Transport Bottlenecks of Edge Computing in 5G Networks

Åke Arvidsson and Lars Westberg

***Abstract*—We consider bottlenecks of TCP throughput in scenarios with edge clouds and 5G cellular networks. By numerical examples from measurements in real networks, we show that edge clouds indeed improve throughput but that several, non-negligible bottlenecks remain. We therefore devise a concept where edge clouds connected directly to the radio access network can increase their transmission rates by relying on built in retransmissions (through quality of service features) and on the built in user fairness (through per-user buffers and scheduler policies). We then return to the numerical examples and show that our solution provides substantial gains and we conclude by identifying and discussing the remaining bottlenecks and the potential of an improved protocol.**

***Index Terms*—5G, TCP, Edge Computing.**

I. Introduction

As noted in our paper [1], data services in cellular networks have evolved over the last decades from 2G/GPRS, with bit rates on the order of 100 kbps and latencies on the order of 1 second, to 4G/LTE, with bit rates higher than 10 Mbps and latencies lower than 100 ms, while future 5G systems aim at bit rates higher than 100 Mbps and latencies lower than 10 ms, *e.g.*, [2], [3]. The increased capabilities combined with flexible smartphones and inventive services have lead to an explosion in traffic and applications, *e.g.*, [4]–[6].

The growing importance of cellular access and an increased focus on user experience, *e.g.*, [7]–[9], has triggered a lot of research into how to evolve and adapt the dominating transport protocol, TCP, to maintain high throughput under higher bandwidths and lower but highly variable delays by modifying the congestion control algorithm, *e.g.*, [10]–[15], and/or by adding cross layer interactions, *e.g.*, [16]–[21]. Without going into details, we note, however, that despite the fact that the above papers span over many years, none of the proposed TCP-versions (which in a comparison typically tend to come out well in some scenarios but to come out poor in other scenarios) or cross layer mechanisms (which typically have difficulties tracking rapid changes over long round trip times) have been widely deployed.

The increased availability of high speed internet accesses is also one of the key factors behind cloud computing [22], *i.e.*, replacing or offloading local computing resources by connecting to centrally located data centres over the Internet. While clouds offer advantages in terms of cost and flexibility [23], the relative remoteness of their locations will typically result in latencies well above those of 5G networks and, in particular, above what some typical 5G applications require [24]. One proposal to bridge this gap and to fully exploit the improvements in 5G is to deploy edge cloud services (ECS) [25] where latencies are minimised by placing computing resources in the immediate vicinity of user access points such as base stations. A typical deployment scenario is depicted in Fig. 1 where user equipment (UE) are connected to cloud services to the base station (in 5G terminology called "next generation NodeB" and abbreviated gNB) and a local breakout (LBO) function which separates services provided from central locations to those provided at the network edge.

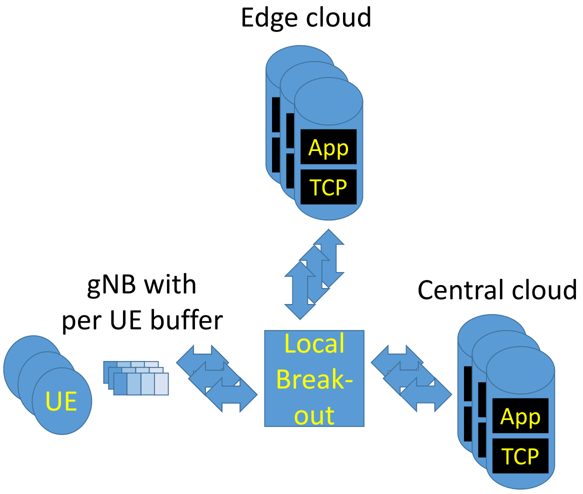

Fig. 1. Simplified view of how an ECS can be deployed at a gNB.

The current paper, which is built upon and extends the work in [1], is organised as follows: In Sec. II we report on our investigation of the performance of different cloud services over 5G, in Sec. III we define the scenario of 5G and ECS in more detail and briefly present our solution to increase link utilisation while the results are reported in Sec. IV. We then extend our analysis in Sec. V by identifying the limiting factors and quantifying their impact after which we discuss and evaluate the potential of a new and further improved

Manuscript received October 05, 2018; revised January 12, 2019. Date of publication March 18, 2019. The associate editor Prof. Dinko Begušić has been coordinating the review of this manuscript and approved it for publication.

Åke Arvidsson is with Kristianstad University, Kristianstad, Sweden (e-mail: ake.arvidsson@hkr.se).

Lars Westberg is with the Royal Institute of Technology, Stockholm, Sweden (e-mail: lgwe@kth.se).

Digital Object Identifier (DOI): 10.24138/jcomss.v15i1.637

protocol. Finally Sec. VI summarises the conclusions and outlines further work in this area.

II. Performance

To assess the performance of different cloud placements we define a set of scenarios for which we compare achieved request bit rates, defined as data volume divided by the time to serve the request, from initial the three way handshake to final receipt of the last acknowledgement.

A. Traffic Model

The scenarios are based on measurements in two real cellular networks A and B. The same data was used in [1] and collected by repeatedly requesting/downloading a set of selected pages/files with different server placements and collecting the resulting packet traces as pcap-files.

In network A, the dominant network in a country with a developed economy, we were able to consider three placements, *viz.* internal (at the edge of the cellular network), a local (domestic) host and a remote (overseas) host. In network B, the market leader in a country with a developing economy, we considered two placements, *viz.* internal (at the edge) and external (further away). While just about any public site can be classified as local/remote or external (hence data collection for these cases is straight forward), only operator sites count as internal (hence data collection for these cases required operator support). The latter was in both cases obtained under condition of anonymity.

While we use the same data as in [1], we apply a slightly different analysis. A specially written programme was used to read the pcap-files and, for each network n and location l, identify all $N_{\mathrm{F}}(n,l)$ flows and, for each flow, identify (i) all $N_{\mathrm{P}}(n,l)$ packets, all $N_{\mathrm{R}}(n,l)$ reordered packets and (iii) all $N_{\mathrm{D}}(n,l)$ duplicate packets. The results are summarised in Tab. I.

TABLE I
Number of Flows, Packets and Recoveries in the Data Set.

Network	Location	Flows	Packets	Recoveries
n	l	$N_{\mathrm{F}}(n,l)$	$N_{\mathrm{P}}(n,l)$	$N_{\mathrm{R}}(n,l)$+$N_{\mathrm{D}}(n,l)$
A	Internal	50,345	17,738,142,653	10,736
	Local	35,500	16,284,862,896	18,949
	Remote	18,324	2,109,021,252	103,886
B	Internal	7,963	2,765,023,472	69,843
	External	9,707	3,653,992,670	414,437

We then for each flow f estimated the RTT τ as the time between the SYN-packet and the SYNACK-packet (provided that neither of them were duplicated), the packet loss probability p as $(N_{\mathrm{R}}(f)+N_{\mathrm{D}}(f))/N_{\mathrm{P}}(f)$ and the fraction of spurious retransmissions η as $N_{\mathrm{D}}(f)/(N_{\mathrm{R}}(f)+N_{\mathrm{D}}(f))$. The results are summarised in Tab. II, Tab. III and Tab. IV in terms of the mean μ, the standard deviation σ and the 50^{th}, 90^{th} and 95^{th} percentiles P_{50}, P_{90} and P_{95} respectively.

From the above we selected the medians P_{50} of τ and the means μ of p and η. The first choice is based on the observation that Tab. II suggests the existence of a few very large values of the unbounded RTT (hence the mean is less relevant) while the other choices are based on the observation that many of the percentiles in Tab. III and Tab. IV are quite extreme and that both lost packets and repeated packets are bounded to the interval 0–100% (hence percentiles are less relevant).

TABLE II
Measured Values of RTT τ (seconds).

Network	Location	μ	σ	P_{50}	P_{90}	P_{95}
A	Internal	0.233	2.880	0.044	0.064	0.078
	Local	0.171	0.373	0.067	0.284	0.303
	Remote	0.371	0.450	0.310	0.526	0.699
B	Internal	0.270	0.379	0.124	0.679	0.885
	External	0.506	1.007	0.268	0.936	1.623

TABLE III
Measured Rates of Lost Packets p (percent).

Network	Location	μ	σ	P_{50}	P_{90}	P_{95}
A	Internal	0.03	238	0	0.0	0.0
	Local	1.24	5029	0	0.0	7.7
	Remote	2.39	4443	0	0.0	25.0
B	Internal	4.31	3764	0	12.5	25.0
	External	17.66	9496	0	72.7	86.7

TABLE IV
Measured Rates of Duplicate Packets η (percent).

Network	Location	μ	σ	P_{50}	P_{90}	P_{95}
A	Internal	1.74	27	0.0	3.6	9.1
	Local	22.07	132	2.1	100.0	100.0
	Remote	1.49	51	0.0	0.5	2.0
B	Internal	34.34	173	0.0	100.0	100.0
	External	5.68	66	0.0	0.0	75.0

TABLE V
Chosen Parameters from the Two Networks.

Network	Location	RTT (s)	Detected (%)	Spurious (%)
n	l	$\tau_{n,l}$	$p_{n,l}$	$\eta_{n,l}$
A	Internal	0.044	0.03	1.74
	Local	0.067	1.24	22.07
	Remote	0.310	2.39	1.49
B	Internal	0.124	4.31	34.34
	External	0.268	17.66	5.68

Using the parameters in Tab. V and the target RTT for 5G radio of 10 ms, we finally created seven different scenarios as specified in Tab. VI. Two of the scenarios (“A edge” and “B edge” apply to ECS. Note that packets in these scenarios only traverse the radio link where the “acknowledged mode” ensures that packets are neither lost nor disordered such that only spurious retransmissions remain. The other five scenarios serve as reference cases; internal clouds placed at the PGW (“A internal” and “B internal”), local and remote clouds in network network A (“A local” and “A remote”) and external clouds in network B (“B external”).

B. Results

The expected times to complete downloads in the above scenarios were then calculated for differently sized objects by means of the relatively complex but highly accurate TCP model in [26]. In the model, the client buffer was set large

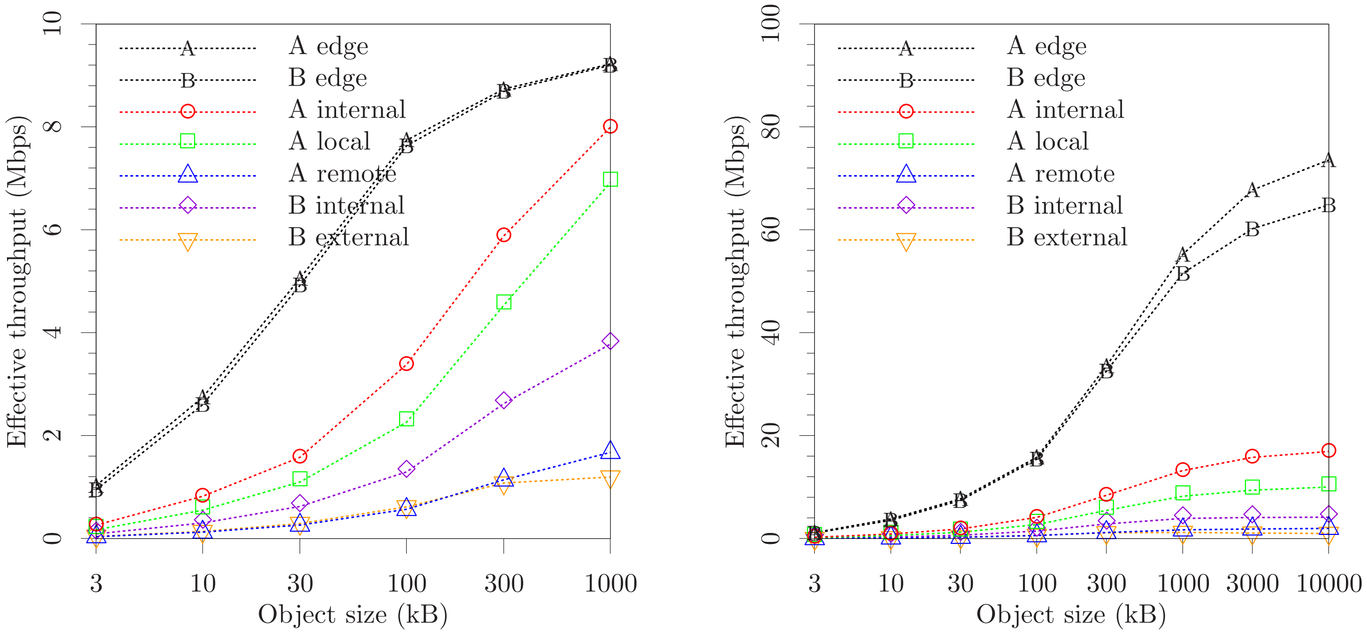

Fig. 2. Effective throughput *vs.* object size for the example scenarios on a 10 Mbps link (left) and a 100 Mbps link (right).

TABLE VI
SCENARIOS USED IN THE PERFORMANCE EVALUATION.

Name	RTT (s)	Loss (%)
A edge	0.01	$p_{\mathrm{A,Int}}\eta_{\mathrm{A,Int}}$
B edge	0.01	$p_{\mathrm{B,Int}}\eta_{\mathrm{B,Int}}$
A internal	$\tau_{\mathrm{A,Int}}$	$p_{\mathrm{A,Int}}$
A local	$\tau_{\mathrm{A,Loc}}$	$p_{\mathrm{A,Loc}}$
A remote	$\tau_{\mathrm{A,Rem}}$	$p_{\mathrm{A,Rem}}$
B internal	$\tau_{\mathrm{B,Loc}}$	$p_{\mathrm{B,Loc}}$
B external	$\tau_{\mathrm{B,Ext}}$	$p_{\mathrm{B,Ext}}$

enough not to become a bottleneck and the network connection was modelled by a manually selected bandwidth and the measured values of latency and loss. It is noted that the rapid variations of the radio channel, although not modelled explicitly, partly impact the results through the measured spurious losses.

We then computed the effective throughputs as the object sizes divided by the calculated download times and the results for two different bit rates are shown in Figure 2.

It is seen in both figures that the two curves for ECS, “A edge” and “B edge” are well above the curves for the other cloud locations which means that ECS, thanks to the shorter round trip times and lower loss rates, offers significant throughput gains even when compared to the second best option internal clouds. It is, however, also noted that link utilisation typically is well below 100% even with ECS in place. This is, first, because of the way TCP probes the channel and reacts to losses and, second, because of fundamental factors like three way handshake *etc.* taking non-zero time. The aim of this paper is to *solve* the first aspect and to *understand* the second aspect.

Finally we remark that all results may be a bit optimistic since radio channels in 5G networks (which are not yet commercially available) will exhibit larger and faster variations [27], [28] and thus will trigger more spurious retransmissions than radio channels in 4G networks (from which our data is taken). By comparing the results in network A, with lower loss rates, to those in network B, with higher loss rates, it is seen that higher rates reduce throughput hence we can conclude that all curves would drop should we assume higher loss rates.

III. IMPROVED TRANSPORT

A. Scenario

To improve throughput we first note that solutions like improved versions of TCP and/or cross layer interactions have certain issues and limitations which, as noted above, have resulted a weak interest (or no interest at all) in such proposals. First, all TCP versions must handle unknown and variable environments and this necessitates conservative congestion control policies with slow increments/fast decrements and excludes exploiting particular properties which under some circumstances might be known. That is, the degree to which TCP can be optimised is limited by the need for generality. Second, cross layer interactions require not only single vendor solutions or internationally agreed standards but also openness about operational data. In this context we note that few or no operators accept being dependent on single suppliers, that standard definitions typically are time consuming processes and, possibly more important, that openly communicating the state of a network can raise concerns with respect to, *e.g.*, network stability, user privacy and market impression.

Our approach is therefore to let operators with ECS connected directly to their gNBs collect and apply knowledge within their own, confined environments without becoming dependent on single vendors or waiting for new standards, and with a minimal amount of additional equipment. In particular, we take advantage the orderly way in which packets are delivered over radio links in acknowledged mode (hence TCP does not need to perform packet retransmissions) and the fairness implemented by the radio scheduler (hence TCP does not need to maintain fair sharing).

B. Concept

As suggested above, our concept is based on two major observations. The first observation is the fact that, by placing the ECS at the gNB, the retransmission mechanisms of the link protocol apply end-to-end and thus provide a virtually lossless connection hence the retransmission function of TCP becomes redundant. We exploit this by turning this mechanism off and in this way we do away with spurious timeouts and other issues which would have reduced throughput without providing any clear benefits. The second observation is the fact that, again because of the placement of ECS, per-user buffers and gNB scheduling take care of user fairness on the entire path hence the fairness considerations in TCP become redundant. We exploit this by applying a carefully inflated initial window (IW) such that high transmission rates are reached immediately and then sustained even in the presence of delayed acknowledgements. We also note that the two aspects are related in that reacting to losses is an important mechanism to achieve fairness. As this does not apply in our scenario we conclude that any possible content verification preferably is placed in the application layer (as is already the case in QUIC).

An implementation of our concept may be illustrated as in Fig. 3. The figure shows how a TCP configuration function controls the settings of TCP in the ECS based on data from a traffic probe (TP). The additional functions may in practise be part of the LBO and/or the ECS.

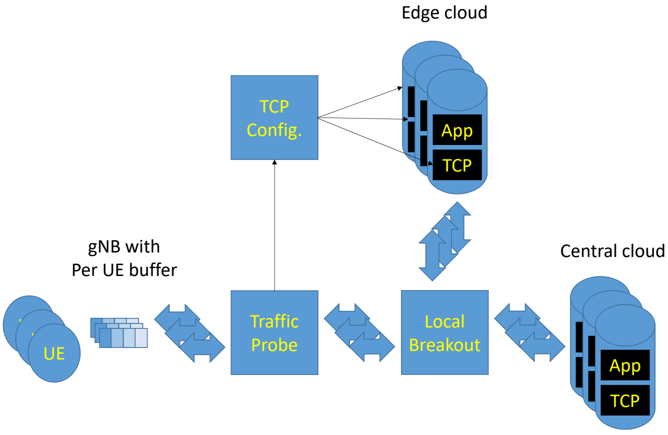

Fig. 3. Simplified view of how our concept can be applied to ECS deployed at the gNB.

A request-response interaction in this scenario would typically include the following steps

1) A client at a UE requests an object.
2) The request is intercepted by the LBO which determines if the object should be provided from the edge cloud or from a central cloud.
3) The selected cloud sends data through the LBO passing the TP and to the gNB which then sends it to the UE.
4) The TP keeps track of this and other flows as discussed below and makes this information available to a TCP configuration function (TCF) in the edge cloud.
5) The TCF uses the information from the TP to set the IW of TCP for any request being served from the edge cloud.

The TP can work in several ways as discussed in [1] and we note that probing does not need to occur in the pipe (which could cause extra delay) but that the probe can operate on copies obtained from, *e.g.*, an optical splitter.

IV. Results

To assess the potential of our concept we extend the scenarios above to include our solution ("Optimised" with the same RTT as the other ECS cases but with a larger initial window and no spurious losses), Tab. VII.

TABLE VII
Additional Scenario Used in the Performance Evaluation.

Case	RTT (s)	Loss (%)
Optimised	0.01	0

Using the same procedure as above we obtained the results shown in Fig. 4.

It is seen that the three curves for ECS almost overlap for 10 Mbps but that the curve for "Optimised" is well above the two other ECS curves for 100 Mbps which means that cloud location alone is sufficient to utilise low speed radio links while optimised protocols like the our concept are necessary to utilise high speed radio links.

Moreover, it is seen that the gap between the curves "A edge" and "B edge" is smaller than the gap to the curve for "Optimised" which, since losses are almost negligible in "A edge" but relatively high in "B edge", suggests that the most important part of the optimised concept is the inflation of the IW. It is, however, important to recall that the impact of spurious losses present in "A edge" and "B edge" may be underestimated in which case the benefits from avoiding spurious timeouts would increase.

V. Analysis

It is noted that throughput still is well below the link capacities. This is because of four limiting factors, *viz.*,

- the time of three way handshake,
- the data carried in packet headers,
- the wait for the last acknowledgement and
- the propagation time of the first packet.

The impact of each of these factors, denoted by "3WHS", "OH", "ACK" and "Delay" respectively, is illustrated in Fig. 5 and where "Optimised" refers to our proposed solution.

The first two factors may be subject of further studies as they to some extent depend on the protocol; *i.e.*, they may be reduced or eliminated by, *e.g.*, resorting to a connectionless protocol or making the connections (semi-)permanent or implicit and applying header compression respectively. Noting that the impact of three way handshake by far exceeds the impact of protocol overhead and that there are existing

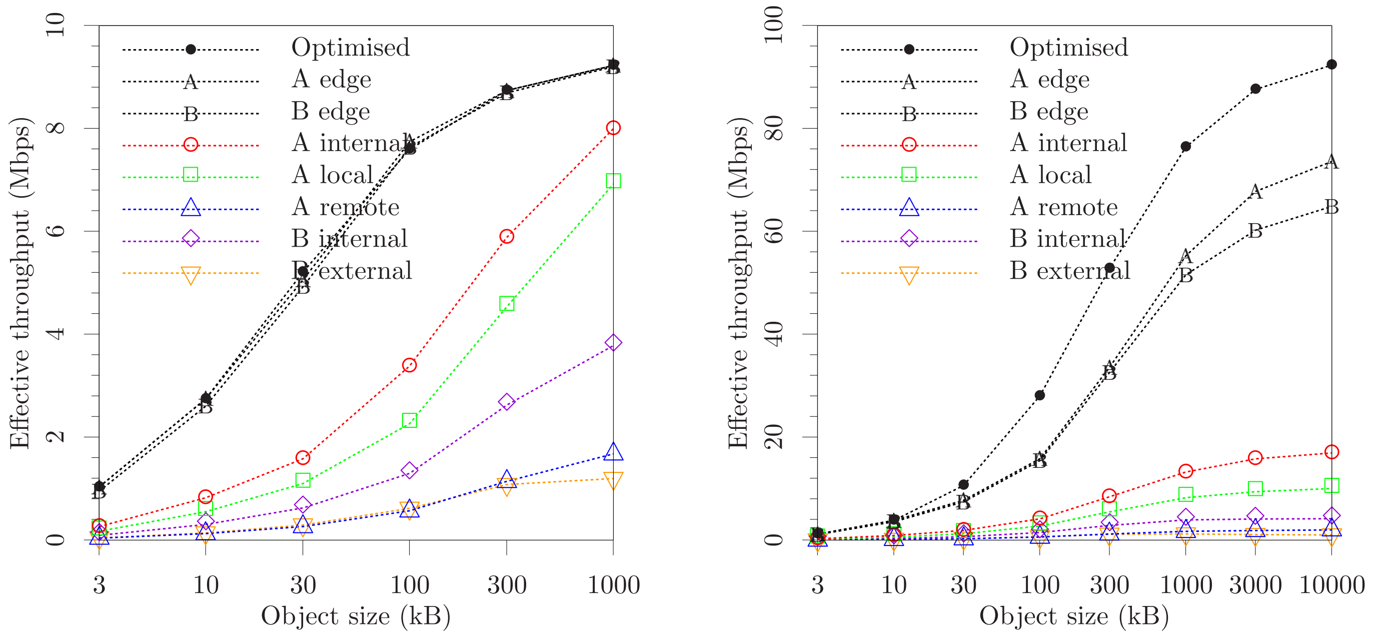

Fig. 4. Effective throughput *vs.* object size for the example scenarios on a 10 Mbps link (left) and a 100 Mbps link (right) including our proposed solution.

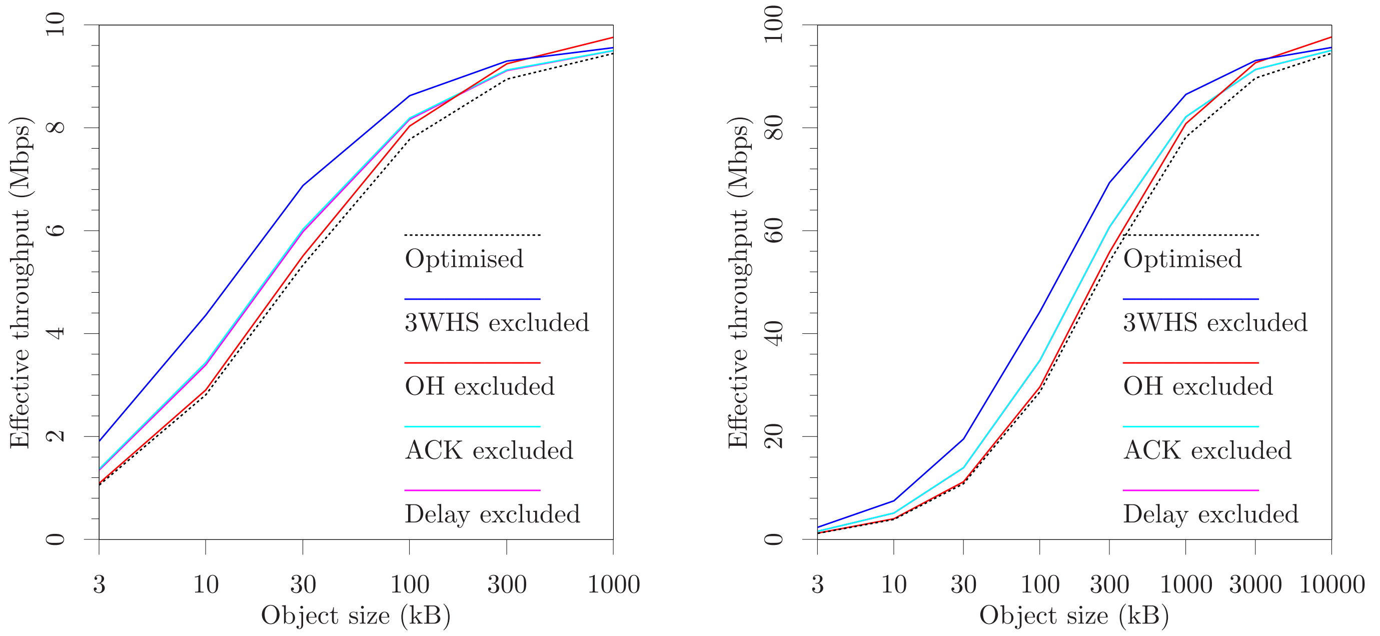

Fig. 5. Impact of different limiting factors in terms of effective throughput *vs.* object size for an edge cloud using 10 Mbps link (left) and a 100 Mbps link (right).

standards for header compression, we conclude that the former factor is the most relevant one for further work. The third factor has no technical implications but is a matter of counting; *i.e.*, shall throughput be based on when the data has arrived at the destination or on when the originator has been informed of this? The fourth factor is determined by the radio link protocol and out of scope of this study. We note that the last two factors both are dominated by the propagation time of half an RTT (hence the two curves tend to overlap) while only factor three also includes the time to transmit an acknowledgement message (hence its impact is somewhat larger especially for the lower bit rate).

The aggregated improvements by factors (i)–(iii) can thus be seen as the potential improvement from a protocol with no three way handshake, reduced overhead and where a final acknowledgement is not necessary. The performance of such a protocol in our scenario is shown in Fig. 6 where "Optimised TCP" refers to our solution whereas "New protocol" refers to the combined effect of also removing limitations (i)–(iii).

It is noted that, despite the considerable improvements offered by our proposed optimisations of TCP, there is significant possibilities for a new protocol to further improve performance. In practical terms such a protocol could be based on UDP along the lines of, *e.g.*, [29]. Such a protocol could also also take further advantage of the flexible nature of the radio link layer and if necessary even designing a separate

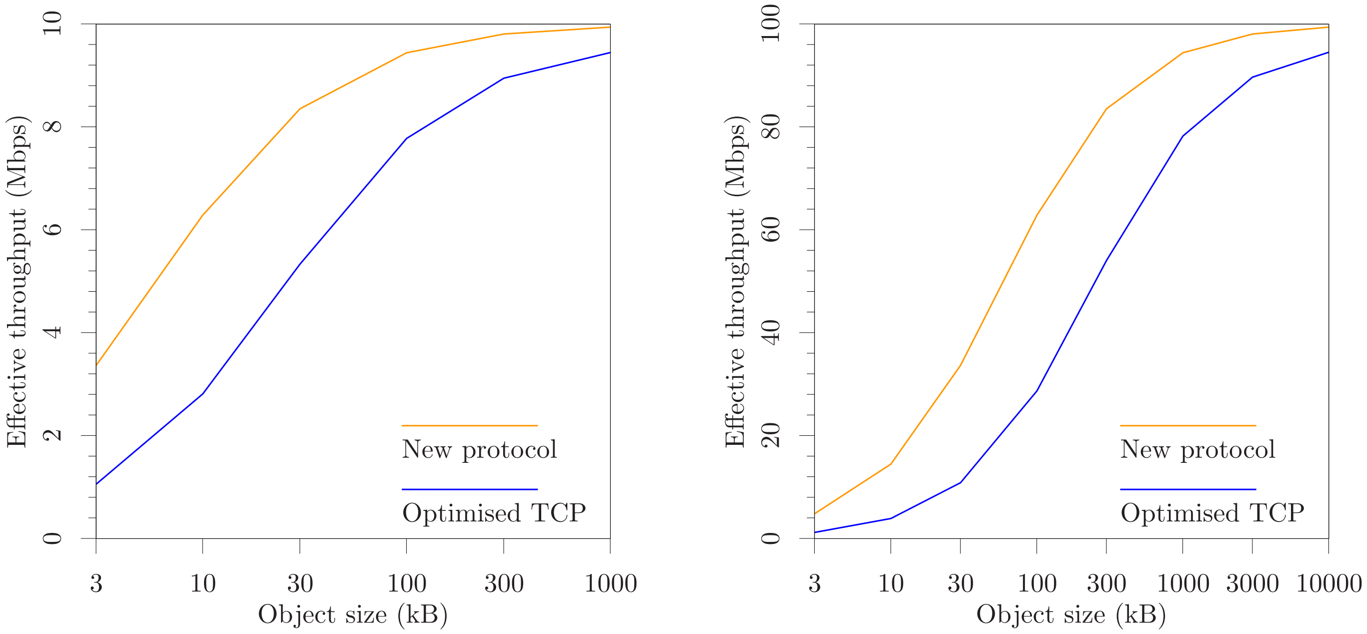

Fig. 6. Remaining potential for a new protocol which eliminates three way handshake, protocol overhead and the need for a final acknowledgement *vs*. object size for an edge cloud using 10 Mbps link (left) and a 100 Mbps link (right).

QCI (Quality-of-service Class Identifier) value.

VI. Conclusions and Further Work

We have demonstrated by means of real data that cloud servers will not be able to take full advantage of the high speed and low RTT offered by 5G networks, and that moving the servers from remote locations to network gateways or to the radio network only partially reduces the problem. The reason for the relatively low effective throughput is that it takes a long time to scale up the congestion window of TCP and that non-negligible losses (actual ones and spurious ones) tend to scale the same window down.

We have also exploited the fact that the gNB has "built in" user fairness (through per-user buffers and scheduler policies) and loss recovery (through quality of service features) and proposed an optimised concept where the congestion window is inflated straight from the start and where no spurious retransmissions are triggered. The calculations demonstrate that our concept is efficient and significantly improves the effective throughput and we have demonstrated by means of a realistic example how our solution helps reducing latencies.

We have finally identified and characterised the remaining bottlenecks as well as the aggregated potential of a new protocol which takes full advantage of edge clouds in 5G networks.

References

[1] Å. Arvidsson and L. Westberg, "Fast Transport for Edge Computing in 5G Networks," in *Proc. IEEE SoftCOM 2018*, September 2018, p. S2.1. doi: 10.23919/SOFTCOM.2018.8555815

[2] Ramnarayan, V. Kumar, and V. Kumar, "A New Generation Wireless Mobile Network — 5G," *International Journal of Computer Applications*, vol. 70, no. 20, pp. 26–29, May 2013. doi: 10.5120/12185-8272

[3] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A Survey on Low Latency Towards 5G: RAN, Core Network and Caching Solutions," *ArXiv e-prints*, Aug. 2017. doi: 10.1109/COMST.2018.2841349

[4] Ericsson, "Ericsson Mobility Report," White paper, pp. 1–36, June 2017. [Online]. Available: https://www.ericsson.com/en/mobility-report

[5] "Number of available applications in the Google Play Store from December 2009 to June 2017," https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/, accessed: 2017-09-15.

[6] "Number of available apps in the Apple App Store from July 2008 to January 2017," https://www.statista.com/ statistics/263795/number-of-available-apps-in-the-apple-app-store/, accessed: 2017-09-15.

[7] S. Egger, T. Hoßfeld, R. Schatz, and M. Fiedler, "Waiting times in quality of experience for web based services," in *Fourth IEEE International Workshop on Quality of Multimedia Experience (QoMEX 2012)*, I. S. Burnett, Ed., July 2012, pp. 86–96. doi: 10.1109/QoMEX.2012.6263888

[8] L. Plissonneau and E. Biersack, "A longitudinal view of http video streaming performance," in *Third ACM Multimedia Systems Conference (MMSys'12)*, 2012, pp. 203–214. doi: 10.1145/2155555.2155588

[9] A. Balachandran, V. Aggarwal, E. Halepovic, J. Pang, S. Seshan, S. Venkataraman, and H. Yan, "Modeling Web Quality-of-experience on Cellular Networks," in *20th IEEE Annual International Conference on Mobile Computing and Networking (MobiCom '14)*, 2014, pp. 213–224. doi: 10.1145/2639108.2639137

[10] K. Brown and S. Singh, "M-TCP: TCP for Mobile Cellular Networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 5, pp. 19–43, October 1997. doi: 10.1145/269790.269794

[11] R. Ludwig and R. H. Katz, "The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions," *SIGCOMM Comput. Commun. Rev.*, vol. 30, no. 1, pp. 30–36, January 2000. doi: 10.1145/505688.505692

[12] T. Goff, J. Moronski, D. S. Phatak, and V. Gupta, "Freeze-TCP: A True End-to-End TCP Enhancement Mechanism for Mobile Environments," in *IEEE INFOCOM*, 2000, pp. 1537–1545. doi: 10.1109/INFCOM.2000.832552

[13] C. Casetti, M. Gerla, S. Mascolo, M. Sanadidi, and R. Wang, "TCP Westwood: End-to-End Congestion Control for Wired/Wireless Networks," *Wireless Networks*, vol. 8, no. 5, pp. 467–479, September 2002. doi: 10.1023/A:1016590112381

[14] C. P. Fu and S. C. Liew, "TCP Veno: TCP enhancement for transmission over wireless access networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 2, pp. 216–228, February 2003. doi: 10.1109/JSAC.2002.807336

[15] K. Xu, Y. Tian, and N. Ansari, "TCP-Jersey for wireless IP communications," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 4, pp. 747–756, 2004. doi: 10.1109/JSAC.2004.825989

[16] I. Cabrera Molero, N. Möller, J. Petersson, R. Skog, Å. Arvidsson, O. Flärdh, and K. H. Johansson, "Cross-layer adaptation for TCP-based applications in WCDMA systems," in *IST Mobile & Wireless Communications Summit, Dresden, Germany*, 2005.

[17] M. Li, “Cross-layer Resource Control to Improve TCP Performance over Wireless Network,” in *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)*, 2007, pp. 706–711. doi: 10.1109/ICIS.2007.86

[18] F. Ren, X. Huang, F. Liu, and C. Lin, “Improving TCP Throughput over HSDPA Networks,” *IEEE Transactions on Wireless Communications*, vol. 7, no. 6, pp. 1993–1998, June 2008. doi: 10.1109/TWC.2008.061007

[19] V. Singh, J. Ott, and I. D. Curcio, “Rate adaptation for conversational 3G video,” in *IEEE INFOCOM Workshops 2009*. IEEE, 2009, pp. 1–7. doi: 10.1109/INFCOMW.2009.5072183

[20] F. Lu, H. Du, A. Jain, G. M. Voelker, A. C. Snoeren, and A. Terzis, “CQIC: Revisiting Cross-Layer Congestion Control f or Cellular Networks,” in *16th International Workshop on Mobile Computing Systems and Applications (HotMobile)*, 2015, pp. 45–50. doi: 10.1145/2699343.2699345

[21] J. D. Beshay, A. T. Nasrabadi, R. Prakash, and A. Francini, “Link-Coupled TCP for 5G networks,” in *25th IEEE/ACM International Symposium on Quality of Service (IWQoS)*, June 2017, pp. 1–6. doi: 10.1109/IWQoS.2017.7969170

[22] M. D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, and A. Vakali, “Cloud Computing: Distributed Internet Computing for IT and Scientific Research,” *IEEE Internet Computing*, vol. 13, no. 5, pp. 10–13, September 2009. doi: 10.1109/MIC.2009.103

[23] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, and A. Ghalsasi, “Cloud computing The business perspective,” *Decision Support Systems*, vol. 51, no. 1, pp. 176–189, 2011. doi: 10.1016/j.dss.2010.12.006

[24] A. Neal, “Edge computing,” Final Deliverable, NGMN Alliance, 5G P1 Requirements & Architecture— Work Stream End-to-End Architecture, Tech. Rep., September 2016. [Online]. Available: https://www.ngmn.org/uploads/media/161010_NGMN_Edge_Computing_v1_0.pdf

[25] S. Islam and J.-C. Grégoire, “Network Edge Intelligence for the Emerging Next-Generation Internet,” *Future Internet*, vol. 2, no. 4, pp. 603–623, 2010. doi: 10.3390/fi2040603

[26] Å. Arvidsson and A. Krzesinski, “A model of a TCP link,” in *15th Specialist Seminar on Internet Traffic Engineering and Traffic Management*. ITC, July 2002, pp. 68–77.

[27] M. Zhang, M. Mezzavilla, R. Ford, S. Rangan, S. Panwar, E. Mellios, D. Kong, A. Nix, and M. Zorzi, “Transport layer performance in 5G mmWave cellular,” in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2016, pp. 730–735. doi: 10.1109/INFCOMW.2016.7562173

[28] M. Polese, R. Jana, and M. Zorzi, “TCP in 5G mmWave networks: Link level retransmissions and MP-TCP,” in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, May 2017, pp. 343–348. doi: 10.1109/INFCOMW.2017.8116400

[29] Tom Herbert, “Transport layer protocols over UDP,” Working Draft, IETF Secretariat, Internet-Draft draft-herbert-transports-over-udp-00, May 2016. [Online]. Available: http://www.ietf.org/internet-drafts/draft-herbert-transports-over-udp-00.txt

Åke Arvidsson obtained his M.Sc. and Ph.D. degrees in Electrical Engineering from Lund University, Sweden, in 1982 and 1990 respectively. He has worked with several consultancy companies and held various academic positions in Sweden and Australia and became full professor of teletraffic systems at Blekinge Institute of Technology in 1995. In 1998 he joined Ericsson as a Technical Expert in Data Traffic Theory and since 2018 he is full professor at Kristianstad University. His current research interests include performance analysis and optimisation of cellular networks, transport protocols and content distribution.

Lars Westberg obtained his M.Sc. and Lic.D. degrees from the Royal Institute of Technology Stockholm, Sweden, in 1990 and 1993 respectively. In 1977 he joined Ericsson and since 1995 he has been with Ericsson Research. His current research interests include content delivery, cloud computing in mobile networks.