

# Improving Throughput of Transmission Control Protocol Using Cross Layer Approach

Jansi Rani Sella Veluswami\*

Sri Sivasubramaniya Nadar College of Engineering, Chennai, 603103, India

\*Corresponding Author: Jansi Rani Sella Veluswami. Email: svjansi@ssn.edu.in

Received: 16 May 2021; Accepted: 02 July 2021

**Abstract:** Most of the internet users connect through wireless networks. Major part of internet traffic is carried by Transmission Control Protocol (TCP). It has some design constraints while operated across wireless networks. TCP is the traditional predominant protocol designed for wired networks. To control congestion in the network, TCP used acknowledgment to delivery of packets by the end host. In wired network, packet loss signals congestion in the network. But rather in wireless networks, loss is mainly because of the wireless characteristics such as fading, signal strength etc. When a packet travels across wired and wireless networks, TCP congestion control theory faces problem during handshake between them. This paper focuses on finding this misinterpretation of the losses using cross layer approach. This paper focuses on increasing bandwidth usage by improving TCP throughput in wireless environments using cross layer approach and hence named the proposed system as CRLTCP. TCP misinterprets wireless loss as congestion loss and unnecessarily reduces congestion window size. Using the signal strength and frame error rate, the type of loss is identified and accordingly the response of TCP is modified. The results show that there is a significant improvement in the throughput of proposed TCP upon which bandwidth usage is increased.

**Keywords:** Bandwidth; wireless networks; transmission control protocol; throughput

## 1 Introduction

Transmission Control Protocol (TCP) is widely used in Internet. TCP is a reliable, connection oriented and end to end protocol. Originally TCP is designed for wired networks. The main functionality of TCP is to control congestion in the network. The reliability is ensured by the acknowledgement sent by the receiver for each packet delivered properly. Packet loss will be recognized using the duplicate acknowledgment or timeout mechanism. This packet loss happens due to congestion in the wired network. Current technological world demands connection between wireless networks to wired network. In wireless networks, packet losses happen due to some wireless characteristics mostly such as signal fading, interference etc. But, the sender on the wired part assumes that packet loss was due to congestion which is in contradiction to the actual fact. Since the non-congestion losses are misinterpreted to be congestion



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

loss [1], there results unnecessary fall in throughput. And, hence even though the bandwidth is available, the bandwidth is underutilized. This work focuses on the bridging this gap.

The packet loss in wireless networks [2] may be a collision loss, random loss or frame loss and it experiences very negligible congestion loss. The data frame may be lost because of collision occurring during competing for channel. The other reason that is considered in this approach is that the frame could have been lost due to transmission error caused by wireless characteristics. Random loss means data will be lost since the signal is very weak. The idea behind this approach is to handle the receiver's reaction according to the type of loss.

The motivation behind this work is due to the proliferation of the mobile computing devices has led to the usage of wireless technology. Whenever wired networks and wireless network collaborate, this problem creeps in. Hence, there is a standing need to improve the performance of Transmission Control Protocol to increase the bandwidth utilization.

The paper is organized as follows. Section 2 presents a survey of related TCP variants. Section 3 gives a detailed design of CRLTCP and Section 4 contains the Results and Discussions.

## 2 Related Work

The research on TCP dates back to two decades since the inception of internet. In the literature had attempted to improve TCP in various directions such as congestion control, delay, and data rate etc., using cross layer approach as in [3].

Reno [4] is the baseline protocol used to adjust congestion window based on the standard TCP procedure but reacts fast when packet loss occurs. TCP Westwood [5] protocol use acknowledgment packets and calculates bandwidth. Based on the calculated bandwidth, it adjusts congestion window. An innovative Jitter based cross layer TCP is developed in [6] based on jitter ratio. But this needs lot of computational power. The modifications of link layer is done in [7]. Some machine learning based loss differentiation is done in [8]. This system aims at using signal strength information to identify wireless losses. Bandwidth is controlled by radio condition over time by developing new window control algorithm in [9]. The self-adaptive contention window is done in [10] based on the QOS parameters.

All cloud services uses mostly TCP connection. The reliable cloud transmission suffers from wireless interference. A Markov chain based analytical model is developed to address wired congestion and wireless radio interference [11]. Ben has also developed cross layer based congestion control for collaborating between transport and mac layer for decision passing to fix contention window value [12]. Many of the schemes involving cross layer mechanisms are discussed in [13]. The contention window management is done using RTS parameters in the receiver side for MANETs [14]. This work focuses only on contention losses. In general, there is a gap in recognizing frame and collision losses and react accordingly in all the above said works. Stochastic modeling of TCP is discussed in [15]. The cross layer approach was adopted in [16] is extended in this paper with mathematical modeling and more investigations are carried out with different performance measures to strengthen the algorithm. In this proposed method CRLTCP, as said earlier collision, frame and random losses are addressed using cross layer approach and accordingly informs the sender.

## 3 System Design

The proposed system CRLTCP aims at increasing bandwidth usage by improving TCP using cross layer approach where the information of link layer is sent to the transport layer for further processing. The sender node is assumed to be in wired network and receiver node in wireless network. Base station is deployed with snoop agent which caches all the data packets passed through it as shown in Fig. 1. Whenever the base station sees an acknowledgment for the cached packet, it deletes the corresponding data packet from the cache. It records the last acknowledgment also.

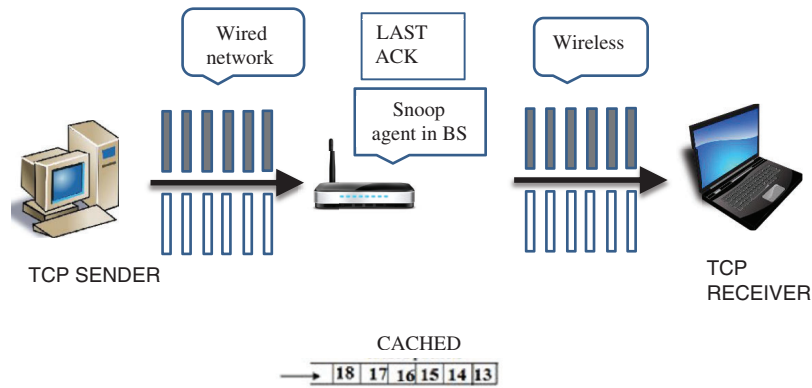


Figure 1: System architecture of CRLTCP

3.1 CRLTCP Algorithm

As depicted in Fig. 2, a new packet is checked to be in order. If it is in order, then the data packet is cached and forwarded, else the sender is informed about the wired loss. If the packet is found to be in out of order, then sequence number is checked against the last acknowledgement saved. If sequence number is larger then, probably ack is lost in wired link itself; else, probability of collision and frame error rate is compared. If there is more contention for medium, then contention window value is doubled and it is identified as collision loss. If not, received signal strength is compared with receive threshold and carrier sense threshold. The frame loss is identified if signal strength is higher else as random loss. This random loss will be intimated to sender in ACK frame.

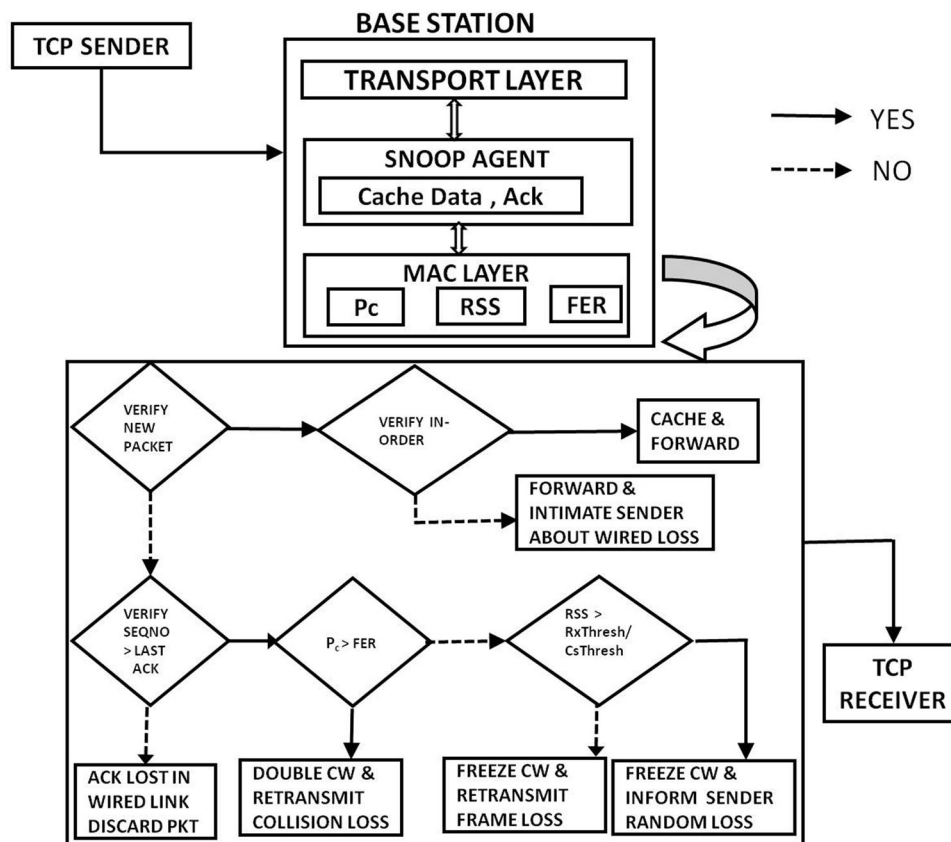


Figure 2: Process flow diagram-CRLTCP

### 3.2 Identifying Losses

The prime objective is to identify the different types of losses as collision loss, frame loss and random loss. As mentioned in the algorithm, random loss is identified using frame error rate which is the direct parameter. For random loss, RxThresh and CsThresh are also direct measurable parameter obtained from the data link layer. The Friss propagation is used to calculate the received signal power as in [17]. The collision loss can be identified by finding the probability of transmission failure ( $P_f$ ).

$$P_f = \frac{\text{Number of Transmission Failures}}{\text{Number of Transmission Attempts}} \quad (1)$$

The update of collision probability happens at the end of each transmission because the channel keeps on changing. Hence, this could reflect the actual status of the channel. Run-time measurement is considered to avoid the undesirable behaviour of rapidly fluctuating channel by using the formula. The runtime estimation of channel status is done to avoid the undesirable behaviour of rapidly fluctuating channel by using the formula

$$P_f = \alpha * P_f + (1 - \alpha)P_{f(\text{Estimated})} \quad (2)$$

where  $\alpha$  is a smoothing factor, ranges from [0, 1]. The value of  $\alpha$  is set as 0.4 to obtain new probability of transmission failure. Where the new value of  $p_f$  is a weighted combination of the previous value of  $p_f$  and the new value for estimated  $p_f$ , and [0, 1] is a smoothing factor.

The probability of transmission failure can be in another way defined as probability of a transmitted frame is received in error or encounters collision.

$$P_f = 1 - (1 - P_c) * (1 - FER) \quad (3)$$

Each node can independently monitor the channel and can calculate this parameter.

### 3.3 Mathematical Modeling

Considering the queuing systems, it is assumed that the packets arrive in Poisson arrival pattern with arrival rate  $\lambda$ , and the service time probability is assumed to be  $1/\mu$ .

The following variables and notations are used:

|                  |   |
|------------------|---|
| $\mu$            | Packet departure rate                           |
| $\lambda$        | Packet arrival rate                             |
| $WT_{BS}$        | Average waiting time in base station cache      |
| $N(P)_{BS}$      | Number of packets waiting in base station cache |
| $T_{sl}$         | Time required for each slot                     |
| $T_S, T_D$       | Duration of short interframe, DCF interframe    |
| $T_{PC}, T_{PH}$ | Duration of a PHCP preamble and header          |
| $CW$             | Minimum back off contention window size         |
| $\overline{CW}$  | Average back off contention window size         |
| $W_{TCP}$        | congestion window size                          |
| $DR_{data}$      | Maximum data rate in wireless LAN               |
| $S_D, S_A$       | Payload size, Acknowledgment frame size         |
| $\tau$           | Propagation delay                               |

$P_n$  Probability of  $n$  packets in the system

$\rho$  System utilization factor,  $0 < \rho < 1$

It is assumed that overall system utilization is assumed to be less than 1 i.e.,  $\rho < 1$ . The steady state probabilities is formulated as,

$$p_{n+1} = \rho p_n, \quad 0 < n < \infty \quad (4)$$

The system utilization factor [18] is given by,

$$\rho = \frac{\lambda}{\mu} \quad (5)$$

$$\rho = \frac{NW}{RTT} (2T_D + \overline{CW} + \frac{S_{D+S_A} + S_{TA}}{DR_{data}} + T_S + 3(T_{PC} + T_{PH} + \tau)) \quad (6)$$

It follows from Eq. (4),

$$p_{n+1} = \rho^{n+1} p_0, \quad 0 < n < \infty \quad (7)$$

All probabilities add up to 1, so

$$1 = \sum_{n=0}^{\infty} p^n = \sum_{n=0}^{\infty} \rho^n p_0 = \frac{p_0}{1 - \rho} \quad (8)$$

The average number of packets in the system is,

$$\rho = \frac{\rho}{\mu - \rho} \quad (9)$$

By little's theorem average waiting time of packets is,

$$T = \frac{1}{\mu - \lambda} \quad (10)$$

$$WT_{BS} = \frac{\rho}{\mu - \lambda} \quad (11)$$

$$N(P)_{BS} = \frac{\rho^2}{1 - \rho} \quad (12)$$

Substitute the value of  $\rho$  from Eq. (6) in Eqs. (11), (12). The average contention window size is calculated as,

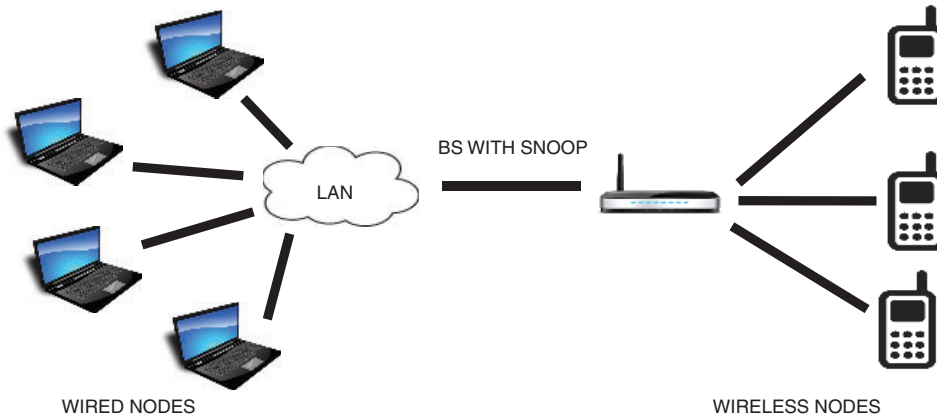
$$\overline{CW} = CW \frac{(1 - P_f - P_f(2P_f)^m)}{1 - 2P_f} \quad (13)$$

The contention window is calculated using this formula and gets doubled when needed.

#### 4 Results and Discussions

The implementation is done using Network Simulator (NS2) [19]. The simulation is carried out by considering the various simulation parameters such as packet size, delay bandwidth and time and uses the topology as shown in Fig. 3. It uses energy model to determine the signal strength of each node. The simulation of a network is written in TCL scripting language. For simulation, the total number of wired

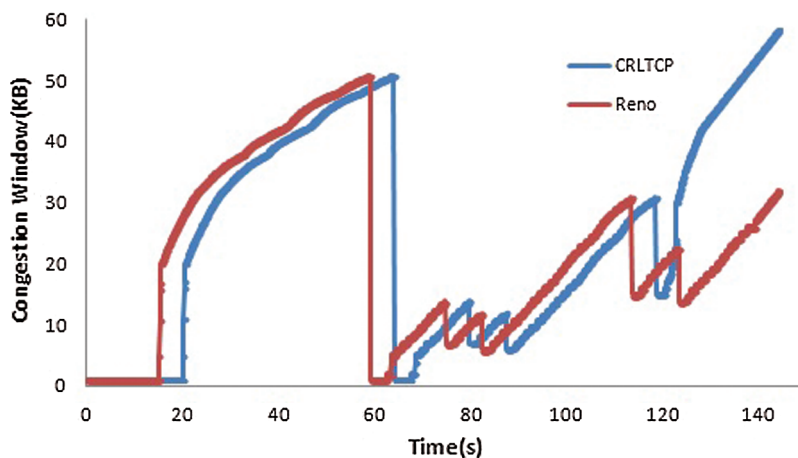
nodes considered is 10 and wireless node is 50. The simulation time is varied between 120 to 300 s. The various performance metrics are evaluated as in [20].



**Figure 3:** Simulation-base topology

#### 4.1 Congestion Window

The congestion window limits the amount of data the TCP can send into the network before receiving an acknowledgment. Congestion Window Dynamics represents how the window gets adjusted based on the state of the network. When there is a loss, it could be inferred from the congestion window drop. In Fig. 4, at 60 s there occurs a drop. Both CRLTCP and Reno moves to slow start phase as a consequence of that drop and re initializes the congestion window. But at 120 s, CRLTCP does not fall to slow start since it could recognize this loss as wireless loss. Hence there will be an improvement in the throughput of CRLTCP compared to Reno.



**Figure 4:** Congestion window dynamics

#### 4.2 Sequence Number

Transmission Control Protocol assigns a number to each and every segment that it transmits. This number is called sequence number. There are some breaks visualized in sequence number at 60 and 120 s due to losses. At 60 s, both the protocols behave in the same way. But at 120 s, Reno break space is

greater than CRLTCP. It performs better because it intimates about the wireless loss happening to the sender. Or in other words, CRLTCP quickly recognizes there is no need to reduce the sending rate and hence it resumes.

### 4.3 Fairness

Fairness is defined as whether the network resources are receiving a fair share of the system resources. Fairness aims at receiving fair-share of the bottleneck link capacity among the N connections sharing the same link. Jain's index is used to find the fairness ratio as said earlier. A graph is drawn by varying the number of connections between 1–12. It is inferred from Fig. 5, that CRLTCP allocates fair share of the bandwidth among the other flows. Hence bandwidth usage is increased since all protocols gets fair share of the bandwidth.

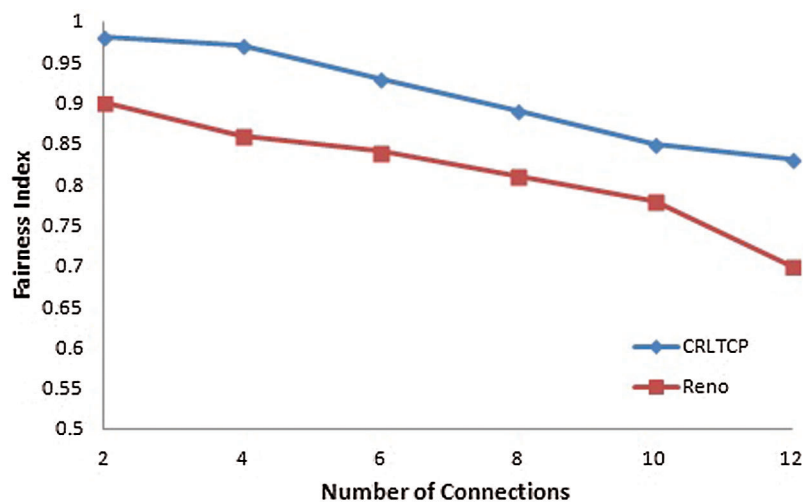
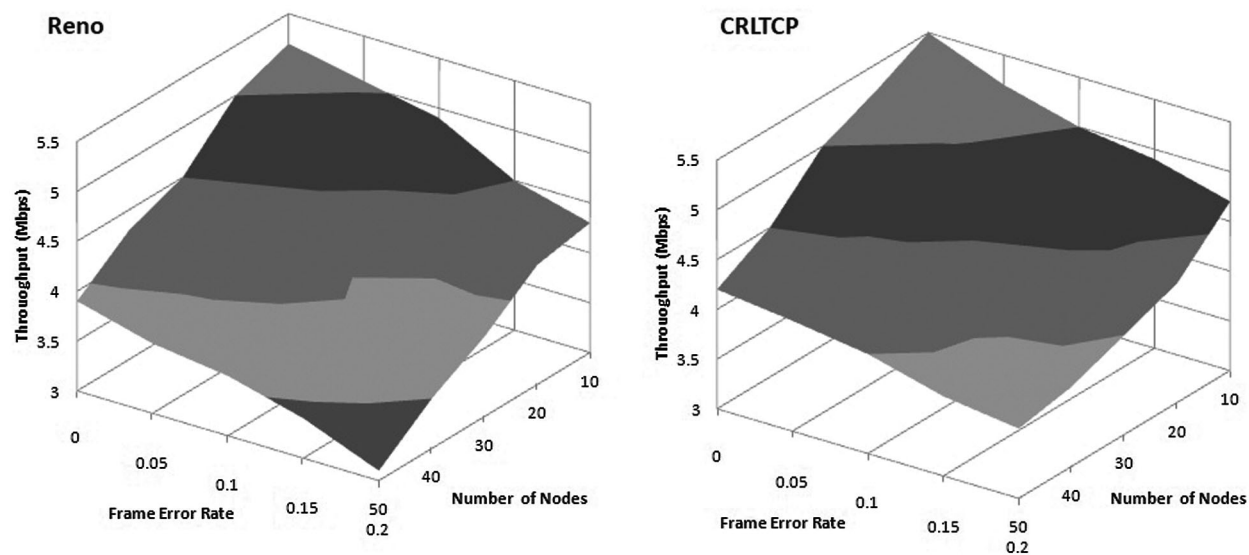


Figure 5: Fairness index

### 4.4 Throughput

Throughput is the average rate of successful data delivery over a communication channel [21]. Fig. 6 shows the throughput of TCP Reno and CRLTCP. It is observed that the throughput of CRLTCP increases when compared to the TCP Reno from 3.8 to 4.2 Mbps when varying number of nodes from 10 to 50. CRLTCP has 9% improvement over Reno since it is sensitive to collision wireless losses. But beyond 30 nodes, it starts declining. To estimate the throughput of CRLTCP in the presence of packet error, performance analysis is done by varying frame error rates (FER) ranging from 0.05 to 0.5. The throughput obtained ranges from 3.2 to 4.6 Mbps for CRLTCP whereas Reno is from 2.5 to 4.2 Mbps. CRLTCP provides better performance than Reno.

Representation of throughput in 3D view is done to visualize it in a better way. Throughput, Number of connections and frame error rate is taken in three different axis. In Reno, when the frame error rate and the number of connections increases, it is inferred that there is a downward sloping pattern from the Fig. 4. In CRLTCP, there is an elevation in the 3D plane and the above mentioned sloping trend is greatly reduced. The visualization of the graph is very clear which depicts the overall improvement. CRLTCP has 13% improvement over Reno because Frame Error Rate and collision losses are considered for differentiating wireless loss. It is also inferred that the usage of bandwidth is increased which is witnessed by the throughput improvement.



**Figure 6:** Throughput vs. frame error rate, number of nodes–Reno, CRLTCP

## 5 Conclusion

This work was developed for increasing the bandwidth usage by improving TCP performance in wireless network. The throughput is improved using CRLTCP by 13% when compared to Reno using cross layer approach between mac and transport layers. Snoop agent is used for discriminating the type of losses using the channel status information available in the mac layer. The CRLTCP algorithm discriminates the wireless losses as collision loss, frame loss and random loss by using received signal strength information. By rectifying the wireless losses due to contention and by communicating the random loss to make the congestion window insensitive helps in the improvement of the traditional TCP throughput through which bandwidth utilization is increased significantly in internet.

**Funding Statement:** The author received no specific funding for this study.

**Conflicts of Interest:** The author declares that they have no conflicts of interest to report regarding the present study.

## References

- [1] K. Pentikousis, “TCP in wired-cum-wireless environments,” *IEEE Communication Surveys and Tutorials*, vol. 3, no. 4, pp. 1–14, 2017.
- [2] A. Afanasyev, N. Tilley, R. Peter and L. Kleinrock, “Host-to-host congestion control for TCP,” *IEEE Communication Surveys and Tutorials*, vol. 12, no. 3, pp. 304–342, 2010.
- [3] S. Lohier, Y. G. Doudane and G. Pujolle, “Cross-layer loss differentiation algorithms to improve TCP performance in WLANs,” *Telecommunication Systems*, vol. 36, no. 1, pp. 61–72, 2007.
- [4] L. A. Grieco and S. Mascolo, “TCP performance evaluation and comparison of Westwood+, New Reno, and Vegas TCP congestion control,” *ACM SIGCOMM Computer Communication Review*, vol. 34, pp. 25–38, 2004.
- [5] C. Casetti, M. Gerla, S. Mascolo, M. Y. Sanadidi and R. Wang, “TCP westwood: End-to-end congestion control for wired/wireless networks,” *Wireless Networks*, vol. 8, no. 5, pp. 467–479, 2002.
- [6] A. Andread, S. Rizzuto and R. Zambon, “A Cross-layer jitter-based TCP for wireless networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 191, pp. 1–11, 2016.
- [7] C. Parsaand and J. J. Garcia-Luna-Aceves, “Improving TCP performance over wireless networks at the link layer,” *EURASIP Mobile Networks and Applications*, vol. 5, no. 1, pp. 57–71, 2004.



- [8] Q. H. Deng and A. N. Cai, "SVM-Based loss differentiation algorithm for wired-cum-wireless networks," *The Journal of China Universities of Posts and Telecommunications*, vol. 16, no. 4, pp. 104–111, 2009.
- [9] H. J. Lee, H. J. Byun and J. T. Lim, "TCP window control for variable bandwidth in wireless cellular networks," *IEEE Communication Letters*, vol. 11, no. 2, pp. 152–154, 2007.
- [10] Y. Rao, C. Deng, G. Zhao, Y. Qiao, L. Y. Fu *et al.*, "Self-adaptive implicit contention window adjustment mechanism for QoS optimization in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 109, pp. 36–52, 2018.
- [11] S. V. Jansi Rani, M. Suryakanth and S. Kaushik, "Cross layer based schemes for improving the performance of TCP in wireless networks," in *Int. Conf. on Computational Intelligence in Data Science IEEE*, Chennai, pp. 1–10, 2017.
- [12] B. J. Chang, Y. H. Liang and J. Y. Jin, "Adaptive cross layer based TCP congestion control for 4G wireless mobile cloud access," in *Int. Conf. on Consumer Electronics*, Taiwan, pp. 1–5, 2016.
- [13] B. J. Chang and S. P. Chen, "Cross-layer-based adaptive congestion and contention controls for accessing cloud services in 5G IEEE 802.11 family wireless networks," *Computer Communications*, vol. 8, pp. 33–45, 2017.
- [14] B. Nithya, R. G. Thivyavignesh and D. Bharat, "Receiver based contention management: A cross layer approach to enhance performance of wireless networks," *Journal of King Saud University*, vol. 32, no. 10, pp. 1–10, 2018.
- [15] A. Abouzeid, S. Royand and M. Azizoglu, "Stochastic modeling of TCP over lossy link," in *Proc. INFOCOM 2000*, Tel Aviv, Israel, pp. 1724–1733, 2000.
- [16] S. V. Jansi Rani, P. Narayanasamy and J. John Shiny, "A cross layer approach for improving TCP performance using channel access information," in *Int. Conf. on Communication and Signal Processing*, Melmaruvathur, India, pp. 587–591, 2013.
- [17] H. Zhai, Y. Kwon and Y. Fang, "Performance analysis of IEEE 802.11 MAC protocols in wireless LANs," *Wireless Communications and Mobile Computing*, vol. 4, pp. 917–931, 2004.
- [18] D. J. Deng, R. S. Chang, H. J. Lin and H. T. Chang, "A Cross-layer congestion and contention window control scheme for TCP performance improvement in wireless lans," *Telecommunication System*, vol. 42, no. 1, pp. 17–27, 2009.
- [19] T. Issariyakul and E. Hossain, *Introduction to network simulator 2 (NS2)*, pp. 1–18. Boston, MA: Springer, 2009.
- [20] H. J. Lee, H. J. Byun and J. T. Lim, "TCP window control for variable bandwidth in wireless cellular networks," *IEEE Communication Letters*, vol. 11, no. 2, pp. 152–154, 2007.
- [21] Y. T. Li, D. Leith and R. N. Shorten, "Experimental evaluation of TCP protocols for high-speed networks," *IEEE/ACM Transaction on Networks*, vol. 15, no. 5, pp. 1109–1122, 2007.