Masking Out Transient Network Issues in Sound Playback

Alexander Midlöv
al6220mi-s@student.lu.se

Department of Electrical and Information Technology
Lund University

Supervisor: Azra Abtahi Fahliani

Examiner: Maria Kihl

June 16, 2024

Printed in Sweden
Tryckeriet i E-huset, Lund

Abstract

Significant advancements in audio transport, encoding, and processing power in embedded systems constitute the foundation of today's speakers. Transitioning the digitalized audio data transmission to Audio over IP (AoIP) has opened the potential to transmit audio with the same network flexibility as other services using packet-based networks, such as voice over IP or video over IP.

This thesis primarily focuses on evaluating different masking techniques for transient network errors in embedded environments, such as an IP speaker using a simulated audio pipeline. The key indicators of success are the algorithms' latency, delay, perceived audio quality, and complexity. Both sender-based and receiver-based, as well as combinations of these methods, have been evaluated. Two techniques were sender-based (Redundant Transmission (RT) and Parity Packet Forward Error Correction (PPFEC)). Five were receiver-based (Silence Insertion (SI), frequency-dependent White Noise Insertion (WNI), Packet Repetition (PR), Waveform Substitution based on Pattern-matching (WSP), and Waveform Similarity Overlap and Add (WSOLA)). The sender-receiver-based techniques were the 10 resulting combinations of sender-based and receiver-based techniques.

Considering all the key indicators, the findings were that the best-performing algorithms for receiver-based Packet Loss Concealment (PLC), sender-based Forward Error Correction (FEC), and the combination sender-receiver-based were PR, PPFEC, and PPFEC with PR. PR had a mean opinion score (MOS) of 57.08 across all evaluated tracks at a drop percentage of 10% with a segmented cross-correlation score roughly at 0.5, which, compared to SI, only had a segmented similarity score in the range of 0.1 to 0.3 for the evaluated excerpts. Further, the execution time ratios between SI and PR were almost equal. Introducing FEC into the masking techniques further improved the results since the reconstruction made by FEC techniques perfectly reconstructs the lost segment if enough redundant data was transmitted correctly. However, implementing FEC using RT or PPFEC results in increased end-to-end latency due to the increased amount of redundant information transmitted, with RT causing a greater increase than PPFEC.

The conclusion was that when latency does not have to be ultra-low, the combination of PPFEC and PR will do a great job of masking occasional transient errors in low-resource AoIP-embedded systems. In the case of ultra-low requirements, using only PR is suggested since PR has low complexity, introduces no new latency, and does not further strain the network.

Acknowledgements

This Master's thesis was made possible by the Lunds Tekniska Högskola and Axis Communications. I want to thank my supervisor at Lund University, Azra Abtahi Fahliani, whose help and guidance have made this thesis a reality.

I would like to thank my industry supervisor, Zeid Bekli, for all the support and assistance throughout the thesis. I would further like to thank Duja El-Khamisi and the Axis New Bussiness audio team members for providing me with the opportunity to perform my thesis at Axis and for helping me with technical aspects during the thesis.

Last, I want to thank my brothers and my parents for all their guidance and support in life.

Popular Science Summary

In recent years, a lot has happened in the audio system industry. The forefront of technology is being pushed forward daily. With the vastly increased processing power in embedded systems (such as speakers or dongles), the developments of bandwidth power, and the evolution of the Internet of Things (IoT), a revolution has sparked regarding audio data transmission. The transition from traditional transmitting technology, analog or previously used digital transmission, to Audio over IP (AoIP).

AoIP has become a fast and resilient alternative for audio transmission and has found its way into several use cases. Whether for live music performances with ultra-low latency or safety-enhancing in cities, AoIP has found its way into several sectors of society. The technology is foremost used in professional industries, providing solutions for various applications. Some use cases include security regarding fast signaling of environmental disasters, warehouse announcements and background music, hospital announcements, producer equipment, live performance wiring, audio tied to surveillance in subways, and so on.

The sectors using AoIP do so because they want better transmission, relaying information faster, and utilizing the benefits tied to operating on a packet-based network. However, as with other packet-based networks, AoIP is sensitive to latency, disruptions, and transient network errors. Further, it is crucial to achieve and maintain good perceived audio quality. Robust systems are in high demand, which is why many in the industry have yet to transition from traditional transmitting technologies to more modern ones. In the neighboring field of Voice over IP (VoIP), the solution to mask the transient network errors has been researched for quite a while. Some of these techniques have been evaluated on music and other signals than speech, but even fewer have been assessed in AoIP systems in embedded environments.

This thesis investigates and evaluates some PLC and FEC techniques when implemented on an IP speaker.

Table of Contents

List of Figures

Abbreviations

AoIP - Audio over IP
BAQ - Basic Audio Quality
CNAME - Canonical Name
DSP - Digital Signal Processing
EVS - Enhanced Voice Services
FEC - Forward Error Correction
FFT - Fast Fourier Transform
HiFi - High-fidelity
IoT - Internet of Things
ITU - International Telecommunication Union
MIFEC - Media-Independent Forward Error Correction
ML - Machine Learning
MUSHRA - MUltiple-Stimuli with Hidden Reference and Anchor
PCM - Pulse Code Modulation
PLC - Packet Loss Concealment
PR - Packet Repetition
PPFEC - Parity Packet Forward Error Correction
PWR - Pitch Waveform Replication
QoS - Quality of Service
RNG - Random Number Generator
RMS - Root Mean Square
RS - Reed-Solomon
RT - Redundant Transmission
RTCP - Real-Time Transport Control Protocol
RTP - Real-Time Transport Protocol
SNR - Signal to Noise Ratio
SI - Silence Insertion
SSRC - Synchronization Source
TCP - Transmission Control Protocol
TSM - Time Scale Modification
TTL - Time To Live
UDP - User Datagram Protocol
VoIP - Voice over IP
WNI - White Noise Insertion
WS - Waveform Substitution
WSOLA - Waveform Similarity Overlap-and-Add
WSP - Waveform Substitution based on Pattern-matching

Chapter 1
Introduction

Significant advancements in audio transport, encoding, and processing power in embedded systems constitute the foundation of today's speakers. Transitioning the digitalized audio data transmission to AoIP has opened the potential to transmit audio with the same network flexibility as other services using packet-based networks, such as voice over IP or video over IP. Several competing companies have introduced new IP speakers and IP-based audio technology, pushing the boundaries of what AoIP is capable of every day [1][2]. The technological development strives to increase the perceived audio quality, reduce latency and jitter, improve transmission speed, and minimize packet loss. These factors are critical indicators for AoIP systems and speakers [3]. Adjustments need to be carefully implemented and revised. If the audio quality is increased, more bandwidth is required, or there is a need for higher latency, and the risk of packet loss increases, leading to perceptual effects like glitches, disruptions, and distortions. Meanwhile, minimizing packet loss comes at the cost of likely increasing the latency or hurting the audio stream's perceived quality. Thus, it is clear that there is a challenge in balancing these metrics.

1.1 Motivation

This master's thesis is done in collaboration with Axis Communications in Lund, Sweden. Axis Communications develops, builds, and sells AoIP equipment like IP speakers. In doing so, a need to investigate and evaluate options for masking out transient network issues in an embedded environment has emerged. Network-related issues might manifest audibly for the listener as glitchy, distorted, or even missing sound segments.

Previous research regarding masking transient network issues specifically for AoIP is somewhat limited [3]. The research has primarily focused on similar areas, such as voice-over-IP and video-over-IP. Some concepts are shared between the areas, but some differences must be considered. AoIP delivers all digital audio, which can be speech, music, alarms, and so on. This differs from VoIP, where human speech is only considered. There are natural micro-pauses regularly occurring during human speech. Meanwhile, music, in general, has melodic continuity. Solutions working for VoIP may thus not necessarily work for AoIP. Given the limited resources available in low-power and small storage embedded environments,

evaluating what algorithms work well is thus of interest.

1.2 Research Goal

This thesis investigates and evaluates various receiver, sender, and sender-receiver-based methods to mask transient network errors in an embedded environment. The primary objective is to apply and evaluate different Packet Loss Concealment (PLC) and Forward Error Correction (FEC) techniques in an Audio over IP (AoIP) speaker device to mask the artifacts caused by packet loss. The thesis focuses on finding solutions suitable for running in environments where ultra-low latency is crucial while power and storage are limited. Evaluations are made with regard to waveform similarity, Signal-to-Noise-Ratio (SNR), perceived audio quality, introduced latency, and execution time comparisons. By performing this investigation, this thesis aims to contribute to the further development of AoIP technology.

1.3 Thesis Outline

The structure of this report follows this outline: In Chapter 2, the background of key aspects of audio and network-based audio transmission is presented with concepts such as psychoacoustics, audio processing, packet-based transmission, and audio pipeline. In Chapter 3, there is a review of masking techniques from previous work in the fields of PLC and FEC. The techniques are introduced and explained. Chapter 4 begins by discussing the chosen selection of algorithms, why they were chosen, how they were implemented, and parameter settings. How latency is affected by the selected parameters is also discussed. In Chapter 5, the developed simulation pipeline is shown. The equipment, audio configurations, evaluation methodology, and other setup-related information are also disclosed. Further, all the experiments that were conducted are presented. Chapter 6 is comprised of the presentation and the discussion of the results. The receiver-based results are first presented and discussed, and then the results of sender and sender-receiver-based are presented and discussed together. In Chapter 7, a short summary and the conclusions drawn in the thesis are presented. Lastly, the potential directions for future work are discussed. The Appendix includes images and further details of the webMUSHRA tool used for the listening tests.

Chapter 2
Networked Based Audio

This chapter presents relevant background on audio and network-based audio transmission. The information presented includes a rough overview of the parts of the audio pipeline, psychoacoustics, and how audio is represented in packet-based networks.

2.1 Audio Over IP

2.1.1 Background

Technological advancements made it possible for audio to traverse packet-switched networks, and this is precisely what AoIP is. Designed to achieve audio stream deliveries for professional applications reliably and with low delay [3]. Using packet-based networks to transport the audio data opened up possibilities for more advanced services and smoother and easier inter-connection possibilities. However, the transition to AoIP did come at the cost of technical challenges due to the stochastically varying nature of these networks [4]. An early investigation based on voice over IP (VoIP) showed that the most prominent challenges affecting the end-to-end quality were packet loss, end-to-end delay, and jitter [5]. Tackling these challenges will result in High-Fidelity (HiFi) audio with the lowest possible delay.

2.1.2 Quality Of Service

Quality of service (QoS) is a phrase that is often seen in the context of IP networks. QoS is a quality describing indicator with a basis in the following four categories: bandwidth, dropped packets, delay, and jitter [3].

Jitter And Jitter buffers

Jitter is the difference in the time between the earliest and latest arrived packets relative to the intended arrival times and can vary in duration [3]. Jitter occurs because packets are not necessarily received in the same order as they are transmitted. Network routing, packet queues, and packet loss are some factors that cause delay jitter. The problem is usually solved in the receiving part of the system with a buffer that delays incoming packets so the packets can be arranged in the correct order and then be released further downstream in batches. Buffers

like that are called jitter buffers and compensate for the jitter with delay [6]. The jitter buffer will rearrange the RTP packets according to their sequence numbers before sending them downstream in the pipeline. Further, jitter buffers require precise configurations since packets held too long will cause an interruption in the playback because some packets will arrive too late and thus be dropped. Meanwhile, jitter buffers with a maximum buffer time that is too short run the risk of missing out on some packets and sending the rest of the batches downstream too soon. Figure 2.1 shows a simplified overview of the information flow.

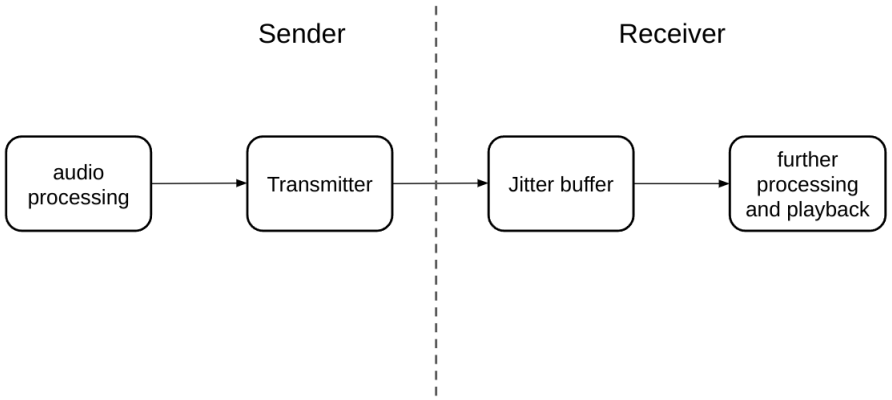

Figure 2.1: Simplified information flow to and from the jitter buffer.

Delay and Latency

Delays of various lengths occur in different steps along the audio stream pipeline. Some factors that might introduce delay are routers, propagation delay through physical links (usually not affecting to any notable degree), jitter buffer delay, and other processing delays [3]. Latency, however, incorporates all the delays and corresponds to the time it takes for a packet to travel from its source to its destination.

Packet Drop

As packets propagate the network, they run the risk of being dropped. Routers can drop packets and will do so for various reasons [3]. They will sometimes drop if a link is overloaded or a packet's Time To Live (TTL) has hit 0. Packet drops might cause disruptions or glitches in the audio stream.

2.1.3 Codecs

Audio codecs are used to compress and reconstruct audio signals. An audio codec system is, in general, made up of three components. These components include an encoder, a quantizer, and a decoder [7]. These play an essential part in the audio pipeline as they directly impact the delay, bit-rate, and bandwidth usage of the information flow [3]. Several codecs have been developed over the years

with various encoding schemes like MP3, AAC, FLAC, etc. It is also possible to digitally represent audio data without using previously mentioned codecs by using Pulse Code Modulation (PCM) with specific bit depth such as PCM-16, PCM-24, or PCM-32 [8]. The numbers following PCM tell which bit depths the signal was quantized with. Characteristics of the ideal IP codec system would include effective and good PLC, a dynamic receive buffer, a mini-maxed bitrate to achieve the best possible fidelity, and an adaptive codec bitrate. Different PLC techniques are thus inherently a part of the audio codec system. Information leaving the jitter buffer typically proceeds through the decoder before proceeding to playback or further signal processing.

2.1.4 Data transmission

Transmitting audio data over the network using IP packets requires synchronization and time precision for playback to be correct. This is even more challenging in large-scale installations where several speakers are to receive and play synchronized audio streams. A transport protocol, the Real-Time Transport Protocol (RTP), is a popular alternative to achieve high precision with fast and secure transmission. RTP was explicitly made to transport real-time data such as audio or video streams [9]. The RTP header contains a sequence number, timestamps, and audio or video encoding. RTP is mainly run over the User Datagram Protocol (UDP) to ensure fast transmissions without retransmission to keep latency down. The protocol can also run over Transmission Control Protocol (TCP), but the added latency makes the TCP-RTP combination less viable for most Real-time applications. The RTP packet header can be seen in figure 2.2.

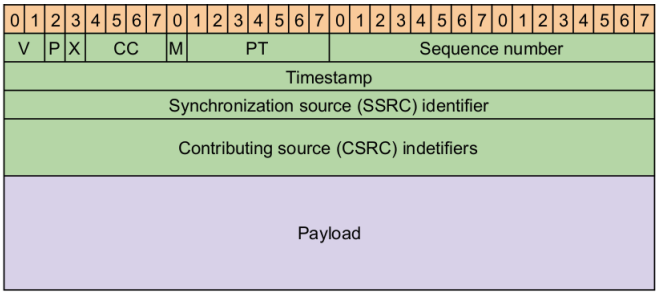

Figure 2.2: RTP packet overlook

RTP packets include a payload field intended for the audio data. The information included in this field will be processed and eventually played by the receiving end, and the data format is identified via a 7-bit numeric identifier [9]. The audio data can be encoded in various formats, such as MP3, AC3, PCM, and more.

Feedback related to the performance of the RTP streams is essential and is possible through the Real-time Transport Control Protocol (RTCP). RTCPs are sent periodically to minimize bandwidth usage, and they uphold four fundamental mechanisms: [9]

1. Informs the quality of the data distribution (congestion control, for example).
2. Carries identifiers for RTP sources, known as Canonical Names (CNAME), to maintain robust RTP streams. CNAME is used if the Synchronization Source (SSRC) identifier changes (this could happen when a conflict is discovered, a service is restarted, or to associate multiple data streams).
3. Calculate what rate packets are sent by observing the participants.
4. The last process is optional and aims to convey minimal session control information. This allows a monitoring entity or a participant in the network to display participant identification or other information in a user interface.

Data transmission in AoIP systems can be directed to one receiver or, through multicast, reach several receivers in a scalable and efficient way [10]. This feature is typically used in larger AoIP systems where several speakers and other potential audio systems are connected. This further highlights the transport layer's need for precise synchronization and timestamps.

2.1.5 Data receiving

RTP packets over UDP are not guaranteed to reach their destination in time or at all [9]. Packet gaps in the jitter buffer can and likely will occur to a lesser or greater extent, depending on the available bandwidth, configuration, or other packet-based network challenges like packet routing. Correcting or concealing these gaps is crucial for the audio stream to stay synchronized and for playback to proceed correctly.

2.2 Psychoacoustics

Psychoacoustics is the science of the hearing system as a receiver of acoustical information [11]. Digital signal processing (DSP), audio transmissions, and other audio configurations should be opted according to the human ear, which ultimately is the end-receiver. The audible spectrum has been mapped to roughly cover the frequencies between 20 to 20000 Hz [12]. A lot of further work has been made in characterizing the human auditory perception [11][13][14]. Processing audio with codecs or using DSP heavily relies on concepts in psychoacoustics since the human ear and sound perception are affected by these processes. This is also true for the masking of transient network errors, as PLC and other masking techniques directly affect the audio signals waveform. Audio coders are even seen to perform compression by exploiting signal information not detectable by the human ear. VoIP error concealment techniques have been seen using similar exploits to make lost packets perceived as less noticeable [15].

2.3 Audio processing

Audio processing is of great importance in the context of AoIP. Since packet loss can occur, audio data will end up missing in playback. There are different potential

approaches to mitigate this problem. One way is by using DSP, by applying filters or interpolations, which have been studied for a long time [16]. Further, DSP can also be used to increase the perceived audio quality. Other methods to mitigate packet loss involve manipulating packets or data at the transmission's sending or receiving end. Several methods are available for both approaches, and techniques can even be combined in some cases [17].

Chapter 3

Error Masking and Repairing Techniques in the Literature

In this chapter, masking techniques investigated in related works are presented. There is not a clear division between concealing and repairing techniques. They are usually split according to the categories sender-, receiver-based, and sometimes sender-receiver-based or similarly phrased divisions [17][18]. This division is strictly focused on where they are processed. According to Church and Pizzi, a state-of-the-art solution to an ideal low-delay and bidirectional IP codec system should encapsulate these four characteristics [3]:

- Effective, inaudible packet-loss concealment.
- Adaptive receiver buffer with time squeeze and stretch capability
- An efficient codec with maximum fidelity to the lowest bitrate.
- Adaptive codec bitrate, adapting according to network conditions.

In the closely related field of voice-over-IP, packet loss concealment is being studied and worked on by Enhanced Voice Services (EVS) with their EVS codec that is based on state-of-the-art speech and audio coding [19]. Among these techniques, a fair share is based on harmonics and pitch, which are essential in reconstructing missing packets in the context of speech.

3.1 Receiver-Based PLC Techniques

Receiver-based schemes use the received information to conceal or repair lost packets. The opting not to use concealing and or repairing techniques is called splicing. Splicing is the most straightforward approach in solving lost packets since the receiver proceeds with the first available packet after the lost packet. The approach has been shown to perform poorly and causes synchronization issues with buffers and during playback [20][17]. Other methods were derived to mask lost segments better. Receiver-based techniques work best with small isolated gaps in the received data and should not be seen as substitutes for sender-based repair techniques.

3.1.1 Insertion-based Error Concealment

Insertion-based error concealment replaces a lost packet by inserting a generated packet. Different approaches can be taken when replacing packets. An introduction to the three most famous ones follows.

Silence Insertion

The baseline compare method inserts silence in case of missing packets so that the signaled is not spliced [21][17]. This has typically been used as a baseline historically in VoIP [22]. In the case of PCM-encoded audio, it generates frames containing only 0s [8]. Filling gaps with silence works well if the loss rates are below 2 percent and the packet lengths are shorter than 4ms [23]. A visual representation of silence insertion can be seen in figure 3.1.

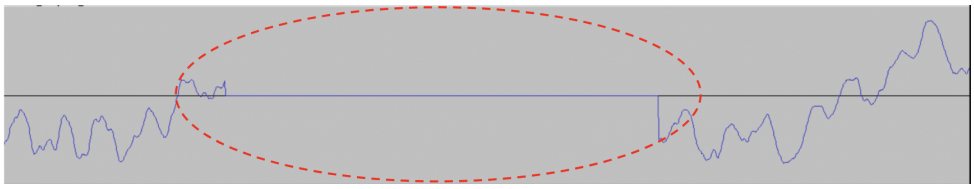

Figure 3.1: A snippet of a waveform that has replaced lost frames with silence using silence insertion.

Packet Repetition

PR is a straightforward algorithm that tracks the latest successfully received packet and repeats this packet in case of lost segments to fill the gap [24]. With small gaps, this works rather well. In the VoIP environment, the GSM system recommended complete repetition for the first lost speech frame followed by the same packet but with a successively more faded-out gain over the next 320ms [25].

Noise Insertion

Noise insertion involves filling lost segments with noise. Research in VoIP environments suggests that using white noise during segments with silence is more effective [26]. Background noise instead of complete silence can be more effective since it allows the human brain to subconsciously restore phonemes through psychoacoustic and provide better-perceived quality, unlike silent packet replacements [27][28]

3.1.2 Interpolation-Based Error Concealment

Interpolation-based error concealment uses packets around the lost segment to produce a replacement. These techniques consider dynamic characteristics when performing the concealment [17]. Three variations of interpolation-based techniques have generally been considered before. These are Waveform Substitution (WS), Pitch Waveform Replication (PWR), and Time Scale Modification (TSM).

Both WS and PWR are pretty similar in that they utilize a form of packet repetition by finding the best-fitting substitution through pitch detection, pattern matching, or both. [29][30]. PWR is essentially a development of WS that considers the pitch surrounding the lost segment. TSM, on the other hand, has been seen to perform even better than WS and PWR, but at the cost of increased complexity [31]. Using different waveform stretching algorithms, TSM stretches the waveform in time and is used to stretch the waveform on either side of the loss across the lost segment.

Waveform Substitution

A study concluded by Goodman et al. used templates to perform repetition-based waveform substitution through pattern matching (WSP) and evaluated the WSP from both the one-sided and the two-sided approach [29]. The study showed waveform substitution outperforming simple packet repetition and silence replacement in packet-based voice systems in both cases. The repetition is based on pattern matching. Three different approaches for pattern matching were presented in their research, and the different methods yielded almost equivalent results, according to Goodman et al. One of the presented pattern-matching methods has a simplified version of the formula that could be useful in practical implementations. The essence of the pattern-matching technique is to use a template and search for a fitting replacement for the lost packets in a search window using normalized samples. In the study, Goodman et al. used the last m samples from the previously received packet as the template. The cross-correlation formula method and the simplified version can be seen in equations 3.1 and 3.2, respectively. M = #samples in the template, N = length of the search window in which the M most template fitting samples are to be found, n is the position of the template in the search window, $x(i)$ and $y(i)$ correspond to the samples of the template and the samples of the search window. The result of the cross-correlation function is thus the n that gives the most significant value in $C(n)$. Further, sgn is the sign-function.

$$C(n) = \frac{\sum_{m=1}^{M} x(m)y(n+m)}{\sum_{m=1}^{M}[y(n+m)]^2}, \text{ where } n = 1, 2, 3, ..., N \tag{3.1}$$

$$S(n) = \sum_{m=1}^{M} sgn[x(m)]sgn[y(n+m)], \text{ where } n = 1, 2, 3, ..., N \tag{3.2}$$

Once the correct position for the template is found, the data following the position for the template position is used as replacement data for the lost packet or packets. Goodman et al. suggested a search window length of 16 ms and a template size of 4 ms regardless of packet size for the one-sided approach [29].

Waveform Similarity Overlap-and-Add

Waveform Similarity Overlap-and-Add (WSOLA) is an effective TSM technique that operates in the time domain without modifying pitch or timbre [32][33][34][35]. The technique stretches the signal before a lost segment across it. See figure 3.2. In the Python library, PyTSMod [36], the WSOLA algorithm has been based on the

work of Dreidiger and Müller [37]. The parameters should be adjusted according to the specification of the audio that is fed to the algorithm. Typically, a Hanning window is used with a size big enough to cover the audible frequencies included in the signal, and a window overlap of 50% should be used [38][37].

The WSOLA PyTSMod implementation begins by validating the parameters. The algorithm continues by generating a window function and determining the output length. Synthesis windows are then created, and interpolation is used to find frame positions for analysis. The algorithm processes each channel separately, iterating through each frame, extracting and aligning them. For each frame, the WSOLA adjusts the position to align by utilizing the previous frame. The current frame gets added to the output signal after it has been extracted and windowed. The progression of the current frame is calculated, and cross-correlation is used to find the continuation with maximum similarity. Once all iterations are done, the output is normalized and trimmed to the desired length.

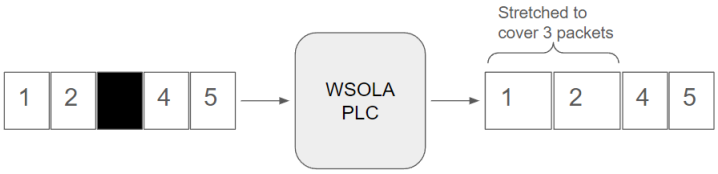

Figure 3.2: Illustration of WSOLA stretching packets across the lost segment.

3.1.3 Machine Learning Based PLC

In recent years, Machine Learning (ML) based PLC has become the center of attention when it comes to repairing and concealing packet loss in multimedia, with several studies trying to find optimal models [39][40][41]. Different models have been evaluated so far, yet newer and better-performing models pop up consistently yearly. Autoregressive models have so far shown to be useful in predicting information from lost segments [42]. The demand for finding a state-of-the-art solution is high. The field has even seen large-scale competitions in finding the best models [43].

3.2 Sender-Based Techniques

In contrast to receiver-based techniques, sender-based techniques try to correct packet loss by including information that can be used to repair missing segments or transmit redundant information.

3.2.1 Forward Error Correction

FEC is a sender-based technique where redundant information regarding other packets is included in the payload. This allows the decoder to better correct errors without the need for retransmission. Different strategies can be applied

when working with FEC. A straightforward take on FEC uses critical information about the signal, allowing the receiving end to repair and replace missing packets better. One approach is to include zero-crossing measurements to interpolate values better, which have been shown to perform well with waveform interpolation by N. Erdöl et al. [44]. Another method is the Media-Independent FEC (MIFEC) method Parity Packet FEC (PPFEC), which can patch solitary scattered losses through XOR of the parity packet and other packets belonging to the same block [45]. A more sophisticated approach in FEC is the well-known Reed-Solomon (RS) codes. Prominent at error-correcting and performs well against burst packet losses [17]. This method is based on converting the message to a polynomial, m with coefficient from a finite field F of degree n, and let RS encoders evaluate m at n different points to generate codewords [46][47]. Though far more complex than the previously mentioned methods, it can perform well at burst losses.

Parity Packet FEC

One relatively straightforward FEC technique studied by N. Shacham [45] is based on including an error-control parity packet for every K packet of media data. In this approach, K packets form a block, and as long as only one packet per block is missing, the technique can restore the lost packet. The PPFEC technique first requires selecting the block size, K. The error-control parity packet, P_E, is then computed using equation 3.3.

$$P_E = c_{i,K+1} = (\sum_{j=1}^{K} c_{i,j}) \mod (2), i = 1, 2, 3, ..., m \tag{3.3}$$

Further, given that a single packet is missing in the block, $\overline{P} = c_{i,a}$ where $a \in (0, 1, ..., k)$, the decoder can reconstruct the packet using the other received packets. A simple overview of the technique can be seen in figure 3.3. The parity of the received packets P, including the error-control packet, will result in the missing packet being a complement to P. P can be calculated as seen in equation 3.4.

$$P = c_{i,K} = (\sum_{j=1}^{K} c_{i,j}) \mod (2), i = 1, 2, 3, ..., m \text{ and } j \neq a \tag{3.4}$$

The summation seen in equation 3.5 thus holds.

$$P_E = (P + \overline{P}) \mod (2) \tag{3.5}$$

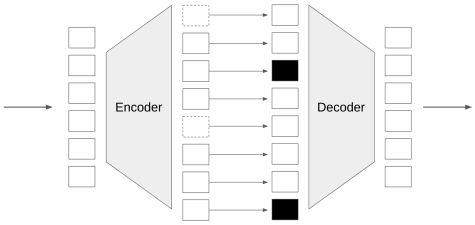

Figure 3.3: Overview of PPFEC.

3.2.2 Redundant Transmission

Redundant Transmission (RT) is perhaps the most straightforward method among the sender-based techniques. It is a simple protection approach against packet loss. The idea behind the scheme is to transmit two or more of every packet, thus reducing the chance that loss occurs while the packets are being transmitted. This comes at the cost of further straining the network by increasing the traffic on the links, which will further increase packet loss probability [17]. It also increases the end-to-end latency since the receiver must receive all those extra packets to attain the same level of information as a transmission without RT requires. RT has thus also been implemented with dynamic redundancy levels that depend on the current drop rate relayed by RTCP packets [48].

3.3 Sender-Receiver-Based Techniques

Sender-receiver-based techniques cover PLC and packet-repairing methods that span across the categories of sender-based and receiver-based techniques. They include, but are not limited to, the combination of sender and receiver-based techniques. They also include methods that use network-specific settings and setups to prevent gaps and errors in the transmission [49][18][50]. These schemes rely on concealment and packet repair, codec-specific information, or other network information to conceal or repair missing segments.

3.3.1 Regeneration Based or Codec Based Error Concealment

Regeneration-based error concealments have been categorized as receiver-based methods [17]. Sender-receiver-based is a more fitting category since the sender typically includes codec-specific or other information used at the receiving end to regenerate the lost segments [18]. These techniques are quite computationally expensive but have been seen to yield quite good results [51][52].

Chapter 4

Implemented Techniques and Parameter Settings

In this chapter, the selection of algorithms is explained. The chapter also discloses how the algorithms have been implemented and the reasoning behind parameter settings. Lastly, the latencies introduced due to the parameter selections are mentioned.

4.1 Technique Selection

Several factors were considered when selecting the techniques to be evaluated for the thesis. Considerations were made to evaluate non-codec-dependent techniques, allowing for AoIP system independent appliance. Techniques should work with signals represented using PCM. Further, it was preferred to evaluate fast techniques that require minimal latency and preferably no packet delay for the algorithm to work. Limitations in manpower and time were pressing factors in selecting the number of algorithms to be evaluated. Another factor was the limited resources of open-source PLC implementations available online. The latest contributions in the field, ML-based PLC, were not considered either. This was due to several reasons. It would lead to less time to evaluate other techniques since converting and implementing models in a network speaker device, finding the correct selection of parameters, and training them are highly time-consuming.

4.1.1 Receiver-Based Selection

Since codec-specific algorithms were not considered in the evaluation techniques, the evaluation would include interpolation and insertion-based techniques. As seen in the literature review, the insertion-based techniques, PR and White Noise Insertion (WNI), performed better in VoIP than silence insertion and are neither complex nor computationally heavy [17], and were thus chosen.

The more sophisticated interpolation-based techniques were selected according to the required packet delay, latency, complexity, and concealing performance. It was interesting to investigate how well the interpolation-based techniques would perform with audio in general (music and speech) and in an embedded environment with limited resources. Seeing as PWR was a refinement of WS but more specified

towards VoIP due to its consideration of silenced speech segments, the WS method waveform substitution based on pattern matching by Goodman et al. was selected for evaluation [17][29][30]. Among the TSM methods, the WSOLA was seen as the basis for the techniques [31][53][54]. Newer versions built on the WSOLA error concealment technique introduced more complexity, further delay, or, in some cases, both. Still, they did, according to Lizhong et al. and Yeh et al., perform slightly better than the original WSOLA algorithm. However, since they only performed marginally better, they introduced more complexity and delay. Thus, the original WSOLA algorithm was selected.

PR Parameters

PR was implemented to reuse the payload of the latest successfully received packet in case of loss. As previously mentioned, the GSM system advocated the application of a fade-out of the gain of over 320ms for packet repetition when used with speech. However, this was not used to implement the PR algorithm in this investigation, as burst losses of that kind are not the main focus of this thesis.

Frequency Filtered WNI Parameters

The result was relatively poor when trying out the white noise appliance in the context of music. While the rest of the audio was continuous and melodic, the sudden insertion of white noise became a harsh contrast. To attune to this, two adjustments were made. Firstly, the root mean square (RMS) of the previously received packet was calculated, and the samples were then Fast Fourier transformed (FFT) so that the dominant frequency f_D could be found. The largest and smallest frequencies with intensities within $\pm 0.8 \cdot f_D$ were used as parameters for two cascading second-order Butterworth bandpass filters with amplifiers in between. This is to increase the steepness of the frequency response since only theoretical filters have a rectangular shape of the transfer function [55]. The samples are amplified after each filter by the difference in RMS before and after passing through the filters. This keeps the gain roughly the same level as the previous packet. Once this is done, the WN that is left is based on the most prominent frequencies of the prior packet. This is then used to fill the gap left by the lost segment.

WSP Parameters

The WSP was implemented using the more practical pattern-matching technique in 3.2. The results of the two pattern-matching techniques were similar in concealing performance, so the one with the least complexity was chosen. The WSP parameters were chosen according to the recommendations made by Goodman et al. [29], a window size of $N = 16ms$ and a template size of $M = 4ms$. These were considered the best choices for the parameters regardless of packet size.

WSOLA Parameters

The WSOLA PLC algorithm evaluated for this thesis followed the open-source implementation in the Python library PyTSMod. The algorithm is applied once

a lost packet is detected. The selection of the parameters has a big impact on the resulting signal.

A Hanning window with a size of 2048 sample was used since the window should cover one pitch period. With a window of 2048, the period of frequencies down to 23.44 Hz is covered when sampled at 48 kHz. $\Delta f = \frac{48000}{2048} \approx 23.44$ Hz. This almost covers the entire audible spectrum but also considers the increased performance cost of having large windows. The hop size was set to have a 50% overlap with the windows and was thus set to 1024. The tolerance was set to 512, as this is the default for the algorithm. The technique was set to use the two latest packets to reconstruct a missing segment by stretching them across the lost segment. The algorithm thus requires a delay of two packets to function properly.

4.1.2 Sender-Based Selection

The sender-based techniques considered were media and codec-independent and did not require specific LAN configurations. The algorithms considered were RT and PPFEC, and they were chosen to be implemented without the need to reconfigure settings outside the sender and the receiver. SI was used if the techniques could not reconstruct the messages to avoid splicing.

Redundant Transmission

RT was implemented with the static redundancy of 1 copy per every original packet. This was based on the fact that the drop rate would be static for the different investigations. If both packets arrive, the copy is ignored and thrown away.

Parity Packet FEC

The PPFEC was implemented according to N. Shacham's single parity packet discussed in section 3.2.1. The block size was set to $K = 6$ to balance the introduced latency from the long waiting times required by the jitter buffers and the introduced latency and strain of transmitting too many parity packets with a smaller block size.

4.1.3 Sender-Receiver-Based Selection

The selection of Sender-Receiver-Based algorithms consisted of combining the selected sender and receiver-based techniques. This decision was based on the available resources and the fact that codec-specific algorithms were not evaluated. Combining techniques allows the sender-based techniques to restore the lost segment perfectly when possible. In contrast, the receiver-based techniques perform concealment in case the sender-based techniques cannot patch the lost segments. This should lead to fewer concealments having to be made. Including the sender-based techniques, which are sender-receiver-based with SI, there were 10 combinations in total between the sender and receiver-based algorithms: RT-SI, RT-PR, RT-WNI, RT-WSP, RT-WSOLA, and the corresponding ones for PPFEC: PPFEC-SI, PPFEC-PR, PPFEC-WNI, PPFEC-WSP, PPFEC-WSOLA.

4.2 PLC Waveform Alignment

The audio signals waveform could end up with non-aligned edges after concealing lost segments. These misalignments were notable when applied with a music audio signal, causing listening experiences similar to SI due to audio clipping. Clipping occurs when the signal's amplitude exceeds the system's maximum level [56]. An example of misalignment due to PR can be seen in figure 4.1. Goodman et al. used cosine weight profiles to handle misalignments in implementing WSP [29]. A similar approach was used for all the receiver-based PLC algorithms that relied on fading. The scheme fades the first 40 samples of the newest received packet after a lost segment or the first 40 samples of the reconstructed packet for PR, WNI, and WSP. The fading algorithm, $sample_i = (sample_i \cdot f_i) + (sample_{prev} \cdot (1 - f_i))$, where f_i is the fade factor at iteration i and $sample_{prev}$ is the last sample from the previous packet.

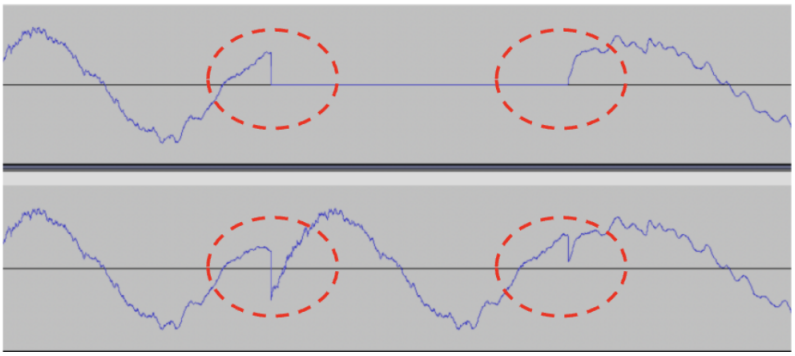

Figure 4.1: An example of PR waveform misalignment.

4.3 Technique Latency Requirement

Other latency-inducing phenomena occurred except for the introduced time required by the techniques to apply the PLC. Among the selected techniques, WSOLA, PPFEC, and RT introduce extra latency so that they work correctly. These latencies are configurable and depend on the chosen parameters for the methods. The latencies described here are based on the configurations presented with the techniques. WSOLA requires a packet delay of 2 packets to stretch the two latest packets across the length of three.

With PPFEC and RT, latency is introduced in terms of the jitter buffer having to wait a particular time to allow the buffer to at least have had the time to wait for enough packets to arrive before sending them further downstream and in terms of the end-to-end latency introduced by transmitting more packets. PPFEC requires sufficient time for the entire block to arrive before letting the information continue to flow downstream. In this case, at least a block must have had the chance to arrive before proceeding. Continuing with RT, the latency arises from the same reasons as for PPFEC and depends on how many redundant packets are transmitted per packet. RT was configured only to send one copy per packet.

To send 60 packets of pure data, RT will transmit 120 packets and PPFEC 66 packets.

Chapter 5

Developed Pipeline and Evaluation Methodology

In this chapter, the developed audio pipeline and the methodology for performing the evaluations are presented. Further, the audio and network configurations used for the investigation and the performed experiments are described.

5.1 Equipment

5.1.1 Audio equipment

The experiments conducted for this investigation have been in collaboration with Axis Communications. Specifically, they provided me with equipment to perform the implementations and executions at an AoIP speaker device. According to a publicly available datasheet, the speaker device had 1024 MB RAM and used the processor NXP i.MX 8M Nano [57].

5.1.2 Audio Specifications

Due to the tremendous potential selection of configurations available with signal processing of audio, limitations regarding the audio used for the investigation had to be made. The audio used throughout all of the investigation used the same configurations. The audio was represented using 16-bit PCM encoding with a sampling frequency of 48kHz, which is more than enough to cover the audible frequency spectrum for humans [12]. Further, all the audio used during this thesis is mono-channeled.

5.1.3 Network Configurations

The experiments are based on error-repairing and error-concealing technologies. Measurements require access to a constricted network, which can also be simulated. Heavily trafficked networks are unreliable work environments. The simulated approach was thus used for the experiments. The approach to simulate a lossy network environment was by dropping a certain percentage of packets. A packet dropper was implemented in the pipeline for these experiments to obtain the measurements and results. The packet dropper drops a specified percentage

of packets based on a seeded random number generator (RNG), thus creating the same baseline conditions for all the evaluations. The implemented packet dropper corresponds to the simulated network transmitter in the implemented audio pipeline, as shown in figure 5.1. The packets are RTP packets with lengths of 16 ms.

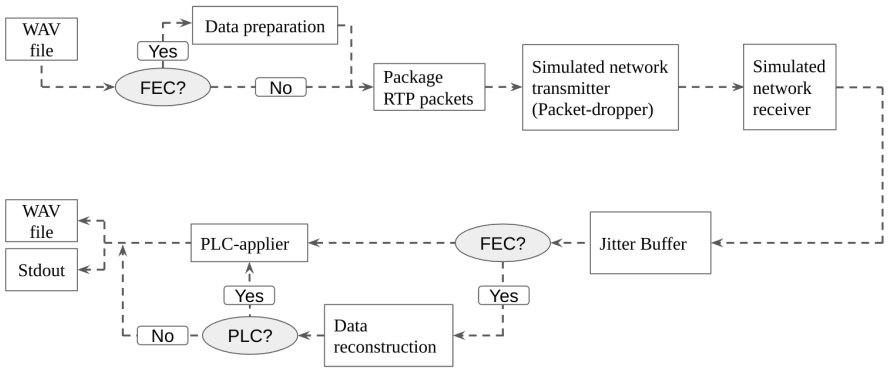

Figure 5.1: The implemented pipeline at device-level, writing results to wav-files or directly to stdout.

5.2 Simulated Audio Pipeline

Due to various factors, the pipeline used in the experiments, figure 5.1, was directly implemented at a speaker device running a Linux OS with embedded programming. The pipeline can be seen as an approach to simulating an actual audio pipeline in devices like the ones used in this investigation. It is far less complex because it contains fewer pipeline elements and allows for easier integration of different PLC and FEC techniques than in the actual audio pipeline.

The audio is read from a WAV file. Depending on whether the technique being evaluated uses FEC, data will be prepared before being packaged into RTP packets. Once the payload has gotten their respective RTP IP headers attached, the packets proceed to the simulated network transmitter, where a given percentage is dropped. The remaining packets are collected in the simulated network receiver, acting as a Jitter buffer, aligning the received packets according to the sequence number in their corresponding RTP headers. Once aligned, the PLC, FEC, or both categories of techniques are applied to conceal or recover the missing information. Once the packet loss has been concealed or reconstructed, it is either written to the standard output (stdout) for audio of the device or a WAV file.

5.3 Evaluation Methodology

The work methodology involves studying and evaluating the different transient network masking techniques according to the key performance indicators. Investigating performance requires measuring and analyzing the following parameters: resulting audio quality, latency, jitter, and computational cost.

5.3.1 Audio quality evaluation

Two approaches are used to evaluate audio quality. The objective approach analyzes the SNR and the segmented cross-correlation CC_{SEG} (waveform similarity) between the original and reconstructed signals at the specific intervals of the lost packets. Using SNR to compare the original signal with the reconstructed signal allows us to detect how much noise the techniques have introduced. It is a measurement used in previous investigations of similar character as this one [58][59]. The SNR is calculated using the reference signal S_{ref}, and the condition signal S_{cond} as seen in the equation 5.1.

$$SNR = 10 \cdot \log_{10} \frac{\sum_{n=1}^{N} (S_{ref}(n))^2}{\sum_{n=1}^{N} (S_{ref}(n) - S_{cond}(n))^2} \tag{5.1}$$

The CC_{SEG}, on the other hand, calculates the average maximum of the cross-correlation between the reference signal and the condition signal at the affected segments. This, in turn, tells us how similar, on average, a condition signal is to the original signal at the affected segments [60]. Since most of the signal are not affected by applying the PLC techniques at the lost frames, the average maximum correlation between segments gives a more concrete similarity assessment at the points of interest. The CC_{SEG} is calculated according to equation 5.2, where M corresponds to the number of segments.

$$CC_{SEG} = \frac{1}{M} \sum_{j=0}^{M-1} \max(\sum_{n} a_{n+k} \cdot \overline{v}_n) \tag{5.2}$$

5.3.2 Perceived audio quality

Perceived audio quality is heavily subjective. As the metrics described in section 5.3.1 do not reliably serve as a precise tool in assessing perceived audio quality, MUltiple-Stimuli with Hidden Reference and Anchor (MUSHRA) listening tests were conducted using the webMUSHRA tool [61]. This test form has been evaluated and proven to be a reliable framework for all listening tests. Further, the webMUSHRA configurations make the tests compliant with the ITU-R Rec. BS.1534-3 [62].

5.3.3 Jitter, Latency, and Synchronization

Delay measurements were obtained by observing the parameters the different algorithms require to function correctly. Latency is when the signal needs to be delayed to make the algorithms behave correctly. Some techniques require information from a packet with a higher sequence number than the currently most recent one to perform PLC or FEC. Latency also affects the synchronization between speakers in multiple-speaker environments since the packet containing the

same audio information needs to reach the playback simultaneously for the speakers. This means the time the jitter buffer will have to wait for late or missing packets is directly tied to the selected FEC or PLC technique. Latency across the audio pipeline will have to take the additional jitter buffer waiting time and the time required for reconstructing or concealing missing segments so that the timestamps of the audio frames are synchronized once they reach playback. The multimedia framework GStreamer solved the synchronization issue by performing a latency query to all the sinks in its pipeline for the maximum latency [63]. This means that minimizing the time required by the techniques in the jitter buffer and their complexity is necessary for achieving low latency in a multi-speaker environment with synchronized speakers.

5.3.4 Execution Time Evaluation

The initial intent was to measure both the CPU performance of the NXP i.MX 8M Nano in real-time and the ratio of times for the different techniques using the original audio pipeline. Since the pipeline had to be simulated and run as a program on the embedded environment instead of as a plugin or service, measuring the CPU performance would not accurately measure the individual techniques. Only the execution time ratio will be used to evaluate the rough estimates of execution times.

5.4 Experiments

Several experiments were conducted to obtain the results. Experiments were run on music and speech signals to evaluate the different techniques in the context of AoIP. To achieve a fair comparison, the packet dropper was seeded with the same seed for every technique. Thus, for every iteration, the seed changes, but the same sequence of seeds is applied to all the different methods. The categories of techniques tested, Sender-based, Receiver-based, and Sender-and-receiver-based, followed the same process in the experiments, except where and how the concealment or reconstruction of the losses was made. The experiments were conducted using the implemented audio pipeline. Further, listening tests for the receiver-based techniques were conducted. Since receiver-based techniques conceal missing audio segments instead of perfectly reconstructing them like sender-based techniques, evaluating the perceived quality of these segments is of interest.

5.4.1 Receiver-Based experiments

The receiver-based techniques experiments evaluate the objective measurements, execution time ratio, segmented cross-correlation, and SNR. They also aim to evaluate the perceived audio quality of the transient network masking PLC techniques.

Objective Measurements of the Signal Waveform

Two different tracks (both mono-channeled) were selected for evaluation, one with music and song and one strictly with speech. 10-second excerpts were made from

the tracks and stored as wave files on the IP speaker. Following the process of the simulated pipeline, 20 files of each PLC technique at all the dropping percentages were generated. All the techniques were given the same sequence of seeds to the RNG. The wave files were then exported to a computer to prepare for the measurements CC_{SEG} and SNR to be calculated as explained in section 5.3.1.

Execution Time Ratio Measurements

The implemented pipeline is a simulation run as a program on the device instead of a service performing PLC in real-time while audio is being processed in the pipeline. Thus, measurements of execution times for the pipeline's PLC application process were taken instead. The measurements consisted of the times it took a specific process to process 10-second music excerpts at different drop percentages. Each technique was measured hundreds of times at every level of drop percentage. The sequence of seeds supplied to the RNG was the same for all the methods to achieve a fair comparison. The average execution time at every drop percentage level, divided by the average time it took the SI technique to perform the same task, will roughly indicate how the different techniques performed execution time-wise by giving a ratio of:

$$\mathbf{Ratio}_i = \frac{\overline{X}_i}{\overline{X}_{\mathrm{SI}}}$$

where $\overline{X}_i$ is the average execution time of the currently investigated PLC technique at a specific drop percentage, and $\overline{X}_{\mathrm{SI}}$ is the average execution time of SI at the same drop percentage.

The experiment thus gives a ratio over performance of execution times by having the NXP i.MX 8M Nano executed Rust code that measures each PLC process (as part of the implemented pipeline).

Listening Test

As mentioned in the section 5.3.2, MUSHRA tests were proposed for evaluating the perceived audio quality. The MUSHRA test measures the Basic Audio Quality (BAQ), corresponding to the participant's overall listening experience [61]. The tests were conducted by having all the participants perform the test at the same place but at different times. All the participants used the Beyerdynamic DT 770 PRO 250 OHM headphones [64] for pro audio during the tests. The participants consisted of a combination of 7 people who are either musically-trained or audio engineers with experience and knowledge in audio processing. The original track that had not sustained any packet loss or other tampering is known as the "hidden reference," the excerpt with silence insertion PLC applied is known as the "anchor". During the test, the participants are met with an interface that lets them toggle between the various stimuli (the resulting audio segments yielded by applying the PLC techniques to the signal once it has passed the packet dropper) and the reference signal at all times. A more in-depth flow and the interface of the webMUSHRA test can be seen in the appendix section A.1.

The participants were asked to assess the conditions by giving a score to 10-second excerpts belonging to five different tracks. These excerpts had been subject to a packet loss at every 10th packet. The scores ranged from 0 (bad) to 100 (excellent). The participants were also tasked with identifying the hidden reference among the conditions and giving the hidden reference a score of 100. This is a crucial part, and if a participant scores the hidden reference below 90 for 15% or more of the tracks, that participant's result must be excluded from the final results according to the ITU-R Rec. BS.1534-3 post-screening guidelines [65].

Tracks with track id 1-4 consisted of segments with varying genres (piano excerpts, rock, pop, and guitar excerpts). The fifth track was strictly a speech excerpt.

5.4.2 Sender and Sender-Receiver-based experiments

The sender-based techniques implemented and evaluated in this thesis are closely related to sender-receiver-based techniques since sender-based techniques are essentially a form of the latter. Thus, these techniques are evaluated together since the only difference is the appliance of PLC techniques at irrecoverable signal segments. SI fills the gap with silence for the sender-based techniques. For sender-receiver-based techniques, other forms of PLC techniques are applied to the segments where the FEC cannot reconstruct the signal.

Unlike PLC techniques, FEC techniques, when possible, perfectly reconstruct the missing audio segment as discussed in section 3.2.1. This means that listening tests are not necessary to the same extent as receiver-based techniques.

5.4.3 Objective Measurements Of The Signal Waveform

Like the receiver-based experiments for objective measurements of the signal waveform described in section 5.4.1, this experiment also used the same two tracks and ran in the implemented simulation pipeline. The primary differences between these two experiments were that the data flowed through the data preparation and the data reconstruction processes in the pipeline. This led to the data being prepared according to the specific sender-based FEC technique and reconstructed accordingly. The rest of the experiment followed the same structure as the receiver-based techniques, resulting in measurements of the average SNR and the CC_{SEG} at specific dropping percentages of the different methods.

5.4.4 Execution Time Ratio Measurements

Obtaining the time ratio measurements for the sender-receiver-based techniques followed a very similar structure to the equivalent measurements for the receiver-based techniques, as described in section 5.4.1. The measurements need to include the PLC techniques as done with the receiver-based experiments, but now it also needs to include the data reconstruction process. The ratios between the different methods and the SI were calculated similarly.

Chapter 6
Results and Discussion

This chapter presents and discusses the results and potential flaws in the experiments. First, the receiver-based techniques are presented and discussed before proceeding with the sender and sender-receiver-based ones.

6.1 Simulation Limitations

There are limitations caused by using a simulation instead of running the experiments described in section 5.4. Specific issues regarding the different experiments are discussed along with their results in the coming sections. In general, it is important to highlight the main differences between running experiments in the created simulation and an actual real-time pipeline.

To begin with, running the experiments on the simulated pipeline is not done in real-time, as it would have been if run in an actual audio pipeline. The real-time aspect affects CPU performance. Playback performances of computationally heavy techniques in a simulated environment might thus not exactly represent a real-time performance, where too much strain might cause further delay. Further, there is a difference in how the network-specific elements in the pipeline behave. Since the simulated audio pipeline did not use a real network transmission, packet loss is simulated through an RNG. Further, the strain on bandwidth and other congestion issues might also arise and are discussed more in the coming sections.

6.2 Receiver-Based PLC Techniques

6.2.1 Evaluating Segmented Cross-correlation and SNR

The results CC_{SEG} and SNR scores were obtained by comparing the receiver-based PLC reconstructed signals with the original music and speech excerpt signals. The CC_{SEG} scores of the music signal can be seen in figure 6.1, and CC_{SEG} scores of the speech signal in figure 6.2. Similarly, the SNR scores of the music signal can be seen in figure 6.3, and the SNR scores of the speech signal in figure 6.4.

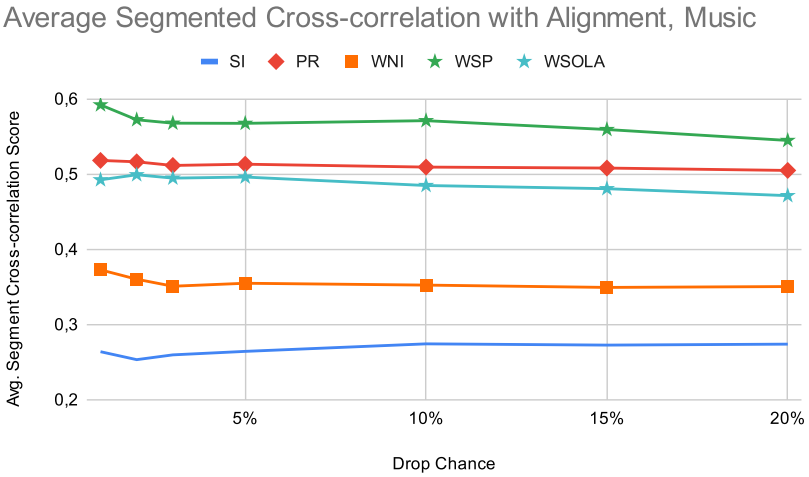

Figure 6.1: Average segmented cross-correlation scores (CC_{SEG}) at different drop chance percentages between the original music excerpt and the reconstructed ones, calculated at the reconstructed regions.

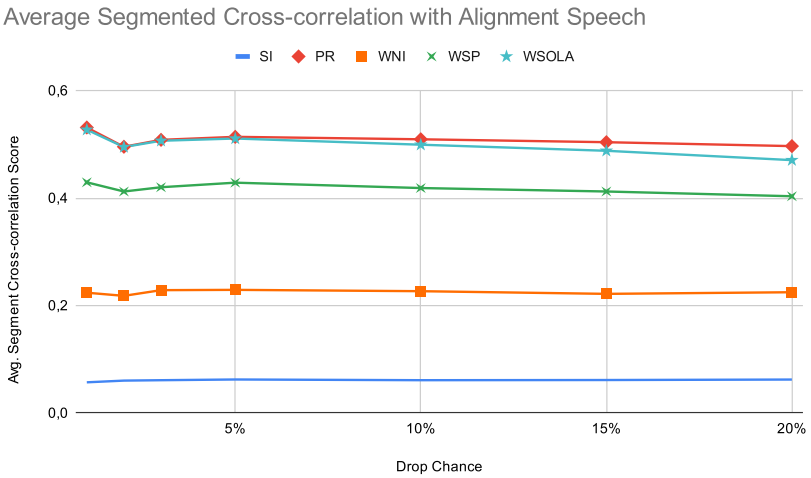

Figure 6.2: Average segmented cross-correlation scores (CC_{SEG}) at different drop chance percentages between the original speech excerpt and the reconstructed ones, calculated at the reconstructed regions.

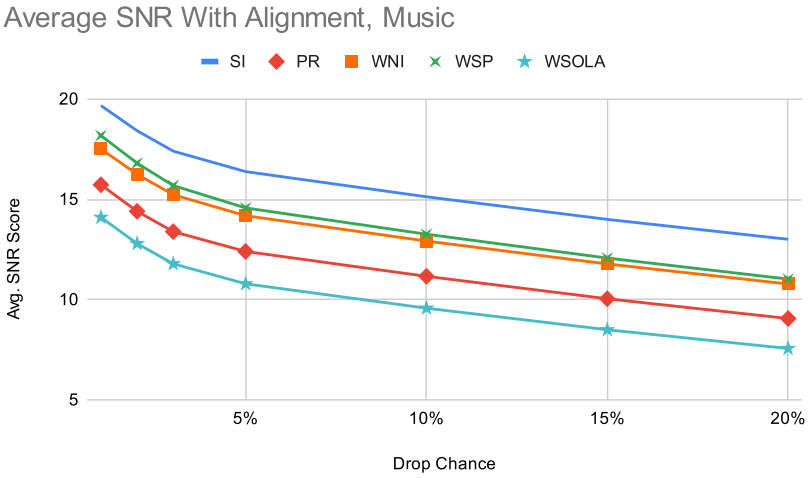

Figure 6.3: Average SNR at different drop chance percentages between the original signal and the reconstructed ones.

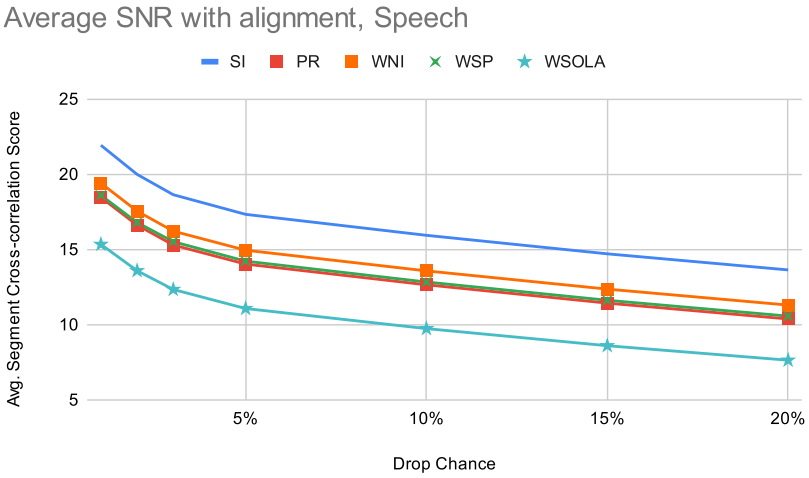

Figure 6.4: Average SNR at different drop chance percentages between the original signal and the reconstructed ones.

Looking at the CC_{SEG} scores for the music and speech excerpt in figures 6.1 and 6.2, the result more or less indicates a more significant waveform similarity for all the implemented PLC techniques than compared to SI. PR and WSOLA performed at about the same level for both the speech and the music excerpt, with a slim margin in favor of PR as the drop percentage increased. In the case of SI, WNI, and WSP, there is a difference ranging from 0.1 to 0.2 in the similarity

score. The fluctuation in similarity score seems to indicate a dependence between the performance of the CC_{SEG} score and the signal's waveform.

Conversely, the SNR in figures 6.3 and 6.4 indicates that more noise is introduced when applying the PLC techniques than SI. This does not quite align with the results of Goodman et al. in their paper on the WSP[29]. It does, however, seem plausible that the reason for this lies in the difference in waveform alignment techniques, the limited choice of waveform signals, and the difference in audio configurations. Like the CC_{SEG}, the SNR varies quite a bit between the two excerpts. Looking at the SNR formula in equation 5.1, the results are feasible because the differences between samples' power might vary more than with SI. Given the results, WSOLA and SI introduce the most and least noise from the two excerpts. WSP and WNI had similar results and a slightly lower SNR for PR.

6.2.2 Issues With The Experiment

As illustrated by the similarity scores and SNR of the PLC techniques, there was a significant discrepancy between the WNI, SI, and WSP scores applied to the speech and music excerpts. The fact that the difference between two different tracks varied this much indicates that more excerpts from different genres and speech excerpts would have to be used in the evaluations to gain a less biased result.

6.2.3 Listening Tests

The listening tests described in section 5.4.1 had to have two participants' results excluded in post-screening because both gave one of the hidden references a score below 90. The results thus consist of the answers of 5 male participants aged between 24 and 33. The results of the listening tests are shown in figure 6.5. The figure showcases a box plot of the results, displaying the means, deviations, and outliers. The figure also showcases the tracks' five Mean Opinion Scores (MOS) or PLCMOS. The PLCMOS was, on average, at roughly 57.08, 52.32, and 45.68 for the PR, WSP, and WSOLA, respectively. This categorized them all as fair overall, but both PR and WSP did manage to achieve PLCMOS at levels of "good" for some tracks. What is certainly interesting is also that all the PLC techniques performed better than SI across all the test cases except for WNI applied to the speech segment. Participants experienced an improved perceived audio quality compared to SI when the other PLC techniques were applied to lost audio segments.

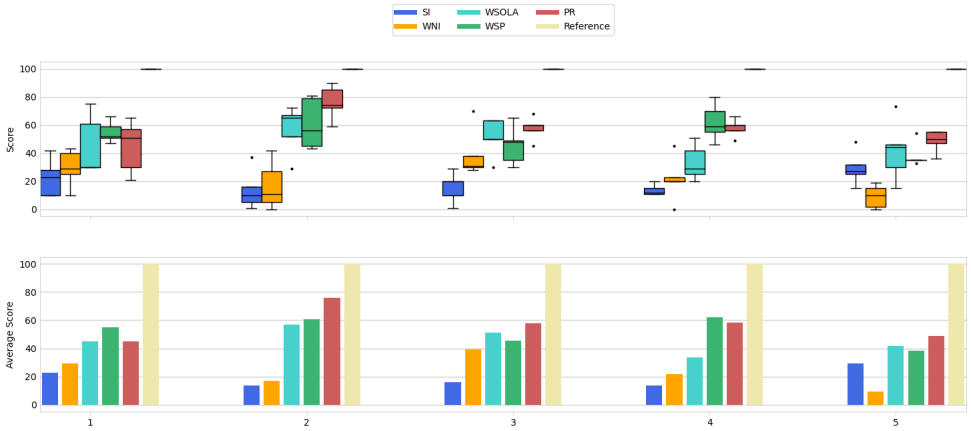

Figure 6.5: Results of the listening tests, displaying MOS and box diagrams of the PLC techniques for five tracks.

Issues With The Experiment

The number of participants in the listening experiment was on the lower end of the spectrum. Despite the participants' expertise in the subject, the fact that there are quite a few outliers in the results displays how much the perceived audio quality can vary. The results would have been more robust with a larger pool of participants. Another part of the listening test that could be improved is the number of stimuli. Due to time and resource limitations, there were only 6 stimuli to assess during the test. The ITU-R Rec. BS.1534-3 does advocate the use of 9 or more stimuli.

6.2.4 Execution Time Performance

An essential detail to disclose when discussing the execution time ratios of the time ratios is that the algorithms are not perfectly optimized or necessarily implemented in the most efficient possible way. They were implemented in Rust without any specific packet or DSP libraries, which differs significantly from Python with optimized libraries such as numpy, which the WSOLA open-source implementation used. These results serve more as a rough guideline when comparing the required time. The ratios obtained by comparing the execution time of SI to other PLC techniques can be seen in figure 6.6. The results do more or less follow the complexity trend discussed by Perkins et al. [17]. The one difference is that WNI is a bit slower due to the implementation of frequency identification and filtering in this thesis WNI.

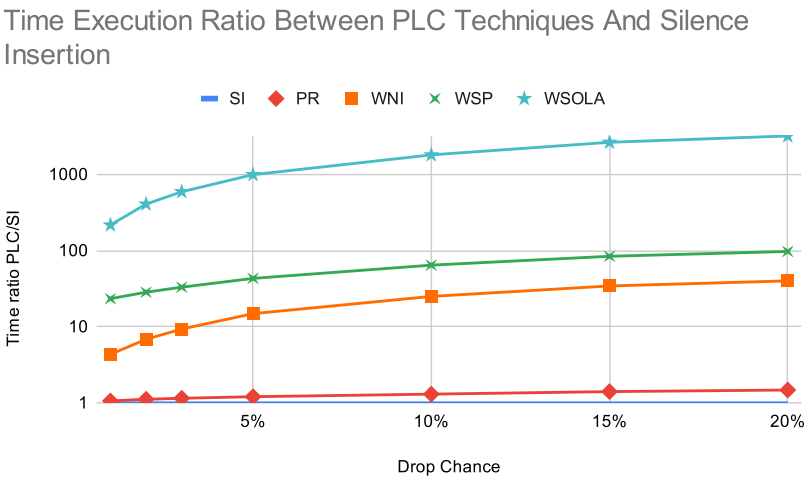

Figure 6.6: Execution time ratio between SI and other PLC techniques.

Issues With The Experiment

It is important to note that the algorithms are not optimized to the limit they could potentially be. They were implemented in Rust, which differs significantly from Python with optimized libraries such as numpy, which the WSOLA open-source implementation used. It is even possible that some of these algorithms could be optimized even further using multithreading. The specific results should thus only be taken as rough indications of computational cost and complexity.

6.3 Sender-Based and Sender-Receiver-Based Techniques

The results of the sender and sender-receiver-based techniques are presented and discussed together similarly to how they were shown in the method section.

6.3.1 Evaluating Segmented Cross-correlation and SNR

The results of the sender and sender-receiver-based techniques were acquired in the same fashion as for the receiver-based techniques. The CC_{SEG} and SNR scores were obtained by comparing the reconstructed signals with the original music and speech excerpt signals. The CC_{SEG} scores with the music excerpt can be seen in Figure 6.7 and the speech score in Figure 6.8. The corresponding SNR scores with the music excerpt in Figure 6.3 and the scores with the speech excerpt in Figure 6.4.

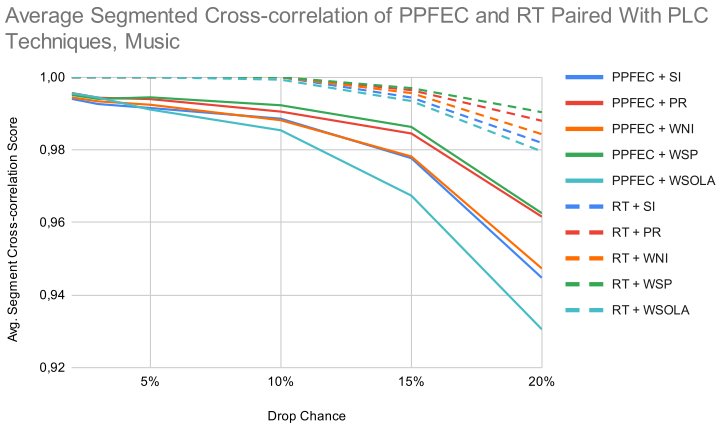

Figure 6.7: Average segmented cross-correlation scores (CC_{SEG}) at different drop chance percentages between the original music excerpt and the reconstructed ones, calculated at the reconstructed regions. The dashed lines represent RT-PLC, and the filled lines PPFEC-PLC.

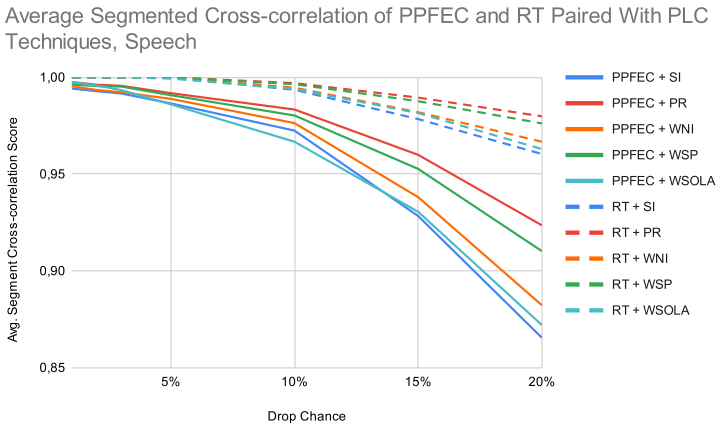

Figure 6.8: Average segmented cross-correlation scores (CC_{SEG}) at different drop chance percentages between the original speech excerpt and the reconstructed ones, calculated at the reconstructed regions. The dashed lines represent RT-PLC, and the filled lines PPFEC-PLC.

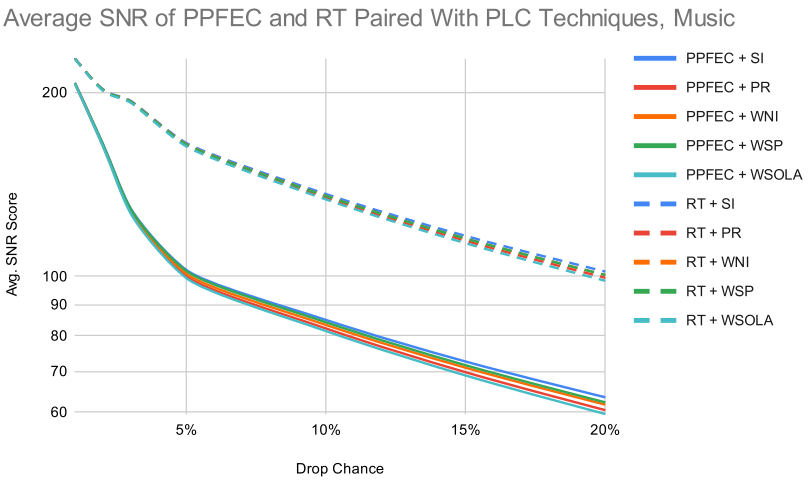

Figure 6.9: Average SNR at different drop chance percentages between the original music signal and the reconstructed ones.

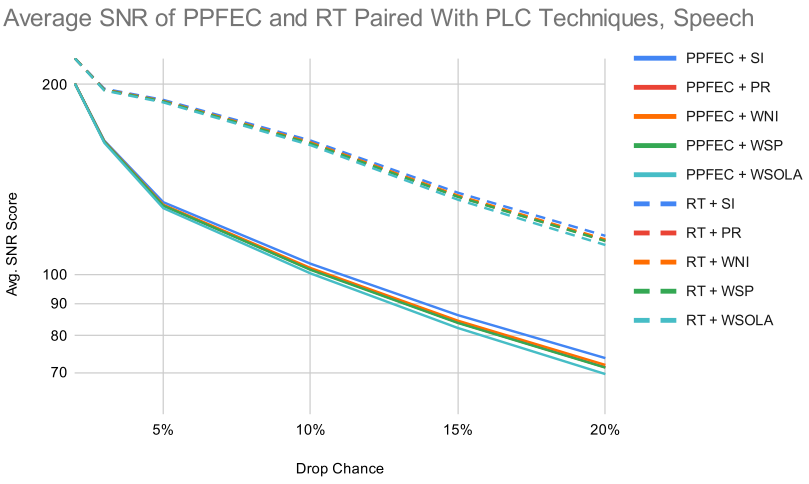

Figure 6.10: Average SNR at different drop chance percentages between the original speech signal and the reconstructed ones.

In contrast to the results for the receiver-based techniques, the sender and sender-receiver-based methods reached very high values of SNR and CC_{SEG} for both of the excerpts. The drastic increase in score lies in the fact that FEC techniques, when able, perfectly reconstruct missing segments. However, since the implemented audio simulation pipeline did not account for network congestion, in other words, it did not simulate a further straining on the bandwidth by having

more data traverse the network, these results do not truly reflect actual network conditions. The results can still roughly indicate how well the techniques would perform at the occasional random packet drop or short burst drops.

In the results, it is clear that RT had overall better SNR and CC_{SEG} scores than PPFEC, regardless of the PLC technique used. However, the further strain RT would cause on the bandwidth is not modeled, and thus, it is not clear how much better RT would perform in real network conditions. The 10-second excerpts required 631 packets when transmitted without FEC, 757 packets with PPFEC, and 1262 with RT. In longer sessions, it becomes evident how many more packets RT will flood the network with than PPFEC.

All the techniques had better CC_{SEG} than SI, meaning that the waveform similarity between the original except and the reconstructed ones were higher when combining sender and receiver-based techniques. However, as with the receiver-based results, the highest SNR score was displayed for the two methods using SI.

For packet loss percentages lower than 5%, there was not much separating the different techniques in terms of SNR and CC_{SEG} score, with WSOLA, WSP, and PR displaying the best results. As ITU has advocated for packet loss percentages below 1% and the impracticability of working with networks experiencing higher loss percentages, the performance on lower loss rates provides the most value [66].

Issues With The Experiment

As with the receiver-based techniques, these results were collected from the same excerpts, meaning there were not many variations in waveform during this experiment either. Further, as discussed, the simulated pipeline did not consider network congestion or any further straining. It allows for comparisons between the different techniques, but exactly how precise the algorithms are requires experiments to be run in real network conditions.

6.3.2 Execution Time Performance

The execution ratio between SI and the sender-based as well as SI and the sender-receiver-based techniques can be seen in figures 6.11 and 6.12. The ratios in execution time followed more or less the same trends as those of receiver-based techniques. The most significant difference is that the PLC techniques are not required to run as often as they would for the strictly receiver-based techniques. This leads to smaller ratios for the more complex and computationally heavy algorithms than when no FEC was utilized. RT managed to reconstruct more packets on average than PPFEC, leading to lower execution ratios for the WSOLA, WSP, and WNI when paired with RT instead of PPFEC. However, looking at the less complex algorithm, PR, the ratio has increased compared to the strictly receiver-based ratios. This is due to the complexity of the introduced FEC technique. Just like with the PLC techniques, the FEC implementations are necessarily not optimized to the extent they could be. These results should thus also act as a rough indication of computational cost and complexity.

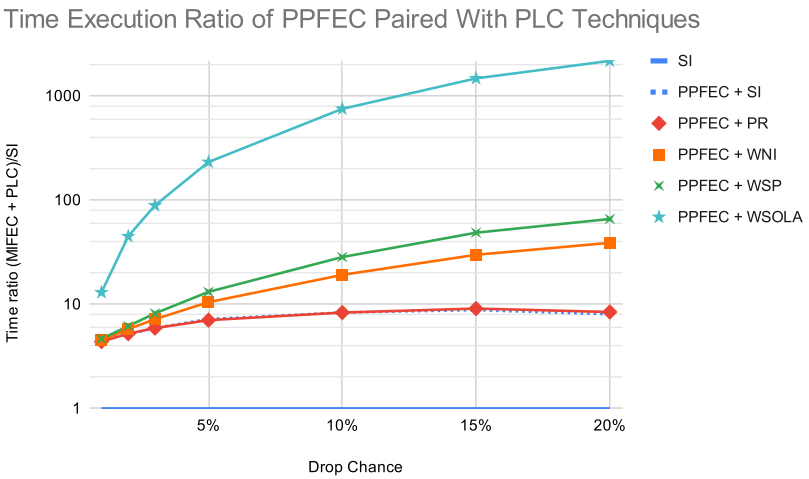

Figure 6.11: Execution time ratio between SI and PPFEC-PLC techniques.

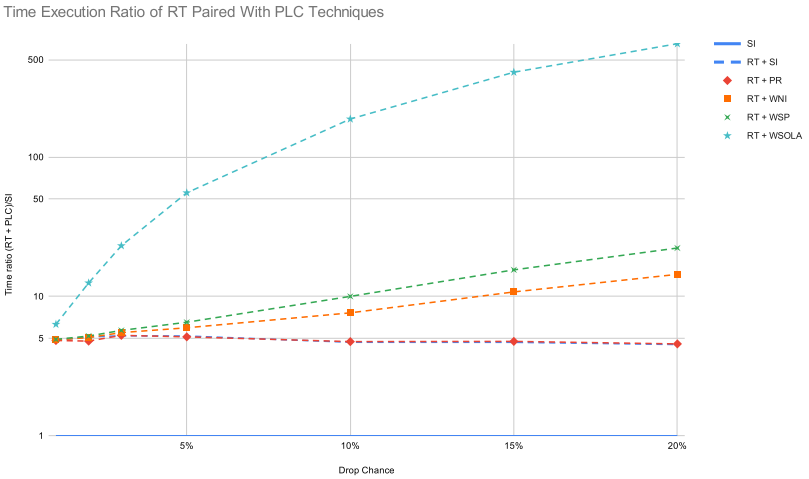

Figure 6.12: Execution time ratio between SI and RT-PLC techniques.

Issues With The Experiment

Similar to the issues with the receiver-based execution ratios experiments, the FEC and PLC techniques used in this experiment might not be optimized to their full extent.

Chapter 7

Summary and Conclusion

In this thesis, various masking techniques for transient network issues in AoIP environments were implemented and evaluated on an IP speaker. Due to integration constraints, a simplified simulation of the audio pipeline was used for evaluation. Three categories of techniques were assessed: receiver-based (SI, PR, WNI, WSP, and WSOLA), sender-based (RT, PPFEC), and combinations of both receiver and sender-based.

Evaluations considered perceived audio quality, waveform similarity, SNR, introduced latency, and execution time complexity. Results indicated a need for more waveform excerpts to achieve a more accurate assessment, as objective measurements like SNR and CC_{SEG} were variable and biased. Execution time performance was assessed via simulation.

Sender-based solutions outperformed receiver-based ones in segment repair but increased bandwidth strain and end-to-end latency. Given that the simulation did not account for bandwidth strain and router congestion, the conclusion is drawn from the perspective of occasional packet loss. RT performed slightly better than PPFEC with a block size of 6 for packet loss probabilities below 5% and notably better for higher packet loss. Given the criticality of latency in AoIP systems, PPFEC is recommended when added latency is acceptable.

Among receiver-based algorithms, PR, WSP, and WSOLA excelled in listening tests, with average MOS scores of 57.8, 52.32, and 45.68, respectively, for every tenth packet dropped. These scores suggest a fair to good listening experience, compared to poor results with SI. PR, WSP, and WSOLA also had the highest CC_{SEG} scores. PR's faster execution time and lower latency make it the recommended technique for the IP speaker.

Combining FEC with PLC techniques provided the best concealment performance, achieving near-perfect reconstruction for packet loss percentages below 5%. Although FEC increases latency, it performs perfect reconstruction for occasional packet loss. For low-latency, high-quality masking, PPFEC combined with PR is recommended, offering minimal end-to-end latency. For ultra-low latency AoIP systems, PR alone is advised.

7.1 Future Work

While several methods showed improvements in BAQ and scored higher in similarity scores than SI, further improvements are still desired to achieve good to excellent listening experiences during packet loss. The forefront of the field is incorporating ML in the PLC techniques. With proper data sets, deep learning, and autoregression, the reconstructed segments are yielding better and better listening experiences [42][39][40][41]. Finding models that are small and fast enough to run with ultra-low latency and limited resources is definitely interesting. Future work should thus focus on testing ML-based PLC directly on AoIP devices. Further work should also try to implement the techniques and models directly into the preexisting audio pipelines through plugins.

References

[1] Axis Communications. Network audio, 2024. URL https://www.axis.com/products/network-audio.

[2] Audinate. Meet dante | audinate, 2024. URL https://www.audinate.com/meet-dante/what-is-dante.

[3] Steve Church and Skip Pizzi. *Audio Over IP. [Elektronisk resurs] Building Pro AoIP Systems with Livewire.* Elsevier, 2009. ISBN 9780240812441. URL https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=cat07147a&AN=lub.6138128&site=eds-live&scope=site.

[4] T.J. Kostas, M.S. Borella, I. Sidhu, G.M. Schuster, J. Grabiec, and J. Mahler. Real-time voice over packet-switched networks. *IEEE Network*, 12(1):18–27, 1998. doi: 10.1109/65.660003.

[5] L. Ding and R.A. Goubran. Speech quality prediction in voip using the extended e-model. In *GLOBECOM '03. IEEE Global Telecommunications Conference (IEEE Cat. No.03CH37489)*, volume 7, pages 3974–3978 vol.7, 2003. doi: 10.1109/GLOCOM.2003.1258975.

[6] F.P. Zhang, O.W.W. Yang, and B. Cheng. Performance evaluation of jitter management algorithms. In *Canadian Conference on Electrical and Computer Engineering 2001. Conference Proceedings (Cat. No.01TH8555)*, volume 2, pages 1011–1016 vol.2, 2001. doi: 10.1109/CCECE.2001.933581.

[7] Yi-Chiao Wu, Israel D. Gebru, Dejan Marković, and Alexander Richard. Audiodec: An open-source streaming high-fidelity neural audio codec. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, June 2023. doi: 10.1109/ICASSP49357.2023.10096509.

[8] Ieee standard methods and equipment for measuring the transmission characteristics of pulse-code modulation (pcm) telecommunications circuits and systems. *IEEE Std 1007-1991*, pages 1–146, Jan 1992. doi: 10.1109/IEEESTD.1992.106964.

[9] Henning Schulzrinne, Stephen L. Casner, Ron Frederick, and Van Jacobson. Rtp: A transport protocol for real-time applications. RFC 3550, jul 2003. URL https://www.rfc-editor.org/info/rfc3550.

[10] DEERING Stephen. Multicast routing in a datagram internetwork. *Ph. D. Thesis, Stanford University*, 1991.

[11] H. Fastl and Eberhard Zwicker. *Psychoacoustics. facts and models.* Springer series in information sciences: 22. Springer, 2007. ISBN 9783540231592. URL https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=cat02271a&AN=atoz.ebs373112e&site=eds-live&scope=site.

[12] T. Painter and A. Spanias. Perceptual coding of digital audio. *Proceedings of the IEEE*, 88(4):451–515, April 2000. ISSN 1558-2256. doi: 10.1109/5.842996.

[13] Harvey Fletcher. Auditory patterns. *Rev. Mod. Phys.*, 12:47–65, Jan 1940. doi: 10.1103/RevModPhys.12.47. URL https://link.aps.org/doi/10.1103/RevModPhys.12.47.

[14] Eberhard Zwicker and U Tilmann Zwicker. Audio engineering and psychoacoustics: Matching signals to the final receiver, the human auditory system. *Journal of the Audio Engineering Society*, 39(3):115–126, 1991.

[15] George A. Miller and J. C. R. Licklider. The Intelligibility of Interrupted Speech. *The Journal of the Acoustical Society of America*, 22(2):167–173, 03 1950. ISSN 0001-4966. doi: 10.1121/1.1906584. URL https://doi.org/10.1121/1.1906584.

[16] Alan V. Oppenheim, John R. Buck, and Ronald W. Schafer. *Discrete-time signal processing.* Prentice-Hall signal processing series. Prentice-HallInternational, 1999. ISBN 0130834432. URL https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=cat07147a&AN=lub.1535186&site=eds-live&scope=site.

[17] C. Perkins, O. Hodson, and V. Hardman. A survey of packet loss recovery techniques for streaming audio. *IEEE Network*, 12(5):40–48, Sep. 1998. ISSN 1558-156X. doi: 10.1109/65.730750.

[18] B.W. Wah, Xiao Su, and Dong Lin. A survey of error-concealment schemes for real-time audio and video transmissions over the internet. In *Proceedings International Symposium on Multimedia Software Engineering*, pages 17–24, Dec 2000. doi: 10.1109/MMSE.2000.897185.

[19] Jérémie Lecomte, Tommy Vaillancourt, Stefan Bruhn, Hosang Sung, Ke Peng, Kei Kikuiri, Bin Wang, Shaminda Subasingha, and Julien Faure. Packet-loss concealment technology advances in evs. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5708–5712, April 2015. doi: 10.1109/ICASSP.2015.7179065.

[20] J. Gruber and L. Strawczynski. Subjective effects of variable delay and speech clipping in dynamically managed voice systems. *IEEE Transactions on Communications*, 33(8):801–808, August 1985. ISSN 1558-0857. doi: 10.1109/TCOM.1985.1096385.

[21] Vicky Hardman, Martina Angela Sasse, Mark Handley, and Anna Watson. Reliable audio for use over the internet. In *Proceedings of INET*, volume 95, pages 171–178. The Internet Society, 1995.

[22] J. Suzuki and M. Taka. Missing packet recovery techniques for low-bit-rate coded speech. *IEEE Journal on Selected Areas in Communications*, 7(5): 707–717, June 1989. ISSN 1558-0008. doi: 10.1109/49.32334.

[23] N. Jayant and S. Christensen. Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation procedure. *IEEE Transactions on Communications*, 29(2):101–109, February 1981. ISSN 1558-0857. doi: 10.1109/TCOM.1981.1094975.

[24] RCF Tucker and JE Flood. Optimizing the performance of packet-switch speech. In *IEEE Conf. on Digital Processing of Signals in Communications*, number 62, pages 227–234, 1985.

[25] 3GPP TS 06.11. Digital cellular telecommunications system (phase 2+) (gsm); full rate speech; substitution and muting of lost frames for full rate speech channels, 2017. URL `https://cdn.standards.iteh.ai/samples/51980/cf56fdf65f754690a1be7c07ec14fb86/ETSI-TS-146-011-V14-0-0-2017-04-.pdf`.

[26] George A. Miller and J. C. R. Licklider. The Intelligibility of Interrupted Speech. *The Journal of the Acoustical Society of America*, 22(2):167–173, 03 1950. ISSN 0001-4966. doi: 10.1121/1.1906584. URL `https://doi.org/10.1121/1.1906584`.

[27] Richard M Warren and Gary L Sherman. Phonemic restorations based on subsequent context. *Perception & Psychophysics*, 16:150–156, 1974. ISSN 1532-5962. doi: 10.3758/BF03203268.

[28] George A Miller and Joseph CR Licklider. The intelligibility of interrupted speech. *The Journal of the Acoustical Society of America*, 22(2):167–173, 1950.

[29] D. Goodman, G. Lockhart, O. Wasem, and Wai-Choong Wong. Waveform substitution techniques for recovering missing speech segments in packet voice communications. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(6):1440–1448, 1986. doi: 10.1109/TASSP.1986.1164984.

[30] O.J. Wasem, D.J. Goodman, C.A. Dvorak, and H.G. Page. The effect of waveform substitution on the quality of pcm packet communications. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(3):342–348, March 1988. ISSN 0096-3518. doi: 10.1109/29.1530.

[31] H. Sanneck, A. Stenger, K. Ben Younes, and B. Girod. A new technique for audio packet loss concealment. In *Proceedings of GLOBECOM'96. 1996 IEEE Global Telecommunications Conference*, volume MiniConfInternet, pages 48–52, Nov 1996. doi: 10.1109/GLOCOM.1996.586117.

[32] D. W. Griffin and J. S. Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Audio, Speech, and Signal Processing*, 32(2):236–243, April 1984. doi: 10.1109/TASSP.1984.1164317.

[33] J. Laroche and M. Dolson. Improved phase vocoder time-scale modification of audio. *IEEE Transactions on Speech and Audio Processing*, 7(3):323–332, May 1999. ISSN 1558-2353. doi: 10.1109/89.759041.

[34] Eric Moulines and Jean Laroche. Non-parametric techniques for pitch-scale and time-scale modification of speech. *Speech Communication*, 16(2):175–205, 1995. ISSN 0167-6393. doi: https://doi.org/10.1016/0167-6393(94)00054-E. URL `https://www.sciencedirect.com/science/article/pii/016763939400054E`. Voice Conversion: State of the Art and Perspectives.

[35] M. Portnoff. Time-scale modification of speech based on short-time fourier analysis. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29 (3):374–390, June 1981. ISSN 0096-3518. doi: 10.1109/TASSP.1981.1163581.

[36] KAIST-MACLab. Pytsmod: Python tools for time series modulation analysis, 2023. URL `https://github.com/KAIST-MACLab/PyTSMod?tab=readme-ov-file`. Accessed on March 20, 2024.

[37] Jonathan Driedger and Meinard Müller. A review of time-scale modification of music signals. *Applied Sciences*, 6(2), 2016. ISSN 2076-3417. doi: 10.3390/app6020057. URL `https://www.mdpi.com/2076-3417/6/2/57`.

[38] W. Verhelst and M. Roelands. An overlap-add technique based on waveform similarity (wsola) for high quality time-scale modification of speech. In *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 554–557 vol.2, April 1993. doi: 10.1109/ICASSP.1993.319366.

[39] Steven Davy, Niamh Belton, Joshua Tobin, Owais Bin Zuber, Liu Dong, and Yuan Xuewen. A causal convolutional approach for packet loss concealment in low powered devices. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Acoustics, Speech and Signal Processing (ICASSP), ICASSP 2023 - 2023 IEEE International Conference on*, pages 1 – 5, 2023. ISSN 978-1-7281-6327-7. URL `https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edseee&AN=edseee.10096505&site=eds-live&scope=site`.

[40] Prateek Verma, Alessandro Ilic Mezza, Chris Chafe, and Cristina Rottondi. A deep learning approach for low-latency packet loss concealment of audio signals in networked music performance applications. In *2020 27th Conference*

of Open Innovations Association (FRUCT), pages 268–275, Sep. 2020. doi: 10.23919/FRUCT49677.2020.9210988.

[41] Mezza Alessandro Ilic, Amerena Matteo, Bernardini Alberto, and Sarti Augusto. Hybrid packet loss concealment for real-time networked music applications. *IEEE Open Journal of Signal Processing*, 5:266 – 273, 2024. ISSN 2644-1322. URL https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edsdoj&AN=edsdoj.4a0867f99b4748b7bb62892cde7d67ac&site=eds-live&scope=site.

[42] Guoqiang Zhang and W. Bastiaan Kleijn. Autoregressive model-based speech packet-loss concealment. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4797–4800, March 2008. doi: 10.1109/ICASSP.2008.4518730.

[43] Microsoft Research Academic Program. Audio deep packet loss concealment challenge - ICASSP 2024. https://www.microsoft.com/en-us/research/academic-program/audio-deep-packet-loss-concealment-challenge-icassp-2024/, 2024. Accessed: 2024-05-10.

[44] N. Erdol, C. Castelluccia, and A. Zilouchian. Recovery of missing speech packets using the short-time energy and zero-crossing measurements. *IEEE Transactions on Speech and Audio Processing*, 1(3):295–303, 1993. doi: 10.1109/89.232613.

[45] N. Shacham. Packet recovery and error correction in high-speed wide-area networks. In *IEEE Military Communications Conference, 'Bridging the Gap. Interoperability, Survivability, Security'*, pages 551–557 vol.2, Oct 1989. doi: 10.1109/MILCOM.1989.103987.

[46] I. S. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960. doi: 10.1137/0108018. URL https://doi.org/10.1137/0108018.

[47] Martyn Riley and Iain Richardson. Reed-Solomon Codes. 2024. URL https://www.cs.cmu.edu/~guyb/realworld/reedsolomon/reed_solomon_codes.html.

[48] Isidor Kouvelas, Orion Hodson, Vicky Hardman, and Jon Crowcroft. Redundancy control in real-time internet audio conferencing. In *Proceedings of AVSPN*, volume 97. Citeseer, 1997.

[49] Erik Hellerud and U. Peter Svensson. Robust transmission of lossless audio with low delay over ip networks. In *2007 IEEE International Symposium on Signal Processing and Information Technology*, pages 590–594, Dec 2007. doi: 10.1109/ISSPIT.2007.4458170.

[50] Jin Ah Kang and Hong Kook Kim. A cross-layer plc algorithm for a real-time audio conferencing system. In *2008 10th International Conference on Advanced Communication Technology*, volume 2, pages 1213–1217, Feb 2008. doi: 10.1109/ICACT.2008.4493983.

[51] You-Li Chen and Bor-Sen Chen. Model-based multirate representation of speech signals and its application to recovery of missing speech packets. *IEEE Transactions on Speech and Audio Processing*, 5(3):220–231, May 1997. ISSN 1558-2353. doi: 10.1109/89.568729.

[52] Xiao Su and B.W. Wah. Streaming video with optimized reconstruction-based dct. In *2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)*, volume 1, pages 271–274 vol.1, July 2000. doi: 10.1109/ICME.2000.869594.

[53] Wang Lizhong, Wu Muqing, and Li Mojia. Waveform similarity over-and-add technique with gain contgrol. In *2009 2nd IEEE International Conference on Broadband Network Multimedia Technology*, pages 735–739, Oct 2009. doi: 10.1109/ICBNMT.2009.5347779.

[54] J.F. Yeh, P.C. Lin, M.D. Kuo, and Z.H. Hsu. Bilateral waveform similarity overlap-and-add based packet loss concealment for voice over ip. *Journal of Applied Research and Technology*, 11(4):559 – 567, 2013. doi: 10.1016/S1665-6423(13)71563-3. URL `https://www.scopus.com/inward/record.uri?eid=2-s2.0-84886679803&doi=10.1016%2fS1665-6423%2813%2971563-3&partnerID=40&md5=2e8ca624691f6753ee03251be41a1778`. Cited by: 3; All Open Access, Bronze Open Access, Green Open Access.

[55] John G. Proakis and Dimitris G. Manolakis. *Digital Signal Processing: Principles, Algorithms, and Applications*. Prentice-Hall International, Inc., 3rd edition, 2006.

[56] Douglas Self. *Audio Engineering Explained*. Elsevier Science, 2009. ISBN 9780240812731. URL `https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=cat07147a&AN=lub.6126086&site=eds-live&scope=site`.

[57] NXP Semiconductors. i.mx 8m nano family - arm cortex-a53, cortex-m7, 2024. URL `https://www.nxp.com/products/processors-and-microcontrollers/arm-processors/i-mx-applications-processors/i-mx-8-applications-processors/i-mx-8m-nano-family-arm-cortex-a53-cortex-m7:i.MX8MNANO`. Accessed: 2024-05-31.

[58] A. Pcjić, P. M. Stanić, and Sz. Pletl. Analysis of packet loss prediction effects on the objective quality measures of opus codec. In *2014 IEEE 12th Inter-*

national Symposium on Intelligent Systems and Informatics (SISY), pages 33–37, Sep. 2014. doi: 10.1109/SISY.2014.6923611.

[59] Naofumi Aoki. A voip packet loss concealment technique taking account of pitch variation in pitch waveform replication. *Electronics amp; Communications in Japan, Part 1: Communications*, 89(3):1 – 9, 2006. ISSN 87566621. URL https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=bth&AN=18898970&site=eds-live&scope=site.

[60] John G. Proakis and Dimitris G. Manolakis. *Digital Signal Processing: Principles, Algorithms, and Applications*. Prentice Hall, 3rd edition, 1996. URL https://uvceee.wordpress.com/wp-content/uploads/2016/09/digital_signal_processing_principles_algorithms_and_applications_third_edition.pdf.

[61] Michael Schoeffler, Sarah Bartoschek, Fabian-Robert Stöter, Marlene Roess, Susanne Westphal, Bernd Edler, and Jürgen Herre. webmushra—a comprehensive framework for web-based listening tests. *Journal of Open Research Software*, 6(1):8, 2018.

[62] Fabian Brinkmann and Stefan Weinzierl. *Audio Quality Assessment for Virtual Reality*, pages 145–178. Springer International Publishing, Cham, 2023. ISBN 978-3-031-04021-4. doi: 10.1007/978-3-031-04021-4_5. URL https://doi.org/10.1007/978-3-031-04021-4_5.

[63] GStreamer project. GStreamer Documentation: Clocks. https://gstreamer.freedesktop.org/documentation/application-development/advanced/clocks.html?gi-language=c, Year of access. Accessed on May 8, 2024.

[64] Beyerdynamic. DT 770 PRO, 2024. URL https://global.beyerdynamic.com/dt-770-pro.html.

[65] ITUR BS. 1534-3,"method for the subjective assessment of intermediate quality level of audio systems,". *International Telecommunication Union, Geneva, Switzerland*, 2015.

[66] International Telecommunication Union. Network performance objectives for ip-based services. Technical report, Geneva, Switzerland, dec 2011. URL https://www.itu.int/rec/T-REC-Y.1541/en.

Appendix A

Appendix

A.1 WebMUSHRA Interface

The webMUSHRA flow starts with the user reading about the tool and the experiment they are about to participate in. There is also a page regarding consent to participate. As seen in figures A.1 and A.2.

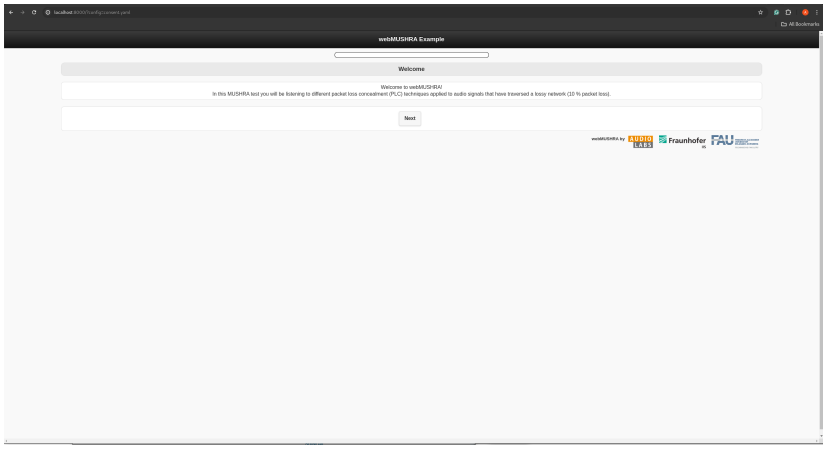

Figure A.1: The index interface of the webMUSHRA tool.

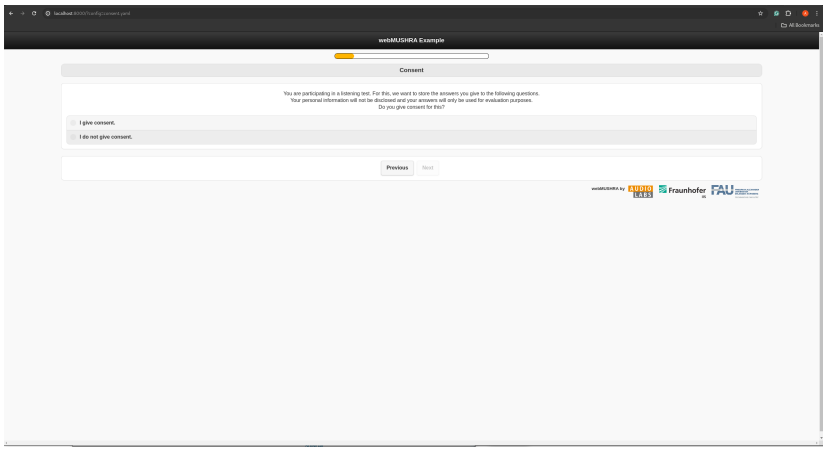

Figure A.2: Consent page for participating in the experiment.

After these two initial pages, they are met with an interface meant to give them a bit better understanding of the tool and more information about their task, this page does not yield any results for the test but instead allows the participants to practice with the tool as seen in figure A.3

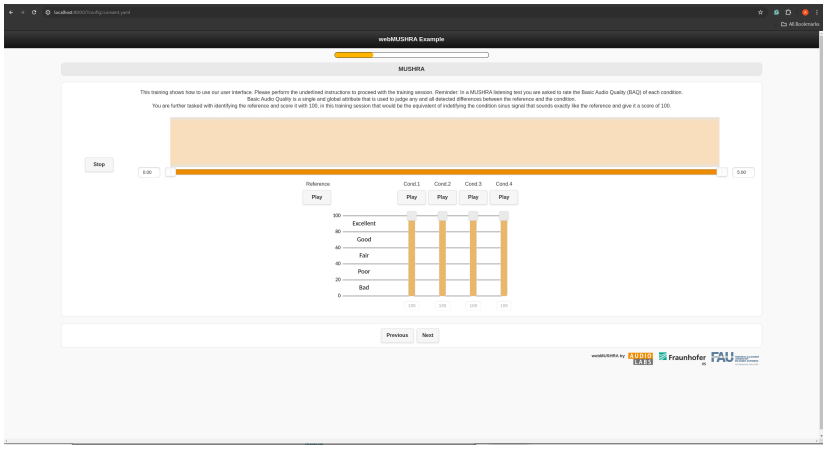

Figure A.3: Consent page for participating in the experiment.

The real test then commences. Each page displays the multiple stimuli and the reference. The participants can freely toggle between the different stimuli and score them while listening. Once satisfied, they proceed to the next track by clicking next. These pages arrive in a randomized order, and the conditions are placed in a random order as well. Figure A.4 showcases an example of a real test page.

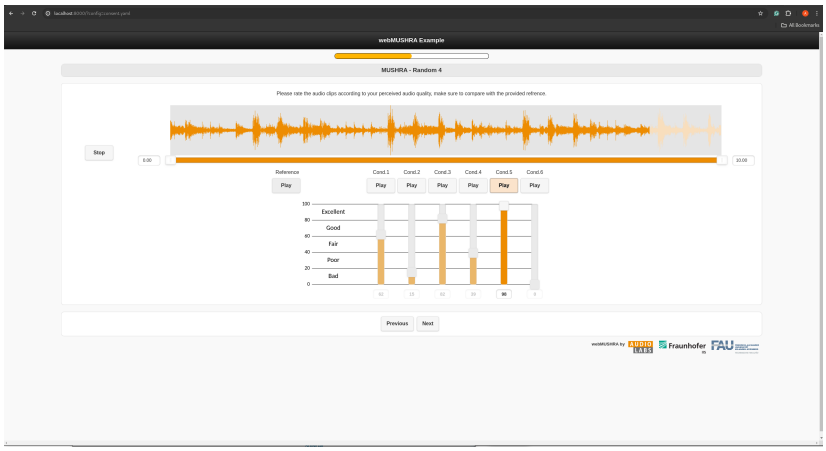

Figure A.4: MUSHRA test for a track, displaying the waveform of the signal, the reference play button, and all the condition play buttons along with their corresponding rating scales.

The participants proceed through all the actual test pages until all tracks have been used to evaluate the stimuli. Once finished, the test results are submitted and stored as CSV. The submission page can be seen in figure A.5.

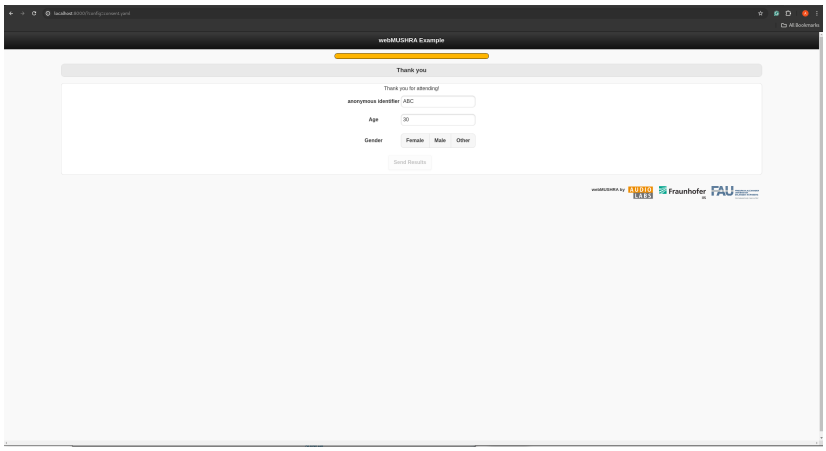

Figure A.5: MUSHRA result submission page.