively. When the calculated COR reaches Th_{COR} or the received signal strength goes below $minTh_{Rs}$, it triggers congestion notification message. The created congestion notification message is transmitted to the source through a flag termed as *Con-notify flag*. While receiving *Con-notify flag*, the source initiates rate control mechanism.

Our congestion triggering scheme is described in the following algorithm,

Algorithm-1

Let Th_{COR} be the maximum threshold value of COR

Consider $minTh_{Rs}$ as the threshold of minimum received signal strength value.

Assume n_1, n_2, … N as a set of mobile nodes in the network

Node n_i calculates COR using equation (1)

Node n_i calculates R_S using equation (3)

COR and R_S values are compared with their threshold values

If (COR $\geq Th_{COR}$ && $R_S < minTh_{Rs}$) then

Data loss is triggered by congestion

n_i transmits Con-notify flag to the source node

End if

Consider the network with nine nodes as shown in figure-1. The source transmits data to the destination through nodes 4-6-7. While data is transmitted between nodes 6 and 7, node 6 finds that the estimated COR value exceeds Th_{COR} and its calculated R_S is lesser than $minTh_{Rs}$. It immediately sends *Con-notify flag* to the source node.

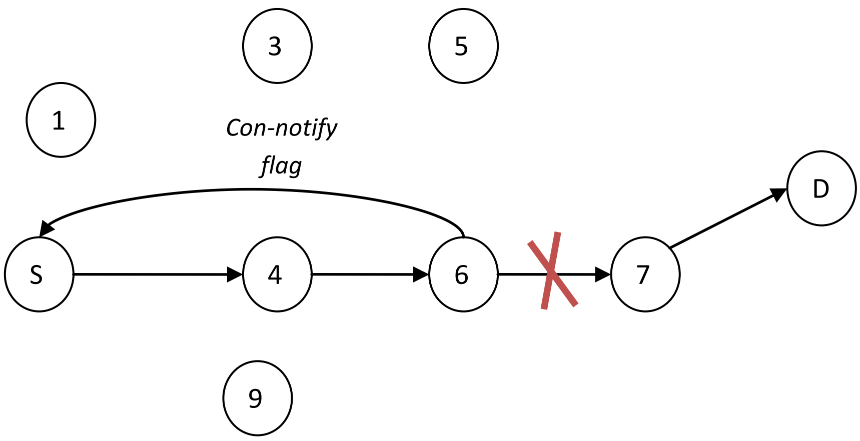

Figure-1 Congestion Triggering Mechanism

3.3.1 *Congestion Control Technique*

By considering three metrics such as bandwidth, delay and COR, end-to-end congestion control technique is operated. This is operated at every node. Available bandwidth, delay of its link and COR are estimated by every node using their respective methods given in equations (4), (5), and (1). This estimations are taken along with the MAC header through the intermediate nodes to the destination, while transmitting data from the source to the destination. The link and physical layers headers encapsulates the cumulative bandwidth, delay and COR values of links that connect the source-destination. it regulates its traffic sending rate when the source node receives this acknowledgement packet.

Consider S and D as source and destination nodes respectively. When the source desires to transmit data to the destination, it forwards the first data packet to an intermediate node (say I_1). When a packet is received, the data link layer of node I_1 measures the bandwidth, delay for its link, and channel busyness ratio. Then it includes this measured information with the MAC header and then forwards it to the next intermediate node I_2. The receiving node I_2 will also do the same measurements for its link and determines the minimum value of delay, bandwidth and COR measurements. It then updates this bandwidth information in the MAC header. The cumulative delay of both the links is calculated and updated by I_2. When the TCP packet finally reaches the destination node D, the MAC header contains the cumulative bandwidth, delay, and COR information of all the links along the path. The destination node D sends this information along with the acknowledgement packet to the source node S, encapsulated by link and physical layer headers. When the source S receives this acknowledgement packet, it will adjust its traffic sending rate using the calculated bandwidth, delay, and COR information. The proposed congestion control technique is illustrated in Figure-2.

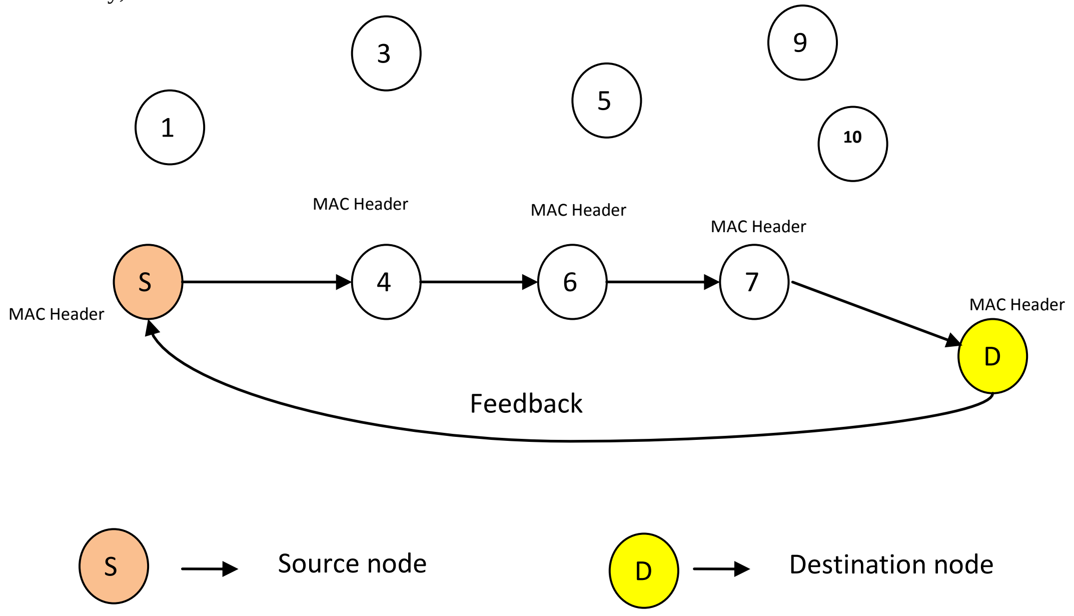

Figure-2 Congestion Control Technique

3.4 Fair Resource Allocation Scheme

Using the equation given in (4), every node in the network can easily measure available bandwidth of its neighbors by considering COR. In our proposed technique, COR is calculates at periodic interval defined as average time interval of COR (Avg T_{COR}).

In this context, we relate the available bandwidth of every node as the channel resource (CR) proportionally to its present traffic load (L_T). Thus, the linear relationship can be expressed as follows:

$$CR = \frac{Th_{COR} - COR}{COR} \times L_T \qquad (6)$$

In the above equation L_T denotes the current traffic load, which is the sum of both incoming and outgoing traffics.

In order to achieve efficiency in data transmission and fairness in resource allocation, the proposed resource allocation scheme utilizes an Additive-Increase Multiplicative Decrease (AIMD) policy along with COR and CR.

In general, when CR > 0, flows increase data sending rate and on the other hand, when CR < 0, flows decrease their data sending rate. By increasing or decreasing data sending rate, flows increase or decrease the throughput of network respectively.

As we discussed earlier, since the channel is utilized by both incoming and outgoing traffic, the number of flows in the network is estimated to accomplish fairness in resource allocation. In this scheme, a flow that originates or terminates at a node is considered as one flow and a flow that processed at a node is considered as two flows. Because flows that process at node consumes twice resources of that originating and terminating flows.

Consider ds_r as the data sending rate, then number of flows (nF) can be calculated as,

$$nF = \sum_{k=1}^{n} \frac{1}{ds_{rk} \, AvgT_{COR}}$$

(7)

where, k is the number of packets monitored by node i in Avg T_{COR}. In short, nF is the summation of both received and transmitted packets.

When $CR \geq 0$ the increasing level of data sending rate for flow F_i and per packet acknowledgment (pACK) are,

$$F_i = \frac{CR}{AvgT_{COR} nF} \quad (8)$$

$$pACK = \frac{F_i}{ds_{rk} AvgT_{COR}} \quad (9)$$

When CR < 0, the proposed technique reduces the flow's throughput related to its present throughput. We can observe in equation (below) that per packet acknowledgement (pACK) is inversely proportional to the expected number of packets monitored by node n_i in AvgT_{COR}. As a result,

$$pACk = \eta \frac{ds_{rk}}{ds_{rk} \times AvgT_{COR}} = \frac{\eta}{AvgT_{COR}} \quad \textbf{(10)}$$

Here, η is a constant and then,

$$\sum_{k=1}^{n} pACK = \frac{CR}{AvgT_{COR}} \quad \textbf{(11)}$$

$$\eta = \frac{CR}{n} \quad \textbf{(12)}$$

To accomplish fair channel resource, COR should be close to Th_{COR} but never it should exceed Th_{COR}. To incorporate fair resource allocation with AIMD law, we define two parameters namely CR^+ and CR^- to represent increased and decreased traffic respectively. In the same way, $pACK^+$ and $pACK^-$ represents positive and negative feedback. Here, $pACK^+$ is measured by the increase law and $pACK^-$ is calculated by decrease law. Therefore,

$$CR = CR^{+} + CR^{-} \quad (13)$$

$$pACK = pACK^{+} + pACK^{-} \quad (14)$$

if $\left(0 < CR < \partial(L_T + CR)\right)$

then $CR^{+} = \partial(L_T + CR)$

If $\left(0 > CR > -\partial(L_T + CR)\right)$

$$CR^{-} = -\partial(L_T + CR)$$

Thus, the incorporation of AIMD law with our proposed scheme achieves a fair resource allocation scheme.

4. CONCLUSION

A cross-layer based approach for improving TCP performance in Multihop Mobile Adhoc Networks (MANETs) is proposed in this paper. The proposed congestion triggering mechanism triggers congestion whenever the channel occupied ratio (COR) reaches a maximum threshold value and the received signal strength is less than a minimum threshold value. Following it, the congestion control scheme controls the data sending rate of the sender by determining available bandwidth, delay of its link and COR. Further, a fair resource allocation scheme is put forwarded.

5. REFERENCES

[1] Vorgelegt von , Ruy de Oliveira and Von Brasilien "Addressing the Challenges for TCP over Multihop Wireless Networks"

[2] Kaixin Xu, Mario Gerla Lantao Qi, Yantai Shu, "Enhancing TCP Fairness in Ad Hoc Wireless Networks Using Neighborhood RED" MobiCom'03, September 14–19, 2003, San Diego, California, USA

[3] K. Nahm, A. Helmy, and C. J. Kuo, "TCP over multihop 802.11 networks: Issues and performance enhancement", ACM, May 2005.

[4] Jin Ye, Jianxin Wang, Qinghua Liu, Yuhong Luo, "An Improved TCP with Cross-layer Congestion Notification over Wired/Wireless Hybrid Networks", ICYCS, pp 368-373, 2008

[5] Daniel Scofield, Lei Wang, Daniel Zappala: "HxH: a hop-by-hop transport protocol for multi-hop wireless networks", WICON 2008: 16

[6] Myungjin Lee, Moonsoo Kang, Myungchul Kim, and Jeonghoon Mo, "A Cross-Layer Approach for TCP Optimization over Wireless and Mobile Networks," ELSEVIER, Computer Communications 31 (2008) 2669–2675

[7] Jingyuan Wang, Jiangtao Wen, Jun Zhang, Yuxing Han, "A demonstration of a new TCP congestion control algorithm over LTE and other challenging networks", MobiCom 2010 Demo, Sept. Chicago, IL.

[8] Shengming Jiang, Qin Zuo and Gang Wei, "Decoupling congestion control from TCP for multi-hop wireless networks: semi-TCP", Proceeding Proceedings of the 4th ACM workshop on Challenged networks, CHANTS '09

[9] Hongqiang Zhai, Xiang Chen, and Yuguang Fang, "Improving Transport Layer Performance in Multihop Ad Hoc Networks by Exploiting MAC Layer Information", IEEE Transactions on Wireless Communications, Vol. 6, No. 5, MAY 2007

[10] Ruy de Oliveira, Torsten Braun, "A Smart TCP Acknowledgment Approach for Multihop Wireless Networks", IEEE Trans. Mob. Comput. 6(2): 192-205 (2007)

[11] X. Wang and D. Perkins, "Cross-Layer Hop-by-Hop Congestion Control in Mobile Ad Hoc Networks", Proceedings of IEEE Wireless Communications and Networking Conference, March 2008.

[12]12. Yao-Nan Lien, "Hop-by-Hop TCP for Sensor Networks", International Journal of Computer Networks & Communications (IJCNC), Vol.1, No.1, April 2009