Experimental Study of the Effects of RLC Modes for 5G-NTN applications using OpenAirInterface5G

Sumit Kumar, Chandan Kumar Sheemar, Jorge Querol, Amirhossein Nik, Symeon Chatzinotas
Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Luxembourg
Email:{sumit.kumar, chandan.kumar.sheemar jorge.querol, amirhossein.nik, symeon.chatzinotas}@uni.lu

***Abstract*—Provisioning of 5G services via Non-Terrestrial Networks (5G-NTN) has become a reality. Currently, 5G-NTN is being developed by adapting the 5G Terrestrial Network (5G-TN) protocol which presents several challenges as the satellite channel is significantly different from terrestrial channels. The retransmission mechanism is one of them. Recent works on 5G-NTN have followed the approach of disabling retransmissions from the 5G protocol stack such as Hybrid Automatic Repeat Request (HARQ) at the MAC layer and Automatic Repeat Request (ARQ) at the RLC layer while delegating it to the application layer. This approach degrades the end-user-throughput and latency. In this work, we experimentally study the effect of enabling ARQ retransmission, by using RLC Acknowledged Mode (RLC-AM) in a GEO-based 5G-NTN which is susceptible to packet loss caused by low SNR. We conduct real-time experiments to compare the effects of RLC-AM and RLC Unacknowledged Mode(RLC-UM) on applications(VoIP, video stream, file transfer) using TCP and UDP for different SNR regimes. We have used OpenAirInterface5G-NTN, which is developed to perform real-time 5G-NTN experiments, and a satellite channel emulator. We observe that for low SNR, RLC-AM performs better than RLC-UM in achieving the required bitrate and packet error rate. The reason being RLC-AM recovers the lost packets earlier in the protocol stack without delegating the retransmissions to the application layer. This becomes especially useful when HARQ is disabled. We believe that our experimental study will complement the ongoing theoretical research and help improve the procedures of the Radio Resource Control(RRC) layer in 5G-NTN-specific applications.**

***Index Terms*—5G-NR, Non-Terrestrial Networks, OpenAirInterface**

I. Introduction

Providing 5G services via satellites, also termed as 5G-NTN has gained significant traction, especially after the inclusion of satellite components in the 3GPP Release-17 onwards [1]; with more new features being included in Release-18; and several improvements planned for Release-19. Protocol for 5G-NTN is being developed by adapting the existing 5G Terrestrial Networks (5G-TN) protocol stack for the sake of interoperability and the lowest impact on the existing 5G-TN system setup. One of the major constraints while developing the protocol stack for 5G-NTN is the retransmission of lost packets (in 5G terminology packets are called Protocol Data Unit: PDU) from the transmitter. Compared to 5G-TN, 5G-NTN is more susceptible to the loss of PDUs at the receiver entity due to very low SNR (for Geostationary Earth Orbit satellite payload)[REF] or significantly high Doppler (for Low Earth Orbit satellite payload)[REF]. Applications requiring reliable links such as file transfer, web browsing, etc cannot afford the loss of PDUs while applications such as voice calls and video calls can afford PDU loss to a certain extent however the end user experience gets degraded. Thus, retransmission techniques from the transmission entity are required to recover the lost PDUs. In 5G-TN, retransmission is done at several layers starting from Hybrid Automatic Repeat Request (HARQ) at the MAC layer and Automatic Repeat Request (ARQ) at RLC layer respectively [2]. Besides, the application layers protocols such as TCP also perform retransmission if loss of PDU is detected. An illustration is shown in Figure-1.

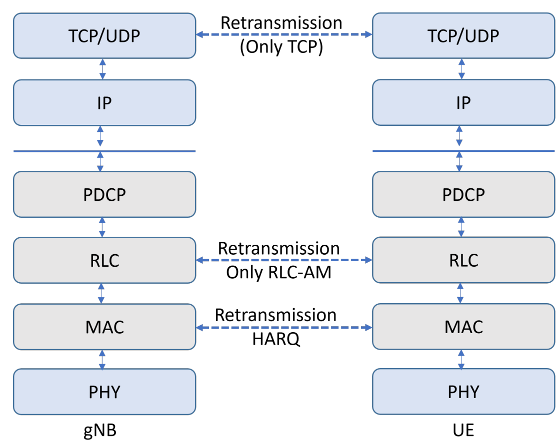

Fig. 1: 5G NR Protocol Stack. Retransmission mechanisms are available from lower layers till higher layers.

Contemplating retransmissions, as we go higher in the protocol stack: MAC $\rightarrow$ RLC $\rightarrow$ Application-Layer, the re-transmission gets more costly in terms of throughput degradation. Hence, it is desirable to perform the re-transmission of lost PDUs as early (or as lower) in the protocol stack. However, due to the challenges posed by satellite channels, especially large distance in GEO-based 5G-NTN, HARQ re-transmissions from the MAC layer becomes impractical [3], [4]. Nonetheless, the ARQ retransmission mechanism from the

RLC layer, which is not as frequent as HARQ, could be still feasible before the retransmission is delegated to the application layer. Previous works on prototype development for 5G-NTN have followed the approach of disabling the re-transmissions from the MAC and RLC layer while completely dependent on application layer retransmissions for lost PDUs or attempting to reduce the number of lost packets by using lower Modulation and Coding Schemes (MCS) - both the approaches result in decreased end-to-end throughput. In this work, we have taken an experimental approach to study the effects of retransmission from the RLC layer on several common applications such as VoIP, video streaming, and file transfer which use application layer protocols such as UDP and TCP. We have used OpenAirInterface5G 5G-NTN suite [5] for the implementation of Software Defined Radio (SDR) transparent payload gNB and UE (see Figure-2). To emulate the channel effects, we have used an in-house developed FPGA-based satellite channel emulator. During the experiments, two different modes of RLC: RLC-AM and RLC-UM are used while at the application layer, TCP and UDP-based traffic corresponding to VoIP, video streaming, and file transfer has been injected into the network. Through this systematic experimental study, we have proved that in the absence of HARQ, a 5G-NTN link can achieve usable throughput and reliable link using ARQ retransmissions from RLC and do not necessarily depend on the application layer completely for the retransmission for lost packets. To the best of our knowledge, experimental evaluation of the effect of re-transmissions RLC has not been done for 5G-NTN over live experiments. Though the expectation of throughput from 5G-NTN is not as high as 5G-TN, we believe that the findings from our experimental study will help improve the ongoing 5G-NTN protocol development.

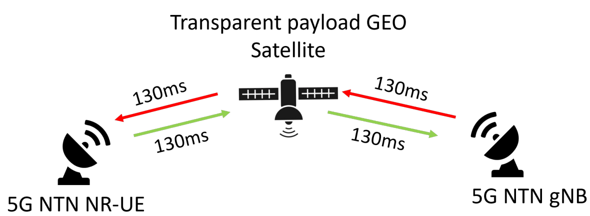

Fig. 2: An illustration of Transparent-Payload GEO based 5G-NTN. The RTT in this set-up is 520ms.

II. Previous Works

Previous works on the adaptation and development of the 5G NTN protocol stack have mainly considered the physical and mac layer with the primary objectives being (a) Establishing the link between gNB and UE via transparent payload satellites (b) Fine-tuning to achieve the best effort throughput under the given channel conditions. Some of the notable projects which have done adaptations in the Terrestrial 5G protocol stack to make it suitable for 5G-NTN include (a) 5G-ALL STAR: 5G AgiLe and flexible integration of SaTellite And cellulaR (5G-ALLSTAR) [6] (b) 5G-EMUSAT: 5G New Radio EMUlation over SATellite(5G-EmuSat) [7]. (c) 5G-GOA: The 5G-Enabled Ground Segment Technologies Over-The-Air Demonstrator [8] (d) 5G-LEO: OpenAirInterface extension for 5G satellite links [9]. 5G-ALLSTAR and 5G-EMUSAT focused mainly on the PHY and MAC layer, and represent a precursor to lay the foundation for 5G-NTN prototype development using open-source SDR implementation. These two projects already implemented several 5G features necessary for integration with NTN. Later, 5G-GOA helped to further develop the stack and included adaptations on the higher layers (RLC, PDCP, RRC) as well as facilitating a Stand-Alone (SA) deployment of 5G-NTN. Besides, 5G-LEO considers LEO-based transparent payload satellites (the earlier three are GEO-based transparent payload satellites) and implements intra-gNB timer-based handover where the UE is handed over from on satellite gNB-DU to another satellite gNB-DU in course of the motion of satellites. In all the aforementioned projects, except 5G-LEO, HARQ at MAC as well as ARQ at RLC was disabled (RLC-UM was used) and the entire responsibility to retrieve the lost PDUs was left for transport layer protocols in case of TCP. While in the case of UDP protocols, such an approach leaves the system completely helpless as there is no re-transmission at all in case of PDU loss, neither from the 5G protocol stack as well as the transport layer. The loss of PDUs was compensated by limiting the Modulation and Coding Schemes (MCS) to the lowest, i.e., MCS-0 which corresponds to Quadrature Phase Shift Keying (QPSK). No analysis of the system throughput was evaluated with re-transmissions facilitated from any of the layers of the 5G protocol stack. Such evaluation is important to understand the limitations of re-transmissions done by the transport layer and 5G protocol stack operating over satellite channels.

III. Radio Link Control(RLC) Modes

RLC in 5G NR protocol stack interfaces PDCP from above via RLC channels and MAC from below via logical channels [2]. The RLC layer in 5G-NR is responsible for mainly (a) Transfer of upper layer PDUs in AM, UM, and TM (b) Error correction via retransmission of lost RLC PDUs (only in AM) using ARQ (c) Segmentation and reassembly of RLC SDUs (both in UM and AM). The RLC sublayer can have multiple RLC entities [1] Every RLC entity in gNB has a corresponding entity in UE and vice-versa. Whenever a radio bearer needs to be set up, depending on the type of service requirements, the RLC entities can be configured in one of the following three modes by the RRC(Radio Resource Control) layer.

- Transparent Mode (TM)
- Unacknowledged Mode (UM)
- Acknowledged Mode (AM)

[1]In software terminology, entity means a thread or a task.

The choice of modes depends on the service requirement. For example, file transfer requires error-free delivery while video streaming and voice calls can afford missing packets to a certain extent. An RLC-TM entity is the simplest. It doesn't add any header or segment to RLC Service Data Units(SDUs). An RLC Protocol Data Unit(PDU) is simply an RLC SDU. An RLC-UM entity segments RLC SDUs and adds a header to each RLC PDU on the transmit side. In the receiver size, the RLC-UM entity reassembles the segments and delivers a complete RLC SDU to PDCP while discarding the incomplete SDUs. RLC-UM supports segmentation and duplicate detection. The last RLC mode, RLC-AM, is similar to RLC-UM with two important distinctions (a)Acknowledgments and (b) Retransmission. The receiving RLC-AM entity sends feedback to the transmitting RLC-AM entity (using STATUS PDUs) to retransmit the PDUs that are not received correctly. If it detects a gap in the sequence of the received PDUs, it starts a reassembly timer (t-ReassemblyTimer) with the assumption that the missing PDU may still get retransmitted using the HARQ protocol at the MAC layer. When the t-ReassemblyTimer expires (usually in the case of HARQ failure):

- An RLC-UM receiver discards the PDU with missing segments and does not attempt to recover them. In this case, the application layer has delegated the burden to recover the packets, however, the application layer retransmissions are triggered very late and affect the throughput.
- A RLC-AM receiver sends a status message comprising the sequence number of the missing PDUs to the sender RLC-AM entity. The ARQ function of the RLC-AM sender entity then performs retransmissions based on the received status message. When the maximum ARQ retransmission count is reached, and the missing segments of the reported PDUs have still not arrived, the PDUs are discarded and the application layer delegates the burden to recover the packets, however, the application layer retransmissions are triggered very late, and affect the throughput. A pictorial illustration of the functionality of RLC-AM is shown in Fig-3.

NTN Adaptations in RLC

Similar to other protocol layers, the RLC too has to cope with the major challenge of extended RTT. For this purpose, the t-ReassemblyTimer (at both UE and gNB) in the RLC-UM mode is adapted as t-Reassembly = (2 x RTT + schedulingOffset) x number-of-HARQ-Retransmissions. For the GEO satellites based 5G-NTN, it has been suggested to disable the HARQ retransmissions at the physical layer [10] and reduce MCS to make the service more robust. For example, in some of the recent works involving live Over-The-Satellite (OTS) 5G-NTN tests, the maximum MCS used was MCS9 [8], [11]. Once, the HARQ is disabled at the physical layer, RLC (only if configured as RLC-AM) has the burden of recovering the

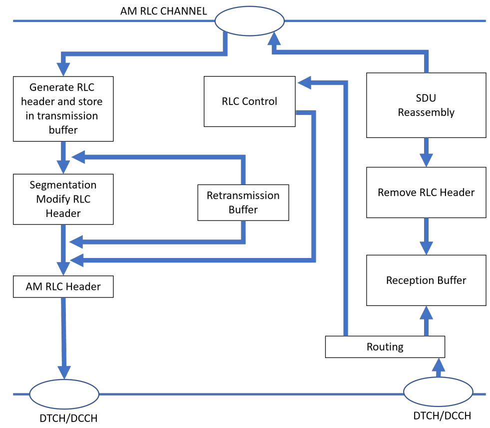

Fig. 3: An illustration of the functionality of RLC-AM in 5G-NR protocol stack [2]

missing packets as the error is propagated from the physical layer to the RLC layer in the absence of HARQ. Besides, ARQ requires that the transmitted packets be buffered in anticipation of potential packet loss and released only after the successful receipt of an acknowledgment, or until a time-out mechanism reinitiates a retransmission. The long RTT requires a larger transmission buffer and potentially limits the number of retransmission allowed for each transmitted packet in both the uplink and downlink. For efficient ARQ operation in GEO networks, UE and gNB must size their transmission buffer and the t-ReassemblyTimer according to the longest RTT to be anticipated. In addition, the maximum retransmission count also requires to be adjusted [4], [12].

IV. OAI 5G-NTN Testbed

In this section, we detail the software and hardware tools used to build the testbed for our experimental study.

Software: OpenAirInterface5G NTN

OpenAirInterface™ (OAI) [13] is an open-source SDR-based implementation of 3GPP compliant 5G gNB, UE and the core network (5G-CN). Using OAI it is possible to build a 4G/5G network quickly at low cost with COTS SDR and general-purpose x86 processors. OAI is well known in the terrestrial communication research regime, especially to applied researchers. Eventually, the capabilities and flexibility offered by OAI have also paved the way toward exploring new application/technology areas such as 5G-NTN. In the past few years, OAI has gained notable interest from the scientific community, industry, and funding agencies to develop it as a tool for 5G-NTN research and experimentation. Please note that all the projects which have been discussed in Section-II have used and contributed towards adaptations in OAI for 5G-NTN use cases. In these projects, OAI is used as the primary tool to develop simulators, in-lab demonstrators, and over-the-satellite testbeds by performing adaptations/suggestions as per 3GPP Release-17. The flavour of OAI which has been used in our work has been developed during the projects 5G-GOA and 5G-LEO. The source code of OAI 5G-NTN can be downloaded and used free of charge from here [5]. [14] lists all the details of adaptations done in OAI for transparent payload-based 5G-NTN.

Besides, we have used iperf3 [15], a popular tool to generate traffic for our experiments and measure the network performance. Iperf3 has client and server functionality and can create data streams corresponding to several QoS(Quality of service).

Hardware: USRP and Channel Emulator

On the hardware side, we have used COTS SDR from Ettus Research, USRP-X310 [16] for gNB and UE which are connected to Intel-i9-based Laptops. The daughter boards in USRP-X310 span the frequency from DC to 6GHz and up to 160MHz of base-band bandwidth. For the channel emulator, we have used an in-house developed FPGA-based (Zynq UltraScale+ RFSoC ZCU111) satellite channel emulator which is capable of providing one-way delay up to 260ms(hence 520ms Round Trip Time) thus mimicking the latency caused by transparent payload based 5G-NTN [17]. Additionally, the channel emulator can add Additive White Gaussian Noise (AWGN) to the uplink and downlink channels, thus, creating a wide range of SNR regimes. Figure-4 shows the schematic of the testbed.

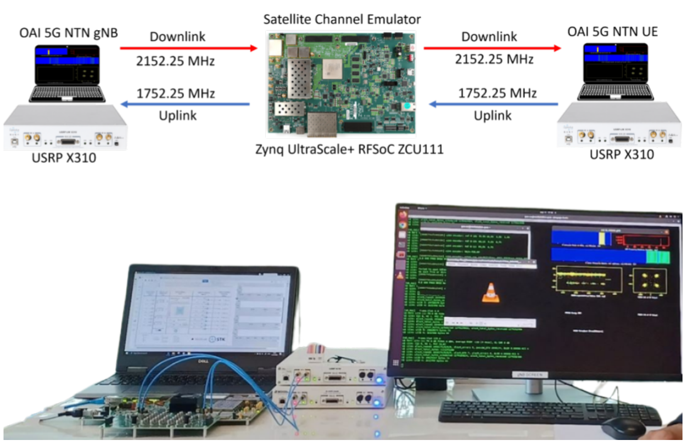

Fig. 4: Testbed set-up for the 5G-NTN experiments. Ettus X310 SDR is used for gNB and UE, FPGA based channel emulator mimics the transparent payload based GEO satellite, OpenAirInterface5G-NTN has been used as the 5G-NTN software protocol stack.

V. Experiments and Results

We have chosen a scenario of transparent payload-based GEO 5G-NTN [4]. In such a configuration, the satellite acts as a reflector, .i.e, the satellite simply reflects the downlink signal coming from gNB towards UE while the uplink signal coming from UE towards the gNB. The schematic of our testbed as shown in Fig-4 replicates this scenario. The major challenges of this scenario are large RTT (approx 520ms) and low SNR. Due to implementation simplicity, we have used only one UE in our experiments. The rest of the experiment parameters are listed in Table-I.

TABLE I: Experiment parameters

Parameters	**Values**
Payload type	Transparent
Constellation type	GEO
Uplink/Downlink freq(MHz)	1752.25/2152.25
RTT(ms)	520
MCS(AMC disabled)	0,4,9
HARQ	Disabled
OAI test mode	PHY Test
Link direction	Downlink
Subcarrier Spacing(KHz)	15
Number of Downlink Slots	10

Experiment-1: VoIP test

In this experiment, we test to compare the performance of RLC-AM over RLC-UM for a VoIP connection. VoIP uses UDP at the application layer. It requires 90 to 156 kbps throughput and less than 1% of packet error rate. Using iperf3, we emulate the VoIP traffic with a suitable ToS(Type of Service) flag. The experiment is performed at different SNR values. For each value of SNR, the iperf UDP traffic is generated for 50 seconds with a target bandwidth of 200kbps and target % PER $< 1\%$. Further, each experiment is repeated 5 times, and the average throughput and %PER loss are logged. We have used only MCS0 for this experiment as the bitrate requirement is low for VoIP compared to other applications.

Experiment-2: Video streaming test

In this experiment, we test to compare the performance of RLC-AM over RLC-UM for a video streaming connection. Most video streaming services use UDP at the application layer. Depending on the quality of the video, the required throughput varies. Nonetheless, a PER of less than 0.5% is desired. We tested a target bitrate of 0.5 Mbps which is the minimum requirement for live streaming [18]. We use MCS4 for live video streaming. Likewise experiment-2, in this experiment, for each value of SNR, the iperf UDP traffic is generated for 50 seconds with the mentioned target bitrate and target % PER $< 0.5\%$. Further, each experiment is repeated 5 times, and the average throughput and %PER loss are logged.

Experiment-3: File transfer test

In this experiment, we compare the reliability of RLC-AM vs RLC-UM. Reliability is a key parameter for file transfer applications such as web browsing. Hence a 0% packet error rate is required. Thus we used TCP for this experiment. In this experiment, we attempt to test the maximum bitrate which can be achieved without focusing on a particular target bitrate. For each value of SNR, the iperf TCP traffic is generated for 50 seconds and each experiment is repeated 5 times. Finally, the average throughput and number of application layer retransmissions are logged.

RESULTS

Experiment-1

The results for experiment-1 are listed in Table-II. We observe that by using RLC-AM, the target bitrate and PER can be achieved for lower SNR compared to RLC-UM. The reason being RLC-AM attempts to recover the lost packets using ARQ retransmissions. While RLC-UM does not attempts to do so. This is detrimental, especially in the case of UDP where the application layer also does not perform retransmission. Note that among the application layer protocols TCP and UDP, only TCP performs retransmission of lost packets.

TABLE II: Percentage of lost packets in RLC-UM vs RLC-AM for VoIP using MCS0 and target bitrate = 200kbps, target PER<1%

SNR(dB)	%Lost Packet in RLC-UM	%Lost Packet in RLC-AM
4.51	0	0
4.17	0	0
4.08	0.025	0.019
4.03	0.068	0.025
4	0.154	0.047
3.91	0.608	0.139
3.82	1.2	0.712
3.77	Unusable	0.864
3.74	Unusable	1.038

Experiment-2

The results for experiment-2 are listed in Table-III. We observe that by using RLC-AM, the target bitrate and PER can be achieved for lower SNR compared to RLC-UM. The reason being RLC-AM attempts to recover the lost packets using ARQ retransmissions. While RLC-UM does not attempts to do so. This is detrimental, especially in the case of UDP where the application layer also does not perform retransmission.

Experiment-3

The result of experiment-3 is shown in Fig-5. We observe that the observed TCP throughput using RLC-AM significantly outperforms when using RLC-UM at lower SNR. However, at higher SNR the effect is not seen, In fact, there is a slight degradation in RLC-AM TCP throughput due to the overhead of acknowledgment packets.

We also list the average number of TCP retransmissions from the application layer while using RLC-AM and RLC-UM in Table-IV. We observe that at low SNR, the number of application layer retransmissions from TCP is significantly lower in RLC-AM compared to RLC-UM. Lower re-transmisisons from

TABLE III: Percentage of lost packets in RLC-UM vs RLC-AM for live video-streaming using MCS4 and target bitrate = 500kbps, target PER<0.5%

SNR(dB)	%Lost Packet in RLC-UM	%Lost Packet in RLC-AM
4.51	0	0
4.34	0	0
4.17	0	0
4	0.044	0.008
3.98	0.053	0.022
3.96	0.085	0.024
3.93	0.136	0.032
3.91	0.664	0.174
3.86	Unusable	0.354
3.81	Unusable	0.572

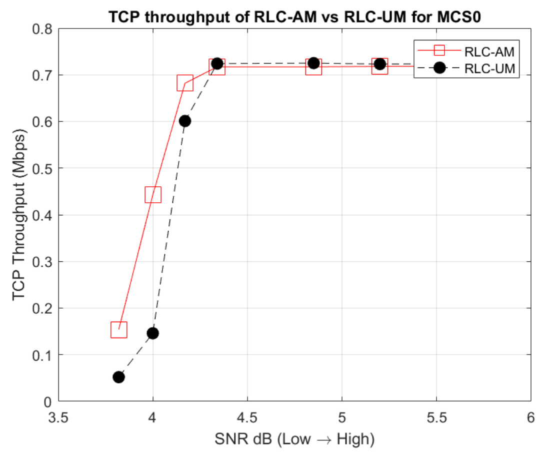

Fig. 5: TCP througput using RLC-AM vs RLC-UM. We observe that at low SNR, RLC-AM provides better throughput, however at higher SNR, the throughput of RLC-AM is slightly lower compared to RLC-UM, the reaon being overhead of acknowledgement packets used in RLC-AM

application layer while using RLC-AM compared to RLC-UM also supports our observation in Fig-5.

VI. CONCLUSION

Early retransmission of lost packets in the 5G-NTN protocol is desirable. However, due to implementation complexity, HARQ retransmissions from the MAC layer are not possible. Nonetheless, the ARQ retransmission mechanism from the RLC layer can still provide better results compared to late retransmissions from the application layer protocols. In this work, by performing exhaustive RF experiments, we have shown that in a GEO-based 5G-NTN, using RLC-AM can provide a better user experience compared to RLC-UM for common applications such as VoIP, live video streaming, and file transfer. We believe that our work will complement the ongoing applied research on 5G-NTN and help improve the

TABLE IV: Average number of TCP retransmissions in RLC-AM vs RLC-UM. We observe that at low SNR, the number of application layer retransmissions from TCP is significantly lower in RLC-AM compared to RLC-UM.

SNR(dB)	Average TCP Retransmissions in RLC-AM	Average TCP Retransmissions in RLC-UM
5.71	0	0
5.20	0	0
4.85	0	0
4.34	0	0
4.17	1	3.66
4	4	5.66
3.82	11	15

RRC functionality for 5G-NTN use cases. Although our experiments concern transparent payload, it is equally significant for regenerative payload as well. Currently, we are working on optimizing various timers which are being used in the RLC and PDCP layer of the 5G-NTN protocol stack.

ACKNOWLEDGMENT

This work has been supported by the project 5G-LEO under the ESA ARTES program with contract no. 4000135152/21/NL/FGL. We acknowledge our consortium Eurescom GMBH, Fraunhofer IIS, and AllBeSmart LDA.

REFERENCES

[1] 3GPP, 2023. [Online]. Available: https://www.3gpp.org/news-events/3gpp-news/nr-ntn

[2] "S. sirotkin (editor), "5g radio access network architecture"," 2020.

[3] S. Cioni, X. Lin, B. Chamaillard, M. El Jaafari, G. Charbit, and L. Raschkowski, "Physical layer enhancements in 5g-nr for direct access via satellite systems," *International Journal of Satellite Communications and Networking*, vol. 41, no. 3, pp. 262–275, 2023.

[4] Q. Ye, C. Lo, J. Jeon, C. Tarver, M. Tonnemacher, J. Yeo, J. Cho, G. Xu, Y. Kim, and J. Zhang, "5g new radio and non-terrestrial networks: Reaching new heights," in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2022, pp. 538–543.

[5] "Openairinterface project, online," 2023. [Online]. Available: https://gitlab.eurecom.fr/oai/openairinterface5g/-/tree/goa-5g-ntn

[6] J. Kim, G. Casati, A. Pietrabissa, A. Giuseppi, E. C. Strinati, N. Cassiau, G. Noh, H. Chung, I. Kim, M. Thary *et al.*, "5G-ALLSTAR: An integrated satellite-cellular system for 5g and beyond," in *IEEE WCNCW*. IEEE, 2020, pp. 1–6.

[7] "5G-EMUSAT:5G New Radio Emulation Over Satellite, online," 2022. [Online]. Available: https://5gmeteors.eurescom.eu/open-calls/1st-open-call-summary/5g-emusat/

[8] "5G-GOA: 5G-Enabled Ground Segment Technologies Over-The-Air Demonstrator, online," 2022. [Online]. Available: https://wwwfr.uni.lu/snt/research/sigcom/projects/5g_goa

[9] "5G-LEO - OpenAirInterface™ extension for 5G satellite links," https://artes.esa.int/projects/5gleo, accessed: 2022-7-31.

[10] "3GPP TR 38.821, "3rd generation partnership project; technical specification group radio access network; solutions for nr to support non-terrestrial networks (ntn) (release 16)," 2021.

[11] F. Kaltenberger, T. Schlichter, T. Heyn, G. Casati, F. Völk, R. T. Schwarz, and A. Knopp, "Building a 5g non-terrestrial network using openairinterface open-source software," *EuCNC/6G Summit*, vol. 2021, 2021.

[12] S. Cioni, X. Lin, B. Chamaillard, M. El Jaafari, G. Charbit, and L. Raschkowski, "Physical layer enhancements in 5g-nr for direct access via satellite systems," *Int. J. Satell. Commun.*, 2022.

[13] "Openairinterface project, online," 2021. [Online]. Available: https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/home

[14] S. Kumar, A. K. Meshram, A. Astro, J. Querol, T. Schlichter, G. Casati, T. Heyn, F. Völk, R. T. Schwarz, A. Knopp *et al.*, "Openairinterface as a platform for 5g-ntn research and experimentation," in *IEEE FNWF*. IEEE, 2022, pp. 500–506.

[15] C. iPerf, "iperf—the ultimate speed test tool for tcp, udp and sctp," 2021.

[16] M. Ettus and M. Braun, "The universal software radio peripheral (usrp) family of low-cost sdrs," *Opportunistic spectrum sharing and white space access: The practical reality*, pp. 3–23, 2015.

[17] J. Querol, A. Abdalla, Z. Bokal, J. C. Merlano Duncan, M. Gholamian, O. Kodheli, J. Krivochiza, S. Kumar, C. Martinez Luna, N. Maturo *et al.*, "5G-SpaceLab," 2021.

[18] "Bandwidth for streaming: How much do you need?, online," 2021. [Online]. Available: https://www.epiphan.com/blog/bandwidth-for-streaming/