ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

A Novel Cross Layer Mechanism Using Clamp Protocol for Performance Enhancement of TCP variant

Sridhar P

Senior Lecturer, Department of Computer Engineering, IRT Polytechnic College, Chennai, Taminadu, India

ABSTRACT*:* The rapidly increasing importance of wireless communications together with the rapid growth of high speed networks, pose new challenges to transmission control protocol (TCP). To overcome them, a wide variety of TCP enhancements has been presented in the literature with different purposes and capabilities. Cross-layering represents a perspective design principle for adapting natively wired protocols to the wireless scenario and for improving their performance. In this paper, s a novel cross-layer approach (Link Layer CLAMP –TCP [L2-CLAMP-TCP]) designed for performance enhancement of TCP over a large variety of wireless networks is proposed. L2-CLAMP-TCP avoids TCP ACK packet transmission over the wireless channel, thus saving time and it can be utilized by the nodes for data packet delivery. The TCP ACK is generated at the base station itself. The congestion measure is also calculated at the base station based on which the receiver advertised window is calculated. The protocol performance is compared with existing TCP New RENO with CLAMP.

KEYWORDS*:* Access Point, Base Station, Cross Layer, Link Layer, Transmission Control Protocol

I. INTRODUCTION

The Transmission Control Protocol (TCP) is the dominant transport-layer protocol in wireless networks. Wireless network violates many assumptions made by TCP, causing degraded end-to-end performance. Wireless communication have some adverse characteristics like Channel Errors handoffs, node mobility, asymmetry, limited bandwidth, medium contention, hidden and exposed terminal problem, multi-path routing etc., compared to wired networks which will deteriorate TCP performance. So TCP needs some adaptations in order to be efficient on a wireless interface. TCP reliability is obtained through the utilization of a positive acknowledgement scheme which specifies TCP receiver to acknowledge data successfully received from the sender. Whenever a TCP segment is transmitted over the wireless link, it is acknowledged three times: one at the transport level and two times at the link layer. This results in a relevant performance reduction. Optimizing the acknowledgement scheme will obviously bring performance improvement through the reduction of medium-busy time and would require interaction between the transport and link layers –thus requiring proper cross-layering schemes.

The problem of network congestion is also considered as the reason for potential performance degradation. Congestion occurs when the amount of data exceeds the available capacity. The congestion window evolution is the key mechanism for TCP congestion control. To prevent congestion loss, an active queue management is required to avoid buffer overflow and also a fair scheduling is necessary to allocate bandwidth.

This paper target cross-layering as a possible solution, presenting a novel cross-layer approach, called Link Layer CLAMP (L2-CLAMP) TCP, where the acknowledgement at the receiver is suppressed and is generated at the base station. The congestion measure is also calculated at the base station and is attached with the acknowledgement. The main performance advantages are achieved through the optimization of interlayer automatic repeat request (ARQ) scheme functionality.

DOI:10.15680/IJIRCCE.2015.0311261

II. LITERATURE SURVEY

The proposals for TCP performance optimization over wireless networks target the root of the problem, i.e. TCP flow control and error recovery mechanisms, introducing different TCP modifications.

A well-known modification in the considered scenario is TCP Westwood [7] with its variations. An estimate of the available capacity on the end-to-end path is computed by using appropriate filtering of returning ACK flow. Then, upon loss detection, the outgoing rate is adjusted to fit the available bandwidth-delay product instead of performing blind window reduction.

TCP Veno [4] implements delay-based congestion control by estimating the number of backlogged packets in the bottleneck buffer – a technique originally proposed in TCP Vegas[14]. In addition, it tries to distinguish between congestion- and loss-related losses based on the estimated bottleneck buffer size preventing window reduction for non-congestion related losses.

On the contrary, flow control solutions adjust outgoing rate without providing reliability of data delivery, for real-time multimedia traffic applications. Proposals in this category can be implemented as an additional transport layer protocol or flow control above transport layer on top of unreliable UDP or RTP protocols.

The main drawback preventing deployment of the proposed TCP modifications and other rate control protocols is in the requirement for sender protocol stack modification which is difficult to perform in world-wide scale. This motivated multiple attempts to hide undesirable characteristics of the wireless links from the transport layer while keeping TCP sender’s stack unchanged.

One of the first of such attempts to mask losses on the wireless link from a fixed sender is proposed in [14], the snoop agent introduced at the base station performs local retransmissions triggered by sniffing duplicate ACKs coming from the wireless receiver. TCP ACK transmission over the wireless link can be avoided through local generation of ACKs at the sender node or at the base station, thus improving throughput [2]

A protocol CLAMP was suggested [1] [3] [4] on the receiver side and provides a separate queue at the AP. The CLAMP protocol removes the window fluctuations and achieves much better fairness than TCP NewReno. The four algorithms, Slow Start, Congestion Avoidance, Fast Retransmit and Fast Recovery are described [13].

The report is organized into 6 chapters

Section III describe the TCP-New Reno variant

Section IV deals with the architecture of the proposed, cross layer L2-CLAMP agent and the receiver agent algorithm.

Section V specifies the simulation requirements, analyses the simulation results, describes the parameters used for the simulation process, specifies the performance metrics, performance obtained presented as Graphs and analysis of the result based on the graphical data

Section VI describes the conclusion and future work towards the enhancement of the existing design for better performance improvement.

III. TCP NEW RENO WITH CLAMP

TCP New RENO is a variant of TCP-RENO. It is able to detect multiple packet losses and thus is more efficient than RENO in the event of multiple packet losses. New-Reno enters into fast-retransmit when it receives multiple duplicate packets, but it differs from RENO in that it doesn’t exit fast-recovery until all the data which was out standing at the time it entered fast recovery is acknowledged. Thus it overcomes the problem faced by Reno of reducing the CWND multiple times.

CLAMP is an algorithm that controls the receiver AWND based on feedback from the AP. This algorithm runs at the mobile device (the receiver) and acknowledgments (ACKs) are sent back to the sender containing the new values of AWND as calculated by CLAMP. Since all versions of TCP interpret the AWND as an upper limit on the allowable window size, which is a mechanism to avoid an overflow of the receiver buffer, this provides an effective method of control provided that the AWND value is smaller than the CWND value calculated by the sender.

IV. PROPOSED PROTOCOL (L2-CLAMP-TCP)

A. L2-CLAMP Architecture

The proposed scheme enhances the protocol stacks of the wireless sender (or a base station /Access Point) and the receiver with Link Layer -CLAMP (L^2CLAMP) agents which support ACK suppression. The network scenario taken is infrastructure based and is shown in Fig. 1

The scenario consists of four sources (S1, S2, S3 and S4) and three mobile receivers (MN1, MN2 and MN3). Four different flows F1, F2, F3 and F4 are shown in Fig. 1. Flow F1 is between source S1 and Mobile node MN1. F2 is between S2 and MN2 and the flows F3 and F4 starting from source S3 and S4 respectively reach the same mobile node MN3 through the access point.

The L2-CLAMP agent suppresses the outgoing L2-CLAMPTCP ACKs at the receiver side and generates them locally at the sender or base station. The basic idea behind approach is to shift TCP ACK generation point from mobile receiver to the base station. L2-CLAMP -TCP requires implementation of a software module, called L2-CLAMP agent, inside base station / access point (BS / AP) protocol stack above the link layer. The L2-CLAMP agent sniffs the ingress traffic from the fixed network assuming to have access to the network and transport layer headers.

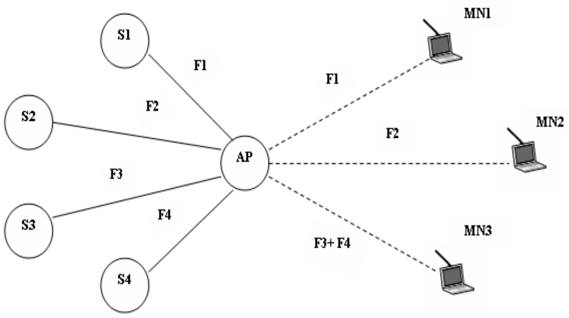


Fig1 - Infrastructure Network

The L2-CLAMP agent suppresses the outgoing L2-CLAMP-TCP ACKs at the receiver side and generates them locally at the sender or base station. The basic idea behind approach is to shift TCP ACK generation point from mobile receiver to the base station. L2-CLAMP -TCP requires implementation of a software module, called L2-CLAMP agent, inside base station / access point (BS / AP) protocol stack above the link layer. The L2-CLAMP agent sniffs the ingress traffic from the fixed network assuming to have access to the network and transport layer headers. Whenever a TCP data packet is detected, the L2-CLAMP Agent in the BA / AP, performs service rate, congestion measurement, received rate and calculates the AWND value, stores flow-related information such as flow sequence number carried by the packet, port numbers, ACK flag and ACK sequence number, along with the AWND value. L2-CLAMP agent calculates the desired size of the AWND window based on the values of congestion measure, received rate and RTT. To calculate congestion measure, a separate Sync packet is sent from the base station to the receiver and the acknowledgement for the Sync is received back by the sender, based on which the transmission time for each packet is calculated. This is done before actual TCP packet transmission.

The parameters used for calculation of congestion measure is b, q,α, μ_c

$$p(q) = (bq - \alpha) / \mu_c \qquad (1)$$

where b = 1 for safer purpose. $\tau$is the parameter (fixed size in bytes) used to calculate the value of AWND and α is used to calculate the congestion price signal. Similarly α is free to be chosen by the AP, but is fixed once chosen.

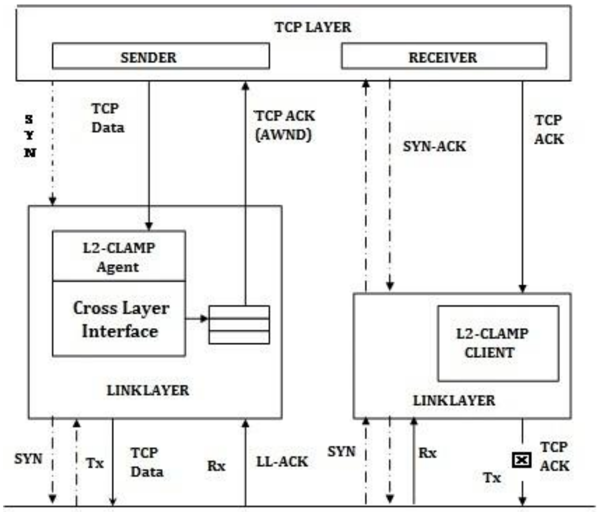


Fig 2 - Architecture of the Proposed System

On the receiver side, module referred as L2-CLAMP client in Fig. 2silently drops all standalone non-duplicate TCP ACK packets. The generated TCP ACK packet is then forwarded to the sender. The AP maintains a separate queue for each receiver and acts as a router, routing the packets to the destined MN based on the address of the mobile. The TCP flows destined for the respective MN are put into the same queue.

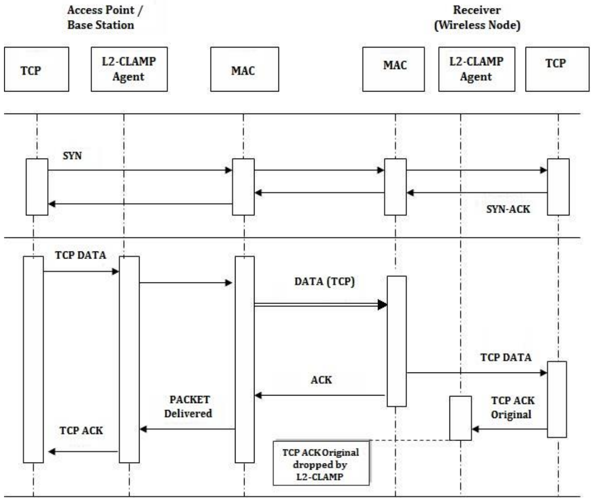


Fig 3 - Sequence of Process for Packet Delivery

In the scenario taken, a separate queue is maintained for flows F1 and F2 as they are destined to different mobile nodes MN1 and MN2. The flows F3 and F4 share a same queue as they are destined to the same mobile receiver MN3. A scheduler incorporated in the access point takes care of scheduling the two flows. The scheduler is based on the principle of multi-user diversity. At each slot, a new scheduling decision is made, and the general preference is for scheduling a mobile that is in a good channel state. At each slot, the scheduler picks the queue i with the largest utility that has data to send in this slot. The utility U_i is calculated as

$$U_i = \mu_i / r_i \quad (2)$$

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

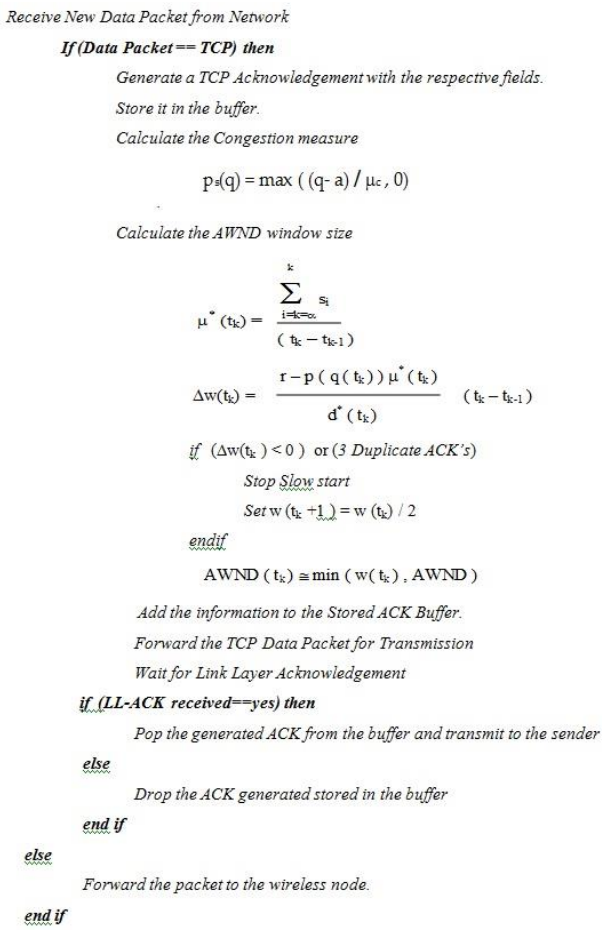
(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

whereμ_i is the is the current rate for the mobile i at the beginning of the slot andr_i is the exponential moving average of the rate obtained by mobile i with an averaging time constant of 100 ms.

Figure 3 represent the sequence process of packet delivery using L2-CLAMP-TCP.

B. Algorithm



```
Receive New Data Packet from Network
    If (Data Packet == TCP) then
        Generate a TCP Acknowledgement with the respective fields.
        Store it in the buffer.
        Calculate the Congestion measure
            $p_s(q) = \max(\, (q- a) / \mu_c , 0)$
        Calculate the AWND window size
            $\mu^*(t_k) = \frac{\sum_{i=k=\alpha}^{k} s_i}{(t_k - t_{k-1})}$
            $\Delta w(t_k) = \frac{r - p(q(t_k))\,\mu^*(t_k)}{d^*(t_k)} \, (t_k - t_{k-1})$
            if $(\Delta w(t_k) < 0)$ or (3 Duplicate ACK's)
                Stop Slow start
                Set $w(t_k + 1) = w(t_k) / 2$
            endif
            $AWND(t_k) \cong \min(w(t_k), AWND)$
         Add the information to the Stored ACK Buffer.
         Forward the TCP Data Packet for Transmission
         Wait for Link Layer Acknowledgement
      if (LL-ACK received==yes) then
         Pop the generated ACK from the buffer and transmit to the sender
      else
         Drop the ACK generated stored in the buffer
      end if
  else
      Forward the packet to the wireless node.
  end if
```

V. PERFORMANCE EVALUATION

To observe how TCP performance is influenced by our L2-CLAMP approach, it is compared with existing TCP New Reno with CLAMP. The simulation is carried with Network Simulator. The simulation parameters are shown in Table 1.

The simulation is carried with Network Simulator-2. The simulation parameters are shown in Table 1. The simulation is carried out considering the various simulation parameters- packet size, queue delay, simulation time and packet error rate.

A. PERFORMANCE METRICS

- *Throughput*

Throughput is defined as the number of bits passing through a point in a second. On extending from bits to packets and from a point to a network, it is defined as the number of packets passing through the network in a unit of time. The unit of throughput is Mb/s. Throughput is represented as

$$T = b / time * 8 / 1000000 \qquad (3)$$

where b = bandwidth (in Mb/s)

- *End-to-end delay*

End-to-end delay refers to the time taken for a packet to be transmitted across a network from source to destination. End-to-End delay is measured in s.

$$d_{end\text{-}end} = N[\, d_{trans} + d_{prop} + d_{proc} \,] \qquad (4)$$

where

dend-end= end-to-end delay
dtrans= transmission delay
dprop= propagation delay
dproc= processing delay

- *Queue Delay*

The queueing delay is the time a job waits in a queue until it can be executed. It is a key component of network delay. The maximum queuing delay is proportional to buffer size. It is measured in milliseconds.

- *Packet Error Ratio*

Packet Error Rate (PER) is used to test the performance of a receiver. PER is the ratio of the number packets not successfully received by the sink nodes to the number of packets sent by source nodes.

- *Simulation Time*

Time duration for determining the service availability or unavailability of end-to-end connections between source and sink nodes. It is measured in seconds.

Table -5.1 Simulation parameters

S.No	Parameter	Value
1	Transmission Range	75 m
2	Propagation Model	Two-Ray ground
3	Wire Nodes	4
4	Base Station	1
5	Wireless nodes	10
6	Packet Size	250 – 1250 Bytes
7	Error Rate (Induced in Wireless Scenario)	0.05 – 0.25
8	Queue Delay	20 – 100 ms
9	Simulation Time	100 – 500 s
10	Propagation Delay (Wired)	20ms
11	Capacity (Wired)	0.5 Mbps
12	Protocols	CLAMP, L2- CLAMP

B. SIMULATION RESULTS

With packet size as 1250 bytes and the delay in queue as 100 ms, simulation is carried over by varying the time between 100s and 500s. The results obtained shows that the L2-CLAMP protocol has better improvement over the CLAMP protocol.

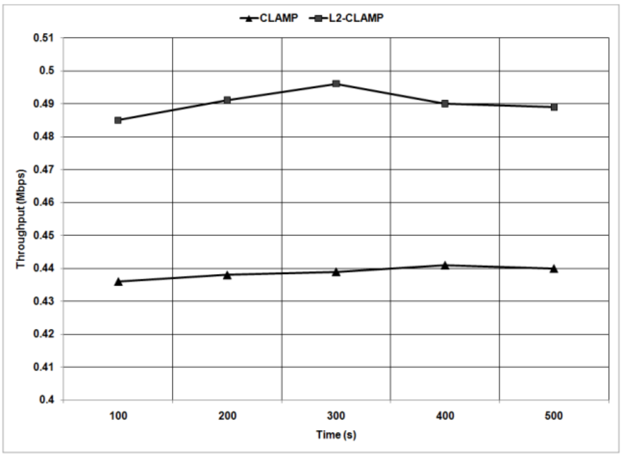


Fig4- Simulation Time Vs Throughput

Figure 4shows that the throughput of L2-CLAMP is better by 0.05 – 0.06 Mbps than that of the CLAMP.

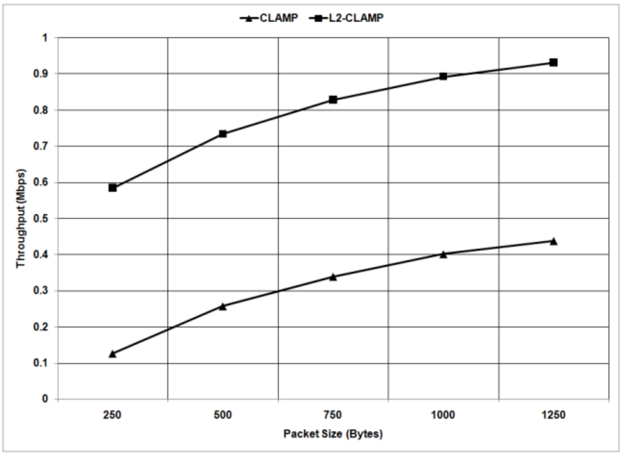


Fig5 - Packet Size Vs Throughput

Simulation is carried over by varying the packet size from 250 bytes to 1250 bytes with a queue delay of 100 ms for a duration of 200s.. The results obtained are shown above in Fig. 4.

Figure 5 shows the throughput for varying packet size. The throughput of the L2-CLAMP is 5 times higher than that of the CLAMP between the packet sizes of 250 – 1250 bytes. The difference in throughput is nearly 0.4 Mbps – 0.5 Mbps between the packet size of 250 bytes and 1250 bytes.

Figure 6 shows the Queue delay with respect to throughput. The results obtained shows that the L2-CLAMP protocol has better improvement over the CLAMP protocol.Performance is also analyzed with respect to Packet Error Rate. The error rate is varied between 0.05 to 0.25 with packet size of 1250 bytes and queue delay of 100 ms.

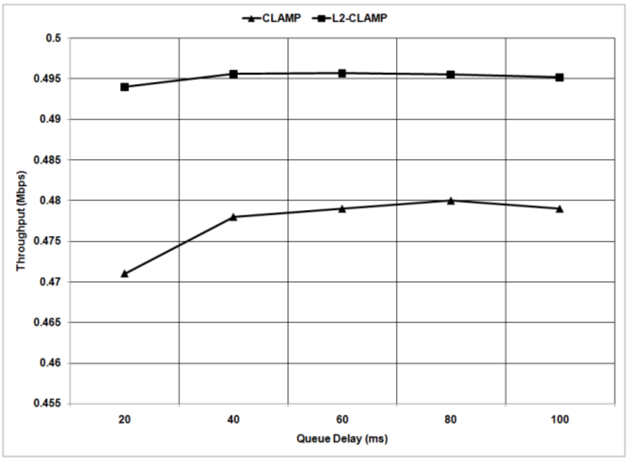


Fig 6- Queue Delay Vs Throughput

Figure 7 represent the error ratio with respect to throughput and delay for a packet size of 1250 bytes with a delay of 100ms and simulation time of 200s. The throughput of the L2-CLAMP proportionally decreases with increase in Packet Error Ratio (PER).

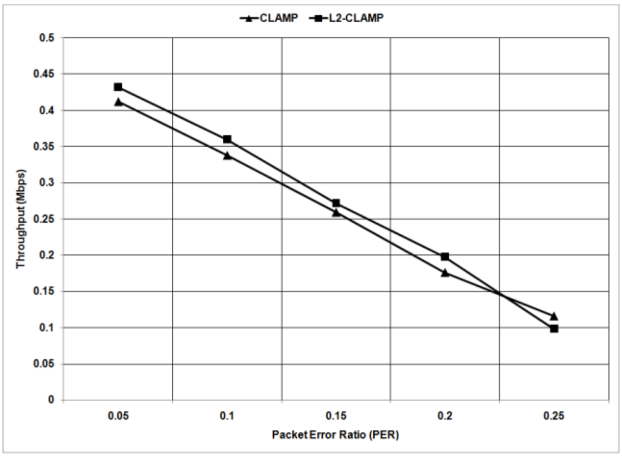


Fig7- Packet Error Rate Vs Throughput

DOI:10.15680/IJIRCCE.2015.0311261

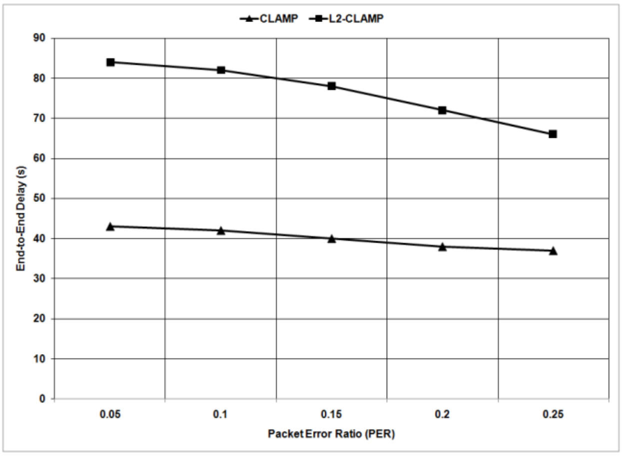


Fig 8- Packet Error Rate Vs Delay

Figure 8 represent PER with respect to delay. The delay decreases with increase in PER in L2- CLAMP. Figure9 represents the available packets in the queue for given the simulation time of 500s.The queue can hold a maximum of 50 packets. The L2CLAMP holds more packets in the queue without making it empty.

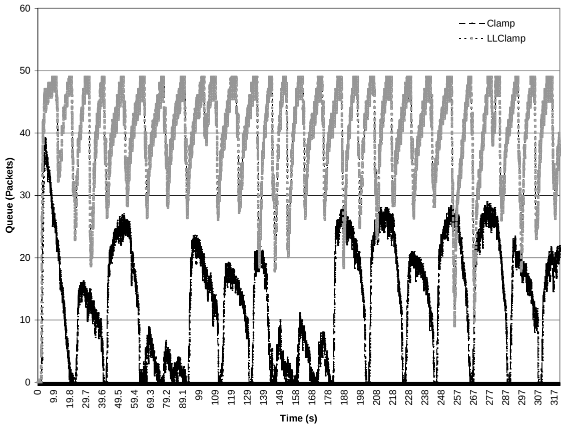


Fig. 9 - Simulation Time Vs Queue

Figure 10 shows the congestion window for the given simulation time. The L2CLAMP reduces the congestion window size whenever it reaches the specified threshold or whenever packet loss occurs and also maintains the queue not be empty.

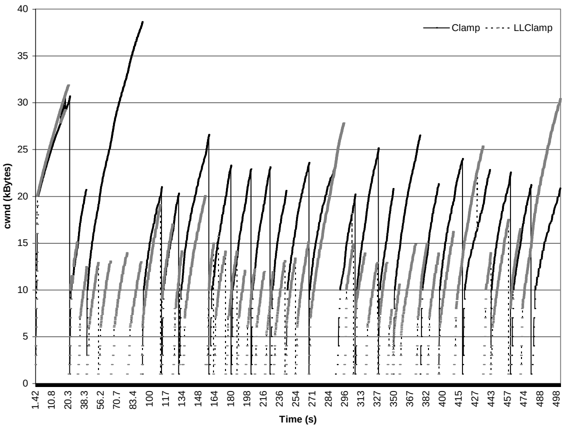


Fig. 10 - Simulation Time Vs Congestion Window

V. CONCLUSION AND FUTURE WORK

This paper presented a novel approach forperformance improvement of TCP over wireless networks.Performance improvement comes from cross-layeroptimization. The proposed solution, L2-CLAMP, avoids TCPACK transmission over the wireless link through localgeneration of ACKs at the sender node or at the base station.The congestion measure is also calculated at the base stationbased on which the receiver advertised window is calculated.The protocol schedules the flow to the destined receiver. The protocol performance is compared with existing TCPNewReno with CLAMP.

The protocol performance is compared with existing TCP NewRenowith CLAMP.From these experiments it is concluded that the results has shown better performance with respect to all the parameters. In the scenarios tested L2-CLAMP has achieved better fairness in error free and error prone environment. The scheduling is done properly by allocating required bandwidth to the users. On employing L2-CLAMP in AP, it is found that the scheduler is able to achieve better fairness than TCP NewReno with CLAMP.

A flexible on/off switch to operate the modules depending upon the operating capacity of the AP can be provided. Thesimulations can also be extended for more number of mobilenodes.

REFERENCES

1. Lachlan L.H. Andrew, Stephen V. Hanly,Rami G. Mukhtar "Active Queue Management for Fair Resource Allocation in Wireless Networks", IEEE transactions on Mobile Computing, vol. 7, no. 2, pp. 231-246, February 2008
2. DzmitryKliazovich, FabrizioGranelli, Mario Gerla, "Performance improvement in wireless networks using cross-layer ARQ", Computer Networks 51 (2007) pp. 4396–4411
3. L.L.H. Andrew, S.V. Hanly, and R.G. Mukhtar, "CLAMP: Maximizing the Performance of TCP over Low Bandwidth Variable Rate Access Links," technical report, Univ. of Melbourne, http://www.ee.unimelb.edu.au/people/hanly/T_Reports/clamp_tr-2004-02-01.pdf, 2007.
4. C.P. Fu, S.C. Liew, "TCP Veno: TCP enhancement for transmission over wireless access networks", IEEE JSAC 21 (2) (2004) pp. 216–228.
5. A. Eryilmaz and R. Srikant, "Fair Resource Allocation in Wireless Networks Using Queue-Length-Based Scheduling," Proc. IEEE INFOCOM, vol. 3, pp. 1794-1803, Mar. 2005
6. R. Kapoor, L. Chen, M. Sanadidi, M. Gerla, "CapProbe: a simple and accurate technique to measure path capacity", Technical Report TR040001, UCLA SCD, 2004.
7. L.A. Grieco, S. Mascolo, "Performance evaluation and comparison of Westwood+, New Reno, and Vegas TCP congestion control", ACM Computing and Communication Review 34 (2004) pp. 25–38.

8. L.L. Andrew, S.V. Hanly, and R. Mukhtar, “CLAMP: A System to Enhance the Performance of Wireless Access Networks,” Proc. IEEE Global Telecomm. Conf. (GLOBECOM ’03), vol. 7, pp. 4142- 4147, Dec. 2003.
9. L.L.H. Andrew, S.V. Hanly, and R.G. Mukhtar, “CLAMP: Differentiated Capacity Allocation in Access Networks,” Proc. 22nd IEEE Int’l Performance Computing and Comm. Conf. (IPCCC ’03), pp. 451-458 Apr 2003
10. Qi Wang, M.A. Abu-Rgheff, “Cross-layer signaling for next generation wireless systems”, in: IEEE Wireless Communications and Networking Conference (WCNC 2003), vol. 2, pp. 1084–1089, March 2003
11. K. Ramakrishnan, S. Floyd, D. Black, “The Addition of Explicit Congestion Notification (ECN) to IP, Request for Comments (RFC)”, September 2001
12. S. Xu, T. Saadawi, “Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?”, IEEE Communications Magazine 39, pp 130–137, June 2001.
13. W. Stevens, “TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms, Request for Comment RFC 2001”, January 1997.
14. L. Brakmo, L. Peterson, “TCP Vegas: end to end congestion avoidance on a global Internet”, IEEE JSAC 13 (8) (1995) pp. 1465–1480.
15. H. Balakrishnan, S. Seshan, E. Amir, R. Katz, “Improving TCP/IP performance over wireless networks”, in: Proceedings of the 1st ACM International Conference on Mobile Computing and Networking (MOBICOM), Berkeley, CA, November, 1995.
16. Sally Floyd,”TCP and Explicit Congestion Notification”, ACM Computer Communications Review,pp. 10-23, October 1994
17. W. Richard Stevens. TCP/IP Illustrated, Volume 1: The Protocols. Addison Wesley, ISBN: 0-201-63346-9. January 1994.
18. NS-2 simulator tool home page, Available from: http://www.isi.edu /nsnam/ns