Journal of Computer Science 7 (12): 1798-1804, 2011
ISSN 1549-3636

Loss Discrimination Algorithm for Wired/Wireless Networks

Liw Jia Seng, Mohd Noor Derahman and Azizol Abdullah
Department of Communication Technology and Network,
Faculty of Computer Science and Information Technology,
University Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia

Abstract: Problem statement: Wireless technology has far growing with an increasing demand of the Wireless networking technologies recent years. The wireless access network and cellular networks are being used to support their need. It is usually connected to a wired backbone network. TCP is used to handle the congestion in wired network. However, it is not well suited for hybrid network consisting of wired and wireless networks. Packet loss occurs in wireless network mostly due to high bit error rate, varying in transmission length, or link failure. These scenarios are always misinterpreted by conventional TCP as a congestion loss. Hence, TCP sender performs congestion control which is unnecessary and leads to a poor network performance. **Approach:** A new algorithm proposed in this study is to differentiate the types of packet loss accurately. The algorithm will classify the packet loss and invoke proper correction mechanisms. **Results:** Our simulation results show the improvement of TCP performance as compared to the existing Selective-TCP and TCP NewReno. **Conclusion:** The proposed algorithm on classifying packet loss shows the improvement of TCP performance.

Keywords: Wireless loss, congestion loss, selective-tcp, available bandwidth, wireless access, bandwidth utilization, adaptive delayed, link layer, wireless transmission, connection starts, packet loss, congestion control

INTRODUCTION

Most of today's Internet traffic is carried out by Transmission Control Protocol (TCP), which accounts for 90% applications and 80% data. TCP was designed and tuned to perform very well in wired networks, where the key functionalities are to provide reliability connection, utilize the available bandwidth and avoid overloading the network (Leung and Li, 2006) .

Wireless networking technologies such as Bluetooth, 802.11 and 3G CDMA had been growing rapidly. This is consequences from the increasing demand for Internet-enabled wireless devices such as cellular phones and personal digital assistants. It utilizes the wireless access network to be connected to a wired backbone network. In this case, TCP is not well-suited for wired/wireless network to handle packet losses between the different communication medium. This is due to the nature of it packet loss. In the wired network, packet losses are due to the network congestion. Whereas in the wireless network, it can be from the handover operation, variable bandwidth as well as due to the dynamic network topology and host mobility (Leung and Li, 2006)-(Mondal and Luqman, 2007).

The link layer TCP congestion control algorithm by reducing the transmission rate is not suitable in wireless network. It results in low bandwidth utilization, unnecessary retransmission and, low goodput and throughput. Thus, we propose a loss discrimination algorithm as an improvement of (Biaz and Vaidya, 1999).

Related works: Proposals for link layer based protocols (Singh and Kankipati, 2004; Bhandarkar *et al.*, 2005) are effective in handling wireless losses but it introduces complexity in intermediate routers and base stations. The mechanisms use to improve TCP performance is either Automatic Repeat Request (ARQ) retransmissions (similar to the split-connection approach), or heavy Forward-Error Correction (FEC).

One of the proposals is associated with a sender site known as Inter-Layer Collaboration Protocol TCP (ILC-TCP) (Mondal and Luqman, 2007). ILC-TCP introduces a new link-layer parallel to the network protocol stack namely State Manager (SM). The SM

Corresponding Author: Azizol Abdullah, Department of Communication Technology and Network, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia

communicates with core layers and storing the link state in order to select the link, base station and the appropriate bandwidth. As a result, ILC-TCP is out perform over TCP in the event of long and frequent disconnection, as well as supporting the movement of mobile devices at certain considerable speed.

TCP Delayed Congestion Response (TCP-DCR) as proposed in (Bhandarkar *et al.,* 2005) improves congestion by identifying retransmission timeout or Duplicating Acknowledged (DUP ACKs) messages. The retransmission timer can also be modified. In the case of DUPACKs, the bounded delay period is set to one round trip time (RTT). By doing this, the sender may recover from link-layer loses; otherwise the retransmission algorithm is activated.

Adaptive Delayed Acknowledgement (TCP-ADA) presented by (Singh and Kankipati, 2004) is also a sender side solution for mobile ad-hoc networks. Instead of duplicating the acknowledgement messages, it uses Delayed Acknowledgement (DelAck) to block the acknowledgements for a specific time period. This is the modification from an actual proposal by (Altman and Jimenez, 2003) . It blocks the message until a specific number of data packets have been received. However, those proposals do not consider the state of the sender and receiver site simultaneously.

Moreover,(Mondal and Luqman, 2007) proposed a method to identify wireless losses and tackling it at the local link level. This is being done through MAC layer retransmissions. Even though the TCP performance is improved, the MAC retransmission strategy however consumes the wireless bandwidth. The same effect is experienced as long as it requires retransmission method in order to recover from congestion and wireless loss as in (Singh and Kankipati, 2004),(Bhandarkar *et al.,* 2005), (Altman Jimenez, 2003).

Meanwhile, in (Kliazovich *et al.,* 2007) proposes a cross layer approach (LLE-TCP) to improve a congestion control using TCP over large variety of wireless networks. This is achieved by avoiding TCP ACK packet transmission over the wireless channel. As a result, the saved time can be utilized by the nodes for data packet delivery. LLE-TCP enhances the protocol stacks of the wireless sender (or a base station) and the receiver with cross-layer ARQ agents which support ACK suppression. ARQ agent suppresses the outgoing ACKs at the receiver side and generates them locally at the sender or base station.

End-to-end TCP connection is first considered by (Kliazovich *et al.,* 2007) over wireless links and satellite links. Instead of setting congestion window size and slow start threshold based on packet drop information as in conventional TCP, TCP Westwood (TCPW) estimates available bandwidth from the TCP sender and sets congestion window size and slow start threshold accordingly. It focuses on the sender-side congestion modification. Thus does not really sensitive to random errors. It also require the sender to sense the channel state prior to transmit. In (Casetti *et al.,* 2002) proposed another end-to-end scheme to infer packet losses using variable threshold associated with Round Trip Time (RTT). The RTT is used to classify the loss is due to either wireless loss or congestion loss. However, this is considered inappropriate as the channel capacity in wired network in this study is very limited.

(Lim and Jang, 2008) proposed Indirect-TCP (I-TCP) that splits the connection on the wireless edge to protect TCP session from wireless media inconsistencies and losses. This results in two different flow and congestion controls in wired and wireless sections separately. It may result in serious inequalities on the two sides. In such attempt, I-TCP eventually violates end-to-end semantics of the conventional TCP. Meanwhile, M-TCP presented by is similar to I-TCP and splits the connection at super host. It differs only in a way, that super host does not generate Acknowledgment (ACK) of last received segment until it receives the ACK from mobile node. In this way it also breaks end-to-end semantics of the TCP and disparities of two sides flow and congestion control mechanism still exist. Another problem that may arise is inaccurate source decision. This will happen when the acknowledgment of last byte is not received. Thus, the sender resends all data which is already received.

This study proposed an end-to-end solution based on a new packet loss differential algorithm to improve TCP performance over hybrid wired/wireless networks.

MATERIALS AND METHODS

Architecture and protocol: We assume that the wireless link is the only bottleneck for the connection. Hence, packets from wired networks will queue in the base station before forwarding to the wireless network. Figure 1 shows the typical scenario for the wired/wireless network.

We propose an end-to-end solution to improve TCP performance over wired/wireless networks. This is based on detecting the type of losses at the TCP receiver. The implementation is an extension to TCP NewReno.

In our case, if an out-of-sequence packet is received, loss differential mechanism will categorize the cause of losses is either due to congestion loss or wireless error. We introduce a new loss detection algorithm which is a modification of the existing loss

detection technique based on packet inter-arrival times at the receiver. When the cause of packet loss is determined, two corrective measures are to be taken to improve TCP performance. Fig. 2 illustrates the Selective-TCP framework of the propose approach.

In the case of wireless transmission losses, receiver sends SNACK instead of duplicate acknowledgements. TCP NewReno's congestion control mechanism is not invoked. As a result, the slow start threshold and congestion window size are not reset unnecessarily, resulting in better bandwidth utilization (Paul and Trajkovic 2006).

In the scenario of congestion loss, the available bandwidth receiver information is sent to the sender for estimation. Thus the sender then amends their congestion window size accordingly. This will help TCP's IAD scheme to set the appropriate sender's congestion window size.

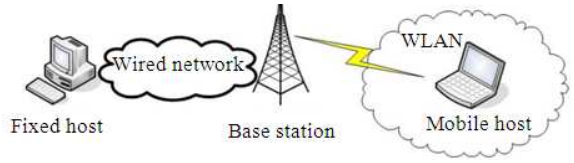

Fig. 1: Network scenario

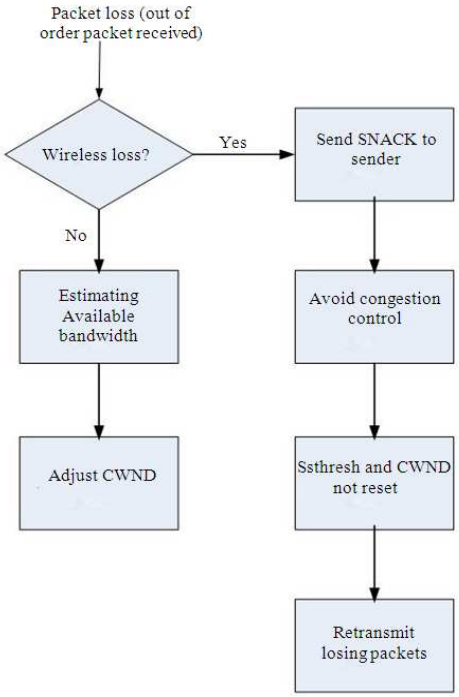

Fig. 2: TCP framework

Proposed schema: In this section, the consideration is given on the mechanism used to differentiate the wireless loss. We then detailed out the algorithm based on inter-arrival time and IAD. Finally we consider the approach to handle the congestion loss and wireless loss.

Wireless loss differentiation mechanism: We assume that the wireless links are the sources for bottlenecks. The sender performs bulk data transfer (assume all packets are same size) and the routers or base station at the boundary of wired and wireless networks buffer more than one packets that are destined to pass through the next wireless link. Hence, packets accumulate at the base station and most packets will be sent back-to-back over the wireless link. Based on the above observation, a boundary host on a wireless link can differentiate between wireless transmission losses and congestion losses based on the inter-arrival packet time.

The inter-arrival time of the packets in the next host at the boundary between the next wireless network and the next wired network is approximately equal to the time T required to transmit one packet on the wireless link. However, if the loss happens due to wireless error, the inter-arrival time between two consecutive packets will be greater than T, since at least T is elapsed in trying to send each of the lost packets. Based on above observations, a simple heuristic has been developed by (Biaz and Vaidya, 1999).

A received frame is called out-of-order if its sequence number is not equal to one plus the sequence number of the frame received immediately before it. Else the frame is called in-sequence. Let Tmin be the minimum inter-arrival time observed so far by the receiver during the connection. If there is no any packet loss or the packets are in sequence, inter-arrival time between two consecutive packets is Tmin as shown in Fig. 3.

If a packet is lost in the wired link due to the congestion, the packet inter-arrival gap is still Tmin because the packets queue at base station before being transmitted on the wireless link. Figure 4 illustrates inter-arrival time between two consecutive packets when packet loss due to congestion.

However, if a packet is lost in the wireless link, the inter-arrival gap at receiver is ~2 Tmin because the lost packet has travelled on the wireless link for some time before being lost. Figure 5 shows the inter-arrival time between two consecutive packets when a packet loss in wireless link.

However, wireless loss differentiation with Tmin is not appropriate as inter-arrival time between

packets would depend on the current network load Fig. 6. Tmin will be far deviate from the current inter-arrival time after a long period of connection. Thus, we propose a new algorithm to differentiate packets loss types using a standard deviation for the packet inter-arrival time. Figure 7 shows the modification algorithm called IAD scheme.

Congestion loss handling procedure: The Selective-TCP increases a counter congestion_count when experienced with congestion loss. The congestion control is initiated by the sender to manipulate the congestion window size. The threshold value for congestion count, k is experimentally chosen to be equal to 10.

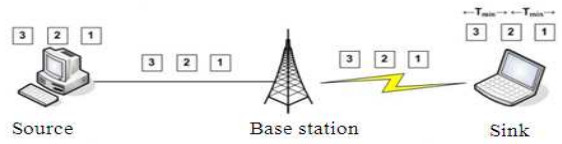

Fig. 3: Inter-arrival time between two consecutive packets

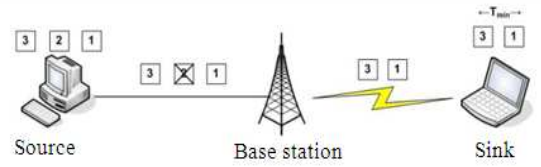

Fig. 4: Inter arrival between two consecutive packets during congestion

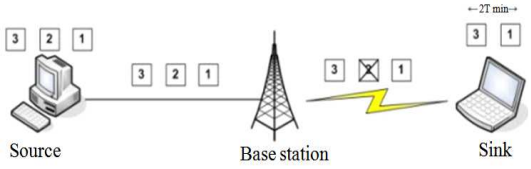

Fig. 5: Inter arrival between two consecutive packets during packet loss in wireless link

N = Number of packet loss between two packet arrival
If packet loss = = TRUE then
If [(n+1)T min<= packet inter-arrival time <(n + 2)T min] then
Wireless loss
Else
Congestion loss
End if
End if

Fig. 6: Wireless loss differentiation over inter arrival time

Tia = current packet inter-arrival time
μia = mean packet inter-arrival time
δia = standard deviation inter-arrival time
Between two packet arrival

If packe tloss = TRUE then
If Tia - > μia >δia then
Wireless loss
Else
Congestion loss
End if
End

Fig.7: Proposed IAD Scheme to differentiate packet loss either due to wireless loss or congestion loss.

This value is critical in deciding when the measured bandwidth is sent to the sender and when congestion window size at the sender is being set. If the congestion window size is set before TCP sends 3 duplicate acknowledgements and reduces congestion window size, it will not be helpful in terms of good put/throughput performance. On the other hand, setting the congestion window size long after TCP IAD algorithm has reduced it. However, it will not be helpful as well. Experiments show any lower/higher value than 10 deteriorates network performance, so we use k = 10 throughout the simulations.

Wireless loss handling procedure: In the case of wireless loss, the receiver sends ACK with SNACK option to the sender. As a consequence, the TCP sender retransmits the missing packets indicated by SNACK. Acknowledgments with SNACK options are sent after a certain delay (snack delay). As a result the chance of unnecessarily retransmitting a delayed or mis-ordered segment is limited. Since the SNACK option triggers a retransmission, there is no reliance on the Fast Retransmit algorithm to detect the loss. This independence from the Fast Retransmit algorithm is important because duplicate ACKs may never be received when operating over a highly loss link.

When a SNACK is received, the TCP sender aggressively retransmits the packet(s) indicated as lost packet(s), without waiting for retransmission timeout to occur. Hence, congestion control mechanism and unnecessary retransmissions are avoided, leading to higher bandwidth utilization.

Congestion info stores the bandwidth measured at the receiver when a packet loss is detected. If the congestion-info field in the TCP header has a non-zero

value, the sender sets its congestion window size equal to congestion info *base_rtt, where base_rtt is the initial round trip time. This prevents the TCP IAD algorithm from setting the congestion window size to be unnecessarily small. Congestion info is multiplied by base_rtt to increase the congestion window size.

Simulation scenario: We compare performance of IAD scheme with Selective-TCP and TCP NewReno in a congested link. In those cases, the 5% burst error in wireless link has been introduced. We also study the goodput performance of IAD Scheme with no wireless error, 1% random error (random statistical error) and 5% burst error (continuous lacking of data) (Paul and Trajkovic, 2006).

There is only CBR/UDP traffic for the first 100 sec of simulation time. After 100 sec, TCP connection starts and exists along with UDP connection. All connections end after 300 sec and 600 sec of simulation time (Paul and Trajkovic 2006). The performance measures we consider are goodput, slow start threshold and size of congestion window. Good put is defined as the number of bits received by the destination host, less the duplicates. We use the maximum sequence number of packets reached at the destination to represent good put in the simulation results.

RESULT AND DISCUSSION

Performance comparison for 300 sec simulation time: Firstly, we evaluate IAD scheme with Selective-TCP and TCP NewReno in a congested link. For the first 100 sec of simulation time, there is only CBR/UDP traffic. After 100 sec, TCP connection starts and exists along with UDP connection. All connections end after 300 sec of simulation time.

Good put comparison of the IAD Scheme, Selective-TCP and TCP NewReno is shown in Fig. 7.

IAD scheme performs better than selective-TCP and TCP NewReno after 300 sec simulation time for goodput comparison Fig. 8. Furthermore, we compare the TCP congestion widow size for IAD scheme, Selective-TCP and TCP NewReno as shown in Fig. 9.

Size of congestion window for IAD scheme is similar with Selective-TCP. Both IAD Scheme and Selective-TCP shows larger congestion window size than TCP NewReno, indicating better utilization of available bandwidth.

The slow start threshold comparison for IAD scheme, Selective-TCP and TCP NewReno is show in Fig. 10.

IAD scheme and Selective-TCP senders maintain a constant value of slow start threshold over a longer period compare to TCP-NewReno. The initial value for slow start threshold is equal to 20 sec.

Performance comparison for different error types: A comparison of good put performance of IAD scheme, without wireless error, with 1% random error () and with 5% burst error. These are associated with random statistical error and continuous lacking of data respectively. The results can be seen in Fig. 11. It portrays the performance of IAD scheme and TCP NewReno should be exactly same. This scenario is also applied to the Selective-TCP as long as there is no wireless error.

However, it experiences the difference of maximum 5% as shown in Fig. 12. The detection accuracy of the mechanism is set to only ~95% is the reason why the difference occurred in IAD scheme. Thus the losses are misinterpreted as either congestion loss or wireless losses.

We evaluate IAD scheme with Selective-TCP and TCP NewReno in a congested link over a longer duration. We first simulate the CBR/UDP traffic prior to 100 sec. After that TCP connection starts and coexist with UDP connection. All connections are terminated after 600 sec.

The performance comparison of the IAD Scheme, Selective-TCP and TCP NewReno is shown in Fig. 13. IAD Scheme out performs over the selective-TCP and TCP NewReno. This also shows that IAD scheme can be used in a longer period of connection. This is due to the correct packet loss classification by IAD Scheme. Thus allowing a proper wireless handling procedure to be undertaken such as described in Section Wireless Loss Handling Procedure.

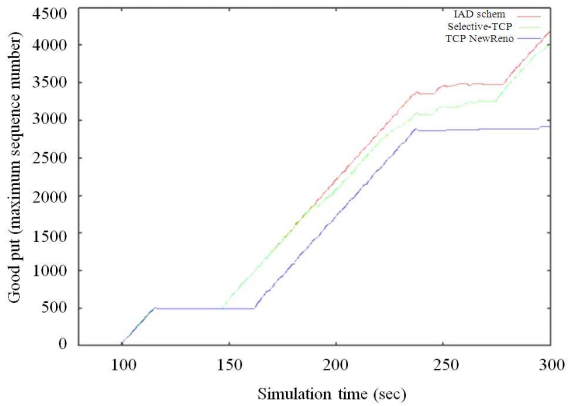

Fig. 8: Goo dput for IAD, Selective-TCP and TCP NewReno over 300 sec

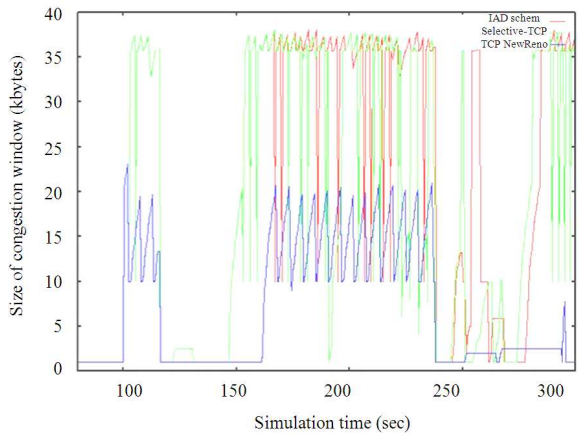

Fig. 9: Size of congestion window for for IAD Scheme, Selective-TCP and TCP NewReno over 300 sec

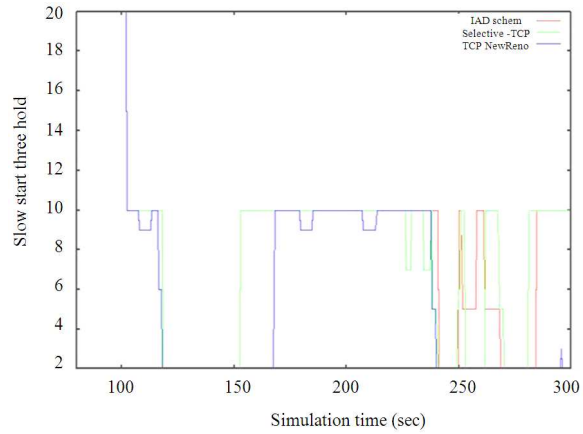

Fig. 101: Slow start threshold for IAD Scheme, Selective-TCP and TCP NEwReno over 300 sec

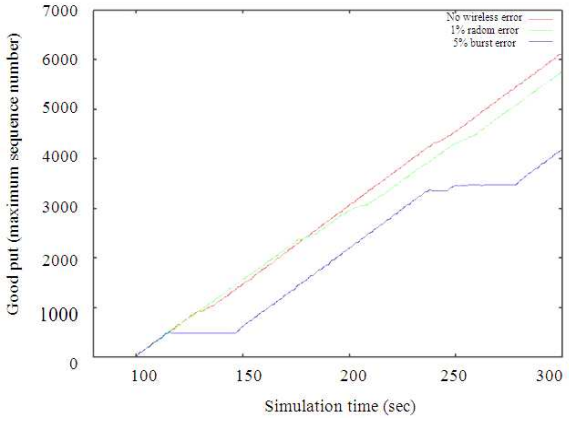

Fig.11: Good put over IAD Scheme without wireless error

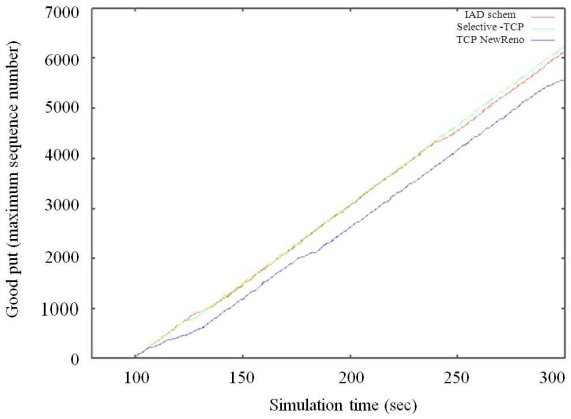

Fig. 122: IAD Scheme with Selective_TCP and TCP-NewReno

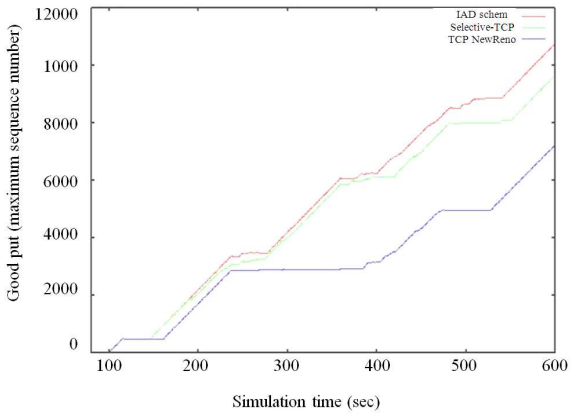

Fig. 13: Good put comparison for IAD scheme, Selective-TCP and TCP-NewReno over 600 sec

CONCLUSION

TCP was designed to provide reliable connection, utilize the available bandwidth and avoid overloading the network. However, TCP is not well-suited for hybrid networks consisting of wired and wireless networks due to the high BER and limited bandwidth of the wireless link. When a packet loss in wireless network, which are quite frequent and often misinterpreted by the TCP sender as loss due to congestion. Hence, TCP sender performs TCP congestion control, which is unnecessary and leads to poor performance of TCP. An end-to-end approach based on the new algorithm for identifying cases of wireless loss and congestion loss was proposed in this project to improve TCP performance over hybrid wired-wireless network. In case of wireless loss, the

receiver sends SNACK to the sender to prevent congestion control. If congestion loss detected, receiver informs the sender of the measured bandwidth at receiver. The sender then sets the congestion window size accordingly. The simulation results show that IAD can be utilized to improve in bandwidth utilization and increase good put as compared to Selective-TCP and TCP NewReno.

REFERENCES

Leung, K.C. and V.O.K. Li, 2006. Transmission control protocol (TCP) in wireless networks: issues, approaches and challenges. Commun. Surveys Tutorials, IEEE, 8: 64-79. DOI: 10.1109/COMST.2006.283822

Singh, A.K. and K. Kankipati, 2004. TCP-ADA: TCP with adaptive delayed acknowledgement for mobile ad hoc networks. Proceedings of the Wireless Communication Networking Conference, Mar. 21-25, IEEE Xplore Press, India, pp: 1685-1690. DOI: 10.1109/WCNC.2004.1311806

Mondal, S.A. and F.B. Luqman, 2007. Improving TCP performance over wired-wireless networks. Comput. Networks, 51: 3799-3811. DOI: 10.1016/j.comnet.2007.04.003

Biaz, S. and N.H. Vaidya, 1999. Discriminating congestion losses from wireless losses using inter-arrival times at the receiver. Proceedings of the Symposium on Application-Specific Systems and Software Engineering and Technology, Mar. 24-27, IEEE Xplore Press, Richardson, TX , USA., pp: 10-17. DOI: 10.1109/ASSET.1999.756746

Bhandarkar, S., N.E. Sadry, A.L.N. Reddy and N.H. Vaidya, 2005. TCP-DCR: A novel protocol for tolerating wireless channel errors. IEEE Trans. Mobile Comput., 4: 517-529. DOI: 10.1109/TMC.2005.72

Altman, E. and T. Jimenez, 2003. Novel delayed ACK techniques for improving TCP performance in multihop wireless networks. Lect. Notes Comput. Sci., 2775: 237-250.

Kliazovich, D., F. Granelli and M. Gerla, 2007. Performance improvement in wireless networks using cross-layer ARQ. Comput. Network, 51: 4396-4411. DOI: 10.1016/j.comnet.2007.06.002

Casetti, C., M. Gerla, S. Mascolo, M.Y. Sanadidi and R. Wang, 2002. TCP westwood: end-to-end congestion control for wired/wireless networks. Wireless Network, 8: 467-479. DOI: 10.1023/A:1016590112381

Lim, C.H. and J.W. Jang, 2008. Robust end-to-end loss differentiation scheme for transport control protocol over wired/wireless networks. Commun., IET, 2: 284-291. DOI: 10.1049/iet-com:20060172.

Paul, R. and L. Trajkovic, 2006. Selective-TCP for wired/wireless networks. Simulat. Series, 38: 339-346.