THROUGHPUT MAXIMIZATION AND LATENCY OPTIMIZATION IN FIFTH-GENERATION NETWORKS USING A MULTISTAGE MACHINE LEARNING FOR EARLY HYBRID AUTOMATIC REPEAT REQUEST

By

Nhlanhla Patrick Hlewane

DISSERTATION

Submitted in fulfilment of the requirements for the degree of

MASTER OF SCIENCE

in the

FACULTY OF SCIENCE AND AGRICULTURE

(School of Mathematical and Computer Sciences)

at the

UNIVERSITY OF LIMPOPO

Supervisor: Prof. Mthulisi Velempini

2023

DEDICATION

This dissertation is dedicated to my beloved people:

My mother, Martha Hlewane.

My sister, Siphokazi Hlewane.

My aunts, Siphiwe and Nomsa Msimango.

DECLARATION

I, ***Nhlanhla Patrick Hlewane*** hereby declare that this dissertation titled: **THROUGHPUT MAXIMIZATION AND LATENCY OPTIMIZATION IN FIFTH-GENERATION NETWORKS USING A MULTISTAGE MACHINE LEARNING FOR EARLY HYBRID AUTOMATIC REPEAT REQUEST** submitted at the University of Limpopo for master's degree is my original work and has not been previously submitted to any university or institution of higher learning. I further declare that all the sources cited are acknowledged and correctly referenced.

Signature: Nhlanhla Patrick Hlewane Date: 17 November 2022

ACKNOWLEDGMENTS

I thank the almighty God for strengthening me throughout this journey.

To myself for not giving up.

To my supervisor, Professor Mthulisi Velempini, thank you for your guidance throughout this journey. It was not simple at all for me, but your availability every time I needed help made the journey much lighter. Your feedback, comments and advice have grown me so much.

To my mother, Martha Hlewane, thank you for caring and loving. You always told me not to give up.

To all my colleagues in the Department of Computer Sciences, thank you a lot for your amazing support. I also acknowledge the University of Limpopo for this opportunity granted to me.

ABSTRACT

The physical layer Hybrid Automatic Repeat Request (HARQ) protocol efficiently achieves low error-rate transmission and high network reliability in the fifth generation (5G) Ultra Reliable Low Latency Communication (URLLC) network. However, this retransmission protocol suffers from increased transmission latency resulting mainly from the delay caused by channel decoding. This problem is caused by the fact that the sender has to wait for acknowledgement of the transmission which is generated after the decoding process at the receiver, resulting in increased latency. To address the latency problem, this study proposed the multistage machine learning Early HARQ (E-HARQ) which uses machine learning algorithms for predicting the acknowledgement before the decoding process. Furthermore, the proposed scheme uses the multistage decision to mitigate the throughput loss resulting from incorrect predictions of the acknowledgement. The multistage decision controls the transmission bandwidth in a multilevel manner depending on channel conditions measured by the Channel State Information (CSI). The study used jupyter notebook and MATLAB for developing the proposed scheme and then evaluating its performance. Simulation results show that the proposed scheme improves the achievable trade-off between the transmission latency and throughput which contributes to the performance of 5G URLLC networks.

Keywords: Multistage decision, Hybrid Automatic Repeat Request (HARQ), machine learning, fifth generation (5G)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

5G -	Fifth generation
16QAM -	16 Quadrature Amplitude Modulation
ACK -	Acknowledgment
AODV -	Ad hoc on-demand distance vector
BLEI -	Block error indicator
BER -	Bit error rate
CC-HARQ -	Chase Combining HARQ
CRC -	Cyclic redundancy check
CSI -	Channel State Information
CSV -	Comma-separated values
DSDV -	Destination sequenced distance vector
E-HARQ -	Early- Hybrid automatic repeat request
FN -	False-negative
FP -	False-positive
HARQ -	Hybrid automatic repeat request
IoT -	Internet of Things
IR-HARQ -	Incremental redundancy hybrid automatic repeat request
LLR –	Log likelihood ratio
MI -	Nutual information
mMTC -	Machine-type communications
NACK -	Negative Acknowledgement
OFDM -	Orthogonal frequency-division multiplexing
QPSK -	Quadrature Phase Shift Keying
RTT -	Round Trip Time
S-ARQ -	Simple Automatic Repeat Request
SC E-HARQ -	Subcodes E-HARQ
SNR -	Signal-to-noise ratio

SVM -	Support vector machine
TTI -	Transmission Time Interval
URLLC -	ultra-reliable low-latency communications

CHAPTER 1 – INTRODUCTION

1.1. Introduction

The fifth generation (5G) supports various use cases which include the Internet of Things (IoT) communication. However, IoT communication has been categorized into massive machine-type communications (mMTC) and ultra-reliable low-latency communications (URLLC) [1]. The URLLC requires high reliability and extremely low latency, but the physical layer Hybrid automatic repeat request (HARQ) poses a bottleneck for achieving the low latency required for URLLC because HARQ introduces additional delays in transmission designated as Round Trip Time (RTT). RTT is "the time interval between receiving the initial transmission and the retransmission" [2], [3].

To solve this latency problem, an Early-HARQ (E-HARQ) was proposed to predict the decoding outcome at an early stage using different methods such as machine learning [4], signal-to-noise ratio (SNR) [5] and Channel State Information (CSI) [6] for predicting the feedback so that the transmitter can receive feedback and react as early as possible to reduce the RTT [2]. However, prediction can be incorrect, leading to throughput loss due to unnecessary retransmission caused by a false alarm. This is the main challenge of the E-HARQ [5].

Moreover, incorrect predictions can also lead to miss-detection that would consequently result in increased latency caused by sending the correct Negative Acknowledgement (*NACK*) at a late stage. The literature review we conducted so far shows that little has been done to solve the throughput loss and increased latency problem caused by incorrect predictions, more especially for E-HARQ which uses machine learning algorithms for prediction in URLLC. This is a research gap which our research seeks to address since "our society [is becoming] increasingly reliant on IoT devices" [7].

1.2. Problem statement

The physical layer retransmission scheme, the HARQ, introduces additional latency to overall transmission [8], [3], [9]. This is caused by the fact that in HARQ, the receiver must decode the entire packet before sending feedback signals which may not be a viable solution for URLLC [10], [4], [11]. To address this problem, various methods for predicting the decoding outcome have been proposed [12]. The prediction allows the transmitter to receive feedback before the data is decoded at the receiver so that the transmitter can make decisions as early as possible [13].

Some of these methods include the use of Orthogonal frequency-division multiplexing (OFDM) symbols [12], the use of machine learning [4], [13] and the use of mutual information (MI) [6] to predict decoding the outcome. Although the prediction of the packet acknowledgement can reduce the latency in HARQ by decreasing the RTT, false alarms and miss-detections can occur, resulting in throughput loss and increased latency. This is because false alarms result in unnecessary retransmission which causes throughput loss [14], [5]. On the other hand, if miss-detections occur, the transmission latency cannot be reduced because the receiver sends a negative acknowledgement (NACK) after the decoding process [5].

The use of machine learning for feedback prediction has shown a significant improvement in terms of prediction accuracy [2]. However, the study in [2] did not develop a solution to mitigate the throughput loss and increased latency resulting from false alarms and miss-detections, which requires further research. To address this research gap, we proposed the use of machine learning algorithms to predict the decoding outcome in HARQ that would then send the acknowledgement (ACK) to the transmitter before the entire packet is decoded.

Our scheme differs from the method proposed in [2], in the sense that we use a multistage decision proposed in [5] to mitigate the throughput loss and the additional latency that could be caused by false predictions. In multistage decision-making, the number of coded bits is reduced in the retransmission packet so that the throughput can be alleviated when unnecessary early retransmission is performed due to false alarms.

1.3. Research motivation

HARQ supports various use cases in 5G URLLC networks which requires extremely low latency and high network reliability. The HARQ protocol offers better network reliability by achieving low error rate transmission. However, the HARQ retransmission protocol suffers from increased transmission latency resulting mainly from the delay caused by the channel decoding process. Various schemes have been proposed to improve the latency performance of the HARQ protocol, this includes the use of machine learning algorithms to predict the packet acknowledgement before the decoding process to reduce latency. However, the use of machine learning comes with some drawbacks such as throughput loss resulting from incorrect predictions, and not many studies have been done to address the throughput loss resulting from incorrect predictions which is a challenge addressed by this study. The study uses machine learning algorithms to predict packet acknowledgement in order to reduce latency. Furthermore, the proposed scheme uses a multistage decision to mitigate the throughput loss resulting from false predictions. In doing so, we hypothesize that the proposed scheme can improve both latency and throughput without affecting the network reliability in 5G URLLC.

1.4. Literature review

The latency of a HARQ system is mainly dominated by the HARQ RTT, which consists of the time required for generating feedback and the time for transmitting this feedback until the retransmission is received [3], [15]. Traditional HARQ protocols require the receiver to decode the entire packet before sending feedback signals, which may result in increased RTT [10].

To address this problem, Nadas et al. in [6] proposed an MI based on early HARQ (E-HARQ) which implements MI to predict the decoding outcome before sending the feedback to the transmitter ahead of the receiver decoding the entire packet. This was done to reduce RTT and the scheme improved the performance of HARQ retransmission schemes in terms of latency. The main drawback of this scheme is that false alarms could occur, resulting in throughput loss caused by unnecessary retransmissions.

As an extension to the study in [6], authors in [5] proposed a multistage decision to address the potential throughput loss due to false alarms. The multistage decision reduces the throughput loss by controlling the transmission bandwidth in a multilevel manner depending on the CSI [5]. Simulation results showed that the use of multistage decision in HARQ increases the achievable throughput compared to conventional methods. Our study uses the multistage decision to improve the achievable throughput in the machine learning E-HARQ.

In [4], a machine learning algorithm was proposed to predict the decoding outcome of HARQ as early as possible. Moreover, the use of machine learning to predict the decoding results of a given transmission using data generated after the first few decoder iterations is promising. This scheme uses the commonly used machine learning classification algorithm known as the logistic regression where the log-odds binary output is modelled as a linear combination of the classifier’s input variable. Simulation results show that machine learning methods have higher prediction accuracy compared to SNR based methods for feedback prediction.

Furthermore, Strodthoff et al. in [2] enhanced machine learning schemes by evaluating their performance with short and long Transmission Time Interval (TTI). However, simulation results show that the probability of false prediction of machine learning schemes is less when the TTI is shorter.

Machine learning E-HARQ schemes have been proven to have a high probability of prediction accuracy in literature. However, not much has been done to mitigate the throughput loss and increased latency caused by false alarms and miss-detections which require further probing. This study seeks to address the throughput and latency challenges associated with false alarms and miss-detections. This is achieved by reducing additional latency at the physical layer through the use of machine learning methods in HARQ to predict the decoding outcome before a packet is decoded at the receiver.

This is done to ensure that feedback to the transmitter is sent earlier so that the transmitter can respond as soon as possible, thereby reducing the RTT. Unlike the scheme proposed in [2], our scheme mitigates the throughput loss caused by unnecessary retransmission due to false alarms. This is achieved by using the multistage decision proposed in [5], where the number of coded bits is reduced in the retransmission packet so that the achievable throughput can be improved when unnecessary early retransmission is performed due to false alarms.

1.5. Aim and objectives

1.4.1 Aim

The aim of this study is to reduce latency and maximize the achievable throughput in HARQ to improve the overall performance of 5G URLLC networks.

1.4.2 Research objectives

i. To investigate the impact of false predictions on throughput and latency in 5G URLLC.
ii. To investigate the best technique for the optimization of end-to-end performance.
iii. To implement multistage decision in machine learning E-HARQ
iv. To evaluate the effectiveness of multistage decision on throughput maximization and latency optimization.
v. To deploy the multistage machine learning E-HARQ scheme on 5G URLLC network and evaluate its performance.

1.6. Research questions

i. What is the impact of false predictions on throughput and latency in E-HARQ and on the overall performance of 5G URLLC network?
ii. Can multistage decision maximize achievable throughput and optimize latency in machine learning E-HARQ?
iii. Can the overall performance of 5G URLLC network be improved by the multistage machine learning E-HARQ?
iv. What is the best retransmission scheme for end-to-end performance optimization?

1.7. Methodology

This research aims to maximize the throughput and optimize the latency in the HARQ of the physical layer in order to optimize the overall network performance in 5G URLLC networks. To achieve these study objectives, MATLAB is used to simulate the proposed scheme. Table 1 and 2 present the details of the platform used and the simulation parameters respectively.

Table 1. 1: Proposed simulation platforms

Computer	**Dell DESKTOP O7ODQ8**
RAM	8,00 GB
Processor	Intel (R) core(TM) i5- 10210u CPU @ 1.60GHz, 4 cores, 8 Logical processors
OS	Microsoft Windows 10 pro

Table 1. 2: Proposed simulation parameters

Parameters	Tools
Network Simulator	MATLAB and Jupyter notebooks
Simulation Area	**500m*500m**
Number of nodes	**2 nodes**

In the simulations, the network scenario of two IoT nodes (the sender and the receiver) is considered. In this scenario, the traditional HARQ is used to handle the communication between the sender and the receiver. However, instead of using the simple HARQ, we use the Incremental Redundancy HARQ (IR-HARQ). The choice was influenced by the fact that IR-HARQ has been proven to maximize throughput compared to simple HARQ and the Chase Combining HARQ (CC-HARQ) in [16].

During network communication, the sender first encodes the message and then transmits it to the receiver. At the receiver, the receiver then decodes the message. If the message is decoded unsuccessfully, the receiver attempts to use error correction codes to rectify the packet received with errors. However, if the decoding process fails after error correction, the receiver then sends the NACK requesting the transmitter to retransmit the damaged packet.

If the receiver decodes the message successfully, the transmitter sends the ACK. After the multiple packet transmissions, the data about the results of the decoding process between the sender and the receiver is stored in MATLAB. This data is then used to train the proposed machine learning predictive model so that it can predict the decoding outcome for future transmissions. We then use Jupyter notebooks to train the predictive model based on the data obtained from MATLAB.

When the predicted feedback requires the transmitter to retransmit the packet, the transmitter uses a certain amount of transmission bandwidth during the retransmission. This transmission bandwidth is determined by the CSI which measures the channel condition to mitigate the throughput loss and additional transmission latency that could occur due to false predictions. The use of the different amounts of bandwidth during the transmission and retransmission depending on the CSI is known as the multistage decision.

Furthermore, network performance is evaluated based on different performance metrics such as transmission latency, RTT, throughput, packet error rate and prediction accuracy. To answer the proposed research questions, we compare the performance of the multistage machine learning E-HARQ to other existing schemes such as the machine learning E-HARQ. Although this section does not discuss details such as routing protocols for routing packets, encoding and decoding methods used during transmission, algorithms required for this study and computations such as the calculation of the RTT are discussed later. They are not the main part of the proposed scheme even though they support our scheme.

1.8. Scientific contribution

This study optimizes the throughput and latency of HARQ in 5G URLLC networks. The study also mitigates the throughput loss and reduces latency caused by false alarms and miss-detections. Lastly, the study implements multistage decisions in machine learning E-HARQ.

1.9. Ethical considerations

The study does not require ethical clearance.

1.10. Availability of research infrastructure

Resources required for this study are available from open access data and tools at the University of Limpopo.

1.11. Overview

The remainder of this study is organised as follows: Chapter two reviews related work conducted for improving the throughput and reducing the latency of HARQ and identifies gaps filled by this research. Chapter three discusses the proposed scheme in detail, focusing on all simulation parameters and the necessary steps taken to achieve the objectives proposed in this research. In Chapter four, we discuss the simulation results of the proposed scheme and compare these to the results of existing schemes to answer all questions proposed in this research. Chapter five concludes the study by summarising the research and the gaps that remain for future research.

CHAPTER 2 - LITERATURE REVIEW

2.1 Introduction

The HARQ can be defined as follows: “The HARQ protocol is a combination of forward error correction (FEC) and automatic repeat request (ARQ), and when an error occurs on the receiving side, HARQ stores the error-occurring packet in the buffer and requests packet retransmission” [17]. The 5G URLLC network is largely dependent on this physical layer HARQ protocol for low error-rate transmission and network reliability. However, this protocol suffers from increased transmission latency resulting mainly from the delay time required for channel decoding [5]. To address this problem, we propose a multistage machine learning E-HARQ which uses machine learning algorithms to predict the decoding outcome and send the feedback at an early stage. Moreover, the main purpose of the multistage in our proposed scheme is to mitigate the throughput loss and increased latency caused by a false alarm and miss-detection. A lot has been done to optimize the physical layer HARQ, however, our work must be a contribution to the available literature instead of replicating the work already done. As a result, this chapter mitigates the risk of duplicating the work already done in the throughput and latency optimization of HARQ by reviewing available literature in the throughput and latency optimization of HARQ to identify the gap filled by this study.

2.2 Likelihood ratio based E-HARQ

Authors in [18] proposed an E-HARQ scheme that generates feedback at an early stage using likelihood ratios outputted by the channel modulator. These likelihood ratios are then sent to a block error indicator (BLEI). Furthermore, the likelihood ratios are also used to estimate the uncoded bit error rate (BER). If the uncoded BER is less than the BLEI, it generates and sends the ACK to the transmitter before the decoding process, otherwise, it sends the NACK. Moreover, this scheme uses the turbo decoder for encoding and decoding the input signal. This can be considered as the main advantage of their study because the turbo decoder was proven to boost the HARQ throughput in the study conducted in [19]. For this reason, our study adopts the use of the turbo decoder for encoding and decoding the signal between the transmitter and the receiver to boost the throughput of our proposed scheme. However, the scheme in [18] was evaluated under different modulation techniques and the simulation results showed that it has a lower rate of false-positive (FP) and false-negative (FN) when it is used with 16-Quadrature Amplitude Modulation (QAM). This was because FP and FN rates tend to increase with modulation order [18]. For this reason, we consider the use of 16-QAM in our study to reduce the rate of false alarm and miss-detection in our proposed scheme. The main focus of the study in [18] was on reducing the rate of FP and FN in E-HARQ. We extend their study by proposing a multistage decision to handle the negative impact that could result from FP and FN.

As an extension to the study in [18], authors in [20] evaluated the performance of the likelihood ratio-based E-HARQ scheme proposed in [20]. However, they considered two modes of operation. In the first mode, the E-HARQ transmits only the predicted feedback. In the second mode, the HARQ transmits the predicted feedback and the regular feedback in case of false predictions. Simulation results showed that the first mode is prone to wrong estimates while the second mode increases the latency in terms of false positives. It is clear that in the second mode, the increase in latency was caused by the retransmission of a packet after miss-detection. The main drawback of their study in [20] is that they only prove that miss-detection increases the transmission latency in E-HARQ without proposing a method to address the problem of transmission latency caused by miss-detection in E-HARQ. As an extension to the study in [20], we propose a multistage decision in our scheme to address the increased latency problem caused by miss-detection.

2.3 Subcodes based E-HARQ

Authors in [3] proposed a subcodes E-HARQ (SC E-HARQ) scheme to provide faster feedback and thus enable earlier retransmission. This SC-EHARQ scheme calculates the early feedback from substructures of c codes in the partially received codewords. The main advantage of this SC E-HARQ scheme is that it was able to achieve a latency of less than 1ms which was proposed as the latency requirement for URLLC in [2], [4], [21], [22], [23]. Unlike our proposed scheme and the study in [18], the SC-HARQ scheme uses the low-density parity-check (LDPC) decoder which was outperformed by the turbo decoder in [24]. The SC E-HARQ scheme was evaluated against the likelihood ratio-based E-HARQ scheme proposed in [18]. Simulation results showed that at higher BLER, the likelihood ratio based E-HARQ has better performance in terms of false positives compared to the SC E-HARQ. However, the author justified the underperforming of SC E-HARQ by suggesting that there was strong noise at a low SNR which prohibits codewords from converging. Similar to the study in [18], this study in [3] did not address the miss-detection and false alarm problem although they showed that their scheme is not good enough in reducing the false feedback predictions since it was outperformed by the likelihood ratio based E-HARQ scheme in some cases.

2.4 Superposition coding-based Early HARQ

To mitigate the transmission latency, authors in [25] proposed a superposition coding-based E-HARQ that uses the channel state information (CSI) obtained before the channel decoding process. Unlike ours, this study in [25] uses multiplexing to address the throughput loss problem and increased latency caused by a false alarm and miss-detection as stated in [5], [20], [18], [3]. Furthermore, the early retransmission packet and the initial packet for the next transmission are multiplexed within the same channel using superposition. When the early retransmission is unnecessary, the receiver applies the interference canceller to offset the interference between superposition-coded packets. Thus, this mitigates the throughput loss caused by unnecessary retransmission due to the false alarm.

Consequently, simulation results showed that the superposition coding-based E-HARQ improves the achievable trade-off between throughput and transmission latency. This is because it achieved higher throughput and less transmission latency compared to the conventional method and the ordinary HARQ during the simulation. This study in [25] was building on the study conducted in [18] and [3] by addressing the unresolved throughput loss and increased latency problems resulting from incorrect predictions. Furthermore, it can be concluded that the study in [25] is similar to our study, the main difference being that the study in [25] used CSI and our proposed study uses machine learning for prediction. Another difference is that our proposed study uses multistage decisions while the study in [25] used multiplexing for mitigating the throughput loss and increased latency.

2.5 E-HARQ based on SNR

Authors in [10] proposed an E-HARQ which predicts early feedback by measuring the instantaneous SNR. If the average input SNR is below a certain threshold, the receiver sends the NACK without attempting to decode the message. The use of SNR makes the predictions possible because if the instantaneous SNR is low, there is a high probability of decoding process failure [10]. However, from observation, this scheme has many drawbacks because the receiver does not attempt to decode the message if the predicted feedback is NACK. Generally, it is unrealistic to assume that false estimates or predictions can be avoided [20]. It is clear then that if the predicted NACK is incorrect, the transmitter keeps on retransmitting the negatively acknowledged but decodable packets. This could lead to the unnecessary retransmission of incorrect negatively acknowledged packets that could be successfully decoded in the receiver. Another drawback of the study in [10] is that it compared the performance of the SNR based E-HARQ scheme with Simple Automatic Repeat Request (S-ARQ) [26] only, instead of considering other existing E-HARQ schemes. Simulation results showed that the proposed E-HARQ has less average error probability when the maximum number of retransmissions is higher. Conversely, the S-ARQ appeared to have less average error probability when the maximum number of retransmissions was low. The main focus of the study in [10] was only based on the probability of decoding error. Moreover, the study in [10] is open for extension, more especially for exploring the prediction errors and the throughput of the proposed scheme.

2.6 Mutual Information based EHARQ

The study in [6] proposed a Mutual Information based E-HARQ that uses the mutual information (MI) calculated from the SNR of each data symbol in the received packet and indicates data that can be correctly received through a given channel for the specific packet. This MI-based E-HARQ scheme calculates the mean of MI in the initial packet and if the mean is lower than the predetermined threshold value, early retransmission is requested without waiting for the decoding process because the channel is likely to contain errors. Furthermore, the study in [6] also considered the a priori log-likelihood ratio (LLR) obtained the signal detection (demodulation) process of the initial packet. The LLR method calculates the average of LLR for all coded bits in the received initial packet. If the average LLR is less than the predefined threshold, early retransmission is requested. Simulation results showed that the MI-based EHARQ method achieves a better trade-off between throughput and latency than the LLR-based method. The throughput loss and increased latency resulting from false alarm and miss-detection of this scheme were further addressed by the study in [5].

2.7 Multistage E-HARQ

The study in [5] proposed a multistage E-HARQ to improve the trade-off between latency and throughput in HARQ. However, the in [5] study used the MI-based E-HARQ method proposed in [6]. This method calculates the MI from the SNR of the received packet for generating early feedback. However, authors in [6] did not mitigate the throughput loss and increased latency caused by a false alarm and miss-detection. The study in [5] proposed a multistage decision for the scheme in [6] to mitigate throughput loss and increased latency caused by incorrect predictions. This was done to improve the trade-off between latency and throughput of the scheme proposed in [6]. The multistage decision controls the transmission bandwidth in a multilevel manner, depending on the measured CSI. It sets the number of bits to a low value when the observed channel state is on the verge of reducing the throughput loss when early retransmission is unnecessary. However, when the channel state is very poor, it sets the number of retransmitted bits to a high value to increase the rate of successful decoding process after early retransmission. Simulation results showed that the multistage E-HARQ scheme significantly improved the trade-off between transmission latency and throughput. For this reason, our proposed scheme uses multistage decisions to alleviate the throughput loss and reduce the transmission latency. Unlike the scheme in [5], our proposed scheme uses machine learning algorithms for feedback predictions because they were proven to have high prediction accuracy in [2] and [4].

2.8 Machine learning E-HARQ.

To address the problem of increased transmission latency in HARQ, the authors in [4] proposed a machine learning E-HARQ scheme. This machine learning E-HARQ scheme uses machine learning classification algorithms and the data available after the first few iterations to predict the decoding outcome of a given transmission. The drawback of this scheme is that it requires some data before performing the predictions. Thus, this results in prediction failure for the first transmission because there is no prior data on the first transmission. Simulation results showed that this scheme has high prediction accuracy with less rate of false positives and false negatives. It can be concluded that machine learning algorithms reduce the probability of false alarms and miss-detection. Another drawback of the study in [4] is that they did not propose a way of mitigating the throughput loss and increased latency that could be caused by their scheme in case of false predictions.

The study in [4] focused on one-dimensional input feature as BER estimates in combination with a hard threshold as a classification algorithm, it only considers a single decoder iteration. However, authors in [2] proposed the enhanced machine learning E-HARQ to further extend the study in [4]. They considered more complex classification algorithms with several decoder iterations and history features that leverage information about the channel state from past submissions available at the receiver. Simulation results showed that the enhanced machine learning E-HARQ improves prediction accuracy better than the scheme proposed in [4] and it also reduces the rate of FP and FN. For the sake of prediction accuracy, our proposed study uses machine learning algorithms for feedback prediction. Although the prediction accuracy was improved in study in [2], the main drawback is that the study in [2] did not identify a way of mitigating the throughput loss and increased latency resulting from false predictions. As an extension to the study in [2] and [4], our proposed study addresses the throughput loss and increased latency caused by a false alarm and miss-detection. This is achieved by the use of multistage in our proposed scheme because it was proven to play a significant role in mitigating the throughput loss and increased latency in [5].

2.9 Conclusion

Different researchers have proposed various schemes for reducing transmission latency in the physical layer HARQ by generating feedback at an early stage before the decoding process. However, this chapter showed that most of the proposed schemes do not mitigate throughput loss caused by a false alarm, leaving a gap, more especially for machine learning E-HARQ even though it was proven to have high prediction accuracy. To fill this gap, our proposed scheme uses the multistage decision because it can mitigate the throughput loss better. In tandem, this chapter showed that the multistage decision was never used with machine learning E-HARQ, and this ensures that we are not running the risk of replicating work already available in the literature. This makes a significant contribution to existing studies. The literature review showed that not much attention has been directed to multiplexing for mitigating the throughput loss. For future studies, it would be interesting to propose and evaluate the performance of machine learning E-HARQ that uses multiplexing for mitigating the throughput loss and compare its performance to our proposed scheme.

CHAPTER 3 - METHODOLOGY

3.1 Introduction

This chapter discusses the processes followed in this study. It provides details of the methods employed throughout this study and discusses the simulation tools and simulation parameters used to achieve the proposed objectives addressing the research questions. Furthermore, it also discusses our proposed scheme in detail and the steps taken to implement and evaluate the scheme.

3.2 Simulation environment and parameters

To implement our proposed scheme for performance evaluation, this study simulated the network with two nodes, the sender and the receiver deployed in a 500m*500m network environment. Simulation parameters used for our study are listed in Table 3.1.

Table 3. 1: Simulation Parameters

Simulation Parameters	**Value**
Routing Protocol	AODV
Retransmission protocol	Multistage Machine Learning E-HRQ, machine learning E-HARQ, and the HARQ
Number of nodes	2
Data modulator or demodulator	16-QAM; {QPSK, 16QAM, 64QAM, 256 QAM}
Packet size	{32, 64, 128, …, 33554432 bits}
Data encoder or decoder	Turbo decoder
HARQ type	IR-HARQ
SNR	{-30, -25, -20, …, 0, 5, 10, …, 30}
Network topography	500m*500m
Maximum number of retransmission	2
Transmission range	30m
Radio type	5G New Radio (NR)
Channel type	AWGN
Machine learning models:	Logistic regression, Random forest classifier and support vector classifier
Transmission bandwidth	25, 50, 75, and 100 MHz

The simulation parameters are discussed in the next subsections.

3.2.1 Routing protocol

The ad hoc on-demand distance vector (AODV) routing protocol was proven to have better throughput performance in [27] and [28] compared to the destination sequenced distance vector (DSDV) protocol. Furthermore, the AODV has better performance for real-time applications compared to DSR owing to its reduced latency [27]. Therefore, for these reasons, our study used the AODV protocol for routing the data from source to destination.

3.2.2 Retransmission protocols

To evaluate the performance of our scheme, we compared the performance of the proposed scheme to the performance of existing schemes. This study compared the performance of our proposed multistage machine learning E-HARQ scheme to the performance of machine learning E-HARQ, and the performance of the HARQ. This was done to investigate if our scheme improves the performance of existing retransmission protocols.

3.2.3 Number of nodes

This study simulated the network scenario of two nodes, the sender and the receiver. This was done to collect data required for developing the predictive model from the existing HARQ protocol. We also used this network scenario to evaluate the performance of the proposed scheme.

3.2.4 Data modulator and demodulator

According to the study in [5], data must be modulated before it can be transmitted across the channel as illustrated in Figure 3.1. However, there are different modulation techniques used in communication systems [29]. This study used the 16QAM modulation technique for data modulation. This choice was supported by the fact that the 16QAM was proven to have a lesser rate of FP and FN in [18] compared to the 64-QAM. This is caused by the fact that the rate of FP and the rate of FN tend to increase with modulation order [18]. As a result, the use of 16QAM reduced the rate of false alarms and miss-detection in our proposed scheme.

3.2.5 Data encoder and decoder

The data packets are encoded before they are transmitted across the channel [5], [30], as illustrated in Figure 3.1. However, various decoding and encoding approaches are possible [31], [32]. This includes the use of LDPC and turbo decoder [32]. Both LDPC and turbo decoder efficiently achieve low error rates [5]. However, the turbo decoder boosts the throughput and has a better performance compared to LDPC [24]. For this reason, our study used the turbo encoder and decoder for encoding and decoding packets to enhance the throughput performance of the proposed scheme.

The figure 3.1 shows the transmission of the message from source to destination.

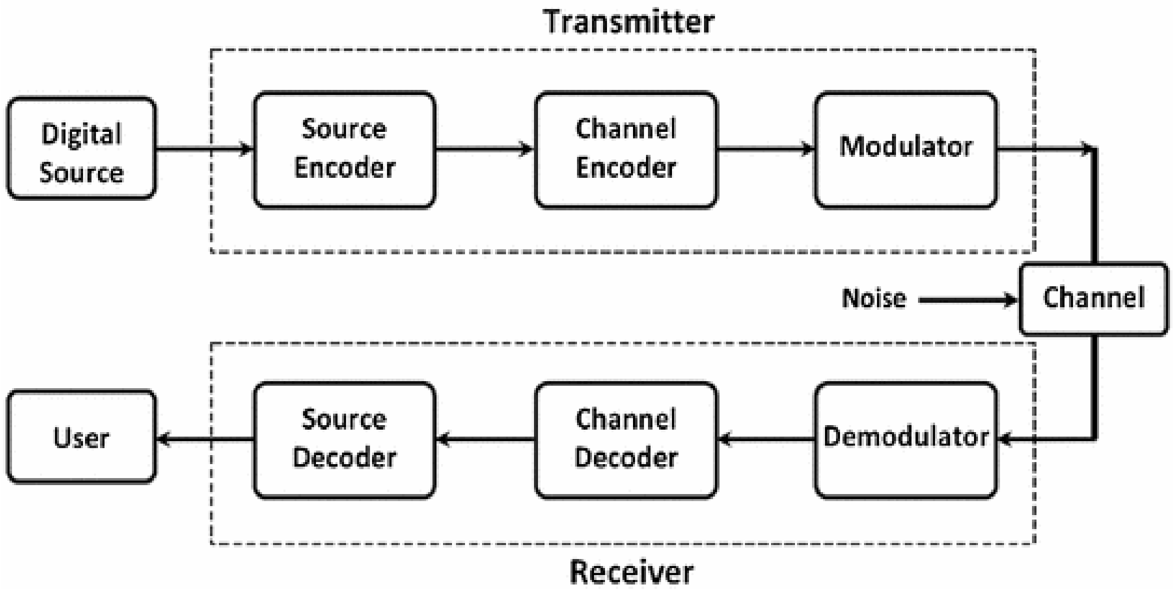


Figure 3. 1: Data transmission between the sender and the receiver

3.2.6 HARQ type

The HARQ is categorized into the simple HARQ, CC-HARQ [33], and the IR-HARQ [34]. The simple HARQ and CC-HARQ are straightforward with vital implications for communication system engineering [35]. However, the IR-HARQ is more powerful and functions more efficiently with the proposed multistage decision method than CC-HARQ [5]. Furthermore, the IR-HARQ achieves higher throughput and has a better performance than the CC-HARQ [36], [33]. For these reasons, our study used the IR-HARQ instead of the simple HARQ and CC-HARQ to improve the throughput performance of the proposed scheme.

3.2.7 Packet size

During the simulation, packets of different sizes were transmitted. This was done to evaluate the performance of the proposed scheme as the packet size increased. This evaluation was significant because the study in [37] showed that packet size has a significant impact on throughput. The study transmitted the packets of different size values of 32, 64, 128, …, 33554432 bits to evaluate the performance of the scheme as the packet size increased.

We considered the packets which larger than 33554432 bits, however, the performance of the computer degraded and generated memory error. It was increased beyond the stipulated size per the standard to effectively investigate the performance of the scheme. We believe that this error was caused by maximum number of bits supported by the radio type and the computer could not handle the bits because it had a limited RAM. However, this problem can still be investigated in the future studies.

3.2.8 Topography

This study used a network topography of 500m*500m to ensure that all nodes are accommodated within the topography. Furthermore, the transmission range is set to 30m, ensuring that all nodes were within transmission range.

3.2.9 Radio type, transmission range and the channel type.

Our study used the 5G new radio (NR) radio type which has a transmission range of 500m. All the nodes in our network are assumed to be within the specified transmission range. Furthermore, using the 5G NR in the physical layer design, accommodate the flexible and scalability which can support many use cases [38]. The study used the additive white gaussian noise (AWGN) channel for adding the noise in the channel because it is widely used with 5G NR and it was the default channel in the MATLAB simulator.

3.2.10 SNR values

The study transmitted packets at different SNR values of -30, -25, -20, …, 30 so that we evaluate, effectively, the performance of the schemes as the channel conditions changed.

3.2.11 Maximum number of retransmissions

The maximum number of retransmissions was set to two in the proposed scheme. This is caused by the fact that latency requirement limits the number of retransmissions in URLLC transmission [39]. Setting the number of retransmissions to two in our proposed scheme allowed the proposed scheme to improve the network performance in terms of latency without degrading the reliability of the network.

3.2.12 Channel bandwidth

Since transmitting the packet from the sender to the receiver requires bandwidth, the proposed scheme used various channel bandwidth values of 50, 75, and 100 MHz to transmit packets from the sender to the receiver in different channel conditions. This allowed the scheme to adjust the bandwidth channel as the noise in the channel changed to support the use of the multistage decision.

3.2.13 Machine learning model

Since the proposed scheme uses the model to predict the packet acknowledgement, we developed three machine learning models, we then selected the best performing model to use for the developing the proposed scheme. The three models developed on the study are: logistic regression, random forest classifier, support vector classifier. We chose these models because they regarded as the most popular and widely used classification models.

3.3 Simulation platforms

This section discusses simulation platforms and tools used for our study as listed in Table 3.2.

Table 3. 2: Simulation platform

Computer	DELL Vostro 153000
RAM	8GB
CPU	Intel(R) Core(TM) i5-10210U CPU
Simulation tools	MATLAB and Jupyter notebooks
Operating System	Windows 10

The study used DELL Vostro with an eight gigabit (8GB) random access memory (RAM) and core i5 processor. Furthermore, the study used MATLAB for simulating the network scenario of two nodes to generate data used to evaluate the performance of the proposed scheme. Moreover, we used Jupyter notebook to create the predictive model using data generated from MATLAB to design the proposed scheme. The MATLAB and Jupyter notebooks were installed on a Windows 10 operating system. The main reason for the study using the simulation platforms listed in Table 3.2 is that they can be used to design the proposed scheme to address the problem of the study. Furthermore, the tools are accessible.

3.4 Processes of the study

Figure 3.2 depicts the process and procedures that were followed to achieve the proposed objectives. In Figure 3.2, we can observe that the first step was to generate data from the existing HARQ retransmission protocol and the last step was to evaluate the performance of the proposed scheme. The next subsections discuss all the steps and processes that were followed to meet the goal of the study as listed in Figure 3.2.

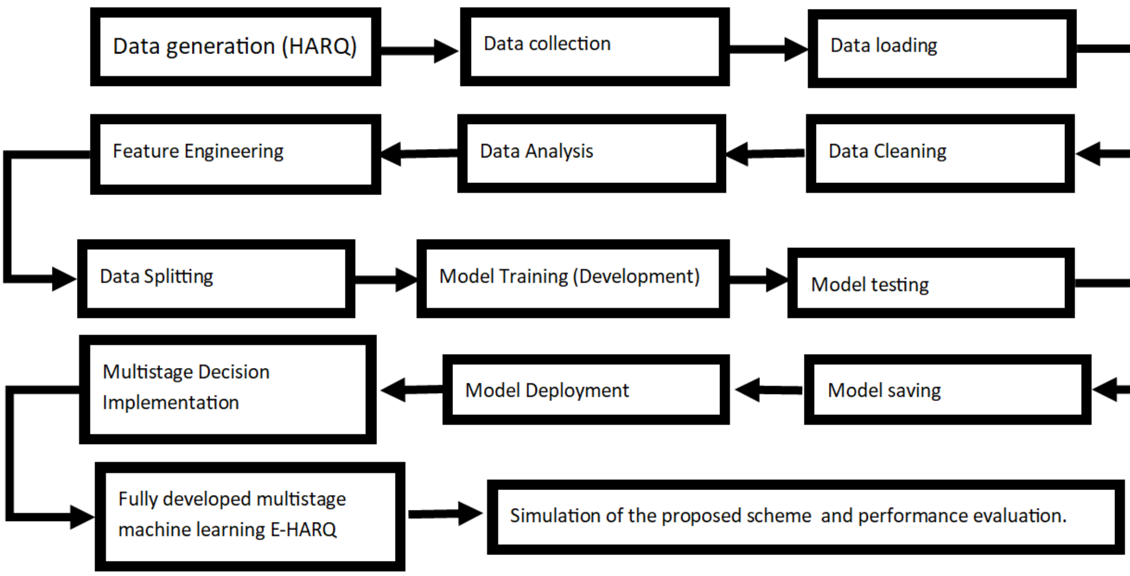


Figure 3. 2: Processes of the study

3.4.1 Data generation

The first step was to generate data required to build the model in order to design the proposed multistage machine learning E-HARQ scheme. The proposed scheme was built from the HARQ and the code for the HARQ was available on the Internet. We simulated the HARQ protocol in MATLAB and transmitted multiple packets from the sender to the receiver to generate the data. The data was then generated during the transmission of packets from the sender to the receiver using the HARQ protocol. To generate a reasonable amount of data, we transmitted 12 000 packets. For each packet transmission, the receiver generated and sent the acknowledgement packet back to the receiver. The generated data was saved in MATLAB variables.

3.4.2 Data collection

After generating the data using the existing HARQ protocol, we collected the generated data from MATLAB variables and stored it in the comma-separated values (CSV) file for building the model. The main reason for saving the data in the CSV file was to build the model in Jupyter notebook and the data stored in a CSV file can be read using the read_csv function in Jupyter notebook [40].

The data generated and collected from the simulations:

- The number of the current packet transmitted
- Whether the packet is the initial transmission or a retransmission of a failed packet
- The number of packet retransmissions
- The size of the packet
- The actual data to be transmitted
- The modulated data
- Encoded data
- The channel noise
- The data generated by the channel
- The demodulated data
- The decoded data
- The check sum value
- The generated packet acknowledgement

3.4.3 Data Loading

Since the data was stored on the CSV file during the data generation process, Python can load the data stored on the CSV file to the Jupyter notebooks for data manipulation. We loaded the data from the CSV file to the Pandas data frame using the read_csv method from the Pandas module in Jupyter notebook using Python. This enabled us to manipulate the data using various python libraries such as numpy, pandas and other libraries required to clean the data and to build the predictive model.

3.4.4 Data Cleaning

Data cleaning is the initial step of any machine learning project [41]. In this stage, we remove data that is incorrect, incomplete, or improperly formatted [42]. The study checked the number of nulls for every column in the data to ensure that the data is clean for all columns. We also checked for duplicated rows and eliminated any data redundancy. We also checked if the data was properly formatted.

3.4.5 Data Analysis

Data analysis is defined as the "process of studying and summarizing data in detail in order to extract useful information" [43]. After the process of data cleaning, the study performed the analysis of the collected data to extract useful information in order to understand this collected data. We performed the data analysis using graphical presentations such as bar graphs and line graphs to investigate the relationship between the features in the collected data. During the data analysis process, we considered the number of transmissions, the number of retransmissions, and the number of initial transmissions. We further investigated the relationship between packet acknowledgement and other features such as packet size, bandwidth, and SNR.

3.4.6 Feature Engineering

Feature engineering "is the process of extracting and generating new features or variables from an existing dataset which helps to improve the performance of machine learning algorithms" [44]. The study performed feature engineering by identifying the new features that can be generated from the collected data features to improve the performance of the model. We changed the data types of some features to reduce memory consumption and to improve performance. Since Jupyter notebook does not allow the models to be built using object type features, we converted object type features to string type features.

3.4.7 Data splitting and preparation

After performing the feature engineering, we prepared the data for building the model. This was achieved by splitting the data into training data which is eighty percent (80%) of the actual data and the remaining twenty percent (20%) of the actual data which was used for testing the predictive model. The process of splitting the data was achieved by using the train_test_split python module. The splitting of data allowed the study to evaluate the performance of the machine learning predictive model on seen and unseen data.

3.4.8 Machine learning model development and training

After splitting the data, we used the sklearn library to develop three machine learning predictive models namely: the logistic regression, random forest classifier, and the support vector classifier. The reason for considering these three machine learning models is that they are widely used in predicting classifications or in categorical outcomes [45]. The main reason for the study developing classification models is that we wanted to create a model that classifies the prediction outcome as either ACK or NACK, therefore, the solution was to develop classification models. Classification models allow the prediction of values that can be classified into two values such as zero (0) or one (1). In the case of our study, the ACK was presented by 1 and the NACK was presented by 0. After developing the three models, we trained the models using the training dataset and the fit method from the Python sklearn module.

For developing the model, we considered the following features:

- Packet size
- Bandwidth
- Number of times retransmitting the packet
- Channel noise

The main reason for considering these features for building the model is that they affect the packet acknowledgment and the study needed to build a model for predicting the packet acknowledgement in order to design the proposed scheme.

3.4.9 Model Testing

Since the data was split into training and testing data, only the training data was used to train the model. This means that the model did not see the testing data. To test the performance of the machine predictive models on unseen data, we used the models to predict the packet acknowledgement of the testing data and compared the predicted acknowledgement with the actual acknowledgement of the testing data. This allowed us to calculate the f1_score of the models on unseen data and hence evaluate the prediction accuracy of the machine learning predictive models. Since the study aims to reduce the latency, we measured the response time of the models and recorded the performance indicators.

3.4.10 Saving the predictive model

We developed and trained three models, namely: the random forest classifier, the logistic regression and the support vector machine classifier. We then saved the best performing model on the pickle file. This was done to ensure that the best performing model is deployed in the HARQ protocol on MATLAB to build the proposed scheme in MATLAB environment.

3.4.11 Deploying the machine learning model on HARQ

After saving the machine learning model on the pickle file, we then deployed it on the HARQ protocol in MATLAB. This was done by importing the pickle file and loading the model on MATLAB as a function that can predict the packet acknowledgement based on specific input features such as SNR. The deployment of the model on the HARQ was achieved by modifying the HARQ protocol code, in the sense that the modified code invokes the machine learning model to predict the packet acknowledgement immediately after demodulating the packet and sending the packet acknowledgement back to the transmitter. This enables the packet acknowledgement to be predicted at an early stage and to reduce latency.

3.4.12 Implementing the multistage decision

This section achieves the third objective of our study. The machine learning model was added to the HARQ, which means that at this stage the HARQ can use the machine learning model to predict the packet acknowledgement and send the acknowledgement to the transmitter. However, it is possible that the predicted packet acknowledgement could be incorrect [20]. The correctness of the predicted packet acknowledgement depends on the prediction accuracy of the machine learning predictive model. We then focused on reducing the throughput loss and increased latency that could occur due to false alarm and miss-detection resulting from incorrect predictions [25]. This was achieved by adding the multistage decision and the machine learning model in the HARQ protocol on MATLAB to develop the proposed scheme.

The multistage decision was implemented in the transmitter. If the predicted feedback requires the transmitter to retransmit, the transmitter determines the channel condition before transmitting. The transmitter achieves this by determining the SNR before retransmitting the packet [5]. In the multistage decision method, the transmitter retransmits the packet using either the low bandwidth channel or the high bandwidth channel depending on the channel condition [5]. If the channel condition is poor, the transmitter retransmits the negatively acknowledged packet using a channel with higher bandwidth to increase the likelihood of successful transmissions which results in improved throughput. If the channel condition is good, the receiver uses a low bandwidth channel, to mitigate the throughput loss resulting from incorrect predictions of the acknowledgements. The method of adjusting the bandwidth channel is known as the multistage decision. The multistage decision was added using the if-else statements in the code of the HARQ protocol in MATLAB on the receiver end.

3.4.13 Fully Proposed Multistage Machine Learning E-HARQ

After implementing the multistage decision on the machine learning E-HARQ, we are finalised with developing the proposed scheme. This section discusses the way the proposed scheme works during the communication of nodes within the network. For simplicity, we consider the communication between the sender and the receiver in a peer-to-peer (device-to-device) network communication. Figure 3.1 presents the communication between the sender and the receiver using the proposed multistage machine learning E-HARQ. The blue solid lines represent the transmission of the message from the transmitter to the receiver through the communication channel. Furthermore, the feedback (ACK/NACK) is presented by the dotted lines.

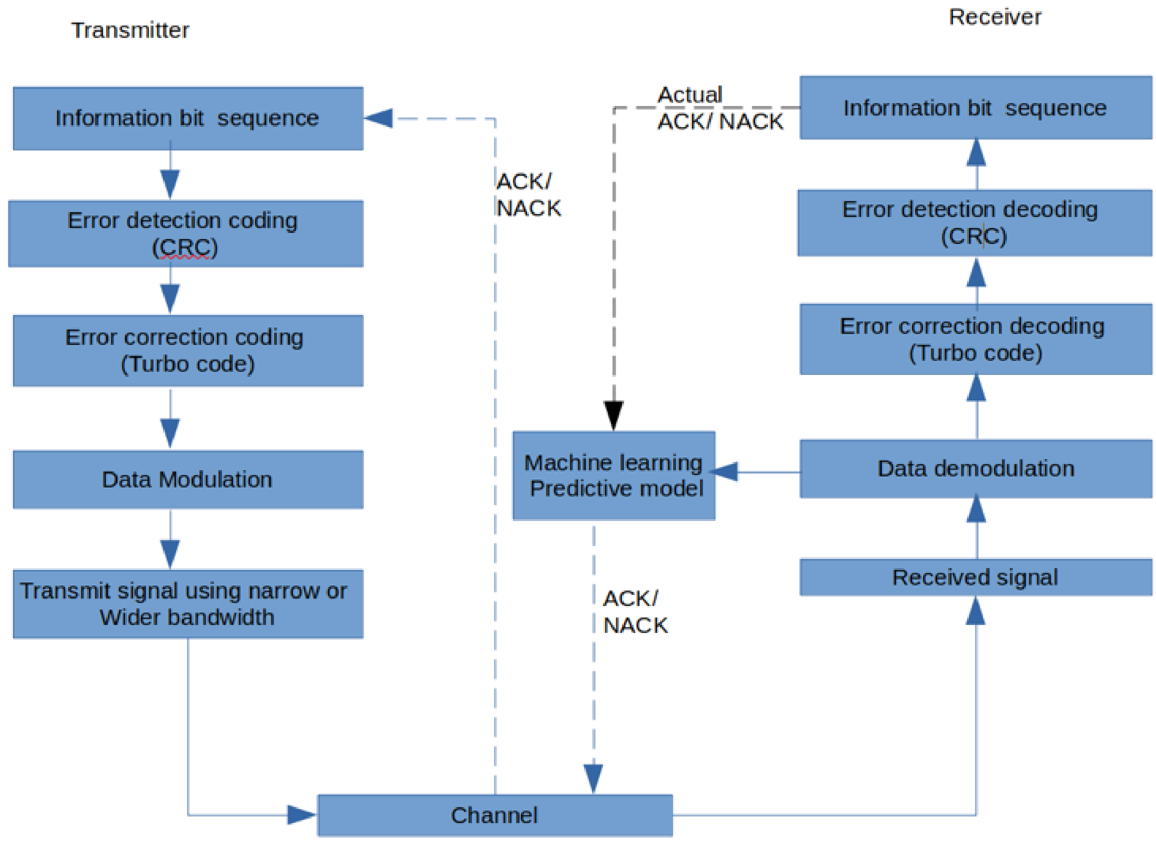


Figure 3. 3: Block diagram of the proposed multistage machine learning E-HARQ scheme

Transmitter:

i. Transmits the message in the form of an information bit sequence from the transmitter.
ii. Adds the cyclic redundancy check (CRC) value for error detection.
iii. Encodes the message using the turbo decoder for error correction.
iv. Modulates the message.
v. Transmits the message using the default bandwidth.
vi. Waits for the acknowledgement of the message.
vii. After receiving the predicted feedback, if the received feedback is NACK, requesting retransmission, the transmitter senses the channel. If the channel condition is poor, it retransmits the message using the high channel bandwidth otherwise it uses a low bandwidth channel. This process is known as the multistage decision as proposed by the study.

Receiver:

i. Receives the message in the form of bits.
ii. Modulates the received message.
iii. Calls the machine learning predictive model stored on the pickle file to predict the feedback (ACK or NACK) of the decoding outcome and pass the message to the turbo decoder.
iv. Sends the predicted feedback or packet acknowledgement of the message to the transmitter.
v. The turbo decoder attempts to decode the message and pass it on to the CRC to detect if there any error occurred during the decoding process and compares the acquired feedback with the predicted feedback. If the predicted feedback is not the same as the actual feedback, the actual feedback is sent to the transmitter.

3.4.14 Performance evaluation

After designing the proposed scheme, we evaluated its performance. This section discusses the evaluation of the performance of the proposed scheme to achieve the second, fourth, and fifth objectives proposed in this study. The proposed scheme was evaluated based on the following performance metrics: transmission latency, RTT, throughput, packet error rate, and prediction accuracy. The performance metrics are discussed in the next subsection.

I. Transmission latency

Transmission latency is defined as the time it takes for a packet to travel from the source to the destination [46]. We measured the time it takes to complete the packet transmission from the sender to the receiver. We evaluated the latency performance of the proposed scheme by transmitting packets of different sizes using different SNR values. We transmitted the packets of 32, 64, 128, …, 33554432 bits using the SNR values of -10, -5, 0, 5, and 10. We also evaluated the transmission latency performance of the proposed scheme.

II. Round trip time

The round trip time is the combination of the time it takes for sending the packet and the time it takes to receive the feedback [47]. The study evaluated the RTT by measuring the time it takes for the sender to receive the packet acknowledgement after sending packets of different sizes. We transmitted the packets with size ranges of 32, 64, 128, …, 33554432.

III. Throughput

Throughput is the rate of error-free packets transmitted from the source to the destination at a given time [48]. The study evaluated the throughput by measuring the percentages of the bits received without errors after transmitting the packet at different SNR values. We used SNR values of -30, -25, -20, …, 30. The study calculated the percentage throughput by measuring the total received bits divided by the total transmitted bits and multiplying it by 100. We evaluated the SNR values in different SNR values to observe the throughput performance of the proposed scheme as the channel conditions changed.

IV. Packet error rate

The packet error rate is the number of incorrectly received data packets divided by the total number of received packets [49]. The study evaluated the percentage of packet error rate by calculating the number of unsuccessful bits divided by the total number of transmitted bits and then multiplying the results by 100. We then evaluated the error rate by transmitting the packet at SNR values of -30, -25, -20, …, 30. We used different SNR values to evaluate the performance of the proposed scheme as the channel conditions changed.

V. Prediction accuracy

The prediction accuracy is the percentage of correct predictions the model has made in the test dataset [50]. Since the proposed scheme uses the model to predict the packet acknowledgement, we evaluated the prediction accuracy of the model by measuring the f1_score of the developed models. We further evaluated the response time of the models to observe their latency performance.

3.5 Evaluated schemes

This section discusses the schemes that were evaluated by the study. The study compared the proposed scheme with the existing HARQ and the machine learning E-HARQ scheme.

3.5.1 HARQ Protocol

The proposed scheme is built from the existing HARQ protocol, and then the study compared the proposed scheme with the HARQ protocol. The code for the existing HARQ protocol is available in the Matlab website. We evaluated the network performance of the existing HARQ protocol and compared it with the performance of our proposed scheme.

3.5.2 Machine learning E-HARQ

The code for the machine learning E-HARQ scheme proposed in [4] is not available. Therefore, this study used the HARQ to develop the model and combined the model with the HARQ protocol to form the machine learning E-HARQ scheme proposed in [4]. However, this machine learning E-HARQ scheme proposed in [4] was not evaluated in terms of the network performance by the study in [4]. The study in [4] evaluated the accuracy of the model without deploying the machine learning E-HARQ scheme on the network. However, since we wanted to compare the network performance of our proposed scheme with the machine learning E-HARQ scheme proposed in [4], we deployed the machine learning E-HARQ scheme on the network and evaluated its network performance and then compared its performance with the performance of our proposed scheme.

3.5.3 Multistage machine learning E-HARQ

The multistage machine learning E-HARQ is the scheme proposed by our study. We built our proposed scheme from the existing HARQ protocol. The code for the HARQ protocol is available on Matlab website. We used the HARQ code to generate and collect the data. We then used the data to build the machine learning model. We combined the model with the HARQ protocol and integrated it with the multistage decision to form the proposed multistage machine learning E-HARQ. We then deployed the proposed scheme on the network and evaluated its performance. The proposed multistage machine learning E-HARQ was discussed in Section 3.4.13.

3.6 Conclusion

This chapter discussed all the steps and tools that were required to achieve the proposed objectives and to address the proposed research questions. We also discussed the simulation parameters and simulation tools required to achieve the goal of the study. We discussed the proposed scheme in detail and all the procedures and steps that were followed to build the proposed scheme. All the tools that were required to build the proposed scheme were available and accessible. As a result, the objectives of the study were met and the problem of the study was addressed.

CHAPTER 4 - RESULTS

4.1 Introduction

This chapter presents the findings of this study. Furthermore, we present the comparative performance results of the proposed scheme and compare these with the performance of existing schemes such as HARQ and machine learning E-HARQ. The performance is evaluated based on throughput, error rate, RTT, and latency to answer the research questions.

4.2 Data collection

For the study to achieve the research objectives and answer the research questions, we needed to develop the proposed scheme. Since the proposed scheme uses a machine learning model and multistage decision to predict the feedback of packet decoding before it is completed and to mitigate the throughput loss resulting from incorrect predictions, the first step was to build a machine learning model for predicting the feedback. We simulated the network with two nodes (the sender and the receiver) to collect data required to build the machine-learning models.

These nodes made multiple transmissions using the existing hybrid automatic repeat request (HARQ) retransmission protocol to generate enough data for building the model. This data was stored in MATLAB in a CSV file. Since the study uses the Jupyter notebook for building the model, the data was loaded from the CSV file to the Pandas' data frame in the Jupyter notebook. The data collected during the simulation for building the model is shown in Figures 4.1 and 4.2, after loading it into the Jupyter notebook.

'num_of_transmission'	'Initial_transmission'	'retransmission'	'number_of_retransmission'	'transport_block_size'	'bits_to_transmit'	'coded_data'
1	1	0	0	32	'[1 1 0 1 1 0 0 1 1 1 0 1 1 0 1 0 0 1 1 1 1 0...	"'[1 0 1 0 1 0 0 1 1 1 1 0 1 1 1 0 0 1 0 0 1 ...
2	1	0	0	32	'[1 1 1 0 0 0 0 1 0 1 1 1 0 0 0 1 1 0 0 0 1 1 ...	"'[0 0 1 0 1 1 1 1 0 0 1 1 0 0 1 0 0 1 0 1 0 ...
3	1	0	0	32	'[1 0 0 0 0 1 0 1 0 0 0 1 1 1 1 1 1 0 0 1 1 1 ...	"'[0 0 1 0 1 1 1 1 0 1 0 0 1 1 0 0 0 1 1 0 0 ...
3	0	1	1	32	'[1 0 0 0 0 1 0 1 0 0 0 1 1 1 1 1 1 0 0 1 1 1 ...	"'[1 0 1 1 0 0 1 1 1 1 0 1 0 0 0 0 0 0 0 1 0 ...
3	0	1	2	32	'[1 0 0 0 0 1 0 1 0 0 0 1 1 1 1 1 1 0 0 1 1 1 ...	"'[1 1 1 1 0 0 0 1 1 1 1 0 1 1 0 1 0 0 0 0 0 ...

Figure 4 1: Data collected for model building from the first simulation of the HARQ

'modulated_data'	'SNR'	'channel_data'	'demodulated_data'	'decoded_data'	'block_crc'	'Acknoledgement'	'ACK/NACK'
"'[0.9487+0.3162i -0.9487+0.3162i -0.3162-0.9...	-11.153851	"'[-1.1186+0.0784i -8.4676+4.2393i 3.357-2.42...	"'[-1.11856999500713 -0.0783578035969243 0.48...	'[1 1 0 1 1 0 0 1 1 1 0 1 1 0 1 0 0 1 1 1 1 0 ...	False	1	'ACK'
"'[-0.9487+0.3162i -0.3162-0.3162i 0.3162+0.3...	-2.137236	"'[-0.5844+0.1459i -1.3557-1.726i 0.1223-0.28...	"'[-0.584367570459123 -0.145923308212743 -0.0...	'[1 1 1 0 0 0 0 1 0 1 1 1 0 0 0 1 1 0 0 0 1 1 ...	False	1	'ACK'
"'[-0.9487+0.3162i -0.3162-0.3162i 0.9487-0.9...	-21.205587	"'[-0.9688+2.1498i -3.491-8.8376i -8.5119+3.8...	"'[-0.968800179363594 -2.1498137950507 0.3363...	'[1 0 1 0 1 1 0 1 0 1 0 1 1 1 1 1 1 0 0 0 1 0 ...	True	0	'NACK'
"'[0.9487+0.9487i 0.3162+0.3162i -0.9487-0.31...	-21.205587	"'[-11.3262+9.5143i 2.7073-13.3458i 7.908-5.5...	"'[-11.3261651698457 -9.51426725704367 10.693...	'[1 0 0 0 0 1 0 1 0 1 1 1 1 1 1 1 0 0 1 1 1 0 ...	True	0	'NACK'
"'[0.9487-0.9487i 0.9487+0.3162i -0.3162-0.94...	-21.205587	"'[-5.5593-3.0496i 10.1895+8.0987i -9.0316-3....	"'[-5.55926575924465 3.049574135715 4.9268102...	'[1 0 0 0 0 1 0 1 0 0 0 0 0 1 1 0 1 1 0 1 1 1 ...	True	0	'NACK'

Figure 4 2: Data collected for model building from the first simulation of the HARQ

From Figure 4.1 and Figure 4.2 above, we can see that our data contains the following features or columns:

- 'num_of_transmission': indicates the number of packets to be transmitted or the number of current transmissions.
- 'initial transmission': indicates whether the current transmission is the initial transmission or the retransmission of the failed transmission. One (1) indicates that the current transmission is the initial transmission and zero (0) indicates that the current transmission is not the initial transmission.

- ‘retransmission’: indicates whether the current transmission is the retransmission of the failed packet or not. 1 indicates the retransmission of failed packet and 0 indicates that the current transmission is not the retransmission of the failed packet.
- ‘number_of_retransmission’: indicates the number of times the packet is being retransmitted.
- ‘transport_block_size’: indicates the size of the data to be transmitted in bits.
- ‘bits_to_transmit’: indicates the actual data to be transmitted in binary form. This data is passed to the turbo encoder.
- ‘coded_data’: is the encoded data outputted by the turbo encoder. This data is then passed on to the 16-QAM modulator.
- ‘Modulated_data’: is modulated data, outputted by the 16-QAM modulator. This data was then passed to the channel for transmission.
- ‘SNR’: is the value of the channel noise. This is a randomly generated value.
- ‘Channel_data’: data generated by the channel after being transmitted to the receiver. This data contains some noise because the channel had noise during the transmission of data.
- ‘demodulated_data’: presents the data outputted by the demodulator, after demodulating the channel data. This data is then passed on to the turbo decoder.
- ‘decoded_data’: represents the decoded data that is generated by the turbo decoder after receiving the demodulated data.
- ‘block_crc’: indicates if there is any erroneous bit on the received data. Presented by true if an error occurred, otherwise false.
- ‘Acknowledgment’: presents the acknowledgement of the transmission, 0 presents an unsuccessful transmission, and 1 presents the successful transmission.
- ‘ACK/NACK’: is the acknowledgement of the transmission that is sent back to the receiver, ACK presents the positive acknowledgement and NACK presents the negative acknowledgement.

4.3 Data Analysis

Data analysis “is the process of studying and summarising data in detail to extract useful information, that is, collecting, sorting, processing and analysing data” [51]. Therefore, for us to understand our data and the relationship between features, we performed data analysis. In the next sections, we analyse the data collected during the first simulation of the HARQ.

4.3.1 Transmissions

Figure 4.3 shows the results of the total transmissions made during the first simulation using the existing HARQ protocol.

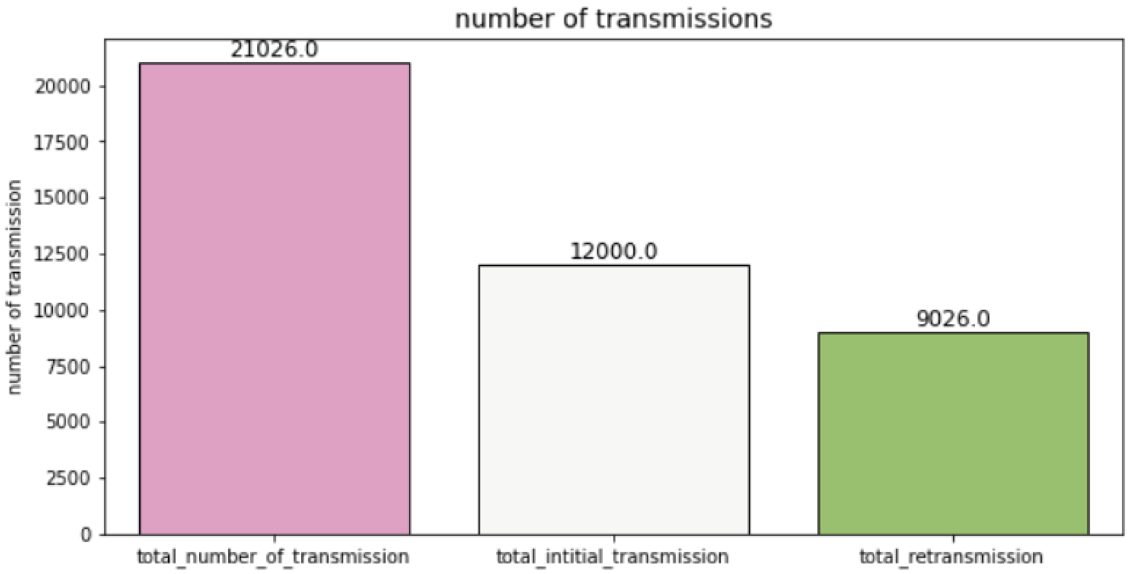


Figure 4 3: Analysis of the transmissions in HARQ

In Figure 4.3, we can see that the total number of transmissions made during the first simulation of HARQ was 21 026. This includes the number of retransmissions made during the simulation. However, during the first simulation of HARQ, the total number of transmissions made was 12 000, excluding the retransmissions. This is the total number of packets transmitted during the first simulation of the HARQ protocol. However, due to transmission failure, some packets were retransmitted and the total number of retransmissions made during the first simulation of HARQ was 9 026

4.3.2 Initial transmissions

We further analysed the initial transmissions of the data collected from the first simulation of the HARQ protocol. In Figure 4.4, we can see that only 8 318 packets were successfully transmitted during the first transmissions. However, only 3 682 out of 12 000 packets failed to be retransmitted successfully in the first retransmission. These packets had to be retransmitted when they were not delivered on first attempt. Some packets were retransmitted more than once. This means that the probability of the packet being transmitted successfully in the first transmission was very high.

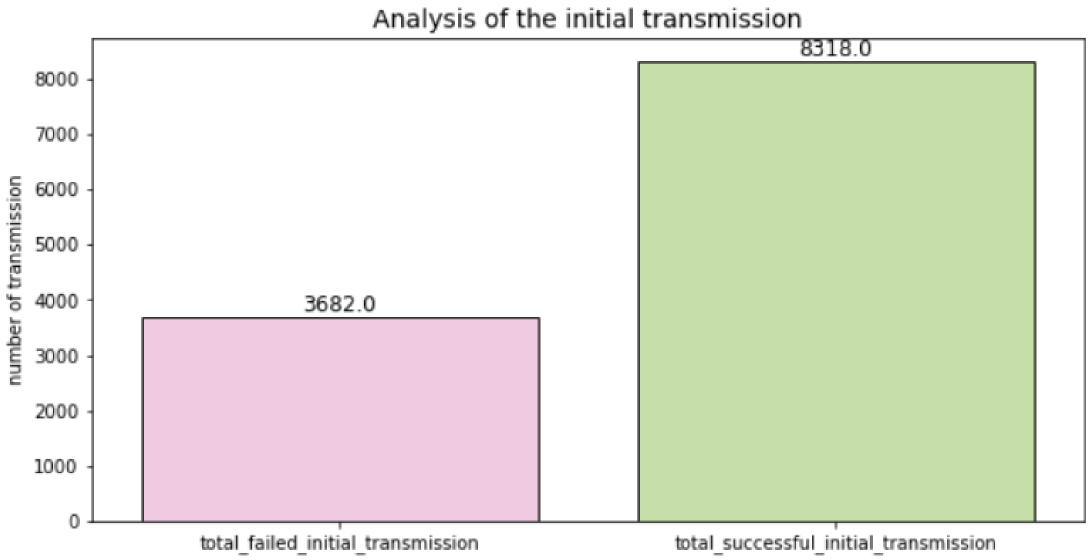


Figure 4 4: Analysis of initial transmissions in HARQ

4.3.3 Retransmissions

In this section, we have the evaluated retransmitted packets during the first simulation of HARQ, using the collected data. In Figure 4.5, we can observe that the packets successfully retransmitted during the first simulation of HARQ are fewer than the packets that were unsuccessfully retransmitted.

This means that the probability of the packet being retransmitted successfully is less than the probability of it being retransmitted unsuccessfully. In other words, there is a high chance of the packet being retransmitted unsuccessfully during the retransmission of failed transmissions. This is caused by the setup of the HARQ scheme that we used for data collection. In this setup, the noise in the channel was randomly generated during the transmission of the new packet. However, retransmission was transmitted at the same channel noise as the initial transmission of that packet. This decreases the chances of having successful retransmissions, as shown in Figure 4.5.

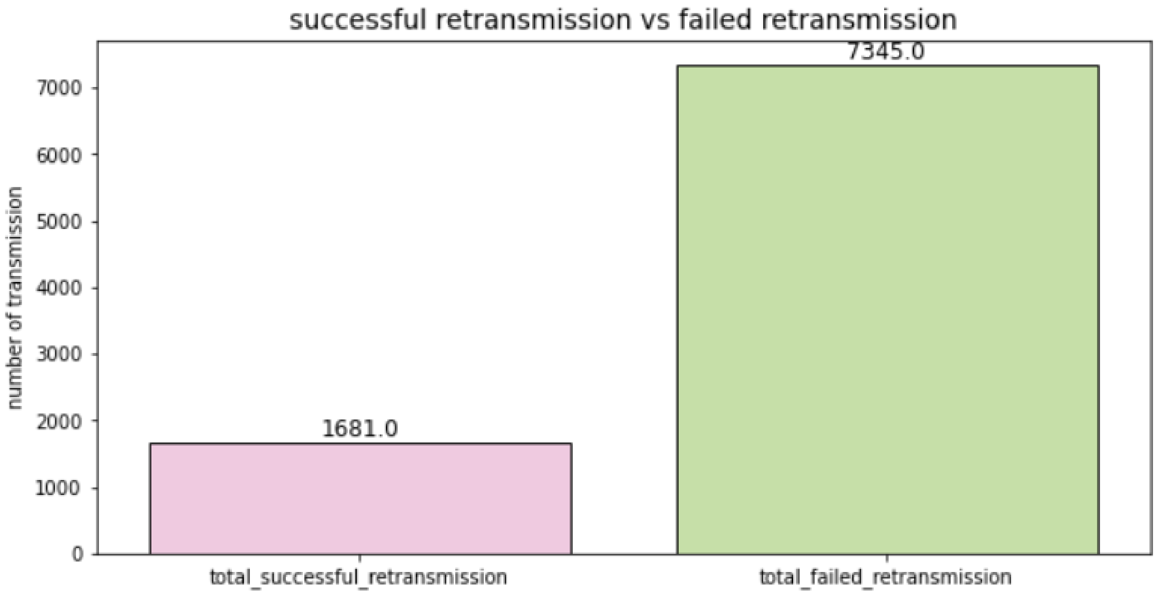


Figure 4 5: Analysis of the retransmissions of the first simulation of HARQ

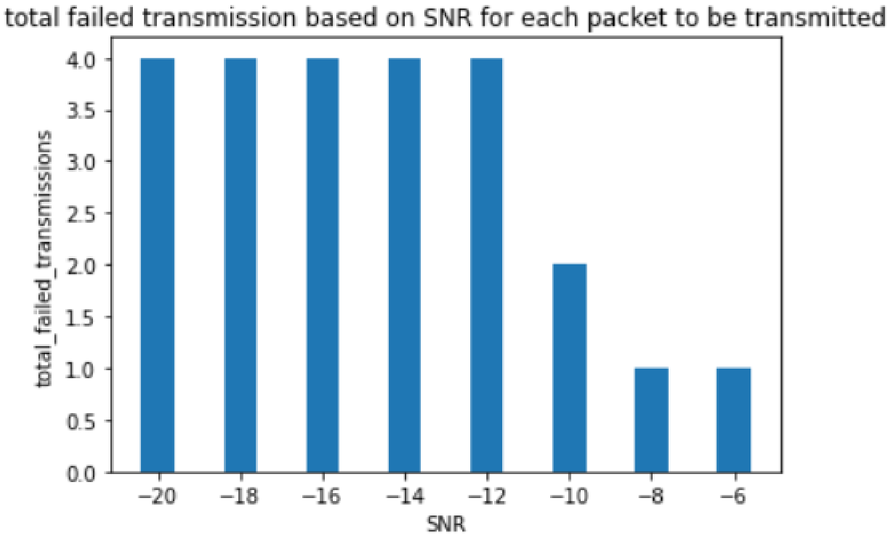
4.4 Relationships between features.

To select appropriate features for building the model, we considered the relationship between features and the response variable, the feedback.

4.4.1 SNR and the feedback

To understand the relationship between the SNR and the generated acknowledgement of the transmission, we then performed the second simulation of HARQ. In the second simulation of HARQ, we kept all parameters constant except the SNR. We used the SNR values of -20, -18, -16, …, 20. Figure 4.6, shows the analysis of the data generated during the second simulation of HARQ.

In Figure 4.6, we can observe that we have more transmission failures at the lower SNR. However, as the SNR increases, the number of failures decreases. This means that at lower SNR, we are likely to receive a negative acknowledgement while at higher SNR we are likely to receive a positive acknowledgement. This is caused by the fact that at lower SNR there is a lot of noise than signal strength and at higher SNR tthere is more signal strength than the noise on the channel [52], [53], [54], [55], [56]. It is clear that the SNR is one of the features influencing feedback. Hence, it must be used for predicting the acknowledgement of the transmission.



<Figure size 216x216 with 0 Axes>

Figure 4 6: Relationship between SNR and the feedback in HARQ

4.4.2 Packet size and the feedback

To analyse the relationship between feedback and packet size, the study performed the third simulation of the HARQ protocol. In this third simulation of HARQ, all parameters were kept constant, except the packet size. We used the packet size values of 400, 600, 800, …, and 1200. The analysis of the data generated during the third simulation of HARQ is shown in Figure 4.7. In Figure 4.7, we can see that as the packet size increases, the number of transmission failures increases. This means that we are likely to receive a negative acknowledgement when we transmit larger packets.

On the other hand, shorter packets are likely to yield positive acknowledgement. From the simulations, it was noted that the cause of this is that larger packets require more bandwidth to be transmitted successfully. So this failure occurred because all parameters were kept constant, except the packet size. This means that there will be inadequate bandwidth for transmitting larger packets since all parameters were kept constant, including the bandwidth. Therefore, the packet size is one of the factors influencing the acknowledgement of the packet. Hence, the packet size must be included as one of the features to predict the feedback.

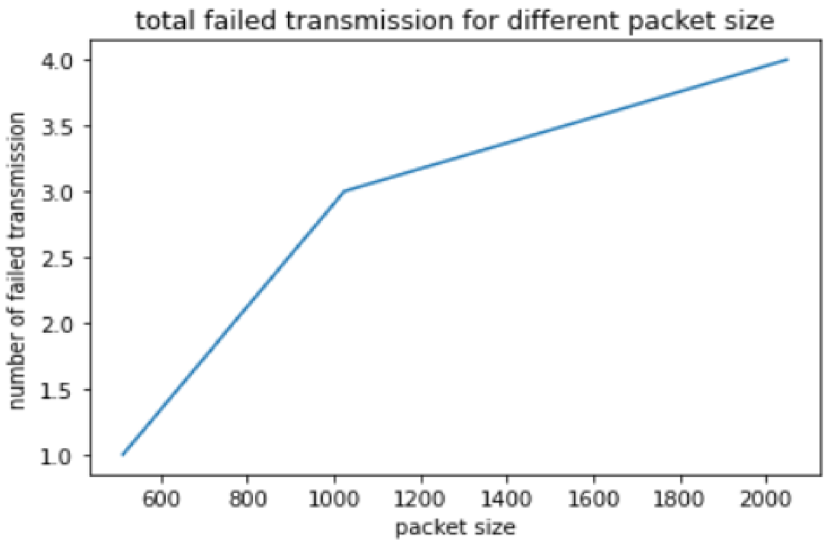


Figure 4 7: The relationship between the packet size and the feedback in HARQ

4.4.3 Relationship between the bandwidth and the feedback

To understand the relationship between the bandwidth and the feedback, the fourth simulation of HARQ was conducted. In this fourth simulation of HARQ, all the parameters were kept constant, except the bandwidth. We used the bandwidth values of 5, 10, 15, …, and 100.

The analysis of the data generated from the simulation is depicted in Figure 4.8. In Figure 4.8, it is clear that the number of failed transmissions decreased as the bandwidth increased. This means that the probability of transmission failure is higher at the lower bandwidth. This is caused by the fact that every feature, except the bandwidth, was kept constant. So decreasing the bandwidth without reducing the packet size is likely to lead to a transmission failure. Clearly, the bandwidth is one of the features influencing the acknowledgement of the packets. Hence, the bandwidth must be included in predicting the packet acknowledgement.

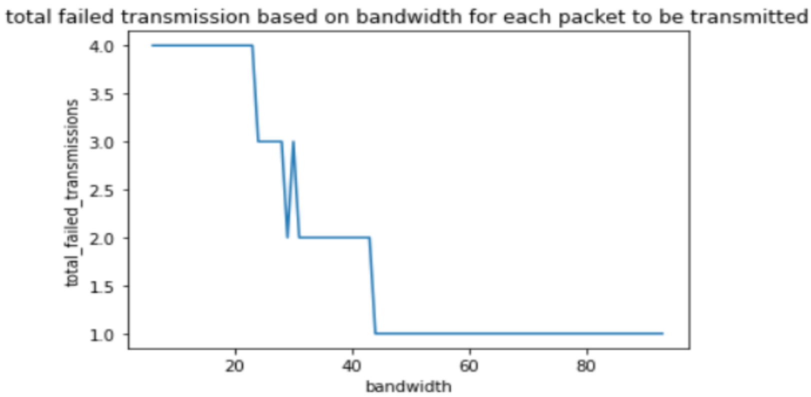


Figure 4 8 Relationship between the bandwidth and the feedback in HARQ

4.5 Data cleaning

Data cleaning is important for developing an efficient machine learning model [57]. This study evaluated the data by calculating the total number of nulls in each column feature to ensure that it is clean before building the model. This was achieved by calculating the number of nulls for every column and the results are shown in Figure 4.9. From the results, it is clear that data for all columns is clean because there is no column with null values. This means that the data was already clean, nullifying the need to clean the data.

```
df.isnull().sum()
```

```
'num_of_transmission'           0
'Initial_transmission'          0
'retransmission'                0
'number_of_retransmission'      0
'transport_block_size'          0
'bits_to_transmit'              0
'coded_data'                    0
'modulated_data'                0
'SNR'                           0
'channel_data'                  0
'demodulated_data'              0
'decoded_data'                  0
'block_crc'                     0
'num_of_resource_blocks'        0
'transmission_bandwidth'        0
'Acknoledgement'                0
'ACK/NACK'                      0
dtype: int64
```

Figure 4 9: Data cleaning

4.6 Data preparation

After over-riding the data cleaning process, the data was prepared for building of the model. In this section, features required for building the model were selected. The selected features are “SNR”, “bandwidth”, “transport_block_size”, and the “number_of_retransmission” to predict the feedback. The main reason for selecting or considering these features is that they have a strong relationship with the feedback as was shown in the Section 4.4. Hence these features are appropriate for predicting the feedback. After selecting the features, the dataset was divided into testing and training datasets. Eighty percent (80%) of the entire dataset was used for training while the remaining twenty percent (20%) was used for testing.

4.7 Model building and performance evaluation

The feedback is in the form of 0 or 1, where 1 represents a positive acknowledgment and 0 presents a negative acknowledgment. So for building the proposed scheme, there was a need to build a binary classification model to predict the binary feedback (in the form of 0 or 1). The study developed three models: logistic regression, random forest, and the support vector machine (SVM).

The three models were trained using the training dataset. After training the models, their performance was evaluated based on the f1_score. The main reason for considering the f1_score is that it is mostly used for measuring the prediction accuracy of classification models [58]. To measure the performance of the models, we predicted the feedback using the test data. Then, the f1_score was measured using the predicted feedback and the actual feedback of the testing data.

The performance results are shown in Figure 4.10. In Figure 4.10, the random forest has the highest f1_score compared to the SVM and the logistic regression. This means that it has the highest prediction accuracy compared to the SVM and the logistic regression. However, the study used the logistic regression model to build the proposed scheme.

This is because the study proposed to add a multistage decision to the scheme that uses the logistic regression model and the logistic regression has a shorter response time compared to other models as shown in Figure 4.11. However, this opens a future research gap for developing a scheme that uses the random forest model since it has higher prediction accuracy (f1_score). In the previous schemes (machine learning E-

HARQ [4]), the logistic regression was considered because of its simplicity and performance [4]. Our study integrated the logistic regression model with the multistage decision.

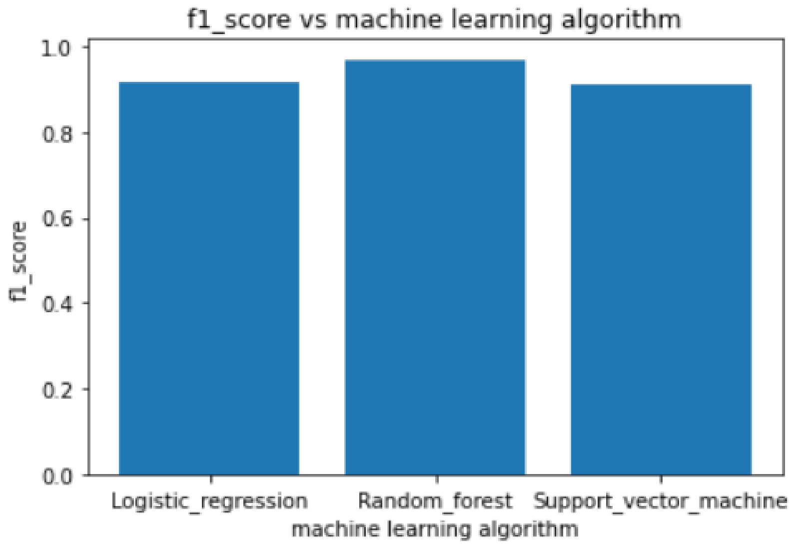


Figure 4 10: Model performance

Figure 4.11 shows the response time of the developed models. In Figure 4.11, we can observe that the logistic regression has a shorter response time compared to the random forest and the support vector machine. This implies that the logistic regression has reduced latency compared to the random forest and the support vector machine. Therefore, for this reason, our scheme uses logistic regression to predict the acknowledgement of the transmission.

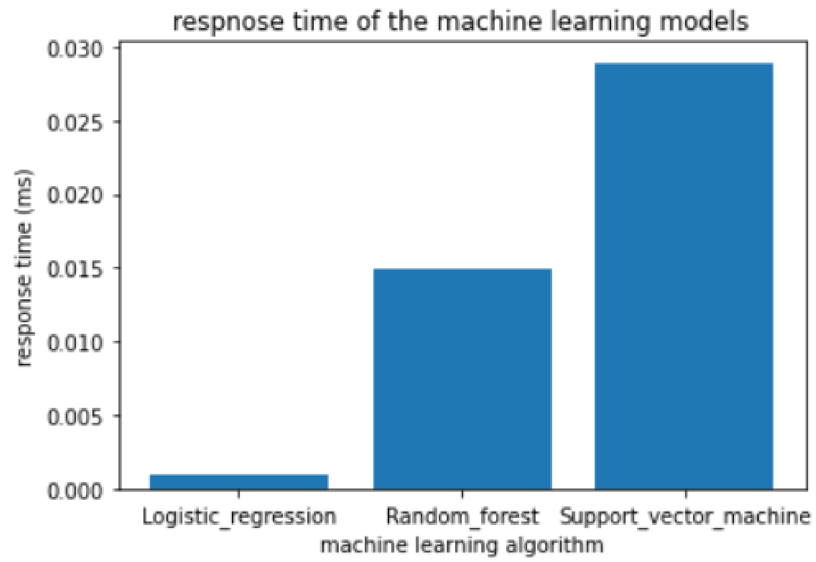


Figure 4 11: Model response time performance

4.8 Model testing

The model was tested to verify if it can predict the feedback for a single transmission before being deployed. This was done to make sure that it would be able to predict the feedback for each transmission in the HARQ retransmission protocol. To perform this test, the following parameters were used:

i. Num_of_retransmission=0
ii. SNR=3
iii. Packet_size=1024 bits
iv. Badwidth= 100 MHz

Then the logistic regression model predicted a value 1, which presents the positive ACK as shown in Figure 4.12. After this test, the model was saved in the pickle file and thereafter it was deployed into MATLAB since it worked efficiently in the testing phase.

```
#call the linear regression model to make predicitons
print("feedback of the current transmission is : ",lr.predict([[0,3,1024,100]])[0])

feedback of the current transmission is :  1
```

Figure 4 12: Model testing for deployment

4.9 Development of the proposed scheme

After deploying the model in the HARQ, it was then integrated with the multistage decision resulting in a fully developed multistage machine learning E-HARQ scheme. In the multistage decision, we defined different bandwidths using 'if-else' statements. In the fully developed multistage machine learning E-HARQ scheme, the transmitter selects the appropriate bandwidth depending on the channel conditions (channel noise). This was done to improve the throughput, especially in poor channel conditions [5]. However, the model on the receiver predicts the feedback to reduce latency. The simulation results of the proposed scheme are shown in the next Section 4.10.

4.10 Simulation results of the transmissions and the retransmissions

In Section 4.3, we performed data analysis of the existing HARQ protocol. This section compares the analysis of the HARQ that was presented in section 4.3 with the analysis of the proposed multistage machine learning E-HARQ scheme. The main reason for presenting the results in different sections is that the results presented in section 4.3 focused on understanding the data for building the proposed scheme. However, this section focuses on comparing the performance of the proposed scheme to the performance of the HARQ protocol to find the best performing scheme between the HARQ and the proposed scheme. The analysis of the HARQ and the analysis of the proposed scheme are combined into single graphical presentations for better comparisons of the two schemes.

4.10.1 Number of transmissions

Figure 4.13 shows the total number of transmissions, the total number of retransmissions, and the number of packets transmitted by the HARQ and the proposed multistage machine learning E-HARQ scheme. The HARQ is denoted by blue bars and the proposed multistage machine learning E-HARQ is denoted by orange bars.

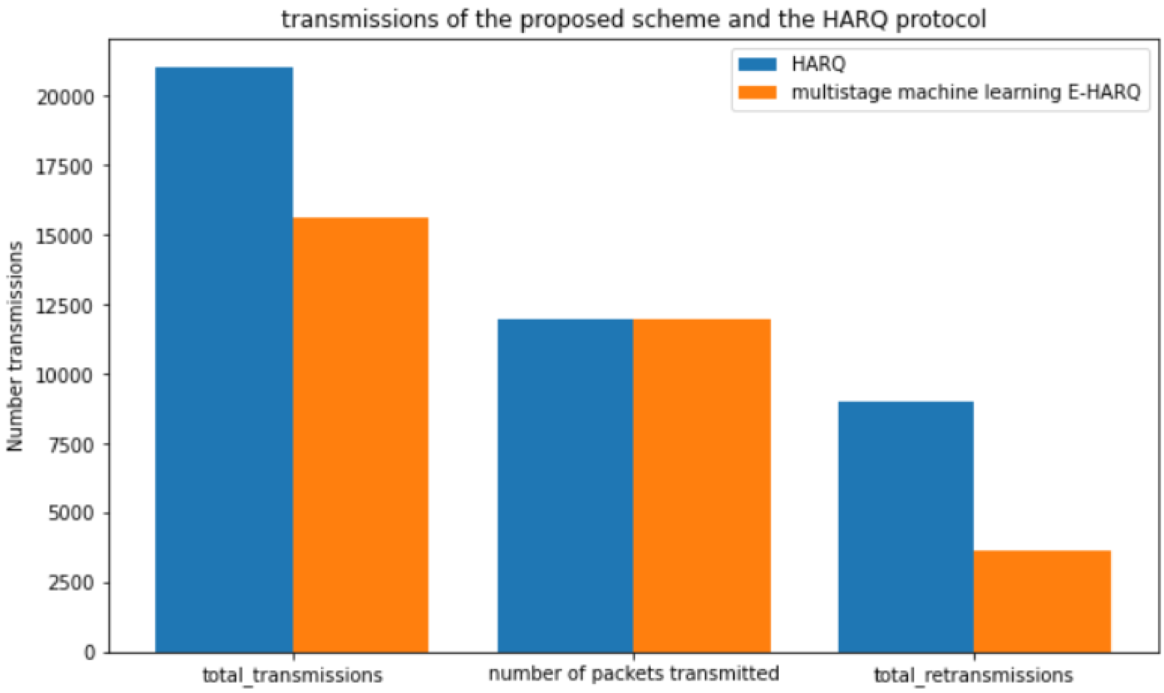


Figure 4 13: Number of transmissions

In Figure 4.13, we can observe that both schemes transmitted the same number of packets which is 12000 packets. However, the HARQ made a number of retransmissions compared to the proposed scheme. This means that the HARQ has more failed packet transmissions that have to be retransmitted. On the other hand, the proposed scheme has a smaller number of retransmissions which means that the number of failed packet transmissions that have to be retransmitted is less compared to the proposed scheme. This is caused by the fact that the proposed scheme uses the multistage decision to increase probability of packets being successfully transmitted and therefore this reduces the number of retransmissions of the failed packets. Therefore, the proposed scheme improved the performance of the proposed schemes in terms of transmissions compared to the HARQ protocol.

4.10.2 Failed and passed transmissions

Figure 4.14 shows the total number of successful and unsuccessful transmissions of the proposed scheme and the HARQ.

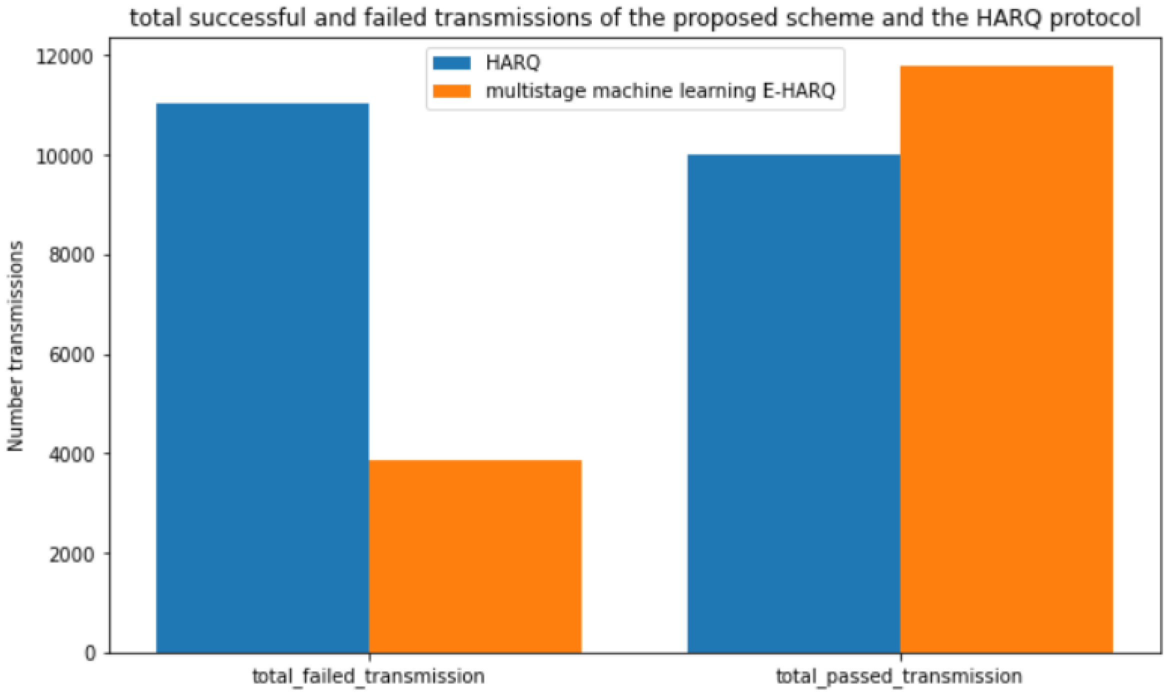


Figure 4 14: The number of successful and unsuccessful transmissions of the proposed scheme and the HARQ protocol

In Figure 4.14, we can observe that the proposed multistage machine learning E-HARQ has more successful transmissions and a smaller number of failed transmissions compared to the HARQ protocol. The proposed scheme uses the multistage decision to adjust the channel bandwidth to increase the probability of the packet to be transmitted successfully if the channel conditions are likely to yield a transmission failure. Therefore, this reduces the number of failed transmissions and increases the number of successful transmissions in the proposed multistage machine learning E-HARQ. Therefore, the proposed scheme improved performance in terms of packet transmission.

4.10.3 Initial transmissions

Figure 4.15 shows the number of successful and unsuccessful initial transmissions of the proposed multistage machine learning E-HARQ scheme and the HARQ protocol.

Figure 4 15: Initial transmission of the HARQ and the proposed scheme

In Figure 4.15, we can observe that the proposed multistage machine learning E-HARQ protocol, has more packets that were transmitted successfully and lesser packets that were transmitted unsuccessfully compared to the existing HARQ protocol. The proposed scheme uses the optimal channel bandwidth to increase the probability of the packets being transmitted successfully even if the packet is transmitted for the first time. This means that the proposed scheme improved the performance of the transmissions when transmitting the packets for the first time.

4.10.4 Retransmissions

Figure 4.16 shows the number of successful and unsuccessful retransmission of the HARQ protocol and the proposed multistage machine learning E-HARQ scheme.

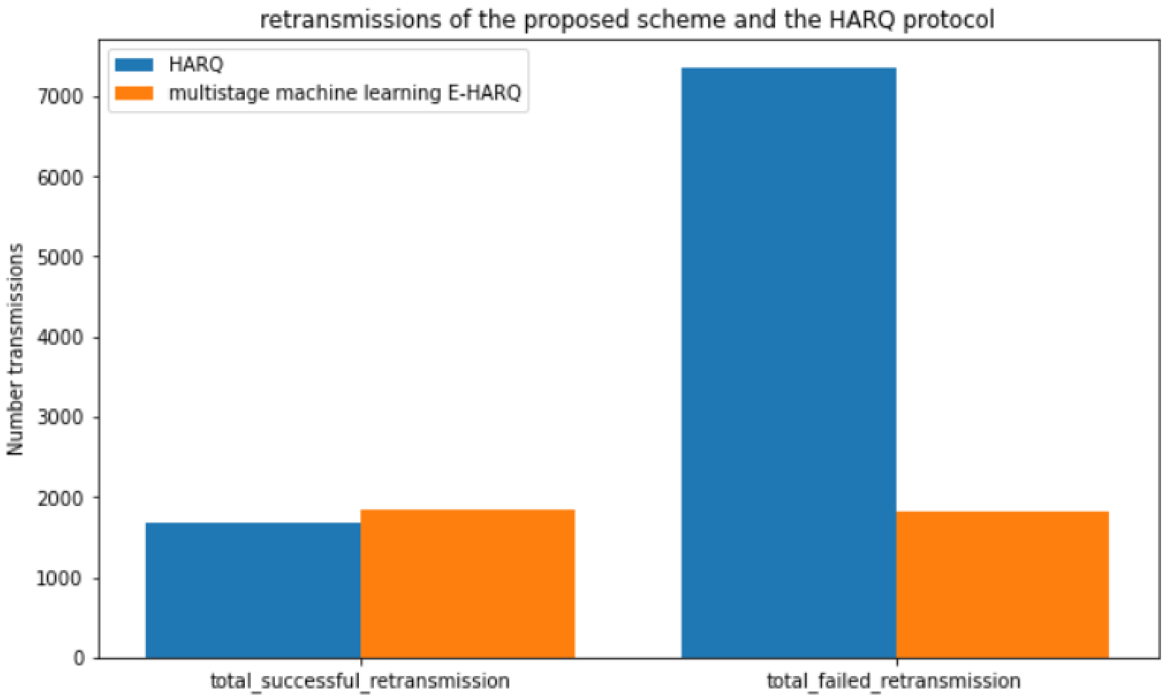


figure 4 16: Successful and unsuccessful retransmissions

In Figure 4.16, we can observe that the number of failed retransmissions for HARQ is higher compared to the number of successful retransmissions of the HARQ. This means that when retransmitting the packet using the HARQ, the probability of successful retransmissions is very low, approximately 17%. However, we can observe that in the proposed multistage machine learning E-HARQ, the number of successful and unsuccessful retransmissions is the same. The probability of retransmitting the packet successfully is 50% when using the proposed scheme. This means that the proposed scheme improved the retransmission performance of the existing HARQ protocol.

4.11 Simulation results of the network performance

This section evaluates the comparative performance results of the proposed scheme to the results of the existing HARQ and the machine learning EHARQ schemes. The performance was evaluated based on throughput, error rate, RTT, and latency.

4.11.1. Throughput performance

Figure 4.17, presents the throughput performance results of the proposed scheme and the results of the HARQ and machine learning E-HARQ. The proposed scheme is presented by the black solid line. In Figure 4.17, we can see that the proposed scheme achieved higher throughput at lower SNR. This is caused by the fact that at lower SNR, there is more noise than the signal in the channel, meaning the channel condition is very poor [52], [53], [54], [55], [56]. Therefore, the proposed scheme uses higher bandwidth in poor channel conditions to increase the probability of successful transmission and hence improving the throughput in bad channel conditions. From the simulation results, at the SNR value greater than 10, the performance of the proposed scheme is the same as the performance of the HARQ scheme.

This is caused by the fact that in good channel conditions, the proposed scheme adjusts the bandwidth to mitigate throughput loss resulting from false predictions. Therefore, the throughput of the proposed scheme is not affected by false predictions. On the other hand, the Machine E-HARQ fails to reduce the throughput loss, which is the main reason for the machine learning E-HARQ having lower achievable throughput at an SNR value greater than 10.

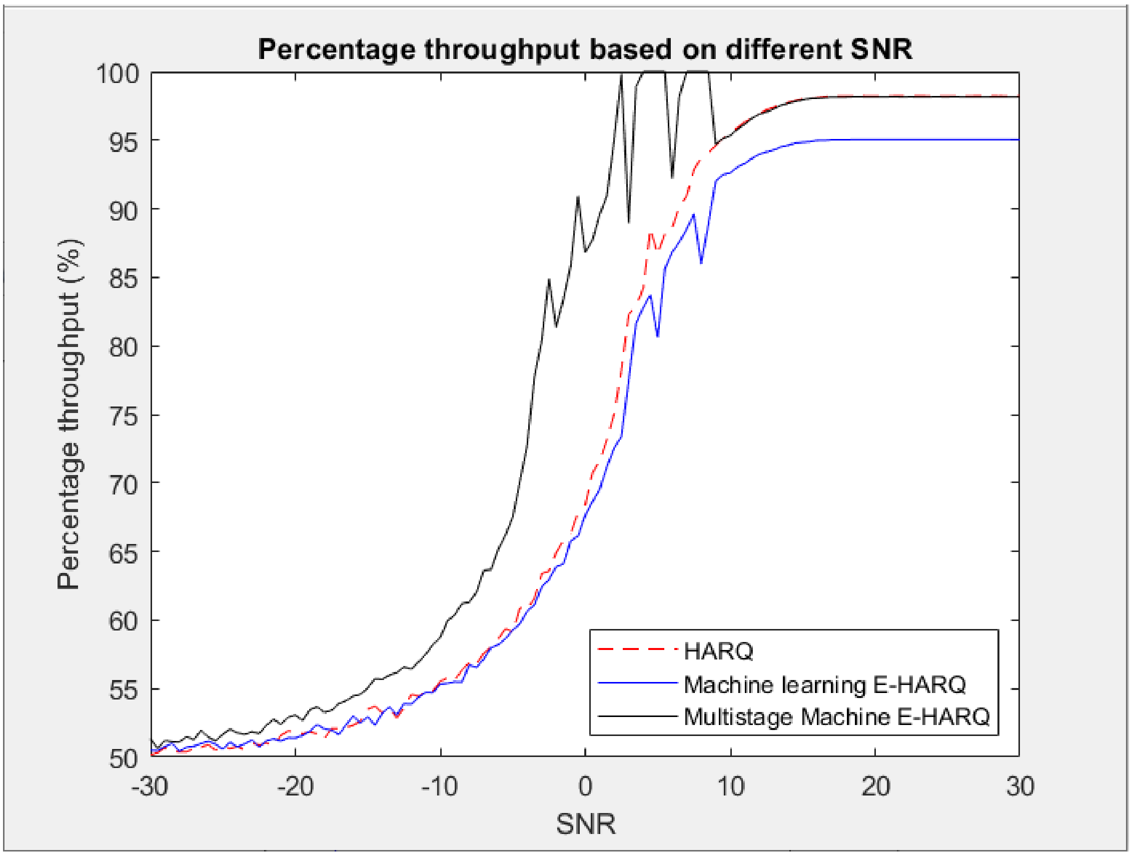


Figure 4 17: Throughput results

4.11.2. Round trip time

This section focuses on the RTT of the proposed scheme and compares it to the RTT performance of the HARQ and machine learning E-HARQ schemes. The simulation results are shown in Figure 4.18.

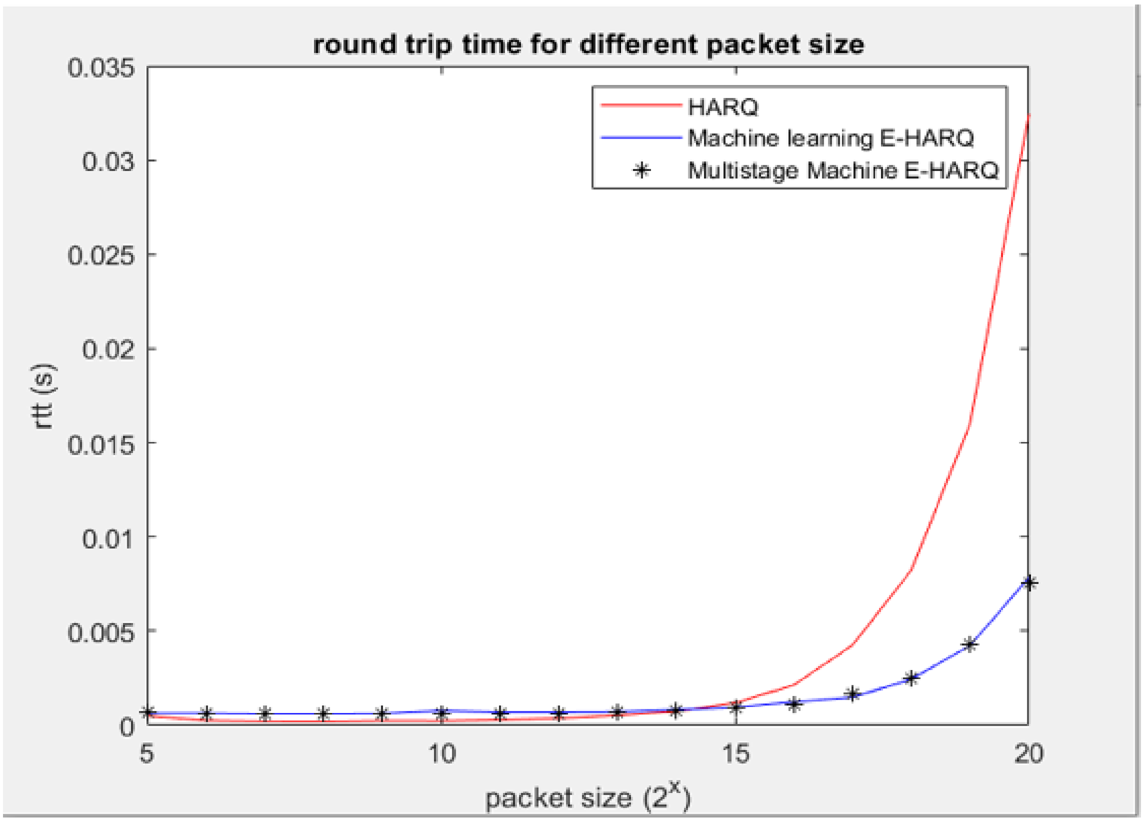


Figure 4 18: Round trip time performance results

In Figure 4.18, we can see that during the transmission of shorter packets, the HARQ has lower RTT compared to the multistage machine learning E-HARQ and the machine learning E-HARQ.

This is caused by the fact that the model takes some time to respond or to make predictions. This time is more than what it takes to decode shorter packets because shorter packets have few bits. That is the main reason for the HARQ having less RTT. However, as the packet size increases, the RTT of the HARQ is extremely high. This is caused by the fact that larger packets have more bits. Hence, it will take more time to decode all the bits, because the decoder must decode every bit. Therefore, more bits result in a longer RTT for the HARQ since it has to decode every bit in the packet.

Furthermore, according to the results, we can see that both the multistage machine learning E-HARQ and the machine learning E-HARQ have less RTT when the packet size increases. This is attributable to the fact that they both use the model to predict the feedback. The feedback is sent before the decoding process is completed, resulting in reduced RTT. Since the simulation results show that the proposed scheme and the machine learning E-HARQ have the same RTT, this means that adding the multistage decision in the machine learning E-HARQ did not affect the RTT.

Given these results, we can conclude that the best scheme for RTT performance is the HARQ when transmitting shorter packets. However, both multistage machine learning E-HARQ and machine learning E-HARQ are efficient in terms of RTT when transmitting larger packets. This means that the proposed multistage machine learning E-HARQ scheme performs better in terms of RTT when transmitting larger packets.

4.11.3. Error rate

This section focuses on the error rate of the proposed scheme, the existing HARQ, and the machine learning E-HARQ. The error rate results are shown in Figure 4.19, where the proposed multistage machine learning E-HARQ scheme has a lower error rate, especially at lower SNR.

This is caused by the fact that the proposed scheme uses the multistage decision to select the appropriate bandwidth depending on the channel condition, the SNR in this case. At lower SNR, the proposed scheme transmits using a channel with higher bandwidth to increase the possibility of successful transmission, thus, reducing the error rate. However, at higher SNR (SNR>=10) it has the same error rate as the HARQ protocol.

This is mainly caused by the fact that at higher SNR, it has reduced the transmission bandwidth to decrease the loss of throughput that could result from false predictions. In conclusion, the proposed scheme has the best performance in terms of error rate.

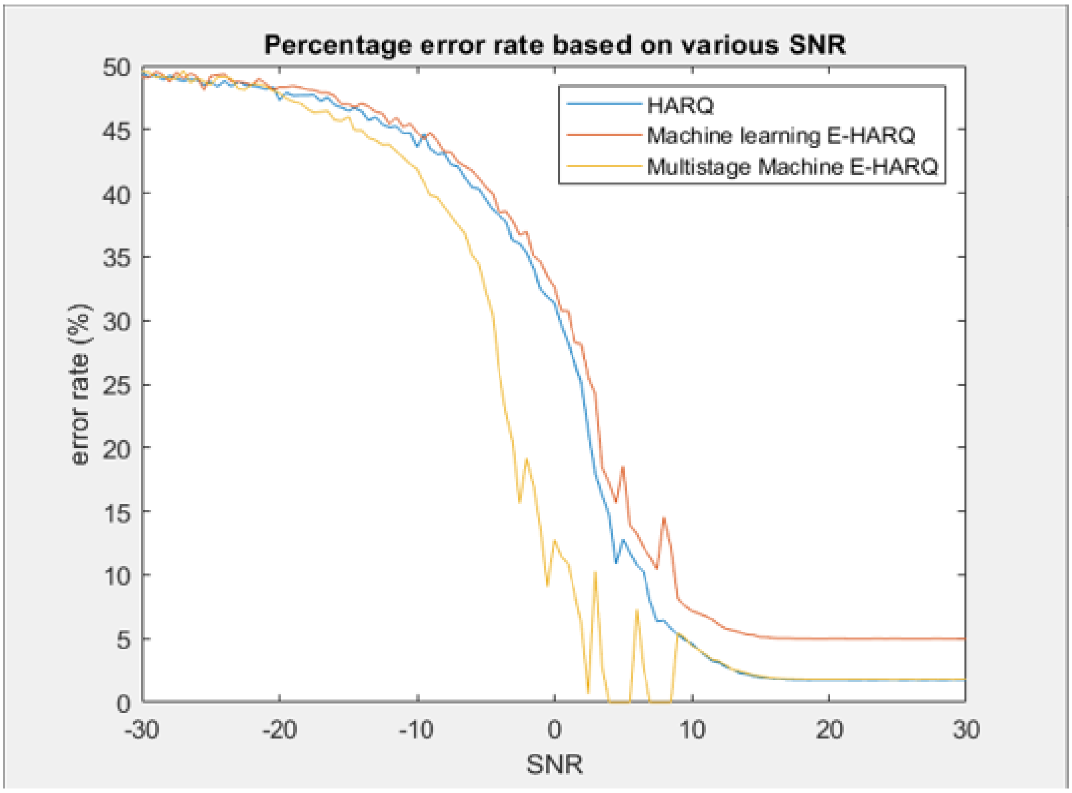


Figure 4 19: Error rate performance results

4.11.4. Latency

This section compares the latency of the proposed scheme to the latency of existing HARQ and machine learning E-HARQ scheme. This is accomplished by analysing the latency in different SNR values during the transmission of various packets with different packet sizes.

4.11.4.1. Latency at SNR=-10

Figure 4.20 presents the latency results of the schemes when transmitting packets of different sizes at the SNR value of -10. The results show that latency for all the schemes is lower when transmitting small packets (packets with less packet size). We observed that the main cause is that smaller packets have few bits to encode in all the schemes. This results in less latency for shorter packets in all three schemes.

However, the latency of the proposed scheme is lower compared to the other schemes in the case of bigger packet sizes. This is caused by the fact that the SNR is negative (snr=-10), meaning that there is more noise in the channel that the signal strength resulting in poor channel condition. Although the channel condition is poor, the proposed scheme is designed to handle such poor channel conditions. So the proposed scheme increases the bandwidth to reduce the transmission failure in these poor channel conditions and therefore reduces the number of retransmissions. It also reduces the latency resulting from retransmissions, which is the main reason for the proposed scheme achieving lower latency compared to the other schemes. Furthermore, the proposed scheme is able to predict outcomes of decoding packets before the process is completed, which also improves its performance.

On the other hand, the other two schemes make a number of retransmissions due to transmission failure in poor channel conditions since they are not designed to handle poor channel conditions. This results in increased latency caused by retransmissions of failed packets, especially when large packets are being transmitted. Furthermore, the HARQ has higher latency when transmitting larger packets.

This is caused by the fact that it has to decode all the bits for every transmission and larger packets have more bits. In essence, larger packets result in increased latency for the HARQ protocol. This is not the case for the proposed scheme and the machine learning E-HARQ since they do not decode the packets. Instead, they call the predictive model to predict the decoding outcome, reducing the latency for both machine learning HARQ and the proposed scheme.

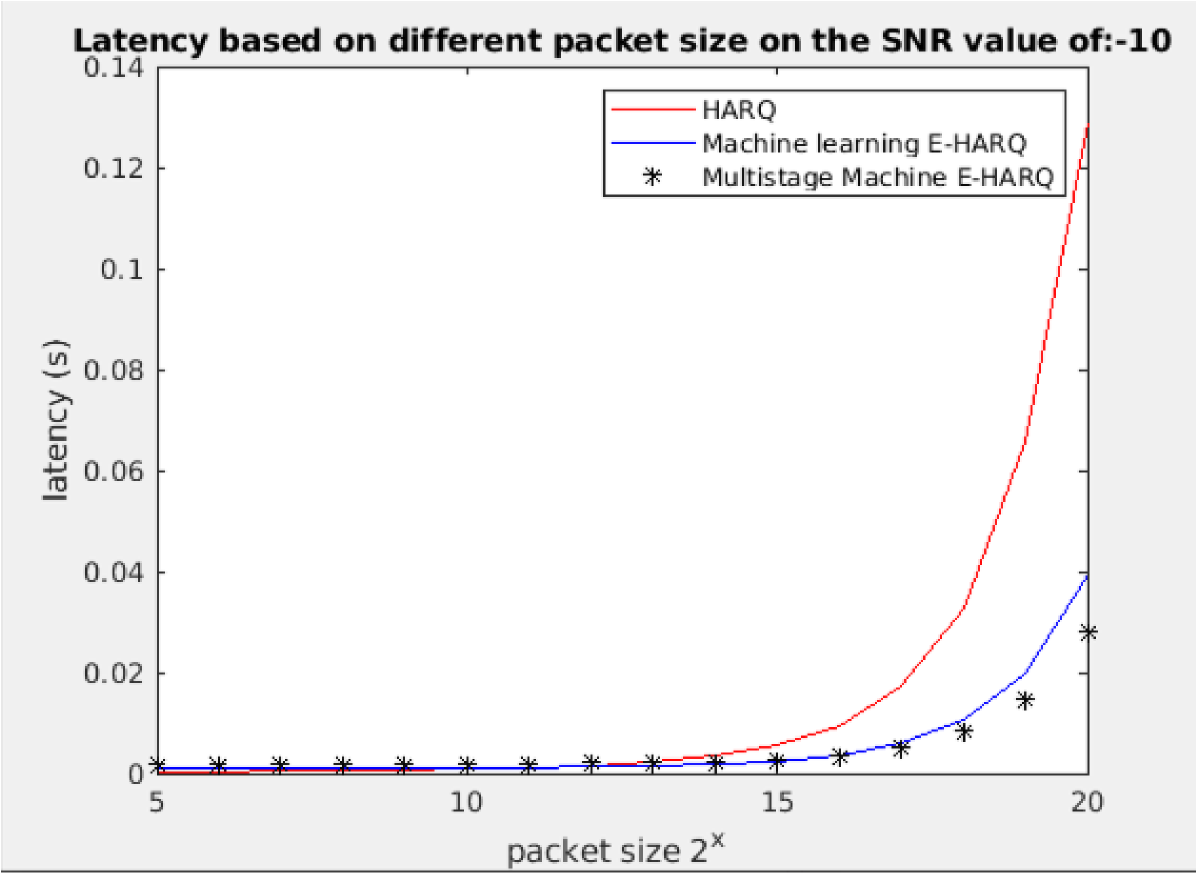


Figure 4 20: Latency results for SNR=-10

4.11.4.2. Latency at SNR= -5

The latency results of the SNR value of -5 were similar to the latency results of the SNR value of -10 shown in Figure 4.20. As a result, the latency results of SNR value of -5 were included in the results.

4.11.4.3. Latency at SNR= 0

In the latency results in Figure 4.21, we increased the SNR to 0 and used packets of different sizes. The latency of the proposed scheme and the latency of the machine learning E-HARQ are the same. This is caused by the fact that at the SNR value of zero, the proposed scheme reduced its bandwidth because, at the SNR value of 0, the noise and the signal are at the same level. So the proposed scheme uses a balanced bandwidth for transmitting.

Therefore, our scheme uses the bandwidth that is approximately equal to the bandwidth of the machine learning E-HARQ. Hence, both schemes yield the same latency performance. However, the HARQ still has more latency when transmitting larger packets. As mentioned earlier, this is because it has to decode all the bits in the packet and larger packets consisting of more bits. This culminates in increased latency for the HARQ protocol when transmitting larger packets.

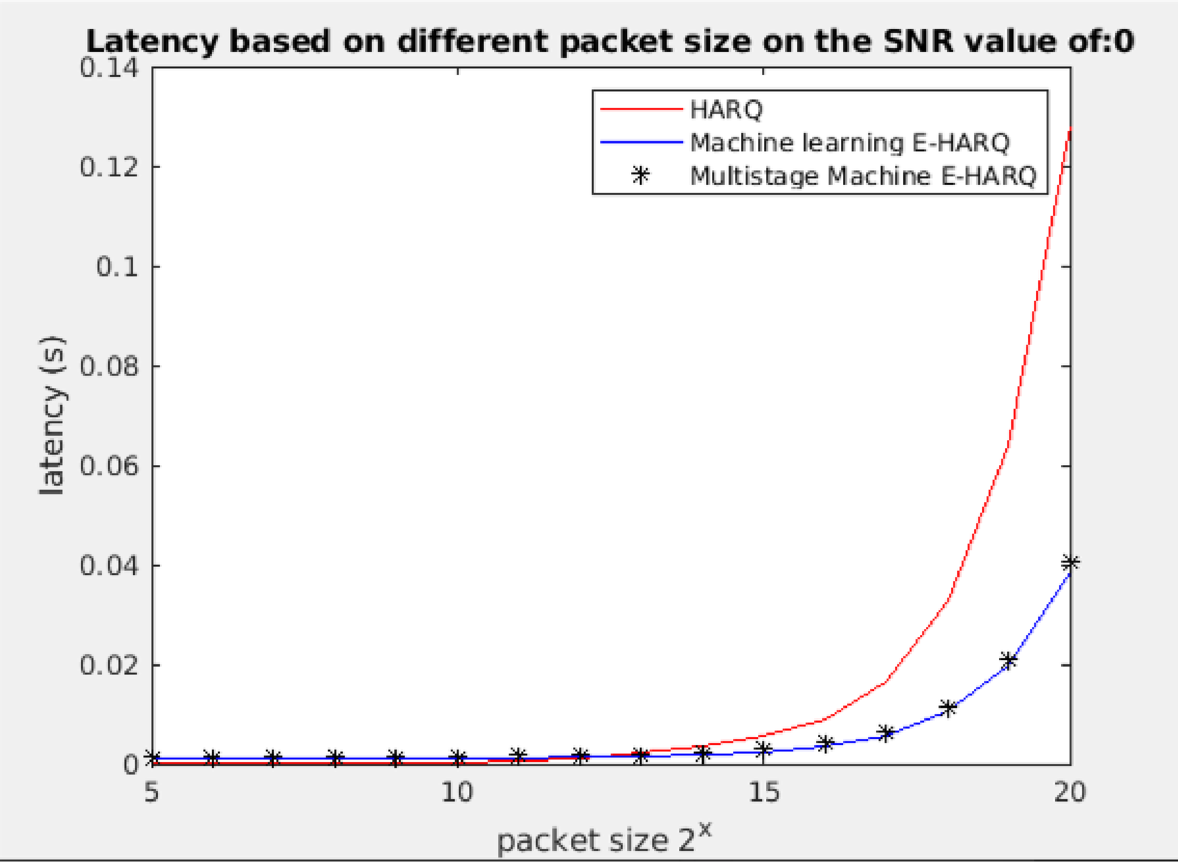


Figure 4 21: Latency results for SNR=0

4.11.4.4. Latency at SNR=5

The latency results of the SNR value of 5 were also similar to the latency performance results of the SNR value of 10 shown in Figure 4.22. The SNR results for value of 5 were not included.

4.11.4.5. Latency at SNR=10

We further increased the SNR value to 10 and transmitted packets of different sizes using the three schemes. The latency results are shown in Figure 4.22. The results show that the proposed scheme has the lowest latency, especially on large packets. This is caused by the fact that with large packets, there is a high possibility of packet failure. However, the SNR is positive, which means that there is more signal than channel noise, hence this increases the possibility of successful transmission. This results in a high probability of false predictions, depending on the accuracy of the model.

The proposed scheme and the machine learning E-HARQ use the model to predict the feedback. The only difference is that our scheme is designed to handle false predictions, meaning that the proposed scheme can mitigate the latency resulting from false predictions. This is not the case with the machine learning E-HARQ, since it can retransmit the incorrectly predicted packet failures and thus result in slightly increased latency for the machine learning E-HARQ.

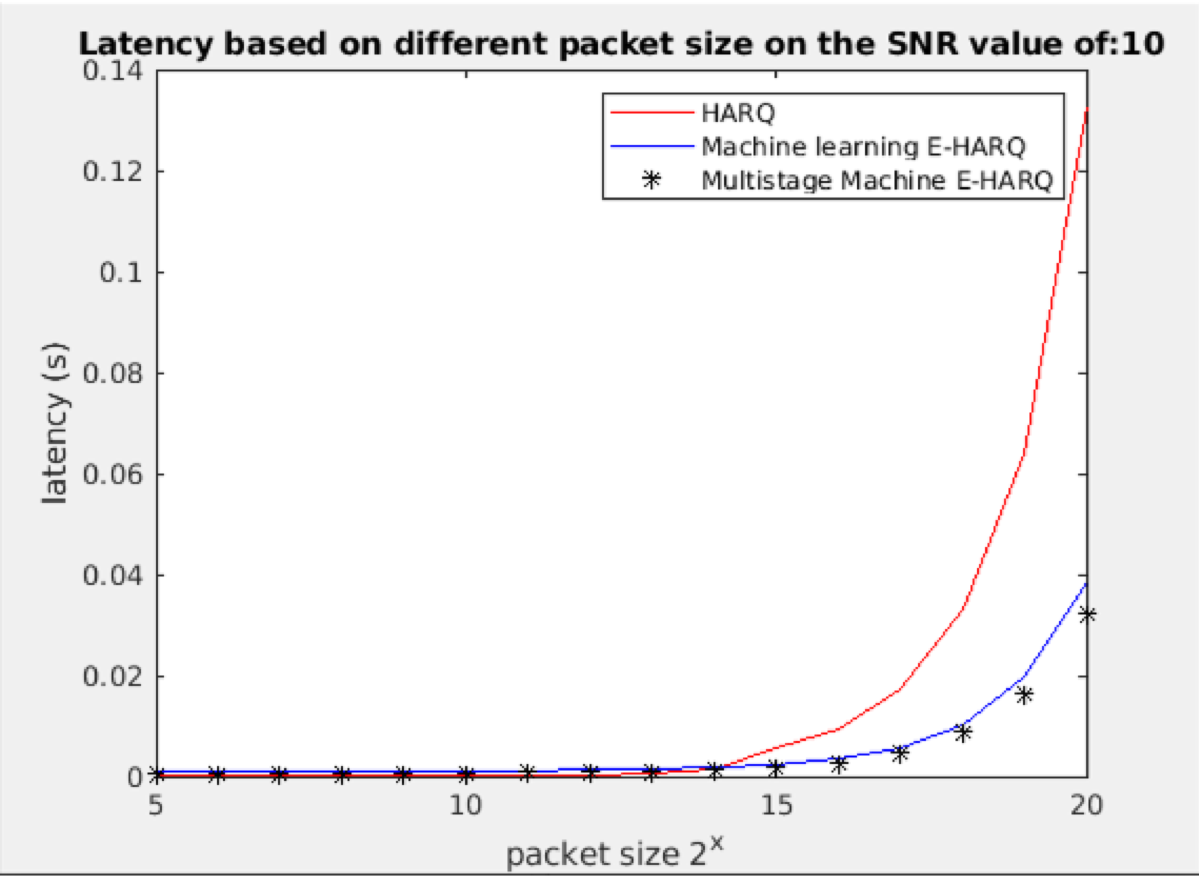


Figure 4 22: Latency results for SNR=10

4.11.4.6. Latency of SNR values ranging between -20 and 20

Figure 4.23 shows the comparative latency results of SNR values ranging between -20 and 20dB.

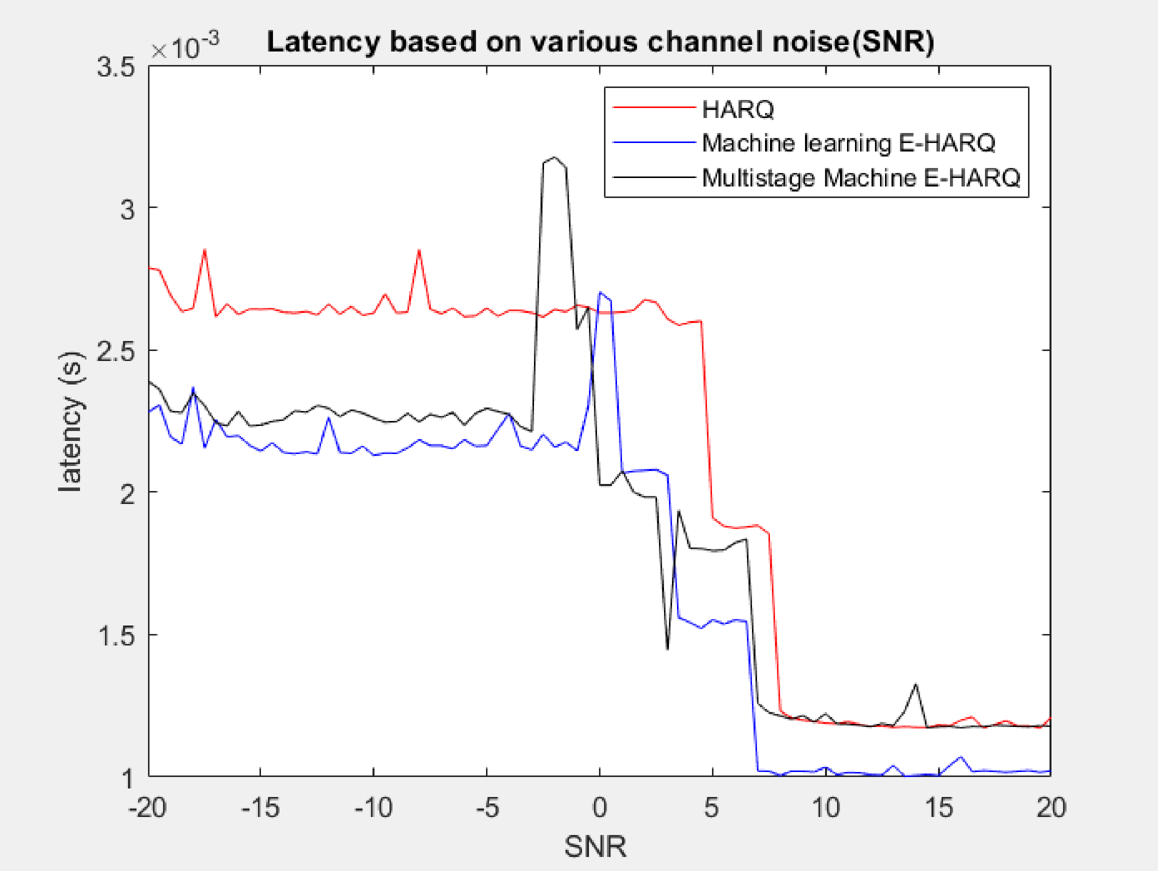


Figure 4 23: Latency results for SNR=10

In Figure 4.23, we can observe that the latency of the proposed scheme (denoted by a solid black line) is marginally higher than the latency of the machine learning E-HARQ. However, between the SNR values of -5 and 0dB, the latency of the proposed scheme increased significantly. This is caused by the fact that the proposed scheme is designed to change the bandwidth channel depending on the SNR value or the channel conditions. Therefore, the change in the channel bandwidth resulted in increased latency as shown in Figure 4.23.

4.11.5. Performance results when larger packet sizes were considered

We further evaluated the performance of the proposed scheme in larger packets to establish if its latency and RTT performance improves as the packet size increases. We observed this trend in the earlier results and strove to confirm this observation using these additional results.

4.11.5.1. Round trip time on larger packets

The RTT performance of the proposed scheme on larger packets is depicted in figure 4.24. In the figure 4.24, the gap between RTT of the proposed multistage machine learning E-HARQ curve and the one of the RTT of the HARQ widens as the packet size increases. This proves that the RTT performance of the proposed scheme improves as the packet size increases. We only examined the packet size of up to 2^25=33554432 bits. This result confirms that the observed trends will continue to be exhibited for larger packets which are greater than 33554432 bits. Hence it can be concluded that the RTT performance of the proposed multistage machine learning E-HARQ will always improve as the packet size increases compared to the existing HARQ.

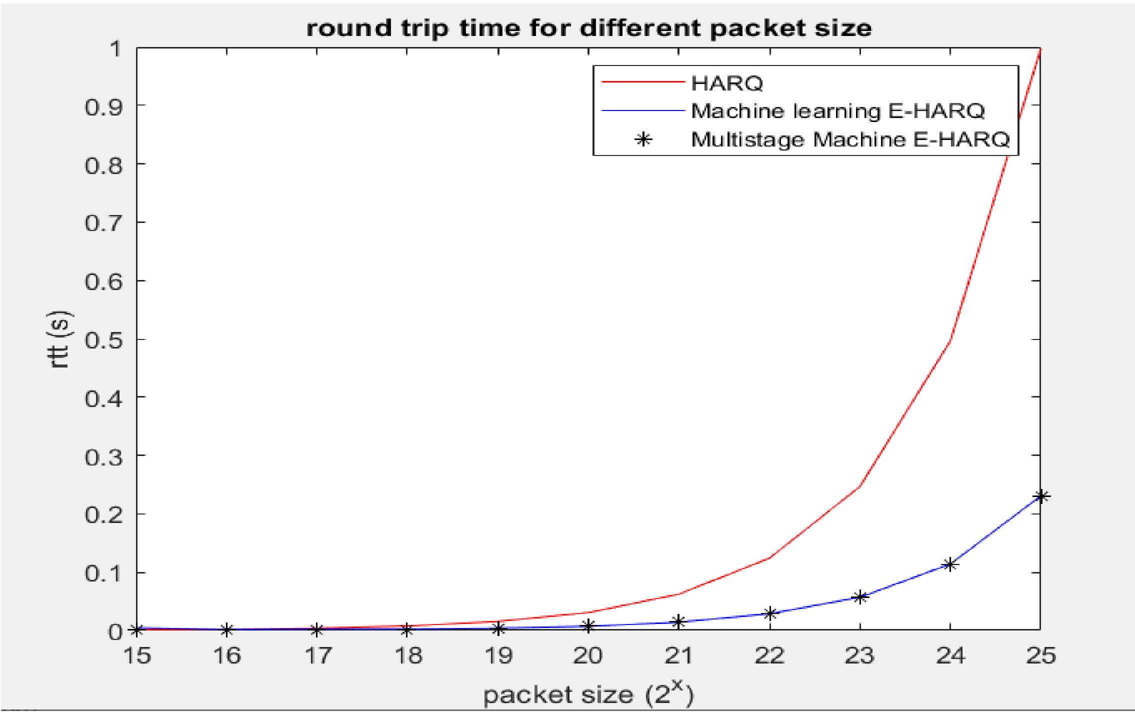


Figure 4 24: RTT on larger packets

4.11.5.2. Latency Results on larger packets

The latency results of the proposed scheme on larger packets is depicted in Figure 4.25. In the Figure 4.25, the gap between the latency performance of proposed multistage machine learning E-HARQ and the latency performance of the HARQ and the machine learning E-HARQ widens as the packet size increases. This proves that the proposed scheme's performance in terms of latency improves as the size of the packet increases.

We only examined the performance of the scheme with packet sizes of up to 2^25=33554432 bits. This observed trend should continue as the size of packets is increased even beyond packet sizes larger than 33554432 bits. Hence, it can be concluded that the performance of the proposed multistage machine learning E-HARQ should always improve as the packet size increases compared to the existing HARQ and the machine.

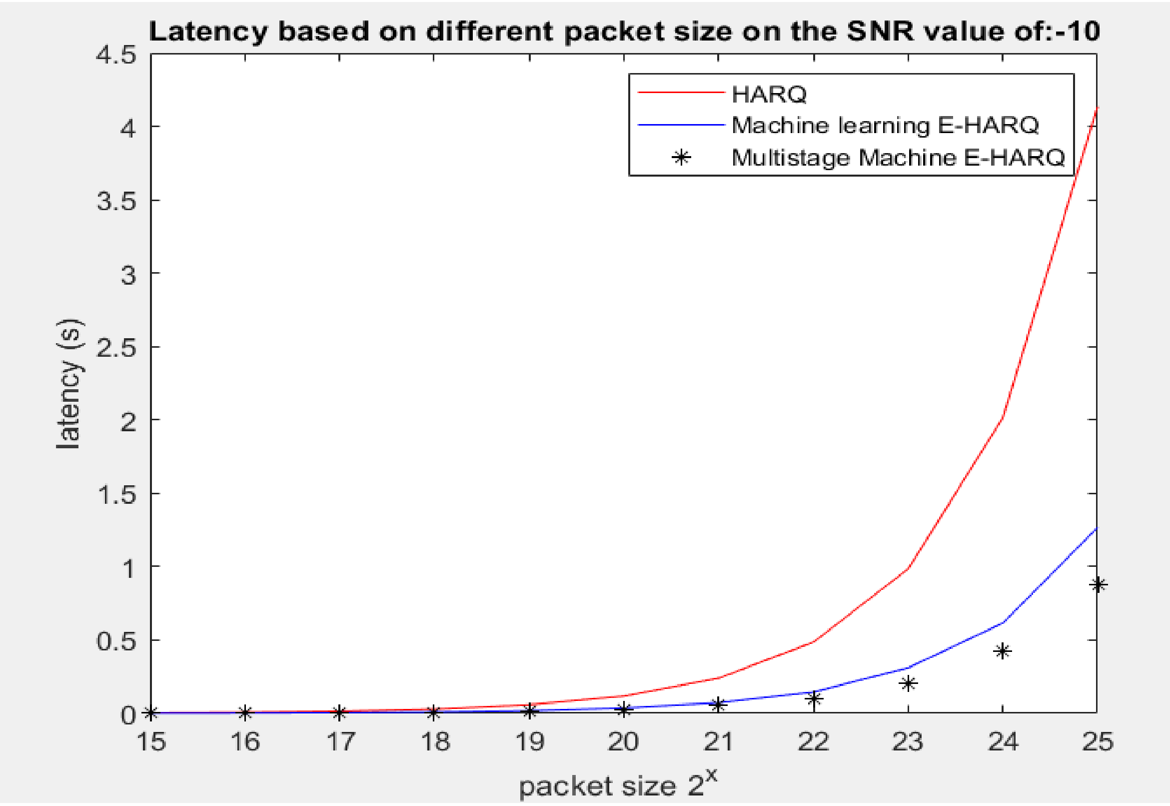


Figure 4 25: Latency on larger packets

4.11.6. Latency results of the proposed scheme using different modulation techniques.

Figure 4.26 shows the latency performance of the proposed scheme when transmitting the data using different modulation techniques.

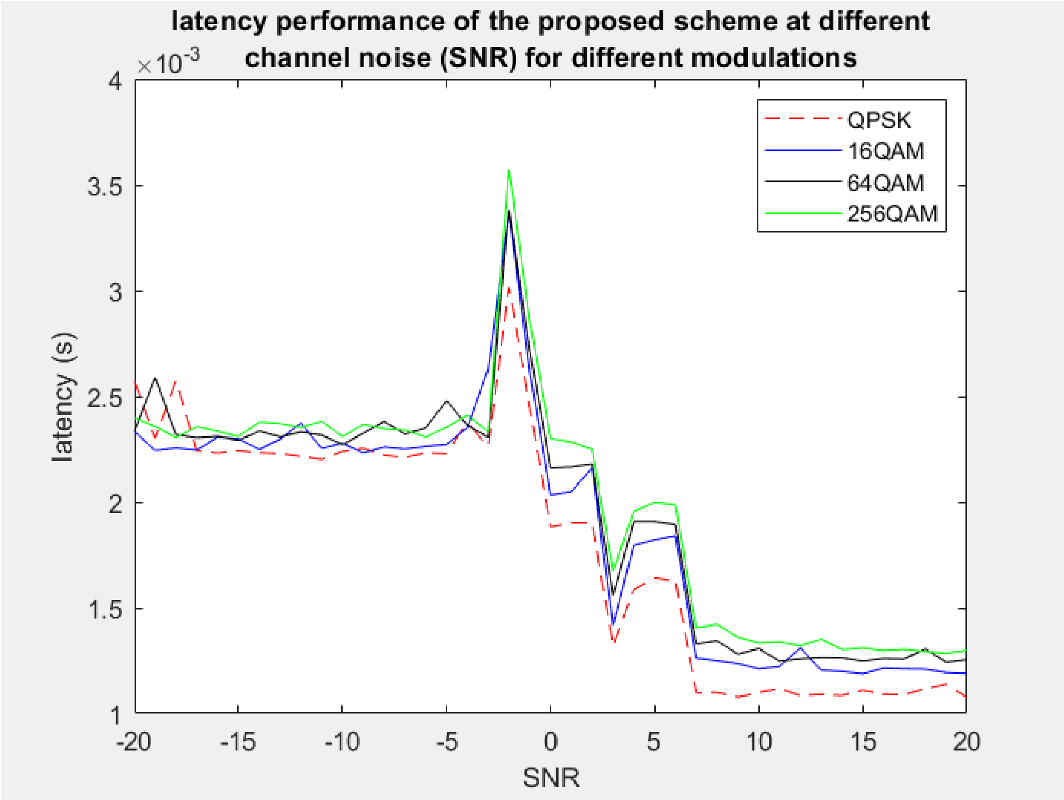


Figure 4 26: Latency of the proposed scheme using different modulation techniques.

In Figure 4.26, we can observe that the proposed scheme recorded good performance when using the quadrature phase shift keying (QPSK). However, the 16 QAM also produces a reasonable performance as compared to the 64QAM and the 256QAM. The reduced latency for the QPSK and the 64QAM is caused by the fact that both QPSK and the 16QAM have a lower error rate. This reduces the latency caused by the retransmission of failed transmissions.

Based on the latency performance results, it is recommended for the proposed scheme to use the QPSK or the 16QAM to reduce the latency resulting from transmission errors in real communications. The significant increase in latency between the SNR value of -5 and 0 in figure 4.26 is mainly caused by the fact that the multistage decision changes the bandwidth channel and this leads to increased latency and can be considered as one of the demerits of the proposed scheme.

4.12 Discussion of the results

From our overall results, we can draw the following conclusion:

The proposed scheme has higher throughput, especially on the lower SNR. The proposed multistage machine learning E-HARQ optimises throughput compared to the HARQ and the machine learning E-HARQ. Furthermore, the RTT performance of the proposed scheme is the same as the performance of the machine learning E-HARQ.

This means that adding the multistage decision on the machine learning E-HARQ did not result in increased RTT. However, in poor channel conditions, the proposed scheme has a lower error rate. Furthermore, the latency of the proposed scheme is the same as the latency of the machine learning E-HARQ. However, the proposed scheme has a slight improvement in the latency when transmitting larger packets. From this observation and discussion, we can conclude that our scheme optimises the throughput with marginal improvement in the optimisation of latency.

4.13 Conclusion

In this chapter, we presented the data collected to build the model. We then analysed the data to understand it and identify the necessary features for building the model. This chapter also presented the accuracy of the developed models. Furthermore, the chapter presented simulation results of the proposed scheme, machine learning E-HARQ, and the HARQ in terms of latency, throughput, error rate, and round trip time. From the results, we conclude that the proposed scheme improves the overall network performance.

CHAPTER 5 - CONCLUSION

5.1. Introduction

The 5G URLLC networks require extremely low latency and high reliability. However, the HARQ achieves the required reliability at the cost of latency in 5G URLLC. The physical layer HARQ poses a bottleneck for achieving the required latency. This is caused by the fact that the receiver must decode the entire packet before sending feedback signals which may result in increased latency and may not be a viable solution for URLLC [10], [4]. To address this problem, this study proposed a multistage machine learning E-HARQ for predicting the feedback before the decoding process is completed to reduce latency. The proposed scheme uses the multistage decision to mitigate the throughput loss that could result from false predictions. We compared the performance of the proposed multistage machine learning E-HARQ with the performance of the HARQ and the machine learning E-HARQ. This chapter summarises the findings of the investigations and outlines future directions.

5.2. Research Summary

This study proposed a multistage machine learning E-HARQ to optimize throughput and latency in 5G URLLC. The proposed scheme uses logistic regression for predicting the feedback to reduce latency. Furthermore, the proposed scheme uses the multistage decision to mitigate throughput loss resulting from false predictions. We compared the performance of the proposed multistage machine learning E-HARQ scheme to the performance of the existing HARQ and the machine learning E-HARQ.

The study demonstrated that false predictions result in increased throughput loss and increased latency. We also observed that adding the multistage decision in machine learning E-HARQ scheme in our proposed scheme improved the throughput performance with marginal improvement in the latency. Furthermore, we confirmed that the proposed multistage machine learning E-HARQ improves performance in terms of latency, throughput, error rate, and round trip time.

5.3. Achieved objectives and addressed questions

In the literature review presented in Chapter two, we observed that false predictions result in throughput loss and increased latency. We proposed the multistage machine learning E-HARQ for addressing the throughput loss and the increased latency resulting from false alarm and miss-detection.

This study developed the existing machine learning E-HARQ scheme by deploying the machine learning model in the HARQ protocol because we could not find its code on the internet. We then developed our proposed scheme by implementing the multistage decision in the machine learning E-HARQ.

After developing the proposed scheme, the study deployed the proposed multistage machine learning E-HARQ scheme in the network and its performance results were presented and evaluated in Chapter 4.

In the results presented in Chapter 4, we have observed that the overall network performance of the proposed multistage machine learning E-HARQ scheme is improved and balanced, in terms of throughput, latency, RTT, and error rate. Hence, we concluded that the proposed schemes have better optimisation of the end-to-end performance.

Furthermore, the results showed that the proposed scheme optimizes the throughput and has a marginal improvement in latency, especially when transmitting large packets. Hence, we concluded that adding the multistage decision to the proposed scheme was effective in the throughput and latency optimization.

This section suffices as evidence that the study achieved all the proposed objectives and addressed all the research questions proposed in Chapter 1.

5.4. Recommendations and limitations of the study

The study did not focus on tuning the model to improve prediction accuracy as this was beyond the scope of the study. Furthermore, the proposed scheme uses only the channel noise to perform the multistage decision. However, for future studies, multistage decision could be performed based on multiple features, such as packet size, channel noise, and the type of channel.

5.5. Future research

Future studies could consider tuning the model to improve prediction accuracy. Furthermore, the scheme that performs the multistage decisions based on multiple features such as the packet size, channel type, and noise is recommended for future studies. The response time of the model in deployment may be optimised. This could benefit the network performance in terms of latency.

5.6. Final conclusion

The study proposed the multistage machine learning E-HARQ to optimise the latency and throughput in 5G URLLC networks. The proposed scheme uses machine learning algorithms to predict the transmission feedback to reduce transmission latency. Furthermore, the proposed scheme uses the multistage decision to mitigate the throughput loss that can result from false predictions. By comparing the proposed scheme to the existing HARQ and the machine learning E-HARQ, we observed that the proposed scheme offers the best overall network performance. Model tuning for better prediction is beyond the scope of the study and could be addressed in future studies.

References

[1] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S.A. Ashraf, B. Almeroth, J. Voigt, I. Riedel and A. Puschmann, "Latency Critical IoT Applications in 5G: Perspective on the Design of Radio Interface and Network Architecture," *in IEEE Communications Magazine,* vol. 55, no. 2, pp. 70-78, 3 Feb 2017.

[2] N. Strodthoff, B. Göktepe, T. Schierl, C. Hellge and W. Samek, "Enhanced Machine Learning Techniques for Early HARQ Feedback Prediction in 5G," *in IEEE Journal on Selected Areas in Communications,* vol. 37, no. 11, pp. 2573-2587, Nov 2019.

[3] B. Goektepe, S. Faehse, L. Thiele, T. Schierl and C. Hellge, "Subcode-Based Early HARQ for 5G," *in 2018 IEEE International Conference on Communications Workshops (ICC Workshops),* pp. 1-6, May 2018.

[4] N. Strodthoff, B. Göktepe, T. Schierl, W. Samek and C. Hellge, "Machine Learning for Early HARQ Feedback Prediction in 5G," *2018 IEEE Globecom Workshops (GC Wkshps),* pp. 1-6, 9 December 2018.

[5] K. Taniyama, Y. Kishiyama and K. Higuchi, "Low Latency HARQ Method Using Early Retransmission Prior to Channel Decoding with Multistage Decision," *in 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall),* pp. 1-5, 22 Sep 2019.

[6] Y. Imamura, D. Muramatsu, Y. Kishiyama and K. Higuchi, "Low latency hybrid ARQ method using channel state information before channel decoding," *In 2017 23rd Asia-Pacific Conference on Communications (APCC),* pp. 1-6, 11 Dec 2017.

[7] T. Alladi, V. Chamola, B. Sikdar and K. R. Choo, "Consumer IoT: Security vulnerability case studies and solutions," *IEEE Consumer Electronics Magazine,* pp. 17-25, 3 Feb 2020.

[8] F. E. Airod, H. Chafnaji and H. Yanikomeroglu, "HARQ in Full-Duplex Relay-Assisted Transmissions for URLLC," *IEEE Open Journal of the Communications Society,* vol. 2, pp. 409-422, 29 Jan 2021.

[9] F. Nadeem, M. Shirvanimoghaddam, Y. Li, and B. Vucetic, "Delay-Sensitive NOMA-HARQ for Short Packet Communications," *Entropy,* vol. 23, no. 7, p. 880, 9 Jul 2021.

[10] J. Nadas, P. Klaine, L. Zhang, G. Zhao, M. Imran and R. Souza, "Performance Analysis of Early-HARQ for Finite Block-Length Packet Transmission," *2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS),* pp. 391-396, May 2019.

[11] A. Ahmed, A. Al-Dweik, Y. Iraqi, H. Mukhtar, M. Naeem and E. Hossain, "Hybrid Automatic Repeat Request (HARQ) in Wireless Communications Systems and Standards: A Contemporary Survey," *in IEEE Communications Surveys & Tutorials,* vol. 23, no. 4, pp. 2711-2752, Jul 2021.

[12] T. Rykova, B. Göktepe, T. Schierl, K. Samouylov, and C. Hellge, "Analytical Model and Feedback Predictor Optimization for Combined Early-HARQ and HARQ," *Mathematics,* vol. 9, no. 17, p. 2104, 31 Aug 2021.

[13] S. AlMarshed, D. Triantafyllopoulou and K. Moessner, "Supervised Learning for Enhanced Early HARQ Feedback Prediction in URLLC," *In 2020 IEEE International Conference on Communication, Networks and Satellite (Comnetsat),* pp. 26-31, 17 Dec 2020.

[14] B. Göktepe, T. Rykova, T. Fehrenbach, T. Schierl and C. Hellge, "Feedback Prediction for Proactive HARQ in the Context of Industrial Internet of Things," *GLOBECOM 2020 - 2020 IEEE Global Communications Conference,* pp. 1-7, 7 Dec 2020.

[15] S. AlMarshed, D. Triantafyllopoulou and K. Moessner, "Deep Learning-Based Estimator for Fast HARQ Feedback in URLLC," *2021 IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (IEEE PIMRC 2021),* pp. 642-647, 2021.

[16] H. Zhu, B. Smida and D. J. Love, "Throughput Analysis of Two-Way NCed-HARQ With Reverse-Link Assistance and Estimated Channel State Information," *IEEE Communications Letters,* vol. 22, no. 2, pp. 352-355, Feb 2018.

[17] B. Kim, and H. Park, "Performance Analysis of HARQ in 2-step RACH Procedure Using Markov Chain Model," *In Advances on Broad-Band Wireless Computing, Communication and Applications: Proceedings of the 17th International Conference on Broad-Band Wireless Computing, Communication and Applications,* pp. 147-158, 18 Oct 2022.

[18] G. Berardinelli, S. R. Khosravirad, K. I. Pedersen, F. Frederiksen and P. Mogensen, "Enabling Early HARQ Feedback in 5G Networks," *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring),* pp. 1-5, 15 May 2016.

[19] C. Liang, J. Xu, Q. Guo and J. Xu, "Soft HARQ for 5G Downlink," *2020 International Wireless Communications and Mobile Computing (IWCMC),* pp. 1059-1064, 15 Jun 2020.

[20] G. Berardinelli, S. R. Khosravirad, K. I. Pedersen, F. Frederiksen and P. Mogensen, "On the benefits of early HARQ feedback with non-ideal prediction in 5G networks," *2016 International Symposium on Wireless Communication Systems (ISWCS),* pp. 11-15, 20 Sep 2016.

[21] A. Anand and G. Veciana, "Resource Allocation and HARQ Optimization for URLLC Traffic in 5G Wireless Networks," *in IEEE Journal on Selected Areas in Communications,* vol. 36, no. 11, pp. 2411-2421, Nov 2018.

[22] K. I. Pedersen, G. Pocovi and J. Steiner, "Preemptive Scheduling of Latency Critical Traffic and Its Impact on Mobile Broadband Performance," *2018 IEEE 87th Vehicular Technology Conference (VTC Spring),* pp. 1-6, 3 Jun 2018.

[23] H. Ren, C. Pan, Y. Deng, M. Elkashlan and A. Nallanathan, "Resource Allocation for URLLC in 5G Mission-Critical IoT Networks," *ICC 2019 - 2019 IEEE International Conference on Communications (ICC),* pp. 1-6, 20 May 2019.

[24] B. Husain and A. Czylwik, "Channel Coding and Low Latency HARQ for Industrial Wireless Sensor Networks," *2019 Wireless Days (WD),* pp. 1-5, Apr 2019.

[25] K. Miura, Y. Kishiyama and K. Higuchi, "Low Latency HARQ Method Using Early Retransmission Before Channel Decoding Based on Superposition Coding," *2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS),* pp. 1-2, 3 Dec 2019.

[26] S. Hao and H. Zhang, "MAC Performance Analysis for Reliable Power-Line Communication Networks with ARQ Scheme," *Sensors,* vol. 21, no. 1, 30 Dec 2020.

[27] A. A. Chavan, D. S. Kurule and P. U. Dere, "Performance Analysis of AODV and DSDV Routing Protocol in MANET and Modifications in AODV against Black Hole Attack," *Procedia Computer Science,* vol. 79, pp. 835-844, 1 Jan 2016.

[28] M. Imran and M. A. Qadeer, "Evaluation Study of Performance Comparison of Topology Based Routing Protocol, AODV and DSDV in MANET," *In 2016 International Conference on Micro-Electronics and Telecommunication Engineering (ICMETE),* pp. 207-211, 22 Sep 2016.

[29] D. Anandkumar and R.G. Sangeetha, "A survey on performance enhancement in free space optical communication system through channel models and modulation techniques.," *Optical and Quantum Electronics,* vol. 53, no. 1, pp. 1-39, 2021.

[30] Y.H. Robinson, E.G. Julie, I.J. Jacob, T.S. Lawrence, V. Saravanan and P.E. Darney, "Enhanced Energy Proficient Encoding Algorithm for Reducing Medium Time in Wireless Networks," *Wireless Personal Communications,* vol. 119, no. 4, pp. 3569-3588, 28 Mar 2021.

[31] J. Dai, K. Tan, Z. Si, K. Niu, M. Chen, H.V. Poor and S. Cui, "Learning to Decode Protograph LDPC Codes," *IEEE Journal on Selected Areas in Communications,* vol. 39, no. 7, pp. 1983-1999, 10 May July 2021.

[32] C. Condo and G. Masera, "Unified turbo/LDPC code decoder architecture for deep-space communications," *in IEEE Transactions on Aerospace and Electronic Systems,* vol. 50, no. 4, pp. 3115-3125, 11 Dec 2014.

[33] F. Nadeem, M. Shirvanimoghaddam, Y. Li and B. Vucetic, "Non-orthogonal HARQ for URLLC: Design and Analysis," *in IEEE Internet of Things Journal,* vol. 8, no. 24, pp. 17596-17610, 15 Dec 2021.

[34] A. Avranas, M. Kountouris and P. Ciblat, "Throughput Maximization and IR-HARQ Optimization for URLLC Traffic in 5G Systems," *ICC 2019 - 2019 IEEE International Conference on Communications (ICC),* pp. 1-6, 20 May 2019.

[35] N. Zahid, G. H. Sodhro, M. B. Janjua, F. A. Chachar, A. H. Sodhro and S. A. K. Abro, "HARQ with chase-combining for bandwidth-efficient communication in MIMO wireless networks," *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET),* pp. 1-6, 3 Mar 2018.

[36] H. Zhu, B. Smida and D. J. Love, "Throughput Analysis of Two-Way NCed-HARQ With Reverse-Link Assistance and Estimated Channel State Information," *IEEE Communications Letters,* vol. 22, no. 2, pp. 352-355, 20 Nov 2018.

[37] B. Zhang, Y. Li, and Y. Liang, , "Impact of Packet Size on Performance of TCP traffic with Small Router," *In MATEC Web of Conferences,* vol. 128, p. 02023, 2017.

[38] H. I. Fitriasari, A. I. Lestari, D. L. Luhurkinanti and R. F. Sari,, "Performance Evaluation of Downlink Multi-user OFDMA Scheduling in 5G New Radio (NR)," *2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON),* pp. 213-223, 2021.

[39] T. K. Le, U. Salim and F. Kaltenberger, "An Overview of Physical Layer Design for Ultra-Reliable Low-Latency Communications in 3GPP Releases 15, 16, and 17," *in IEEE Access,* vol. 9, pp. 433-444, 23 Dec 2021.

[40] A. Pajankar, "Getting Started with Pandas," in *Practical Python Data Visualization*, Springer, 2021, pp. 117-136.

[41] G.Y. Lee, L. Alzamil, B. Doskenov and A. Termehchy, "A Survey on Data Cleaning Methods for Improved Machine Learning Model Performance," *arXiv preprint arXiv:2109.07127,* 15 Sep 2021.

[42] A. Vatreš and Z. Mašetić, "Exploring Customers' Behavior--Analysing Customer Data, Customer Segmentation and Predicting Customers' Behavior on Black Friday," in *International Symposium on Innovative and Interdisciplinary Applications of Advanced Technologies*, Springer, 2021, pp. 15-26.

[43] K. Xie, L. Han, M. Jing, J. Luan, T. Yang and Fan, R, "Review of Intelligent Data Analysis and Data Visualization," *In International Conference on Broadband and Wireless Computing, Communication and Application,* pp. 365-375, 28 Oct 2020.

[44] A.M. Abdullah, R.S.A. Usmani, T.R. Pillai, I.A.T. Hashem and M. Marjani, "Feature Engineering Algorithms for Traffic Dataset," *International Journal of Advanced Computer Science and Applications,* vol. 12, no. 4, pp. 1-8, 2021.

[45] D. Riswantini and E. Nugraheni, "Machine learning in handling disease outbreaks: a comprehensive review," *Bulletin of Electrical Engineering and Informatics,* vol. 11, no. 4, pp. 2169-2186, 1 Aug 2022.

[46] P. Prakasam, T. S. Kumar, T. Velmurugan and S. Nandakumar, "Efficient power distribution model for IoT nodes driven by energy harvested from low power ambient RF signal," *Microelectronics Journal,* vol. 95, p. 104665, 1 Jan 2020.

[47] H. Gujral, S. Mittal, and A. Sharma, "A novel data mining approach for analysis and pattern recognition of active fingerprinting components," *Wireless Personal Communications,* vol. 105, no. 3, pp. 1039-1068, 2019.

[48] M. E. Bayrakdar, "Employing sensor network based opportunistic spectrum utilization for agricultural monitoring," *Sustainable Computing: Informatics and Systems,* vol. 27, p. 100404, 1 Sep 2020.

[49] M. E. Bayrakdar, "Priority based health data monitoring with IEEE 802.11 af technology in wireless medical sensor networks," *Medical & Biological Engineering & Computing,* vol. 57, no. 12, pp. 2757-2769, 2019.

[50] J. Pettersson, "The Use of Decision Trees to Detect Alzheimer's Disease," p. 19, 2021.

[51] K. Xie, L. Han, M. Jing, J. Luan, T. Yang, and R. Fan, "Review of Intelligent Data Analysis and Data Visualization," *In International Conference on Broadband and Wireless Computing, Communication and Applications,* pp. 365-375, 28 Oct 2020.

[52] B. Petro, N. Kasabov and R. M. Kiss, "Selection and optimization of temporal spike encoding methods for spiking neural networks," *in IEEE Transactions on Neural Networks and Learning Systems,* vol. 31, no. 2, pp. 358-370, 12 Apr 2019.

[53] A. Ali, C. Sheng-Chang, and M. Shah, "Continuous wavelet transformation of seismic data for feature extraction," *SN Applied Sciences,* vol. 2, no. 11, pp. 1-12, Nov 2020.

[54] M. Saban, O. Aghzout, L.D. Medus, and A. Rosado, "Experimental Analysis of IoT Networks Based on LoRa/LoRaWAN under Indoor and Outdoor Environments: Performance and Limitations," *IFAC-PapersOnLine,* vol. 54, no. 4, pp. 159-164, 1 Jan 2021.

[55] L.S. Yousuf, "Nonlinear Dynamics Suppression of Bending Deflection in a Composite Laminated Plate Using a Beam Stiffener Due to Critical Buckling Load and Shear Force," *Applied Sciences,* vol. 12, no. 3, p. 1204, 24 Jan 2022.

[56] K. Pagels, "Classification of EEG Data Using Convolutional Neural Networks and the Scaled Reassigned Spectrogram.," *LUNFMS-4054-2021,* pp. 1-42, 2021.

[57] M.S. Hasan, A. Kordijazi, P.K. Rohatgi, and M. Nosonovsky, "Triboinformatic modeling of dry friction and wear of aluminum base alloys using machine learning algorithms," *Tribology International,* vol. 161, p. 107065, 1 Sep 2021.

[58] P. Jia, Q. Liu, and Y. Sun, "Detection and classification of astronomical targets with deep neural networks in wide-field small aperture telescopes.," *The Astronomical Journal,* vol. 159, no. 5, p. 212, 16 Apr 2020.