Sliding Window Network Coding Enables NeXt Generation URLLC Millimeter-Wave Networks

Eurico Dias, Duarte Raposo, Homa Esfahanizadeh, Alejandro Cohen,
Tânia Ferreira, Miguel Luís, Susana Sargento and Muriel Médard

***Abstract*—Ultra-reliability and low-latency are pivotal requirements of the emerging 6th generation of communication systems (xURLLC). The transition in millimeter-wave (mmWave) technology, from omni-directional to highly directional antennas, has been seen as an enabler for high bandwidth communications, still susceptible to high loss and high latency variation. Classical error recovery approaches cannot close the rising gap between high throughput and low delay in such systems. In this work, we incorporate effective sliding window network coding solutions in mmWave communications. While legacy systems such as rateless codes improve the delay, cross-layer results show that they do not provide Low Latency Communications (LLC), due to the lossy behaviour of mmWave channel and the lower-layers' retransmission mechanisms. On the other hand, fixed sliding window random linear network coding (RLNC) is able to achieve LLC, and even better, adaptive sliding window RLNC obtains Ultra-Reliable LLC (URLLC) in mmWave backhaul networks.**

I. Introduction

Millimeter-wave (mmWave) networks enable multi-gigabit-per-second data rates between 57 GHz and 64 GHz, the so-called V-Band, that uses the unlicensed spectrum available worldwide. It is an attractive option for Integrated and Access Backhaul (IAB), which is part for the new generation of communications – 6G – to reduce deployment expenses of fiber optics with the increase of connection density [1]. However, these frequency bands have been heretofore mostly idle because mmWave communications suffer from strong path loss and heavy propagation challenges with obstacles, rain and atmospheric absorption, making them only suitable for short and Line-of-Sight (LoS) communications.

The challenges of mmWave are particularly salient when we seek to use them, as would be the case in IAB, for the neXt generation Ultra-Reliable and Low-Latency Communications (xURLLC) in 6G services (e.g., tactile Internet, Virtual/Augmented Reality, and intelligent transportation). In-order delivery delay is the main target for both Low Latency Communications (LLC) and Ultra-Reliable and Low-Latency Communications (URLLC), and in addition, URLLC would require the bulk of the packets to be delivered in a timely fashion to their destinations, *i.e.*, with a failure probability of less than $1-10^{-5}$, and within a latency of 1 ms for 32 bytes and 3-10 ms for 300 bytes [2].

E. Dias, D. Raposo, T. Ferreira, M. Luís and S. Sargento are with Instituto de Telecomunicações, 3810-193 Aveiro, Portugal (e-mail:{eurico.omdias, dmgraposo, tania.s.ferreira, nmal}@av.it.pt, susana@ua.pt). E. Dias, T. Ferreira and S. Sargento are also with University of Aveiro, 3810-193 Aveiro, Portugal. M. Luís is also with Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa, 1959-007 Lisboa, Portugal. H. Esfahanizadeh and M. Médard are with EECS, MIT, Cambridge, MA 02139 USA (email:{medard, homaesf}@mit.edu). A. Cohen is with ECE, Technion, Israel (e-mail: alecohen@technion.ac.il).

This work is supported by the European Regional Development Fund (FEDER), through the Regional Operational Programme of Centre (CENTRO 2020) of the Portugal framework and FCT under the MIT Portugal Program [Project SNOB-5G with Nr. 045929 (CENTRO-01-0247-FEDER-045929), and partially by Northrop Grumman Corporation (NGC).

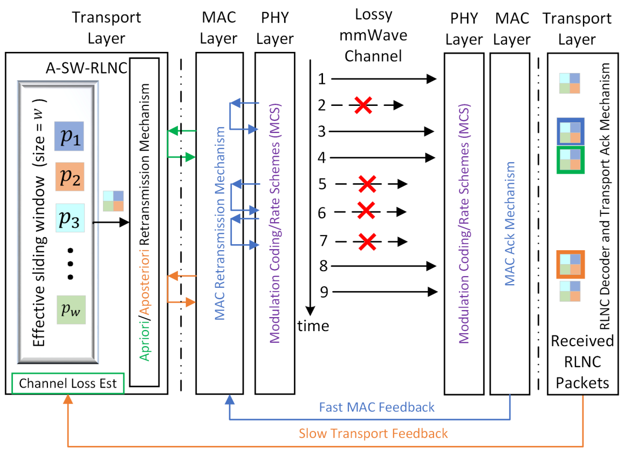

Fig. 1: Cross-layer sliding window network coding approach. Section II describes the system model. Section III presents different RLNC approaches, in particular, the A-SW-RLNC. Only involved layers in the protocol are illustrated here.

The lossy nature of mmWave introduces new challenges in MAC and transport layers, such as link quality assessment, rate adaptation and bufferbloat [3]. Several techniques have been used to correct failures in the wireless channels, e.g., rateless erasure codes [4], which were recently deployed by Verizon [5], systematic codes [6], and streaming codes [7]. In order to manage delay, transport protocols commonly use windowing schemes, such as TCP [8]. Combining windowing with coding can be done with Random Linear Network Coding (SW-RLNC) either in a fixed way (F-SW-RLNC) [9], in an adaptive way (A-SW-RLNC) [6], or in a causal variant [10].

The contributions of this paper are as follows: we look into the unique dynamic behaviour of mmWave communication environment and propose how to use SW-RLNC, both F-SW-RLNC and A-SW-RLNC schemes [10], [11], to capture rapid changes and mitigate the high losses that are intrinsic to mmWave, for LLC and URLLC. In particular, we study the interplay of cross-layer solutions as illustrated in Fig. 1, and show how to mitigate the large delay caused by the lower-layer error control mechanisms.

Our study shows that the combination of a priori Forward Error Correction (FEC) mechanism and a slow posteriori retransmission mechanism at the transport layer [10], [12] could be used to relax the conservative modulation and the coding rate requirement at the PHY layer [13], while keeping the fast retransmissions at the MAC layer [14]. This solution is according to the mmWave channel quality as represented to the sender, by passive handshaking via acknowledgements between MAC and transport layers. Our findings show that cross-layer approaches could be used by system designers to improve current technologies.

We evaluate the performance of our approach over a cityscale mmWave testbed deployed in the city of Aveiro, Portugal. Our approach demonstrates that communication protocols can notably take benefit from relaxing the PHY and MAC layer error control mechanisms and delegating the task to the transport layer using the proposed network coding solution.

II. System Model and Problem Formulation

We consider a real-time slotted mmWave communication with fast MAC feedback and slow transport feedback as illustrated in Fig. 1. In particular, a Single-Path (SP) communication setting is considered between two points, sender and receiver. We assume that the data that needs to be transmitted consists of N packets of the same size, *i.e.*, $\{P_1, \ldots, P_N\}$. At the i-th time step, the sender transmits a coded packet E_i over the noisy mmWave forward channel. The receiver may acknowledge the sender by sending an Acknowledgment (ACK) for any delivered coded packet over both noisy feedback channels, one for the MAC layer and the second for the transport layer. The distance between the first time data is transmitted and the time the corresponding feedback is received at each layer is called Round Trip Time (RTT). The transmission delay of coded packets t_d is the time it takes for the sender at each layer to transmit one packet (push the packet into the medium), and the propagation delay t_{prop} is the amount of time it takes for one packet to be received from the sender to the receiver and vice versa.

Due to the layered protocol stack mechanisms in mmWave communications [15], t_d is significantly faster in the MAC layer than in the transport layer. We assume that the size of the feedback acknowledgment is negligible, and that the propagation delay can vary for any transmitted coded packet according to the channel's condition. Hence, the RTT for each coded packet is equal to $t_d + 2t_{\text{prop}}(E_i)$, where $t_{\text{prop}}(E_i) \leq t_{\text{prop}}$. Let the timeout $t_{\text{o}} \geq 2t_{\text{prop}}$ denote an adaptive parameter the sender may choose to declare the packets that were not delivered at the receiver in each layer. That is, for any coded packet transmitted, if an ACK is not received at the sender after $t_d + t_{\text{o}}$ time slots, the sender at each layer declares a Negative Acknowledgment (NACK) for the corresponding packet.

Our main performance metrics are defined as follows:
(1) **Throughput η.** This is defined as the rate, in units of bits per time slot, at which the information is delivered at the receiver. In this paper, we focus on a normalized throughput, denoted by η, which corresponds to the total number of information bits delivered to the receiver divided by the total amount of bits transmitted by the sender.
(2) **In-order delivery delay of packets D.** This is the difference between the time slot in which an information packet is first transmitted at the sender and the time slot in which the packet is decoded, in order, by the receiver.

III. Joint Scheduling and Network Coding

This section elaborates on using RLNC as an error correction mechanism in the *transport layer* as illustrated in Fig. 1. This mechanism mitigates the rigid requirements of the physical layer error correction to provide reliable communications for the worst mmWave channel condition.

In classical RLNC schemes [16], each encoded packet E_i that is transmitted over the lossy communication is a random linear combination of the original uncoded packets, *i.e.*,

$$E_i = \sum_{j=1}^{N} \rho_{i,j} P_j, \tag{1}$$

where the coefficients $\{\rho_{i,j} : i \in \{1, 2, \ldots\}, j \in \{1, \ldots, N\}\}$ are drawn from a sufficiently large field, and N is the total number of original uncoded packets. When the coefficients are randomly sampled from a large field, the receiver can decode the original packets once N coded packets are received, for example using the Gaussian elimination technique.

Although classical RLNC schemes can achieve the desired communication rates in the realm of large N, it imposes a large latency to the system. This is because, to decode the first packet, at least N coded packets need to be received.

A. Rateless RLNC (R-RLNC)

In this variation, the sender's packets are split into non-overlapping blocks, called batches, each with size of n packets. The batches are encoded and transmitted in order. For each batch, the encoded packets are random linear combinations of the packets within the same batch, and the ratio of the number of original packers n and the number of encoded packets m is the rate of the scheme. In a well-designed scheme, the receiver is able to recover the whole batch per receipt of n out of m encoded packets. More precisely, let E_i^k be the k-th encoded packet of the i-th batch, where $k \in \{1, \ldots, m\}$, then

$$E_i^k = \sum_{j=1}^{n} \rho_{i,j}^k P_{(i-1)n+j}. \tag{2}$$

Here, it is assumed that the total number of packets is divisible by the block size. If not, one can easily use the zero-padding techniques. In this variation, the code designer in advance can try to manage the performance, in terms of throughput and latency trade-off, by choosing the size of n and m.

If the sender does not receive an acknowledgement showing that at least n packets are delivered, by the end of transmitting the m-th coded packet of a batch, it starts sending another m coded packets (with different coefficients) for the same batch. This process continues until the sender ensures that n coded packets are received at the destination. Therefore, this RLNC scheme is by-definition a rateless code [17], and we call this scheme R-RLNC.

Recently, there are new solutions in the literature where the size of the i-th uncoded batch $n(i)$ and the size of the i-th coded batch $m(i)$ can be time-variant and adapted based on the channel estimation [18], [19]. However, those solutions are only adaptive and reactive to the average packet loss probability. In mmWave communications, the channel conditions vary extremely fast; hence, although the above solutions are adaptive, one can pay in performance, as those solutions do not track the specific erasure pattern of each packet and batch.

B. Adaptive and Causal RLNC

This is an adaptive and causal variant (A-SW-RLNC method), as given in [10]. In this method, at a time slot, according to the cumulative transport feedback information, the sender at the transport layer can decide either to transmit a new coded linear combination, i.e., *new packet*, or repeat the last sent combination, *same packet*. Here, *same* and *new* refer to the raw information packets contained in the linear combination, such that sending the same linear combination means that the raw information packets are the same but with different random coefficients. Thus, using a sliding window mechanism, the i-th coded packet can be described as follows,

$$E_i = \sum_{j=w_{\min}}^{w_{\max}} \rho_{i,j} P_j, \tag{3}$$

where $w_{\min}$ corresponds to the oldest raw information packet that is not yet decoded, and $w_{\max}$ is incremented each time a new raw information packet is decided to be included in the linear combination by the sender.

The A-SW-RLNC solution tracks the channel conditions, and adjusts the retransmission rate at the transport layer based on the channel quality and the transport feedback acknowledgments. For the channel estimation, the behavior of the channel parameters (*i.e.* erasure probability and its variance) is tracked using the transport feedback acknowledgments over time. A-SW-RLNC envisions two different FEC mechanisms to add redundancy (retransmissions) and cope with the errors and failures, according to the channel conditions. The first one is $apriori$ and the second one is $aposteriori$, and they both interplay to obtain a desired throughput-delay trade-off. The first FEC mechanism is $apriori$, as it sends redundant packets in advance (before the failure occurs) according to the average estimation of the channel behavior. The second FEC mechanism is $aposteriori$, as it sends redundant packets according to the realization of errors, identified using the transport feedback information.

It is via the second mechanism that the sender ensures that decoding is eventually possible at the receiver. We note that, the higher the number of $apriori$ FECs is, the lower is the delay and the throughput, as it pro-actively recovers (possibly more than needed) for future lost coded packets. On the other hand, the higher the number of $aposteriori$ FECs is, the higher is the delay and the throughput, as it only recovers for the needed lost coded packets at the cost of a delay proportional to RTT. The way to adjust this trade-off is through an adaptive approach, which is described in [10].

IV. Experimental Study and Performance Evaluation

A. Experimental Outdoor mmWave Setup

The mmWave network is composed of three Cambridge Communication Systems (CCS) Metnet nodes [20], which are presented in Fig. 2. These nodes were deployed in an outdoor environment, as part of the Aveiro Tech City Living Lab [21], which allowed running tests under a fully controlled environment. The deployed network adopts an architecture where the Personal basic service set Control Point (PCP) node has a wired connection to the core network. On the other hand, nodes A (normal connection) and B (affected connection), the remote nodes, access the network through the radio links they establish with the node PCP. For each node, there is a single board unit (APU) connected, that will communicate using the mmWave backhaul. In the experimental scenario, a metallic obstacle was placed between two Stations (STAs) to simulate the blockage scenario.

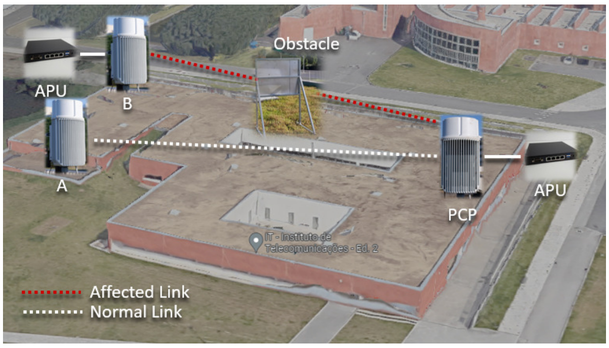

Fig. 2: Experimental mmWave network test scenario.

B. Data Collection: Recording the mmWave Channel Profile

The dataset characterizing the mmWave channel profile, *i.e.*, the RTT and the packet loss event at each time slot, was collected using the mmWave setup illustrated in Fig. 2. UDP traffic was generated and, at the same time, the RTT and the state of the packet (being erased or not) were collected using the Two-Way Active Measurement Protocol (TWAMP) tool. The tool implements the standard defined in the RFC 5357 [22], which is capable of performing round-trip measurements. The unprocessed dataset consists of a 5-minute execution of the TWAMP tool for each MCS mode. The time slot duration is 450 μs, obtained by trial and error as the maximum value without losing packets due to processing limitations.

In order to evaluate the channel behaviour and collect the metrics needed by the adaptive communication solutions, we retrieved from the unprocessed dataset a sequence of tuples, represented as (RTT, packet loss) per time slot. We call this sequence a *Channel Profile (CP)*, and in total 5 CPs were obtained (one per each mode in $\{auto, 3, 4, 5, 6\}$).

C. Emulation: Communication over Transport-Layer mmWave Channel

To evaluate the performance of the transport-layer communication solutions under the collected channel profiles *CP*s, an emulator was developed on top of Steinwurf's Kodo FEC components [23] (using research license). Our implementation of the Rateless RLNC (sec. III-A), the Sliding Window variant of the Rateless (F-SW-RLNC), and the A-SW-RLNC (sec. III-B) are built upon the *Block* and *Slide* RLNC schemes in these libraries, which we extended accordingly.

The baseline scenario is represented as a UDP transmission of a pseudo-randomized binary file, emulated using the recorded *CP*. The scenario consists of a file divided into 100 datagrams of 1000 bytes each. Each successful delivery under the defined scenario using a communication algorithm is called an *experience*. The completion of an *experience* outputs a triple, consisting of the normalized throughput, the mean and the maximum in-order delay metrics. A *datapoint* is then defined as the mean of each metric over 10 experiences. The simulation will complete when all time slots from a *CP* are used (*i.e.*, the length of the *CP* sequence is exhausted). The total set of datapoints is then collected per tested *CP* and communication solution.

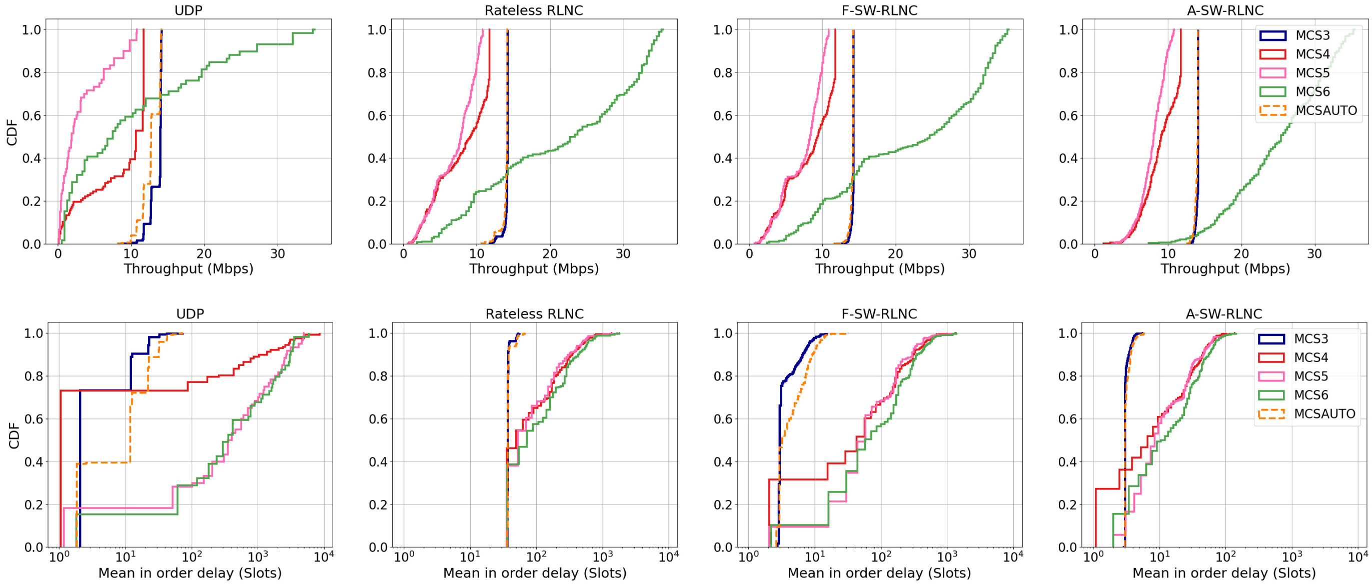

Fig. 3: Normalized throughput (top) and mean in-order delay (bottom) for each communication solution (UDP, R-RLNC, F-SW-RLNC and A-SW-RLNC).

D. *Performance Analysis of RLNC Solution over mmWave*

After running the emulator with the collected channel profiles, we obtained the main performance metrics for several transport-layer communication solutions, presented in Fig. 3 and Table I. Our results show a significant improvement regarding the in-order delivery delay while maintaining a high throughput, when using adaptive network coding to improve the communication in the transport layer. Next, we compare the various communication protocols regarding each performance metric.

1) Throughput: Fig. 3 shows that deploying A-SW-RLNC over mmWave network link results in a higher overall throughput compared to the UDP baseline across all MCS modes, despite the high RTT variance. Besides, the higher MCS modes obtain higher throughput as expected. For MCS 3 and Auto, A-SW-RLNC slightly outperforms other coding solutions, and at the same time presents a consistent behaviour (*i.e.*, with less fluctuations). Table I confirms, for the MCS Auto mode, there is no notable difference on the 99^{th} percentile between the Rateless-RLNC and the A-SW-RLNC. This means that A-SW-RLNC has no significant throughput penalty, while offering significant delay improvements as we will show.

2) Mean In-Order Delivery Delay: Regarding the mean in-order delay, A-SW-RLNC results in a dramatic improvement over R-RLNC and UDP solutions, which is attained thanks to its adaptive and dynamic components (Section III-B). With MCS 3 and MCS Auto, Fig. 3 shows that a simple UDP transmission achieves lower delays than the R-RLNC scheme. This is because for R-RLNC, the sender transmits batches of encoded data for the same generation until ensuring the successful decoding, and this limits the minimum theoretical achievable delay (Section III-A). The F-SW-RLNC implementation removes this limitation and achieves better results for the upper quartile and above with an order of two. For high MCS modes, the sharp increase in the channel erasure rate makes a simple UDP file transfer to be unusable. This is because the retransmission probability is substantial, and this increases the packet in-order delivery delay. The R-RLNC implementation mitigates these losses via a generous code redundancy, improving over the in-order delay of a UDP transmission. From the CDF curves, gains start for the upper steps of the 20^{th} percentile. For the MCS 6, regarding the 99^{th} percentile of the mean in-order delay, the R-RLNC outperforms the UDP transmission by a factor of 4.60, and the A-SW-RLNC outperforms the R-RLNC with a factor of 9.96, see Table I.

3) Maximum In-Order Delivery Delay: With respect to the maximum in-order delivery delay, Table I shows that the R-RLNC algorithm does not differ much from the previous average delay analysis, showing that its bounds are quite close. For the MCS 6 and in the 99^{th} percentile, the statistical values for the maximum delay show a significant improvement with a factor of 4.22 times for R-RLNC compared to UDP transmission and 11.36 times for A-SW-RLNC compared to UDP transmission, respectively. Similar to the average analysis, the F-SW-RLNC shows a slight improvement in the maximum in-order delivery delay. In fact, A-SW-RLNC achieves an improvement with a ratio of 2 to 2.5 across all percentile bounds over the former approach (F-SW-RLNC), from MCS Auto to MCS 6.

4) LLC and URLLC Performance Indicators: Regarding the support of low latency and ultra-reliablity in the mmWave IAB, Table I highlights the schemes that are capable to achieve LLC and URLLC requirements. For LLC applications, we target a mean in-order delay below 10 ms (22 slots). For URLLC applications (only $P_{99\%}$), a max in-order delay below 10 ms (22 slots) is targeted. As presented in Table I, only the A-SW-RLNC can support URLLC applications by obtaining a max in-order delay below 10 ms for $P_{99\%}$. In such lossy links, transport protocols like UDP cannot be used in URLLC scenarios, as well as the rateless and the F-SW-RLNC schemes. When addressing LLC applications, the A-SW-RLNC scheme is capable to achieve a delay below 10 ms in all the MCSs evaluated, allowing the increase of the network bandwidth by using a higher MCS. As presented, low-layer techniques are very conservative and do not allow to take the

TABLE I: Statistics for simulation results of tested algorithms over MCS 4, MCS 6 and Auto modes. For a time slot of $450\mu s$, the schemes achieving LLC and URLLC are marked.

Mode	Algorithm	Throughput (Mbps)			Mean In-Order Delay (slots)			Max In-Order Delay (slots)		
		Mean	Stdev	$P_{99\%}$	Mean	Stdev	$P_{99\%}$	Mean	Stdev	$P_{99\%}$
MCS4	UDP transmission	8.62	4.27	0.23	425.44	1 180.34	5 106.57	427.41	1 180.38	5 108.70
	Rateless RLNC	8.10	3.59	1.40	140.53	180.12	765.62	148.18	181.56	779.55
	F-SW-RLNC	8.33	3.39	1.59	114.29	163.71	681.71	145.74	175.30	734.11
	A-SW-RLNC	9.11	2.36	3.87	16.66	20.04	80.23	63.86	77.37	324.84
MCS6	UDP transmission	10.64	9.78	0.80	1 029.80	1 299.62	4 601.51	1 031.76	1 299.59	4 604.08
	Rateless RLNC	21.39	10.36	3.45	177.93	228.41	999.80	186.20	229.38	1 009.81
	F-SW-RLNC	22.04	10.07	3.97	154.31	192.85	848.23	193.42	199.65	899.50
	A-SW-RLNC	25.17	6.36	11.01	22.28	23.63	100.38	87.16	88.85	405.18
Auto	UDP transmission	12.82	1.21	14.15	12.96	11.63	43.20	15.60	11.63	45.82
	Rateless RLNC	13.98	0.50	14.22	38.31	4.03	55.08	43.27	4.20	61.43
	F-SW-RLNC	14.08	0.27	14.22	5.15	3.29	15.43	11.25	6.48	29.90
	A-SW-RLNC	13.92	0.24	14.08	3.22	0.42	4.90	4.78	2.71	14.85

full benefits of the mmWave link capacity.

V. Conclusions and Future Visions

In this work, we proposed a significant enhancement on the mmWave performance by incorporating network coding algorithms to stabilize the high-frequency communication sensitivity. In particular, we showed that, using A-SW-RLNC, it is possible to obtain ultra-reliable high bandwidth while reducing by up to an order of two the mean in-order delay. Our results demonstrate that the communication protocols can notably take benefit from relaxing the PHY and MAC layer error control mechanisms, and delegating the task to the upper layers using the proposed network coding solution. In fact, the retransmissions that occur due to MAC error control mechanisms are effectively not needed once the network coding solution and FEC mechanisms are utilized.

As for future work, we plan to use the gain of Multi-Path (MP) network coding communication through splitting the mmWave band into several sub-bands. For this end, we will extend the proposed SP solution to several frequency links via an effective MP coded communication [11]. To use the proposed solution over highly-meshed backhaul of novel communication networks, we plan to incorporate software-defined controllers for collecting information that can enhance the communication performance over meshed mmWave links [24]. Last but not the least, we plan to exploit the recent trend in the estimation of error patterns using deep-learning solutions to further improve our adaptive solutions over mmWave networks [25].

References

[1] C. D. Alwis, A. Kalla, Q.-V. Pham, P. Kumar, K. Dev, W.-J. Hwang, and M. Liyanage, "Survey on 6G Frontiers: Trends, Applications, Requirements, Technologies and Future Research," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 836–886, 2021.

[2] 3GPP, "Study on scenarios and requirements for next generation access technologies," 3rd Generation Partnership Project (3GPP), Technical Report (TR) 38.913, 4 2022, version 17.0.0. [Online]. Available: http://www.3gpp.org/DynaReport/38913.htm

[3] Y. Ren, W. Yang, X. Zhou, H. Chen, and B. Liu, "A survey on TCP over mmWave," *Computer Communications*, vol. 171, no. February, pp. 80–88, 2021. [Online]. Available: https://doi.org/10.1016/j.comcom.2021.01.032

[4] A. Shokrollahi, "Raptor codes," *IEEE/ACM Transactions on Networking (TON)*, vol. 14, no. SI, pp. 2551–2567, 2006.

[5] (2021, December) BitRipple, Qualcomm Technologies and Verizon Work Together to Enable the First 8K HDR Smartphone Video Call over 5G. [Online]. Available: https://6park.news/hawaii/bitripple-qualcomm-technologies-and-verizon-work-together-to-enable-the-first-8k-hdr-smartphone-video-call-over-5g-news.html

[6] J. Cloud, D. Leith, and M. Médard, "A coded generalization of selective repeat ARQ," in *2015 IEEE Conf. Comp. Comm*, 2015, pp. 2155–2163.

[7] G. Joshi, Y. Kochman, and G. W. Wornell, "The effect of block-wise feedback on the throughput-delay trade-off in streaming," in *Proc. IEEE Conf. Computer Commun. Wkshps*, 2014, pp. 227–232.

[8] V. Cerf, Y. Dalal, and C. Sunshine, "Specification of internet transmission control program," RFC 675, December, Tech. Rep., 1974.

[9] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Commun.*, vol. 1, pp. 3–20, Mar. 2008.

[10] A. Cohen, D. Malak, V. B. Bracha, and M. Médard, "Adaptive causal network coding with feedback," *IEEE Trans. on Comm*, vol. 68, no. 7, pp. 4325–4341, 2020.

[11] A. Cohen, G. Thiran, V. B. Bracha, and M. Médard, "Adaptive causal network coding with feedback for multipath multi-hop communications," *IEEE Trans. on Comm*, vol. 69, no. 2, pp. 766–785, 2020.

[12] F. Michel, A. Cohen, D. Malak, Q. De Coninck, M. Médard, and O. Bonaventure, "FlEC: Enhancing QUIC with application-tailored reliability mechanisms," *IEEE/ACM Transactions on Networking*, 2022.

[13] B. Schultz, "802.11 ad - wlan at 60 GHz - a technology introduction," *Rohde & Schwarz*, 2013.

[14] IEEE Std 802.11e™-2005, *Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements*, 2005, vol. 2005.

[15] M. Dahhani, G. Jakllari, and A. L. Beylot, "Association and Reliability in 802.11ad Networks: An Experimental Study," *Conference on Local Computer Networks, LCN*, vol. 2019-Octob, pp. 398–405, 2019.

[16] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, 2006.

[17] N. Bonello, Y. Yang, S. Aissa, and L. Hanzo, "Myths and realities of rateless coding," *IEEE Communications Magazine*, vol. 49, no. 8, pp. 143–151, 2011.

[18] L. Yang, Y. E. Sagduyu, J. Zhang, and J. H. Li, "Deadline-aware scheduling with adaptive network coding for real-time traffic," *IEEE/ACM Transactions on Networking*, vol. 23, no. 5, pp. 1430–1443, 2014.

[19] Y. Shi, Y. E. Sagduyu, J. Zhang, and J. H. Li, "Adaptive coding optimization in wireless networks: Design and implementation aspects," *IEEE Trans. on Wireless Comm*, vol. 14, no. 10, pp. 5672–5680, 2015.

[20] "Metnet 60G unlicensed mmwave mesh datasheet - ccsl.com," 02 2020. [Online]. Available: https://www.ccsl.com/v1/uploads/files/Metnet-60G-Mesh-datasheet.pdf

[21] P. Rito, A. Almeida, A. Figueiredo, C. Gomes, P. Teixeira, R. Rosmaninho, R. Lopes, D. Dias, G. Vítor, G. Perna, M. Silva, C. Senna, D. Raposo, M. Luís, S. Sargento, A. Oliveira, and N. B. de Carvalho, "Aveiro Tech City Living Lab: A Communication, Sensing and Computing Platform for City Environments," *IEEE Internet of Things Journal*, pp. 1–1, 2023.

[22] J. Babiarz, R. M. Krzanowski, K. Hedayat, K. Yum, and A. Morton, "A Two-Way Active Measurement Protocol (TWAMP)," RFC 5357, Oct. 2008. [Online]. Available: https://www.rfc-editor.org/info/rfc5357

[23] Steinwurf, "Kodo-python," https://github.com/steinwurf/kodo-python.

[24] A. Cohen, H. Esfahanizadeh, B. Sousa, J. P. Vilela, M. Luís, D. Raposo, F. Michel, S. Sargento, and M. Médard, "Bringing Network Coding into SDN: Architectural Study for Meshed Heterogeneous Communications," *IEEE Communications Magazine*, vol. 59, no. 4, pp. 37–43, 2021.

[25] A. Cohen, A. Solomon, and N. Shlezinger, "DeepNP: Deep Learning-Based Noise Prediction for Ultra-Reliable Low-Latency Communications," in *2022 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2022, pp. 2690–2695.