RESEARCH ARTICLE OPEN ACCESS

An improvement of REM Active Queue Management algorithm using the Hedge Algebra Controller applied in a wireless TCP IP network

Thu May Duong*, Mai Thuong Duong Thi ** Van Hoan Mai ***
**(Faculty of Information Technology, ICTU, Thai Nguyen, Vietnam*
*** (Faculty of Information Technology, ICTU, Thai Nguyen, Vietnam*
**** (Faculty of Information Technology, ICTU, Thai Nguyen, Vietnam*

ABSTRACT
The Random Exponential Marking (REM) algorithm is a traditionally proposed Active Queue Management (AQM) algorithm for congestion control on a wired TCP network, where packet loss is an indication of congestion. However, in wireless TCP networks, packet loss is also caused by communication and processing at the wireless links. Therefore, in order to apply REM algorithm on wireless TCP network, in this article, we propose two solutions to improve REM active queue management algorithm. Solution 1 is to use an Hedge Algebra Controller (HAC) so that the REM algorithm can more adapt to the dynamics of the TCP network. Solution 2 is to separate packet loss caused by wireless link from packet loss caused by congestion control. This solution uses an agent that has a combination of an HAC and Explicit Congestion Notification (ECN) at the interface between the wired and wireless network. The simulation results have demonstrated the effectiveness and adaptability of the proposals.

Keywords - Active Queue Management, Hedge Algebra Controller, Random Exponential Marking, Wireless TCP

Date of Submission: 26-03-2021 Date of Acceptance: 09-04-2021

I. INTRODUCTION

These days, all communication networks tend to converge on a single TCP / IP network. Therefore, this network must accommodate both wired and wireless services. This leads to some traditional TCP/IP techniques that need to be improved to accommodate this convergence including congestion control through active queue management [1].

Congestion control algorithms are installed in routers on wired TCP networks to proactively remove packets to prevent and overcome congestion. When the packet loss is detected, the TCP terminals assume these packets have been discarded due to the congestion. Flow control algorithms will be applied to throttle the terminals to reduce the size of the congestion window, thereby reducing the amount of information entering the TCP network. [2].

However, in wireless TCP networks, in addition to packet loss due to congestion, packets are lost mainly due to higher bit error rates, fading, radio channel interference, intermittent connections, and handoff process. If there is no solution for the terminal to distinguish between these two types of packet loss, even the loss due to transmission, the terminal will actively reduce the speed, thereby reducing the efficiency of the network. To overcome this, a lot of studies have been conducted individually on flow control in wired networks and on interference suppression in wireless networks. [3].

REM is a traditional queue management algorithm that has many advantages due to packet rejection based on both the queue length and the rate of incoming information flow [4]. However, REM needs to be improved to be applicable over mixed networks, including both wired and wireless.

First, the REM needs to be more adaptive to the dynamics of the TCP network, when the various types of information and the number of information flows in / out of the network change continuously. The solution to this problem is to use an HAC. This is a solution that has been applied on many different control objects and is said to be suitable for objects with variable dynamics properties.

Second, to use REM in a wireless TCP network, it is necessary to separate loss recovery from congestion control in order to improve the overall network performance. This problem is achieved when using an agent that exists between the interface of the wired and the wireless network. On this Agent, in addition to using HAC, it also integrates ECN method to distinguish packet loss due to congestion or transmission error.

In the next sections of this article, after analyzing the problems that exist on the wireless TCP network and their corrective solutions, we will describe in detail how to proceed with these two solutions, analyzing the simulation results to demonstrate the effectiveness and adaptability of the two solutions.

II. WIRELESS TCP NETWORK PROBLEMS AND SOLUTIONS

2.1. Features of the wireless TCP network

In a wireless network, mobile terminals expect data transport services to be provided as fixed terminals. However, using TCP on a network that contains wireless links causes the network throughput to be severely reduced due to the following important factors: [6], [7]:

Capacity limits: Spectrum is a valuable resource in radio communications. The spectrum of the radio channel will limit the maximum rate at which packets can be transmitted over wireless channels. Meanwhile, to ensure reliable transmission, we must use forward error coding (FEC) and automatic repeat request (ARQ) techniques. To do this requires an additional bandwidth limitation which is inherently limited. Error control mechanisms, on the other hand, can only partially correct failures in the wireless environment.

High bit error rate: Wireless links have a much higher bit error rate than wired links. Faulty packets cause timeouts in the TCP sender, which increases frequent retransmission speed, which is always associated with a slow start period in TCP. Repeated errors can lead to low network throughput. Furthermore, the wrong packets generate end-to-end retransmissions that overwhelm the traffic throughout the network. On the other hand, the error on the wireless link is usually a cluster error caused by fading, loss of connection depending on the location of the mobile device. With the high bit error rate characteristic of wireless transmission, we have to choose the much smaller Maximum Transmission Unit than the data unit in wired networks. The small size of this data unit also leads to the use of small packets in parts of the network with a wired link, although larger data packets may be used on wired links. . The consequence of this is that costs for data packet processing (data encapsulation, data separation ...) at the nodes on the transmission line increase and decrease the throughput of the network.

Higher end-to-end latency: Obviously the presence of a wireless link on a terminal-to-end connection slows down data throughput and increases end-to-end latency due to limited capacity. Furthermore, retransmissions caused by erroneous packets also add an additional delay that should be taken into account when calculating overall latency. In addition, wireless systems often use encoding and interleaving techniques to overcome high busty failure rates. These methods also increase wireless latency.

Disconnect frequently: Disconnecting means the link quality is temporarily poor, resulting in the inability of the mobile device to receive the information. In mobile wireless communication systems, handover often results in changes in packet delay or loss of packets. This can cause a disconnection that extends from a number of packets up to a few frames. If a cell has a high density of users, such as a pico-cell, some connections (of new calls) may not receive bandwidth for a long time.

2.2. Some solutions to overcome

To deal with problems in wireless networks, several new versions of TCP that include numerous solutions have been introduced. Proposals to improve TCP performance in wireless networks can be divided into three categories: end-to-end method, connection spliting method, and link layer method.

The end-to-end methods attempt to make the TCP sender handle the packet loss on the radio channel through two techniques. First, they use some form of selective confirmation, such as SACK or quick retransmission, to allow the sender to recover from multiple packet loss in a single window without having to time out. The main advantage of these methods is that it reduces the disconnection time caused by the handover and it can be used to tune TCP to a mobile computing environment without terminal modification. Second, they try to make the sender differentiate between congestion and other forms of loss by using an ECN mechanism [8], [9].

The split method of connection attempts to separate loss recovery over the wireless link from the wired link because the two links have completely different characteristics. These solutions include Indirect-TCP, M-TCP. The split methods generally cause high packet overhead when using protocol overlap. Since then, a lot of buffer capacity in the base station is required [10].

Link layer methods attempt to conceal link-related losses with a TCP sender by using ARQ and FEC techniques over the wireless link. ARQ and FEC are successful in reducing BER and they naturally fit into the layered architecture of the protocol stack. However, all of these methods limit the available wireless bandwidth.

III. CONSTRUCTION OF HAC TO IMPROVE REM ALGORITHM

3.1 The REM algorithm

Like other AQM algorithms, REM is used to achieve high transmission efficiency, low average packet loss rate, and small queue delay. The special feature is that the REM uses a unit of measure of congestion called price. This unit is calculated from the difference in the queue of the instantaneous queue length with the target queue length, the difference in the rate of information to the line capacity (1) [4], [5].

$$p(kT) = \max(0, p(k-1)T + \gamma(\alpha(q(kT) - TQL) + x(kT) - c)) \quad (1)$$

Where c is the route capacity (packet rate by time), q (kT) is the length of the queue, and x(kT) is the rate at which the packet arrives. The probability of REM's packet marking or drop is calculated using (2):

$$prob(kT) = 1 - \varphi^{-p(kT)} \text{ where } \phi > 1 \text{ is a constant} \quad (2)$$

Table 1. REM parameters

Parameters	Description
q_{ref}	Reference queue length
α and γ	Constants for computing the "congestion price"
φ	Constant for computing the drop probability
T	Sampling interval

3.2 Building the controller according to the Hedge Algebra (HA) approach

HA was developed in [11] to model semantics based on the order of linguistic words (linguistic values) of linguistic variables. Using the HA can quantify the semantic value of the linguistic variables in the rule. Then, the fuzzy laws can be seen as a point on the surface. Thus, the approximation methodology in control can be converted into an interpolation method on a real-surface determined by these points using fuzzy parameter values [12]. This transformation is defined by the Semantically Quantifying Mappings (SQMs) of the HA that can maintain relationships between variables based on the semantic order of linguistic values in the control rule system. The result can be considered as an appropriate mathematical model of control knowledge. Thus, the HA can provide a solid basis for developing new coherent and effective methods of reasoning for a type of controller, which is HAC (Fig.1).

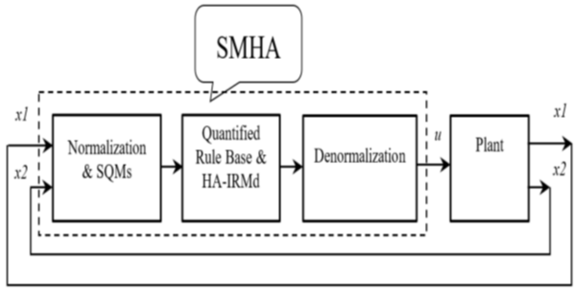

Figure 1. Diagram of control system using HA [12]

To design a HAC we must perform the following main steps [13], [14]:

- Step 1: Selecting the elements of the HA for input / output variables.
- Step 2: Building a mathematical model for the input-output relationship of the HAC
- Step 3: Calculating semantic quantitative values for language labels, build input-output relational surface (with defined fuzzy parameter set).

When the controller is working, the approximation problem is calculated using the interpolation method on the surface.

3.3 The HAC for REM algorithm improvement problem

HAC implements the REM algorithm (HACREM), which is also based on the price variable to measure congestion like the REM algorithm. We use two inputs, one that represents the price at the current time $P_r(kT)$ and one that represents the price at the previous cycle time $P_r(kT - T)$. Based on these two input values, the HA controller determines the value of packet rejection probability (DVP) that represents the output of the system.

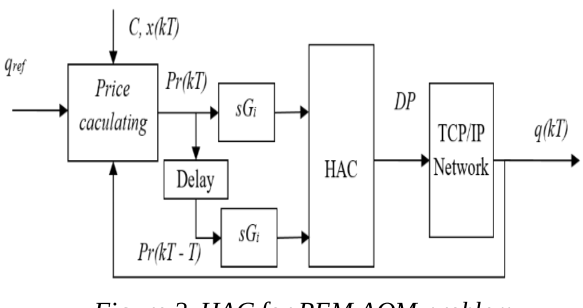

Figure 2. HAC for REM AQM problem

The inputs variables $P_r(kT)$, $P_r(kT - T)$ and the output *DVP(T)* are defined as follows: $P_r(kT - T) = \{VS, S, W, B, VB\}$, $P_r(kT) = \{VS, S, W, B, VB\}$, $DP = \{VS, S, LS, W, LB, B, VB\}$. When choosing $sG_i = \frac{1}{TQL + C}$ the crips domain of the inputs that will be in the range [-1; 1] this domain is mapped to the domain [0,1] of the HA.

The fuzzy parameters of the HA for input/output variables are pre-selected as shown in Table 1. The law of inference in the HAC is shown in Table 2.

Table 1. Fuzzy parameters of the HA for $P_r(kT)$, $P_r(kT-T)$, DVP

	$P_r(kT)$	$P_r(kT-T)$	*DP*
fm(S)	0.5	0.5	0.5
$\gamma = \mu(L)$	0.5	0.5	0.5

Table 2. Fuzzy rule system for REM AQM problem

$P_r(kT-T)$ \ $P_r(kT)$	***0***	*VS*	*S*	***W***	*B*	*VB*	***1***
0	***0***	***0***	*VS*	*VS*	*S*	*LS*	***W***
VS	***0***	*VS*	*VS*	*S*	*LS*	***W***	*LB*
S	*VS*	*VS*	*S*	*LS*	***W***	*LB*	*B*
W	*VS*	*S*	*LS*	***W***	*LB*	*B*	*VB*
B	*S*	*LS*	***W***	*LB*	*B*	*VB*	*VB*
VB	*LS*	***W***	*LB*	*B*	*VB*	*VB*	***1***
1	***W***	*LB*	*B*	*VB*	*VB*	***1***	***1***

Using fuzzy parameters in Table 1 to calculate semantic quantitative values of linguistic labels in Table 2 we obtain a semantic quantitative table of variables (Table 3). This table is also equivalent to the inference surface depicted in Figure 3.

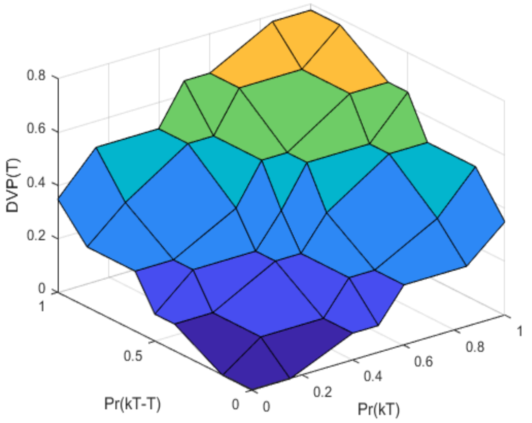

Figure 3. Inference surface of HAC

Table 3. Semantic quantitative values of variable

$P_r(kT)$ \ $P_r(kT-T)$	0.0000 *(0)*	0.1513 *(VS)*	0.3988 *(S)*	0.5000 *(W)*	0.6012 *(B)*	0.8488 *(VB)*	1.0000 *(1)*
0.0000 *(0)*	0.0000 *(0)*	0.0000 *(0)*	0.1012 *(VS)*	0.1012 *(VS)*	0.2250 *(S)*	0.3488 *(LS)*	0.5000 *(W)*
0.1513 *(VS)*	0.0000 *(0)*	0.1012 *(VS)*	0.1012 *(VS)*	0.2250 *(S)*	0.3488 *(LS)*	0.5000 *(W)*	0.6512 *(LB)*
0.3988 *(S)*	0.1012 *(VS)*	0.1012 *(VS)*	0.2250 *(S)*	0.3488 *(LS)*	0.5000 *(W)*	0.6512 *(LB)*	0.7750 *(B)*
0.5000 *(W)*	0.1012 *(VS)*	0.2250 *(S)*	0.3488 *(LS)*	0.5000 *(W)*	0.6512 *(LB)*	0.7750 *(B)*	0.8988 *(VB)*
0.6012 *(B)*	0.2250 *(S)*	0.3488 *(LS)*	0.5000 *(W)*	0.6512 *(LB)*	0.7750 *(B)*	0.8988 *(VB)*	0.9636 *(VB)*
0.8488 *(VB)*	0.3488 *(LS)*	0.5000 *(W)*	0.6512 *(LB)*	0.7750 *(B)*	0.8988 *(VB)*	0.9636 *(VB)*	1.0000 *(1)*
1.0000 *(1)*	0.5000 *(W)*	0.6512 *(LB)*	0.7750 *(B)*	0.8988 *(VB)*	0.9636 *(VB)*	1.0000 *(1)*	1.0000 *(1)*

IV. SEPARATION OF LOSS RECOVERY FROM CONGESTION CONTROL

The popularity of the Internet is, at least in part, due to the technology independent design of the IP network layer, which seamlessly connects diverse networks. Meeting the integration of wired and wireless TCP / IP networks is therefore an essential requirement of any new networking standard. In addition, a good strategy must be able to handle the coexistence of both ECN-enabled and ECN-enabled routers.

Packet loss due to buffer overflow can certainly happen on the Internet. In addition, routers do not have ECN capability, even routers that have AQM algorithms such as REM or RED perform packet reject when experiencing heavy congestion. For routers that do not support ECN, packet loss is considered congestion. So, for networks with such routers, the ambiguity between packet loss due to transmission failure and congestion loss can lead to congestion control failure.

To resolve this issue, we propose the following solution:

- Clearly inform the cause of the data packet loss (ECN)
- Conceal the loss of data packets due to transmission errors by creating a WHACREM Agent at the interface between wired and wireless networks.

4.1 Explicit Congestion Notification

ECN was initiated in 1999 from the idea of early detection of system bottlenecks and sending notification signals to the system before queue overflows. In today's TCP / IP networks, dropped packets are seen as a congestion signal. The majority of routers in a TCP / IP network are unprepared for embryo congestion detection. Therefore, when the queue overflows, the packets are dropped. Through acknowledgment packets sent to the source TCP detects packets discarded and detects current congestion in the network **Error! Reference source not found.**.

To limit packet rejection, ECN proposed a solution to develop the detection of system embryo congestion by calculating the average queue size and placing one ECN bit in the IP packet header. When the average queue size exceeds a certain threshold, the ECN detects and reports the congestion without having to rely on discarded packets.

4.2 Agent Building the WHACREM Agent

We propose to design an Agent as the interface between the wireless network and the wired network at the base station or wireless router. The essence of this Agent (WHACREM) is to install a HACREM controller in conjunction with the ECN. Before the data enters the wireless link, WHACREM Agent monitors both the passing IP packet and the re-provisioning ACK. When congestion loss is detected, an ECN bit is set immediately. This makes it possible to transport all congestion messages to the TCP source correctly. Whenever a packet is lost due to a transmission error, the router just invokes the retransmission mechanism without reducing its congestion window size. Since packet loss due to transmission failure cannot cause TCP to reduce window size, the WHACREM Agent can significantly improve network performance.

In practice, there may be cases where multiple packets are marked with the WHACREM Agent in a single window, so this algorithm performs at most one ECN marker per RTT time. This also provides stability compared to the possibility that a wireless ECN packet is dropped in both directions. The above solution gives WHACREM two main advantages:

- It effectively improves TCP performance in wireless network and is suitable for existing TCP mechanism.
- More importantly, it is compatible with routers that do not support ECN, making this algorithm highly acceptable in both wired and wireless networks.

V. SIMULATION RESULTS

Figure 4 illustrates the network architecture used to evaluate performance over wireless networks with and without WHACREM on the NS2.35 emulation program. Where packets are transmitted in two directions between the LAN terminals to the wireless terminals, the propagation delay is said to be negligible. We also assume that the ACK message from wireless terminals will immediately reach the LAN server. Since the ACK packet is relatively much smaller than the data packet (40-byte ACK vs. 560-1500-byte packet), this is a reasonable assumption.

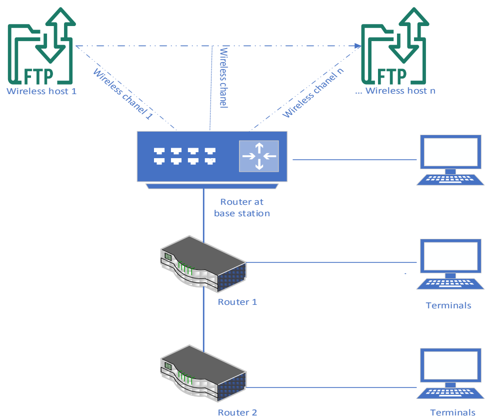

Figure 4. Architecture of the mixed network simulation

5.1 Efficiency

To test WHACREM's effectiveness, we set up simulated conditions with both a wired and wireless link that both has a bandwidth capacity of 2Mbps. The buffer capacity of the routers is 120 packets. Wireless link is shared by 100 NewReno TCP servers. They all implement the file transfer protocol (FTP), ie all are greedy. FTP sources can pass through a single link or multiple links, in which routers have REM or HACREM installed for comparison. The wireless channel model selected is a two-state Markov chain model. In order to change the dynamics of the network, we initially activated 20 sources and every 50 seconds thereafter, 20 other sources activated until all 100 were active.

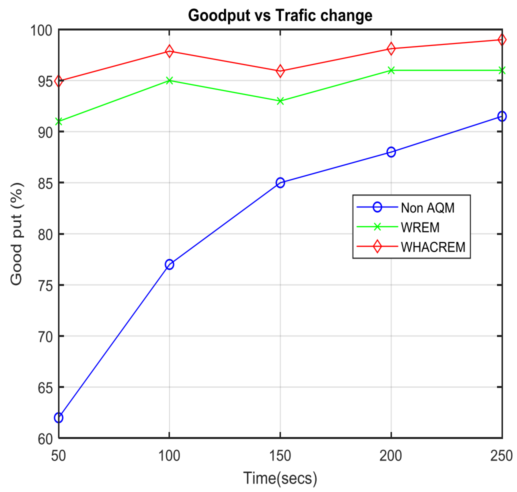

Figure 5. Goodput as the number of information sources increases

Figure 5 shows that both WREM and WHACREM methods are very effective in improving the goodput of Newreno (over 90%). Where WHACREM shows more adaptation to network dynamics. When the number of sources to the wireless network changes, WHACREM always maintains a goodput greater than 95%. Whereas, when the network is not supported by WHACREM, the goodput is very low, especially when the number of sources involved is small.

Cumulative packet loss due to buffer overflow is shown in Figure 6. It can be seen that WHACREM and WREM make negligible loss while Newreno without AQM will suffer increasing packet loss. This is consistent with the instantaneous queue length situation depicted in Figure 7.

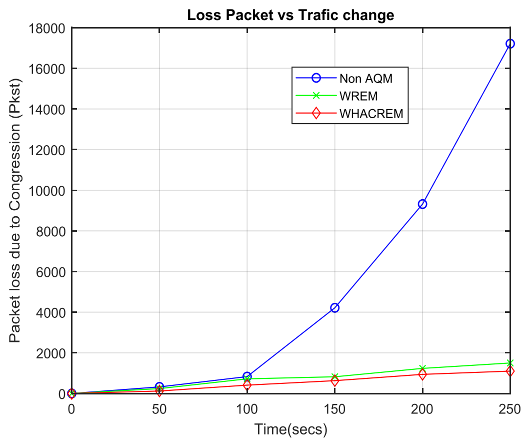

Figure 6. Cumulative packet loss as the source of information increases

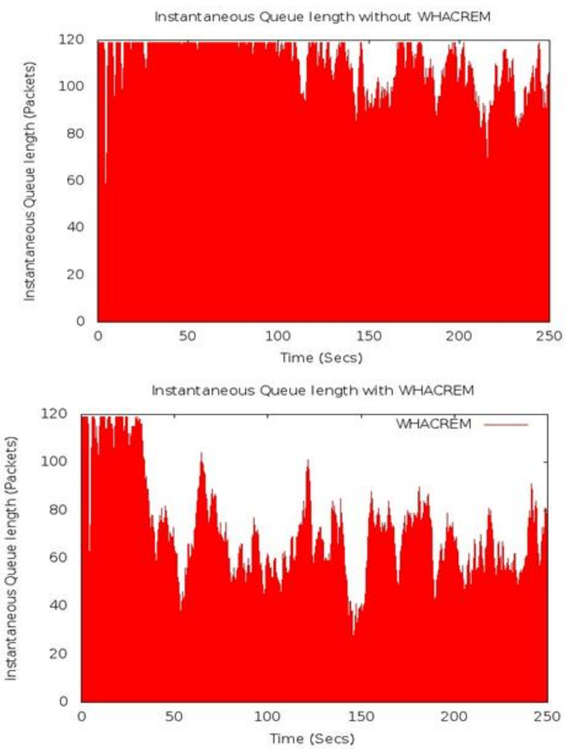

Figure 7. Instant queue without and using WHACREM

5.2 Compatibility

To test WHACREM compatibility, we need to perform this simulation with multiple links including routers with and without ECN support. For simplicity, we just put two routers in the path: One is running HACREM and the other doesn't support ECN. As above, we use a two-state Markov chain model to simulate the wireless link, using 100 Newreno sources with 20 sequentially activated sources every 50 seconds. We compare performance by results obtained on the same topology but both routers are ECN supported to confirm its compatibility.

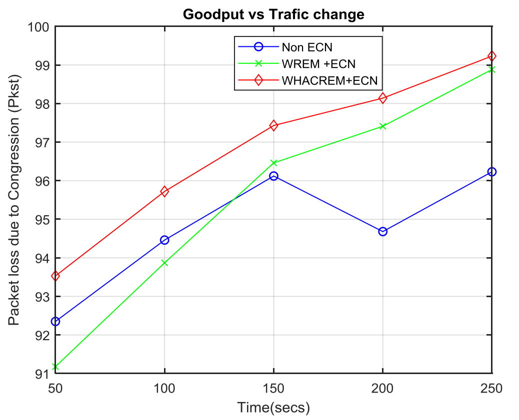

Figure 8 Goodput when a router with and without using ECN

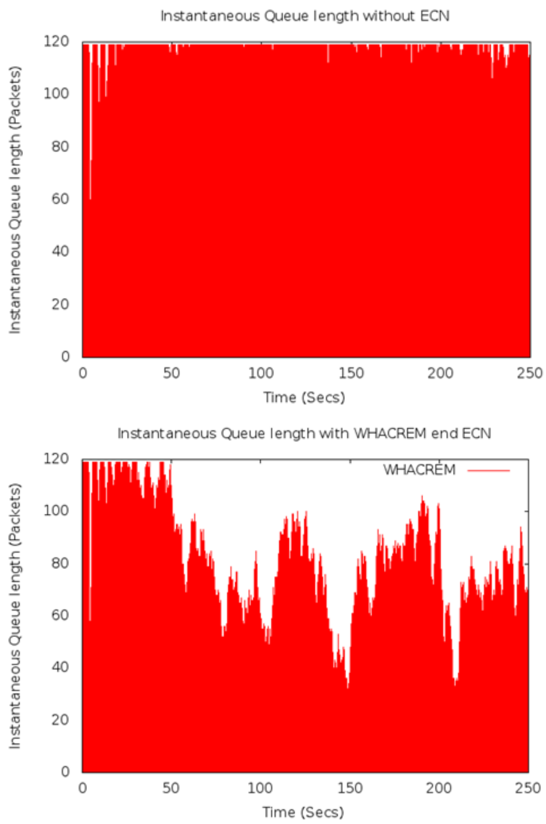

Figure 9. Instant queue when router with and without ECN

Figure 8 shows that when placing the WHACREM Agent at the interface between the two networks, the efficiency of the network usage is high whether or not the internal routers use ECN. However, ECN-enabled routers should be used for best performance. Observations of the queue length in Figure 9 show that HACREM can stabilize the average queue at a low level, thus leading to low queuing delay and low packet loss rates.

VI. CONCLUSION

The active queue management algorithms have been widely recognized as an effective way to improve wired TCP performance. However, with wireless TCP, these algorithms need to be improved to separate between congested packet loss and packet loss due to line loss. The WHACREM algorithm is proposed on the basis of improved REM performance by HAC in package marking and removal. At the same time, WHACREM integrates with the ECN and is installed at the interface between the wireless network and the rest. The simulation results show that WHACREM ensures efficiency by increasing googput and reducing packet loss accumulation. Also, WHACREM is compatible with the rest of the wired routers in the network whether or not they support ECN.

ACKNOWLEDGEMENTS

This research was supported by the ICTU research grant (under project # T2020-07-12). We thank our colleagues from the ICTU who provided insight and expertise that greatly assisted the research.

REFERENCES

[1]. Ma, L., Liu, X., Wang, H. et al. Congestion Tracking Control for Wireless TCP/AQM Network Based on Adaptive Integral Backstepping. Int. J. Control Autom. Syst. 18, 2289–2296 (2020). https://doi.org/10.1007/s12555-019-0724-y

[2]. Sharma, Anup & Behera, Ashok. (2016). A Survey on Active Queue Management Techniques. International Journal Of Engineering And Computer Science. 10.18535/ijecs/v5i11.47.

[3]. K. A. Yadav and S. Kumar, "A review of congestion control mechanisms for wireless networks," 2017 2nd International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2017, pp. 109-115, doi: 10.1109/CESYS.2017.8321245.

[4]. Athuraliya S., Lapsley D. E., Low S. H. (2001), “Random early marking for Internet congestion control”. IEEE/ACM Transactions on Networking, Vol. 15, No:3, pp. 48-53.

[5]. Huy Nguyen Phuong, Dung Le Ba, An application of genetic fuzzy system to improve the Random Exponential Marking algorithm in Active queue Management, 5th International Conference on BioMedical Engineering and Informatics (BMEI'12), pp. 1552-1556

[6]. Victor H. Li, Zhi-Qiang Liu, Steven H. Low, Using AQM to improve TCP performance over wireless networks, Proceedings Volume 4866, Quality of Service over Next-Generation Internet; (2002) https://doi.org/10.1117/12.473023

[7]. Wang, H.; Tang, J.; Hong, B. Research of Wireless Congestion Control Algorithm Based on EKF. Symmetry 2020, 12, 646. https://doi.org/10.3390/sym12040646

[8]. Casetti, Claudio & Gerla, Mario & Mascolo, Saverio & Sanadidi, M.Y. & Wang, Ren. (2002). TCP Westwood: End-to-End Congestion Control for Wired/Wireless Networks. Wireless Networks. 8. 467-479. 10.1023/A:1016590112381.

[9]. Haile, H K., Grinnemo, K-J., Ferlin, S., Hurtig, P., Brunström, A. (2021) End-to-end congestion control approaches for high throughput and low delay in 4G/5G cellular networks Computer Networks, 186: 1-22,https://doi.org/10.1016/j.comnet.2020.107692

[10]. Chen, Xiang & Zhai, Hongqiang & Wang, Jianfeng & Fang, Yuguang. (2005). A Survey on Improving TCP Performance over Wireless Networks. 10.1007/0-387-23808-5_23.

[11]. C.H. Nguyen, W. Wechler, Hedge algebras: An algebraic approach to structures of sets of linguistic domains of linguistic truth variable, Fuzzy Set. Syst., 35 (1990) 281-293.

[12]. C.H. Nguyen, N. L. Vu, X. V. Le, Optimal hedge-algebra-based controller: Desgn and application, Fuzzy Set. Syst., 159 (2008) 968-989.

[13]. D. Vukadinović, M. Bašić, C. H. Nguyen, T. Duy Nguyen, N. L. Vu and I. Grgić, "Hedge-Algebra Speed Controller – Experimental Implementation in a Sensorless Stand-Alone WECS," 2020 International Symposium on Industrial Electronics and Applications (INDEL), Banja Luka, Bosnia and Herzegovina, 2020, pp. 1-7, doi: 10.1109/INDEL50386.2020.9266203.

[14]. Thu May Duong , Hien Trinh Nguyen , Hop Le The, An Application of Hedge Algebras in Active Queue Management for TCP/IP Multiple Congestion Networks, Journal of Electronics and Communication Engineering (IOSR-JECE), pp. 45-53

[15]. S. Shahzad, E. Jung, J. Chung and R. Kettimuthu, "Enhanced Explicit Congestion Notification (EECN) in TCP with P4 Programming," 2020 International Conference on Green and Human Information Technology (ICGHIT), Hanoi, Vietnam, 2020, pp. 35-40, doi: 10.1109/ICGHIT49656.2020.00015.

Thu May Duong, et. al. "An improvement of REM Active Queue Management algorithm using the Hedge Algebra Controller applied in a wireless TCP IP network." *International Journal of Engineering Research and Applications (IJERA)*, vol.11 (4), 2021, pp 24-31.