Easing Network Traffic in the 5G Networks

Ivan Petrov[1] and Toni Janevski[2]
[1]Telecom, Sales Excellence Unit
Kej 13 Noemvri, No.6. 1000, Skopje, Macedonia
ivan.petrov@telekom.mk

[2]Faculty of Electrical Engineering and Information Technologies
Rugjer Boshkovik bb
1000 Skopje, Macedonia
tonij@feit.ukim.edu.mk

ABSTRACT: *In the 5 G networks, effective and dynamic data use is important for its utilization for which we have designed a transport protocol. In the environment where the low speed exists, the high rate can up to 400 Gbps with the help of the heavy mobile situation. This paper has introduced a powerful algorithm which can help to evade congestion issues in the back network traffic. Thus, the proposed protocol can perform better than the existing common high speed transport protocols.*

Keywords: 5G Mobile Networks, Congestion Control, TCP, Throughput

Received: 28 December 2021, Revised 1 March 2021, Accepted 17 March 2022

DOI: 10.6025/jnt/2022/13/2/29-36

1. Introduction

From the very beginning until today TCP is evolving and the document roadmap describing its development is frequently updated [1]. One of the key triggers in the development process was its poor performance in high speed networks.

Congestion avoidance phase was studied and identified as generator of this unpleasant behavior. Numbers of TCP versions were created with primary aim to improve the protocol utilization in high speed environment and most of them were addressing and redesigning the congestion control mechanism. It is important to notice that during the design phase variety of constraints must be carefully addressed especially when creating protocol for usage in future 5G mobile networks [2-5].

Today we distinguish three main groups of high speed protocols: loss based [6-12], delay based [13-16] and loss based with bandwidth estimation [17-22], each of them improves the protocol performances.

High speed TCP proposals use direct and indirect approach of more aggressive network probing with aim to create congestion control mechanism that will be aggressive enough in case of underutilized network and will remain gentle in case of utilized resources.

There are two main reasons for poor performance of standard TCP protocols in high speed networks:

- Cwnd linear increase by one packet per round trip time in congestion avoidance phase is too slow and multiplicative decrease in case of loss is too drastic.

- Maintaining large average congestion windows at flow level require extremely small equilibrium loss probability and a stable flow dynamic design.

In this paper we present novel high speed transport protocol (TCP Ohrid) capable to achieve speeds up to 400Gbps in future 5G networks.

In section 2 we discuss future 5G network design scenarios and we express the need for development of TCP Ohrid. TCP Ohrid protocol is defined in section 3. In Section 4 we present the simulation environment and discuss the results. Section 5 concludes the paper.

2. Future Networks

Next Generation Networks (NGN) and Future Networks [2] are all-IP networks, meaning that all data, control and signaling will be carried through IP-based communication and based on Internet technologies form network protocol layer up to the application layer, with heterogeneous access technologies (fixed and mobile) on the lower layers of the protocol stack. TCP Ohrid meets 5G characteristics which are set prior to the standardization, which is expected to be finalized around 2020.

Most researchers are focused on development of 5G network that will provide improved future communication. Fundamental design, requests and drivers are defined and argued in [3]. 5G Network design scenarios and evaluations are conducted in [4].

Ideas for mobile cloud computing in order to offload most energy consuming tasks to nearby fixed servers and to develop mobile cloud computing are reality and are subject of future development [5].

As we know, small cells and their ultra dense deployment could become main feature of 5G networks indicating that the number of small cells will increase in unit area so the corresponding backhaul gateway traffic will increase exponentially if the conventional centralized control model is adopted as 5G backhaul network architecture. If this network design is chosen during the selection process we have to notice that massive backhaul traffic will create congestion which might result in a collapse of the backhaul network. Therefore, there is a need for development of ultra fast protocol similar to TCP Ohrid able to assure data transfer rates up to 400 Gbps or lower rates in case of heavy mobility of the terminals, making the protocol more efficient than the traditional one. This protocol is able to reduce and dynamically manage the backhaul congestion.

Hybrid backhaul networks using wireless and fiber access technologies can be used to increase the energy efficiency of the backhaul network.

We identify several requests that should be satisfied by future networks. 5G networks will have to employ energy harvesting mechanism at core base station network. For example,”harvest-then-transmit” mechanism or similar can be used in order to save energy [23].

Optimization at terminal application level with offload of most energy consuming tasks to nearby fixed servers presented as mobile cloud computing is additional value that can be added and employed at application layer in order to save terminal battery. Both centralized and decentralized approaches can be used in order to provide this network function.

5G Networks must represent an all IP mobile platform that will provide infrastructure as a service, platform as a service and software as a service. (Unified access of the platform will be provided regardless the medium access control, e.g., WiFi, 2G, 3G, LTE, Copper, Fiber, RF, etc.). The network must offer higher data transfer rates in static and mobile (nomadic) mode of operation.

3. TCP Ohrid Protocol

TCP Ohrid protocol development is inspired by HSTCP protocol. We have noticed that HSTCP response function can be modified by preserving the main request to remain convex. It is possible to define different increase and decrease parameter functions by usage of different interpolation mechanisms. Equation (1) given below describes all loss based TCP protocols and in combination with (2) it defines HSTCP.

$$\begin{aligned} &w \leftarrow w + \frac{\alpha}{w} && CA\ \ phase\ ,\ when\ \ ACK\ \ received \\ &w \leftarrow w - \beta w && in\ case\ \ of\ \ drop \\ &w \leftarrow w + \gamma && Slow\ \ Start \end{aligned} \tag{1}$$

$$\alpha\left(w\right) = \frac{w^{2}\,2\,\beta\left(w\right)p\left(w\right)}{2 - \beta\left(w\right)} \tag{2}$$

α is window increase parameter with default value of 1 Maximal Segment Size, β is decrease parameter with default value of 0.5 MSS, w denotes congestion window size, p is packet loss probability and γ is slow start parameter with default value of 1 MSS. When default values of α, β and γ are used equation (1) describes the standard TCP (Reno, Newreno) protocol versions.

TCP Ohrid protocol is defined with several response functions where linear interpolation is used for six known points. Usage of several response functions for different speed improves protocol friendliness at lower rates. The algorithm combines response functions in order to provide improved protocol friendliness at low and high rates. We know that fiber optic has error rate of 1 error in 10^{10} bits meaning that the probability of bit error in fiber is 10^{-10} . In our scenario we use packets with size of 1500 bytes that limits packet loss probability in fiber networks at $P = 12000 \times 10^{-10} = 1.2 \times 10^{-6}$. TCP Ohrid protocol preserves usage of switch point defined with $w = 38$ packets, $p = 0{,}0015$ and parameter set at 0.5 in order to represent fair protocol that will be friendly with competing TCP. If we want to create a protocol that will achieve Gbps throughput we have to modify the basic definition (1) in order to get faster window increase. However, this implies that we cannot use existing congestion mechanism. We decided to use modified parameter equation in predefined range of known constraint values. TCP Ohrid protocol is defined with several response functions in dependence of the sending rate. The main idea is to use one primary switch point and five end/switch point's instead one switch and one endpoint at loglog scale. It is defined with points that correspond at speed of 5Gbps, 10Gbps, 40Gbps, 100Gbps and 400Gbps throughput that is supported by the transport medium. Knowing that packet loss probability of fiber networks is in rang of 10^{-7} (when 1500 byte packets are used) we decided to observe real packet loss probability values for optical networks and future packet loss probability of 5G packet transmission networks. TCP Ohrid protocol is 5G protocol that can provide data rates higher than 5Gbps under heavy user mobility and 400 Gbps backhaul data rates in the core network. Today there are patented solutions for providing ultra high speed transmission with capacity of 1015 bits/s. In 2005, 100Gbps Ethernet was tested in laboratory and in 2010 commercial active and passive equipment was available. IEEE Taskforce has completed standardization of local area network interfaces for 100Gbps ethernet. We can recall that IEEE P802.3bj supports BER better or equal to 10^{-12} and 100 Gbps (over copper), IEEE P802.3bs supports BER better or equal to 10-13 and MAC data rate of 400Gbps. IEEE 802.3ba supports BER better than or equal to 10^{-12} and MAC data rates of 40Gbps and 100 Gbps. Considering all above stated we can decide which packet loss probability values and which data rates in the switch points can be used in future 5G network. Note that the suggested bit error probabilities are real if we use 1500 byte packets. TCP Ohrid is defined with maximal speed of 400Gbps, $p=10^{-12}$ and =0.1 or 0.4 in dependence of the used decrease parameter variation, the primary switch point preserve its parameter values. Interpolation can be conducted with help of following defined points 5Gbps, p = 10-7; 10Gbps, $p = 10^{-8}$; 40Gbps, $p = 10^{-9}$ and 100Gbps, $p = 10^{-10}$. It is expected in the future networks BER to be over 10-15 which represents the main reason why we have decided to use it as a limit. Line equations are defined among following points w-w1, w1-w2, w2-w3, w3-w4, w4-w5 and ww2, w-w3, w-w4, w-w5 as presented at Figure 1.

Five basic response functions denoted as TCPO1, TCPO2, TCPO3, TCPO4, TCPO5 and additional four TCPO6, TCPO7, TCPO8 and TCPO9 are defined.

All of the response functions are defined with line equation with different slope. Several algorithms can be defined. The protocol will start with standard TCP Reno/NewReno mechanism, afterwards it will use TCPO1 and in case of lossless environment after passing w1 value it will use TCPO2 function, if loss is detected than TCPO6 is used as response function (even if the calculated window value after loss is below the switch point value after passing $w = 38$, TCPO6 will be used because it will provide improved window growth). After reaching w2 value TCPO3 will be used and if loss is detected TCP07 will be used. When $w3$ value is reached TCPO4 is used and if loss is detected TCPO8 will be used or TCP05 will be used after w4 value is reached. When loss is detected it switches to use of TCPO9 and after passing w5 value current response function will be preserved. In case of three consecutive timeouts the protocol counters are restarted and it restarts the initial usage of the response functions according the actual congestion window value (cwnd). It is interesting to notice that this protocol modifies the initial behavior of Reno/Newreno protocol used, during slow mode of operation. In order to calculate the window value after the loss we use:

$$w_{new} = \frac{w_{\max} + w_{\min}}{2} \tag{3}$$

where w_{max} denotes highest window value achieved prior the loss occurs and w_{min} is the decreased window value after the loss that corresponds with the second line of equation (1). Obtained value places the cwnd in the middle between highest achieved value before the loss and the calculated value that should be obtained after the loss.

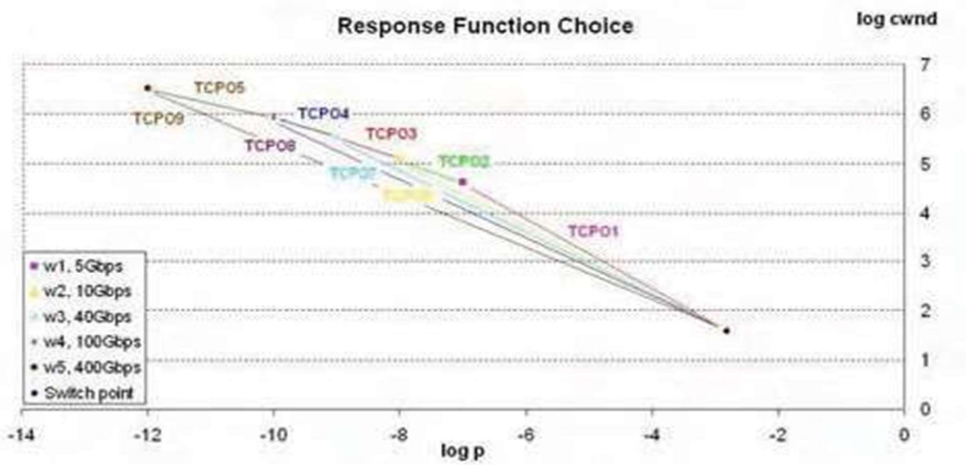

Figure 1. Response function choice of TCPOhrid

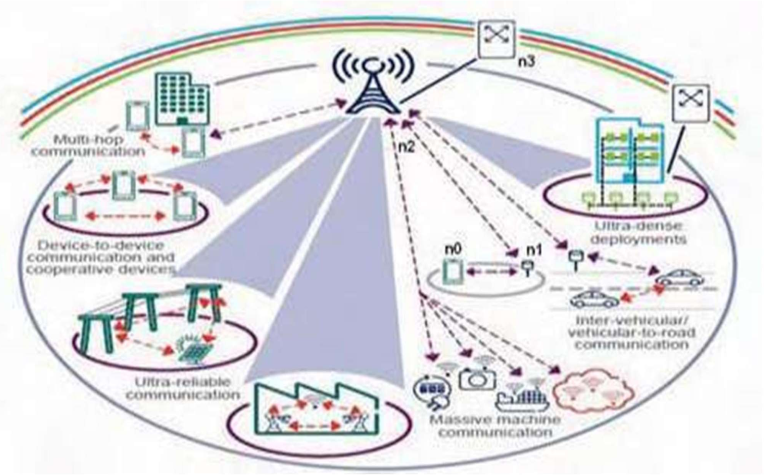

Figure 2. Simulation scenario. High speed backhaul link is used between MBS and core network gateway. We define wireless medium between n0 and n1, n1 is wirelessly connected with macro base station n2 and optical link n2-n3 is defined between the macro base station n2 and the network gateway n3

Similar window calculation algorithm is used in BIC protocol [8]. We find usage of (3) more interesting because it impacts β calculation. Till now we have defined the protocol response function variation and its loss window calculation. Packet loss probability p can be calculated for a given value of cwnd and α parameter is defined with equation (2). The β parameter is:

$$\beta_{calculated}(w) = (0.1 - 0.5)\frac{\log(w) - \log(38)}{\log(w_5) - \log(38)} + 0.5 \tag{4}$$

Where w represents the present value of the congestion window of interest and w5 is the congestion window size that defines the fifth switch point. Calculated value β of should be calibrated having in mind that the window calculation after the loss is modified, when equation (3) is used to calculate the window value after loss in slow mode of operation. If we substitute all known values in (3) we obtain

$$w_{new} = (1 - \frac{\beta}{2}) w_{\max} \tag{5}$$

Equation that indicates new β value is:

$$\beta_{new} = \frac{\beta_{calculated}}{2} \tag{6}$$

This equation is used only in case of time out, afterwards the protocol proceeds with standard β calculation. Since we have defined all important protocol parameters we can implement them in Network Simulator 2 (ns2). In the simulation calculation β is done by using:

$$\beta_{new} = 0.5 - \beta_{calculated} \tag{7}$$

With this equation we achieve increased concavity β of function that provides low cwnd decrement for low values and high cwnd decrement when timeout occurs for large window values. Usage of (7) provides improved protocol performance for large cwnd values when congestion occurs because the window will be decreased for 40% of the previous value. Lowest β_{new} value is bounded by 0.1 and the highest with 0.4. If the result of equation (7) is lower than 0.1 than β_{new} value is rounded at 0.1.

4. Simulation Scenario

We study simple simulation scenario presented at Figure 2 implemented with help of Network Simulator 2 (ns2). We simulate only one mobile user (n0) communication with micro base station (n1) and macro base station (n2) through backhaul high speed link n2-n3 connected with the adequate network gateway (n3).

Parameter of interest is congestion window size variation; we know that it is directly related with throughput values. Packet size is set at 1500 bytes. Simulation lasts 230 sec. The both cases, when maximum bound of window size is 20 packets and 80000 packets result in large trace files. Buffers are Drop Tail; buffer size is set at 100% of the product of bottleneck capacity by the largest RTT divided by packet size. Link speed can vary and high speed link has RTT value of 100 ms. We have conducted simulations for 5, 10, 40 and 100 Gbps and adequate buffer size limitations. Since the max window size is predefined, similar results are obtained regardless the speed and buffer size constraints.

Obtained results are presented at Figure 3 and Figure 4.

On Figure 3 we have presented cwnd change when link capacity is set at 5, 10, 40 and 100 Gbps. Simulation last 230 sec and the window size is set to ns2 default value of 20 packets. This figure justifies previously stated design parameters of Ohrid protocol. It can be noticed that Reno protocol has smallest window growth, directly impacting the throughput that can be achieved when this protocol is used. Ohrid has shown larger window growth compared to Reno but smaller than HSTCP. This proves that Ohrid protocol is more friendly with existing protocols than HSTCP and still it sustains its high speed performances. This protocol behavior assures that can be used to buffer the negative impact of wireless medium especially in case of heavy mobile users. At On Figure 4 we have obtained results when the window parameter is set at 80000. At the beginning of the simulation Reno protocol achieves lowest cwnd values while HSTCP cwnd change is better that the one achieved by Ohrid protocol mainly because of the rapid growth of the HSTCP response function. The Ohrid protocol has better performance which justifies the novel protocol design. The protocol uses more efficient set of response functions in case when the window size has larger value than w1 packets. It is obvious that the Ohrid protocol provides higher data rates than HSTCP protocol. Figure 4 assures us that the protocol is robust in case of increased terminal mobility, cwnd growth will be lower, hence making the protocol capable to cope wireless medium impact at the given throughput when the user is mobile and away from the base station. When the terminal is static or near the base station and the number of active users of the network is low than higher data rates can be achieved. Therefore, this protocol is appropriate for use in cell communication and for backhaul data transfer.

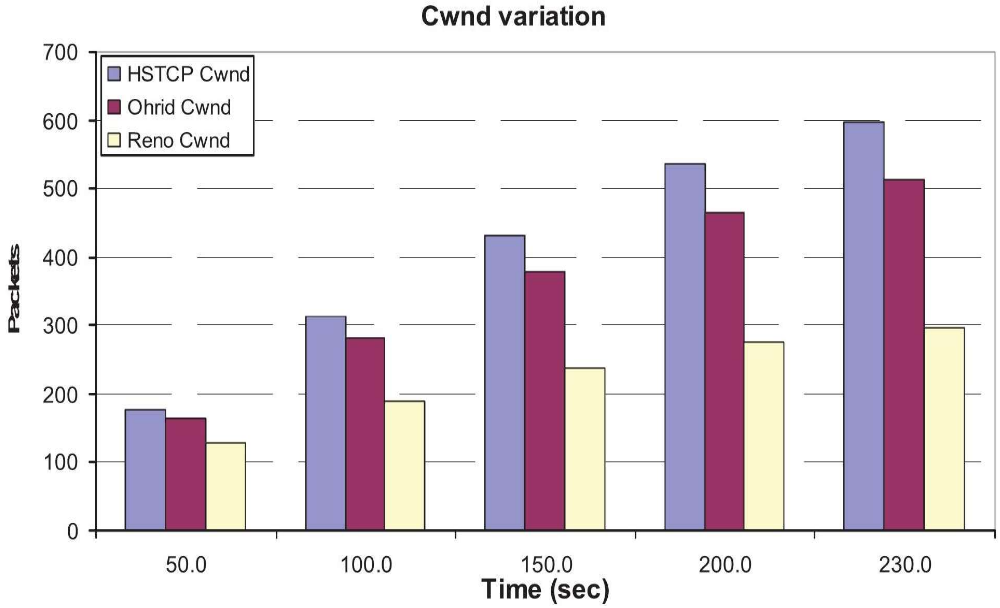

Figure 3. cwnd change when 5, 10, 40 and 100 Gbps links are used with adequate buffer sizes

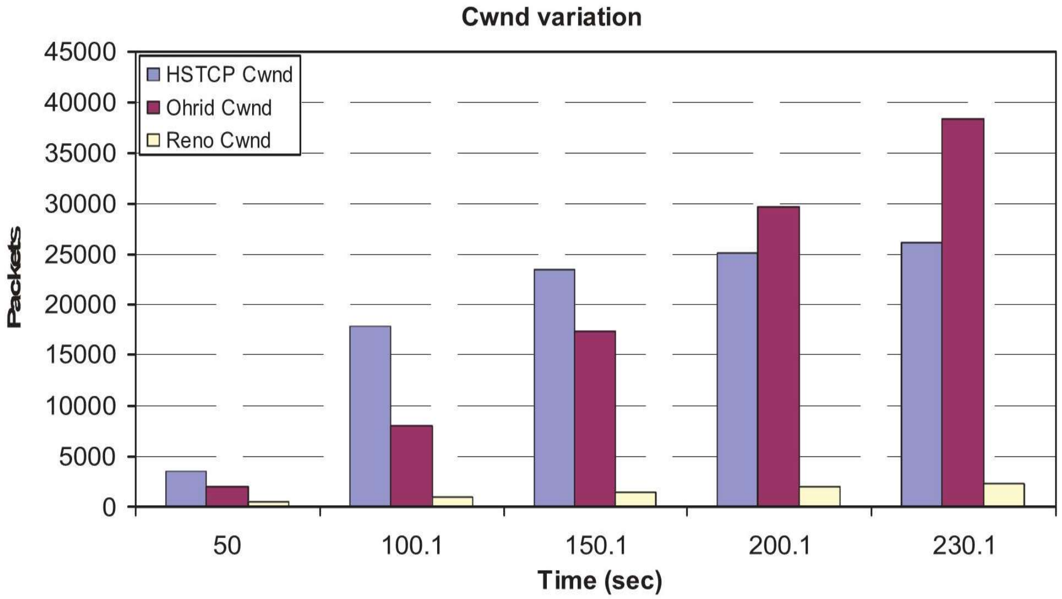

Figure 4. cwnd change when 5, 10, 40 and 100 Gbps links are used with adequate buffer sizes (max window bound is set at 80000)

5. Conclusion

In this paper we have presented and evaluated the TCP Ohrid protocol that has shown improved cwnd behavior. Increased friendliness and higher data rate performance of TCP Ohrid protocol are justified. The protocol is defined to be used in the future 5G data networks regardless of the engaged MAC Layer. It is defined with several response functions for speeds of 5, 10, 40, 100 and 400 Gbps. The used β parameter expressed with equation (7) guarantees improved behavior especially in high congested networks. Equation (3) is used to improve Reno/Newreno protocol performance till reaching cwnd value of 38 packets which provides faster increase when loss occurs during slow mode of operation. All used parameters can be fine tuned in order to properly design the protocol. Development of high speed protocols that will sustain 5G network capacity is essential hence user can benefit the network capacity. TCP Ohrid is a good candidate since it can support high transmission speeds and provide efficient usage of network capacity up to speeds of 400 Gbps.

References

[1] Duke, M., Braden, R., Eddy, W., Blanton, E. & Zimmermann, A. A road map for transmission control protocol (TCP). *Internet Engineering Task Force (IETF)*, RFC7414 (February 2015).

[2] Janevski, T. (2014). NGN Architectures, Protocols, and Services. John Wiley & Sons: UK.

[3] Rodriguez, J. (2015). Fundamentals of 5G Mobile Networks. John Wiley & Sons: UK.

[4] Ge, X., Cheng, H., Guizani, M. & Han, T. (2014). 5G wireless backhaul networks: Challenges and research advances. IEEE Network, 28, 6–11 [DOI: 10.1109/MNET.2014.6963798].

[5] Barbarossa, S., Sardellitti, S. & Di Lorenzo, P. (2014) Communicating While Computing: Distributed mobile cloud computing over 5G heterogeneous networks. *IEEE Signal Processing Magazine*, 31, 45–55 [DOI: 10.1109/MSP.2014.2334709].

[6] Kelly, T. (2003) Scalable TCP: Improving performance in highspeed wide area networks. ACM SIGCOMM Computer Communication Review, 33, 83–91 [DOI: 10.1145/956981.956989].

[7] Floyd, S. (2003). HighSpeed TCP for large congestion windows. RFC, 3649.

[8] Xu, L., Harfoush, K.. Rhee, I. (2004) Binary increase congestion control for fast, long distance networks. *In: Proceedings of the IEEE Infocom*, Vol. 4, pp. 2514–2524.

[9] Leith, D. (2008) H-TCP:TCP congestion control for high bandwidthdelay product paths. *IETF Internet Draft*. tools.ietf.org/html/draftleith-tcp-htcp-06.

[10] Ha, S., Rhee, I. & Xu, L. (2008) CUBIC: A new TCP-friendly high-speed TCP variant. ACM SIGOPS Operating Systems Review, 42, 64–74 [DOI: 10.1145/1400097.1400105].

[11] Marfia, G., Palazzi, C., Pau, G., Gerla, M., Sanadidi, M. & Roccetti, M. (2005) “TCP Libra: Exploring RTT-Fairness for TCP, *UCLA computer science department, tech. Rep*. UCLA-CSD. TR-050037.

[12] Caini, C., Firrincieli, R. (2004) TCP Hybla: A TCP enhancement for heterogeneous networks. *International Journal of Satellite Communications and Networking*, 22, 547–566 [DOI: 10.1002/sat.799].

[13] Baiocchi, A., Castellani, A.P., Vacirca, F. (2007). YeAH-TCP: Yet another highspeed TCP. In: Proceedings of the PFLDnet. Intercollegiate Studies Institute: Marina del Rey (Los Angeles), CA, USA.

[14] King, R., Baraniuk, R. & Riedi, R. (2005) TCP-Africa: An adaptive and fair rapid increase rule for scalable TCP. *In: Proceedings of the IEEE Infocom*, Volume 3, p 1838–1848.

[15] Tan, K., Song, J., Zhang, Q. & Sridharan, M. (2005). A Compound TCP Approach for High-Speed and Long Distance Networks.

[16] Liu, S., Basar, T. & Srikant, R. (2006) TCP-Illinois: A loss and delay-based congestion control algorithm for high-speed networks. *In: Proceedings of the First International Conference on Performance Evaluation Methodologies and Tools* (VALUETOOLS).

[17] Wei, D.X., Jin, C., Low, S.H. & Hegde, S. (2006) FAST TCP: Motivation, architecture, algorithms, performance. *IEEE/ACM Transactions on Networking*, 14, 1246–1259 [DOI: 10.1109/TNET.2006.886335].-

[18] Sing, J. Soh, B. (2005) TCP new Vegas: Improving the performance of TCP Vegas over high latency links. *In: Proceedings of the 4th IEEE International Symposium on Network Computing and Applications (IEEE NCA05)*, pp. 73–80.

[19] Yamada, K., Wang, R., Sanadidi, M.Y. & Gerla, M. (2004) TCP Westwood with agile probing:dealing with dynamic, large, leaky pipes. *IEEE Communication Society*.

[20] Kliazovich, D., Granelli, F. & Miorandi, D. (2008) Logarithmic window increase for TCP Westwood+ for improvement in high speed, long distance networks. *Computer Networks*, 52, 2395–2410 [DOI: 10.1016/j.comnet.2008.04.018].

[21] Shimonishi, H., Hama, T. & Murase, T. (2006) TCP-adaptive Reno for improving efficiency-friendliness tradeoffs of TCP congestion control algorithm. *In: Proceedings of the 4th International Workshop on Protocols for Fast Long Distance Networks*.

[22] Kaneko, K., Fujikawa, T., Su, Z. & Katto, J. (2007) TCP-Fusion: A hybrid congestion control algorithm for high-speed networks. *In:* Proceedings of the PFLDnet. *Intercollegiate Studies Institute: Marina del Rey* (Los Angeles), CA, USA.

[23] Gunduz, D., Stamatiou, K., Michelusi, N. & Zorzi, M. (2014) Designingintelligent energy harvesting communication systems. *IEEE Communications Magazine*, 52, 210–216 [DOI: 10.1109/MCOM.2014.6710085].