International Journal of Electrical Electronics & Computer Science Engineering
Volume 3, Issue 1 (February, 2016) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

# Efficient Method For Congestion Detection And Avoidance For TCP Using SNR And Buffer Value

Anup A. Kawathekar, Supriya A. Jadhav, Kedar R. Joshi
SRTTC-VIT, Kamshet, Pune
anup2690@gmail.com, indalkar.supriya@gmail.com, kedarjoshi128@gmail.com

***Abstract:*** ***As TCP is most popular and widely used network transmission protocol. In actual for communication using internet, TCP is widely used and very much reliable for wired network. However, networking becomes more heterogeneous for both wired as well as wireless topologies. TCP is known to have poor performance in wireless links where packet losses are due to transmission error, mobility is to be treated by congestion in network. In case of wired network TCP considers all packet timeouts due to congestion in network, but in wireless network TCP is not able to distinguish between error losses and congestion losses. TCP only considers the losses due to congestion and not due to network problem. As a result it directly slows down the data rate and hence affecting the Quality of service (QOS) and throughput of a network. This paper propose new scheme to distinguish error losses and congestion losses in both wired as well as wireless network. The scheme is based on use of reserved field of the TCP header, increasing the SNR ratio to detect the reliability of the link and decide whether to reduce packet data rate or retransmit a timed-out packet.***

***Keywords: TCP Congestion, Wireless Network, Reserved Field, SNR, Packet data rate, Packet loss and Buffer size.***

## I. INTRODUCTION

When too many packets are present in a subnet, performance degrades. This situation is called Congestion. In any network when there is heavy data traffic at a node, that network slows down or starts loosing data, it is known as network congestion. It degrades quality of service and also can lead to delays, data loss or e.g. dropped calls on VOIP network. When excess of data travels in a network there is high probability of Congestion in the network. This congestion will lose the packet and make time delay in the networks. Modern Telecommunication, Computer Networks and both wired and wireless communications including the Internet, are being designed for fast transmission of large amounts of data, for which Congestion Control is very important. Transmission Control Protocol (TCP), the most widely used reliable transport protocol, was designed mainly for wired networks where transmission errors are rare and the majority of packet losses are caused by congestion. When applied to wireless networks where transmission errors are frequent, TCP is found to have poor performance if proper enhancement is not used. In wireless, loss is treated as a congestion loss, where as there are several other reasons for data loss in wireless environment. As TCP is most popular and widely used network transmission protocol. In actual for communicating on internet TCP is widely used and very much reliable for wired network. However, networking becomes more heterogeneous for both wired as well as wireless topologies. TCP is known to have poor performance in wireless links where packet loss due to transmission error, mobility is treated as congestion in network. In case of wired network TCP considers all packet timeouts as congestion in network, but in wireless network TCP is not able to distinguish between error losses and congestion losses in network. So TCP directly consider the losses due to congestion and not due to network problem. As results it directly slows down the data rate hence effecting on Quality of service (QOS) and throughput of a network. Due to this reasons one of the most suitable protocol in internet protocol suit is TCP protocol, which gives the concept of congestion control with controlling the flow of packets in network and prevents network performance [1].

In early days it was difficult to determine the cause of packet loss, as acknowledgments were not received from the receiver. But it does not mean that the packet is lost due to congestion and not because of network error or noise. However, with the tremendous advancement in technology, the packet losses are reduced due to more reliability on network infrastructure and communicating nodes. Hence, we conclude that the packets timeouts in wired network is due to congestion. The reliability in network has led the TCP protocol to be optimized for that the packet loss in wired network is due to congestion but not due to network error. So instead of distinguishing the problem it will directly slows down the data rate and announces the congestion and it affects the throughput of TCP and reduces overall speed of network. This case is not suitable for wireless networks. In wireless networks many problems affecting while communicating such as collision, interference, mobility, noise, hand-off, fading and other radio frequencies; Consequently the packet loss in wireless network cannot be considered only because of congestion. As a result TCP may give wrong decision by slows down the data rate instead of retransmitting lost packets [2].

## II. CONGESTION CONTROL IN TCP

Main aspect of TCP is congestion control. TCP uses a number of mechanisms to achieve high performance and avoid congestion collapse, where network performance fails. It controls rate of data entering the network, keeping the data flow below a threshold value at which TCP triggers or announce congestion state. TCP's senders and receivers alter the behaviour of flow of data.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 3, Issue 1 (February, 2016) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

This is referred to as congestion control or network congestion avoidance. Enhancing TCP to reliably handle loss, minimize errors, manage congestion and go fast in very high speed transmission. TCP congestion avoidance algorithm is a primary base for congestion control in the network. Problem occurs when many concurrent TCP flows are experiencing port queue buffer tail drop, then TCP's automatic congestion avoidance is not enough. Some of the congestion avoidance methods are as follows.

*A. Random Early Detection (RED):* Also known as random early discard or random early drop is a queuing discipline for a network scheduler suited for congestion avoidance. In the conventional tail drop algorithm, a router or other network component buffers as many packets as it can, and simply drops the ones it cannot buffer. If buffers are constantly full, the network is congested. Tail drop distributes buffer space unfairly among traffic flows. Tail drop can also lead to TCP global synchronization as all TCP connections "hold back" simultaneously, and then step forward simultaneously.

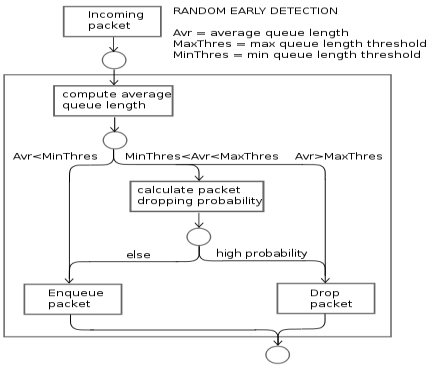


Fig.1. Random Early Detection

RED monitors the average queue size and drops packets based on statistical probabilities. If the buffer is almost empty, all incoming packets are accepted. As the queue grows, the probability for dropping an incoming packet grows too. When the buffer is full, the probability has reached 1 and all incoming packets are dropped. RED is more fair than tail drop, in the sense that it does not possess a bias against burst traffic that uses only a small portion of the bandwidth. The more the host transmits, the more likely it is that its packets are dropped as the probability of a host's packet being dropped is proportional to the amount of data it has in a queue.

*B. Robust Random Early Detection (RRED):* Is a queuing discipline for a network scheduler. The existing random early detection (RED) algorithm and its variants are found vulnerable to emerging attacks, especially the Low-rate Denial-of-Service attacks (LDOS). The Robust RED (RRED) algorithm was proposed to improve the TCP throughput against LDOS attacks. The basic idea behind the RRED is to detect and filter out attack packets before a normal RED algorithm is applied to incoming flows. RRED algorithm can significantly improve the performance of TCP under Low-rate denial-of-service attacks. A detection and filter block is added in front of a regular RED block on a router. The basic idea behind the RRED is to detect and filter out LDOS attack packets from incoming explosions before they feed to the RED. Within a benign TCP flow, the sender will delay sending new packets if loss is detected (e.g., a packet is dropped). Consequently, a packet is suspected to be an attacking packet if it is sent within a short-range after a packet is dropped. This is the basic idea of the detection algorithm of Robust RED (RRED).

*C. TCP Window Shaping:* Congestion avoidance can also efficiently be achieved by reducing the amount of traffic flowing into a network. When an application requests a large file, graphic or web page, it usually advertises a "window" of between 32K and 64K. This results in the server sending a full window of data (assuming the file is larger than the window). When there are many applications simultaneously requesting downloads, this data creates a congestion point at an upstream provider by flooding the queue much faster than it can be emptied. By using a device to reduce the window advertisement, the remote servers will send less data, thus reducing the congestion and allowing traffic to flow more freely. This technique can reduce congestion in a network by a factor of 40.

*D. Explicit Congestion Notification:* Another approach is to use IP Explicit Congestion Notification (ECN). ECN is only used when the two hosts signal that they want to use it. With this method, a protocol bit is used to signal explicit congestion. This is better than the indirect packet delete congestion notification performed by the RED/WRED algorithms, but it requires explicit support by both hosts to be effective. Some outdated or buggy network equipment drops packets with the ECN bit set, rather than ignoring the bit [7]. When a router receives a packet marked as ECN capable and anticipates (using RED) congestion, it sets the ECN flag notifying the sender of congestion. The sender should respond by decreasing its transmission bandwidth, e.g., by decreasing the TCP window size (sending rate) or by other means. In TCP/IP, routers operate mostly on the Internet layer, while transmission rate is handled by the endpoints at the transport layer. Congestion may be handled only by the transmitter, but since it is known to have happened only after a packet was sent, there must be an echo of the congestion indication by the receiver to the transmitter. Without ECN congestion indication echo is achieved indirectly by the detection of lost packets. With ECN, the congestion is indicated by setting the ECN field within an IP packet to CE and is recall back by the receiver to the transmitter by setting proper bits in the Transport Layer's protocol header. For example, when using TCP, the congestion indication is recall back by setting the ECE bit.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 3, Issue 1 (February, 2016) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

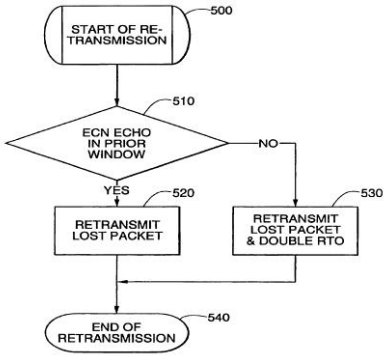


Fig.2. ECN

## III. SIDEEFFECTS OF CONGESTION COLLAPSE AVOIDANCE

*A. Radio Links:* The protocols that avoid congestive collapse are often based on the idea that data loss on the Internet is caused by congestion. This is true in nearly all cases; errors during transmission are rare on today's fibre based Internet. However, this causes Wi-Fi, 3G or other networks with a radio layer to have poor throughput in some cases since wireless networks are susceptible to data loss due to interference. The TCP connections running over a radio based physical layer see the data loss and tend to believe that congestion is occurring when it isn't and erroneously reduce the data rate sent.

*B. Short-Lived Connections:* The slow-start protocol performs badly for short-lived connections. Older web browsers would create many consecutive short-lived connections to the web server, and would open and close the connection for each file requested. This kept most connections in the slow start mode, which resulted in poor response time. To avoid this problem, modern browsers either open multiple connections simultaneously or reuse one connection for all files requested from a particular web server. However, the initial performance can be poor, and many connections never get out of the slow-start regime, significantly increasing latency.

## IV. TCP OVER WIRELESS NETWORK

Communication over wireless links is characterized by limited bandwidth, high latency, high bit error rate and link connection and disconnection that can be handled by network protocol and applications. TCP congestion algorithm assumes that timeouts are caused by congestion, and not by transmission errors. This is well suitable for wired network but in wireless networks it is not suitable. TCP has been optimized for wired networks. Any packet loss is considered to be the result of network congestion and the congestion window size is reduced dramatically as a precaution. However, wireless links are known to experience sporadic and usually temporary losses due to fading, shadowing, hand off, and other radio effects, which cannot be considered as congestion, after the erroneous back-off of the congestion window size. For this reason, TCP falsely consider any packet loss in wireless transmission is due to congestion which triggers the congestion algorithm to reduce the window size to one segment and consequently reducing transmission speed and packet throughput due to wireless packet loss.

There can be a congestion avoidance phase with a conservative decrease in window size. This causes the radio link to be underutilized. For e.g. if the packet loss is 20% and if sender is sending 100 packets per second, the throughput is 80 packets per second, So due to packet loss TCP slows down data rate of packet transmission. If sender sends 60 packets per second after slow down data rate then the throughput is 48 packets per second [3]. As the TCP congestion algorithm was not initially designed to support the error-prone wireless network, but for very reliable wired network, it is impossible for the sender to differentiate between congestion loss and error loss. As a result, in timeout situations over wireless networks, the TCP often makes the wrong decision by slowing down the burst of packets while it should instead retransmit lost packets.

## V. EXISTING TCP CONGESTION CONTROL TECHNIQUES OVER WIRELESS NETWORKS

*A. Indirect TCP (I-TCP):* I-TCP is a transport layer protocol for mobile host which is based on the indirect protocol model. I-TCP is fully compatible with TCP/IP on fixed network. It works as a transport layer connection and establish two different connections between mobile host and fixed host. One over wireless medium and another over fixed network. It separates the flow control and congestion control performance on wireless link that on wired link. While in TCP/IP if any wireless node wants to communicate with wired network at that time it have to display or show the mobile support routers (MSR) [4]. However, in working of TCP in wired network while communicating with another wired network that time for communication both the network have to publish their IP's and route table for better communication.

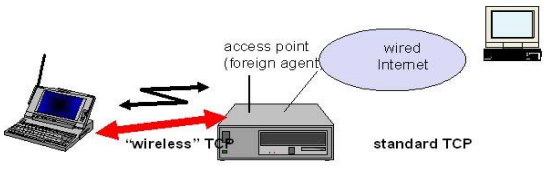


Fig.3. I-TCP

Same as when mobile host want to communicate with the fixed host then I-TCP sends one request to mobile support router to establish a TCP connection. In general the wired is between fixed host and a middleware station known as hub or proxy server, while wireless network is between hub or proxy server to mobile host.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 3, Issue 1 (February, 2016) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

So due to hub or proxy server the packet has to travel twice within communication and which affects on end-to-end transmission.

*B. Snoop:* Snoop is used for data transfer from fixed host to mobile host. Snoop keeps the record of unacknowledged packets. When packet loss is detected then Snoop retransmit the lost packet and in the record of unacknowledged packet Snoop keeps the record of fixed host and mobile host address, after retransmission of packet Snoop removes address of fixed host from retransmitted packet known as duplicate acknowledgment which prevent congestion control, while transfer of data if non sequence packet come Snoop marked as having congestion loss and forward message to mobile host. When acknowledgement is received from mobile host Snoop erase its record data and update its round trip time and forward acknowledgement to sender [5]. The lost packet are send on priority basis and Snoop erase the information of duplicate acknowledgement and retransmission of packet is done before any reflection from mobile host.

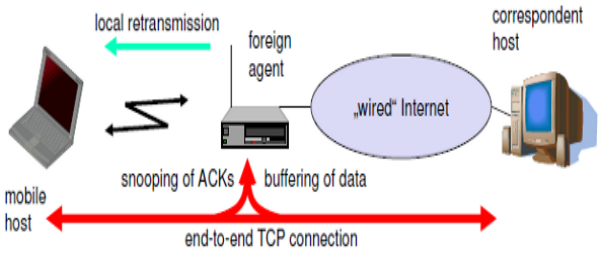


Fig.4. Snoop

While in case of communicating from mobile host to fixed host TCP will modify process of negative acknowledgement and then lost packets are retransmitted. Each mobile host is assigned with home address and temporary or bias IP address. Instead of home address packets the foreign address packets are transmitted to base station and that to mobile host and after acknowledgement receiver from base station it will proceed towards fixed host and works on the basis of FIFO mechanism.

*C. Multicast TCP (MTCP):* Multicast TCP is intended for low bit rate wireless links. In this each TCP connection is divided into two. One between supervisor host and fixed host and another connection is between fixed host and supervisor host. When the connection is established in between fixed host and mobile host first packet is received at supervisor host and then it forwarded to mobile host. The supervisor host will adjust the window size of network and works on the Round trip time period of acknowledgement. When acknowledgement is received from mobile host to supervisor host it will forwarded to fixed host with constant window size [6], if mobile host is disconnected or no any acknowledgment received to supervisor host then supervisor host will forward acknowledgement to fixed host and set window size to zero, after that fixed host stops its timer till it receives any further acknowledgement with nonzero window size.

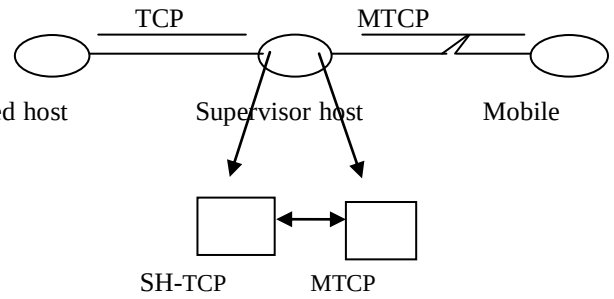


Fig.5. Setting up a TCP Connection.

After that supervisor host will set one suitable window size for remaining communication and restarts timer. As a result the sender resumes transmitting data at full speed. Because, no any acknowledgement from mobile host it will stop timer so it results in end-to-end semantic.

*D. Westwood:* Westwood is a new TCP protocol with a sender side modification of the window congestion control scheme. TCP Westwood controls the window using end-to-end transmission rate estimation which is public to both routers and destination. TCP Westwood will reset the window according to the allowable bandwidth. Hence, results in overcome of losses in wireless network due to this the source uses the available bandwidth and to use bandwidth estimation for faster recovery. Hence, results in high throughput [7]. In Westwood while communication between fixed host and mobile host it does not have any intermediate host that is hub or proxy server while communication. So due to this it is end-to-end schematic. TCP Westwood avoids reduction of window and threshold values. TCP Westwood does not require any support from lower layer. By receiving acknowledgement TCP Westwood continuously shows or estimates the packet data rate. Bandwidth estimate is equal to packet rate when data is delivered to the TCP receiver. After packet lost it will transfer data with slow threshold or slow data rate.

*E. Jitter TCP (JTCP):* There exist some unsolved problems in TCP over wireless network such as end-to-end congestion control and fair traffic or data transfer. So to overcome this to major problems a new TCP protocol works known as Jitter TCP. Jitter TCP is a TCP congestion control scheme to detect whether the packet drops are due to congestion or network error or bit errors. Basically the name jitter indicates as the time requires transferring a data from sender to receiver with packet lost period known as jitter. Main function is to calculate the packet-by-packet delay and jitter ratio that can be calculated through the overall jitter occurred while communication [8]. That is the spacing of packet at sender and spacing of packet at receiver in a frame. If the congestion occurs then Jitter TCP will keep packets in queue with longer transmission time and release after

International Journal of Electrical Electronics & Computer Science Engineering
Volume 3, Issue 1 (February, 2016) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

congestion problem overcomes. So the advantage of Jitter TCP is it provides end-to-end schematic with fairness. To achieve this result Jitter TCP will be implemented in transport layer in TCP, nodes and base stations.

*F. Wireless TCP (WTCP):* Wireless TCP is an end-to-end schematic mechanism, it works on inter arrival time of packets that is packet arrival time and packet delivery time. WTCP will compare packet arrival time with packet delivery time and able to distinguish the exact problem of packet loss either due to network loss or due to congestion [9]. WTCP uses rate based transmission control rather than window based transmission control. So it never allows to transfers burst of packets due to this more fairness will observed. WTCP will take ratio of inter packet arrival rate at receiver to that of inter packet arrival rate at sender. Like another TCP protocols WTCP does not slows down the data rate but by calculating the inter packet arrival time and packet delivery time WTCP will conclude to a better result and able to distinguish problem of packet loss. WTCP works as point-to-point mechanism by calculating packet rate at each node but it is different than original TCP so it has to implement at each node and base stations.

## VI. PROPOSED SOLUTION

*A. Overview:* In this paper, we propose a congestion prediction and controlled packet transfer rate for TCP/IP. Initially we define range according to buffer values as A1, A2 and A3. For example range varies according to buffer size as:

Buffer Values Considered in explanation

A1 is less than 256 kb

A2 is from 256 kb to 512 kb

A3 is from 512kb to 900 kb.

To determine the connection type (wireless or wired) Reserved bits (b) are set asb=1 for wired and b=0 for wireless and accordingly we define control packet transfer rates and forwarded it to sender as

A1=full ⟶ Packet will be sent at default rate

A2=80% ⟶ Packet sending rate will be reduced by 20%

A3=60% ⟶ Packet sending rate will be reduced by 40%

>A3=40% ⟶ Packet sending rate will be reduced by 60%

*B. Buffer Value Based Congestion Avoidance Algorithm:* At the time of communication, buffer monitoring mechanism is initialized and buffer size will be set to default according to bandwidth.

*First Stage A1:* When packets are received it will check buffer size, if it is less than 256 kb then it will transfer those packets to receiver with initializing congestion window (CWND) to 1 and also notify buffer status message to Control packet Transfer Rate Mechanism which eventually sends message to sender with controlled packet data rate status . If buffer value is equal to or more than 256 kb then it will sends message to increase buffer size up to 512 kb to buffer size set block. At this stage it will also start predicting the chances of congestion occurrence. As far as the receiving data rate is within the default buffer size limits it is considered as there is no congestion and controlled packet rate will be set to A1 i.e. data can be received at full or default speed.

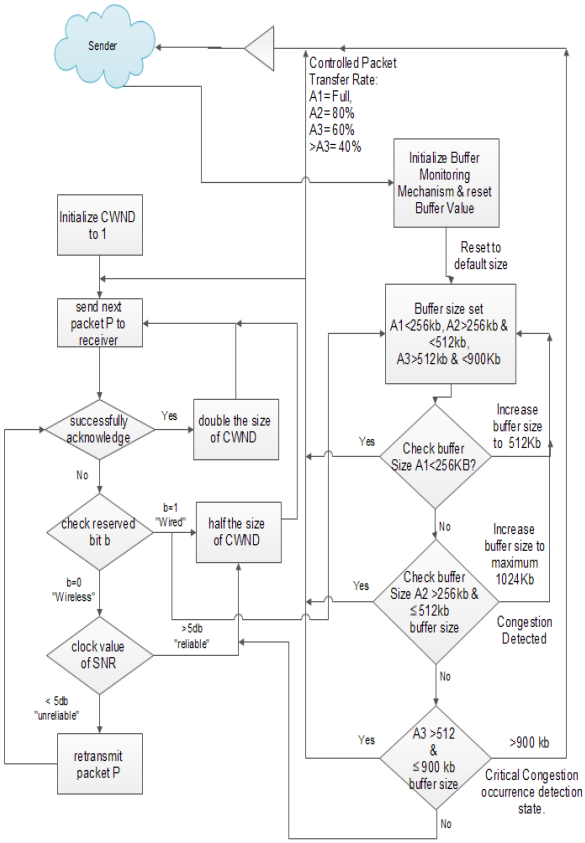


Fig.6. Proposed Congestion Detection and Avoidance Flowchart

*Second Stage A2:* If buffer value is more than 256 kb and less than 512 kb the packets will be sent to receiver with initializing congestion window (CWND) from its updates value and will also sends message to sender with updated buffer status to control packet data rate according to A2 i.e. reduce packet transfer rate by 20% and send 80% packets of current transfer rate as congestion is occurring. At a same time if the buffer value still keeps on increasing and is equal to or more than 512 kb then it will immediately request to increase the buffer size to its maximum value of 1024 kb to the buffer size set mechanism.

*Third Stage A3:* when the congestion occurs the buffer value keeps on overflowing and tends to happen packet

loss in this condition, if the buffer value is more than 512 kb and less than 900 kb, the packets will be sent to receiver as it is still under controllable congestion state and not exceeding its max threshold value of 900 kb, simultaneously notification message will be sent to sender to control data packet transfer rate to 60% and reduce data transfer rate by 40% of the current transfer rate. When the buffer size reaches equal to or more than 900 kb it will send notification to the sender to reduce the data transfer rate to 40% i.e. transfer rate will be lowered down by 60%. At this stage the mechanism identifies the critical congestion state has occurred and will send buffered data to half the size of CWND.

If packets are successfully acknowledged then it will double the congestion window and sends packets to receiver. If acknowledgement is not received then it will check for reserved bit 'b'. If b=1 means wired network then it will set the buffer size as per the type of link. If b is not equal to 1 then buffer size set mechanism assumes that buffer value has to set according to the wireless network for efficient communication system. If b=0 means wireless network then it will check value of SNR by following formulae.

SNR= Input power / Noise power

If the clock value of SNR is greater than 5 db means link is reliable and it will reduce to half the size of congestion window and sends packets to receiver. If it is less than 5 db means link is not reliable then it will retransmit those packets and check for successful acknowledgement. While operating in wireless environment this mechanism exploits the SNR ratio of the communication to decide whether a timed-out packet was due to congestion or error loss. When it identifies that the first bit of the reserved field is set according to the link type i.e. b=1 means wired connection and b=0 means wireless connection packet are sent to receiver, if acknowledge is received the CWND is increased by one segment doubling its size till the congestion occurs. When congestion occurs and packets are timed out the system will check the reserved bit b. If b=1 then it will consider packet loss is due to congestion and the size of the CWND will set to half and burst packets will be resumed. If b=0 then the connection is considered to be wireless and SNR ratio is checked if SNR ratio is less than 5 (SNR<5) then packet loss is considered to be because network error and the packets are retransmitted. If the SNR ratio is greater than 5 (SNR>5) the packet loss is considered due to congestion and CWND is reduces by half to resume the packets. The flowchart of proposed algorithm is depicted in Fig.6.

Buffer is reset accordingly when data transmission rate is set back within the A1, A2 and A3 slabs. Buffer monitoring is continuously done for immediate buffer size adjustments and to avoid Congestion and packet loss for any type of networks.

## VII. CONCLUSION

The proposed scheme efficiently uses Buffer Management System along with reserved field and SNR ratio to reduce packet loss during communication and to avoid congestion by reducing transmission speed. This system also avoids problem of congestion occurring in network. Need of packet retransmission due to packet loss is reduced by this proposed solution. It also reducesretransmission attempts which results in energy efficient communication. By using different controlled packet data rates the problem of occurrence of congestion is reduced. The scheme also identifies the cause of packet loss in wireless environment.

## VIII. REFERENCES

[1] Van Jacobson, and Michael J. Karels, "Congestion Avoidance and Control", Proceedings of ACM SIGCOMM '88, pp: 314-329, 1988.

[2] G. Xylomenos, G.C. Polyzos, P. Mahonen, and M. Saaranen, "TCP Performance Issues over Wireless Links", IEEE Communications Magazine, 2001.

[3] Ye Tian, Kai Xu, and Ansari N, "TCP in Wireless Environments: Problems and Solutions", IEEE Communications Magazine, vol. 43, no. 3, pp. 27-32, 2005.

[4] Bakre, Badrinath, "I-TCP: Indirect TCP for Mobile Hosts", In Proceedings of ICDCS 95, 1995.

[5] Hari Balakrishnan, Srinivasan Seshan, Elan Amir, and Randy H. Katz, "Improving TCP/IP Performance over Wireless Networks", Proceedings of the 1st ACM Conference on Mobile Computing and Networking, Berkeley, CA, November 1995.

[6] K. Brown and S. Singh, "M-TCP: TCP for Mobile Cellular Networks", ACM Computer Communications Review, vol. 27, no. 5, pp. 19 43, 1997.

[7] Ramakrishnan and Floyd, "A Proposal to add Explicit Congestion Notification (ECN) to IP", Internet Draft, January 1999.

[9] P. Sinha, N. Venkitaraman, R. Sivakumar, and V. Bharghavan, "TCP: A Reliable Transport Protocol for Wireless Wide-Area Networks", ACM Mobicom, Seattle, WA, 1999.

[10] Saverio Mascolo, Claudio Casetti, Mario Gerla, M. Y. Sanadidi, and Ren Wang, "TCP Westwood: Bandwidth Extimation for Enhanced Transport over Wireless Links", ACM Mobicom, 2001.

[11] Wu E.H.-K., and Mei-Zhen Chen, "JTCP: Jitter-based TCP for Heterogeneous Wireless Networks", Selected Areas in Communications, IEEE Journal, vol. 22, no. 4, pp. 757-766, 2004.