Performance of TCP CUBIC and TCP BBR on IEEE802.11s Mesh Network

Katsuya Atsuta, Satoshi Kouya, and Naoshi Sakamoto
Tokyo Denki University, Tokyo, Japan
Email: 18kmc03@ms.dendai.ac.jp; kouya@mail.dendai.ac.jp; sakamoto@c.dendai.ac.jp

Abstract —In this study, IEEE802.11s mesh networking was formulated as it expected to be widely used because of its convenience. The properties of IEEE802.11s mesh networking were investigated and the results revealed its faulty communication performance in a multi-hop network. This could be explained based on Mathis's theory for loss-based TCP congestion control algorithm that suggests that with an increase in delay, the loss probability decreases the communication performance. Recently, Google proposed TCP Bottleneck Bandwidth and Round-trip propagation time (BBR), both of which may tolerate a high bit error rate. This algorithm does not follow Mathis's model. In this study, we measure the performance of TCP CUBIC and TCP BBR on a mesh network, followed by an evaluation of the properties of this mesh network.

Index Terms—IEEE802.11s, mesh network, TCP BBR, throughput

I. Introduction

Wireless LANs, especially WiFi (IEEE802.11 [1]), have spread widely. IEEE802.11 defines the infrastructure and ad-hoc modes. In 2012, IEEE802.11s mesh networking was proposed as an expansion of the ad-hoc mode. This introduced the notion of a mesh point (MP). The MPs form a mesh network by communicating, routing, and relaying with each other. While the former access point requires a wired connection to the Internet, a mesh network can be extended by adding MPs within reach, facilitating its extension without wired connections. IEEE802.11s has been in use since 2003. The open80211s group developed a Linux driver that was officially installed in the kernel 2.6.26.

The authors investigated the properties of IEEE802.11s mesh networking through experiments [2]. They evaluated the communication speed by using iperf command, which measures the communication performance of transport control protocol (TCP). Further, it was determined that the performance decreases by obeying Mathis's model [3]. This could be due to the increase of the round-trip time and bit error rate for hops.

Recently, Google proposed TCP Bottleneck Bandwidth and Round-trip propagation time (BBR) that might tolerate high bit error rate [4]. As this algorithm does not obey Mathis's model, it may be effective on a mesh network. Thus, in this study, we measure the performance of TCP CUBIC and TCP BBR on a mesh network, followed by the evaluation of the properties of the mesh network. The remainder of this paper is organized as follows: Section II explains IEEE802.11s, TCP, and congestion control algorithms of TCP. Section III introduces the related works. Section IV discusses and expands Mathis's model. Section V explains our experiments and the results. Section VI discusses the experimental results. Further, Section VII concludes the paper.

Manuscript received December 8, 2019; revised July 3, 2020.
Corresponding author email: sakamoto@c.dendai.ac.jp.
doi: 10.12720/jcm.15.8.639-645

II. Preliminaries

A. *IEEE802.11s*

IEEE802.11 [1] is a wireless network that uses 2.4 GHz and 5 GHz (IEEE802.11ad uses 60 GHz). Its physical layers are defined in 11b, 11g, 11n, 11ac, 11ad, and 11ax. IEEE802.11 defines the infrastructure and ad-hoc network modes. In the infrastructure mode, one access point connects with the terminals. This mode is the one that is widely used. On the other hand, in the ad-hoc mode, the terminals have peer-to-peer connection with each other. This mode is usually used by game terminals and printers. Further, IEEE802.11s is the extended version of the ad-hoc network mode.

IEEE802.11s is a specification of mesh networking, which was officially mentioned in 2012. In IEEE802.11s, once a node joins a mesh network, the network is organized using dynamic routing based on the metrics calculated in terms of each other. At most 32 MPs can join the mesh network. The MP also plays the role of an access point.

B. *TCP*

TCP is usually used to exchange a file over the Internet. It is an end-to-end protocol to exchange a file on a network. It detects packet losses and retransmits them, and also controls the band width. To evaluate network traffic, the file transport speed of TCP is used, which is measured using Iperf software.

There are several TCP congestion control algorithms. These are classified into the following three groups: loss-based, delay-based, and hybrid. A loss-based algorithm considers congestion by detecting a packet loss. A delay-based algorithm considers congestion by detecting if the established time has exceeded the round-trip time. A hybrid algorithm combines the loss-based and delay-based algorithms.

Mathis *et al*. demonstrated the throughput model of loss-based TCP congestion control algorithm [3]. By considering MSS to be the maximum segment size, RTT to be the round-trip time, and p to be the probability of packet loss, the bandwidth (BW) of TCP is shown using the following:

$$BW < \left(\frac{MSS}{RTT}\right)\frac{1}{\sqrt{p}} \quad (1)$$

Moreover, Antunes et al. confirmed that TCP CUBIC obeys Mathis's model [5].

C. TCP Congestion Control Algorithms

There are several TCP congestion control algorithms. In this section, we introduce TCP CUBIC and TCP BBR.

TCP CUBIC was proposed in 2004 as a loss-based congestion control algorithm, which is a modified version of *Binary Increase Congestion control* (BIC)-TCP [6]. This enters the congestion control mode when it detects a packet loss such as TCP New Reno. However, while TCP New Reno largely narrows the congestion window, followed by widening it like a linear function, TCP CUBIC narrows the congestion window slightly, followed by widening it like a cubic curve to widen it gradually around the maximum width. Recently, many operating systems, such as Linux 2.6.19 and later, and Windows 10.1709 and later, have installed TCP CUBIC.

TCP BBR has been developed as a new congestion control algorithm by the Google research team [4]. It is called neither loss-based nor delay-based, but congestion-based.

A loss-based algorithm progressively widens the congestion window monotone until it causes packet losses, then, narrows it. Subsequently, the behavior of the bottleneck network just before the occurrence of a packet loss is considered. It is assumed that eventually the congestion window reaches the communication capacity. As long as packet loss does not occur, the loss-based algorithm continues to widen the congestion window. Further, the router starts to buffer packets. This implies that the communication speed slows down. Finally, only when the buffer reaches the limit, a packet loss occurs. Therefore, the congestion window when a packet loss occurs is wider than the congestion window at the communication capacity of the bottle neck. However, the round-trip time when a packet loss occurs is slower than the round-trip time at the communication capacity of the bottle neck, indicating that the throughput decreases.

BBR, congestion-based algorithm, measures the smallest round-trip time in long span, and measures the communication rate in short span. Further, it calculates the congestion window ratio for the maximum size of the congestion window in short round-trip time by comparing the amount of in-flight data (where their ACK packets are not received yet). BBR does not react to packet losses, instead the algorithm calculates the optimized size of the congestion window for the average behavior of the network. Thus, BBR might have tolerance to simple packet losses. BBR is available for Linux 4.9 and later.

III. Related Works

A. IEEE802.11s

Firstly, in this section, previous studies about IEEE802.11s are described. These studies involved operational experiments. In the experiments, while the distances between the nodes were fixed, the number of hops was set to be variable. Moreover, they considered line and square grid shapes as the network topologies.

Lv *et al*. simulated a mesh network where 16 nodes were placed in a 4 by 4 grid at intervals of 100 m by using ns-3 [7]. Further, they showed that the throughput decreased, and both the round-trip time and packet loss ratio increased when the packet flow increased.

González et al. made four nodes by using Raspberry Pies and wireless USB adapters [8]. They measured the response time of ping, throughput of TCP, and loss ratio of User Datagram Protocol (UDP). Finally, they compared the results with the simulation of ns-3.

Lin *et al*. measured the TCP throughput for nine real nodes in 11b, 11g, and 11n modes [9]. They considered the following three topologies: five nodes placed at intervals of 0.5 m in a line shape; nodes arranged in a 3 by 3 grid at intervals of 0.5 m; and nodes arranged in a 3 by 3 grid at widths and heights of 25 m with the use of a building.

Rethfeldt *et al*. added a register to the antenna of nodes to reduce the antenna gain, then arranged mesh nodes in a line shape, and finally, measured the throughput of both TCP and UDP among the four nodes [10].

Robitzsch *et al*. arranged six nodes in a line shape, and then, measured the throughput of both TCP and UDP in 802.11g and 802.11n on 2.4 GHz and in 802.11n on 5 GHz [11].

Hiertz *et al*. introduced IEEE802.11s and the open80211s project [12] [13]. Further, they evaluated the driver developed by the open80211s project by measuring the discovery time and throughput from one to 11 hops.

Sakamoto *et al*. examined the property of TCP communication on multi-hop in a real environment [2]. Moreover, they identified a difference between the open80211s protocol and Google WiFi. However, they also determined that it is difficult to effectively use a mesh network beyond the adopted three hops either.

B. TCP BBR

Few researchers evaluated the property of TCP BBR. Cardwell *et al*. compared TCP BBR to TCP CUBIC based on the goodput for the loss ratio obtained through experiments [14]. While in TCP CUBIC, goodput decreases by less than a half when the loss ratio increases from 0.001% to 0.01%; moreover, the goodput decreases to approximately zero when the loss ratio is 0.1%. TCP

BBR has almost constant goodput until the loss ratio is 5%. This shows that TCP BBR is tolerant to packet losses.

Hock *et al*. observed the behavior of plural BBR flows for changing their round-trip time and the buffer size [15]. They found that plural BBR flows increase delay, result in packet losses, and create unfairness among the flows. They also performed experiments for CUBIC flows. Further, they showed that BBR flows drive CUBIC flows away and occupy bandwidth.

Atxutegi *et al*. measured the round-trip time of TCP BBR on 4G network under three conditions where the bandwidth and buffer size were different [16]. Further, they identified that although BBR works quite well with a sufficient buffer size, with a small buffer size, it causes longer delays than the existing congestion control algorithms.

IV. Expansion of Mathis's Model

In this section, we describe Mathis's model, which is based on the loss-based TCP congestion control algorithm.

To analyze the traffic for TCP Reno microscopically, the following were assumed [3]:

- A cycle was called in the fast recovery mode from start for a packet loss to occur.
- All ACKs were returned until a packet loss occurred. A round trip was described as the time from packets of the size of the window until all ACKs were returned.
- The number of round trips in a cycle was R. Further, the total number of sent packets could be calculated.
- The packet loss rate was p. Further, it can be said that a packet loss occurred every $\frac{1}{p}$ packets on an average. Thus, it is assumed that a cycle ends after $1/p$ packets are sent.
- *MSS* denotes the packet size and *RTT* denotes the round-trip time. Further, we can estimate the throughput by dividing *MSS* (the total number of packets) by *RTT* (the number of round trips).

Thus, by denoting R as the number of round trips, and $\frac{1}{p}$ as the total number of packets, we can estimate the throughput using the following:

$$BW = \frac{MSS \cdot \frac{1}{p}}{RTT \cdot R} \tag{2}$$

Now, in case of TCP Reno, we have $R = \frac{W}{2}$, and $\frac{1}{p} = \frac{3}{8W^2}$. These imply $R = \sqrt{\frac{2}{3p}}$. Thus, we have the following:

$$BW_{Reno} = \frac{MSS \cdot \frac{1}{p}}{RTT\sqrt{\frac{2}{3p}}} = \frac{MSS}{RTT}\sqrt{\frac{3}{2p}} \tag{3}$$

Now, we expand this notion. Let *p(w)* be the packet loss rate function for the window size. Let us assume that the window width function *w(r)* is a monotone increasing sequence. By allowing the approximation of summation using integral calculus, we have the following:

$$\min_{0 \le r < R} \frac{1}{p\left(w(r)\right)} = \int_0^R w(r)\,dr \tag{4}$$

Moreover, by assuming that p(w) is also a monotone increasing sequence, we can approximate the average number of packets in a cycle by using an inverse number of the packet loss rate on the maximum window size. This would give the following:

$$\frac{1}{p\left(w(R)\right)} = \int_0^R w(r)\,dr \tag{5}$$

Based on R satisfying (5), the throughput can be estimated using the following:

$$BW = \frac{MSS\int_0^R w(r)\,dr}{RTT \cdot R} \tag{6}$$

V. Performance Tests

To determine the communication property of IEEE802.11s, we performed various experiments.

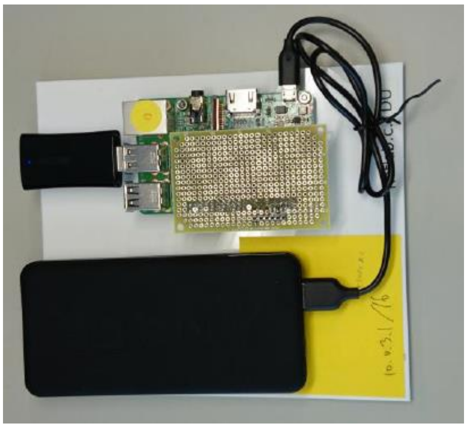

Fig. 1. Mesh node developed by us.

We developed and used our own mesh nodes. These mesh nodes consist of Raspberry Pi3 with Raspbian OS, Buffalo WLI-UCG301N (IEEE802.11n) as the physical layer, and a battery (Fig. 1). The driver was a rt2x00. Further, we used open80211s for constricting a mesh network. We referred to this as a node. We performed the following experiments in a radio wave darkroom and a laboratory room in a University to simulate an actual indoor environment. We arranged the nodes in a line shape (Fig. 2).

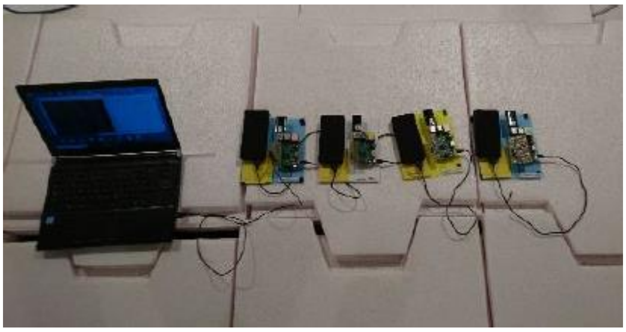

Fig. 2. Mesh nodes in the radio wave darkroom.

The laboratory room is situated in a building of reinforced concrete. There are many similar rooms near it. Consequently, there are many Wi-Fi access points near the room. Fig. 3 shows the beacon map of a 2.4 GHz band in the room. We used channel six.

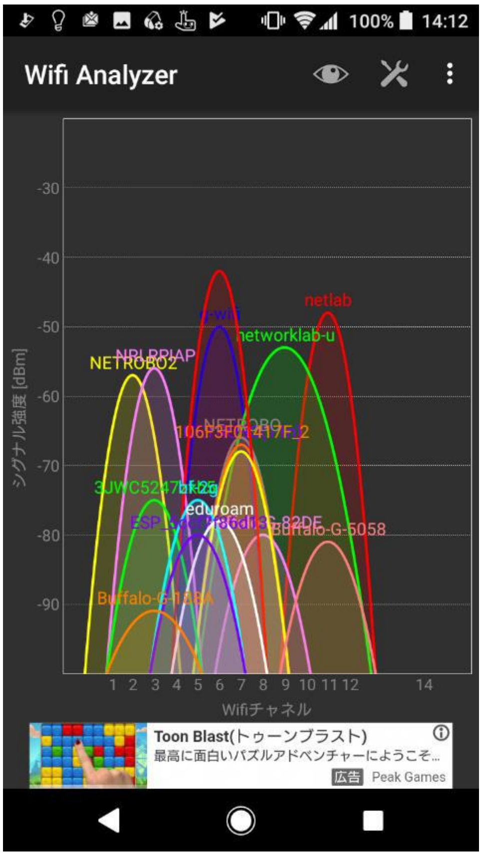

Fig. 3. Beacon map in an actual indoor environment.

Experiment 1: We measured the throughput for constant bit rate by UDP in both the radio wave darkroom and an indoor environment.

The results of this experiment are depicted in Figs. 4 and 5.

Experiment 2: In this experiment, we set the configuration of the nodes such that they were connected to only their neighboring nodes in order to construct a mesh network with line topology.

Further, we measured the TCP throughput of the mesh network in 100 s from one to seven hops for TCP CUBIC and TCP BBR by using Iperf3.

This experiment was conducted in the radio wave darkroom and an indoor environment.

The result obtained is shown in Fig. 6.

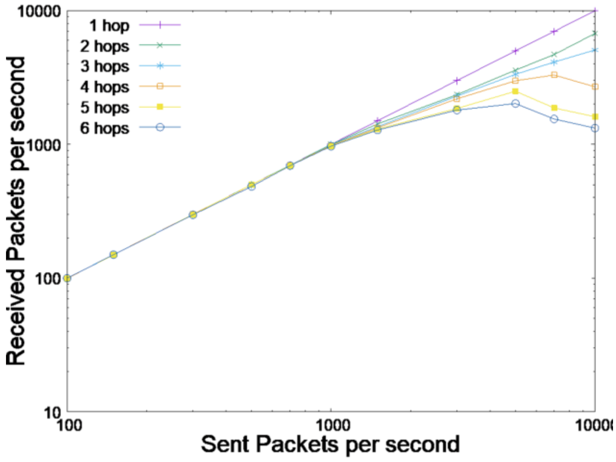

Fig. 4. Constant bit rate in the radio wave darkroom.

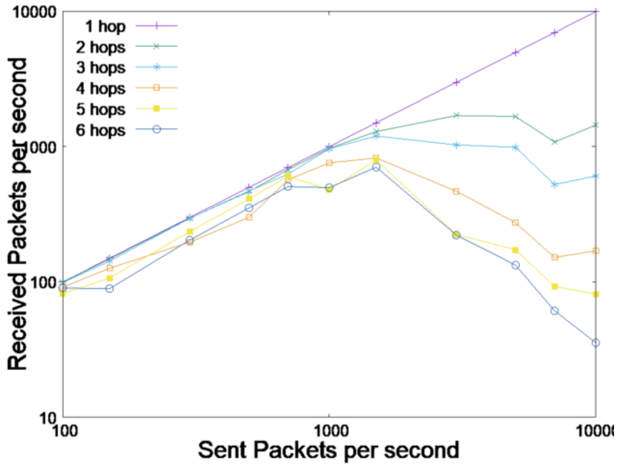

Fig. 5. Constant bit rate in an indoor environment.

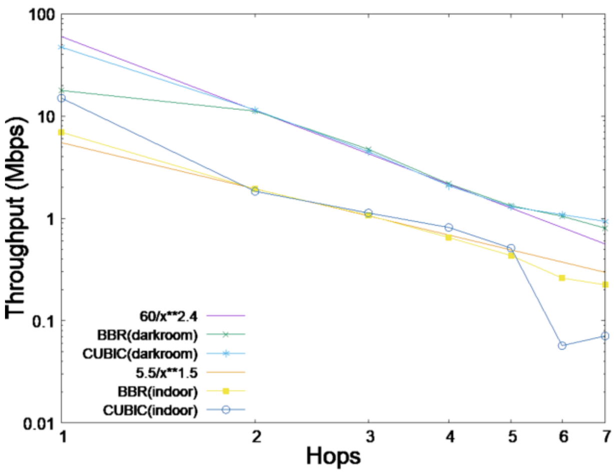

Fig. 6. TCP throughput for hops.

Based on the results, it is noted that both TCP CUBIC and TCP BBR demonstrate similar straight lines in the log-log graph. Therefore, an approximate curve was obtained using the least square method for the part with good linearity between two and five hops. As a result, the approximate curves in the darkroom and indoor environment were (7) and (8), respectively. These are also shown in Fig. 6.

$$BW_{darkroom} \sim \frac{60}{n^{2.4}} \tag{7}$$

$$BW_{indoor} \sim \frac{5.5}{n^{1.5}} \tag{8}$$

VI. Discussion

Firstly, based on the result of Experiment 1, the properties of mesh network hops are discussed. By observing Figs. 4 and 5, we note that there are two modes. In one mode, when the input bit rate increases, the throughput also increases. In the other mode, though the input bit rate increases, the throughput decreases. The mode where the throughput decreases can be understood by assuming that every packet requires a constant time to process. However, as our focus is on the TCP throughput, we concentrate only on the mode where the throughput increases. While according to Fig. 4, the packet loss rate is quite low, in reality, the packet loss occurs normally. Fig. 7 shows the packet loss ratio.

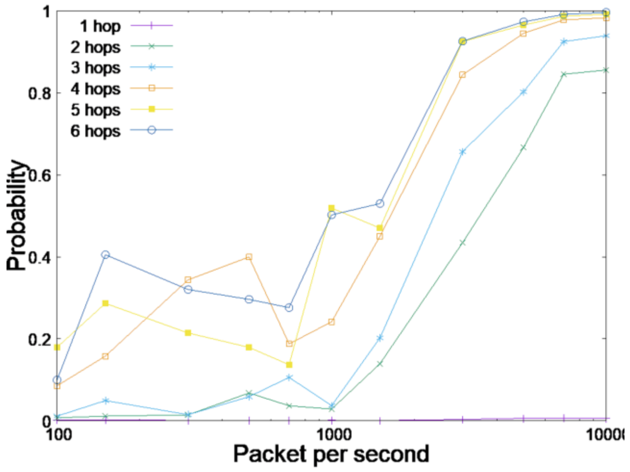

Fig. 7. Packet loss ratio for constant bit rate.

According to Fig. 7, more than one hop mesh network causes packet losses. By increasing the input bit rate, the packet loss rate can also be increased. Further, we calculate the approximate line $p_n(r)=a_n \log r b_n$ to the packet loss rate on a semi-log paper up to 1000 packets per second using the least square method. Fig. 8 shows the parameters a_n and b_n. Moreover, according to Fig. 8, as the obtained coefficient changes linearly with respect to the number of hops, (9) is assumed as a linear expression with respect to the number of hops.

$$p(w,n)=a(n-b)(\log w-c)+d \tag{9}$$

Thus, we induce the coefficients of (9) by using the least square method as (10).

$$\begin{aligned} a &= 0.0179, b = 1.15, \\ c &= 1.92, d = -0.0167 \end{aligned} \tag{10}$$

We plot the graphs of these approximation lines in Fig. 9.

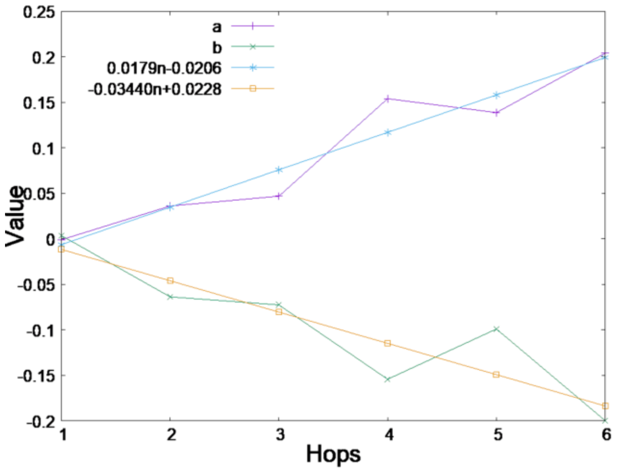

Fig. 8. Approximation coefficients for hops.

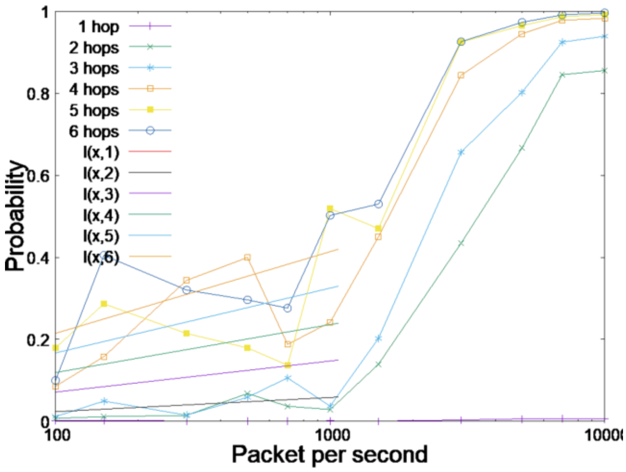

Fig. 9. Packet loss ratio for constant bit rate with theoretical curve.

Next, let us consider the TCP throughput. Let $W_{\max}$ denote the maximum window size, *w(r)* be the window size for round *r*, *p(w)* be the packet loss rate, and *R* denote the number of round trips, where $w(R)=W_{\max}$.

By Further, TCP CUBIC window size is defined as follows:

$$w(r)=C(r-K)^3+W_{\max} \tag{11}$$

where $C=0.4$, $\beta=0.2$, and $K=\sqrt[3]{\beta W_{\max}/C}$.

By substituting (9) and (11) in (5), we obtain the following:

$$\int_0^R w(r)\,dr = \int_0^K \left(C(r-K)^3 + W_{\max}\right)dr$$
$$= KW_{\max}\left(1-\frac{\beta}{4}\right)$$
$$= \left(W_{\max}\right)^{\frac{4}{3}}\sqrt[3]{\frac{\beta}{C}}\left(1-\frac{\beta}{4}\right) \tag{12}$$

Based on the condition where (12) is equal to (9), by substituting (9) for (12), and assuming that $\log W_{\max}$ is constant when solving the equation, we can obtain the relationship between $W_{\max}$ and *n* as follows:

$$W_{\max} = \sqrt[4]{\frac{64C}{\beta(4-\beta)^3}}$$
$$\times\left(a(n-b)\left(\log W_{\max}-c\right)+d\right)^{-\frac{3}{4}}$$
$$\propto n^{-1.75} \tag{13}$$

Based on the fact that when the number of hops increases, RTT also increases, and $W_{\max}$ is proportional to $n^{-0.75}$, we finally obtain the following:

$$BW \propto n^{-1.75} \tag{14}$$

Next, we discuss Experiment 2. Firstly, the reason behind the result not obeying Mathis's model is because our network has a high loss probability. However, while the graph in [3] lost linearity for high loss probability, Fig. 6 does not lose linearity.

Further, we propose a conjecture that a mesh network has an optimal window size for TCP. Lv *et al*. studied the bit rate and loss probability for multi-hop mesh networks. They found that the throughput at a lower bit rate than its limit is greater than the throughput closer to the bit rate limit.

While Cardwell *et al*. observed that TCP CUBIC and TCP BBR have different properties at a high packet loss rate [4], through our experiments, we observed that TCP CUBIC and TCP BBR have similar properties, as seen in Fig. 6. Therefore, we conclude that when the number of hops increases, the decrease in TCP throughput on multi-hop mesh networks does not occur due to an increase in packet losses. Particularly, it needs to be noted that TCP BBR is the algorithm that adjusts the window size to optimize the throughput despite packet losses. Though TCP CUBIC and TCP BBR are different algorithms, both identify the same window size. Thus, it can be concluded that two superior TCP algorithms can find an identical optimum window size.

VII. Conclusions

In this study, the traffic-related property of TCP congestion control algorithms has been evaluated on 802.11s mesh networks through experiments on a mesh network including mesh nodes developed by us. It was determined that the throughput of both TCP CUBIC and TCP BBR drops drastically when the number of hops increases. This is different from the observation made by Cardwell *et al*. [4] where TCP BBR had a higher throughput than TCP CUBIC with high packet loss. Thus, it was found that neither CUBIC nor BBR, which are the current TCP algorithms, can suppress the decrease in throughput with respect to the number of hops in the mesh network.

In the future, we would like to determine a method for effective communication on a mesh network.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Sakamoto conducted this research. Atsuta and Sakamoto investigated and analyzed. Then they proposed and verified the hypothesis. Atsuta and Kouya proposed and performed the experiments. Atsuta and Sakamoto wrote the paper. All authors had approved the final version.

References

[1] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.11, 2016.

[2] N. Sakamoto and K. Atsuta, "Measurement of the performance of IEEE 802.11s," presented at the 11th International Conference on Computer Science and Information Technology (ICCSIT 2018), Paris, France, December 21-23, 2018.

[3] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," *SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 3, pp. 67-82, Jul. 1997.

[4] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "BBR: Congestion-based congestion control," *Queue*, vol. 14, no. 5, pp. 50:20-50:53, Oct. 2016.

[5] J. Antunes. (Jul. 2013). Validating a very simple model for TCP throughput. [Online]. Available: https://blog.thousandeyes.com/a-very-simple-model-for-tcp-throughput/

[6] S. Ha, I. Rhee, and L. Xu, "Cubic: A new TCP-friendly high-speed TCP variant," *SIGOPS Oper. Syst. Rev.*, vol. 42, no. 5, pp. 64-74, Jul. 2008.

[7] P. Lv, X. Wang, M. Xu, and Y. Chen, "Network-leading association scheme in IEEE 802.11 wireless mesh networks," in *Proc. IEEE Int. Cnf. Commun.*, 2011, pp. 1-5.

[8] V. González, A. L. Santos, C. Pinart, and F. Milagro, "Experimental demonstration of the viability of IEEE 802.11b based inter-vehicle communications," in *Proc. TridentCom'08. ICST*, 2008, pp. 1:1-1:7.

[9] Y. D. Lin, S. L. Chang, J. H. Yeh, and S. Y. Cheng, "Indoor deployment of IEEE 802.11s mesh networks:

Lessons and guidelines," *Ad Hoc Netw.*, vol. 9, no. 8, pp. 1404-1413, Nov. 2011.

[10] M. Rethfeldt, B. Beichler, H. Raddatz, F. Uster, P. Danielis, C. Haubelt, and D. Timmermann, "Mini-mesh: Practical assessment of a miniaturized IEEE 802.11n/s mesh testbed," in *Proc. WCNC*. IEEE, 2018, pp. 1-6.

[11] S. Robitzsch, C. Niephaus, J. Fitzpatrick, and M. Kretschmer, "Measurements and evaluations for an IEEE 802.11a based carrier-grade multi-radio wireless mesh network deployment," in *Proc. Winicom2009*, 2009, pp. 272-278.

[12] open80211s. [Online]. Available: https://www.open80211s.org/

[13] G. Hiertz, D. Denteneer, S. Max, R. Taori, J. Cardona, L. Berlemann, and B. Walke, "IEEE 802.11s: The WLAN mesh standard," in *Proc. IEEE Wireless Communications*, 2010, pp. 104-111.

[14] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "BBR: Congestion-based congestion control," *Commun. ACM*, vol. 60, no. 2, pp. 58-66, Jan. 2017.

[15] M. Hock, R. Bless, and M. Zitterbart, "Experimental evaluation of BBR congestion control," in *Proc. ICNP2017*. 2017, pp. 1-10.

[16] E. Atxutegi, F. Liberal, H. K. Haile, K. J. Grinnemo, A. Brunstrom, and A. Arvidsson, "On the use of TCP BBR in cellular networks," *Comm. Mag.*, vol. 56, no. 3, pp. 172-179, Mar. 2018.

Katsuya Atsuta was born in Japan in 1996. He graduated from Tokyo Denki University. His area of interest lies in wireless mesh networks.

Satoshi Kouya was born in Japan in 1969. He received the B.E., M.E., and Ph.D. degrees from Tokyo Denki University, Tokyo, Japan, in 1993, 1995, and 1998, respectively. From 1999 to 2000, he worked as a researcher at Polytechnic University, New York, USA. Since then, he has been associated with Tokyo Denki University, where he is currently working as an Assistant Professor. His primary areas of research interest lies in wireless power transmission, WiFi security, and small antenna system. Dr. Kouya is a member of the Institute of Electronics, Information and Communication Engineers, the Japan Society for Simulation Technology, and the Institute of Electrical and Electronics Engineers.

Naoshi Sakamoto was born in Japan in 1964. He graduated from the University of Electro-Communications, Japan, in 1987, and received the M.S. and doctoral degree from Tokyo Institute of Technology in 1989 and 2001, respectively.

From 1992 to 1997, he was a research assistant at the computer center of Hitotsubashi University. From 1997 to 2001, he was a research assistant at Tokyo Institute of Technology. Since 2001, he worked as an Associate Professor at Tokyo Denki University. Thereafter, since 2014, he has been a Professor with the university. He received the IEICE excellent paper award in 1999. His research interests are distributed algorithms, randomized algorithms, and complexity theory.