

# TCP in 5G mmWave Networks: Link Level Retransmissions and MP-TCP

Michele Polese\*, Rittwik Jana<sup>†</sup>, Michele Zorzi\*

\*Department of Information Engineering, University of Padova, Italy

e-mail: {polesemi, zorzi}@dei.unipd.it

<sup>†</sup>AT&T Labs-Research, Bedminster NJ, USA

e-mail: rjana@research.att.com

**Abstract**—MmWave communications, one of the cornerstones of future 5G mobile networks, are characterized at the same time by a potential multi-gigabit capacity and by a very dynamic channel, sensitive to blockage, wide fluctuations in the received signal quality, and possibly also sudden link disruption. While the performance of physical and MAC layer schemes that address these issues has been thoroughly investigated in the literature, the complex interactions between mmWave links and transport layer protocols such as TCP are still relatively unexplored. This paper uses the ns-3 mmWave module, with its channel model based on real measurements in New York City, to analyze the performance of the Linux TCP/IP stack (i) with and without link-layer retransmissions, showing that they are fundamental to reach a high TCP throughput on mmWave links and (ii) with Multipath TCP (MP-TCP) over multiple LTE and mmWave links, illustrating which are the throughput-optimal combinations of secondary paths and congestion control algorithms in different conditions.

## I. INTRODUCTION

MmWave communications are expected to play a major role in reaching the performance target of the next generation of mobile networks (5G) [1]. At mmWave frequencies (i.e., above 10 GHz), indeed, there is a high availability of contiguous bandwidth that can be allocated to cellular networks. However, mmWave communications also present challenges and issues that must be faced in order to make this technology market-ready. In fact, these frequencies suffer from high isotropic pathloss (compensated by massive MIMO and beamforming gains) and blockage by solid materials, as for example buildings, cars, and also the human body [2].

These extreme propagation conditions demand a careful design of the PHY and MAC layers [3], but also have an impact on the interplay with the higher layers of the protocol stack. In particular, the congestion control mechanisms of the Transmission Control Protocol (TCP) may suffer from long blockages that trigger a Retransmission Timeout (RTO), and may not timely track the channel state when the link from a User Equipment (UE) to the serving evolved Node Base (eNB) switches from a Line-of-Sight (LOS) to a Non-Line-of-Sight (NLOS) condition [4]. Finally, TCP may take a long time to fill the huge amount of bandwidth available, and this penalizes short-lived TCP sessions such as those used for browsing or instant messaging applications.

In this paper, we systematically analyze the performance of the Linux kernel TCP/IP stack implementation over mmWave links using the mmWave module for the ns-3 simulator. First, we study the interaction with lower-layer retransmission protocols, in terms of both throughput and latency, proving

that without these retransmissions TCP is only able to reach a fraction of the potential mmWave NLOS link throughput. Secondly, we show that multi-path transmissions improve the performance of the mmWave network, by using Multipath TCP (MP-TCP) with different congestion control algorithms over (i) an LTE and a mmWave link or (ii) two mmWave links with different carrier frequencies. Finally, we test the multi-path transmission of TCP ACKs only, measuring in which conditions sending the TCP ACK packets over an LTE link (and data over a mmWave one) improves the throughput in a mobility scenario.

The rest of the paper is organized as follows. Sec. II describes the simulation setup, while Sec. III illustrates the performance analysis involving the lower-layer retransmission. Sec. IV presents the results for MP-TCP in mmWave networks, and Sec. V those for TCP ACK transmission on LTE links. Finally in Sec. VI some conclusions are drawn, and future work is suggested.

## II. SIMULATION SETUP

The simulations use the NYU ns-3 module [5], [6], with the extensions developed in [7], plugged to the TCP implementation of the Linux kernel (also including MP-TCP [8]) using a custom version of the Direct Code Execution (DCE) library [9]. In this way it is possible to test the real Linux implementation of TCP and MP-TCP with the flexibility of a network simulator. This approach can be applied to any application layer software or transport protocol, provided they only use the subset of kernel methods also available in DCE. It is possible to plug different applications on top, and the following experiments use IPERF [10] and Linux wget, in order to test respectively the throughput of the end-to-end connection and the time it takes to download files of different sizes.

The NYU ns-3 module [6] simulates an end-to-end cellular network, with a complete mobile stack (with custom TDD-based PHY and MAC layers, RRC layer, and legacy LTE RLC and PDCP layers) which is able to transmit packets

Parameter	Value
mmWave carrier frequency	28 GHz, 73 GHz
mmWave bandwidth	1 GHz
mmWave TX power	30 dBm
LTE carrier frequency (DL)	2.1 GHz
LTE carrier frequency (UL)	1.9 GHz
LTE bandwidth	20 MHz
LTE downlink TX power	30 dBm
LTE uplink TX power	25 dBm

Table I: Simulation parameters

from the UE to a remote host, or vice versa. The main feature of the NYU ns-3 module is the channel model for the 28 GHz and the 73 GHz carrier frequencies, based on real measurements [11], which can either statistically simulate LOS-NLOS-outage transitions [5] or rely on the ns-3 building module to track when a mobile terminal is in LOS or NLOS [6], [12]. The main simulation parameters are those typically used in the performance analysis of mmWave networks [12], and are summarized in Table I.

### III. INTERACTION WITH LOWER-LAYER RETRANSMISSION PROTOCOLS

Current and future mobile networks deploy different retransmission mechanisms in order to prevent packet loss and increase the throughput at the mobile devices. When using mmWave links, these retransmission protocols become a key element in hiding the highly dynamic and consequently unstable behavior of the channel to the higher layer transport protocols.

At the MAC layer, Hybrid Automatic Repeat reQuest (HARQ) is used. When the PHY layer at the receiver receives a packet, but detects the presence of some errors that prevent reliable decoding, it asks for a retransmission. The sender then transmits additional redundancy that helps retrieve the correct version of the packet [13]. Moreover, in 3GPP networks (e.g., LTE), there is a layer on top of the MAC layer that may perform additional retransmissions, called Radio Link Control (RLC) layer, that will also likely be a part of the 5G protocol stack [14]. Since the number of retransmissions at the MAC layer is usually limited (typically only 3 attempts are performed), the RLC layer Acknowledged Mode (AM) offers another way of recovering lost packets. Thanks to periodic reports from the receiver, the RLC AM sender knows which packets are missing and can retransmit them. The number of attempts that RLC AM can perform is also limited, and, if some packets are still missing, a Radio Link Failure is declared. RLC Unacknowledged Mode instead does not perform any retransmission in addition to those of the HARQ at the MAC layer. These retransmission mechanisms operate based on information related to the link and with a greater timeliness with respect to TCP, which instead uses packet losses to detect congestion and operates on the larger timescale of retransmission time-outs (RTOs), of the order of a second.

In order to test the effectiveness of coupling TCP with lower-layer retransmission mechanisms, we performed some simulations using the framework described in Sec. II, where we considered an uplink connection from a User Equipment (UE) placed at different distances from an evolved Node Base (eNB). We use IPERF on top of the Linux implementation of TCP CUBIC, with the statistical channel model [5], and perform Monte-carlo simulations for each distance  $d \in \{50, 75, 100, 150\}$  m.

RLC AM introduces additional redundancy in order to perform the retransmissions, but, when the distance between the eNB and the UE is equal to  $d = 50$  m and the UE is in LOS with very high probability, these retransmissions are not actually needed, because of the low packet error rate of the channel. Therefore, as also shown in Fig. 1, the throughput is lower when RLC AM is used (though by only a minimal

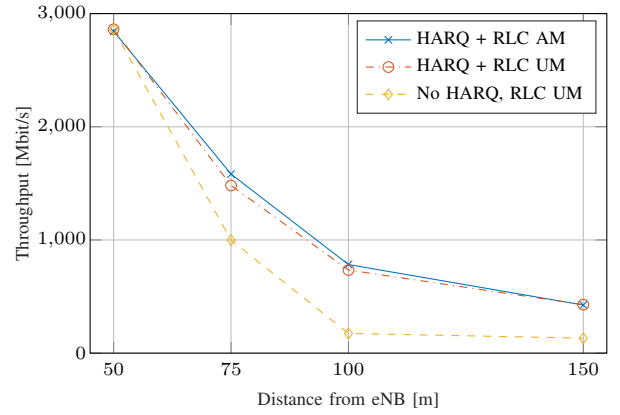


Figure 1: Throughput of TCP CUBIC with and without the different retransmission mechanisms of the mmWave protocol stack.

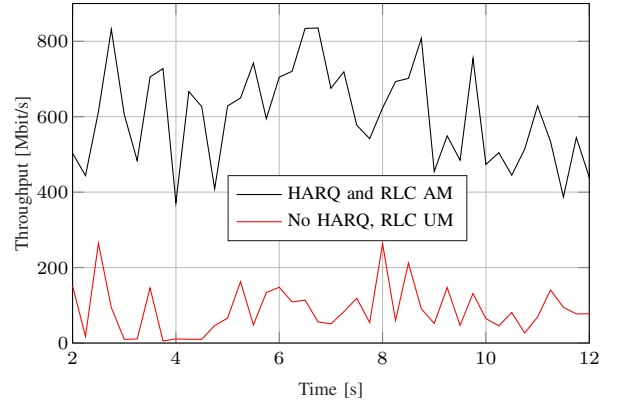


Figure 2: Example of TCP CUBIC flow with and without HARQ and RLC AM retransmissions. The UE is at 150 m from the eNB.

amount). As the distance increases, instead, the performance of TCP without HARQ and without RLC AM collapses, because the TCP congestion control algorithm sees a very lossy link and triggers congestion avoidance mechanisms or, worse, a RTO. Fig. 2 compares the traces at the PDCP layer of the throughput of a simulation over time, for  $d = 150$  m. It can be seen that the lack of HARQ and RLC AM places the whole burden of retransmissions on TCP, which does not manage to reach the high throughput allowed by the available bandwidth.

If instead we compare the performance of HARQ with RLC UM and that of HARQ with RLC AM, it can be seen from Fig. 1 that the additional retransmissions given by RLC AM increase the throughput by 100 Mbit/s at  $d = 75$  m and 50 Mbit/s at  $d = 100$  m. For  $d = 150$  m, instead, RLC AM does not improve the performance of RLC UM, showing that at such distance even further transmission attempts fail to successfully deliver packets (for example, because of extended outage events).

RLC AM at large distances instead increases the latency of successfully received packets, as shown in Fig. 3, because of retransmissions and additional segmentation that may introduce Head of Line (HoL) blocking delays. The smallest latency is achieved without HARQ and with RLC UM, because no retransmissions are performed, but this option is not able to deliver a high TCP throughput in general.

Fig. 4 shows the download time for a file of different sizes

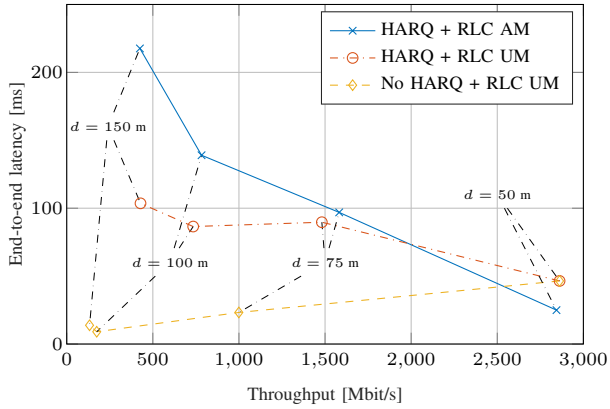


Figure 3: Latency throughput tradeoff for TCP CUBIC, with and without the different retransmission mechanisms of the mmWave protocol stack.

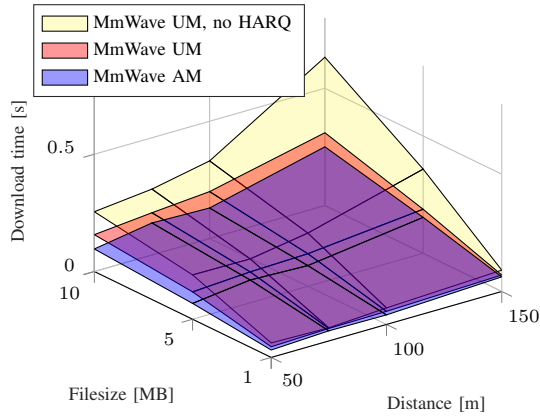


Figure 4: Download time as a function of the file size and of the distance, for TCP CUBIC with and without lower-layer retransmissions.

(from 1 MB to 10 MB) using wget (the file is hosted in the UE and retrieved by the remote server, in order to be consistent with the previous uplink simulations). The results show that lower-layer retransmission mechanisms help decrease the download time, and that the performance gain increases as the distance and the file size increase. Moreover, the difference between the download times with RLC AM and with RLC UM (no retransmissions) is more noticeable than that between the throughput values of Fig. 1, showing that for short-lived TCP sessions it is important to perform retransmissions as fast as possible, i.e., at a layer as close to the radio link as possible.

These results are well known when applied to traditional LTE networks [15], but these are the first simulations that show how much TCP depends on lower-layer retransmissions in mmWave networks, using the real Linux TCP/IP implementation. They show that, also in mmWave networks, the support of lower-layer retransmission mechanisms is fundamental for reaching a high TCP throughput even at large distances between transmitter and receiver, at the price of additional latency. In particular, in the simulated scenario the most effective retransmission scheme is HARQ at the MAC layer, since it provides the greatest throughput gain, but also the acknowledged mode of the RLC layer helps improve the performance of the mmWave link by reducing the download time for short-lived TCP sessions.

#### IV. MULTIPATH TCP

Multipath TCP (MP-TCP) has been proposed as a way of allowing vertical and seamless handovers between cellular networks and Wi-Fi hotspots and is currently under discussion for standardization at the IETF. It may also be used to provide path diversity in mmWave cellular networks. The three main design goals of MP-TCP are [16]:

- 1) Improve throughput: an MP-TCP flow should perform at least as well as a traditional single path TCP (SP-TCP) flow on the best path available.
- 2) On shared links, MP-TCP should not get more resources than standard TCP flows.
- 3) MP-TCP should prefer less congested paths, subject to the previous two conditions.

There are three RFCs that describe MP-TCP [16]–[18]. They discuss the signaling and setup procedures [17], the architectural choices for the deployment of MP-TCP [18], and a congestion control (CC) algorithm [16]. Finally the document in [19] discusses the impact on the application layer.

There are several studies that propose coupled congestion control algorithms for MP-TCP connections. By coupling over the different subflows, the authors of [16] claim that it is possible to reach goals 2 and 3 above. In particular they propose a first coupled CC, that is however criticized in [20] and in [21], because it (i) transmits too much traffic on congested paths and (ii) is unfriendly with respect to SP-TCP. Therefore two more coupled CC were proposed:

- In [20] the Opportunistic Linked Increases Algorithm (OLIA) is designed to overcome these two issues, but presents non-responsiveness problems with respect to congestion changes in the subflows;
- In [21] the Balanced Linked Adaptation algorithm (BALIA) addresses both the problems of the original CC and those of OLIA. In particular, the parameters of the protocol are derived through a theoretical analysis of the performance of multipath congestion control algorithms.

However, these schemes are based on the legacy design of Reno and New Reno congestion control algorithms (Additive Increase - Multiplicative Decrease, AIMD), which are shown to suffer from the highly dynamic behavior of mmWave links more than the newer TCP CUBIC congestion control algorithm [4].

MP-TCP could be used as an end-to-end solution for multi-connectivity, i.e., next generation mobile devices may connect both to an LTE and to a mmWave eNB, or to two or more mmWave eNBs with no need for coordination at the lower layers. However, there are some issues with its performance in mmWave networks, as we will show in the following paragraphs. In this performance evaluation campaign we used the real Linux implementation of MP-TCP (v0.90), which includes several CC algorithms, namely the original coupled CC, OLIA, BALIA, uncoupled (with any desired TCP flavor, e.g., CUBIC), and others. We co-deploy an LTE eNB and a mmWave eNB, or a mmWave eNB capable of transmissions at different frequencies (28 and 73 GHz, with the same bandwidth and the maximum number of antennas available in the ns-3 NYU simulator), and vary the distance of the multi-connected

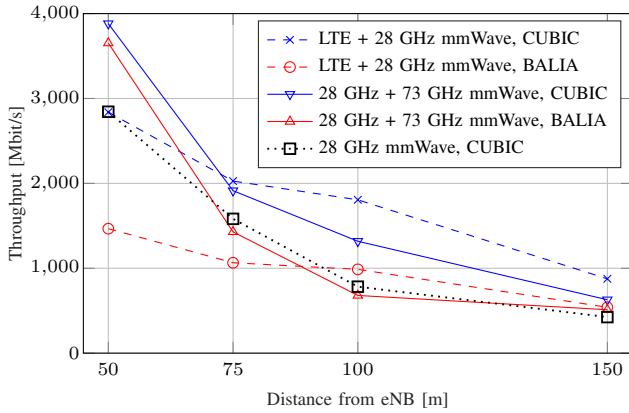


Figure 5: MP-TCP throughput for different distances  $d$  and different MP-TCP options. The black dotted line shows the performance of a SP-TCP connection with TCP CUBIC, as a reference.

UE from the eNBs, using the statistical channel model. The remote host is a multi-homed server, supporting MP-TCP connections. The UE uses IPERF, and starts the connection on the 28 GHz mmWave link. Then another subflow is added on the LTE link, or on the 73 GHz mmWave link.

Fig. 5 shows the performance in terms of throughput of different MP-TCP congestion control algorithms over different connections, with respect to the baseline of a SP-TCP connection with TCP CUBIC. The dashed lines represent a scenario with paths on LTE and on mmWave (28 GHz), while the solid ones refer to paths on mmWave links with 28 GHz and 73 GHz as carrier frequencies.

**LTE as mmWave secondary path:** When the UE is close to the eNB and has a LOS link most of the time on both the 28 and the 73 GHz connections (e.g., for  $d = 50$  m), then the solution with multipath TCP on mmWave-only links outperforms SP-TCP, with a gain that ranges from 800 Mbit/s (28%) to 1 Gbit/s (36%). Instead, due to the limit of the LTE uplink, the performance of a multipath on LTE and mmWave is close to that of SP-TCP (when CUBIC is used, because BALIA has much worse performance, as will be discussed later).

However, it can be seen from Fig. 5 that MP-TCP with LTE and mmWave links performs better than with only mmWave connections for  $d \geq 100$  m, and with the CUBIC uncoupled CC algorithm also for  $d = 75$  m. Indeed, the 73 GHz link offers a potentially larger throughput than an LTE uplink connection, but it has a lossy behavior that penalizes the overall throughput, except for small distances. In particular, for  $d = 150$  m, MP-TCP with LTE and 28 GHz mmWave offers a gain of more than 450 Mbit/s (i.e., 100%) with respect to the SP-TCP (i.e., more than the LTE uplink throughput), showing that the presence of the secondary and reliable LTE path improves the throughput on the mmWave link. This can be seen also in Fig. 6, where we plot the contribution of the two subflows of MP-TCP connections at  $d \in \{100, 150\}$  m when the second subflow is LTE or mmWave. It can be seen that the contribution given by the reliable LTE uplink subflow is smaller than that of the 73 GHz mmWave subflow, but the primary 28 GHz mmWave subflow reaches a higher throughput when coupled with the LTE secondary subflow.

For short-lived TCP sessions, instead, using a secondary

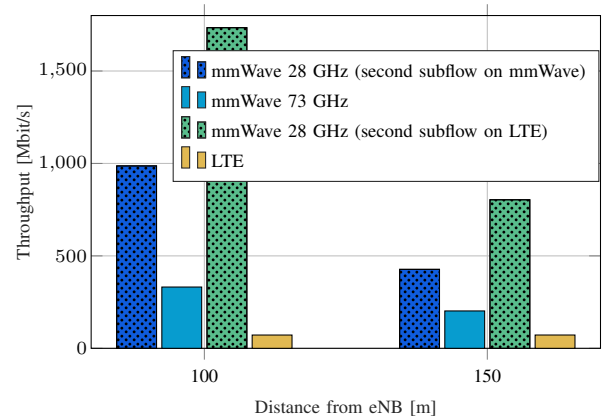


Figure 6: Contribution of the two subflows as a function of the distance  $d$ , for MP-TCP with CUBIC CC.

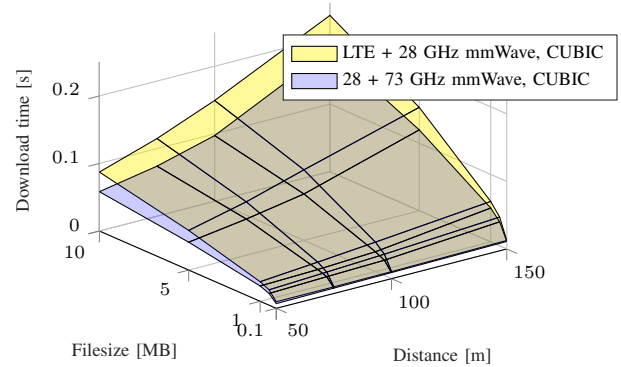


Figure 7: Download time as a function of the file size and of the distance, for MP-TCP with CUBIC CC and secondary subflow on LTE or mmWave at 73 GHz.

subflow on mmWave links improves the system performance. This can be seen in Fig. 7, which shows the download time of a file using wget with the same setup described in Sec. III. However, the performance gain, especially for smaller files, is minimal, showing that the LTE link makes up for its smaller capacity with a higher reliability that benefits the performance of TCP.

**Coupled vs uncoupled CC:** Another important observation is that MP-TCP with the BALIA CC algorithm fails to meet target 1, since in many cases its throughput is lower than that of SP-TCP, as shown in Fig. 5. The most striking cases are those with MP-TCP on LTE and mmWave, and  $d \in \{50, 75\}$  m. Here the congestion control algorithm sees the losses on the 28 GHz mmWave link as congestion, and, according to design goal 3, it steers the whole traffic to the LTE subflow, degrading the performance of the end-to-end connection. Instead, the uncoupled congestion control algorithm is not affected by this issue, since each path behaves independently. However, in this case design goal 2 is not met. An example of this behavior is shown in Fig. 8, where we compare the throughput over time of two different AIMD coupled CC algorithms (OLIA and BALIA) and of an uncoupled CC algorithm with CUBIC. It can be seen that at time  $t = 7$  s both OLIA and BALIA start using only the LTE connection, and that the throughput of the mmWave subflow goes to zero. A similar behavior for OLIA was observed in [21].

When considering short-lived TCP sessions and file down-

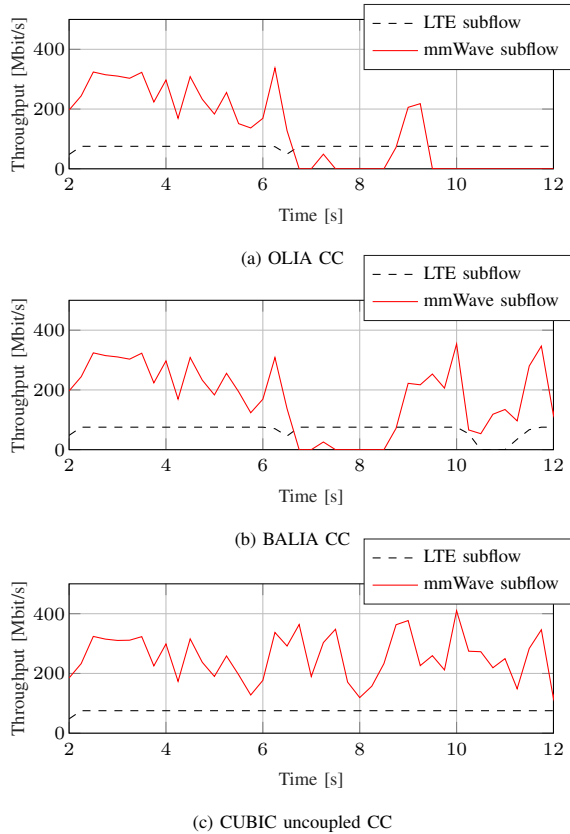


Figure 8: Throughput over time for the two subflows, for different MP-TCP CC algorithms.

load times, there are two different outcomes according to the file size. As shown in Fig. 9a, when the file is smaller than 1 MB the BALIA coupled congestion control algorithm exhibits a slightly smaller download time than the CUBIC uncoupled CC. Instead, when the file is larger than 5 MB, as in Fig. 9b, the MP-TCP solution with CUBIC as CC mechanism manages to download the file in less than a fifth of the time required by MP-TCP with BALIA. This behavior can be explained by considering the shape of the window growth function of CUBIC, which recalls a cubic function, i.e., flat at the beginning and then rapidly increasing.

The main conclusions from this performance analysis of MP-TCP for mmWave networks are that at larger distances and for long-lived TCP sessions it is preferable to use a more stable LTE-like link, and that the deployment of MP-TCP coupled congestion control algorithms on mmWave links is not able to satisfy the original design goals of [16]. A possible improvement of MP-TCP CC algorithms should adapt the TCP CUBIC scheme to a coupled scenario, so that the reactivity and stability of TCP CUBIC enhance the performance of the transport protocol while not harming other legacy TCP flows.

## V. MULTI-CONNECTED UE WITH ACKS ON LTE AND DATA ON MMWAVE

In this section, we test the performance of another kind of multipath deployment. The UE receives downlink data from the eNB on a mmWave connection, and sends the TCP ACKs either on LTE or on the same mmWave link. We consider a mobility scenario, with the eNB at coordinates (-1, 20) m and the UE moving from (151, 0) m to (151, 40) m at speed  $s \in$

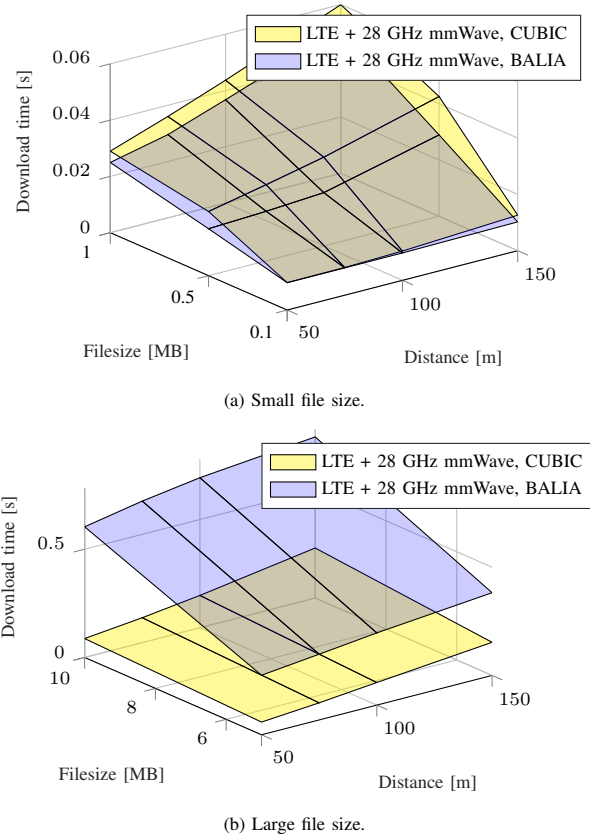


Figure 9: Download time as a function of the file size and of the distance, for MP-TCP with secondary subflow on LTE and CUBIC or BALIA CC.

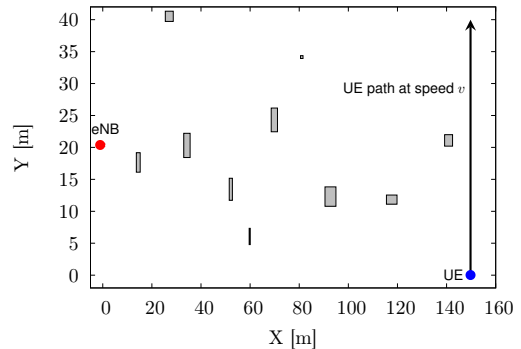


Figure 10: Random realization of the simulation scenario. The grey rectangles are 10 randomly deployed non-overlapping obstacles (e.g., cars, buildings, people, trees).

{2, 5} m/s. 10 small obstacles are deployed randomly in the area between the eNB and the UE, and the simulations exploit an ns-3 channel model that uses real traces to model the LOS to NLOS (or vice versa) transitions [12]. An example of random deployment is shown in Fig. 10.

The results of these simulations are shown in Fig. 11 for different RLC buffer sizes (2, 20 MB) and UE speed  $s$  (2, 5 m/s). It can be seen that with a larger buffer the throughput is slightly higher when sending the ACKs on the LTE connection, while with the smaller one it is preferable to use the mmWave connection, but the throughput is 100 Mbit/s lower. Indeed, when the buffer is small there is a need for more timely updates of the TCP congestion window, since there are more chances to cause buffer overflow and lose packets. If LTE is used, the latency of the ACKs increases, therefore the timeliness



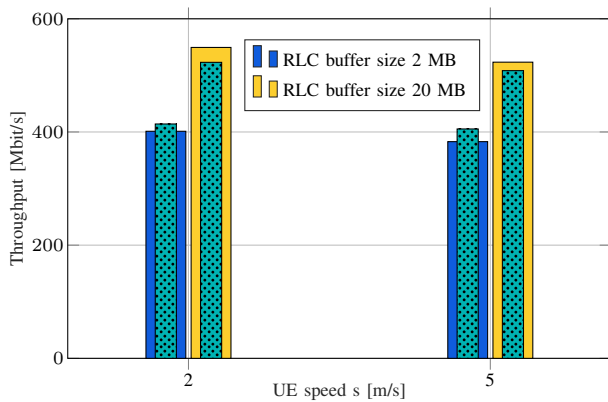


Figure 11: TCP throughput for different ACK reporting links. The narrow dotted bars refer to the setup with ACKs on mmWave links, while the large ones to that with ACKs on the LTE link.

of congestion control is reduced. However, the difference in throughput between the two solutions is small. Instead, with a larger buffer it is possible to queue more packets, and the transport layer is less sensitive to the latency in reporting the ACKs. In this case it is better to receive the ACKs on a more reliable LTE connection. However, notice that when the ACKs are sent on LTE the RTT increases, thus it takes more time to fill the capacity of the LOS link. This explains why also for the large buffer case the difference in performance between the system with ACKs on LTE and that with ACKs on mmWave is minimal. The main difference is thus on the choice of the buffer size: an undersized buffer degrades the throughput up to 27%.

## VI. CONCLUSIONS

The large bandwidth available at mmWave frequencies could allow a link capacity of the order of gigabits per second. However, the interaction with legacy transport protocols could prevent the full exploitation of the rate potentially available in mmWave communications. In this paper, we presented the first comprehensive performance evaluation of the interaction of Single Path and Multi Path TCP with mmWave links, with and without lower-layer retransmission mechanisms, using (i) a simulator with a channel model based on real measurements and a complete 3GPP-like cellular stack, and (ii) the actual TCP implementation of Linux. We firstly remarked that for mmWave it is very important to mask the channel losses to the higher TCP layer with link retransmission mechanisms, otherwise it is not possible to reach high throughput. Secondly, we studied the behavior of MP-TCP on 28 GHz mmWave links with LTE or 73 GHz mmWave links as secondary subflows. We showed that when the mmWave link has a high probability of being in NLOS state, a secondary LTE subflow improves the throughput performance of long-lived TCP sessions more than a mmWave subflow, and that the design goals of MP-TCP may not be met with mmWave links. Finally, we evaluated whether or not using LTE as the uplink connection for TCP ACKs helps improve the throughput, and showed that there is not a clear gain, because of the additional latency introduced by the LTE radio link.

As part of our future work, we will study how it is possible to exploit connections over multiple paths and lower-layer retransmission mechanisms to reach a high throughput on

mmWave links, while trying to reduce the additional delay introduced by retransmissions to meet the 1 ms latency 5G design goal. For example, we will extend our study to account for multiple mmWave eNBs deployed in different locations, so that the UE can use more than two MP-TCP subflows.

## REFERENCES

- [1] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, February 2014.
- [2] S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter-wave cellular wireless networks: Potentials and challenges," *Proc. IEEE*, vol. 102, no. 3, pp. 366–385, Mar. 2014.
- [3] H. Shokri-Ghadikolaei, C. Fischione, G. Fodor, P. Popovski, and M. Zorzi, "Millimeter wave cellular networks: A MAC layer perspective," *IEEE Transactions on Communications*, vol. 63, no. 10, pp. 3437–3458, Oct 2015.
- [4] M. Zhang, M. Mezzavilla, R. Ford, S. Rangan, S. S. Panwar, E. Mellios, D. Kong, A. R. Nix, and M. Zorzi, "Transport Layer Performance in 5G mmWave Cellular," in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Apr. 2016, pp. 730–735.
- [5] M. Mezzavilla, S. Dutta, M. Zhang, M. R. Akdeniz, and S. Rangan, "5G MmWave Module for the ns-3 Network Simulator," in *Proceedings of the 18th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Nov 2015, pp. 283–290.
- [6] R. Ford, M. Zhang, S. Dutta, M. Mezzavilla, S. Rangan, and M. Zorzi, "A Framework for End-to-End Evaluation of 5G mmWave Cellular Networks in ns-3," in *Proceedings of the Workshop on Ns-3*, June 2016, pp. 85–92.
- [7] M. Polese, M. Mezzavilla, and M. Zorzi, "Performance Comparison of Dual Connectivity and Hard Handover for LTE-5G Tight Integration," in *Proceedings of the 9th EAI International Conference on Simulation Tools and Techniques*, 2016, pp. 118–123.
- [8] C. Paasch, S. Barre, and al., "Multipath TCP in the Linux Kernel," available at <http://www.multipath-tcp.org>.
- [9] H. Tazaki, F. Uarbani, E. Mancini, M. Lagace, D. Camara, T. Turletti, and W. Dabbous, "Direct Code Execution: Revisiting Library OS Architecture for Reproducible Network Experiments," in *Proceedings of the Ninth ACM Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT '13, 2013, pp. 217–228.
- [10] "Iperf 2.0," available at <https://iperf.fr>.
- [11] M. Akdeniz, Y. Liu, M. Samimi, S. Sun, S. Rangan, T. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE Journal on Sel. Areas in Commun.*, vol. 32, no. 6, pp. 1164–1179, June 2014.
- [12] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, "Improved Handover Through Dual Connectivity in 5G mmWave Mobile Networks," *Submitted to IEEE JSAC Special Issue on Millimeter Wave Communications for Future Mobile Networks*, Nov. 2016. [Online]. Available: <https://arxiv.org/abs/1611.04748>
- [13] S. Choi and K. G. Shin, "A class of adaptive hybrid ARQ schemes for wireless links," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 3, pp. 777–790, May 2001.
- [14] Korea Telecom (KT) 5G-SIG, "5G.322, KT 5th Generation Radio Access, Radio Link Control protocol (5G pre-specifications)," 2016.
- [15] R. Li, M. Shariat, and M. Nekovee, "Transport protocols behaviour study in evolving mobile networks," in *2016 International Symposium on Wireless Communication Systems (ISWCS)*, Sept 2016, pp. 456–460.
- [16] C. Raiciu, M. Handley, and D. Wischik, "Coupled congestion control for multipath transport protocols," RFC 6356, 2011.
- [17] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses," RFC 6824, 2013.
- [18] A. Ford, C. Raiciu, M. Handley, S. Barre, and J. Iyengar, "Architectural Guidelines for Multipath TCP Development," RFC 6182, 2011.
- [19] M. Scharf and A. Ford, "MPTCP Application Interface Considerations," RFC 6897, 2012.
- [20] R. Khalili, N. Gast, M. Popovic, and J. Y. L. Boudec, "MPTCP is Not Pareto-Optimal: Performance Issues and a Possible Solution," *IEEE/ACM Transactions on Networking*, vol. 21, no. 5, pp. 1651–1665, Oct 2013.
- [21] Q. Peng, A. Walid, J. Hwang, and S. H. Low, "Multipath TCP: Analysis, Design, and Implementation," *IEEE/ACM Transactions on Networking*, vol. 24, no. 1, pp. 596–609, Feb 2016.