Fractional-Order Noise Estimation for Speech Enhancement in Acoustic Environments with 1/f Noise

- A Grünwald-Letnikov Modification of MCRA -

MSc Sound and Music Computing Thesis

Stephan Alejandro Horvath Muñoz de Escalona

Aalborg University
Department of Architecture, Design and Media Technology

Department of Architecture, Design and Media Technology
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
Fractional-Order Noise Estimation for Speech Enhancement in Acoustic Environments with 1/f Noise: A Grünwald-Letnikov Modification of MCRA

Theme:
Master's Thesis

Project Period:
Sound and Music Computing Semester 10

Author:
Stephan Horvath

Supervisor:
Razvan Paisa

Page Numbers: 56

Date of Completion:
May 25, 2026

Abstract:

This thesis proposes GL-MCRA, a fractional-order modification of the minima controlled recursive averaging (MCRA) noise estimator. Standard recursive averaging imposes an exponentially decaying memory structure poorly matched to the power-law correlation decay of $1/f$ noise. GL-MCRA replaces the first-order recursive update with a Grünwald-Letnikov power-law weighted sum over past speech-presence-gated power observations, with fractional order $r \in \{0.1, \ldots, 0.9\}$ controlling the rate of decay.
Evaluated on the NOIZEUS corpus across seven noise conditions and four SNR levels, GL-MCRA outperforms standard MCRA in segmental SNR improvement across all conditions at $r = 0.9$, with statistically significant improvements of 0.23–0.48 dB (5.4–11.5%, $p < 0.001$). Perceptual evaluation of speech quality results are consistent with the objective findings. A subjective MOS listening test with 17 participants reveals a condition-dependent perceptual benefit. The primary limitation of GL-MCRA is its higher computational complexity relative to standard MCRA.

The content of this report is freely available, but publication (with reference) may only be pursued due to agreement with the author.

Contents

List of Figures **v**

List of Tables **vi**

Preface **vii**

List of Abbreviations **ix**

1 Introduction **1**
1.1 Motivation and Background 1
1.2 Research Gap 1

2 Problem Statement **3**

3 Literature Review **5**
3.1 Voice Activity Detectors 5
3.1.1 Global System for Mobile Communications VAD 5
3.1.2 Cellular Network VAD 6
3.1.3 Decision-Directed Method 7
3.1.4 Spectral Entropy VAD 7
3.1.5 Convolutional Neural Network VAD 8
3.1.6 Discussion 9
3.2 Acoustic Noise Characteristics 9
3.2.1 Stationarity 9
3.2.2 Colours of Noise 10
3.2.3 Short and Long-Range Dependence 13
3.3 Noise Estimation Methods 15
3.3.1 Minima Controlled Recursive Averaging 15
3.3.2 Improved Minima Controlled Recursive Averaging 18
3.3.3 DFT Subspace (DFT-SS) 18
3.3.4 Minimum Mean-Squared Error 20
3.4 Evaluation of Noise Estimators 22

3.4.1 Signal-to-Noise Ratio . . . 22
3.4.2 Segmental Signal-to-Noise Ratio . . . 22
3.4.3 Signal-to-Noise Ratio Levels . . . 23
3.4.4 Perceptual Evaluation of Speech Quality . . . 23
3.4.5 Mean Opinion Score . . . 24
3.5 Fractional-Order Filters . . . 24
3.5.1 Fractional Calculus . . . 24

4 Final Problem Statement **27**

5 Methodology **29**
5.1 Signal Preparation . . . 29
5.2 Noise Types . . . 29
5.2.1 White Gaussian Noise . . . 29
5.2.2 Stationary $1/f$ Noise . . . 30
5.2.3 Non-Stationary $1/f$ Noise . . . 30
5.2.4 Train and Babble Noise . . . 30
5.3 Signal Mixing . . . 30
5.4 The Proposed Estimator . . . 31
5.4.1 Speech-Presence Gating . . . 31
5.4.2 Grünwald-Letnikov Noise Estimate Update . . . 32
5.5 Experimental Design . . . 32
5.5.1 Estimator Configurations . . . 32
5.5.2 Enhancement and Evaluation Pipeline . . . 33

6 Results **34**
6.1 Aggregate SegSNR Improvement . . . 34
6.2 Per-Condition SegSNR Improvement . . . 35
6.2.1 Stationary $1/f$ Noise . . . 36
6.2.2 White Gaussian Noise . . . 37
6.2.3 Non-Stationary $1/f$ Noise . . . 37
6.2.4 Train Noise . . . 37
6.2.5 Babble Noise . . . 39
6.2.6 Summary . . . 39
6.2.7 Statistical Significance . . . 40
6.3 Perceptual Evaluation of Speech Quality . . . 41
6.3.1 Aggregate PESQ . . . 41
6.3.2 Per-Condition PESQ . . . 42
6.3.3 PESQ vs SNR Level . . . 42
6.4 Mean Opinion Score . . . 43

7 Discussion **45**
7.1 Effect of Fractional Order on Performance 45
7.2 Contribution of Gating vs Fractional Memory 46
7.3 Performance Under Stationary Noise 47
7.3.1 Stationary $1/f$ Noise 47
7.3.2 White Gaussian Noise 47
7.4 Performance Under Non-Stationary and Real-World Noise 48
7.4.1 Non-Stationary $1/f$ Noise 48
7.4.2 Train Noise 48
7.4.3 Babble Noise 48
7.5 Divergence Between Objective and Subjective Metrics 48
7.6 Limitations of the Proposed Estimator 49
7.7 Limitations of the Evaluation 49
7.8 Future Work 50

8 Conclusion **51**
8.1 Summary of Findings 51
8.2 Answer to the Research Question 52
8.3 Future Work 52

Bibliography **54**

A Source Code **a**
A.1 MATLAB Implementation . . . . . a
A.2 AI Tool Usage . . . . . h

List of Figures

3.1 White Noise's Flat Power Spectral Density 11
3.2 Pink Noise's -3 dB/Octave Slope 12
3.3 Brown Noise's -6 dB/Octave Slope 13

6.1 Segmental SNR Improvement Per Estimator, averaged across all noise conditions, as a function of fractional-order r 35
6.2 SegSNR Improvement Per GL-MCRA per each value of r, MCRA and Gated MCRA, Per Condition 36
6.3 Relative SegSNR improvement of GL-MCRA over standard MCRA (dB), per noise condition and fractional order r. Positive values indicate GL-MCRA outperforms standard MCRA. 37
6.4 Spectrograms of clean, noisy, MCRA-enhanced, and GL-MCRA-enhanced speech for utterance sp15 under fGn H=0.9 noise at 10 dB SNR. 38
6.5 Hedge's g effect size for GL-MCRA at $r = 0.9$ vs Standard MCRA, per noise condition and input SNR level ($n = 30$ pairs per cell, $p < 0.001$ for all entries). 40
6.6 Mean PESQ scores averaged across all noise conditions and SNR levels, as a function of fractional order r 41
6.7 Mean PESQ scores per noise condition at $r = 0.9$. Higher is better; scale is 1.0–4.5. 42
6.8 Mean PESQ scores by input SNR level at $r = 0.9$, averaged across all noise conditions. 43

List of Tables

6.1 Mean MOS ratings ($\pm$ Standard Deviation) at 0 dB SNR, $n = 17$ participants. GL-MCRA at $r = 0.9$. Δ denotes GL-MCRA minus MCRA; $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$ (Wilcoxon signed-rank test). 44
6.2 Mean MOS ratings ($\pm$ Standard Deviation) at 10 dB SNR, $n = 17$ participants. GL-MCRA at $r = 0.9$. Δ denotes GL-MCRA minus MCRA; $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$ (Wilcoxon signed-rank test). 44

Preface

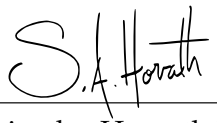
Aalborg University May 25, 2026

This thesis was written as part of the Master of Science programme in Sound and Music Computing at Aalborg University, Copenhagen, during the spring semester of 2026.

The project investigates fractional-order noise estimation for speech enhancement in acoustic environments with 1/f noise, proposing a modification to the MCRA noise estimator based on the Grünwald-Letnikov fractional derivative.

I would like to thank Razvan Paisa for guidance and feedback throughout the project, and my friends and family who supported me throughout the programme. Thank you as well to the 17 participants who volunteered their time for the listening test.

Parts of this thesis were produced with the assistance of AI language model tools, which were used for generating data automation scripts to batch-run analysis functions, and for formatting LaTeX tables. All technical content, methodology, results, analysis, and conclusions are the author's own work.



Stephan Alejandro Horvath Muñoz de Escalona
sescal24@student.aau.dk

Abbreviations

ARMA	Autoregressive Moving Average
AWGN	Additive White Gaussian Noise
BNVAD	Babble Noise Voice Activity Detector
CNN	Convolutional Neural Network
DD	Decision-Directed
DFT-SS	Discrete Fourier Transform Subspace
FC	Fractional Calculus
fBm	Fractional Brownian Motion
fGn	Fractional Gaussian Noise
FIR	Finite Impulse Response
GL	Grünwald-Letnikov
GL-MCRA	Grünwald-Letnikov Minima Controlled Recursive Averaging
GSM	Global System for Mobile Communications
HMM	Hidden Markov Model
IIR	Infinite Impulse Response
IMCRA	Improved Minima Controlled Recursive Averaging
ITU-T	International Telecommunication Union Telecommunication
LOG-Err	Symmetric Log-Error Distortion
LRD	Long-Range Dependence
MCRA	Minima Controlled Recursive Averaging
MMSE	Minimum Mean-Squared Error
MOS	Mean Opinion Score
NHR	Noise Hit Rate
OM-LSA	Optimally Modified Log-Spectral Amplitude
PESQ	Perceptual Evaluation of Speech Quality
PSD	Power Spectral Density
SegErr	Segmental Relative Estimation Error
SegSNR	Segmental Signal-to-Noise Ratio
SEVAD	Spectral Entropy Voice Activity Detector
SNR	Signal-to-Noise Ratio
SRD	Short-Range Dependent
STFT	Short-Time Fourier Transform
VAD	Voice Activity Detector
VNVAD	Vehicle Noise Voice Activity Detector
wGn	White Gaussian Noise
WSS	Wide-Sense Stationary

Chapter 1

Introduction

1.1 Motivation and Background

Speech enhancement is the process of recovering a clean speech signal from a recording corrupted by additive background noise. It is a fundamental problem in audio signal processing with applications across mobile communications, hearing aids, speech recognition, and voice-activated systems. The performance of speech enhancement systems depends critically on the accuracy of the underlying noise estimator, which must track the power spectral density of the background noise continuously and adapt to changing acoustic conditions.

The predominant class of noise estimators in the literature, including the minima controlled recursive averaging (MCRA) method of Cohen and Berdugo [1] and the improved MCRA (IMCRA) of Cohen [2], relies on first-order recursive averaging to smooth past spectral power observations. This imposes an exponentially decaying memory structure on the noise estimate where recent observations are weighted heavily and older observations decay rapidly — the estimator has "short memory". This memory structure is well suited to short-range dependent noise, where temporal correlations decay exponentially or are absent entirely, as in the case of white Gaussian noise. However, many real-world acoustic noise sources exhibit $1/f$ spectral characteristics and long-range dependence (LRD), where correlations decay as a power law and persist over time. For such types of noise, the exponential memory of standard recursive averaging is a structural mismatch.

1.2 Research Gap

This thesis investigates whether replacing the first-order recursive averaging in MCRA with a Grünwald-Letnikov (GL) fractional-order weighted sum improves noise power spectral density estimation accuracy in $1/f$ acoustic environments. The GL definition of the fractional derivative expresses the operator as a weighted sum of past samples with weights that decay as a power law — a natural match to the correlation structure of

$1/f$ noise. The proposed estimator, GL-MCRA, additionally introduces a speech-presence gating mechanism to prevent speech energy from contaminating the power-law memory buffer. The fractional order $r \in \{0.1, 0.2, \ldots, 0.9\}$ controls the rate of power-law decay and is swept systematically to characterise its effect on estimation performance.

Research Questions

To what extent does replacing the first-order recursive averaging in MCRA with a Grünwald-Letnikov fractional-order weighted sum improve noise power spectral density estimation and speech enhancement performance in $1/f$ acoustic noise environments?

The primary contributions of this thesis are:

- The derivation and implementation of GL-MCRA, a fractional-order modification of the MCRA noise estimator that replaces first-order recursive averaging with a Grünwald-Letnikov power-law weighted sum
- A speech-presence gating mechanism that prevents speech energy from entering the fractional memory buffer
- A systematic evaluation of GL-MCRA across seven noise conditions — including stationary and non-stationary $1/f$ noise at three Hurst exponents, white Gaussian noise, train noise, and babble noise — at four SNR levels, using segmental SNR improvement, PESQ, and a subjective MOS listening test with 17 participants

Thesis Structure

- Chapter 3 reviews the relevant literature, covering voice activity detection, acoustic noise characteristics, noise estimation methods, and fractional calculus.
- Chapter 5 describes the experimental setup, the proposed GL-MCRA estimator, and the evaluation procedure.
- Chapter 6 presents the objective and subjective evaluation results.
- Chapter 7 interprets the results, discusses the limitations of GL-MCRA and the evaluation, and outlines directions for future work.
- Chapter 8 summarises the findings and answers the research question.

Chapter 2

Problem Statement

In speech technology and processing, the suppression of noise is vitally important for applications such as speech detection and recognition, hearing aids, real-time voice transmission such as mobile communication [3], and other such applications where the efficiency and output of a voice activity detector (VAD) depends on the separation of speech and non-speech signals. The properties and characteristics of the noise can vary in amplitude and in spectral shape depending on the environment where the speech signal is captured. The gentle splashing of waves on a shore is remarkably different from the loud babble of a multitude of voices at a party. In order to reduce the noise in a signal, an assumption must be made about the type of noise, or its properties must be estimated [1, 4].

Traditional noise reduction methods such as Wiener filtering rely on the noise being a stationary stochastic process with predictable statistical properties [4]. Another widely used method, spectral subtraction [5], needs first to estimate the spectral shape of the noise during pauses in the speech signal, before it is able to subtract the noise from the noisy signal and recover an estimate of the clean speech signal. Both of these methods perform best with stationary noise [1, 4], though they have been extended to also handle non-stationary noise.

However, the standard methods (which operate with integer-order filters) assume exponentially decaying autocorrelation functions. This assumption breaks down for $1/f^{\alpha}$-type noise (pink noise and similar long-range dependence (LRD) processes commonly found in natural acoustic environments), which exhibits power-law autocorrelation decay. Wiener filtering and spectral subtraction cannot accurately model these long-range temporal dependencies, leading to biased noise estimates and residual noise artifacts. Fractional-order filters, by contrast, naturally incorporate power-law dynamics through their fractional derivative operators, enabling them to capture the LRD properties inherent in $1/f^{\alpha}$ noise and provide more accurate spectral estimates of the noise process.

A review of the literature reveals minimal prior research on fractional-order filters for VAD noise estimation. Based on this, the purpose of this research is to determine in what quantifiable way can fractional-order filters, which in their calculations include the LRD

properties of non-stationary noise, improve noise estimation in VADs operating on signals with conditions from severely degraded to mildly noisy.

Chapter 3

Literature Review

This chapter reviews literature relevant to the modified estimator, beginning with voice activity detectors and their dependence on accurate noise estimation, before describing the statistical properties of acoustic noise with particular attention to $1/f$ noise and long-range dependence. The noise estimation methods that form the basis of the proposed modification are then reviewed, followed by an introduction to fractional calculus and the Grünwald-Letnikov definition that the GL-MCRA estimator is based on. The evaluation metrics used to assess noise estimator performance are described at the end of the chapter.

3.1 Voice Activity Detectors

A voice activity detector (VAD) is a system that distinguishes between speech and non-speech activity in an audio signal. The task appears straightforward when the signal is clean, yet becomes considerably more difficult in the presence of noise [3]. Despite this challenge, they are fundamental to many speech processing applications: in mobile communications, VADs are used to detect when a participant is speaking or silent – when there is a pause or silence during the call, the VAD decides that the amount of data transmitted over the network can be reduced[6], thus saving bandwidth [7]; they are used in speech recognition of keywords for voice activated applications, to determine whether or not to begin the process; and in hearing aids to facilitate noise suppression between utterances [8]. In these applications, the VAD must classify successive segments of the signal as speech or non-speech in real-time and in unpredictable acoustic environments.

3.1.1 Global System for Mobile Communications VAD

The Global System for Mobile Communications (GSM) standard VAD was developed by British Telecom for the Pan-European digital mobile telephone system in 1989 [9]. Freeman et al. argue that distinguishing speech and noise through their level is impossible in low-SNR environments and that the distinction must be made through the differences in

spectral characteristics of the signal; and under the assumption that background noise is stationary over relatively long periods of time and thus the spectrum of the noise varies little each frame [9], speech can therefore be identified by detecting when the incoming signal's spectral characteristics diverge from those of the estimated background noise. The VAD proposed in their research employs a secondary VAD which detects the noise-only moments in the signal, from which coefficients can be derived for an inverse filter to attenuate the noise in the signal. The deviations that remain after inverse filtering the signal must be the speech signal, and the energy of this signal is compared to a threshold set to the energy of the filtered noise signal. This threshold level is updated during noise-only moments, and if the energy exceeds the threshold, then speech has been detected [9].

3.1.2 Cellular Network VAD

Srinivasan and Gersho [10] build on the approach presented by Freeman et al., proposing two VAD algorithms for cellular networks: one designed for vehicular noise environments, which assumes stationary noise, and one for babble noise environments, where the noise is non-stationary. The vehicle noise VAD (VNVAD) builds on the GSM approach by passing the signal through a whitening filter to flatten the noise's spectral envelope, such that the noise energy is spread uniformly across all frequencies, preventing the adaptive threshold from being skewed by the concentration of energy at lower frequencies typical of vehicular noise [10]. Thus, any energy exceeding the threshold is likely to be speech. This makes the detector more sensitive and consistent across the whole spectrum in low SNR environments. Srinivasan and Gersho further improve upon the VAD by adding a multi-band energy comparison scheme to improve the discrimination between speech and non-speech activity; evaluating the spectral flatness at the output of the noise-reduction filter to further improve the detection of speech; and an adaptive hang-over scheme to prevent clipping during or at the end of active speech, which ranges from 40 ms to 180 ms, compared to the fixed 100 ms from the GSM VAD [10]. The VNVAD was tested using speech signals with added vehicle noise at different SNRs. The tests revealed that in high-level noise environments audible clipping is substantially reduced in comparison to the GSM VAD. In very low-noise environments, only a marginal improvement was found over the GSM VAD.

The second VAD, designed for babble noise environments, takes a different approach due to the time-varying nature of babble noise. This VAD is composed of a primary VAD, which distinguishes between speech and non-speech, and a secondary VAD which adaptively sets the energy level threshold. The key difference with the babble noise VAD (BNVAD) is a separate comparison of the fraction of the low-frequency energy to a secondary adaptive threshold, also set during noise-only moments. The secondary VAD monitors the variability of the energy of the whole signal and the fraction of low-frequency energy over 400 ms windows; if neither measure varies significantly, the period is declared as noise-only and the thresholds are updated. Srinivasan and Gersho report that the BNVAD

performs significantly better than the spectral-difference based approach in babble-noise environments. At a 20 dB SNR, the voice activity factor is reduced by 10%, and by 25% at 5 dB SNR, without any noticeable clipping effect. Lastly, as in real environments the noise type may not be known in advance, Srinivasan and Gersho describe a method to combine the two VADs by giving control to the VNVAD when noise is detected to be stationary over a period of 400 ms, or passing control to the BNVAD if it is detected as non-stationary.

ITU-T G.729 Annex B

The International Telecommunication Union Telecommunication (ITU-T) G.729 Annex B standard, introduced in 1996, defined a VAD for wired telephony systems based on a similar heuristic approach [7].

3.1.3 Decision-Directed Method

In 1999, Sohn et al. present a foundational contribution to the field: a VAD intended for variable-rate speech coding, adopting the statistical framework of Ephraim and Malah, namely that the DFT coefficients of the speech and noise signals are asymptotically independent Gaussian variables [11]. They propose using a decision-directed (DD) method to estimate the *a priori* signal-to-noise ratio (SNR) (the expected speech-to-noise ratio at frequency bin k), which together with the measured *a posteriori* SNR (the SNR at frequency bin k, obtained from the magnitude of the signal's DFT), is used to compute the likelihood ratio of speech or non-speech for each frequency bin of each frame. They further propose a hang-over scheme based on the hidden Markov model (HMM) to prevent clipping of weak tails [6]. To evaluate the algorithm, they compared their DD method to a maximum likelihood method on their speech detection and false-alarm probabilities, with and without the HMM hang-over, and to the ITU standard G.729 Annex B [7]. A manually labelled clean speech signal was used as a reference, and the VAD algorithms were tested using the NOISEX-92 database, which consists of noisy speech samples corrupted by vehicular and white noise sources at 5 dB, 15 dB and 25 dB SNRs. Their results show that the proposed DD VAD provides a substantial performance improvement over the G.729B VAD in speech detection at nearly all SNR levels and types of noise, though at the cost of a higher false alarm rate across most conditions, that is, more misclassification of noise as speech [6]. The performance of the VAD is therefore critically dependent on the accuracy of the noise estimate, motivating the need for robust noise estimation methods.

3.1.4 Spectral Entropy VAD

More recently, a statistical method not based on signal energy has been developed. Liu and Demosthenous propose a VAD for low SNR in hearing aids, based on a spectral entropy method [12]. Spectral entropy is a concept derived from Shannon entropy, which quantifies the distribution of a signal's spectral power. The normalised power spectrum is interpreted

as a probability distribution and its Shannon entropy is computed to yield the spectral entropy, which can be useful for feature extraction in speech recognition [13, 14, 15]. Liu and Demosthenous make use of spectral entropy's more stable performance when applied to level-varying noise and more effectively characterises speech signal properties, resulting in improved accuracy at low SNRs. The spectral entropy VAD (SEVAD) they present is made more computationally efficient by using the output of hearing aid filterbanks to compute the spectral entropy instead of the FFT, as the filterbank outputs are lower spectral resolution [12]. Results show that the SEVAD is more accurate at detecting speech at SNR levels below 0dB, and is more computationally efficient than Sohn et al.'s statistical VAD and a separate power-based method. However, spectral entropy methods still rely on fixed or adaptively-set threshold for classification, making their performance sensitive to noise type, as the threshold must accurately track the noise spectral entropy across varying acoustic conditions [12].

3.1.5 Convolutional Neural Network VAD

Sehgal and Kehtarnavaz [16] demonstrate that statistical VADs, including G.729B and Sohn et al.'s, have a bias towards misclassifying noise as speech. They propose a convolutional neural network (CNN) in combination with log-mel filterbank energy features to enable real-time speech detection. The decision to use log-mel filterbank energy images as the input to the CNN is motivated by their consistently superior performance over linear-scaled Short Time Fourier Transform (STFT) spectrograms, constant-Q transform spectrograms, among other representations in audio classification tasks [16]. The log-mel energy spectrum captures the distribution of spectral power across mel-scaled frequency bands over short time periods. The mel scale is a frequency scale mapped to human pitch perception, where listeners judge all intervals as equidistant. The mel-frequency spectral coefficients, which comprise the log-mel energy spectrum, are concatenated across successive frames into a two-dimensional image which serves as the input to the CNN. By processing this image, the CNN examines local patterns in both time and frequency simultaneously, rather than the spectrum as a whole, allowing it to focus on cleaner regions of the spectrum and compensate for noisy areas when discriminating between speech and noise-only frames [16].

This CNN approach demonstrates a substantial performance increase over the statistical methods in identifying noise-only frames. The CNN was trained using speech signals with noise at different SNR levels, and its offline noise-hit-rate (NHR) was compared against G.729B (39.3%), Sohn et al.'s method (69%), and a random forest method (97.4%), with the CNN achieving 99.3%. The research also evaluates the NHR of the CNN VAD in real-time using 40 sentences in a crowded noise environment, the result of which were compared against the same manually labelled sentences. Results show that the CNN VAD achieves an average NHR of 99% compared to 27.9% for Sohn et al.'s method and 63.6% for G.729B [16], reducing the risk of noise frames being misclassified as speech.

The methods reviewed here represent a progression from heuristic energy-based approaches to statistical and deep learning methods, with improving accuracy in challenging noise conditions. Despite these advances, a common thread runs through all VAD-dependent noise estimators: their performance is dependent on the accuracy of the underlying noise estimate. As Sohn et al. [6] demonstrate, a corrupted noise estimate leads to higher false alarm rates, which in turn corrupts the noise estimate further. This motivates the development of noise estimators that do not rely exclusively on VAD-detected speech pauses to update, and instead operate continuously across both speech and noise frames. The MCRA-based approach reviewed in Section 3.3 addresses this limitation directly, and forms the basis of the proposed GL-MCRA estimator.

3.1.6 Discussion

Despite the substantial improvements in NHR that deep learning approaches such as the one presented by Sehgal and Kehtarnavaz demonstrate, statistical VADs retain some practical advantages in real-world scenarios. They require no offline training data, and thus are designed to adapt to changing noise without the need for retraining, and they are more computationally efficient. However, as Sohn et al. illustrate, their performance remains critically dependent on the accuracy of the underlying noise estimate. This justifies the need for more precise and adaptive noise estimation techniques, which are discussed in Section 3.3.

3.2 Acoustic Noise Characteristics

Acoustic noise in speech processing environments exhibits a wide range of statistical properties. This section outlines the key properties of acoustic noise relevant to this thesis, beginning with the concept of stationarity, before describing the spectral characteristics of different noise types – with particular interest in $1/f$ noise – and the distinction between short and long-range dependence. An important assumption often made in speech processing research is that of the modelled acoustic noise being a random process, that is, its behaviour cannot be predicted deterministically [17].

3.2.1 Stationarity

Stochastic processes are said to be stationary if their joint probability distribution does not change when shifted in time. There are several types of stationarity.

Strict Stationarity

A random process $X_{t_i} = \{X_{t_1}, X_{t_2}, \dots, X_{t_n}\}$ is *strictly stationary* if it has the same probability density function as its time-shifted version $X_{t_i+\tau} = \{X_{t_1+\tau}, X_{t_2+\tau}, \dots, X_{t_n+\tau}\}$, that is: $p(X_{t_1}, X_{t_2}, ..., X_{t_n}) = p(X_{t_1+\tau}, X_{t_2+\tau}, ..., X_{t_n+\tau})$ for all τ and all n. An example of a strictly

stationary random process is white noise, discussed further in Section 3.2.2. One well-known type of white noise is Gaussian white noise, defined by the normal distribution of its samples.

Wide-Sense Stationarity

Wide-sense stationarity (WSS) is more relaxed than strict stationarity. These kind of processes only require that the mean function, defined by $\mu_X(t) = \mu_X$, for all $t \in \mathbb{R}$, and the correlation function, defined by $R_X(t_1, t_2) = R_X(t_1 - t_2)$, for all $t_1, t_2 \in \mathbb{R}$, do not change with offset time, and that the autocorrelation function depends only on the lag between two time points rather than their absolute positions in time [17].

In practical terms, the statistical behaviour of WSS noise does not drift over time. The signal's average power and its correlational structure stay constant. This is what allows noise estimators to take a measurement of the noise spectrum during noise-only periods and assume that measurement is still valid for subsequent frames, when speech is present. Real acoustic noise is rarely truly stationary over long periods. The quasi-stationarity assumption treats noise as locally stationary. The noise properties change slowly enough that it can be approximated as WSS over short analysis windows of around 20–40 ms [11, 5]. VADs such as [6, 7, 9] and [10] rely on this assumption, updating their noise estimates during detected noise-only periods under the expectation that the noise spectrum will remain approximately stable in the subsequent speech frames. Quasi-stationarity breaks down when noise changes faster than the noise estimator can track it. As a consequence:

- If the estimator updates during a speech frame, the speech energy is added to the noise estimate, suppressing speech
- If noise increases during speech, the estimator will underestimate the noise and leave it in the enhanced signal
- Since VADs depend on the noise estimate, a corrupted estimate will cause more misclassification, compounding the interdependency between the VAD and the noise estimator described in Section 3.1

3.2.2 Colours of Noise

White Noise

A signal is termed "white noise" when its samples are drawn independently from the same probability distribution regardless of when they are observed, are uncorrelated at all non-zero lags and its power is distributed uniformly across the frequency spectrum. Mathematically, white noise exhibits a flat power spectral density (PSD) (Figure 3.1) and an autocorrelation function $R(\tau) = \sigma^2\delta(\tau)$, where correlations vanish for any lag time $\tau \neq 0$. This implies that past samples provide no information about future samples.

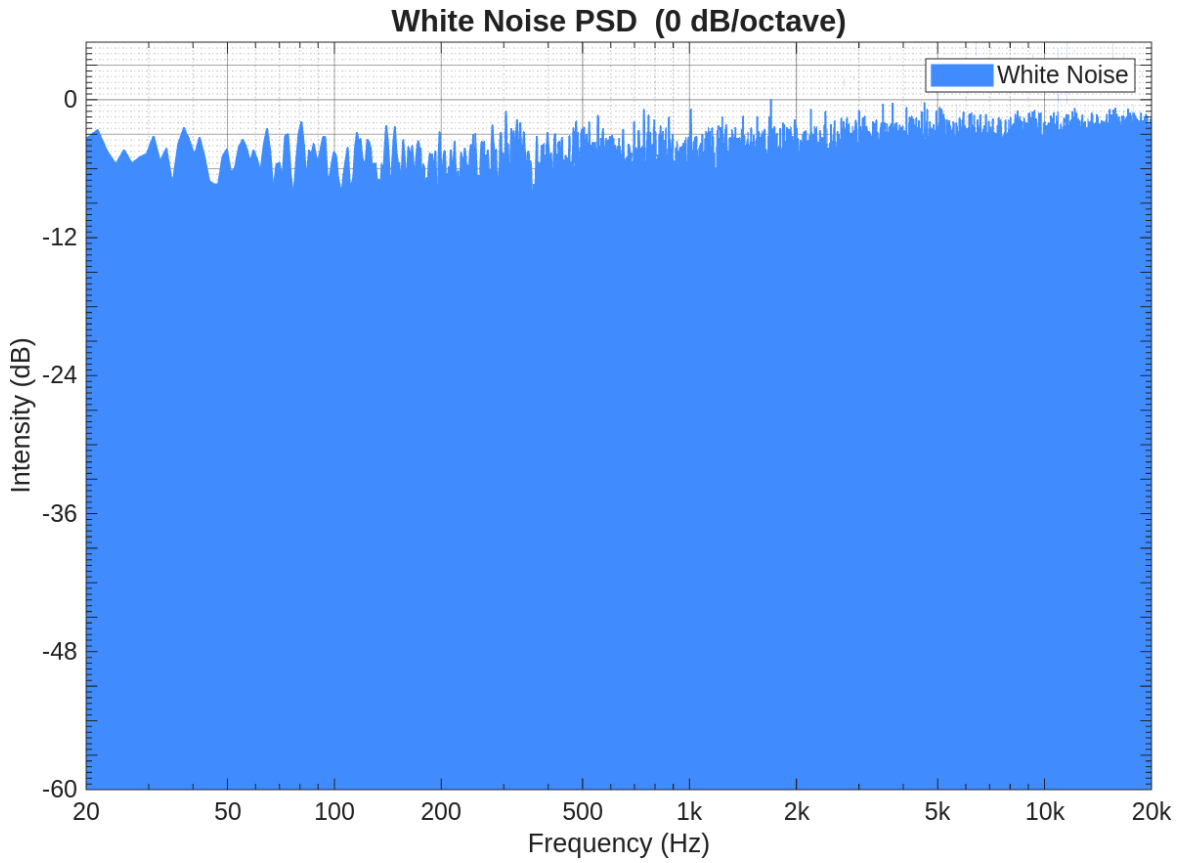


Figure 3.1: White Noise's Flat Power Spectral Density

The signal is therefore *memoryless* and can thus be characterised by a single parameter, its variance σ^2. When Gaussian white noise is added to a clean signal it is referred to as *additive white Gaussian noise (AWGN)*, and serves as the standard noise model in many speech processing algorithms [11].

Coloured Noise - $1/f^{\alpha}$

Real-world acoustic noise frequently deviates from the white noise ideal. Coloured noise exhibits frequency-dependent power distributions, characterised by power spectral densities of the form $S(f) \propto 1/f^{\alpha}$, where α determines their spectral shape. Common examples include pink noise ($\alpha \approx 1$) and brown noise ($\alpha \approx 2$). As α increases, power becomes progressively concentrated at lower frequencies, with steeper roll-off.

Pink noise ($\alpha \approx 1$) is one of the most commonly encountered forms of coloured noise in natural and acoustic environments. Its power decreases by 3 dB per octave , meaning each doubling of frequency carries half the power. This gives pink noise equal energy per octave. It arises in a wide variety of physical, biological, and electronic systems.

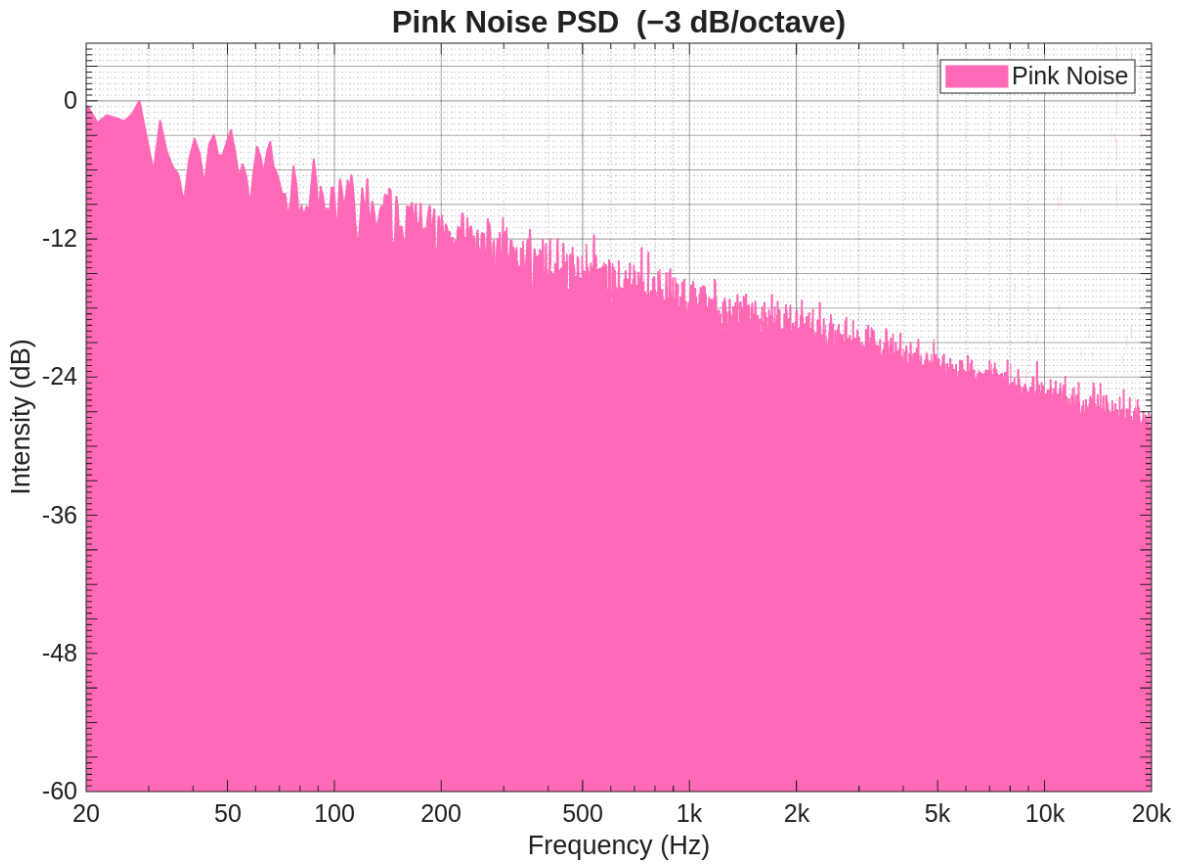


Figure 3.2: Pink Noise's -3 dB/Octave Slope

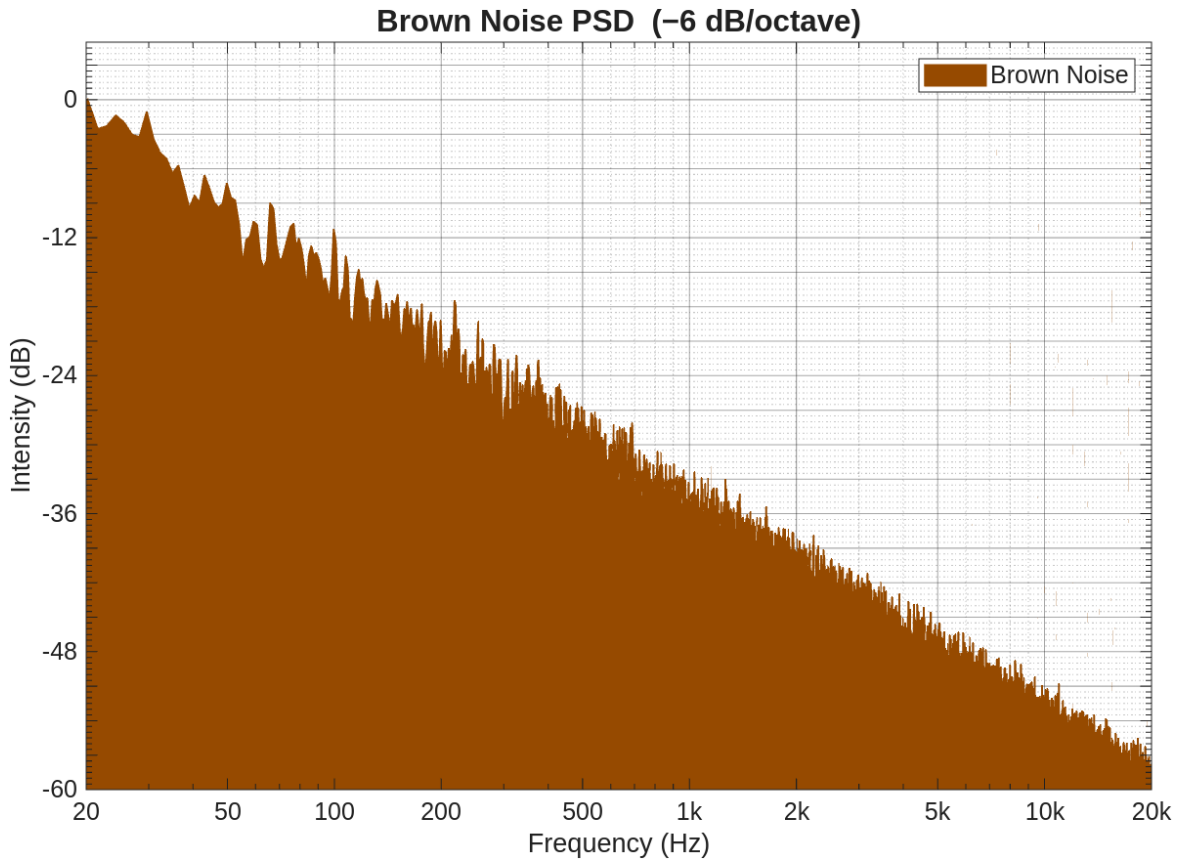


Figure 3.3: Brown Noise's -6 dB/Octave Slope

Brown noise ($\alpha \approx 2$), also known as Brownian or red noise, has a steeper spectral slope of 6 dB per octave. Its power is more heavily concentrated at low frequencies, giving it a rumbling character.

The $1/f^{\alpha}$ family of noise processes is closely related to long-range dependence. For $0 < \alpha < 1$, the power spectral density diverges at low frequencies, which is the frequency-domain signature of LRD. This connection is central to the motivation for the proposed estimator.

3.2.3 Short and Long-Range Dependence

In time series analysis, different models are developed to make predictions of future values. Autoregressive moving average (ARMA) models are a type of short-range dependent models for signals whose correlation functions decrease exponentially fast over time. They are widely used, but cannot accurately model signals where correlation exhibits a power-law decay [18, 19]. In the 1950s, while studying the water levels of the Nile River over long periods of time, to determine the optimal size for a dam, Harold Edwin Hurst discovered

that the variation in water level was not random: high and low levels persisted over long periods of time. The persistence meant that the river's current behaviour depended on its past over very long timescales. This phenomenon is known as long-range dependence (LRD) or long memory.

Properties of Short-Range Dependent Noise

Short-range dependent (SRD) processes are characterised by rapidly decaying correlations. More formally, their autocorrelation function $\rho(k)$ decays exponentially fast as the lag k increases, and the sum of autocorrelations is finite:

$$\sum_{k=-\infty}^{\infty} |\rho(k)| < \infty. \tag{3.1}$$

This implies that observations far apart in time are effectively uncorrelated, and the process has only short-term memory. As a result, SRD processes are well described by models such as ARMA, where dependence is captured by a finite number of parameters.

In the frequency domain, SRD processes have a bounded power spectral density that does not diverge at low frequencies. A typical example is white noise, where no temporal correlation exists between samples.

Properties of Long-Range Dependent Noise

Long-range dependent (LRD) processes exhibit fundamentally different behaviour. Their autocorrelation function decays according to a power law:

$$\rho(k) \sim k^{-\alpha}, \quad \text{as } k \to \infty, \quad 0 < \alpha < 1, \tag{3.2}$$

which implies that correlations persist over arbitrarily long time lags. In contrast to SRD processes, the sum of autocorrelations diverges:

$$\sum_{k=-\infty}^{\infty} |\rho(k)| = \infty. \tag{3.3}$$

This slow decay means that distant observations remain statistically dependent, giving rise to long memory in the process.

An equivalent characterisation can be given in the frequency domain: LRD processes exhibit a power spectral density that behaves as:

$$S(f) \sim \frac{1}{f^{\alpha}}, \quad \text{as } f \to 0, \tag{3.4}$$

with $0 < \alpha < 1$. This divergence at low frequencies reflects the dominance of long-term fluctuations and is characteristic of $1/f$ noise.

The degree of long-range dependence is often quantified by the Hurst exponent $H \in (0.5, 1)$, where larger values indicate stronger persistence. In such processes, high values are more likely to be followed by high values, and low values by low values, even across long time intervals.

3.3 Noise Estimation Methods

3.3.1 Minima Controlled Recursive Averaging

Cohen and Berdugo [1] introduce the minima controlled recursive averaging (MCRA) method for noise estimation, which addresses the fundamental limitation of VAD dependent noise estimators: relying exclusively on detected speech pauses to update the noise estimate. They explain that the weakness comes from low SNRs degrading detection reliability, and that it is possible that there are not enough noise-only periods between speech periods to correctly track the noise and calculate its spectrum [1]. The research they improve on used minima values of a smoothed power estimate of the noisy signal, but this was affected by outliers and had large variance. With MCRA, Cohen and Berdugo present a method to estimate noise by recursively averaging past spectral power measurements, and using a smoothing parameter, adjusted by the signal presence probability in subbands. By using the ratio between local energy of the noisy speech in a frame and its minimum within a specified time period, and comparing the ratio to a threshold value, speech can be detected in a given frame of a subband. A higher ratio indicates speech has been detected.

The discrete-time estimation works by dividing the speech signal, denoted by $x(n)$, and the uncorrelated additive noise, denoted by $d(n)$ into overlapping frames through a window function and analysing them via STFT [1]:

$$Y(k, \ell) = \sum_{n=0}^{N-1} y(n + \ell M) h(n) e^{-j(2\pi/N)nk} \tag{3.5}$$

where k and ℓ are the frequency bin and time frame respectively, h is an analysis window of length N, and M is the hop size between successive frames. A binary hypothesis test is employed to determine whether a frame contains only noise, or speech plus noise, where the STFT of the clean speech and the noisy speech signals are represented respectively by $X(k, \ell)$ and $D(k, \ell)$ [1]:

$$H_0(k, \ell) : Y(k, \ell) = D(k, \ell) \tag{3.6}$$

$$H_1(k, \ell) : Y(k, \ell) = X(k, \ell) + D(k, \ell) \tag{3.7}$$

This is the statistical framework the method is built on. The outcome of the hypothesis test drives the speech presence probability estimate, and controls how aggressively the noise estimator updates. The variance of the noise in the kth subband is expressed by

$\lambda_d(k,\ell) = E[|D(k,\ell)|^2]$, to which a temporal recursive smoothing is applied to obtain its estimate during noise-only periods, shown by:

$$H_0'(k,\ell) : \hat{\lambda}_d(k,\ell+1) = \alpha_d \hat{\lambda}_d(k,\ell) + (1-\alpha_d)|Y(k,\ell)|^2 \tag{3.8}$$

$$H_1'(k,\ell) : \hat{\lambda}_d(k,\ell+1) = \hat{\lambda}_d(k,\ell) \tag{3.9}$$

where H_0' and H_1' represent the hypothetical speech absence and presence respectively, and $\alpha_d (0 < \alpha_d < 1)$ is a smoothing parameter. Cohen and Berdugo distinguish between the two hypotheses pairs: H_0 and H_1 govern the estimation of the clean speech signal, while H_0' and H_1' control the adaptation of the noise spectrum [1]. Under H_0', the noise estimate is updated as a weighted average of the previous estimate and the current observed power, with α_d controlling the balance between past and present observations. Under H_1', the estimate is held constant to prevent speech energy from corrupting the noise estimate.

However, since speech presence is never known with certainty, a hard binary switch between H_0' and H_1' risks corrupting the noise estimate when speech presence is incorrectly declared absent. Cohen and Berdugo address this by replacing the binary decision with a soft probability-weighted combination, introducing the conditional signal presence probability $p'(k,\ell)$. The noise estimate is then given by [1]:

$$\begin{aligned}\hat{\lambda}_d(k,\ell+1) = \; & \hat{\lambda}_d(k,\ell)p'(k,\ell) \\ & + [\alpha_d \hat{\lambda}_d(k,\ell) + (1-\alpha_d)|Y(k,\ell)|^2] \\ & \times (1-p'(k,\ell)) \\ = \; & \tilde{\alpha_d}(k,\ell)\hat{\lambda}_d(k,\ell) \\ & + [1-\tilde{\alpha_d}(k,\ell)]|Y(k,\ell)|^2 \end{aligned} \tag{3.10}$$

where $\tilde{\alpha}_d(k,\ell)$ is a time-varying smoothing parameter, defined as [1]:

$$\tilde{\alpha}_d(k,\ell) \triangleq \alpha_d + (1-\alpha_d)p'(k,\ell) \tag{3.11}$$

Thus, when speech is absent, the value of p' is low and the estimate updates quickly, and when speech is present, the value of p' is high, so the estimate updates much more slowly. Crucially, the estimator never stops entirely, allowing it to track abrupt changes in the noise spectrum without relying on a VAD.

The speech presence probability $p'(k,\ell)$ is central to the MCRA estimator. Cohen and Berdugo estimate it using the ratio between the local energy of the noisy speech and its minimum within a specified time window [1]. The local energy of the noisy speech is calculated by smoothing the power of its STFT in time and frequency. The frequency smoothing is provided by:

$$S_f(k,\ell) = \sum_{i=-w}^{w} b(i)|Y(k-i,\ell)|^2 \tag{3.12}$$

where b is the window function with length $2w + 1$. Using the result of the frequency smoothing, the time smoothing is given by:

$$S(k, \ell) = \alpha_s S(k, \ell - 1) + (1 - \alpha_s) S_f(k, \ell) \tag{3.13}$$

The minimum for the current frame is then obtained by comparing the local energy sample-by-sample to the minimum of the previous frame. Two variables are used to track the minimum:

$$S_{min}(k, \ell) = min\{S_{min}(k, \ell - 1), S(k, \ell)\} \tag{3.14}$$
$$S_{tmp}(k, \ell) = min\{S_{tmp}(k, \ell - 1), S(k, \ell)\} \tag{3.15}$$

Every L frames, S_{tmp} is used to refresh S_{min}, ensuring the minimum reflects a window of at least L frames but no more than $2L$ frames [1]. The resolution of the minima search is controlled by L [1]. Cohen and Berdugo recommend a window length of 0.5–1.5 seconds. The ratio between the local energy of the noisy speech and its minimum, is represented by: $S_r(k, \ell) \triangleq S(k, \ell)/S_{min}(k, \ell)$. If S_r is close to 1, the current energy is close to the minimum, suggesting noise only. If S_r is large, the current energy is larger than the minimum, suggesting speech is present.

To avoid incorrectly deciding whether speech is present or absent, Cohen and Berdugo use a Bayes minimum-cost decision rule:

$$\frac{p(S_r|H_1)}{p(S_r|H_0)} \underset{H_0'}{\overset{H_1'}{\gtrless}} \frac{c_{10} P(H_0)}{c_{01} P(H_1)} \tag{3.16}$$

The decision rule depends on the likelihood ratios $p(S_r|H_1)$ and $p(S_r|H_0)$, which are the probability of observing the value of S_r given that speech is present, or the probability of observing the value of S_r given that only noise is present. This ratio informs which hypothesis makes the observed S_r more likely. The right-hand side is a threshold determined by the prior probabilities of speech absence and presence, $P(H_0)$ and $P(H_1)$, and the costs c_{10} and c_{01}, which represent the penalties for falsely declaring speech present and falsely declaring noise-only respectively. Together they adjust the decision based on how likely speech is and how costly each type of mistake is [1]. Since the likelihood ratio $p(S_r|H_1)/p(S_r|H_0)$ is monotonically increasing in S_r, comparing it against a threshold is equivalent to comparing S_r directly against a threshold δ [1], simplifying the decision rule to:

$$S_r(k, \ell) \underset{H_0'}{\overset{H_1'}{\gtrless}} \delta \tag{3.17}$$

A frame is therefore declared to contain speech when S_r exceeds the threshold δ, and noise-only when it does not. However, rather than feeding this hard binary decision di-

rectly into the noise estimator, Cohen and Berdugo use the smoothing operator α_p along with the speech presence indicator $I(k, \ell)$ for the estimator $p'(k, \ell)$ [1]:

$$\hat{p}'(k, \ell) = \alpha_p \hat{p}'(k, \ell - 1) + (1 - \alpha_p) I(k, \ell) \tag{3.18}$$

where $I(k, \ell) = 1$ if $S_r > \delta$, and 0 if $S_r < \delta$. The resulting probability $p'(k, \ell)$ drives the time-varying smoothing operator $\tilde{\alpha}_d$, completing the MCRA estimation loop.

Cohen and Berdugo compared their MCRA method against a weighted average method, using three measures: segmental SNR improvement, spectrogram analysis, and informal listening tests, across a range of stationary and non-stationary noise conditions at SNR levels from -5 to 10 dB [1]. MCRA demonstrates significantly faster recovery from abrupt noise increases, adapting in approximately 3 seconds compared to over 12 seconds for the weighted average method, while maintaining accurate tracking under stationary conditions.

3.3.2 Improved Minima Controlled Recursive Averaging

Cohen subsequently refined the MCRA method in a follow-up work, introducing the improved minima controlled recursive averaging (IMCRA) method [2], which improves the minimum tracking during speech activity, addressing limitations in the speech presence probability estimate, and derives a bias compensation factor. IMCRA uses two passes of smoothing and minima tracking. The first pass provides a rough voice activity detection per frequency band. The second pass then excludes strong speech components from the smoothing process, making minimum tracking more robust during active speech [2].

3.3.3 DFT Subspace (DFT-SS)

An alternative approach to noise power spectral density estimation is presented by Hendriks et al. [20]. The method uses the discrete Fourier transform (DFT) subspace, which overcomes a fundamental limitation of minimum statistics based methods: the inability to update the noise estimate when speech is continuously present in a frequency bin. The method constructs correlation matrices from time-series DFT coefficients at each frequency bin, and applies eigenvalue decomposition (a technique that breaks a matrix down into its principal directions and the energy associated with each) to separate the noisy-speech signal into a signal subspace and a noise-only subspace. Since voiced speech at a particular frequency bin can be expressed as a linear combination of a few complex exponentials, its energy concentrates in only a few directions of the observation space, leaving the remaining directions free of speech entirely. The noise PSD is therefore estimated from the eigenvalues belonging to these speech-free directions, enabling continuous updates even when speech is actively present. Hendriks et al. demonstrate this property of noise tracking by comparing it to a minimum statistic method, simulating a speech signal with a 190 Hz sinusoid, and show that their method tracks the noise PSD even in the presence of constant speech energy [20].

The proposed method models the observed DFT coefficients at frequency bin k and time-frame i as $Y(k,i) = X(k,i) + D(k,i)$, where $X(k,i)$ and $D(k,i)$ denote the clean speech and noise DFT coefficients respectively. As with previous methods, the additive noise is assumed to be zero mean and uncorrelated with the clean speech signal. To exploit the temporal structure of the signal at each frequency bin, $M = p_1 + p_2 + 1$ consecutive DFT coefficients are collected into a vector $Y(k,i) = [Y(k,i-p_1), \ldots, Y(k,i+p_2)]^T \in \mathbb{C}^M$, where p_1 and p_2 denote the number of past and future frames relative to the current frame i. The noisy speech correlation matrix is then defined by $R_Y(k,i) = E[Y(k,i)Y^H(k,i)]$, the speech correlation matrix by $R_X(k,i) = E[X(k,i)X^H(k,i)]$ and the noise correlation matrix by $R_D(k,i) = E[D(k,i)D^H(k,i)]$. Following the previously stated assumption that the speech and noise are additive and uncorrelated, the noisy speech correlation matrix can be rewritten as: $R_Y(k,i) = R_X(k,i) + R_D(k,i)$. Furthermore, the noise DFT coefficients are assumed to be uncorrelated across frames, such that $R_D(k,i) = \sigma_D^2(k,i)I_M$, where σ_D^2 is the noise PSD to be estimated [20].

Since voiced speech is low-rank, it occupies only Q out of M dimensions of the observation space. The remaining $M - Q$ dimensions — the noise-only subspace — are untouched by speech. Because the noise is assumed to be white, it contributes equally to all dimensions without disturbing this structure, meaning the eigenvalues in the noise-only subspace each equal the noise PSD σ_D^2. The noise PSD is therefore estimated as the average of these eigenvalues enabling continuous updates even during speech activity:

$$\hat{\sigma}_D^2(k,i) = \frac{1}{M-Q} \sum_{l=Q+1}^{M} \hat{\lambda}_{Y_l}(k,i) \tag{3.19}$$

However, computing $\hat{\sigma}_D^2(k,i)$ requires knowledge of the signal subspace dimension Q, which is unknown in practice and must be estimated from the data. To estimate Q, Hendriks et al. propose a hypothesis testing approach that exploits the previous frame's noise PSD estimate as prior information, and assumes slow noise variations. An eigenvalue is assigned to the signal subspace if the probability of observing a value equal or larger than $\hat{\lambda}_{Y_l}$ is smaller than a minimum probability P_{min} [20].

$$\int_{\hat{\lambda}_{Y_l}}^{+\infty} f_{\Lambda_D}(\lambda_D) d\lambda_D < P_{min} \tag{3.20}$$

Eigenvalues are evaluated in increasing order, and once an eigenvalue fails this test, it and all subsequent larger eigenvalues are assigned to the signal subspace, determining Q.

To evaluate its performance, the DFT-SS method was compared with a minimum statistics-based method, using speech signals degraded by several noise types at input SNRs of 0, 5, 10, and 15 dB. In comparison to minimum statistics, the proposed method greatly improves tracking, due to its ability to track noise during speech-active frames, reduces the estimation error and improves the segmental SNR, across all noise types and all SNRs. The largest improvement was for the non-stationary noise of a passing train, where the estimation error at 0 dB SNR was 0.33 for minimum statistics and 0.16 for the DFT-SS

method. The estimation error for white Gaussian noise was again 0.33 for minimum statistics, but 0.068 for DFT-SS. The objective improvements are further confirmed by subjective evaluation, where listeners preferred the proposed method over minimum statistics under all tested conditions, with preference reaching 96% for non-stationary white noise. Despite this, the eigenvalue decomposition required at each frequency bin per frame results in an increase in computational complexity, motivating the development of less complex noise estimation methods.

3.3.4 Minimum Mean-Squared Error

Hendriks et al. subsequently propose a method based on a minimum mean-squared error (MMSE) estimator [21]. Similar to their DFT-SS method, the proposed MMSE method also uses the DFT and shows similar performance, but exhibits lower computational complexity, better suited for less powerful devices such as mobile phones or hearing aids. The standard assumptions of additive, zero-mean noise, uncorrelated with the clean speech signal, are retained. Hendriks et al. derive the noise PSD estimate as the MMSE of the noise magnitude-squared coefficients. Assuming complex Gaussian distributions for both speech and noise DFT coefficients, this yields the closed-form estimator:

$$E\{N^2|Y\} = \left(\frac{1}{(1+\xi)^2} + \frac{\xi}{(1+\xi)\zeta} \right) |Y|^2 \tag{3.21}$$

where ξ and ζ denote the *a priori* and *a posteriori* SNR respectively. The noise PSD estimate is therefore a scalar multiple of the observed power $|Y|^2$, with a weight that adapts to the local SNR conditions, and it requires only simple scalar operations per time-frequency bin. Equation 3.21 gives a theoretically unbiased estimator, assuming the true ξ is known. As the value of ξ is not known it must be estimated. Hendriks et al. use a maximum likelihood (ML) estimator for ξ based on the previous frame:

$$\hat{\xi}(k,i) = \max\left(\frac{|Y(k,i)|^2}{\hat{\sigma}_W^2(k,i-1)} - 1, 0 \right) \tag{3.22}$$

This `max` operator in the estimator clips negative values to zero, which occur when $|Y|^2$ is smaller than the previous noise estimate, particularly at low SNR. This introduces a downward bias, causing the noise PSD to be underestimated. To correct for this, a bias compensation factor $B(\hat{\xi}_{DD})$ is applied, where $\hat{\xi}_{DD}$ denotes the decision-directed *a priori* SNR estimate — a smoother and more stable alternative to the ML estimate, better suited for computing a reliable correction. The final bias compensation is then given by:

$$\widehat{\sigma}_W^2 = E\{N^2|y;\hat{\xi}\} B(\hat{\xi}_{DD}) \tag{3.23}$$

As demonstrated in Fig. 2 of Hendriks et al. [21], this compensation leads to essentially unbiased estimates across a wide range of SNR conditions.

To evaluate the performance of the MMSE method, it was compared to five other methods: minimum statistics (MS), DFT-SS, high resolution DFT (HR-DFT), weighted noise estimation (WNE) and a recursive smoothing MMSE method by Yu [21], using speech signals degraded by several noise sources at SNRs of 0, 5, 10, and 15 dB. Noise tracking performance is measured using the symmetric log-error distortion, and speech performance using segmental SNR and perceptual evaluation of speech quality (PESQ) [22]. Results show that the MMSE method's noise tracking and speech enhancement performance is comparable to DFT-SS and HR-DFT, while outperforming MS and WNE across all conditions, with improvements of approximately 1dB in segmental SNR and 0.25 PESQ points for non-stationary noise sources. The key advantage over DFT-SS is its computational efficiency. This can be seen in Table 1 in Hendriks et al. [21], which shows the normalised MATLAB processing-time for each of the methods, where the proposed MMSE method — with its low computational complexity — processes 42 times faster than the DFT-SS method. This low complexity, combined with strong noise tracking performance, makes the method particularly well suited for resource-constrained applications.

Like MCRA and IMCRA, the MMSE estimator relies on first-order recursive averaging — the noise PSD estimate at each frame depends on the estimate from the previous frame through $\hat{\sigma}_W^2(k, i-1)$ in Equation 3.22, imposing the same exponentially decaying memory structure [21].

The methods reviewed in this section represent a progression from VAD-dependent noise estimation toward increasingly robust VAD-independent approaches. MCRA establishes continuous tracking, freeing itself from speech pause detection, and instead relying on averaged spectral power values from past frames and a smoothing parameter that adapts to the speech presence probability at each frequency bin. IMCRA refines this through more robust minima tracking. Hendriks et al.'s DFT-SS method takes this further, exploiting the low-rank structure of voiced speech to cleanly separate the observation space into a speech subspace and a noise-only subspace — enabling noise tracking to continue uninterrupted even during active speech. The MMSE method subsequently achieves comparable tracking performance at substantially lower computational cost. Despite their differences, MCRA, IMCRA, and the MMSE method share a common constraint: the recursive averaging is governed by a first-order filter, imposing an exponentially decaying memory structure. The DFT-SS method takes a different approach, estimating the noise PSD from the eigenstructure of the observed signal rather than through recursive smoothing. The following section examines fractional-order filters, which allow the recursive averaging to move beyond integer order. Unlike first-order recursive averaging, which retains only an exponentially decaying memory of past observations, fractional-order filters can capture long-range dependence, potentially making them better suited to noise processes whose statistical properties evolve slowly over time.

3.4 Evaluation of Noise Estimators

Evaluating a noise estimator requires metrics that capture both the accuracy of the noise estimate itself and the perceptual quality of the resulting enhanced speech. These are related, but distinct quantities — an accurate noise estimate does not necessarily produce perceptually superior speech, and perceptual quality alone cannot reveal where estimation errors occur. Therefore, a combination of objective and subjective measures is employed.

3.4.1 Signal-to-Noise Ratio

The Signal-to-Noise Ratio (SNR) quantifies the ratio between the power of a desired signal and the power of background noise. It provides an objective measure of how much a signal is corrupted by noise.

In practical scenarios, the observed signal $x_d(n)$ is typically modelled as the sum of a clean speech signal $x_\phi(n)$ and additive noise $v(n)$:

$$x_d(n) = x_\phi(n) + v(n). \tag{3.24}$$

where (n) is the sample index.

When the clean reference signal is available, the SNR can be computed as:

$$\text{SNR} = 10 \log_{10} \frac{\sum_n x_\phi^2(n)}{\sum_n v^2(n)}. \tag{3.25}$$

Since the noise signal $v(n)$ is typically unknown, it is estimated as the difference between the noisy and clean signals [23]:

$$v(n) = x_d(n) - x_\phi(n), \tag{3.26}$$

yielding the practical form:

$$\text{SNR} = 10 \log_{10} \frac{\sum_n x_\phi^2(n)}{\sum_n (x_d(n) - x_\phi(n))^2}. \tag{3.27}$$

3.4.2 Segmental Signal-to-Noise Ratio

Global SNR averages power over the entire signal, which is problematic given the highly non-stationary nature of speech. High-energy plosives or loud voiced segments can bias the ratio positively, making it appear larger than it is. The opposite is true for low-energy segments or segments with multiple pauses — skewing the ratio negatively. Segmental SNR (SegSNR) provides a frame-by-frame measure that is then averaged. This prevents loud parts of the signal under analysis from biasing the ratio. It also makes SegSNR more sensitive to local errors. SegSNR is defined as [23]:

$$\text{SegSNR} = \frac{10}{M} \sum_{m=0}^{M-1} \log_{10} \sum_{n=Nm}^{Nm+N-1} \left(\frac{x_\phi^2(n)}{[x_d(n) - x_\phi(n)]^2} \right) \tag{3.28}$$

where x_ϕ is the clean speech, x_d is the distorted speech, N is the frame length, and M is the number of frames being evaluated. Window lengths are typically 15–20 ms. The advantage of segmental SNR over global SNR, is that it assigns equal weight to both loud and quiet parts of the utterance [23]. Silent frames are excluded from the sum, since near-zero speech energy can produce large negative SNR values that would bias the measure [23]. Cohen and Berdugo use segmental SNR improvement — the difference between the enhanced and noisy SegSNR — as a primary objective metric for MCRA [1], as does Cohen for IMCRA [2] and Hendriks et al. for both DFT-SS and MMSE [20, 21].

Cohen's IMCRA paper introduces a more direct measure of estimation accuracy: the segmental relative estimation error (SegErr) [2]:

$$\text{SegErr} = \frac{1}{L} \sum_{\ell=0}^{L-1} \frac{\sum_k \left[\hat{\lambda}_d(k,\ell) - \lambda_d(k,\ell)\right]^2}{\sum_k \lambda_d^2(k,\ell)} \tag{3.29}$$

which measures the normalised squared error between the estimated and true noise PSD, averaged uniformly across all frames. However, Hendriks et al. note that SegErr is asymmetric — it is more sensitive to overestimation of the noise PSD than to underestimation [20].

3.4.3 Signal-to-Noise Ratio Levels

In the evaluation of their respective methods, Ephraim and Malah use an SNR range of $-5, 0$, and 5 dB [4], Cohen and Berdugo a range of $-5, 0, 5, 10, 15$, and 20 dB [1]. Hendriks et al. use the NOIZEUS corpus in [20] and thus a range of $0, 5, 10$, and 15 dB, and the same range in [21]. Thus, as this thesis will make use of the NOIZEUS corpus, the additive noise will be mixed at SNR ranges of 0, 5, 10 and 15 dB.

3.4.4 Perceptual Evaluation of Speech Quality

Perceptual Evaluation of Speech Quality (PESQ) is an objective measure of perceived speech quality standardised as ITU-T P.862 [24]. It compares a degraded speech signal against a clean reference and produces a score mapped to a predicted mean opinion score (MOS) value on a scale from 1.0 (bad) to 4.5 (excellent). As Rix et al. write, PESQ is well-suited to noise processing, noting that it can be assessed by presenting PESQ with the clean unprocessed original and the noise-suppressed signal [22]. Furthermore, PESQ was standardised for narrowband telephone speech sampled at 8 kHz, which matches the sampling rate of the NOIZEUS corpus, ensuring that the evaluation operates within the validated scope of the measure [22].

3.4.5 Mean Opinion Score

Subjective evaluation is conducted using the Mean Opinion Score (MOS), in which participants rate the perceived quality of enhanced speech on a five-point scale from 1 to 5. This approach is consistent with the informal listening tests used by Cohen and Berdugo [1] and Cohen [2], and the preference listening test conducted by Hendriks et al. [20]. Participants rate a representative subset of enhanced utterances covering each noise condition, providing a subjective complement to the objective PESQ scores.

3.5 Fractional-Order Filters

In signal processing, observed signals are often assumed to be independent and identically distributed, with correlations that decay exponentially as the time difference between samples increases [19]. These assumptions simplify statistical analysis of the signals, and promote widely used models such as autoregressive moving average (ARMA) and the Gaussian noise model. However, many real-world signals often go against these assumptions, exhibiting non-stationary behaviour and correlations that decay far more slowly than exponentially [19]. As discussed in Section 3.2, $1/f$ noise is a canonical example: its correlations decay as a power law, a property known as long-range dependence, which exponential memory models cannot adequately capture. This makes the case for the use of fractional calculus, which generalises integer-order differentiation and integration to non-integer orders, and naturally gives rise to power-law memory structures [19].

3.5.1 Fractional Calculus

Fractional calculus (FC) is the branch of mathematics concerned with integrals and derivatives of arbitrary real order, generalising the classical notions of integer-order operators such as D^{-1}, D^{1} and D^{2} to non-integer orders such as $D^{0.5}$ and $D^{-\pi}$ [25]. Integer-order linear differential equations are generally characterized by sums of exponential functions, while their fractional-order counterparts are governed by the Mittag-Leffler function (the FC analogue of the exponential function), known for its power-law asymptotic behaviour. For this reason, FC provides a natural framework for modelling and processing signals with power-law correlation structure [19]. Just as with integer-order calculus, integration and differentiation are inverse operations, but in fractional-order calculus, integration and differentiation are unified under the same operator, with integration corresponding to negative values of the order. A fractional integral of order α is therefore denoted $D^{-\alpha}$, and the Grünwald-Letnikov (GL) definition covers both cases within the same framework [25].

Riemann-Liouville Definition

The Riemann-Liouville definition expresses the fractional derivative as an integer-order derivative applied to a fractional integral [25, 19]:

$$ {}_aD_t^{\alpha} f(t) = \frac{1}{-(n-\alpha)} \left(\frac{d}{dt}\right)^n \int_a^t (t-\tau)^{n-\alpha-1} f(\tau) d\tau \tag{3.30} $$

While widely used in mathematics, the Riemann-Liouville definition requires fractional-order initial conditions at the lower terminal $t = a$, which Podlubny notes have no known physical interpretation [25]. The integer-order derivative $\left(\frac{d}{dt}\right)^n$ is applied *after* the integral, producing boundary terms at $t = a$ that take the form of fractional derivatives of f. In the context of noise estimation, these initial conditions correspond to fractional derivatives of the observed signal power at the starting point — quantities that cannot be estimated from finite samples and unknown prehistory. This contrasts with the first-order IIR recursive smoothing used in MCRA:

$$ \hat{\lambda}_d(k, \ell+1) = \alpha \hat{\lambda}_d(k, \ell) + (1-\alpha)|Y(k, \ell)|^2 \tag{3.31} $$

which requires only $\hat{\lambda}_d(k, 0)$ — the noise estimate at the first frame — a physically meaningful quantity that can be approximated from the observed signal power.

Caputo Definition

To address this limitation, Caputo proposed an alternative definition in which the integer-order derivative is applied to f before integration [25]:

$$ {}_a^C D_t^{\alpha} f(t) = \frac{1}{-(n-\alpha)} \int_a^t (t-\tau)^{n-\alpha-1} f^{(n)}(\tau) d\tau \tag{3.32} $$

By applying the integer-order derivative $f^{(n)}(\tau)$ inside the integral, the resulting initial conditions take the form of integer-order derivatives $f(a), f'(a), f''(a), \ldots$, which are physically interpretable and measurable [25]. However, both the RL and Caputo definitions are continuous-time operations, requiring differentiable functions and making them unsuitable for direct discrete-time implementation.

Grünwald-Letnikov Definition

The Grünwald-Letnikov (GL) definition (Equation 3.33) is particularly suited to discrete-time signal processing, as it expresses the fractional derivative directly as a weighted sum of past samples [25].

$$ {}_aD_t^r f(t) \approx \frac{1}{h^r} \sum_{j=0}^{[(t-a)/h]} w_j^{(r)} f(t-jh) \tag{3.33} $$

The weights $w_j^{(r)}$ are computed recursively by:

$$ w_0^{(r)} = 1, \quad w_j^{(r)} = \left(1 - \frac{r+1}{j}\right) w_{j-1}^{(r)}, \quad j = 1, 2, \ldots \tag{3.34} $$

where h is the step size, r is the fractional order, a and t are the lower and upper limits of differentiation, and j is the lag index over past samples. Since the GL definition operates directly on discrete past values, its initial conditions are simply past observations of the signal — actual values that can be obtained directly from the data. Furthermore, the weights $w_j^{(r)}$ decay as a power law with increasing lag j, in contrast to the exponential decay of integer-order filters, making the GL integrator naturally suited to signals with long-range dependence such as $1/f$ noise [19].

A practical consideration for implementing the GL integrator is the growing length of the sum in Equation 3.33. For $t \gg a$, the number of past samples to sum increases without bound. Podlubny addresses this through the short-memory principle, which truncates the sum to a finite memory window of length L [25].

$$ {}_aD_t^r f(t) \approx_{t-L} D_t^r f(t), \quad (t > a + L) \tag{3.35} $$

Substituting the GL approximation into the short-memory principle gives:

$$ {}_{t-L}D_t^r f(t) \approx \frac{1}{h^r} \sum_{j=0}^{[L/h]} w_j^{(r)} f(t - jh) \tag{3.36} $$

Although the proposed method is implemented offline, L remains a meaningful design parameter — too short a window discards the long-range dependence that motivates the GL approach, while too long a window incorporates observations that no longer reflect current noise conditions.

Chapter 4

Final Problem Statement

The preceding sections reveal a common limitation across the noise estimation methods reviewed. Despite their different approaches, MCRA (Equations 3.8 and 3.13), IMCRA, and the MMSE method of Hendriks et al. (Equation 3.22) all rely on first-order recursive averaging, imposing an exponentially decaying memory on the noise estimate. VADs operating at low SNR depend critically on the accuracy of this estimate, yet the exponential memory structure is fundamentally mismatched to the power-law correlation decay of $1/f$ noise — a property of many real acoustic environments discussed in Section 3.2. As Sheng et al. note, long-range dependent processes such as $1/f$ noise are better modelled using fractional-order systems, whose solutions exhibit power-law rather than exponential behaviour [19].

This thesis therefore proposes replacing the first-order IIR recursive averaging in MCRA with a Grünwald-Letnikov fractional-order integrator of order r, where $0 < r < 1$. Because the GL weights decay as $w_j^{(r)} \sim j^{-r-1}$ (Equation 3.34), the resulting estimator retains a longer and more appropriately weighted history of past observations than an exponentially forgetting first-order filter, making it structurally aligned with the correlation decay of $1/f$ noise.

This structural alignment prompts two related research questions:

1. To what extent does a fractional-order estimator produce more accurate noise power spectral density estimates than standard MCRA in stationary $1/f$ environments.

2. To what extent does this alignment extend to non-stationary $1/f$ conditions, where the long-range dependence structure changes over time and the estimator must track a shifting noise spectrum. The hypothesis is that the power-law memory of the GL integrator will reduce estimation error and improve speech enhancement quality under $1/f$ noise conditions, while producing no meaningful difference from standard MCRA under white Gaussian noise, where no long-range dependence exists and the exponential memory of the first-order filter is already adequate.

To investigate these questions, the proposed estimator is evaluated using the NOIZEUS corpus [26], which contains 30 IEEE sentences spoken by male and female speakers. Clean utterances are corrupted by additive white Gaussian noise, stationary $1/f$ noise, and non-stationary $1/f$ noise at SNR levels of 0, 5, 10, and 15 dB — to adhere to the same values used in the studies from Section 3.3. Pre-corrupted recordings with train and babble noise from the corpus are included to evaluate performance under real-world conditions. The fractional order is swept through $r \in \{0.1, 0.2, \ldots, 0.9\}$ to characterise how the decay of long-range memory affects performance. Once the noise PSD is obtained, noisy speech will be processed for speech enhancement. Full details of the experimental design are given in Chapter 5.

In accordance to previous studies, performance is assessed using segmental SNR improvement [23], and perceptual evaluation of speech quality (PESQ) [22]. Objective measures are supplemented by subjective listening tests conducted with 17 participants, who rate the perceptual quality of enhanced speech on a five-point Mean Opinion Score (MOS) scale.

Chapter 5

Methodology

This chapter describes the experimental setup used to evaluate the GL modification. The speech corpus and noise conditions are described first, followed by the signal mixing procedure. The proposed estimator is then derived in detail, including the speech-presence gating mechanism and the Grünwald-Letnikov weighted sum update rule. The chapter concludes with a description of the three estimator configurations evaluated, the enhancement pipeline, and the evaluation metrics and procedures.

5.1 Signal Preparation

The speech material used in all experiments is drawn from the NOIZEUS corpus [26], which contains 30 IEEE sentences spoken by male and female speakers, sampled at 8 kHz. The files were originally recorded at 25 kHz, but were downsampled to 8 kHz to reduce computational load while maintaining full clarity of speech. Clean utterances are corrupted by additive noise at segmental SNR levels of 0, 5, 10, and 15 dB, spanning conditions from severely degraded to mildly noisy.

5.2 Noise Types

Five noise types are used, chosen to evaluate the new estimation method under conditions of increasing statistical complexity.

5.2.1 White Gaussian Noise

Additive white Gaussian noise (AWGN) serves as the baseline condition. As discussed in Section 3.2, wGn is memoryless and strictly stationary, with a flat power spectral density. Under this condition, the exponential memory of standard MCRA is already well matched to the noise statistics, and the proposed fractional-order modification is not expected to

offer improvement. This condition therefore establishes whether the GL modification degrades performance where it confers no theoretical advantage.

5.2.2 Stationary $1/f$ Noise

Stationary $1/f$ noise is generated as fractional Gaussian noise (fGn) using the Davies-Harte algorithm, implemented via circulant embedding. The auto-covariance sequence is given by:

$$r(k) = \frac{1}{2}\left(|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H}\right) \tag{5.1}$$

where H is the Hurst exponent. A value of $H > 0.5$ produces long-range dependence, with the power spectral density behaving as $S(f) \sim f^{1-2H}$, giving the characteristic $1/f^{\alpha}$ spectral shape. Three different batches of noise are generated, with Hurst exponents of $H = 0.6$, $H = 0.75$, and $H = 0.9$. The generated sequences are normalised to unit variance prior to mixing, so that the SNR is controlled independently of H.

5.2.3 Non-Stationary $1/f$ Noise

Non-stationary $1/f$ noise is constructed by splicing two independent fGn sequences with different Hurst exponents $H_1 = 0.6$ and $H_2 = 0.9$ at the temporal midpoint of each utterance. The splice introduces an abrupt change in the LRD structure mid-utterance, challenging the estimator's ability to track a shifting noise spectrum. Each segment is generated independently using the same Davies-Harte procedure described above and normalised to unit variance before concatenation, ensuring that the splice introduces a statistical change without an accompanying energy discontinuity.

5.2.4 Train and Babble Noise

Train and babble noise are taken directly from the pre-corrupted recordings available in the NOIZEUS corpus, which provides noisy utterances at the target SNR levels. Train noise is selected for its periodic spectral-temporal structure, which presents a distinctive tracking challenge for the estimator. Babble noise is selected for its spectral similarity to speech, which challenges the accuracy of the speech presence probability estimate within MCRA.

5.3 Signal Mixing

For the synthetically generated noise (wGn, stationary fGn, and non-stationary fGn), the noisy signal is formed by scaling the signal to achieve the desired global SNR before addition to the clean speech signal:

$$y(n) = x(n) + \sigma_d \cdot d(n) \tag{5.2}$$

where $x(n)$ is the clean speech signal, $d(n)$ is the noise signal, and σ_d is the scaling factor chosen such that the global SNR equals the target level.

5.4 The Proposed Estimator

The proposed GL-MCRA estimator introduces two modifications to standard MCRA. First, the observed power spectrum is replaced by a speech-presence-gated power $\tilde{P}(k, \ell)$, which suppresses speech energy from entering the noise estimate. Second, the first-order recursive averaging in the noise estimate update is replaced by a Grünwald-Letnikov fractional-order weighted sum. All other components of MCRA — the STFT analysis, frequency and time smoothing of the local energy, minima tracking, speech presence indicator, and speech presence probability — are retained unchanged from Cohen and Berdugo [1]. GL-MCRA modifies standard MCRA rather than IMCRA because the GL substitution targets the first-order recursive averaging update directly. IMCRA's two-pass smoothing and bias compensation structure [2] would introduce additional variables alongside the fractional-order modification, making it harder to attribute observed differences to the GL memory structure alone. Extending the GL modification to IMCRA remains a direction for future work.

5.4.1 Speech-Presence Gating

In standard MCRA, the noise estimate update is modulated by the time-varying smoothing parameter $\tilde{\alpha}_d(k, \ell)$, which slows the update rate when speech is present. In GL-MCRA, the update is instead controlled by gating the observed power directly. The speech presence indicator $I(k, \ell)$ and speech presence probability $p'(k, \ell)$ are computed identically to standard MCRA:

$$I(k, \ell) = \begin{cases} 1 & \text{if } S_r(k, \ell) > \delta \\ 0 & \text{otherwise} \end{cases} \tag{5.3}$$

$$\hat{p}'(k, \ell) = \alpha_p \hat{p}'(k, \ell - 1) + (1 - \alpha_p) I(k, \ell) \tag{5.4}$$

The speech presence probability $p'(k, \ell)$ is then used to gate the observed power spectrum before it enters the GL memory buffer:

$$\tilde{P}(k, \ell) = (1 - p'(k, \ell)) \cdot |Y(k, \ell)|^2 + p'(k, \ell) \cdot \hat{\lambda}_d(k, \ell - 1) \tag{5.5}$$

When speech is absent, p' is low and $\tilde{P}$ reflects the observed power. When speech is present, p' is high and $\tilde{P}$ is replaced by the previous noise estimate, preventing speech energy from entering the memory buffer and corrupting the estimate.

5.4.2 Grünwald-Letnikov Noise Estimate Update

The noise estimate is computed as a normalised power-law weighted sum of past gated power observations:

$$\hat{\lambda}_d(k,\ell) = \sum_{j=0}^{L_{mem}-1} \bar{w}_j^{(r)} \tilde{P}(k,\ell-j) \tag{5.6}$$

where the weights $\bar{w}_j^{(r)}$ are the normalised Grünwald-Letnikov weights:

$$\bar{w}_j^{(r)} = \frac{w_j^{(r)}}{\sum_{j=0}^{L_{mem}-1} |w_j^{(r)}|} \tag{5.7}$$

and $w_j^{(r)}$ are computed recursively by Equation 3.34. Normalisation ensures the weighted sum is a proper average on the same scale as the observed power, making the GL estimate directly comparable to the standard MCRA estimate. The memory length L_{mem} truncates the otherwise infinite GL sum according to the short-memory principle [25], and is set to $L_{mem} = 150$ frames — approximately 2.4 seconds at 8 kHz sampling rate with 50% frame overlap. This value is determined by the utterance length of the NOIZEUS corpus rather than by the GL weight decay criterion alone, since for $r = 0.1$ the weights decay sufficiently slowly that the point at which they fall below 10^{-3} of $\bar{w}_0^{(r)}$ occurs at $j = 63$ frames — approximately one second of signal history. Setting $L_{mem} = 150$ therefore ensures the full decay is captured within the available utterance length, while the weight at the boundary has decayed to 3.81×10^{-4} of $\bar{w}_0^{(r)}$, confirming that truncation error is negligible across all tested values of r.

These two substitutions — replacing $|Y(k,\ell)|^2$ with $\tilde{P}(k,\ell)$, and replacing Equation 3.8 with Equation 5.6 — are the complete modification. The fractional order r controls the rate of power-law decay of the weights: higher r produces faster decay and shorter effective memory; lower r produces slower decay and longer memory, retaining a greater influence of distant past observations.

5.5 Experimental Design

The experiment is designed to isolate the contribution of the fractional-order memory structure from other potential sources of improvement.

5.5.1 Estimator Configurations

Three estimator configurations are evaluated to isolate the contribution of each component of the proposed modification:

- **Standard MCRA**: the baseline estimator of Cohen and Berdugo [1] as described in Section 3.3, with no modification
- **Gated MCRA**: standard MCRA with the observed power replaced by the speech-presence-gated power $\tilde{P}(k, \ell)$, isolating the effect of the gating mechanism independently of the GL memory
- **GL-MCRA**: the proposed estimator, combining speech-presence gating with the Grünwald-Letnikov fractional-order memory structure

The fractional order is swept over $r \in \{0.1, 0.2, \ldots, 0.9\}$ for GL-MCRA. Standard MCRA and gated MCRA serve as fixed baselines independent of r. All shared MCRA parameters are fixed at the values recommended by Cohen and Berdugo: $\alpha_d = 0.85$, $\alpha_s = 0.9$, $\alpha_p = 0.2$, $\delta = 5$, and $L = 75$ frames (approximately one second at 8 kHz with 50% overlap), consistent with the window length of 0.5–1.5 seconds recommended by Cohen and Berdugo [1].

5.5.2 Enhancement and Evaluation Pipeline

All 30 utterances from the NOIZEUS corpus [26] are corrupted by each noise type at each SNR level as described in Section 5.1. Each corrupted utterance is processed by all three estimator configurations. The resulting noise PSD estimates are used by a modified Wiener filter, producing enhanced speech signals.

Performance is evaluated using segmental SNR improvement, and PESQ score, computed for each utterance, noise condition, SNR level, and value of r, and averaged across 30 utterances. Segmental SNR improvement is the difference between the SegSNR of the enhanced signal and the SegSNR of the noisy signal (Equation 3.28), where a positive value indicates improvement over the unprocessed input. PESQ is computed using the `pysepm` library [27], which uses the Hu and Loizou implementation [28], returning a predicted mean opinion score on a scale of 1.0 to 4.5 [22, 24].

Subjective evaluation is conducted using a mean opinion score listening test with 17 participants, who rated enhanced clips on a five-point scale from 1 to 5 for each noise condition at 0 dB and 10 dB SNR. Clips were presented with neutral labels to prevent participants from identifying the estimator configuration. GL-MCRA clips use $r = 0.9$, the best-performing fractional order as determined by the objective evaluation.

Results are reported at two levels of aggregation. Aggregate results, averaged across all noise conditions and SNR levels, characterise how GL-MCRA performance varies with r relative to the baselines. Per-condition results test the central hypothesis: that the improvement of GL-MCRA over standard MCRA is greater for noise conditions with stronger long-range dependence. Statistical significance is assessed using paired t-tests across utterances with threshold $p < 0.05$.

Chapter 6

Results

This chapter presents the objective and subjective evaluation results for all three estimator configurations under all seven conditions and SNR levels. Results are reported first as aggregate segmental SNR improvement averaged across all conditions, then broken down per noise condition and per SNR level. PESQ scores and mean opinion scores from the listening test are presented subsequently. Statistical significance of differences between GL-MCRA and standard MCRA is assessed throughout using paired t-tests.

6.1 Aggregate SegSNR Improvement

Figure 6.1 shows the mean segmental SNR improvement — obtained by subtracting the SegSNR of the noisy signal from the SegSNR of the enhanced signal — of all three estimator configurations, averaged across all noise conditions and SNR levels, as a function of the fractional order r. Standard MCRA and gated MCRA appear as horizontal lines at 2.65 dB and 2.51 dB respectively, since neither depends on r. GL-MCRA increases at every step across the r range, from 1.88 dB at $r = 0.1$ to 3.02 dB at $r = 0.9$ (an improvement of 30%), crossing gated MCRA at approximately $r = 0.5$ and standard MCRA at approximately $r = 0.6$.

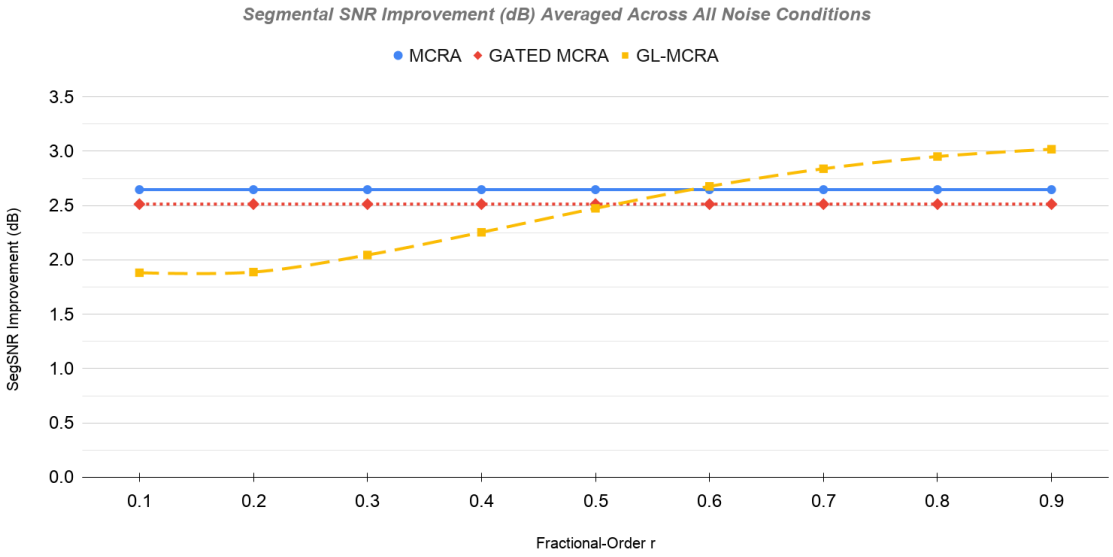


Figure 6.1: Segmental SNR Improvement Per Estimator, averaged across all noise conditions, as a function of fractional-order r

Immediately apparent is how GL-MCRA outperforms both baselines for $r \geq 0.6$, with a mean improvement of 0.37 dB (8.9%) over standard MCRA, and 0.51 dB (12.5%) for gated MCRA at $r = 0.9$. Interestingly, gated MCRA performs consistently below standard MCRA, with a mean deficit of 0.13 dB. At low values of r, GL-MCRA performs below both baselines, reaching a minimum of 1.88 dB at $r = 0.1$.

6.2 Per-Condition SegSNR Improvement

Figure 6.2 shows the mean segmental SNR improvement of MCRA, Gated MCRA, and GL-MCRA as a function of r, reported separately for each noise condition. GL-MCRA improves monotonically with r across all conditions, crossing the standard MCRA baseline at different values of r depending on the condition. Gated MCRA falls below standard MCRA in every condition, consistent with the aggregate results. The relative improvement of GL-MCRA over standard MCRA at $r = 0.9$ is reported in Figure 6.3 and discussed per condition below.

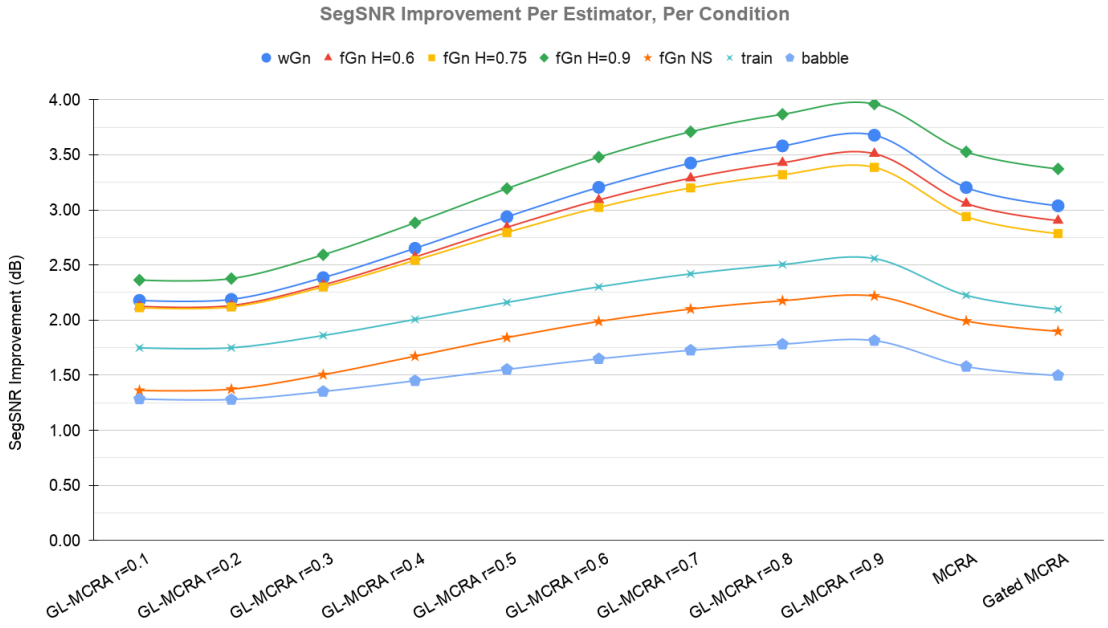


Figure 6.2: SegSNR Improvement Per GL-MCRA per each value of r, MCRA and Gated MCRA, Per Condition

6.2.1 Stationary $1/f$ Noise

For all stationary fGn conditions, GL-MCRA improves with r and outperforms standard MCRA for $r \geq 0.6$. At $r = 0.9$, the mean improvement reaches 3.51 dB for $H = 0.6$, 3.39 dB for $H = 0.75$, and 3.96 dB for $H = 0.9$.

However, the improvement does not increase with the Hurst parameter as expected. $H = 0.6$ consistently outperforms $H = 0.75$ across all values of r, which is contrary to the prediction that stronger LRD should produce greater benefit. Standard MCRA shows the same result, achieving 3.06 dB for $H = 0.6$ but 2.94 dB for $H = 0.75$.

Figures 6.4(a) through (d) show the spectrograms of the clean reference, noisy speech, MCRA-enhanced, and GL-MCRA-enhanced signals for utterance sp15 under fGn H=0.9 noise at 10 dB SNR. The noisy spectrogram shows the characteristic low-frequency concentration of energy from the $1/f$ noise floor obscuring the speech. Both estimators reduce the noise floor visibly relative to the noisy input, but GL-MCRA produces a cleaner residual between speech segments, consistent with the SegSNR improvement of 0.43 dB reported in Figure 6.3. The harmonic structure of the speech visible in the clean reference is preserved in both enhanced signals, indicating that neither estimator introduces significant speech distortion at this SNR level.

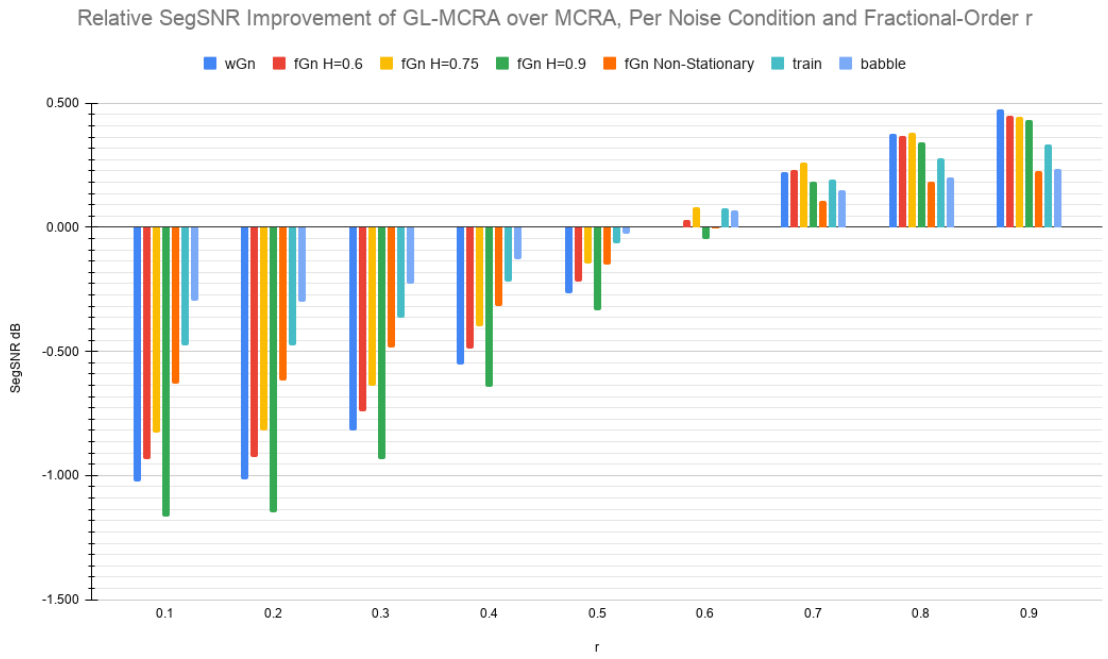


Figure 6.3: Relative SegSNR improvement of GL-MCRA over standard MCRA (dB), per noise condition and fractional order r. Positive values indicate GL-MCRA outperforms standard MCRA.

6.2.2 White Gaussian Noise

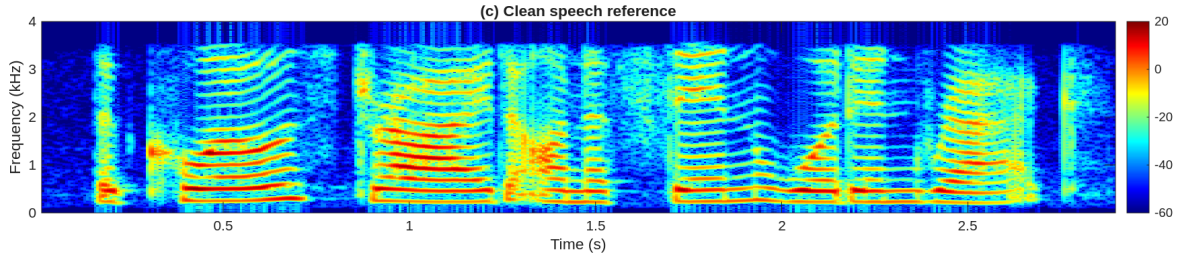
Under wGn, GL-MCRA at $r = 0.9$ reaches a segmental SNR improvement of 3.68 dB over the noisy signal — comparable to the fGn conditions and substantially above standard MCRA (3.20 dB, +11.5%). This is the second largest absolute improvement of any condition, and is notable given that wGn has no LRD.

6.2.3 Non-Stationary $1/f$ Noise

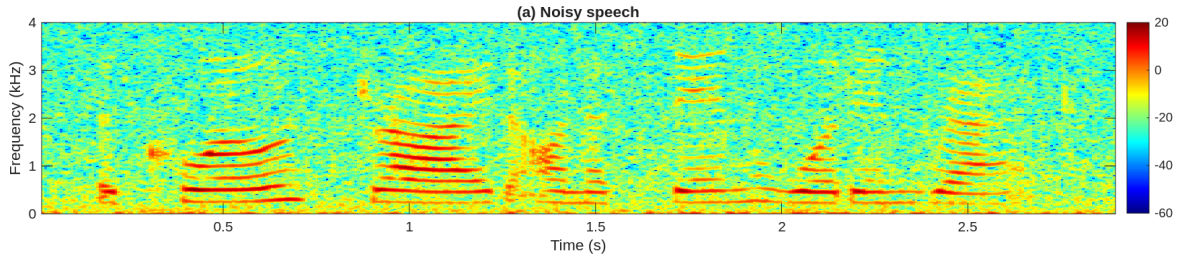
The non-stationary fGn condition, consisting of spliced fGn segments with $H_1 = 0.6$ and $H_2 = 0.9$, concatenated at the utterance midpoint, shows the lowest improvement among the $1/f$ conditions. At $r = 0.9$, the mean improvement for GL-MCRA reaches 2.22 dB — only a slight improvement above standard MCRA's 1.99 dB (+5.4%). This is the smallest improvement among all conditions alongside babble. Performance rises steadily with r.

6.2.4 Train Noise

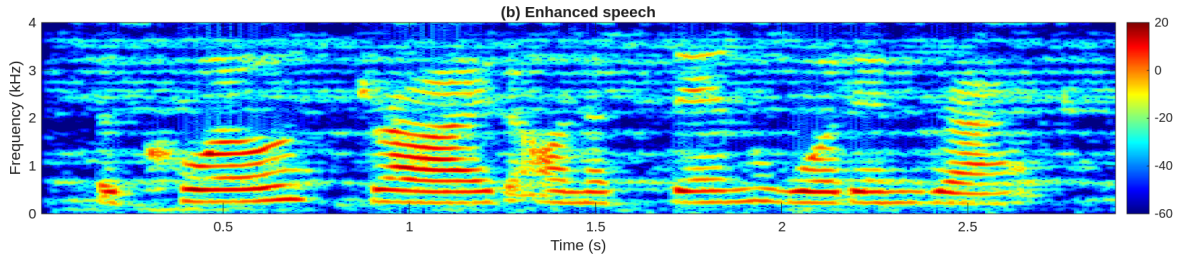
Train noise yields 2.56 dB at $r = 0.9$ — an improvement of 0.33 dB (7.9%) over MCRA. GL-MCRA crosses the MCRA baseline at approximately $r = 0.5$, earlier than most other



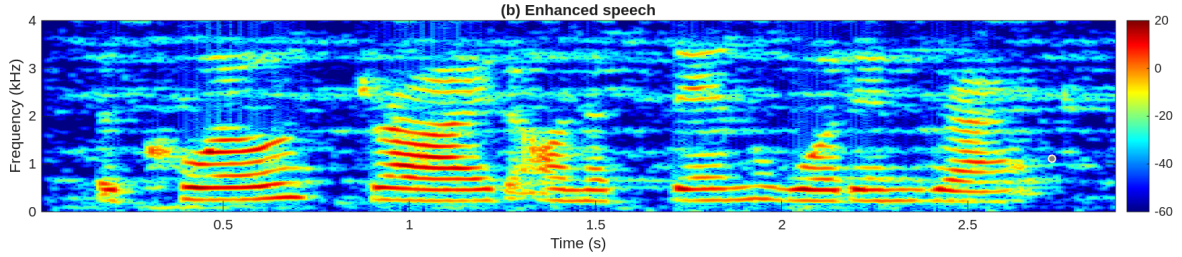
(a) Clean Reference



(b) Noisy Speech (10 dB SNR)



(c) MCRA enhanced



(d) GL-MCRA enhanced

Figure 6.4: Spectrograms of clean, noisy, MCRA-enhanced, and GL-MCRA-enhanced speech for utterance sp15 under fGn H=0.9 noise at 10 dB SNR.

conditions. The improvement is moderate relative to the stationary fGn conditions but consistent across all values of r above the crossover point.

6.2.5 Babble Noise

Babble noise is the most challenging condition for all estimators, with the lowest absolute SegSNR improvement across all configurations. GL-MCRA reaches 1.81 dB at $r = 0.9$, compared to MCRA's 1.58 dB (+5.4%).

6.2.6 Summary

GL-MCRA outperforms standard MCRA at $r = 0.9$ across all seven noise conditions, with improvements ranging from 0.23 dB (5.4%) for non-stationary fGn and babble, to wGn's 0.48 dB (11.7%). Gated MCRA falls below standard MCRA in every condition.

These results have concrete implications for real-world speech enhancement applications. The strong performance under stationary $1/f$ suggests that GL-MCRA would be particularly beneficial in environments where background noise is stable and exhibits long-range dependence, for example:

- the low-frequency hum of air conditioning or ventilation systems in office environments
- the steady rumble of a train carriage interior during a journey

In these settings, the power-law memory of GL-MCRA is better matched to the structure of the noise than the exponential memory of standard MCRA, and the improvement in noise estimation accuracy would translate to cleaner speech in hearing aids, mobiles, or conferencing systems.

The improvement under train noise (+0.33 dB, +7.9%) and the strong subjective improvement at 10 dB SNR (+0.76 MOS, $p < 0.001$) suggest that GL-MCRA could be practically useful in transit environments such as train stations, bus interiors, or airport lounges, where noise has a structured and quasi-periodic spectral character. A person making a phone call on a train, or a hearing aid user walking through a busy station, would likely benefit from the improved noise tracking that GL-MCRA provides in these conditions.

The smallest improvements are observed for babble noise (+0.23 dB, 5.4%) and non-stationary fGn (+0.23 dB, +5.4%), which represent the hardest practical cases — a crowded party or a restaurant where the noise character changes rapidly as different speakers dominate. In these environments, the spectral similarity between babble and speech makes it difficult for any MCRA-based estimator to reliably distinguish noise from speech, and the abrupt changes in noise statistics limit the benefit of long-memory estimators. Nevertheless, GL-MCRA still outperforms standard MCRA even in these challenging conditions, suggesting it is a better choice than standard MCRA across all tested acoustic environments.

6.2.7 Statistical Significance

Paired t-tests were conducted between the SegSNR improvement of GL-MCRA at $r = 0.9$ and standard MCRA across all 30 utterances and four SNR levels for each noise condition ($n = 120$ pairs per condition). All per-condition tests returned $p < 0.001$, confirming that the SegSNR improvements of GL-MCRA over standard MCRA are statistically significant across all seven noise conditions.

Hedge's g effect sizes are reported in Figure 6.5. All values exceed the large effect threshold of $g = 0.8$ [29], ranging from 1.18 (babble at 15 dB) to 3.62 (fGn $H = 0.6$ at 5 dB). Effect sizes decrease consistently with increasing SNR across all conditions, indicating the diminishing benefit of improved noise estimation as noise becomes less dominant at higher input SNR levels.

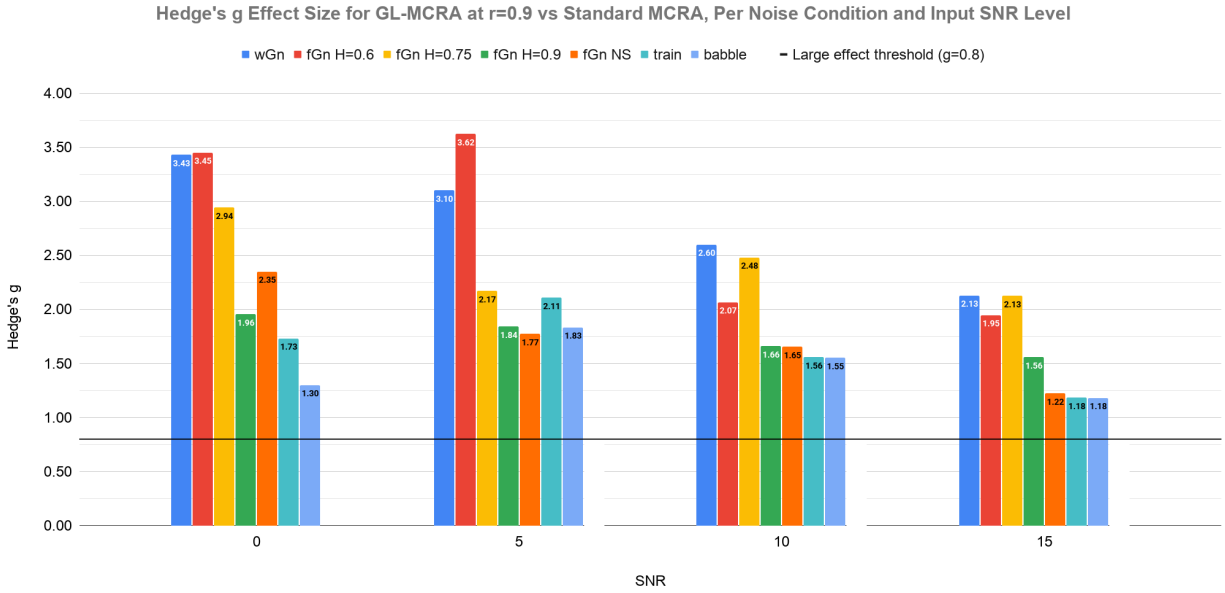


Figure 6.5: Hedge's g effect size for GL-MCRA at $r = 0.9$ vs Standard MCRA, per noise condition and input SNR level ($n = 30$ pairs per cell, $p < 0.001$ for all entries).

6.3 Perceptual Evaluation of Speech Quality

6.3.1 Aggregate PESQ

Figure 6.6 shows the mean PESQ score of all three estimators, averaged across all noise conditions and SNR levels as a function of r. Standard MCRA and gated MCRA each have the same values at every r as they do not use r. GL-MCRA increases monotonically from 2.205 at $r = 0.1$ to 2.399 at $r = 0.9$, crossing gated MCRA at approximately $r = 0.7$ and standard MCRA at $r = 0.8$.

At $r = 0.9$, GL-MCRA achieves a mean PESQ score of 2.399 — an improvement of 0.014 (+0.6%) PESQ points over standard MCRA. The PESQ crossover points occur later than the SegSNR crossover points.

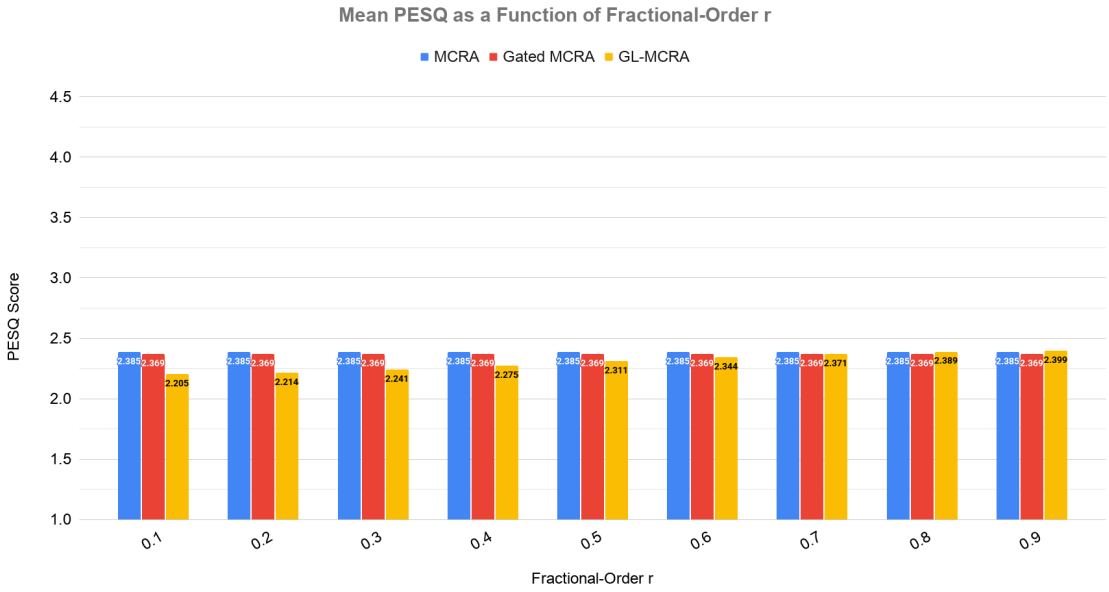


Figure 6.6: Mean PESQ scores averaged across all noise conditions and SNR levels, as a function of fractional order r

6.3.2 Per-Condition PESQ

The mean PESQ scores for all three estimators (with GL-MCRA at $r = 0.9$), separated by noise condition is shown in Figure 6.7.

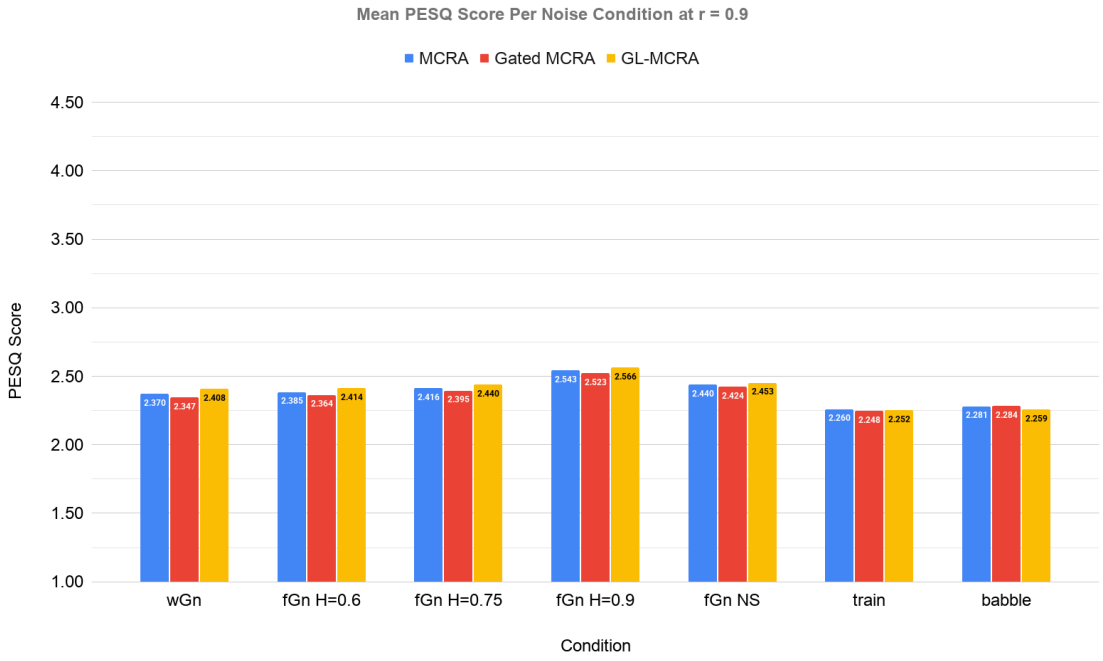


Figure 6.7: Mean PESQ scores per noise condition at $r = 0.9$. Higher is better; scale is 1.0–4.5.

GL-MCRA outperforms standard MCRA in five of the seven conditions at $r = 0.9$ (+1.6%, +1.2%, +1.0%, +0.9%, +0.5%). In the train and babble conditions, GL-MCRA scores marginally below standard MCRA (-0.4%, -1.0%).

6.3.3 PESQ vs SNR Level

Figure 6.8 shows mean PESQ scores across all conditions at each SNR level for $r = 0.9$.

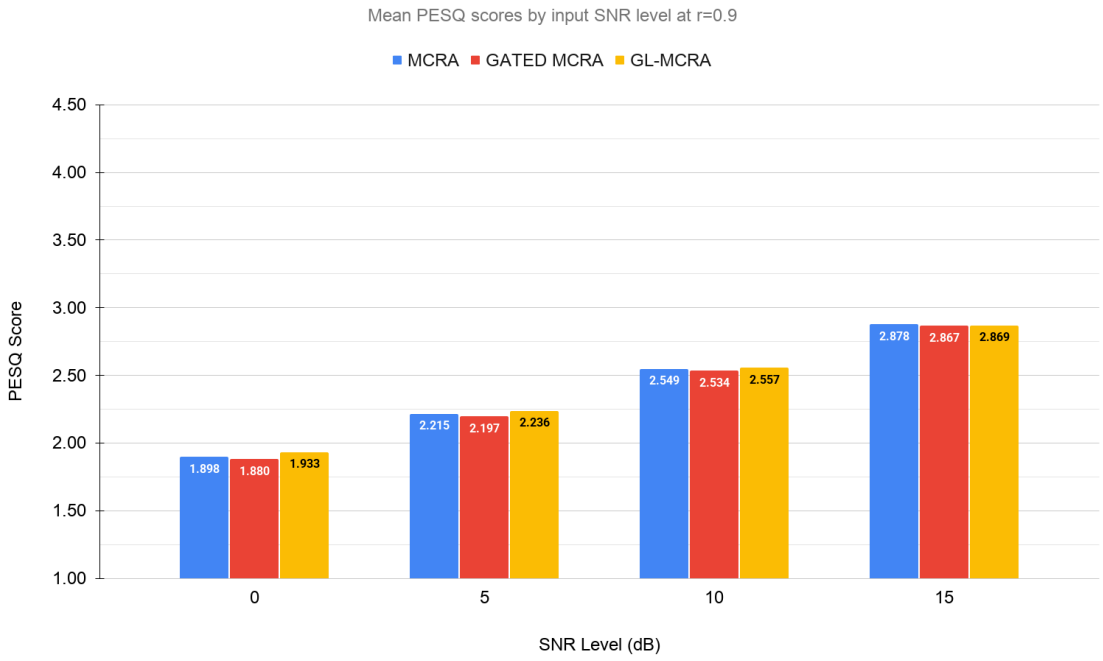


Figure 6.8: Mean PESQ scores by input SNR level at $r = 0.9$, averaged across all noise conditions.

GL-MCRA outperforms standard MCRA at 0, 5, and 10 db SNR, with the largest advantage at 0 dB. At 15 dB SNR, GL-MCRA falls slightly below standard MCRA.

6.4 Mean Opinion Score

Seventeen participants rated clips on a five-point scale from 1 (bad) to 5 (excellent). Clips were presented without descriptive labels at 0 dB and 10 dB SNR for each noise condition. Shapiro-Wilk tests on the paired differences rejected normality in all 14 condition–SNR combinations ($p < 0.05$), which is expected given the ordinal and bounded nature of MOS ratings. Wilcoxon signed-rank tests were therefore used to assess statistical significance of difference between GL-MCRA and standard MCRA.

Tables 6.1 and 6.2 show the mean MOS ratings and standard deviations for GL-MCRA and standard MCRA at 0 dB and 10 dB SNR respectively.

At 0 dB SNR, GL-MCRA is rated significantly higher than standard MCRA for wGn ($+0.24$, $p = 0.046$ and fGn $H = 0.75$ ($+0.41$, $p = 0.020$) and significantly lower for train noise (-0.24, $p = 0.046$). At 10 dB SNR, GL-MCRA is rated significantly higher for fGn $H = 0.9$ ($+0.41$, $p = 0.020$) and train noise ($+0.76$, $p < 0.001$), and significantly lower for wGn (-0.59, $p = 0.008$), fGn $H = 0.75$ (-0.59, $p = 0.020$), and fGn NS (-0.71, $p = 0.005$).

Table 6.1: Mean MOS ratings ($\pm$ Standard Deviation) at 0 dB SNR, $n = 17$ participants. GL-MCRA at $r = 0.9$. Δ denotes GL-MCRA minus MCRA; $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$ (Wilcoxon signed-rank test).

Condition	Noisy	MCRA	GL-MCRA	Δ
wGn	1.18 ± 0.39	1.71 ± 0.92	1.94 ± 0.97	$+0.24^{*}$
fGn H=0.6	1.47 ± 0.72	1.76 ± 0.83	1.59 ± 0.80	-0.18
fGn H=0.75	1.12 ± 0.33	1.41 ± 0.71	1.82 ± 0.73	$+0.41^{*}$
fGn H=0.9	1.65 ± 0.70	2.41 ± 0.94	2.29 ± 0.69	-0.12
fGn NS	1.88 ± 0.99	1.65 ± 0.70	1.94 ± 0.75	$+0.29$
Train	1.41 ± 0.51	1.65 ± 0.86	1.41 ± 0.87	-0.24^{*}
Babble	1.24 ± 0.44	1.35 ± 0.49	1.47 ± 0.51	$+0.12$

Table 6.2: Mean MOS ratings ($\pm$ Standard Deviation) at 10 dB SNR, $n = 17$ participants. GL-MCRA at $r = 0.9$. Δ denotes GL-MCRA minus MCRA; $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$ (Wilcoxon signed-rank test).

Condition	Noisy	MCRA	GL-MCRA	Δ
wGn	3.06 ± 0.90	3.59 ± 0.71	3.00 ± 0.79	-0.59^{**}
fGn H=0.6	2.53 ± 0.87	3.82 ± 0.88	4.06 ± 0.83	$+0.24$
fGn H=0.75	2.82 ± 1.07	3.53 ± 1.01	2.94 ± 0.74	-0.59^{*}
fGn H=0.9	3.00 ± 0.70	3.65 ± 1.06	4.06 ± 1.10	$+0.41^{*}$
fGn NS	3.12 ± 0.99	3.82 ± 1.24	3.12 ± 1.17	-0.71^{**}
Train	2.88 ± 1.05	2.35 ± 0.86	3.12 ± 1.05	$+0.76^{***}$
Babble	2.47 ± 0.87	3.29 ± 1.16	3.18 ± 1.13	-0.12

Chapter 7

Discussion

This chapter interprets the results presented in Chapter 6, examining the effect of the fractional order r on performance, the relative contribution of gating and fractional memory, and the per-condition behaviour of GL-MCRA across stationary, non-stationary, and real-world noise conditions. The divergence between objective and subjective metrics is discussed, followed by the limitations of GL-MCRA and the evaluation, and directions for future work.

7.1 Effect of Fractional Order on Performance

The monotonic improvement with r (Figure 6.1) suggests that within the tested range, shorter power-law memory is consistently more beneficial than longer memory when averaged across all conditions. This is in line with the weight structure at low r: the weights decay very slowly, giving nearly equal weight to observations far in the past, which introduces a long lag in the estimate's response to changing noise conditions. The PESQ crossover points appear later than the SegSNR crossover points, suggesting that objective noise estimation does not translate proportionally to perceptual quality improvement.

The per-condition analysis in Section 6.2 reveals three patterns:

1. The largest improvements are consistently observed under stationary conditions, where the noise statistics are stable over time and the GL weighted sum can accumulate a history of past observations.

2. The smallest improvements occur under non-stationary fGn and babble, where the quality of the memory buffer is possibly compromised by stale observations following a statistical change, or by unreliable speech presence estimation.

3. The relationship between LRD strength and GL-MCRA benefit is partially supported: fGn $H = 0.9$ yields the largest absolute improvement, but the relative improvement over the per-condition MCRA baseline is approximately equal across all three fGn

conditions (0.44 dB, 10.7%), suggesting that GL-MCRA provides a consistent proportional benefit regardless of the specific Hurst exponent.

As mentioned in Section 6.2.7, the Hedge's g effect sizes confirm the improvement of GL-MCRA over standard MCRA. All values exceed the large effect threshold of $g = 0.8$, which indicates that the distribution of GL-MCRA and standard MCRA SegSNR improvements are well separated at the utterance level. In practice, this means GL-MCRA reliably outperforms standard MCRA on individual utterances rather than only on average — reinforcing the statistical significance.

Effect sizes decrease consistently with increasing SNR across all conditions, from a mean of $g = 2.52$ at 0 dB to $g = 1.61$ at 15 dB. This reflects the pattern seen in the mean SegSNR improvements and confirms that the benefit of GL-MCRA is greatest where accurate noise estimation matters most, i.e. heavily degraded conditions where speech and noise are closest in energy.

The per-SNR breakdown also sheds light on the ordering anomaly between fGn $H = 0.6$ and fGn $H = 0.75$. The non-monotonic ordering — with $H = 0.75$ performing below both $H = 0.6$ and $H = 0.9$ — does not have a straightforward explanation. Importantly, the same ordering is present in the standard MCRA baseline, which achieves 3.059 dB for $H = 0.6$ but only 2.939 dB for $H = 0.75$, despite $H = 0.75$ representing stronger LRD. Since standard MCRA applies the same algorithm and parameters to all conditions, this confirms that the anomaly is a property of the condition itself rather than of the GL modification. The spectral slope of fGn at $H = 0.75$ ($1 - 2H = -0.5$) lies between that of $H = 0.6$ (-0.2) and $H = 0.9$ (-0.8), and it is possible that this intermediate spectral shape is particularly challenging for MCRA's minima tracking — neither flat enough to be well approximated by a stationary estimate, nor strongly enough coloured for the power-law memory to exploit its correlation structure effectively. Resolving this would require a denser sweep of Hurst exponents to characterise the relationship between H and estimator difficulty more precisely.

7.2 Contribution of Gating vs Fractional Memory

Gated MCRA performs consistently below standard MCRA (Figure 6.1, Figure 6.3), which indicates that replacing the observed power with the gated power $\tilde{P}$ was detrimental to performance. This is initially counter-intuitive, as gating is designed to prevent speech energy from corrupting the noise estimate.

The explanation lies within the soft modulation of the update rate via $\tilde{\alpha}_d$, which in standard MCRA already provides protection against speech contamination, and the additional hard gating of the input disrupts this balance. When speech is present, p' is high and $\tilde{\alpha}_d$ approaches 1, and the noise estimate updates very slowly. The modulation is continuous and proportional to the degree of speech presence. Gated MCRA adds a second layer of suppression on top of this: $\tilde{P}$ replaces the observed power with the previous noise

estimate when p' is high, and afterwards this value is smoothed by $\tilde{\alpha}_d$. The result is double suppression during speech frames as the estimate is prevented from updating both through the input and through the smoothing parameter simultaneously.

The improvement of GL-MCRA over standard MCRA at high r therefore cannot be attributed to gating alone, and must come from the interaction between the gated power input and the power-law memory structure of the GL weighted sum. In GL-MCRA, the gating controls the contents of the memory buffer rather than suppressing the update rate of the smoother. By replacing speech-contaminated observations with recycled noise estimates before they enter the buffer, the gating ensures that the power-law weighted sum operates on a cleaner history, without introducing the double suppression effect that degrades gated MCRA.

7.3 Performance Under Stationary Noise

7.3.1 Stationary $1/f$ Noise

The ordering of $H = 0.6$ and $H = 0.9$ (Figure 6.3) confirms the hypothesis: stronger long-range dependence produces greater benefit from the GL memory structure. The condition with the highest Hurst exponent, $H = 0.9$, produced the largest improvement across all values of r, followed surprisingly by $H = 0.6$ and $H = 0.75$. As r increases, the gap between conditions widens, suggesting that the fractional memory is increasingly effective at exploiting the long-range correlation structure of the noise as the order approaches 0.9.

Notably, Figure 6.3 shows that at $r = 0.6$ — the approximate crossover point in the aggregate results — GL-MCRA in the fGn $H = 0.9$ and the non-stationary conditions does not improve upon the MCRA baseline (underperforming by -0.048 and -0.003 dB respectively), whereas the improvement for $H = 0.6$ and $H = 0.75$ has already crossed into the positive at $r = 0.6$. This could be related to the high MCRA baseline for fGn $H = 0.9$, suggesting that despite the better match between power-law memory and strong LRD, GL-MCRA might need a higher fractional order before it can surpass standard MCRA in those noise conditions.

7.3.2 White Gaussian Noise

The strong performance of GL-MCRA under wGn (Figure 6.3) is unexpected, as wGn has no long-range dependence and the GL power-law memory should not offer theoretical advantage over exponential memory. One possible explanation is that the combination of the gating mechanism and the power-law weighted sum produces a more stable estimate, as the weights decay slowly, resulting in less sensitivity to individual noisy frames. This finding suggests that the benefit of GL-MCRA is not exclusively attributable to its match to $1/f$ statistics, and that the power-law memory in combination with the gating, may confer a more general advantage.

7.4 Performance Under Non-Stationary and Real-World Noise

7.4.1 Non-Stationary $1/f$ Noise

The non-stationary fGn condition shows the smallest improvement among the $1/f$ conditions (Figure 6.3), showing that an abrupt change in the LRD structure limits the benefit of the GL memory: the power-law weighted sum retains observations from H_1 after the instant transition to H_2, whose statistics no longer reflect the current noise, slowing adaptation. Once again, performance climbs as r increases, suggesting that faster decay is consistently beneficial, allowing the pre-splice observations to dissipate more quickly.

7.4.2 Train Noise

GL-MCRA shows moderate benefit from the fractional memory under the train noise condition. Train noise has a periodic spectral-temporal structure that is more predictable than babble, but less stationary than the synthetic fGn conditions, which may account for its intermediate level of SegSNR improvement. The largest MOS improvement of any condition at 10 dB ($+0.76$, $p < 0.001$), shown on Table 6.2, suggests that the fractional memory structure is particularly well suited to the periodic structure of train noise at moderate SNR levels, despite the modest SegSNR improvement.

7.4.3 Babble Noise

The spectral similarity between babble and speech is one possible explanation for the low improvement for this noise condition, and may cause p' to be overestimated, leading to excessive gating and a memory buffer populated by recycled estimates rather than true observations. Despite this, GL-MCRA still outperforms standard MCRA at $r = 0.9$ in SegSNR (Figure 6.3), suggesting that even imperfect gating combined with power-law memory provides a consistent benefit over the first-order recursive smoother. In practice, babble noise represents the cocktail party problem — a crowded restaurant, an open office, or a social gathering where multiple speakers are active simultaneously. The result that GL-MCRA still outperforms standard MCRA even under these difficult conditions despite the spectral confusion between noise and speech, suggests that the power-law memory grants a robust advantage that persists even when the gating mechanism is unreliable.

7.5 Divergence Between Objective and Subjective Metrics

Figure 6.7 shows that GL-MCRA outperforms standard MCRA in five of the seven conditions at $r = 0.9$ in PESQ, with the largest improvements observed for wGn and the stationary fGn conditions, which mirrors the SegSNR findings. Despite showing a positive SegSNR improvement in the train noise condition, GL-MCRA does marginally worse than

standard MCRA in PESQ, hinting towards the noise estimation improvement not translating to a perceptual benefit for this condition. For babble, GL-MCRA falls below both standard MCRA and gated MCRA in PESQ — the only condition where GL-MCRA performs worst of all three estimators. This is surprising, as it does not follow from the SegSNR results, where GL-MCRA outperformed MCRA on babble, and suggests that some speech distortion may be introduced under babble conditions during the enhancement pipeline that is penalised by PESQ but not captured by SegSNR.

GL-MCRA performs better than standard MCRA at 0, 5, and 10 dB SNR, with the largest advantage at 0 dB — the noisiest condition. At high SNR, GL-MCRA falls slightly below standard MCRA (Figure 6.8) in PESQ at 15 dB (2.869 and 2.878 respectively). This could suggest that at high SNR the power-law memory structure offers no advantage, and could be due to the noise being less perceptually dominant.

The MOS results at 10 dB (Table 6.2) reinforce this — GL-MCRA is rated significantly lower than standard MCRA for wGn (-0.59, $p = 0.004$), fGn $H = 0.75$ (-0.59, $p = 0.013$), and fGn NS (-0.71, $p = 0.002$) despite positive SegSNR improvements, suggesting that distortion introduced in the enhancement pipeline becomes more perceptually noticeable when noise is less dominant.

7.6 Limitations of the Proposed Estimator

The primary limitation of GL-MCRA relative to standard MCRA is its higher computational complexity. Standard MCRA updates $\hat{\lambda}_d$ in $O(K)$ per frame. GL-MCRA computes a weighted sum over L_{mem} past frames per bin, giving $O(K \cdot L_{mem})$ per frame. With $K = 129$ frequency bins and $L_{mem} = 150$ frames, this represents a factor of 150 increase in the number of operations required for the noise estimate update, making GL-MCRA unsuitable for real-time deployment on resource-constrained devices in its current form. Optimisations such as reducing L_{mem} for higher values of r, where the weights decay more rapidly, or exploiting the recursive structure of the GL weights, could reduce this overhead.

7.7 Limitations of the Evaluation

The evaluation has several limitations. The primary noise estimation accuracy metric, LogErr, used by Hendriks et al. [20, 21] was not computed due to time constraints, meaning the direct accuracy of $\hat{\lambda}_d$ relative to the true noise PSD cannot be assessed. A Wiener filter was used as the enhancement stage rather than OM-LSA, which is used by Cohen and Berdugo [1] with MCRA, and differences in enhancement stage may contribute to the observed divergence between objective and subjective metrics. The MOS listening test used a single utterance per condition per SNR level, limiting the generalisability of the subjective results to the specific utterances selected.

Standard MCRA was chosen as the baseline estimator rather than more recent methods

such as IMCRA, DFT-SS, or MMSE, because the GL modification directly replaces the first-order recursive averaging update in MCRA. Evaluating against a fundamentally different estimator would conflate the effect of the fractional-order modification with differences in the underlying estimation algorithm. Cohen and Berdugo's [1] recommended parameters allowed the three different estimators to be fixed at those values, ensuring that observed differences are attributable to the GL modification alone. Comparison against IMCRA and the MMSE estimator of Hendriks et al. remains a direction for future work.

7.8 Future Work

The monotonically increasing performance of GL-MCRA across the full tested range of r, with no sign of plateauing at $r = 0.9$, suggests that values of r beyond 0.9 may offer further improvement and warrants investigation. The non-stationary fGn results motivate the development of a dynamically adapted r that responds to detected changes in noise statistics, combining the long memory of low r during stationary periods with the faster adaptation of high r following abrupt transitions. Replacing the Wiener filter with OM-LSA as the enhancement stage would align the evaluation more closely with the original MCRA framework [1]. Computing Log-Err from saved noise PSD estimates would provide a direct measure of estimation accuracy to complement the speech quality metrics used here. Comparison of GL-MCRA against IMCRA and MMSE would situate the proposed method within the broader landscape of VAD-independent noise estimators. Finally, a larger MOS listening test with multiple utterances per condition would strengthen the subjective conclusions.

Chapter 8

Conclusion

This thesis investigated whether replacing the first-order recursive averaging in the MCRA noise estimator with a Grünwald-Letnikov fractional-order weighted sum improves noise power spectral density estimation and speech enhancement performance in $1/f$ acoustic noise environments. The proposed GL-MCRA estimator introduces two modifications to standard MCRA: a speech-presence gating mechanism that prevents speech energy from entering the fractional memory buffer, and a power-law weighted sum over past gated power observations in place of the first-order recursive update.

8.1 Summary of Findings

The objective evaluation demonstrates that GL-MCRA outperforms standard MCRA in segmental SNR improvement across all seven noise conditions and all four SNR levels at $r = 0.9$, with improvements ranging from 0.23 dB (5.4%) for non-stationary fGn and babble noise to 0.48 dB (11.5%) for white Gaussian noise. All per-condition differences are statistically significant ($p < 0.001$, paired t-test, $n = 120$ pairs per condition). The improvement increases monotonically with r across the full tested range, with GL-MCRA crossing the standard MCRA baseline at approximately $r = 0.6$ when averaged across all conditions.

The study of gated MCRA confirms that the improvement cannot be attributed to the gating mechanism alone. Gated MCRA, which introduces the same speech power gating into the standard MCRA algorithm, performs consistently below standard MCRA across all conditions. The benefit of GL-MCRA arises from the interaction between the gated power input and the power-law memory structure of the GL weighted sum.

The hypothesis that GL-MCRA would benefit most under conditions with stronger long-range dependence is partially supported. The fGn condition with $H = 0.9$ yields the largest absolute improvement among the fGn conditions, and the relative improvement over the per-condition MCRA baseline is approximately equal across all three fGn conditions at 0.44 dB (10.7%), suggesting a consistent proportional benefit regardless of

the specific Hurst exponent. The unexpected strong performance under white Gaussian noise suggests that the benefit of GL-MCRA extends beyond its theoretical match to $1/f$ statistics, and that the combination of gating and power-law memory may confer a more general advantage over first-order recursive averaging.

PESQ results are consistent with the SegSNR findings, with GL-MCRA outperforming standard MCRA in five of the seven conditions at $r = 0.9$, though the magnitude of improvement is smaller than in SegSNR. The subjective MOS evaluation with 17 participants reveals a more complex picture, as GL-MCRA is rated significantly higher than standard MCRA in some conditions and significantly lower in others, with the pattern depending on both the noise condition and the SNR level. The most consistent subjective finding is the train noise result at 10 dB, where GL-MCRA achieves the largest MOS improvement of any condition ($+0.76$, $p < 0.001$). The difference between objective and subjective metrics at high SNR suggests that the enhancement pipeline introduces perceptible speech distortion when driven by the GL noise estimate under conditions where noise is less dominant, a distortion that SegSNR does not capture but listeners detect.

8.2 Answer to the Research Question

The research question asked whether replacing the first-order recursive averaging in MCRA with a Grünwald-Letnikov fractional-order weighted sum improves noise power spectral density estimation and speech enhancement performance in $1/f$ acoustic noise environments. The answer is yes. GL-MCRA consistently outperforms standard MCRA in objective speech enhancement quality across all tested noise conditions and SNR levels, and the improvement is statistically significant in all cases. The benefit is not limited to $1/f$ conditions, as it has been shown that comparable improvements are observed under white Gaussian noise, which suggests that the power-law memory structure grants a general advantage over first-order recursive averaging rather than one specific to long-range dependent noise. The perceptual benefit is condition-dependent and not always statistically significant in the subjective evaluation, reflecting both the limitations of the listening test and the role of the enhancement stage in mediating between noise estimation accuracy and perceived quality.

8.3 Future Work

The monotonic increase of GL-MCRA performance across the full tested range of r, with no sign of plateauing at $r = 0.9$, suggests that values beyond 0.9 may offer further improvement. The non-stationary fGn results motivate the development of a dynamically adapted r that responds to detected changes in noise statistics, combining the long memory of low r during stationary periods with the faster adaptation of high r following abrupt transitions. Replacing the Wiener filter with OM-LSA as the enhancement stage would align

the evaluation more closely with the original MCRA framework [1] and may reduce the speech distortion observed under babble and high-SNR conditions. Computing LOG-Err from saved noise PSD estimates would provide a direct measure of estimation accuracy to complement the speech quality metrics used here. Comparison of GL-MCRA against IMCRA [2] and the MMSE estimator of Hendriks et al. [21] would situate the proposed method within the broader landscape of VAD-independent noise estimators. Finally, a larger MOS listening test with multiple utterances per condition would strengthen the subjective conclusions.

Bibliography

[1] I. Cohen and B. Berdugo. "Noise estimation by minima controlled recursive averaging for robust speech enhancement". In: *IEEE Signal Processing Letters* 9.1 (2002), pp. 12–15. DOI: 10.1109/97.988717.

[2] I. Cohen. "Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging". In: *IEEE Transactions on Speech and Audio Processing* 11.5 (2003), pp. 466–475. DOI: 10.1109/TSA.2003.811544.

[3] Michael Grimm and Kristian Kroschel. *Robust Speech Recognition and Understanding*. eng. 1st ed. Vienna: IntechOpen, 2007. ISBN: 9789535158035.

[4] Y. Ephraim and D. Malah. "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.6 (1984), pp. 1109–1121. DOI: 10.1109/TASSP.1984.1164453.

[5] S. Boll. "Suppression of acoustic noise in speech using spectral subtraction". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 27.2 (1979), pp. 113–120. DOI: 10.1109/TASSP.1979.1163209.

[6] Jongseo Sohn, Nam Soo Kim, and Wonyong Sung. "A statistical model-based voice activity detection". In: *IEEE Signal Processing Letters* 6.1 (1999), pp. 1–3. DOI: 10.1109/97.736233.

[7] *ITU–T Recommendation G.729 Annex B: A silence compression scheme for G.729 optimized for terminals conforming to Recommendation V.70*. Recommendation (Annex). Superseded. International Telecommunication Union, ITU–T. URL: https://handle.itu.int/11.1002/1000/3946.

[8] K. Itoh and M. Mizushima. "Environmental noise reduction based on speech/non-speech identification for hearing aids". eng. In: *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 1. Washington DC: IEEE, 1997, 419–422 vol.1. ISBN: 0818679190.

[9] D.K. Freeman et al. "The voice activity detector for the Pan-European digital cellular mobile telephone service". In: *International Conference on Acoustics, Speech, and Signal Processing*, 1989, 369–372 vol.1. DOI: 10.1109/ICASSP.1989.266442.

[10] K. Srinivasan and A. Gersho. "Voice activity detection for cellular networks". In: *Proceedings., IEEE Workshop on Speech Coding for Telecommunications*, 1993, pp. 85–86. DOI: 10.1109/SCFT.1993.762351.

[11] Y. Ephraim and D. Malah. "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 33.2 (1985), pp. 443–445. DOI: 10.1109/TASSP.1985.1164550.

[12] Fangqi Liu and Andreas Demosthenous. "A Computation Efficient Voice Activity Detector for Low Signal-to-Noise Ratio in Hearing Aids". In: *2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*. 2021, pp. 524–528. DOI: 10.1109/MWSCAS47672.2021.9531915.

[13] Jia-lin Shen, Jeih-weih Hung, and Lin-Shan Lee. "Robust Entropy-based Endpoint Detection for Speech Recognition in Noisy Environments". In: *5th International Conference on Spoken Language Processing (ICSLP 98)*. 1998.

[14] Kun-Ching Wang and Yi-Hsing Tasi. "Voice Activity Detection Algorithm with Low Signal-to-Noise Ratios Based on Spectrum Entropy". In: *2008 Second International Symposium on Universal Communication*. 2008, pp. 423–428. DOI: 10.1109/ISUC.2008.55.

[15] Thein Htay Zaw and Nu War. "The combination of spectral entropy, zero crossing rate, short time energy and linear prediction error for voice activity detection". In: *2017 20th International Conference of Computer and Information Technology (ICCIT)*. 2017, pp. 1–5. DOI: 10.1109/ICCITECHN.2017.8281794.

[16] Abhishek Sehgal and Nasser Kehtarnavaz. "A Convolutional Neural Network Smartphone App for Real-Time Voice Activity Detection". In: *IEEE Access* 6 (2018), pp. 9017–9026. DOI: 10.1109/ACCESS.2018.2800728.

[17] John G. Proakis and Dimitris G. Manolakis. *Digital Signal Processing: Principles, Algorithms, and Applications*. 3rd ed. Upper Saddle River, NJ: Prentice-Hall International, Inc, 1996.

[18] Paul Doukhan, George Oppenheim, and Murad S. Taqqu, eds. *Theory and Applications of Long-Range Dependence*. Boston, MA: Birkhäuser Boston, 2002, p. 720. ISBN: 978-0-8176-4168-9.

[19] Hu Sheng, TianShuang Qiu, and YangQuan Chen. *Fractional Processes and Fractional-Order Signal Processing: Techniques and Applications*. eng. 1st ed. Signals and Communication Technology. London: Springer Nature, 2011. ISBN: 9781447122333.

[20] Richard C. Hendriks, Jesper Jensen, and Richard Heusdens. "Noise Tracking Using DFT Domain Subspace Decompositions". In: *IEEE Transactions on Audio, Speech, and Language Processing* 16.3 (2008), pp. 541–553. DOI: 10.1109/TASL.2007.914977.

[21] Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. "MMSE based noise PSD tracking with low complexity". In: *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2010, pp. 4266–4269. DOI: 10.1109/ICASSP.2010.5495680.

[22] A.W. Rix et al. "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs". In: *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*. Vol. 2. 2001, 749–752 vol.2. DOI: 10.1109/ICASSP.2001.941023.

[23] Schuyler R. Quackenbush, Thomas P. Barnwell, and Mark A. Clements. *Objective Measures of Speech Quality*. Prentice Hall Signal Processing Series. Prentice Hall, 1988.

[24] ITU-T. *Perceptual Evaluation of Speech Quality (PESQ): An Objective Method for End-to-End Speech Quality Assessment of Narrow-Band Telephone Networks and Speech Codecs*. ITU-T Recommendation P.862. International Telecommunication Union, 2000.

[25] Igor Podlubny. *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*. 1st ed. Vol. 198. Mathematics in Science and Engineering. Academic Press, Oct. 1998. ISBN: 9780125588409.

[26] Yi Hu and Philipos C. Loizou. "Subjective comparison and evaluation of speech enhancement algorithms". In: *Speech Communication* 49.7–8 (2007), pp. 588–601.

[27] Schmiph. *pysepm: Python Speech Enhancement Measures*. https://github.com/schmiph2/pysepm. 2019.

[28] Yi Hu and Philipos C. Loizou. "Evaluation of Objective Quality Measures for Speech Enhancement". In: *IEEE Transactions on Speech and Audio Processing* 16.1 (2008), pp. 229–238.

[29] Larry V. Hedges. "Distribution Theory for Glass's Estimator of Effect Size and Related Estimators". eng ; jpn. In: *Journal of educational statistics* 6.2 (1981), pp. 107–128. ISSN: 0362-9791.

Appendix A

Source Code

A.1 MATLAB Implementation

The following listings present the core MATLAB functions used in the evaluation pipeline. All functions operate on 8 kHz signals using a 256-point Hamming-windowed STFT with 50% overlap.

```
function fgn = generate_fgn(N, H)
    % Generates fractional Gaussian noise using Davies-Harte
        algorithm
    % N: number of samples
    % H: Hurst exponent (0.5, 1) for long-range dependence
    H = min(H, 0.9999); % clamp to avoid numerical breakdown at H = 1

    % Compute autocovariance sequence
    r = nan(N+1, 1); r(1) = 1;
    for k = 1:N
        r(k+1) = 0.5 * ((k+1)^(2*H) - 2*k^(2*H) + (k-1)^(2*H));
    end

    % Embed in circulant matrix and compute eigenvalues
    r = [r; r(end-1:-1:2)];
    lambda = real(fft(r)) / (2*N);
    lambda = max(lambda, 0); % guard against numerical noise

    % Generate fGn in frequency domain and transform back
    W = fft(sqrt(lambda) .* complex(randn(2*N, 1), randn(2*N, 1)));
    fgn = real(W(1:N))';  % trim to N samples, ensure row vector
    fgn = fgn / std(fgn); % normalise to unit variance
end
```

Listing A.1: Fractional Gaussian noise generation using the Davies-Harte algorithm.

```
function lambda_d = mcra(y, fs, alpha_d, alpha_s, alpha_p, delta, L)
% MCRA Noise Estimator - Cohen and Berdugo, 2002
%
% Inputs:
%   y       - noisy time-domain signal
%   fs      - sample rate (8000 for NOIZEUS)
%   alpha_d - noise estimate smoothing
%   alpha_s - local energy smoothing
%   alpha_p - speech presence probability
%   delta   - speech presence threshold
%   L       - minima window length in frames
%
% Output:
%   lambda_d - noise PSD estimate (K x num_frames)


%  STFT Matrix, Frequency Vector, Time Vector
[S, ~, ~] = stft(y, fs, 'Window', hamming(256), 'OverlapLength', 128, 'FFTLength', 256);

% Power spectrum
P = abs(S).^2; % the '.^' operator is element-wise exponentiation

% Dimensions
K      = size(P, 1);
num_frames = size(P, 2);

% Initialisation
S      = P(:, 1);
S_min = min(P(:,1:min(L, num_frames)), [], 2);
S_tmp = S_min;
p      = zeros(K, 1);
lambda_d = zeros(K, num_frames);
lambda_d(:, 1) = P(:, 1); % initial noise estimate for each bin at frame 1
frame_count = 0;

% Main loop
for ell = 2:num_frames
    % 1. Frequency smoothing then time smoothing
    w = 1; % window half-width, giving 3 bins
    S_f = zeros(K, 1);
    for k = 1:K
        k_min = max(1, k - w);
        k_max = min(K, k + w);
```

```
            S_f(k) = mean(P(k_min:k_max, ell));
        end

        % Time smoothing of locally frequency-smoothed power
        S = alpha_s * S + (1 - alpha_s) * S_f;

        % 2. Minima tracking
        S_min = min(S_min, S);
        S_tmp = min(S_tmp, S);

        frame_count = frame_count + 1;
        if frame_count >= L
            S_min = S_tmp;
            S_tmp = S;
            frame_count = 0;
        end

        % 3. Ratio
        S_r = S ./ (S_min + 1e-10);

        % 4. Speech presence indicator
        I = S_r > delta;

        % 5. Speech presence probability
        p = alpha_p * p + (1 - alpha_p) * I;

        % 6. Time-varying smoothing parameter
        alpha_tilde = alpha_d + (1 - alpha_d) * p;

        % 7. Noise estimate update
        lambda_d(:, ell) = alpha_tilde .* lambda_d(:, ell-1) + ...
                           (1 - alpha_tilde) .* P(:, ell);

    end
end
```

Listing A.2: Standard MCRA noise estimator from Cohen and Berdugo, 2002.

```
function lambda_d = mcra_gated(y, fs, alpha_d, alpha_s, alpha_p, delta, L)
% Gated MCRA Noise Estimator
%
% Inputs:
%   y       - noisy time-domain signal
%   fs      - sample rate (8000 for NOIZEUS)
%   alpha_d - noise estimate smoothing
%   alpha_s - local energy smoothing
%   alpha_p - speech presence probability
%   delta   - speech presence threshold
%   L       - minima window length in frames
%
% Output:
%   lambda_d - noise PSD estimate (K x num_frames)


%  STFT Matrix, Frequency Vector, Time Vector
[S, ~, ~] = stft(y, fs, 'Window', hamming(256), 'OverlapLength', 128, 'FFTLength', 256);

% Power spectrum
P = abs(S).^2; % the '.^' operator is element-wise exponentiation

% Dimensions
K      = size(P, 1);
num_frames = size(P, 2);

% Initialisation
S      = P(:, 1);
S_min = min(P(:,1:min(L, num_frames)), [], 2);
S_tmp = S_min;
p      = zeros(K, 1);
lambda_d = zeros(K, num_frames);
lambda_d(:, 1) = P(:, 1); % initial noise estimate for each bin at frame 1
frame_count = 0;

% Main loop
for ell = 2:num_frames
    % 1. Frequency smoothing then time smoothing
    w = 1; % window half-width, giving 3 bins
    S_f = zeros(K, 1);
    for k = 1:K
        k_min = max(1, k - w);
        k_max = min(K, k + w);
```

```
            S_f(k) = mean(P(k_min:k_max, ell));
        end

        % Time smoothing of locally frequency-smoothed power
        S = alpha_s * S + (1 - alpha_s) * S_f;

        % 2. Minima tracking
        S_min = min(S_min, S);
        S_tmp = min(S_tmp, S);

        frame_count = frame_count + 1;
        if frame_count >= L
            S_min = S_tmp;
            S_tmp = S;
            frame_count = 0;
        end

        % 3. Ratio
        S_r = S ./ (S_min + 1e-10);

        % 4. Speech presence indicator
        I = S_r > delta;

        % 5. Speech presence probability
        p = alpha_p * p + (1 - alpha_p) * I;

        % 6. Time-varying smoothing parameter
        alpha_tilde = alpha_d + (1 - alpha_d) * p;

        P_gated_frame = (1 - p) .* P(:, ell) + ...
                        p .* lambda_d(:, ell-1);

        % 7. Noise estimate update
        lambda_d(:, ell) = alpha_tilde .* lambda_d(:, ell-1) + ...
                           (1 - alpha_tilde) .* P_gated_frame;

    end
end
```

Listing A.3: Gated MCRA noise estimator.

```
function lambda_d_frac = gl_mcra(y, fs, alpha_d, alpha_s, alpha_p, delta, L, L_mem, r)
% Grunwald-Letnikov Modification for MCRA Noise Estimator
%
% Inputs:
%   y       - noisy time-domain signal
%   fs      - sample rate (8000 for NOIZEUS)
%   alpha_d - noise estimate smoothing (unused)
%   alpha_s - local energy smoothing
%   alpha_p - speech presence probability
%   delta   - speech presence threshold
%   L       - minima window length in frames
%   r       - fractional order
%
% Output:
%   lambda_d_frac - noise PSD estimate (K x num_frames)


%  STFT Matrix, Frequency Vector, Time Vector
[S, ~, ~] = stft(y, fs, 'Window', hamming(256), ...
            'OverlapLength', 128, 'FFTLength', 256);

% Power spectrum
P = abs(S).^2; % the '.^' operator is element-wise exponentiation

% Dimensions
K      = size(P, 1); % number of frequency bins
num_frames = size(P, 2); % number of frames

% Initialisation
S      = P(:, 1);
% take the minimum value of each frequency bin
S_min = min(P(:,1:min(L, num_frames)), [], 2);
S_tmp = S_min; % assign this minimum as the temporary value
p      = zeros(K, 1); % speech probability at each bin, starts as 0
lambda_d_frac = zeros(K, num_frames); % initialise noise estimate for each bins as 0
lambda_d_frac(:, 1) = P(:, 1); % initial noise estimate for each bin at frame 1
frame_count = 0;

% GL variables
weights = zeros(1, L_mem); % create vector of weights, 1xL_mem
weights(1) = 1; % as per GL definition, first weight = 1
for j = 2:L_mem
```

```
43          % loop through rest of the weights with weighing function
                from GL
44          weights(j) = (1 - (r+1) / (j - 1)) * weights(j - 1);
45      end
46      weights = abs(weights) / sum(abs(weights)); % normalise weights
47
48
49      P_gated = zeros(K, num_frames);
50      P_gated(:, 1) = P(:, 1);
51
52      % Main loop
53      for ell = 2:num_frames
54          % 1. Frequency smoothing then time smoothing
55          w = 1; % window half-width, giving 3 bins
56          S_f = zeros(K, 1);
57          for k = 1:K
58              k_min = max(1, k - w);
59              k_max = min(K, k + w);
60              S_f(k) = mean(P(k_min:k_max, ell));
61          end
62
63          % Time smoothing of locally frequency-smoothed power
64          S = alpha_s * S + (1 - alpha_s) * S_f;
65
66          % 2. Minima tracking
67          S_min = min(S_min, S);
68          S_tmp = min(S_tmp, S);
69
70          frame_count = frame_count + 1;
71          if frame_count >= L
72              S_min = S_tmp;
73              S_tmp = S;
74              frame_count = 0;
75          end
76
77          % 3. Ratio
78          S_r = S ./ (S_min + 1e-10);
79
80          % 4. Speech presence indicator
81          I = S_r > delta;
82
83          % 5. Speech presence probability
84          p = alpha_p * p + (1 - alpha_p) * I;
85
86          % 6. Gate the power using speech presence probability
87          % Speech must be gated because otherwise, the GL sum includes
                power
```

```
        % from speech frames
        P_gated(:, ell) = (1 - p) .* P(:, ell) + ...
                          p .* lambda_d_frac(:, ell-1);

        % GL noise estimate update
        lags = min(ell, L_mem);
        lambda_d_frac(:, ell) = P_gated(:, ell-lags+1:ell) * ...
                                fliplr(weights(1:lags))';
    end
end
```

Listing A.4: GL-MCRA noise estimator (proposed).

A.2 AI Tool Usage

The following prompts were used to generate parts of analysis scripts with the assistance of an AI language model:

1. "Write a Python script using the `scipy` package to load audio files"
2. "Write a Python script using the `pandas` package to output and save a CSV file"

```
import os
import numpy as np
import pandas as pd
import pysepm
from scipy.io import wavfile

# --- Config ------------------------------------------------

CLEAN_DIR    = '../speech clips/clean'
NOISY_DIR    = '../output/'
ENHANCED_DIR = '../enhanced'
RESULTS_CSV  = '../results/results.csv'

CONDITIONS = ['wgn', 'fgn_0.6', 'fgn_0.75', 'fgn_0.9',
              'fgn_ns_0.6_0.9', 'train', 'babble']
SNR_LEVELS = [0, 5, 10, 15]
R_VALUES   = [round(r,1) for r in np.arange(0.1, 1.0, 0.1)]
FS         = 8000

# --- Helper functions --------------------------------------
# AI generated start
def load_wav(path):
    fs, x = wavfile.read(path)
```

```
    if x.dtype != np.float32 and x.dtype != np.float64:
        x = x.astype(np.float64) / np.iinfo(x.dtype).max
    return fs, x.flatten()
# AI generated end

def match_length(a, b):
    n = min(len(a), len(b))
    return a[:n], b[:n]

def get_clean_name(noisy_fname):
    return noisy_fname.split('_')[0] + '.wav'

def safe_metric(fn, *args):
    try:
        return float(fn(*args))
    except Exception as e:
        print(f'    Metric error: {e}')
        return np.nan

def pesq_score(clean, degraded, fs):
    pesq = pysepm.pesq(clean,degraded,fs)[0]
    return float(pesq)

# --- Main loop ---------------------------------------------
def main():
    os.makedirs(os.path.dirname(RESULTS_CSV), exist_ok=True)
    rows = []

    for cond in CONDITIONS:
        for snr in SNR_LEVELS:
            snr_str = f'snr{snr}'
            noisy_path = os.path.join(NOISY_DIR, cond, snr_str)

            if not os.path.isdir(noisy_path):
                print(f'Skipping missing: {noisy_path}')
                continue

            for noisy_fname in sorted(os.listdir(noisy_path)):
                if not noisy_fname.endswith('.wav'):
                    continue

                clean_path = os.path.join(CLEAN_DIR, get_clean_name(
                    noisy_fname))
                noisy_file = os.path.join(noisy_path, noisy_fname)
                base       = os.path.splitext(noisy_fname)[0]

                if not os.path.isfile(clean_path):
```

```
                print(f'    Missing clean: {clean_path}')
                continue

            fs, clean = load_wav(clean_path)
            fs, noisy = load_wav(noisy_file)
            clean, noisy = match_length(clean, noisy)

            seg_snr_noisy = safe_metric(pysepm.SNRseg, clean,
               noisy, FS)
            print(f' {cond} | snr{snr} | {noisy_fname}')

            # --- Standard MCRA ---

            mcra_file = os.path.join(ENHANCED_DIR, cond, snr_str,
                                     'mcra', f'{base}
                                        _enhanced_mcra.wav')

            if os.path.isfile(mcra_file):
                fs, mcra_enh = load_wav(mcra_file)
                c, m = match_length(clean, mcra_enh)
                seg_snr_mcra   = safe_metric(pysepm.SNRseg, c, m,
                    FS)
                pesq_mcra      = safe_metric(pesq_score, c, m, FS
                   )
                delta_snr_mcra = seg_snr_mcra - seg_snr_noisy
            else:
                seg_snr_mcra = pesq_mcra = delta_snr_mcra = np.
                   nan

            # --- Gated MCRA
            gated_file = os.path.join(ENHANCED_DIR, cond, snr_str
                , 'mcra_gated',
                                      f'{base}
                                         _enhanced_mcra_gated.wav
                                         ')
            if os.path.isfile(gated_file):
                _, gated_enh    = load_wav(gated_file)
                c, g            = match_length(clean, gated_enh)
                seg_snr_gated   = safe_metric(pysepm.SNRseg,c, g,
                    FS)
                pesq_gated      = safe_metric(pesq_score, c, g,
                   FS)
                delta_snr_gated =seg_snr_gated - seg_snr_noisy
            else:
                seg_snr_gated = pesq_gated = delta_snr_gated = np
                   .nan

```

```
                # --- GL-MCRA for each r ---
                for r in R_VALUES:
                    gl_file = os.path.join(
                        ENHANCED_DIR, cond, snr_str, f'gl_r{r:.1f}',
                        f'{base}_enhanced_gl_r{r:.1f}.wav')

                    if os.path.isfile(gl_file):
                        fs, gl_enh = load_wav(gl_file)
                        c, g = match_length(clean, gl_enh)
                        seg_snr_gl   = safe_metric(pysepm.SNRseg, c,
                            g, FS)
                        pesq_gl      = safe_metric(pesq_score, c, g,
                            FS)
                        delta_snr_gl = seg_snr_gl - seg_snr_noisy
                    else:
                        seg_snr_gl = pesq_gl = delta_snr_gl = np.nan

    # AI generated start
                    rows.append({
                        'condition':        cond,
                        'snr_level':        snr,
                        'filename':         noisy_fname,
                        'r':                r,
                        # Noisy baseline
                        'segsnr_noisy':     seg_snr_noisy,
                        # standard MCRA
                        'segsnr_mcra':      seg_snr_mcra,
                        'delta_snr_mcra':   delta_snr_mcra,
                        'pesq_mcra':        pesq_mcra,
                        # Gated MCRA
                        'segsnr_gated':     seg_snr_gated,
                        'delta_snr_gated':  delta_snr_gated,
                        'pesq_gated':       pesq_gated,
                        # GL-MCRA
                        'segsnr_gl':        seg_snr_gl,
                        'delta_snr_gl':     delta_snr_gl,
                        'pesq_gl':          pesq_gl,
                    })

    df = pd.DataFrame(rows)
    df.to_csv(RESULTS_CSV, index=False)
    # AI generated end
    print(f'\nResults saved to {RESULTS_CSV}')
    print(f'Total rows: {len(df)}')
    print(df.head())
```

```
if __name__ == '__main__':
    main()
```

Listing A.5: Process MATLAB results to CSV

1. "Write a Python script using the pandas package to load a CSV file, rename the columns by iterating through a list of condition names and appending MCRA and GL-MCRA suffixes."

```
import pandas as pd
import numpy as np
from scipy import stats

# AI generated start
# Load skipping first row labels
df = pd.read_csv('../responses_for_shapiro_test.csv', header=1)

# Rename columns
conditions_raw = ['wGn␣0dB', 'wGn␣10dB', 'fGn␣0.6␣0dB', 'fGn␣0.6␣10dB',
                  'fGn␣0.75␣0dB', 'fGn␣0.75␣10dB', 'fGn␣0.9␣0dB', 'fGn␣0.9␣10dB',
                  'fGn␣NS␣0dB', 'fGn␣NS␣10dB', 'train␣0dB', 'train␣10dB',
                  'babble␣0dB', 'babble␣10dB']

new_cols = []

for cond in conditions_raw:
    new_cols.append(f'{cond} MCRA')
    new_cols.append(f'{cond} GL')
df.columns = new_cols

df = df.astype(float)

conditions = {
    'wGn':         ['wGn␣0dB',      'wGn␣10dB'],
    'fGn␣H=0.6':  ['fGn␣0.6␣0dB', 'fGn␣0.6␣10dB'],
    'fGn␣H=0.75': ['fGn␣0.75␣0dB','fGn␣0.75␣10dB'],
    'fGn␣H=0.9':  ['fGn␣0.9␣0dB', 'fGn␣0.9␣10dB'],
    'fGn␣NS':     ['fGn␣NS␣0dB',  'fGn␣NS␣10dB'],
    'Train':      ['train␣0dB',   'train␣10dB'],
    'Babble':     ['babble␣0dB',  'babble␣10dB'],
}
# AI generated end

```

```
print("=== Shapiro-Wilk + Wilcoxon Signed-Rank for MOS ===\n")

for snr_idx, snr in enumerate(['0dB', '10dB']):
    print(f"--- {snr} SNR ---")
    print(f"{'Condition':<14} {'SW p':>8} {'Normal?':>8} "
          f"{'Wilcoxon p':>12} {'sig':>5} {'Delta':>8}")
    print("-" * 62)
    for cond, snr_list in conditions.items():
        key = snr_list[snr_idx]
        m    = df[f'{key} MCRA'].values
        g    = df[f'{key} GL'].values
        diff = g - m

        _, sw_p = stats.shapiro(diff)
        normal  = 'yes' if sw_p > 0.05 else 'no'

        try:
            _, wx_p = stats.wilcoxon(g, m)
        except ValueError:
            wx_p = 1.0

        sig   = '***' if wx_p < 0.001 else \
                '**'  if wx_p < 0.01  else \
                '*'   if wx_p < 0.05  else 'n.s.'
        delta = np.mean(diff)

        print(f"{cond:<14} {sw_p:>8.4f} {normal:>8} "
              f"{wx_p:>12.4f} {sig:>5} {delta:>+8.2f}")
    print()
```

Listing A.6: Automated Shapiro-Wilk and Wilcoxon tests.