Randomized Sampling and Multiplier-Less Filtering

by

Sourav R. Dey

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2008

Author ....
Department of Electrical Engineering and Computer Science
January 30, 2008

Certified by ....
Alan V. Oppenheim
Ford Professor of Engineering
Thesis Supervisor

Accepted by ....
Terry P. Orlando
Chairman, Department Committee on Graduate Students

Randomized Sampling and Multiplier-Less Filtering
by
Sourav R. Dey

Submitted to the Department of Electrical Engineering and Computer Science
on January 31, 2008, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering

Abstract

This thesis considers the benefits of randomization in two fundamental signal processing techniques: sampling and filtering. The first part develops randomized non-uniform sampling as a method to mitigate the effects of aliasing. Randomization of the sampling times is shown to convert aliasing error due to uniform under-sampling into uncorrelated shapeable noise. In certain applications, especially perceptual ones, this form of error may be preferable.

Two sampling structures with are developed in this thesis. In the first, denoted simple randomized sampling, non-white sampling processes can be designed to frequency-shape the error spectrum, so that its power is minimized in the band of interest. In the second model, denoted filtered randomized sampling, a pre-filter, post-filter, and the sampling process can be designed to further frequency-shape the error to improve performance. The thesis develops design techniques using parametric binary process models to optimize the performance of randomized non-uniform sampling. In addition, a detailed second-order error analysis, including performance bounds and results from simulation, is presented for each type of sampling.

The second part of this thesis develops randomization as a method to improve the performance of multiplier-less FIR filters. Static multiplier-less filters, even when carefully designed, result in frequency distortion as compared to a desired continuous-valued filter. Replacing each static tap with a binary random process is shown to mitigate this distortion, converting the error into uncorrelated shapeable noise. As with randomized sampling, in certain applications this form of error may be preferable.

This thesis presents a FIR Direct Form I randomized multiplier-less filter structure denoted binary randomized filtering (BRF). In its most general form, BRF incorporates over-sampling combined with a tapped delay-line that changes in time according to a binary vector process. The time and tap correlation of the binary vector process can be designed to improve the error performance. The thesis develops design techniques using parametric binary vector process models to do so. In addition, a detailed second-order error analysis, including performance bounds, error scaling with over-sampling, and results from simulation, is presented for the various forms of BRF.

Thesis Supervisor: Alan V. Oppenheim
Title: Ford Professor of Engineering

Acknowledgments

I would first like to thank God for giving me the perseverance to finish. Life has brought me many blessings – support from my friends and family, numerous opportunities, and achievements. I am eternally grateful for all of them. I promise use all my knowledge to make the world a better place. Thank you Thakur.

They say the only good thesis is a finished thesis and I couldn't agree more. Though it is a single author document, it is not the product of a single individual. Rather it is the sum of all those who have supported me throughout this long, trying journey. Before getting into the personal acknowledgments, I want to thank all my teachers and mentors over the years for believing in me. I have come this far only because of your encouragement.

After God, I have to thank Al. It's been six years since I first came into the group as a green undergraduate – living out of my car and generally a complete mess. Under his guidance and supervision I've become a professional – a mature engineer with the ability to think creatively about any problem. Al provided encouragement when I was being too hard on myself it and focus when I was not pushing myself hard enough. This thesis would not exist without his guidance. Al, you're now a lifelong colleague, mentor, and friend.

In addition, I want to thank my thesis committee, Vivek and Charlie. You guys were phenomenal. Your comments and feedback made the thesis better. Your mentorship helped me grow as a professional and as a person.

Then there are the members of DSPG with whom I not only share the suite with, but with whom I have become very good friends with. On those bitterly cold January mornings, you guys are the reason I want to come into the office. I couldn't have finished this without you all: Petros, Maya, Tom, Melanie, Dennis, Shay, Zahi, Little Al. The thesis grew out all of our numerous conversations and whiteboard brainstorming sessions. Thanks for letting me interrupt your work and talk about some ”random ass sh**”.

And then to all my friends, you guys kept me sane. You have all supported me through bouts of crippling depression and moments of pure mania. I know I'm unstable. But hey, that's just who I is. Suresh, I don't know what I would have done if we hadn't gone to India together. It was a random decision on my part then, but without it there would have been no Suresh. And MIT would have been a lonely place. Ankit, we've lived together for six years and, yeah you're dirty, but I couldn't have asked for a better roommate and friend. It was a random decision to live together, but without it there would have been no Braveheart, Chris Rock, Jack Bauer, or love of anger. Who would I even be? To everyone else – Danilo, Demba, Tommy, Sachin, Vibhav, Josh, and others – you guys really made the last six years great. From hiking to working out, from vacations to television – these are the memories I will take away from Cambridge.

My family is the building block upon which I have built my life. They are my Rock (and yes that is a pun). Ma, Baba, Piya, thank you for your unconditional love and constant support. This thesis is dedicated to you. I did it!

And lastly to Pallabi, my onta, you've been the greatest support of all. Weve grown so much over the years together. You gave me the power to finish, the faith to carry on, and always remind me of how good life is. I'm looking forward to spending the rest of my life with you :) We've had so many good times and they will only get better! California here we come!

Contents

1 Introduction **17**

2 Randomized Non-Uniform Sampling **21**

2.1 Motivation 21

2.2 Randomized Sampling Model 23

2.2.1 Continuous-Time Model 23

2.2.2 Discrete-Time Model 24

3 Simple Randomized Sampling **29**

3.1 Problem Statement 29

3.2 White SRS 32

3.3 Frequency-Shaped SRS 32

3.3.1 Design Problem 33

3.3.2 Relaxation Bound 34

3.3.3 Parametric Frequency-Shaped SRS 36

3.4 Error Analysis 40

3.4.1 White SRS 40

3.4.2 Optimal High-Rate Sampling Frequency 41

3.4.3 Frequency-Shaping SRS 42

3.5 Numerical Experiments 44

3.5.1 Example 44

3.5.2 Peakiness 46

3.6 Extensions 47

4 Filtered Randomized Sampling **53**

4.1 Introduction 53

4.2 Distortion-Free White FRS 57

4.2.1 Design Problem 57

4.2.2 Optimal Design 58

4.2.3 Error Analysis 61

4.3 Distortion-Free Frequency-Shaped FRS 61

4.3.1 Design Problem 61

4.3.2 Optimal Design 63

4.3.3 Error Analysis 65

4.4 Unconstrained White FRS 66

4.4.1 Design Problem 66
4.4.2 Optimal Design 69
4.4.3 Error Analysis 71
4.5 Unconstrained Frequency-Shaped FRS 72
4.6 Numerical Experiments 73
4.6.1 Examples 73
4.6.2 Peakiness 85
4.7 Extensions 87

5 Randomized Multiplier-Less Filtering **89**
5.1 Introduction 89
5.2 Binary Randomized Filtering 90
5.2.1 Direct Form I FIR BRF Model 90
5.2.2 Randomization Error 93
5.2.3 Types of Binary Randomized Filters 96
5.3 Extensions 97
5.3.1 Other Filter Architectures 97
5.3.2 Multi-Level Coefficient Quantization 97

6 Memoryless Standard Direct Form I
FIR Binary Randomized Filtering **99**
6.1 Problem Statement 99
6.2 Tap-Independent Memoryless BRF 101
6.2.1 Problem Statment 101
6.2.2 Optimal Solution 101
6.2.3 Scale Invariance 102
6.2.4 Error Analysis 104
6.3 Tap-Correlated Memoryless BRF 109
6.3.1 Problem Statement 109
6.3.2 Optimal Solution 111
6.3.3 Scale Invariance 116
6.3.4 Parametric Memoryless BRF 119
6.3.5 Relaxation Bound 122
6.3.6 Error Analysis 123
6.4 Numerical Experiments 125
6.4.1 Example 125
6.4.2 Peakiness 126

7 Oversampled Direct Form I FIR
Binary Randomized Filtering **137**
7.1 Oversampled BRF Model 137
7.1.1 Oversampled Direct Form I FIR Structure 137
7.1.2 Oversampled Direct Form I FIR BRF 138
7.1.3 Multiplier-less Rate Conversion 139
7.2 Error Analysis 140
7.2.1 General Analysis 140

7.2.2 Memoryless Oversampled BRF 143
7.2.3 Frequency-Shaping BRF 144
7.3 Numerical Experiments 147

List of Figures

1-1 Thesis Roadmap 19

2-1 Block diagrams for signal and sampling model. $h(t)$ is a fixed linear time-invariant filter. $x(t)$ is band-limited to π/T. The maximum rate of the sampling process is restricted to be $r_{\text{max}} = 1/T_r < r$. The goal is to get the reconstructed output $\hat{y}(t)$ close to the desired output $y(t)$. 21

2-2 Block diagrams for continuous-time randomized sampling models studied in this thesis. 26

2-3 Important frequency parameters in randomized sampling. (a) illustrates the CT frequency parameters. The blue area denotes the range of possible values for Ω_{hr}. The red area denotes the possible values for Ω_{avg}. (b) illustrates the DT frequency parameters when sampled at rate T_{hr}. 27

2-4 Block diagrams for discrete-time randomized sampling models studied in this thesis. 28

3-1 Coefficient space for $\mu < 1/2$ reproduced from [5]. The shaded area is the set of coefficients for which the algorithm is guaranteed not to overflow. As μ increases, the two constraints pivot around the point $(1,1)$, as shown in the plot. 39

3-2 Markov chain for binary-AR model with $p = 2$. The states are $(r[n-2], r[n-1])$. 39

3-3 Possible rate implementation pairs $(\mu, \Omega_{\text{hr}})$. Pairs shown for $\Omega_{\text{avg}} = 400/1\pi$ and $160/2\pi$. The Ω_{hr} axis on this figure matches with Figure 3-4 below. . . 43

3-4 MSE scaling as function of Ω_{hr} for example discussed in Section 3.5. Error curves shown for white-SRS and frequency-shaped SRS at two fixed average sampling rates, $\Omega_{\text{avg}} = 400/2\pi$ and $160/2\pi$. 43

3-5 Empirical plots of randomized sampling power spectra for the example discussed in Section 3.5. Average sampling rate fixed to $\Omega_{\text{avg}} = 160/2\pi$. Gray regions denote the stop-band of $H(\omega)$. White regions denote the pass-band of $H(\omega)$. 48

3-6 Results of Boufounos process design for the example discussed in Section 3.5. Design done using numerical optimization with $\epsilon = 0.05$. The optimal parameters are $a_1 = -0.3394$ and $a_2 = 0.4783$. 49

3-7 SNR and shaping-gain as a function of μ for example discussed in Section 3.5. $\Omega_{\text{hr}} = 400/2\pi$, the Nyquist rate for the input. 50

3-8 Shaping gain of MA(1) process defined in Eqn.(3.57) as a function of θ. $\mu = 2/5$. 51

3-9 Shaping gain of AR(1) process defined in Eqn.(3.58) as a function of ρ. $\mu = 2/5$. 51

4-1 FRS models. 54

4-2 Important frequency parameters in filtered randomized sampling. (a) illustrates the CT frequency parameters. The pass-bands for $H(j\Omega)$, $G_1(j\Omega)$, and $G_2(j\Omega)$ are Ω_h. The red area denotes the possible values for Ω_{avg}. Ω_{hr} should be set to $2\Omega_h$. The hatched region denotes the frequencies of $S_{xx}(j\Omega)$ that are removed by $G_1(j\Omega)$. (b) illustrates the DT frequency parameters when sampled at $\Omega_{\text{hr}} = 2\Omega_h$. 56

4-3 D*PCM block diagram. 59

4-4 Relaxation parameters as a function of μ 66

4-5 Uniform and white-SRS randomized sampling for the example discussed in Section 4.6. Average sampling rate fixed to $\Omega_{\text{avg}} = 220/2\pi$. The reconstruction filter $H(e^{j\omega})$ is assumed to be unity across the band. 76

4-6 Distortion-free filtered randomized sampling for the example discussed in Section 4.6. Average sampling rate fixed to $\Omega_{\text{avg}} = 220/2\pi$. The reconstruction filter $H(e^{j\omega})$ is assumed to be unity across the band. 77

4-7 Unconstrained white FRS for example discussed in Section 4.6. SNR = 5.586 dB. Average sampling rate fixed to $\Omega_{\text{avg}} = 220/2\pi$. The reconstruction filter $H(e^{j\omega})$ is assumed to be unity across the band. 78

4-8 Distortion-free white FRS filter for the example discussed in Section 4.6. Ideal filters are approximated using 2048-pt linear phase FIR filters. 79

4-9 Distortion-free frequency-shaped FRS filters for the example discussed in Section 4.6. Filters are found using relaxed block coordinate descent on discrete frequency grid. The filters are approximated using 2048-pt linear phase FIR filters. 80

4-10 Relaxed block coordinate descent error for distortion-free frequency-shaped FRS design. MSE plotted agains iteration. Stationary point achieved in about 10 iterations. 81

4-11 Distortion-free frequency-shaped FRS binary-AR characteristics for the example discussed in Section 4.6. Parameters optimized using relaxed block coordinate descent. 82

4-12 Unconstrained white FRS filters for the example discussed in Section 4.6. Ideal filters are approximated using 2048-pt linear phase FIR filters. 83

4-13 SNR scaling as a function of μ for example discussed in Section 4.6. $f_{\text{hr}} = 400$ Hz, the Nyquist rate for the filter cutoff Ω_h. 84

4-14 Shaping gain of MA(1) process defined in Eqn.(4.70) as a function of θ. $\mu = 11/20$. 86

4-15 Shaping gain of AR(1) process defined in Eqn.(4.72) as a function of ρ. $\mu = 11/20$. 86

4-16 Possible feedback FRS structure inspired by DPCM. 87

5-1 Frequency distortion caused by binary quantization of impulse response. Desired response is 33 tap linear-phase low-pass filter designed using the Parks-McClellan algorithm. 91

5-2 Hierarchy of multiplier-less randomized filters. The types of filters developed in this thesis are illustrated in bold boxes. Those not developed in detail are in italics. 92
5-3 Block diagrams of Direct Form I FIR filter implementations (a) the continuous-valued counterpart, (b) the BRF implementation and, (c) an over-sampled BRF implementation. 94

6-1 Block diagram of N-tap simple BRF. 100
6-2 The red line denotes the set of possible implementation points for a continuous-valued filter $\mathbf{b}$ along the ray $C\mathbf{e}_b$. The optimal implementation point for timBRF is on the face of the cube. 103
6-3 Plot of optimal MSE and implementation manifold for 3-tap timBRF FIR filters on the unit-power manifold defined by $\sum_{i=0}^{2} b_i^2 = 1$. 106
6-4 Joint density $p_{\mathbf{h}}(\mathbf{h}[n])$ in three dimensions 112
6-5 Plot of optimal MSE $\mathcal{E}^*(K)$ for $\mathbf{b} = [\frac{1}{2}\ \frac{1}{2}\ \frac{1}{\sqrt{2}}]$. Minimum is not at $K = \max\{|b_i|\}$. 115
6-6 Plot of optimal MSE and implementation manifold for 3-tap tcmBRF FIR filters on the unit-power manifold defined by $\sum_{i=0}^{2} b_i^2 = 1$. 118
6-7 Correlation gain, $\mathcal{G}_{\text{tcm}}$ for 3-tap tcmBRF FIR filters on the unit-power manifold defined by $\sum_{i=0}^{2} b_i^2 = 1$. 128
6-8 Worst-case MSE as a function of N. timBRF curve is analytic expression from Eqn.(6.33). tcmBRF relaxation bound curves are a result of numerical optimization using `fmincon` using random restarts. Input is normalized so $R_{xx}[0] = 1$. 128
6-9 Desired continuous-valued filter. $N = 33$ length Parks-McClellan filter designed using the specifications of Eqn.(6.83). 129
6-10 Power spectra of input and desired output. 130
6-11 Error power spectrum and time-domain output for tap-independent memoryless BRF implementation of the example in Section 6.4.1. 131
6-12 Error power spectrum and time-domain output for tap-correlated memoryless BRF implementation of the example in Section 6.4.1 with Qaqish order $\kappa = 4$.132
6-13 Optimal covariance matrices from relaxation bound and Qaqish parametric optimization of the $N = 33$ tap example in Section 6.4.1. 133
6-14 MSE scaling as a function of N for the Parks-McClellan filter with specifications given in Eqn.(6.83) and input spectrum given in Section 6.4.1. 134
6-15 Correlation gain, $\mathcal{G}_{\text{tcm}}$, as a function of N for the Parks-McClellan filter with specifications given in Eqn.(6.83) and input spectrum given in Section 6.4.1. 134
6-16 SNR scaling as a function of N for the Parks-McClellan filter with specifications given in Eqn.(6.83) and input spectrum given in Section 6.4.1. 135
6-17 Max ripple error as a function of N for Parks-McClellan filters designed with specification Eqn.(6.83). 135
6-18 Correlation gain, $\mathcal{G}_{\text{tcm}}$, as a function of Qaqish order, κ, for example of Section 6.4.1. 136
6-19 Correlation gain, $\mathcal{G}_{\text{tcm}}$, of a AR(1) input process as a function ρ. Desired filter is 33 tap Parks-McClellan example of Section 6.4.1 136

7-1 Diagrams for continuous-valued oversampled Direct Form I FIR filtering. The rate-conversion filters are assumed to be ideal and the coefficients b_i are assumed to be continuous-valued. $B(e^{j\omega})$ is the frequency response of the standard structure without oversampling. 139
7-2 Block Diagram of Oversampled Direct Form I FIR BRF 140
7-3 Error analysis block diagrams for oversampled Direct Form I BRF 141
7-4 Error power spectrum and time-domain output for oversampled tap-independent memoryless BRF implementation of the example in Section 6.4.1. Oversampling rate $L = 4$. 148
7-5 Error power spectrum and time-domain output for oversampled tap-correlated memoryless BRF implementation of the example in Section 6.4.1. Oversampling rate $L = 4$ and Qaqish order $\kappa = 4$. 149
7-6 SNR of memoryless BRF as a function of oversampling rate L for example of Section 6.4.1. 150

List of Tables

4.1 Types of FRS 57

5.1 Types of binary randomized filtering 96

Chapter 1

Introduction

Randomization has been used in signal processing as a technique to mitigate certain forms of error. For example, randomized dither is used in quantization to convert perceptually disturbing quantization artifacts into less disturbing uncorrelated noise. This thesis extends randomization to two fundamental building blocks of signal processing: sampling and filtering. In both contexts, we find that randomization leads to better control over the error, often converting it into a preferable form.

With uniform sampling, if the input signal is sampled below its Nyquist rate, aliasing occurs. Aliasing is problematic in certain applications, especially those based on perception. For example, in ray-traced computer graphics aliasing appears as visually disturbing moir artifacts. By contrast, randomization of the sampling times can convert aliasing into shapeable noise that is uncorrelated, which can be preferable to aliasing. In the ray-tracing example, the error due to randomized non-uniform sampling appears as additive noise across all frequencies, which the eye is less sensitive too.

In this way, randomized non-uniform sampling can algorithmically extend the effective bandwidth of sampling. The lowered sampling rate could lead to improvements in power-consumption, computational-cost, and hardware complexity in analog-to-digital converters. This could potentially be useful in a wide range of applications, from sensor network nodes to ray-traced computer graphics.

Randomization can also be used to improve the quality of filtering. In certain low-complexity, low-latency applications multiplier-less filters are desirable. However, implementing filters without multiplies often leads to severe coefficient quantization error. Specifically, the frequency response of the multiplier-less implementation is distorted from the desired continuous-valued filter. In certain applications, especially where the fine structure of the frequency response is important, this distortion can be problematic. For example, in audio equalization the human ear is sensitive to distortion of the frequency response. In this thesis, we find that randomization of multiplier-less filter coefficients by switching between coefficient quantization levels leads to shapeable noise that is uncorrelated with the input, which can be preferable to frequency response distortion. In the audio equalization example, randomized multiplier-less filtering can be used to turn frequency distortion into uncorrelated noise that is shaped to higher frequencies where the ear is less sensitive to it.

Randomized multiplier-less filtering is a new filtering paradigm that could potentially replace standard filter implementations with coefficient multiplies. With proper optimization, randomized filters could offer an alternative implementation with a smaller hardware footprint and lower-latency than standard DSP architectures. They could be potentially useful in a host of applications from space to medical devices – anywhere low-complexity, low-latency processing is necessary.

Figure 1-1 illustrates the roadmap for the thesis. At the root of the tree we place the randomization principle which is applied in two contexts, non-uniform sampling and multiplier-less filtering. In Chapters 2 through 4, we discuss randomized non-uniform sampling. In Chapter 2, we introduce randomized non-uniform sampling and discuss relevant background, potential applications, and our particular discrete-time/LTI-reconstruction model.

Two forms of randomized sampling are developed in this thesis, simple randomized sampling (sRS) and filtered randomized sampling (FRS). In Chapter 3 we develop SRS, which has no pre-filters or post-filters. The main result of this chapter is that the sampling process correlation can be designed to frequency shape the sampling error out of band. This requires the design of binary processes with specified auto-covariances, which is a non-trivial problem. It is addressed in detail in Chapter 3.

In Chapter 4 we develop FRS, which has a tunable pre-filter and post-filter that can be used to further frequency-shape the error spectrum. One main result of this chapter is that FRS has a strong duality with classical quantization theory. In addition, FRS is shown to improve performance over SRS because of the ability to pre-shape the input.

In Chapters 5 through 7, we discuss randomized multiplier-less filtering. Chapter 5 introduces randomized multiplier-less filtering, discussing their benefit over static filters, potential applications, and our particular model. As illustrated in the roadmap, there are a number of models for randomized multiplier-less filtering – in different filter structures and different multiplier-less coefficients. In this thesis, we focus on the randomization of binary coefficients in the Direct Form I FIR filter structure. We denote this as BRF for short. As discussed in Chapter 5, the model is a vector extension of the randomized sampling model of Chapter 2.

Two forms of BRF are developed in this thesis, standard and oversampled. In Chapter 6 we present memoryless standard BRF, where the taps are uncorrelated in time. The main result is that the noise due to memoryless BRF is white and can be reduced by designing the correlation between the taps appropriately. We develop design techniques to design optimal binary tap correlation. The development is similar to SRS but extended to a vector form. In Chapter 7 we develop oversampled BRF, a more advanced structure that incorporates rate-conversion. This structure improves the SNR of BRF significantly due to an averaging effect. We show that SNR can be made arbitrarily high with enough oversampling. As such, oversampled BRF could potentially be useful as a technique to implement general digital filters – as an alternative to the standard multiply and accumulate architecture.

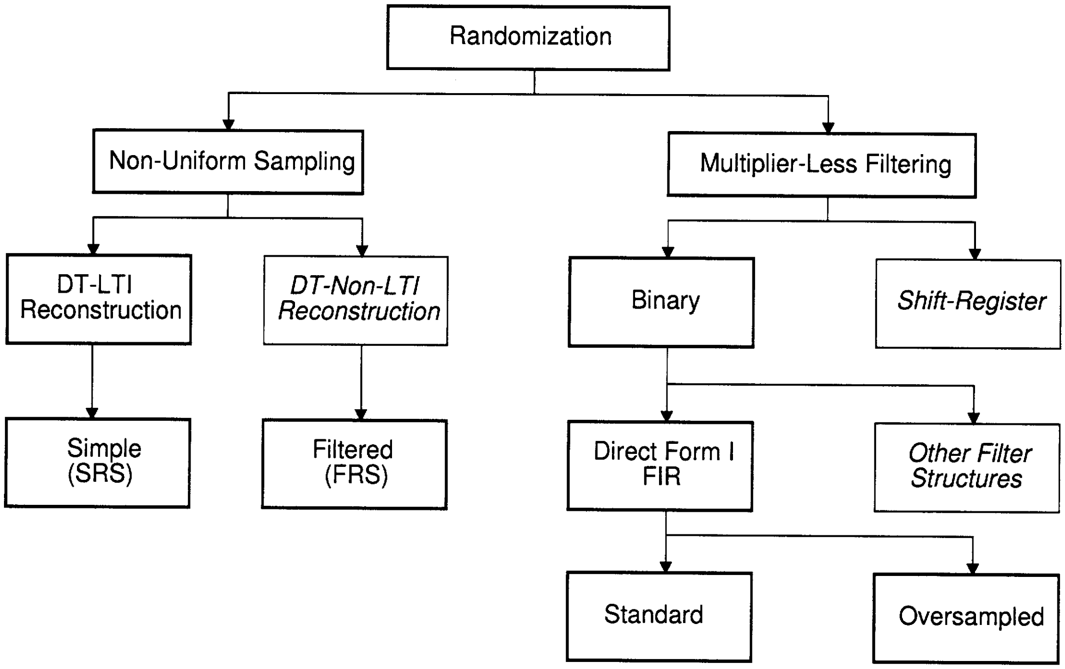


Figure 1-1. Thesis Roadmap

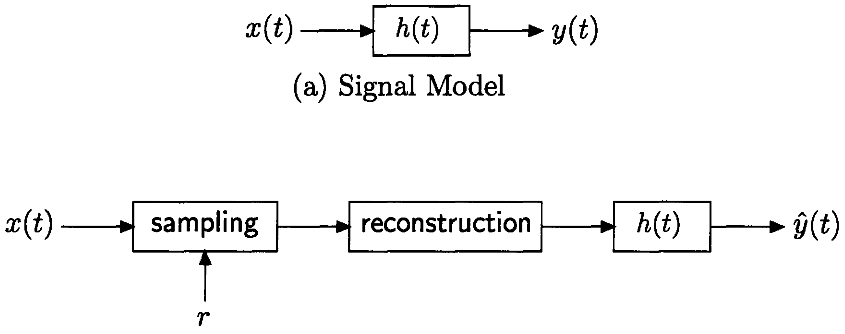
Chapter 2

Randomized Non-Uniform Sampling

This chapter motivates randomized non-uniform sampling as a method to mitigate aliasing. It introduces the basic randomized non-uniform sampling model which is an extension of that presented in Said and Oppenheim [28, 29]. Two structures are presented, simple randomized sampling, without pre/post filters, and filtered randomized sampling, which incorporates tunable pre/post filters.

2.1 Motivation

In this thesis, the goal is to create a faithful reproduction of a signal of interest (SOI) from a set of samples. We assume that the SOI, $y(t)$, is a filtered version of another signal $x(t)$. Figure 2-1(a) illustrates this signal model. $h(t)$ is a fixed LTI filter. The goal is to sample and reconstruct $x(t)$ in such a way that, after filtering with $h(t)$, the resulting reconstruction, $\hat{y}(t)$, is close to the desired signal, $y(t)$. Note that the objective is not the accuracy of the reconstruction of $x(t)$, but rather, the final signal, $\hat{y}(t)$, after filtering. Figure 2-1(b) illustrates this sampling model.



(a) Signal Model

(b) Sampling Model

Figure 2-1. Block diagrams for signal and sampling model. $h(t)$ is a fixed linear time-invariant filter. $x(t)$ is band-limited to π/T. The maximum rate of the sampling process is restricted to be $r_{\max} = 1/T_r < r$. The goal is to get the reconstructed output $\hat{y}(t)$ close to the desired output $y(t)$.

This generic model can represent a number of applications. For example, $x(t)$ can be light waves from a scene and $h(t)$ can be an LTI approximation for the response of the eye. $y(t)$ is the resulting image that would be perceived if there were no sampling. In this case, the goal is to create an imaging system that uses spatial samples to reconstruct the scene, $\hat{y}(t)$, such that the perceived visual distortion is minimized.

We assume $x(t)$ is band-limited to Ω_x, i.e. $X(j\Omega) = 0$ for $\Omega > \Omega_x$. If we could sample above the Nyquist rate at $\Omega_s > 2\Omega_x$, then $x(t)$ could be perfectly recovered from a set of uniform samples. Consequently, $y(t)$ can also be perfectly recovered using band-limited interpolation. Because of system constraints though, we assume that the maximum possible sampling rate is constrained such that $\Omega_s < 2\Omega_x$.

Such a sampling rate constraint could occur in various applications. For example, in wide-band surveillance, hardware constraints may constrain the sampling rate of an ADC below the Nyquist rate for signals of interest. In this context, it may be desirable to have a sampling strategy that algorithmically extends the effective bandwidth of the ADC. As another example, for low power applications such as sensor network nodes, sampling consumes a large portion of the power. In this case, it may be desirable to reduce the sampling rate, and correspondingly the power consumption, while still achieving a suitable SNR. Such a scenario may even occur in ray-traced computer graphics, where computational constraints may constrain the number of ray-traces that can be done. In this case, it may be desirable to have a ray-tracing procedure that reduces the amount of computation while still achieving a visually pleasing rendering of the scene.

In all of these scenarios, periodic sampling at rate $\Omega_s < 2\Omega_x$, aliases the input signal. Aliasing has two problems. First is that the user has little control over the coloration of the noise across the band. The aliases can occur at frequencies that contain critical information. In addition, the functional dependence of aliasing is problematic in certain applications, especially perceptual ones. For example, in imaging applications, aliasing manifests as moire patters, which a visually very disturbing.

The standard anti-aliasing technique is to low-pass filter the input to no more than half the sampling frequency before periodic sampling. This imposes a severe restriction on the range of frequencies that can be represented. It removes high frequency components that could contain critical information that needs to be preserved. In this sense, an anti-aliasing filter is not an ideal solution to the aliasing problem.

Furthermore, in certain applications, anti-aliasing filters are impossible to implement. For example, in ray-traced computer graphics there is no input signal before ray-tracing, only a mathematical model of the scene [10]. Traditional anti-aliasing requires first ray-tracing onto a dense grid, filtering, and then down-sampling the resulting output. Since each ray-trace requires a set of computations, this process is computationally intensive. In this scenario, it may be desirable to directly sample onto a lower rate grid without aliasing rather than traditional anti-aliasing.

Uniform sampling has its shortcomings when under-sampled. By contrast, as shown in the subsequent chapters, with randomized non-uniform sampling, the error due to under-sampling becomes shapeable noise that is uncorrelated with the input signal. In certain applications, particularly perceptual ones, this form of noise may be preferable to aliasing artifacts. In fact, a form of randomized sampling is used in the computer graphics community to anti-alias ray-traced images, [13, 10, 23]. In this thesis, we extend these randomized sampling techniques using non-white sampling processes, shaping filters, and other techniques to achieve better control over the error.

2.2 Randomized Sampling Model

Though there are a number of applications for randomized sampling, as documented in [4], in this thesis, we focus on using randomized sampling as a technique to reduce the number of samples to represent a signal. In particular, we do not address the question of performing operations, either linear or non-linear, on the randomized samples. Instead we focus on reconstruction. Our goal is create a faithful reproduction of the signal of interest (SOI) from a set of randomized samples.

There are numerous models for randomization of the sampling times. A number are presented in [4]. In this thesis, we model randomized non-uniform sampling as a downsampling process. The formulation follows the randomized down-sampling framework presented in [28, 29]. We extend the simple models of [28, 29] to more complex architectures that can further reduce the error.

The basic randomized sampling model is in continuous-time (CT). It is presented in Section 2.2.1. This section also introduces the two sampling structures, simple randomized sampling (SRS) and filtered randomized sampling (FRS). Under certain conditions, this CT model is equivalent to a purely discrete-time (DT) model. This conversion, in addition to certain subtleties regarding the transformation, are presented in Section 2.2.2.

2.2.1 Continuous-Time Model

Figure 2-2(a) illustrates a mathematical model of the basic randomized sampling architecture. The input $x(t)$ is assumed to be a band-limited wide-sense stationary (WSS) random process with auto-correlation, $R_{xx}(\tau)$, and maximum frequency Ω_x. In certain forms of randomized sampling, we assume that prior knowledge of $R_{xx}(\tau)$ is available.

In this model, samples cannot be taken at arbitrary times. They can only be taken at integer multiples of a high-rate grid with spacing T_{hr}. This can be interpreted as an ADC that is clocked with a high-rate stable digital clock, but does not sample on each rising-edge. Instead it samples only on rising-edges where it is enabled by a randomized signal. In Figure 2-2(a), this is modeled as an uniform sampling C/D, with rate $\Omega_{\text{hr}} = 2\pi/T_{\text{hr}}$, followed by multiplication with a discrete-time (DT) binary process, $r[n]$. The binary process, $r[n]$, can only take the values 1 and 0. When $r[n] = 1$, the sample is kept, when $r[n] = 0$, the sample is erased. Assuming that $r[n]$ is WSS with $E\{r[n]\} = \mu$, the effective average sampling rate can be mathematically expressed as:

$$\Omega_{\text{avg}} = \mu\Omega_{\text{hr}} \tag{2.1}$$

We assume that there is no aliasing on the high-rate grid, i.e. $\Omega_{\text{hr}} \geq 2\Omega_x$. Furthermore, we assume that the average sampling rate is fixed below the Nyquist rate for the signal $x(t)$, i.e. $\Omega_{\text{avg}} < 2\Omega_x$ and usually below the the Nyquist rate corresponding to the band-limit of the reconstruction filter, $\Omega_{\text{avg}} < 2\Omega_h$. Figure 2-3(a) illustrates these frequencies relative to each other in the frequency domain.

The DT signal, $q[n]$, after randomized sampling is at the rate Ω_{hr}. It is composed of sampled signal amplitudes and zeros. The non-zero amplitudes in $q[n]$ and their associated sampling times are assumed to be available upon reconstruction. Note that the time-indices do not have to be explicitly stored. Rather, if we assume that the sampling process, $r[n]$, is generated from a pseudo-random number generator, then only the state of the generator has to be stored. Using this state and an identical pseudo-random number generator, the non-uniform sampling times can be regenerated upon reconstruction, i.e. the samples can be placed at the correct positions on the high-rate grid.

The fixed LTI filter, $h(t)$, represents a frequency-dependent error weighting, denoting frequency bands of interest with high values and less important bands with low values. In certain contexts, $h(t)$ can be interpreted as a reconstruction filter. For example, in audio and imaging, $h(t)$ can represent a model of human perception. For simplicity, $h(t)$ is often assumed to be an ideal low-pass filter (LPF) with cutoff $\Omega_h < \Omega_x$. Though crude, the ideal (LPF) models the essential aspect of $h(t)$: the existence of a stop-band for which the frequency is significantly attenuated. Figure 2-3(a) illustrates Ω_h relative to the other frequency parameters.

This thesis develops two randomized sampling structures, simple randomized sampling (SRS) and filtered randomized sampling (FRS). In both we assume that the reconstruction is LTI. We briefly discuss potential extensions of randomized sampling using non-LTI reconstruction techniques in Section 3.6, but these methods are not explored in detail in this thesis.

SRS is illustrated in Fig.2-2(b). It does not incorporate any additional pre-filters or post-filters. The fixed filter $h(t)$ is used as a LTI reconstruction filter. There is a scaling by $1/\mu$ that compensates for the loss of energy due to sampling. There are two forms of SRS depending on the correlation of the sampling process. In white-SRS, the sampling process is constrained to be a white Bernoulli process. In frequency-shaped SRS, the correlation of the sampling process can be designed to shaped the error spectrum so there is less energy in the band of interest. SRS is developed in detail in Chapter 3

FRS is illustrated in Fig.2-2(c). It has an additional user-definable pre-filter and post-filter. The front-end sampling scheme consists of an LTI pre-emphasis filter, $g_1(t)$, followed by randomized sampling. The pre-emphasis filter $g_1(t)$ can be an arbitrary LTI filter, not necessarily an anti-aliasing filter. Upon reconstruction the non-uniform impulse train, $q(t)$, is filtered through an LTI filter, $g_2(t)$, to achieve an intermediate reconstruction, $w(t)$. This signal is filtered by $h(t)$ to produce the reconstruction $\hat{y}(t)$. There are four forms of FRS depending the correlation of the sampling process and a constraint on the filters. All four types are developed in Chapter 4.

2.2.2 Discrete-Time Model

The CT randomized sampling models of the previous section can be converted into equivalent DT models that are simpler to analyze. Figure 2-4(a) illustrates a manipulated form of the CT SRS architecture where all of the filtering is done in the DT domain. The C/D blocks represent perfect continuous-to-discrete transformations and the D/C blocks represent perfect discrete-to-continuous transformations, both at rate $\Omega_{\text{hr}} = 2\pi/T_{\text{hr}}$. Here, $h(t)$ is assumed to be a band-limiting filter, such that $H(j\Omega) = 0$ for $\Omega > \Omega_{\text{hr}}/2$. With this assumption, $h(t)$ can be split into an ideal low-pass filter (LPF) with cutoff $\Omega_{\text{hr}}/2$ and a DT filter $h[n]$ operating on a Ω_{hr}-rate sample stream. Combining the ideal LPF with the S/I constructs a D/C converter which cancels the C/D converter that follows.

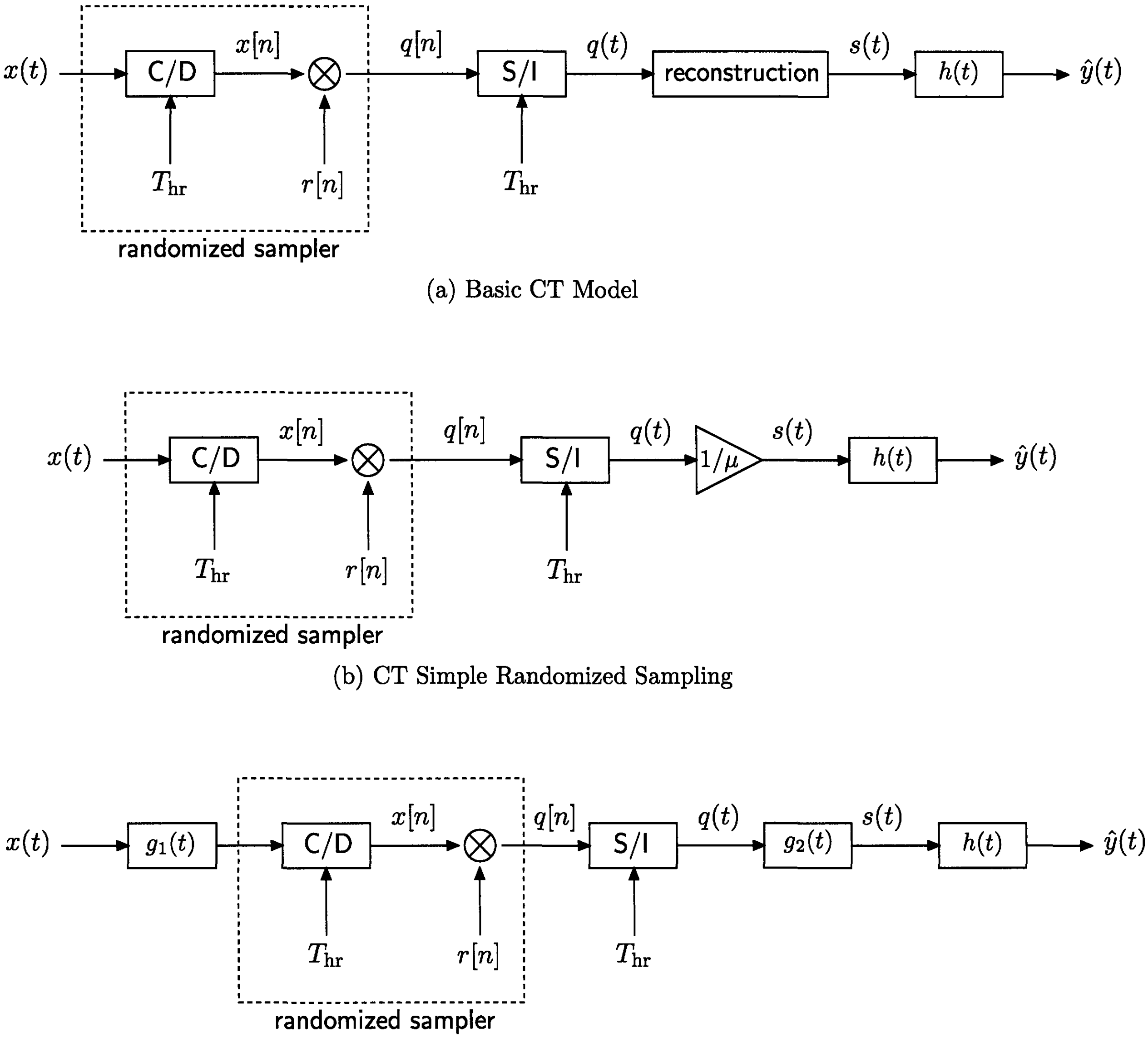
Combined with the assumption that $\Omega_{\text{hr}} \geq 2\Omega_x$, the CT SRS model can be transformed into the DT SRS system illustrated in Figure 2-4(b). Figure 2-3(b) illustrates the relative placement of the important DT frequency parameters. The CT to DT frequency mapping is:

$$\omega = \Omega T_{\text{hr}} \tag{2.2}$$

In the discrete-time formulation, the Ω_{avg} constraint from continuous-time becomes a constraint on the mean: $E\{r[n]\} = \mu = \Omega_{\text{avg}}/\Omega_{\text{hr}}$.

An analogous transformation can be done for FRS. In this case, $g_1(t)$ and $g_2(t)$ are assumed to be band-limited filters, with $G_1(j\Omega) = G_2(j\Omega) = 0$ for $\Omega > \Omega_{\text{hr}}/2$. As such, both have equivalent DT filter representations, $g_1[n]$, $g_2[n]$, on a Ω_{hr}-rate sample stream. Figure 2-4(c) illustrates the discrete-time FRS system. In the remainder of this thesis, analysis is done primarily on these discrete-time models.

There is an important subtlety associated with the DT transformation. In continuous-time the average sampling rate is fixed to Ω_{avg}. As expressed in Eqn.(2.1), this average rate is a function of both Ω_{hr} and the randomized down-sampling rate μ. There are many Ω_{hr} and μ pairs that can achieve a fixed Ω_{avg}. All pairs are not equal though. As shown in Chapter 3, for LTI reconstruction, the optimal operating point is $\Omega_{\text{hr}} = 2\Omega_x$, i.e. the lowest possible value without aliasing. This high-rate grid has the least mismatch with LTI reconstruction.

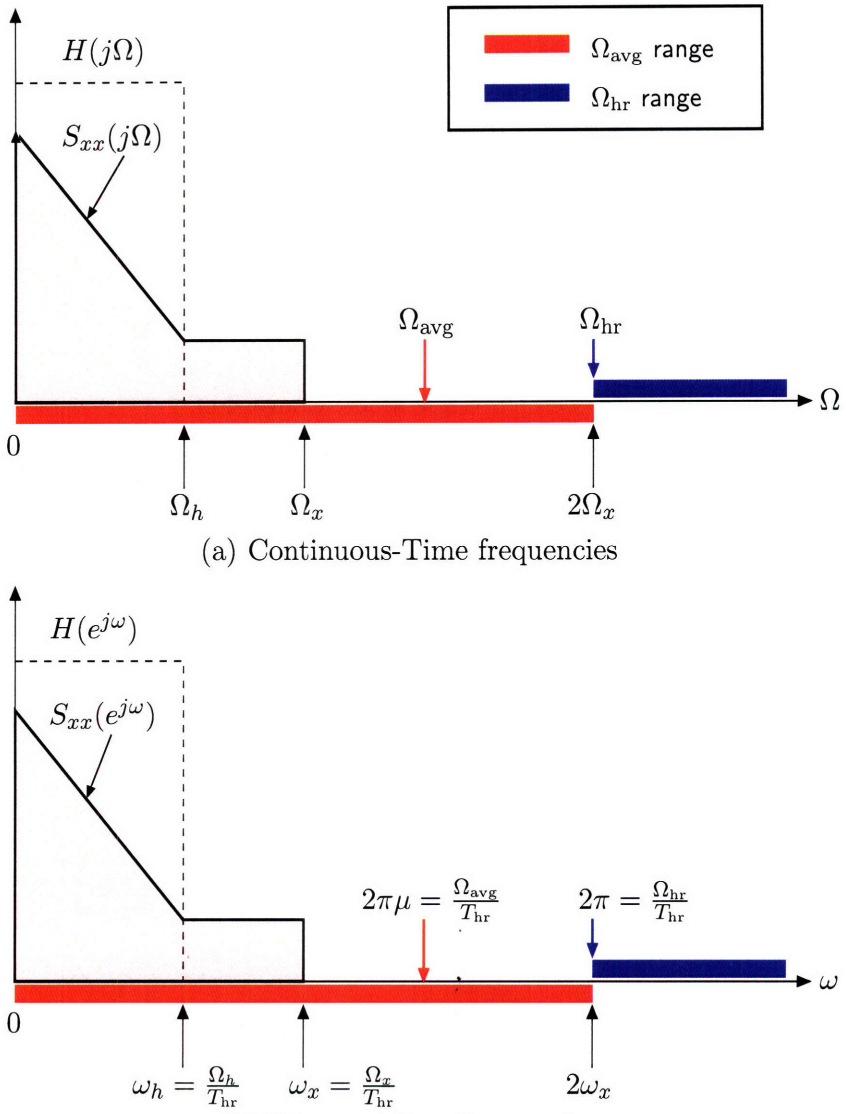


(a) Basic CT Model

(b) CT Simple Randomized Sampling

(c) CT Filtered Randomized Sampling

Figure 2-2. Block diagrams for continuous-time randomized sampling models studied in this thesis.

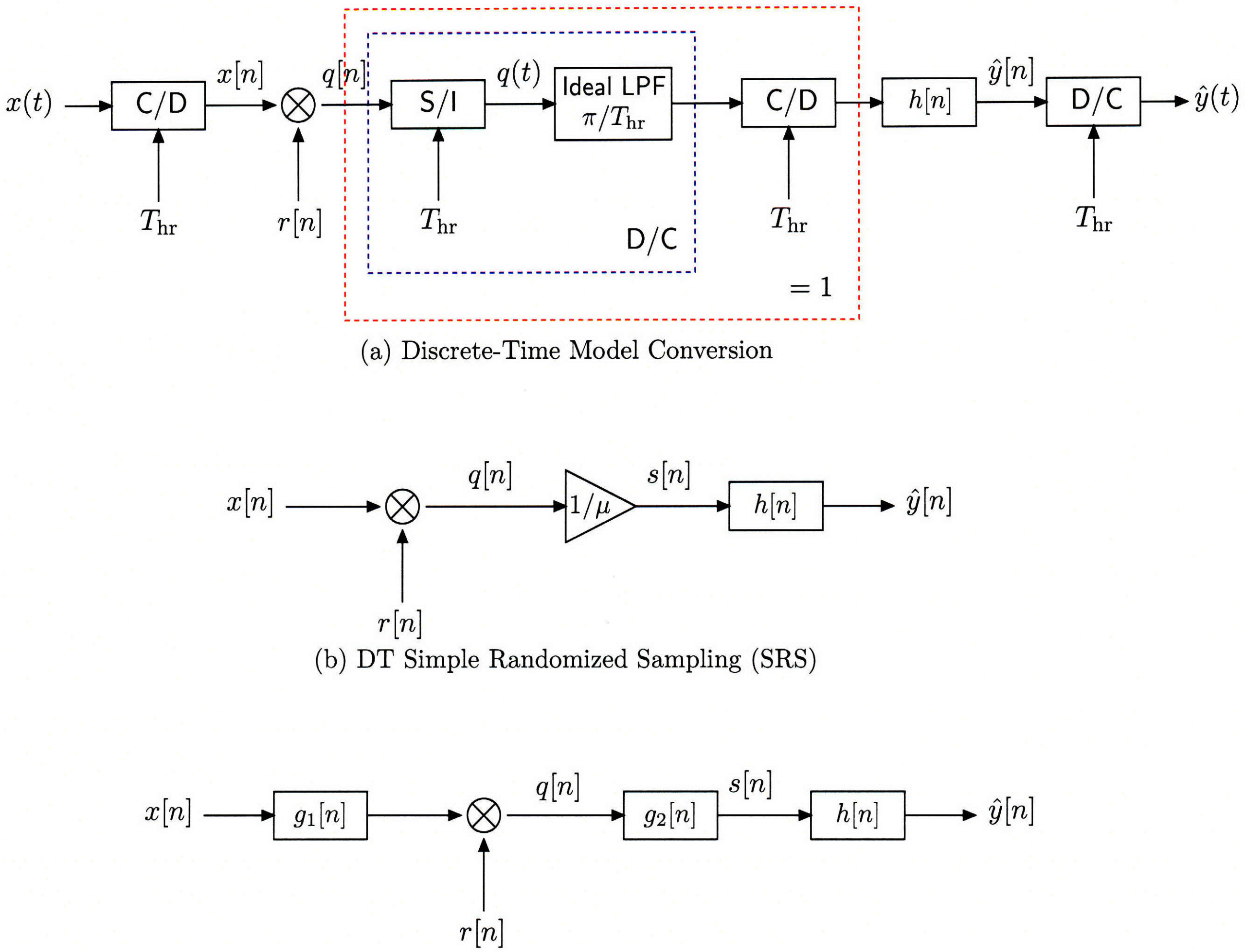


$H(j\Omega)$
$S_{xx}(j\Omega)$
🟥 Ω_{avg} range
🟦 Ω_{hr} range
Ω_{avg}
Ω_{hr}
Ω
0
Ω_h
Ω_x
$2\Omega_x$
$H(e^{j\omega})$
$S_{xx}(e^{j\omega})$
$2\pi\mu = \frac{\Omega_{\text{avg}}}{T_{\text{hr}}}$
$2\pi = \frac{\Omega_{\text{hr}}}{T_{\text{hr}}}$
ω
0
$\omega_h = \frac{\Omega_h}{T_{\text{hr}}}$
$\omega_x = \frac{\Omega_x}{T_{\text{hr}}}$
$2\omega_x$

(a) Continuous-Time frequencies

(b) Discrete-Time frequencies

Figure 2-3. Important frequency parameters in randomized sampling. (a) illustrates the CT frequency parameters. The blue area denotes the range of possible values for Ω_{hr}. The red area denotes the possible values for Ω_{avg}. (b) illustrates the DT frequency parameters when sampled at rate T_{hr}.



(a) Discrete-Time Model Conversion

(b) DT Simple Randomized Sampling (SRS)

(c) DT Filtered Randomized Sampling (FRS)

Figure 2-4. Block diagrams for discrete-time randomized sampling models studied in this thesis.

Chapter 3

Simple Randomized Sampling

This chapter develops simple randomized sampling (SRS). Two forms of SRS, white SRS and frequency-shaped SRS, are discussed. Section 3.1 presents the basic properties of SRS and its design. Section 3.2 develops white SRS, where the sampling process is constrained to be uncorrelated in time. Section 3.3 presents frequency-shaped SRS, where the correlation of the sampling process can be used to frequency-shape the error spectrum. It presents a parametric solution and develops a performance bound. Section 3.4 does a detailed error analysis of SRS. The theoretical predictions are validated with numerical experiments in Section 3.5.

3.1 Problem Statement

The upper-branch of Figure 3.1(a) depicts the continuous-time SRS model. The lower-branch illustrates the desired output $y(t)$. The model is drawn in its simplified form with the filter $h[n]$ defined in discrete-time. The LTI filter $h[n]$ is assumed to be fixed and known. The input $x(t)$ is assumed to be a band-limited, wide-sense stationary (WSS) process with auto-correlation $R_{xx}(\tau)$ and maximum frequency Ω_x. The high-rate sampling frequency is constrained to be above the Nyquist rate, $\Omega_{\text{hr}} \geq 2\Omega_x$. There is no aliasing on the high-rate grid. As mentioned in Chapter 2, Ω_{hr} is a parameter that can be changed. Its optimal choice is discussed in Section 3.4.2. For now, we assume it is fixed to some specified value.

The average sampling rate, Ω_{avg}, is fixed to a value below the Nyquist rate, i.e. $\Omega_{\text{avg}} < 2\Omega_x$. The fixed Ω_{hr} and Ω_{avg} fixes the mean of the discrete-time sampling process to:

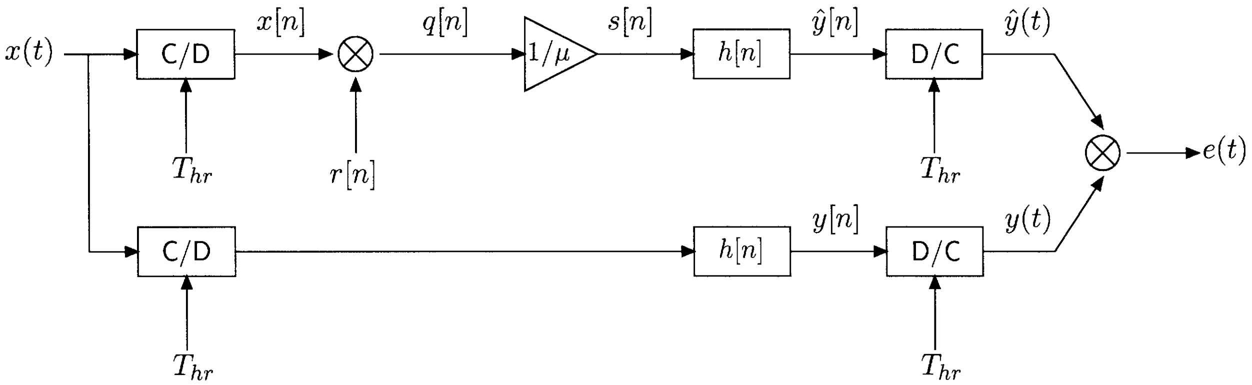
$$E\{r[n]\} = \mu = \frac{\Omega_{\text{avg}}}{\Omega_{\text{hr}}} \tag{3.1}$$

The input process $x(t)$ and the sampling process $r[n]$ are assumed to be independent. The only tunable facet of the system is the correlation of the sampling process.

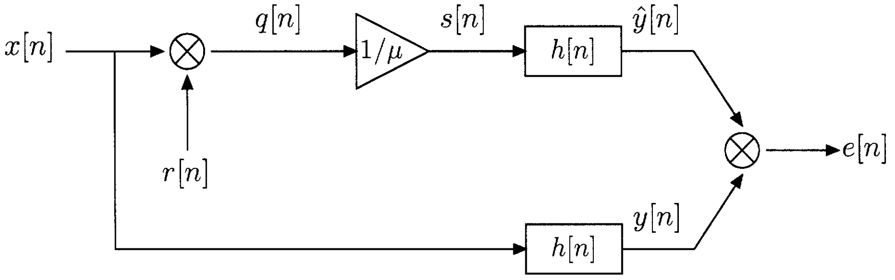
As noted in Chapter 2, the CT SRS problem can be recast in DT under these conditions. The DT SRS model is illustrated in the upper-branch of Figure 3.1(b). The desired output, $y[n]$, is illustrated as output of the lower-branch, i.e. $y[n] = x[n] * h[n]$. The goal in DT SRS is to design the WSS binary process, $r[n]$, subject to the fixed μ from Eqn(3.1), such that the MSE error is minimized:

$$\mathcal{E} = E\left\{e^2[n]\right\} = E\left\{(y[n] - \hat{y}[n])^2\right\} \tag{3.2}$$

We focus on this DT formulation, with a fixed Ω_{hr}, Ω_{avg}, and μ, for the remainder of this



CT Simple Randomized Sampling (CT-SRS)



DT Simple Randomized Sampling (SRS)

section.

The sampling process $r[n]$ is a WSS binary process with fixed mean μ. As such, it has a first-order probability mass function:

$$p_r(r[n]) = \begin{cases} \mu & \text{, for } r[n] = 1 \\ 1-\mu & \text{, for } r[n] = 0 \\ 0 & \text{, otherwise} \end{cases} \tag{3.3}$$

where $0 \leq \mu \leq 1$. The process can be decomposed into the sum of its mean and a zero-mean process, $\tilde{r}[n]$:

$$r[n] = \mu + \tilde{r}[n] \tag{3.4}$$

where $\tilde{r}[n] = \{1-\mu, -\mu\}$. Using this decomposition, the reconstruction $\hat{y}[n]$ can be expressed as the sum of the desired signal $y[n]$ and an error signal $e[n]$:

$$\begin{aligned} \hat{y}[n] &= \frac{1}{\mu} h[n] * q[n] \\ &= y[n] + \underbrace{\frac{1}{\mu} h[n] * (x[n]\tilde{r}[n])}_{e[n]} \end{aligned} \tag{3.5}$$

The error signal can be expanded as the sum:

$$e[n] = \frac{1}{\mu} \sum_{k=-\infty}^{\infty} x[k]\tilde{r}[k]h[n-k] \tag{3.6}$$

The error is unbiased and uncorrelated with the input. The bias can be found directly by linearity of expectation and the fact that $\tilde{r}[n]$ and $x[n]$ are independent processes. Mathematically:

$$\begin{aligned} E\{e[n]\} &= \frac{1}{\mu}E\left\{\sum_{k=-\infty}^{\infty} x[k]\tilde{r}[k]h[n+m-k]\right\} \\ &= \frac{1}{\mu}\sum_{m=-\infty}^{\infty} E\{x[k]\}\underbrace{E\{\tilde{r}[k]\}}_{=0}h[n+m-k] = 0 \end{aligned} \tag{3.7}$$

The error can be shown to be uncorrelated with the input directly:

$$\begin{aligned} E\{e[n+m]x[n]\} &= \frac{1}{\mu}E\left\{\sum_{k=-\infty}^{\infty} x[k]\tilde{r}[k]h[n+m-k]x[n]\right\} \\ &= \frac{1}{\mu}\sum_{m=-\infty}^{\infty} E\{x[k]x[n]\}\underbrace{E\{\tilde{r}[k]\}}_{=0}h[n+m-k] = 0 \end{aligned} \tag{3.8}$$

This uncorrelated error may be less disturbing in certain applications, especially perceptual ones, as compared to aliasing.

Since $r[n]$ and $x[n]$ are independent, the auto-covariance of the error before filtering, $w[n] = s[n] - x[n] = \frac{1}{\mu}x[n]\tilde{r}[n]$, can be expressed as:

$$\begin{aligned} R_{ww}[m] = E\{w[n+m]w[n]\} &= \frac{1}{\mu^2}E\{x[n+m]x[n]\}E\{\tilde{r}[n+m]\tilde{r}[m]\} \\ &= \frac{1}{\mu^2}R_{xx}[m]K_{rr}[m] \end{aligned} \tag{3.9}$$

In the frequency-domain, this can be expressed as a circular convolution:

$$S_{ww}(e^{j\omega}) = \frac{1}{\mu^2}\int_{-\pi}^{\pi} S_{xx}(e^{j\omega})\Phi_{rr}(e^{j(\omega-\theta)})\frac{d\theta}{2\pi} \tag{3.10}$$

where $\Phi_{rr}(e^{j\omega})$ is the auto-covariance spectrum of $r[n]$. Since $e[n] = h[n] * w[n]$, the power-spectrum of the error can be expressed as:

$$S_{ee}(e^{j\omega}) = |H(e^{j\omega})|^2\left\{\frac{1}{\mu^2}\int_{-\pi}^{\pi} S_{xx}(e^{j\theta})\Phi_{rr}(e^{j(\omega-\theta)})\frac{d\theta}{2\pi}\right\} \tag{3.11}$$

The MSE can be expressed in the frequency-domain by integrating Eqn.(3.11):

$$\begin{aligned} \mathcal{E} &= \int_{-\pi}^{\pi} S_{ee}(e^{j\omega})\frac{d\omega}{2\pi} \\ &= \frac{1}{\mu^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} |H(e^{j\omega})|^2 S_{xx}(e^{j\theta})\Phi_{rr}(e^{j(\omega-\theta)})\frac{d\theta}{2\pi}\frac{d\omega}{2\pi} \end{aligned} \tag{3.12}$$

There are two forms of SRS depending on the correlation of the sampling process. In white-SRS, the sampling process is constrained to be a Bernoulli process. Section 3.2

develops white-SRS in detail. The sampling process can be correlated in time to shape the error out of the pass-band of $h[n]$ though. We denote this as frequency-shaped SRS. It is developed in Section 3.3.

3.2 White SRS

In this section we develop white SRS, where the process $r[n]$ is restricted to be uncorrelated in time. It was first developed in Said and Oppenheim in [28]. White-SRS is used in practice in the computer graphics [13, 10, 23] to anti-alias ray-tracing. There is is often denoted as stochastic ray-tracing or randomized ray-tracing.

Since $r[n]$ is a binary process, fixing $E\{r[n]\} = \mu$ also fixes its first-order variance, σ_r^2. Mathematically:

$$\sigma_r^2 = \mu(1-\mu) \tag{3.13}$$

Consequently, the auto-covariance is

$$K_{rr}[m] = \sigma_r^2 \delta[m] \tag{3.14}$$

Combining Eqn. (3.13), (3.14) and Eqn.(3.9), we can express $K_{ww}[m]$ as:

$$\begin{aligned} K_{ww}[m] &= \frac{\sigma_r^2}{\mu^2} R_{xx}[m]\delta[m] \\ &= \left(\frac{1}{\mu} - 1\right) R_{xx}[0]\delta[m] \end{aligned} \tag{3.15}$$

which implies that $w[n]$ is white. Substituting Eqn.(3.15) into Eqn.(3.11), it follows that the error spectrum $S_{ee}(e^{j\omega})$ has the same shape as the filter $H(e^{j\omega})$:

$$S_{ee}(e^{j\omega}) = \left(\frac{1}{\mu} - 1\right) R_{xx}[0]|H(e^{j\omega})|^2 \tag{3.16}$$

There are no tunable parameters to optimize white SRS. Expanding $R_{xx}[0]$, the white SRS MSE can be expressed as the integral of the error spectrum:

$$\mathcal{E}_{\text{w}} = \left(\frac{1}{\mu} - 1\right) \left(\int_{-\pi}^{\pi} S_{xx}(e^{j\theta}) \frac{d\theta}{2\pi}\right) \left(\int_{-\pi}^{\pi} |H(e^{j\omega})|^2 \frac{d\omega}{2\pi}\right) \tag{3.17}$$

Further error analysis of white SRS is done in Section 3.4.1. White-SRS is the simplest form of SRS, serving as a performance baseline for the other more advanced techniques developed in this thesis.

3.3 Frequency-Shaped SRS

In frequency-shaped SRS, the time correlation of the sampling process can be used to frequency shape the error spectrum. This can be used to reduce the in-band error and improve the SNR. This section discusses the design of frequency-shaped SRS in detail.

3.3.1 Design Problem

By designing the sampling auto-covariance, $\Phi_{rr}(e^{j\omega})$, the error spectrum can be shaped to minimize the energy in the passband of $h[n]$. We can formally express the design of $\Phi_{rr}(e^{j\omega})$ as an optimization. The first step is to re-express Eqn.(3.12) by expanding the convolution integral and swapping the order of integration:

$$\begin{aligned}\mathcal{E} &= \frac{1}{\mu^2}\int_{\omega=-\pi}^{\pi}\left(\int_{\theta=-\pi}^{\pi}\Phi_{rr}(e^{j\theta})S_{xx}(e^{j(\omega-\theta)})\frac{d\theta}{2\pi}\right)|H(e^{j\omega})|^2\frac{d\omega}{2\pi}\\ &= \frac{1}{\mu^2}\int_{\theta=-\pi}^{\pi}\Phi_{rr}(e^{j\theta})\underbrace{\left(\int_{\omega=-\pi}^{\pi}S_{xx}(e^{j(\omega-\theta)})|H(e^{j\omega})|^2\frac{d\omega}{2\pi}\right)}_{F(e^{j\theta})}\frac{d\theta}{2\pi} \qquad (3.18)\end{aligned}$$

We define $F(e^{j\theta})$ as the convolution of $|H(e^{j\omega})|^2$ with a frequency reversed version of $S_{xx}(e^{j\omega})$. Since $S_{xx}(e^{j\omega})$ is a power-spectrum, it is even and symmetric, with $S_{xx}(e^{j\omega}) = S_{xx}(e^{-j\omega})$. Thus,

$$\begin{aligned}F(e^{j\theta}) &= \int_{\omega=-\pi}^{\pi}|H(e^{j\omega})|^2 S_{xx}(e^{-j(\theta-\omega)})\frac{d\omega}{2\pi}\\ &= \int_{\omega=-\pi}^{\pi}|H(e^{j\omega})|^2 S_{xx}(e^{j(\theta-\omega)})\frac{d\omega}{2\pi} \qquad (3.19)\end{aligned}$$

Combining Eqn.(3.19) with Eqn.(3.18), the MSE can be expressed as:

$$\mathcal{E} = \frac{1}{\mu^2}\int_{-\pi}^{\pi}\Phi_{rr}(e^{j\omega})F(e^{j\omega})\frac{d\omega}{2\pi} \qquad (3.20)$$

Mathematically, the goal in frequency-shaped SRS is to design $\Phi_{rr}(e^{j\omega})$ such that this objective function is minimized. There are two important constraints on $\Phi_{rr}(e^{j\omega})$. First, since μ is fixed and $r[n]$ is a binary process, the first-order variance is also fixed. Consequently, the area under $\Phi_{rr}(e^{j\omega})$ is constrained:

$$\int_{-\pi}^{\pi}\Phi_{rr}(e^{j\omega})\frac{d\omega}{2\pi} = \sigma_r^2 = \mu(1-\mu) \qquad (3.21)$$

Secondly, and more importantly, $\Phi_{rr}(e^{j\omega})$ and μ must be achievable using a binary process, i.e. there must exist a stationary binary process with mean μ and auto-covariance spectrum $\Phi_{rr}(e^{j\omega})$. This is a problematic constraint, because not all valid auto-covariance spectra are achievable by binary processes [11, 19, 5]. The set of achievable spectra has been studied in [11, 19]. We denote this set at $\mathcal{B}(\mu)$. Unfortunately, this set is not tractable for optimization. Combining the two constraints and the objective function Eqn.(3.20), the design of $\Phi_{rr}(e^{j\omega})$ can be posed formally as the optimization:

$$\boxed{\begin{array}{rl}\underset{\Phi_{rr}(e^{j\omega})}{\text{minimize}} & \frac{1}{\mu^2}\int_{-\pi}^{\pi}\Phi_{rr}(e^{j\omega})F(e^{j\omega})\frac{d\omega}{2\pi}\\ \text{subject to} & \int_{-\pi}^{\pi}\Phi_{rr}(e^{j\omega})\frac{d\omega}{2\pi} = \mu(1-\mu)\\ & \Phi_{rr}(e^{j\omega}) \in \mathcal{B}(\mu)\end{array}} \qquad (3.22)$$

3.3.2 Relaxation Bound

We can find a performance bound for frequency-shaped SRS by relaxing the binary achievability constraint in Eqn.(3.22) and replacing it with the constraint that $\Phi_{rr}(e^{j\omega})$ must be a valid auto-covariance spectrum, i.e. positive semi-definite [26]. The relaxed optimization (3.22) can be expressed as,

$$\begin{array}{ll} \underset{\Phi_{rr}(e^{j\omega})}{\text{minimize}} & \frac{1}{\mu^2}\int_{-\pi}^{\pi}\Phi_{rr}(e^{j\omega})F(e^{j\omega})\frac{d\omega}{2\pi} \\ \text{subject to} & \frac{1}{2\pi}\int_{-\pi}^{\pi}\Phi_{rr}(e^{j\omega})\frac{d\omega}{2\pi} = \mu(1-\mu) \\ & \Phi_{rr}(e^{j\omega}) \geq 0 \end{array} \tag{3.23}$$

The solution to this relaxed optimization is a performance bound, because, as mentioned in the previous section, not all valid auto-covariance functions can be achieved using a binary random process. The optimal binary solution to frequency-shaped SRS can do no better than this relaxation bound.

In the optimization of Eqn.(3.23), the variance constrains the amount of energy $\Phi_{rr}(e^{j\omega})$ must have. In the objective function, $F(e^{j\omega})$ can be interpreted as an shaping factor on $\Phi_{rr}(e^{j\omega})$. Intuitively, the solution to the relaxed optimization is to put all the energy of $\Phi_{rr}(e^{j\omega})$ where the shaping by $F(e^{j\omega})$ will be minimal. Mathematically, define

$$\omega_0 = \arg\min_{\omega} F(e^{j\omega}) \tag{3.24}$$

The optimal relaxed input covariance spectrum, $\Phi^{\star}_{rr}(e^{j\omega})$, is then:

$$\Phi^{\star}_{rr}(e^{j\omega}) = 2\pi\sigma_r^2\left(\frac{1}{2}\delta(\omega-\omega_0)+\frac{1}{2}\delta(\omega+\omega_0)\right) \tag{3.25}$$

If $F(e^{j\omega})$ has multiple global minima then there are many solutions to the relaxed optimization. If $F(e^{j\omega})$ has zeros, then the MSE can be made zero by putting all the energy of $\Phi_{rr}(e^{j\omega})$ at this zero. The next section gives a more formal argument for this result.

Substituting Eqn.(3.25) back into the objective function Eqn.(3.23) the relaxation bound on the MSE can be expressed as:

$$\begin{aligned} \mathcal{E}_{\mathrm{r}} &= \frac{F(e^{j\omega_0})}{2\pi}\left(\frac{1}{\mu}-1\right) \\ &= \frac{1}{2\pi}\left(\min_{\omega}\int_{-\pi}^{\pi}|H(e^{j\theta})|^2 S_{xx}(e^{j(\omega-\theta)})d\theta\right)\left(\frac{1}{\mu}-1\right) \end{aligned} \tag{3.26}$$

Formal Argument for Relaxed Solution

In this section we develop a more formal argument for the optimal relaxed solution. Assume, without loss of generality, that $F(e^{j\omega})$ has only one global minimum at $\omega = \omega_0$. For multiple minima, the argument below can be generalized in a straightforward manner. We proceed by contradiction. Assume that the optimal relaxed solution $\Phi_{rr}(e^{j\omega})$ is not Eqn.(3.25),

$$\Phi^{\star}_{rr}(e^{j\omega}) \neq 2\pi\sigma_r^2\left(\frac{1}{2}\delta(\omega-\omega_0)+\frac{1}{2}\delta(\omega+\omega_0)\right) \tag{3.27}$$

This implies the existence of an open interval (ω_1, ω_2) with $\omega_0 \notin (\omega_1, \omega_2)$ and $\omega_1, \omega_2 > 0$, for which,

$$\int_{\omega_1}^{\omega_2} \Phi^{\star}_{rr}(e^{j\omega})d\omega = \alpha > 0 \tag{3.28}$$

Since $\Phi^{\star}_{rr}(e^{j\omega})$ is a power spectrum that is even and symmetric, the same is true for an open interval $(-\omega_1, -\omega_2)$, where $-\omega_0 \notin (-\omega_1, -\omega_2)$. Construct a function $\Phi^{\star\star}_{rr}(\omega)$, of the form,

$$\Phi^{\star\star}_{rr}(e^{j\omega}) = \begin{cases} \Phi^{\star}_{rr}(e^{j\omega}) + \alpha\left\{\delta(\omega-\omega_0) + \delta(\omega+\omega_0)\right\} & , \text{for } \omega \notin \{(\omega_1,\omega_2) \cup (-\omega_1,-\omega_2)\} \\ 0 & , \text{for } \omega \in \{(\omega_1,\omega_2) \cup (-\omega_1,-\omega_2)\} \end{cases} \tag{3.29}$$

where the energy in the intervals (ω_1, ω_2) and $(-\omega_1, -\omega_2)$ has been moved to the points ω_0 and $-\omega_0$, respectively. As before, $\omega_0 = \arg\min_\omega F(e^{j\omega})$. The function $\Phi^{\star\star}_{rr}(e^{j\omega})$ satisfies all of the constraints, i.e. it is a valid power spectrum and integrates to the same value as $\Phi^{\star}_{rr}(e^{j\omega})$.

We now show that $\Phi^{\star\star}_{rr}(e^{j\omega})$ has a smaller MSE than $\Phi^{\star}_{rr}(e^{j\omega})$, contradicting the assumption that $\Phi^{\star}_{rr}(e^{j\omega})$ is the optimal relaxed solution. The integral of Eqn.(**??**) can be separated into the sum of three integrals over disjoint intervals,

$$\int_0^{\pi} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega = \int_0^{\omega_1} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega + \int_{\omega_1}^{\omega_2} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega + \int_{\omega_2}^{\pi} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega \tag{3.30}$$

where we have dropped the leading scale factor terms since they are not relevant to the proof. Similarly, the integral of $F(e^{j\omega})$ with $\Phi^{\star\star}_{rr}(e^{j\omega})$ can be separated into the sum of integrals over the same three disjoint intervals. Substituting Eqn.(3.29) for $\Phi^{\star\star}_{rr}(e^{j\omega})$, the resulting integral can be simplified as,

$$\begin{aligned}\int_0^{\pi} \Phi^{\star\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega &= \int_0^{\omega_1} \Phi^{\star\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega + \underbrace{\int_{\omega_1}^{\omega_2} \Phi^{\star\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega}_{=0} + \int_{\omega_2}^{\pi} \Phi^{\star\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega \\ &= \int_0^{\omega_1} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega + \int_{\omega_2}^{\pi} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega + \int_0^{\pi} \alpha\delta(\omega-\omega_0)F(e^{j\omega})d\omega \\ &= \int_0^{\omega_1} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega + \int_{\omega_2}^{\pi} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega + \alpha F(e^{j\omega_0})\end{aligned} \tag{3.31}$$

The difference between Eqns. (3.30) and (3.31) is,

$$\int_0^{\pi} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega - \int_0^{\pi} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega = \int_{\omega_1}^{\omega_2} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega - \alpha F(e^{j\omega_0}) \tag{3.32}$$

If this difference is strictly greater than zero then $\Phi^{\star\star}_{rr}(e^{j\omega})$ has a lower MSE than $\Phi^{\star}_{rr}(e^{j\omega})$. Define ω_3 as,

$$\omega_3 = \arg \min_{\omega\in(\omega_1,\omega_2)} F(e^{j\omega}) \tag{3.33}$$

Since both $F(e^{j\omega})$ and $\Phi^{\star}_{rr}(e^{j\omega})$ are positive, we can bound the integral,

$$\begin{aligned}\int_{\omega_1}^{\omega_2} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega &\geq \int_{\omega_1}^{\omega_2} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega_3})d\omega \\ &= F(e^{j\omega_3}) \int_{\omega_1}^{\omega_2} \Phi^{\star}_{rr}(e^{j\omega})d\omega \\ &= \alpha F(e^{j\omega_3}) \end{aligned} \tag{3.34}$$

Since we have assumed that $F(e^{j\omega})$ has a single global minimum as ω_0, this implies by definition that $F(e^{j\omega_3}) > F(e^{j\omega_0})$. Consequently,

$$\int_{\omega_1}^{\omega_2} \Phi^{\star}_{rr}(e^{j\omega})F(e^{j\omega})d\omega - \alpha F(e^{j\omega_0}) \geq \alpha F(e^{j\omega_3}) - \alpha F(e^{j\omega_0}) > 0 \tag{3.35}$$

This is a contradiction: a lower MSE can be achieved by moving the energy in the interval (ω_1, ω_2) to the point ω_0. Our initial assumption was therefore false. The optimal relaxed solution is therefore given by Eqn.(3.25).

Inachievability of Relaxed Solution

The optimal relaxed solution of Eqn.(3.25) cannot be achieved by a WSS binary process though. We can see this informally using the Einstein-Wiener-Khinchin theorem. Given a WSS random process $\tilde{r}[n]$, the Fourier transform squared at any specific frequency, ω, converges with probability one to a non-negative random-variable $|\tilde{R}(e^{j\omega})|^2 > 0$, [26]. The Einstein-Wiener-Khinchin theorem relates the expectation of this random variable with the value of the power-spectrum, [26]

$$\Phi_{rr}(e^{j\omega}) = E\{|\tilde{R}(e^{j\omega})|^2\} \tag{3.36}$$

Consequently, if $\Phi_{rr}(e^{j\omega}) = 0$ for some ω, this implies $E\{|R(e^{j\omega})|^2\} = 0$. Since $|R(e^{j\omega})|^2 \geq 0$, this means that the random variable $|R(e^{j\omega})|^2$ converges, with probability one, to a distribution that has mass only at 0. This implies that every realization of the random process, except a set of measure zero, has no spectral energy at ω. Since our optimal solution $\Phi^{\star}_{rr}(e^{j\omega})$ is zero for all ω other than ω_0, this implies that all realizations, with probability one, are sinusoids of the form:

$$\tilde{r}[n] = A\cos(\omega_0 n + \phi_1) + B\sin(\omega_0 n + \phi_2) \tag{3.37}$$

where ϕ_1 and ϕ_2 are random phases and A and B are constants such that $A^2 + B^2 = \sigma_r^2$. Such realizations, except for very special cases, are not WSS binary processes. Thus, in general, the relaxed solution cannot be achieved using a WSS binary process.

3.3.3 Parametric Frequency-Shaped SRS

As mentioned in Section 3.3.1, the set of covariance spectra achievable using a binary process has been characterized in [11, 19]. Unfortunately, this set is defined recursively and is not tractable for optimization in Eqn.(3.22). To make the optimization tractable, we use parametric models for $r[n]$. A desirable parametric model should satisfy three properties. First, the model should have the ability to constrain the mean of the binary process. Second, the covariance spectrum, $\Phi_{rr}(e^{j\omega})$, should have a tractable expression in terms of the parameters. In this way the parameters can be optimized for frequency-shaped SRS. Lastly, for practical purposes, the model should be amenable to hardware implementation.

Though there are numerous techniques to generate binary processes, most do not give a tractable expression for the resulting auto-covariance spectrum. We use Boufounous processes, first presented in [5], as our model. The Boufounos model generates binary processes with a fixed mean and auto-regressive (AR) spectrum. The basic Boufounos model is presented below for completeness, but the proofs are omitted. For a more detailed description, the reader is referred to the original paper [5].

In the Boufounos model, a binary process, $r[n]$, is generated iteratively from p previous samples, $r[n-1], r[n-2], \ldots, r[n-p]$ as follows,

1. The bias, $r_b[n]$, for the generation of $r[n]$ is computed according to the relationship:

$$r_b[n] = \mu + \sum_{k=1}^{p} a_k(r[n-k] - \mu) \tag{3.38}$$

   where μ is the desired mean of $r[n]$ and the a_k are parameters of the algorithm.

2. The sample $r[n]$ is randomly generated from a binary distribution biased by $r_b[n]$ as follows:

$$r[n] = \begin{cases} 1 & \text{with probability } r_b[n] \\ 0 & \text{with probability } 1 - r_b[n] \end{cases} \tag{3.39}$$

The resulting process is a p^{th} order Markov process, in which the $\{r[n-k], k = 1, \ldots, p\}$ determine the state at any given time n.

In steady-state, the binary process can be proved to be wide-sense stationary with mean μ. The auto-covariance spectrum can be shown to be auto-regressive, of the form:

$$\Phi_{rr}(e^{j\omega}) = \frac{A}{|H(e^{j\omega})|^2} = \frac{A}{|1 - \sum_{k=1}^{p} a_k e^{-j\omega k}|^2} \tag{3.40}$$

A is a scale factor that ensures that the variance constraint is satisfied. Mathematically it can be expressed as:

$$A = \mu(1-\mu)\left(\int_{-\pi}^{\pi} \frac{1}{|1 - \sum_{k=1}^{p} a_k e^{-j\omega k}|^2} \frac{d\omega}{2\pi}\right) \tag{3.41}$$

The binary process converges to this auto-covariance as long as the following constraint on the parameters a_k and μ are satisfied,

$$\left(\sum_{k=1}^{p} a_k - 1\right) > \frac{1}{|1-2\mu|}\left(\sum_{k=1}^{p} |a_k| - 1\right) \tag{3.42}$$

This constraint ensures that there is no overflow in Eqn.(6.63), i.e. the bias, $r_b[n]$ is bounded between 0 and 1, [5]. Note that this inequality is strict. The constraint is illustrated graphically in Figure 3-1 for $\mu < 1/2$. The axes are $\sum a_k$ and $\sum |a_k|$. The shaded area denotes the set of coefficients for which the algorithm is guaranteed not to overflow. As μ increases, the two constraints pivot around the point $(1, 1)$, as shown in the plot. The shaded area is maximized when $\mu = 1/2$. For $\mu > 1/2$ the constraints cross over each other, and the shaded area is identical to the shaded area for $1 - \mu$. Note that in the limit as $\mu \to 0$ or $\mu \to 1$ the shaded area converges toward the origin where $\sum a_k \to 0$ and $\sum |a_k| \to 0$. In these limits, $a_k \to 0$ and the Boufounos process approaches a Bernoulli process.

Since the $r[n-k]$ only take discrete binary values, the process can represented using a 2^p-state Markov chain. The state transition probabilities are,

$$P((r[n], \ldots, r[n-p+1]) | (r[n-1], \ldots, r[n-p])) = \begin{cases} \mu + \sum_{k=1}^{p} a_k (r[n-k] - \mu) & \text{if } r[n] = 1 \\ 1 - \left(\mu + \sum_{k=1}^{p} a_k (r[n-k] - \mu)\right) & \text{if } r[n] = 0 \end{cases} \tag{3.43}$$

which is equal to $r_b[n]$ and $(1 - r_b[n])$ if $r[n] = 1$ and 0, respectively. The transition probabilities are zero for all other state transitions. The strict inequality of Eqn.(3.42) ensures that all the probabilities in Eqn.(3.43) are strictly positive. This in turn ensures that any state can be reached with positive probability within p transitions from any other state. The Markov chain is thus ergodic with an unique stationary distribution [5]. Figure 3-2 illustrates the Markov chain for a two pole Boufounos process, i.e. $p = 2$. The states are all the length two binary sequences, i.e. $(r[n-2], r[n-1])$. The transition probabilities are labeled accordingly.

The Boufounos model is simple and readily amenable to optimization for use in frequency-shaped SRS. We could potentially achieve a larger set of binary processes by removing the auto-regressive constraint. As an alternative, we could develop binary processes generated by general, fully connected, Markov chains. Such a model could have a closed-form auto-covariance spectrum using the work of [14, 15]. This is beyond the scope of this thesis and we restrict ourselves to Boufounos processes.

By substituting Eqn.(3.40) into Eqn.(3.20), fixing the number of a_k parameters to p, and imposing the constraint Eqn.(3.42), the SRS design problem can be expressed as the optimization,

$$\boxed{\begin{aligned} \underset{a_k}{\text{minimize}} \quad & \tfrac{1}{\mu^2} \textstyle\int_{-\pi}^{\pi} F(e^{j\omega}) \frac{A}{|1-\sum_{k=1}^{p} a_k e^{-j\omega k}|^2} \frac{d\omega}{2\pi} \\ \text{subject to} \quad & A = \mu(1-\mu) \left(\int_{-\pi}^{\pi} \frac{1}{|1 - \sum_{k=1}^{p} a_k e^{-j\omega k}|^2} \frac{d\omega}{2\pi} \right) \\ & \left(\sum_{k=1}^{p} a_k - 1 \right) > \frac{1}{|1-2\mu|} \left(\sum_{k=1}^{p} |a_k| - 1 \right) \end{aligned}} \tag{3.44}$$

The binary compatibility is implicitly satisfied because the Boufounos model guarantees that a binary process with parameters μ and a_k exist.

Numerical optimization can be used to solve (3.44) for the optimal values of a_k. The optimization requires discretization of the continuous functions $F(e^{j\omega})$ and $\Phi_{rr}(e^{j\omega})$. For simplicity, we discretize onto a grid of uniformly spaced frequencies. Though inexact, if

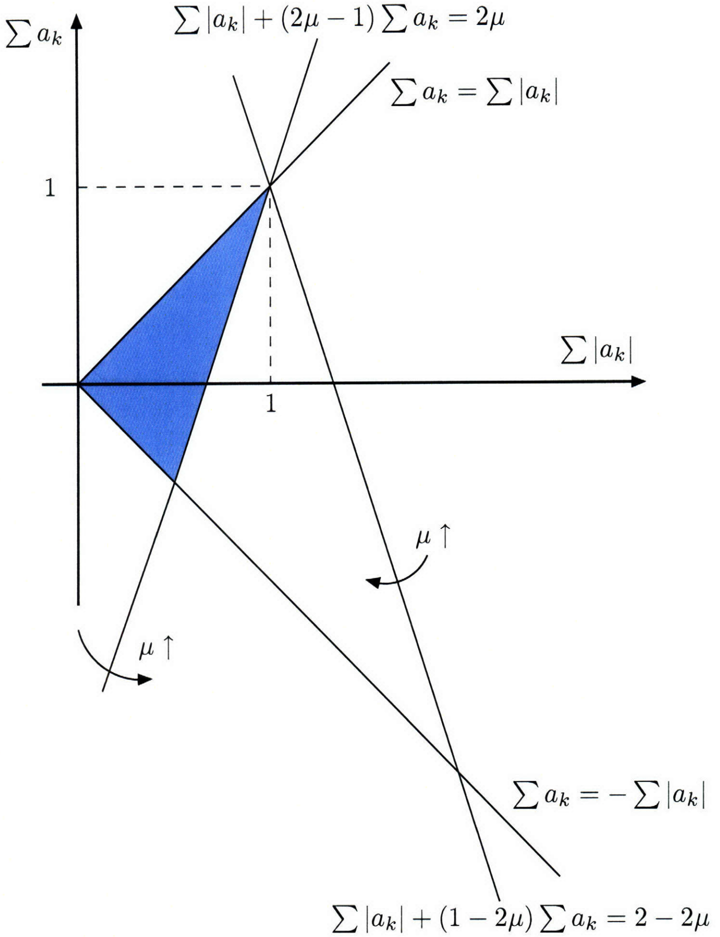


Figure 3-1. Coefficient space for $\mu < 1/2$ reproduced from [5]. The shaded area is the set of coefficients for which the algorithm is guaranteed not to overflow. As μ increases, the two constraints pivot around the point $(1, 1)$, as shown in the plot.

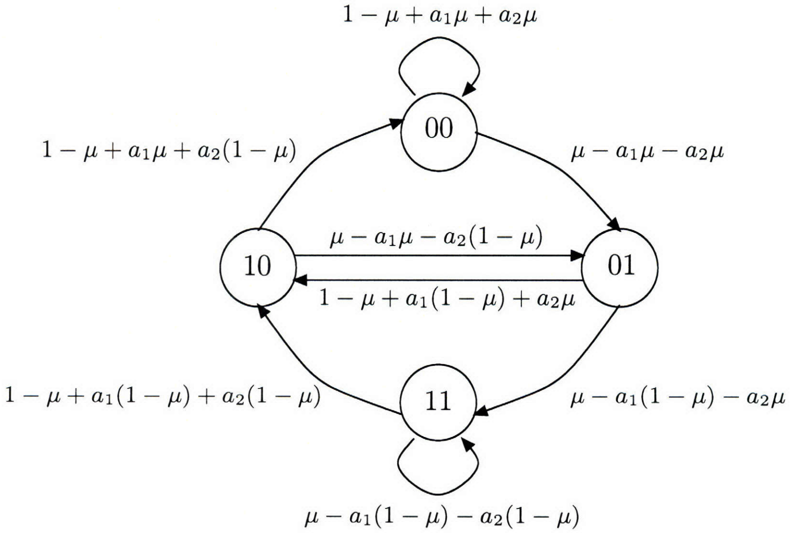

$1 - \mu + a_1\mu + a_2\mu$
00
$1 - \mu + a_1\mu + a_2(1 - \mu)$
$\mu - a_1\mu - a_2\mu$
$\mu - a_1\mu - a_2(1 - \mu)$
10
01
$1 - \mu + a_1(1 - \mu) + a_2\mu$
$1 - \mu + a_1(1 - \mu) + a_2(1 - \mu)$
11
$\mu - a_1(1 - \mu) - a_2\mu$
$\mu - a_1(1 - \mu) - a_2(1 - \mu)$

Figure 3-2. Markov chain for binary-AR model with $p = 2$. The states are $(r[n-2], r[n-1])$.

done on a dense enough grid, the solution is assumed to be close to the desired continuous optimization. In addition, since almost all numerical solvers require closed constraint sets, with non-strict inequalities, a parameter ϵ is added to the constraint to ensure it is met strictly. Mathematically, the constraint becomes

$$\left(\sum_{k=1}^{p} a_k - 1\right) \geq \frac{1}{|1-2\mu|}\left(\sum_{k=1}^{p} |a_k| - 1\right) + \epsilon \tag{3.45}$$

The parameter ϵ is usually set to a small value, e.g. $\epsilon \approx 0.05$. It ensures ergodicity of the chain. It can be interpreted as a mixing parameter, i.e. the higher ϵ is the more 'mixed' the Markov chain is.

Once the optimal a_k are found, the sampling process can be easily simulated by following the steps of the algorithm. Frequency-shaped SRS using the resulting $r[n]$ has lower MSE than white SRS, and in many cases, even lower than aliased uniform sampling.

3.4 Error Analysis

In this section we do an error analysis on the MSE of both forms of SRS, both white and frequency-shaped. Section 3.4.1 presents the MSE scaling of white-SRS. Section 3.4.2 discusses the optimal choice of Ω_{hr}, the high-rate sampling rate. Section 3.4.3 develops the scaling of frequency-shaped SRS using the relaxation bound.

3.4.1 White SRS

In this section we develop a closed-form expression for the white-SRS MSE. The calculation is done on the continuous-time SRS model to incorporate the high sampling rate Ω_{hr} as a parameter.

Equation (3.16) gives an expression for the discrete-time error spectrum. Since there is no aliasing on the high-rate grid, i.e $\Omega_{\text{hr}} \geq 2\Omega_x$, the continuous-time error spectrum, after reconstruction, can be expressed as [25]:

$$S_{ee}(j\Omega) = T_{\text{hr}} S_{ee}(e^{j\omega})|_{\omega=\Omega T_{\text{hr}}} \tag{3.46}$$

Substituting Eqn.(3.16) into Eqn.(3.46), the CT error spectrum can be expressed as:

$$\begin{aligned} S_{ee}(j\Omega) &= T_{\text{hr}}\left(\frac{1}{\mu} - 1\right) R_{xx}[0]|H(e^{j\Omega T})|^2 \\ &= 2\pi\left(\frac{1}{\Omega_{\text{avg}}} - \frac{1}{\Omega_{\text{hr}}}\right) R_{xx}[0]|H(e^{j\Omega T})|^2 \end{aligned} \tag{3.47}$$

where we have made the substitution $T_{\text{hr}} = 2\pi/\Omega_{\text{hr}}$ and $\Omega_{\text{avg}} = \mu\Omega_{\text{hr}}$. Since $x[n] = x(nT_{\text{hr}})$, the DT auto-correlation is the sampled CT auto-correlation [25], i.e. $R_{xx}[m] = R_{xx}(\tau = mT_{\text{hr}})$. Consequently, in the expression above, $R_{xx}[m=0] = R_{xx}(\tau = 0)$. In addition, since $H(j\Omega)$ is a band-limiting filter with cutoff $\Omega_h \leq \Omega_x$, the CT filter is related to the DT filter though the mapping $H(j\Omega) = H(e^{j\omega})|_{\omega=\Omega T_{\text{hr}}}$, [25]. Making these substitutions, the error spectrum can be expressed as:

$$S_{ee}(j\Omega) = 2\pi R_{xx}(0)\left(\frac{1}{\Omega_{\text{avg}}} - \frac{1}{\Omega_{\text{hr}}}\right)|H(j\Omega)|^2 \tag{3.48}$$

The continuous-time MSE can be found by integrating Eqn.(3.48):

$$\begin{aligned} E_w = \frac{1}{2\pi}\int_{-\pi/T}^{\pi/T} S_{ee}(j\Omega)d\Omega &= R_{xx}(0)\left(\frac{1}{\Omega_{\text{avg}}} - \frac{1}{\Omega_{\text{hr}}}\right)\underbrace{\int_{-\Omega_h}^{\Omega_h}|H(j\Omega)|^2 d\Omega}_{2\pi E_h} \\ &= 2\pi R_{xx}(0)E_h\left(\frac{1}{\Omega_{\text{avg}}} - \frac{1}{\Omega_{\text{hr}}}\right) \end{aligned} \tag{3.49}$$

The CT white-SRS MSE is a function of Ω_{hr} and Ω_{avg} where $R_{xx}(0)$ and E_h are fixed parameters independent of the sampling frequencies. For a fixed Ω_{hr}, the DT white-SRS MSE is presented in Eqn.(??). The white-SRS MSE is used as a performance baseline throughout this thesis. We evaluate the performance the other, more complex sampling techniques in terms of a "shaping gain" over white SRS.

3.4.2 Optimal High-Rate Sampling Frequency

As mentioned in Chapter 2, there are many $(\mu, \Omega_{\text{hr}})$ pairs that can achieve a fixed Ω_{avg}. Mathematically, the two are related by the expression:

$$\Omega_{\text{avg}} = \mu\Omega_{\text{hr}} \tag{3.50}$$

The set of possible pairs can be plotted as a curve as in $(\mu, \Omega_{\text{hr}})$ space as in Figure 3-3. Here the Nyquist rate is $2\Omega_x = 400/2\pi$. The high-rate sampling rate is constrained such that $\Omega_{\text{hr}} \geq 400/2\pi$. The two curves in Figure 3-3 illustrate the $(\mu, \Omega_{\text{hr}})$ pairs for two different average sampling rates: $\Omega_{\text{avg}} = 400/2\pi$ and $\Omega_{\text{avg}} = 160/2\pi$.

Not all operating points along this curve are equal. From Eqn.(3.49) we observe that for a fixed Ω_{avg}, the white SRS MSE increases with Ω_{hr}. Figure 3-4 plots both the theoretical and empirical white SRS MSE for the example process discussed in Section 3.5. The high-rate sampling frequency is varied from $\Omega_{\text{hr}} = 400/2\pi$, its minimum value, to $\Omega_{\text{hr}} = 1200/2\pi$. Results are shown for two different average sampling rates: $\Omega_{\text{avg}} = 400/2\pi$ at the Nyquist rate and $\Omega_{\text{avg}} = 160/2\pi$ which is well below the Nyquist rate. The theoretical results match the empirical results closely.

Though the theoretical analysis is only done for white SRS, the same scaling holds for frequency-shaped SRS. Figure 3-4 plots the frequency-shaped SRS MSE as function of Ω_{hr} for $\Omega_{\text{avg}} = 400/2\pi$ and $160/2\pi$. This curve is found empirically from numerical simulation. Though the frequency-shaped MSE is below the white SRS MSE, it also grows as $(1-1/\Omega_{\text{hr}})$.

Clearly from both the theoretical and empirical analysis, the optimal choice of Ω_{hr} is its minimum possible value: the Nyquist rate for $x(t)$.

$$\Omega_{\text{hr}}^* = 2\Omega_x \tag{3.51}$$

This choice minimizes the white SRS and frequency-shaped SRS MSE. At first glance this result may seem counterintuitive because, though the high-rate grid is changing, the average sampling rate is constant. The high-rate grid matters though, because LTI filtering is not a perfect reconstruction system for non-uniform samples. In particular, as Ω_{hr} increases there is greater mismatch between LTI reconstruction and the non-uniform grid. This is seen most clearly in Figure 3-4 for the white SRS curve at $\Omega_{\text{avg}} = \Omega_x = 400/2\pi$. In this case, since we are at the Nyquist rate, we should be able to always achieve perfect reconstruction. This is true when $\Omega_{\text{hr}} = 2\Omega_x = 400/2\pi$ and the effective sampling grid is uniform. For larger values of Ω_{hr} though, the effective grid is non-uniform making the LTI reconstruction have an error that grows according to Eqn.(3.49). The same is true for frequency-shaped SRS though there is a dip at $\Omega_{\text{hr}} = 2\Omega_{\text{avg}} = 800/2\pi$. At this point, the Boufounos process optimization finds the uniform Nyquist grid again so the LTI filter achieves perfect reconstruction.

With non-LTI reconstruction, the value of Ω_{hr} may be less important. This is because the non-LTI reconstruction can achieve a perfect reconstruction from non-uniform samples. Such extensions are further discussed in Section 3.6. For the remainder of this chapter, we assume that the high-rate sampling frequency is fixed to its optimal value: $\Omega_{\text{hr}} = 2\Omega_x$.

3.4.3 Frequency-Shaping SRS

Throughout this section, we assume that the high-rate sampling frequency has been optimally chosen as $\Omega_{\text{hr}} = 2\Omega_x$. Since the high-rate is fixed, we can do the error analysis in discrete-time without loss of generality. It is difficult to find a closed-form expression for the MSE of frequency-shaped SRS because the Boufounos process is redesigned for each Ω_{hr} and Ω_{avg}. Consequently, we compute frequency-shaped SRS MSE numerically. We can however find upper and lower bounds on the MSE. The white SRS MSE gives an lower-bound on the performance of frequency-shaped SRS. The relaxation bound of Eqn.(3.26) gives an upper-bound.

We study the performance of frequency-shaped SRS relative to white SRS using the dimensionless shaping gain defined as:

$$\mathcal{G}_{\text{fs}} = \frac{\mathcal{E}_{\text{fs}}}{\mathcal{E}_{\text{w}}} \tag{3.52}$$

Like the MSE, the shaping gain can be bounded using the relaxation bound. This relaxation bound can expressed as the quotient of Eqn.(3.26) and Eqn.(3.17),

$$\mathcal{G}_{\text{r}} = \frac{\mathcal{E}_{\text{w}}}{\mathcal{E}_{\text{r}}} = \frac{\left(\frac{1}{2\pi}\int_{-\pi}^{\pi} S_{xx}(e^{j\omega})d\omega\right)\left(\int_{-\pi}^{\pi} |H(e^{j\theta})|^2 d\theta\right)}{\left(\min_\omega \int_{-\pi}^{\pi} |H(e^{j\theta})|^2 S_{xx}(e^{j(\omega-\theta)})d\theta\right)} \geq \mathcal{G}_{\text{fs}} \tag{3.53}$$

The shaping gain is always greater than or equal to one. The worst-case occurs when the input is white, i.e. $S_{xx}(e^{j\omega}) = R_{xx}[0]$. In this case, substituting $S_{xx}(e^{j\omega}) = R_{xx}[0]$ into Eqn.(3.53) gives $\mathcal{G}_{\text{r}} = 1$. The worst-case is also achieved when $H(e^{j\omega})$ is an all-pass filter. In both of these cases, the relaxation bound is tight, i.e. $\mathcal{G}_{\text{r}} = \mathcal{G}_{\text{fs}}$, because the set of binary auto-covariance matrices is equal to the set of all auto-covariance matrices. In general though, the relaxation bound is loose because the set of binary auto-covariance matrices is a subset of all auto-covariance matrices.

Other than these worst-case scenarios, some shaping gain is always possible. Even if

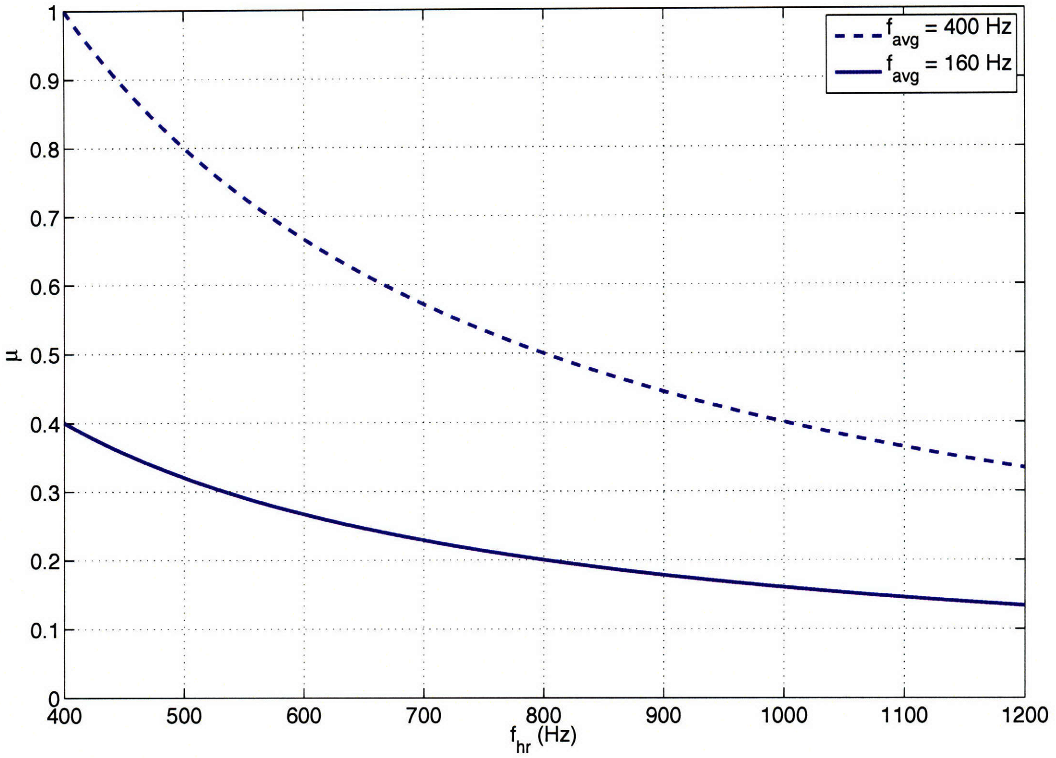


Figure 3-3. Possible rate implementation pairs $(\mu, \Omega_{\mathrm{hr}})$. Pairs shown for $\Omega_{\mathrm{avg}} = 400/1\pi$ and $160/2\pi$. The Ω_{hr} axis on this figure matches with Figure 3-4 below.

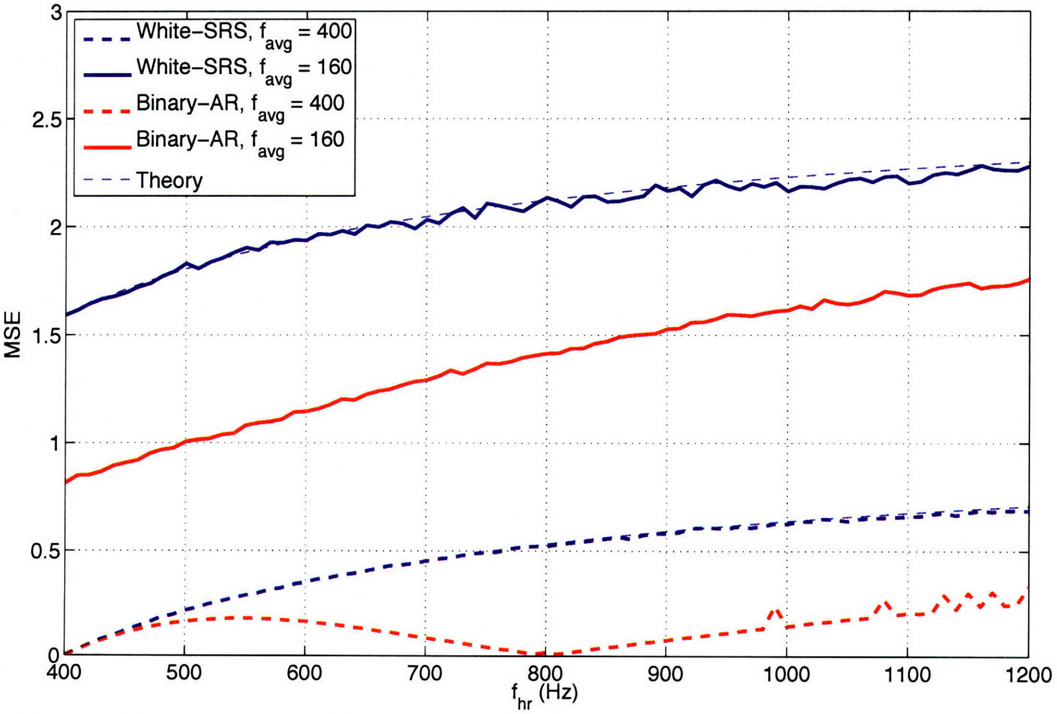


Figure 3-4. MSE scaling as function of Ω_{hr} for example discussed in Section 3.5. Error curves shown for white-SRS and frequency-shaped SRS at two fixed average sampling rates, $\Omega_{\mathrm{avg}} = 400/2\pi$ and $160/2\pi$.

$H(e^{j\omega})$ has no stop-band, i.e. $\omega_h = \pi$, shaping gain is possible as long as the response is not constant across the band. In general, given a fixed $H(e^{j\omega})$, the shaping gain is better for more peaky $S_{xx}(e^{j\omega})$. Specifically, the more concentrated the energy in $S_{xx}(e^{j\omega})$ is in the frequency domain, the lower the side-lobes are and the lower $F(e^{j\omega_0})$ can be. The shaping gain is consequently higher. This property is illustrated using AR(1) and MA(1) numerical examples in Section 3.5.

When the relaxation bound is high, the bound becomes so loose that it is uninformative. For example, the relaxation bound can, in theory, be infinite. This situation occurs if there is a shift $S_{xx}(e^{j(\omega-\theta)})$ that has zero overlap with $H(e^{j\omega})$. In this case $F(e^{j\omega_0}) = 0$ and the relaxation bound is infinite. In practice though, since the relaxed solution solution cannot be implemented, an infinite shaping gain cannot be achieved.

It may seem that oversampling, where $\Omega_{\text{hr}} > 2\Omega_x$, could improve the error. This is because with more oversampling, $H(e^{j\omega})$ has a larger effective stop-band which in turn reduces the value of $F(e^{j\omega_0})$. In fact, with enough oversampling the relaxation bound implies infinite shaping gain. Increasing the oversampling at a fixed Ω_{avg} decreases the value of μ though. This limits our ability to do binary shaping. For binary processes, the reduction in μ increases the error by a factor larger than the reduction in $F(e^{j\omega_0})$. Consequently, in practice, oversampling leads to a larger MSE. As discussed in Section 3.4.2, for frequency-shaped SRS the optimal high-rate sampling frequency is still $\Omega_{\text{hr}} = 2\Omega_x$.

The expression for the relaxation bound should be used with caution. It can lead to erroneous conclusions when the $\mathcal{G}_r$ is large. In general though, the relaxation bound builds important intuition about the limits of frequency-shaped SRS.

3.5 Numerical Experiments

In this section, we present an example of both white and frequency-shaped SRS along with a numerical error analysis. In addition, we present numerical simulations that illustrate the effect of $S_{xx}(e^{j\omega})$ on shaping gain.

3.5.1 Example

For our example, we assume a band-limited input $x(t)$ with maximum frequency $\Omega_x = 200/2\pi$. The high-rate sampling frequency is fixed to its minimum value $\Omega_{\text{hr}} = 2\Omega_x = 400/2\pi$, the Nyquist rate for $x(t)$. As discussed in Section 3.4.2, this is the optimal value of Ω_{hr}. The resulting DT signal $x[n]$ on the high-rate grid is a WSS DT ARMA process generated by shaping white Gaussian noise through a filter:

$$G(z) = \frac{(z - z_0)(z - z_1)}{(z - p_0)(z - p_1)(z - p_1^*)(z - p_2)(z - p_2^*)} \tag{3.54}$$

$$z_0 = 0.05,\ z_1 = e^{j\pi/4},\ p_0 = -0.5,\ p_1 = 0.9e^{j\pi/8},\ p_2 = 0.9e^{j3\pi/8}$$

The power spectrum is $S_{xx}(e^{j\omega}) = G(z)G(z^{-1})$. The continuous-time filter $h(t)$ is assumed to be an ideal LPF with cutoff $\Omega_h = 100$ Hz, half the bandwidth of $S_{xx}(e^{j\omega})$. In discrete-time this is implemented as a 2048-point FIR filter designed by applying a Hamming window to an ideal LPF with cutoff at $\omega_h = \Omega_h T_{\text{hr}} = \pi/2$. The average sampling rate is fixed to $\Omega_{\text{avg}} = 160/2\pi$, which is well below the Nyquist rate. The mean of $r[n]$ is consequently fixed to $\mu = \Omega_{\text{avg}}/\Omega_{\text{hr}} = 2/5$.

Figure 3-5 illustrates the discrete-time power spectrum $S_{xx}(e^{j\omega})$ and the filter $H(e^{j\omega})$. This example has been constructed to illustrate a situation where frequency-shaped SRS may be useful. We assume that anti-aliasing is not possible. The sampling rate is below the Nyquist rate, so uniform sampling leads to aliasing. On the other hand, the reconstruction filter has a large stop-band that we can take advantage of.

Two million samples of $w[n]$ are generated for each test case. Periodogram averaging with a 2048-point Hamming window with 50% overlap is used to approximate 2048 points of the power spectrum $S_{ww}(\omega)$. The MSE is estimated numerically by averaging the squared difference between $y[n]$ and $\hat{y}[n]$ after filtering. The in-band SNR, defined below, is used as a power-independent performance metric.

$$\text{SNR} = \frac{E\{y^2[n]\}}{\mathcal{E}} \tag{3.55}$$

Fig.3-5(a) shows the result of uniform sampling at the rate $\Omega_{\text{avg}} = 160$ Hz. Note the strong aliases in the band of interest. The in-band SNR is 3.600 dB. Fig.3-5(b) shows the result of white-SRS sampling with $\Omega_{\text{hr}} = 400$ Hz and $\mu = 2/5$. As predicted, the noise has a flat power spectrum with height given by Eqn.(3.16). The in-band SNR is 0.763 dB. Fig 3-5(c) shows the result of frequency-shaped SRS with a two-pole Boufounos process. The sampling process is generated according to the model of [5], with parameters $a_1 = -0.3394$ and $a_2 = 0.4783$. These values are found through numerical optimization for this specific $H(e^{j\omega})$ and $S_{xx}(e^{j\omega})$. Note how the noise has been shaped out of band, so that the in-band SNR is 3.826 dB, greater than both white SRS and aliased uniform sampling.

Figure 3-6 illustrate the results of the numerical optimization of the Boufounos process for this example leading to the parameters used above. A mixing parameter $\epsilon = 0.05$ was used to ensure the strict inequality of Eqn.(3.42). Figure 3-6(a) illustrates $F(e^{j\omega})$ along with the optimal auto-covariance spectrum $\Phi_{rr}(e^{j\omega})$. As expected, most of the energy of $\Phi_{rr}(e^{j\omega})$ is near the minimum point of $F(e^{j\omega})$ at $\omega_0 = \pi$. Since the process is restricted to be binary though, $\Phi_{rr}(e^{j\omega})$ cannot be made impulsive at this point. Figure 3-6(b) illustrates a realization of the random process $r[n]$ in steady-state. Note how the sampling process is close to $\frac{1}{2}(1 + (-1)^n)$, the optimal relaxed solution, aside from the random intervals where it is all zero. These zero excursions ensure that the mean remains stationary at $\mu = 2/5$.

Figure 3-7(a) illustrates the SNR as a function of μ for this example. Since Ω_{hr} is fixed, μ serves as a DT proxy for the average sampling rate Ω_{avg}. The SNR is computed empirically using numerical simulation for $\Omega_{\text{avg}} = 0$ to $\Omega_{\text{avg}} = 2\Omega_x = 400/2\pi$. Figure 3-7(b) illustrates the results from the same simulation in terms of shaping gain, $\mathcal{G}_{\text{fs}}$.

There are a number of points to note. First, both the white SRS SNR and the relaxation bound grow as $O(1/\mu - 1)$. This is expected: with more samples we do better. Next, the empirical white SRS SNR curve matches closely with the theoretical one. Thirdly, as expected, the frequency-shaped SRS SNR is always above that of white SRS, i.e. shaping gain is always greater than 1. As $\mu \to 1$ or $\mu \to 0$, the frequency-shaped SNR approaches the performance of white SRS. This is expected because, as noted in Section 3.3.3, the set of achievable binary processes becomes smaller and closer to a white process.

The shaping gain peaks at $\mu = 1/2$. Frequency-shaped SRS is most useful in the regime where $\Omega_{\text{avg}} \approx \Omega_x$ because the set of achievable binary auto-covariance functions is the largest. Note that for this particular input process, frequency-shaped SRS nearly hits the relaxation bound when $\mu = 1/2$. This is a special case, because for this example process $\omega_0 = \pi$ and when $\mu = 1/2$, the Boufounos process can achieve the optimal relaxed solution,

$$r_{\text{r}}[n] = \frac{1}{2}(1 + (-1)^n) \tag{3.56}$$

Empirically, there is still a small gap from the bound at $\mu = 1/2$. This is from the mixing parameter $\epsilon = 0.05$. The non-zero value of ϵ prevents the binary-AR solution from precisely achieving the non-ergodic $r_{\text{r}}[n]$. In any case, the solution at $\mu = 1/2$ is aliasing. It is uniform sampling at half the Nyquist rate. For practical purposes, if the goal is to anti-alias, this is not a good operating point. It may be better to increase the mixing parameter ϵ in the Boufounos process optimization to get a randomized grid closer to that of white SRS.

3.5.2 Peakiness

In this section, we illustrate the effect of input peakiness on SRS performance using two experiments. For both, the reconstruction filter $H(e^{j\omega})$ is a ideal LPF with cutoff $\omega_h = \pi/2$. The input $S_{xx}(e^{j\omega})$ is assumed to be band-limited to $\Omega_x = 200/2\pi$ and the high-rate sampling frequency is $\Omega_{\text{hr}} = 400/2\pi$. The average sampling rate is fixed to $\Omega_{\text{avg}} = 160/2\pi$ so $\mu = 2/5$. In the first experiment $x[n]$ is a first-order moving average MA(1) process with power spectrum,

$$S_{xx}(z = e^{j\omega}) = (1 - \theta e^{j\omega})(1 - \theta e^{j\omega}) \tag{3.57}$$

Figure 3-8 plots the shaping gain and the relaxation bound as θ is varied from zero to one. Each point on the curve is found by numerically computing the shaping gain from 600,000 samples. There are a number of observations to make. When $\theta \approx 0$ the spectrum is nearly white. Consequently, as predicted, the shaping gain near 1. As θ increases the spectrum becomes more peaky as the zero moves closer to the unit circle. Accordingly, the shaping gain is higher. The shaping gain flattens out near $\theta = 1$ because an MA(1) process is not infinitely peaky, even when the zero is on the unit circle. The non-binary bound is relatively tight in this case because the process does not become infinitely peaky.

In the second experiment $x[n]$ is a first-order auto-regressive AR(1) process with power spectrum,

$$S_{xx}(e^{j\omega}) = \frac{1}{(1 - \rho e^{-j\omega})(1 - \rho e^{j\omega})} \tag{3.58}$$

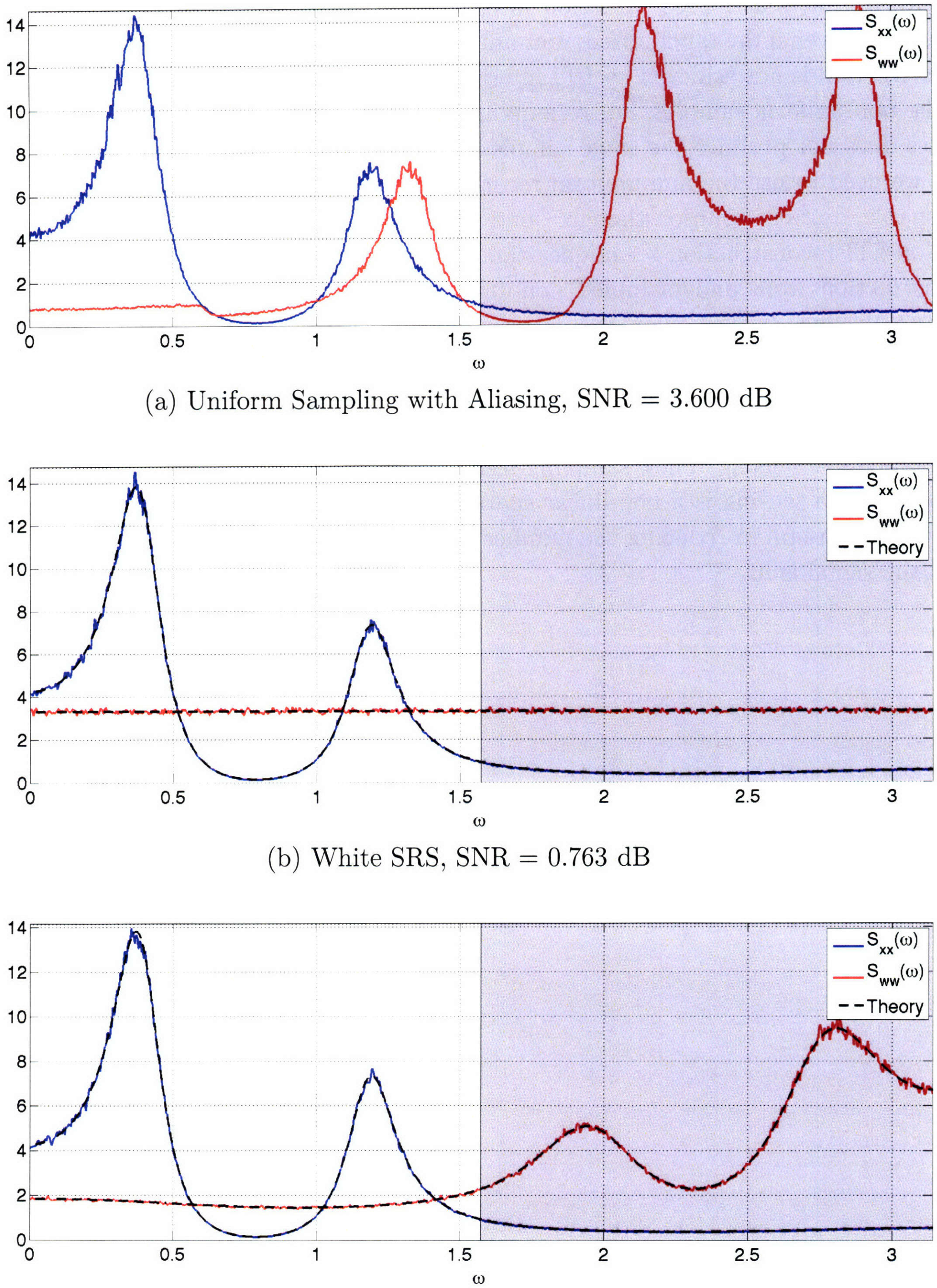
Figure 3-8 plots the shaping gain and the relaxation bound as ρ is varied from zero to one. Each point on the curve is found by numerical simulation of 600,000 samples. Similar to the MA(1) experiment, when $\rho \approx 0$, the spectrum is nearly white and the shaping gain is near 1. As ρ increases the pole moves closer to the unit circle, making $S_{xx}(e^{j\omega})$ more peaky. Unlike the MA(1) experiment, the AR(1) process becomes infinitely peaky as the pole approaches the unit circle. Consequently, the shaping gain increases exponentially as $\rho \rightarrow 1$. The relaxation bound becomes looser in this limit for the same reason. In short, the peakiness of the input controls the extent of the possible shaping gain using frequency-shaped SRS. The peakier the input, the more potential gain is possible.

3.6 Extensions

In SRS, $h(t)$ acts as a reconstruction filter. This is not a perfect reconstruction scheme. Specifically, even if the randomized, non-uniform samples are above the Nyquist rate, perfect reconstruction is not achieved by LTI filtering. In fact, LTI reconstruction is not consistent for the non-uniform samples, i.e. resampling the reconstruction, $\hat{y}(t)$, onto the same time-indices does not produce the same samples. For perfect reconstruction above the Nyquist rate, we need to use the non-uniform reconstruction theorem [22]. LTI reconstruction is a reasonable, first-cut solution though – one that is commonly used in practice.

Non-LTI reconstruction techniques can potentially do better. The question of non-LTI reconstruction from undersampled non-uniform samples brings up the question of non-uniform aliasing. In particular, the nature of aliasing with non-uniform samples is not well studied. Potential extensions of this thesis can consider using a deeper understanding of non-uniform aliasing to design a better non-LTI reconstruction technique for SRS.

Another potential extension could involve using frequency-shaped SRS as a front-end for compressive sensing, a new sampling technique for sparse signals [8]. The reconstruction in compressive sensing is a non-linear sparse approximation technique. Frequency-shaped SRS may be useful in reducing the number of samples given prior information about what band the signal is in.

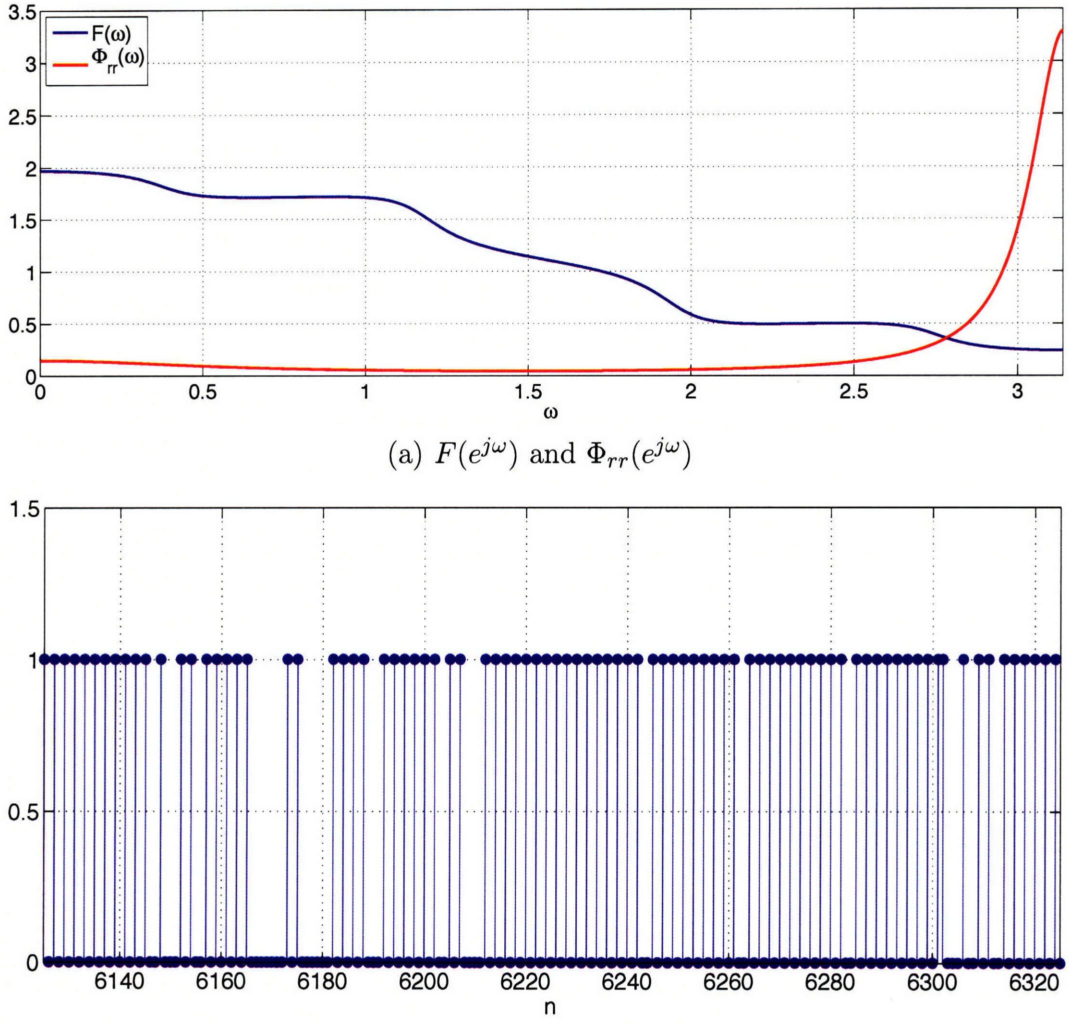


(a) Uniform Sampling with Aliasing, SNR = 3.600 dB

(b) White SRS, SNR = 0.763 dB

(c) Frequency-Shaped SRS, $\epsilon = 0.05$, SNR = 3.826 dB

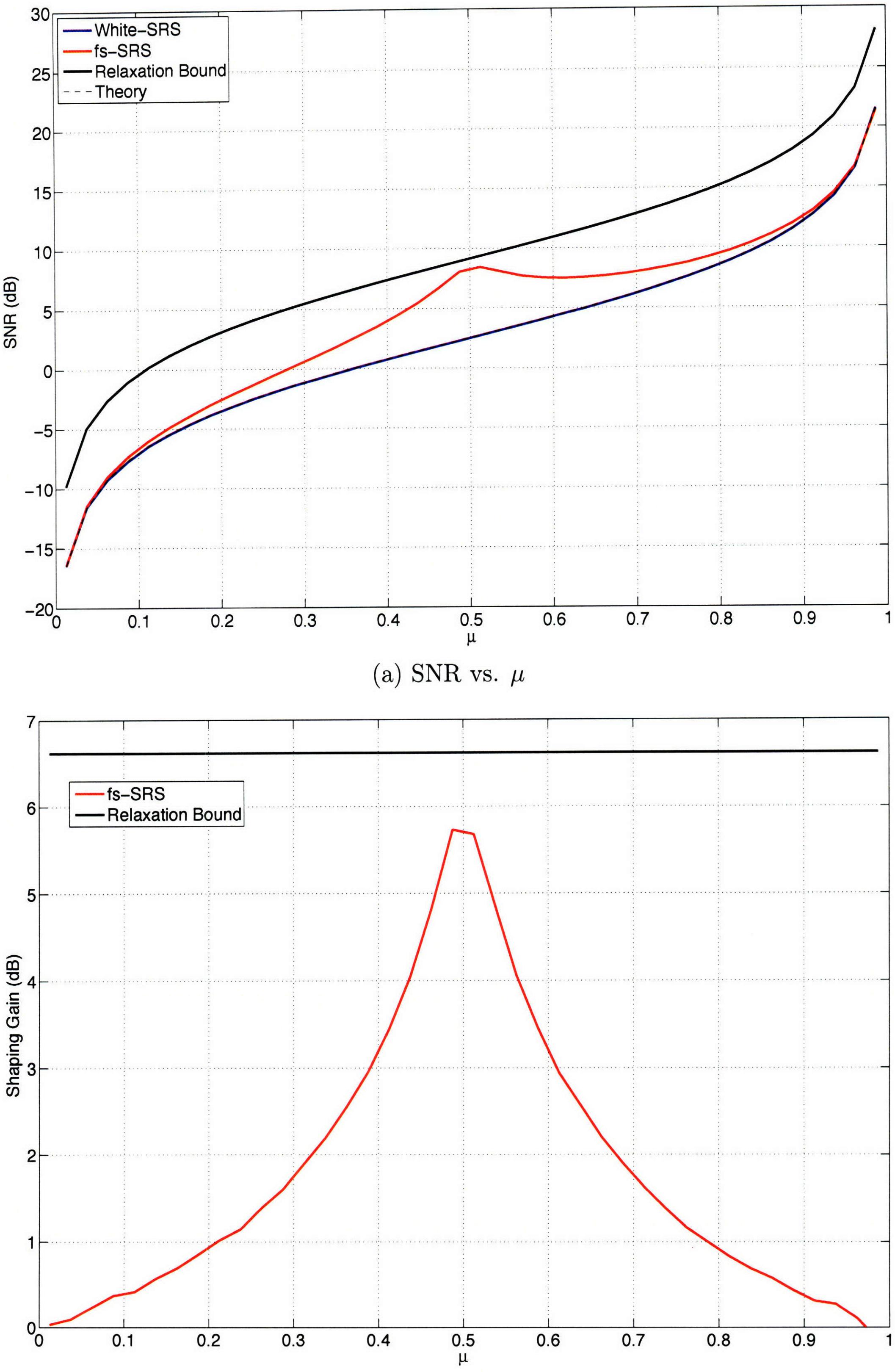
Figure 3-5. Empirical plots of randomized sampling power spectra for the example discussed in Section 3.5. Average sampling rate fixed to $\Omega_{\text{avg}} = 160/2\pi$. Gray regions denote the stop-band of $H(\omega)$. White regions denote the pass-band of $H(\omega)$.



(a) $F(e^{j\omega})$ and $\Phi_{rr}(e^{j\omega})$

(b) Realization of $r[n]$ in steady-state, $\mu = 2/5$

Figure 3-6. Results of Boufounos process design for the example discussed in Section 3.5. Design done using numerical optimization with $\epsilon = 0.05$. The optimal parameters are $a_1 = -0.3394$ and $a_2 = 0.4783$.



(a) SNR vs. μ

(b) Shaping gain $\mathcal{G}_{\text{fs}}$ vs. μ

Figure 3-7. SNR and shaping-gain as a function of μ for example discussed in Section 3.5. $\Omega_{\text{hr}} = 400/2\pi$, the Nyquist rate for the input.

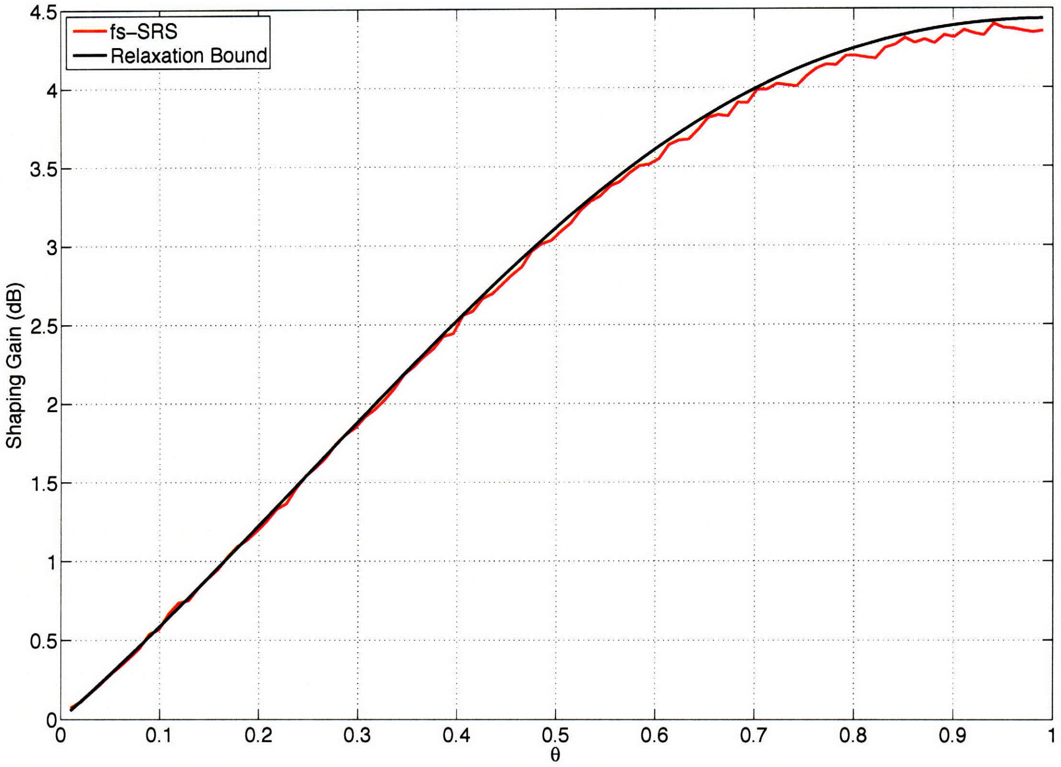


Figure 3-8. Shaping gain of MA(1) process defined in Eqn.(3.57) as a function of θ. $\mu = 2/5$.

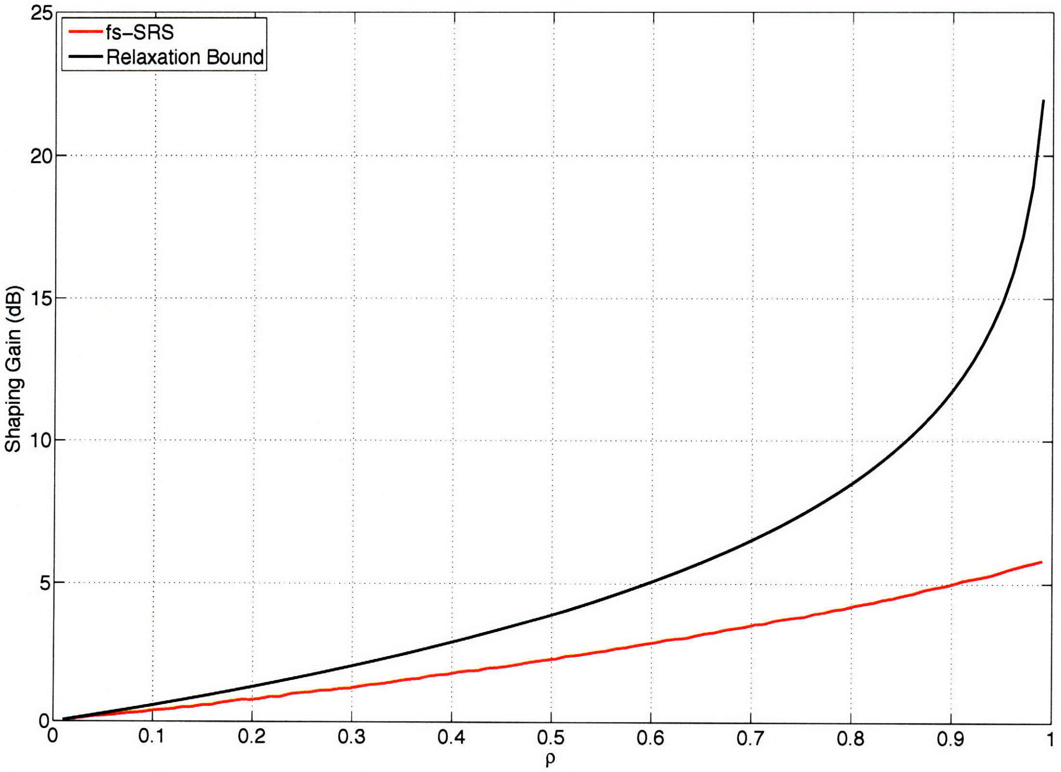


Figure 3-9. Shaping gain of AR(1) process defined in Eqn.(3.58) as a function of ρ. $\mu = 2/5$.

Chapter 4

Filtered Randomized Sampling

In this chapter filtered randomized sampling (FRS) is developed as an extension of SRS that incorporates a pre-filter and post-filter. Section 4.1 introduces the FRS model and basic issues in its design. The possibility of an invertibility constraint between the pre-filter and post-filter leads to two different forms of FRS: distortion-free FRS where the filters are inverses of one another and unconstrained FRS where the filters are unconstrained. In addition, similar to SRS, there are two forms of FRS depending on the correlation of the sampling process: white or frequency-shaped. The combination of these two forms leads us to consider four types of FRS in this chapter: distortion-free white FRS, distortion-free frequency-shaped FRS, unconstrained white FRS, and unconstrained frequency-shaped FRS. Each is discussed in Sections 4.2 through 4.5, respectively. Section 4.7 discusses potential extensions of FRS.

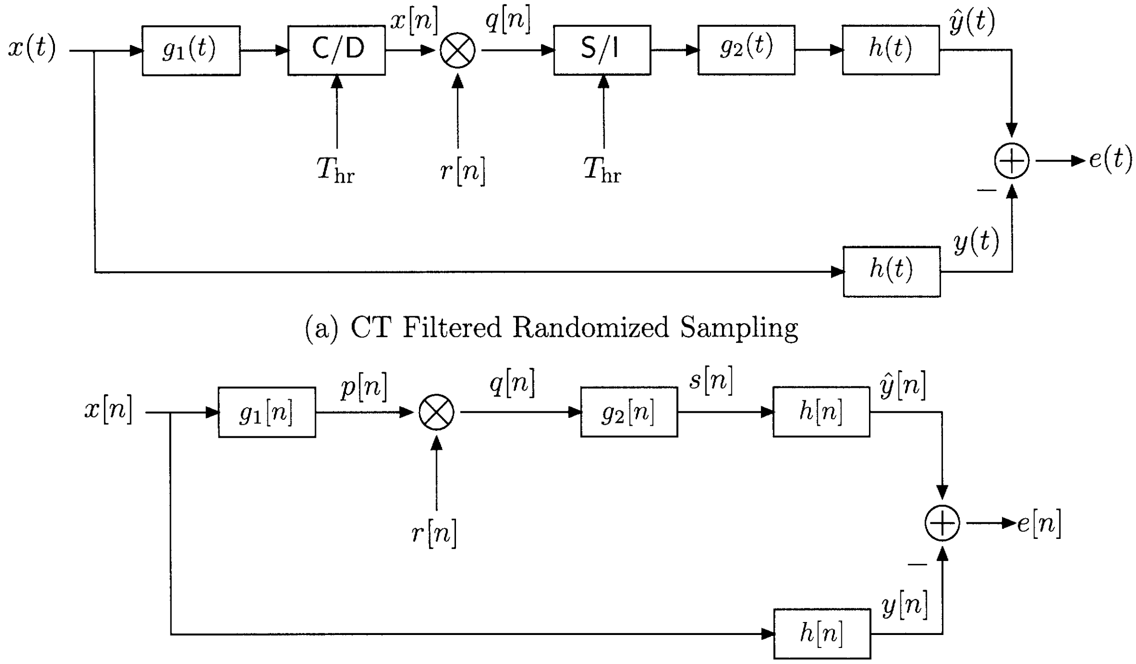
4.1 Introduction

The model for FRS is analogous to SRS, except for two additional design parameters, a pre-filter $g_1(t)$, and a post-filter $g_2(t)$. As in SRS, the covariance of the sampling process, $\Phi_{rr}(e^{j\omega})$, can potentially be designed. The upper-branch of Figure 4-1(a) illustrates the continuous-time FRS model. The lower branch illustrates the desired output $y(t)$.

The FRS model is fundamentally different from SRS because it assumes the ability to pre-filter before sampling. For certain applications, such as ray-tracing [10], pre-filtering is not possible because the signal is not accessible before sampling. In this sense, SRS, without a pre-filter, is more broadly applicable. Nevertheless, for classical analog-to-digital conversion, analog pre-filtering is possible and often easy to implement using modern analog electronics.

Similar to SRS, the average sampling rate in FRS, Ω_{avg}, is constrained to the under-sampled regime. For SRS, this implies that Ω_{avg} is constrained below the Nyquist rate for $x(t)$. This changes in FRS. Because of the ability to pre-filter, any reasonable choice of $G_1(j\Omega)$ should remove the out of band energy in $S_{xx}(j\Omega)$, i.e. for $\Omega > \Omega_h$. This energy is not part of the desired reconstruction and can reduce performance by aliasing into band. Consequently, in a properly optimized FRS system, $G_1(j\Omega)$ is a band-limiting filter with cutoff Ω_h. Additionally, since $G_2(j\Omega)$ is cascaded with $H(j\Omega)$, without loss of generality $G_2(j\Omega)$ can be chosen as a band-limiting filter with cutoff Ω_h. The effective Nyquist rate is thus $2\Omega_h$. The under-sampled regime for FRS occurs when Ω_{avg} is constrained to,

$$\Omega_{\text{avg}} < 2\Omega_h \tag{4.1}$$



(a) CT Filtered Randomized Sampling

(b) DT Filtered Randomized Sampling

Figure 4-1. FRS models.

Though $G_1(j\Omega)$ removes energy above Ω_h, it is not a classical anti-aliasing filter. If that were the case, its cutoff would be at $\Omega_{\text{avg}}/2 < \Omega_h$.

Since the effective Nyquist rate is $2\Omega_h$ and FRS does LTI reconstruction, the logic of Section 3.4.2 implies that the optimal high-rate sampling frequency is,

$$\Omega_{\text{hr}} = 2\Omega_h \tag{4.2}$$

This is the minimal Ω_{hr} for which there is no aliasing on the high-rate grid. It minimizes the MSE for a fixed Ω_{avg}. The argument is analogous to that presented in Section 3.4.2.

Combining the fact that $\Omega_{\text{hr}} = 2\Omega_h$ and that $g_1(t)$ and $g_2(t)$ are band-limited to Ω_h, the CT FRS model can be recast purely in DT. The DT FRS model is illustrated in the upper-branch of Figure 4-1(b). The desired output $y[n]$ is illustrated as the output of the lower-branch. Similar to SRS, the goal is to design the pre-filter $g_1[n]$, post-filter $g_2[n]$, and the sampling process auto-covariance $\Phi_{rr}(e^{j\omega})$ such that the MSE is minimized.

Since $\Omega_{\text{hr}} = 2\Omega_h$, this implies that $H(e^{j\omega})$ has no stop-band. This is different from SRS. In SRS the goal of frequency-shaping was to place most of the error in the stop-band of $H(e^{j\omega})$. By contrast, in FRS the goal is to shape the error in band to locations where it is attenuated by the post-filter. Figure 4-2(a) and (b) illustrates the relative placement of the important CT and DT frequency parameters, respectively. Note how the energy of $S_{xx}(j\Omega)$ out of band is removed before conversion to DT.

For the same input and average sampling rate, FRS has the potential to perform better than SRS because of the additional degrees of freedom. With more freedom also comes more complexity. In particular, there is an issue of distortion that occurs in FRS. The output of the DT FRS system is:

$$\hat{y} = \mu(h * g_2 * g_1 * x) + h * g_2 * ((g_1 * x)\tilde{r}) \tag{4.3}$$

and the error can be expressed as:

$$e = \underbrace{(\mu h * g_1 * g_2 - h) * x}_{u[n]} + \underbrace{h * g_2 * ((g_1 * x)\tilde{r})}_{v[n]} \tag{4.4}$$

where the dependence on n has been dropped for notational clarity. If we restrict the pre-filter and post-filter such that they are inverses of one another in the pass-band of $h[n]$:

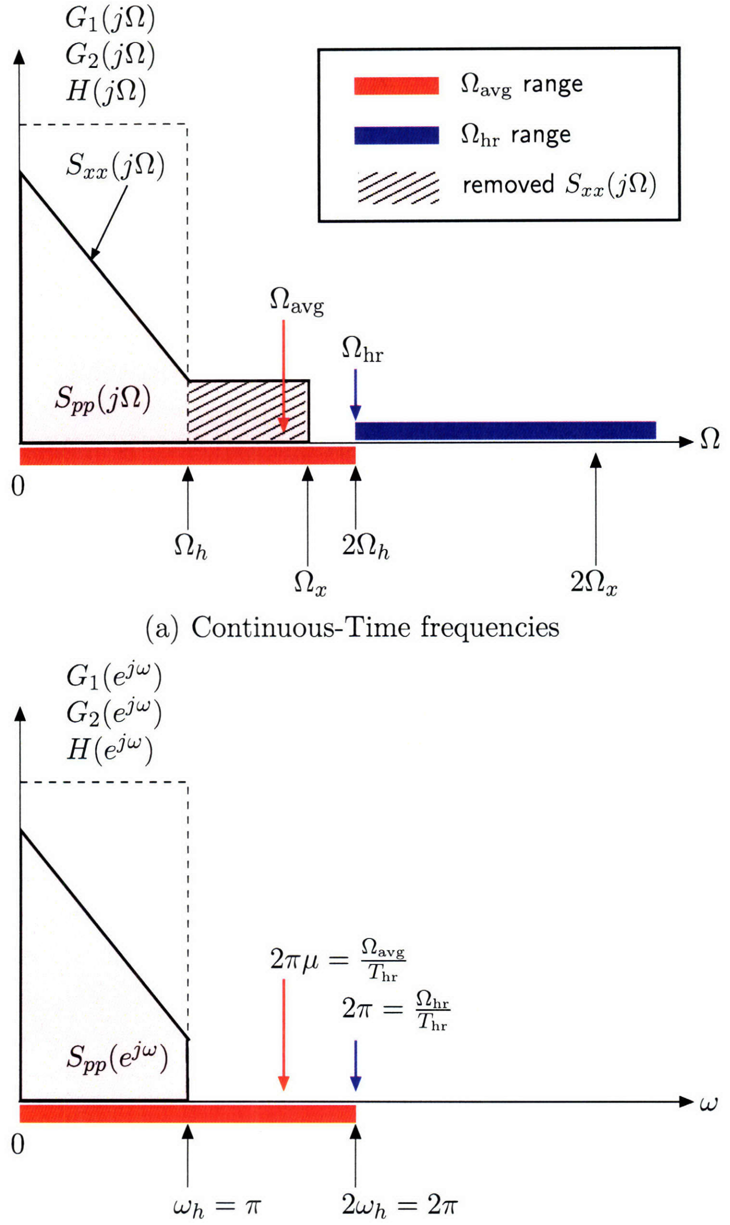
$$\mu g_1[n] * g_2[n] * h[n] = h[n] \tag{4.5}$$

the first term in Eqn.(4.4) becomes zero, i.e. $u[n] = 0$. Consequently, $e[n] = v[n]$ and since $E\{\tilde{r}[n]\} = 0$, it is straightforward to show that $E\{v[n]\} = 0$ and $K_{xv}[m] = 0$. This, in turn, implies that $K_{xe}[m] = 0$. With this constraint, the error is unbiased and uncorrelated with the input like in SRS. We denote this as distortion-free FRS because the the first term in Eqn.(4.3) is not impacted by the filters. Noise is only due to the second additive term that is unbiased and uncorrelated with the input. As with SRS, distortion-free FRS could be desirable in certain applications where this uncorrelated, additive error is preferable distortion of the system frequency response.

Alternatively, the filters can remain unconstrained. We denote this as unconstrained FRS. In this case, the error is biased and correlated, but can achieve a lower MSE. In certain contexts, the lower MSE is the only metric of importance. For example, in a surveillance context where the goal is to detect radar signals, the bias and correlation of the error with the input is largely irrelevant as long as the detection performance is improved.

In addition to the filter constraints, there are two forms of FRS depending on how the sampling process is correlated in time. The simplest form is white FRS, where $r[n]$ is restricted to be a Bernoulli process. In white FRS, only the filters shape the error. It has a strong duality with classical quantization theory. In fact, as developed in this thesis, there is an exact correspondence with D*PCM, differential pulse coded modulation without a feedback loop [17]. Sections 4.2 and 4.4 develop distortion-free white FRS and unconstrained white FRS, respectively.

More generally, $\Phi_{rr}(e^{j\omega})$, the auto-covariance of $r[n]$, can be tuned in addition to the filters. We denote this as frequency-shaped FRS. Frequency-shaped FRS is a difficult problem. In this thesis, we present the problem statement and an iterative design technique for distortion-free frequency-shaped FRS. We briefly discuss the extension of the algorithm for unconstrained frequency-shaped FRS, but do not develop it in detail. Though the resulting solutions improve MSE performance over white FRS, we have not fully characterized the limits of frequency-shaped FRS. Sections 4.3 and 4.5 develop distortion-free frequency-shaped FRS and unconstrained frequency-shaped FRS, respectively. The four variants of FRS are summarized in Table 4.1 along with their respective abbreviations.



(a) Continuous-Time frequencies

(b) Discrete-Time frequencies

Figure 4-2. Important frequency parameters in filtered randomized sampling. (a) illustrates the CT frequency parameters. The pass-bands for $H(j\Omega)$, $G_1(j\Omega)$, and $G_2(j\Omega)$ are Ω_h. The red area denotes the possible values for Ω_{avg}. Ω_{hr} should be set to $2\Omega_h$. The hatched region denotes the frequencies of $S_{xx}(j\Omega)$ that are removed by $G_1(j\Omega)$. (b) illustrates the DT frequency parameters when sampled at $\Omega_{\mathrm{hr}} = 2\Omega_h$.

	White $K_{rr}[m] = \sigma_r^2\delta[m]$	**Frequency-Shaped** $K_{rr}[m]$ unconstrained
Distortion-Free $\mu g_1[n] * g_2[n] * h[n] = h[n]$	Section 4.2 dfw-FRS	Section 4.3 dffs-FRS
Unconstrained $g_1[n]$, $g_2[n]$ unconstrained	Section 4.4 uw-FRS	Section 4.5 ufs-FRS

Table 4.1. Types of FRS

4.2 Distortion-Free White FRS

In this section we discuss distortion-free white FRS (dfw-FRS), where the sampling process is restricted to be white and the invertibility constraint of Eqn.(4.5) is imposed.

4.2.1 Design Problem

Because of the invertibility constraint, the error signal in dfw-FRS is given by Eqn.(4.4) with $u[n] = 0$:

$$e[n] = v[n] = h[n] * g_2[n] * \underbrace{(p[n]\tilde{r}[n])}_{z[n]} \tag{4.6}$$

where $p[n] = g_1[n] * x[n]$. Since $r[n]$ is Bernoulli process, $z[n]$ is a white process with power spectrum:

$$S_{zz}(e^{j\omega}) = \sigma_p^2\sigma_r^2 \tag{4.7}$$

where, as always, $\sigma_r^2 = \mu(1-\mu)$ and σ_p^2 denotes the variance of $p[n]$. It can be expressed as the area under $S_{pp}(e^{j\omega})$:

$$\sigma_p^2 = \int_{-\pi}^{\pi} S_{pp}(e^{j\omega})\frac{d\omega}{2\pi} = \int_{-\pi}^{\pi} |G_1(e^{j\omega})|^2 S_{xx}(e^{j\omega})\frac{d\omega}{2\pi} \tag{4.8}$$

Combining Eqn.(4.7) with Eqn.(4.6) the error power-spectrum can be expressed as:

$$S_{ee}(e^{j\omega}) = \sigma_r^2\sigma_p^2|G_2(e^{j\omega})|^2|H(e^{j\omega})|^2 \tag{4.9}$$

As described in the previous section, it is assumed that $H(e^{j\omega})$ has no stop-band. In what follows, $H(e^{j\omega})$ is assumed to be a invertible filter with $H(e^{j\omega}) \neq 0$, for $|\omega| < \pi$. Consequently Eqn.(4.5) can be equivalently expressed as:

$$G_1(e^{j\omega}) = \frac{1}{\mu G_2(e^{j\omega})} \tag{4.10}$$

From Eqn.(4.9), we observe that the error spectrum has the same shape as $|G_2(e^{j\omega})|^2|H(e^{j\omega})|^2$. Naively, it may seem that any desired spectra can be achieved by appropriately choosing

$G_2(e^{j\omega})$. This is not the case however. First, note that the desired error spectrum must be invertible, i.e. $S'_{ee}(e^{j\omega}) > 0$, for $|\omega| < \pi$. Otherwise, $G_1(e^{j\omega})$ cannot be designed to maintain the constraint Eqn.(4.5). Assuming invertibility, the shape of a desired error spectrum, $S'_{ee}(e^{j\omega})$, can be matched by choosing $|G_2(e^{j\omega})|^2$ in the pass-band as:

$$|G_2(e^{j\omega})|^2 = \alpha \frac{S'_{ee}(e^{j\omega})}{|H(e^{j\omega})|^2} \tag{4.11}$$

where α is an arbitrary positive scale factor. Unfortunately, since $G_1(e^{j\omega})$ is coupled to $G_2(e^{j\omega})$, the scaling, determined by σ_p^2, is dependent on the choice of $S'_{ee}(e^{j\omega})$. Combining Eqn.(4.5), (4.11), (4.8) and substituting into Eqn.(4.9), the factor α cancels and the achieved error spectrum is:

$$S_{ee}(e^{j\omega}) = \sigma_r^2 \left(\int_{-\omega_p}^{\omega_p} \frac{|H(e^{j\omega})|^2 S_{xx}(e^{j\omega})}{S'_{ee}(e^{j\omega})} d\omega \right) S'_{ee}(e^{j\omega}) \tag{4.12}$$

Equation (4.12) is invariant to scaling of $S'_{ee}(e^{j\omega})$. Consequently, given a desired shape for the error spectrum, we have no control over the scaling. This degeneracy occurs because a degree of freedom is lost in imposing the constraint of Eqn.(4.5). This is reasonable though, because otherwise the error could be made arbitrarily small by scaling $G_2(e^{j\omega})$. The fact that $|G_1(e^{j\omega})|^2 = 1/|G_2(e^{j\omega})|^2$, amplifies the total error, preventing such a trivial solution. These two competing effects fix the scaling for a given spectral shape.

The goal in dfw-FRS design is to find the error spectrum shape that minimizes the MSE. By integrating Eqn.(4.9), the MSE can be expressed as:

$$\mathcal{E}_{\text{dfw}} = \sigma_r^2 \sigma_p^2 \int_{-\pi}^{\pi} |G_2(e^{j\omega})|^2 |H(e^{j\omega})|^2 \frac{d\omega}{2\pi} \tag{4.13}$$

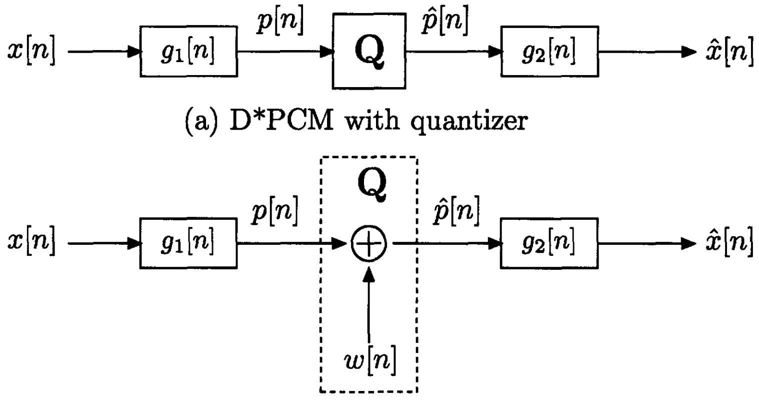
Substituting Eqn.(4.8), and incorporating the constraint of Eqn.(4.10), the design goal can be expressed as a constrained optimization over $|G_1(e^{j\omega})|^2$:

$$\min_{|G_1(e^{j\omega})|^2} \frac{\sigma_r^2}{\mu^2} \left(\int_{-\pi}^{\pi} |G_1(e^{j\omega})|^2 S_{xx}(e^{j\omega}) \frac{d\omega}{2\pi} \right) \left(\int_{-\pi}^{\pi} \frac{|H(e^{j\theta})|^2}{|G_1(e^{j\theta})|^2} \frac{d\theta}{2\pi} \right) \tag{4.14}$$

$$\text{subject to } |G_1(e^{j\omega})|^2 \geq 0 \ \forall \omega$$

4.2.2 Optimal Design

As mentioned in Section 4.1, white FRS has an exact correspondence with D*PCM, a waveform coding strategy for quantization. D*PCM is a special case of DPCM formed by eliminating the noise feedback. Figure 4-3(a) illustrates a block diagram for D*PCM [17]. The Q block represents a quantizer. Assuming high-rate quantization, the quantizer can be modeled as an additive white noise source independent of the input $p[n]$ [24, 17]. This is illustrated as Figure 4-3(b). The quantization noise variance is linearly dependent on the input variance, i.e. $\sigma_q^2 = \beta\sigma_p^2$. With this quantizer model, the error enters the D*PCM



(a) D*PCM with quantizer

(b) D*PCM with quantizer replaced by additive noise model

Figure 4-3. D*PCM block diagram.

system in the same way as white FRS. Mathematically, the MSE for D*PCM is:

$$\mathcal{E}_{\text{D*PCM}} = \beta\sigma_p^2 \int_{-\pi}^{\pi} |G_2(e^{j\omega})|^2 \frac{d\omega}{2\pi} \tag{4.15}$$

Comparing this to Eqn.(4.13), we observe that the error is the same except for the additional term $|H(e^{j\omega})|^2$ in the expression for white FRS. The constant of proportionality in white-FRS is $\beta = \sigma_r^2$. This correspondence between D*PCM and white-FRS is useful because D*PCM has been extensively studied in the quantization literature. Solutions to both distortion-free white FRS and unconstrained white FRS can be adapted from the D*PCM derivations. The optimal design for distortion-free white FRS is analogous to the 'half-whitening' solution for D*PCM. Though the derivation is relatively straightforward in [17], we include the derivation here for completeness, and to incorporate the extra term $H(e^{j\omega})$ in FRS.

Define ℓ_2 as the Hilbert space of finite-energy discrete-time signals. Using Parseval's relationship, the inner product in this space can be expressed in the frequency domain as $\langle V_1(e^{j\omega}), V_2(e^{j\omega})\rangle = \int_{-\pi}^{\pi} V_1^*(e^{j\omega})V_2(e^{j\omega})d\omega/2\pi$, [1]. Define two elements in ℓ_2 which have Fourier transforms:

$$V_1(e^{j\omega}) = \frac{\sigma_r}{\mu}|G_1(e^{j\omega})|\sqrt{S_{xx}(e^{j\omega})} \tag{4.16}$$

$$V_2(e^{j\omega}) = \frac{\sigma_r}{\mu}\frac{|H(e^{j\omega})|}{|G_1(e^{j\omega})|} \tag{4.17}$$

Substituting Eqns. (4.16) and (4.17) into Eqn. (4.14), the dfw-FRS MSE can be expressed as the product of the norms of $V_1(e^{j\omega})$ and $V_2(e^{j\omega})$. The dfw-FRS MSE can be bounded below using the Cauchy-Schwartz inequality:

$$\left(\int_{-\pi}^{\pi} |V_1(e^{j\omega})|^2 \frac{d\omega}{2\pi}\right)\left(\int_{-\pi}^{\pi} |V_2(e^{j\omega})|^2 \frac{d\omega}{2\pi}\right) \geq \left|\int_{-\pi}^{\pi} V_1^*(e^{j\omega})V_2(e^{j\omega})\frac{d\omega}{2\pi}\right|^2 \tag{4.18}$$

This lower bound is met with equality if and only if $V_1(e^{j\omega})$ is linearly-dependent on $V_2(e^{j\omega})$, i.e. $V_1(e^{j\omega}) = \alpha V_2(e^{j\omega})$, [1]. Applying this relation to Eqns. (4.16) and (4.17), and using the invertibility constraint Eqn.(4.10), the magnitude squared of the optimal filters can be expressed as:

$$|G_1(e^{j\omega})|^2 = \alpha \frac{|H(e^{j\omega})|}{\sqrt{S_{xx}(e^{j\omega})}} \tag{4.19}$$

$$|G_2(e^{j\omega})|^2 = \frac{1}{\alpha\mu^2} \frac{\sqrt{S_{xx}(e^{j\omega})}}{|H(e^{j\omega})|} \tag{4.20}$$

up to an indeterminate scale factor α. Without loss of generality we can set $\alpha = 1$. The phase response for these filters should be chosen so that their cascade has zero phase – to satisfy the invertibility constraint of Eqn.(4.5). In practice though, these filters can be chosen so that the cascade is approximately an all-pass filter with linear-phase.

Aside from the extra term $H(e^{j\omega})$, this solution is exactly the 'half-whitening' solution from [17] for D*PCM. It is called 'half-whitening' because the pre-filter half-whitens the input so that:

$$S_{pp}(e^{j\omega}) = |H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})} \tag{4.21}$$

It is a counter-intuitive result because one would expect full-whitening to be optimal in some sense. This intuition is correct when noise feedback is possible, i.e. DPCM. In that case, full-whitening is the optimal solution and it has a lower MSE than optimal D*PCM. This result encourages a closer look at feedback structures for randomized sampling. They are considered in Section 4.7.

Though the solutions of Eqns.(4.19) and (4.20) are optimal, in general, these filters are non-causal and not implementable. For practical implementation, finite-order filters must be used. Mathematically, the design then requires optimization over the frequency response parametrized by the poles and zeros. In addition, because of the invertibility constraint the pre-filter and post-filter should be a complementary pair, where the poles and zeros cancel. For example, with a first-order FIR pre-filter and a complementary first-order IIR post-filter, the dfw-FRS design problem can be expressed as an optimization over a single parameter a_0 as:

$$\min_{a_0} \frac{\sigma_r^2}{\mu^2} \left(\int_{-\pi}^{\pi} |1 - a_0 e^{j\omega}|^2 S_{xx}(e^{j\omega}) \frac{d\omega}{2\pi} \right) \left(\int_{-\pi}^{\pi} \frac{|H(e^{j\theta})|^2}{|1 - a_0 e^{j\omega}|^2} \frac{d\theta}{2\pi} \right) \tag{4.22}$$

$$\text{subject to } |a_0| \leq 1 \ \forall \omega$$

In the numerical experiments of this chapter, we use a more approximate approach. Instead of using parametric models, we discretize the continuous frequency axis by uniformly sampling it with N points, i.e. with spacing $2\pi/N$. The frequency response is computed on the discrete grid and a N-point inverse DFT is used to get an FIR approximation to the optimal filter. Both the pre-filter and post-filter are thus approximated using FIR filters. Though not inverses of one another, with enough points, it is a close approximation. Section 4.6 explains the FIR approximation in more detail.

4.2.3 Error Analysis

The 'half-whitening' solution gives a shaping gain over white SRS by exploiting the spectral shape of $S_{xx}(e^{j\omega})$. Its performance is a lower bound for the other, more complex forms of FRS considered in this chapter. From the Cauchy-Schwartz inequality, the minimum dfw-FRS MSE is:

$$\mathcal{E}_{\text{dfw}} = \left(\frac{1}{\mu} - 1\right)\left(\int_{-\pi}^{\pi} \sqrt{S_{xx}(e^{j\omega})}|H(e^{j\omega})|\frac{d\omega}{2\pi}\right)^2 \tag{4.23}$$

The MSE scales with μ at the same rate as white SRS, i.e. $O(1/\mu - 1)$. The shaping gain over white SRS can be expressed as the quotient of Eqn.(3.17) and Eqn.(4.23):

$$\mathcal{G}_{\text{dfw}} = \frac{\left(\int_{-\pi}^{\pi} S_{xx}(e^{j\omega})d\omega\right)\left(\int_{-\pi}^{\pi} |H(e^{j\omega})|^2 d\omega\right)}{\left(\int_{-\pi}^{\pi} \sqrt{S_{xx}(e^{j\omega})}|H(e^{j\omega})|d\omega\right)^2} \tag{4.24}$$

The Cauchy-Schwartz inequality implies that the white SRS MSE is always greater than the distortion-free white FRS MSE, i.e. $\mathcal{G}_{\text{dfw}} \geq 1$. The worst-case shaping gain, $\mathcal{G}_{\text{dfw}} = 1$, occurs when the reconstruction filter has the same shape as the input power spectrum, i.e. $H(e^{j\omega}) = \alpha S_{xx}(e^{j\omega})$. For example, if $H(e^{j\omega})$ is an all-pass filter, then a white input has no shaping gain. If the filter has some structure though, shaping gain is possible even for a white input. Contrast this with SRS where no gain is possible if $S_{xx}(e^{j\omega})$ is white.

More shaping gain is possible as $S_{xx}(e^{j\omega})$ and $H(e^{j\omega})$ become more orthogonal to one another. Practically speaking, this amounts to a result like SRS: Since $H(e^{j\omega})$ is often close to an all-pass filter, the shaping gain increases as $S_{xx}(e^{j\omega})$ becomes more peaky. The error properties of dfw-FRS are illustrated alongside the other FRS techniques in Section 4.6 using numerical experiments.

4.3 Distortion-Free Frequency-Shaped FRS

In this section we discuss distortion-free frequency-shaped FRS (dffs-FRS) where the invertibility constraint of Eqn.(4.5) is imposed, but the sampling process can be correlated in time.

4.3.1 Design Problem

With $r[n]$ non-white, the power spectrum of $q[n]$, the output of randomized sampling, can be expressed as:

$$S_{qq}(e^{j\omega}) = \int_{-\pi}^{\pi} \Phi_{rr}(e^{j\omega})S_{xx}(e^{j(\omega-\theta)})|G_1(e^{j(\omega-\theta)})|^2\frac{d\theta}{2\pi} \tag{4.25}$$

The error spectrum after filtering with $g_2[n]$ and $h[n]$ is:

$$S_{ee}(e^{j\omega}) = |H(e^{j\omega})|^2|G_2(e^{j\omega})|^2\int_{-\pi}^{\pi} \Phi_{rr}(e^{j\omega})S_{xx}(e^{j(\omega-\theta)})|G_1(e^{j(\omega-\theta)})|^2\frac{d\theta}{2\pi} \tag{4.26}$$

As in the previous section, we assume that $H(e^{j\omega})$ is an invertible filter. Substituting Eqn.(4.10) and integrating Eqn.(4.26), the distortion-free frequency-shaped FRS MSE can be expressed as:

$$\mathcal{E} = \frac{1}{\mu^2}\int_{-\pi}^{\pi} \Phi_{rr}(e^{j\omega}) \left(\int_{-\pi}^{\pi} \frac{|H(e^{j\theta})|^2}{|G_1(e^{j\theta})|^2} S_{xx}(e^{j(\omega-\theta)}) |G_1(e^{j(\omega-\theta)})|^2 \frac{d\theta}{2\pi} \right) \frac{d\omega}{2\pi} \tag{4.27}$$

where we have swapped the order of integration. The goal in dffs-FRS is to design $\Phi_{rr}(e^{j\omega})$ and $G_1(e^{j\omega})$ such that this MSE in minimized. There are three constraints. First, because $r[n]$ is a binary process with a fixed mean, its variance must be constrained to $\sigma_r^2 = \mu(1-\mu)$. Secondly, $\Phi_{rr}(e^{j\omega})$ must be compatible with a binary process with mean μ, i.e. $\Phi_{rr}(e^{j\omega}) \in \mathcal{B}(\mu)$. In practice, this can be ensured by using the parametric Boufounos model of Section 3.3.3. Lastly, $|G_1(e^{j\omega})|^2$ must be positive for all ω since it is a magnitude. To simplify the notation, we define:

$$F(e^{j\omega}; |G_1|^2) = \int_{-\pi}^{\pi} \frac{|H(e^{j\theta})|^2}{|G_1(e^{j\theta})|^2} S_{xx}(e^{j(\omega-\theta)}) |G_1(e^{j(\omega-\theta)})|^2 \frac{d\theta}{2\pi} \tag{4.28}$$

As the notation suggests, $F(e^{j\omega}; |G_1|^2)$ takes a similar role as $F(e^{j\omega})$ in Eqn.(3.22), the objective function for frequency-shaped SRS. Here it is a function of $|G_1(e^{j\omega})|^2$, rather than a constant. Incorporating these constraints and simplifications, the design problem can be posed as a constrained optimization over $\Phi_{rr}(e^{j\omega})$ and $|G_1(e^{j\omega})|^2$:

$$\begin{array}{ll} \underset{\Phi_{rr}(e^{j\omega}), |G_1|^2}{\text{minimize}} & \frac{1}{\mu^2}\int_{-\pi}^{\pi} \Phi_{rr}(e^{j\omega}) F(e^{j\omega}; |G_1|^2) \frac{d\omega}{2\pi} \\ \text{subject to} & \int_{-\pi}^{\pi} \Phi_{rr}(e^{j\omega}) \frac{d\omega}{2\pi} = \mu(1-\mu) \\ & \Phi_{rr}(e^{j\omega}) \in \mathcal{B}(\mu) \\ & |G_1(e^{j\omega})|^2 \geq 0, \ \forall\omega \end{array} \tag{4.29}$$

The joint optimization is difficult, but the problem has a natural decomposition with respect to the decision variables $\Phi_{rr}(e^{j\omega})$ and $|G_1(e^{j\omega})|^2$. In the next section we propose an iterative algorithm that splits this optimization into two steps: (1) optimization of $\Phi_{rr}(e^{j\omega})$ for a fixed $F(e^{j\omega}; |G_1|^2)$ and (2) the optimization of $|G_1(e^{j\omega})|^2$ for a fixed $\Phi_{rr}(e^{j\omega})$. It is unclear whether this algorithm finds the optimal solution. Empirically though, it finds dffs-FRS designs that have a lower MSE than dfw-FRS.

In contrast with white-FRS, frequency-shaped FRS is fundamentally different from D*PCM. Specifically, in classical quantization theory, the additive error is either white or fixed to some colored spectrum. In dffs-FRS, on the other hand, we have control over the shape of the additive error spectrum. This extra degree of freedom allows us to further reduce the MSE. The duality with classical quantization theory breaks down. We must develop our own design algorithms.

4.3.2 Optimal Design

In this section we present an iterative design technique that attempts to find a solution to the optimization of Eqn.(4.29). We begin by presenting the steps of the algorithm followed by a detailed discussion.

We assume that the binary sampling process is generated by a Boufounos process with a fixed number of parameters. We denote the parametric dependence of $\Phi_{rr}(e^{j\omega})$ on a_k using the notation $\Phi_{rr}(e^{j\omega}; a_k)$. Additionally, the numerical optimization requires discretization of the continuous functions $\Phi_{rr}(e^{j\omega}; a_k)$ and $F(e^{j\omega}; |G_1|^2)$ onto a grid of frequencies. We assume that this is done densely so that the discrete solution is a close approximation to the desired continuous optimization. Alternatively, we can restrict the form of $G_1(e^{j\omega})$ to some fixed-order IIR or FIR filter and optimize over the parameters. Though important for practical implementation, we do not pursue such techniques here. Rather we approximate the ideal filters with an FIR approximation found by taking the inverse DFT of the discretized frequency response.

The steps of the iterative design technique are as follows:

1. Initialize $a_k^{(0)} = 0$ which in turn initializes $\Phi_{rr}^{(0)}(e^{j\omega}, a_k) = \sigma_r^2$, a white sampling process. Initialize $|G_1^{(0)}(e^{j\omega})|^2 = 1$, an all-pass filter.

2. Keeping $a_k^{(i)}$ and thus $\Phi_{rr}^{(i)}(e^{j\omega}, a_k)$ fixed, find $|G_{1,\text{opt}}^{(i+1)}(e^{j\omega})|^2$ as the solution to the partial optimization:

$$\boxed{\begin{aligned} &\underset{|G_1|^2}{\text{minimize}} && \frac{1}{\mu^2}\int_{-\pi}^{\pi} \Phi_{rr}^{(i)}(e^{j\omega}; a_k) F(e^{j\omega}; |G_1|^2)\frac{d\omega}{2\pi} \\ &\text{where} && F(e^{j\omega}; |G_1|^2) = \int_{-\pi}^{\pi} \frac{|H(e^{j\theta})|^2}{|G_1(e^{j\theta})|^2} S_{xx}(e^{j(\omega-\theta)})|G_1(e^{j(\omega-\theta)})|^2 \frac{d\theta}{2\pi} \\ &\text{subject to} && |G_1(e^{j\omega})|^2 \geq 0, \ \forall\omega \end{aligned}} \tag{4.30}$$

   For $i = 0$ and $\Phi_{rr}^{(i)}(e^{j\omega}, a_k)$ white, $|G_{1,\text{opt}}^{(i+1)}(e^{j\omega})|^2$ is the optimal distortion-free white solution, given by Eqn.(4.19). For all other i the optimization must be done numerically. Our implementation uses `fmincon` in MATLAB which uses a subspace trust-region method based on the interior-reflective Newton's method, [2].

3. Update the value of $|G_1(e^{j\omega})|^2$ to:

$$|G_1^{(i+1)}(e^{j\omega})|^2 = \lambda_g |G_1^{(i)}(e^{j\omega})|^2 + (1-\lambda_g)|G_{1,\text{opt}}^{(i+1)}(e^{j\omega})|^2 \tag{4.31}$$

   where $0 < \lambda_g < 1$. This is a relaxation step. The algorithm takes a step in the direction of the optimal solution but does not take the new optimal value. $|G_1^{(i+1)}(e^{j\omega})|^2$ is a feasible point by the convexity of the constraint set $|G_1(e^{j\omega})|^2 \geq 0, \ \forall\omega$. The value of λ_g can be optimized to get the best possible convergence rate. In our implementation we choose λ_g as a function of μ.

4. Keeping $|G_1^{(i+1)}(e^{j\omega})|^2$ and thus $F^{(i+1)}(e^{j\omega};|G_1|^2)$ fixed, find $a_{k,\text{opt}}^{(i+1)}$ as the solution to the partial optimization:

$$\boxed{\begin{aligned} &\underset{a_k}{\text{minimize}} && \frac{1}{\mu^2}\int_{-\pi}^{\pi}\Phi_{rr}(e^{j\omega};a_k)F^{(i+1)}(e^{j\omega};|G_1|^2)\frac{d\omega}{2\pi} \\ &\text{where} && \Phi_{rr}(e^{j\omega};a_k)=\frac{A}{|1-\sum_{k=1}^{p}a_ke^{-j\omega k}|^2} \\ & && A=\mu(1-\mu)\left(\int_{-\pi}^{\pi}\frac{1}{|1-\sum_{k=1}^{p}a_ke^{-j\omega k}|^2}\frac{d\omega}{2\pi}\right) \\ &\text{subject to} && \left(\sum_{k=1}^{p}a_k-1\right)>\frac{1}{|1-2\mu|}\left(\sum_{k=1}^{p}|a_k|-1\right) \end{aligned}} \tag{4.32}$$

This optimization is exactly the same as the one presented in Section 3.3.3 for frequency-shaped SRS. It must be done numerically. Similar to Step 2, our implementation uses `fmincon` in MATLAB.

5. Update the value of a_k to:

$$a_k^{(i+1)}=\lambda_a a_k^{(i)}+(1-\lambda_a)a_{k,\text{opt}}^{(i+1)} \tag{4.33}$$

where $0<\lambda_a<1$. This is another relaxation step. The algorithm takes a step in the direction of the optimal solution but does not take the new optimal value. $a_k^{(i+1)}$ is a feasible point by the convexity of the polyhedral constraint set $\left(\sum_{k=1}^{p}a_k-1\right)>\frac{1}{|1-2\mu|}\left(\sum_{k=1}^{p}|a_k|-1\right)$. In our implementation we choose λ_a as a function of μ.

6. Compute the new MSE as:

$$\mathcal{E}^{(i+1)}=\frac{1}{\mu^2}\int_{\pi}^{\pi}\Phi_{rr}^{(i+1)}(e^{j\omega};a_k)F^{(i+1)}(e^{j\omega};|G_1|^2)\frac{d\omega}{2\pi} \tag{4.34}$$

7. Go to Step 2 until the MSE improvement is below some pre-specified tolerance ϵ, i.e. $\mathcal{E}^{(i+1)}-\mathcal{E}^{(i)}<\epsilon$. Alternatively, we can terminate after a pre-specified maximum number of iterations.

This algorithm falls under a class of constrained optimization techniques called block coordinate descent. Such algorithms are useful in situations, like this one, where there is a natural decomposable structure to the problem. Block coordinate descent is described in detail in [2]. As proved in [2], block coordinate descent converges to a stationary point if each of the constraint sets are convex and if the minimum is uniquely attained for each sub-optimization. In our case, the constraint sets are convex but it is unclear that the minimum is uniquely achieved for each sub-optimization. Future work will address this issue. In practice though, we observe that our algorithm converges quickly to a stationary point.

The relaxation step is included because the direct block coordinate descent, with $\lambda_G=\lambda_a=0$, exhibits some pathological oscillatory behavior. In certain situations, the solution bounces around wildly, taking a long time to converge. With the relaxation step, the convergence becomes smoother and, in general, faster. Relaxation is a standard technique used in many optimization algorithms, from gradient descent to projection onto convex sets, [2, 30]. In our implementation, we choose λ_g and λ_a as the following functions of μ:

$$\lambda_g = -\left|\mu - \frac{1}{2}\right| + \frac{1}{2} \tag{4.35}$$

$$\lambda_a = 1 - \lambda_g \tag{4.36}$$

Figure 4-4 illustrates a graph of these relaxation parameters as a function of μ. Changing the λ as a function of μ allows us to encodes prior information about the expected optimal solution so that convergence is faster. In particular, when μ is near 1 or 0, the sampling process is more white. Consequently, we set $\lambda_a \approx 1$ so the solution is rigid near the white initial condition. Conversely, for these extremal μ, most of the gain comes from the filter, so we set $\lambda_g \approx 0$ so each step toward the optimal solution is larger. When $\mu = 1/2$, both parameters are the loosest with $\lambda_g = \lambda_a = 1/2$. In this case, the iterative optimization is not biased toward either solution.

Though our algorithm converges to a stationary point, it is unclear if this is the unique global minimum or just a local minimum. The optimization may have a convex structure, which would imply that the stationary point is the unique minimizer, but the issue remains unclear. A more detailed analysis of this iterative algorithm should address the convexity and uniqueness issue in more depth.

Empirically, our iterative algorithm converges to a local minimum in approximately 10 steps. Though reasonable, Steps 2 and 4 in this algorithm are expensive. A better algorithm would not compute the optimal solution for each sub-optimization but rather just find a suitable direction of descent. For example, a modified version of gradient projection or conditional gradient descent could be amenable for use in this problem, [2]. Additionally, if we can prove that the sub-optimizations are convex we can use powerful convex solvers to significantly speed up convergence, [7]. Further work on this algorithm can address these issues. As it stands, our iterative algorithm, though computationally burdensome, is suitable for the filter lengths of interest.

4.3.3 Error Analysis

Similar to frequency-shaped SRS, finding a closed-form for distortion-free frequency-shaped FRS is difficult because of the numerical optimization. The distortion-free white FRS MSE from Eqn.(4.23) gives an upper-bound on the dffs-FRS MSE. Accordingly, the dfw-FRS shaping gain gives a lower bound on the dffs-FRS shaping gain:

$$\mathcal{G}_{\text{dffs}} \geq \mathcal{G}_{\text{dfw}} \tag{4.37}$$

This bound is met with equality when $S_{xx}(e^{j\omega})$ is white and $H(e^{j\omega})$ is an all-pass filter. In this case, no amount of shaping by the filter or sampling process will improve the SNR.

We can use the SRS relaxation bound to get a lower bound to the shaping gain. Specifically, relaxing the binary achievability constraint for a fixed $G_1(e^{j\omega})$ the optimal autocovariance is $\Phi_{rr}(e^{j\omega}) = \pi\sigma_r^2\left(\delta(\omega - \omega_0) + \delta(\omega + \omega_0)\right)$ where $\omega_0 = \arg\min F(e^{j\omega}, |G_1|^2)$.

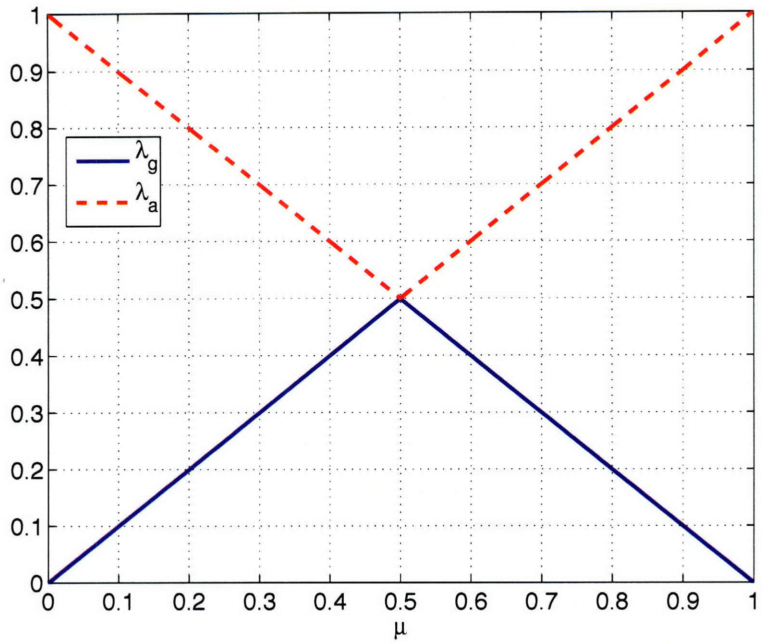


Figure 4-4. Relaxation parameters as a function of μ

The relaxation bound for a fixed $G_1(e^{j\omega})$ is thus:

$$\mathcal{E}_{\text{dfnb}}(|G_1|^2) = \left(\frac{1}{\mu} - 1\right) \min_{\omega} F(e^{j\omega}; |G_1|^2) \tag{4.38}$$

The full dffs-FRS relaxation bound can thus be expressed as a minimization over $|G_1(e^{j\omega})|^2$:

$$\mathcal{E}_{\text{dfnb}} = \left(\frac{1}{\mu} - 1\right) \min_{|G_1|^2} \left(\min_{\omega} F(e^{j\omega}; |G_1|^2) \right) \tag{4.39}$$

We can get a solution from numerical optimization, but we conjecture that a closed-form expression exists, although we have been unable to develop one. Further work should explore the existence of a closed form solution.

4.4 Unconstrained White FRS

In this section we discuss unconstrained white FRS (uw-FRS), the case in which the filters are not constrained to be inverses of one another but the sampling process is restricted to be white. Because it is less constrained than dfw-FRS, uw-FRS has a better MSE performance. The resulting error is biased and correlated with the desired output though. Under certain situations, this distortion may be an acceptable trade-off for a lower MSE.

4.4.1 Design Problem

Like dfw-FRS, uw-FRS has an exact correspondence with D*PCM. The quantizer analog to uw-FRS, where pre-filter and post-filter are not restricted to be inverses of one another, is studied in detail in a paper by Tuqan and Vaidyanathan, [31]. In this section, we follow the development from [31] to derive a tractable problem statement for the uw-FRS design problem.

In what follows, the filter $H(e^{j\omega})$ is assumed to be invertible, i.e. $H(e^{j\omega}) \neq 0$ for $|\omega| < \pi$. Without loss of generality, we can then interchange the order of the post-filters so that filtering with $h[n]$ occurs before filtering with $g_2[n]$. This is because both are LTI filters with the same pass-band. Define $v[n]$ as the output after the sampled process is interpolated by $h[n]$, but before $g_2[n]$:

$$v[n] = h[n] * q[n] \tag{4.40}$$

To develop the optimum closed-form solution, we first fix the pre-filter and optimize the post-filter. For a fixed pre-filter, $G_1(e^{j\omega})$, the optimum post-filter, $G_2^{\text{opt}}(e^{j\omega})$, is the optimal linear estimator of the process $y[n]$ from the data $v[n]$. Since $y[n]$ and $v[n]$ are jointly WSS, the optimal linear estimator is the non-causal Wiener filter given by:

$$G_2^{\text{opt}}(e^{j\omega}) = \frac{S_{yv}(e^{j\omega})}{S_{vv}(e^{j\omega})} \tag{4.41}$$

The denominator in Eqn.(4.41) can be expressed as:

$$\begin{aligned} S_{vv}(e^{j\omega}) &= |H(e^{j\omega})|^2 S_{qq}(e^{j\omega}) \\ &= |H(e^{j\omega})|^2 \left(\mu^2 |G_1(e^{j\omega})|^2 S_{xx}(e^{j\omega}) + \sigma_r^2 \sigma_p^2\right) \end{aligned} \tag{4.42}$$

where, as before, σ_p^2 is the signal variance after pre-filtering, defined as:

$$\sigma_p^2 = \int_{-\pi}^{\pi} |G_1(e^{j\omega})|^2 S_{xx}(e^{j\omega}) \frac{d\omega}{2\pi} \tag{4.43}$$

The numerator in Eqn.(4.41) can be expressed as:

$$\begin{aligned} S_{yv}(e^{j\omega}) &= S_{xv}(e^{j\omega}) H(e^{j\omega}) \\ &= S_{xq}(e^{j\omega}) |H(e^{j\omega})|^2 \\ &= \mu |H(e^{j\omega})|^2 G_1^*(e^{j\omega}) S_{xx}(e^{j\omega}) \end{aligned} \tag{4.44}$$

Substituting Eqns.(4.44) and (4.42) in to Eqn.(4.41) and simplifying, the optimal post-filter, given a fixed $G_1(e^{j\omega})$ can be expressed as:

$$G_2^{\text{opt}}(e^{j\omega}) = \frac{1}{G_1(e^{j\omega})} \cdot \frac{S_{xx}(e^{j\omega})}{\mu S_{xx}(e^{j\omega}) + (1-\mu)\frac{\sigma_p^2}{|G_1(e^{j\omega})|^2}} \tag{4.45}$$

which, as expected, has the same form as a Wiener filter for the MMSE estimation of $x[n]$ degraded by noise with the spectrum $\sigma_p^2/|G_1(e^{j\omega})|^2$. The first term in Eqn.(4.45) is the inverse filter, the distortion-free solution. The second term is a correction that can be made because the filters are not constrained to be inverses anymore.

Following the analysis of Tuqan and Vaidyanathan from [31], we can express the MSE as:

$$\begin{aligned}\mathcal{E} &= E\{e^2[n]\} = E\{e[n](y[n] - \hat{y}[n])\} \\ &= E\{e[n]y[n]\} = E\{(y[n] - \hat{y}[n])y[n]\} \\ &= R_{yy}[0] - E\{y[n]\hat{y}[n]\} \\ &= R_{yy}[0] - \sum_{k=-\infty}^{\infty} g_2[k]E\{y[n]\hat{v}[n-k]\}\end{aligned} \tag{4.46}$$

where we have used the orthogonality principle in the second line, i.e. $E\{e[n]\hat{y}[n]\} = 0$. Using Parseval's relation, the MSE can be expressed in the frequency-domain as:

$$\mathcal{E} = \int_{-\pi}^{\pi} \left(S_{yy}(e^{j\omega}) - S_{yv}^{*}G_2(e^{j\omega})\right) \frac{d\omega}{2\pi} \tag{4.47}$$

Substituting $S_{yv}^{*}(e^{j\omega})$ from Eqn.(4.44) and simplifying, the MSE is:

$$\mathcal{E} = \int_{-\pi}^{\pi} S_{yy}(e^{j\omega}) \left(1 - \mu G_1(e^{j\omega})G_2^{\text{opt}}(e^{j\omega})\right) \frac{d\omega}{2\pi} \tag{4.48}$$

Substituting $G_2^{\text{opt}}(e^{j\omega})$ from Eqn.(4.45) and σ_p^2 from Eqn.(4.43), the MSE can be expressed as a function of $|G_1(e^{j\omega})|^2$. After certain simplifications it can be expressed as:

$$\mathcal{E}(|G_1|^2;\mu) = \int_{-\pi}^{\pi} \frac{|H(e^{j\omega})|^2 S_{xx}(e^{j\omega}) \left(\frac{1}{\mu} - 1\right) \int_{-\pi}^{\pi} S_{xx}(e^{j\theta})|G_1(e^{j\theta})|^2 \frac{d\theta}{2\pi}}{S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 + \left(\frac{1}{\mu} - 1\right) \int_{-\pi}^{\pi} S_{xx}(e^{j\theta})|G_1(e^{j\theta})|^2 \frac{d\theta}{2\pi}} \frac{d\omega}{2\pi} \tag{4.49}$$

The goal in uw-FRS design is to minimize this objective function subject to the constraint that $|G_1(e^{j\omega})| \geq 0$. Trying to derive a closed-form solution from the objective Eqn.(4.49) is tedious and difficult. The problem can be transformed from this integral into an equality constrained optimization with a power constraint on the pre-filter output. From [31] the transformed optimization is:

$$\boxed{\begin{aligned} &\underset{|G_1(e^{j\omega})|^2}{\text{minimize}} \quad \int_{-\pi}^{\pi} \frac{\zeta |H(e^{j\omega})|^2 S_{xx}(e^{j\omega})}{S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 + \zeta} \frac{d\omega}{2\pi} \\ &\text{subject to} \quad \int_{-\pi}^{\pi} S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 \frac{d\omega}{2\pi} = 1 \end{aligned} \qquad (4.50)}$$

where $\zeta = \frac{1}{\mu} - 1$. This problem, in the class of isoperimetric calculus of variations problems, is more mathematically tractable and a closed-form expression can be obtained. The proof of this transformation comes from scale-invariance. It is detailed in [31]. Intuitively, because the solution to Eqn.(4.49) is invariant to scaling of the pre-filter output variance, σ_p^2, we can set this variance arbitrarily without loss of generality.

4.4.2 Optimal Design

In this section, we closely follow the steps in [31] to derive a closed-form solution for the optimal filter that minimizes Eqn.(4.50). The derivation is essentially the same as in [31] except for the additional term $H(e^{j\omega})$. We reproduce the key steps in the derivation for completeness but omit certain details. For a more detailed description, the reader is referred to [31].

To solve the constrained optimization of Eqn.(4.50), we use the calculus of variations. The first step is to incorporate the constraints using the Lagrangian:

$$\mathcal{L}(|G_1(e^{j\omega})|^2, \lambda, \beta(\omega)) = \int_{-\pi}^{\pi} \frac{\zeta |H(e^{j\omega})|^2 S_{xx}(e^{j\omega})}{S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 + \zeta} \frac{d\omega}{2\pi} + \\ \lambda \int_{-\pi}^{\pi} S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 \frac{d\omega}{2\pi} + \beta(\omega)|G_1(e^{j\omega})|^2 \quad (4.51)$$

There are two Lagrange multipliers in this problem for the two constraints: λ for the equality constraint and $\beta(\omega)$ for the non-negative inequality contraint. Note that the multiplier λ is independent of frequency, but $\beta(\omega)$ is a function of frequency. Tuqan and Vaidyanathan prove the frequency independence of λ in [31].

The Euler-Lagrange condition from the calclulus of variations gives a necessary condition for the stationary point, [31]. For our Lagrangian, the condition implies that the solution must satisfy the following equation at all frequencies:

$$\frac{\partial}{\partial |G_1(e^{j\omega})|^2} \left(\frac{\zeta |H(e^{j\omega})|^2 S_{xx}(e^{j\omega})}{S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 + \zeta} + \lambda S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 \right) = -\beta(\omega) \quad (4.52)$$

Here we give an abbreviated derivation of the solution from this expression with the additional $H(e^{j\omega})$ terms. The detailed steps and proofs are in [31]. As denoted in [31], there are two cases: $\beta(\omega) = 0$ and $\beta(\omega) \neq 0$. When $\beta(\omega) = 0$, the solution is the same as that from unconstrained minimization. In this case, the Euler-Lagrange condition can be expressed as:

$$\frac{\zeta |H(e^{j\omega})|^2 S_{xx}^2(e^{j\omega})}{(S_{xx}(e^{j\omega})|G_1(e^{j\omega})|^2 + \zeta)^2} + \lambda S_{xx}(e^{j\omega}) = 0 \quad (4.53)$$

Solving for $|G_1(e^{j\omega})|^2$ after some simplification gives,

$$|G_1(e^{j\omega})|^2 = \frac{1}{\sqrt{\lambda}} \sqrt{\frac{\zeta |H(e^{j\omega})|^2}{S_{xx}(e^{j\omega})}} - \frac{\zeta}{S_{xx}(e^{j\omega})} \quad (4.54)$$

Substituting Eqn.(4.54) back into the constraint of Eqn.(4.50), we can solve for the Lagrange multiplier $\sqrt{\lambda}$. After simplification it can be expressed as:

$$\sqrt{\lambda} = \left(\frac{\sqrt{\zeta}}{1+\zeta} \right) \int_{-\pi}^{\pi} |H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})} \frac{d\omega}{2\pi} \quad (4.55)$$

Substituting the value of $\sqrt{\lambda}$ into Eqn.(4.54) gives an expression for the optimal pre-filter as long as the value is non-negative:

$$|G_1^{\text{opt}}(e^{j\omega})|^2 = \frac{|H(e^{j\omega})|}{\sqrt{S_{xx}(e^{j\omega})}} \left(\frac{1+\zeta}{\int_{-\pi}^{\pi} |H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})} \frac{d\omega}{2\pi}} - \frac{\zeta}{|H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})}} \right) \tag{4.56}$$

if this expression is greater than zero for a particular ω. For $\beta(\omega) \neq 0$, as proved in [31], $|G_1^{\text{opt}}(e^{j\omega})|^2 = 0$. This is a form of the complementary slackness condition from Lagrange multiplier theory [2]. Combining the two results, the complete form of the optimum pre-filter can be expressed as:

$$|G_1^{\text{opt}}(e^{j\omega})|^2 = \max\left(0, \alpha \frac{|H(e^{j\omega})|}{\sqrt{S_{xx}(e^{j\omega})}} \left(\frac{1+\zeta}{\int_{-\pi}^{\pi} |H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})} \frac{d\omega}{2\pi}} - \frac{\zeta}{|H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})}} \right)\right) \tag{4.57}$$

where α is a scale factor that ensures that the constraint of Eqn.(4.50) is satisfied, i.e. that the variance $\sigma_p^2 = 1$. Mathematically, α is the solution to the expression:

$$\alpha \left[(1+\zeta) \left(\frac{\int_{\Omega} |H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})} \frac{d\omega}{2\pi}}{\int_{-\pi}^{\pi} |H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})} \frac{d\omega}{2\pi}} \right) - \zeta \right] = \sigma_p^2 = 1 \tag{4.58}$$

where Ω is the range of frequencies for which Eqn.(4.56) is non-negative, i.e. $|G_1(e^{j\omega})|^2 > 0$. Though the Euler-Lagrange solution gives only a necessary condition for a stationary point, Tuqan and Vaidyanathan prove in [31] that this is also a sufficient condition for this problem because of convexity. Consequently, this solution is a minimizing extremum of the objective function Eqn.(4.50). In addition, as long as $S_{xx}(e^{j\omega})$ and $H(e^{j\omega})$ are piece-wise smooth, the solution is guaranteed to be piece-wise smooth, [31]. Lastly, for the bands where the pre-filter $|G_1^{\text{opt}}(e^{j\omega})| = 0$, we can set the post-filter $|G_2^{\text{opt}}(e^{j\omega})| = 0$ without affecting the MSE.

Note that the phase response is not specified by Eqn.(4.57). From Eqn.(4.45) we observe that the MSE is minimized with respect to the phase response if the product $G_1(e^{j\omega})G_2^{\text{opt}}(e^{j\omega})$ has linear phase. Consequently, the phase response of $G_1^{\text{opt}}(e^{j\omega})$ should be complementary to the phase of $G_2^{\text{opt}}(e^{j\omega})$, up to a linear phase factor. In this thesis, we approximate $G_1^{\text{opt}}(e^{j\omega})$ and $G_2^{\text{opt}}(e^{j\omega})$ using linear-phase FIR filters, so this constraint is always satisfied. In general though, for parametric filters the phase complementarity must be imposed as a constraint between the two filters.

The pre-filter of Eqn.(4.57) has an intuitive interpretation. The filter is non-zero only when Eqn.(4.54) is non-negative. This implies that the pre-filter is zero where the signal energy is below a threshold dependent on σ_r^2, the sampling variance. In particular, rearranging Eqn.(4.54), the pre-filter is zero when:

$$\mu^2 |H(e^{j\omega})|^2 S_{xx}(e^{j\omega}) = \mu^2 S_{yy}(e^{j\omega}) < \lambda \sigma_r^2 \tag{4.59}$$

where $\lambda = (\sqrt{\lambda})^2$ is defined as in Eqn.(4.55). As quoted in [31]: *"It is therefore better not to transmit the signal at those frequencies where the [sampling] noise level is higher (by a certain threshold) than the signal level."*

Collecting the two filter expressions in one place, the optimal uw-FRS filters are,

$$|G_1^{\text{opt}}(e^{j\omega})|^2 = \max\left(0, \alpha\frac{|H(e^{j\omega})|}{\sqrt{S_{xx}(e^{j\omega})}}\left(\frac{1+\zeta}{\int_{-\pi}^{\pi}|H(e^{j\omega})|\sqrt{S_{xx}(e^{j\omega})}\frac{d\omega}{2\pi}} - \frac{\zeta}{|H(e^{j\omega})|\sqrt{S_{xx}(e^{j\omega})}}\right)\right) \tag{4.60}$$

$$G_2^{\text{opt}}(e^{j\omega}) = \frac{1}{G_1(e^{j\omega})}\cdot\frac{S_{xx}(e^{j\omega})}{\mu S_{xx}(e^{j\omega}) + (1-\mu)\frac{\sigma_p^2}{|G_1(e^{j\omega})|^2}} \tag{4.61}$$

These ideal filters are in general non-causal and not implementable. In practice, finite-order filters must be used. This requires optimization over the frequency response parametrized by the poles and zeros. In this thesis though, similar to dfw-FRS, we use the inverse DFT to find a FIR approximation of the optimal filters.

There are certain features of these filters to observe. First, when $\mu \approx 1$, the correction term in $G_2(e^{j\omega})$ is near unity and the post-filter is nearly the inverse of the pre-filter. Further, since the threshold is very low when $\mu \approx 1$, $\alpha \approx 1$ and $\zeta \approx 0$. Consequently, the optimal uw-FRS filters approach the dfw-FRS solutions,

$$|G_1^{\text{opt}}(e^{j\omega})|^2 \approx \frac{|H(e^{j\omega})|}{\sqrt{S_{xx}(e^{j\omega})}}\cdot\frac{1}{\int_{-\pi}^{\pi}|H(e^{j\omega})|\sqrt{S_{xx}(e^{j\omega})}\frac{d\omega}{2\pi}} \tag{4.62}$$

$$|G_2^{\text{opt}}(e^{j\omega})|^2 \approx \frac{1}{|G_1(e^{j\omega})|^2} \tag{4.63}$$

Consequently, at a high sampling rate uw-FRS does not have much benefit. As μ decreases the correction terms dominate and uw-FRS does better than dfw-FRS. This gain is quantified in the next section on error analysis. In the limit as $\mu \approx 0$, the threshold becomes very high and both filters approach zero. In this limit, the signal is not passed at all and the error is the signal itself.

4.4.3 Error Analysis

An expression for the uw-FRS MSE can be found by substituting the expression for the optimal pre-filter, Eqn.(4.57), back into the objective Eqn.(4.50). There are two distinct terms in the expression depending on whether the pre-filter is zero or not. We denote the band where $G^{\text{opt}}(e^{j\omega}) = 0$ with Ω and the complementary band, where it is non-zero, with Ω'. With some simplification the MSE can be expressed as:

$$\mathcal{E}_{\text{uw}} = \int_{\Omega'} S_{yy}(e^{j\omega})\frac{d\omega}{2\pi} + \left(\int_{-\pi}^{\pi}|H(e^{j\omega})|\sqrt{S_{xx}(e^{j\omega})}\frac{d\omega}{2\pi}\right) \tag{4.64}$$

$$\left(\int_{\Omega}\frac{|H(e^{j\omega})|^2 S_{xx}(e^{j\omega})}{\frac{\alpha}{1-\mu}|H(e^{j\omega})|\sqrt{S_{xx}(e^{j\omega})} + (1-\alpha)\int_{-\pi}^{\pi}|H(e^{j\omega})|\sqrt{S_{xx}(e^{j\omega})}\frac{d\omega}{2\pi}}\frac{d\omega}{2\pi}\right)$$

The first term corresponds to the distortion caused by removing the low-power bands with energy below the threshold. The second term is more complex. If the threshold of Eqn.(4.59) is low enough, none of the signal is nulled by the pre-filter, $\alpha = 1$, and $\Omega = \phi$, the empty set. In that case, the first term in Eqn.(4.64) is zero, and substituting $\alpha = 1$ in the second term, the MSE can be simplified to:

$$\mathcal{E}_{\text{uw}} = \mu \left(\frac{1}{\mu} - 1 \right) \left(\int_{-\pi}^{\pi} |H(e^{j\omega})| \sqrt{S_{xx}(e^{j\omega})} \frac{d\omega}{2\pi} \right)^2 = \mu \mathcal{E}_{\text{dfw}} \tag{4.65}$$

As noted, Eqn.(4.65) is exactly the same as the dfw-FRS MSE except for the multiplicative gain factor μ. The shaping gain in this case is:

$$\mathcal{G}_{\text{uw}} = \frac{1}{\mu} \mathcal{G}_{\text{dfw}} \tag{4.66}$$

This implies that $\mathcal{G}_{\text{uw}} > 1$, independent of input, except when $\mu = 1$. Consequently, even for a white input $S_{xx}(e^{j\omega})$ and all-pass filter $H(e^{j\omega})$, shaping gain is possible. As long as the energy in the white $S_{xx}(e^{j\omega})$ is above the threshold, the shaping-gain is $1/\mu$. By contrast, dfw-FRS and dffs-FRS has no shaping gain when the input is white and $H(e^{j\omega})$ is an all-pass filter. Extra gain is possible using uw-FRS because of the Wiener post-filter.

When $\mu \approx 1$, the uw-FRS performance is similar to that of dfw-FRS. The benefit of uw-FRS is greater at lower sampling rates. In the limit as $\mu \to 0$, the threshold becomes infinite and the MSE is dominated by the first term in Eqn.(4.64):

$$\lim_{\mu \to 0} \mathcal{E}_{\text{uw}} = \int_{-\pi}^{\pi} S_{yy}(e^{j\omega}) \frac{d\omega}{2\pi} \tag{4.67}$$

The error is finite and equal to the signal energy. Contrast this with dfw-FRS, where the MSE increases without bound because the invertibility constraint forces amplification of the error.

4.5 Unconstrained Frequency-Shaped FRS

Of the four techniques mentioned in the introduction, this thesis does not develop unconstrained frequency-shaped FRS (ufs-FRS) in detail. In ufs-FRS, the filters are not constrained to be inverses of one another and the sampling process can be correlated in time. Since it is the least constrained, ufs-FRS will have the best performance, as measured by MSE, of any of the randomized sampling methods we have considered. The resulting error will be biased and correlated with the desired output though. As with uw-FRS, this may be acceptable in certain circumstances.

Though not studied in detail, we propose an iterative optimization of the following form to design the filters and sampling process for ufs-FRS:

1. Initialize $G_1^{(0)}(e^{j\omega})$ and $G_2^{(0)}(e^{j\omega})$ to be all-pass filters.

2. Fix $G_1^{(i)}(e^{j\omega})$ and $G_2^{(i)}(e^{j\omega})$, compute the optimal SRS solution $\Phi_{rr}^{(i)}(e^{j\omega}, a_k)$ using a Boufounos process.

3. Fix $\Phi_{rr}^{(i)}(e^{j\omega}, a_k)$ and $G_1^{(i+1)}(e^{j\omega})$, compute $G_2^{(i+1)}(e^{j\omega})$ as the Wiener filter.

4. Fix $\Phi_{rr}^{(i)}(e^{j\omega}, a_k)$ and $G_2^{(i+1)}(e^{j\omega})$, compute $G_1^{(i+1)}(e^{j\omega})$ using an optimization routine. The optimization in this step has not been fully characterized yet.

5. Goto Step 2 until the iteration step defined by the squared error difference, $|G_1^{(i)}(e^{j\omega}) - G_1^{(i+1)}(e^{j\omega})|^2$, is less than a pre-specified tolerance ϵ.

We conjecture that this iterative algorithm will converge to a decent, perhaps even optimal, solution under the MMSE metric. Further work can study the performance and limitations of this algorithm in detail.

4.6 Numerical Experiments

In this section, numerical simulations are used to illustrate the spectral shaping and error scaling of the various forms of FRS. We also present simulations which illustrate the effect of $S_{xx}(e^{j\omega})$ on shaping gain.

4.6.1 Examples

The FRS example presented here is different from the SRS example presented in Section 3.5. This is because the context in which FRS is useful is different from when SRS is useful. Specifically, in our FRS example, Ω_x, the band-width of $S_{xx}(e^{j\omega})$ is equal to the band-width of the filter, Ω_h. By contrast in the SRS example, $\Omega_x > \Omega_h$. As noted in Section 4.1, because we can pre-filter any reasonable FRS implementation should remove the frequencies above Ω_h and set $\Omega_{\text{hr}} = \Omega_h$. Consequently, the SRS example where $\Omega_x > \Omega_h$ is not relevant to FRS.

In our FRS example, we assume a band-limited input, $x(t)$, with maximum frequency $\Omega_x = 200/2\pi$. The high-rate sampling frequency is fixed to its minimum value $\Omega_{\text{hr}} = 400/2\pi$, the Nyquist rate for $x(t)$. As shown in Section 3.4.2, this is the optimal value of Ω_{hr}. The resulting DT signal $x[n]$ on the high-rate grid is a WSS DT ARMA process generated by shaping white Gaussian noise through the fourth-order filter:

$$G(z) = \frac{(z-z_0)(z-z_1)(z-z_2)(z-z_2^*)}{(z-p_0)(z-p_1)(z-p_1^*)(z-p_2)(z-p_2^*)} \tag{4.68}$$

$$z_0 = 0.7,\ z_1 = -0.7,\ z_2 = 0.9e^{j\pi/2}\ p_0 = 0.2,\ p_1 = 0.9e^{j\pi/4},\ p_2 = 0.9e^{j3\pi/4}$$

The power spectrum is $S_{xx}(e^{j\omega}) = G(z)G(z^{-1})$. The continuous-time filter, $h(t)$, is assumed to be an ideal LPF with cutoff $\Omega_h = 200$ Hz. In discrete-time, after sampling at $\Omega_{\text{hr}} = 400/2\pi$, this becomes an all-pass filter. The average sampling rate is fixed to $\Omega_{\text{avg}} = 220/2\pi$, which is below the Nyquist rate. The mean downsampling rate is thus fixed to $\mu = \Omega_{\text{avg}}/\Omega_{\text{hr}} = 11/20$. Figure 4-5 illustrates the discrete-time power spectrum $S_{xx}(e^{j\omega})$. This example has been constructed to illustrate a situation where FRS may be useful. The sampling rate is below the Nyquist rate, so uniform sampling leads to aliasing. On the other hand, simple anti-aliasing removes the second peak and potential information of interest there.

Two million samples of the additive error, $w[n]$, are generated for each test case. Periodogram averaging with a Hamming window of size 2048 with 50% overlap is used to approximate 2048 samples of the power spectrum $S_{ww}(\omega)$. The MSE is estimated numerically by averaging the squared difference between $y[n]$ and $\hat{y}[n]$ after filtering.

Figure 4-5(a) shows the result of uniform sampling at the rate $\Omega_{\text{avg}} = 220/2\pi$. Note the strong aliases in the band of interest. Fig.4-5(b) shows the result of white-SRS with $\Omega_{\text{hr}} = 400/2\pi$ and $\mu = 11/20$. The noise has a flat power spectrum with height given by Eqn.(3.16).

Figure 4-6(a) shows the result of distortion-free white FRS. The filters, $G_1(e^{j\omega})$ and $G_2(e^{j\omega})$ are approximated using 2048-point linear-phase FIR filters. They are computed by taking the inverse DFT of a sampled version of the optimal frequency responses given by Eqn.(4.19) and (4.20). The dfw-FRS SNR is 3.113 dB, which is about 2.2 dB better than white SRS. The noise has been shaped into the peaks of $S_{xx}(e^{j\omega})$. The empirical results closely match the theoretical predictions. Figure 4-8 illustrates the results of dfw-FRS design. Figures 4-8(a) and 4-8(b) illustrate the magnitude responses $|G_1(e^{j\omega})|$ and $|G_2(e^{j\omega})|$. Figure 4-8(c) illustrates the magnitude response of the cascade $|G_1(e^{j\omega})||G_2(e^{j\omega})|$. Because of the invertibility constraint, the cascade is an all-pass filter with gain $1/\mu$ across the band.

Figure 4-6(b) shows the result of distortion-free frequency-shaped FRS. The filters are approximated using 2048-point linear-phase FIR filters. They are found using the discretized relaxed block coordinate descent algorithm presented in Section 4.3.2. Figure 4.6.1 illustrates how the MSE converges to a stationary point quickly. Note the smooth descent due to the relaxation. The dffs-FRS SNR is 5.01 dB, which is about 2 dB better than dfw-FRS because the sampling process shifts the error spectrum into the nulls of $|G_2(e^{j\omega})|$. Figure 4-9 illustrates the filters resulting from the relaxed block coordinate descent. They are very similar to the dfw-FRS optimal filters, except slightly more pinched in near the peaks. Intuitively, the filters emphasize the peaks so the sampling process can better shift the error into the nulls of $|G_2(e^{j\omega})|$. Figure 4-11 illustrates the properties of the dffs-FRS sampling process. In this example, we restricted the order of the Boufounos process to $p = 4$. Figure 4-11(a) illustrates $F(e^{j\omega}; |G_1|^2)$ along with the optimal $\Phi_{rr}(e^{j\omega})$. The peaks of $\Phi_{rr}(e^{j\omega})$ are in the troughs of $F(e^{j\omega}; |G_1|^2)$. Figure 4-11(b) illustrates a realization of the sampling process $r[n]$. It has mean $\mu = 11/20$ and a clear quasi-periodic structure.

Figure 4-7 shows the result of unconstrained white FRS. There are three curves plotted, because the error is correlated with the input. The uncorrelated portion is still given by $S_{ww}(e^{j\omega})$. There is an additional correlated portion though. It can be visualized by plotting the power spectrum of the signal through the filter cascade, $\mu^2|G_1(e^{j\omega})|^2|G_2(e^{j\omega})|^2S_{xx}(e^{j\omega})$. In distortion-free FRS, this signal would be equivalent to $S_{xx}(e^{j\omega})$, but in the unconstrained case it is not. In this sense, it represents the distortion caused by the filter cascade. The SNR of uw-FRS is 5.586 dB, which is better than both dfw-FRS and dffs-FRS for this example. Figure 4-12 illustrates the optimal uw-FRS filters. Unlike dfw-FRS, the unconstrained filters have stop-bands. Figure 4-12(c) illustrates the frequency response of the cascade. The filters clearly are not inverses of one another. The filters emphasize the peaks of the signal, where the power is higher. They also remove all power below the threshold of Eqn.(4.59). Figure 4-12(c) plots this threshold along with $S_{yy}(e^{j\omega})$. As expected, the filters are zero where $S_{yy}(e^{j\omega})$ below the threshold. It is not worth transmitting the signal below this power.

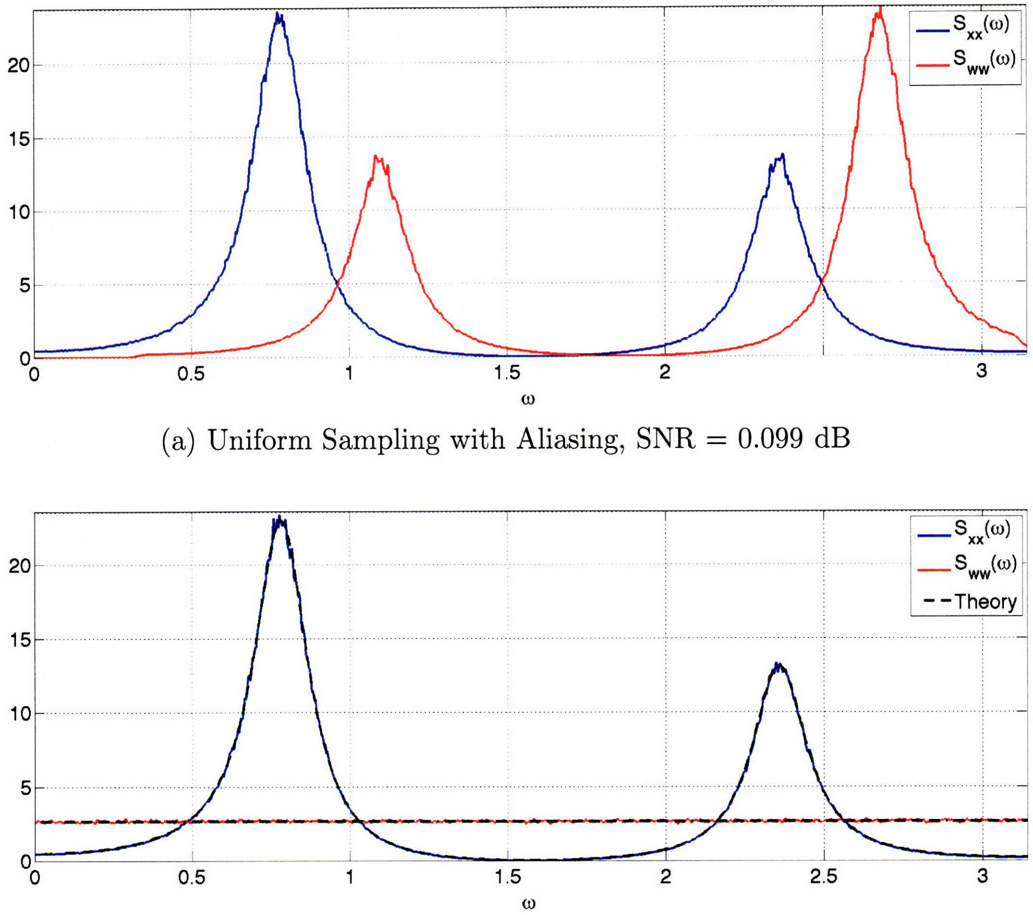
Figure 4-13(a) illustrates the SNR scaling of the various techniques as a function of μ, for this example. Figure 4-13(b) illustrates the same results in terms of shaping gain. There are an number of observations to make. First, all of the techniques have similar shaping gain when $\mu \approx 1$. The curves diverge as μ decreases. As expected, dfw-FRS has a constant shaping gain of $\mathcal{G}_{\text{dfw}} = 2.32$ dB over white-SRS, independent of μ, from Eqn.(4.24). Dffs-

FRS always has a higher shaping gain dfw-FRS. Similar to frequency-shaped SRS, it peaks when $\mu = 1/2$, and in general is higher when μ has mid-range values. As μ approaches extremal values, performance becomes similar to dfw-FRS.

The unconstrained FRS curves have a qualitatively different shape. The shaping gain is low when μ is close to one, but monotonically increases as μ decreases. For the regime $1/2 < \mu < 1$, the uw-FRS shaping gain is a linear dB factor above the dfw-FRS shaping gain. This is expected because in this regime, the threshold is low and from the approximation of Eqn.(4.66),

$$\begin{aligned} 10\log_{10}\mathcal{G}_{\text{uw}} &\approx 10\log_{10}\mathcal{G}_{\text{dfw}} + 10\log_{10}(1/\mu) \\ &\approx 10\log_{10}\mathcal{G}_{\text{dfw}} - 10\log_{10}(\mu) \end{aligned} \tag{4.69}$$

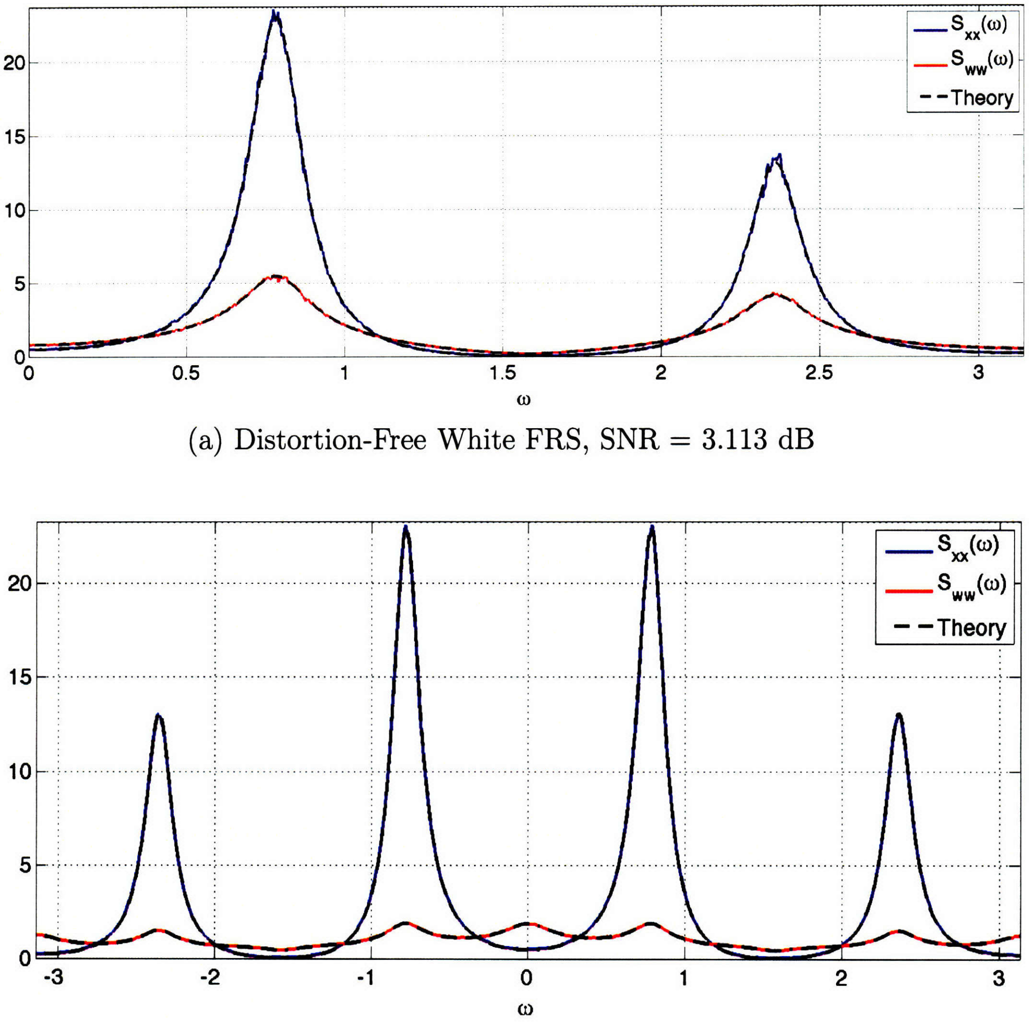
where $\mu < 1$, so $-10\log_{10}(\mu) > 0$, i.e. there is positive gain over $\mathcal{G}_{\text{dfw}}$. In addition, the performance of uw-FRS is always above dffs-FRS for this particular input. It is unclear if that is the case for all inputs. Note that unlike the distortion-free FRS and white SRS the uw-FRS SNR approaches 0 dB as $\mu \to 0$, rather than $-\infty$.



(a) Uniform Sampling with Aliasing, SNR = 0.099 dB

(b) White SRS, SNR = 0.871 dB

Figure 4-5. Uniform and white-SRS randomized sampling for the example discussed in Section 4.6. Average sampling rate fixed to $\Omega_{\text{avg}} = 220/2\pi$. The reconstruction filter $H(e^{j\omega})$ is assumed to be unity across the band.



(a) Distortion-Free White FRS, SNR = 3.113 dB

(b) Distortion-Free Frequency-Shaped FRS, SNR = 5.006 dB

Figure 4-6. Distortion-free filtered randomized sampling for the example discussed in Section 4.6. Average sampling rate fixed to $\Omega_{\text{avg}} = 220/2\pi$. The reconstruction filter $H(e^{j\omega})$ is assumed to be unity across the band.

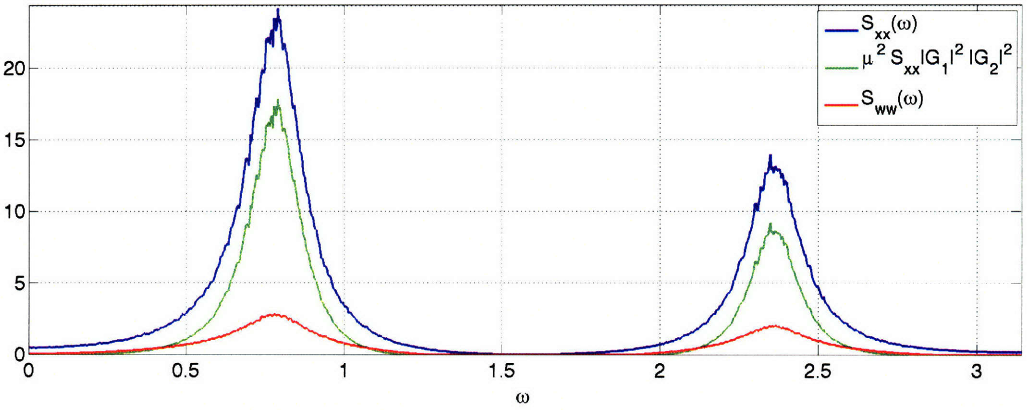
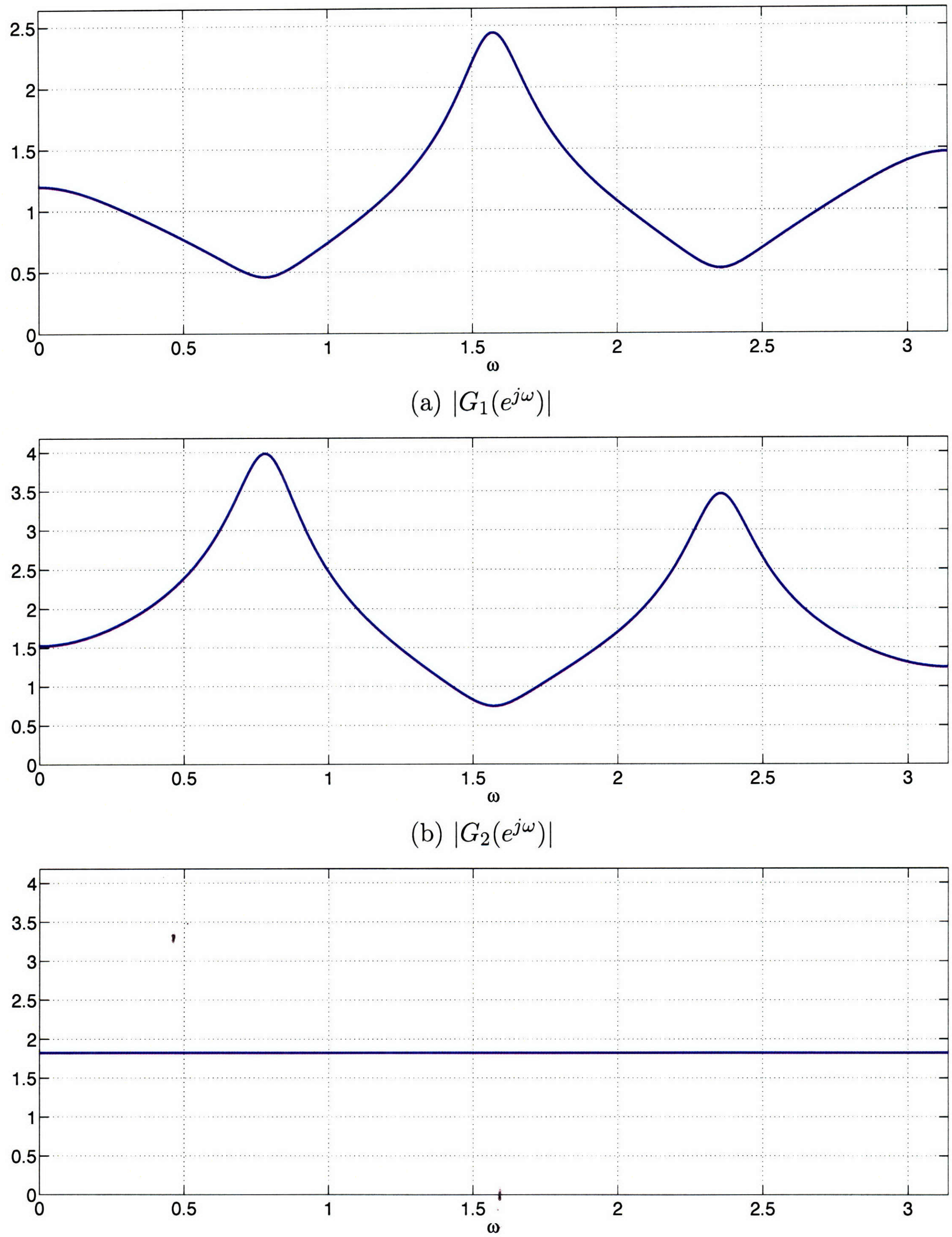


Figure 4-7. Unconstrained white FRS for example discussed in Section 4.6. SNR = 5.586 dB. Average sampling rate fixed to $\Omega_{\text{avg}} = 220/2\pi$. The reconstruction filter $H(e^{j\omega})$ is assumed to be unity across the band.

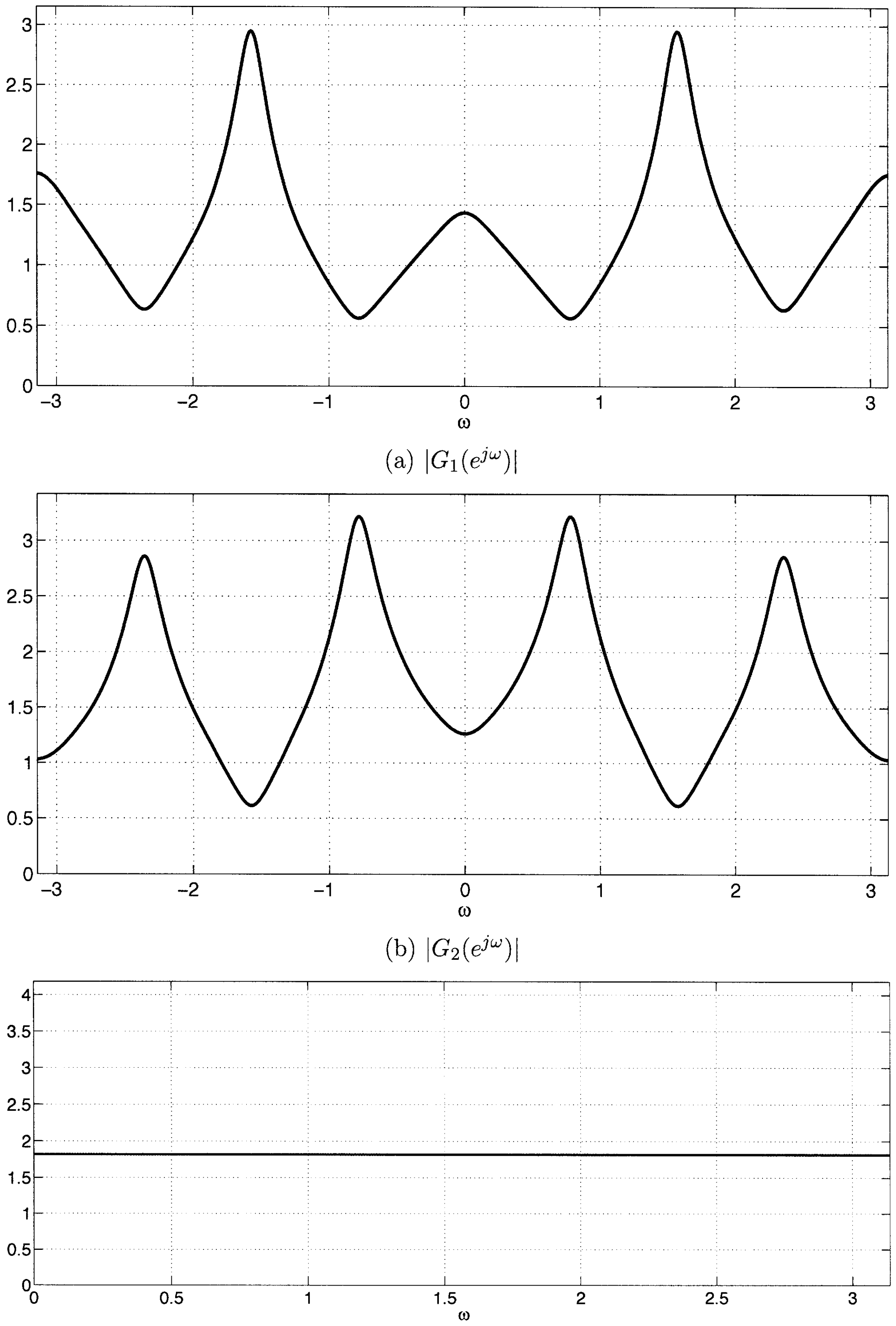


(a) $|G_1(e^{j\omega})|$

(b) $|G_2(e^{j\omega})|$

(c) Cascade $|G_1(e^{j\omega})||G_2(e^{j\omega})|$

Figure 4-8. Distortion-free white FRS filter for the example discussed in Section 4.6. Ideal filters are approximated using 2048-pt linear phase FIR filters.



(a) $|G_1(e^{j\omega})|$

(b) $|G_2(e^{j\omega})|$

(c) Cascade $|G_1(e^{j\omega})||G_2(e^{j\omega})|$

Figure 4-9. Distortion-free frequency-shaped FRS filters for the example discussed in Section 4.6. Filters are found using relaxed block coordinate descent on discrete frequency grid. The filters are approximated using 2048-pt linear phase FIR filters.

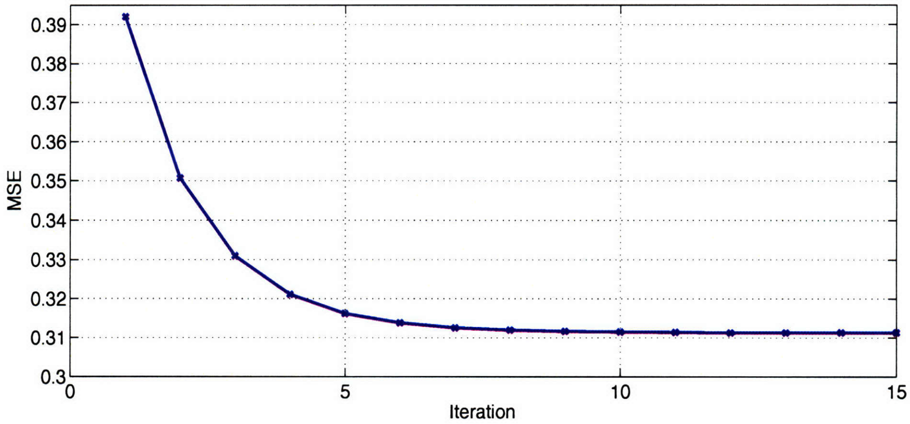
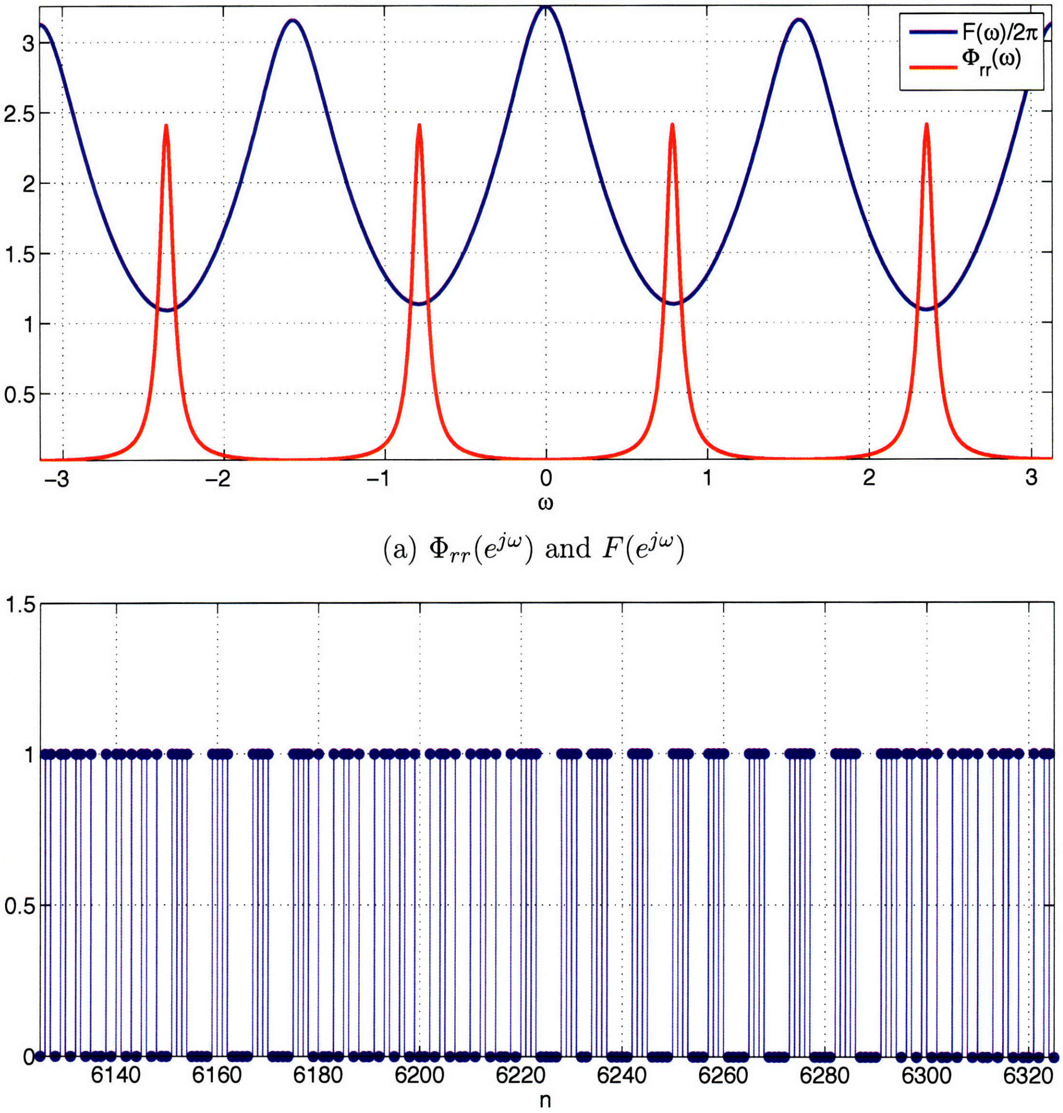


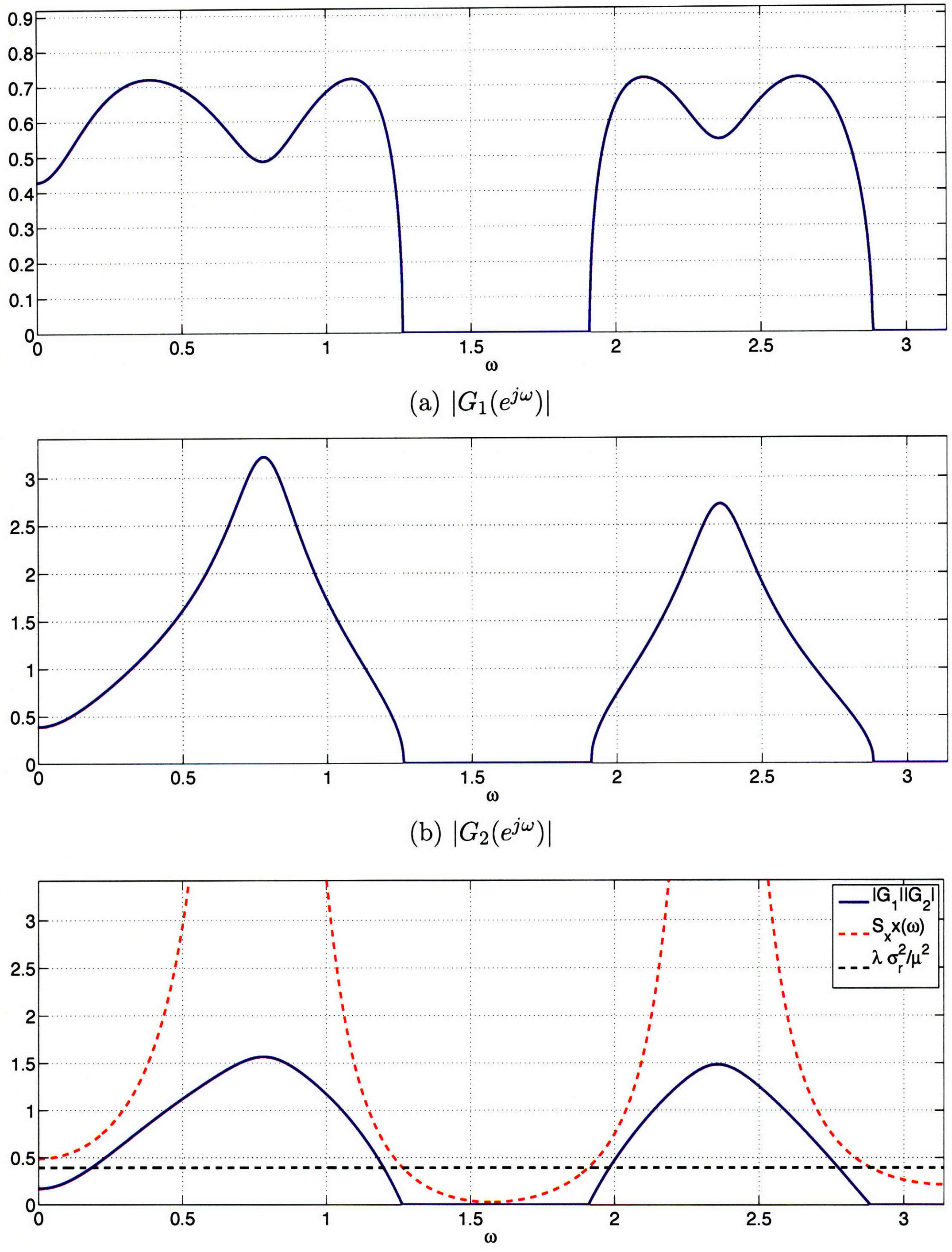
Figure 4-10. Relaxed block coordinate descent error for distortion-free frequency-shaped FRS design. MSE plotted agains iteration. Stationary point achieved in about 10 iterations.



(a) $\Phi_{rr}(e^{j\omega})$ and $F(e^{j\omega})$

(b) Realization of $r[n]$

Figure 4-11. Distortion-free frequency-shaped FRS binary-AR characteristics for the example discussed in Section 4.6. Parameters optimized using relaxed block coordinate descent.

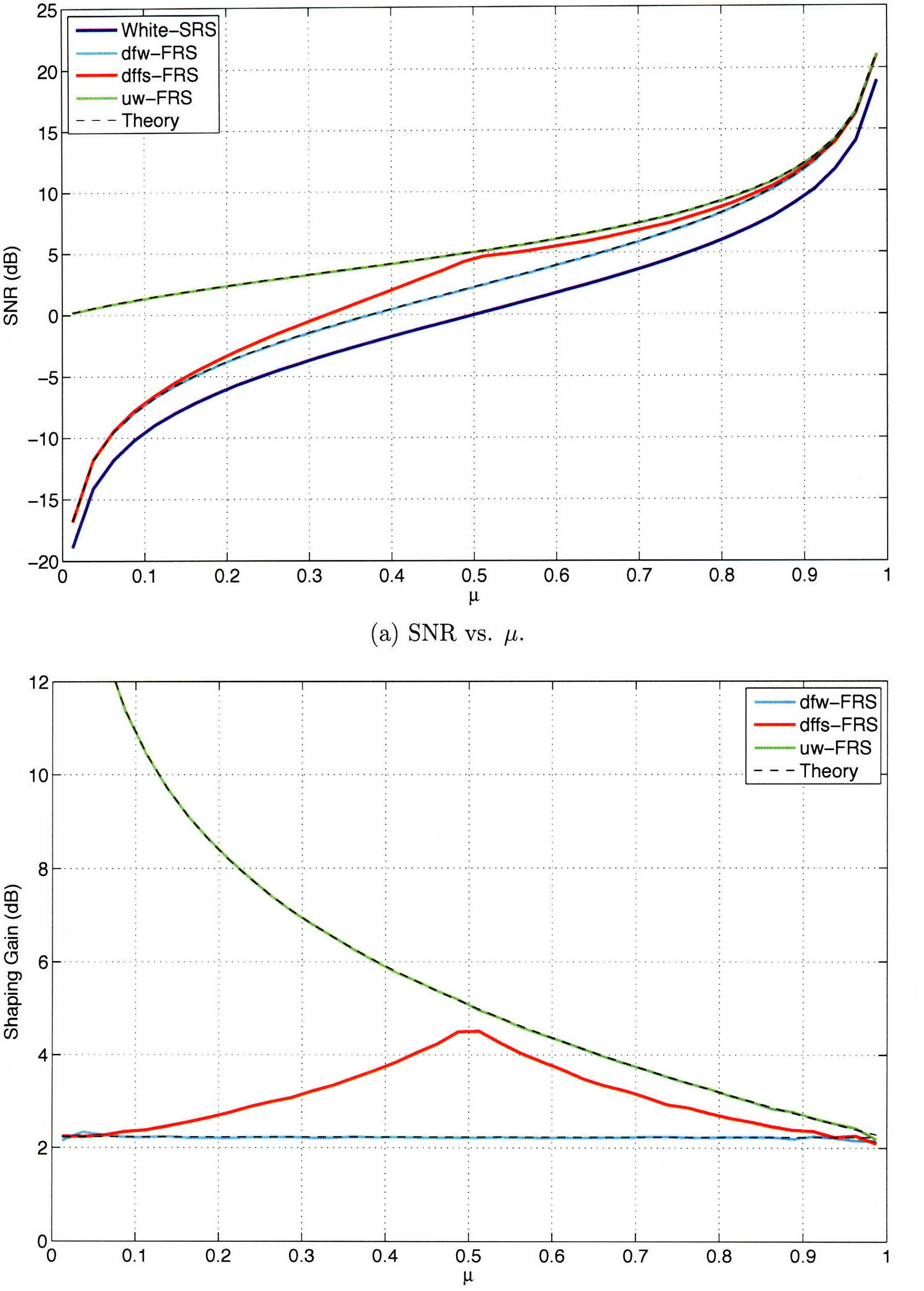


(a) $|G_1(e^{j\omega})|$

(b) $|G_2(e^{j\omega})|$

(c) Cascade $|G_1(e^{j\omega})||G_2(e^{j\omega})|$

Figure 4-12. Unconstrained white FRS filters for the example discussed in Section 4.6. Ideal filters are approximated using 2048-pt linear phase FIR filters.



(a) SNR vs. μ.

(b) Shaping gain vs. μ.

Figure 4-13. SNR scaling as a function of μ for example discussed in Section 4.6. $f_{\mathrm{hr}} = 400$ Hz, the Nyquist rate for the filter cutoff Ω_h.

4.6.2 Peakiness

Similar to SRS, we illustrate the effect of peakiness on FRS performance using an MA(1) and AR(1) experiment. For both, the reconstruction filter $H(e^{j\omega})$ is a unity gain all-pass filter and the average sampling rate is fixed to $\mu = 11/20$. In the first experiment, $x[n]$ is a first-order moving average MA(1) process with power spectrum,

$$S_{xx}(e^{j\omega}) = (1 - \theta e^{-j\omega})(1 - \theta e^{j\omega}) \tag{4.70}$$

Figure 4-14 plots the FRS shaping gain for the various FRS techniques as θ is varied from zero to one. Each point on the curve is found numerically by computing the shaping gain from one-million samples. The dffs-FRS curve is jagged because the numerical optimization has some variation in the solution it finds, i.e. running the optimization from different initial conditions leads to different final solutions. There are a number of observations to make. When $\theta = 0$ the spectrum is white and the same as $H(e^{j\omega})$, an all-pass filter. Consequently the dfw-FRS and dffw-FRS shaping gain is 0 dB. As θ increases the spectrum becomes less like $H(e^{j\omega})$. Accordingly, the dfw-FRS shaping gain is higher. The filter $|G_2(e^{j\omega})|$ also has a deeper null when $S_{xx}(e^{j\omega})$ is peakier. Consequently, the gain over dfw-FRS due to frequency shaping increases with θ. The shaping gain flattens our near $\theta = 1$ becuase an MA(1) process is not infinitely peaky, even when the zero is on the unit circle.

For the MA(1) process, most of the input signal is above the uw-FRS threshold of Eqn.(4.59) for all the values of θ. The approximation of Eqn.(4.66) holds and mathematically:

$$10\log_{10}\mathcal{G}_{\text{uw}} = 10\log_{10}\mathcal{G}_{\text{dfw}} + 10\log_{10}(1/\mu) = 10\log_{10}\mathcal{G}_{\text{dfw}} + 2.6 \tag{4.71}$$

This is exactly what is observed in Figure 4-14: the uw-FRS shaping gain is a constant 2.6 dB above the the dfw-FRS gain, independent of θ. Note that this is true even for $\theta = 0$, a white input.

In the second experiment $x[n]$ is a first-order auto-regressive AR(1) process with power spectrum,

$$S_{xx}(e^{j\omega}) = \frac{1}{(1 - \rho e^{-j\omega})(1 - \rho e^{j\omega})} \tag{4.72}$$

Figure 4-15 plots the FRS shaping gain for the various FRS techniques as ρ is varied from zero to one. Each point on the curve is found by numerical simulation of one-million samples. As before, the dffs-FRS curve is jagged because of variation in the solution from numerical optimization. When $\rho = 0$, the spectrum is white and the same as $H(e^{j\omega})$. The dfw-FRS and dffw-FRS shaping gain is 0 dB. As ρ increases the pole moves closer to the unit circle, making $S_{xx}(e^{j\omega})$ more peaky. The dfw-FRS and dffs-FRS shaping gain increase exponentially. As with the MA(1) experiment, dffs-FRS is always above dfw-FRS and the additional gain increases as $S_{xx}(e^{j\omega})$ becomes more peaky. Unlike the MA(1) experiment though, the AR(1) process becomes infinitely peaky as the pole approaches the unit circle, so the shaping gain does not flatten out near $\rho = 1$.

At low values of ρ, the uw-FRS shaping gain is still 2.6 dB over dfw-FRS. With more peakiness, the gain becomes lower because a significant portion of the signal is removed by $G_1(e^{j\omega})$, i.e. the low-threshold approximation is not valid anymore. At extremely high peakiness, $\rho \approx 1$, the uw-FRS and dfw-FRS gain become similar.

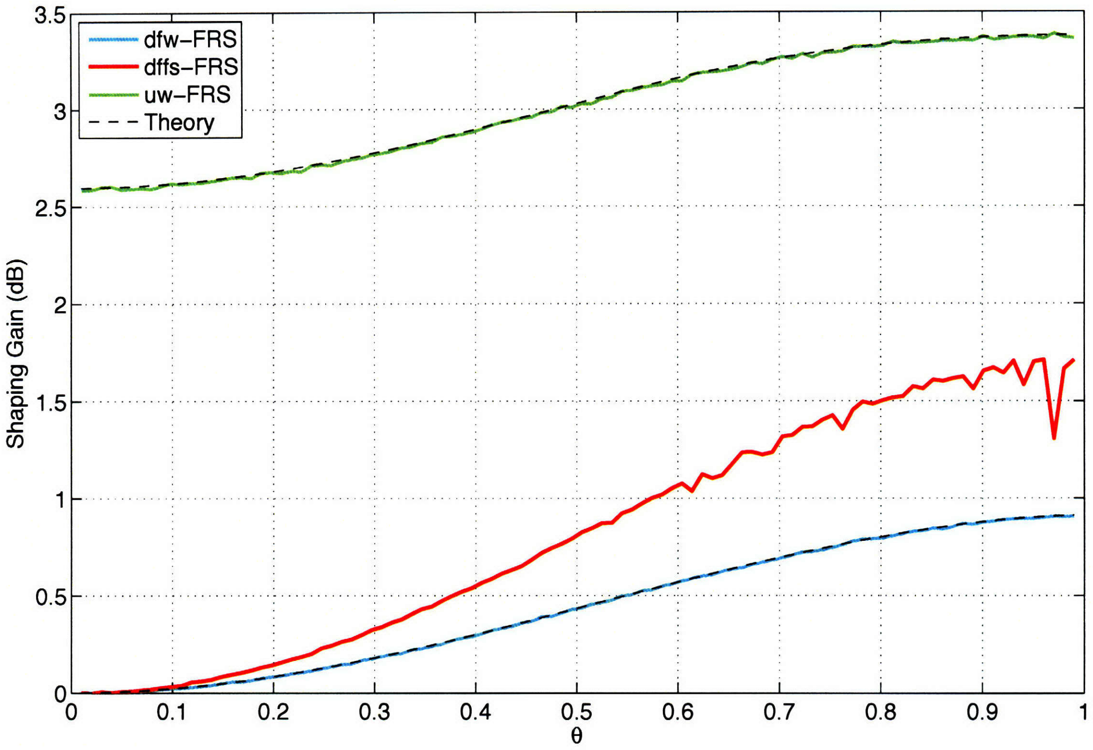


Figure 4-14. Shaping gain of MA(1) process defined in Eqn.(4.70) as a function of θ. $\mu = 11/20$.

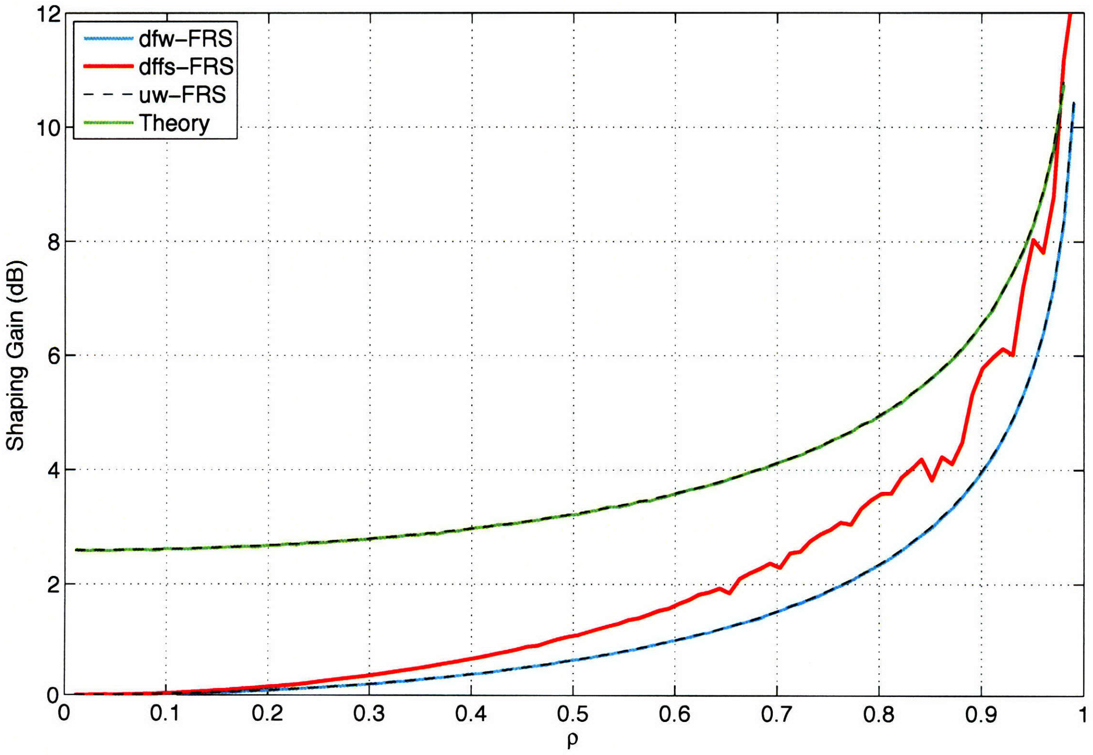


Figure 4-15. Shaping gain of AR(1) process defined in Eqn.(4.72) as a function of ρ. $\mu = 11/20$.

4.7 Extensions

In this chapter, we have explored the use of LTI filtering to reduce randomized sampling error. A large part of the theory is analogous to classical quantization theory. From this duality, we can consider more complex structures with feedback. Such techniques, like DPCM and Σ-Δ noise-shaping, are among the most useful quantizer architectures. Further work could potentially use analogous coding techniques for randomized sampling. One possible architecture is illustrated in Figure 4-16. It is inspired by DPCM. The sampled signal is fed back and used to "encode more information" on the next sample that will be taken. It is unclear what this means at this point, but we conjecture that using such a technique, more shaping gain can be realized.

The literature on LTI erasure compensation, in particular [12, 6], may also be relevant to the design of FRS feedback structures. In essence, the feedback should do some sort of pre-compensation for the information erased by the randomized sampler. In this sense, the problems are similar. On the other hand, the erasure compensation literature primarily considers signals in an oversampled, frame-based setting, whereas in this case we are in the undersampled regime. In any case, these are all potential directions for future work in FRS.

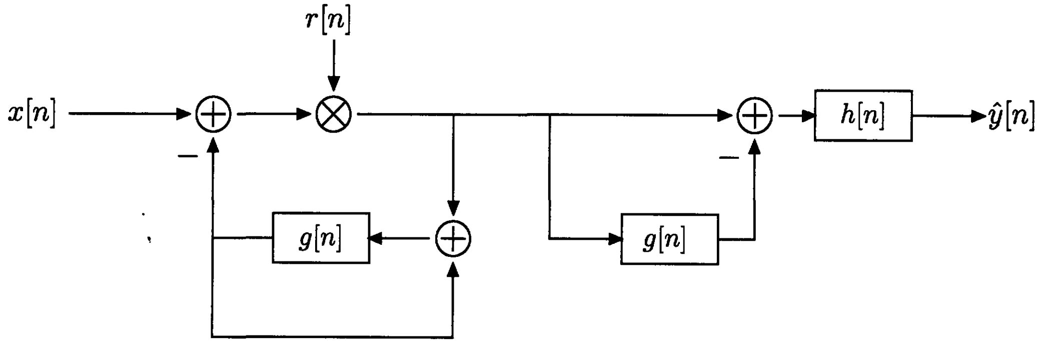


Figure 4-16. Possible feedback FRS structure inspired by DPCM.

Chapter 5

Randomized Multiplier-Less Filtering

This chapter introduces randomized multiplier-less filtering as a method to mitigate coefficient quantization error in multiplier-less filter structures. The formulation is a vector extension of the randomized sampling techniques of the previous chapters. This chapter begins by motivating the use of randomization. Section 5.2.1 introduces one particular structure, the Direct Form I binary randomized filter, that is studied in this thesis. Section 5.3 discusses potential extensions of randomization to other filter structures.

5.1 Introduction

Multiplier-less filter implementations are desirable because they reduce hardware complexity and speed up computation. One method to implement multiplier-less filters is to do static one-bit coefficient quantization in a filter structure. However, the resulting coefficient quantization error significantly distorts the frequency response from the desired continuous-valued one. This distortion can be disturbing in certain applications, especially perceptual ones.

As an example, Figure 5-1 illustrates the frequency response of a Parks-McClellan filter and a one-bit quantized version of it. Though qualitatively similar, the quantized filter introduces a significant distortion in the frequency response. This distortion may be acceptable for approximate filtering operations, but becomes increasingly problematic for filters where the finer structure of the response is important, e.g. equalization, beamforming.

Static multiplier-less filters have been studied extensively in the literature. There are two general techniques that have been developed to mitigate frequency response distortion. The first is smart-factoring of the desired filter into multiplier-less subsections. A good example of this technique is the class of cascaded-integrator-comb (CIC) filters used in sampling rate conversion, [16, 20]. These filters cascade multiplier-less IIR integrator stages, $H_I(z) = 1/1 - z^{-1}$, with multiplier-less FIR comb stages, $H_C(z) = 1 - \sum_{i=0}^{N} z^{-iL}$ to create approximate low-pass filters. Though the frequency response of these CIC filters are far from ideal, in practice they are suitable for interpolation and anti-aliasing in low-fidelity rate conversion, [16, 20].

The second technique used to mitigate frequency response distortion is to replace multiplies with bit shifts, [21]. Unlike a multiply, bit shifting has little cost in terms of hardware area and latency. Effectively, filters with bit-shifts have multiplies that are factors of two. These shift-register filters are often combined with smart factoring to achieve a subset of possible filters, [21].

Though these static multiplier-less techniques are useful and can achieve a subset of possible filters exactly, the essential problem of frequency response distortion still exists. Given an arbitrary desired continuous-valued filter, static multiplier-less filters in general cannot achieve the frequency response. In certain applications, like approximate resampling, the frequency response distortion may be acceptable, but in other, especially perceptual applications, this distortion can be problematic. In this thesis, we propose a new paradigm for multiplier-less filtering using randomization of the filter coefficients. It mitigates many of the problems of frequency response distortion seen in static multiplier-less filters.

The basic principle in our randomized multiplier-less filtering technique is to replace the desired continuous-valued coefficients in a filter structure with a multiplier-less random process that has the same value on average. The replacement can be done in any filter structure, e.g. direct form, cascaded, parallel, lattice, etc. The performance, design, and error analysis of BRF is different in each structure though. In this thesis, we focus on the simplest structure: Direct Form I FIR filters. In more complex structures there are additional issues of stability, error feedback, and error propagation that must be addressed. Though restrictive, Direct Form I is a filter structure that is useful in a broad range of applications, especially ones where linear phase is important. Potential extensions to other filter structures are briefly discussed in Section 5.3.

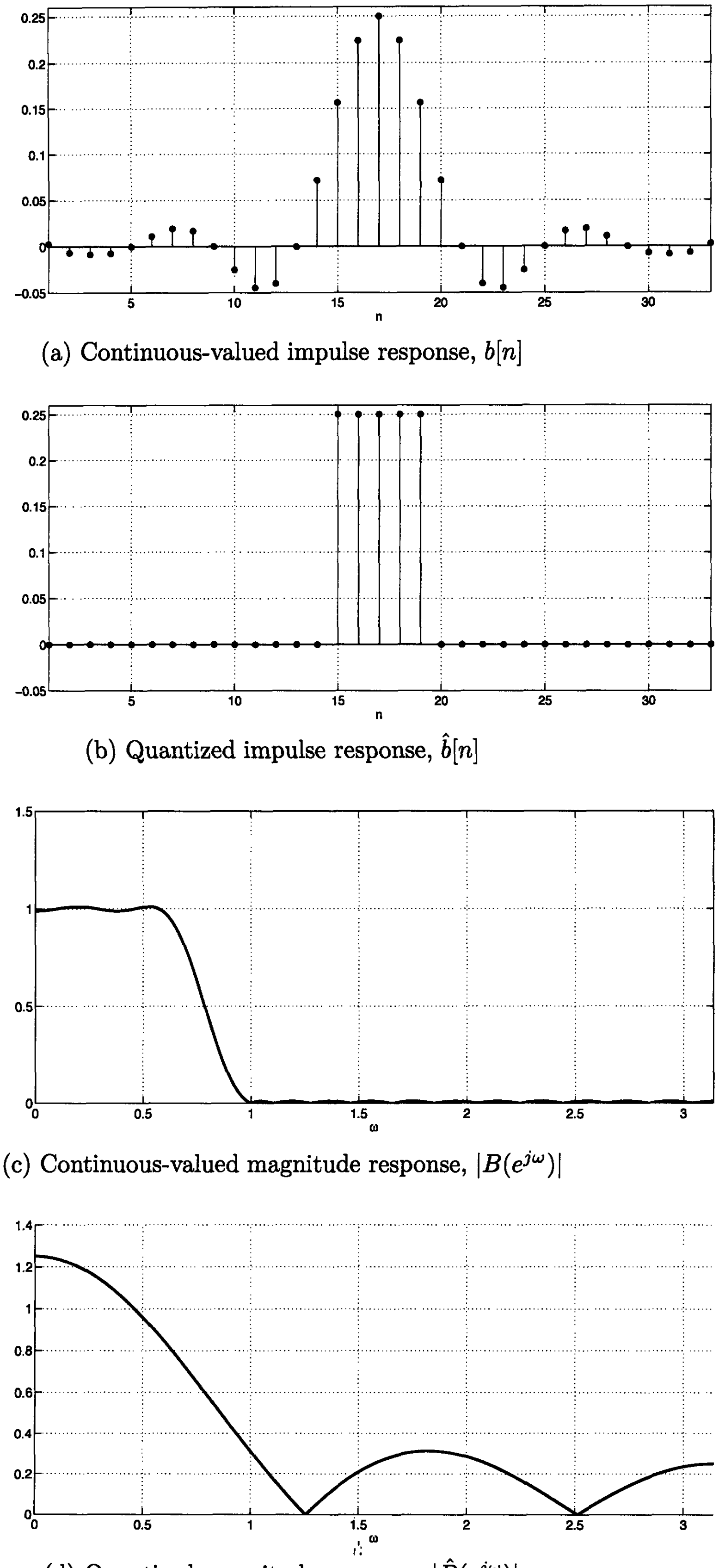
Randomization can be done in both the one-bit quantized or shift-register type multiplier-less filters. In this thesis, we focus on randomization of one-bit quantized binary coefficients where the coefficients can only be 1, 0, or -1. The principles developed for binary randomization can be extended to shift-register type filters in straightforward manner. Such extensions are briefly discussed in Section 5.3. A hierarchical tree diagram showing the various types of randomized multiplier-less filters is illustrated in Figure 5-2. The focus of this thesis is on the bold boxed parts of the tree.

5.2 Binary Randomized Filtering

We denote our binary multiplier-less randomized scheme as Direct Form I FIR binary randomized filtering, or BRF for short. The basic principle in BRF is to replace the desired continuous-valued coefficients with a binary random process that has the same value on average. Section 5.2.1 develops the Direct Form I BRF structure, introducing the two basic forms: standard and oversampled BRF. Section 5.2.2 does a brief analysis of the randomization error, discussing its potential benefits over static coefficient quantization. Section 5.2.3 introduces the different types of BRF depending on tap and time correlation.

5.2.1 Direct Form I FIR BRF Model

We assume that a specified FIR filter, with N continuous-valued taps, $\{b_i\}_{i=0}^{N-1}$, is given in Direct Form I. Such a filter can be designed using any technique, e.g. Parks-McClellan for linear-phase design, Yule-Walker for Wiener filter design, etc. Figure 5-3(a) illustrates the tapped delay line Direct Form I FIR filter structure, [24]. This is denoted the continuous-valued counterpart of the BRF. Figure 5-3(b) illustrates the binary randomized filter structure. Each continuous-valued tap, b_i, has been replaced by a binary random process, $h_i[n] = \{0, 1\}$, perhaps followed by a sign change if necessary, $s_i = \text{sgn}(b_i) = \{-1, 1\}$. Effectively, positive-valued taps are represented by binary processes that takes values 0 and



(a) Continuous-valued impulse response, $b[n]$

(b) Quantized impulse response, $\hat{b}[n]$

(c) Continuous-valued magnitude response, $|B(e^{j\omega})|$

(d) Quantized magnitude response, $|\hat{B}(e^{j\omega})|$

Figure 5-1. Frequency distortion caused by binary quantization of impulse response. Desired response is 33 tap linear-phase low-pass filter designed using the Parks-McClellan algorithm.

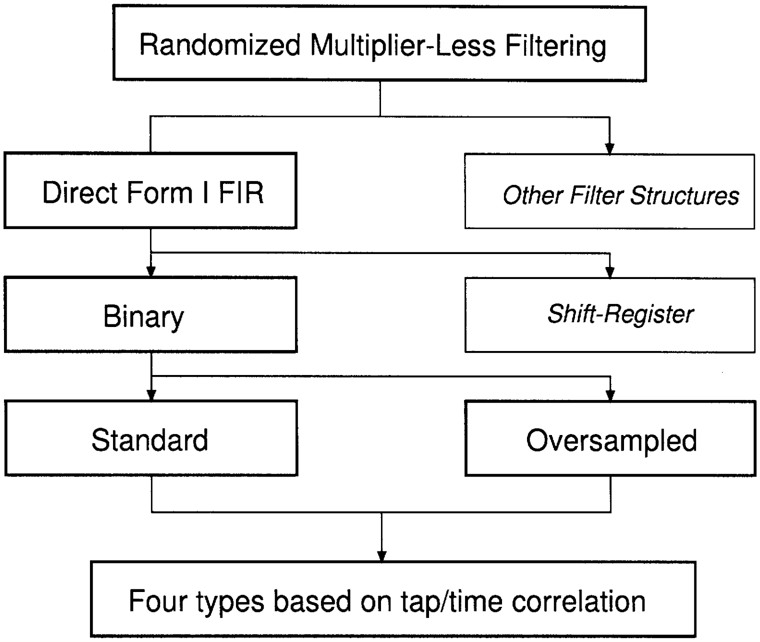


Figure 5-2. Hierarchy of multiplier-less randomized filters. The types of filters developed in this thesis are illustrated in bold boxes. Those not developed in detail are in italics.

1, and negative-valued taps are represented by binary processes that take values 0 and −1. Note that, since each tap is a function of time, the BRF is effectively a time-varying filter.

Since the binary taps have a restricted range, an overall scaling, K, after filtering, is included so that the BRF has the same gain as its continuous-valued counterpart. This scaling can be implemented digitally as a single high-fidelity multiply or as an analog gain element after the digital stage. We assume it is a positive value, $K > 0$. In this sense, BRF has a single multiply but all the tap multiplies have been replaced with multiplier-less switching processes.

The means of the binary tap processes, $h_i[n]$, are constrained such each has the same mean value as the desired continuous-valued counterpart, i.e. they are constrained such that:

$$KE\{h_i[n]\} = Ks_i\mu_i = b_i \tag{5.1}$$

Since the mean of a binary process is bounded below by 0 and above by 1, the scaling K must be chosen so that all the means are in the range $\mu_i \in [0, 1]$. This implies the constraint:

$$K \geq \max\{|b_i|\} = ||\mathbf{b}||_\infty \tag{5.2}$$

With this constraint the filter on average has the desired continuous-valued response, i.e. the expected value of the BRF response is that of the desired continuous-valued filter. There is error due to the binary randomization though. The next section discusses the structure of this error further.

In terms of hardware complexity, the BRF architecture trades multiplication for randomization. This thesis does not explore the hardware implementation details of this trade off, but generally speaking the BRF architecture will have a significantly lower complexity. In particular, the randomized binary taps in a BRF can be generated with minimal overhead using pseudo-random numbers generated from a linear shift feedback register (LSFRs) and combinational logic. The sign changes in BRF can be implemented with negligible cost as a set of bit flips, depending on the digital representation of $x[n]$.

The structure developed above is termed standard BRF. We develop another more advanced structure denoted oversampled BRF. Figure 5-3(c) illustrates a block diagram of the oversampled BRF architecture. It has three significant changes from standard BRF. First, the BRF is preceded by an L-fold upsampling stage composed of an expander and an interpolation filter, $g_u[n]$. Second, the tapped delay-line is expanded, i.e. the unit-delays in the tapped-delay line are replaced with L-element delays. Lastly, the BRF is followed by a L-fold downsampling stage composed of an anti-aliasing filter $g_d[n]$ followed by a compressor. The hardware complexity of this architecture is higher, requiring the implementation of rate-converters, which can have multipliers or can be multiplier-less, and a larger memory buffer. But, as discussed in Chapter 7, this structure has a better error performance than standard BRF. In fact the performance improves monotonically as a function of L. The oversampled BRF implementation is developed in Chapter 7.

5.2.2 Randomization Error

With the mean constraint, BRF has the same response on average as its continuous-valued counterpart. There is however error due to binary randomization. In this section, we study the properties of this error and its benefits over frequency response distortion.

We assume a stochastic model for the error analysis. In particular, we assume the input $x[n]$ is a continuous-valued WSS random process and the taps, $h_i[n]$, are WSS binary processes independent of the input. Each tap process can be expressed as:

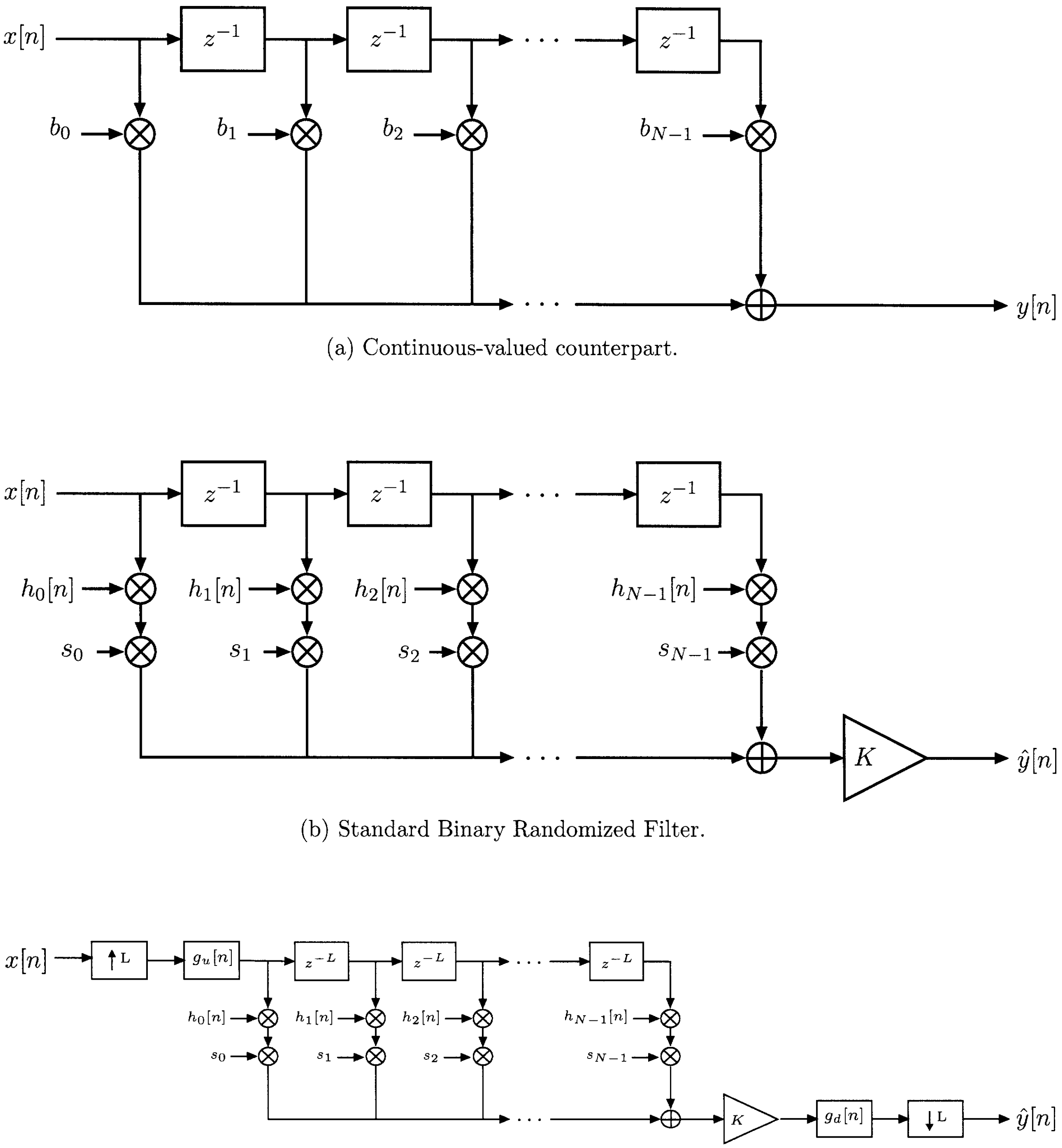
$$h_i[n] = \mu_i + \tilde{h}_i[n] \tag{5.3}$$

where $\tilde{h}_i[n]$ is the centered, zero-mean WSS random process with $\tilde{h}_i[n] = \{1 - \mu_i, -\mu_i\}$. With this decomposition, the output of the BRF, $\hat{y}[n]$ can be expressed as:

$$\begin{aligned}
\hat{y}[n] &= K \sum_{i=0}^{N-1} h_i[n] s_i x[n-i] \\
&= \sum_{i=0}^{N-1} \underbrace{(K\mu_i s_i)}_{b_i} x[n-i] + K \sum_{i=0}^{N-1} \tilde{h}_i[n] s_i x[n-i] \\
&= \underbrace{\sum_{i=0}^{N-1} b_i x[n-i]}_{y[n]} + \underbrace{K \sum_{i=0}^{N-1} \tilde{h}_i[n] s_i x[n-i]}_{e[n]}
\end{aligned} \tag{5.4}$$

The output is the sum of the desired output, $y[n]$, from the desired continuous-valued filter $b[n]$, and the error $e[n]$, from the zero-mean time-varying filter with kernel $\tilde{h}_i[n]$.

The error, $e[n]$, is a zero-mean WSS process that is uncorrelated with the input signal, $x[n]$. This follows in a straightforward manner using the independence of $h_i[n]$ and $x[n]$.



(a) Continuous-valued counterpart.

(b) Standard Binary Randomized Filter.

(c) Oversampled Binary Randomized Filter.

Figure 5-3. Block diagrams of Direct Form I FIR filter implementations (a) the continuous-valued counterpart, (b) the BRF implementation and, (c) an over-sampled BRF implementation.

The bias can be computed as:

$$E\{e[n]\} = K \sum_{i=0}^{N-1} \underbrace{E\{\tilde{h}_i[n]\}}_{0} s_i E\{x[n-i]\} = 0 \tag{5.5}$$

which proves $e[n]$ is zero-mean. In addition, the cross-correlation of $e[n]$ with the input $x[n]$ is:

$$E\{e[n]x[n]\} = K \sum_{i=0}^{N-1} \underbrace{E\{\tilde{h}_i[n]\}}_{0} s_i E\{x[n-i]x[n]\} = 0 \tag{5.6}$$

i.e. $e[n]$ is uncorrelated with the input. Intuitively, as in randomized sampling, the randomization of the coefficients has the effect of decorrelating the error with the input. The error is WSS and the auto-correlation is,

$$\begin{aligned} E\{e[n]e[n+m]\} &= E\left\{ \sum_{i=0}^{N-1} \tilde{h}_i[n+m] s_i x[n+m-i] \sum_{j=0}^{N-1} \tilde{h}_j[n] s_j x[n-j] \right\} \\ &= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} E\left\{\tilde{h}_i[n]\tilde{h}_j[n+m]\right\} s_i s_j E\left\{x[n-i]x[n+m-j]\right\} \end{aligned} \tag{5.7}$$

This auto-correlation is a function of the time correlation in the binary tap processes $h_i[n]$. There are two classes of BRF, memoryless and frequency-shaped, depending on how this time correlation is constrained. In memoryless BRF, the taps $h_i[n]$ are restricted to be independent in time. This implies,

$$E\left\{\tilde{h}_i[n]\tilde{h}_j[n+m]\right\} = \begin{cases} \sigma_{ij} & \text{for } m = n \\ 0 & \text{for } m \neq n \end{cases} \tag{5.8}$$

Substituting Eqn.(5.8) into Eqn.(5.7), the error auto-correlation can be expressed as,

$$R_{ee}[m] = K^2 \sum_{i=0}^{N-1} \sum_{j=1}^{N-1} \sigma_{ij} s_i s_j R_{xx}[i-j]\delta[m] \tag{5.9}$$

which implies that $e[n]$ is a white process. With frequency-shaped BRF, the taps $h_i[n]$ can be correlated in time to frequency-shape the error spectrum in a potentially desirable manner, e.g. outside the band of interest.

This is the benefit of BRF. As opposed to frequency response distortion in static multiplier-less filtering, the error is uncorrelated with the input and we have much greater control over it. In certain applications, especially perceptual ones, the uncorrelated shapeable randomization error may be preferable to the frequency response distortion caused by static multiplier-less filters. The effect is similar to that of dither in waveform coding to break up cyclical quantization errors. In a sense, BRF can be viewed as a form of dither in multiplier-less filters.

5.2.3 Types of Binary Randomized Filters

The correlation of the tap processes play a central role in the performance of BRF. For simplicity, we can express all of the tap processes at once as a binary vector process $\mathbf{h}[n]$, defined as,

$$\mathbf{h}[n] = \begin{bmatrix} h_0[n] & h_1[n] & \cdots & h_{N-1}[n] \end{bmatrix} \tag{5.10}$$

This vector process can be correlated both in time and across the taps. As mentioned in previous section, there are two types of BRF depending on the time correlation: memoryless and frequency shaped. In addition to the time correlation, the taps can be correlated with each other at a fixed time n. There are two types of BRF depending on this tap correlation, tap-independent and tap-correlated. Unlike the time-correlation, the tap-correlation does not affect the shape of the error spectrum, but it can reduce the total error power at the output.

As the name implies, in tap-independent BRF all of the taps are independent processes. Tap-independent BRF sets a baseline for performance and is easy to design. Correlation across the taps can reduce the total error at the output. Intuitively, the correlation of $x[n]$ can be used to design the tap processes so that there is partial cancellation of the error at the accumulator. The design of tap-correlated BRFs is significantly more complex though. It involves the design of vector binary processes with tractable auto-covariance matrices. Their design is non-trivial and requires the development of parametric models for the generation of vector binary processes. These models and their application to tap-correlated BRF design is presented in Chapter 6.

Combining the two types of filters, there are four types of BRF : tap-independent memoryless BRF, tap-correlated memoryless BRF, tap-independent frequency-shaped BRF, and tap-correlated frequency-shaped BRF. The four types of BRF are summarized in Table 5.1 along with the abbreviations we use and the section that develops them. In this thesis, we only develop Direct Form I FIR memoryless filtering, both tap-independent and tap-correlated, in detail. Their design and error analysis is presented in Chapter 6. We briefly mention the benefits of frequency shaping BRF with regard to oversampled BRF in Chapter 7, but do not develop it in detail. Further work should perform a detailed study of frequency-shaped BRF.

	Memoryless $\mathbf{h}[n]$ independent in time	**Frequency-Shaped** $\mathbf{h}[n]$ correlated in time
Tap-Independent $\mathbf{h}[n]$ independent across taps	Section 6.2 tim-BRF	briefly in 7.2.3 tifs-BRF
Tap-Correlated $\mathbf{h}[n]$ correlated across taps	Section 6.3 tcm-BRF	briefly in 7.2.3 tcfs-BRF

Table 5.1. Types of binary randomized filtering

5.3 Extensions

The techniques developed in this thesis can be extended to other filter structures and multi-level coefficient quantization. In this section, we present a brief overview of these two potential extensions. Though not developed in detail, it shows that BRF is more broadly applicable than the narrow context in which is developed in this thesis.

5.3.1 Other Filter Architectures

This thesis only randomizes coefficients in the Direct Form I FIR structure. This is done for simplicity because the Direct Form I FIR structure only has a feed-forward path for the error. The randomization of coefficients can be done in any filter structure though. One important case to consider in future work is BRF for Direct Form I or II IIR filter structures with feedback. Such structures will benefit from BRF in the same way as Direct Form I FIR filters, but there is an additional issue of error feedback. Specifically, the error will be shaped by the feedback path of the filter. This could be problematic if the error is placed at undesirable frequencies. In addition, the scaling factor K is accumulated through an IIR BRF, so it can potentially drive the filter unstable if not designed properly. Future work should consider the design, error analysis, and numerical simulation of IIR binary randomized filtering in detail.

Cascaded architectures are another important case to consider in future work. A filter can be implemented as cascaded stages. BRF in such a cascaded implementation would have error propagation, i.e. the error of the first stage would be shaped by the next, etc. This error propagation could be used as an advantage however. Using frequency-shaping BRF, the error of the first stage could be shaped into the stop-band of the next stage and so on. If frequency-shaping and cascading are designed properly, the error could be significantly reduced from a Direct Form implementation. Future work should consider such cascaded design, perhaps developing certain rules of thumb for cascaded implementation.

5.3.2 Multi-Level Coefficient Quantization

Though this thesis focuses on binary multiplier-less filters, the BRF theory developed can potentially be extended to general multi-level coefficient quantization. For example, assume that the input, $x[n]$, is continuous-valued and that the coefficient quantization is uniform with step-size Δ. The coefficient quantization error can then be modeled mathematically as,

$$\tilde{b}_i = b_i - Q(b_i) \tag{5.11}$$

where $Q(\cdot)$ is a uniform quantizer with step-size Δ, b_i is the desired tap-value, and $\tilde{b}_i$ is the coefficient quantization error. By construction, $|\tilde{b}_i| < \Delta$. Without randomization, the filter can be implemented as a static quantized filter with taps $Q(b_i)$. Such a quantized filter exhibits frequency response distortion though. Using the techniques developed in this thesis, we can add a binary dither process, $d_i[n]$, such that the taps randomly take the values,

$$b_i[n] = Q(b_i) + \text{sgn}(\tilde{b}_i)\Delta d_i[n] \tag{5.12}$$

where the dither process takes the binary values $d_i[n] = \{0, 1\}$ and has a mean, $E\{d_i[n]\} = \tilde{b}_i$. With this dither, the tap effectively switches between the two quantization levels, $Q(b_i)$

and $Q(b_i) + \text{sgn}(\tilde{b_i})\Delta$. The output of the dithered filter, $\hat{y}[n]$, can thus be expressed as,

$$\hat{y}[n] = \underbrace{\sum_{i=0}^{N-1} Q(b_i)x[n-i]}_{\text{static quantized filter}} + \Delta \underbrace{\sum_{i=0}^{N-1} \text{sgn}(\tilde{b_i})d_i[n]x[n-i]}_{\text{binary randomized filter}} \tag{5.13}$$

The first term is the output of the static quantized filter. The second term is the output of a BRF. Essentially, the filter can be implemented as the parallel combination of a static quantized filter, $Q(b_i)$, and a BRF with $K = \Delta$ which randomizes between the quantization levels. The resulting randomized filter has the desired dither effect; the frequency response distortion is replaced with uncorrelated, shapeable noise. The BRF techniques developed in this thesis can be used to design the binary dither to minimize this error.

In addition to uniform quantization, the BRF theory can be used in conjunction with shift-register filters, dithering between different bit-shift levels. For example, a bit-shift of 2.3 can be achieved, on average, by dithering between a shift of 2 and 3. The dithered shift-register filters are still multiplier-less, but instead of frequency response distortion the error is again uncorrelated, shapeable noise.

Chapter 6

Memoryless Standard Direct Form I FIR Binary Randomized Filtering

In this chapter standard memoryless Direct Form I FIR binary randomized filters are developed. They are a class of multiplier-less FIR filters that have additive white error rather than frequency response distortion. Section 6.1 presents the formal design problem. Section 6.2 develops tap-independent memoryless BRF (timBRF), where the filter is uncorrelated in time and across the taps. Section 6.3 develops tap-correlated memoryless BRF (tcmBRF) where tap-correlation can be used to reduce the error beyond timBRF. The theoretical analysis is validated with numerical experiments in Section 6.4.

6.1 Problem Statement

The block diagram of standard Direct Form I FIR BRF is reproduced in Figure 6-1. As noted in Chapter 5, we assume that a desired continuous-valued filter, $\mathbf{b} = \left[b_0 \ldots b_{N-1}\right]$ is given. The design goal for BRF is to optimize K and $\mathbf{h}[n]$ such that $\hat{y}[n]$ as close as possible to the output $y[n]$ from the desired continuous-valued filter. As shown in Section 5.2.2, the output $\hat{y}[n]$ can be decomposed as the sum of the desired $y[n]$ plus an uncorrelated additive error, $e[n]$. We use the error power, as measured by the mean squared error (MSE), as our error metric. With $e[n]$ defined as in Eqn.(5.4), the MSE can be expressed as:

$$
\begin{aligned}
\mathcal{E} = E\{e^2[n]\} &= E\left\{K^2 \sum_{i=0}^{N-1}\sum_{j=1}^{N-1} \tilde{h}_i[n]\tilde{h}_j[n] s_i s_j x[n-i]x[n-j]\right\} \\
&= K^2 \left\{\sum_{i=0}^{N-1}\sum_{j=1}^{N-1} \underbrace{E\left\{\tilde{h}_i\tilde{h}_j[n]\right\}}_{\sigma_{ij}} s_i s_j \underbrace{E\left\{x[n-i]x[n-j]\right\}}_{R_{xx}[i-j]}\right\} \\
&= K^2 \sum_{i=0}^{N-1}\sum_{j=1}^{N-1} \sigma_{ij} s_i s_j R_{xx}[i-j] \qquad (6.1)
\end{aligned}
$$

where, as noted, $\sigma_{ij} = \operatorname{cov}(h_i[n], h_j[n])$.

Note that in Eqn.(6.1) the time correlation of $\mathbf{h}[n]$ does not affect the MSE. Consequently, $\mathbf{h}[n]$ can be restricted to be a uncorrelated in time, i.e. memoryless, without loss of generality. Frequency-shaped standard BRF has not no additional benefit because even though the error spectrum may be frequency-shaped, there is no post-filter to remove it. Note that in memoryless BRF the error is always white, with a constant error spectrum.

From Eqn.(6.1), the design goal is to choose K and the covariances σ_{ij} such that the MSE is minimized. The MSE objective function, Eqn.(6.1), can be split into two terms, the first composed of the N diagonal terms with $i = j$, and the second composed of the remaining $N^2 - N$ terms with $i \neq j$:

$$\mathcal{E} = K^2 \sum_{i=0}^{N-1} \sigma_i^2 s_i^2 R_{xx}[0] + K^2 \sum_{i=0}^{N-1} \sum_{j=1:j\neq i}^{N-1} \sigma_{ij} s_i s_j R_{xx}[i-j] \tag{6.2}$$

In tap-independent memoryless BRF (timBRF), the covariances $\sigma_{ij}, i \neq j$ are all restricted to be zero. With this constraint, the second term in Eqn.(6.2) is zero. Tap-independent memoryless BRF is the simplest form of BRF serving as a performance baseline for the other, more complex techniques. The design and error analysis of timBRF is discussed in Section 6.2.

The MSE can be reduced by allowing correlation between the taps. In particular, timBRF makes the second term in Eqn.(6.2) zero, but by appropriately picking σ_{ij} we can make the second term in negative – thus reducing the total MSE. Intuitively, this corresponds to designing the taps randomization such that there is partial cancellation of the error at the accumulator. The design of tap-correlated memoryless BRF (tcmBRF) is non-trivial, requiring the optimization and generation of correlated binary vectors. The design and error analysis of tcmBRF is discussed in Section 6.3

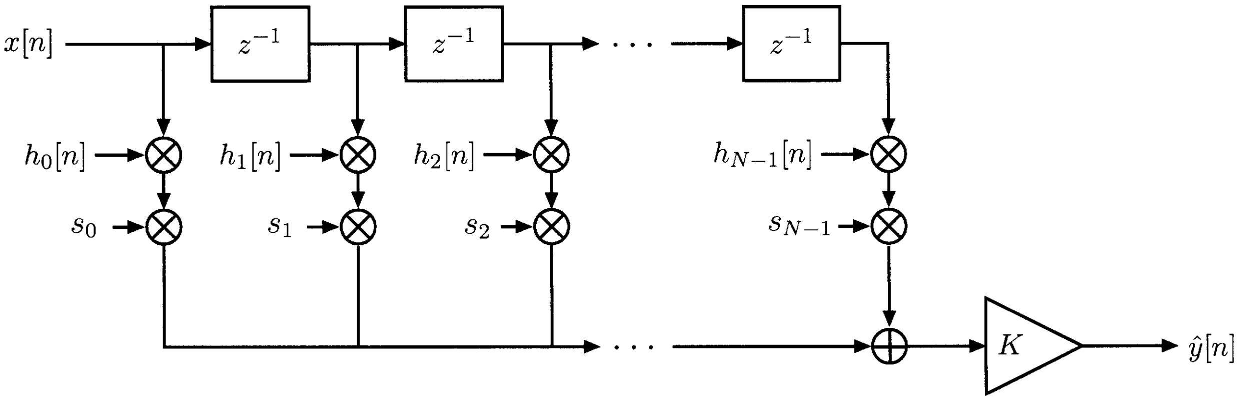


Figure 6-1. Block diagram of N-tap simple BRF.

6.2 Tap-Independent Memoryless BRF

Tap-independent memoryless BRF (timBRF) is the simplest form of BRF where binary vector process $\mathbf{h}[n]$ is independent in time and across the taps. In this section we derive the optimal timBRF design and do a detailed error analysis.

6.2.1 Problem Statment

The MSE for timBRF corresponds to the first term in Eqn.(6.2),

$$\mathcal{E} = K^2 \sum_{i=0}^{N-1} \sigma_i^2 s_i^2 R_{xx}[0] \tag{6.3}$$

Recall that in a binary process the mean fixes the variance. Substituting the mean constraint, $\mu_i = \frac{s_i b_i}{K} = \frac{|b_i|}{K}$, into the expression for the variance, $\sigma_i^2 = \mu_i(1 - \mu_i)$, and this into Eqn.(6.3), the timBRF MSE can be expressed as a function of K and b_i,

$$\begin{aligned} \mathcal{E} &= K^2 R_{xx}[0] \sum_{i=0}^{N-1} \frac{|b_i|}{K}\left(1 - \frac{|b_i|}{K}\right) \\ &= K R_{xx}[0] \sum_{i=0}^{N-1} |b_i| - R_{xx}[0] \sum_{i=0}^{N-1} |b_i|^2 \end{aligned} \tag{6.4}$$

Since the b_i are fixed by the desired continuous-valued filter, the only design parameter in timBRF is K. Combining the constraint on K from Eqn.(5.2) with Eqn.(6.4), the timBRF design problem can be formally posed as the constrained optimization:

$$\boxed{\begin{aligned} &\underset{K}{\text{minimize}} \quad K R_{xx}[0] \textstyle\sum_{i=0}^{N-1} |b_i| - R_{xx}[0] \sum_{i=0}^{N-1} |b_i|^2 \\ &\text{subject to} \quad K \geq \max\{b_i\} \end{aligned}} \tag{6.5}$$

6.2.2 Optimal Solution

Note that for design purposes, only the absolute values of the taps, $|b_i|$, matter. Negative valued taps are accounted for by the sign change after the binary tap process. Thus, in what follows, without loss of generality we can focus solely on filters with non-negative taps. The objective function in Eqn.(6.5) is linear in K and since $R_{xx}[0] > 0$ and $\sum_{i=0}^{N-1} |b_i| > 0$, the optimal solution is to choose K as the minimal feasible value:

$$K^* = \max\{|b_i|\} \tag{6.6}$$

Intuitively, each random tap is a source of noise, so the optimal solution is to make one tap static. This is what the scaling of Eqn.(6.6) does – it scales the means so that the maximal tap has $\mu_i = 1$, i.e. it is always on and not random anymore.

The optimal solution implies that the choice of K matters. Naively, it may seem that the timBRF implementation should be independent of K, but this is not the case. For example, say the desired continuous-valued filter has three taps:

$$\mathbf{b} = \begin{bmatrix}\frac{1}{2} & \frac{1}{2} & \frac{1}{2}\end{bmatrix} \tag{6.7}$$

There are an unlimited number of $(\boldsymbol{\mu}, K) = (\frac{\mathbf{b}}{K}, K)$ pairs that can be used to implement this timBRF. Two possible pairs are:

$$(\boldsymbol{\mu}_1, K_1) = (\begin{bmatrix}\frac{1}{2} & \frac{1}{2} & \frac{1}{2}\end{bmatrix}, 1) \tag{6.8}$$

$$(\boldsymbol{\mu}_2, K_2) = (\begin{bmatrix}1 & 1 & 1\end{bmatrix}, \frac{1}{2}) \tag{6.9}$$

For the first pair $(\boldsymbol{\mu}_1, K_1)$, the MSE is:

$$\begin{aligned}\mathcal{E}(\boldsymbol{\mu}_1, K_1) &= K^2 \sum_{i=0}^{2} R_{xx}[0]\sigma_i^2 = R_{xx}[0] \sum_{i=0}^{2} \mu_i(1-\mu_i) \\ &= R_{xx}[0] \sum_{i=0}^{2} \frac{1}{4} = \frac{3}{4} R_{xx}[0]\end{aligned} \tag{6.10}$$

In contrast, the second pair, $(\boldsymbol{\mu}_2, K_2)$, can be implemented perfectly as a static binary filter. The three timBRF taps are always on, i.e. $h_i[n] = 1$. There is no randomization so $\sigma_i^2 = 0$. Consequently, the MSE is zero:

$$\mathcal{E}(\boldsymbol{\mu}_2, K_2) = K^2 \sum_{i=0}^{2} R_{xx}[0]\sigma_i^2 = 0 \tag{6.11}$$

Clearly the second pair is a better timBRF implementation and the choice of $(\boldsymbol{\mu}, K)$ matters. Geometrically, the set of possible means is the unit-cube in $\mathbb{R}^N$, i.e. $\boldsymbol{\mu} \in [0,1]^N$. The desired continuous-valued filter, $\mathbf{b}$, is a point along the ray $C\mathbf{e}_{\mathrm{b}}$, where $C \geq 0$ and $\mathbf{e}_{\mathrm{b}}$ is the unit vector in the direction $\mathbf{b}$. The set of possible implementation points for $\mathbf{b}$ is defined by the intersection of the ray with the unit-cube. Geometrically, this is a line-segment from the origin to the face of the unit-cube in the direction $\mathbf{e}_{\mathrm{b}}$. Figure 6-2 illustrates an example of this set for a three tap filter in $\mathbb{R}^3$. Each point on this line-segment corresponds to an implementation point $\boldsymbol{\mu}$, and a scaling K. The value of K increases towards the origin. The extremal pair on the face is $(\boldsymbol{\mu} = \frac{\mathbf{b}}{\max\{|b_i|\}}, K = \|\mathbf{b}\|_\infty)$. The origin is $(\boldsymbol{\mu} = \mathbf{0}, K = \infty)$.

The timBRF solution implies that the optimal implementation point along this line-segment, denoted $\boldsymbol{\mu}^*(\mathbf{e}_{\mathrm{b}})$, is always the extremal point on the face of the cube. Geometrically, the set of optimal implementation points in $\mathbb{R}^N$ is the $N-1$-dimensional surface of the unit-cube. We denote this surface as the implementation manifold for timBRF. For timBRF, this manifold is independent of the input auto-correlation $R_{xx}[m]$.

6.2.3 Scale Invariance

The optimal timBRF solution implies that the optimal implementation point is scale invariant. Specifically, any filter in the direction $\mathbf{e}_{\mathrm{b}}$ such that $\mathbf{b} = C\mathbf{e}_{\mathrm{b}}$, with $C > 0$, has an optimal timBRF implementation point, $\boldsymbol{\mu}^*(\mathbf{e}_{\mathrm{b}})$, on the face of the cube $[0,1]^N$ independent of C. In other words, all timBRFs with $\mathbf{b}$ in the direction $\mathbf{e}_{\mathrm{b}}$, regardless of their scale, are optimally implemented at the point $\boldsymbol{\mu}^*(\mathbf{e}_{\mathrm{b}})$.

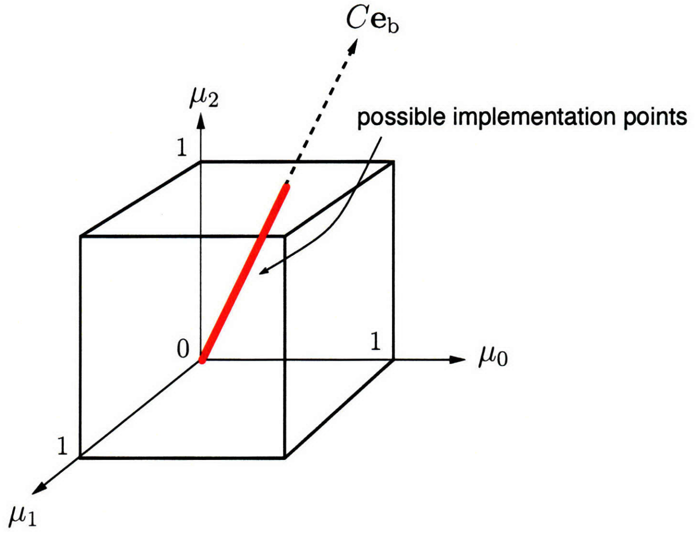


Figure 6-2. The red line denotes the set of possible implementation points for a continuous-valued filter **b** along the ray $C\mathbf{e}_\text{b}$. The optimal implementation point for timBRF is on the face of the cube.

The scale invariance of the optimal implementation point implies scale invariance of the SNR too. Since all filters in the direction $\mathbf{e}_\text{b}$ can be expressed as $\mathbf{b} = C\boldsymbol{\mu}(\mathbf{e}_\text{b})$, the minimal MSE for a filter **b** can be expressed as:

$$\mathcal{E}^*(\mathbf{b}) = R_{xx}[0]\sum_{i=0}^{N-1}|b_i|(\max\{|b_i|\} - |b_i|) \tag{6.12}$$

$$= \underbrace{C^2\, R_{xx}[0]\sum_{i=0}^{N-1}\mu_i(\mathbf{e}_\text{b})(1-\mu_i(\mathbf{e}_\text{b}))}_{\mathcal{E}(\mathbf{e}_\text{b})}$$

$$= C^2\mathcal{E}^*(\mathbf{e}_\text{b}) \tag{6.13}$$

where in the second line we have used the substitutions $|b_i| = C\mu_i(\mathbf{e}_\text{b})$ and $\max\{|b_i|\} = C$ to simplify our expression. Since $\mathbf{b} = C\boldsymbol{\mu}(\mathbf{e}_\text{b})$, the signal power through a timBRF can be expressed as:

$$\mathcal{S}(\mathbf{b}) = E\{y^2[n]\} = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{xx}(e^{j\omega})|B(e^{j\omega})|^2 d\omega$$

$$= \frac{C^2}{2\pi}\int_{-\pi}^{\pi} S_{xx}(e^{j\omega})|M_{\mathbf{e}_\text{b}}(e^{j\omega})|^2 d\omega$$

$$= C^2\mathcal{S}(\mathbf{e}_\text{b}) \tag{6.14}$$

where $B(e^{j\omega})$ is the frequency-response of the continuous-valued LTI filter **b** and $M_{\mathbf{e}_\text{b}}(e^{j\omega})$ is the frequency-response of the filter $\boldsymbol{\mu}(\mathbf{e}_\text{b})$. The SNR is given by dividing the signal power, Eqn.(6.14), by the MSE, Eqn.(6.61). The leading C^2 terms cancel in the division, making the SNR invariant of the scale, C, and only dependent on the direction $\mathbf{e}_\text{b}$ and the input spectrum $S_{xx}(e^{j\omega})$,

$$\text{SNR}(C\mathbf{e}_\text{b}) = \frac{\mathcal{S}(\mathbf{e}_\text{b})}{\mathcal{E}^*(\mathbf{e}_\text{b})} \tag{6.15}$$

This scale invariance is desirable because the performance of the filter does not depend upon the total filter power. On the other hand, the performance is dependent on the desired filtering application. This is discussed further in the next section.

6.2.4 Error Analysis

In this section we do a detailed error analysis of timBRF. The first section compares the timBRF MSE to that of static binary filtering. The second section discusses the performance of timBRF as a function of the desired filter. The final section discusses the error scaling of timBRF as a function of N.

BRF vs. Static

The timBRF has a larger MSE than the optimal static binary implementation. This follows from a simple argument. Given N taps, there are 2^N possible static binary filters, $\hat{B}_j(e^{j\omega})$, where $j = 1, \cdots, 2^N$. Each filter has an associated optimal scaling, K_j, and an MSE which can be expressed as,

$$\mathcal{E}_j = \int_{-\pi}^{\pi} S_{xx}(e^{j\omega})|B(e^{j\omega}) - K_j\hat{B}_j(e^{j\omega})|^2 \frac{d\omega}{2\pi} \tag{6.16}$$

Of the finite possibilities, at least one filter has the minimum MSE because a finite set of real numbers always has a minimum. Randomization does not improve performance because the other static binary filters have a higher MSE. Switching randomly to these non-minimum configurations as a function of time cannot improve the MSE. This implies that the minimum MSE filter is a static binary filter.

The benefit of randomization comes from the fact that the form of the error is better. It is uncorrelated shapeable noise rather than frequency distortion. As noted in Section 5.2.2, in certain applications, especially perceptual ones, this form of error might be much less disturbing than the correlated, colored error of frequency distortion.

Fixed Length Analysis

The timBRF MSE is dependent on the desired continuous-valued filter we are trying to implement. Certain filters, like one where the coefficients are all identica, e.g. $\mathbf{b} = \begin{bmatrix} b & b & b \end{bmatrix}$, can be perfectly implemented using a timBRF and have zero MSE. Other filters have a much higher MSE. Mathematically, the minimal MSE for a timBRF is a function $\mathcal{E}(\mathbf{b})$ given by,

$$\mathcal{E}^*(\mathbf{b}) = R_{xx}[0] \sum_{i=0}^{N-1} |b_i|(\max\{|b_i|\} - |b_i|) \tag{6.17}$$

As an example of how the MSE varies as a function of the taps, Figure 6-3(a) illustrates the $\mathcal{E}(\mathbf{b})$ for all positive-valued, three-tap timBRFs with unit-energy, i.e. $b_0^2 + b_1^2 + b_2^2 = 1$ and $b_i \geq 0$. The figure is plotted as a projection into the (b_0, b_1) plane. The b_2 coordinate is implicit from the unit-energy constraint. Figure 6-3(b) illustrates the implementation manifold for 3-tap white-UMLRF filters. It is the surface of the unit-cube in three dimensions.

There are certain symmetries to note in the plot. First, as noted earlier, $\mathcal{E}(\mathbf{b})$ for filters with negative taps is equivalent filters with positive taps under the mapping, $b_i \to |b_i|$. Accordingly, $\mathcal{E}(\mathbf{b})$ in the non-positive orthants are symmetric to this portion plotted in the positive orthant. In addition, $\mathcal{E}(\mathbf{b})$ is symmetric with respect to which element of $\mathbf{b}$ is maximal. This can be seen in Figure 6-3(a) as a three-fold symmetry around the point $\begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix}$.

The analysis above is only for the MSE. The SNR is a better measure of actual performance. The SNR is affected by the MSE, but is also dependent on the output signal power, $\sigma_y^2 = E\{y^2[n]\}$. If the output signal power is small then the SNR will be small despite what the MSE is. In these situations, BRF will perform poorly. For example, if the goal of filtering was to extract a low-power signal in surrounding high-power noise, then BRF will likely perform poorly. Intuitively, the high-power noise will spread across all of the band with timBRF, burying the desired low-power signal in noise. Mathematically, the SNR will be low because the numerator, $\mathcal{S}(\mathbf{e}_b)$, is small. On the other hand, in other applications, like audio equalization the output signal power is high. In this case, BRF could be very useful is removing frequency distortion and replacing it with noise that is masked by the higher power output signal.

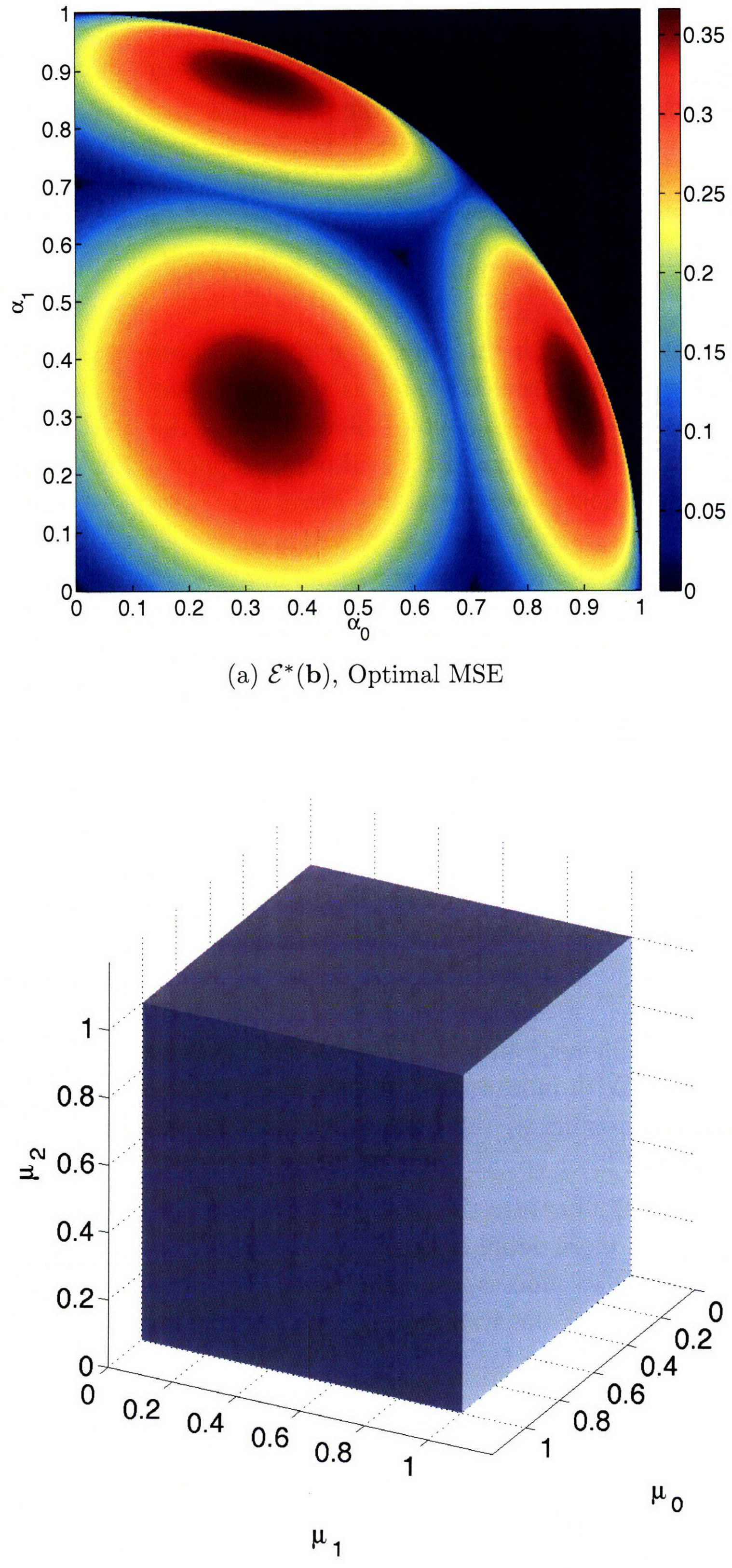
Error Scaling

The timBRF MSE increases as the number of taps N increases. Intuitively this is because with more taps there are more noise-sources injecting error into the filter. The growth of the error depends on the ideal filter specification. However, with more taps we can implement a better filter on average, i.e. we are closer to the ideal filter specification. For example, assume our goal is to implement an ideal low-pass filter with a certain specification using a Parks-McClellan filter. With more taps, the max ripple error is lower and the continuous-valued filter better approximates the ideal response. There is thus a tradeoff between randomization error and filter approximation error.

Figures 6-14 and 6-17 illustrate this tradeoff for a specific Parks-McClellan design example. It is described in more detail in Section 6.4. For now, note that there with a longer length, N, the filter has lower max ripple error, but higher randomization MSE. There is no ideal operating length, rather the system designer must choose a suitable operating length by trading off between these two forms of error.

Regardless of the exact specification though, any well-posed specification has a constant power constraint on the filter, i.e. the ideal filter has a total power $\sum |b_i|^2 = C$. For example, with our Parks-McClellan low-pass filter design, the ideal filter has a constant power given by the square of the integral under the passband. This is constant regardless of the number of taps used to implement a filter. The worst-case MSE given a constant power constraint on the filter thus gives an upper-bound on how the randomization error for any filter specification scales as N increases.

Without loss of generality we can fix the the filter to have unit power. The worst-case scaling of unit power timBRFs can be found analytically. For a fixed number of taps, N,



(a) $\mathcal{E}^*(\mathbf{b})$, Optimal MSE

(b) Implementation Manifold

Figure 6-3. Plot of optimal MSE and implementation manifold for 3-tap timBRF FIR filters on the unit-power manifold defined by $\sum_{i=0}^{2} b_i^2 = 1$.

the worst-case filter is the solution to the maximization:

$$\begin{aligned} \underset{\mathbf{b}}{\text{maximize}} \quad & R_{xx}[0] \textstyle\sum_{i=0}^{N-1} b_i(\|\mathbf{b}\|_\infty - b_i) \\ \text{subject to} \quad & \textstyle\sum_{i=0}^{N-1} b_i^2 = 1 \\ & b_i \geq 0 \end{aligned} \tag{6.18}$$

The optimal MSE, $\mathcal{E}^*(\mathbf{b})$, is symmetric over which element of $\mathbf{b}$ is maximal. Thus, without loss of generality, we can assume that $\mathbf{b} = b_0$. The maximization of Eqn.(6.18) can then be re-formulated as a minimization:

$$\begin{aligned} \underset{b_0, b_i}{\text{minimize}} \quad & -R_{xx}[0] \textstyle\sum_{i=1}^{N-1} b_i(b_0 - b_i) \\ \text{subject to} \quad & \textstyle\sum_{i=1}^{N-1} b_i^2 = 1 - b_0^2 \\ & b_i \geq 0, \text{ for } i \neq 0 \\ & b_0 \geq b_i, \text{ for } i \neq 0 \end{aligned} \tag{6.19}$$

The Lagrangian for the constrained optimization of Eqn.(6.19) is expressed in Eqn.(6.20). λ is the Lagrange multiplier for the unit-energy equality constraint, γ_i are the Lagrange multipliers for the non-zero inequality constraints, and θ_i are the Lagrange multipliers for the $b_0 \geq b_i$ inequality constraints.

$$\mathcal{L}(\mathbf{b}, \lambda, \gamma_i, \theta_i) = -R_{xx}[0] \sum_{i=1}^{N-1} b_i(b_0 - b_i) + \lambda \left(\sum_{i=1}^{N-1} b_i^2 + b_0^2 - 1 \right) + \sum_{i=1}^{N-1} \gamma_i b_i + \sum_{i=1}^{N-1} \theta_i(b_0 - b_i) \tag{6.20}$$

Empirically, we have observed that none of the inequality constraints are active at the worst-case filter. For example, in Figure 6-3(a), the worst case point occurs at an interior point, not at the boundary. Specifically, at the worst-case filter, all the b_i are strictly greater than zero and a single tap is maximal, i.e. $\|\mathbf{b}\|_\infty = b_0 \neq b_i$. Intuitively, this observation is sensible since both a $b_i = 0$ tap and a $b_i = \|\mathbf{b}\|_\infty$ tap have zero contribution to the total MSE. We would expect that, other than $b_0 = \|\mathbf{b}\|_\infty$, the other taps do not take these values in the worst-case filter.

From these observations, we assume that none of the inequality constraints are active at the worst-case filter. Complementary slackness implies that their corresponding Lagrange multipliers are zero [2]. Consequently, the last two terms in Eqn.(6.20) are zero. The reduced Lagrangian is:

$$\mathcal{L}(\mathbf{b}, \lambda) = -R_{xx}[0] \sum_{i=1}^{N-1} b_i b_0 + R_{xx}[0] \sum_{i=1}^{N-1} b_i^2 + \lambda \left(\sum_{i=1}^{N-1} b_i^2 + b_0^2 - 1 \right) \tag{6.21}$$

Applying the Karush-Kuhn-Tucker (KKT) conditions to this reduced Lagrangian results in $N+1$ equations,

$$\frac{\partial \mathcal{L}}{\partial b_i} = -Rxx[0]b_0 + 2R_{xx}[0]b_i + 2\lambda b_i = 0, \text{ for } i \neq 0 \tag{6.22}$$

$$\frac{\partial \mathcal{L}}{\partial b_0} = -R_{xx}[0] \sum_{i=1}^{N-1} b_i + 2\lambda b_0 = 0 \tag{6.23}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \sum_{i=1}^{N-1} b_i^2 + b_0^2 - 1 = 0 \tag{6.24}$$

The first $N-1$ equations, Eqn.(6.22) can be solved for b_i in terms of b_0,

$$b_i = \frac{R_{xx}[0]b_0}{2R_{xx}[0] + 2\lambda} \tag{6.25}$$

Substituting Eqn.(6.25) into the constraint, Eqn.(6.24), and manipulating the resulting expression, we can solve for the denominator $2R_{xx}[0] + 2\lambda$:

$$2R_{xx}[0] + 2\lambda = R_{xx}[0]b_0\sqrt{\frac{N-1}{1-b_0^2}} \tag{6.26}$$

Substituting this back into Eqn.(6.25), b_i can be expressed as a function of b_0,

$$b_i = \sqrt{\frac{1-b_0^2}{N-1}} \tag{6.27}$$

From the second KKT condition, Eqn.(6.23), we get an expression for b_0 in terms of the other b_i,

$$b_0 = \frac{R_{xx}[0]\sum_{i=1}^{N-1} b_i}{2\lambda} \tag{6.28}$$

Substituting Eqn.(6.27) for b_i and Eqn.(6.26) for 2λ, we can eliminate these variables and after considerable simplification get a quartic equation that b_0 must satisfy,

$$4Nb_0^4 - 4Nb_0^2 + (N-1) = 0 \tag{6.29}$$

This quartic equation is a quadratic equation in b_0^2. There are two solutions for b_0^2 using the quadratic formula. After some simplifications, the solutions are,

$$b_0^2 = \frac{1}{2}\sqrt{1 \pm \frac{1}{\sqrt{N}}} \tag{6.30}$$

Consequently, there are four solutions for b_0. The two negative roots are not feasible, since we have restricted all our taps to be positive. Of the two positive roots, from simulation we see that the maximal root is the worst-case b_0. Thus, as a function of N, the maximal tap in the worst-case filter is,

$$b_0 = \frac{1}{\sqrt{2}}\sqrt{1 + \frac{1}{\sqrt{N}}} \tag{6.31}$$

Substituting Eqn.(6.31) into Eqn.(6.27), we can get a closed form expression for other taps. They all have the same value:

$$b_i = \sqrt{\frac{\frac{1}{2} - \frac{1}{\sqrt{N}}}{N-1}}, \text{ for } i \neq 0 \tag{6.32}$$

The worst-case MSE is computed by substituting the derived values of b_i and b_0 back into the objective function of (6.19). After some tedious simplifications, the worst-case scaling of the MSE can be expressed as a function of N as:

$$\mathcal{E}_{\text{wc}} = \frac{R_{xx}[0]}{2} \left(\sqrt{N - 2 + \frac{1}{N}} - 1 + \frac{1}{\sqrt{N}} \right) \tag{6.33}$$

As N increases the $\frac{R_{xx}[0]}{2}\sqrt{N}$ term dominates. So for white-UMLRF the randomization error scales as the $\sqrt{N}$:

$$\mathcal{E}_{\text{wc}} \approx \mathrm{O}(\sqrt{N}) \tag{6.34}$$

Figure 6-8 plots the analytic $\mathcal{E}_{\text{wc}}$, Eqn.(6.33), as a function of N. Numerical results match exactly with this analytic expression. Equation (6.33) is an upper-bound on the MSE scaling for all memoryless BRF with unit-energy. No matter what the exact filter specification, the MSE does not grow faster than Eqn.(6.33).

6.3 Tap-Correlated Memoryless BRF

This section discusses tap-correlated memoryless BRF (tcmBRF) where the binary vector tap process, $\mathbf{h}[n]$, can be correlated across the taps but is still independent in time. The randomization error is still white, but the noise floor can be reduced beyond timBRF by using tap correlation. Section 6.3.1 presents the formal design problem. Section 6.3.2 develops an algorithm to find the optimal binary solution. Section 6.3.3 shows the scale invariance of the optimal solution. The design of the optimal solution is shown to be computationally intractable. In Section 6.3.4, we develop a parametric design technique that is more tractable. Section 6.3.6 does an error analysis on tcmBRF using a relaxation bound.

6.3.1 Problem Statement

In tcmBRF, the randomization error is still a white process because the taps are uncorrelated in time. As noted in Section 6.1 though, the noise floor can be reduced by correlating across the taps. Specifically, in the MSE, which is reproduced below as Eqn.(6.35), if $R_{xx}[k-j] > 0$ then σ_{ij} can be made negative to reduce the error beyond the timBRF MSE, which is the first term. Similarly, if $R_{xx}[k-j] < 0$, then σ_{ij} can be made positive to reduce the error.

$$\mathcal{E} = K^2 \sum_{i=0}^{N-1} \sigma_i^2 s_i^2 R_{xx}[0] + K^2 \sum_{i=0}^{N-1} \sum_{j=1:j\neq i}^{N-1} \sigma_{ij} s_i s_j R_{xx}[i-j] \tag{6.35}$$

Intuitively, the tap correlation can be designed to get partial cancellation of the error at the accumulator. The error due to binary randomization at one tap can be partially cancelled by the error at another tap. An important pre-requisite to the design process is prior knowledge of $R_{xx}[m]$, the input auto-correlation. In this sense, tcmBRF is a form of data-dependent filtering where the filter is tuned to the statistics of the input signal. In what follows, we assume perfect knowledge of the input auto-correlation. In practice though, this auto-correlation must be adaptively estimated from the input signal.

The goal in tcmBRF is to design K and σ_{ij} to minimize the MSE. The design can be posed formally as a constrained optimization. The MSE objective, Eqn.(6.35), can be expressed more compactly in matrix notation using the trace operator as:

$$K^2 \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sigma_{ij} R_{xx}[i-j] = K^2 \text{tr}\left(\mathbf{R_x \Sigma_h}\right) \tag{6.36}$$

In Eqn.(6.36), $\mathbf{\Sigma_h}$ is the zero-lag auto-covariance matrix of the random vector $\mathbf{h}[n]$. $\mathbf{R_x}$ is the auto-covariance matrix of the random vector, $\mathbf{x} = \begin{bmatrix} x[n] & \cdots & x[n-N] \end{bmatrix}$. It is a Toeplitz matrix of the form:

$$\mathbf{R_x} = \begin{bmatrix} R_{xx}[0] & R_{xx}[1] & \dots & R_{xx}[N] \\ R_{xx}[1] & R_{xx}[0] & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ R_{xx}[N] & \dots & R_{xx}[1] & R_{xx}[0] \end{bmatrix} \tag{6.37}$$

There are three constraints on K and $\mathbf{\Sigma_h}$. As in timBRF, the scaling, K, must be chosen so that all the of the means are in the range $[0, 1]$:

$$K \geq \max\{|b_i|\} \tag{6.38}$$

Secondly, the mean vector is constrained to be a scaled value of $\mathbf{b}$, $E\{\mathbf{h}[n]\} = \boldsymbol{\mu} = \mathbf{b}/K$. Consequently, because the mean fixes the variance of a binary process, the diagonal elements of $\mathbf{\Sigma_h}$ are constrained to be:

$$\sigma_i^2 = \mu_i(1-\mu_i) = \frac{|b_i|}{K}\left(1 - \frac{|b_i|}{K}\right) \tag{6.39}$$

Note how this constraint couples $\mathbf{\Sigma_h}$ to the scaling K.

Lastly, the pair $(\mathbf{\Sigma_h}, \boldsymbol{\mu})$ must be achievable using a binary random vector, i.e there must exist a multivariate binary distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{\Sigma_h}$, [9]. We denote the set of achievable binary covariance matrices with mean $\boldsymbol{\mu}$ as $\mathcal{B}(\boldsymbol{\mu})$. This constraint is very similar to the binary achievability constraint discussed in Chapter 3 for frequency-shaped SRS. As it was there, this constraint is problematic because not all valid covariance matrices, i.e. positive-semi definite matrices $\mathbf{\Sigma_h} \succeq 0$, are achievable using a binary random vector [9, 27]. The set of achievable matrices has been studied to some extent [9], but this set is not tractable for optimization. Combining the objective, Eqn.(6.36), with the three constraints, the tcmBRF design problem can be formally posed as a constrained optimization:

$$\begin{array}{ll} \underset{K,\boldsymbol{\Sigma}_{\mathbf{h}}}{\text{minimize}} & K^2\text{tr}\left(\mathbf{R}_{\mathbf{x}}\boldsymbol{\Sigma}_{\mathbf{h}}\right) \\ & \\ \text{subject to} & K \geq \max\{b_i\} \\ & \mu_i = b_i/K \\ & \sigma_i^2 = \mu_i(1-\mu_i) \\ & \boldsymbol{\Sigma}_{\mathbf{h}} \in \mathcal{B}(\boldsymbol{\mu}) \end{array} \tag{6.40}$$

Tap-correlated BRF design is significantly more difficult than the timBRF design. As with frequency-shaped SRS, we cannot hope for an analytic solution, but even a numerical solution is difficult. Section 6.3.2 develops the optimal solution by posing the design optimization over the joint-density of $\mathbf{h}[n]$. Though optimal, this algorithm is shown to be computationally intractable. As a tractable alternative, Section 6.3.4 develops a parametric model that can be optimized for tcmBRF implementation.

6.3.2 Optimal Solution

The two variables of optimization in Eqn.(6.40), K and the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{h}}$, are coupled through the mean-constraint $\boldsymbol{\mu} = \mathbf{b}/K$. Joint optimization over both is difficult, so we separate the optimization into two stages, an inner optimization over $\boldsymbol{\Sigma}_{\mathbf{h}}$ for a fixed K and an outer optimization over K.

We first develop the inner optimization for a fixed K. Fixing K, fixes the mean vector $\boldsymbol{\mu} = \mathbf{b}/K$. The inner optimization can be formally posed as,

$$\begin{array}{ll} \underset{\boldsymbol{\Sigma}_{\mathbf{h}}}{\text{minimize}} & \mathcal{I}(\boldsymbol{\Sigma}_{\mathbf{h}}, \boldsymbol{\mu}) = \text{tr}\left(\mathbf{R}_{\mathbf{x}}\boldsymbol{\Sigma}_{\mathbf{h}}\right) \\ & \\ \text{subject to} & \sigma_i^2 = \mu_i(1-\mu_i) \\ & \boldsymbol{\Sigma}_{\mathbf{h}} \in \mathcal{B}(\boldsymbol{\mu}) \end{array} \tag{6.41}$$

As discussed in the previous section, the compatibility constraint is problematic. We can incorporate it in a more tractable way by recasting the optimization of Eqn.(6.41) over the joint density of $\mathbf{h}[n]$.

The multi-variate binary joint-density, $p_{\mathbf{h}}(\mathbf{h}[n])$, is a discrete PMF with probability mass at the 2^N vertices of a N-dimensional hypercube. The probability masses $p_{\mathbf{h}}(\mathbf{h}[n])$ can be ordered using a binary counting scheme and collected in a mass-vector, $\mathbf{p}$. For example, with $N = 3$, $\mathbf{p}$ is of size $2^3 = 8$. In our formulation, the masses are ordered using the binary counting scheme below, where $h_0[n]$ represents the least significant bit and $h_2[n]$ represents the most significant bit.

$$
\begin{aligned}
p_0 &= p(h_2[n] = 0, h_1[n] = 0, h_0[n] = 0)\\
p_1 &= p(h_2[n] = 0, h_1[n] = 0, h_0[n] = 1)\\
p_2 &= p(h_2[n] = 0, h_1[n] = 1, h_0[n] = 0)\\
p_3 &= p(h_2[n] = 0, h_1[n] = 1, h_0[n] = 1)\\
p_4 &= p(h_2[n] = 1, h_1[n] = 0, h_0[n] = 0)\\
p_5 &= p(h_2[n] = 1, h_1[n] = 0, h_0[n] = 1)\\
p_6 &= p(h_2[n] = 1, h_1[n] = 1, h_0[n] = 0)\\
p_7 &= p(h_2[n] = 1, h_1[n] = 1, h_0[n] = 1)
\end{aligned}
$$

Figure 6-4 illustrates the labeled joint-density $p_{\mathbf{h}}(\mathbf{h}[n])$ in three dimensions. The extension to N dimensions is done in a straightforward manner, with $h_0[n]$ as the least significant bit and $h_{N-1}[n]$ as the most significant bit.

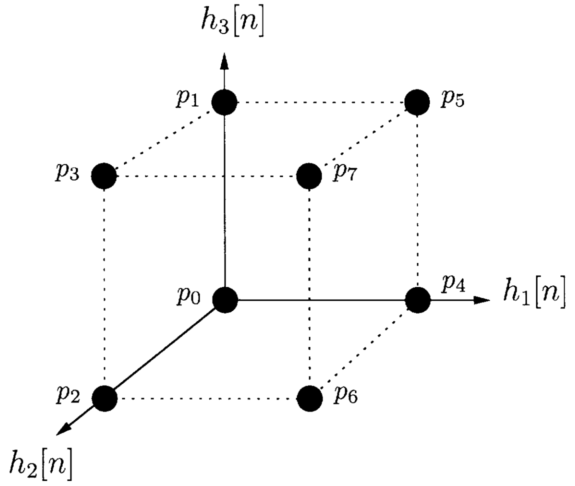


Figure 6-4. Joint density $p_{\mathbf{h}}(\mathbf{h}[n])$ in three dimensions

In posing the optimization, it is necessary to define a binary counting matrix, $\mathbf{A}$, of size $2^N \times N$. The i-th row of $\mathbf{A}$ is the N-element binary representation of the number $i \in \{0, 1, \ldots, 2^N - 1\}$. For example, with $N = 3$, $\mathbf{A}$ is an 8×3 matrix of the form:

$$
\mathbf{A} = \begin{bmatrix} 0 & 0 & 0\\ 0 & 0 & 1\\ 0 & 1 & 0\\ 0 & 1 & 1\\ 1 & 0 & 0\\ 1 & 0 & 1\\ 1 & 1 & 0\\ 1 & 1 & 1 \end{bmatrix} \tag{6.42}
$$

The decision variable vector, $\mathbf{p}$, is subject to three constraints. First, each element, representing a probability mass, must be non-negative,

$$p_i \geq 0 \tag{6.43}$$

Second, the total probability mass must sum to 1,

$$\sum_{i=0}^{2^N-1} p_i = \mathbf{1}'\mathbf{p} = 1 \tag{6.44}$$

where $\mathbf{1}$ is a 2^N-dimensional vector of all ones. Lastly, the joint-density represented by $\mathbf{p}$ must be consistent with the given mean-vector $\boldsymbol{\mu}$, which is fixed by the choice of K. With the covariance matrix, this was the problematic constraint. On the binary joint-density though, this imposes N constraints on each of the marginal densities, $p(h_i[n] = 1) = \mu_i$. These can be expressed as N linear equality constraints on the joint-density,

$$\underbrace{\sum_{h_1=0}^{1} \cdots \sum_{h_{N-1}=0}^{1}}_{N-1 \text{ sums, no } i \text{ term}} p(h_1[n], \ldots, h_i[n] = 1, \ldots, h_{N-1}[n]) = p(h_i[n] = 1) = \mu_i \tag{6.45}$$

These N constraints on $\mathbf{p}$ can be expressed in matrix notation as,

$$\mathbf{A}'\mathbf{p} = \boldsymbol{\mu} \tag{6.46}$$

The ordering of $\mathbf{A}$ and $\mathbf{p}$ coincide such that the inner-product of the i-th row of $\mathbf{A}'$ with $\mathbf{p}$ corresponds to computing the marginal probability $p(h_i[n] = 1)$.

The inner objective, $\mathcal{I}(\boldsymbol{\Sigma}_{\mathbf{h}}, \boldsymbol{\mu})$, of Eqn.(6.41) can be expressed as a linear function of $\mathbf{p}$. Each cross-covariance, σ_{ij}, can be expressed as,

$$\sigma_{ij} = E\{\tilde{h}_i[n]\tilde{h}_j[n]\} = E\{h_i[n]h_j[n]\} - \mu_i\mu_j \tag{6.47}$$

Since $h_i[n] = \{0, 1\}$, the correlation $E\{h_i[n]h_j[n]\}$ is the probability that $h_i[n] = 1$ and $h_j[n] = 1$ together, i.e.

$$E\{h_i[n]h_j[n]\} = p(h_i[n] = 1, h_j[n] = 1) \tag{6.48}$$

This probability can be expressed as a linear function of the joint-density,

$$\begin{aligned} & p(h_i[n] = 1, h_j[n] = 1) \\ &= \underbrace{\sum_{h_1=0}^{1} \cdots \sum_{h_{N-1}=0}^{1}}_{N-2 \text{ sums, no } i,\ j \text{ terms}} p(h_1[n], \ldots, h_i[n] = 1, \ldots, h_j[n] = 1, \ldots, h_{N-1}[n]) \\ &= \mathbf{c}'_{ij}\mathbf{p} \end{aligned} \tag{6.49}$$

where $\mathbf{c}_{ij}$ is a vector representing the locations in $\mathbf{p}$ where both $h_i[n] = 1$ and $h_j[n] = 1$. These locations are computed using the matrix $\mathbf{A}$. Specifically, $\mathbf{c}_{ij}$ is the element-wise multiplication of the i-th column of $\mathbf{A}$, $\mathbf{a}_i$, with the j-th column of $\mathbf{A}$, $\mathbf{a}_j$. In MATLAB syntax this can be expressed as,

$$\mathbf{c}_{ij} = \mathbf{a}_i . * \mathbf{a}_j \tag{6.50}$$

Using the indicator vectors, $\mathbf{c}_{ij}$, the inner objective $\mathcal{I}(\mathbf{\Sigma_h}, \boldsymbol{\mu})$ can be expressed as a function of $\mathbf{p}$ and $\boldsymbol{\mu}$ which we denote $\mathcal{I}(\mathbf{p}, \boldsymbol{\mu})$,

$$\begin{aligned}
\mathcal{I}(\mathbf{p}, \boldsymbol{\mu}) &= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} R_{xx}[i-j](\mathbf{c}'_{ij}\mathbf{p} - \mu_i \mu_j) \\
&= \sum_{i=0}^{N-1} \underbrace{\left[R_{xx}[i-0] \quad \dots \quad R_{xx}[i-N-1]\right]}_{\mathbf{r}'_i} \begin{bmatrix} \mathbf{c}'_{i0}\mathbf{p} - \mu_i\mu_0 \\ \vdots \\ \mathbf{c}'_{iN}\mathbf{p} - \mu_i\mu_N \end{bmatrix} \\
&= \sum_{i=0}^{N-1} \mathbf{r}'_i \underbrace{\begin{bmatrix} \mathbf{c}'_{i0} \\ \vdots \\ \mathbf{c}'_{i(N-1)} \end{bmatrix}}_{\mathbf{C}'_i} \mathbf{p} - \sum_{i=0}^{N-1} \mu_i \mathbf{r}'_i \underbrace{\begin{bmatrix} \mu_0 \\ \vdots \\ \mu_{N-1} \end{bmatrix}}_{\boldsymbol{\mu}} \\
&= \underbrace{\left(\sum_{i=0}^{N-1} \mathbf{r}'_i \mathbf{C}'_i \right)}_{\mathbf{f}'} \mathbf{p} - \boldsymbol{\mu}' \mathbf{R_x} \boldsymbol{\mu} \\
&= \mathbf{f}'\mathbf{p} - \boldsymbol{\mu}' \mathbf{R_x} \boldsymbol{\mu}
\end{aligned} \tag{6.51}$$

where $\mathbf{r}'_i$ is the i-th row of the Toeplitz, covariance matrix $\mathbf{R_x}$, i.e. $\mathbf{r}'_i = \mathbf{e}_i \mathbf{R_x}$. $\mathbf{C}_i$ is a matrix composed of the stacked columns, $\mathbf{c}_{i0}$ through $\mathbf{c}_{i(N-1)}$. The linear cost vector $\mathbf{f}$ can be expressed as,

$$\mathbf{f}' = \sum_{i=0}^{N-1} \mathbf{e}'_i \mathbf{R_x} \mathbf{C}_i \tag{6.52}$$

The quadratic form, $\boldsymbol{\mu}' \mathbf{R_x} \boldsymbol{\mu}$, is a constant independent of the decision variables $\mathbf{p}$. The resulting optimization, incorporating the constraints of Eqns.(6.43), (6.44), and (6.46), and the linear cost vector $\mathbf{f}$ defined in Eqn.(6.52) is an affine program. It can be posed formally as,

$$\boxed{\begin{aligned}
\underset{\mathbf{p}}{\text{minimize}} \quad & \mathcal{I}(\mathbf{p}, \boldsymbol{\mu}) = \mathbf{f}'\mathbf{p} - \boldsymbol{\mu}' \mathbf{R_x} \boldsymbol{\mu} \\
\text{subject to} \quad & \mathbf{1}'\mathbf{p} = 1 \\
& \mathbf{A}'\mathbf{p} = \boldsymbol{\mu} \\
& p_i \geq 0
\end{aligned}} \tag{6.53}$$

This affine program can be solved efficiently using standard interior-point methods for linear programming, [3]. We denote its optimal value for a fixed $\boldsymbol{\mu}$ as $\mathcal{I}^*(\boldsymbol{\mu})$. The outer optimization over K can then be posed as,

$$\boxed{\begin{aligned} \underset{K}{\text{minimize}} \quad & \mathcal{E}(K) = K^2 \mathcal{I}^*(\boldsymbol{\mu}) \\ \text{subject to} \quad & K \geq \max\{|b_i|\} \\ & \boldsymbol{\mu} = \mathbf{b}/K \end{aligned}} \tag{6.54}$$

In timBRF, the optimal scaling, K^*, is always the minimal feasible value, $K = \max\{|b_i|\}$. This is not always the case in tcmBRF. For example, assume the input is a narrowband auto-regressive process with $R_{xx}[m] = 0.9^{|m|}$ and the desired continuous-valued filter has three taps $\mathbf{b} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{2}} \end{bmatrix}$. Figure 6-5 plots the MSE objective of Eqn.(6.54) as a function of K.

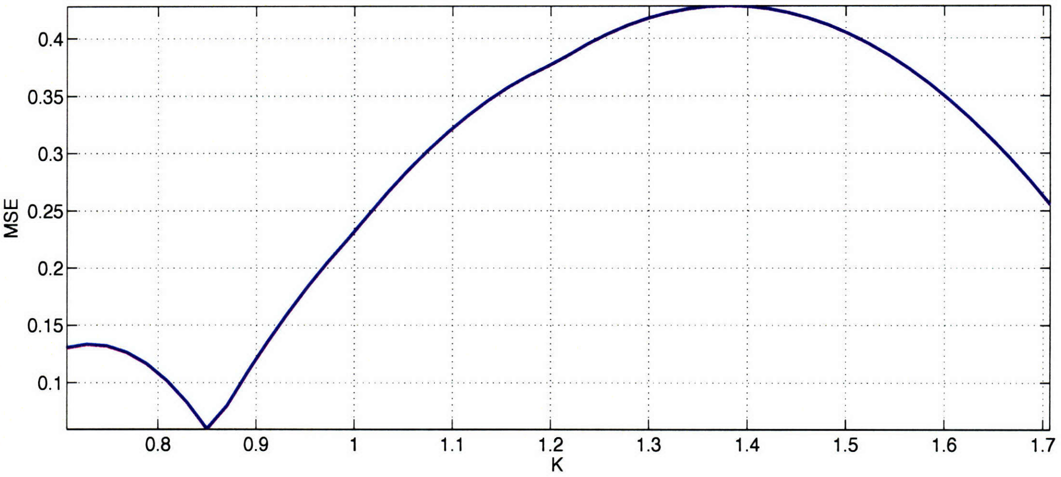


Figure 6-5. Plot of optimal MSE $\mathcal{E}^*(K)$ for $\mathbf{b} = [\frac{1}{2}\ \frac{1}{2}\ \frac{1}{\sqrt{2}}]$. Minimum is not at $K = \max\{|b_i|\}$.

As illustrated, for this example the minimum value does not occur at the extremal value $K = \max\{|b_i|\} = \frac{1}{\sqrt{2}}$, but rather at an interior point around $K \approx 0.85$. In addition, note the non-convex structure of the objective in Figure 6-5. These two properties of $\mathcal{E}(K)$ complicate the design process. Because the objective is not convex over K, gradient-descent can converge to local minima. To find the global minimum over K, a direct line-search must be done. This line-search is computationally expensive because it requires multiple evaluations of the inner affine program for $\mathcal{I}^*(\boldsymbol{\mu})$. We can speed up the line-search by using an adaptive evaluation technique, moving from a coarser grid to a finer one in the vicinity of the coarse minimum.

Despite these speed-up techniques, the full optimization, consisting of a line-search composed with an affine program, becomes computationally intractable as the number of filter taps, N, increases. This is because the number of decision variables in joint-density, $\mathbf{p}$, grows exponentially as 2^N. The exponentially growing decision space becomes computationally intractable even for the most powerful linear programming solvers. Even reasonable filter lengths, like $N = 50$, lead to 2^{50} decision variables, an intractably large number for a LP solver. Even with powerful computers, we can only find the optimal solution for very small filter lengths, on the order of $N \approx 10 - 20$. For practical filter lengths, a more tractable design procedure is necessary. We present one using a parametric model in Section 6.3.4.

6.3.3 Scale Invariance

Like timBRF, the optimal tcmBRF implementation point of a filter $\mathbf{b}$ is scale invariant. Any filter in the direction $\mathbf{e}_\mathrm{b}$ such that $\mathbf{b} = C\mathbf{e}_\mathrm{b}$, with $C > 0$, has a optimal tcmBRF implementation point, $\boldsymbol{\mu}^*(\mathbf{e}_\mathrm{b}) \in [0,1]^N$ that is independent of C. In other words, all tcmBRFs in the direction $\mathbf{e}_\mathrm{b}$, regardless of their total power, are optimally implemented at a point $\boldsymbol{\mu}^*(\mathbf{e}_\mathrm{b})$. Unlike timBRF though, this optimal implementation point is not necessarily one such that $\|\boldsymbol{\mu}^*(\mathbf{e}_\mathrm{b})\|_\infty = 1$, i.e. on the face of the unit cube.

The proof of scale invariance can be done by contradiction. Assume that for a deterministic filter $\mathbf{b}_0$, the optimal implementation point is $\boldsymbol{\mu}_0 = \frac{\mathbf{b}_0}{K_0}$. We denote the maximum element of $\mathbf{b}_0$ using the infinity-norm notation, i.e. $\|\mathbf{b}_0\|_\infty = \max\{|\mathbf{b}_0|\}$ Optimality implies that,

$$K_0^2\mathcal{I}^*\left(\frac{\mathbf{b}_0}{K_0}\right) \leq K^2\mathcal{I}^*\left(\frac{\mathbf{b}_0}{K}\right) \quad , \forall K \neq K_0,\, K \in [\|\mathbf{b}_0\|_\infty, \infty) \tag{6.55}$$

In addition, assume that for $\mathbf{b}_1 = C\mathbf{b}_0$, a scaled version of $\mathbf{b}_0$, with $C > 0$, the optimal implementation point is different. Mathematically, this means $\boldsymbol{\mu}_1 = \frac{\mathbf{b}_1}{K_1} \neq \boldsymbol{\mu}_0 = \frac{\mathbf{b}_0}{K_0}$. This in turn implies that,

$$K_0 \neq \frac{K_1}{C} \tag{6.56}$$

Since $\boldsymbol{\mu}_1 = \frac{\mathbf{b}_1}{K_1}$ is the optimal implementation point for $\mathbf{b}_1$, this implies that,

$$K_1^2\mathcal{I}^*\left(\frac{\mathbf{b}_1}{K_1}\right) \leq K^2\mathcal{I}^*\left(\frac{\mathbf{b}_1}{K}\right) \quad , \forall K \neq K_1,\, K \in [\|\mathbf{b}_1\|_\infty, \infty) \tag{6.57}$$

Multiplying both sides of Eqn.(6.57) by $\frac{1}{C^2} > 0$ gives the inequality,

$$\frac{K_1^2}{C^2}\mathcal{I}^*\left(\frac{\mathbf{b}_1}{K_1}\right) \leq \frac{K^2}{C^2}\mathcal{I}^*\left(\frac{\mathbf{b}_1}{K}\right) \quad , \forall K \neq K_1,\, K \in [\|\mathbf{b}_1\|_\infty, \infty) \tag{6.58}$$

Substituting our assumption that $\mathbf{b}_1 = C\mathbf{b}_0$ and doing some manipulation the expression becomes,

$$\left(\frac{K_1}{C}\right)^2\mathcal{I}^*\left(\frac{\mathbf{b}_0}{(K_1/C)}\right) \leq \left(\frac{K}{C}\right)^2\mathcal{I}^*\left(\frac{\mathbf{b}_0}{(K/C)}\right) \quad , \forall K \neq K_1,\, K \in [C\|\mathbf{b}_0\|_\infty, \infty) \tag{6.59}$$

Since by Eqn.(6.56), $K_0 \neq \frac{K_1}{C}$, let $K_0 = \frac{K}{C}$. From the fact that $K \in [C\|\mathbf{b}_0\|_\infty, \infty)$, this implies $\frac{K}{C} \in [\|\mathbf{b}_0\|_\infty, \infty)$. In addition, since $K_1 \in [C\|\mathbf{b}_0\|_\infty, \infty)$, this implies that $\frac{K_1}{C} \in [\|\mathbf{b}_0\|_\infty, \infty)$. Consequently, $\frac{K_1}{C}$ is a point such that,

$$\left(\frac{K_1}{C}\right)^2\mathcal{I}^*\left(\frac{\mathbf{b}_0}{(K_1/C)}\right) \leq K_0^2\mathcal{I}^*\left(\frac{\mathbf{b}_0}{K_0}\right) \quad , \frac{K_1}{C} \neq K_0 \tag{6.60}$$

If this inequality is strict then $\frac{K_1}{C}$ is a point for which the MSE is lower than K_0 for $\mathbf{b}_0$. This contradicts our original assumption that $\boldsymbol{\mu}_0 = \frac{\mathbf{b}_0}{K_0}$ is the optimal implementation point for $\mathbf{b}_0$. If Eqn.(6.60) is met with equality, then $\boldsymbol{\mu}_1 = \frac{\mathbf{b}_0}{(K_1/C)}$ was another implementation point for b_0 to begin with. This proves that the optimal implementation point for any filter in the direction $\mathbf{e}_\text{b}$ is at the same point $\boldsymbol{\mu}^*(\mathbf{e}_\text{b})$.

Since all filters in the direction $\mathbf{e}_\text{b}$ can be expressed as $\mathbf{b} = C\boldsymbol{\mu}^*(\mathbf{e}_\text{b})$, the optimal MSE for $\mathbf{b}$ can be expressed as:

$$\mathcal{E}^*(\mathbf{b}) = C^2 \underbrace{K^*(\mathbf{e}_\text{b})\mathcal{I}^*(\boldsymbol{\mu}^*(\mathbf{e}_\text{b}))}_{\mathcal{E}(\mathbf{e}_\text{b})} = C^2\mathcal{E}(\mathbf{e}_\text{b}) \tag{6.61}$$

Following the same argument as in Section 6.2.3, we can prove that the tcmBRF SNR is scale invariant. The SNR only depends upon the direction of the filter $\mathbf{e}_\text{b}$. Mathematically, this can be expressed as,

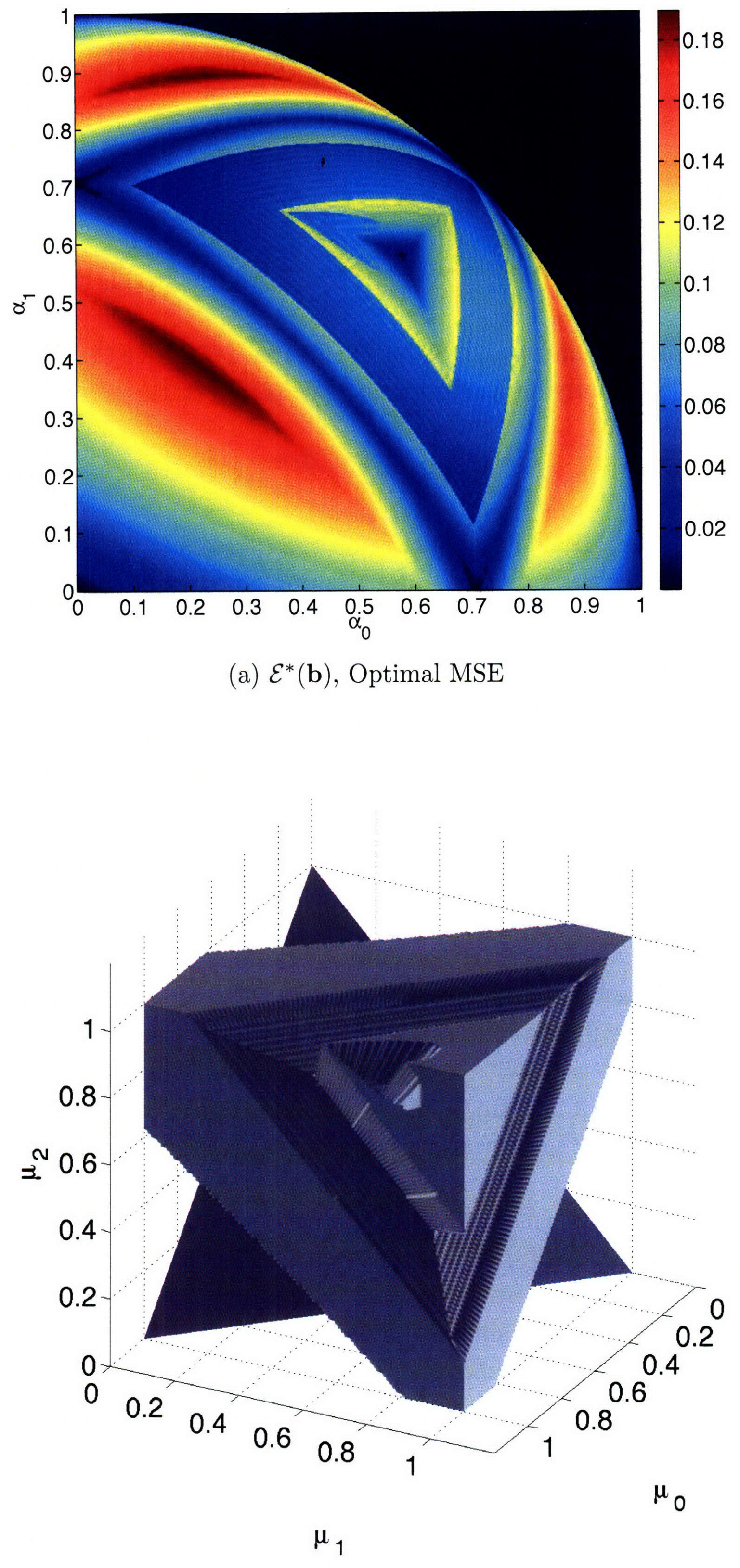
$$\text{SNR}(C\mathbf{e}_\text{b}) = \frac{\mathcal{S}(\mathbf{e}_\text{b})}{\mathcal{E}(\mathbf{e}_\text{b})} \tag{6.62}$$

Geometrically, the tcmBRF implementation manifold is a $N-1$-dimensional surface inside the unit-cube. Within the unit-cube, the implementation manifold is closer to the outer faces of the cube than the origin. This is because the leading K^2 term in the MSE quadratically penalizes small values of $\|\boldsymbol{\mu}^*(\mathbf{e}_\text{b})\|_\infty$, i.e. large values of K. On the other hand, the effect from the $\mathcal{I}^*(\boldsymbol{\mu})$ term can offset the K^2 term in certain directions so that the optimal implementation point is not of the face of the cube.

The shape of the implementation manifold is dependent on the input auto-correlation, $\mathbf{R}_\mathbf{x}$. For a white input, $\mathbf{R}_\mathbf{x} = \mathbf{I}$, the implementation manifold is exactly the faces of the unit-cube, i.e. $\|\boldsymbol{\mu}(\mathbf{e}_\text{b})\|_\infty = 1$. For a rank-one auto-correlation, $\mathbf{R}_\mathbf{x} = \mathbf{1}\mathbf{1}'$, the implementation manifold is the $N-1$-dimensional unit-simplex inside the unit cube, $\sum_{i=0}^{N-1} \mu_i = 1$. From simulation, we observe that for a general, full-rank $\mathbf{R}_\mathbf{x}$ the implementation manifold is in between these two extremal cases. It is often on the face of the unit-cube, but there are certain directions for which the manifold is inside it.

As an example, Figure 6-6(b) illustrates implementation manifold for all three-tap unit-energy filters with $R_{xx}[m] = 0.9^{|m|}$. It is found numerically using a line-search composed with an affine program, as in Section 6.3.2. Note the complex, non-smooth structure of the implementation manifold with many indentations and discontinuous jumps.

Since the line-search over K is expensive, in certain situations, we can set $K = \max\{|b_i|\}$ and implement the tcmBRF on the face of the unit-cube. Though the solution in this case sub-optimal, fixing K simplifies the tcmBRF design process by removing the outer optimization over K. It saves the computation required for repeated evaluations of the inner optimization. In addition, simulation implies that for full-rank $\mathbf{R}_\mathbf{x}$, fixing K does not have a significant loss in performance. Especially in higher dimensions, it seems that $K = \max\{|b_i|\}$ is often the optimal implementation point. Even in cases that it is not, the error penalty for implementing on the face of the cube is small.



(a) $\mathcal{E}^*(\mathbf{b})$, Optimal MSE

(b) Implementation Manifold

Figure 6-6. Plot of optimal MSE and implementation manifold for 3-tap tcmBRF FIR filters on the unit-power manifold defined by $\sum_{i=0}^{2} b_i^2 = 1$.

6.3.4 Parametric Memoryless BRF

Though the optimal tcmBRF solution can be posed, the numerical optimization is intractable because the number of decision variables grow exponentially with N. To make the design tractable we use a parametric model to generate the random binary vector $\mathbf{h}[n]$. As with SRS, a suitable parametric model should satisfy three properties. First, we must be able to constrain the mean of the binary vector. Second, the covariance matrix, $\mathbf{\Sigma_h}$, must have a tractable expression in terms of a small number of parameters, as opposed to exponentially many. This way the parameters can be optimized for tcmBRF. Lastly, the generation of the binary vector should be easily implementable in hardware.

Though there are numerous techniques to generate random binary vectors, most do not give a tractable expression for the resulting covariance matrix. We use a modified version of the model presented by Qaqish in [27], which alleviates this problem. It can generate correlated binary vectors with a fixed mean and specified covariance matrix. The basic model is presented below for completeness, but the proofs are omitted. For a more detailed description, the reader is referred to [27]. After presenting the model, we discuss the optimization of Qaqish correlated binary vectors for use in tcmBRF.

In our model, modified from [27], a N-element correlated binary vector, $\mathbf{h}[n]$, is generated iteratively. The i-th element of the vector, $h_i[n]$, is generated from the κ previous samples, $h_{i-1}[n], h_{i-2}[n], \ldots h_{i-\kappa}[n]$ as follows,

1. The conditional bias, $\lambda_i[n]$, for the generation of $h_i[n]$ is computed according to the relationship:

$$\lambda_i[n] = \mu_i + \sum_{j=\zeta}^{i-1} \gamma_{ij}(h_j[n] - \mu_j) \tag{6.63}$$

   where $\zeta = \max(0, i-\kappa)$, μ_i is the desired mean of $h_i[n]$, and the constants γ_{ij} are the parameters of the algorithm. The lower limit of the sum, ζ, takes either the value 0 or $i-\kappa$, depending on which is larger. Mathematically, this denotes that for the first κ elements, the lower limit cannot go below zero. For example, the first element can only depend on the zeroth element, regardless of the value of κ.

2. The sample $h_i[n]$ is randomly generated from a binary distribution biased by $\lambda_i[n]$ as follows:

$$h_i[n] = \begin{cases} 1 & \text{with probability } \lambda_i[n] \\ 0 & \text{with probability } 1 - \lambda_i[n] \end{cases} \tag{6.64}$$

The resulting vector $\mathbf{h}[n]$ can be proved to be binary random vector with mean vector, $\boldsymbol{\mu}$, and a covariance matrix, $\mathbf{\Sigma_h}$, that can be expressed as,

$$\mathbf{\Sigma_h}(\gamma) = \mathbf{G}_N \tag{6.65}$$

The matrix $\mathbf{G}_N$ is computed recursively starting from the 1×1 matrix, $\mathbf{G}_0 = \sigma_0^2 = \mu_0(1-\mu_0)$ and the recursion defined by,

$$\mathbf{G}_i = \begin{bmatrix} \mathbf{G}_{i-1} & \mathbf{s}_i \\ \mathbf{s}_i^T & \sigma_i^2 \end{bmatrix} \tag{6.66}$$

where $\sigma_i^2 = \mu_i(1-\mu_i)$ and the i-element column vector, $\mathbf{s}_i$, can be expressed as,

$$\mathbf{s}_i = \mathbf{G}_{i-1}\boldsymbol{\gamma}_i \tag{6.67}$$

where the i-element column vector $\boldsymbol{\gamma}_i$ is a vector of the parameters γ_{ij}, which is of the form,

$$\boldsymbol{\gamma}_i = \begin{bmatrix} \gamma_{i0} \\ \vdots \\ \gamma_{i(i-1)} \end{bmatrix} \tag{6.68}$$

when $\zeta = \max(0, i-\kappa) = 0$, and of the form,

$$\boldsymbol{\gamma}_i = \begin{bmatrix} \gamma_{i0} \\ \vdots \\ \gamma_{i(\kappa-1)} \\ \gamma_{i\kappa} \\ \vdots \\ \gamma_{i(i-1)} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \gamma_{i\kappa} \\ \vdots \\ \gamma_{i(i-1)} \end{bmatrix} \tag{6.69}$$

when $\zeta = \max(0, i-\kappa) = i-\kappa$. As before, the two different forms of $\boldsymbol{\gamma}_i$ denotes that for the first κ elements, the lower limit cannot go below zero.

There are constraints on the parameters γ_{ij}. Specifically, the generated binary vector has the prescribed covariance matrix only if the conditional means are all feasible with $\lambda_i[n] \in [0,1]$. From [27], this implies the restriction that,

$$\sum_{j=\zeta}^{i-1} \gamma_{ij}\mu_j - \sum_{-} \gamma_{ij} \le \mu_i \le 1 + \sum_{j=\zeta}^{i-1} \gamma_{ij}\mu_j - \sum_{+} \gamma_{ij} \tag{6.70}$$

where, as before, $\zeta = \max(0, i-\kappa)$ and $\sum_+$ and $\sum_-$ denote the summation over $\{j : \zeta \le j < i, \gamma_{ij} > 0\}$ and $\{j : \zeta \le j < i, \gamma_{ij} < 0\}$, respectively. We can simplify this constraint by expressing it using absolute values,

$$\frac{1}{2}\sum_{j=\zeta}^{i-1} |\gamma_{ij}| + \sum_{j=\zeta}^{i-1} \gamma_{ij}\left(\frac{1}{2} - \mu_j\right) \le 1 - \mu_i \tag{6.71}$$

$$\frac{1}{2}\sum_{j=\zeta}^{i-1} |\gamma_{ij}| - \sum_{j=\zeta}^{i-1} \gamma_{ij}\left(\frac{1}{2} - \mu_j\right) \le \mu_i \tag{6.72}$$

The constraints of Eqns.(6.71) and (6.72) can be derived in a straightforward manner from Eqn.(6.70). We omit the derivation for the sake of brevity.

This model is a modified version of the Qaqish model presented in [27]. As presented in [27], $\kappa = N$, and each element of the $\boldsymbol{\gamma}_i$ vectors can be non-zero. The total number of parameters in the unmodified model is,

$$\|\gamma\|_0 = \sum_{j=0}^{N-1}\sum_{i=0}^{j-1} I(\gamma_{ij}) = \frac{N(N-1)}{2} \tag{6.73}$$

which grows quadratically in N as $O(N^2)$. In this expression, γ is a stacked vector of all the γ_i for all j. The function $I(\cdot)$ represents an indicator function and the $\|\cdot\|_0$ notation represents the zero pseudo-norm which is a count of all the non-zero elements in γ. In the modified model, we can restrict the level of correlation $\kappa < N$. In each γ_i vector there is then, at most, κ non-zero elements. The total number of parameters in the modified model is,

$$\|\gamma\|_0 = \frac{(\kappa+1)\kappa}{2} + (N-\kappa-1)\kappa \tag{6.74}$$

which grows as linearly in N as $O(\kappa N)$. In essence, κ is a complexity control parameter. We denote it as the Qaqish order. The higher the Qaqish order is, the more covariance matrices we can achieve, but the ensuing optimization is more complex. In practical applications, the system designer must choose an appropriate value of κ based on hardware constraints and performance requirements.

The Qaqish model is a restricted representation of binary vectors. There exist valid binary vector distributions that cannot be achieved using this model. Especially with $\kappa < N$, the coverage of the space of binary vector distributions is far from complete. Consequently, the solution from optimizing over this model is sub-optimal. But, with a linearly growing number of decision variables, the optimization is tractable for large N.

Mathematically, fixing the Qaqish order, κ, and incorporating the constraints of Eqns.(6.71) and (6.72), the tcmBRF design problem can be expressed as a constrained optimization over the parameters γ and K.

$$\boxed{\begin{array}{ll} \underset{K,\gamma}{\text{minimize}} & K^2 \text{tr}\left(\mathbf{R_x}\mathbf{\Sigma_h}(\gamma)\right) \\ & \\ \text{subject to} & K \geq \max\{|b_i|\} \\ & \mu_i = |b_i|/K \\ & \frac{1}{2}\sum_{j=i-\kappa}^{i-1}|\gamma_{ij}| + \frac{1}{2}\sum_{j=i-\kappa}^{i-1}\gamma_{ij}\left(\frac{1}{2}-\mu_j\right) \leq 1-\mu_i \\ & \frac{1}{2}\sum_{j=i-\kappa}^{i-1}|\gamma_{ij}| - \frac{1}{2}\sum_{j=i-\kappa}^{i-1}\gamma_{ij}\left(\frac{1}{2}-\mu_j\right) \leq \mu_i \end{array}} \tag{6.75}$$

As with the optimal solution, the joint optimization over K and γ is difficult, so we split the optimization into an inner optimization over γ for a fixed K and an outer optimization over K. With the Qaqish model the inner optimization is tractable and can be solved using standard non-linear programming techniques, e.g. `fmincon` in MATLAB.

The optimization, though tractable for large N is still problematic. Especially for large κ, the convergence is slow and often the numerical optimization fails to converge to a feasible solution. Though problematic, this constrained optimization is presented as a proof of concept solution. Future work will further study the structure of the inner optimization to see if it can be solved using more robust optimization algorithms, e.g. convex solvers. This is beyond the scope of this thesis though.

It should be noted that this model correlates the binary vector $\mathbf{h}[n]$ sequentially. This works well when the input is low-pass and the strongest correlations in $R_{xx}[m]$ are sequential. More generally, we can consider correlating across a permuted version of $\mathbf{h}[n]$. This is potentially useful when input is band-pass and the strongest correlations in $R_{xx}[m]$ are not sequential across the tapped-delay line, but rather skip taps. In that case, a well designed permutation of $\mathbf{h}[n]$ before optimization can further reduce the MSE. We do not explore the effects of permutations in this thesis though. Future work may consider the development of permutations as a preliminary step in the Qaqish tcmBRF design.

As a final point, note that is is relatively easy to generate correlated binary vectors using the Qaqish algorithm. It requires pseudo-random number generation and the calculation of $\lambda_i[n]$ using Eqn.(6.63). The pseudo-random number generation can be done using LFSRs. The summation of Eqn.(6.63) using the coefficients γ_{ij} can be done using a table lookup. The essential point is that the generation does not require multiplies and can hopefully be done with a small hardware footprint.

6.3.5 Relaxation Bound

Because the binary compatibility constraint is problematic, in this section, we replace it with two less problematic ones. The resulting relaxed optimization is easier to solve and gives a lower bound on the MSE. It is useful for error analysis.

There are two natural restrictions that any covariance matrix, $\mathbf{\Sigma_h} \in \mathcal{B}(\boldsymbol{\mu})$, must satisfy. First $\mathbf{\Sigma_h}$ should be positive semi-definite, i.e. $\mathbf{\Sigma_h} \succeq 0$. Secondly, the correlation $E\{h_i[n]h_j[n]\}$ is constrained by μ_i and μ_j. In particular, for binary random variables, the marginal means limit the ranges of the covariances to, [27],

$$\max(0, \mu_i + \mu_j - 1) - \mu_i\mu_j \leq \sigma_{ij} \leq \min(\mu_i, \mu_j) - \mu_i\mu_j \tag{6.76}$$

The upper bounds in Eqn.(6.76) are all simultaneously achievable by a binary covariance matrix. On the other hand, the lower bounds may not be all simultaneously achievable, [27].

Using these two restrictions, rather than the full compatibility constraint, gives a relaxed version of tcmBRF design that can be expressed as the constrained optimization,

$$\begin{array}{ll} \underset{K,\mathbf{\Sigma_h}}{\text{minimize}} & K^2 \text{tr}\left(\mathbf{R_x}\mathbf{\Sigma_h}\right) \\ \\ \text{subject to} & K \geq \max\{|b_i|\} \\ & \mu_i = b_i/K \\ & \sigma_i^2 = \mu_i(1-\mu_i) \\ & \sigma_{ij} \leq \min(\mu_i, \mu_j) - \mu_i\mu_j \\ & \sigma_{ij} \geq \max(0, \mu_i + \mu_j - 1) - \mu_i\mu_j \\ & \mathbf{\Sigma_h} \succeq 0 \end{array} \tag{6.77}$$

The feasible region for optimal design problem, Eqn.(6.40), with the binary compatibility constraint, is a subset of the feasible region of this relaxed program. Consequently, the optimal value of Eqn.(6.77), which is denoted $\mathcal{E}_{\mathrm{r}}(\mathbf{b})$, gives a lower bound on the achievable tcmBRF MSE.

As before, the relaxed optimization can be solved by separating the joint optimization, Eqn.(6.77), into an inner optimization over $\mathbf{\Sigma_h}$ for fixed K and outer optimization over K.

The inner optimization can be expressed as a constrained optimization over the semi-definite matrix $\mathbf{\Sigma_h}$ with element-wise constraints,

$$\begin{aligned} \underset{\mathbf{\Sigma_h}}{\text{minimize}} \quad & \mathcal{I}_{\mathrm{r}}(\mathbf{\Sigma_h}, \boldsymbol{\mu}) = \mathrm{tr}\left(\mathbf{R_x}\mathbf{\Sigma_h}\right) \\ \\ \text{subject to} \quad & \sigma_i^2 = \mu_i(1-\mu_i) \\ & \sigma_{ij} \leq \min(\mu_i, \mu_j) - \mu_i\mu_j \\ & \sigma_{ij} \geq \max(0, \mu_i + \mu_j - 1) - \mu_i\mu_j \\ & \mathbf{\Sigma_h} \succeq 0 \end{aligned} \tag{6.78}$$

This is a semi-definite program; a conic, convex optimization. It can be efficiently solved using standard interior-point solvers like SDPT-3 or CVX, [7, 18]. Defining the minimum value of the inner optimization as $\mathcal{I}_{\mathrm{r}}^*(\boldsymbol{\mu})$, the outer optimization over K can be expressed as,

$$\begin{aligned} \underset{K}{\text{minimize}} \quad & K^2\mathcal{I}_{\mathrm{r}}^*(\boldsymbol{\mu}) \\ \\ \text{subject to} \quad & K \geq \max\{|b_i|\} \\ & \boldsymbol{\mu} = \mathbf{b}/K \end{aligned} \tag{6.79}$$

As before, the outer optimization can be solved using a line-search. The benefit of relaxation comes from the fact that there are very efficient solvers for semi-definite programs. It is far easier to solve this relaxed program than either the optimal affine program of Section 6.3.2 or the parametric optimization of 6.3.4.

The drawback, of course, is that the solution to the relaxed program only gives a lower bound on the MSE. We denote this as the relaxation bound. We conjecture that, in general, the relaxation bound is not tight, i.e. there does not exist a binary distribution that achieves the optimal relaxed covariance matrix, $\mathbf{\Sigma}_{\mathbf{r}}^*$. In addition, there does not seem to be any simple method to approximate $\mathbf{\Sigma}_{\mathbf{r}}^*$ using a binary distribution. For example, we have attempted to approximate $\mathbf{\Sigma}_{\mathbf{r}}^*$ using the parametric Qaqish model, but the back-solving procedure is extremely unstable. In particular, the back-solved parameters, γ, are highly infeasible when $\mathbf{\Sigma}_{\mathbf{r}}^*$ is not compatible with a binary distribution. Despite these drawbacks, the relaxation bound is useful because it gives a fundamental limit on the correlation gain. In addition, since it is easy to compute, the relaxation bound is useful for error analysis.

6.3.6 Error Analysis

As with timBRF, the tcmBRF MSE is dependent on the desired continuous-valued filter we are trying to implement. Certain filters, like $\mathbf{b} = \left[\frac{1}{2} \;\; \frac{1}{2} \;\; \frac{1}{2}\right]$, can be perfectly implemented using a tcmBRF and have zero MSE. Other filters have a much higher MSE. As an example of how the MSE varies as a function of the taps, Figure Figure 6-6(a) illustrates the optimal MSE, $\mathcal{E}^*(\mathbf{b})$, for all positive-valued, three-tap tcmBRFs with unit-energy, i.e. $b_0^2+b_1^2+b_2^2 = 1$ and $b_i \geq 0$. The input auto-correlation is fixed to $R_{xx}[m] = 0.9^{|m|}$. The figure is plotted as a projection into the (b_0, b_1) plane. The b_2 coordinate is implicit from the unit-energy constraint. Note that the symmetries that existed in timBRF are broken in tcmBRF because the input auto-correlation affects the MSE surface.

Whatever the filter though, the tcmBRF MSE lower than the timBRF MSE. We denote the gain over timBRF as correlation gain. Mathematically, for a fixed filter the correlation gain is defined as the ratio between the optimal timBRF and tcmBRF MSE,

$$\mathcal{G}_{\text{tcm}}(\mathbf{b}) = \frac{\mathcal{E}^*_{\text{tim}}(\mathbf{b})}{\mathcal{E}^*_{\text{tcm}}(\mathbf{b})} \tag{6.80}$$

As with the MSE, the correlation gain is a function of the desired continuous-valued filter $\mathbf{b}$. Figure 6-7 illustrates the correlation gain for all positive-value three-tap tcmBRFs with unit-energy. For certain filters, there is little correlation gain because both the timBRF and tcmBRF implementations have similar MSE, e.g. $\mathbf{b} = \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix}$ which has zero MSE for both types of BRF. For other filters, there is a significant correlation gain, e.g. $\mathbf{b} = \begin{bmatrix} \frac{1}{\sqrt{2}}\sqrt{1+\frac{1}{\sqrt{3}}} & \sqrt{\frac{1}{4}-\frac{1}{2\sqrt{3}}} & \sqrt{\frac{1}{4}-\frac{1}{2\sqrt{3}}} \end{bmatrix}$ which has the maximal timBRF MSE.

In addition to the desired filter, the input auto-correlation, $\mathbf{R_x}$, affects the correlation gain. For example, with a white input, $R_{xx}[m] = R_{xx}[0]\delta[m]$, there is no correlation gain, i.e. $\mathcal{G}_{\text{tcm}} = 0$ dB for all filters $\mathbf{b}$. Mathematically, in Eqn.(6.35) the contributions from the cross terms are all zero because $R_{xx}[i-j] = 0$ for $i \neq j$. Intuitively, with no correlation in the input, we cannot design the tap processes to have partial cancellation at the accumulator.

On the other hand, with a constant input, $R_{xx}[m] = C$, there is infinite correlation gain, i.e. $\mathcal{G}_{\text{tcm}} = \infty$ for all filters $\mathbf{b}$. In this case, there are an infinite number of solutions that have zero MSE. One possible solution is to choose $K = \left|\sum_{i=0}^{N-1} b_i\right| = |A(e^{j0})|$ and set the zeroth tap to $h_0[n] = \text{sgn}\{A(e^{j0})\}$ and the rest to zero. Intuitively, with a constant input the output is a constant with scaling $\sum_{i=1}^{N-1} b_i = A(e^{j0})$. This can be always matched using the single scaling element in the BRF.

In between these two extremal cases, correlation gain increases the more narrowband the input is, i.e the closer $R_{xx}[m]$ is to a constant. Intuitively, the more narrowband the input is, the more predictable the process $x[n]$ is from samples of its past. This, in turn, leads to a better ability to cancel errors at the accumulator because, in its essence, the cancellation from tcmBRF is like a binary vector prediction system. We explore the effect $\mathbf{R_x}$ on correlation gain in more detail using numerical experiments in Section 6.4.

Lastly, for parametric tcmBRF, the Qaqish order, κ, affects the correlation gain. For a fixed $\mathbf{b}$ and $\mathbf{R_x}$, increasing κ improves correlation gain because the feasible region is larger. Most of the benefit of correlation comes from the largest elements in the auto-correlation, $R_{xx}[m]$ though. Once these have been taken into account, the contribution from the remaining terms is much smaller. Consequently, depending on the decay of the auto-correlation, $R_{xx}[m]$, there is diminishing correlation gain with κ. Figure 7-6 in Section 6.4 illustrates the effect of κ on correlation gain for an example filter.

As with timBRF, for a fixed ideal filter specification, the tcmBRF MSE increases as the number of taps N increases. Because of correlation gain though, the tcmBRF MSE increases at slower rate. As such, tcmBRF gives a better error tradeoff than timBRF. This allows us to make a longer, better filter for same MSE constraint. Figure 6-14 in Section 6.4 illustrates the tcmBRF MSE as a function of N for and example specification. It illustrates that the correlation gain is about constant for all N.

The error scaling is different for different filters $\mathbf{b}$, but like in timBRF, we can do a worst-case error analysis for unit-power filters with fixed $\mathbf{R_x}$. This gives an upper-bound on how the tcmBRF MSE scales with N for any filter specification. Doing the worst-case analysis with the binary compatibility constraint is difficult, especially since the optimal solution is intractable. Instead, we develop the worst-case analysis using the relaxation bound. Mathematically, for a fixed N and $\mathbf{R_x}$, the relaxation bound on the worst-case MSE can be expressed as the maximization of the form,

$$\begin{array}{ll} \underset{\mathbf{b}}{\text{maximize}} & \mathcal{E}_{\mathrm{r}}^*(\mathbf{b}) \\ \\ \text{subject to} & \sum_{i=0}^{N-1} b_i^2 = 1 \\ & b_i \geq 0 \end{array} \tag{6.81}$$

Where $\mathcal{E}_{\mathrm{r}}^*(\mathbf{b})$ is the solution to the relaxation program for a fixed $\mathbf{b}$. An analytic solution is desirable, but unfortunately, we have been unable to find a closed form expression for this optimization. We can compute the worst-case relaxation bound numerically instead. We conjecture that the worst-case scaling depends upon the minimum singular value of $\mathbf{R_x}$ in some way. Future work should consider a more robust error analysis of tcmBRF.

6.4 Numerical Experiments

In this section, numerical simulations are used to illustrate the effect of BRF for an example. The properties of both timBRF and tcmBRF are shown in the time and frequency domains. In addition, simulations are used to illustrate the error scaling and the effect of peakiness on the error.

6.4.1 Example

In this section, we illustrate the effect of timBRF and tcmBRF for a low-pass filtering example. The input $x[n]$ is a WSS DT ARMA process generated by shaping white Gaussian noise through a filter:

$$G(z) = \frac{(z - z_0)(z - z_0^*)}{(z - p_0)(z - p_1)} \tag{6.82}$$

$$z_0 = e^{j\pi/2}, \; p_0 = 0.9, \; p_1 = -0.6$$

The input power spectrum is $S_{xx}(e^{j\omega}) = G(z)G(z^{-1})$. It is illustrated in Figure 6-10(a). The desired continuous-valued filter is a $N = 33$ tap, linear-phase, Parks-McClellan low-pass filter designed using the specifications:

$$H(e^{j\omega}) = \begin{cases} 1 & \text{for } \omega_p = [0, 3\pi/16] \\ 0 & \text{for } \omega_s = [5\pi/16, \pi] \end{cases} \tag{6.83}$$

The impulse response and frequency response of this filter is illustrated in Figure 6-9. Figure 6-10(b) illustrates the power spectrum of the desired output, $S_{yy}(e^{j\omega})$. In this example, the low-pass filter removes the unwanted high-frequency components in $S_{xx}(e^{j\omega})$ while preserving the low-pass components.

We design a timBRF implementation using the optimization Eqn.(6.5). We design a tcmBRF implementation with Qaqish order $\kappa = 4$ using the optimization Eqn.(6.75). For each, we simulate the filter in MALTAB and generate two million samples of the error $e[n]$. Periodogram averaging with a Hamming window of size 2048 with 50% overlap is used to approximate 2048 samples of the power spectrum $S_{ee}(\omega)$. The MSE is estimated numerically by averaging the squared difference between $y[n]$, the desired output of the continuous-valued filter, and $\hat{y}[n]$ output of the BRF.

Figure 6-11 illustrates the results of timBRF. Figure 6-11(a) illustrates $S_{ee}(e^{j\omega})$. As expected, it is white with height given by Eqn.(6.17). Figure 6-11(b) illustrates a section of the output, $\hat{y}[n]$, in the time domain along with the desired output, $y[n]$. The output of the timBRF is a degraded version of the desired output $y[n]$. As noted, for this example the timBRF SNR is 7.18 dB.

Figure 6-12 illustrates the results of tcmBRF for the same signal. The error spectrum is still white, but the noise floor has been reduced due to correlation gain. Accordingly, in the time-domain, the tcmBRF output, $\hat{y}[n]$, more closely follows the desired output, $y[n]$. The SNR is 10.58 dB, with a correlation gain of 3.40 dB over timBRF. Bound is at 13.81 dB. The gap to bound is 3.24 dB.

Figure 6-13(a) illustrates the optimal covariance matrix $\mathbf{\Sigma_h}^*$ found using parametric tcmBRF optimization with $\kappa =$. Compare this to Figure 6-13(b) which illustrates the optimal covariance matrix $\mathbf{\Sigma_r}^*$ from the relaxation bound. Because of the sequential correlation with order $\kappa = 4$, the Qaqish covariance matrix cannot have high covariances away from the main diagonal. Consequently, the Qaqish matrix matches the relaxation bound matrix close to the diagonal, but has significant differences away from it.

Figure 7-6 illustrates the correlation gain for this example as a function of κ, the Qaqish order. We observe that for larger κ there is more correlation gain, but there are diminishing returns. Above $\kappa = 8$ the gain plateaus. This is because beyond $m = 8$ the auto-correlation $R_{xx}[m]$ has decayed away considerably – enough so that the remaining terms have a small contribution to correlation gain.

Figure 6-14(a) illustrates MSE scaling for this particular example as a function of N. Figure 6-14(b) illustrates the same results in terms of correlation gain. Figure 6-16(b) illustrates the same results in terms of SNR. There are a number of observations to make. First, as noted in the error analysis sections, the MSE grows with N on the order $O(\sqrt{N})$. Comparing Figure 6-14(a) with Figure 6-8, the error scaling is slower than the worst-case by a large multiplicative factor. Secondly, the correlation gain is approximately a constant as a function of N. It is about 2.6 dB for $\kappa = 2$ and about 3.5 dB for $\kappa = 4$. From the relaxation bound, it seems that there is room for more improvement with a potential correlation gain of up to 6.6 dB. Of course, the relaxation bound may not be tight, so the full gain of 6.6 dB may not be achievable.

As noted in the error analysis, even though the MSE increases, with more taps we can implement a better filter. For this Parks-McClellan example, we can measure the the filter performance using the max ripple error. Figure 6-17 illustrates the max ripple error as a function of N for the example specifications. It decays quickly with N. On the other hand, the MSE due to BRF implementation grows with N. The system designer must make a tradeoff between the max ripple error in Fig.6-17 and the randomization error MSE in 6-14 to choose an operating point.

6.4.2 Peakiness

A discussed in Section 6.3.6, the peakiness of the input power spectrum affects the tcmBRF performance. The more narrowband the input is, the more correlation gain we can achieve.

We illustrate this dependence using a numerical experiment. We assume the desired continuous valued filter is the same as in the previous section, the $N = 33$ tap Parks-McClellan low pass filter illustrated in Fig. 6-9.

In the experiment, we generate the input $x[n]$ as first-order auto-regressive AR(1) process with power spectrum,

$$S_{xx}(z) = \frac{1}{(1 - \rho z^{-1})(1 - \rho z)} \tag{6.84}$$

Figure 6-19 plots the correlation gain, $\mathcal{G}_{\text{tcm}}$, as ρ is varied from zero to one. Each point on the curve is found by numerical simulation of one-million samples. As expected, when $\rho = 0$, the spectrum is white and there is no correlation gain, i.e. $\mathcal{G}_{\text{tcm}} = 0$ dB. As ρ increases the pole moves closer to the unit circle, making $S_{xx}(e^{j\omega})$ more peaky. The correlation gain increases exponentially for all κ. As ρ approaches 1, the AR(1) process becomes more like a constant and the correlation gain approaches infinity. Also, as expected, the correlation gain for $\kappa = 4$ is always above that of $\kappa = 2$ because with more degrees of freedom we can find a better solution. Another observation is that as ρ increases, the gap to the relaxation bound increases. This is because with a peakier input there is more potential correlation gain that the Qaqish model with a restricted κ cannot take advantage of.

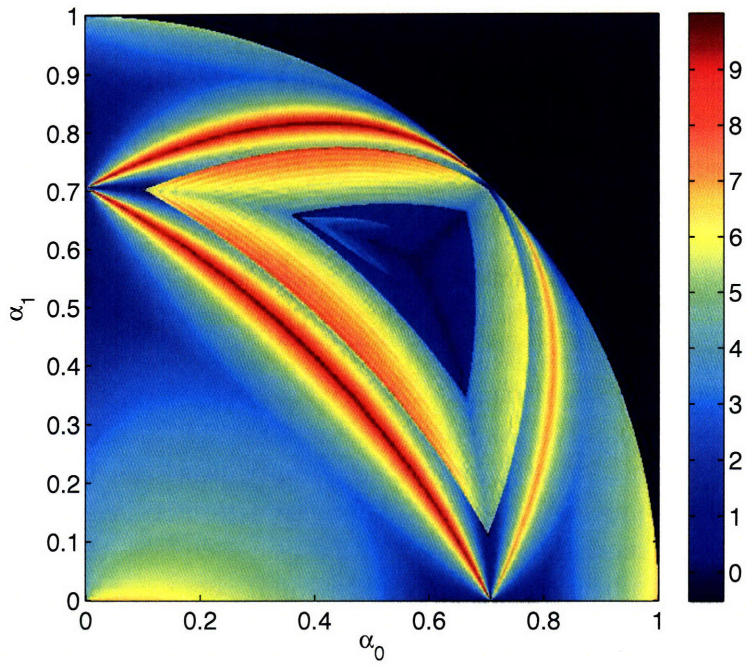


Figure 6-7. Correlation gain, $\mathcal{G}_{\text{tcm}}$ for 3-tap tcmBRF FIR filters on the unit-power manifold defined by $\sum_{i=0}^{2} b_i^2 = 1$.

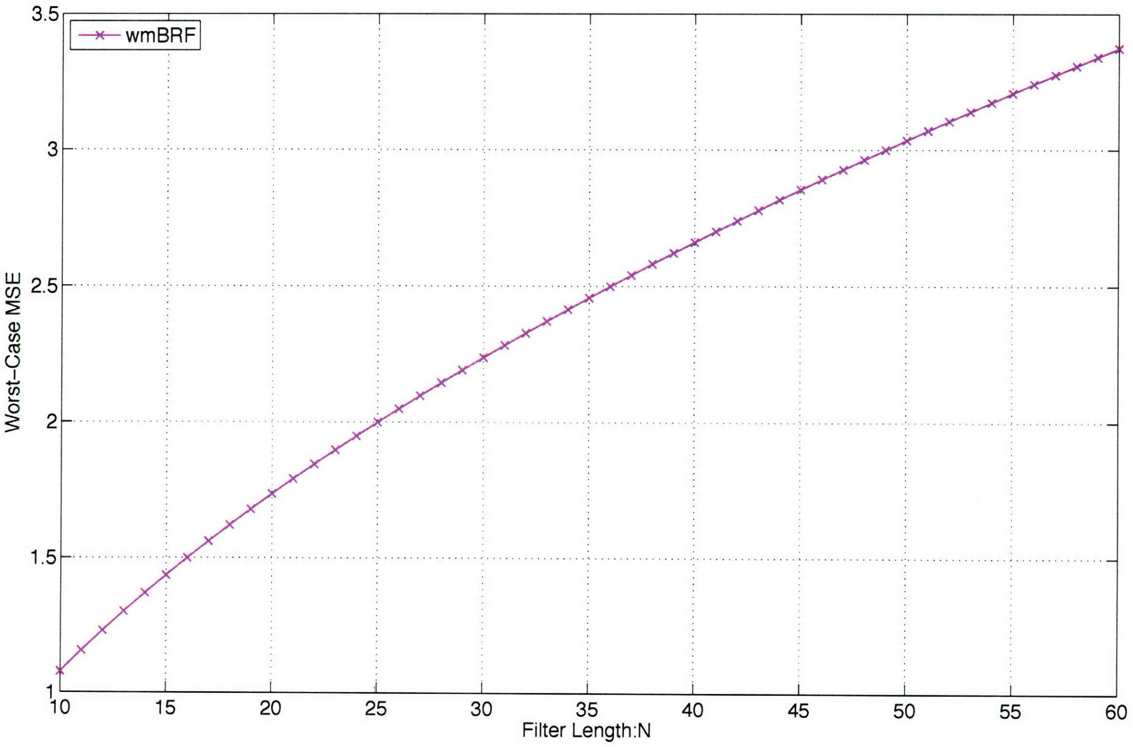
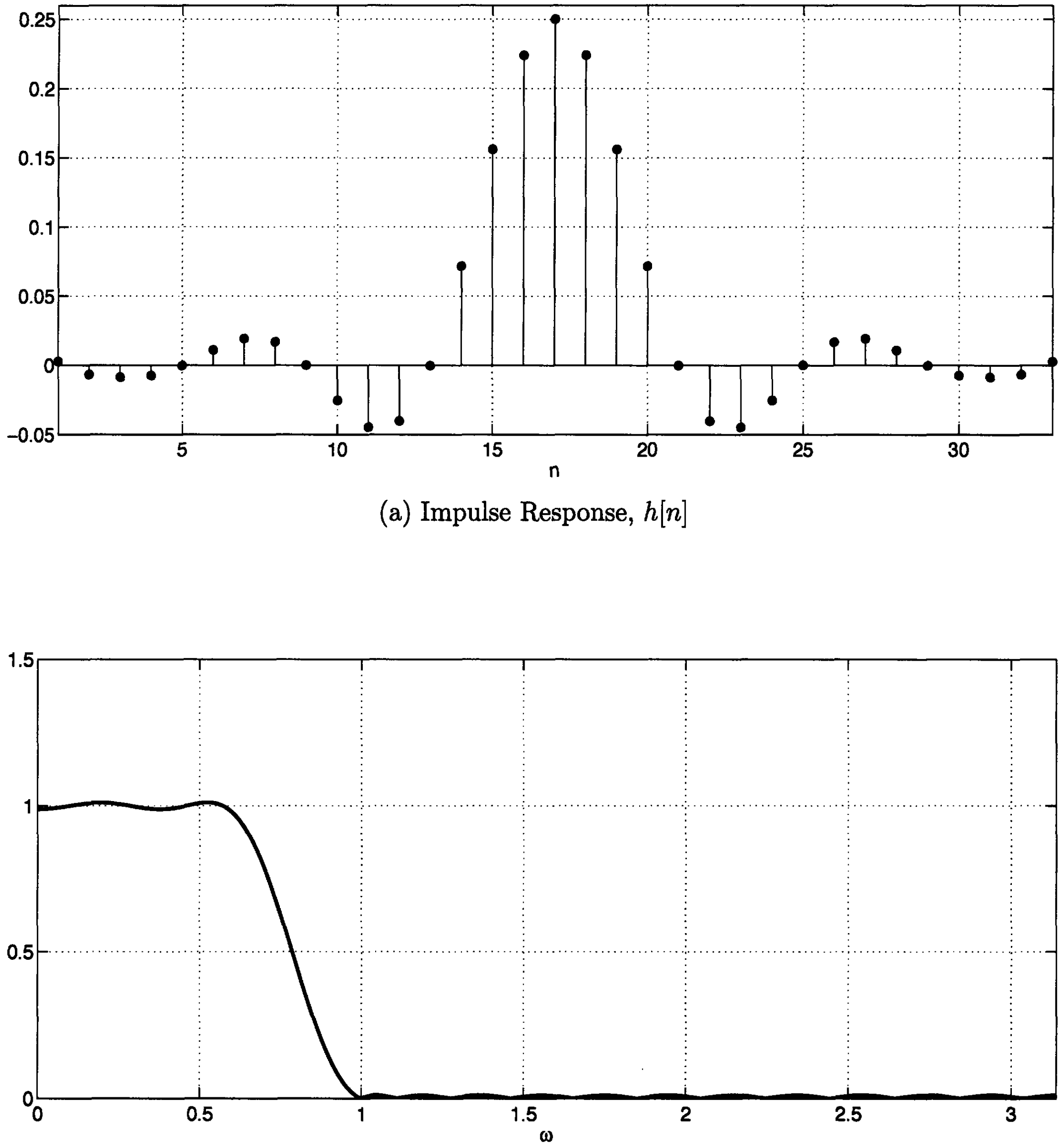


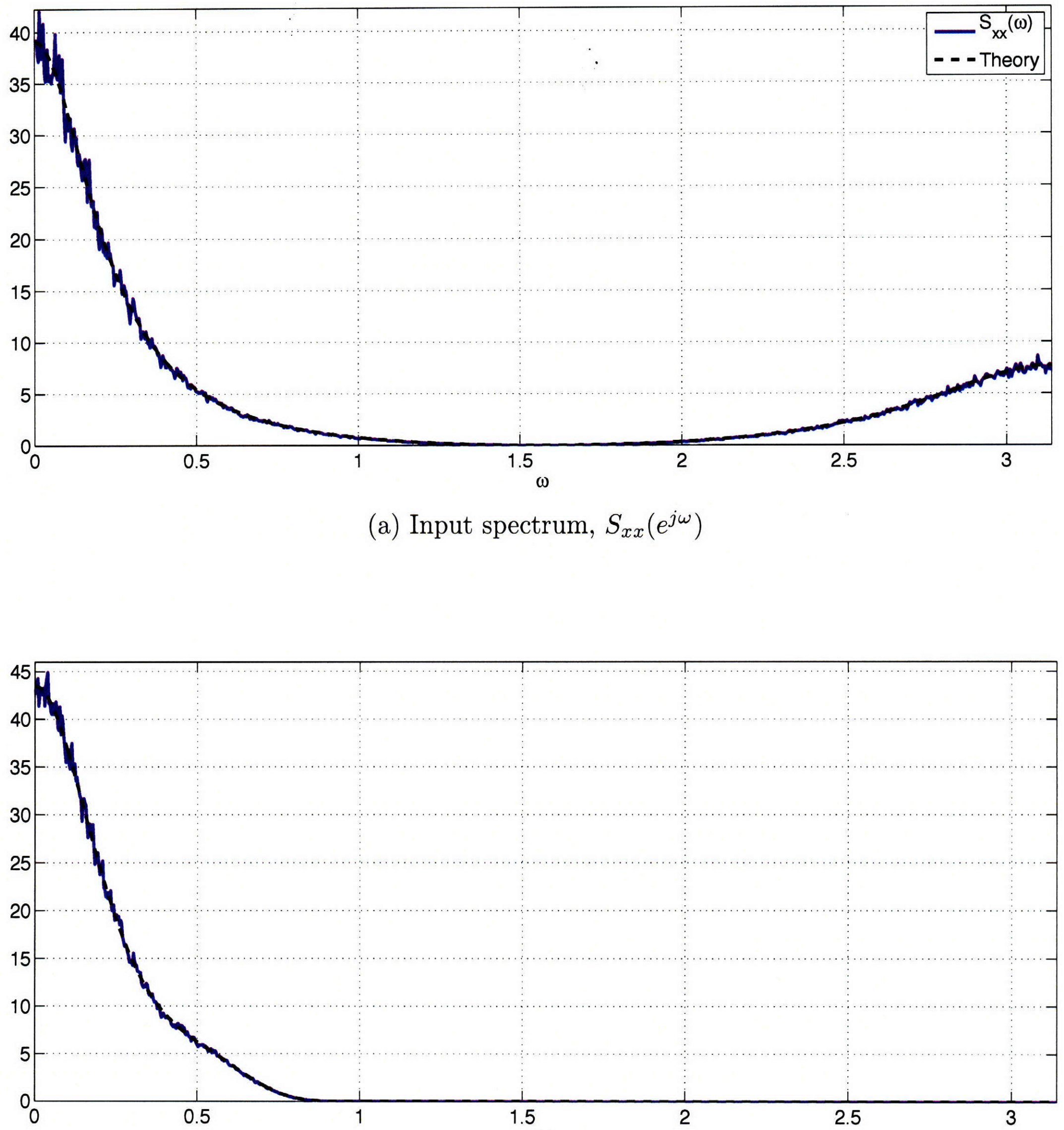
Figure 6-8. Worst-case MSE as a function of N. timBRF curve is analytic expression from Eqn.(6.33). tcmBRF relaxation bound curves are a result of numerical optimization using `fmincon` using random restarts. Input is normalized so $R_{xx}[0] = 1$.



(a) Impulse Response, $h[n]$

(b) Magnitude Response, $|H(e^{j\omega})|$

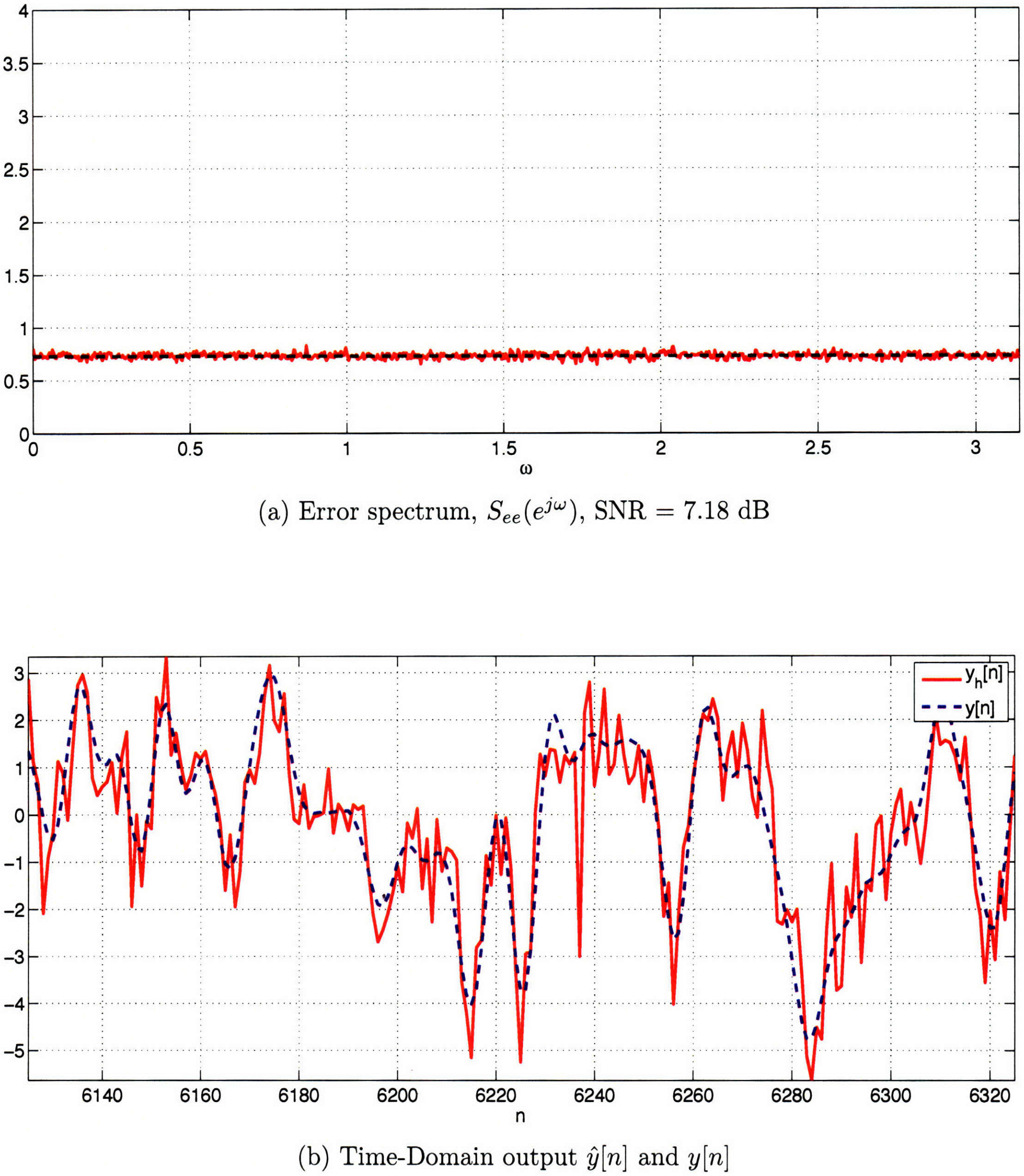
Figure 6-9. Desired continuous-valued filter. $N = 33$ length Parks-McClellan filter designed using the specifications of Eqn.(6.83).



(a) Input spectrum, $S_{xx}(e^{j\omega})$

(b) Desired output spectrum, $S_{yy}(e^{j\omega})$

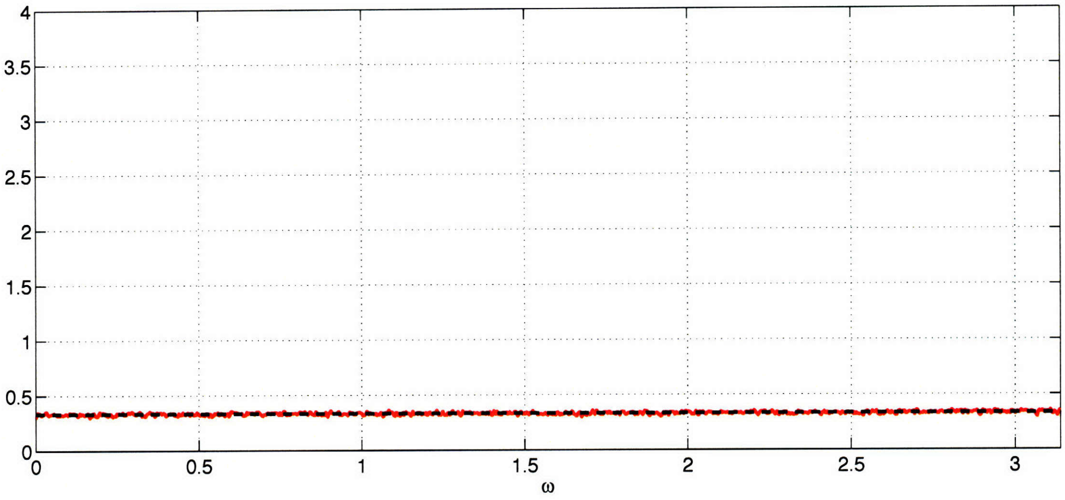
Figure 6-10. Power spectra of input and desired output.



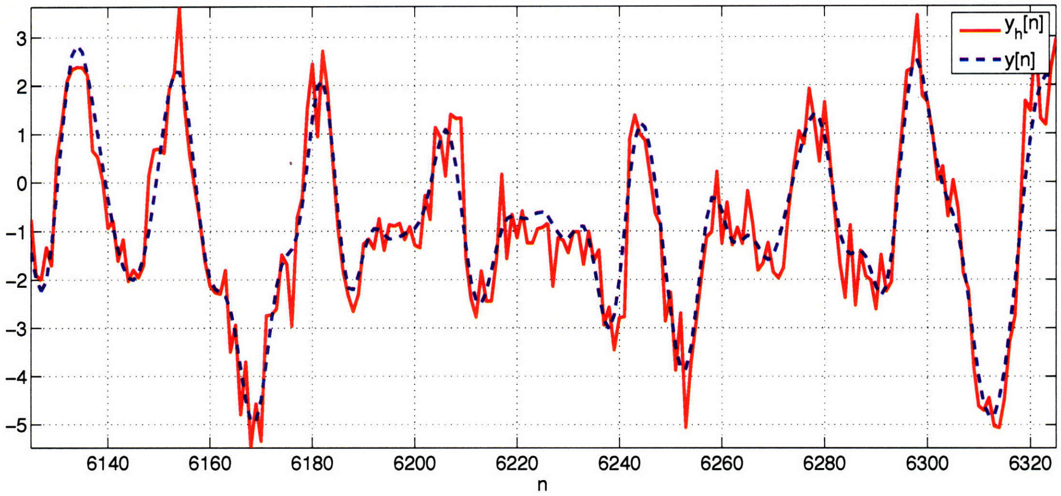
(a) Error spectrum, $S_{ee}(e^{j\omega})$, SNR = 7.18 dB

(b) Time-Domain output $\hat{y}[n]$ and $y[n]$

Figure 6-11. Error power spectrum and time-domain output for tap-independent memoryless BRF implementation of the example in Section 6.4.1.

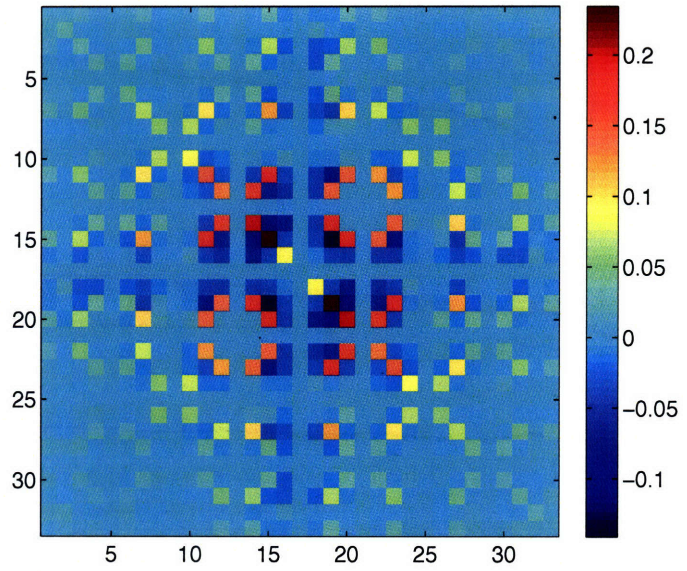


(a) Error spectrum, $S_{ee}(e^{j\omega})$, SNR = 10.58 dB

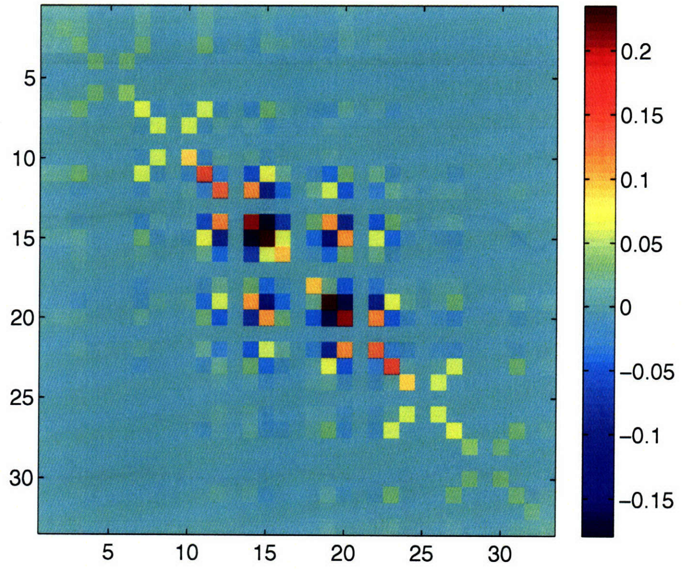


(b) Time-Domain output $\hat{y}[n]$ and $y[n]$

Figure 6-12. Error power spectrum and time-domain output for tap-correlated memoryless BRF implementation of the example in Section 6.4.1 with Qaqish order $\kappa = 4$.



(a) $\mathbf{\Sigma_r}^*$, optimal covariance matrix from relaxation bound.



(b) $\mathbf{\Sigma_h}^*$, optimal covariance matrix from Qaqish optimization with $\kappa = 4$.

Figure 6-13. Optimal covariance matrices from relaxation bound and Qaqish parametric optimization of the $N = 33$ tap example in Section 6.4.1.

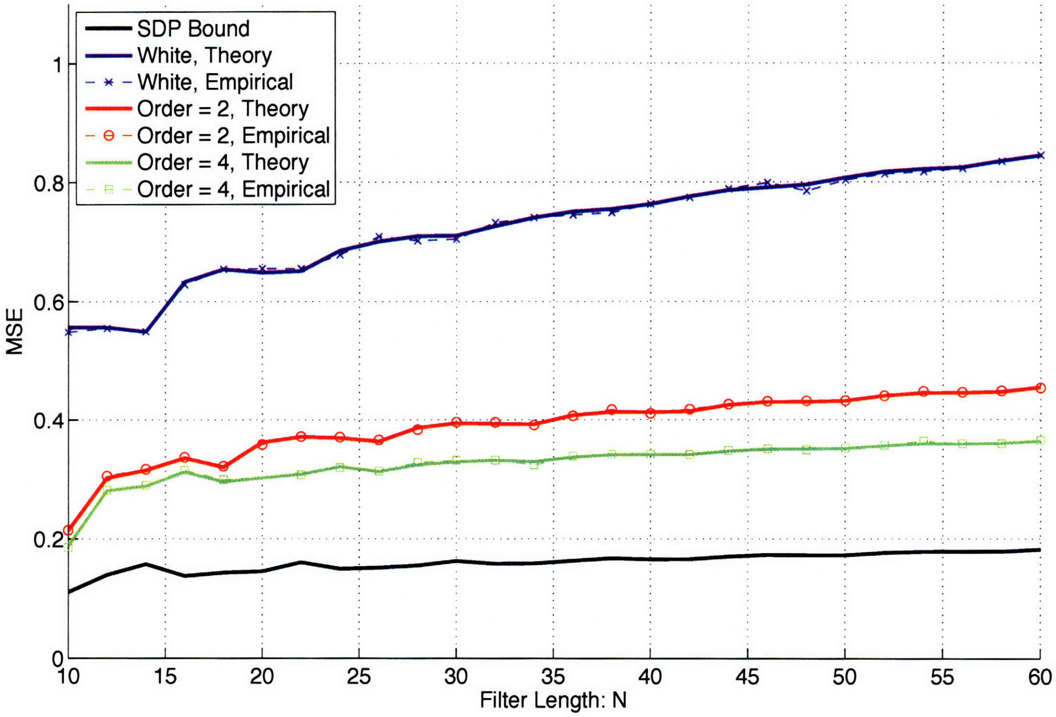


Figure 6-14. MSE scaling as a function of N for the Parks-McClellan filter with specifications given in Eqn.(6.83) and input spectrum given in Section 6.4.1.

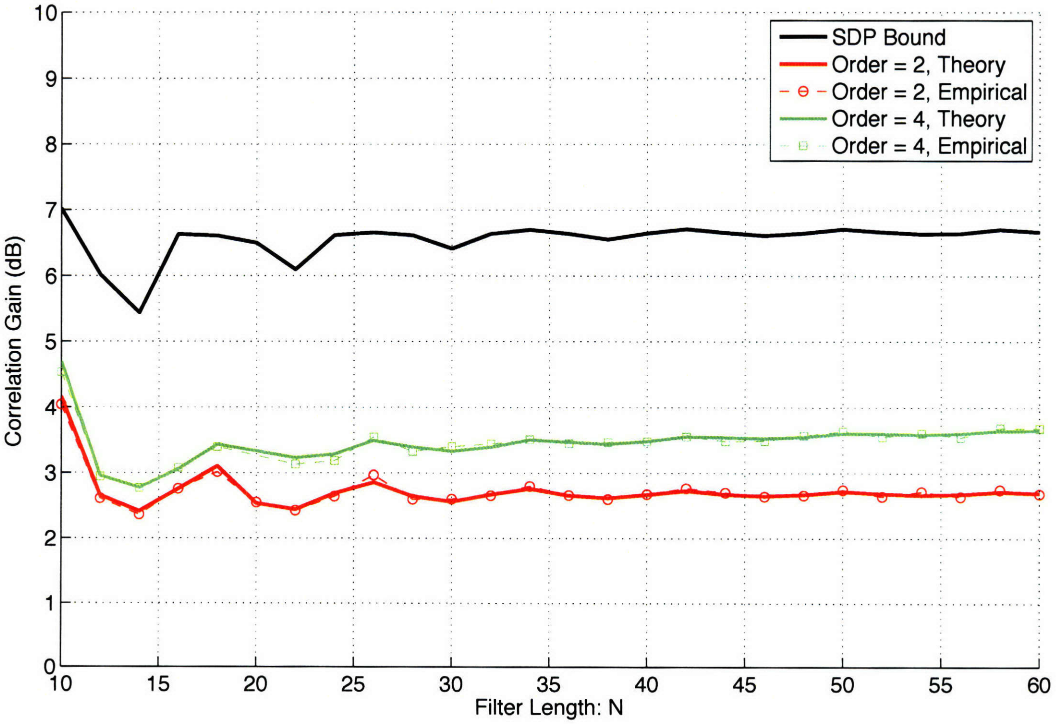


Figure 6-15. Correlation gain, $\mathcal{G}_{\text{tcm}}$, as a function of N for the Parks-McClellan filter with specifications given in Eqn.(6.83) and input spectrum given in Section 6.4.1.

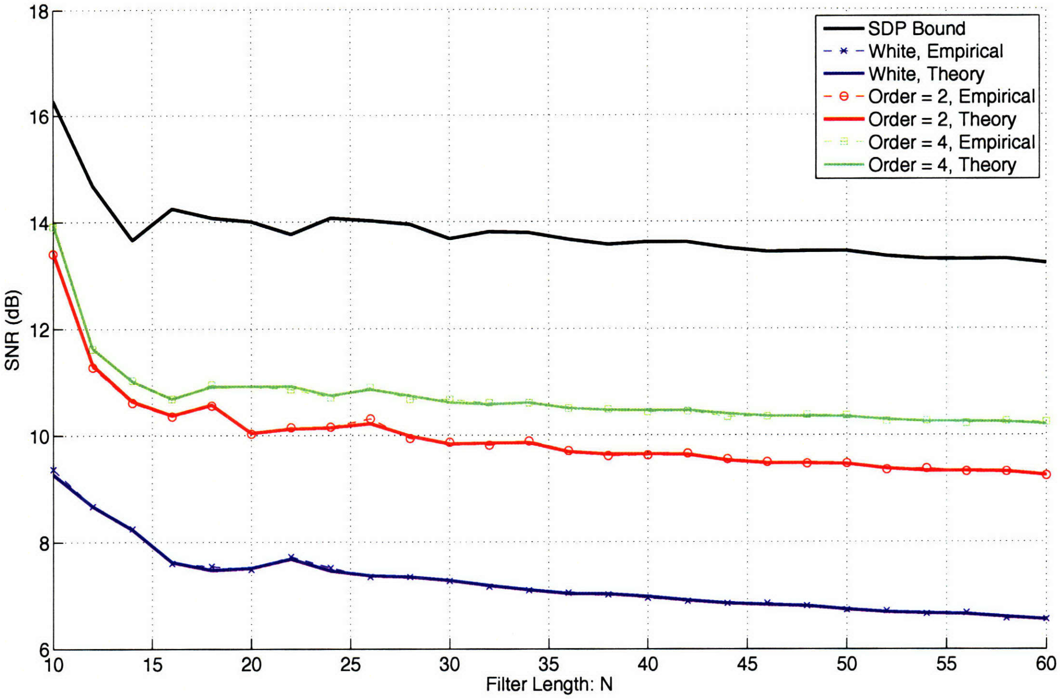


Figure 6-16. SNR scaling as a function of N for the Parks-McClellan filter with specifications given in Eqn.(6.83) and input spectrum given in Section 6.4.1.

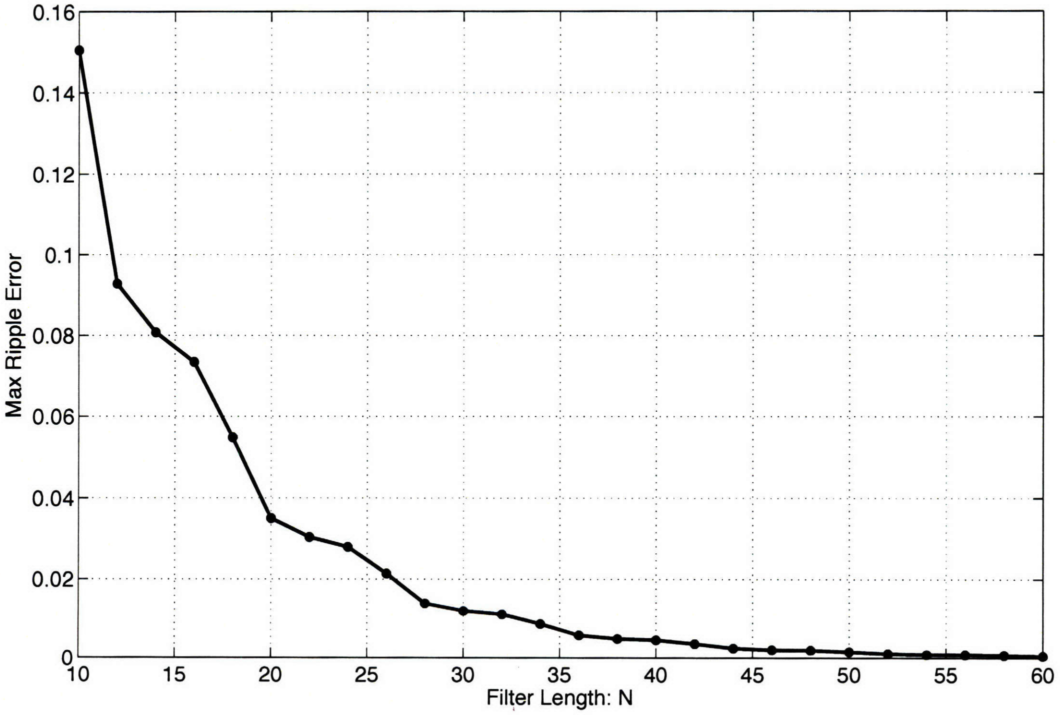


Figure 6-17. Max ripple error as a function of N for Parks-McClellan filters designed with specification Eqn.(6.83).

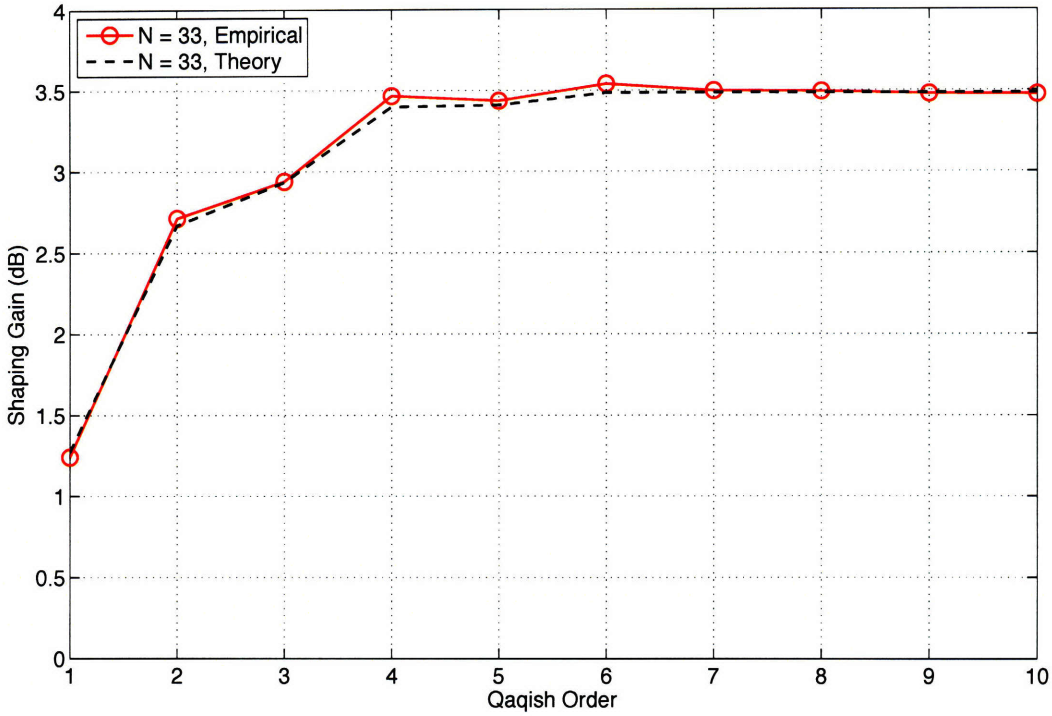


Figure 6-18. Correlation gain, $\mathcal{G}_{\mathrm{tcm}}$, as a function of Qaqish order, κ, for example of Section 6.4.1.

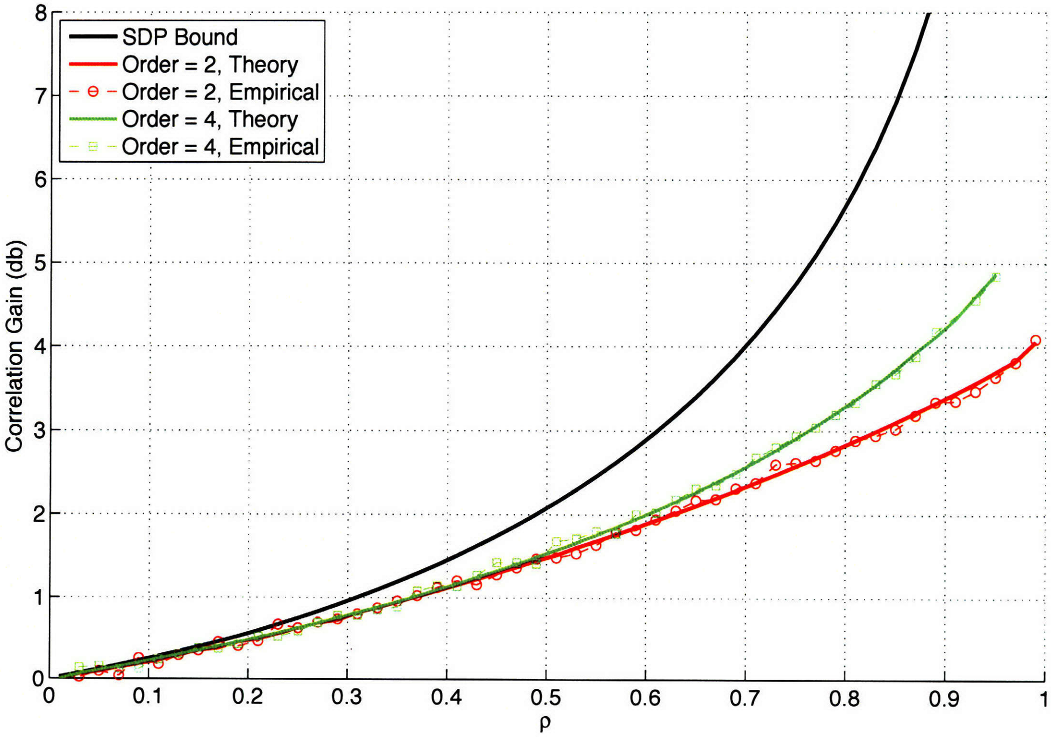


Figure 6-19. Correlation gain, $\mathcal{G}_{\mathrm{tcm}}$, of a AR(1) input process as a function ρ. Desired filter is 33 tap Parks-McClellan example of Section 6.4.1

Chapter 7

Oversampled Direct Form I FIR Binary Randomized Filtering

This chapter develops oversampled Direct Form I FIR BRF, an extension of standard BRF that incorporates rate conversion. Due to an averaging effect, oversampled BRF has a higher SNR that standard BRF. In addition, because of oversampling, the randomization error can be moved out of band to further improve the SNR. Section 7.1 presents the oversampled BRF structure and discusses certain implementation issues. Section 7.2 presents an error analysis of oversampled BRF. Section 7.3 concludes this chapter with a set of numerical experiments to validate the theory.

7.1 Oversampled BRF Model

Section 7.1.1 presents the continuous-valued oversampled Direct Form I FIR filter structure as an alternative implementation of FIR filters. Section 7.1.1 replaces the continuous-valued coefficients in this structure with binary processes to implement oversampled BRF. Section 7.1.3 raises certain implementation issues regarding multiplier-less rate conversion.

7.1.1 Oversampled Direct Form I FIR Structure

Figure 7-1(a) illustrates what we denote the continuous-valued oversampled Direct Form I FIR filter structure. There are three major differences from the standard Direct Form I tapped-delay line structure. First, the tapped-delay line is preceded by an upsampling stage. As illustrated, The input, $x[n]$, is expanded by a factor of L and interpolated with, $G_u(e^{j\omega})$, a LPF with gain L and cutoff π/L.

Secondly, the tapped-delay line is expanded, i.e. the unit delays are replaced with L-element delays. Note that the continuous-valued coefficients b_i are the same as in the standard structure. The expansion compresses the frequency response by a factor of L and introduces periodic copies of the frequency response at integer multiples of π/L. Specifically, defining $B(e^{j\omega})$ as the frequency response of the standard structure, the frequency response of expanded structure is $B(e^{j\omega L})$.

Thirdly, in the oversampled structure, the tapped delay-line is followed by a downsampling stage. As illustrated in Figure 7-1, the output of the tapped delay line is anti-aliased with a unity-gain LPF filter, $G_d(e^{j\omega})$, with cutoff π/L, and then compressed by L. This returns the output, $y[n]$, to the same sampling rate as the input, $x[n]$.

The continuous-valued oversampled structure has the same frequency response as the standard structure, i.e. $B(e^{j\omega})$. This can be shown in a straightforward manner. We present an informal argument here. For analysis purposes, we assume the interpolation and anti-aliasing filters in the rate-conversion stages are ideal, i.e. $G_u(e^{j\omega})$, the interpolation filter, has a frequency response:

$$G_u(e^{j\omega}) = \begin{cases} L & \text{for } |\omega| < \frac{\pi}{L} \\ 0 & \text{otherwise} \end{cases} \tag{7.1}$$

and $G_d(e^{j\omega})$,the anti-aliasing filter, has a frequency response:

$$G_d(e^{j\omega}) = \begin{cases} 1 & \text{for } |\omega| < \frac{\pi}{L} \\ 0 & \text{otherwise} \end{cases} \tag{7.2}$$

Figure 7-1(b) illustrates a block diagram of the oversampled structure. Because the filters are all LTI, we can interchange the order of $B(e^{j\omega L})$ and $G_d(e^{j\omega})$. Using the down-sampling noble identity, the order of the compressor and $B(e^{j\omega L})$ can also be interchanged to achieve the block diagram of Figure 7-1(c). The first four stages in Figure 7-2(b), upsampling by L followed by down-sampling by L, constitute an identity system. This shows that the oversampled structure has the same frequency response, $B(e^{j\omega})$, as the standard one. This, in turn, implies that for a static filter, implementation using the oversampled structure has no effect. In particular, for static multiplier-less filters, the oversampled structure does not remove frequency response distortion. The MSE is constant as a function of L. This is in contrast to randomization where, as shown in this chapter, the MSE decreases as a function of L.

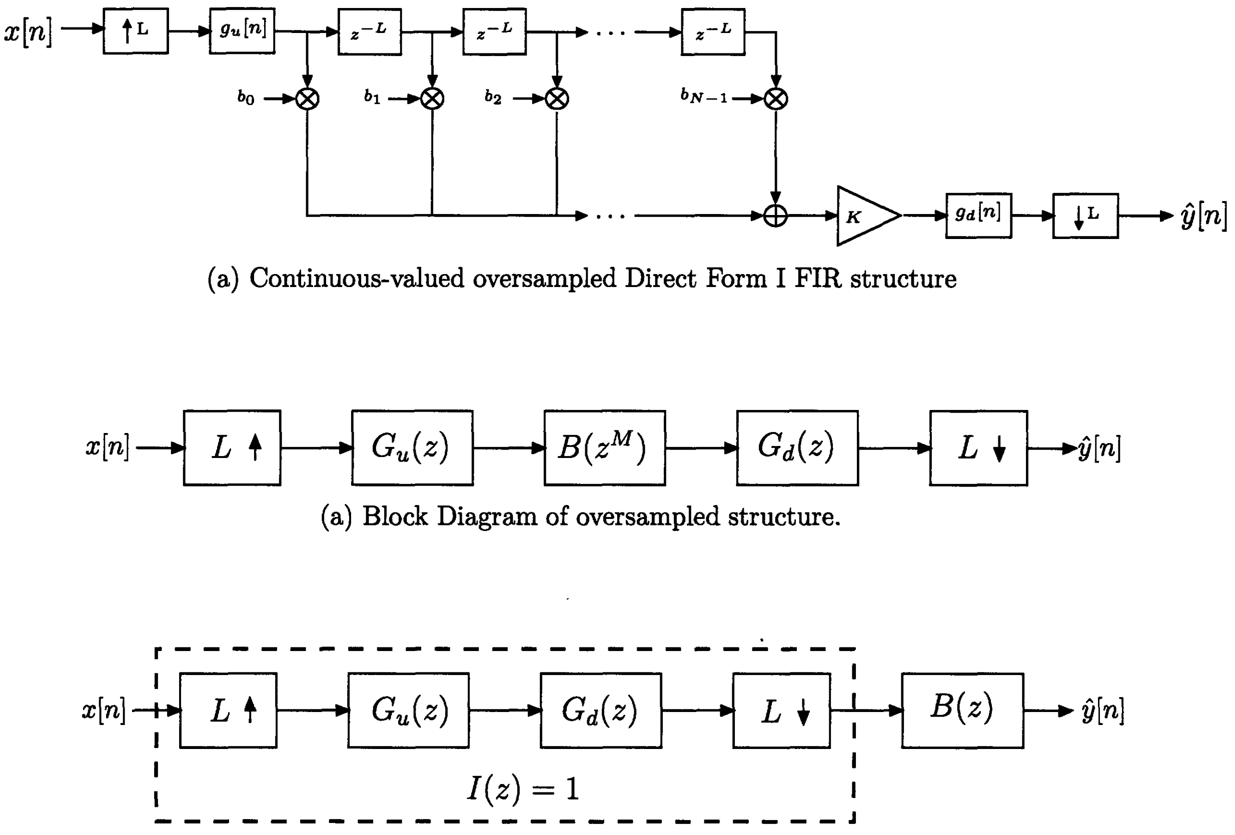
7.1.2 Oversampled Direct Form I FIR BRF

Figure 7-2 illustrates the oversampled Direct Form I FIR BRF structure. The continuous-valued taps in Figure 7-1(a) have been replaced with binary random processes that run at L times the input sample rate. As in standard BRF, the mean of each tap is constrained such that $K\mu_i s_i = b_i$. With this constraint, oversampled BRF (oBRF) on average has the desired continuous-valued response, i.e. the expected frequency response is $B(e^{j\omega})$.

There is an important implementation issue with oBRF: the rate-conversion filters, $g_u[n]$ and $g_d[n]$, in general require multiplies for implementation. Section 7.1.3 discusses this issue in more detail, but in short, this issue is not problematic. Either these filters can be implemented without multiplies or they can be implemented as a fixed filter using an ASIC. Basically, since they are not reconfigurable, like the BRF, they can be viewed as a fixed cost of oBRF implementation.

The oBRF structure still has randomization error, but it is lower than that of standard BRF. In fact, as shown in Section 7.2, oBRF exhibits an L-fold gain in SNR over standard BRF. The precise analysis is presented in Section 7.2. Intuitively, the error is lower due to an averaging effect. Basically, the oBRF can be viewed as the average of L standard BRFs running independently. The averaging reduces the noise variance while keeping the mean unchanged – leading to an increase in SNR.

One important point to note is that the tapped delay-line has N non-zero tap processes, not LN. This is important because, as shown in Chapter 6, the randomization error scales with the number of non-zero tap processes. By fixing the number of taps to N, the total



(a) Continuous-valued oversampled Direct Form I FIR structure

(a) Block Diagram of oversampled structure.

(b) Modified Block Diagram of oversampled structure.

Figure 7-1. Diagrams for continuous-valued oversampled Direct Form I FIR filtering. The rate-conversion filters are assumed to be ideal and the coefficients b_i are assumed to be continuous-valued. $B(e^{j\omega})$ is the frequency response of the standard structure without oversampling.

power of the randomization error is fixed independent of the rate L. As discussed in Section 7.2, this is a critical element to achieving any oversampling SNR gain.

As with standard BRF, there are four types of oBRF depending on the correlation of the tap processes. Memoryless oBRF, both tap-independent and tap-correlated, are discussed in detail in Section 7.2.2. Frequency-shaping, using time-correlation of the tap processes, is useful in oBRF. This is because the anti-aliasing filter, $G_d(e^{j\omega})$, can be used to remove the error shaped into its stop-band. Section 7.2.3 discusses the potential of frequency-shaped oBRF.

7.1.3 Multiplier-less Rate Conversion

As mentioned in the previous section, there is an implementation issue with oversampled BRF: the rate-conversion filters, $g_u[n]$ and $g_d[n]$, require multiplies for implementation. These filters are fixed though, so they can be implemented and optimized for speed in an ASIC. The resulting implementation can be viewed as a fixed cost of the oBRF structure.

In other cases, we may want to implement the rate-conversion without multiplies too. There is a class of specialized multiplier-less filters, called cascaded-integrator-combs (CICs), designed for this purpose, [16, 20]. As mentioned in Chapter 5, CICs are static multiplier-

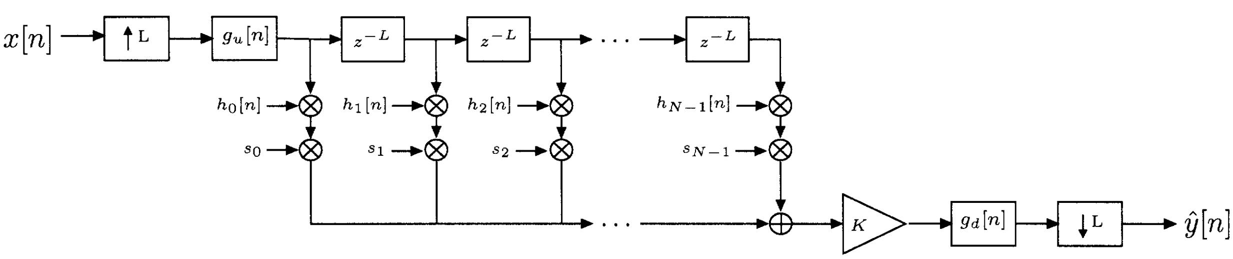


Figure 7-2. Block Diagram of Oversampled Direct Form I FIR BRF

less filters composed of a cascade of single pole integrator IIR stages, $H_I(z) = 1/(1-z^{-1})$, with single zero FIR comb stages, $H_C(z) = 1 - \sum_{i=0}^{N} z^{-iL}$. These CIC filters have been studied in detail in the literature, [16, 20]. They can handle arbitrary and large rate changes and are well suited for hardware implementation.

Though suitable, CIC filters, like all static multiplier-less filters have frequency distortion. The most problematic distortion caused by CIC filters is a type of passband attenuation called passband "droop", [16, 20]. The randomized tapped delay-line can equalize the passband "droop", i.e. changing the desired response of the oBRF can compensate for the droop from the CIC interpolation and anti-aliasing filters. In short, implementing $g_u[n]$ and $g_d[n]$ using multiplier-less CIC filters does not pose a significant design obstacle. It is a mature technique which can be adapted for use in oversampled BRF.

In this chapter, for simplicity, we assume that the rate-conversion filters are implemented with multiplies. This simplifies the analysis but the computed gains are then overestimates relative to what is possible using CIC filters. The ideal analysis can be interpreted as an upper-bound on performance using CICs. With a properly designed CIC filters, the error will be close to this bound.

7.2 Error Analysis

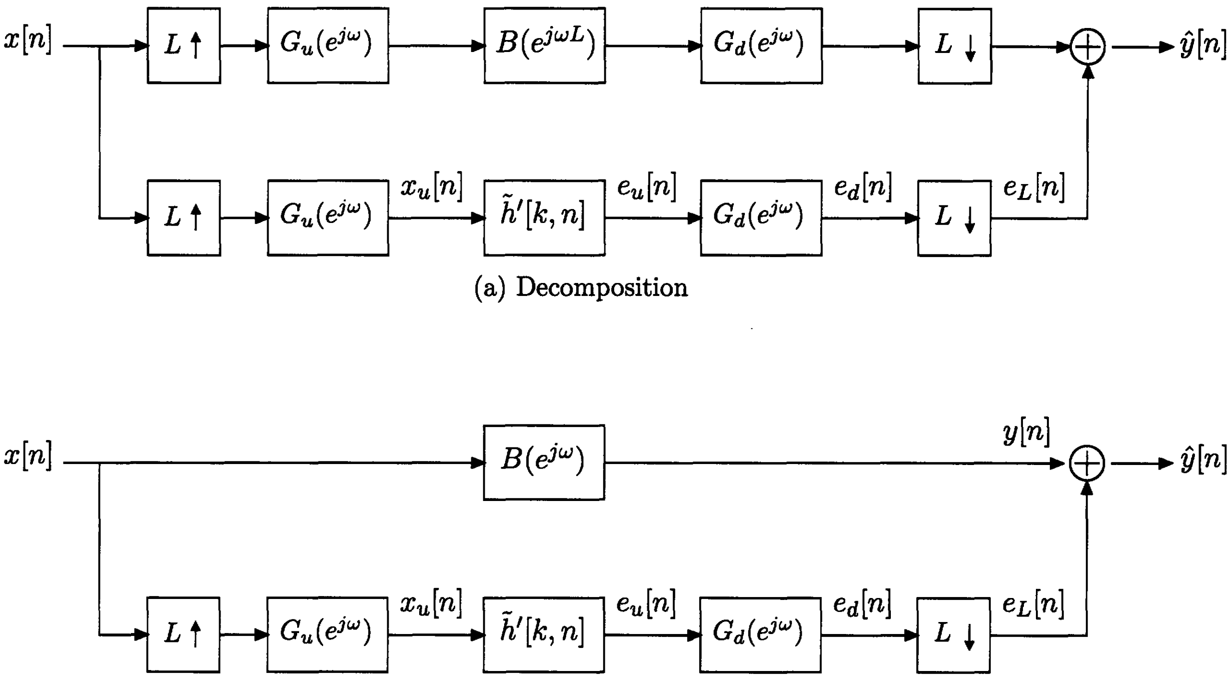
This section presents an error analysis for oversampled BRF. As with standard BRF, the oBRF error is shown to be uncorrelated with the input and shapeable. Section 7.2.2 presents the analysis of memoryless oBRF. Oversampled BRF is shown to have a L-fold SNR gain over standard BRF. Section 7.2.3 discuses potential benefits of frequency-shaping in oBRF and presents preliminary design techniques for it.

7.2.1 General Analysis

Define b'_k as the NL continuous-valued tap coefficients in the oversampled Direct Form I structure. These coefficients have the values:

$$b'_k = \begin{cases} b_i & , \ k = iL \text{ for } i = 0, \dots, N-1 \\ 0 & , \ \text{otherwise} \end{cases} \tag{7.3}$$

where the b_i are the non-zero taps in Figure 7-1(a) which have the same values as the N taps of a standard Direct Form I structure. As mentioned earlier, the frequency response



(a) Decomposition

(b) Modified decomposition

Figure 7-3. Error analysis block diagrams for oversampled Direct Form I BRF

of the expanded tapped delay-line in oversampled structure is $B'(e^{j\omega}) = B(e^{j\omega L})$, where $B(e^{j\omega})$ is the frequency response of the standard structure.

Define $h'_k[n]$ as the tap-processes in the expanded tapped delay-line for oversampled BRF. These processes can be expressed as:

$$h'_k[n] = \begin{cases} h_i[n] & , \ k = iL \text{ for } i = 0, \dots, N-1 \\ 0 & , \ \text{otherwise} \end{cases} \tag{7.4}$$

where the $h_i[n]$ are the non-zero binary processes in Figure 7-2. Define the sign changes similarly using s'_k. As noted in Section 7.1.2, the non-zero tap processes have the same mean constraint, $E\{h_i[n]\} = \mu_i = s_i b_i / K$. In the expanded tapped delay-line the means, μ'_k, can be expressed as,

$$\mu'_k = \begin{cases} \mu_i & , \ k = iL \text{ for } i = 0, \dots, N-1 \\ 0 & , \ \text{otherwise} \end{cases} \tag{7.5}$$

The output of the expanded tapped delay-line before downsampling, $\hat{y}_u[n]$, can be defined as:

$$\hat{y}_u[n] = \sum_{k=0}^{L(N-1)} h'_k[n] s'_k x_u[n-k] \tag{7.6}$$

where $x_u[n]$ is the upsampled input. Similar to our randomized sampling analysis, we can use the decomposition $h'_k[n] = \mu'_k + \tilde{h}'_k[n]$, to express the output as the sum of a static portion, from the constant mean, and an error term from the zero-mean time-varying filter

$\tilde{h}'_k[n]$:

$$\begin{aligned}\hat{y}_u[n] &= \sum_{k=0}^{L(N-1)} \underbrace{K\mu'_k s'_k}_{b'_k} x_u[n-k] + K \sum_{k=0}^{L(N-1)} \tilde{h}'_k[n] s_i x_u[n-k] \\ &= \sum_{k=0}^{L(N-1)} b'_k x_u[n-k] K \sum_{k=0}^{L(N-1)} \tilde{h}'_k[n] s'_k x_u[n-k] \end{aligned} \tag{7.7}$$

As denoted in Eqn.(7.7), the first term is the LTI convolution of the expanded filter, b'_k, with upsampled input, $x_u[n]$. The second term is the convolution of $x_u[n]$ with the time-varying kernel, $\tilde{h}'[k,n] = \tilde{h}'_k[n]$. The decomposition of Eqn.(7.7) can be represented in block diagram form as Figure 7-3(a). In this figure the rate conversion stages have been distributed into each of the branches. The upper branch can be simplified into a single filter $B(e^{j\omega})$ using the identity from Section 7.1.1. This implies that the output of the upper branch is, $y[n]$, the desired output from the desired continuous-valued filter. Figure 7-3(b) illustrates the block diagram with this simplification.

We denote the output of the lower branch as $e_L[n]$, the error after L-fold oBRF. To characterize this error, we must first characterize the error $e_u[n]$, the output of the zero-mean time-varying filter $\tilde{h}[k,n]$. Using an analogous argument to that in Section 5.2.2, we can show that $e_u[n]$ is a WSS random process that is uncorrelated with the input $x_u[n]$. It follows that the output after downsampling, $e_L[n]$, is uncorrelated with the input $x[n]$.

The auto-correlation of $e_u[n]$ can be expressed as,

$$\begin{aligned} R_{e_u e_u}[m] &= E\{e_u[n]e_u[n+m]\} \\ &= K^2 \sum_{k=0}^{L(N-1)} \sum_{l=0}^{L(N-1)} E\{h'_k[n]h'_l[n+m]\} s'_k s'_l E\{x_u[n-k]x_u[n+m-l]\} \\ &= K^2 \sum_{k=0}^{L(N-1)} \sum_{l=0}^{L(N-1)} E\{h'_k[n]h'_l[n+m]\} s'_k s'_l R_{x_u x_u}[m+k-l] \end{aligned} \tag{7.8}$$

where we have used the independence of $h'_k[n]$ and $x_u[n]$ to separate the expectation. Since $h'_k[n]$ only has N non-zero processes, the cross terms can be simplified such that,

$$E\{h'_k[n]h'_l[n]\} = \begin{cases} E\{h_i[n]h_j[n+m]\} & , k = iL,\ l = jL \text{ for } i,j = 0,\ldots,N-1 \\ 0 & , \text{otherwise} \end{cases} \tag{7.9}$$

Substituting Eqn.(7.9) into Eqn.(7.8) and reindexing the summations, the error auto-correlation can be simplified as,

$$R_{e_u e_u}[m] = K^2 \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} E\{h_i[n]h_j[n+m]\} s_i s_j R_{x_u x_u}[m + L(i-j)] \tag{7.10}$$

For further error analysis we must assume some structure on the tap processes. As mentioned earlier, there are four forms of oBRF depending on the correlation of the tap processes. The next section analyzes memoryless oBRF. Section 7.2.3 analyzes the potential benefits of frequency-shaped oBRF.

7.2.2 Memoryless Oversampled BRF

In memoryless oBRF, the tap processes are restricted to be white in time. This implies the constraint $E\{h_i[n]h_j[n+m]\} = \sigma_{ij}\delta[m]$. Incorporating this constraint into Eqn.(7.10), the upsampled error auto-correlation can be expressed as,

$$R_{e_u e_u}[m] = K^2 \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sigma_{ij} s_i s_j R_{x_u x_u}[L(i-j)]\delta[m] \tag{7.11}$$

We can show that $R_{x_u x_u}[Lm] = R_{xx}[m]$. We present an informal argument here for completeness. The signal $x_u[n]$ is the bandlimited interpolation of $x[n]$ after expansion by L. It is thus a WSS random process. Compressing $x_u[n]$ by L returns the original signal $x[n]$. This follows from zero-ISI property of bandlimited interpolation. Compressing a WSS process without aliasing results in another WSS signal which has an auto-correlation that is the downsampled auto-correlation of the input signal, [25]. It follows that:

$$R_{x_u x_u}[Lm] = R_{xx}[m] \tag{7.12}$$

Substituting Eqn.(7.12) into Eqn.(7.11), the upsampled error auto-correlation can be expressed as,

$$R_{e_u e_u}[m] = K^2 \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sigma_{ij} s_i s_j R_{xx}[i-j]\delta[m] = R_{ee}[m] \tag{7.13}$$

The error is white and equivalent to $R_{ee}[m]$ from Eqn.(5.9), the auto-correlation of standard memoryless BRF error, without oversampling. This is intuitively sensible because in oBRF the error is from the same number of noise sources, with the same means, and same input correlation values. Note that the error is the same only because the tapped delay-line in oBRF has N non-zero taps, rather than LN.

Since the error auto-correlation is the identical, the optimal design of memoryless oBRF is the same as that for standard BRF. The standard BRF solutions can be used directly. Oversampling does not add any new degrees of freedom that can be exploited using memoryless oBRF. In what follows, we assume that memoryless BRF has been optimally designed according to the techniques presented in Chapter 6. We denote the optimum noise floor as $\mathcal{E}^*(\mathbf{b})$. In the frequency domain, the power spectrum of the upsampled error is flat and can be expressed as,

$$S_{e_u e_u}(e^{j\omega}) = S_{ee}(e^{j\omega}) = \mathcal{E}^*(\mathbf{b}) \tag{7.14}$$

The upsampled error is filtered by the anti-aliasing filter $G_d(e^{j\omega})$. The output of this filter, $e_d[n]$, has a power spectrum,

$$S_{e_d e_d}(e^{j\omega}) = S_{ee}(e^{j\omega})|G_d(e^{j\omega})|^2 = \begin{cases} \mathcal{E}^*(\mathbf{b}) & , |\omega| \leq \pi/L \\ 0 & , \text{otherwise} \end{cases} \tag{7.15}$$

As noted, the anti-aliasing filter removes the energy of $S_{ee}(e^{j\omega})$ in the stop-band. It is this noise reduction that results in oversampling gain. For memoryless BRF the filter removes all but $1/L$-th the noise power. With frequency-shaping BRF we could potentially make this ratio lower by frequency shaping $S_{ee}(e^{j\omega})$ so that there is less energy in passband of $G_d(e^{j\omega})$. This is discussed in the next section.

For memoryless BRF, after anti-aliasing the signal $e_d[n]$ is compressed by a factor of L. This stretches the power spectrum $S_{e_d e_d}(e^{j\omega})$ and scales it down by factor of L. The output error power spectrum can be expressed as:

$$S_{e_L e_L}(e^{j\omega}) = \frac{1}{L} S_{e_d e_d}(e^{j\omega/L}) = \frac{\mathcal{E}^*(\mathbf{b})}{L} \text{ for} |\omega| < \pi \tag{7.16}$$

The noise floor is scaled down by a factor of L. Consequently, the oBRF output MSE, denoted $\mathcal{E}_L^*(\mathbf{b})$, is $1/L$-th the power of the standard BRF MSE, $\mathcal{E}^*(\mathbf{b})$:

$$\mathcal{E}_L^*(\mathbf{b}) = \frac{\mathcal{E}^*(\mathbf{b})}{L} \tag{7.17}$$

On the other hand, the power of the desired output signal, $y[n]$, from the upper branch of Figure 7-3(b) is unchanged. Accordingly, the oBRF SNR in is L times higher than that of standard BRF:

$$\text{SNR}_L = \frac{E\{y^2[n]\}}{\mathcal{E}_L^*(\mathbf{b})} = L \frac{E\{y^2[n]\}}{\mathcal{E}^*(\mathbf{b})} = L \cdot \text{SNR} \tag{7.18}$$

Intuitively, with oversampling the number of non-zero taps remains N so the same total noise power is spread over L replications of $S_{xx}(e^{j\omega})$ with the filtering then reducing the noise power total. Both forms of memoryless BRF, timBRF and tcmBRF, exhibit this L-fold oversampling SNR gain. Note that, unlike correlation gain, there is no limit on oversampling gain. It can be used to arbitrarily improve the output SNR. In practice though, the hardware constraints will impose a limit because increasing L increases the memory requirements of the implementation, i.e. the tapped delay-line must be longer.

In addition, with a multiplier-less CIC implementation of the rate-conversion filters the full L-fold gain is probably not achievable. The gain will still be on the order L though. We conjecture that the performance loss from CIC implementation will be small. Section 7.3 illustrates the results of memoryless oBRF for two examples. The empirical results match well with the theory developed in this section.

7.2.3 Frequency-Shaping BRF

As mentioned in the previous section, frequency-shaping BRF could have benefits in the oversampled BRF structure. In particular, the error spectrum can be shaped so that the error in the passband of $G_d(e^{j\omega})$ is reduced. Frequency-shaped BRF requires time correlation in the binary tap processes $h_i[n]$. There are two types of frequency-shaping BRF: tap-independent frequency-shaping (tifsBRF), with each tap process is correlated in time but independent of one another, and tap-correlated frequency-shaping (tcfsBRF), where the tap processes can be correlated in time and across the taps. We briefly discuss the potential benefits and possible design issues of both of these techniques in this section. The ideas presented are preliminary and not developed in detail.

Tap-Independent Frequency Shaped BRF

In tap-independent frequency-shaping BRF (tifsBRF) each tap is uncorrelated with one another but they can be correlated in time. This implies:

$$E\{\tilde{h}_i[n]\tilde{h}_j[n+m]\} = \begin{cases} E\{\tilde{h}_i[n]\tilde{h}_i[n+m]\} = K_{h_ih_i}[m] & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \tag{7.19}$$

Substituting this into Eqn.(7.10), the auto-correlation of the upsampled error can be expressed as:

$$R_{e_ue_u}[m] = K^2 \sum_{i=0}^{N-1} K_{h_ih_i}[m] R_{x_ux_u}[m] \tag{7.20}$$

In the frequency domain the power-spectrum can be expressed as:

$$S_{e_ue_u}(e^{j\omega}) = K^2 \sum_{i=0}^{N-1} \left(\int_{-\pi}^{\pi} \Phi_{h_ih_i}(e^{j\theta}) S_{x_ux_u}(e^{j(\omega-\theta)}) \frac{d\theta}{2\pi} \right) \tag{7.21}$$

After anti-aliasing with the filter $G_d(e^{j\omega})$, the power spectrum of the error, $e_d[n]$, can be expressed as:

$$\begin{aligned} S_{e_de_d}(e^{j\omega}) &= |G_d(e^{j\omega})|^2 S_{e_ue_u}(e^{j\omega}) && (7.22) \\ &= K^2 \left(\sum_{i=0}^{N-1} |G_d(e^{j\omega})|^2 \int_{-\pi}^{\pi} \Phi_{h_ih_i}(e^{j\theta}) S_{x_ux_u}(e^{j(\omega-\theta)}) \frac{d\theta}{2\pi} \right) && (7.23) \end{aligned}$$

The output MSE can be expressed as,

$$\begin{aligned} \mathcal{E}_L &= \int_{-\pi}^{\pi} S_{e_de_d}(e^{j\omega}) \frac{d\omega}{2\pi} && (7.24) \\ &= K^2 \sum_{i=0}^{N-1} \left(\int_{\omega=-\pi}^{\pi} \int_{\theta=-\pi}^{\pi} \left(\Phi_{h_ih_i}(e^{j\theta}) S_{x_ux_u}(e^{j(\omega-\theta)}) \frac{d\theta}{2\pi} \right) |G_d(e^{j\omega})|^2 \frac{d\omega}{2\pi} \right) && (7.25) \\ &= K^2 \sum_{i=0}^{N-1} \left(\int_{\theta=-\pi}^{\pi} \Phi_{h_ih_i}(e^{j\theta}) \underbrace{\left(\int_{\omega=-\pi}^{\pi} S_{x_ux_u}(e^{j(\omega-\theta)} |G_d(e^{j\omega})|^2 \frac{d\omega}{2\pi} \right)}_{F(e^{j\theta})} \frac{d\theta}{2\pi} \right) && (7.26) \\ &= K^2 \sum_{i=0}^{N-1} \left(\int_{\theta=-\pi}^{\pi} \Phi_{h_ih_i}(e^{j\theta}) F(e^{j\theta}) \frac{d\theta}{2\pi} \right) && (7.27) \\ &= K^2 \sum_{i=0}^{N-1} \mathcal{E}_i && (7.28) \end{aligned}$$

Because of tap-independence, the design of each tap process can be considered independently. For each, the goal is to design $h_i[n]$, a binary process with fixed mean μ_i, such that the MSE, $\mathcal{E}_i$, through the filter $G_d(e^{j\omega})$ is minimized. The design problem is exactly the same as that of frequency-shaped SRS. In fact, tifsBRF design can be decomposed into N

independent frequency-shaped SRS design problems for each tap.

The optimization for each tap process in tifsBRF is even simpler than frequency-shaped SRS because it does not require precise knowledge of the power spectrum $S_{x_u x_u}(e^{j\omega})$. In particular, since $S_{x_u x_u}(e^{j\omega})$ is bandlimited to π/L and $G_d(e^{j\omega})$ is a bandlimiting filter to π/L, the function $F(e^{j\omega})$ has support only on $[-2\pi/L, 2\pi/L]$. Consequently, the optimal design of $\Phi_{h_i h_i}(e^{j\theta})$ is to place most of the energy near π, independent of the exact shape of $S_{x_u x_u}(e^{j\omega})$. Tap-independent frequency-shaping BRF is not implemented in this thesis. We conjecture that tifsBRF will have a large oversampling SNR gain for small L and diminishing returns as L increases. Asymptotically, for large L, the tifsBRF SNR should approach the timBRF oversampling gain.

Tap-Correlated Frequency Shaped BRF

Tap-correlated frequency-shaping BRF (tcfsBRF) has the most degrees of freedom so it can achieve the best oversampling gain. However, the many degrees of freedom make tcfsBRF difficult to design. As with the other correlated techniques discussed in this thesis, the fundamental issue is characterization of the set of binary vectors that can be correlated both in time and across taps. Similar to frequency-shaped SRS and tcmBRF, we propose the use of a parametric model.

Our parametric model is inspired by a vector extension of scalar Boufounos processes [5]. Preliminary analysis shows that this extension can be used to generate vector AR binary processes. We present the basic model here, but omit the proofs. Future work will prove and resolve certain issues about this model.

In the proposed model, the vector AR binary process, $\mathbf{h}[n]$, is generated iteratively from p previous values, $\mathbf{h}[n-1], \ldots \mathbf{h}[n-p]$, as follows:

1. The bias vector $\mathbf{h_b}[n]$ for the generation of $\mathbf{h}[n]$ is computed according to the relationship:

$$\mathbf{h_b}[n] = \boldsymbol{\mu} + \sum_{i=1}^{p} \mathbf{A_k}(\mathbf{h}[n-k] - \boldsymbol{\mu}) \tag{7.29}$$

   where $\boldsymbol{\mu}$ is a vector of desired means and the $\mathbf{A_k}$ are matrices that are parameters of the algorithm.

2. The i-th element, $h^i[n]$, of the vector $\mathbf{h}[n]$ is randomly generated, independently of the other indices, from a binary distribution biased by $h_b^i[n]$, the i-th element of $\mathbf{h_b}[n]$, as follows:

$$h^i[n] = \begin{cases} 1 & \text{with probability } h_b^i[n] \\ 0 & \text{with probability } 1 - h_b^i[n] \end{cases} \tag{7.30}$$

In steady-state it can be shown that this vector process is wide-sense stationary with mean $\boldsymbol{\mu}$. The matrix power spectrum can be shown to be,

$$\mathbf{S_{hh}}(\omega) = \mathbf{H}(\omega)^{\mathrm{T}}\mathbf{D}\mathbf{H}(\omega) \tag{7.31}$$

where $\mathbf{D}$ is a diagonal matrix which scales the diagonal elements of $\mathbf{S_{hh}}(\omega)$ such that the fixed mean constraint is satisfied. $\mathbf{H}(\omega)$ is a matrix transfer function,

$$\mathbf{H}(\omega) = \left(\mathbf{I} - \sum_{i=1}^{p} \mathbf{A_k} e^{-j\omega k} \right)^{-1} \tag{7.32}$$

As in the scalar case, there are constraints on the matrices $\mathbf{A_k}$ to ensure that the bias generated by Eqn.(7.29) does not overflow. Only if these constraints are satisfied will the resulting matrix power spectrum be given by Eqn.(7.32). These constraints still remain to be determined. Once all these constraints have been derived, the tcfsBRF design problem can be posed over the parameters $\mathbf{A}_i$ of this model. The ensuing constrained optimization is likely non-trivial and difficult to solve.

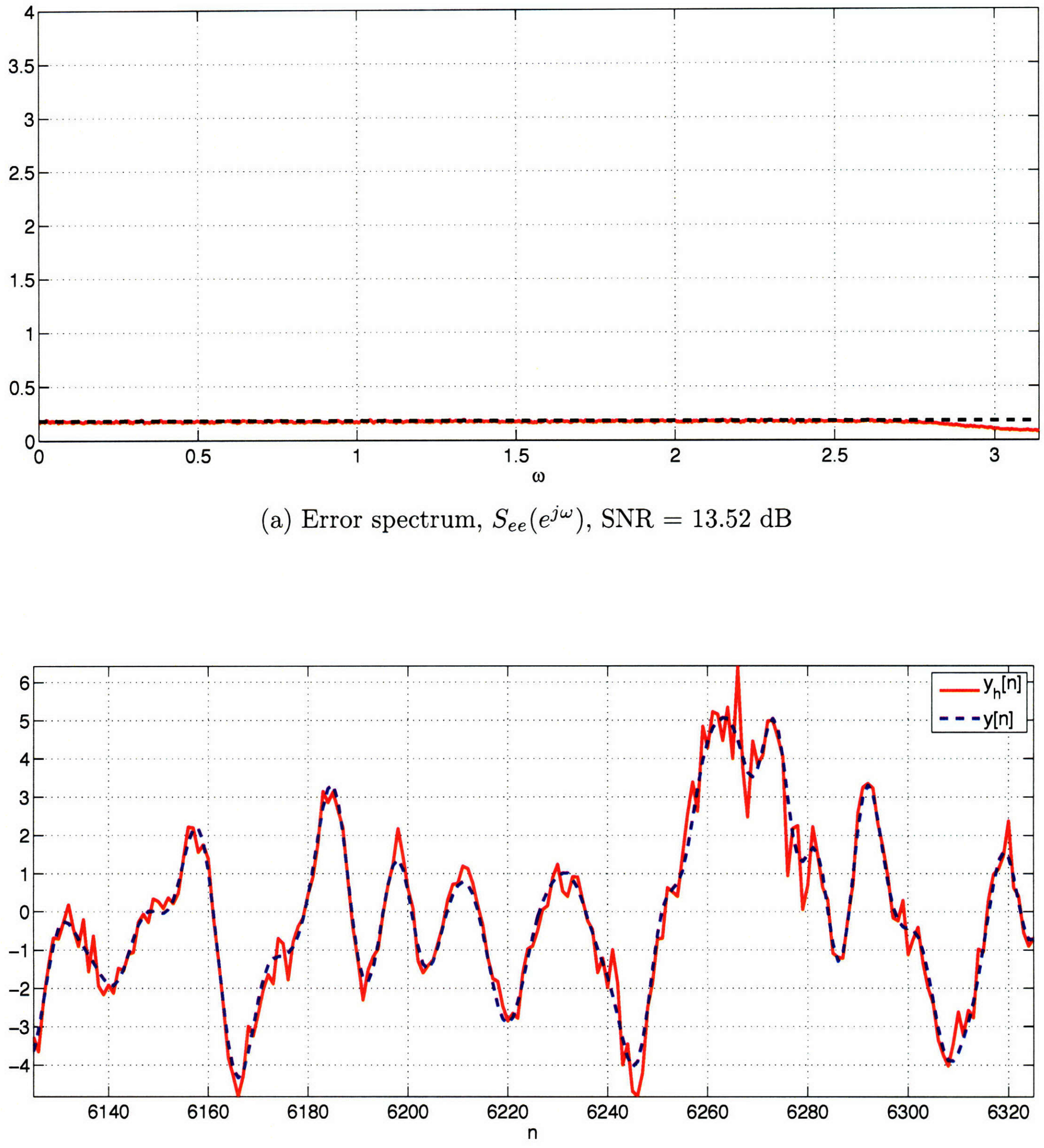
7.3 Numerical Experiments

In this section, we illustrate the performance of oversampled BRF using the same example as in Section 6.4.1 of Chapter 6. The input power spectrum $S_{xx}(e^{j\omega})$ is as given in Eqn.(6.82) and illustrated in Figure 6-10(a). The desired continuous-valued filter is the 33-tap Parks-McClellan low-pass filter with specifications given in Eqn.(6.83) and illustrated in Figure 6-9. We present two examples, one using timBRF the other tcmBRF. The rate-conversion filters are implemented using multiplies for simplicity.

The first example, illustrated in Figure 7-4 is the result of tap-independent memoryless BRF coupled with $L = 4$ times oversampling. As expected the error spectrum is still white, but the noise floor has been reduced due to oversampling gain. There is a small amount of distortion in the error spectrum near $\omega = \pi$ due to the use of non-ideal rate-conversion filters. In the time-domain the output of the oversampled BRF, $\hat{y}[n]$, more closely follows the desired output, $y[n]$. The SNR is 13.52 dB. Compare these results to Figure 6-11 in Chapter 6 with $L = 1$. There is an oversampling gain of 6.34 dB.

The second example, illustrated in Figure 7-5 is the result of tap-correlated BRF with $\kappa = 4$ coupled with $L = 4$ times oversampling. The error spectrum is still white, but the noise floor has been reduced due to correlation gain and oversampling gain. Again, there is a small amount of distortion in the error spectrum due to the use of non-ideal rate-conversion filters. In the time-domain, $\hat{y}[n]$, follows the desired output, $y[n]$, very closely. The SNR is 16.90 dB, with a correlation gain of 3.40 dB plus oversampling gain of 6.32 dB. Compare this to the results of Figure 6-12 in Chapter 6 with $M = 1$.

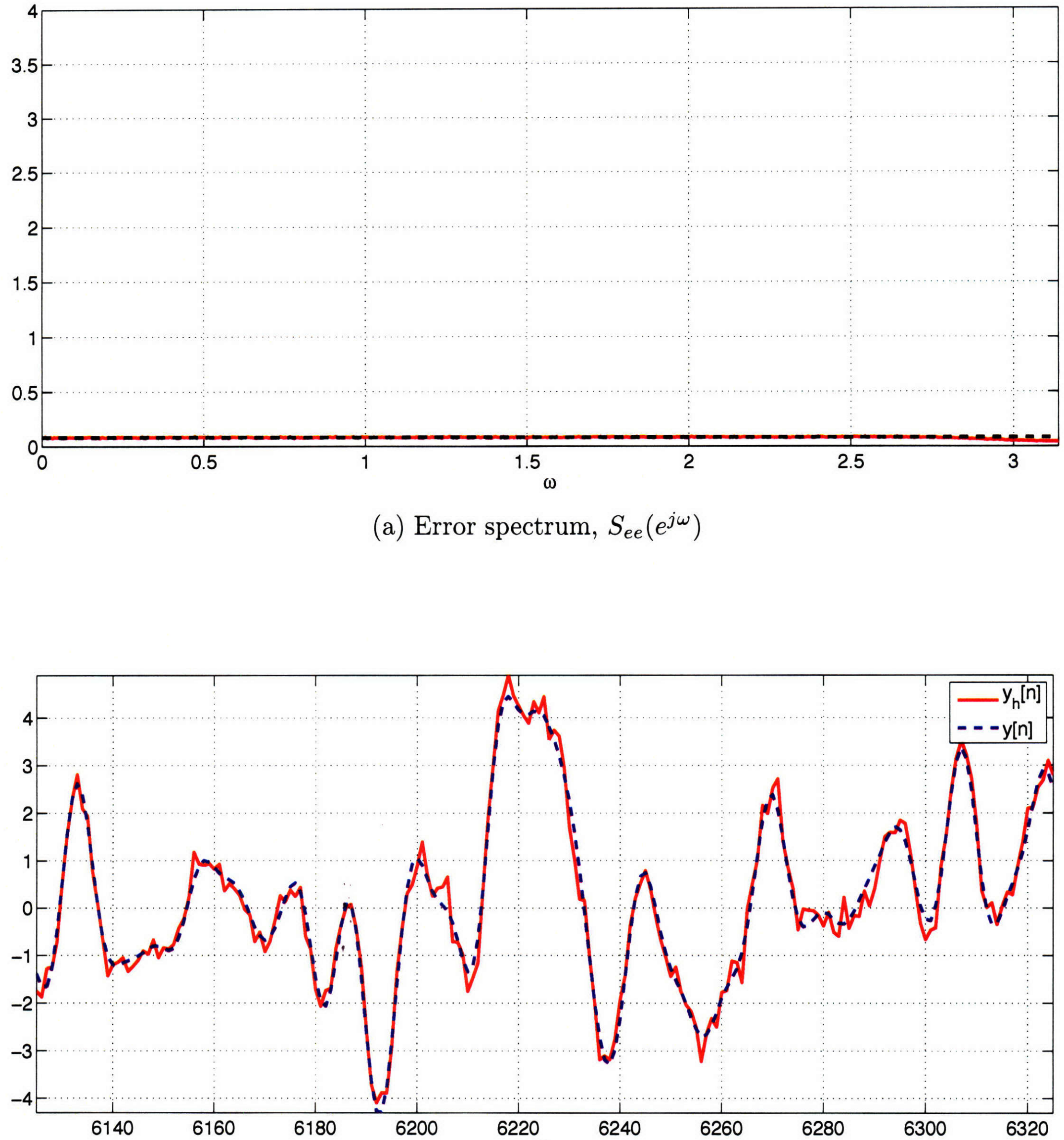
Figure 7-6 illustrates the SNR as function of oversampling rate, L, for this example filter and input. As expected, the SNR grows linearly in L and as $10 \log_1 0L$ on the dB plot. Figure 7-6 also illustrates the MSE of a static multiplier-less filter. As expected, the static multiplier-less MSE is a constant function of L. This contrast illustrates the benefits of randomization over a static multiplier-less structure. With oversampling we can achieve arbitrarily high SNR with enough oversampling. By contrast, with a static implementation we are limited to a fixed SNR.



(a) Error spectrum, $S_{ee}(e^{j\omega})$, SNR = 13.52 dB

(b) Time-Domain output $\hat{y}[n]$ and $y[n]$

Figure 7-4. Error power spectrum and time-domain output for oversampled tap-independent memoryless BRF implementation of the example in Section 6.4.1. Oversampling rate $L = 4$.



(a) Error spectrum, $S_{ee}(e^{j\omega})$

(b) Time-Domain output $\hat{y}[n]$ and $y[n]$

Figure 7-5. Error power spectrum and time-domain output for oversampled tap-correlated memoryless BRF implementation of the example in Section 6.4.1. Oversampling rate $L = 4$ and Qaqish order $\kappa = 4$.

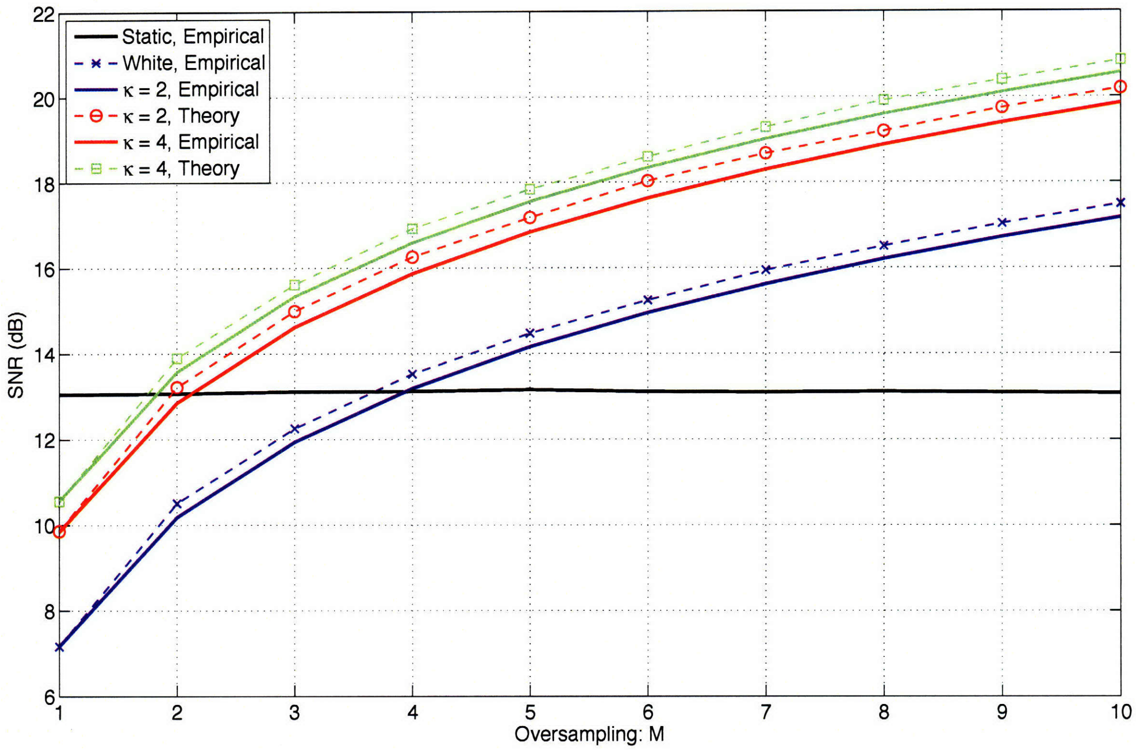


Figure 7-6. SNR of memoryless BRF as a function of oversampling rate L for example of Section 6.4.1.

Bibliography

[1] Sheldon Axler. *Linear Algebra Done Right.* Springer-Verlag New York, Inc., 1997.

[2] Dmitri P. Bertsekas. *Nonlinear Programming.* Athena Scientific, 1999.

[3] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, 1997.

[4] I. Bilinkis and A. Mikelsons. *Randomized Signal Processing.* Prentice-Hall, 1992.

[5] P. Boufounos. Generating binary processes with autoregressive spectra. In *Proceedings of ICASSP'07*, May 2007.

[6] Petros Boufounos, Alan V. Oppenheim, and Vivek K. Goyal. Causal compensation for erasures in frame representations. *IEEE Transactions on Signal Processing*, Accepted for publication.

[7] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization.* Cambridge University Press, 2006.

[8] Emmanuel Candes. Compressive sampling. In *Proceedings of the International Congress of Mathematicians*, 2006.

[9] N. Rao Chaganty and Harry Joe. Range of correlation matrices for dependent Bernoulli random variables. *Biometrika*, 93(1):197–206, 2006.

[10] R. L. Cook. Stochastic sampling in computer graphics. *ACM Transactions on Graphics*, 5(1):51–72, Jan 1986.

[11] J.L. Martins de Carvalho and J.M.C. Clark. Characterizing the autocorrelations of binary sequences. *IEEE Transactions on Information Theory*, IT-29(4), July 1983.

[12] Sourav R. Dey, Andrew I. Russell, and Alan V. Oppenheim. Pre-compensation for anticipated erasures in LTI interpolation systems. *IEEE Transactions on Signal Processing*, Jan 2006.

[13] M. Dippe and E. H. Wold. Antialiasing through stochastic sampling. In *Proceedings of SIGGRAPH'85*, pages 69–78. SIGGRAPH, July 1985.

[14] G. Pierobon G. Bilardi, R. Padovani. Spectral analysis of functions of markov chains with applications. *IEEE Transactions on Communications*, COM-31(7), July 1983.

[15] P. Galko and S. Pasupathy. The mean power spectral density of Markov chain driven signals. *IEEE Transactions on Information Theory*, IT-27(6), November 1981.

[16] E. B. Hogenauer. An economical class of digital filters for decimation and interpolation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(2):155–162, 1981.

[17] N. S. Jayant and P. Noll. *Digital Coding of Waveforms.* Prentice-Hall, Inc., 1984.

[18] R. H. Tutuncu K. C. Toh and M. J. Todd. On the implementation and usage of sdpt3 - a matlab sofware package for semidfinite-quadratic-linear programming, version 4.0. Technical report, National University of Singapore, 2006.

[19] K.X. Karakostas and H.P. Wynn. On the covariance function of stationary binary sequences with given mean. *IEEE Transactions on Information Theory*, 39(5), Sept 1993.

[20] Alan Y. Kwentus, Zhongnong Jiang, and Jr. Alan N. Wilson. Application of filter sharpening to cascaded integrator-comb decimation filters. *IEEE Transactions on Signal Processing*, 45(2), 1997.

[21] Marcos Martinex-Peiro, Eduardo I. Boemo, and Lars Wanhammar. Design of high-speed multiplierless filters using a nonrecursive signed common subexpression algorithm. *IEEE Transactions on Circuits and Systems – II: Analog and Digital Signal Processing*, 49(3), March 2002.

[22] Farokh Marvasti, editor. *Nonuniform Sampling: Theory and Practice.* Kluwer Academic/Plenum Publishers, 2001.

[23] D. P. Mitchell. Generating antialiased images at low sampling densities. *Computer Graphics*, 21(4):65–72, July 1987.

[24] Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-Time Signal Processing.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1999.

[25] Alan V. Oppenheim and George C. Verghese. Signals, systems, and inference. Class notes for 6.011: Introduction to Communication, Control, and Signal Processing, 2007.

[26] Athanasios Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, 1991.

[27] Bahjat F. Qaqish. A family of multivariate binary distributions for simulating correlated binary variables with specified marginal means and correlations. *Biometrika*, 90(2):455–463, 2003.

[28] M. Said and A.V. Oppenheim. Discrete-time randomized sampling. In *Proceedings of ICECS'01*. ICECS, Sept 2001.

[29] Maya R. Said. Discrete-time randomized sampling. Master's thesis, Massachusetts Institute of Technology, 2001.

[30] Henry Stark and Yongyi Yang. *Vector Space Projections.* John Wiley and Sons, Inc, 1998.

[31] J. Tuqan and P. P. Vaidyanathan. Statistically optimum pre- and postfiltering in quantization. *IEEE Transactions on Circuits and Systems – II: Analog and Digital Signal Processing*, 44(1), December 1997.