Abstract—Autonomous systems and mission-critical applications demand ultra-reliable low-latency communication (URLLC). To build wireless communication networks capable of accommodating such applications, optimization of the air interface characteristics is vital. This paper leverages recent advancements in the field of Artificial Intelligence (AI) technologies to optimize specific aspects of the air interface design to satisfy these stringent link reliability and latency requirements. The precise aim of this research is to reduce the link latency caused by the presence of the Hybrid Automatic Repeat reQuest (HARQ) mechanism. To this end, we propose a novel deep learning-based algorithm (Deep-HARQ), employing a deep neural network (DNN) with fully connected layers to estimate the decodability of the coded-received in-phase and quadrature (I/Q) signals prior to accomplishing the majority of the complex reception tasks. This enables the receiver to respond faster, allowing for the reduction of the signal round-trip time (RTT). To evaluate Deep-HARQ with a realistic dataset, we collected training and validation samples from a waveform compatible with 3GPP 5G NR Release 15 standards. The simulation results reveal a faster estimation response, with an accuracy enhancement of 12% compared to relevant algorithms in the literature.

Index Terms—5G, 6G, deep learning, HARQ, latency, machine learning, URLLC.

I. INTRODUCTION

Increased demand for interactive and high-performance applications has motivated stakeholders to research and develop fifth-generation (5G) networks. Complementing well-known broadband services, 5G networks will aim to support use cases with diverse resource requirements in terms of latency, capacity, and massive connectivity, making the waveform design extraordinarily challenging. Ultra-Reliable and Low-Latency Communication (URLLC) is one of the vital service classes of 5G, as it is intended to support applications dependent on mission-critical links [1], delivering a short packet within up to 1 ms at a success rate of at least 99.999%. To this end, a set of 5G New Radio (NR) specifications has been standardized by the 3rd Generation Partnership Project (3GPP) in Rel. 15, which mainly covered the fundamental features of URLLC. Indeed, the air interface latency aspect has been implicitly addressed by considering the new 5G numerology; however, there are still open issues that need to be solved to fully achieve the target latency requirement. For instance, the HARQ retransmission mechanism over the air-interface is still not standardized; in particular, the required time to transmit and then successfully receive a packet, including all its possible retransmission rounds, has to be minimized to meet the 1 ms latency target [2]. Otherwise, the contribution of traditional HARQ to URLLC would be doubtful. However, there are significant URLLC related enhancements introduced in 3GPP Rel. 16 and 17. These research efforts can lead the advancements towards the next generation of URLLC, referred to as eXtreme ultra-reliable and low-latency communication [3]. Among the research efforts driven by academia and industry to fulfill URLLC requirements, some proposals have focused on achieving the latency targets by reducing the round trip time (RTT). This is defined as the interval between sending a packet, followed by the receipt of the acknowledgment message (ACK)/negative-acknowledgment message (NACK) from the receiver, up to the re-sending of the previous (or new) packet [4]. Employment of a mini-slot frame structure and utilization of a wider subcarrier space are concepts that are already specified in the new 5G numerology to minimize the RTT. Other proposals consider optimizing the HARQ process. HARQ combines forward error-correction coding and error detection, both performed at the receiver, to release a NACK when a decoding failure occurs due to poor channel conditions, or an ACK when decoding is successful. Despite the network reliability goal that can be achieved with HARQ, it can cause a high-latency cost due to the multiple retransmissions and multiple signal decoding operations at the receiver. Therefore, the delay associated with HARQ can create a bottleneck, compromising the achievement of the challenging URLLC latency requirement. The above challenges motivated our work, in which we design an estimator capable of delivering early HARQ (E-HARQ) feedback regarding the decodability of the received codeword prior to completion of the reception task. In this design, the transmitter receives feedback at an earlier stage, allowing a more rapid response in the form of a new message or re-transmission of the same message. It is noteworthy that the concept of E-HARQ feedback has previously been discussed and proposed for 5G. In line with this, the authors in [5] propose an E-HARQ algorithm relying on the log-likelihood ratio (LLR) to estimate prior feedback,

Deep Learning-Based Estimator for Fast HARQ Feedback in URLLC

Saleh AlMarshed
5G & 6G Innovation Centres
Institute for Communication Systems
University of Surrey, UK
s.almarshed@surrey.ac.uk

Dionyssia Triantafyllopoulou
5G & 6G Innovation Centres
Institute for Communication Systems
University of Surrey, UK
d.triantafyllopoulou@surrey.ac.uk

Klaus Moessner
Professorship for Communications Engineering
Technical University Chemnitz, Germany
and 5G & 6G Innovation Centres
Institute for Communication Systems
University of Surrey, UK
klaus.moessner@etit.tu-chemnitz.de
where a decision is made following partial signal decoding. The authors in [4] aimed to further expedite feedback release by allowing an estimation decision to be made after only a part of the codeword has been decoded. This approach relied on Low-Density Parity-Check (LDPC) for two reasons. Firstly, it is a potential candidate for URLLC services, since it is being adopted by 3GPP as a channel coding scheme for Enhanced Mobile Broadband (eMBB). Secondly, it offers excellent opportunities to exploit substructures, allowing faster reactions to estimate feedback utilizing only parts of the codeword. Furthermore, a machine learning-based (ML) approach is investigated in [6]. This involves enhancing feedback prediction by training the ML model to classify the decodability status, rather than utilizing the traditional method (i.e., hard threshold-based). Although E-HARQ feedback schemes (such as [4]–[6]) appear to produce a good classification performance, they rely on channel decoding to estimate the early feedback. We believe that coupling the early feedback with the decoding process can form a barrier to achieving the strict latency requirement of URLLC. Specifically, when considering the significant delay caused by the channel decoding task, i.e., consuming approximately 60% of the total processing time in LTE [7], we aimed to address this issue by releasing feedback ahead of the channel decoding [8]. Despite the achieved latency and reliability enhancement, the proposed ML-based scheme still relied on LLR to estimate the early feedback, which is considered a delay source at the user equipment (UE) end. Thus, in this paper, we propose a novel Deep-HARQ scheme empowered by deep learning technologies (DL), capable of estimating the feedback without the need to manually extract features, taking advantage of the fact that DNNs demonstrate excellent classification capability by learning and analyzing the characteristics of sophisticated wireless channels [9]. The principal contributions of this paper are:

- A Deep-HARQ algorithm, capable of estimating early HARQ feedback with an accuracy enhancement of up to 12% compared to other techniques introduced in the literature.
- The proposed estimator is decoupled from most complex receiver tasks, i.e., channel decoding and LLR calculating, as the estimation relies solely on the coded-received I/Q samples; hence feedback is released at an earlier stage than possible with pre-existing algorithms.
- We conduct a feasibility study in applying deep learning techniques on wireless communication networks from a waveform designer perspective. More specifically, we identify a trade-off between potential gain under different channel conditions and evaluate the respective model complexity.

The paper is structured as follows: Section II describes the system model, including the model utilized to collect the data along with the proposed Deep-HARQ estimator. Section III presents the performance evaluation, including the simulation results and complexity evaluation, and finally, section IV provides concluding remarks and outlines for future work.

II. SYSTEM MODEL

Supervised learning-based models need to be trained using a realistic dataset to deliver reliable performance. However, applying DL technologies with a physical layer design is a novel concept, and because URLLC standardization activities are still in progress there is no suitable public dataset available that is applicable to our research. Thus, to collect a training dataset, we modeled a 5G-waveform compatible with 3GPP NR Release 15 standards [10]. As the URLLC has strict latency and reliability requirements, we modified some of the waveform parameters to be in line with potential URLLC aspects by adopting the newly proposed concepts, such as the wider subcarrier spacing and mini-slot strategy. This can lead to a shorter Transmission Time Interval (TTI), thereby minimizing the latency by reducing the processing and buffering times, which is one of the primary methods to satisfy the strict latency requirements of URLLC [11]. The following sub-section offers a brief discussion of the modeled waveform, devised to collect the training and validation dataset, alongside the Deep-HARQ algorithm.

![Typical transmitter diagram that is used to collect the training and validation dataset.](image-url)

Fig. 1. Typical transmitter diagram that is used to collect the training and validation dataset.

A. Data Collecting Model

In this model, we consider a wireless multiple-antenna communication system, which has \( N_t \cdot N_r \) transmitter and receiver antennas, respectively. Each user is allocated with a group of \( L \) orthogonal Resource Blocks (RBs) in the frequency domain. Each RB contains a grid of resource elements (REs) spanning \( S_f \) OFDM symbols over the frequency domain, and \( S_t \) consecutive subcarriers over the time domain. The RB has a varying time duration that depends mainly on the slot size and the symbol’s subcarrier spacing. Hence, each packet contains a total of \( P = L \times S_f \times S_t \) symbols and has a time duration referred to as TTI. The URLLC message consists of a number of bits, referred to as \( B \), that are encoded to a codeword via an LDPC encoder. Each bit on the codeword is mapped to a symbol using an appropriate modulation scheme (i.e., 8PSK, 16-QAM). Then, the reference symbols are inserted to the modulated codeword \( S \) while processing the other transmitter...
functions, such as MIMO precoding and resource mapping. The details of the transmitter functions are shown in Fig.1. The channel input-output relation within the $i$th RB can be expressed as follows:

$$Y_i = S_i H_i + W_i,$$

where $S_i \in C^{S \times N_I}$ and $Y_i \in C^{S \times N_R}$ are the transmitted and received signal, respectively, $H_i \in C^{N_N \times N_R}$ is the fading matrix, that is assumed to support clustered delay line (CDL) [12]. $W_i \in C^{S \times N_R}$ denotes the additive white Gaussian noise (AWGN), which has independent $W \sim CN(0,1)$. Finally, The real and imaginary parts of the received signal, representing I/Q samples, will be collected as features (input) and the dataset labels, i.e., the actual decodability outcome, will be collected following completion of all the reception functions, including the signal decoding and integrity check processing of the decoded bits.

### B. Deep-HARQ estimator

In the conventional E-HARQ mechanism, i.e., non-ML-based, the early estimation is performed based on manually-extracted features that are inferred from the log-likelihood ratio (LLR), i.e., a partial process from the signal decoding. The decision then is made by mapping each extracted feature to an ACK or NACK message utilizing a hard threshold obtained experimentally [6]. On the contrary, our proposed Deep-HARQ mechanism is decoupled from the decoding task, where the received-coded I/Q input is directly employed to estimate the early feedback. For this, many pre-collected samples are utilized for training the DNN without human intervention. A brief description of the Deep-HARQ structure is provided below, including the training and validation procedure.

1) **Structure of Deep-HARQ**: The idea behind the Deep-HARQ mechanism is to build a DNN that has the capability to early estimate the decodability outcome from the received-coded signal. The structure of the proposed network is illustrated in Fig.2, where a long short-term memory (LSTM) network is employed to classify the received I/Q blocks. The network structure contains two fully connected (FC) layers; the hidden layer has $M$ nodes, while there are $K$ nodes in the output layer. The activation function that represents each node on the hidden layer is either the rectifier linear unit (Relu) or the hyperbolic tangent (Tanh) function, while the Sigmoid function is applied on the last layer to obtain an output that lies between 0 and 1. The decodability outcome, therefore, will be assumed to be 0 when its value is less than 0.5 or 1 otherwise. Table I shows the adopted activation functions on the model. At this point, it is essential to highlight the key factors that play a role in the design of a reliable classifier. The first factor is the input and the output size of the DNN. This is mainly determined based on the application requirements. In our problem, the classification is performed for each codeword independently; therefore, the model input length relies on the available RBs for each codeword. To be in-line with the potential URLLC frame structure, we chose the minimum scheduling requirement based on 5G standards (i.e., 20 RBs/scheduling). We also assume that the mini slot concept is applied (2 symbols/slot), thus each RB can contain $(12 \times 2)$ resource elements; so the number of symbols in each codeword is 480. The second factor is the number of neurons in the hidden layer. This mainly depends on the input length and the desired performance. As the number of nodes increases, the network performance is generally enhanced but also results in an increased complexity. Therefore, adopting the optimal number of nodes that can guarantee a balanced trade-off between the model complexity and the classification accuracy, is considered one of the main challenges of obtaining a reliable classifier, and is out of the scope of this paper.

2) **Training Procedure**: To train a reliable Deep-HARQ estimator, we need to obtain a vast labeled dataset. More specifically, various codewords will be randomly generated at diverse physical channel conditions. On the receiver side, the received-coded I/Q of each codeword will be collected as the network input, and the corresponding decodability outcome $(b_0, b_1)$ will be recorded as labels after completing all the reception tasks, as described above. It should be noted that the number of training samples should be large enough to prevent the model from capping with over-fitting when the trained model is generalized. Using the collected samples, the model will be trained to minimize the difference between the original decodability outcomes $p$ and the predicted outcomes $p^\prime$. Thus, we initially find the loss function using the mean-squared error (MSE) as follows:

$$L(p, p'; \theta) = \frac{1}{N} \sum_{i=1}^{N} \|p - p'|^2,$$

where $N$ is the number of samples and $\theta = (W_\theta, b)$ are the model parameters, i.e., weights and biases. To find the optimal
values of $\theta$ that return the minimum cost value, stochastic gradient (SGD) algorithm is adopted as follows:

$$\theta^{+} = \theta - \alpha \nabla L(p, y'; \theta), \tag{3}$$

where $\alpha$ is the learning rate that defines the step size of the SGD and $\theta^{+}$ are the updated parameters. During the training stage, the algorithm will find the optimal value of $\theta^{+}$ by repeatedly passing the training samples at different values of $W_0$ and $b$. Each time, the partial derivatives (i.e., gradient) for the cost function will be computed with respect to the current value of $\theta$. The value of $\theta^{+}$ will keep being updated until the algorithm converges.

III. PERFORMANCE EVALUATION

A. Simulation Setup

The performance of the proposed Deep-HARQ algorithm is compared to algorithms previously suggested in the literature, including two approaches relying on the LLR to estimate the decodability feedback. The first approach, referred to as Hard Threshold (HT) method, utilizes a hard-threshold for the classification decision [5], whereas in the second approach, referred to as Logistic Regression (LR) method, the decision is made based on ML [6]. To evaluate such algorithms, we performed link-level simulations, as shown in section II.A. In all instances, we applied 600K transmissions with independent channel realizations. As external parameters, we varied the SNR values under low channel quality conditions for two reasons. Firstly, erroneous reception is most likely to occur at low SNR; thus, early feedback will inevitably be more effective. Secondly, we can obtain a balanced dataset in this region, one characterized by fair equality between decodability outcomes, i.e., the number of correct blocks being similar to the number of corrupted blocks. This strategy aims to prevent the classifier from failing to manage an imbalanced training dataset, and allows evaluation of the classifier in a critical operational region. The simulation parameters applied when collecting the training and validation samples are illustrated in Table II. Meanwhile, obtaining a reliable Deep-HARQ estimator requires determination of the appropriate hyperparameters of the DNN, such as epochs, batches, and learning rate. In our design, we set the number of epochs, i.e., defined as the number of times the entire dataset is shown to the network while training, to be 128. The dataset in each epoch is grouped into mini batches comprising 1000 samples. The model parameters can then be updated after each batch is processed based on the learning rate $\alpha$, which is chosen to be 0.001. Finally, the hidden layer is set to have 12 nodes. However, no specific mathematical criteria exist to establish the optimal network hyperparameters, as parameter determination takes place empirically based on observation of the models’ response during the training and validation processes.

B. Classification Performance

The classification performance can be evaluated using learning curves (LCs), as shown in Fig.3. This method is an efficient diagnostic tool designed to observe the training progress over time. The training curve explains the model learning progress, whereas the validation curve can be utilized to examine the model’s generalizability, which is known as the capability of the trained model to classify previously unseen data during the training process, i.e., a validation dataset. As the training goal is to obtain a reliable classifier, there must be a good fit-relationship between the training and validation curves, i.e., avoiding resulting in an overfit or underfit model. As can be seen from the learning curves, the classifier response achieved the target effectively, as the loss curves decrease smoothly over time to a point of stability. Meanwhile, both accuracy curves increase significantly up to 96%. It is also notable that the gap between the curves is relatively small until the conclusion of the training process. This gap is referred to as the generalization gap and is acceptable assuming that the curves converge and are reasonably stable. To compare Deep-HARQ to the other classification algorithms, we evaluated the classification accuracy by finding the ratio of number of correct predictions to the total number of input samples. The results, as illustrated in Table III-B, demonstrate a 9% and 12% accuracy enhancement in favor of the proposed Deep-HARQ compared to the ML-based and HT-based classifiers, respectively, i.e., when the CR is 2/5. The promising simulation results are further extended to evaluate the model’s generalizing ability at different channel coding rates. The performance gain is still shown to be effective, with an acceptable degradation of the overall system performance for all classifiers arising from the use of a higher channel coding rate.

C. System Performance

We start by discussing the Block Error Rate (BLER) performance of the three classification algorithms under comparison. The simulation results, as set out in Fig.4, demonstrate the superiority of the proposed Deep-HARQ across all SNR regions. Such behavior can be attributed to the complexity of the DNN structure, which allows the network to develop
Fig. 3. i) The model accuracy improvements over the training iteration, ii) The loss between the actual outcome and the predicted outcome.

TABLE III
CLASSIFIERS ACCURACY COMPARISON IN TERMS OF DIFFERENT CODING RATES (CR).

<table>
<thead>
<tr>
<th>Method</th>
<th>Feature</th>
<th>CR= 2/5</th>
<th>CR= 5/6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard Threshold (HT)</td>
<td>LLR</td>
<td>84%</td>
<td>81%</td>
</tr>
<tr>
<td>Logistic Regression (ML)</td>
<td>LLR</td>
<td>87%</td>
<td>84%</td>
</tr>
<tr>
<td>Deep-HARQ (DL)</td>
<td>IQ</td>
<td>96%</td>
<td>91%</td>
</tr>
</tbody>
</table>

the capability to remember and analyze the characteristics of sophisticated wireless channels. Compared to the standard wireless transmission without E-HARQ, the emerging gap in performance narrows in favor of the new proposal. Notably, all the classifiers have a similar performance in low SNR regions; this is a result of the fact that the majority of the decodability outcomes in these SNR regions have a monopolistic majority over one-class from among the others. Therefore, the classifiers tend to be biased toward the outcome of the major class. In contrast, the performance difference is more apparent when the SNR values fall between -5 dB to -2 dB. To examine this behavior in-depth, let us find the ratio of the incorrectly estimated outcomes to the total observations, which refers to the misclassification rate; this can help understand the classifiers’ behavior at different SNRs. As shown in Fig. 5, the difference in the model’s performance peaks at SNR=-3.8 dB, where the decodability outcomes are relatively even, rendering the classification decision more challenging. However, the DL-based algorithm surpasses the other methods by minimizing the misclassification rate peak in this critical region. In general, employing E-HARQ techniques causes a slight deterioration of the BLER performance. This is a result of the fact that the E-HARQ feedback estimation is performed before the channel decoding and CRC processes. Therefore, the estimation is based on a noisy coded-signal, i.e., I/Q signal, that has not yet benefited from the error correction. However, our proposal shows a significant improvement with a faster response time associated with the release of the E-HARQ feedback.

D. Complexity and Latency Comparison

It is worth evaluating the complexity of the proposed Deep-HARQ algorithm compared to the proposed classifiers in the relevant literature. We use MATLAB to empirically compute the average runtime to train various classifiers besides the time required to estimate the early feedback. To obtain a fair comparison, the examination is employed for all the algorithms in the same environment, i.e., unified dataset, hardware, and software. Table IV illustrates that training of the proposed Deep-HARQ consumes a significant time due to the complex structure of the DNN. However, this computational
cost is considered acceptable as the model relies on supervised learning, where the model can be previously trained off-board at various channel conditions. Unlike the training behavior, the early feedback estimation, i.e., at the UE end, consumes less time than the other classifiers. Notably, this difference results from the capability of the new algorithm to release early feedback ahead of the decoding task, leading to a latency saving that is equivalent to the consumption time allocated to the costly codeword decoding task. However, despite this promising delay reduction, it is difficult to deduce computational complexity by simply relying on measuring the execution time for two reasons. Firstly, the various parameters of modern computers, such as the Central Processing Unit (CPU) architecture and cooling power, which are not part of the algorithm, can significantly affect the algorithm’s execution. Secondly, typical URLLC radio hardware differs from what is used in the test environment, i.e., from an implementation perspective. Besides, URLLC hardware is yet to be standardized. Ultimately, the execution time can reflect some aspects of complexity throughout initial investigations, mainly when algorithms are performed on a unified platform applying the same operating conditions.

### IV. Conclusion

This paper investigated a deep learning-based approach for Early HARQ feedback. An estimator was modeled as complementary DNN, then trained to classify the received-coded signal to decodability outcome. The simulation results demonstrate the capability of the DNN to perform the feedback estimation ahead of the most complex reception tasks, with improved estimation accuracy compared to other algorithms in the relative literature. It is essential to mention that employing a DNN can deliver outstanding accuracy, but comes at the cost of high computational complexity, which requires a lengthy training period compared to other algorithms. However, in our application, the complexity of training is less important, as Deep-HARQ can be pre-trained offline to determine its optimal network parameters, then employ the trained-model on radio devices, i.e., on-board. Our future plans include further generalizing the model to wider SNR regions, by accounting for the minority of erroneous blocks that occur occasionally at high SNRs. The plan also includes studying the capacity to re-tune the DL model on-board, specifically when the trained model experiences new channel conditions that do not match the channel models considered during training.

### ACKNOWLEDGMENT

We would like to acknowledge the support of the University of Surrey 5GIC (http://www.surrey.ac.uk/5gic) members for this work.

### REFERENCES


[12] 3GPP, “Study on channel model for frequencies from 0.5 to 100 GHz,” 3GPP TSG RAN, TR 38.901.