

TCP Performance in Multi-Polling Game Theory-Based IEEE 802.11 Networks

Tomas Cuzanauskas (*Doctoral Student, Vilnius Gediminas Technical University*),
Aurimas Anskaitis (*Assistant Professor, Vilnius Gediminas Technical University*)

Abstract – Easy usage and integration with various applications made IEEE 802.11 one of the most used technologies these days, both at home and business premises. Over the years, there have been many additional improvements to the 802.11 standards. Nevertheless, the algorithms and Media Access Control (MAC) layer methods are almost the same as in previous Wi-Fi versions. In this paper, a set of methods to improve the total system capacity is proposed – such as efficient transmit power management based on Game Theory with a custom wireless medium protocol. The transmit power management and wireless medium protocol is verified by both simulation and real application scenarios. The results conclude that the capacity of the proposed wireless medium protocol is overall 20 percent higher than the standard 802.11 wireless medium access protocols. Additional TCP Acknowledgment filtering, which was tested together with the proposed wireless medium access protocol, can provide up to 10-percent-higher TCP throughput in high-density scenarios, especially for asymmetrical traffic cases. The conducted research suggests that efficient power management could result in lighter transmit power allocation rules that are currently set by the local regulators for current Wi-Fi devices. Thus, better propagation characteristics and wireless medium management would lead to an overall higher wireless system capacity.

Keywords – Cognitive radio; Communications technology; IP networks; Radio spectrum management; Wireless networks.

I. INTRODUCTION

As the demand of users increases both for the throughput and the growing base of various applications [1], it was noticed that the current IEEE 802.11 standard, which is used as a base on many systems, could no longer cope with required efficiency.

The new IEEE 802.11ac standard with higher modulation capabilities (256QAM), Multi-user MiMo as well as improved Carrier Sense Multiple Access Collision Avoidance (CSMA/CA) capabilities, gave a new boost to Wi-Fi networks. However, the recent results [2] suggest that the efficiency decreased compared to IEEE 802.11n standard.

Basically, IEEE 802.11 uses two wireless medium approaches: Point Coordination Function (PCF) and Distributed Coordination Function (DCF). The first one is based on the main entity, which controls all the connected nodes, while the second approach relies on fairness mechanism, allowing the wireless nodes to compete for wireless medium. An updated IEEE 802.11e version introduced a capability to have different queues for voice, video, best effort and background data types, thus providing the Quality of Service (QoS) capabilities.

In past, there have been many attempts to improve the quality of service for real-time applications. A few of the solutions suggested a combination of fair scheduled algorithms [3]–[5], while others proposed proprietary-based Time Division Multiple Access (TDMA) algorithms. The TDMA-based protocol is focused towards scaling capabilities as well as stability and low jitter, which is needed for real-time applications; however, it has drawbacks in traffic flow management.

The aim of this paper is to present the results of multi-polling based solution, which has a good scaling factor and stability in dense wireless networks. The results are provided based on the real outdoor network installation of off-the-shelf IEEE 802.11n devices. During the measurements, we ran multiple tests to verify the system capacity when different numbers of users utilized the wireless medium. The obtained results showed that the multi-polling solution is able to provide far better results compared to standard Round-Robin-based PCF protocol. The multi-polling solution in core is similar to Hybrid Coordination Function (HCF); however, differently than in HCF, we are considering all types of traffic, not just those of high priority. Besides the achieved results, we performed additional improvement for TCP throughput. The research, which we conducted, was based on TCP Acknowledgement (ACKs) frame filtering. The results of such filtering showed that TCP ACKs filtering plays a major role in increasing the overall performance in multi-polling-based solution, especially in those cases when traffic is asymmetrical.

The rest of the paper is organized as follows. In Section II, we overview the recent advances both in wireless networks and in TCP performance improvement over wireless networks. In Section III, we introduce Game Theory as one of the handlers for power allocation in the IEEE 802.11-based network. We will overlook the possibilities of the Game Theory and appliance in multi-polling solution. In Section IV, we describe the implementation and results of multi-polling on the IEEE 802.11-based devices. In Section V and VI, we will show a case study of possibilities to improve TCP performance in multi-polling based solution. We make final conclusions in Section VII.

II. RELATED WORK

In this section, we will review the most recent advances in IEEE 802.11 as well as possible improvements for TCP over asymmetric wireless links.

Over the years, there have been many attempts to improve IEEE 802.11 operation. In the following articles [6], [7], the authors are re-using one of the few popular polling schemes. The basic idea behind this scheme is the possibility of collecting the information from the stations and re-use that information for selecting the most active clients. From this statistically collected information, a scheduler compiles a polling list of connected stations. The algorithm uses the status collection period (SCP), where a small frame is broadcasted for multiple connected stations. In status collection period, stations mark the buffered amount of data and activity. After status collection period is finished, each active station receives calculated Transmission Opportunity (TxOP) allocation based on the currently used data rate and their frame size. In the next article [8], the authors have proposed a different approach. Similarly to the first methodology, here AP sends RAL frame, which includes corresponding TXOP, Traffic ID (TID) as well as transmission sequence index value (SIV). When a SIV value decreases, a different station has an opportunity to transmit its data.

In order to increase the TCP performance over the error prone wireless networks, we analyzed the existing solutions over asymmetrical radio links. However, there have not been that many attempts to implement this over commercial devices. Some of the articles study how the TCP reacts over the asymmetric networks. Few of them [9], [10] mark that fundamental cause of TCP degradation in asymmetric wireless networks is due to uneven access to shared wireless medium. As different stations gain different time to access the wireless medium, an asymmetry is created between downlink and uplink. The consequences of this are lower TCP throughput. As a solution, the authors of these articles provide a way to handle an uneven uplink and downlink ratio with managing the bandwidth usage of uplink used by ACKs and other type of traffic. The TCP ACKs filtering, TCP header compression, ACKs reconstruction, and TCP sender adaption methods are provided as possible key elements to increase the TCP performance [9].

In the next article [11], the authors propose a way to handle TCP sessions by providing an equal access to wireless medium for all created TCP sessions. The solution in the article is quite simple. They manipulate the received ACKs from the device in order to decrease or increase the TCP window size. By doing this they can control the TCP sender window size and limit it to whatever value is needed. However, this approach might require a considerable amount of CPU power on the wireless device, which is not always available; thus, it is not an efficient way, especially when the number of connected wireless nodes can be high.

In the next article [12], the authors consider custom-built snooping protocols, which are tracking the TCP packets over the base station. The idea behind snooping protocol is to track the packet loss faster than the TCP sender would react to the packet loss. The lost packet is retransmitted from snoop agent cache. The last article [13] proposes additional changes to TCP by re-using both delay-based and loss-based congestions avoidance approaches. However, this requires TCP

modification, which is hard to implement on every system and therefore not applicable for our case study approach.

III. GAME THEORY

Game Theory (GT) modeling considers a system of selfish players and their interactions between them. Depending on one player actions, the payoff for another player in the system might be different for every strategy, which was selected at the time. Any game model is classified based on the number of players, the sum of gains or the strategies and data types, which are used inside the game system. In wireless networks, we usually adapt non-cooperative game systems, since signaling between each player usually is not possible. Therefore, in our power game, we use non-cooperative game type. The non-cooperative game can be extended based on the data available for each player. In our case, we consider an extended form game [14].

Due to nature and dynamics of Radio environment, which is constantly changing in unlicensed band, it is hard to apply linear methods. Therefore, we apply a game theory model as additional tool to describe interactions between players in wireless medium. As we know, wireless medium is shared between multiple stations, where contention-based approaches are applied; in our system, we consider a non-cooperative game model as a base for marking these interactions. In game theory it is not always possible to have all the information – only some of information is available to stations or Access Points. The interference between different stations or Access Points results in a collision with an increased packet error rate (PER). Therefore, we seek a strategy for clients and AP which would result in the best outcome of the game. Strategy can be defined as the best response function. In the articles [14], [15], we considered a power allocation game as an additional improvement to MAC layer, which in some scenarios can be a benefit.

A system consisting of N players (Tx-Rx pairs of connected stations) with their predefined strategies $P = \{p_1, p_2, \dots, p_N\}$, the utility function of each player can be described as [9]:

$$u_i = \log \left(1 + \frac{h_{ii} p_i}{n_0 + \sum_{j \neq i} h_{ji} p_j} \right), \quad (1)$$

where

- p_i, p_j – the transmit powers of players i and j ;
- h_{ii} – the direct gain;
- h_{ji} – the channel gain between transmitter j and receiver i ;
- n_0 – the noise power.

The player's payoff function represents a difference between the utility function (1) and a cost function $c_i p_i$ [15], [16]:

$$\pi(p_i, p_{-i}) = \log \left(1 + \frac{h_{ii} p_i}{\sum_{j \neq i} h_{ji} p_j + n_0} \right) - c_i p_i, \quad (2)$$

where c_i – a trade-off between utility and power efficiency, which does not let the stations to raise its power indefinitely;

$$b_i(p_{-i}) = \frac{1}{c_i} - \frac{\sum_{j \neq i} h_{ji} p_j + n_0}{h_{ii}} = \frac{1}{c_i} - \frac{I + n_0}{h_{ii}}, \quad (3)$$

where

$b_i(p_{-i})$ – the best response of player (Tx-Rx pair) i given the current state of the game;

c_i represents transmit power cost;

h_{ij} – the channel gains;

p_j – the transmitted power for all other players;

n_0 – the noise.

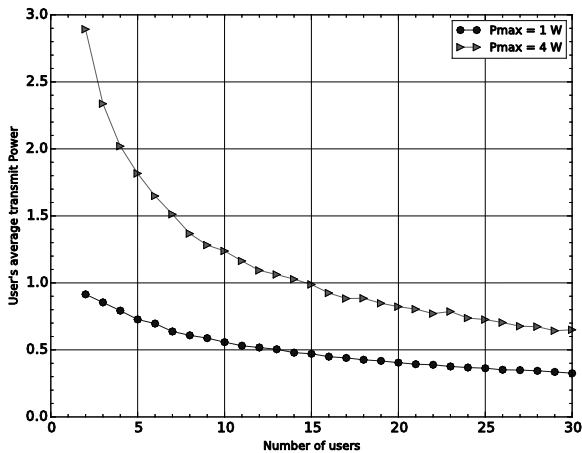


Fig. 1. User's average transmit power under different allowed power levels.

If the Nash equilibrium is reached, the game is stopped [15], [16]. The verification of algorithms was tested with a python-based simulation tool [17]. The result of python-based simulation tool is displayed in Fig. 1. The resulting graph allows us to see that users, which are part of the system, follow their strategy and do not raise the power indefinitely. The maximum limit of transmit power was set to 4 W, and the average power seen in Fig. 1 is around 3 W, which concludes our assumption that this is a self-controlled system and the power will not rise indefinitely.

We also tried to remove the transmit power restrictions. As can be seen in Fig. 2, the result was that the more power we allocate to users the more the overall system capacity increases. However, this is not always a right assumption, since we can consider that higher power might result in higher interference possibilities among the clients. We could consider this efficient only when applied outdoors with high antennas, which would decrease possible interference impact in coverage area. To conclude, we can consider a game theory-based power allocation approach as a valid tool for some modifications in the IEEE 802.11 MAC layer for power distribution between different users.

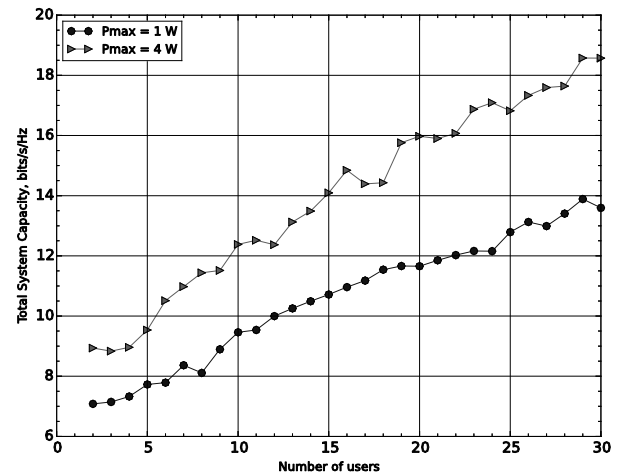


Fig. 2. Total system capacity with 30 users in the system with different allowed power levels.

IV. THE STUDY CASE OF MULTI-POLLING

In the first tests of multi-polling, we will start with standard polling tests such as maximum throughput test, latency test, and latency-under-load test. During measurements, we will compare the test results. The results should reveal if there is any improvement in real setup deployment. The multi-polling algorithm merges two DCF and Polling protocols into one – creating a Hybrid Version of both these protocols similar to the IEEE 802.11e [7] standard. Differently than the standard IEEE 802.11e, ours is not based on the quality stations (QSTA). On IEEE 802.11e, only stations with high priority packets are polled, while the others are left to use DCF for wireless medium access. AP contains two lists of connected stations. The stations which are more active will be put to polling list, and the stations which are less active or idling will use DCF approach to gain access to wireless medium. The stations which have high-priority packets receive capability to access wireless medium without waiting for polling tokens. The station activity is measured based on their sent packets per second (PPS) and bytes per second (BPS). The thresholds are controllable on the device and can be changed to have different PPS or BPS values. The implementation was carried out on 83 commercial wireless devices. The Access Point is Atheros AR9342 chipset-based device with an 18 dBm sectorial antenna. The stations are Atheros AR9342 with a 15 dBi directional antenna. The maximum output power of devices is set at 25 dBm and 15 dBm respectively. The devices are placed in outdoor environment at different distances: 61 devices are 200 m to 300 m and 22 devices are placed 5 km to 6 km range from the AP. This is done in order to create a more dynamic environment and a hidden node issue.

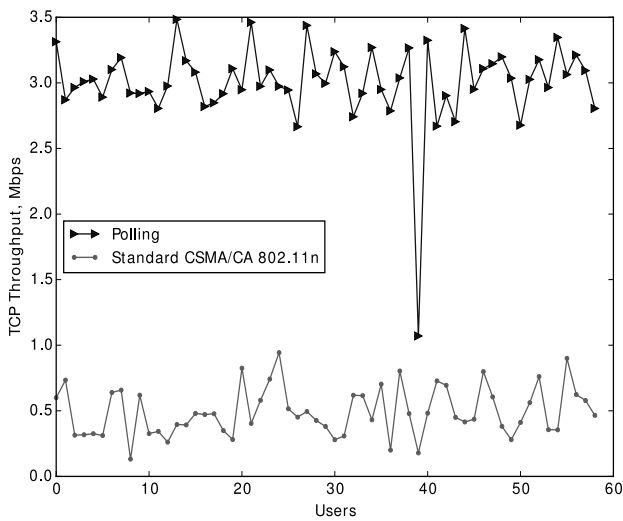


Fig. 3. Throughput under different protocols with 60 clients. Average RSSI – 5 dBm.

In Fig. 3, we can notice that polling even with standard Round-Robin (RR) functionality provides better performance compared to the standard IEEE 802.11 DCF protocol, which is usually used for the current off-the-shelf wireless devices. Such difference is due to additional self-interference between the stations. When multiple stations try to send their data at the same time, the back-off mechanism is activated for each of these stations. Any packets loss results in auto rate decrease on the device. At the end, total system capacity is lower compared to standard polling solution. However, when re-using the multi-polling solution, where only active clients are polled and inactive are left inside the inactivity scheduler, we achieve better results. The results are provided in Fig. 4. We can see that with multi-poll solution, we acquire higher performance figures when the amount of connected clients per AP increases. The measurement was done with outdoor-based devices placed at distance of approximately 200 to 300 meters.

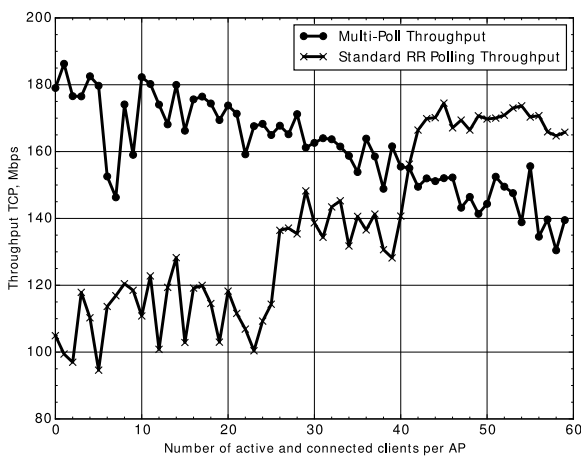


Fig. 4. Performance with different numbers of clients per AP.

In the next test, we will overview the real-time response under load conditions. This is done to evaluate the round-trip-

time (RTT) when there is a different amount of traffic generated over the connected stations and some of them are in idle state. For RTT measurement, we used Flent [18], which measures RTT under load conditions.

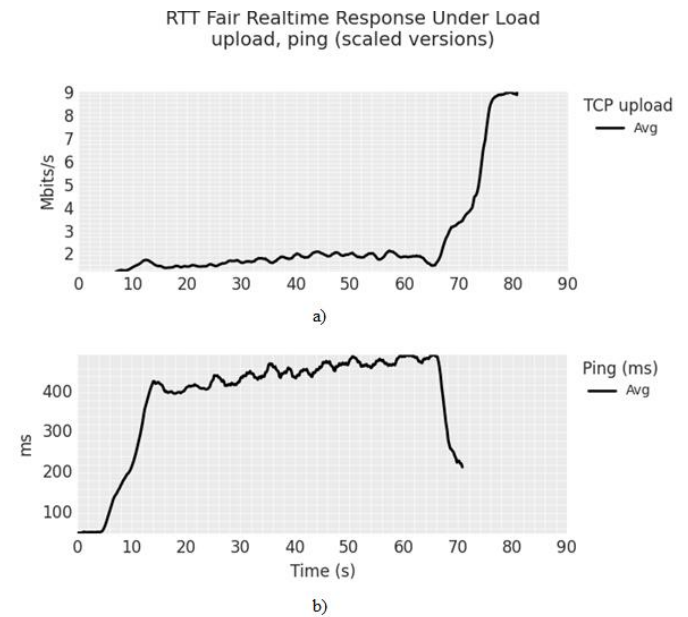


Fig. 5. a) User average throughput over time; b) User average latency.

The results depicted in Fig. 5 show that under high wireless network load conditions, there is a high latency impact. The graph a) demonstrates average throughput per individual station, and the graph b) demonstrates average latency over station. Since every station has predefined data slots (buffer size), the time that is required for transmission might vary from environment conditions. In the next test, we have checked CDF (Cumulative Distribution Function), which is applied for latency.

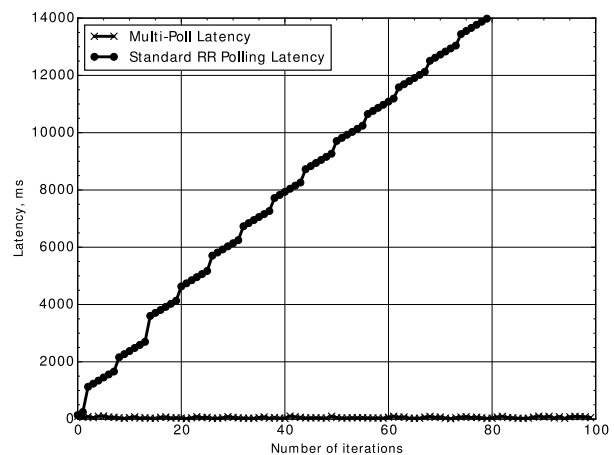


Fig. 6. RTT Fair Realtime Response under load.

Fig. 6 shows that under high load conditions, latency increased. The first wireless scheme is a standard-based Point Coordination Function, which equally polls all the connected

wireless devices in a row, called a Round-Robin approach. The second scheme is a multi-polling-based approach, where we are adding clients to activity list based on their throughput requirements. Since this allows building a smaller schedule list, we can guarantee a lower round-trip-time. The less active wireless nodes in this case are competing for wireless medium access like standard DCF-based devices in between the polled ones.

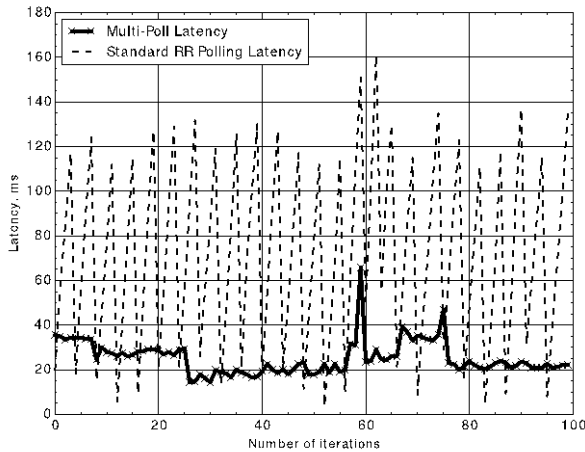


Fig. 7. User's latency under standard polling and multi-polling conditions. When total of 80 clients are connected to AP.

Fig. 7 shows the results when only one active client exists out of 80 connected. The active client is generating 90 Mbps throughput, while the one which was randomly selected for RTT measurements was idled. The multi-poll result provides more steady latency compared to the standard Round-Robin-based polling solution. The idling clients by default re-use the standard DCF approach, while the more active ones are moved to Round-Robin scheduler and the access is controlled by using token mechanism. Compared to standard polling solution, where all stations are polled, no matter how active the station is, we handle wireless medium activity more efficiently. The low PPS and BPS applications such as voice or Internet Control Message Protocol (ICMP) will have lower latency compared to standard polling. The smaller active scheduler list provides better latency to the active stations and results in higher TCP performance, which is latency and packet loss sensitive. The latency between new solution and standard polling can differ up to five times.

In Fig. 8, we tested a case when there is only one active station. During the test, there were a total of 83 stations. It is seen that multi-poll total throughput is 20 % higher compared to the standard polling solution. This is due to the fact that AP in multi-poll is polling only one station, which does not have data to send, instead of 83 stations. In standard polling mode, AP polls all the stations even if they do not have any data or have only small number of frames to send.

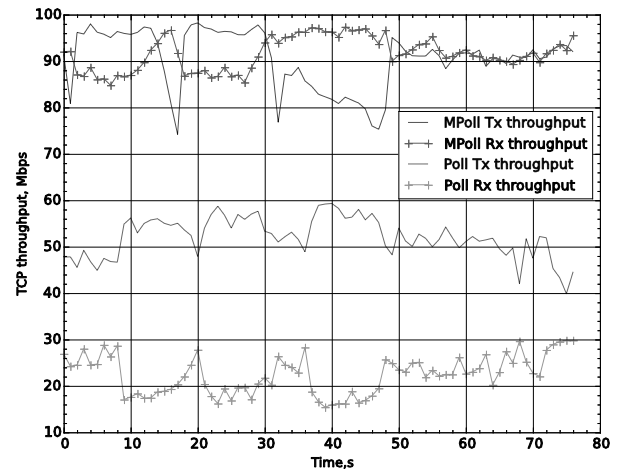


Fig. 8. Maximum throughput through one active station when the total of 83 stations are connected to AP.

V. TCP OPTIMIZATION IN MULTI-POLLING

TCP is used to provide reliable connection for applications like file transfer or important routing protocols. The reliability is created by using Acknowledgement system [19]. The main idea of TCP is to build a system which could recover data when it was lost or damaged due to various circumstances. In order to build such a system, an acknowledgement mechanism with timeout period is needed. If acknowledgement is not received over a defined timeout period, which is 200 ms according to RFC 813 [20], the data is retransmitted. However, if the ACK is delayed, the result might be that the host will receive duplicated data chunks. In this case, the duplicated data has to be discarded upon acknowledgement number [9], [19]. If ACK is lost completely, the transmitting host will repeat the sent data again. Nevertheless, sending multiple acknowledgment packets for every data chunk is an additional overhead for entire network and consumes unnecessary amount of traffic [21]. TCP can increment the sequence number of data chunks from the last segment, thus returned ACK sequence number is also increased, which informs the transmitter that all the data is received. Another important key role of TCP is the window size, which is highly dependent on the channel features such as latency or packet loss RFC 2581. TCP congestion window is related with packet loss rate as shown in [13]:

$$W = \frac{1.22 \cdot MSS}{\sqrt{p}}, \quad (4)$$

where

- W – an average TCP window;
- p – an average packet loss rate;
- MSS – the maximum segment size.

It is seen that TCP window size depends relatively on small packet loss. Therefore, we can see many new TCP algorithms which try to solve the window growth and congestion problems [9]–[13]. One of the key elements to get a stable window growth is the guarantee that ACKs packets will be delivered with the lowest possible latency and there will not be significant ACKs drops in the network. The most used TCP congestion algorithms are HTSC (RFC 3649), STCP (RFC 3286) [22], and Cubic, which are currently employed on Linux-based machines [23].

In wireless networks, it is not always possible to guarantee a steady flow of TCP ACKs, which are required for stable throughput. It is especially noticeable in asymmetric networks, which are common among IEEE 802.11 network users. If outdated ACKs and new ACKs are transmitted on a single wireless medium access, the outcome might be packet loss or throughput degradation. One of the ways of controlling the amount of TCP ACKs in wireless network is to use the TCP header compression [9], [10]. During header compression, ACK size is reduced; however, this still does not solve the unnecessary overhead on IEEE 802.11 networks. Therefore, we will rely on the TCP ACK filtering technique [9], [24], [25]. This method allows reducing the number of TCP ACK, which is sent uplink from the station to Access Point, thus removing unneeded overhead and allowing more efficient usage of available bandwidth. The only challenge in handling such a mechanism is to guarantee that the sender will receive enough TCP ACK and will not stall. Only certain ACKs are removed in order not to impact the throughput performance [26]. In this way, we may free up the buffer for other data. The error control of TCP ACK mechanism is handled by leaving only up-to-date ACKs which sequence number contains the information of outdated ACK. Since this technique was mainly tested in wired network, i.e. ADSL, we will implement a similar approach on wireless devices where downlink is more important than uplink performance.

Our TCP ACK handling mechanism is implemented in simple, yet robust way. By default, every station and AP has a queue where packets are buffered until the time comes to send the data. In order to handle TCP ACKs, additional buffers are created to separate TCP ACK from all the rest normal data. In this buffer, the device periodically checks if there exist outdated ACKs before transmitting them. If there are redundant ACKs which acknowledge the same TCP frame, they are simply removed. When the station receives the token from AP, the station is allowed to transmit its data. Firstly, TCP ACKs buffer is flushed, and after that the data buffer is handled. This leads to two important key components. By removing redundancy ACKs, we decrease the time required to transmit unnecessary frames, and by moving TCP ACKs firstly, we decrease both latency and jitter, which impacts TCP performance.

VI. TCP PERFORMANCE MEASUREMENTS

During testing, we will rely on standard CUBIC TCP implementation, where congestion window can be described as [27]:

$$W(t)_c = C \left(t - 3 \sqrt{\frac{W(t)_{\max} \cdot \beta}{C}} \right)^3 + W(t)_{\max}, \quad (5)$$

where

C – the scaling factor;

$W(t)_{\max}$ – the maximum window size before the last congestion;

β – the decrease factor applied to $W(t)$.

From the above formula it is seen that the initial phase when the TCP window is growing to its maximal value is critical and requires that ACKs would be stable without any loss; otherwise congestion will be triggered and throughput will decrease. When maximum value for $W(t)_{\max}$ is reached, growth becomes minimal and congestion is less important at this phase. Most of the current Linux versions are based on CUBIC TCP. CUBIC TCP uses an optimized congestion control algorithm. Another important keynote is the CUBIC capability not to rely on the ACKs to increase the window size, which is important to our testing, since we will try to remove as many redundant TCP ACKs as possible. In the last test, we will try to measure the impact of TCP ACKs on standard Microsoft Windows-based machines which use Compound TCP. It is a Microsoft implementation of TCP which incorporates a scalable delay-based algorithm into the standard TCP congestion algorithm. When the network is sensed under-utilized, the delay-based element rapidly increases the window size; when the network is sensed to be fully utilized, the delay-based algorithm becomes less aggressive on the TCP window size.

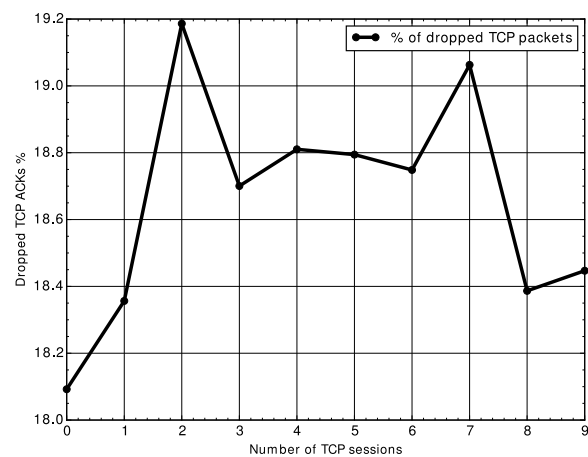


Fig. 9. Percentage of dropped TCP ACKs compared to the total amount of packets on Ethernet interface.

Again, for the measurements, we used the same setup as in section IV. Firstly, we measured how many TCP ACKs are dropped by our solution. We loaded one of the 80 connected devices with simplex 90 Mbps traffic, which was going from AP to the station over the wireless link. For traffic generation, we used iPerf. The 90 Mbps is limited by the Ethernet port speed, which is 100BaseT. We measured the total number of sent packets on Ethernet interface and compared the amount of dropped TCP ACKs. Fig. 9 illustrates that with the increasing

amount of TCP session we can get approximately 18 % of TCP ACKs drop rate compared to the total amount of sent packets over Ethernet interface. We dropped only redundant TCP ACKs, so the performance did not decrease.

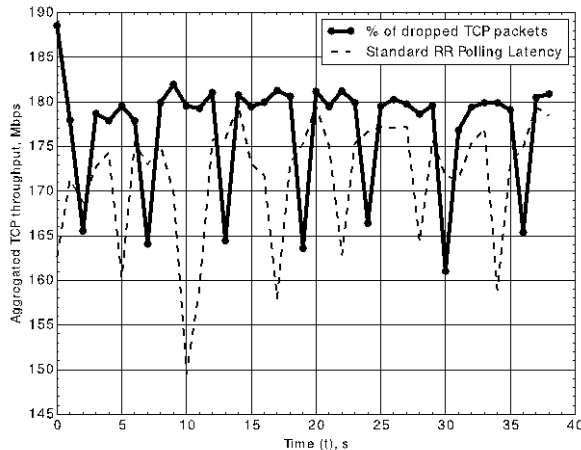


Fig. 10. Performance with different TCP ACKs handling solutions.

Similar approach [27] was used for TCP ACKs filtering in wired networks by using wired devices as well as wireless devices. However, our approach is less aggressive, since in later approach, TCP ACKs were removed completely when a certain TCP Window size was reached. We took less aggressive approach since wireless medium is less predictable.

We continued our measurements with comparing the standard polling without TCP ACKs filtering mechanism, when traffic was generated over 60 clients with iPerf. We generated TCP Duplex traffic. As it is seen, with TCP ACK-based multi-poll solution, higher performance figures are obtained, which is more stable compared to the solution without TCP ACK filtering mechanism. This is the result of generating less TCP ACKs for each of TCP session created by iPerf. Every connected station creates two TCP sessions. In total, 120 TCP sessions are created over the wireless link. From Fig. 10 it is seen that throughput increase is around 10 %.

In the last test (Fig. 11), we measured the throughput generated by single TCP session, when the direction of traffic was sent from AP to the station. It can be noticed that at the same conditions, the average throughput, which is 94 Mbps, generates the stream of 2 Mbps to 2.5 Mbps TCP ACKs from the station to AP. This is the result of redundant TCP ACKs. In our solution, when we remove those redundant TCP ACKs, the performance from AP to the station is left unchanged – the same 94 Mbps; however, the TCP ACKs stream was within the range of 0.3 Mbps to 0.5 Mbps from the station to AP compared to the previous 2.5 Mbps without TCP ACKs filtering. This provides a five times less generated throughput of TCP ACKs. The amount of traffic, which is generated by only TCP ACKs, will rise in IEEE 802.11ac while more traffic can be sent from AP to the station.

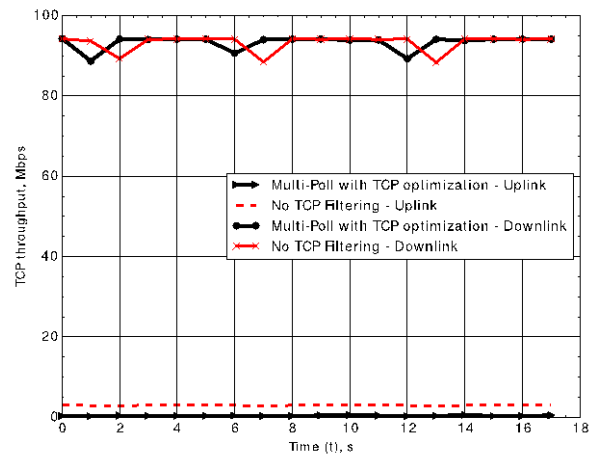


Fig. 11. Comparison of uplink and downlink when TCP filtering was turned off and on.

VII. CONCLUSION AND FUTURE WORK

In this article, we overview the possibilities of multi-poll solution together with the TCP ACKs filtering mechanism. When a high number of clients is connected to AP, which is a typical scenario in point-to-multipoint networks, the existing IEEE 802.11 mechanism cannot cope with handling wireless medium. However, the multi-poll-based solution can increase throughput by station since stations are moved between active and idle state. The throughput, which can be acquired by using multi-poll scheduler, can be up to 20 % higher compared to typical Round-Robin-based solution. The less active stations are still able to access wireless medium by using a standard DCF method; this allows getting lower latency, which might be required for less throughput-demanding applications like VoIP. The TCP ACKs filtering showed that it is possible to increase the throughput up to 10 % in different cases. As it is seen from the last measurements, typical TCP ACKs in simplex throughput measurement generates around 2 Mbps to 2.5 Mbps throughput of TCP ACKs. However, when applying TCP ACK filtering, the stream up to 0.5 Mbps is achieved. By reducing the number of required ACKs, the station takes less time for transmission and more time is handed out to AP. Since most of the traffic in these days is usually downlink, we consider this as a good improvement in asymmetrical IEEE 802.11 wireless communication. In future, we plan to perform similar tests on the IEEE 802.11ac platform, where TCP ACKs can take even bigger part, since simplex throughput is six times higher.

REFERENCES

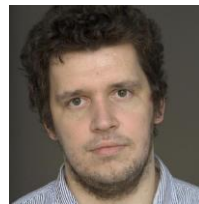
- [1] O. Tetervenoks, I. Galkin and J. Armas, "Autonomous Power Supply System for Light Sensor of Illumination Measurement Test Bench," *Electrical, Control and Communication Eng.*, vol. 1, issue 1, pp. 30–35, Dec. 2012. <https://doi.org/10.2478/v10314-012-0005-0>
- [2] Aruba Networks. *802.11ac In-Depth*, 2015. [Online]. Available: http://www.arubanetworks.com/pdf/technology/whitepapers/WP_80211acInDepth.pdf

- [3] Z. Chen and A. Khokhar, "Improved MAC Protocols for DCF and PCF Modes over Fading Channels in Wireless LANs," in *2003 IEEE Wireless Commun. and Networking. WCNC 2003*, New Orleans, LA, USA, 2003, pp. 1297–1302. <https://doi.org/10.1109/WCNC.2003.1200560>
- [4] S. Cheng, M. Wu. "Contention-Polling Duality Coordination Function for IEEE 802.11 WLAN Family," *IEEE Trans. on Commun.*, vol. 57, no. 3, pp. 779–888, Mar. 2009. <https://doi.org/10.1109/TCOMM.2009.03.070073>
- [5] Y. P. Fallah and H. Alnuweiri, "Hybrid polling and contention access scheduling in IEEE 802.11e WLANs," *J. of Parallel and Distributed Computing*, vol. 67, issue 2, pp. 242–256, Feb. 2007. <https://doi.org/10.1016/j.jpdc.2006.07.003>
- [6] B.-S. Kim, S. W. Kim, Y. Fang and T. F. Wong, "Two-Step Multipolling MAC Protocol for Wireless LANs," *IEEE J. on Selected Areas in Commun.*, vol. 23, no. 6, June 2005. <https://doi.org/10.1109/jsac.2005.845635>
- [7] J. Chen, C.-A. Lin, "HMM: Hybrid Multipolling Mechanism with Pre-allocation Admission Control for Real-Time Transmissions in WLANs," in *IEEE 60th Vehicular Technology Conf. (VTC2004-Fall)*, 2004, pp. 3040–3044. <https://doi.org/10.1109/vetecf.2004.1400619>
- [8] Y. Fang, D. Gu, A. B. McDonald and J. Zhang, "On the Performance Enhancement of Wireless LAN – A Multi-polling Mechanism with Hidden Terminal Solution," in *2005 Global Telecommun. Conf. (GLOBECOM '05)*, 2005. <https://doi.org/10.1109/glocom.2005.1577689>
- [9] H. Balakrishnan and V. N. Padmanabhan, "How Network Asymmetry Affects TCP," *IEEE Commun. Magazine*, vol. 39, issue 4, pp. 60–67, Apr. 2001. <https://doi.org/10.1109/35.917505>
- [10] A. Statkus and S. Paulikas, "Improving TCP Performance in IEEE 802.11 Networks," *Research J. Elektronika ir Elektrotechnika*, vol. 19, no. 5, 2013. <https://doi.org/10.5755/j01.eee.19.5.2571>
- [11] S. Pilosof, R. Ramjee, D. Raz, Y. Shavitt and P. Sinha, "Understanding TCP fairness over Wireless LAN," in *Twenty-second Annu. Joint Conf. of the IEEE Computer and Commun. Societies, IEEE INFOCOM 2003*. San Francisco, CA, 2003, vol. 2, pp. 863–872. <https://doi.org/10.1109/infcom.2003.1208924>
- [12] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, M. Stemm, E. Amir and R. H. Katz, "TCP Improvements for Heterogeneous Networks: The Daedalus Approach," in *Proc. of the 35th Annu. Allerton Conf. on Communication, Control, AND Computing*, 1997.
- [13] K. Tan, J. Song, Q. Zhang and M. Sridharan, "A Compound TCP Approach for High-speed and Long Distance Networks," in *Proc. IEEE INFOCOM 2006. 25TH IEEE Int. Conf. on Computer Commun.*, Barcelona, Spain, 2006, pp. 1–12. <https://doi.org/10.1109/INFOCOM.2006.188>
- [14] M. Felegyhazi and J. P. Hubaux, "Game Theory in Wireless Networks: A Tutorial," EPFL Technical report: LCA-REPORT-2006-002, Switzerland, Jan. 2006.
- [15] A. Medeisis, J. Sydor, L. Cremene, O. Holland, A. Anskaitis, D. Wiecek, Y. Haddad and T. Cuzanauskas, "ISM-Advanced: Improved Access Rules for Unlicensed Spectrum," in *2014 IEEE Int. Symp. on Dynamic Spectrum Access Networks (DYSPAN)*, McLean, VA, 2014, pp. 194–205. <https://doi.org/10.1109/DySPAN.2014.6817796>
- [16] G. Fang et al., "Distributed Inter-Network Interference Coordination for WBANs," in *2010 IEEE Global Telecommun. Conf. GLOBECOM 2010*, Miami, FL, 2010, pp. 1–5. <https://doi.org/10.1109/glocom.2010.5683900>
- [17] *Python language programme used for simulation of power control game*. [Online]. Available: <http://www.cost-terra.org/our-publications>
- [18] T. Høiland-Jørgensen (2016, Dec. 3). "Flent: The FLExible Network Tester," [Online]. Available: <https://flent.org/flent-the-flexible-network-tester.pdf>
- [19] M. Allman, V. Paxson and W. Stevens, *TCP Congestion Control*. RFC2581, 1999. [Online]. Available: <https://tools.ietf.org/html/rfc2581>
- [20] D. D. Clark, *Window and Acknowledgment strategy in TCP*. RFC813, 1982. [Online]. Available: <https://tools.ietf.org/html/rfc813>
- [21] J. D. Brouer and H. S. Hansen, "Experiences with Reducing TCP Performance Problems on ADSL," DIKU – Technical Report 04/07, 2004.
- [22] L. Ong and J. Yoakum, *An Introduction to the Stream Control Transmission Protocol (SCTP)*, RFC3286, 2002. [Online]. Available: <https://www.ietf.org/rfc/rfc3286.txt>
- [23] I. Rhee, L. Xu and S. Ha, *CUBIC for Fast Long-Distance Networks*, 2007. [Online]. Available: <https://tools.ietf.org/html/draft-rhee-tcp-cubic-00>
- [24] C. Barakat and E. Altman. *On ACK Filtering on a Slow Reverse Channel*. INRIA, 2004. [Online]. Available: <http://www-sop.inria.fr/members/Chadi.Barakat/Qofis2000-2.pdf>
- [25] L. Salameh, A. Zhushi, M. Handley, K. Jamieson and B. Karp, "HACK: Hierarchical ACKs for Efficient Wireless Medium Utilization," in *2014 USENIX Annu. Technical Conf.*, Philadelphia, PA, June 19–20, 2014, pp. 359–370.
- [26] H. Balakrishnan, V. N. Padmanabhan and M. Sooriyabandara, *TCP Performance Implications of Network Path Asymmetry*. RFC3449, 2002. [Online]. Available: <https://www.rfc-editor.org/rfc/rfc3449.txt>
- [27] L. Pavilanskas and A. Statkus, "Evaluation of TCP Acknowledgment Mechanism Influence on Router Performance," *Research J. Elektronika ir Elektrotechnika*, vol. 103, no. 7, pp. 95–100, 2010.



Tomas Cuzanauskas received the degrees of B.Sc. and M.Sc. in telecommunication engineering from Vilnius Gediminas Technical University in 2011 and 2013, respectively. He is currently a Ph.D. student in telecommunication engineering at Vilnius Gediminas Technical University. Since 2014, he is an Engineer in one of the IEEE 802.11 wireless manufactures companies. His main research interests are wireless communications, smart networks and resource management in the IEEE 802.11 wireless networks. His current research includes wireless networks optimization in dense wireless networks.

T. Cuzanauskas is a student member of the IEEE.
Address: Naugarduko str. 41, Vilnius, Lithuania.
E-mail: tomas.cuzanauskas@vgtu.lt



Aurimas Anskaitis received the degrees of B.Sc., M.Sc. and Ph.D. in electronic engineering from Vilnius Gediminas Technical University in 2001, 2005, and 2009, respectively. He is an Associate Professor at the Department of Telecommunication Engineering at Vilnius Gediminas Technical University. A. Anskaitis has participated in numerous research projects such as COST Action IC0905, COST 290 and similar, where various smart wireless network topics were researched. His main area of research interest is wireless network optimization and the problems related to the quality of service. A. Anskaitis is a co-author of the Telecommunication Theory Workbook.

Address: Naugarduko str. 41, Vilnius, Lithuania.
E-mail: aurimas.anskaitis@vgtu.lt