

## Increasing transmission control protocol speed by reducing the acknowledgement collision probability

Suherman<sup>1</sup>, Ali Hanafiah Rambe<sup>2</sup>, Norshakila Haris<sup>3</sup>, Anhar<sup>4</sup>

<sup>1,2</sup>Electrical Eng. Department, Universitas Sumatera Utara, Kampus USU, Medan, Indonesia

<sup>3</sup>Marine Electrical and Electronics Tech., Universiti Kuala Lumpur, Malaysia

<sup>4</sup>Electrical Eng. Department, Universitas Riau, Pekanbaru, Indonesia

---

### Article Info

#### Article history:

Received Apr 15, 2020

Revised Nov 10, 2020

Accepted Dec 5, 2020

---

#### Keywords:

802.11, 802.16

Collision probability

TCP/IP

TCP-ACK

---

### ABSTRACT

Transmission control protocol (TCP) employs acknowledgement (TCP-ACK) for every transmitted packets to ensure reliable transmission. As a result, it sends the next window packets after receiving the TCP-ACK packet of previous window. This means that the earlier the TCP-ACK packet arrives, the faster the TCP next window transmission and the better the TCP performances. To do so, there should be a special treatment to the transmitted TCP-ACK to fasten next window transmission. This paper proposes a collision probability reduction for the transmitted TCP-ACK packets so that the overall TCP delay reduces. Collision probability reduction can be implemented in many ways. Initially, mathematical analysis is provided to prove that method can work as expected. The mathematic analysis shows that when TCP-ACK collision probability is reduced, the overall TCP delay is also reduced. The proposed method is then implemented in 802.11, 802.16 and complex networks. The NS-2 simulations evaluations for the aforementioned networks and the proposed method proved that collision reductions on TCP-ACK exert average TCP delay reductions about 11.86%, 28.04% and 9.46% subsequently. The proposed method is also applicable for other TCP types.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



---

### Corresponding Author:

Suherman

Electrical Eng. Department

Universitas Sumatera Utara

Kampus USU, Medan-Indonesia

Email: suherman@usu.ac.id

---

## 1. INTRODUCTION

Transmission control protocol, shortened as Transmission control protocol (TCP), provides data transmissions reliability by providing acknowledgement and retransmission services. TCP makes sure that each packet is received successfully and sequentially [1]. TCP drawback is that the acknowledgement and retransmission services inject additional delay, which is often annoying for real-time applications. Some low bit rate applications, such as youtube employ TCP [2], high bit rate application compensates delay by introducing application buffering; however, buffering may be unacceptable for some users [3]. Other transport protocol is user datagram protocol (UDP), which minimizes service delay by limiting the headers and omitting connection-oriented services [4]. Some applications such as real-time high definition video uses UDP for data transfer and employs TCP for control commands.

There have been many proposals to enhance TCP and UDP performances. Some works embedded TCP properties into UDP [5-7] or employed a negative acknowledgement [8]. TCP enhancement are dominated by flow control techniques that concerned with window management. The flow control and

window adjustments produced various TCP variants such as TCP Vegas, TCP Reno and others [9-11]. TCP and cross-layer techniques have also been developed to obtain lower layer supports [12-14]. Most existing works on TCP dealt with how to manage TCP window [9-11] and obtain lower layer support [12-14] to avoid congestion. This paper aims at reducing TCP acknowledgement (TCP-ACK) delay, without changing TCP itself. One of the closest related research by Yoshiaki Ohta [15] showed that asymmetry bandwidth in LTE may cause congestion and resource shortage on uplink for transporting TCP-ACK. Ohta confirmed that a proper scheduling on transmission may improve the performance. Since TCP sends the next window after receiving previous window acknowledgement, the faster TCP-ACK reception leads to faster TCP transmission. This paper focuses on how to speed up TCP transmission by reducing TCP-ACK collision probability. Mathematical analysis is presented in next section and TCP-ACK collision probability reduction implementations in 802.11 network; 802.16 network; and in larger networks are presented.

## 2. THE PROPOSED METHOD

Since contention request is employed in both 802.11 and 802.16 network, the analysis on the effectiveness of TCP-ACK collision probability reduction is carried out on the contending networks. In such network, a mobile node should sense the network availability by sensing carrier frequency. When the carrier signal is not detected, it means the channel is idle. However, if the carrier signal is detected, channel is busy and the node should defer its transmission. It then picks a random back-off time to wait [16]. Random access analysis provided by [17, 18], show that the collision probability of one transmission of  $n$  nodes is expressed by the (1), where  $\tau$  is the attempt probability.

$$p = 1 - (1 - \tau)^{n-1} \quad (1)$$

Since back-off influences the collision probability, the proposed model that included back-off duration [19] is employed in this paper. The model approximates back-off duration as the function of back-off multiplier,  $\lambda$ , and initial contention window,  $W$ . The  $i$ th back-off duration  $U(i)$  is given by the back-off uniform distribution of  $m$  attempts out of  $K$  maximum attempts (2). The average back-off duration is given by (3).

$$U^{(i)} = \begin{cases} (0, \lambda^i W - 1) & \text{for } i = 0 \dots, m - 1 \\ (0, \lambda^m W - 1) & \text{for } i = m \dots, K - 1 \end{cases} \quad (2)$$

$$E[U^{(i)}] = \begin{cases} (\lambda^i W - 1)/2 & \text{for } i = 0 \dots, m - 1 \\ (\lambda^m W - 1)/2 & \text{for } i = m \dots, K - 1 \end{cases} \quad (3)$$

$$\tau^{-1} = \sum_0^K \pi_i E[U^{(i)}] \quad (4)$$

The reciprocal attempt probability is given by (4), where  $\pi_i$  is the relative frequency of the node entering back off period. When all nodes are treated equally, the collision probability of each node remains similar. In order to reduce collision probability on the TCP-ACK packet, MAC layer should prioritize TCP-ACK frames by reducing the inter-frame time (tif) by factor of  $\alpha < 1$ , the node sending TCP-ACK frames will have a shorter waiting time. Suppose there are  $x$  nodes with TCP connections in the network, since number of TCP data and TCP-ACK are similar, there will be  $x/2$  nodes compete after  $\alpha \cdot \text{tif}$ . The reciprocal attempt probability is given by (5). The collision probability is then computed based on (6).

$$\tau^{-1} = \sum_0^{x/2} \pi_i E[U^{(i)}] \quad (5)$$

$$p = 1 - \left( 1 - 1 / \sum_1^{x/2} \pi_i E[U^{(i)}] \right)^{x/2} \quad (6)$$

Other nodes with non-TCP-ACK payload will compete at least to  $n-x/2$  nodes. Despite  $x/2$  nodes decrement in collision probability equation, there will be an increase on the frequency of entering back-off period  $\pi_i$  as a result of prioritization of TCP-ACK frames. The other nodes have collision probability as given by (7), where  $\pi_r$  is additional back-off frequency.

$$p = 1 - \left( 1 - \left( \frac{1}{\sum_1^{n-x/2} (\pi_t + (\pi_r) E[U^{(l)}])} \right) \right)^{n-x/2} \tag{7}$$

By using the parameters plotted in Table 1 [19] for total of n saturated stations and 5 TCP connections in an ideal channel condition and by ignoring the propagation delay, the approximated collision probability of the contending nodes and the proposed TCP-ACK collision reduction node are shown in Figure 1. It is shown that the collision probability of nodes sending TCP-ACK packet is smaller than the baseline nodes.

Since TCP-ACK collision probability is reduced, the TCP-ACK frames are prioritized than the other packets, the overall round trip time of the TCP flow within the evaluated network is expected to be shortened. Sakurai *et al.* [19] approximated the average access delay (T) by a unified analytical model to include delays of data (tdata), inter-frame (tif : tsifs and tdifs) and TCP-ACK delays (tack) as the access delay:

$$C=T^*=C^* = t_{data} + t_{sifs} + t_{ack} + t_{difs} \tag{8}$$

The waiting time reduction with factor of  $\alpha$  for TCP-ACK frames minimizes the number of possible collided nodes from n to x, influences tsifs and tdifs, reduces tack and finally reduces the average duration of the TCP transmission by (9).

$$E[D] = \frac{(t_{slot} + p \cdot E[C^*])}{\pi \cdot (1-p)} + \frac{(n-1) \cdot (E[T^*] - E[C^*])}{1-\pi} + \frac{p \cdot E[C]}{1-p} + E[T] \tag{9}$$

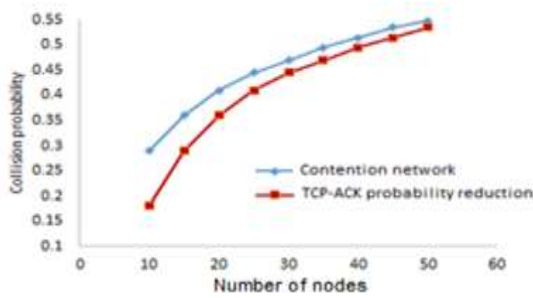


Figure 1. Callision probability vs number of nodes

Table 1. Contention parameters [19]

Parameters	Value
Data bit rate ( $r_{data}$ )	11 Mbps
Control bit rate ( $r_{cont}$ )	1 Mbps
PHY header ( $t_{PHY}$ )	192 us
MAC Header ( $l_{MAC}$ )	224 bits
TCP/IP Header ( $l_{TCP/IP}$ )	320 bits
TCP-ACK Header ( $l_{ACK}$ )	112 bits
Slot time ( $t_{slot}$ )	20 us
Inter-frame time ( $t_{if}$ )	50 us
Min CW ( $W$ )	32
Doubling limit ( $M$ )	5
Retry limit ( $K$ )	7
TCP-ACK inter-frame reduction ( $\alpha$ )	0.75

### 3. THE EVALUATION METHOD AND RESULTS

#### 3.1. TCP-ACK collision reduction in the 802.11 network

Since the 802.11 network employs contention schema through the carrier sense multiple access/collision avoidance or CSMA/CA, collision reduction for TCP-ACK is simply applied by reducing the distributed inter-frame space (DIFS) time of nodes that transmitting frames containing the TCP-ACK packet. DIFS reduction is set in factor of  $\alpha < 1$ . NS-2 simulations consisted of a single 802.11 access point serving some nodes were conducted to evaluate the proposed method. Instead of using the standard traffic pattern such as constant or poison distributed traffics, simulations evaluated a real video streams generated by the Evalvid framework as mentioned in [20], so that the proposed method is accessed for a real application.

Existing solutions that applied cross-layer schemas on TCP over 802.11 network such as TCP-ELN that applied explicit loss notification (ELN) from lower layer [21] or TCP-CC that integrated the contention ratio values and channel utilization into TCP cwnd value [22] can also make use the proposed method. As contention reduction is applied in medium access layer for speeding TCP-ACK transmission, existing TCP enhancement can also take benefits from the proposed method. In addition, this paper evaluated the existing TCP cross-layer technique and TCP-ELN [21] performance when combined with the proposed method.

The number of mobile nodes was set from 2 nodes to 20 nodes including one access point as the receiver. Traffics were sent two ways: upward and downward using TCP connections. MPEG4 code with the packet sequence of IPP and group of pictures (GOPs) of 30 and bit rate of 561 kbps were employed. Defaults DIFS was set 20  $\mu$ s and the TCP-ACK DIFS was 15  $\mu$ s. Figure 2 shows the simulation results. Since NS-2 is a discrete event simulator, simulation results are quite rough. However, the pattern shows that by reducing the TCP-ACK collision probability, the overall TCP delay decreases. The proposed method with 15  $\mu$ s DIFS for acknowledgement frame shortens average TCP delay by 11.86% as shown in Figure 2(a). TCP jitter on

the other hand, is not distinguishable as shown in Figure 2(b). Since jitter depends on delay variations, the value is not influenced by the proposed technique. It can be explained that number TCP packets in the same window may vary depending upon the window size and TCP type. In average, jitter value is in a good range.

The collision probability reduction of TCP-ACK implies TCP transmission assurance. As a result, packet loss rate is suppressed. Packet loss decreases for the number of users 1 to 14 as shown in Figure 2(c). However, losses increase for more nodes in the network. It is caused by the total rate of transmitted video which is higher than 802.11 capacities. This also means that DIFS reduction is not effective when the number of TCP-ACK frame is high. However, the average packet loss reduces up to 5%. When DIFS value for the TCP-ACK is adjusted from 10 $\mu$ s to 15  $\mu$ s as shown in Figure 2(d), delay characteristics for the proposed method is consistently lower than the baseline. This means that TCP-ACK collision probability reduction in 802.11 successfully increases the TCP speed. Since the proposed method is applied in MAC layer and is independent of TCP protocol, the proposed method can be combined with existing TCP enhancement such as TCP-ELN [21]. The combination of TCP-ELN and the proposed method experienced 7.61% delay decrement as shown in Figure 2(e) and 4.91% jitter decrement as shown in Figure 2(f).

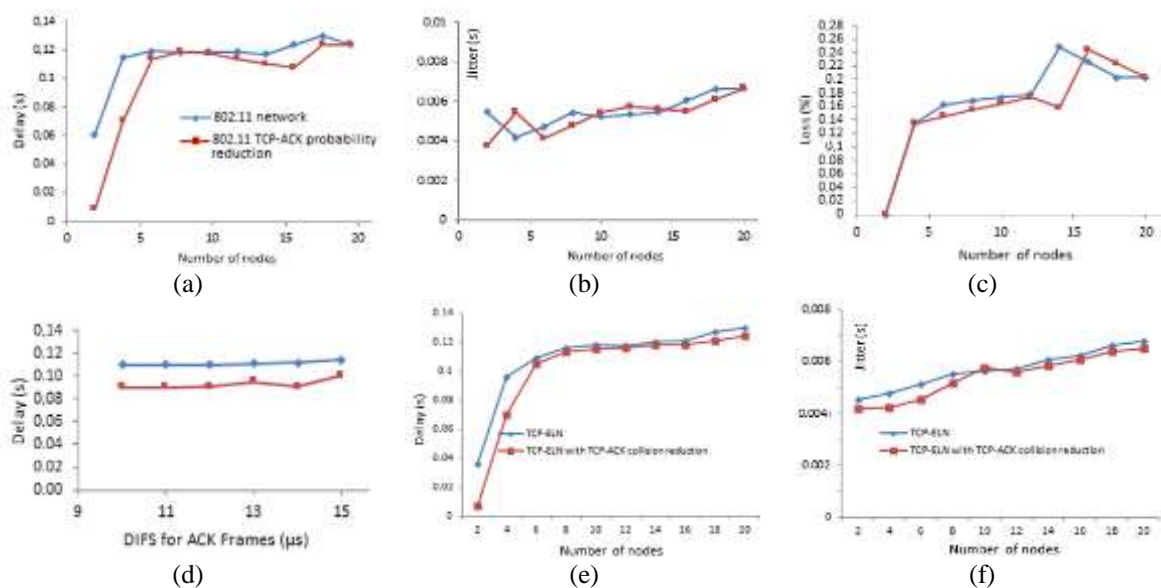


Figure 2. The proposed method impact to TCP and TCP-ELC performances in the 802.11 network; (a) TPC delay, (b) TPC jitter, (c) TPC loss, (d) DIFS variation, (e) TPC-ELN delay, and (f) TPC-ELN jitter

### 3.2. TCP-ACK collision reduction in the 802.16 network

The 802.16 network or widely known as worldwide interoperability microwave access (WiMAX) is a broadband radio access technology that employs quality of services (QoS) that means its base station (BS) is able to differentiate services for its customers or subscriber stations (SS). TCP-ACK collision reduction can be achieved by applying different QoS for TCP-ACK packets and TCP-DATA Packets. However, the scheme may be costly as QoS is generally implemented per SS or per connections (one SS may have more than one connection). However, there are some mechanisms employed in each connection that can be used to give TCP-ACK collision reduction; namely: bandwidth request mechanism and transmission scheduling [20].

Since 802.16 also employs contention schema quit similar to CSMA to request bandwidth, the same mathematical analysis is coherent to WiMAX. In order to evaluate TCP-ACK collision probability reduction in WiMAX network, downlink TCP traffic flow is considered as in Figure 3(a). Base station receives TCP data from server and forwards it to SS via the 802.16 link. Figure 3(b) shows the strategy of collision reduction implementation that surpasses bandwidth request mechanism. The implementation method modifies transport-mac cross-layer scheme proposed in [12] which employed negative acknowledgement. This paper uses TCP-ACK, a positive acknowledgement. The architecture is based on NIST WiMAX module [23]. The incoming packet is denoted by flag to determine either TCP-DATA contained or not. If TCP-DATA is detected, frame re-assembler within WiMAX MAC layer notifies BytesACK in WiMAX scheduler to allocate bandwidth slot to the intended SS for the upcoming TCP-ACK. By performing this method, SS is allocated bandwidth for sending TCP-ACK before SS makes bandwidth request. This in turn

reduces collision probability of bandwidth request for TCP-ACK, which eventually fastens the next TCP window transmission.

In order to apply and observe the proposed method in WiMAX environment, NS-2 simulations were performed by using NIST WiMAX module [23]. WiMAX transmit and receive power thresholds were adjusted to cover 1000m radius, 64 quadrature amplitude modulation and the two-ray ground model were used, downlink (DL) to uplink (UL) ratio was 0.3. The simulated network had four SSs within one base station. Traffic sources were obtained from akiyo\_cif video to produce simulated traces, taken from [24]. The traffic generation, reconstruction and evaluation followed the Evalvid video evaluation framework [25, 26]. Round robin and contention were used for medium access schedule and bandwidth request [27].

Figures 3(c) and 3(d) shows the influence of the designed method to TCP and TCP fast retransmission over the 802.16 network. The proposed method is consistently reducing TCP delay for all I-frame rate values as shown in Figure 3(c). This significant reduction is caused by bandwidth allocation for TCP-ACK is performed prior bandwidth request by SS. In average, the proposed method reduced delay about 28.04%. Meanwhile, jitter and packet loss differences are not significant (not shown). Further assessment on TCP modification, Fast Retransmission TCP [28] demonstrates that the designed method is capable in reducing number of duplicate ACK packets that eventually reduces packet delay. Reduction achieves 16.05% in average as shown in Figure 3(d).

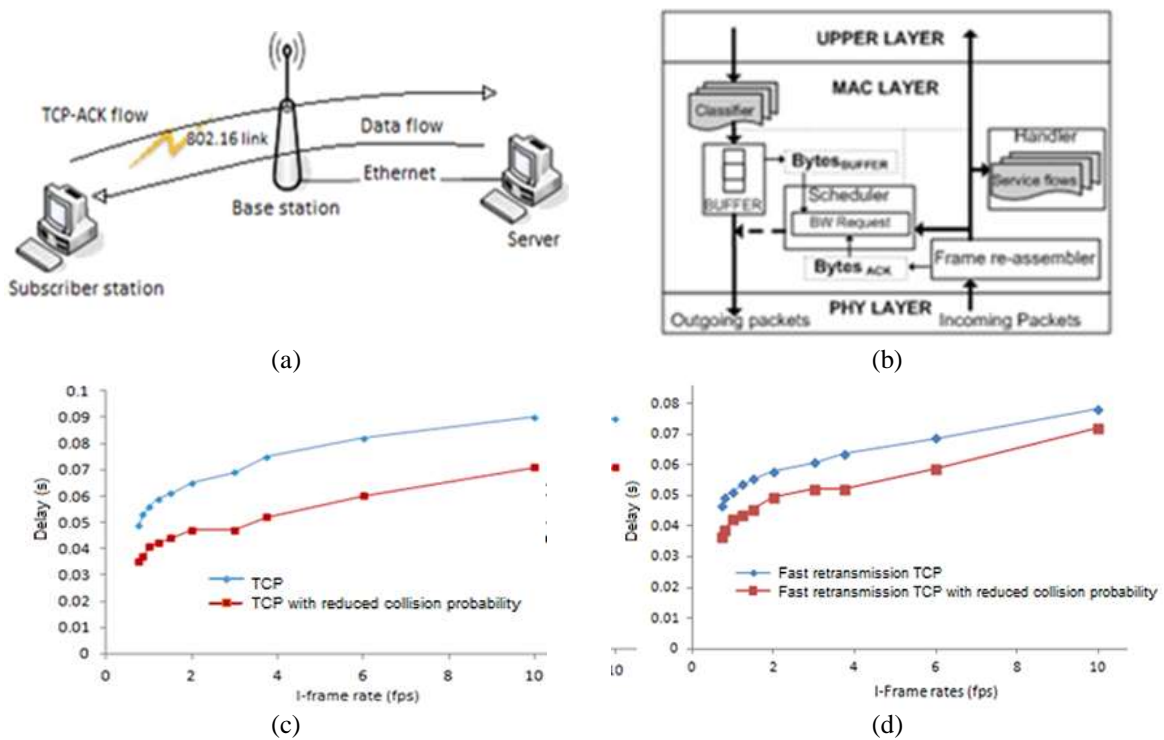


Figure 3. Delay performance for various DIFSS; (a) TCP traffic flow, (b) TP\_ACK Collision reduction in WiMAX, (c) TCP delay, and (d) TCP fast retransmission delay

### 3.3. TCP-ACK collision reduction in the complex network

Previous TCP-ACK collision reductions on 802.11 and 802.16 are performed in a single link. Complex dynamic network always involves multiple links that keep changing overtime. Implementation based on link by link may not be efficient. In order to effectively implementing TCP-ACK reduction, routing scheme may be preferred. Figure 4(a) shows how routing schema can do TCP-ACK collision reduction. TCP connection from PC A to PC B sends data from the complex networks. Initially, both TCP-DATA and TCP-ACK are sent through the same network. Figure 4(b) shows how to implement TCP-ACK collision reduction. PC B replies TCP-DATA using TCP-ACK by routing TCP-ACK through separated, dedicated link. By doing this, TCP-ACK will arrive to PC A at the shortest time. Eventually, the next window data are sent quicker. Bandwidth requirement for separated link is relatively low as TCP-ACK rate is much lower than TCP-DATA rate.



In order to evaluate the method, NS-2 simulations were performed. There are 20 nodes created in NS-2. Node 9 sends TCP-DATA to node 8. Normally, TCP-DATA and TCP-ACK are sent through the same network (bi-directionally) as shown in Figure 4(c). TCP-ACK will experience collision and queue as TCP-DATA. TCP-ACK reduction is achieved by using directional link from node 8 to node 9 as shown in Figure 4(d), so that TCP-ACK arrives without significant obstacles. Traffic set up was similar to section 4 with rates of 548 kbps to 581 kbps. Standard TCP protocol was tested in both conditions.

Simulation results are plotted in Figure 4(e). Delay increases to bit rate. The average delays of the TCP with the proposed scheme consistently lower than TCP with existing architecture. Modifications in WiMAX MAC layer shortens the bandwidth request time, so that TCP-ACK sent earlier and eventually the TCP delay decreases. The proposed method reduces TCP delay about 9.46% in average.

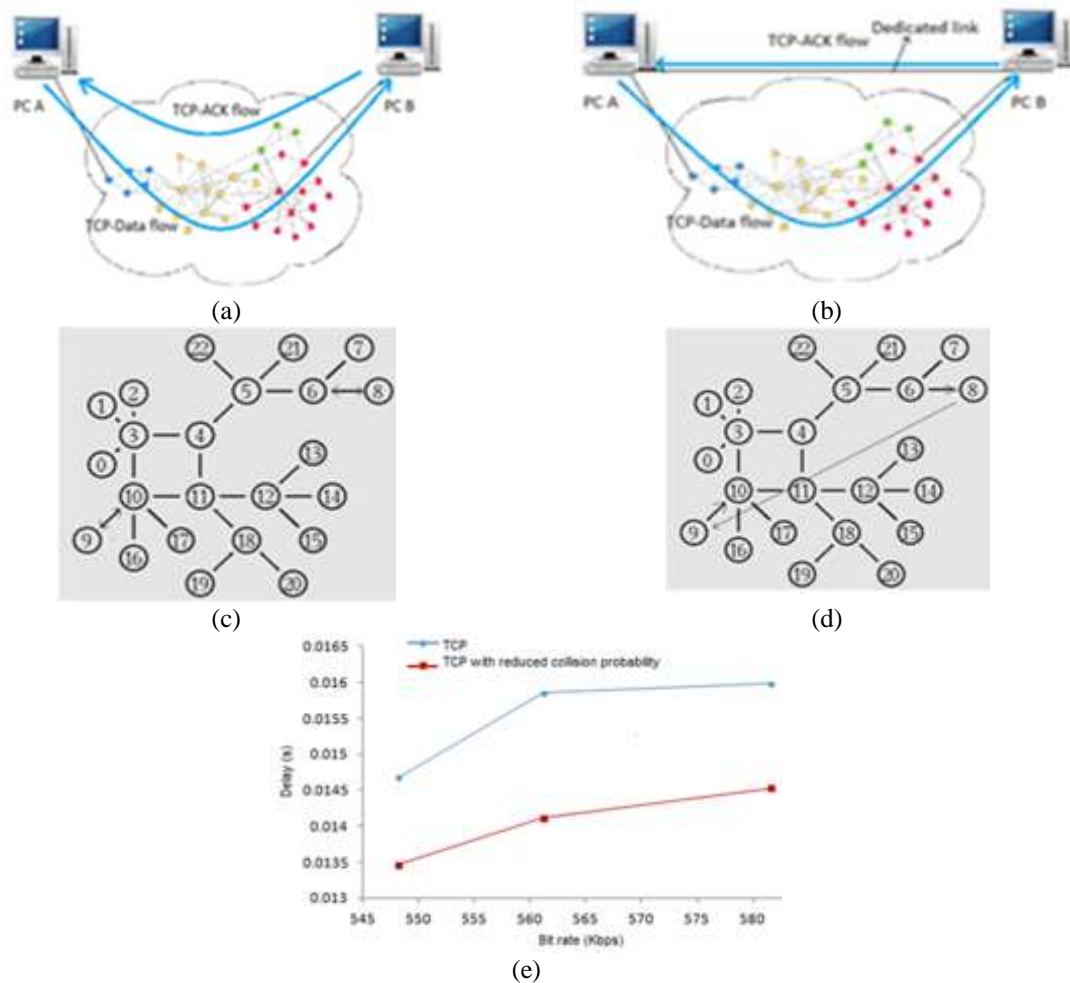


Figure 4. Proposed method impact to TCP in complex networks; (a) mixed TCP data and TCP-ACK traffics flow, (b) separated TCP-Data and TCP-ACK traffics flow, (c) complex network configuration for TCP flow, (d) configuration for TCP with reduced collision probability, and (e) delay reduction

#### 4. CONCLUSIONS

This paper proposes TCP-ACK collision reduction method to fasten TCP transmission. Mathematical analysis in contention network proved the method effectiveness. NS-2 implementations and simulations show that the proposed method works for 802.11, 802.16 and complex networks. The proposed method implementation in 802.11 network is achieved by reducing DIFS value by factor  $\alpha$  for TCP-ACK contained frames. The proposed method successfully reduces TCP delay about 11.86%. Delay reduction is also occurred when method is applied to TCP variants such as TCP-ELN. However, it is noted that the use of the proposed method in 802.11 should consider the maximum traffic, as saturated network may not be effective for the method to be implemented. Meanwhile, the proposed method implementation in 802.16

network is achieved by notifying bandwidth scheduler in base station when TCP-DATA is received in incoming packet re-assembler. As results, the average delay reduces about 28.04%. Existing fast-retransmission TCP that made use the proposed method also experiences delay reduction. The implementation of the proposed method in a complex network that contains more than one link requires a dedicated direct link for the route of TCP-ACK packet. Simulations show that TCP delay reduces about 9.46% when the proposed method applied. However, jitter and packet loss characteristics are not influenced by the method. Future work may explore a more efficient way to implement the proposed method in various networks.

## ACKNOWLEDGEMENT

This research is supported by Skema Penelitian Dasar of DIKTI, Indonesia, 2019.

## REFERENCES

- [1] Ramanjaneya R., Pakanati C., Padmavathamma M., "Performance enhancement of TCP friendly rate control protocol over wired networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, no. 6, pp. 2949-2954, 2016, doi: 10.11591/ijece.v6i6.pp2949-2954.
- [2] Muñoz R., et al., "Characteristics of mobile youtube traffic," *IEEE Wireless Com.*, vol. 21, no. 1, pp. 18-25, 2014, doi: 10.1109/MWC.2014.6757893.
- [3] Stéphanie M., Guillemin F., and Houdoin T., "Correlation between QoS and QoE for HTTP YouTube content in Orange cellular networks," *IEEE 9th Latin-American Conference on Communications*, pp. 1-6, 2017, doi: 10.1109/LATINCOM.2017.8240146.
- [4] Carvalho J., Veiga H., Pacheco C., and Reis A., "Extended research on effects of encryption and topology on performance of IEEE 802.11 a Laboratory Links," *IAENG IJCS*, vol. 44, no.2, 2017.
- [5] Huh J., "Reliable user datagram protocol as a solution to latencies in network games," *Electronics*, vol. 7, no. 11, p. 295, 2018, doi: 10.3390/electronics7110295.
- [6] Hussein W., "Performance evaluation of UDP, DCCP, SCTP and TFRC for different traffic flow in wired networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 6, pp. 3552-3557, 2017, doi: 10.11591/ijece.v7i6.pp3552-3557.
- [7] Wirz R., Marin R., Ferre M., Barrio J., Claver J., "Bidirectional transport protocol for teleoperated robots," *IEEE Trans. on Industrial Electronics*, vol. 56, no. 9, pp. 3772-3781, 2009, doi: 10.1109/TIE.2009.2025291.
- [8] Masirap M. et al., "Evaluation of reliable UDP-based transport protocols for Internet of Things (IoT)," *2016 IEEE Symposium on Computer Applications & Industrial Electronics*, pp. 200-205, 2016, doi: 10.1109/ISCAIE.2016.7575063.
- [9] Bisoy S., Pattnaik P., "Fairness between TCP Reno and TCP Vegas in wired and wireless network," *Int. J. of Comput. Syst. Engineering*, vol. 3, no. 1-2, pp. 14-26, 2017, doi: 10.1504/IJCSYSE.2017.083137
- [10] Abolfazli E., Shah M. V., "Dynamic adjustment of queue levels in TCP Vegas-based networks," *Electronics Letters*, vol. 52, no. 5, pp. 361-363, 2016, doi: 10.1049/el.2015.281
- [11] Kaur H., Singh G., "TCP congestion control and its variants," *Ad. in Computational Sciences and Technology*, vol. 10, no. 6, pp. 1715-1723, 2017. [Online]. Available: [http://www.ripublication.com/acst17/acst17\\_10n6\\_17.pdf](http://www.ripublication.com/acst17/acst17_10n6_17.pdf)
- [12] Al-Akaidi M., Mubarakah N., "A transport layer protocol for uplink WiMAX video streaming," *IJMUE*, vol. 10, no. 1, pp. 19-32, 2015.
- [13] Reddy N., Pakanati C., Padmavathamma M., "An enhanced queue management scheme for eradicating congestion of TFRC over wired environment," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 3, pp. 1347-1354, 2017, doi: 10.11591/ijece.v7i3.pp1347-1354.
- [14] Abdurrahman H., Sitompul O. S., Mubarakah N., "UDP-Lite enhancement Through checksum protection," *In IOP Conference Series: Materials Science and Engineering*, IOP Publishing, vol. 180, p. 012146, 2017, doi: 10.1088/1757-899X/180/1/012146.
- [15] Yoshiaki O., Nakamura M., Kawasaki Y., and Ode T., "Controlling TCP ACK transmission for throughput improvement in LTE-Advanced Pro," *IEEE Conf. on Standards for Communications and Networking*, 2016, pp. 1-6, doi: 10.1109/CSCN.2016.7785188.
- [16] IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN medium access control (MAC) and physical layer (PHY) spec.," [Online]. Available: <http://standards.ieee.org/getieee802/download/802.11-2007.pdf>.
- [17] Chatzimisios P., Boucouvalas A., and Vitsas V., "Packet delay analysis of IEEE 802.11 MAC protocol," *IEE Electronics Letters*, vol. 39, no. 18, pp. 1358- 1359, 2003, doi: 10.1049/el:20030868.
- [18] Tadayon N., Askari E., Aïssa S., Khabazian M., "A novel analytical model for service delay in IEEE 802.11 Networks," *IEEE Systems J.*, vol. 6, no. 4, pp. 627-634, 2012, doi: 10.1109/JSYST.2012.2185350.
- [19] Sakurai T., Hai Vu L., "MAC Access Delay of IEEE 802.11 DCF," *IEEE Trans. on Wireless Comm.*, vol. 6, no. 5, 2007, doi: 10.1109/TWC.2007.360372.
- [20] Periyasamy P., "H. 264/MPEG-4 AVC Video Streaming Analysis of AODV, OLSR and ZRP in MANET," *Int. J. of Advanced Networking and Applications*, vol. 10, no. 6, pp. 4076-4082, 2019, doi: 10.35444/IJANA.2019.10065.
- [21] Rani S., Suryakanth M., and Kaushik S., "Cross layer-based schemes for improving the performance of TCP in wireless networks," *Int. Conference on Computational Intelligence in Data Science (ICCIDS)*, 2017, doi: 10.1109/ICCIDS.2017.8272672.

- [22] Xinming Z., *et al.*, "TCP transmission rate control mechanism based on channel utilization and contention ratio in ad hoc networks," *IEEE Communication Letters*, vol. 13, no. 4, pp. 280-282, April 2009, doi: 10.1109/LCOMM.2009.082217.
- [23] Al-Akaidi, M. *et al.*, "A review on transport layer protocol performance for delivering video on an adhoc network," *IOP MSE.*, vol. 237, p. 012018, 2017, doi: 10.1088/1757-899X/237/1/012018.
- [24] "Test media," [Online]. Available: <http://media.xiph.org> [accessed: May 2019].
- [25] Marzouk H., Abdelmajid B., and Khadija S., "PSNR analysis of video transmission in VANETS, using NS2 and Evalvid framework," 2019. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-02298768>.
- [26] Ke C., Shieh C., Hwang W., Ziviani A., "An evaluation framework for more realistic simulations of MPEG video transmission," *Journal of Information Science and Engineering*, vol. 24, pp. 425-440, 2008. [Online]. Available: <https://jise.iis.sinica.edu.tw/JISESearch/pages/View/PaperSearch.jsf?searchBy=TITLE&title=An+evaluation+framework+for+more+realistic+simulations+of+MPEG+video+transmission>
- [27] Fakhrizal S., Al-Akaidi M., "A subjective scheduler for subjective dedicated networks," *In IOP MSE*, vol. 237, p. 012019, 2017, doi:10.1088/1757-899X/237/1/012019.
- [28] Farah N., Yasir A., and Elhadi Y., "Investigation and estimation of fast Retransmission method for reliable data transmission in TCP," *Int. Conf. on Computer, Control, Electrical, and Electronics Eng.*, 2018, pp. 1-5, doi: 10.1109/ICCCEE.2018.8515820.

## BIOGRAPHIES OF AUTHORS



**Suherman** is a Senior Lecturer in Electrical Engineering Department, Universitas Sumatera Utara. He completed B.Eng from this university in 2000 and obtained MSc and PhD from RMIT, Australia and De Montfort University, UK subsequently in 2009 and 2013. He also took some courses in Ilmenau TU, Germany and European Bioinformatics Institute, Cambridge, UK. His research interest covers communication network and protocols, computer network as well as applied communication techniques.



**Ali Hanafiah Rambe** completed his doctorate in the field of microstrip antenna in 2014. He has been lecturing within Universitas Sumatera Utara since 2004. Currently He serves as program secretary of Electrical Engineering master degree in Universitas Sumatera Utara. His research interest is related to antenna design, transmission lines and applied electromagnetic.



**Norshakila Haris** is currently a Senior Lecturer and a Head of Section with the Marine Electrical and Electronics Technology (MEET) Section. She received the B.Eng. (Hons) degree in Electronics Engineering from Cardiff University of Wales, U.K and the M.Eng. degree in Electrical-Electronics & Telecommunications from Universiti Teknologi Malaysia (UTM), Malaysia in 2006 and 2008, respectively. In 2017, she received her PhD degree in Electrical and Electronics Engineering from The University of Manchester, UK. She has been working with Universiti Kuala Lumpur Malaysian Institute of Marine Engineering Technology (UniKL MIMET) since July 2008.



**Anhar** received his Bachelor and Master Degree in Electrical Engineering from Universitas Sumatera Utara and Bandung Institute of Technology, Indonesia, in 2000 and 2008, respectively. He received his PhD degree in School of Engineering and Design at Brunel University, London, UK. Currently, He is a lecturer in Riau University, Pekanbaru, Indonesia. His research interests include routing and medium access control protocols for Wireless Sensor Network (WSN) and Internet of Things (IoT).