Reconstruction of Functions From Non-uniformly Distributed Sampled Data in Shift-Invariant Frame Subspaces

Presented by:
Mkhuseli Bruce Mkhaliphi

Prepared for:
Professor Alireza Baghai-Wadji
Department of Electrical and Computer Engineering
University of Cape Town

Submitted to the Department of Electrical and Computer Engineering at the University of Cape Town in partial fulfilment of the academic requirements for a Masters of Science degree in Electrical Engineering.

February, 2018

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.

Signature: Signed by candidate

M. B. Mkhaliphi

Date: 29.10.2018

Supervisor

As the candidate's supervisor, I have approved this dissertation for submission.

Signature: Signed by candidate

Prof. Alireza Baghai-Wadji

Date:

Acknowledgements

This research was possible through the supervision and mentorship of Prof. Alireza Baghai-Wadji and additional support from peer researchers, Lateef Akinyemi and Wasim Lorgaat in the Electronics and Computational Engineering Group (ECEG). I am grateful for receiving The Ledger Grant, a prestigious bursary, granted by ARMSCOR and CSIR.

I am grateful for Prof. Alireza's encouragement to join the MSc program, and for him taking me on as a student. I particularly enjoyed our research discussions, and the atmosphere of freedom which characterized them. His enthusiasm and approach to problem solving since my undergraduate has given me confidence to seek and tackle challenging problems. His methods and perspective, although difficult to replicate at his level of depth, has served me, outside academics, to identify and seize business opportunities.

What I took from him, mostly, was the mindset of not being a passive consumer of knowledge. Now, I have the urge to contribute to the legacy of societal advancement through scientific work.

Thank you!

Abstract

The focus of this research is to study and implement efficient iterative reconstruction algorithms. Iterative reconstruction algorithms are used to reconstruct bandlimited signals in shift-invariant L2 subspaces from a set of non-uniformly distributed sampled data. The Shannon-Whittaker reconstruction formula commonly used in uniform sampling problems is insufficient in reconstructing function from non-uniformly distributed sampled data. Therefore new techniques are required. There are many traditional approaches for non-uniform sampling and reconstruction methods where the Adaptive Weights (AW) algorithm is considered to be the most efficient. Recently, the Partitions of Unity (PoU) algorithm has been suggested to outperform the AW although there has been much literature covering its numerical performance.

A study and analysis of the implementation of the Adaptive Weights (AW) and Partitions of Unity (PoU) reconstruction methods is conducted. The algorithms consider the missing data problem, defined as reconstructing continuous-time (CT) signals from non-uniform samples which resulted from missing samples on a uniform grid. Mainly, the algorithms convert the non-uniform grid to a uniform grid. The implemented iterative methods construct CT bandlimited functions in frame subspaces. Bandlimited functions are considered to be a superposition of basis functions, named frames. PoU is a variation of AW, they differ by the choice of frame because each frame produces a different approximation operator and convergence rate.

If efficiency is defined as the norm convergence and computational time of an algorithm, then among the two methods, discussed, the PoU method is more efficient. The AW method is slow and converged to a higher error than that of the PoU. However, AW compensates for its slowness and less accuracy by being convergent and robust for large sampling gaps and less sensitive to the sampling irregularities. The impact of additive white Gaussian noise on the performance of the two algorithms is also investigated. The numerical tools utilized in this research consist of the theory of discrete irregular sampling, frames, and iterative techniques. The developed software provides a platform for sampling signals under non-ideal conditions with real devices.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The field of Digital Signal Processing (DSP) relies on the fact that analog signals are represented digitally through sampling methods, so that they are processed by digital systems through reconstruction methods. The problem of sampling and reconstructing analog signals is one of the standard problems in modern DSP and telecommunication systems. An analog, continuous-time, and bandlimited function $f(t) \in L_2(\mathbb{R})$ cannot be stored in its entirety. In practice, only a finite number of samples are measured and stored as a sequence of discrete uniform or non-uniform and or noisy samples. In some applications, it is justified to assume that the sampling set is uniform. However, in many realistic applications the data are known only on a non-uniformly spaced sampling set. This non-uniformity prevents the use of the standard methods from Fourier analysis.

In the *missing data* problem, a loss of data or samples from a uniformly spaced sampling set, generally, result in a sequence of non-uniform samples. Hence, the discussion is the problem of *resampling*, defined as the process of converting between two digital representations of the same analog signal, which have different sampling sets. The non-uniformly sampled signal is resampled onto a uniform set to allow for more straightforward reconstruction procedures in simulation software packages such as MATLAB. This resampling problem has motivation in various engineering applications, including problems relating to communication, medical and underwater imaging, and analog to digital conversion. The question then arises whether and how the lost data can be recovered and sufficiently approximate the analog signal from its non-uniformly distributed samples.

1.1 Scope of Work

This research limits itself to reconstructing bandlimited continuous-time (CT) signals of finite energy modelled by the mathematical set $\mathbb{B}_{\omega_m}$. Formally, $\mathbb{B}_{\omega_m}$ is the space of all bandlimited functions $f(t)$, expressed in the form

$$\mathbb{B}_{\omega_m} = \left\{ f(t) = \frac{1}{2\pi} \int_{-\omega_m}^{\omega_m} d\omega \quad \hat{f}(\omega) e^{-j\omega t} \quad | \quad \text{supp}\left(\hat{f}(\omega)\right) = [-\omega_m, \omega_m] \right\}, \tag{1.1}$$

where $\hat{f}(\omega)$, the Fourier transform of $f(t)$, is square integrable over $[-\omega_m, \omega_m]$, and zero outside $[-\omega_m, \omega_m]$. The set of all bandlimited functions is large and rich to model many signals of practical engineering interest. This is because most real-world signals have some effective bandwidth, where above some frequency they contain insignificant energy. An analog low-pass filter may sometimes be introduced to approximately limit the bandwidth of a signal. This bandlimitedness assumption is often used when dealing with sampling and reconstruction of analog signals. However, recently, some work has been done in the area of sampling and reconstruction with spline type functions and other non-bandlimited CT functions which lie in more general spaces [11].

Two steps are required to represent bandlimited CT signals digitally. Firstly, a finite discrete sequence of real numbers are used to represent the signal (sampling). Secondly, each of the real numbers are represented by a finite number of bits (quantization). For the cases considered here, it is assumed there is no information lost in the sampling step. In the second step there is usually an approximation error between the values of the real signal and the values of the measured signal because of the quantization involved. Floating-point digital signal processors of 32-bit or more are accurate enough to neglect the approximation error, however, this error is significant for the smaller 16-bit fixed point digital signal processors. For a stable representation the approximation error can in principle be made as small as needed by choosing an appropriate number of bits in the digital representation. The emphasis of this thesis is not quantization, therefore, infinite precision is assumed for all quantities which are considered.

The idea that a bandlimited CT signal could be perfectly represented by a discrete sequence of numbers was formalized by Shannon [13]. The sampling theorem states that a bandlimited CT signal $f(t) \in \mathbb{B}_{\omega_m}$ is uniquely determined by its values at the set of uniformly spaced sampling time-instants $\{t_n = nT_s \quad : \quad n \in \mathbb{Z}\}$. $T_s = \pi/\omega_m$, and $1/T_s$ is the minimum sampling rate needed, referred to by Shannon as the Nyquist rate in recognition of the work done by Nyquist [14] on telegraph transmission theory. Furthermore, the function $f(t)$ is expressed as the pointwise convergent series

$$f(t) = \sum_n c_n \phi(t - t_n), \tag{1.2}$$

where $\phi(t)$ is provided by

$$\phi(t) = \frac{\sin(\omega_m t)}{\omega_m t}. \tag{1.3}$$

The coefficients in the expansion $\{c_n : n \in \mathbb{Z}\}$ are the required samples of the signal, provided by

$$c_n = f(t_n). \tag{1.4}$$

A sampled signal then, consists of a set of amplitudes $\{c_n\}$, and a corresponding set of time-instants $\{t_n\}$, and may be written as a collection of amplitude-time pairs $\{c_n, t_n\}$. The set of times $\{t_n\}$ is referred to as the grid, which may be uniform or non-uniform. This thesis is restricted to non-uniform grids which result from a loss of data in uniform grids.

1.2 Discrete Signal

Uniform sampling is the most commonly known form of discretization and it occurs when samples or values of CT signals are obtained at equally-spaced time intervals. Under Nyquist's conditions, a CT signal is recoverable from this set of samples through the use of Shannon's cardinal series. A sampled signal is representable as the amplitude-time pairs, $\{c_n, t_n\}$, or amplitude-index pairs, $\{c_n, n\}$. A CT signal is discretized into N samples, from 1 to N, called the index set $\{n\}$ which also corresponds to the amplitude-time pairs. Then, the amplitude is written as

$$c_n = f(t_n) = f[n]. \tag{1.5}$$

The time-instant variable, t_n, with unit of seconds, is interchangeable with the discrete index n, with unit of samples. The time-instants are normalized by the sampling time interval T_s with unit seconds per sample, which causes time to have convenient integer values at the moments of sampling. The simplicity offered by normalised units is favoured because real units are incidental to the point of a theorem or proof. Uniform sampling is framed in the following

way. A discretized version $f[n]$ is obtained by sampling $f(t)$ on a uniform discrete index set $n = 1, \ldots, N$, which are integers representing sampling positions, and N is possibly a large integer. The index set $\{n\}$ is identified with the finite cyclic group $\mathbb{Z}_N$, thus all discrete signals $f[n]$ are understood as periodic sequences with period N, [1]:

$$f[n] = f[n + mN], \quad m \in \mathbb{Z}, \tag{1.6}$$

or

$$f(t_n) = f(t_n + mt_N), \quad m \in \mathbb{Z}. \tag{1.7}$$

While bandlimited functions on $\mathbb{R}$ are entire functions of exponential type, discrete bandlimited signals also have some smoothness that distinguishes them from general signals on $\mathbb{Z}_N$, and are defined similarly to bandlimited functions on $\mathbb{R}$ [1]. Using the periodicity of $f[n]$ negative indices are admitted and define for $0 < \omega_m < N/2$, the space of discrete bandlimited functions of bandwidth ω_m [1] by

$$\mathbb{B}_{\omega_m} = \left\{ f[n] \in l^2(\mathbb{Z}_N) \quad | \quad \hat{f}(\omega) = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} f[n] e^{-j2\pi\omega \frac{n}{N}} = 0, \quad \text{for } |\omega| > \omega_m \right\}. \tag{1.8}$$

Reconstruction from uniform samples is convenient to implement, however, its limitation lies in the need for the samples to be on a uniform grid. In many applications, sampled data are collected in non-uniform sets or are partly lost or unavailable from uniform sets. When a signal is transmitted, usually, there is jitter errors or even a loss of data which result in larger sampling gaps and, of course, noise when sampled in the receiver. A more general sampling scheme involves the notion of non-uniform sampling, which is an extension of uniform sampling without constraining the sampling instants onto a uniform grid. Numerically, it is required to convert non-uniformly sampled functions to uniformly sampled ones to restore missing data. In practice, $f \in \mathbb{B}_{\omega_m}$ is non-uniformly sampled at a subset $1 \leq n_1 < n_2 < \cdots < n_r \leq N$ [1]. The problem is viewed in the following way. provided the index set n_k and the samples $f(t_{n_k})$, $k = 1, 2, \ldots, r$, of a signal $f \in \mathbb{B}_{\omega_m}$, is f uniquely determined by its samples? If so, what is a practical method to reconstruct f [1]? This is the missing data problem of non-uniform sampling and the focus of this research. This problem deals with uniform discrete-time signals missing samples. Therefore, it is a step closer for numerical implementation than algorithms that deal with continuous-time signals on $\mathbb{R}$ because algorithms are analysed in the version in which they are implemented digital systems [1].

1.3 Problem Statement

In general, it is difficult to perfectly recover signals from samples taken at an arbitrary set of time-instants. Existing reconstruction methods for non-uniform sampling assume some form of structure inherent in the sampling process which imposes a limit on the maximum sampling gap $\delta = \sup_k(t_{n_k} - t_{n_{k-1}})$ such that the CT signal may still be sufficiently reconstructed [1, 4, 8, 9]. Reconstruction methods are different based on the choice of $\phi(t)$ in the pointwise convergent series. The choice of $\phi(t)$ is crucial because it has its limitations which it imposes on the maximum gap. If the maximum gap is exceeded, then the CT signal may not be reconstructed. The Adaptive Weights and Partitions of Unity methods, researched here, use the concept of Riesz bases or, in general, frames for the choice of $\phi(t)$. The concept of frames is an excellent tool to study non-uniform sampling problems. The frame approach has the advantage that it gives rise to deep theoretical results and also to the construction of efficient numerical algorithms. Shift-invariant subspaces in Hilbert spaces, $\mathcal{H}$ of the form

$$\mathbb{V}_\phi = \{\sum_{n\in\mathbb{Z}} c_n\phi_n(t) \quad : \quad c_n \in l_2\}, \tag{1.9}$$

have been used extensively in the analysis and synthesis of signals to introduce and explain the theory of frames [5]. In shift-invariant subspaces bandlimited CT signals, $f(t)$, are considered as a superposition of functions $\phi(t)$, where the sequence $\{\phi_n(t)\}$ either builds an orthonormal basis, or a Riesz basis, or, generally, a frame or weighted frame. Hilbert spaces and the associated concept of orthonormal bases are fundamental importance in signal processing. However, linear independence and orthonormality of the bases impose constraints that make it difficult for the bases to accommodate non-orthogonal functional analysis and synthesis. A frame in a Hilbert space is a generalization of an orthonormal basis that is used to provide non-orthogonal expansions of functions in that space. Frames are relatively easier to construct and implement in numerical analysis than orthogonal or Riesz bases. Frame possess over-completeness properties, meaning that a d-dimensional space is spanned by m frames, where $m > d$ [30]. Then, any $f(t) \in \mathbb{V}_\phi$ can be represented by its samples by the pointwise convergent series

$$f(t) = \sum_n f(t_n)\phi(t - t_n). \tag{1.10}$$

In practice, the reconstruction problem is a finite dimensional problem and is understood as a question concerning over-determined system of linear equations. If the problem is solved numerically, the choice of a basis makes a considerable difference. The basis determines the convergence of the algorithms and it is dependent on the sampling gaps. If the samples are uniformly spaced and obey Nyquist's sampling condition, then the basis is the sinc function

defined in (1.3) and the classical Shannon Theorem provides an explicit reconstruction.

1.3.1 Background

Many well-known non-uniform sampling theorems are reformulations about earlier results on non-harmonic Fourier series. The theory of non-harmonic Fourier series is concerned with the completeness of and expansion properties of sets of complex exponentials $\{e^{j\lambda_n t}\}$ in $L2(-\pi, \pi)$ [4]. It's origin lie in the work of Paley-Wiener in [23]. The bandwidth is normalized, this unitless characterization of bandlimited signals is provided by Paley-Wiener. Results of Paley and Wiener [23], and Kadec [9] relate Riesz bases consisting of complex exponentials to sampling sets that are perturbations of $\mathbb{Z}$. In the context of non-harmonic Fourier series, Paley and Wiener [23] studied the relationship between the Lagrange interpolation formula and the complex exponentials $\{e^{j\omega_m t}\}$ by imposing the constraint $|\lambda_n - n| < 1/\pi^2$ where $n \in \mathbb{Z}$. By relating their work to non-uniform sampling, this corresponded to having the sample instants deviate by a bounded amount from a uniform grid. Levinson [8] then showed that the best possible bound is $|\lambda_n - n| < 1/4$ and provided this constraint, Kadec [9] showed that the set of complex exponentials, $\{e^{j\lambda_n t}\}$, forms a Riesz basis in $L2(-\pi, \pi)$.

Frame theory generalizes and encompasses the theory of Riesz bases. It enables the translation of the sampling problem into a problem of functional analysis. The connection between frames and sets of sampling is established by means of reproducing kernel Hilbert spaces or, more generally, Banach spaces [4]. Reconstruction from non-uniform samples is usually iterative, therefore, the focus of this work is on iterative reconstruction methods using frame theory. Other approaches of direct recovery of bandlimited CT signals from non-uniform samples involve the use of non-uniform splines [32 ,33], reconstruction from recurrent non-uniform sampling [31], and Lagrange interpolation.

1.4 Research Overview

Chapter 1 provides the introduction to the problem of reconstructing bandlimited CT signals from non-uniform samples. It describes the basic properties of the considered signals by defining function spaces in which they lie in. The fundamentals of the missing data problem, defined as lost samples on a uniform grid is formulated to explore suitable iterative reconstruction algorithms.

Chapter 2 discusses the necessary literature for one to have an understanding of uniform sampling, non-uniform sampling, and reconstruction methods. Three types of non-uniform

sampling, Additive Random Sampling, Missing Data Sampling, and Stochastic Jitter Sampling, are introduced. The generalization of the iterative reconstruction methods is provided as the inversion of a linear operator by a Neuman series.

Chapter 3 begins the discussion of reconstruction methods by introducing the Paley-Wiener theorem. The theorem is complex and difficult to implement, so it serves provided the fundamental algorithm of non-uniform sampling. The theorems and proofs relating to Adaptive Weights and Partitions of Unity algorithms is provided.

Chapter 4 discusses the numerical implementation and results of the Adaptive Weights and Partitions of Unity algorithms. A full description of the used equipment, test signals, and testing procedure are provided. The simulation results consist of reconstructed signals, convergence rates, effects of sampling gaps and noise on each iterative method is also provided and compared.

Chapter 5 draws conclusions from the results obtained. This chapter concludes that the Partitions of Unity is faster and convergent for small sampling gaps and that the Adaptive weights method is the preferable method for obtaining convergent results, especially, in the case of large sampling gaps. It also provided the shortcomings of the simulations and provides suggestions for future work.

Appendix A Appendix A describes the author's written software used for numerical simulations in this work. The software removes samples of deterministic and randomly generated signals from a uniform grid. Consecutively, it creates a uniform grid by approximating the missing samples over one hundred thousand iterations. It studies noise by adding additive Gaussian noise. A Chebyshev low-pass filter is used to remove the high frequency components. The filter introduces a delayed output. The delay is removed by shortening the length of the signal, hence the reconstructed signal, in the presence of noise, is truncated.

Chapter 2

Literature Review

Non-uniform, non-equidistant, staggered, irregular, and uneven sampling are some of the common names that have been given to the type of sampling dealt with in this thesis. Non-uniform sampling occurs naturally in many applications, due to imperfect sensors, mismatched clocks, data loss, or event-triggered phenomena. It occurs in various industries or applications, however, herein the focus is on it occurring in data communication applications. Examples of occurrences, in data communication, are data loss in the network core routers due to queueing, and cyber-attacks if the type of attack eliminates data on a uniform grid.

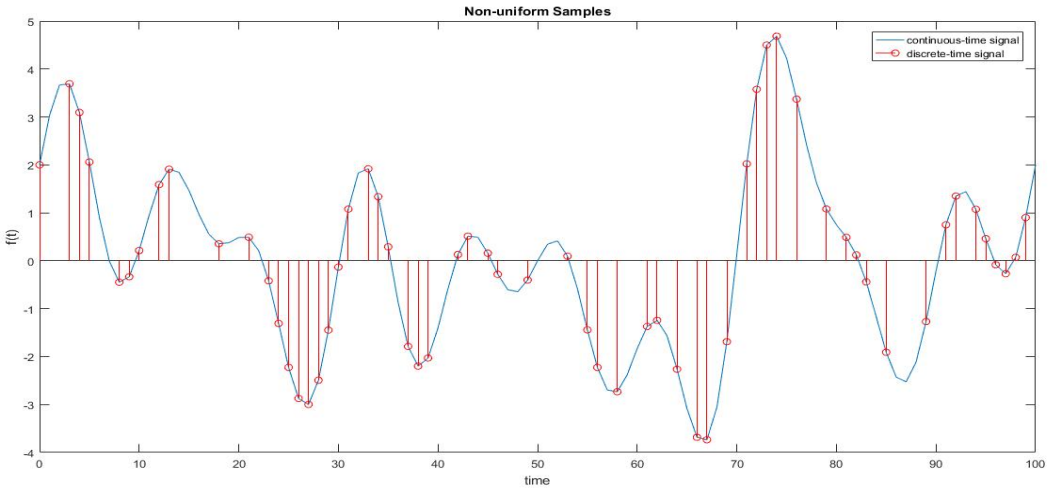

Figure 2.1: A non-uniformly sampled continuous-time signal with a maximum gap of 5 consecutive missing samples on a uniform grid.

Although non-uniform sampling is common, the literature and implementation of discrete-time systems to a large extent focus only on uniform functional analysis and synthesis methods. One very important application of the concept of sampling is its role in processing continuous-

time signals using discrete-time systems. Specifically, continuous-time signals, which either are assumed to be band-limited or are forced to be band-limited by first applying an anti-aliasing filter. These signals are sampled and represented in discrete-time. The discrete-time signal is called non-uniformly sampled when it is possible for the distance between consecutive samples to be unequal. This occurs when samples are collected irregularly or are partly lost or unavailable on a uniform grid. When samples are lost on a uniform grid, the intervals of missing samples are sampled below the critical rate if the uniform samples were retained at the critical rate. The reconstruction methods solve the missing data problem by restoring the missing samples, so uniform discrete-time systems are able to accurately and efficiently approximate band-limited continuous-time signals. In the discrete-time representation, signal amplitude and sampling instant stamps are delivered in pairs, called amplitude-time pairs, $\{c_n, t_n\}$, or amplitude-index pairs, $\{c_n, n\}$. It is common in literature for discrete-time signals and systems to use amplitude-index pairs, where the sequence index $\{n\}$ is an integer variable without reference to a sampling period since discrete-time signals arise in a wide variety of ways besides periodic time sampling. If the sampling index $\{n\}$ of length N, possibly a large integer, is of uniform samples, then $\{n_k\} \subset \{n\}$, $1 \leq n_1 < n_2 < \cdots < n_r \leq N$, is the sampling index of non-uniform samples. Then in amplitude-time or amplitude-index pairs, the sampling is performed at times t_{n_k} and n_k, respectively, to obtain sample values c_{n_k} from a continuous-time signal $f(t)$,

$$c_n = f(t_n) = f[n], \text{ uniform samples,} \tag{2.1a}$$

$$c_{n_k} = f(t_{n_k}) = f[n_k], \text{ non-uniform samples.} \tag{2.1b}$$

The research in this work addresses the non-uniform sampling problem – in the above sense - in a framework of a discrete sampling theorem for bandlimited continuous -time signals, $f(t)$, with finite energy and Fourier transform $\hat{f}(\omega)$,

$$f(t) = \frac{1}{2\pi} \int_{-\omega_m}^{\omega_m} d\omega \quad \hat{f}(\omega) e^{-j\omega t}, \tag{2.2}$$

in the shift-invariant space $\mathbb{V}_\phi$ of the form

$$\mathbb{V}_\phi = \left\{ \sum_{n=-\infty}^{\infty} c_n \phi(t - t_n) : c_n \in l_2 \right\}, \tag{2.3}$$

a subspace of the Hilbert space $\mathcal{H}$. Beutler [22] asserted that for signals $f(t)$ with a Fourier transform there exist functions $\phi(t)$ such that

$$f(t) = \sum_{n=-\infty}^{\infty} f(t_n)\phi(t - t_n) \tag{2.4}$$

converges uniformly for a fixed t. It remains for one to find the synthesis functions $\phi_n(t)$ and hope that the error when using a finite series is small enough. Several contributions follow in this direction. Signals in the subspace $\mathbb{V}_\phi$ are considered a linear superposition of a given basis function $\phi(t)$. The basis functions $\{\phi_n = \phi(t-t_n)\}$ either build an orthonormal basis or a Riesz basis or, more general, a frame. As mentioned $\{n_k\} \subset \{n\}$, the choice of the basis function is very important such that $\{\phi_{n_k} = \phi(t - t_{n_k})\}$ still forms a frame and

$$f(t) = \sum_{k=1}^{r} c_{n_k}\phi_{n_k}, \quad r < N. \tag{2.5}$$

The sampling times, t_{n_k}, are stochastic variables with probability density functions, $p_k(t)$, such that

$$t_{n_k} = t_{k-1} + \tau_k, \quad t_0 = 0, \tag{2.6}$$

where t_{k-1} is the previous uniform sampling time instant and τ_k is the sampling deviation determined by the probability density function (PDF) $p_k(t)$. From the continuous-time function $f(t)$, there is a stochastic observation, c_{n_k}, of the corresponding deterministic sample value

$$c_{n_k} = f(t_{n_k}) = f(t_k + \tau_k). \tag{2.7}$$

The sampled value, $f(t_{n_k})$, is a function of the stochastic variable t_{n_k} and its stochastic properties can be investigated accordingly, by the mean

$$E[c_{n_k}] = E[f(t_{n_k})] \tag{2.8a}$$
$$= \int dt\ f(t)p_k(t), \tag{2.8b}$$

and variance

$$\mathrm{Var}(c_{n_k}) = E[c_{n_k}^2] - (E[c_{n_k}])^2 \tag{2.9a}$$
$$= E[f^2(t_{n_k})] - (E[t_{n_k}])^2 \tag{2.9b}$$
$$= \int dt\ f^2(t)p_k(t) - \left(\int dt\ f(t)p_k(t)\right)^2. \tag{2.9c}$$

Any transform of these stochastic function measurements, can be investigated to study its *a priori* properties, such as frequency analysis. All receivers or propagation channels lead to measured samples corrupted with noise, usually independent and identically distributed with zero mean. In certain analyses, processing the additive measured noise is not crucial, and the effects are included *a posteriori*. For example, adding zero-mean measurement noise results in

$$c_{n_k} = f(t_{n_k}) + \eta_{n_k}, \tag{2.10a}$$

$$E[f(t_{n_k}) + \eta_{n_k}] = E[f(t_{n_k})], \tag{2.10b}$$

$$\mathrm{Var}(f(t_{n_k}) + \eta_{n_k}) = \mathrm{Var}(f(t_{n_k})) + \mathrm{Var}(\eta_{n_k}). \tag{2.10c}$$

2.1 Analysis and Synthesis

The sampling and reconstruction of band-limited continuous-time signals is expressed more generally as signal analysis and synthesis in shift-invariant frame subspaces $\mathbb{V}_\phi \in \mathcal{H}$. The following discussion, demonstrates, that in general, the choice of analysis and synthesis basis functions are dependent on sets of time instants. Assume that the arbitrary signals of interest are $f(t) \in \mathbb{V}_\phi$. Consider two sets of functions $\{s_n(t)\}$ and $\{\phi_n(t)\}$, both of which span the space $\mathcal{H}$, sampling and reconstructing the signal is interpreted as a system composed of analysis and synthesis functions. In the case of uniform sampling, the reconstruction formula is expressed as

$$f(t) = \sum_{n=-\infty}^{\infty} c_n \phi_n(t) \tag{2.11a}$$

$$= \sum_{n=-\infty}^{\infty} f(t_n)\phi_n(t) \tag{2.11b}$$

$$= \sum_{n=-\infty}^{\infty} |\phi_n(t) >< s_n(t)|f(t) > \tag{2.11c}$$

where $\{s_n(t)\}$ are the analysis or sampling basis functions, giving the samples through the inner product

$$c_n = f(t_n) =< s_n(t)|f(t) >= \int_{\mathbb{R}} dt\ s_n(t)f(t), \tag{2.12}$$

and $\{\phi_n(t)\}$ are the synthesis or reconstruction functions used to reconstruct band-limited

continuous-time signals by (2.11). Equivalently by duality, the roles of $\{s_n(t)\}$ and $\{\phi_n(t)\}$ can be interchanged such that $\{\phi_n(t)\}$ is the set of analysis functions and $\{s_n(t)\}$ is the set of synthesis functions. Evidently (2.11) holds only if

$$\hat{\mathbb{1}} = \sum_{n=-\infty}^{\infty} |\phi_n(t) >< s_n(t)| \tag{2.13}$$

forms an identity operator. In practice sampling problems have finite samples, uniform discrete-time signals have N samples and the reconstructed signal is only an approximation of f(t),

$$\tilde{f}_u(t) = \sum_{n=1}^{N} |\phi_n(t) >< s_n(t)|f(t) >, \tag{2.14}$$

approximated by the approximation operator

$$\mathcal{A}_u = \sum_{n=1}^{N} |\phi_n(t) >< s_n(t)|, \tag{2.15a}$$

$$\tilde{f}_u(t) = \mathcal{A}_u f(t). \tag{2.15b}$$

In the case of irregular sampling the approximation is

$$\tilde{f}_{irr}(t) = \sum_{k=1}^{r} |\phi_{n_k}(t) >< s_{n_k}(t)|f(t) >, \quad r < N, \tag{2.16}$$

approximated by the approximation operator

$$\mathcal{A}_{irr} = \sum_{k=1}^{r} |\phi_{n_k}(t) >< s_{n_k}(t)| \tag{2.17a}$$

$$\tilde{f}_{irr}(t) = \mathcal{A}_{irr} f(t). \tag{2.17b}$$

The accuracy of the approximation operator strongly depends on the number of samples. The condition $r < N$ informs one that $\mathcal{A}_{irr}$ is less accurate than $\mathcal{A}_u$. In practice, the analysis function

is unknown, therefore, the choice of the synthesis function plays a crucial role in minimizing the approximation error, such that $\mathcal{A}_{irr} \approx \mathcal{A}_u$. The sets of functions, $\{s_n(t)\}$ and $\{\phi_n(t)\}$, are bases when the sampling is of uniform density at critical sampling. Otherwise, the sets of functions correspond to frames. The set of functions $\{s_n(t)\}$ and $\{\phi_n(t)\}$ are Riesz bases, or generally frames, if for every $f \in \mathbb{V}_\phi$, and $\{c_n\} \in l_2$

$$A \sum_n |c_n|^2 \leq ||\mathcal{A}_u f(t)||^2 \leq B \sum_n |c_n|^2 \tag{2.18}$$

hold, where A and B are called frame bounds. Then, the set of functions $\{s_{n_k}(t)\}$ and $\{\phi_{n_k}(t)\}$ are also frames if there exists a constant $a \in (0, 1]$ such that, [5]:

$$||\mathcal{A}_{irr} f(t) - \mathcal{A}_u f(t) \leq a||\mathcal{A}_u f(t)||^2. \tag{2.19}$$

These equations are also used to establish the existence of the dual frames of the analysis- and synthesis functions. The left inequality of (2.18) implies the convergence of the square-summable sequence while the right inequality shows the stable sampling property [4]. In virtue of the ideas involved in non-uniform sampling theory, uniform sampling is a special case of sampling problems because it only has one constant sampling rate, whereas non-uniform is a problem of varying sampling rates. Additionally, the ability to approximate a signal from non-uniform samples directly translates to the ability to reconstruct a signal from uniform sampling while the converse is not true. In practice, when analysing and synthesising signals, it is sometimes a challenge to determine the dual of a set of functions and the frame bounds, rendering the relationships in (2.18- 2.19) difficult to prove. Usually one is satisfied, merely, with the notion that they exist.

2.2 Uniform Sampling

Before exploring the common types of irregular sampling, regular sampling is, briefly, discussed. The sampling theorem, which is a relatively straightforward consequence of the modulation theorem, is elegant in its simplicity. It states that a band-limited continuous-time function is exactly reconstructed from equally spaced samples provided that the sampling rate is sufficiently high, specifically, that it is greater than twice the highest frequency present in the signal,

$$f_s \geq 2f_m. \tag{2.20}$$

f_s is the sampling rate and f_m is the highest frequency component in the considered signal. One of the important consequences of the sampling theorem is that it provides a mechanism for exactly representing a band-limited continuous-time signal by a sequence of samples, that is, by a discrete-time signal. The reconstruction procedure consists of processing the impulse train of samples by an ideal low-pass filter, by Shannon's theorem

$$f(t) = \sum_n f(t_n)\ \mathrm{sinc}\left(2\omega_m[t - t_n]\right), \tag{2.21}$$

where $\omega_m = 2\pi f_m$. Central to the sampling theorem is the assumption that the sampling frequency is greater than twice the highest frequency in the signal. The reconstructing low-pass filter will always generate a reconstruction consistent with this constraint, even if the constraint was purposely or inadvertently violated in the sampling process. Said differently, the reconstruction process will always generate a signal that is band-limited to less than half the sampling frequency and that matches the given set of samples. If the original signal meets these constraints, the reconstructed signal is identical to the original signal. On the other hand, if the conditions of the sampling theorem are violated, then frequencies in the original signal above half the sampling frequency become reflected down to frequencies less than half the sampling frequency. This distortion is commonly referred to as aliasing, a name suggestive of the fact that higher frequencies (above half the sampling frequency) take on the alias of lower frequencies. It is important to understand that in sampling and reconstruction with an ideal low-pass filter, the reconstructed output will not be equal to the original input in the presence of aliasing, but samples of the reconstructed output will always match the samples of the original signal.

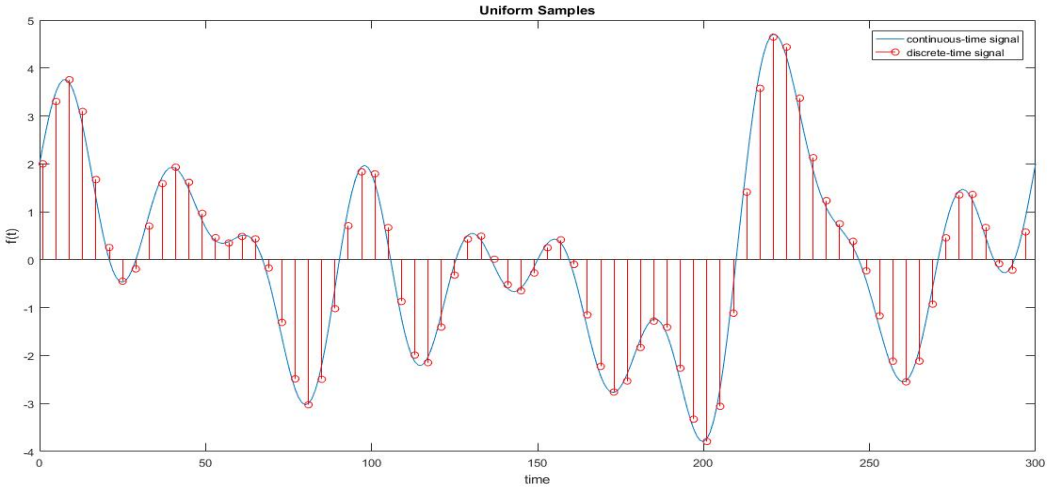

Figure 2.2: Uniformly sampled signal.

To illustrate this concept in an example, it is preferable to work with amplitude-index pairs, $\{f[n], n\}$, because in simulation packages, such as MATLAB, a continuous signal is represented

by uniform samples $f[n]$, then, the uniformly sampled signal is represented by $\{f[n_l]\}$, where $\{n_l\} \subset \{n\}$. If the signal length of $f[n]$ is M, then, the the signal is sampled at a distance $\delta = n_l - n_{l-1} \leq M/f_m$, being kept fixed throughout the vector $f[n]$. Fig.2.2 shows the sampling of a $f_m = 30$ Hz "continuous" signal represented by $M = 300$ samples by a constant distance of $\delta = 4$ samples, therefore, the uniform sample sequence $\{f[n_l]\}$ has the length $N = 72$ samples.

A sampled signal has a periodized frequency response and a complete reconstruction of the signal $f[n]$ is possible aslong as the shifted copies of the periodized spectrum do not overlap. The phenomenon of the overlapping shifted copies is called aliasing. Then, the reconstructed signal will not be the original signal. To avoid aliasing, the sampling has to obey Nyquist's sampling criteria. This case obey Nyquist's criteria, therefore, one has to simply multiply the periodized spectrum by a low pass filter $H(\omega) = 1$ in the interval $\Omega = [-2\pi f_m, 2\pi f_m]$ and zero outside the interval Ω. Since multiplication on the frequency side corresponds to convolution on the time-side, it is of interest also to have a look at a window or basis function $\phi[n]$ whose Fourier transform equals to $H(\omega)$. An ideal brick wall filter is a rectangular pulse function therefore it's time (index) domain representative is a sinc function, which is suitable for the critical sampling case $\delta = M/f_m$.

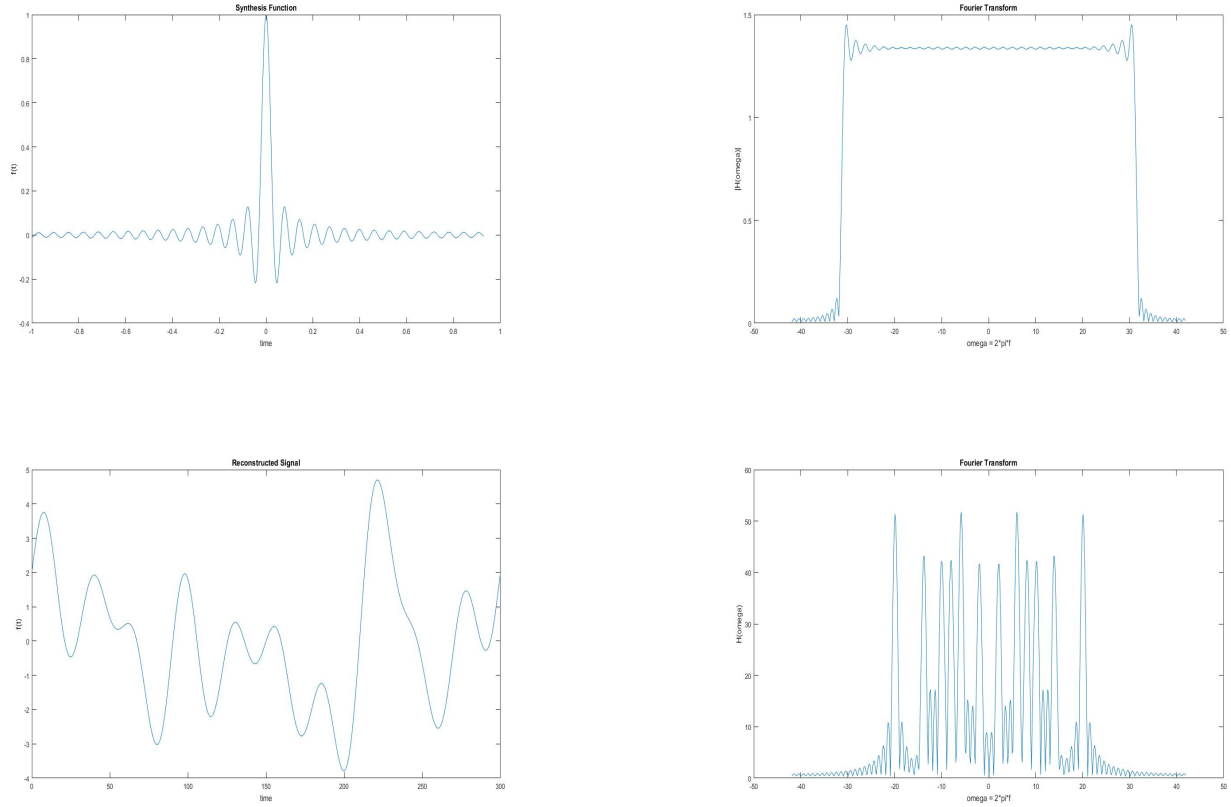

Figure 2.3: Uniform Reconstruction: **top-left:** Synthesis function. **top-right:** Fourier transform of synthesis function. **bottom-left** Reconstructed Function. **bottom-right** Fourier transform of reconstructed function

.

Shannon's theorem in (2.21) captures both of these statements. However, the sinc function is known to have fairly poor decay properties and practically brick wall filters do not exist. In contrast, for the case of oversampling, where one has more samples than absolutely necessary, and for practical purposes of implementation rather than just simulation, gives freedom in the choice of $H(\omega)$.

Since convolution of the sampled signal with the window ϕ is interpreted as forming a series of shifted copies of ϕ, with the given samples being the coefficients of the series expansion, the better decay of ϕ will result in better locality properties of the reconstruction algorithm. In other words, it is not necessary to have all the samples available in order to reconstruct a portion of a signal. The use of well localized bases implies, given some interval of interest it is only necessary to use the samples in some neighbourhood of the given interval in order to achieve a small reconstruction error.

2.3 Non-uniform Sampling

The irregular sampling problem is concerned with the problem of recovering a band-limited signal $f(t)$ with bandwidth ω_m from a sequence of samples $\{f(t_{n_k})\}$ taken in an irregular way. Typically, iterative algorithms are used to recover the signal step by step from the sampling values since direct methods, such as Legrange interpolation, are extremely difficult to numerically implement or simulate and are computationally costly. This thesis describes the situation for the one-dimensional case of vectors of finite length. Such a vector (real- or complex-valued) $f(t_{n_k})$ has spectrum in the interval $\Omega = [-\omega_m, \omega_m]$ around the origin of the frequency domain, if all of the Fourier coefficients of $f(t_{n_k})$ which correspond to parts of the frequency domain outside Ω are zero.

Non-uniform sampling occurs in different forms and this section lists the most common descriptions, Additive Random Sampling (ARS), Missing Data Sampling(MD), and Stochastic Jittered Sampling (SJS) [30]. The non-uniform sampling times, t_{n_k}, are considered to have stochastic properties defined by the probability density function (PDF) p_k(t). Every sampling type constructs the non-uniform sampling instants, t_{n_k}, based on different sampling models. The sampling models of the ARS, MD, and SJS are provide in the table, below, [30]:

Type	t_{n_k}	$E[t_{n_k}]$	$\tau_{n_k} \in$	$p_k(t)$
ARS	$t_{n_k} = t_{k-1} + \tau_k$	kT	$(0, \infty)$	$p_\tau^k(t)$
MDS	$t_{n_k} = t_{k-1} + \tau_k$	$> kT$	$\{T, 2T, \dots\}$	
SJS	$t_{n_k} = kT + \tau_k$	kT	$(-\frac{T}{2}, \frac{T}{2})$	$p_\tau(t - kT)$

Table 2.1: Non-uniform sampling time-instant properties.

2.3.1 Additive Random Sampling

To form Additive Random Sampling (ARS), the sampling times are constructed by adding the sampling deviation, τ_k, to the previous sampling time, [30]:

$$t_{n_k} = t_{k-1} + \tau_k,\ t_0 = 0, \tag{2.22}$$

where $\tau_k \in (0, \infty)$and $E[\tau_k] = T$. This means that $E[t_k] = kT$, while the variance increases with k. The PDF is given as a convolution of the sampling deviation PDF n times, [30]

$$p_k(t) = p_\tau^k(t). \tag{2.23}$$

For example, the exponential distribution,

$$p_\tau(t) = T^{-1}e^{-t/T}, \tag{2.24}$$

gives a Poisson sampling process. The central limit theorem gives that $p_k(t)$ will approach a Gaussian distribution when k goes to infinity [30], since it is the PDF of a sum of k independent identically distributed variables.

2.3.2 Missing Data Sampling

Another case considered, the primary focus of this research, is the non-uniform sampling problem of Missing Data Sampling (MDS), where the underlying sampling procedure is uniform but some samples are missing. This is described with a discrete sampling, [30]:

$$t_{n_k} = t_{k-1} + \tau_k, \quad t_0 = 0, \tag{2.25}$$

and $\tau_k \in \{T, 2T, \dots\}$. This can be seen as a special case of ARS, with a non-trivial PDF for the sampling distribution [30]. Data are lost at random. It is difficult to predict its distribution considering in applications such as a packet data network there are data losses due to queueing in the core network and cyber-attacks which may have different distributions.

2.3.3 Stochastic Jitter Sampling

To formulate the problem of non-uniform sampling as Stochastic Jitter Sampling (SJS) the sampling deviation is added to the expected sampling time,

$$t_{n_k} = kT + \tau_k, \tag{2.26}$$

with $\tau_k \in (-T/2, T/2)$ and $E[\tau_k] = 0$ [30]. In this case the variance is constant over time and the PDF is given directly by the PDF for τ_n, [30]:

$$p_k(t) = p_\tau(t_{n_k} - kT). \tag{2.27}$$

One natural distribution is the rectangular distribution, $p_\tau(t) = 1/T$, $-T/2 < t < T/2$, but it is also possible to imagine a truncated Gaussian distribution or other bounded distributions. The sampling noise can both be known and unknown.

2.4 Spectral Analysis

The computation and study of spectral content is an important part of signal analysis. Conventional spectral analysis techniques require the input signal to be regularly sampled. When the sampling is irregular one can resample or interpolate the signal onto a regular sample grid. However, this can add undesired artefacts onto the spectrum and leads to analysis errors. This section introduces a method to directly analyse the spectrum of an irregularly sampled signal by focusing on the following problem, given irregular measurements c_n at times t_n, how can one best characterize the frequency content in the original signal, $f(t)$? Wojtkiewicz and Tuszynski [15] chose to start from the z-transform [30]

$$\mathcal{C}[z] = \sum_{n=0}^{\infty} c_n z^{-n} \tag{2.28}$$

for sampled signals and construct the Dirichlet transform,

$$\mathcal{C}(s) = \sum_{n=1}^{N} c_n e^{-st_n}, \tag{2.29}$$

where $s = \sigma + j\omega$. This transform is argued to be better suited for analysis of non-uniformly sampled signals, since it preserves information about the time instants. The sampling is considered deterministic and the inverse transform is also derived. Only the case of jitter sampling is discussed in the analysis.

Lomb [16] and Scargle [17] use

$$c_n = a\sin(\omega[t_n - \tau]) + b\cos(\omega[t_n - \tau]) + y_n \tag{2.30}$$

as a model and least squares fitting to find a and b. The time shift τ is chosen such that

$$\tan(2\omega\tau) = \frac{\sum_{n=1}^{M}\sin(2\omega t_n)}{\sum_{n=1}^{M}\cos(2\omega t_n)}, \tag{2.31}$$

for easier computations, since it ensures that the cross-term

$$\sum_{n=1}^{M}\sin(2\omega[t_n - \tau])\cos(2\omega[t_n - \tau]) = 0, \quad \forall\omega. \tag{2.32}$$

This gives the periodogram

$$P_{\mathcal{C}}(\omega) = \frac{1}{M}([\sum_n c_n\sin(2\omega[t_n - \tau])]^2 + [\sum_n c_n\cos(2\omega[t_n - \tau])]^2) \tag{2.33a}$$

$$\approx a^2 + b^2. \tag{2.33b}$$

Comparing the Dirichlet transform (2.29), with $s = j\omega$, and the Lomb-Scargle periodogram, one obtains $P_{\mathcal{C}}(\omega) = |\mathcal{C}(j\omega)|^2/M$. Lomb and Scargle also perform probability calculations and correlation analysis between frequencies when the true signal is sinusoidal and the measurement noise is Gaussian.

In [18], the same signal model is used but extended to a sum over several frequencies,

$$c_n = a_0 + y_n + \sum_k a_k sin(\omega_k t_n) + b_k cos(\omega_k t_n). \tag{2.34}$$

The coefficients a_k and b_k are then considered varying and are estimated recursively using the Kalman filter, for an *a priori* chosen set of frequencies ω_k. This gives a useful algorithm when the frequency content varies over time, but there is no closed form expression like (2.33).

Estimation of the spectrum when c_n is given as samples from a stochastic process is given attention in [19-21]. They consider Poisson sampling, i.e., the sampling times are given from $t_n = t_{n-1} + \tau_n$ and τ_n is taken from an exponential distribution with mean β. First the spectrum is estimated using

$$P_{\mathcal{C}}(\omega) = \frac{1}{\pi M \beta} \sum_{k=1}^{log(M)} \sum_{n=1}^{M-k} c_n c_{n+k} \cos(\omega[t_{n+k} - t_n]), \tag{2.35}$$

and the estimate is shown to be asymptotically unbiased, when $M \to \infty$, for any value β,

$$E[P_{\mathcal{C}}(\omega)] \to \int_0^{\infty} d\tau \quad Cov(f(t), f(t+\tau)) cos(\omega\tau), \tag{2.36}$$

when the sample values $c_n = f(t_n)$ are taken from a Gaussian process, $f(t)$. In [21] the estimate is generalized with the inclusion of a window function so that

$$P_{\mathcal{C}}(\omega) = \frac{1}{M \pi \beta} \sum_{k=1}^{log(M)} \sum_{n=1}^{M-k} c_n c_{n+k} w_M(t_{n+k} - t_n) \cos(\omega(t_{n+k} - t_n), \tag{2.37}$$

where $w_n(t)$ is represented by

$$w_M(t) = \int d\omega \quad \hat{v}(\omega) e^{\frac{j\omega t}{log(M)}}, \tag{2.38}$$

and $\hat{v}(\omega)$ is any symmetric real-valued function scaled so that $w_M(0) = 1$. [20] compares estimation of the spectrum from uniform sampling to Poisson sampling, and in particular the case for finite sample sizes, as opposed to the asymptotic analysis performed in [19,20]. As mentioned the analysis is based on Poisson sampling, so the usefulness is limited when sampling times are given.

2.5 Reconstruction and Estimation

This section is presented as solving the following problem. Given measurements $c_{n_k} = f(t_{n_k})$, what is the best approximation of the original continuous-time signal $f(t)$ at times t_{n_k}? The approximation of $f(t)$ is denoted $\tilde{f}(t)$ in this presentation.

2.5.1 Interpolation

In developing sampling theorems, the reconstruction procedure for recovering an original signal from its samples is based on the use of a low-pass filter in the frequency domain. Correspondingly, in the time domain, the reconstruction is represented by the convolution of the samples $\{f(t_{n_k})\}$ with the synthesis function $\{\phi(t - t_{n_k})\}$, forming the interpolation operator

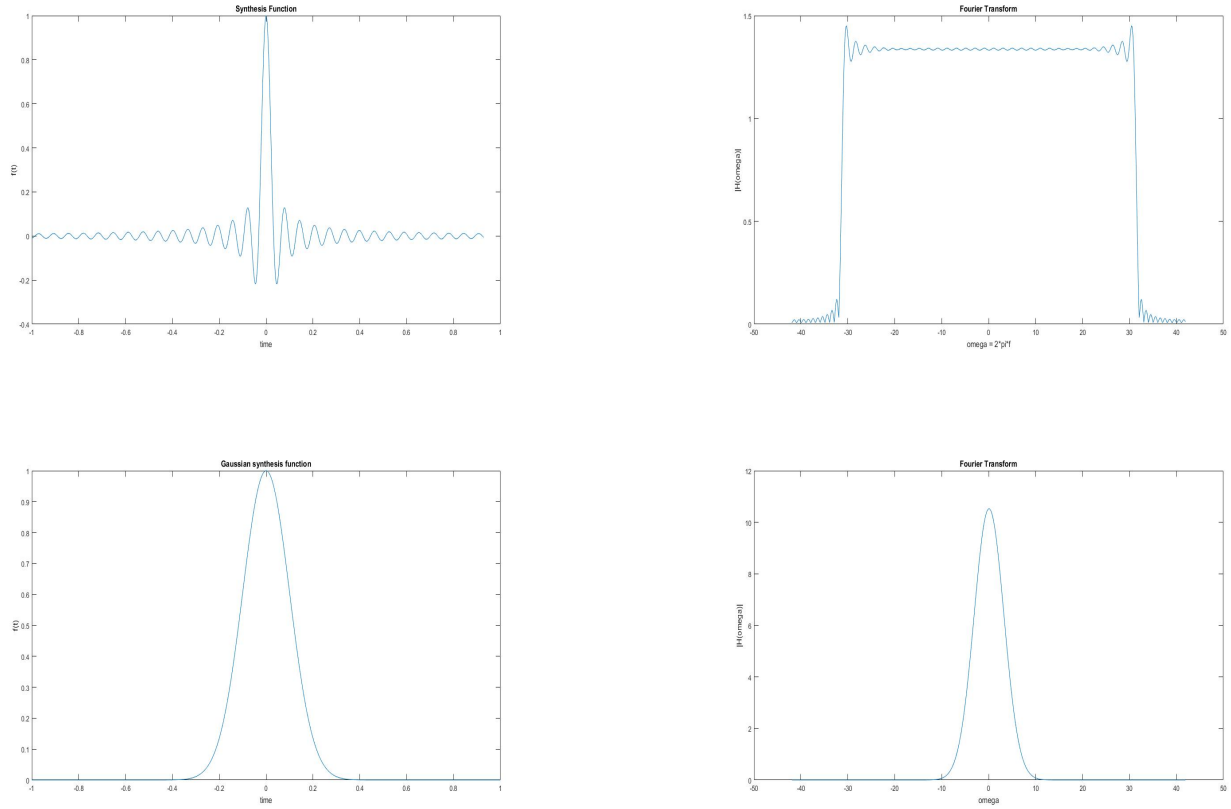

Figure 2.4: Frequency responses of different synthesis function: **top-left:** Time response of a sinc function. **top-right:** Frequency response of sinc function. **bottom-left:** time response of a normalised Gaussian function. **bottom-right** Frequency response of Gaussian

.

$$\mathcal{I}f(t) = \sum_{n=1}^{r} f(t_{n_k})\phi(t - t_{n_k}). \tag{2.39}$$

This superposition represents an interpolation process between the samples. When the reconstruction filter is an ideal low-pass filter, the synthesis function is a sinc function. This is often referred to as band-limited interpolation because it interpolates between sample points by explicitly assuming that the original signal is band-limited to less than half the sampling frequency.

In addition to band-limited interpolation, a variety of other interpolation procedures are commonly used. Different choices of synthesis functions result in different interpolation schemes with different decaying properties in the frequency domain as illustrated in Fig.2.3 and Fig.2.4.

2.5.1.1 Lagrange Interpolation

One approach to reconstruction of band-limited signals from non-uniform samples is through Lagrange interpolation. However, it is extremely difficult to implement. The Lagrange interpolation formula originated as an attempt to find a polynomial function that takes on M function values, ϕ_{n_k} associated with N distinct time instants, t_{n_k}. For reconstruction from non-uniform samples of band-limited signals, the Lagrange interpolation series can be regarded as having infinitely many constraint points.

Yen [24] introduced several reconstruction theorems, mainly to deal with a finite number of non-uniform samples on an otherwise uniform grid, the missing sample problem and recurrent non-uniform sampling. These reconstruction theorems were shown without reference to Lagrange interpolation. The reconstruction of bandlimited signal through Lagrange interpolation functions was introduce by Yao and Thomas [25].

In this section, the series expansion of a band-limited function based on its non-uniform samples is discussed, and the properties related to Lagrange reconstruction are elaborated upon. Yao and Thomas [25] states that a signal $f(t)$, belonging to the class of functions band-limited to ω_m, can be represented as a series expansion given by

$$f(t) = \sum_{k} f(t_{n_k})\phi_{n_k}(t), \tag{2.40}$$

if the sequence of sampling time instants, $\{t_{n_k}\}$ is constrained by

$$|t_{n_k} - nT_s| < \frac{T_s}{4}, \quad \forall n \in \mathbb{Z}, \tag{2.41}$$

where T_s is the sampling period. The set of basis functions $\{\phi_{n_k}(t)\}$ is unique and comprises of Lagrange interpolation functions, given by

$$\phi_{n_k}(t) = \frac{g(t)}{g'(t_{n_k})(t - t_{n_k})}, \tag{2.42}$$

where the prime on $g'(t_{n_k})$, denotes the derivative of

$$g(t) = (t - t_0) \prod_{k=-\infty}^{\infty} (1 - \frac{t}{t_{n_k}})(1 - \frac{t}{t_{n_{-k}}}). \tag{2.43}$$

with respect to its argument, evaluated at the non-uniform sampling instants. The following discussion examines the properties of the Lagrange reconstruction formula described in (2.41) - (2.43). It is shown in [25] that the basis functions $\{\phi_{n_k}(t)\}$ belongs to $\mathbb{B}_{\omega_m}$. If the constraint on the sample time instants in (2.43) is satisfied, then the set $\{e^{-j\omega t_{n_k}}\}$ forms a Riesz basis for the Hilbert space $L_2[-\omega_m, \omega_m]$. Since $\{e^{-j\omega t_{n_k}}\}$ is a Riesz basis, there also exists a unique sequence, $\{\hat{\phi}_n(\omega)\}$ which by duality is also a Riesz basis [26].

Using the interpolation property, where

$$\phi_{n_k}(t_k) = \{1, \quad n = k \quad or \quad 0, \quad n \neq k\}, \tag{2.44}$$

the set of Lagrange interpolation functions, $\{\phi_{n_k}(t)\}$ can be shown to be biorthonormal to the set of shifted sinc functions. If $\{\phi_{n_k}(t)\}$ is the inverse Fourier transform of the set of dual basis $\{\hat{\phi}_n(\omega)\}$,

$$\phi_{n_k}(t) = \frac{g(t)}{g'(t_{n_k})(t - t_{n_k})} \tag{2.45a}$$

$$= \frac{1}{2\pi} \int_{-\omega_m}^{\omega_m} d\omega \quad \hat{\phi}_n(\omega) e^{j\omega t}, \tag{2.45b}$$

then $\hat{\phi}_n(\omega)$ is biorthonormal to $\{e^{-j\omega t_{n_k}}\}$. The shifted sinc functions, $s_n(t) = (\omega_m/\pi)\text{sinc}\,(\omega_m[t - t_{n_k}])$ is the inverse Fourier transform of $\{e^{-j\omega t_{n_k}}\}$ in the interval $\omega_m < \omega < \omega_m$. Then by the isometric

property of Fourier transforms, the two set of sequences $\{s_n(t)\}$ and $\{\phi_{n_k}(t)\}$ are biorthonormal.

2.5.2 Oblique and Orthogonal Projections

In iterative approximation methods, the approximation operator, $\mathcal{A}$ is composed of the interpolation operator, $\mathcal{I}$, and orthogonal projection operator $\mathcal{P}$,

$$\mathcal{A}f = \mathcal{P}\mathcal{I}f. \tag{2.46}$$

The interpolation operator takes the irregular samples of f as its input and provides a continuous function, f_a, an approximation of f as its output, defined as

$$f_a = \mathcal{I}f \tag{2.47a}$$
$$= \sum_{k\in\mathbb{Z}} f[n_k]\phi_{n_k}. \tag{2.47b}$$

The orthogonal projection operator $\mathcal{P}$ which projects a function in Hilbert space $\mathcal{H}$ to $\mathbb{V}_\phi$, given as, [3]:

$$\mathcal{P}f_a = \sum_{k,l} S^{-1}_{k,l} |\phi_{n_k} >< \phi_{n_l}|f_a >, \quad k,l \in \mathbb{Z}, \tag{2.48}$$

and $S^{-1}_{k,l}$ is the inverse of the auto-correlation matrix, $S_{k,l}$, given by the equation

$$S_{k,l} =< \phi_{n_k}|\phi_{n_l} > . \tag{2.49}$$

It is sufficient that $S_{k,l}$ is positive and self-adjoint, so the space $\mathbb{V}_\phi$ is a well-defined closed subset of $\mathcal{H}$ with Riesz basis or frames ϕ_n such that

$$\alpha_0 \sum_k |c_k|^2 \leq |c_k|\phi_{n_k} > |^2 \leq \alpha_1 \sum_k |c_k|^2 \tag{2.50a}$$

$$\alpha_0 \sum_k |c_k|^2 \leq (c_k|\phi_{n_k}>)^*(c_k|\phi_{n_k}>) \leq \alpha_1 \sum_k |c_k|^2 \tag{2.50b}$$

$$\alpha_0 \sum_k |c_k|^2 \leq c_l < \phi_{n_l}|\phi_{n_k} > c_k \leq \alpha_1 \sum_k |c_k|^2 \tag{2.50c}$$

$$\alpha_0 \sum_k |c_k|^2 \leq c_l S_{k,l} c_k \leq \alpha_1 \sum_k |c_k|^2, \tag{2.50d}$$

where $\alpha_0 = ||S_{k,l}||$ and $\alpha_1 = ||S_{k,l}^{-1}||$.

2.5.3 Iterative Reconstruction

This section introduces the concept of iterative reconstruction algorithms that are used in signal analysis and in some areas of mathematics by generalizing their form as the inversion of a linear operator by a Neuman series. Most of the irregular sampling algorithms are iterative in nature. Starting from some initial guess, typically based on the given sampling values, further approximations are obtained step by step. This class of algorithms is distinguished by (a) their linearity, (b) being iterative, and (c) their geometric convergence of successive approximations. A particular case of these methods are algorithms based on frames.

Proposition 1: Let $\mathcal{A}$ be a bounded operator on Hilbert space, $\mathcal{H}$, that satisfies

$$||f - \mathcal{A}f||_{\mathcal{H}} \leq \gamma ||f||_{\mathcal{H}} \quad \forall \quad f \in \mathcal{H}, \tag{2.51}$$

for some positive constant $\gamma < 1$. Then $\mathcal{A}$ is invertible on $\mathcal{H}$ and f can be recovered from $\mathcal{A}f$ by the following iterative algorithm,

$$f_0 = \mathcal{A}f \tag{2.52a}$$

$$f_i = f_{i-1} + \mathcal{A}(f - f_{i-1}), \tag{2.52b}$$

where $i = 1, \ldots, I$ representing the number of iterations or successive approximations and I is the final approximation. Numerically, I is finite, however, theoretically, it is considered to tend to infinity and the iterative algorithm leads to

$$\lim_{i \to \infty} f_i = f, \tag{2.53}$$

with the norm convergence of the $n-th$ iteration given as

$$||f - f_n||_{\mathcal{H}} \leq \gamma^{n+1}||f||_{\mathcal{H}}. \tag{2.54}$$

Proof: In (2.51), the operator norm $||\hat{\mathbb{I}} - \mathcal{A}||_{op}$ is less than γ. This implies that $\mathcal{A}$ is invertible and that the inverse can be represented as a Neumann series:

$$\mathcal{A}^{-1} = \sum_{i=1}^{\infty}(\hat{\mathbb{I}} - \mathcal{A})^i \tag{2.55}$$

and any $f \in \mathcal{H}$ is determined by $\mathcal{A}f$ and the norm-convergent series

$$\hat{\mathbb{I}}f = \mathcal{A}^{-1}\mathcal{A}f \tag{2.56a}$$

$$= \sum_{i=0}^{\infty}(\hat{\mathbb{I}} - \mathcal{A})^i\mathcal{A}f. \tag{2.56b}$$

The reconstruction (2.54) and the error estimate (2.55) follow easily after it is shown that the $n-th$ approximation f_n as defined in (2.53) coincides with the $n-th$ partial sum $\sum_{k=0}^{n}(\hat{\mathbb{I}}-\mathcal{A})^k\mathcal{A}f$ It is evident for $n = 0$, since $f_0 = \mathcal{A}f$ by definition. Next, the $i-th$ iteration is given by

$$f_i = \sum_{k=0}^{i}(\hat{\mathbb{I}} - \mathcal{A})^k\mathcal{A}f \tag{2.57a}$$

$$= \mathcal{A}f + \sum_{k=1}^{i}(\hat{\mathbb{I}} - \mathcal{A})^k\mathcal{A}f \tag{2.57b}$$

$$= \mathcal{A}f + (\hat{\mathbb{I}} - \mathcal{A})\sum_{k=0}^{i-1}(\hat{\mathbb{I}} - \mathcal{A})^k\mathcal{A}f \tag{2.57c}$$

$$= \mathcal{A}f + (\hat{\mathbb{I}} - \mathcal{A})f_{i-1} \tag{2.57d}$$

$$= \mathcal{A}f + \hat{\mathbb{I}}f_{i-1} - \mathcal{A}f_{i-1} \tag{2.57e}$$

$$= f_{i-1} + A(f - f_{i-1}). \tag{2.57f}$$

Now, clearly $f = \lim_{i\to\infty} f_i$. The error estimate in (2.55) is derived as

$$||f - f_i||_{\mathcal{H}} = ||\sum_{k=0}^{\infty}(\hat{\mathbb{I}} - \mathcal{A})^k \mathcal{A}f - \sum_{k=0}^{i}(\hat{\mathbb{I}} - \mathcal{A})^k \mathcal{A}f||_{\mathcal{H}} \tag{2.58a}$$

$$= ||\sum_{k=0}^{i}(\hat{\mathbb{I}} - \mathcal{A})^k \mathcal{A}f + \sum_{k=i+1}^{\infty}(\hat{\mathbb{I}} - \mathcal{A})^k \mathcal{A}f - \sum_{k=0}^{i}(\hat{\mathbb{I}} - \mathcal{A})^k \mathcal{A}f||_{\mathcal{H}} \tag{2.58b}$$

$$= ||\sum_{k=i+1}^{\infty}(\hat{\mathbb{I}} - \mathcal{A})^k \mathcal{A}f||_{\mathcal{H}} \tag{2.58c}$$

$$= ||(\hat{\mathbb{I}} - \mathcal{A})^{i+1}\sum_{k=0}^{\infty}(\hat{\mathbb{I}} - \mathcal{A})^k \mathcal{A}f||_{\mathcal{H}} \tag{2.58d}$$

$$= ||(\hat{\mathbb{I}} - \mathcal{A})^{i+1}\mathcal{A}^{-1}\mathcal{A}f||_{\mathcal{H}} \tag{2.58e}$$

$$= ||(\hat{\mathbb{I}} - \mathcal{A})^{i+1} f||_{\mathcal{H}} \tag{2.58f}$$

$$= (\hat{\mathbb{I}} - \mathcal{A})^{i+1}||f||_{\mathcal{H}} \leq \gamma^{i+1}||f||_{\mathcal{H}}. \tag{2.58g}$$

To prove an iterative reconstruction of f with a geometric rate of convergence, it is necessary to determine a linear approximation operator, $\mathcal{A}$, that requires only the samples of f. Then, prove the norm convergence of the form (2.52). This strategy has been applied successfully to prove irregular sampling theorems in spaces of analytic functions for short time Fourier transforms and wavelet transforms, and for a general class of spaces of band-limited functions. An important instance of this algorithm occurs in the presence of frames in a Hilbert space. Frames were introduced as a generalization of Riesz bases by R. Duffin and A. Schaeffer in their fundamental work on irregular sampling of band-limited functions[2].

Definition: A sequence of functions ϕ_n, $n \in \mathbb{Z}$ in a separable Hilbert space $\mathcal{H}$ is said to constitute a frame, if the following inequalities hold true

$$0 < A \leq B \leq \infty \tag{2.59}$$

and

$$A||f>|^2 \leq \sum_{n=1}^{\infty}|<\phi_n|f>|^2 \leq B||f>|^2 \quad \forall f \in \mathcal{H}. \tag{2.60}$$

The constants A and B are referred to as the frame bounds. The inequality in (2.60) must hold true for arbitrary function $f \in \mathbb{V}_\phi$. As it shall been shown momentarily, the middle term in

(2.60) involves a frame operator. The inequalities in (2.60) subject to the conditions in (2.59) ensure the existence of the inverse of the frame operator. The resulting inverse operator is crucial in constructing the dual frame corresponding to the assumed frame,

$$A||f>|^2 \leq \sum_{n=1}^{\infty} |<\phi_n|f>|^2 \leq B||f>|^2 \tag{2.61a}$$

$$A<f|f> \leq \sum_{n=1}^{\infty} <f|\phi_n><\phi_n|f> \leq B<f|f> \tag{2.61b}$$

$$<f|\left\{A\hat{\mathbb{I}}\right\}|f> \leq <f|\left\{\sum_{n=1}^{\infty}|\phi_n><\phi_n|\right\}|f> \leq <f|\left\{B\hat{\mathbb{I}}\right\}|f> \tag{2.61c}$$

$$<f|\left\{A\hat{\mathbb{I}}\right\}|f> \leq <f|\left\{\hat{\mathbb{S}}\right\}|f> \leq <f|\left\{B\hat{\mathbb{I}}\right\}|f> . \tag{2.61d}$$

Note that the frame operator $\hat{\mathbb{S}}$ in virtue of its construction according to $\sum_{n=1}^{\infty}|\phi_n><\phi_n|$, only requires the frame vectors $|\phi_n>$ and the corresponding hermitian conjugate vectors $<\phi_n|$. In order to interpret the frame operator, apply it onto the function $f \in \mathbb{V}_\phi$,

$$\hat{\mathbb{S}}f = \sum_{n=1}^{\infty}|\phi_n><\phi_n||f> . \tag{2.62}$$

The validity of the inequality (2.60) ensures the existence of the inverse of the frame operator. It was shown that the operator is bounded above and below. Thus multiplying both sides of (2.66) from the left by $\hat{\mathbb{S}}^{-1}$, it results in:

$$f = \sum_{n=1}^{\infty}\hat{\mathbb{S}}^{-1}|\phi_n><\phi_n||f> \tag{2.63a}$$

$$= \sum_{n=1}^{\infty}|\tilde{\phi}_n><\phi_n||f> \tag{2.63b}$$

$$= \hat{\mathbb{I}}|f>, \tag{2.63c}$$

introducing the dual frame $\tilde{\phi}_n$. Consequently, any function $f \in \mathbb{V}_\phi$ can be analysed in terms of the over-complete set of functions ϕ_n and synthesized in terms of the dual frame functions $\tilde{\phi}_n$. Note that the two frame bounds A and B play a vital role in the algorithm and determine the speed of convergence of the algorithm. It is therefore of practical importance to estimate A, B as sharp as possible. The upper estimate in (2.60) is rarely a problem. It expresses the continuity of $f \mapsto <\phi_n|f>$, from $\mathcal{H}$ into l_2 and it is not difficult to derive reasonable estimates

for B. On the other hand, the lower estimate is usually the difficult one and deep part of the argument, and good estimates for the lower frame bound A are mostly beyond reach.

This problem is often dealt with in the following way: one is content with the mere existence of frame bounds A and B and considers the family of operators

$$\hat{\mathbb{S}}_\lambda f = \lambda \sum_n |\phi_n >< \phi_n | f >, \tag{2.64}$$

where λ is a relaxation parameter. With an estimate similar to (2.58) one obtains

$$||f - \hat{\mathbb{S}}_\lambda f|| \leq \gamma(\lambda)||f||, \tag{2.65}$$

where $\gamma(\lambda) = max(|1 - \lambda A|, |1 - \lambda B|)$ and $\gamma(\lambda) < 1$ for small values of λ. The choice

$$\lambda_0 = \frac{2}{A + B} \tag{2.66}$$

gives the optimal value

$$\gamma(\lambda_0) = \frac{B - A}{A + B}, \tag{2.67}$$

and the actual rate of convergence is closer to the optimal value $\gamma(\lambda_0)$, the closer λ is to λ_0.

In the applications to irregular sampling of band-limited functions the following strategy leads to better and explicit estimates of the frame bounds. This method is based on the observation that in most cases it is more natural to work with an approximation operator of the form

$$\mathcal{A}f = \sum_n |h_n >< \phi_n | f > \tag{2.68}$$

for two sequences ϕ_n and h_n in $\mathcal{H}$.

Proposition 2: Suppose that the sequences ϕ_n and h_n have the following properties: there

exist constants $C1, C2 > 0$, $0 \leq \gamma < 1$, so that

$$\sum_n | < \phi_n | f > |^2 \leq C_1 ||f||^2, \tag{2.69a}$$

$$|| \sum_n \gamma_n h_n ||^2 \leq C_2 \sum_n |\gamma|^2, \tag{2.69b}$$

and from the definition of $\mathcal{A}f$ given in (2.68)

$$||f - \mathcal{A}f|| \leq \gamma ||f||, \tag{2.70}$$

$\forall f \in \mathcal{H}$ and $\lambda_n \in l_2$. Then ϕ_n is a frame with frame bounds $(1-\gamma)^2/C_2$ and C_1, and h_n is a frame with bounds $(1-\gamma)^2/C_1$ and C_2

Proof: Let $\mathcal{A}$ be defined as in (2.68), then $\mathcal{A}$ is a bounded operator on $\mathcal{H}$ by (2.69a) and (2.69b) and S is invertible by (2.70) with

$$\mathcal{A}^{-1} = \sum_{n=0}^{\infty} (\hat{\mathbb{I}} - \mathcal{A})^n \tag{2.71}$$

and the operator norm of $\mathcal{A}^{-1}$ is less than $(1-\gamma)^{-1}$. Using (2.63) and (2.64) obtain

$$||\mathcal{A}^{-1}\mathcal{A}f||^2 \quad \leq \quad (1-\gamma)^{-2} || \sum_n < \phi_n | f > h_n ||^2 \quad \leq (1-\gamma)^{-2} C_1 C_2 ||f||^2. \tag{2.72}$$

Next, verify two inequalities which are dual to (2.87a) and (2.87b):

$$\sum_n | < f | h_n > |^2 \leq \quad C_2 ||f||^2, \tag{2.73a}$$

$$|| \sum_n \lambda_n \phi_n ||^2 \leq C_1 \sum_n |\lambda_n|^2. \tag{2.73b}$$

It follows that f can be reconstructed from the frame coefficients $< \phi_n | f >$ in two ways: (a) use the approximation operator $\mathcal{A}$ defined in (2.68), which requires the use of both frames ϕ_n and h_n, and the iteration defined in Proposition 1, or (b) use the frame operator $\hat{\mathbb{S}}_\lambda$ for a suitable relaxation parameter $\lambda = 2C_2[(1-\gamma)^2 + C_1 C_2]^{-1}$. In its abstract form, Proposition 2 does

not have much substance, however in the following sections it will be a valuable guide to prove the existence of frames, irregular sampling theorems and explicit numerical constants. Through different choices of the approximation operator different iterative methods are obtained.

Chapter 3

Investigated Methods

3.1 Paley-Wiener Theorem

This section provides the amended Paley-Wiener reconstruction theorem, amended by Kadec[8] and Levinson [9]. Paley and Wiener [4] where the first to initiate the study of non-harmonic Fourier series and to show that the system $\{e^{j\omega t_{n_k}}\}$ is a Riesz basis for $L2(-\pi,\pi)$ whenever each t_{n_k} is real and $|t_{n_k} - kT_S| \leq L < 1/\pi^2 T_s$. They were interested in determining which perturbations $\{e^{i\omega kT_S}\}$ of the Riesz basis $\{e^{i\omega t_{n_k}}\}$ are still a Riesz basis for $L_2(-\pi,\pi)$. In the context of irregular sampling their result is stated by the following theorem [4]:

Theorem 3.1: *Suppose that the irregular sampling set* $\{n_k, k \in \mathbb{Z}\}$ *satisfies*

$$\sup_k |t_{n_k} - kT_s| \leq L < \frac{T_s}{4}. \tag{3.1}$$

Then, there exists a unique sequence of synthesis functions, $\{g_{n_k}(t)\} \in \mathbb{V}_\phi$*, so that every* $f \in \mathbb{V}_\phi$ *has the representations,*

$$f(t) = \sum_{k\in\mathbb{Z}} f[n_k] g_{n_k}(t) \tag{3.2a}$$

$$= \sum_{k\in\mathbb{Z}} |g_{n_k}(t) >< s_{n_k}(t)| f > \tag{3.2b}$$

$$= \sum_{k\in\mathbb{Z}} |s_{n_k}(t) >< g_{n_k}(t)| f >, \tag{3.2c}$$

where $s_{n_k}(t) = (\omega_m/\pi)\ sinc(\omega_m[t - t_{n_k}])$ *The series converges in* $L_2(\mathbb{R})$ *uniformly on compact sets. The collections* $\{g_{n_k}\}, k \in \mathbb{Z}$ *and* $\{s_{n_k}\}, k \in \mathbb{Z}$ *are Riesz bases of* $\mathbb{V}_\phi$ *and* $< s_{n_k}|g_{n_l} >= \delta_{k,l}$.

Proof: Suppose that the system $\{e^{j\omega t_{n_k}}\}$ is a Riesz basis for $L2(-\pi, \pi)$ and the inequalities

$$A\sum |c_n|^2||e^{j\omega t_{n_k}}||^2 \leq ||c_n e^{j\omega t_{n_k}}||^2 \leq B\sum |c_n|^2||e^{j\omega t_{n_k}}||^2 \tag{3.3}$$

hold for the convergent series

$$f(t) = \sum_k c_k e^{j\omega t_{n_k}}, \tag{3.4}$$

and let $\{kT_S\}$ satisfy $|t_{n_k} - kT_S| \leq L$. It is to be shown that if L is sufficiently small, then

$$||\sum_k c_k \left(e^{j\omega t_{n_k}} - e^{j\omega kT_S}\right)|| \leq \gamma||\sum_k e^{j\omega t_{n_k}}||, \tag{3.5}$$

where $0 \leq \gamma < 1$ and $\{c_n\}$ is a finite sequence of scalars. This criterion ensures that the mapping $e^{j\omega t_{n_k}} \to e^{j\omega kT_S}$ can be extended to a bounded linear operator $\mathcal{A}$ on all of $L2(-\pi, \pi)$ and the the norm operator $||\hat{\mathbb{1}} - \mathcal{A}|| \leq \gamma$. The equation (3.5) is established as follows:

$$||\sum_k c_k \left(e^{j\omega t_{n_k}} - e^{j\omega kT_S}\right)|| = ||\sum_k e^{j\omega t_{n_k}} \left(e^{j\omega(kT_S - t_{n_k})} - 1\right)|| \tag{3.6a}$$

$$= ||\sum_k c_k \sum_{l=1}^{\infty} \frac{[j\omega(kT_S - t_{n_k})]}{l!} t^l e^{j\omega t_{n_k}}|| \tag{3.6b}$$

$$\leq \sum_{l=1}^{\infty} \frac{\pi^l}{l!} ||\sum_k c_k \left[j\omega(kT_S - t_{n_k})\right] e^{j\omega t_{n_k}}|| \tag{3.6c}$$

$$\leq \sum_{l=1}^{\infty} \frac{\pi^l}{l!} \left\{B\sum_k |c_k|^2 |kT_S - t_{n_k}|^{2l} ||e^{j\omega t_{n_k}}||^2\right\}^{1/2} \tag{3.6d}$$

$$\leq \sum_{l=1}^{\infty} \frac{\pi^l}{l!} L^l \left\{B\sum_k |c_n k|^2 ||e^{j\omega t_{n_k}}||^2\right\}^{1/2} \tag{3.6e}$$

$$\leq \sum_{l=1}^{\infty} \frac{(\pi L)^l}{l!} \frac{B}{A} ||\sum_k c_k e^{j\omega t_{n_k}}||^2 \tag{3.6f}$$

$$= \frac{B}{A}(e^{\pi L} - 1)||\sum_k c_k e^{j\omega t_{n_k}}||^2. \tag{3.6g}$$

Note that both inequalities in (3.5) and $||t^x(t)|| \leq \pi^k||x(t||)$ are used. Setting $\gamma = (B/A)\left(e^{\pi L} - 1\right)$,

shows that $\gamma < 1$ provided that L is sufficiently small. By expanding the complex exponentials not as a Taylor series, but in terms of orthonormal basis

$$\{1, \cos(nt), \sin(nt)\}_{n=1}^{\infty}, \tag{3.7}$$

Kadec [8] deduced that (3.5) holds with $\gamma = 1 - \cos(\pi L) + \sin(\pi L)$, provided that $L < 1/4T_s$. If $s_n(t) = (\omega/\pi)\text{sinc}(\omega_m[t - t_{n_k}])$, Levinson [9] found explicit formulas for the g_{n_k} in terms of Lagrange interpolation functions

$$h(t) = t \prod_{k=1}^{\infty} \left(1 - \left[\frac{t}{t_{n_k}}\right]^2\right), \tag{3.8}$$

then,

$$g_{n_k}(t) = \frac{h(t)}{h(t - t_{n_k})h'(t_{n_k})} \tag{3.9}$$

Therefore, an explicit reconstruction formula is formulated. However, implementation of this method is difficult to execute and would be computationally intense. Additionally, in this work, reconstruction methods are applied in solving the missing data problem, therefore, $L > 1/4T_s$ if the uniform samples are assumed to be retained at the Nyquist rate, which is the case. So, although this method is discussed it is not implemented. The discussion only meant to introduce the initial work in non-uniform sampling.

3.2 Adaptive Weights Method

This section covers the Adaptive Weights method for reconstructing bandlimited functions $f \in \mathbb{V}_\phi$ from irregular samples. All of the theory, such as theorems ad lemmas, were taken from the work conducted by Grochenig and Feichtinger [37]. Adaptive Weights provide explicit frame bounds and estimates for the rate of convergence. It is proven completely. Assume that the sampling points t_{n_k}, $k \in \mathbb{Z}$ are ordered by magnitude. The sampling density is measured by the maximal length of the gaps between the samples

$$\delta = \sup_{k \in \mathbb{Z}}(t_{n_k} - t_{n_{k-1}}). \tag{3.10}$$

Denote the midpoints between the samples by $m_k = (t_{n_k} + t_{n_{k+1}})/2$, and the distance between the midpoints by $\Delta m_k = (t_{n_{k+1}} - t_{n_{k-1}})/2$. If the maximal gap length is δ, then $(m_k - t_{n_k}) \leq \delta/2$ and $(t_{n_k} - m_{k-1}) \leq \delta/2$. According to Proposition 1 the approximation operator, $\mathcal{A}$, on $\mathbb{V}_\phi$ that requires only the samples $f(t_{n_k})$ as input must be designed, in order to obtain an iterative reconstruction of $f \in \mathbb{V}_\phi$. An obvious and simple approximation procedure is to firstly, interpolate samples $f(t_{n_k})$ by a characteristic function using the interpolation operator $\mathcal{I}$, followed by the orthogonal projection $\mathcal{P}$ from L_2 onto $\mathbb{V}_\phi$

$$\mathcal{A}f = \mathcal{P}\mathcal{I}f. \tag{3.11}$$

Denoting the characteristic function of the interval $[m_{k-1}, m_k)$ by χ_k, then the first guess for an approximation of f by its samples is

$$f_0 = \mathcal{A}f = \mathcal{P}\sum_{k\in\mathbb{Z}} f(t_{n_k})\chi_k. \tag{3.12}$$

It is easy to verify that $\mathcal{A}$ maps $\mathbb{V}_\phi$ into $\mathbb{V}_\phi$, see also below, so the question is how well $\mathcal{A}$ approximates the identity operator on $\mathbb{V}_\phi$.

Lemma 1: (Bernstein's inequality) If $f \in \mathbb{V}_\phi$, then $f' \in \mathbb{V}_\phi$ and

$$||f'|| \leq \omega||f||. \tag{3.13}$$

Proof: Using

$$< \hat{f}|\hat{g} >= 2\pi < f|g >, \tag{3.14}$$

obtain

$$||f'||^2 = \frac{1}{2\pi}||\hat{f'}||^2 \tag{3.15a}$$

$$= \frac{1}{2\pi}||j\xi\hat{f}(\xi)||^2 \tag{3.15b}$$

$$\leq \frac{\omega_0^2}{2\pi}||\hat{f}||^2 = \omega_0^2||f||^2. \tag{3.15c}$$

Lemma 2: (Wirtinger's inequality) If f, $f' \in L_2(a,b)$ and either $f(a) = 0$ or $f(b) = 0$, then

$$\int_a^b dt\ |f(t)|^2 \leq \frac{4(a-b)^2}{2\pi} \int_a^b dt\ |f'(t)|^2. \tag{3.16}$$

Lemma 2 follows from [7] by a change of variables. It is interesting to note that it has also been used to obtain uniqueness results for band-limited functions.

Proposition 3: If $\delta = \sup_{k\in\mathbb{Z}}(t_{n_k} - t_{n_{k+1}}) < \frac{\pi}{\omega_m}$ then for all $f \in \mathbb{V}_\phi$

$$||f - \mathcal{A}f|| < \frac{\delta\omega_m}{\pi}||f||. \tag{3.17}$$

Consequently, $\mathcal{A}$ is bounded and invertible on $\mathbb{V}_\phi$ with bounds

$$||\mathcal{A}f|| \leq (1 + \frac{\delta\omega_m}{\pi})||f|| \tag{3.18}$$

and

$$||\mathcal{A}^{-1}f|| \leq (1 - \frac{\delta\omega_m}{\pi})^{-1}||f||. \tag{3.19}$$

Proof: Since $f = \mathcal{A}f = P\sum_{k\in\mathbb{Z}} f[n]\chi_k$ for $f \in \mathbb{V}_\phi$, and since the characteristic functions χ_k have mutually disjoint support,

$$||f - \mathcal{A}f||^2 = ||Pf - P\sum_{k\in\mathbb{Z}} f(t_{n_k})\chi_k||^2 \tag{3.20a}$$

$$\leq ||\sum_{k\in\mathbb{Z}}(f(t_n) - f(t_{n_k}))\chi_k)||^2, \tag{3.20b}$$

where

$$||\sum_{k\in\mathbb{Z}}(f(t_n) - f(t_{n_k}))\chi_k)||^2 = \sum_{k\in\mathbb{Z}} \int_{m_{k-1}}^{m_k} dt\ |f(t) - f(t_{n_k})|^2. \tag{3.21}$$

Then apply Wirtinger's inequality to obtain

$$\int_{m_{k-1}}^{m_k} dt\ |f(t) - f(t_{n_k})|^2 \leq \frac{4(t_{n_k} - m_{k-1})^2}{\pi^2} \int_{m_{k-1}}^{t_{n_k}} dt\ |f'(t)|^2 + \frac{4(m_k - t_{n_k})^2}{\pi^2} \int_{t_{n_k}}^{m_k} dt\ |f'(t)|^2 \tag{3.22a}$$

$$\leq \frac{\delta^2}{\pi^2} \int_{m_{k-1}}^{m_k} dt\ |f'(t)|^2, \tag{3.22b}$$

sum over k and use Bernstein's inequality to obtain

$$\sum_{k\in\mathbb{Z}} \int_{m_{k-1}}^{m_k} dt\ |f(t) - f(t_{n_k})|^2 \leq \frac{\delta^2}{\pi^2} \int_{m_{k-1}}^{m_k} dt\ |f'(t)|^2, \tag{3.23}$$

and

$$\frac{\delta^2}{\pi^2} \int_{m_{k-1}}^{m_k} dt\ |f'(t)|^2 = \frac{\delta^2}{\pi^2} ||f'||^2 \tag{3.24a}$$

$$\leq \frac{\delta^2 \omega_0^2}{\pi^2} ||f||^2. \tag{3.24b}$$

A combination of these estimates yields

$$||\mathcal{A}f|| < ||f|| + ||\mathcal{A}f - f|| \leq (1 + \frac{\delta\omega_m}{\pi})||f||, \tag{3.25}$$

and

$$||\mathcal{A}^{-1}f|| = ||\sum_{k=0}^{\infty} (\hat{\mathbb{I}} - \mathcal{A})^k f|| \leq \sum_{k=0}^{\infty} (\frac{\delta\omega_m}{\pi})^k ||f||. \tag{3.26}$$

To obtain a quantitative sampling theory apply Propositions 1-3 in the following ways [37].

Theorem 3.2: *If* $\delta = \sup_{k\in\mathbb{Z}}(t_{n_k} - t_{n_{k-1}}) < \frac{\pi}{\omega_m}$ *then* $f \in \mathbb{V}_\phi$ *is uniquely determined by its*

samples $f(t_{n_k})$ and can be reconstructed iteratively as follows

$$f_0(x) = P\sum_{k\in\mathbb{Z}} f(t_{n_k})\chi_k \tag{3.27a}$$

$$f_i(x) = f_{i-1} + P\sum_{k\in\mathbb{Z}} (f(t_{n_k}) - f_{i-1}(t_{n_k}))\chi_k. \tag{3.27b}$$

Proof: Combine Propositions 1 and 3. To deduce further conclusions look at the approximation operator $\mathcal{A}$ [37],

$$\mathcal{A}f = P\frac{\omega_m}{\pi}\sum_{k\in\mathbb{Z}} |\frac{1}{\sqrt{\Delta m_k}}\chi_k >< \sqrt{\Delta m_k}\text{sinc}(\omega_m[t - t_{n_k}])|f > . \tag{3.28}$$

This normalization is natural, because $\frac{1}{\sqrt{\Delta m_k}}\chi_k \leq 1$. Now, it is easy to verify the assumptions of Proposition 2. Therefore the collections

$\{\sqrt{\Delta m_k}\text{sinc}(\omega_m[t - t_{n_k}]) >, k \in \mathbb{Z}\}$ and $\{\frac{1}{\sqrt{\Delta m_k}}\chi_k\, k \in \mathbb{Z}\}$ are frames for $\mathbb{V}_\phi$.

Theorem 3.3: *Let $\{t_{n_k}, k \in \mathbb{Z}\}$ be a sampling set with $\delta = \sup_{k\in\mathbb{Z}}(t_{n_k} - t_{n_{k-1}}) < \frac{\pi}{\omega_m}$. Then,*

(a) Weighted Frames: *For any $f \in \mathbb{V}_\phi$*

$$(1 - \frac{\delta\omega_m}{\pi^2})^2||f||^2 \leq \sum_{k\in\mathbb{Z}} \Delta m_k|f(t_{n_k})|^2 \leq (1 + \frac{\delta\omega_m}{\pi^2})^2||f||^2. \tag{3.29}$$

Consequently, the collection $\{\sqrt{\Delta m_k}\text{sinc}(\omega_m[t - t_{n_k}]) >, k \in \mathbb{Z}\}$ is a frame with frame bounds $A = (1 - \frac{\delta\omega_m}{\pi^2})^2$ and $B = (1 + \frac{\delta\omega_m}{\pi^2})^2$.

(b) The Adaptive Weights Method: *For any $f \in \mathbb{V}_\phi$ can be reconstructed from its samples $f(t_{n_k})$ by the adapted frame algorithm:*

$$f_0 = P\frac{\pi^2}{\pi^2 + \delta^2 + \omega_0^2}\sum_{k\in\mathbb{Z}} f(t_{n_k})\sqrt{\Delta m_k}\text{sinc}(\omega_m[t - t_{n_k}]) \tag{3.30a}$$

$$f_i = f_i + P\frac{\pi^2}{\pi^2 + \delta^2 + \omega_0^2}\sum_{k\in\mathbb{Z}} (f(t_{n_k}) - f_{i-1}(t_{n_k})\sqrt{\Delta m_k}\text{sinc}(\omega_m[t - t_{n_k}]). \tag{3.30b}$$

The convergence of the the number iterations are defined by

$$f = \lim_{i\to\infty} f_i, \tag{3.31}$$

and

$$||f - f_i|| \leq (\frac{2\pi\delta\omega_m}{\pi^2 + \delta^2\omega_0^2})^{i+1}||f||. \tag{3.32}$$

Proof of (a): For the left hand inequality of (3.29) use Proposition 3 and obtain

$$||f||^2 = ||\mathcal{A}^{-1}\mathcal{A}f||^{-2} \leq (1 - \frac{\delta\omega_m}{\pi})^2||P\sum_{k\in\mathbb{Z}} f(t_{n_k})\chi_k||^2 \tag{3.33a}$$

$$\leq (1 - \frac{\delta\omega_m}{\pi})^2||\sum_{k\in\mathbb{Z}} f(t_{n_k})\chi_k||^2. \tag{3.33b}$$

Since the supports of χ_k are mutually disjoint

$$||f(t_{n_k})\chi_k||^2 = \int dt\ |\sum_{k\in\mathbb{Z}} f(t_{n_k})\chi_k|^2 \tag{3.34a}$$

$$= \sum_{k\in\mathbb{Z}} \int dt\ |f(t_{n_k})|^2\chi_k \tag{3.34b}$$

$$= \sum_{k\in\mathbb{Z}} |f(t_{n_k})|^2 \int dt\ \chi_k \tag{3.34c}$$

$$= \sum_{k\in\mathbb{Z}} |f(t_{n_k})|^2\Delta m_k \tag{3.34d}$$

For the right-hand side use

$$\sum_{k\in\mathbb{Z}} |f(t_{n_k})|^2\Delta m_k = ||f(t_{n_k})\chi_k||^2 \leq (||f|| + ||\sum_{k\in\mathbb{Z}} f(t_{n_k})\chi_k - f)^2. \tag{3.35}$$

Since $f(t_{n_k}) = \frac{\omega_m}{\pi} < f|\ \mathrm{sinc}(\omega_m[t - t_{n_k}]) >$, the second statement is clear.

Proof of (b) follows from Proposition 1, (9) and (10), when the relaxation is parameter

$$\lambda = \frac{2}{A + B} \tag{3.36a}$$

$$= \frac{1}{1 + \frac{\delta^2\omega_m^2}{\pi^2}}, \tag{3.36b}$$

which provides a rate of convergence

$$\lambda = \frac{B - A}{B + A} \tag{3.37a}$$

$$= \frac{2\pi\delta\omega_m}{\pi^2 + \delta^2\omega_m} < 1. \tag{3.37b}$$

3.3 Partitions of Unity Method

This section presents the concept of Partitions of Unity (PoU) used for reconstructing band-limited continuous-time functions. The theory is taken from lecture notes [34]. Partitions of unity is a variation of the Adaptive Weights method, they differ by the different choice of frames. A partition of nity is a decomposition of the constant 1 into a sum of continuous windowing functions

$$\sum_n \phi_n(t) = 1, \quad t \in T, \tag{3.38}$$

on a given domain of interest, T. Let T be a topological space, then a collection $U_n \subset T$ such that

$$T = \bigcup_n U_n, \tag{3.39}$$

form an open cover of T and one is interested in the existence of partitions of unity subordinated to the cover. The continuous function, ϕ_n is required to be differentiable and concentrated in a given, usually very small, open U_n. When it comes to applications to Geometry and Analysis the discussion includes a set S of continuous functions. To specify the axioms for S, consider the space of continuous functions on T:

$$\mathcal{C}(T) = \{g : T \to \mathbb{R} : g \text{ is continuous}\}. \tag{3.40}$$

Definition 3.1: Given a topological space T, a subset $S \subset \mathcal{C}(T)$:

a. *is closed under finite sums*: if $g + h \in S$ whenever $g, h \in S$.

b. *is closed under quotients*: if $g/h \in S$ whenever $g, h \in S$ and h is nowhere vanishing.

S is closed under multiplication by real numbers, or multiplication of continuous functions as well. However, the most important condition on S is lowing topological one:

Definition 3.2: Given a topological space T and $S \subset \mathcal{C}(T)$ then S is normal if for any two closed disjoint subsets $T_k, T_l \subset T$, there exists $g : T \to [0,1]$ which belongs to S and such that $g|_{T_k} = 0, g|_{T_l} = 1$. The existence of such continuous functions implies that T must be *normal*; any two closed disjoint subsets $T_k, T_l \subset T$ can be separated topologically.

Lemma 3: In a normal space T, if $T_n \subset U \subset T$ with T_n-closed and U-open in T, then there exists an open V in T such that $T_n \subset V \subset \bar{V} \subset U$.

Proof: If $T_n \subset U$, then T_n is disjoint to $T - U$ They are both closed, hence it is possible to find disjoint opens W and V such that $T_n \subset V$ and $T - U \subset W$. The condition $V \cap W = \emptyset$ is equivalent to $V \subset T - W$. Since $T - W$ is closed containing V, this implies $\bar{V} \subset T - W$.. On the other hand, $T - U \subset W$ can be re-written as $T - W \subset U$. Hence $\bar{V} \subset T - W \subset U$.

3.3.1 Finite Partitions of Unity

This section provides a precise meaning to the statement that a continuous function ϕ_n is concentrated in an open $T_n \subset T$,

$$\{g \neq 0\} = \{t \in T : g(t) \neq 0\} \tag{3.41}$$

Definition 3.3: *Given a topological space T and $\phi : T \to \mathbb{R}$ define the support of ϕ as the closed set*

$$\text{supp}(\phi) = \overline{\{\phi \neq 0\}}. \tag{3.42}$$

It is said that ϕ is supported in an open U if $\text{supp}(\phi) \subset U$.

It is important that the support is defined as the closure of $\{\phi \neq 0\}$. This condition allows one to perform globalization. Next the discussion of finite partitions of unity is presented.

Definition 3.4: *Let T be a topological space, $U = \{U_1, \ldots, U_n\}$ a finite open cover of T. A*

partition of unity subordinated to U is a family of functions $\phi_n : T \to [0,1]$ *satisfying*:

$$\sum_i \phi_i(t) = 1, \quad \text{supp}(\phi_i) \subset U_i. \tag{3.43}$$

Given $S \subset \mathcal{C}(X)$*, it said that* $\{\phi_n\}$ *is a S-partition of unity if* $\phi_i \in S \ \forall \ i$.

Theorem 3.4: *Let T be a topological space and assume that* $S \subset C(X)$ *is normal and closed under finite sums and quotients. Then, for any finite open cover U, there exists an S-partition of unity subordinated to U.*

Proof: The main topological ingredient in the proof is the following shrinking lemma

Lemma 4: *The finite shrinking lemma - For any finite open covering* $U = \{U_i : 1 \leq i \leq n\}$ *of a normal space T, there exists a covering* $V = \{V_i : 1 \leq i \leq n\}$ *such that*

$$\overline{V}_i \subset U_i, \quad \forall i = 1, \ldots, N \tag{3.44}$$

Proof: Let

$$A = T - (U_2 \cup \cdots \cup U_N) \tag{3.45}$$
$$D = U_1 \tag{3.46}$$

Then A is closed, D is open, and $A \in D$. V_1 is open such that

$$A \subset V_1 \subset \overline{V}_1 \subset D \tag{3.47}$$

This means that $\{V_1, U_2, \ldots, U_n\}$ is a new open cover of T with $\overline{V}_1 \subset U_1$. Applying the same argument to this new cover to refine U_2, a new open cover $\{V_1, V_2, U_3, \ldots, U_n\}$ with $V_1 \subset U_1$, $\overline{V}_2 \subset U_2$ is found. Continuing this argument, the desired open cover V is obtained.

To prove the theorem, let $U = \{U_i\}$ be the given finite open cover. Apply the previous lemma twice and choose open covers $V = \{V_i\}$, $W = \{W_i\}$, with $\overline{V}i \subset Ui,$, $\overline{W}_i \subset V_i$. For each i, the separation property of S for the disjoint closed sets $(\overline{W}_i, T - V_i)$ is used. As a result, $g_i : T \to [0,1]$ belongs to S, with $f_i = 1$ on $\overline{W}_i$ and $f_i = 0$ outside V_i. Note that

$$g = \sum_i^N g_i \tag{3.48}$$

is nowhere zero. Indeed, if $f(t) = 0$, then $f_i(x) = 0, \quad \forall i, \quad t \notin W_i$. But this contradicts the fact that W is a cover of T. From the properties of S, each

$$\phi_i = \frac{g_i}{\sum_i^N g_i} : T \to [0, 1] \tag{3.49}$$

is continuous. Clearly, their sum is 1. Finally, $\text{supp}(\eta_i) \subset U_i$ because $\overline{V}_i \subset U_i$ and $\{\phi \neq 0\} \subset V_i$.

3.3.2 Arbitrary Partitions of Unity

For arbitrary partitions of unity one has to deal with infinite $\sum_i g_i$ of continuous functions on T, where i is some index in an infinite set I. In such cases it is natural to require that, for each $t \in T$, the $\sum_i g_i$ is finite, in example $g_i = 0$ for all but a finite number of i's. Although the sum is well defined as a function on T, to retain continuity, a slightly stronger notion is needed.

Definition 3.5: *Let T be a topological space and let $\mathcal{S} = \{S_i\}$ be a family of subsets of T. It is said that $\mathcal{S}$ is locally finite in the space T if for any $t \in T$, there exists a neighbourhood V_x such that it intersects only finitely many subsets that belong to $\mathcal{S}$.*

Definition 3.6: *Given a topological space T, a family $\{\tilde{g}_i : i \in I\}$ of continuous functions $\tilde{g}_i : T \to \mathbb{R}$ is called a locally finite family of continuous functions if $\{supp(\tilde{g}_i) : i \in I\}$ is locally finite.*

Definition 3.7: Given a topological space T and $S \subset \mathcal{C}(X)$, then S is closed under locally finite sums if for any locally finite family $\{\tilde{g}_i : i \in I\}$ of functions from S, $\sum_i g_i \in S$.

Definition 3.8: Let T be a topological space, $U = \{U_i : i \in I\}$ an open cover of T. S-partition of unity subordinated to U is a locally finite family of functions $\phi_i : T \to \mathbb{R}$ satisfying:

$$\sum_i \phi_i = 1, \quad \text{supp}(\phi_i) \subset U_i \tag{3.50}$$

Given $S \subset \mathcal{C}(X)$, then $\{\phi_i\}$ is an S-partition of unity if $\phi_i \in S$ $\forall i$.

The existence of partitions of unity, for arbitrary covers, forces T to have a special topological property, called paracompactness. As in the case of compactness, paracompactness is best characterized in terms of open covers.

Definition 3.9: *Let T be a topological space and let $\mathcal{A}$ be a cover of T. A refinement of $\mathcal{A}$ is any other cover $\mathcal{B}$ with the property that any $B \in \mathcal{B}$ is contained in some $A \in \mathcal{A}$.*

Definition 3.10: *A topological space T is called paracompact if any open cover admits a locally finite refinement.*

As in the previous subsection the shrinking lemma is needed for partitions of unity.

Lemma 5: *If T is a paracompact Hausdorff space then T is normal and, for any open cover $U = \{U_i : i \in I\}$ there exists a locally finite open cover $V = \{V_i : i \in I\}$ with the property that $\overline{V}_i \subset U_i$ for all $i \in I$.*

Proof: First show that T is normal. It suffices to show that, for $Y, Z \in T$ if Z is closed and $Y|\{z\}$ for all $z \in Z$, then $Y|Z$. The condition $Y|Z$ is implies, and it is actually equivalent to, the existence of an open neighbourhood V of Z such that $Y \cap \overline{V} = \emptyset$. Indeed, if $U \cap V = \emptyset$ for some open neighbourhoods U of Y and V of Z, then $V \subset T - U$ where the last set is closed, hence $\overline{V} \subset T - U$, hence $\overline{V} \cap U = \emptyset$; since $Y \subset U$, then $\overline{V} \cap Y = \emptyset$, for the converse, $U = X - \overline{V}$.

Hence assume now that $Y|\{z\}$ for all $z \in Z$ and prove $Y|Z$. For each $z \in Z$ choose an open neighbourhood V_z such that $Y \cap \overline{V}_z = \emptyset$. Then $\{V_z : z \in Z\} \cup \{T - Z\}$ is an open cover of T. Let U be a locally finite refinement and let $W = \{W_i : i \in I\}$ consisting of those members of U which intersect Z. Define $V = \bigcup_i W_i$. This is an open neighbourhood of Z. Note that $Y \cap \overline{W}_i = \emptyset$ for all i, since each W_i is inside some V_z and $Y \cap \overline{V}_z = \emptyset$ by construction). Also, due to local finiteness

$$\overline{V} = \bigcup_i \overline{W}_i. \tag{3.51}$$

Hence $\overline{V} \cap Y = \emptyset$, proving that $Y|Z$. In conclusion T must be normal. To prove the second part. Consider $\mathcal{A} = \{V \subset T \text{ open} : \overline{V} \subset U_i \text{ for some } i \in I\}$. Since T is normal, Lemma 3 implies that $\mathcal{A}$ is an open cover of T. Let $\mathcal{B} = B_j : j \in J$ be a locally finite refinement of $\mathcal{A}$ which is an open cover of T. Then, for each $j \in I$ there is an element e_j such that $\overline{B}_j \subset U_{e_j}$, defining a function $h : J \to I$. By definition

$$V_i = \bigcup_{j \in e_j} B_j \tag{3.52}$$

this is empty if e_j^{-1} is empty, and $\overline{V} \subset U_i \quad \forall i$.Finally, remark that $\{V_i\}$ is locally finite: if a neighbourhood of a point intersects V_i then it intersects B_j for some $j \in e_j$ hence it intersects an infinite number of V_i's, then it would also intersect an infinite number of B_j's.

Theorem 3.5: *Let T be a paracompact Hausdorff space and assume that $S \subset \mathcal{C}(T)$ is normal, closed under locally finite sums and closed under quotients. Then, for any open cover U of T, there exists an S-partition of unity subordinated to U.*

Proof. The proof is completely similar to the proof from the finite case. Apply the shrinking lemma twice to find coverings $\{V_i\}$ and $\{W_i\}$ with $\overline{V}_i \subset U_i$, $\overline{W}_i \subset V_i$. Then choose $\eta_i : T \to [0,1]$ such that $\eta_i = 1$ on $\overline{W}_i$ and 0 on $T - V_i$, with $\eta_i \in S$. Finally, since the families are locally finite,

$$\phi_i = \frac{\eta_i}{\sum_j \eta_j} \tag{3.53}$$

makes sense and is the desired partition of unity.

3.3.3 Algorithm

Grouping neighbouring windowing functions forms arbitrary non-uniform partitions of unity, which significantly reduce computation times. When used in conjunction with analysis and synthesis windowing, they ensure an overall amplitude-preserving transformation. Another expected quality of a windowing functions is a high order of differentiability, or smoothness. This provides good spectral domain behaviour and avoids undesirable effects such as Gibbs phenomenon, or spectral ringing [6]. A theorem from [35] is used to formulate the theory of Partitions of Unity iterative reconstruction algorithm.

Theorem 3.6: *Let $\phi \in \mathbb{V}_\phi$ and $\mathcal{P}$ a bounded orthogonal projection from L_2 to $\mathbb{V}_\phi$. Assume that the index set $\{n_k\}$ is a subset sequence $1 \leq n_1 < n_2 < \cdots < n_r \leq N$ of $n = 1, \ldots, N$ with maximum gap, $\delta = \sup_k(t_{n_k} - t_{n_{k-1}})$ such that $\delta < \pi/\omega_m$. Then $f \in \mathbb{V}_\phi$ is completely determined by the non-uniform samples $f(t_{n_k})$,$k = 1, \ldots, r$, and can be reconstructed iteratively by the algorithm*

$$f_0 = \mathcal{P}\mathcal{Q}f \tag{3.54}$$

$$f_i = \mathcal{P}\mathcal{Q}(f - f_{i-1}) + f_{i-1} \tag{3.55}$$

where the quasi-interpolant operator $\mathcal{Q}$ is defined by

$$\mathcal{Q}f = \sum_{k=1}^{r} f(t_{n_k})\phi_{n_k} \tag{3.56}$$

To create a suitable quasi-interpolant, let $\{\phi_i\}_{i\in\mathbb{Z}}$ be a basis that satisfies the conditions

a. $0 < \phi_{n_k} \leq 1$

b. $\text{supp}(\phi_{n_k}) \subset U_k$

c. $\sum_k \phi_{n_k} = 1$

The definition of the orthogonal projection operator $\mathcal{P}$ is provided by Eq. (2.54). Since partitions of unity are variations of adaptive weights, it has similar proofs hence none are provided here.

Chapter 4

Numerical Simulations

This chapter provides numerical evidence for the theory of iterative reconstruction methods covered in previous sections. Through different choices of frames, $\phi(t)$, different approximation operators, $\mathcal{A}$, are obtained, leading to different iterative methods with different performances. A study and analysis of the implementation of the Adaptive Weights and Partition of Unity reconstruction methods was conducted. The algorithms considered the *missing data* problem, defined as reconstructing CT signals from non-uniform samples which resulted from missing samples on a uniform grid. Mainly, the algorithms convert the non-uniform grid to a uniform grid which allows conventional uniform reconstruction methods implemented in digital machines to work. Additionally, the effects of noise are studied on the developed algorithms.

Implementation of signal recovery from non-uniformly sampled data requires the inversion of the corresponding operator $\mathcal{A}$, which is, in general, computationally a challenging and costly procedure. In applications, one is usually satisfied with signal reconstruction with certain limited accuracy and apply for the reconstruction in an iterative procedure. Iterative methods are chosen since it may be difficult to directly determine the inverse of $\mathcal{A}$. The iterative methods, approach the inversion of the approximation operator $\mathcal{A}$ as the inversion of a linear operator by a Neuman series. The aim in this chapter is to demonstrate that the choice of Adaptive Weights is, in many respects, the method of choice, when compared with Partitions-of-Unity method, due to guaranteed favorable rates of convergence for large sampling gaps, $\delta = \sup_k(t_{n_k} - t_{n_{k-1}})$.

MATLAB was chosen as the primary development environment for the computations of the numerical simulations. The experiments were conducted on a machine running an Intel(R) Core(TM) i5-5200U CPU @ 2.20 GHz, 4GB installed RAM, and 64 bit operating system. The following two iterative reconstruction methods are compared against each other:

a. Adaptive Weights (AW) using weighted sinc functions as the chosen frame.

b. Partitions-of-Unity (PoU) using Gaussian functions as the chosen frame.

4.1 Signal Simulation

The performance of the AW and PoU iterative reconstruction algorithms was studied by recovering deterministic and randomly generated band-limited signals, with known spectrum. It was assumed that the spectrum or maximum frequency in the signals was available. The iterative methods were applied on recovering ten different deterministic and randomly generated signals. The following subsections provide information of how the signals were formed, however, for illustrative purposes, the analysis was conducted on one signal in each class.

4.1.1 Deterministic Signals

This work considered deterministic functions defined by the trigonometric Fourier series. The spectrum was given and defined the signals as a superposition of admissible frequencies with random amplitudes,

$$f(t) = a_0 + \sum_k (a_k \sin(k\omega_o t) + b_k \cos(k\omega_0 t)) \tag{4.1}$$

where

$$a_0 = \frac{1}{T}\int_0^T dt \quad f(t), \tag{4.2a}$$

$$a_k = \frac{2}{T}\int_0^T dt \quad f(t)\sin(k\omega_o t), \tag{4.2b}$$

$$b_k = \frac{2}{T}\int_0^T dt \quad f(t)\cos(k\omega_o t). \tag{4.2c}$$

A simple continuous-time signal defined in (4.3) was constructed and simulated to illustrate and analyse the performance of the investigated reconstruction methods on simple deterministic signals before moving onto more complicated one including stochastic features.

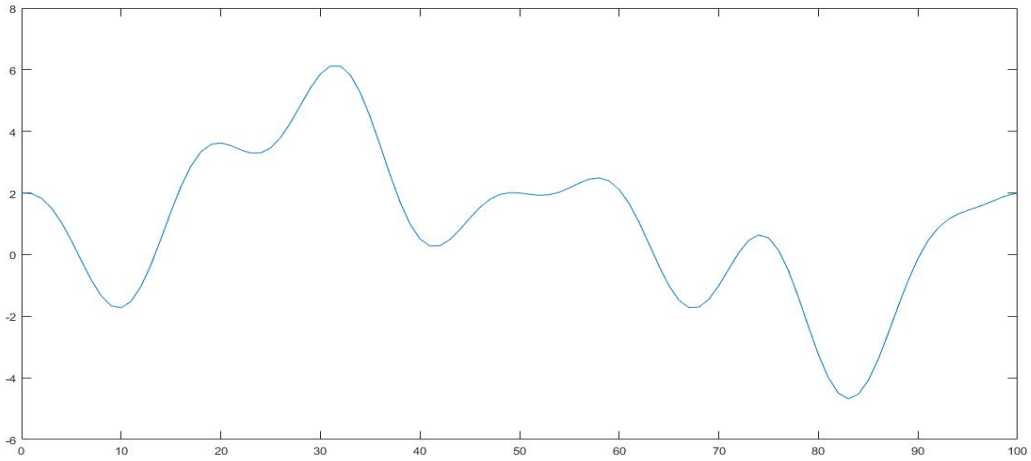

Figure 4.1: Deterministic signal, $f(t)$, used to analyse Adaptive Weights and Partition of Unity reconstruction methods.

To simulate and analyse the performance of the reconstruction methods on deterministic signal $f(t)$, the simulation environment, MATLAB, which is a discrete-time application, required the signal to be represented by a uniformly distributed discrete-time signal $\{f(t_n)\}$. By removing samples from the uniform grid, a sequence of non-uniform samples $\{f(t_{n_k})\}$ were formed. The reconstruction from non-uniform samples converted the non-uniform grid to a uniform one before MATLAB reconstructed the uniform samples to a continuous-time representation. MATLAB has its own choice of synthesis function $m(t)$ such that $\tilde{f}_u(t) = \sum_{n=1}^{N} f(t_n)m(t - t_n)$. The continuous-time signal

$$f(t) = 1 + \sqrt{5}\sin(\omega_0 t) - \cos(\omega_0 t) - \sqrt{\frac{3}{2}}\sin(3\omega_0 t) + \cos(3\omega_0 t) + \cos(4\omega_0 t) - \sin(5\omega_0 t) + \sin(7\omega_0 t) \tag{4.3}$$

was represented as a uniform discrete-time signal by a sequence of $N = 101$ uniform sampling time instants $t_n = 0, 1 \ldots, 100$ with amplitude values $\{f(t_n)\}$ determined by (4.3). The maximum frequency available in the signal was given as $\omega_m = 7\omega_o$, where $\omega_0 = 2\pi/N$ provided sufficiently fine timings such that the reconstruction from the samples $\{f(t_n)\}$, with MATLAB's internal approximation method, provided a good approximatiion of the continuous-time signal $f(t)$. The signal was assumed to be periodic, implying, that MATLAB's internal reconstruction from the uniform samples $\{f(t_n)\}$ approximated one period of $f(t)$.

4.1.2 Randomly Generated Signals

As already pointed out, this study focuses on the performance analysis of the investigated iterative reconstruction algorithms when applied to the recovery of band-limited signals. To simulate a randomly generated band-limited signal $f(t)$ as a discrete-time signal, a sequence of $N = 101$ uniform sampling time instants $t_n = 0, 1, \ldots, 100$ and samples $\{f(t_n)\}$ with random amplitude values lying between -1 and 1. The random signals were high frequency signals defined by the trigonometric Fourier series. In order to obtain a band-limited discrete signal, the sequence was filtered by a low-pass or bandpass filter with a specified bandwidth ω_m. By removing undesired frequencies, the filter formed different signals for the algorithms to reconstruct. The signals in Fig. 4.2 were formed by applying a low-pass filter at different cut-off frequencies.

.

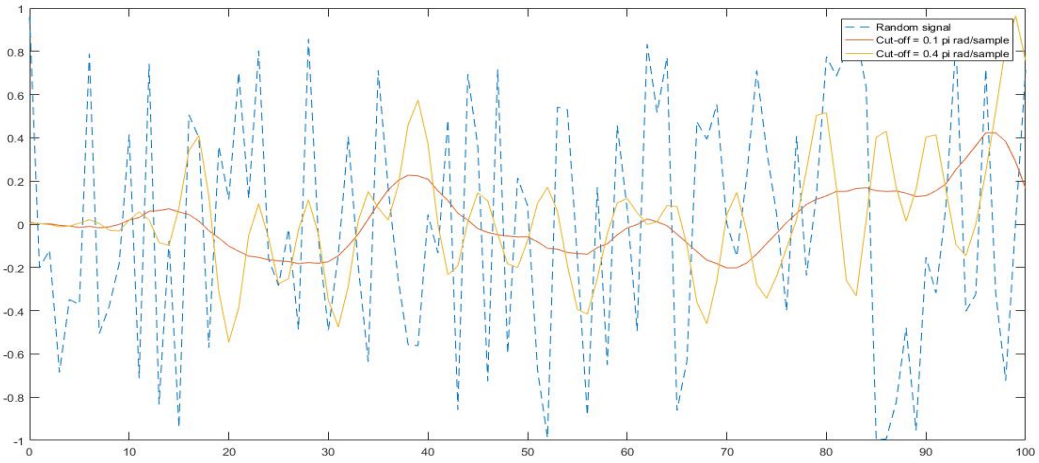

Figure 4.2: Randomly generated signal filtered at different cut-off frequencies

The low-pass filter created a smoothing effect. It produced slow changes in output values which made it easier to observe trends and boost the overall signal-to-noise ratio with minimal signal degradation. As expected the resulting signal is smoother, however, it lags behind due to the delay caused by filtering. This means that the filtered signal is shifted in time with respect to the original. Finite impulse response filters often delay all frequency components by the same amount. This property made it easy to correct for the delay by shifting the signal in time.

The filtered signal is defined by the trigonometric Fourier series; consequently, in addition to randomly generated signals, the reconstruction algorithms were tested on deterministic sinusoidal signals so that the performance of each algorithm could be more carefully evaluated. In studying deterministic input signals, the reference signal was $f(t)$ defined in (4.3).

4.1.3 Non-uniform Sampling

To formulate the irregular sampling problem of missing data on a uniform grid, it was assumed that 41 out of the 101 samples were lost and 60 were recovered. A random permutation of 60

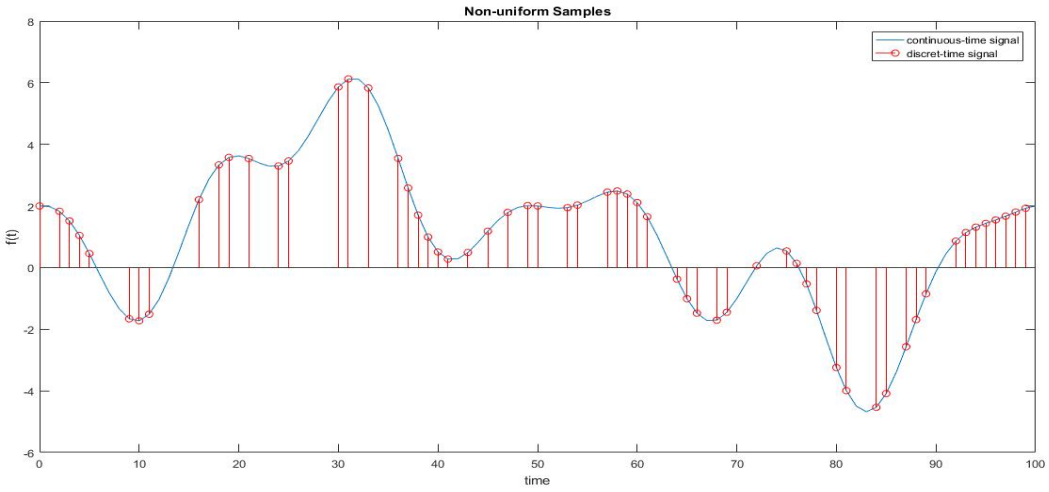

Figure 4.3: Non-uniformly sampled signal with maximum sampling gap $\delta = 5$.

integers between 0 and 100 with maximum gap, δ, were generated to form the non-uniform time grid $\{t_{n_k}\}$. The non-uniform sampling index was defined as the subset $1 \leq n_1 < n_2 < \cdots < n_{60} \leq 101$. Initially the maximum sampling gap was set as $\delta = 5$, before it was varied to evaluate its influence on the performance of the two algorithms. Fig. 7.2 shows the deterministic signal and its non-uniform sampled values, which are the samples from which the iterative reconstruction process had to start from.

4.2 Adaptive Weights

The general form of the iterative algorithm given by

$$f_0 = \mathcal{A}f \tag{4.4a}$$

$$f_i = f_{i-1} + \mathcal{A}(f - f_{i-1}) \tag{4.4b}$$

was used. To form the Adaptive Weights method, the approximation operator

$$\mathcal{A} = \sum_{j=1}^{r}\sum_{l=1}^{r} \mathcal{S}_{j,l}^{-1} |\phi_{n_j}(t) >< \phi_{n_l}(t)| \sum_{k=1}^{r} |\phi_{n_k}(t) >< s_{n_k}(t)|, \tag{4.5}$$

frame

$$\phi_{n_k}(t) = \phi(t - t_{n_k}) \tag{4.6a}$$

$$= \frac{\pi^2\sqrt{\Delta m_k}}{\pi^2 + \delta^2 + \omega_m^2} sinc(\omega_m[t - t_{n_k}]), \tag{4.6b}$$

and the auto-correlation matrix

$$S_{j,l} =< \phi_{n_j}|\phi_{n_l} >= \int_{\mathbb{R}} dt \quad \phi_{n_j}\phi_{n_l} \tag{4.7a}$$

$$= \sqrt{\Delta m_j \Delta m_l}(\frac{\pi^2}{\pi^2 + \delta^2 + \omega_m^2})^2 \int_{\mathbb{R}} dt \quad sinc(\omega_m[t - t_{n_j}])sinc(\omega_m[t - t_{n_l}]) \tag{4.7b}$$

$$= \sqrt{\Delta m_j \Delta m_l}(\frac{\pi^2}{\pi^2 + \delta^2 + \omega_m^2})^2 \int_{\mathbb{R}} dt \quad \frac{\sin(\omega_m[t - t_{n_j}])}{\omega_m[t - t_{n_j}]} \frac{\sin(\omega_m[t - t_{n_l}])}{\omega_m[t - t_{n_l}]} \tag{4.7c}$$

$$= \frac{\sqrt{\Delta m_j \Delta m_l}}{2\omega_m^2}(\frac{\pi^2}{\pi^2 + \delta^2 + \omega_m^2})^2 \int_{\mathbb{R}} dt \quad \frac{cos(\omega_m[t_{n_l} - t_{n_j}]) - cos(\omega_m[(t - t_{n_l}) + (t - t_{n_j})])}{[t - t_{n_j}][t - t_{n_l}]} \tag{4.7d}$$

$$= \frac{\sqrt{\Delta m_j \Delta m_l}}{2\omega_m^2}(\frac{\pi^2}{\pi^2 + \delta^2 + \omega_m^2})^2 \int_{\mathbb{R}} du \quad \frac{cos(\omega_m[t_{n_l} - t_{n_j}]) - cos(\omega_m[2u + t_{n_l} - t_{n_j}])}{u(u + t_{n_l} - t_{n_j})} \tag{4.7e}$$

$$= \frac{\sqrt{\Delta m_j \Delta m_l}}{2\omega_m^2}(\frac{\pi^2}{\pi^2 + \delta^2 + \omega_m^2})^2 \int_{\mathbb{R}} du \quad \frac{cos(\omega_m a_{j,l}) - cos(\omega_m[2u + a_{j,l}])}{u(u + a_{j,l})} \tag{4.7f}$$

were defined. The analysis function $s_{n_k(t)}$ was unknown and contained in the samples

$$f(t_{n_k}) =< s_{n_k}(t)|f(t) > . \tag{4.8}$$

The assumption $f \in \mathbb{V}_\phi \subset \mathbb{B}_\omega$, hence a superposition of the frame $\{\phi_{n_k}(t)\}$ and its samples, $\{f(t_{n_k})\}$, made the knowledge of the analysis function irrelevant.

4.2.1 Results

This section discusses the results obtained using the algorithm in (4.4). Fig 4.4 is the reconstructed function from the samples in Fig. 4.3, using the Adaptive Weights iterative reconstruction algorithm. The maximum sampling gap $\delta = 5$. The algorithm required the number of iterations to be known and set beforehand instead of an acceptable final precision.

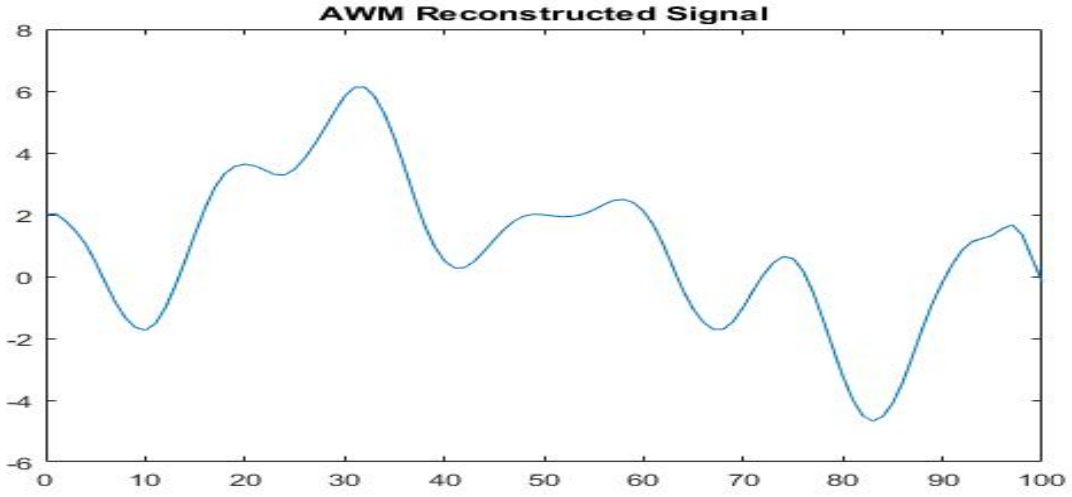

Figure 4.4: Adaptive weights iterative reconstruction of irregularly sampled signal with maximum gap $\delta = 5$.

This approach was taken because the algorithm may not converge to the specified error, especially for large sampling gaps, and may get stuck in an infinite loop. The number of iterations set a point were the algorithm breaks and for the user to evaluate and decide if there are more iterations needed. To obtain the reconstruction in Fig 4.4 and final error in Fig. 4.7, the algorithm ran for one hundred thousand (10^5) iterations in 223.1582 seconds.

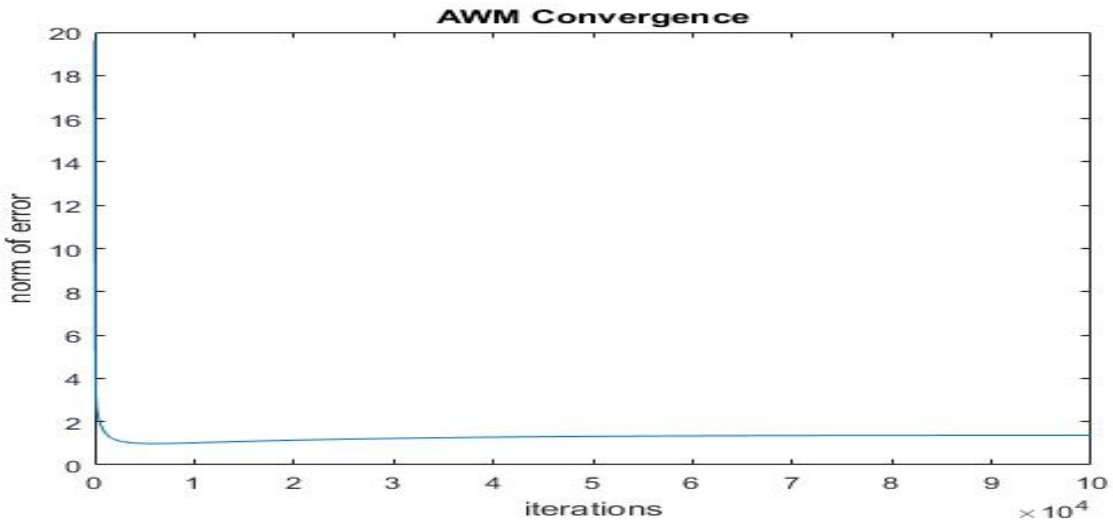

Figure 4.5: Convergence of the Adaptive Weights iterative reconstruction algorithm, $||f - f_i||$.

Fig 4.5 shows the norm convergence, $||f - f_i||$. In most cases the speed of convergence does not depend very much on the specific weights, $\pi^2/(\pi^2 + \delta^2 + \omega_m^2)\sqrt{\Delta m_k}$. Areas of high sampling density have smaller weights. Areas of low sampling density have larger weights adding too much weight, and a negative effect on the convergence. It is best to introduce a global relaxation parameter $0 < \alpha \leq 1$ creating the weight, $(\alpha\pi^2)/(\pi^2 + \delta^2 + \omega_m^2)\sqrt{\Delta m_k}$, which helps to compensate the irregularities of the sampling geometry and guarantee convergence. In this case $\alpha = 0.9$ because it is has the optimum effect on the convergence for the given sampling set, as illustrated in Fig 4.6 below.

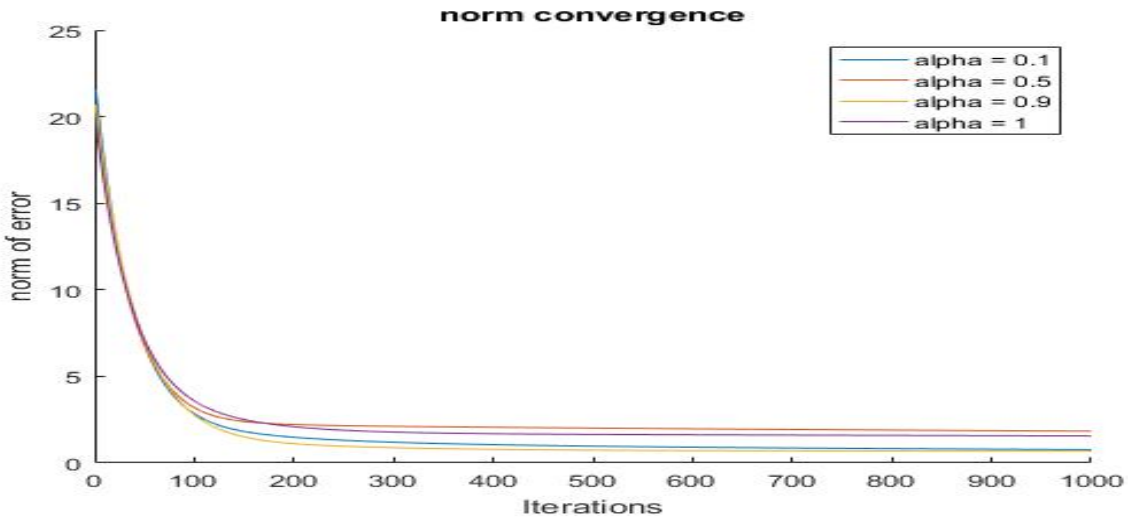

Figure 4.6: Effects of α on the convergence of the Adaptive Weights iterative reconstruction algorithm.

The influence of α is dependent on the sampling set. Therefore, the knowledge of the optimal relaxation parameter for this set cannot be used to make a guess for the new optimal relaxation

parameter given a new set. A good choice of the relaxation parameter often has a significant influence on the actual speed of convergence. From frame-theory, the optimal choice is

$$\alpha = \frac{2}{(A+B)} \tag{4.9}$$

given the frame bound constants A and B. In practice, A and B cannot be obtained numerically from the sampling geometry. The computational complexity for determining A and B is higher than that for solving the irregular sampling problem directly. Knowledge of the frame bounds only gives an optimal rate of convergence.. In general, it is not true that knowledge of them gives a better relaxation parameter. The optimal relaxation parameter for Adaptive Weights method is usually very close to one, often within 2 decimals.

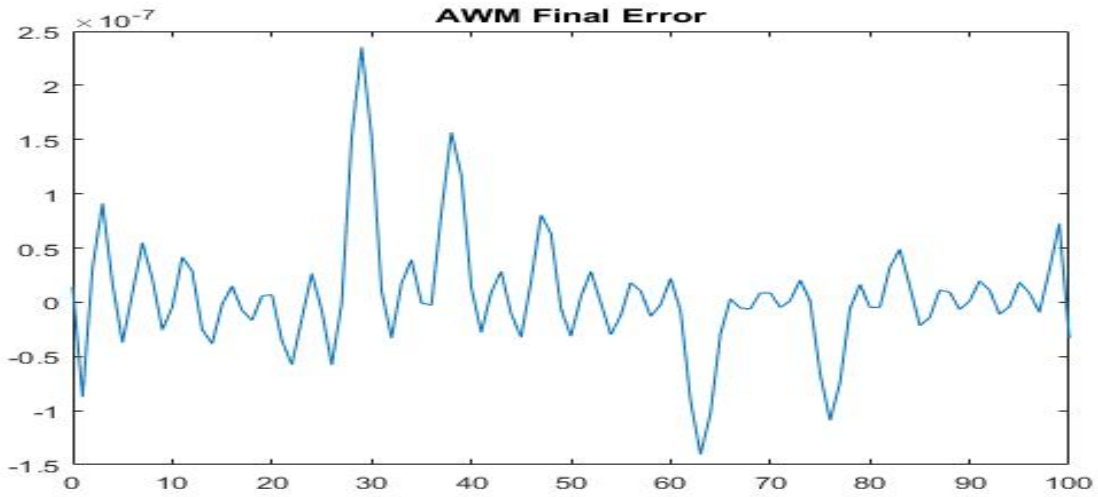

Figure 4.7: Adaptive Weights method's final error, $f - f_{10^5}$, after 10^5 iterations.

Since there is no efficient practical method knowing the optimal choice given by (4.9), it only provides help with theoretical analysis, practically, one has to grammatically vary α between zero and one, and manually choose from the results. Fig. 4.7 shows the final error after one hundred thousand iterations.

4.2.1.1 Effects of Maximum Sampling Gap

To test the effect of δ, the sampling gaps were varied from $\delta = 3$ to $\delta = 15$. Theorem 5.2 states that if $\delta < \pi/\omega_m$, then for any $f \in \mathbb{V}_\phi$ can be reconstructed, iteratively, by the algorithm in (4.4). When $N = 101$ and $\omega_m = 14\pi/N$, then $\delta < 7$ satisfies the sampling condition. However, as shown in Fig 4.8, the algorithm still converges for $\delta > 7$ and was observed to be, seemingly, independent on the sampling gap. However, in practice any method under consideration will

diverge if the maximum sampling gap is too large. Very small relaxation parameters will in principle enforce convergence, but at a rate which is too poor for practical consideration.

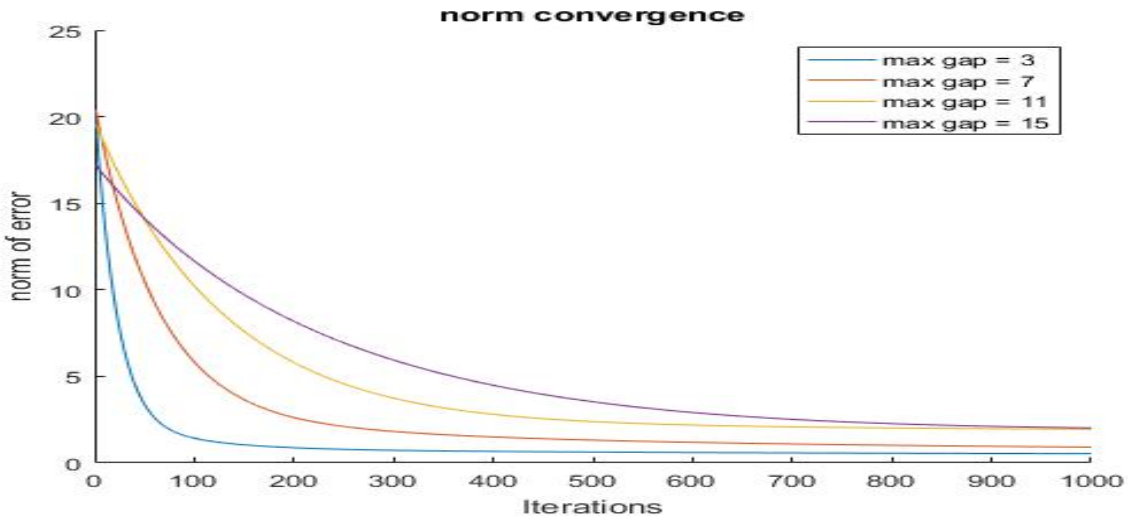

Figure 4.8: Effects of maximum gap on the convergence of the Adaptive Weights iterative algorithm

Of course, there are many other geometric features that a sampling set may have. For example, the degree of irregularity plays an important role. A sampling family may have slowly varying sampling density or set with almost uniform density, but spoiled by few sparse points - regions of lower density. In any case the quality of an algorithm should include statements about its sensitivity with respect to those different forms of irregularity. Adaptive Weights is less sensitive to this geometric irregularity compared to Partitions-of-Unity because its relaxation parameter compensates the irregularities of the sampling geometry.

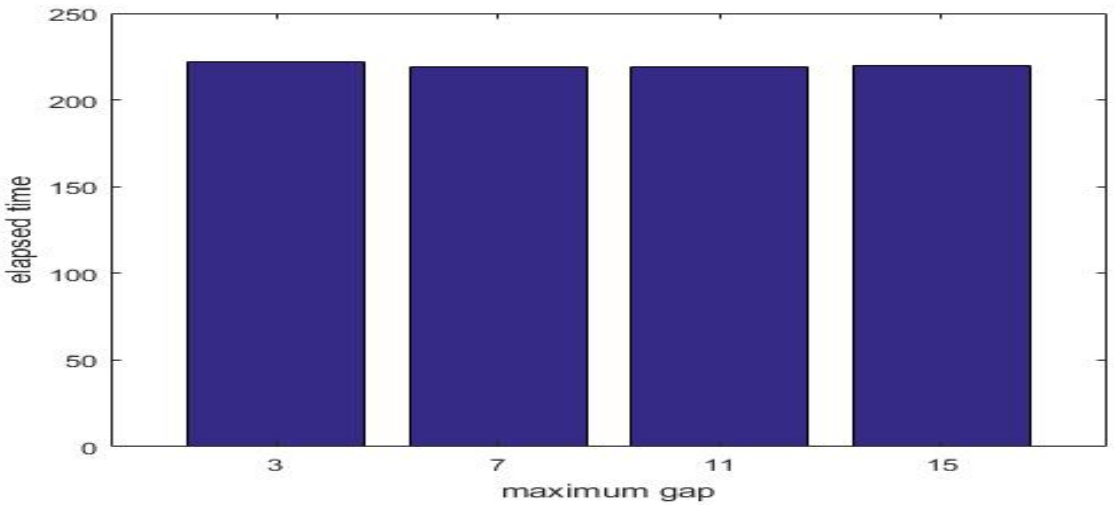

Figure 4.9: Effects of maximum gap on the computational time for the Adaptive Weights iterative algorithm

It was not surprising that the size of the maximal gap in a sampling set had considerable influence on the norm convergence, hence, time to reach a certain precision. However, the time to compute a number of iterations, ten thousand (10^4) in this scenario, was not considerably influenced by the maximum sampling gap because the number of samples computed were the same.

4.3 Partitions-of-Unity Method

The Partitions-of-Unity is a variation of the Adaptive Weights method and its algorithm has the same general form in Eq. (4.4). They are differentiated by the frame of choice because of its innate properties. Partitions-of-Unity play an important role as amplitude-preserving windows. A frame that was used to form a partition of unity is a Gaussian function of the form

$$\phi(t - t_{n_k}) = \frac{\Delta t_{n_k}}{\sigma\sqrt{\pi}} e^{-\frac{(t-t_{n_k})^2}{\sigma^2}} \tag{4.10}$$

where $\Delta t_{n_k} = t_{n_k} - t_{n_{k-1}}$ is the separation between samples and σ is the standard deviation of the samples $\{f(t_{n_k})\}$. The Gaussian is infinitely differentiable and has optimal resolution properties in the time, spatial and Fourier domains [6]. Although it was used to form a partition of unity it does not have a compact support, it has support on all of $\mathbb{R}$, nor yield an exact partition of unity.

$$\sum_k \phi_{n_k} = 1 + 2\cos(\frac{2\pi t}{\Delta t_{n_k}}) e^{(\frac{\pi\sigma}{\Delta t_{n_k}})^2} \tag{4.11}$$

The deviation from 1 is approximately 10^{-4} when $\Delta t_{n_k} = \sigma$. Numerically, the best approximation of a partition of unity, illustrated in Fig. 4.10, was achieved when the samples were on a uniform grid.

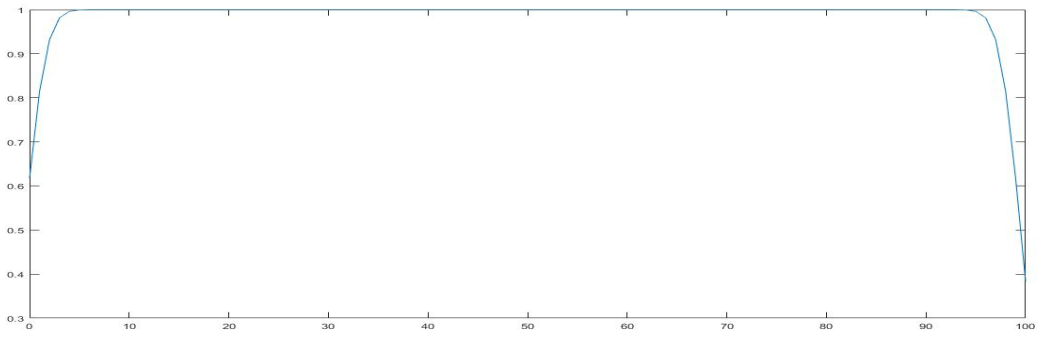

Figure 4.10: Partition of unity generated by the Gaussian for regularly spaced samples.

The approximated partition of unity by a Gaussian frame had roll-ff's on the sides in the domain of interest. Roll-off effects occured because a finite number of Gaussians were used. It is possible to eliminate them by having, on both sides, more Gaussians extending outside the domain of interest. The choice of a Gaussian allowed the auto-correlation matrix to be represented in closed form, eliminating the need to numerically compute integrals, hence significantly improved the numerical efficiency of the algorithm.

$$\mathcal{S}_{k,l} =< \phi_{n_j} | \phi_{n_l} >= \int_{\mathbb{R}} dx \quad \phi(t - t_{n_j})\phi(t - t_{n_l}) \tag{4.12a}$$

$$= \int_{\mathbb{R}} dx \quad \frac{\Delta t_{n_k}}{\sigma\sqrt{\pi}} e^{-\frac{(t-t_{n_j})^2}{\sigma^2}} \frac{\Delta t_{n_l}}{\sigma\sqrt{\pi}} e^{-\frac{(t-t_{n_l})^2}{\sigma^2}} \tag{4.12b}$$

$$= \frac{\Delta t_{n_j} \Delta t_{n_l}}{\pi\sigma^2} \int_{\mathbb{R}} dx \quad e^{-\frac{(t-t_{n_j})^2}{\sigma^2}} e^{-\frac{(t-t_{n_l})^2}{\sigma^2}} \tag{4.12c}$$

$$= \frac{\Delta t_{n_j} \Delta t_{n_l}}{\pi\sigma^2} \int_{\mathbb{R}} dx \quad e^{-\frac{(t-t_{n_j})^2+(t-t_{n_l})^2}{\sigma^2}} \tag{4.12d}$$

$$= \frac{\Delta t_{n_j} \Delta t_{n_l}}{\pi\sigma^2} \int_{\mathbb{R}} dx \quad e^{-\frac{2x^2-2x(t_{n_j}+t_{n_l})+(n_k^2+n_l^2)}{\sigma^2}} \tag{4.12e}$$

$$= \frac{\Delta t_{n_j} \Delta t_{n_l}}{\pi\sigma^2} e^{-\frac{t_{n_j}^2+t_{n_l}^2}{\sigma^2}} \int_{\mathbb{R}} dx \quad e^{-\frac{2x^2-2x(t_{n_j}+t_{n_l})}{\sigma^2}} \tag{4.12f}$$

$$= \frac{\Delta t_{n_j} \Delta t_{n_l}}{\pi\sigma^2} e^{-\frac{t_{n_j}^2+t_{n_l}^2}{\sigma^2}} \int_{\mathbb{R}} dx \quad e^{-2\frac{(t-\frac{1}{2}(t_{n_j}+t_{n_l}))^2-(\frac{1}{2}(t_{n_j}+t_{n_l}))^2}{\sigma^2}} \tag{4.12g}$$

$$= \frac{\Delta t_{n_j} \Delta t_{n_l}}{\pi\sigma^2} e^{-\frac{1}{\sigma^2}[t_{n_j}^2+t_{n_l}^2-2(\frac{1}{2}(t_{n_j}+t_{n_l}))^2]} \int_{\mathbb{R}} dx \quad e^{-[\frac{\sqrt{2}}{\sigma}t-\frac{\sqrt{2}}{2\sigma}(t_{n_j}+t_{n_l})]^2} \tag{4.12h}$$

$$= \frac{\Delta t_{n_j} \Delta t_{n_l}}{\sigma\sqrt{2\pi}} e^{-\frac{(t_{n_j}-t_{n_l})^2}{2\sigma^2}} \tag{4.12i}$$

4.3.1 Results

This section discusses the results obtained using the algorithm in (4.6)-(4.7) with the Gaussian selected as a frame of the shift invariant space $\mathbb{V}_\phi$ and the approximation operator $\mathcal{A}$ defined in (4.8).

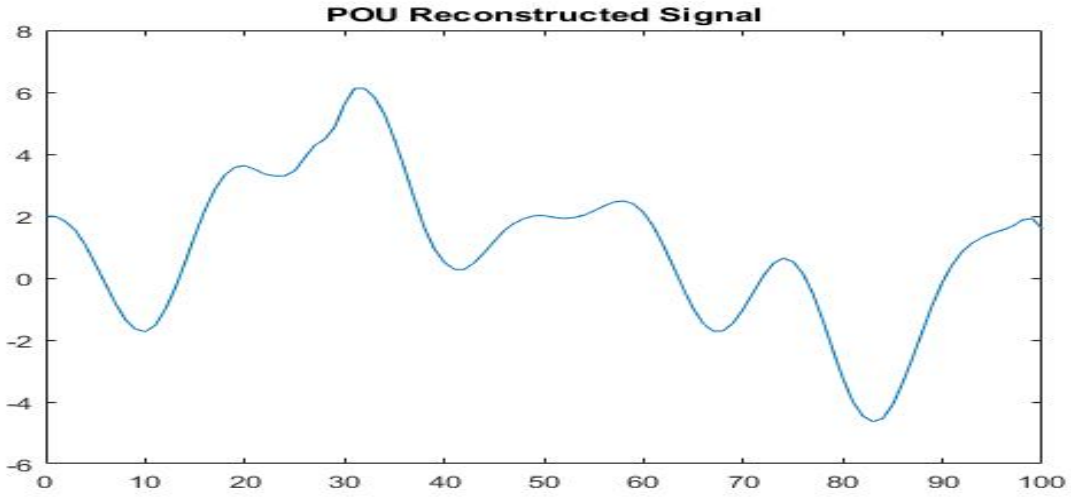

Figure 4.11: Adaptive weights iterative reconstruction of irregularly sampled signal with maximum gap $\delta = 5$.

The reconstructed function, illustrated in Fig 4.11, was reconstructed over one hundred thousand iterations in 8.1658 seconds. The algorithm was quite fast and sufficiently accurate due to the fact the choice of a Gaussian enabled a closed-form representation and easy computations of the autocorrelation matrix elements and that Partitions-of-Unity preserve amplitudes. However, it was observed that this method was sensitive to irregularities in the sampling geometry and was apparent in this example as it was unable to adequetly approximate the curve between 20 and 30 seconds. The sensitivity was analysed by varying the maximum sampling gap.

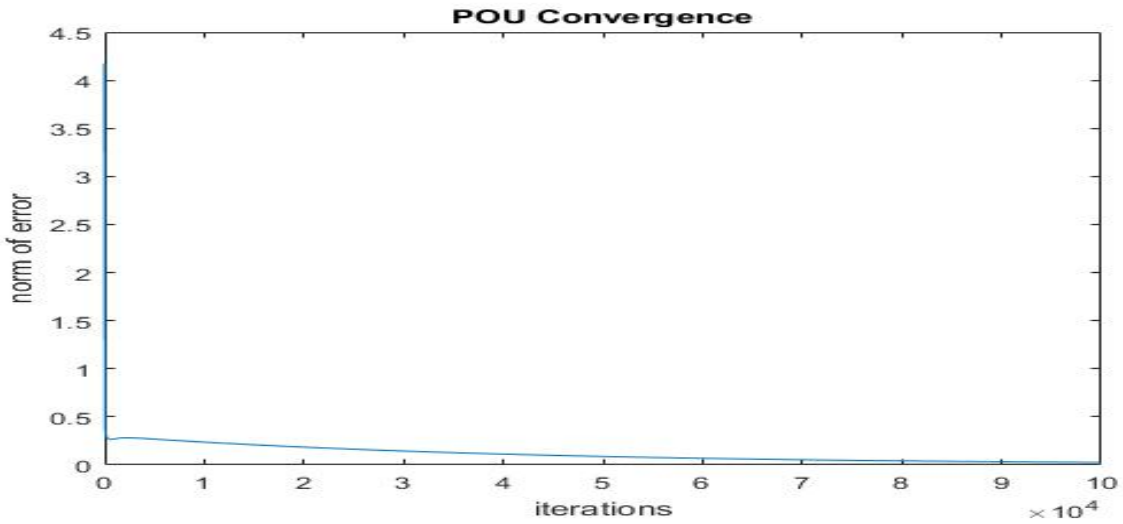

Figure 4.12: Convergence of the Partitions-of-Unity's iterative reconstruction algorithm, $||f - f_i||$.

Fig 4.12 shows the convergence of the norm, which had a sharp decay for the first few iterations, attributed to the amplitude preserving window formed by Partitions-of-Unity. It had a better initial approximation, f_0, than the adaptive weights, clearly, demonstrating its, superior, amplitude preserving ability.

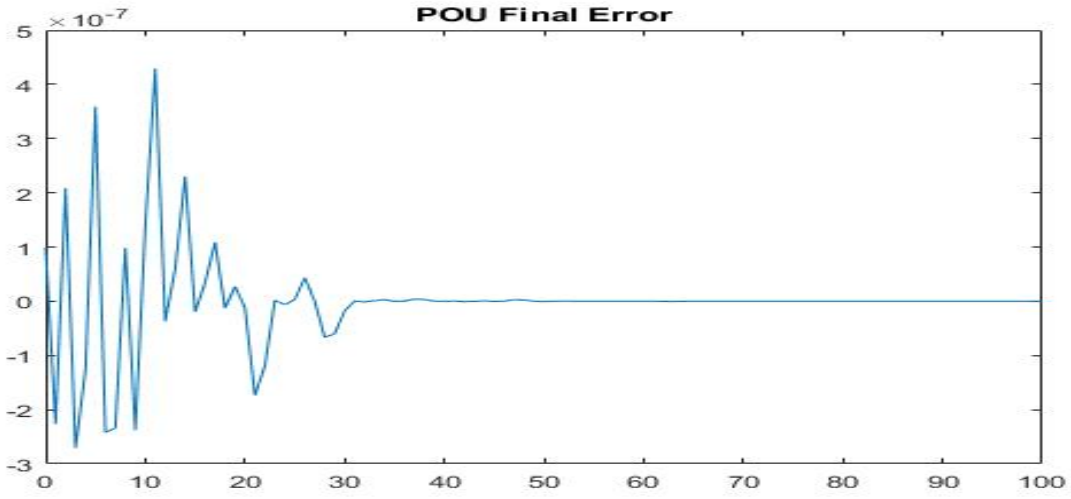

Figure 4.13: Partitions-of-Unity method's final error, $f - f_{10^5}$ after 10^5 iterations.

The final error and the convergence rate of the norm depend on the given non-uniform samples; the PoU method is efficient at approximating the given samples, however, its sensitivity to irregularities in sampling geometry caused it to poorly approximate curves in regions of low sampling density.

4.3.1.1 Effects of Maximum Sampling Gap

It is of great practical importance to know to which extent the sampling geometry has an influence on the performance of a given method. To test the effect of δ, the sampling gaps was varied from $\delta = 3$ to $\delta = 11$.

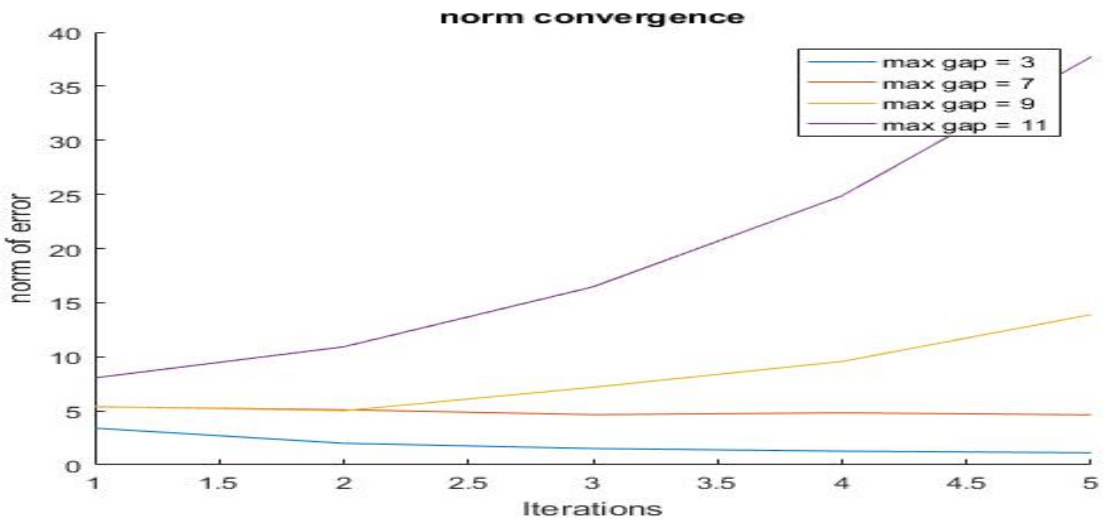

Figure 4.14: Effects of maximum gap on the convergence of the Partition-of-Unity iterative algorithm

As mentioned before, PoU is a variation of AWM and also requires $\delta < \pi/\omega_m$, for any $f \in \mathbb{V}_\phi$ to be reconstructed, iteratively. When $N = 101$ and $\omega_m = 14\pi/N$, then $\delta < 7$ satisfies the sampling condition. This method confirms the theoretical prediction as the algorithm diverges when $\delta > 7$, proving that its performance dependents on the irregularities in the sampling grid.

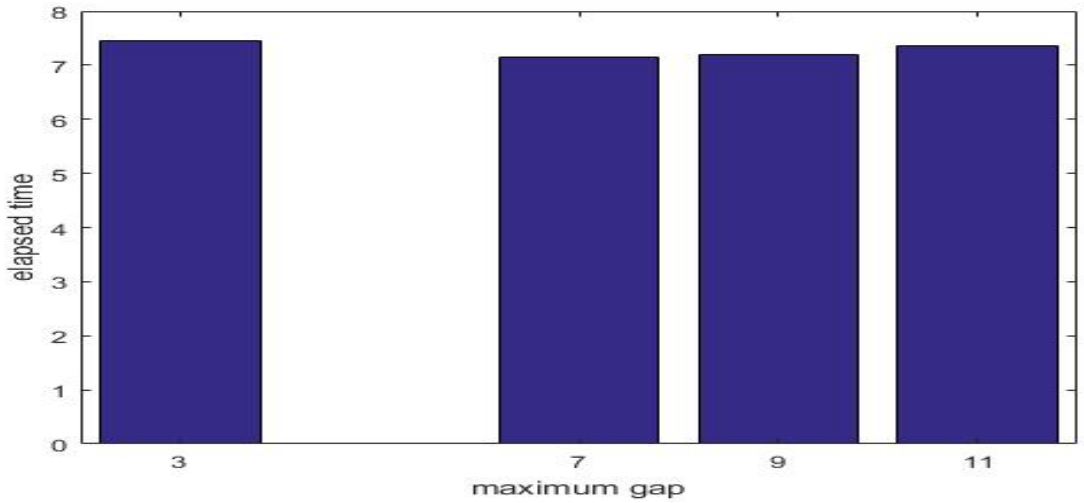

Figure 4.15: Effects of maximum gap on the computational time for the Partition of Unity iterative algortihm

This is no surprise if the choice of a relaxation parameter is understood as the choice of a constant weight. Therefore, it cannot compensate the irregularities of the sampling set as the Adaptive Weights method does.

4.4 Comparison of Methods

The iterative methods were implemented on the filtered random signals to observe their consistency and perform a comparison between the methods. Although, the filtered signal was shifted in time, that didn't effect the algorithms performance because the concern was can the algorithms determine the filtered signal. For all the signals the maximum gap was kept at $\delta = 5$. Plots such as those in Figs. 4.7, 4.14, and 4.16, allow one to read off easily how many iterations or how many units of time and maximum sampling gap are needed in order to reach a certain precision with a given method. This is useful for practical applications to fairly determine the efficiency of an algorithm. Typically, after an initial phase of few iterations the algorimths have steeper decay and later approach a certain final error which is dependent on the maximum sampling gap. In any case the actual rate of convergence and computational time of PoU was superior to that of AWM

If efficiency is defined as the norm convergence and computational time of an algorithm, then among the two methods, discussed, the Partitions-of-Unity method is more efficient. The Adaptive Weights method is slow and converged to a higher error than that of the Partitions-of-Unity. One notable reason that affected the convergence rate and computational time was the fact that this author was unable to compute and represent the autocorrelation matrix elements in closed-form. The numerical accuracy and computational time were affected by MATLAB's integral function and rounding off errors found in computing the inverse of the matrix. However, the Adaptive Weights method compensates for slowness and less accuracy by being convergent and robust for large sampling gaps and less sensitive to the irregularities

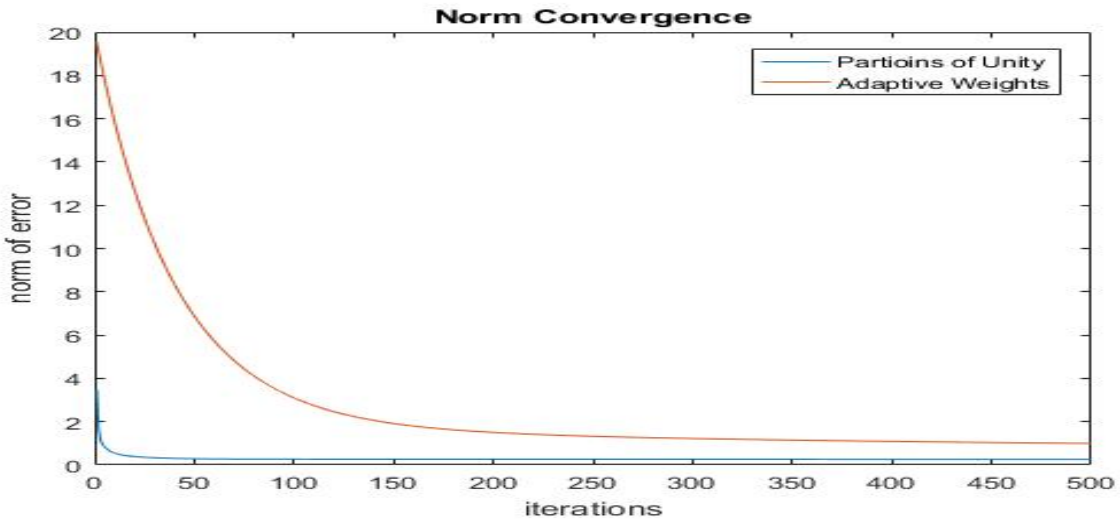

Figure 4.16: Comparison of convergence rate

The Partitions-of-Unity method tends to be more sensitive towards irregularities of the sampling geometry. The Adaptive Weights method is less sensitive and has robust approximations for larger sampling gaps. In the presence of additive Gaussian noise the problem can still be viewed as reconstructed a randomly generated signal at a lower frequency. Noise is a high frequency randomly generated signal added to a baseband signal of bandwidth ω_m. When the cut-off frequency of a low pass filter is set to ω_m, then the result is time-shifted response of the baseband signal with amplitude deviation due to attenuation. The accuracy of the filter also depends on the number of samples and distribution, uniform samples provided better filtered approximation.

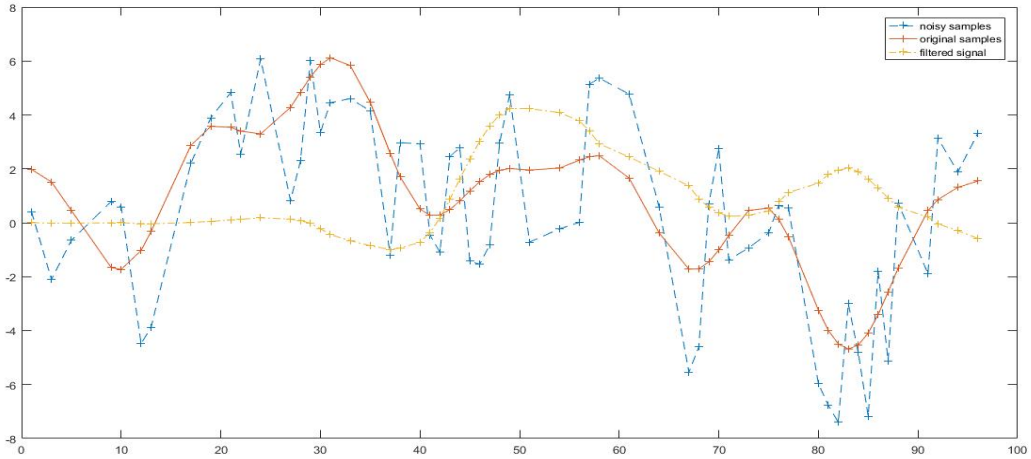

Figure 4.17: Baseband signal corrupted with noise

The algorithms work with the available irregular samples, in Fig. 4.17, of the filtered signal. Additionally, the time-shift delay was introduced by the filter. To rectify this, the delay was

eliminated. Removing the delay required the samples denoting the delay to be removed. Since the signal was represented by finite samples the reconstruction was performed on incomplete or truncated signals.

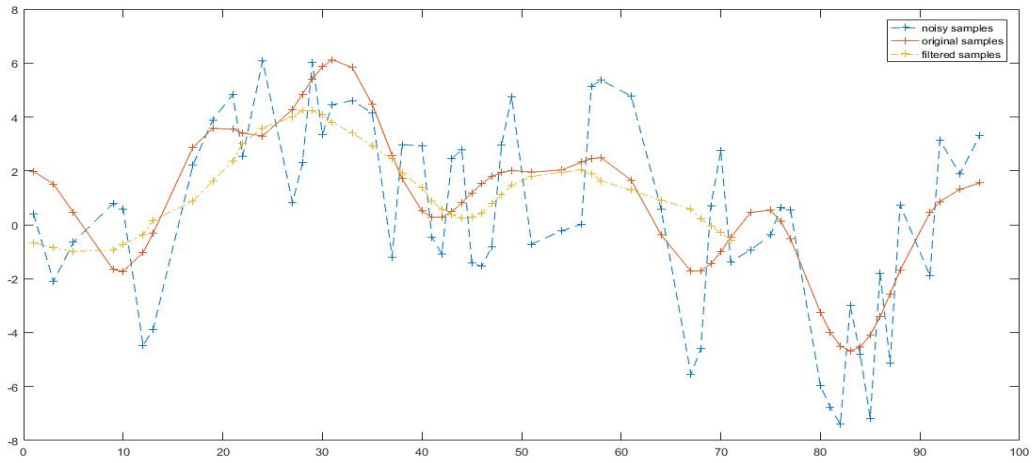

Figure 4.18: Removed delay from the filtered signal

As illustrated in Fig. 4.18 after the removal of the delay the signal is truncated. Filtering using non-uniform samples produced a smooth function which, slightly, differs from the original signal. Evidently, there is an notable error between the filtered and original signals. The algorithms reconstructed the filtered signals with an innate error; therefore, even if, in some cases, the final error of the approximation was less than the approximation of the original, there was a comparatively large error, and thus, a significant difference between the original and approximation.

Chapter 5

Conclusion and Future Work

Sampling is the step to preserve, retrieve and extract information embedded in the continuous signals of interest. The Shannon-Nyquist sampling theorem provided a theoretical bound as how fast a continuous-time (CT) signal must be uniformly, so that information is not lost in the sampling process. This thesis, however, is dedicated to the exploration of ways to break through the minimum sampling frequency by applying non-uniform sampling and reconstruction techniques. Based on the theoretical framework, the thesis viewed three random sampling methods Additive Random Sampling (ARS), Missing Data (MD), and Stochastic Jittered Sampling (SJS) and proposed two iterative reconstruction methods, Partitions-of-Unity (PoU) and Adaptive Weights (AW), to solve the MD non-uniform sampling problem.

One of the surprising observations made was the efficiency of the PoU method in terms of computational time to perform iterations, compared against the AW method. These results can be used to calculate the number of iterations which will be sufficient to obtain a given precision. The results of the PoU method confirmed the theoretical convergence rates of the norm of the error and restrictions on the maximum sampling gap. For the AW method the situation is quite different, and numerical convergence curves are substantially better than the predicted ones for maximum sampling gaps larger than the theoretical limit. Whether this is rather a statistical phenomenon dependent on the non-uniform grids used to test the algorithm such that for a high percentage of cases this is true and for few exceptional cases abide by the theoretical limit, or rather the theory has to be adjusted to actual observations, is a question that has to be analyzed in the future.

The description of the AW method includes the choice of weights, defined as the square root of half the distance between the left and the right neighbour. The speed of convergence is not highly effected by the chosen weights as long as they compensate the irregularities of the sampling geometry. In areas of higher sampling density, the weights are small and in areas of

lower sampling density the weight is big having a negative effect on irregularity compensation. Hence, in this research a relaxation parameter less than one was introduced to compensate for irregularities and guarantee convergence. In general it is true that a small relaxation parameter improves stability of the method, and helps to overcome situations which are somewhat more irregular, at the cost of speed. However, optimal relaxation parameters in the case of AW are very close to one, hence, 0.9 gave the best result from the tested set, because the small weights for areas of high sampling density compensate for the irregularities in the sampling geometry, automatically. However, instead of combining a weight sequence with a relaxation parameter, as done here, one could determine the maximum weight that all weights have to be below or equal to. Another way would be to ensure the sum of the weights is equal to the length of the signal.

There is sufficient evidence, in the current implementation, that the PoU method is faster and more precise than the AW method when the maximal sampling gaps are, $\delta < \pi/\omega_m$. The PoU method tends to be more sensitive towards irregularities of the sampling geometry than AW. Although slower, the AW outperforms PoU for cases where the maximal sampling gaps are greater than the theoretical limit, merely by being convergent because PoU is divergent. Certainly, if efficiency is viewed as computational time and accuracy without considering the effects of maximal gaps, then the PoU is the method of choice. However, in practice there's no particular way of limiting the number of consecutively missing samples on a uniform grid. Therefore, when the effects of maximal sampling gaps is considered, the Adaptive Weights is the preferable method. To improve the AW method's computational time and accuracy it is recommended that the closed form for the computation of the elements in the auto-correlation matrix is determined. This will, certainly, lead to a much more fair and accurate assessment and comparison of the two methods because there would be then a high level of similarity in implementation.

Future Work

Missing data sampling is the case when a uniform sampling sequence is distorted with missing samples. Therefore, the distribution of the sampling set $\{t_{n_k}\}$ is almost uniform. It is worth investigating other distributions which are far from uniform, example ARS where the distribution is exponential.When only looking at the convergence behavior in simulations, the AW method seems not to have a maximal sampling gap, although theoretically all algorithms do have a maximum sampling gap which if exceeded the algorithm will diverge. Therefore, it is worthwhile to investigate the true maximal gap of the AW method, or expand on the theoretical limit.

Appendix A

Source Code

A.1 Signal Simulation

Deterministic Signal

To have a controlled assessment of the algorithms deterministic functions were formed. The number of uniform samples were assumed to be given as $N_c = 101$.

```
% ========================================================Creation of function
Nc = 101;                           % Number of expected uniform samples
Nd = 0.6*Nc;                        % Number of recovered irregular samples
t = linspace(0,100,Nc);             % time axis
w0 = 2*pi/Nc;
f = 1 + sqrt(5)*sin(w0*t) - sqrt(3/2)*sin(3*w0*t) - sin(5*w0*t) + sin(7*w0*t)...
    - cos(w0*t) + cos(4*w0*t) + cos(3*w0*t);
wm = 7*w0;                          % maximum frequency
```

To study the effects of noise the deterministic or randomly generated function was corrupted by additive Gaussian noise generated by the MATLAB function **wgn(**m, n,power**)**.The function generates an m by n matrix of white Gaussian noise with a specified power in decibels relative to a watt. The default load impedence is 1 ohm. Therefore the code snippet generated a 101 element row vector containing real white Gaussian noise of power 0 dBW added to the signal of interest.

```
noise = wgn(1,length(g),0);             % additive white gaussian noise
fn = f + noise;
```

Randomly Generated

To test the algorithms on non predetermined signals such that the test signals are vast in form, randomly generated signals were constructed. The signals were generated randomly within a specific range. The code created a 101 element row vector of random floating numbers drawn from a uniform distribution in the open interval (-1,1).

```
% Set range for sample values
a = -1;                                 % lower bound
b = 1;                                  % upper bound
r = (b-a).*rand(N,1) + a;               % randomly generated signal
r_range = [min(r) max(r)];              % global min and max
```

By default the MATLAB function **rand()** returned normalized values between zero and one that are drawn from a uniform distribution. To change the range to (a, b), each value was multiplied by the width of the new range, $(b - a)$, and then shifted by a.

Filter and Delay Correction

The non-uniformly sampled randomly generated and noisy signals were filtered by a low pass filter for the algorithms to reconstruct low frequency signals. The finite impulse response (FIR) filter was used. The advantages of FIR is that they can have exactly linear phase, always stable, efficiently realizable in hardware, and the startup transients have finite duration. One notable disadvantage is that they require a much higher order than infinite impulse response (IIR) filters. The filter function **fir1()** was chosen, it applied a window to the truncated inverse Fourier transform of the chosen 34th order Chebyshev low pass filter. A cut off frequency of $\omega_m = \frac{14}{N}\pi$ rad/sample and a Chebyshev window with 30 dB of ripple was implemented, where $N = 101$.

```
% create a low pass filter
fc = 2*7/N;                             % cut off frequency [pi rad/sec]
lpf = firl(34,fc,chebwin(35,30));       % Low Pass Filter
% apply low pass filter
rf = filter(lpf,1,r);                   % filtered random signal
```

Filtering a signal introduces a delay. This means that the filtered signal is shifted in time with respect to the input. The snippet of code below is how the delay was counteracted. The delay for FIR filters is equal and constant over the frequency band. This makes it easier to counteract. For an $n - th$ order FIR filter, the delay is $\frac{n}{2}$, therefore in this example it was 17 time instants.

```
% Correct Filter time-shift delay
delay = mean(grpdelay(lpf));              % determine delay
td = t(1:end-delay);                      % revised time interval
rfd = rf;
rfd(1:delay)=[];                          % non delayed filtered signal
```

Nonuniform Sampling

The purpose of this work was to investigate algorithms that can efficiently reconstruct a signal from its non-uniform samples. Below is a source code snippet that randomly removed samples on a uniform grid to form a non-uniform one. The while loop was used to control the maximum sampling gap. The loop was commented out to have random maximum gaps. The maximum gaps were not arbitrary because of the number of non-uniform samples that were assumed to be retained.

```
% ========================================Sampling gap and Sampled function
n = sort(randperm(Nc,Nd));        % irregular sampling index
tn = t(n);                        % irregualr time instants
di = max(diff(tn));               % maximum difference between irregular samples
dr = max(diff(t));                % maximum difference between regular samples

% Set maximum gap
while di~=5
    n = sort(randperm(Nc,Nd));        % sampling index
    tn = t(n);                        % sampling points
    di = max(diff(tn));
end
% sample function
fs = f(n);                            % irregular sample values
```

A.2 Iterative Methods

The algorithms required the number of iterations to be specified, therefore one-hundred thousand iterations were initialized. The number of iterations needed were chosen iteratively from viewing the final error and the smoothness of the approximated function when the iterations were lower.

```
itr = 100000;                     % number of iterations
```

A.2.1 Adaptive Weights

Interpolation

The first step for the algorithms was to interpolate the given samples with the chosen synthesis function ϕ

```
w_Qxf =  w_Qx(fs,t,tn,di,dr,wm,Nc,Nd);              %1st Interpolation
```

Below is the created function to interpolate given samples using Adaptive Weights method. Firstly, the function determined the adaptive weight for each sample before multiplying it with the sinc function and summing all elements.

```
function [ Qf ] = w_Qx(fn,x,xn,di,dr,wm,Nc,Nd,alpha)
% Computation of Quasi Interpolation
Qxf = 0;  % Initialize
for k = 1:length(xn)-1

    % form adaptive weights
    if k==1
        dmk = 0.5* ( dr + x(Nc)- x(Nd)+ xn(2) );
    end
    if k>1
        dmk = 0.5*(xn(k+1)-xn(k-1));
    end
    % Interpolation
        Qxf = fn(k)*sqrt(dmk)*(wm/pi)*sinc(wm*(x-xn(k)))+ Qxf;
end
Qf = alpha*pi^2/(pi^2+di^2+wm^2)*Qxf; % complete interpolated signal
end
```

Auto-corrolation Matrix

The algorithm required the orthogonal projection to be applied on the interpolated result. However, the orthogonal projector required the inverse matrix of the autocorrelation matrix.

```
[w_A,w_iA] = w_Akl(tn,wm,dr,di,Nc,Nd,t);            %Autocorrelation Matrix
```

Below is the created function to compute the autocorrelation matrix and its inverse.

```
for k = 1:length(xn)-1
    % form the adaptive weights
    if k==1
        dmk = 0.5* ( dr + x(Nc)- x(Nd)+ xn(2) );
    end
    if k>1
        dmk = 0.5*(xn(k+1)-xn(k-1));
    end
    for l=1:length(xn)-1
    if l==1
        dml = 0.5* ( dr + x(Nc)- x(Nd)+ xn(2) );
    end
    if l>1
        dml = 0.5*(xn(l+1)-xn(l-1));
    end
    %compute integral to determine matrix elements
       f = @(x) sqrt(dmk*dml)*(cos( pi*wm*( xn(l)-xn(k) ) ) - cos(pi*wm* (2.*x-( xn(l)+xn(k) ) ))./...
          ( (2.*x-( xn(l)+xn(k) ) ).^2 - ( xn(l)-xn(k) ).^2);
       B(k,l)= integral(f,-inf,inf);                % form matrix
    end
end
A = 2/(pi^3)*(alpha*pi^2/(pi^2+di^2+wm^2)).^2*B;    % auto-corrolation matrix
iA = inv(A);                                        % inverted matrix
```

Orthogonal Projection

When the inverse of the autocorrelation matrix was complete the orthogonal projection could be performed in the two steps below. The complete projection and approximation operator is defined by `w_E`.

```
%orthogonal Projection
w_D = w_Dx(w_Qxf,tn,t,Nc,Nd,wm,dr,di);
w_E = w_D*w_iA;
```

Below is the function created to compute `w_D`.

```
function [ D ] = w_Dx(Qxf,xn,x,Nc,Nd,wm,dr,di,alpha)

for k=1:length(xn)-1
    if k==1
        dmk = 0.5* ( dr + x(Nc)- x(Nd)+ xn(2) );
    end
    if k>1
        dmk = 0.5*(xn(k+1)-xn(k-1));
    end
    D(1,k)  = dot( Qxf,(alpha*pi^2/(pi^2+di^2+wm^2))*sqrt(dmk)*(wm/pi)*sinc(wm*(x-xn(k))) );

end
```

Approximation

When the approximation operator was ready it was called in the main function as follows to form an approximated function.

```
% first approximation
w_f1 = w_apprx( w_E,tn,t,Nc,Nd,wm,dr,di );       %
w_fx = w_f1;
```

Below is the function created to compute the approximation of the function. It took the approximation operator `w_E` and applied it to samples.

```
function [ fx ] = w_apprx( E,xn,x,Nc,Nd,wm,dr,di)
fx=0;
% Approximate function 'f(x)'
for k= 1:length(xn)-1
    if k==1
        dmk = 0.5* ( dr + x(Nc)- x(Nd)+ xn(2) );
    end
    if k>1
        dmk = 0.5*(xn(k+1)-xn(k-1));
    end

     fx = E(k)*(0.9*pi^2/(pi^2+di^2+wm^2))*sqrt(dmk)*(wm/pi)*sinc(wm*(x-xn(k))) + fx;
end
end
```

Iterations

The snippet, below, performed the one-hundred thousand iterations. All of the functions defined in the previous sections were used.

```
%for loop iterations
for r =1:itr
    w_ex = f-w_fx;
    w_Qxe = w_Qx(w_ex(n),t,tn,di,dr,wm,Nc,Nd);
    w_xi(1,r) = r;
    w_conv(1,r) = norm(w_ex(n));
    %orthogonal Projection
    w_D = w_Dx(w_Qxe,tn,t,Nc,Nd,wm,dr,di);
    w_Ex = w_D*w_iA;
    w_ex = w_apprx( w_Ex,tn,t,Nc,Nd,wm,dr,di );
    %approximation
    w_fx = w_ex + w_fx;
end
```

A.2.2 Partitions of Unity

This method required the standard deviation of the non-uniform samples to be determined because it was a parameter of the Gaussian synthesis function.

```
std_fs = std(fs);                    % Standard Deviation of the samples
```

Interpolation

The first step for the algorithms was to interpolate the given samples with the chosen synthesis function ϕ

```
p_Qxf = Qx(fs,std_fs,t,tn,n);        % 1st Quasi-Interpolation
```

Below is the created function to interpolate given samples using Partitions of Unity. Firstly, the function determined the adaptive weight for each sample before multiplying it with the Gaussian function and summing all elements.

```
function [ Qxf ] = Qx( fn, std_fn,x,xn,n)
% Computation of Quasi Interpolation
Qxf = 0;  % Initialize
for r = 1:length(n)
    if r==1 && xn(r)==0
        d = min(diff(xn));
    elseif r==1 && xn(r)~=0
        d = xn(r);
    else
        d = xn(r)-xn(r-1);
    end
        Qxf = fn(r)*d/(sqrt(pi)*std_fn)*exp( - ( x -  xn(r) ).^2 / (std_fn^2)  ) + Qxf;
end
end
```

Auto-correlation Matrix

The algorithm required the orthogonal projection to be applied on the interpolated result. However, the orthogonal projector required the inverse matrix of the autocorrelation matrix.

```
% Auto Correlation Matrix
[A,iA,dxk,dxl,cond] = Akl(n,std_fs,tn);
```

Below is the created function to compute the autocorrelation matrix and its inverse.

```
function [A,iA,dxk,dxl,cond] = Akl( n,std_fn,xn)
% Determination of the auto-correlation matrix using the Gaussian function
fprintf('Computing auto-correlation matrix [A(k-l)] \nPlease wait...\n')
for k = 1:length(n)

    for l=1:length(n)
            if l==1 && k==1 && xn(l)==0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 1;

            elseif l==1  && k==1 && xn(l)==0 && xn(k)~=0
                dxk(k,l) = xn(k);
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 2;

            elseif l==1  && k==1 && xn(l)~=0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = xn(l);
                cond(k,l) = 3;

            elseif l==1  && k==1 && xn(l)~=0 && xn(k)~=0
                dxk(k,l) = xn(k);
                dxl(k,l) = xn(l);
                cond(k,l) = 4;
```

```
            elseif l==1  && k~=1 && xn(l)==0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 5;

            elseif l==1  && k~=1 && xn(l)==0 && xn(k)~=0
                dxk(k,l) = xn(k) - xn(k-1);
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 6;

            elseif l==1  && k~=1 && xn(l)~=0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = xn(l);
                cond(k,l) = 7;

            elseif l==1  && k~=1 && xn(l)~=0 && xn(k)~=0
                dxk(k,l) = xn(k)-xn(k-1);
                dxl(k,l) = xn(l);
                cond(k,l) = 8;

            elseif l~=1  && k==1 && xn(l)==0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 9;
```

```
            elseif l~=1  && k==1 && xn(l)==0 && xn(k)~=0
                dxk(k,l) = xn(k);
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 10;

            elseif l~=1  && k==1 && xn(l)~=0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = xn(l) - xn(l-1);
                cond(k,l) = 11;

            elseif l~=1  && k==1 && xn(l)~=0 && xn(k)~=0
                dxk(k,l) = xn(k);
                dxl(k,l) = xn(l) - xn(l-1);
                cond(k,l) = 12;

            elseif l~=1  && k~=1 && xn(l)==0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 13;

            elseif l~=1  && k~=1 && xn(l)==0 && xn(k)~=0
                dxk(k,l) = xn(k) - xn(k-1);
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 14;
```

```
            elseif l~=1  && k~=1 && xn(l)==0 && xn(k)~=0
                dxk(k,l) = xn(k) - xn(k-1);
                dxl(k,l) = min(diff(xn));
                cond(k,l) = 14;

            elseif l~=1  && k~=1 && xn(l)~=0 && xn(k)==0
                dxk(k,l) = min(diff(xn));
                dxl(k,l) = xn(l) - xn(l-1);
                cond(k,l) = 15;

            elseif l~=1  && k~=1 && xn(l)~=0 && xn(k)~=0
                dxk(k,l) = xn(k) - xn(k-1);
                dxl(k,l) = xn(l) - xn(l-1);
                cond(k,l) = 16;
            end


        % Computation of [A(k-l)]
        A(k,l) = ( dxk(k,l)*dxl(k,l) ) / (std_fn*sqrt(2*pi))* exp(-1/(2*std_fn^2)*(xn(k) - xn(l) )^2 );
    end
end

iA = inv(A);
end
```

Orthogonal Projection

When the inverse of the autocorrelation matrix was complete the orthogonal projection could be performed in the two steps below. The complete projection and approximation operator is defined by `p_E`.

```
%orthogonal Projection
p_D = Dx(p_Qxf,n,t,tn,std_fs);
p_E = p_D*iA;
```

Below is the function created to compute p_D.

```
function [ D ] = Dx(Qxf,n,x,xn,std_fn)

for r=1:length(n)
    if r==1 && xn(r)==0
        d = min(diff(xn));
    elseif r==1 && xn(r)~=0
        d = xn(r);
    else
        d = xn(r)-xn(r-1);
    end
    D(1,r)  = dot( Qxf,d/(sqrt(pi)*std_fn)*exp( - ( x -  xn(r) ).^2 / (std_fn^2)  ) );

end
```

Approximation

When the approximation operator was ready it was called in the main function as follows to form an approximated function.

```
% initial approximation
f1 = apprx( p_E,std_fs,t,tn,f0,n );       %
p_fx = f1;
```

Below is the function created to compute the approximation of the function. It took the approximation operator p_E and applied it to samples.

```
function [ fx ] = apprx( E,std_fs,x,xn,fx,n )
% Approximate function 'f(x)'
for r= 1:length(n)

    if r==1 && xn(r)==0
        d = min(diff(xn));
    elseif r==1 && xn(r)~=0
        d = xn(r);
    else
        d = xn(r)-xn(r-1);
    end

     fx = E(r)*d/(sqrt(pi)*std_fs)*exp( - ( x -  xn(r) ).^2 / (std_fs^2)  ) + fx;
end
end
```

Iterations

The snippet, below, performed the one-hundred thousand iterations. All of the functions defined in the previous sections were used.

```
  % for loop iteration
for r =1:itr
    p_ex = f-p_fx;
    p_Qxe = Qx(p_ex(n),std_fs,t,tn,n);
    p_xi(1,r) = r;
    p_ex = f-p_fx;
    d_conv(1,r) = norm(p_ex(n));
    %orthogonal Projection
    p_D = Dx(p_Qxe,n,t,tn,std_fs);
    p_Ex = p_D*iA;
    p_ex = apprx( p_Ex,std_fs,t,tn,f0,n );
    %approximation
    p_fx = p_ex + p_fx;
end
```

Bibliography

[**1**] K. Grochenig, *A Discrete Theory of Irregular Sampling*, Linear Algebra and its Applications 193 (1993), pp. 129-150

[**2**] R. Duffin, A. Schaeffer, *A class of nonharmonic Fourier series*, Trans. Amer. Math. Soc. 72 (1952), pp. 341-366.

[**3**] A. Aldroubi, *Obllique Projections in Atomic Spaces*, American Mathematical Society, Vol. 124, pp. 2051-2060

[**4**] R.Young, *An Introduction to Nonharmonic Fourier Series*, New York: Academic 1980

[**5**] W. Chen, S. Itoh, J. Shiki, *Irregular Sampling Theorems for Wavelets*, IEEE Trans. on Info. Theory, Vol. 44, No. 3 (1998), pp. 1131-1142

[**6**] R. Bale, J. Grossman, G. Margrave, M. Lamoureux, *Multidimensional partitions of unity and Gaussian terrains*

[**7**] G. Hardy, J. E. Littlewood, G. Polya, *Inequalities. 2nd Ed.*, Cambridge Univ. Press. 1952.

[**8**] M.I. Kadec, *The Exact Value of the Paley-Wiener Constant*, Soviet Math. Dokl. 5 (1964), pp. 559-561.

[**9**] N. Levinson, *Gap and Density Theorems*, Coll. Publ. 26, Amer. Amer. Math. Soc., New York, 1940.

[**10**] H. Feichtinger, K. Grochenig, T. Strohmer, *Efficient Numerical Methods in Non-Uniform Sampling Theory*, Numer. Math 69 (1995): pp. 423-440.

[**11**] M. Unser, *Splines: A perfect fit for signal and image processing*, IEEE Signal Processing Magazine, vol. 16 (1999).

[**12**] G. Marven, C. Ewers. *A simple aproach to digital signal processing.* Texas Instruments, 1993.

[**13**] C. Shannon, *Communication in the presence of noise*, Proceedings of the IRE, vol. 37 (1949) ,pp. 10–21.

[**14**] H. Nyquist,*Certain topics in telegraph transmission theory*, AIEE Transactions, vol. 35 (1928), pp. 617-644.

[**15**] Wojtkiewicz, Andrzej and Tuszynski, Michal, *Application of the Dirichlet transform in analysis of nonuniformly sampled signals*, IEEE Int. Conf. on Acoustics, Speech, and Signal Proc.(1992)

[**16**] N. Lomb, *Least-squares frequency analysis of unequally spaced data*, Astro-

physics and Space Science, vol. 39 (1976), pp. 447-462.
[**17**] J. Scargle, *Studies in astronomical time series analysis II: Statistical aspects of spectral analysis of unevenly spaced data*, The Astrophysical Journal, vol. 263 (1982), pp. 835-853.
[**18**] Y. Qi, T. Minka, and R. Picard, *Bayesian spectrum estimation of unevenly sampled nonstationary data*, IEEE Int. Conf. on Acoustics, Speech, and Signal Proc. Orlando, Florida, USA. (2002)
[**19**] E. Masry, M. Lui, *Discrete-time spectral estimation of continuous-parameter processes – a new consistent estimate*. IEEE Transactions on Information Theory, vol. 22 (1976), pp. 298-312.
[**20**] E. Masry, D. Klamer, C. Mirabile, *Spectral estimation of continuous-time processes: Performance comparison between periodic and Poisson sampling schemes*. IEEE Transactions on Automatic Control, vol. 23 (1978), pp. 679-685
[**21**] E. Masry, *Poisson sampling and spectral estimation of continuous-time processes*. IEEE Transactions on Information Theory, vol. 24 (1978), pp. 173-183.
[**22**] F. Beutler, *Error-free recovery of signals from irregularly spaced samples*, SIAM Review, vol. 8 (1966), pp. 328-335.
[**23**] R. Paley, N. Wiener, *Fourier Transforms in the Complex Domain*, New York: A.M.S. (1934).
[**24**] J. Yen, *On nonuniform sampling of bandwidth-limited signals*, IRE Trans. Circuit Theory, vol. 3, no. 4 (1956), pp. 251-257
[**25**] K. Yao and J. B. Thomas, *On some stability and interpolatory properties of nonuniform sampling expansions*, IEEE Trans. Circuit Theory, vol. 14, no. 4 (1967), pp. 404-408.
[**26**] J. Higgins, *Sampling Theory in Fourier and Signal Analysis: Foundations*, Oxford Science Publications, Clarendon Press, Oxford, (1996).
[**27**] F. Marvasti, M. Analoui, M. Gamshadzahi, *Recovery of signals from nonuniform samples using iterative methods*, IEEE Transactions on Signal Processing, vol. 39 (1991), pp. 872-878.
[**28**] F. Gunnarsson, F. Gunnarsson, F. Gustafsson, *Issues on performance measurements of TCP*, Radiovetenskapoch Kommunikation (2002). Stockholm, Sweden.
[**29**] F. Gunnarsson, F. Gunnarsson, F. Gustafsson, *Controlling Internet queue dynamics using recursively identified models*, IEEE Conference on Decision and Control (2003), Maui, Hawaii, USA.
[**30**] *Non-uniform sampling in statistical signal processing*,Linköping Studies in Science and Technology. Dissertations. No. 1082
[**31**] Y. Eldar and A. Oppenheim, *Filterbank reconstruction of bandlimited signals from nonuniform and generalized samples*,"IEEE Trans. Signal Processing, vol. 48, no. 10, pp. 2864-2875, October 2000.
[**32**] A. Aldroubi, M. Unser, and M. Eden, *On the asymptotic convergence of*

B-spline wavelets to Gabor functions,"IEEE Trans. Information Theory, vol. 38, no. 2, pp. 864-872, 1992.

[**33**] A. Aldroubi, M. Unser and M. Eden, "Cardinal spline filters: stability and convergence to the ideal sinc interpolator", Signal Process., vol. 28, no. 2, pp. 127-138, August 1992.

[**34**] P. Cassaza, *Hilbert space frames containing a Riesz basis and Banach spaces which have no subspace Isomorphic to* c_0, Journal of Mathematical Analysis and Applications, vol 202, pp 940 - 950, 1996.

[**35**] M. Crainic, S. Igonin, *Chapter 5: Partitions of Unity and Paracompactness*, lecture notes, Inleiding Topologie, Utrecht University, 2011

[**36**] J. Melenk, I. Babusaka, *The partition of unity finite element method basic theory and applications*, ETH Zurich Research Collection, 1996

[**37**] H. G. Feichtinger and K. Gröchenig, *Theory and practice of irregular sampling*, Wavelets: Mathematics and Applications, vol. 1994, pp. 305 - 363, 1994