ABSTRACT

Reconstruction of Under-sampled Super High Frequency Signals using Recurrent Nonuniform Sampling

(May 2018)

Teja Renukaradhya B.E., Vemana Institute of Technology

Chairman of Advisory Committee: Dr. Sung-won Park

Sampling devices form an interface between front ends of radio receivers and signal processing units and therefore play a vital role in digital radio and radar communications. In order to obtain signal outputs from devices without any loss of information, effective signal reconstruction is essential. In general, a signal can be faithfully reconstructed if it is sampled at a rate greater than or equal to the Nyquist rate. In some high frequency applications however, it is not possible to sample the signal at the Nyquist rate. In this thesis, we deal with the under-sampling and effective reconstruction of very high frequency signals. Here, we first further the discussion of accurately determining the center frequency using the main lobe to side lobe ratio, as discussed in paper [1], and then develop a practical implementation for the same. Next, we tackle the problem of aliasing, faced when nonuniformly sampling the signal at one-third and two-third of the sampling frequency. The aim is to develop a practical solution that comes within 3% of the theoretical results. Finally, we reconstruct the input signal with the help of bandpass filtering (to remove the side lobes) and interpolation.

ACKNOWLEDGEMENTS

First and foremost, I would first like to thank my thesis advisor Dr. Sung-won Park of the Frank H. Dotterweich College of Engineering at Texas A&M University – Kingsville. His ideas and suggestions formed the foundations of the work in this thesis. I am thoroughly grateful for his contributions of time and effort and also for his immense patience in guiding me through every step of this process.

Next, I would like to thank the committee members who were involved in the validation and verification of this research project: Dr. Chung S. Leung and Dr. Zhaohui Wang. Their active participation and input, was what made the entire process smooth and successful and I immensely appreciate their help and suggestions.

Last, but by no means the least, I want to express my gratitude to my family, for their unwavering support and encouragement. I also want to thank my friends who have been my family here and whose contributions have made this entire process both easy and fun. It is to this entire family, that I dedicate this project.

TABLE OF CONTENTS

Page

ABSTRACT iii

ACKNOWLEDGEMENTS iv

TABLE OF CONTENTS v

LIST OF TABLES viii

LIST OF FIGURES ix

CHAPTER I 1

INTRODUCTION 1

CHAPTER II 3

MULTI-RATE SIGNAL PROCESSING 3

2.1 Under-sampling and over-sampling 3

2.2 Interpolation and decimation 5

2.3 Filtering 6

2.3.1 FIR (All-zero system) and IIR (All-pole system) filters 8

2.3.2 Windows 10

CHAPTER III 12

COMPRESSIVE SENSING 12

3.1 Sparsity 13

3.2 Incoherence 14
3.3 Application of Compressive Sensing 14
CHAPTER IV 18
NONUNIFORM SAMPLING 18
4.1 Uniform sampling 18
4.2 Nonuniform sampling 18
4.2.1 Recurrent nonuniform sampling 19
CHAPTER V 22
DTFT OF RECURRENT NONUNIFORM SAMPLING 22
CHAPTER VI. 28
ALGORITHM 28
6.1 Algorithm 28
6.2 Frequency detection block diagram 29
6.3 Results 29
CHAPTER VII 41
RECONSTRUCTION 41
CHAPTER VIII 46
APPLICATIONS 46
CHAPTER IX 48
CONCLUSION 48

CHAPTER X 50

FUTURE WORK.. 50

REFERENCES 51

APPENDIX.. 54

MATLAB SIMULATED CODE.. 54

VITA... 65

LIST OF TABLES

Page

Table 1. Description of elementary filters [3] 7

Table 2. Practical and theoretical ratios for different frequencies of a pure sinusoidal signal 30

Table 3. Test and reference ratios for different frequencies of an FM signal 30

Table 4. Amplitude of the side lobe for one-third and two-third sampling frequencies............... 39

LIST OF FIGURES

Page

Figure 1. Aliasing due to under-sampling [2] 4

Figure 2. Under-sampling and over-sampling of a signal 4

Figure 3. Up-sampling and down-sampling a signal by a factor of 2 6

Figure 4. Lowpass filtering followed by down-sampling 6

Figure 5. Recovery of a sparse signal using l_1 and l_2 norms [6] 17

Figure 6. Uniform Sampling 18

Figure 7. Nonuniform Sampling 19

Figure 8. Interleaved ADC system [1] 20

Figure 9. Uniform sampling versus recurrent nonuniform sampling [1] 21

Figure 10. The coefficients of $X(\theta)$, $P(\theta)$, $Q(\theta)$ and $X_{non}(\theta)$ for $\theta_0 = 2\pi \times 0.3$ (rad) [1] 24

Figure 11. The coefficients of $X(\theta)$, $P(\theta)$, $Q(\theta)$ and $X_{non}(\theta)$ for $\theta_0 = 2\pi \times 1.3$ (rad) [1] 25

Figure 12. Magnitude of $R(k)$ for $\theta_0 = 2\pi \times 0.3$ and $\theta_0 = 2\pi \times 1.3$ and $k = 0,1,2$ [1] 26

Figure 13. Detection of center frequency of input Test signal 29

Figure 14. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.2 .. 32

Figure 15. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.4 .. 32

Figure 16. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.6 .. 33

Figure 17. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.8 .. 33

Figure 18. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.9 .. 34

Figure 19. Test and Reference ratio spectra of FM signal for normalized frequency 0.4 35

Figure 20. Test and Reference ratio spectra of FM signal for normalized frequency 0.8 35

Figure 21. Test and Reference ratio spectra of FM signal for normalized frequency 2.4 36

Figure 22. Test and Reference ratio spectra of FM signal for normalized frequency 2.8 36

Figure 23. Test and Reference ratio spectra of FM signal for normalized frequency 4.4 37

Figure 24. Test and Reference ratio spectra of FM signal for normalized frequency 4.8 37

Figure 25. Nonuniform sampling and reconstruction of an unfolded sinusoidal signal (0.4) 41

Figure 26. Nonuniform sampling and reconstruction of a folded input sinusoidal signal (2.7) ... 42

Figure 27. Nonuniform sampling and reconstruction of a folded input sinusoidal signal (4.6) ... 43

Figure 28. Nonuniform sampling and reconstruction of an unfolded input FM signal (0.4) 43

Figure 29. Nonuniform sampling and reconstruction of a folded input FM signal (2.7) 44

Figure 30. Nonuniform sampling and reconstruction of a folded input FM signal (4.6) 44

Figure 31. Applications of high frequency signals [22] 47

CHAPTER I

INTRODUCTION

The most common sampling methods used in Digital Signal Processing (DSP) employ uniform sampling. But in many practical cases, uniform sampling of data is not possible. And since most of the traditional reconstruction methods are for uniformly sampled signals, the effective reconstruction of nonuniformly sampled signals still remains a problem. Furthermore, the requirement of sampling at the Nyquist rate may exceed the capacity of currently available analog-to-digital converters. This shows that, it is not always possible to sample signals above the Nyquist rate, in which case under-sampling is the only option. But effective reconstruction of under-sampled nonuniform signals is essential in many fields like image processing, speech processing, medical imaging, radar communication, geophysics, communication theory and astronomy.

Recently, it has been found that reconstruction of a narrow band signal is possible even when it is sampled at a rate lower than the Nyquist rate, provided the center frequency is known. Recurrent nonuniform sampling results in the generation of side lobes which in turn can be used to obtain the center frequency. This can be done by first obtaining the Discrete Fourier Transform of the nonuniformly sampled signal and then estimating the frequency of the original signal using the side lobe to main lobe ratio. So, even if the main lobe folds down due to aliasing, the center frequency of the signal can be found with the help of the side lobes [1].

In this thesis, we discuss the practical sampling and reconstruction of a narrow band nonuniformly under-sampled signal. This is done by first nonuniformly under-sampling the signal which results in side lobes. These help in estimating the center frequency about which the signal can be reconstructed. Finally, the under-sampled signal is passed through a band pass filter to eliminate the side lobes and provide a reconstructed output. If the center frequency estimate is higher than half the sampling frequency, the signal appears folded and must therefore be bandpass filtered and then interpolated. This method is feasible and economical and can be used in frequency hopping and radar communications, bringing down the cost of expensive high speed digital systems.

CHAPTER II

MULTI-RATE SIGNAL PROCESSING

The world we live in uses, analog continuous time signals for a variety of applications. However, we use digital devices to both process and store these signals. Therefore it is required for them to be first converted to digital form. This area of converting analog signals into their digital form and processing them is called Digital Signal Processing (DSP). Sampling is a vital element of these DSP systems. It is the process of converting continuous time signals into discrete time signals by recording the values of the continuous signal at various instances in time. Many times we use the highest frequency (f_H) as the bandwidth (B) of the signal, so that we can work on all the frequencies between 0 Hz and the highest frequency.

To perfectly reconstruct the signal later, the sampling rate (f_S) must be twice the bandwidth of the signal, $f_S \geq 2B$. This rate is called the Nyquist rate and this criteria forms the sampling theorem.

2.1 Under-sampling and over-sampling

Under-sampling is the process of sampling a signal at a rate lower than the Nyquist rate. It is difficult to analyze an under-sampled signal. However, it is still possible to reconstruct under-sampled signals using some special techniques.

If a signal is under-sampled, its frequency components overlap to cause errors. The process of formation of these new unwanted frequency components is called aliasing, shown in Figure 1.

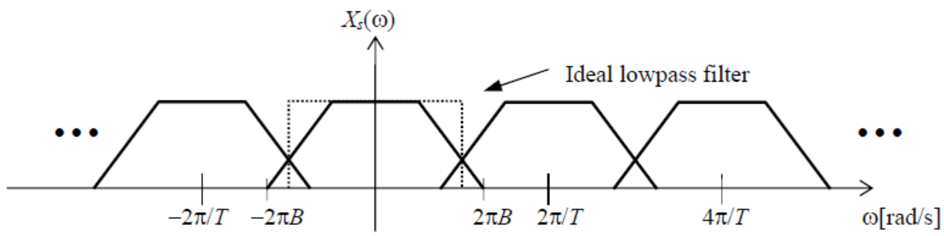


Figure 1. Aliasing due to under-sampling [2]

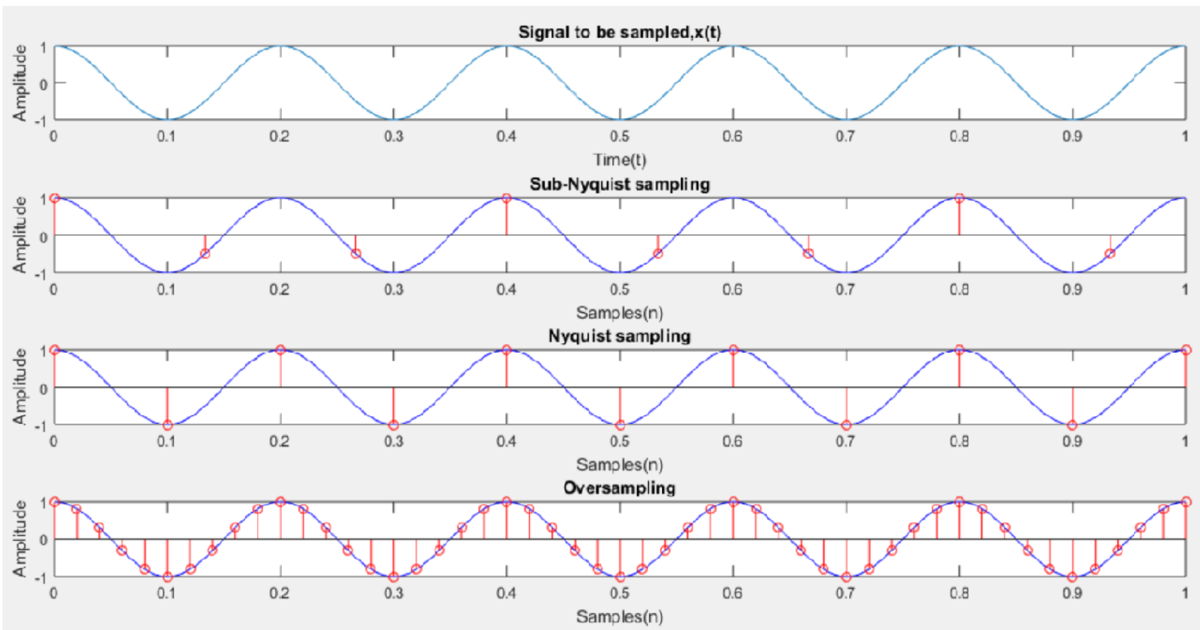


Figure 2. Under-sampling and over-sampling of a signal

The process of sampling a signal at a rate much higher than the Nyquist rate is called over-sampling. A higher rate of sampling means more number of samples, which help in achieving a much more accurate reconstruction of the signal. It also reduces the complexity of the reconstruction system. Over-sampling has many advantages in antialiasing applications.

2.2 Interpolation and decimation

The number of samples in discrete time signals can be changed. In signal processing a signal can be up-sampled or down-sampled according to its application. Interpolation or up-sampling is the process of increasing the number of samples in a discrete signal by inserting zeros between the samples.

Example: Let us consider the sampled signal shown in Figure 3. The length of the sampled signal is N. The number of samples N is increased by a factor n, by inserting $n-1$ zeros between the samples, for all n. This compresses the frequency axis by a factor of n. In Figure 3, the factor n is equal to 2.

Decimation or down-sampling is the process of decreasing the number of samples in a discrete signal by selecting every n^{th} sample and discarding the rest. Thereby expanding the frequency axis by a factor of n. Figure 4 shows down-sampling of a sampled signal by a factor of 2. Down-sampling causes aliasing of frequency components when the original signal is reconstructed. In such cases the signal must be passed through a low pass filter to prevent aliasing, as shown in Figure 4.

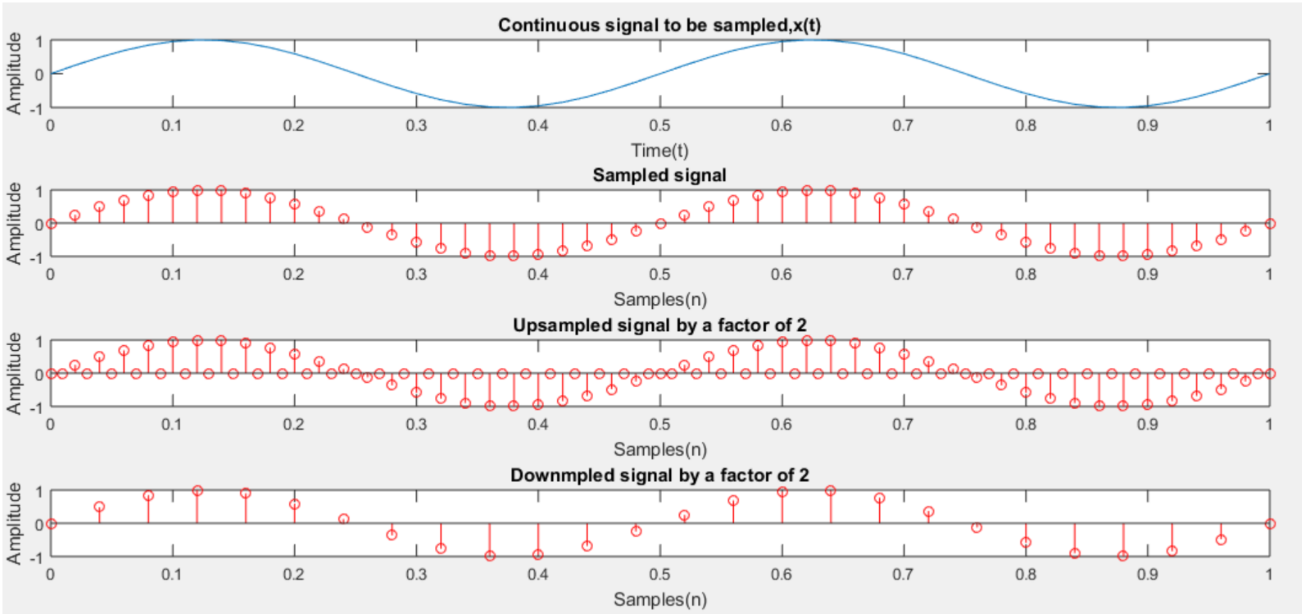


Figure 3. Up-sampling and down-sampling a signal by a factor of 2

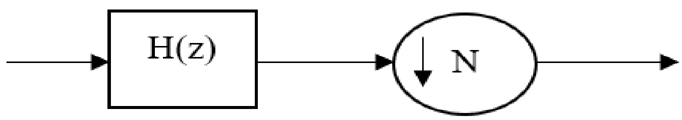
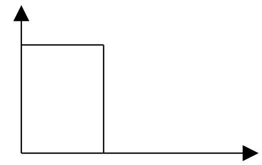
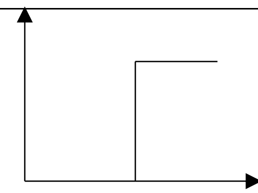
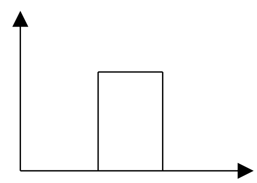
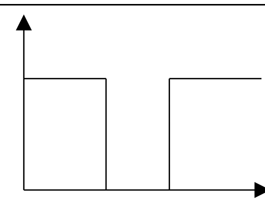


Figure 4. Lowpass filtering followed by down-sampling

2.3 Filtering

In DSP, digital filtering is an important tool. Filtering is the process of selecting a frequency or a range of frequencies from a given signal. Simply put, it is the process of filtering out unwanted frequency components from a given signal. Filtering is used in a variety of applications such as speech recognition, noise cancellation, etc. The range of frequencies allowed to pass through the filter forms the **pass-band,** and the range that is filtered out is called the **stop-band.** The threshold that separates the pass-band and stop-band is known as the **cut-off frequency**.

Table 1. Description of elementary filters [3]

Filter name	Mathematical Expression	Description	Graphical Representation
Low Pass Filter (LPF)	$\frac{1}{2}(1 + z^{-1})$	Allows low frequency components of f_p	
High Pass Filter (HPF)	$\frac{1}{2}(1 - z^{-1})$	Allows high frequency components of f_p	
Band Pass Filter (BPF)	$\frac{1}{2}(1 - z^{-2})$	Passes frequencies inside a specific range	
Band Stop Filter (BSF)	$\frac{1}{2}(1 + z^{-2})$	Attenuates frequencies inside a specific range	
All Pass Filter (APF)	z^{-N}	Passes all frequencies uniformly in gain and change the phase relationship	

A filter that allows lower frequencies to pass through, while blocking the remaining is called a **Lowpass filter**. On the other hand, a filter that allows higher frequencies to go through, while blocking the rest is called a **High pass filter**. If a filter allows only certain range or band of frequencies to pass through, while blocking the rest, it is called a **Band pass filter**.

Table 1 shows some common elementary filters used in signal processing. The term f_p represents the pass-band while f_s represents the stop-band. In band pass filters there are two pass-band edges $-f_{p1}$ and f_{p2}, where $(f_{p1} < f_{p2})$, and all the frequencies between these two bands are allowed. Similarly, in a band stop filter, there are two stop-band edge $-f_{s1}$ and f_{s2}, where $(f_{s1} < f_{s2})$, and only the frequencies above f_{s1} and below f_{s2} are allowed. In this thesis, we use a bandpass filter since we know the frequency range in which the narrow band signal must be reconstructed.

2.3.1 FIR (All-zero system) and IIR (All-pole system) filters

Digital filters can be divided into two main categories namely, Finite Impulse Response (FIR) filters and Infinite Impulse Response (IIR) filters, based on the length of their impulse response. FIR filters have a finite number of nonzero samples or impulse responses, because of which, they are stable. These filters have a linear phase, which means that, they have the same phase response for all frequencies. They are time invariant (their response doesn't change with time) and easy to design. The equation for an FIR filter is given by,

$$y(n) = h(n) * x(n) \tag{2}$$

where $x(n)$ is the input of the filter and $y(n)$ is the output of the filter. The output of the filter is obtained by the convolution sum

$$y(n) = \sum_{m=0}^{M} h_m x(n-m) \tag{3}$$

The transfer function is given by,

$$H(z) = \frac{Y(z)}{X(z)} = h_0 + h_1 z^{-1} + \ldots + h_m z^{-m} \tag{4}$$

The roots of the numerator polynomial are called zeros while the roots of the denominator polynomial are called poles. Since FIR filters have M zeros and no non-zero poles in their transfer function, they form an **all-zero system**.

IIR filters have an infinite number of non-zero samples. They are non-linear and computationally efficient. IIR filters are also recursive, which means that they have feedback and the output of the filter depends on the present input as well as the output of the previous stage. The equation for an IIR filter is as follows:

$$y(n) = h(k) * x(n-k) \tag{5}$$

$$y(n) = \sum_{m=0}^{M} a_m x(n-k) - \sum_{m=1}^{M} b_m y(n-k) \tag{6}$$

where $x(n)$ is the input of the filter, $y(n)$ is the output of the filter, a_m represents the feedforward filter coefficients and b_m represents the feedback filter coefficients. The transfer function is given by,

$$H(z) = \frac{X(z)}{Y(z)} = \frac{a_0 + a_1 z^{-1} + \ldots + a_N z^{-N}}{1 + b_1 z^{-1} + \ldots + b_M z^{-M}} \tag{7}$$

There are many ways of designing an FIR filter: [2]

- Frequency sampling method
- Window method
- Weighed least square method
- Equiripple method

The window method is the most convenient method of FIR filter designing out of all the methods mentioned above.

2.3.2 Windows

Common types of windows used in the design of filters are given below:

- Rectangular Window: It is the simplest window and has great resolution. It reduces the smoothing effect by narrowing [3] but the side lobes are larger. It is defined by,

$$w(n) = \begin{cases} 1, & \text{for all } n=0,1,2,\ \ldots\ ,M \\ 0, & \text{elsewhere} \end{cases} \tag{8}$$

- Bartlett Window: It is a triangular shaped window function. Attenuation of the side lobe is greater in this window than in the rectangular window [3]. It is defined by,

$$w(n) = \frac{\left| n - \frac{M-1}{2} \right|}{\frac{M-1}{2}} \tag{9}$$

- Hanning window: It is a general-purpose window used for continuous signal analysis. It allows good frequency resolution with lower side lobes compared to the rectangular window [3]. It is defined by,

$$w(n) = 0.5(1 - \cos \frac{2\pi n}{M-1}) \tag{10}$$

- Hamming window: It is similar to the Hanning window but suppresses the first side lobe even more than the Hanning window. It is defined by,

$$w(n) = 0.54 - 0.46 \cos(\frac{2\pi n}{M-1}) \tag{11}$$

- Blackmann window: This window is similar to the above mentioned windows but has a smaller pass-band ripple and better stop-band attenuation [3]. It is defined as,

$$w(n) = 0.42 - 0.5\cos(\frac{2\pi n}{M-1}) + 0.08\cos(\frac{4\pi n}{M-1}) \tag{12}$$

In this thesis, we use a band pass filter designed using the hamming window, to reconstruct the signal. The advantages of using a Hamming window are:

1. It suppresses the first side lobe, thereby giving better selectivity for larger signals.
2. It has narrow transition bands compared to other windows.

CHAPTER III

COMPRESSIVE SENSING

In many practical applications, especially in the radar and communication sectors, we use super high frequency radio signals. Higher frequencies require higher sampling rates as per the Nyquist criteria and this puts immense stress on the present-day systems, especially the analog-to-digital converters (ADCs). Also, a large number of samples makes compression necessary prior to storage or transmission. This requirement increases the cost of these signal processing systems. However, it has been observed that the level of information contained in a signal is often much lower than the actual bandwidth, which means that with better reconstruction schemes we can overcome the present requirement of high sampling rates for high frequency signals, and consequently reduce the overall cost of these systems.

Most people are familiar with JPEG that compresses Megabytes of input data into a few Kilobytes without much or any perceptual loss. A similar compression scheme could be used in signal processing too. A new field known as Compressive Sensing (CS) has been introduced which is based on the fact that an ADC's sampling rate must correspond to the amount of essential information contained in a signal rather than the entire bandwidth of the signal. While CS seemingly debunks Shannon's theorem by sampling at a much lower rate than required, it does not completely disprove it. There are some rules as to what signals can be sampled using CS. For example, in order to deploy CS and sample below the Nyquist rate, a signal must be compressible by some transform such as Fourier or Wavelet.

Basically, Compressive Sensing attempts to combine the data acquisition and data compression processes by gathering only those samples that represent the information carried in the signal or image. As mentioned above, the underlying requirement is a signal that can be compressed using some type of transform. The Fourier transform is one of the most popular ones used in the processing of communication signals, but there are many other ways to compress signals and images like wavelets, spikes, or even tight frames including curvelet or Gabor representations. The idea here is to identify the basis of the signal using which the information in the signal can be described as a weighted sum of a family of functions. With this, even a signal with nearly all nonzero values can be compressed and represented as a function of the basis with only a few nonzeros. Compressive sampling relies on two main principles to make this possible: **sparsity** (pertains to the signals of interest) and **incoherence** (pertains to the sensing modality) [5].

3.1 Sparsity

Sparsity calls upon the idea that the level of information contained in a signal is less than what its bandwidth suggests. This principle states that the important information in a signal depends only on a few components, and the number of such components is comparably much smaller than the length of the signal. These components form a sparse signal which is nothing but a concise or compressed version of the discrete signal expressed in a different basis Ψ. A sparse signal rapidly decays to zero with the coefficients of the transform basis sorted from highest to lowest. However, this is relative to the total number of samples used to represent the signal. Simply put, sparsity dictates how much signal compression can be achieved using any transform.

3.2 Incoherence

Incoherence refers to the idea that objects which have a sparse representation in ψ, must be spread out in the domain in which they are acquired, similar to a Dirac or spike in the time domain that is spread out in the frequency domain representation. In other words, this principle iterates that the widely spread sampling/sensing waveforms of discrete time signals have an extremely dense representation in the basis Ψ.

The operating point here is that, by using better sensing or sampling methods, the necessary information in a signal can be captured and compressed giving a small amount of useful data. These methods involve correlating the signal with a small number of fixed waveforms that are incoherent with the sparsifying basis Ψ [5]. As per this methodology, a sensor can efficiently obtain the information contained in a sparse signal without trying to comprehend that signal. Numerical optimization then allows us to reconstruct the entire signal from just a small amount of data.

3.3 Application of Compressive Sensing

Consider a signal comprising of several unknown and real vectors of length N, all of which are sparse in a specified basis. Now consider one such vector $\boldsymbol{x}$ of this signal and assume it has K-sparse, which means that it can be expressed using a linear combination of K elements all of which belong to the basis set $\{\psi_i\}$ [5].

$$\boldsymbol{x} = \sum_{i=1}^{k} \alpha_i \psi_i \tag{13}$$

It is possible to recover or reconstruct $\boldsymbol{x}$ using traditional methods like sampling each value of $\boldsymbol{x}$. However, to achieve such a sampling rate would be impossible. Besides, it would mean wasting energy to encode bits that may not be useful in the final representation. The real challenge is to find the data that really matter in the reconstruction of the signal. The answer lies in using the compressive sensing method to reconstruct $\boldsymbol{x}$ using a minimal number of samples.

Prior to the process of sampling, if the right basis elements were known, $\{\psi_i\}$ could be represented by simply extracting the K useful entries of the signal. However, it is impossible to attain prior knowledge of the signal. Therefore, compressive sensing suggests using a second basis set $\{\phi_i\}$ that is incoherent with $\{\psi_i\}$, and collecting samples that are projections of the unknown signal onto this new incoherent basis set. Here, incoherent refers to the sparse representation of any element of the basis set $\{\phi_i\}$ that is independent of basis vector $\{\psi_i\}$ and vice versa. The linear model $\boldsymbol{y} = \Phi \boldsymbol{x}$ can be used to describe this process, where $\boldsymbol{y}$ is a length M vector and $K < M << N$ [5].

Basically, in any CS application, there are two orthonormal bases that describe a signal. One is the basis Φ which is used for sensing the signal, and the other is the basis Ψ which is used to represent the signal. We can also say that Φ is the sampling method used while Ψ is the transform basis representation. The coherence between these terms is given by:

$$\mu(\Phi, \Psi) = \sqrt{n} \max_{1 \le k, j \le n} \left| \left\langle \Phi_k, \Psi_j \right\rangle \right| \qquad (14)$$

where n gives the number of elements in each orthonormal base.

The coherency of a CS system is simply maximum correlation between any two components of the bases $-\Phi$ and Ψ. So if the two bases tend to have similar or correlated components then the coherence is large and if the components are uncorrelated, the coherence is said to be small. For CS to be successful, the coherence is required to be small. Incoherent bases components which have a sparse representation in Ψ, must be spread out in the domain in which they are acquired. Just as a Dirac or a spike in the time domain is spread out in the frequency domain. Therefore, for μ to be minimized at 1, each of the measurement vectors has to be spread out in the Ψ domain. The reason for incoherency is now obvious - the less the coherence, fewer the samples required. With the condition of incoherency, it is essential to choose the right sensing mechanism. This brings us back to the concept of random sampling. Random sampling is incoherent with almost all transform basis.

No prior knowledge of the signal is required to obtain these incoherent projections. Reconstruction can be accomplished even if the basis in which the signal is sparse is unknown. It is done by solving the l_i minimization problem and finding the estimate for $\boldsymbol{x}$ in the underdetermined set of linear equations given below [6]:

$$\hat{\alpha} = \arg\min \| \alpha \|_1 \;\; \Rightarrow \;\; \boldsymbol{y} = \Phi \boldsymbol{x} = \Phi \Psi \boldsymbol{\alpha} \tag{15}$$

$$\boldsymbol{x} = \Psi \hat{\boldsymbol{\alpha}} \tag{16}$$

On solving the l_i minimization problem, it is seen that the l_0 minimization simply counts the number of nonzero components in the signal. The l_2 minimization measures signal energy instead of signal sparsity and almost never finds a K-sparse solution, returning instead a non-sparse result with several nonzero components. Optimization based on l_1 minimization however, can exactly recover K-sparse signals and closely approximate the input signal with high probability. It is therefore, the most widely adopted norm in CS. That said, the extensively computational nature of the l_1-norm optimization method has prompted research into other alternatives for signal recovery in CS.

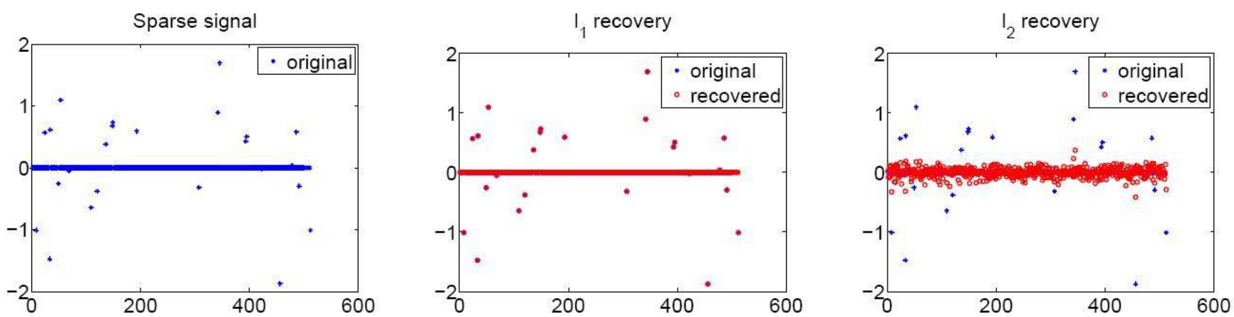


Figure 5. Recovery of a sparse signal using l_1 and l_2 norms [6]

In summary, compressive sampling is a simple and efficient method that allows us to sample a signal at a low rate and then use certain computations and optimizations to reconstruct the signal from a small set of samples. It is therefore, a method of sensing and compressing data simultaneously, hence the name compressive sensing. In this research we use compressive sampling in the reconstruction of a nonuniformly under-sampled narrow band signal.

CHAPTER IV

NONUNIFORM SAMPLING

4.1 Uniform sampling

Uniform or periodic sampling is the process of sampling a continuous-time signal at equally spaced time intervals. It is the most common form of sampling and has many advantages. Uniform sampling is easy to implement and more viable for digital signal processing. An example of periodic sampling is shown in Figure 6. The general equation for uniform sampling is given below:

$$x[n] = x(nT), \quad -\infty < n < \infty \tag{17}$$

Here T is the sampling period in second and $f_s = 1/T$ is the sampling frequency in Hz.

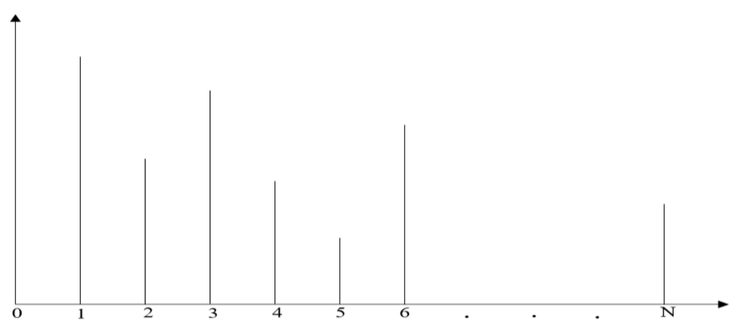


Figure 6. Uniform Sampling

4.2 Nonuniform sampling

Non-uniform sampling is the process of sampling a signal by taking samples randomly or at unequally spaced time intervals. For a very high frequency signal, the frequency above the

Nyquist rate gets folded in the interval of $-f_s/2$ to $+f_s/2$ (f_s being the sampling rate). This is known as the aliasing effect. An example of nonuniform sampling is seen in Figure 7.

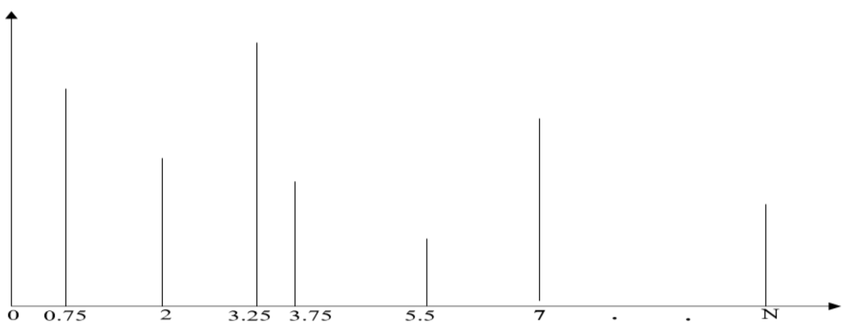


Figure 7. Nonuniform Sampling

4.2.1 Recurrent nonuniform sampling

In this paper, the process of recurrent nonuniform sampling is employed. Here, the total sampling points are split into groups of K points each. These groups recur after every period T_S, where T_S is equal to K times the Nyquist period T_0. Therefore, representing the sampling points in a period T_S by $\tau_k \cdot T_0$, for $k = 0,1,\ldots,K-1$, the complete set of sampling points within $n \in (-\infty, \infty)$ are given by [10]:

$$\tau_k \cdot T_0 + n \cdot T_S \text{ with } 0 \le \tau_k < K \qquad (18)$$

where $T_S = K \cdot T_0$

These recurrent nonuniform samples can be seen as a combination of several sequences of uniform samples taken at one K^{th} the Nyquist rate.

As shown in Fig. 8, one way of performing recurrent nonuniform sampling is with the help of interleaved ADCs. The sampling rate can be controlled by increasing or decreasing the number of ADCs used. For example, if M ADCs are employed each with a sampling rate of $1/MT$ [Hz], then the overall sampling rate of the system is $1/T$ [Hz]. This means that, in an ideal case, each delay must be exactly T seconds. However, due to some overheads, the actual cumulative delay is seen to be $(m+r_m)T$ where the term r_m represents the nonuniform sampling ratios.

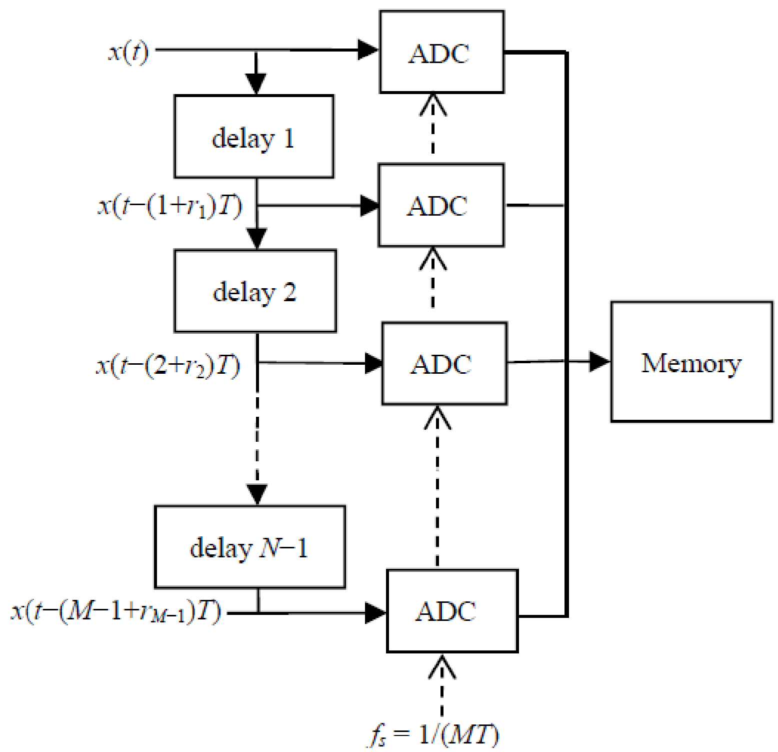


Figure 8. Interleaved ADC system [1]

Simply put, recurrent nonuniform sampling refers to the nonuniform sampling of a continuous time signal with a periodic pattern. Fig. 9 compares the result of recurrent nonuniform sampling versus uniform sampling using three ADCs. For uniform sampling, the sampling interval is T while the nonuniform sampling ratios are represented in terms of r_m sampling. Recurrence lies in the fact that the sampling patterns repeat themselves periodically. In this case, the recurrence period is 3.

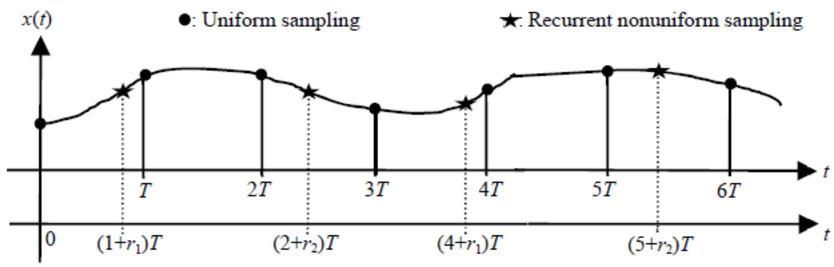


Figure 9. Uniform sampling versus recurrent nonuniform sampling [1]

CHAPTER V

DTFT OF RECURRENT NONUNIFORM SAMPLING

As seen in the previous chapter, the actual cumulative delay resulting from the recurrent nonuniform sampling of a signal is $(m+r_m)T$ when M ADCs are used to sample the signal at a rate of $1/MT$ [Hz]. From this, we can write the equation for a nonuniformly sampled continuous-time signal $x(t)$ as [1]:

$$(kM+m)T+r_mT=(kM+m+r_m)T \tag{18}$$

Here k ranges from $-\infty$ to ∞, m goes from 0 to $M-1$ and T is the average sampling interval. Let us assume that our input is a complex exponential signal given by:

$$x(t)=e^{j\theta_0 t} \tag{19}$$

If the signal uniformly sampled with interval $T=1$, the result is given by:

$$x(n)=e^{j\theta_0 n} \text{ for } n=0,1,\dots,N-1 \tag{20}$$

The DTFT of this sequence can then be calculated as follows [1]:

$$X(\theta)=\sum_{n=0}^{N-1}e^{j\theta_0 n}e^{-j\theta n}=\sum_{n=0}^{N-1}e^{-j(\theta-\theta_0)n}$$

$$\Rightarrow X(\theta)=\frac{(e^{j(\theta-\theta_0)N/2}-e^{-j(\theta-\theta_0)N/2})e^{-j(\theta-\theta_0)N/2}}{(e^{j(\theta-\theta_0)/2}-e^{-j(\theta-\theta_0)/2})e^{-j(\theta-\theta_0)/2}}$$

$$\Rightarrow X(\theta)=\frac{\sin((\theta-\theta_0)N/2)}{\sin((\theta-\theta_0)/2)}e^{-j(\theta-\theta_0)(N-1)/2} \tag{21}$$

Performing recurrent nonuniform sampling on $x(t)$ results with $M=3$ gives:

$$x_{non}(n)=\begin{cases} e^{j\theta_0 n}, & n=0,3,6,\dots \\ e^{j\theta_0(n+r_1)}, & n=1,4,7,\dots \\ e^{j\theta_0(n+r_2)}, & n=2,5,8,\dots \end{cases} \tag{22}$$

The DTFT of the above $x_{non}(n)$ is calculated as follows [1]:

$$X_{non}(\theta) = \sum_{n=0}^{\frac{N}{3}-1}\left[e^{j\theta_0 3n} e^{-j\theta 3n} + e^{j\theta_0 (3n+1+r_1)} e^{-j\theta(3n+1)} + e^{j\theta_0(3n+2+r_2)} e^{-j\theta(3n+2)} \right]$$

$$\Rightarrow X_{non}(\theta) = \frac{1-e^{-j(\theta-\theta_0)N}}{1-e^{-j(\theta-\theta_0)3}} \left[1 + e^{j\theta_0 r_1} e^{-j(\theta-\theta_0)} + e^{j\theta_0 r_2} e^{-j2(\theta-\theta_0)} \right]$$

$$\Rightarrow X_{non}(\theta) = \frac{(e^{j(\theta-\theta_0)N/2} - e^{-j(\theta-\theta_0)N/2}) e^{-j(\theta-\theta_0)N/2}}{(e^{j(\theta-\theta_0)3/2} - e^{-j(\theta-\theta_0)3/2}) e^{-j(\theta-\theta_0)3/2}} Q(\theta)$$

$$\Rightarrow X_{non}(\theta) = P(\theta) Q(\theta) \tag{23}$$

where,

$$P(\theta) = \frac{\sin((\theta-\theta_0)N/2)}{\sin((\theta-\theta_0)3/2)} e^{-j(\theta-\theta_0)(N-3)/2} \tag{24}$$

and

$$Q(\theta) = 1 + e^{j\theta_0 r_1} e^{-j(\theta-\theta_0)} + e^{j\theta_0 r_2} e^{-j2(\theta-\theta_0)} \tag{25}$$

Figure 9 shows the coefficients of $X(\theta)$, $P(\theta)$, $Q(\theta)$ and $X_{non}(\theta)$ for the nonuniform sampling ratios $r_1 = r_2 = 0.1$ and $\theta_0 = 2\pi \times 0.3$ [rad].

It is seen that the DTFT of the uniformly sampled sequence $X(\theta)$ has one main lobe, as expected, at $\theta_0 = 2\pi \times 0.3$ [rad]. However, the DTFT of $X_{non}(\theta)$ shows a main lobe along with two side lobes at $2\pi \times 0.6\bar{3}$ [rad] and $2\pi \times 0.9\bar{3}$ [rad].

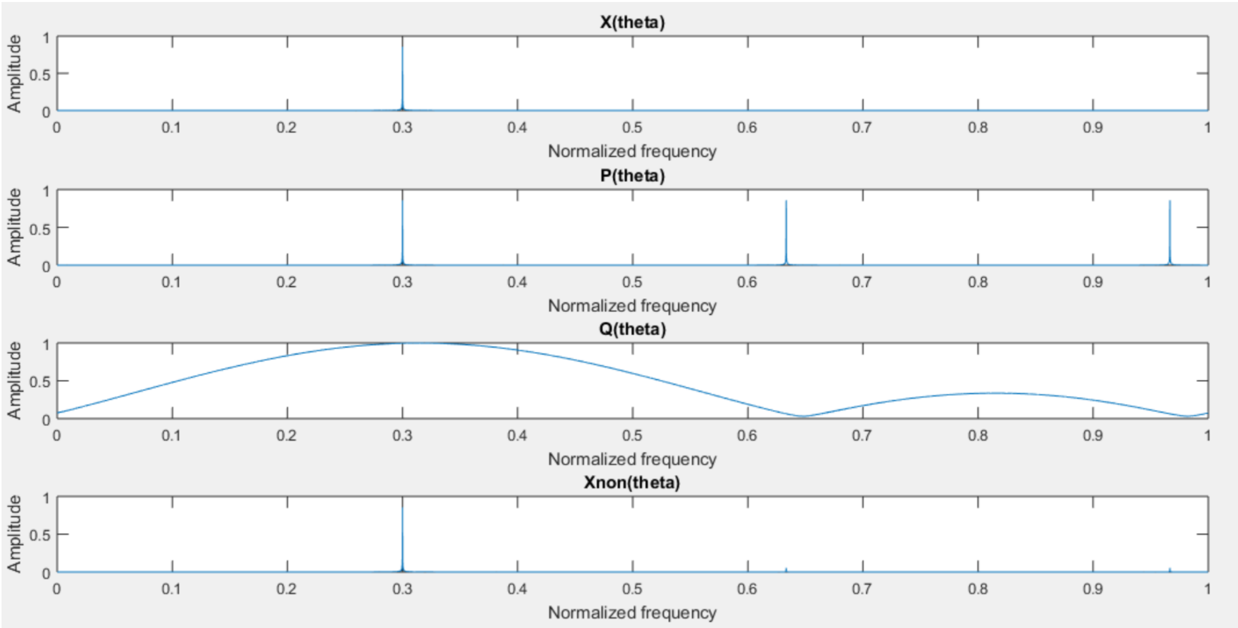


Figure 10. The coefficients of $X(\theta)$, $P(\theta)$, $Q(\theta)$ and $X_{non}(\theta)$ for $\theta_0 = 2\pi \times 0.3$ (rad) [1]

Now consider the case when $\theta_0 = 2\pi \times 1.3$ [rad] as seen in Figure 10. We see that for $X(\theta)$, the frequency folds down from $2\pi \times 1.3$ [rad] to $2\pi \times 0.3$ [rad]. This occurs due to the effect of aliasing. This frequency folding makes it impossible to determine the actual frequency of the original signal from the frequency of the DTFT of a uniformly sampled signal. However, it is seen that, the side lobes in the DTFT of the nonuniformly sampled signal $X_{non}(\theta)$ for $\theta_0 = 2\pi \times 1.3$ [rad] are at the same locations when compared to the ones for $\theta_0 = 2\pi \times 0.3$ [rad], except that they have larger amplitudes. It is also seen that the magnitude of these side lobes is directly related to the product of $P(\theta)$ and $Q(\theta)$.

Let us now consider the sequence $r(m)$:

$$r(m) = \left\{1,\ e^{j\theta_0 r_1},\ e^{j\theta_0 r_2}\right\} \tag{26}$$

The DTFT of $r(m)$ gives us [1]:

$$R(\theta) = 1 + e^{j\theta_0 r_1} e^{-j\theta} + e^{j\theta_0 r_2} e^{-j2\theta} \tag{27}$$

On solving, we obtain [1]:

$$Q(\theta) = R(\theta - \theta_0) \tag{28}$$

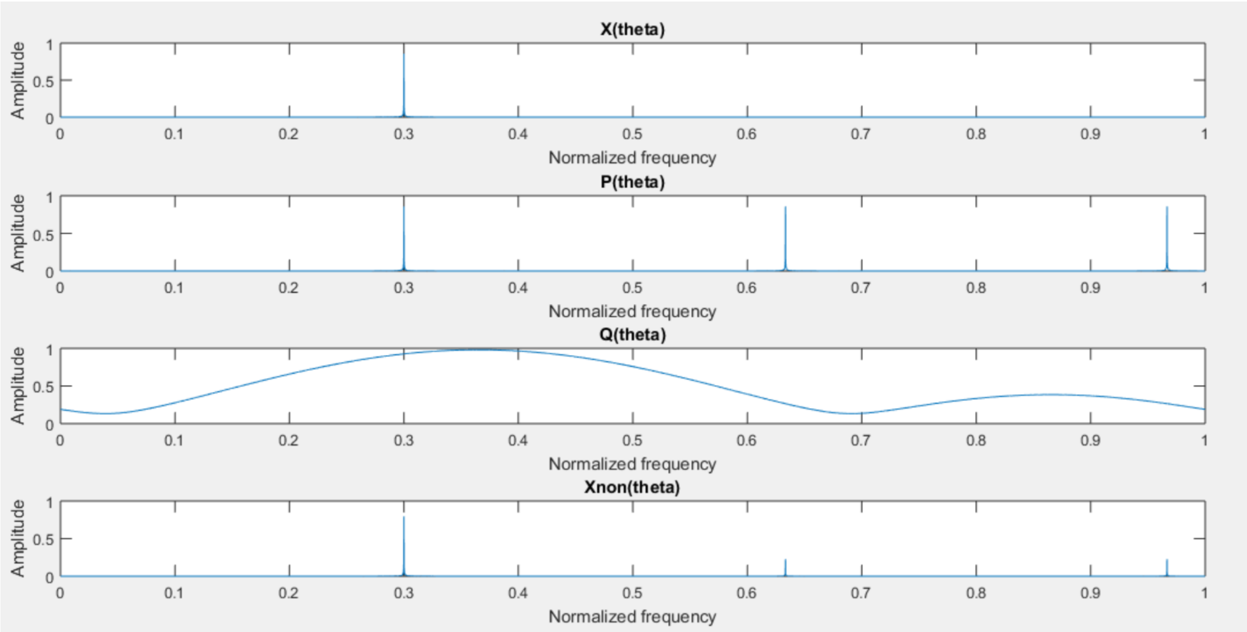
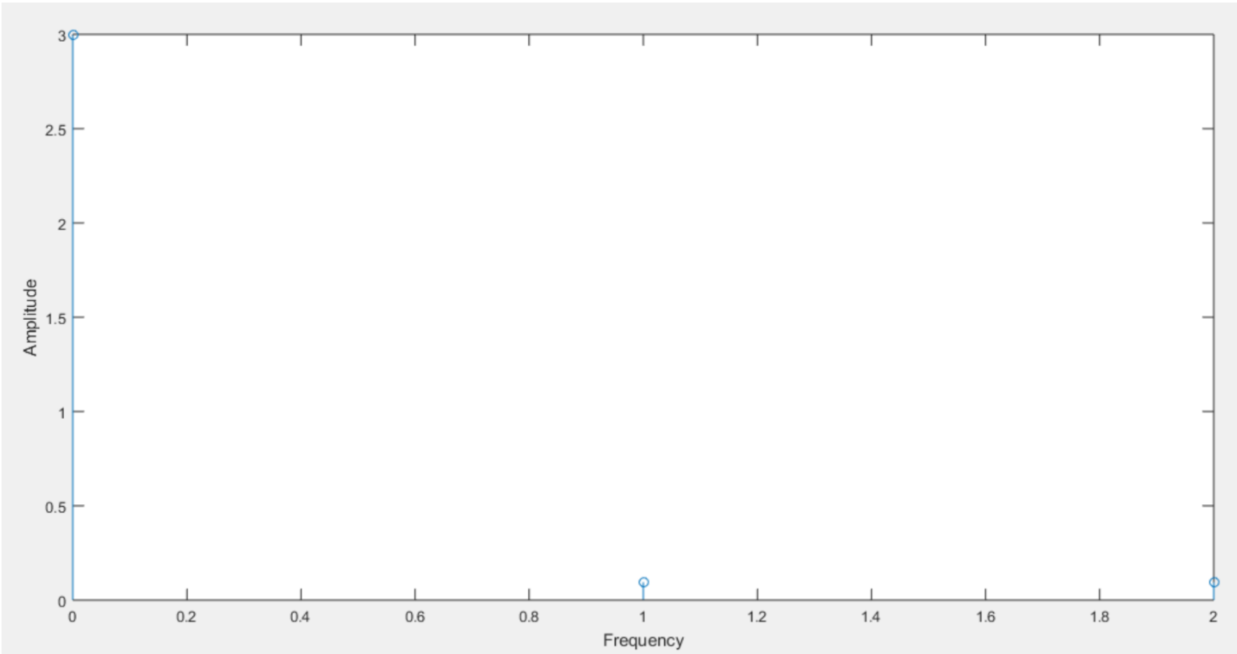


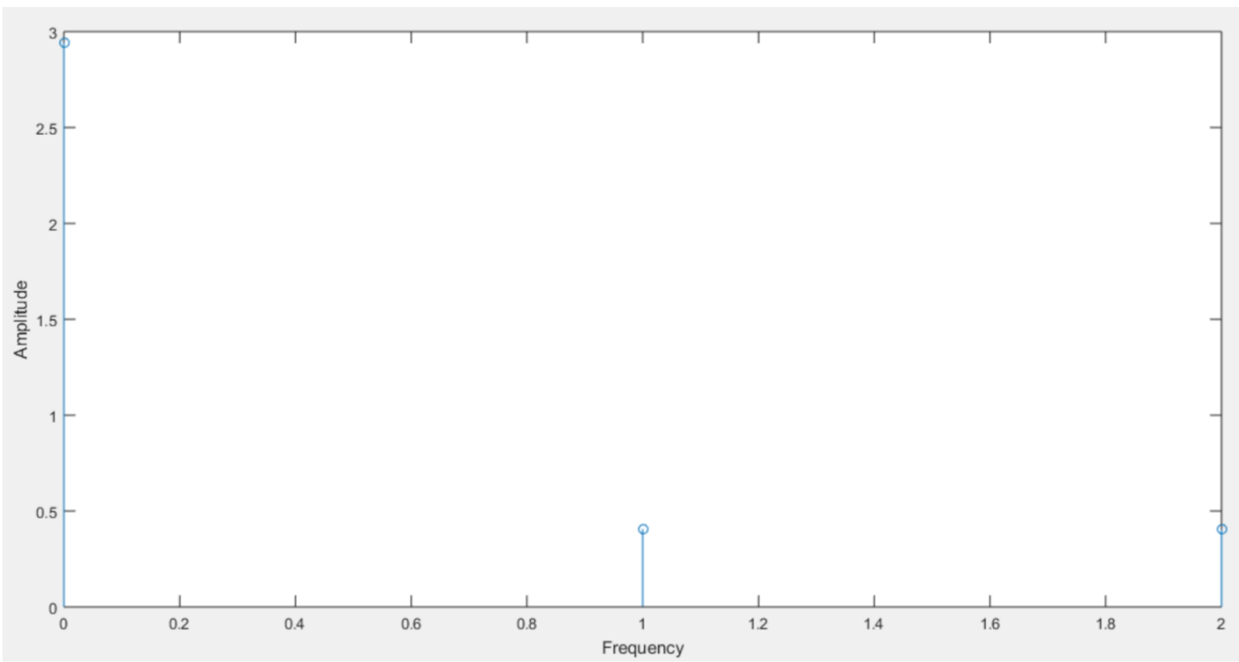
Figure 11. The coefficients of $X(\theta)$, $P(\theta)$, $Q(\theta)$ and $X_{non}(\theta)$ for $\theta_0 = 2\pi \times 1.3$ (rad) [1]

Now $R(0)$ gives us to the main lobe while $R(2\pi/3)$ and $R(4\pi/3)$ give us the side lobes. It is observed that $R(0)$, $R(2\pi/3)$ and $R(4\pi/3)$ are actually the DFT of $r(m)$ fitting the equation:

$$R(k) = \sum_{m=0}^{2} e^{j\theta_0 r_m} e^{-j\frac{2\pi}{M}mk} \text{ for } k = 0,1,2. \tag{29}$$



(a)



(b)

Figure 12. Magnitude of $R(k)$ for $\theta_0 = 2\pi \times 0.3$ and $\theta_0 = 2\pi \times 1.3$ and $k = 0,1,2$ [1]

The coefficients of the main lobe along with the side lobes for $\theta_0 = 2\pi \times 0.3$ and $\theta_0 = 2\pi \times 1.3$ are shown in Figure 11 for all three values of k .

The DFT of $r(m)$ can therefore be used to estimate the side lobe to main lobe ratio as follows [1]

$$\textit{Side Lobe to Main lobe ratio} = \frac{[R(1) + R(2)]/2}{R(0)} \tag{30}$$

Using this ratio, the center frequency of the original signal can be determined.

CHAPTER VI.

ALGORITHM

6.1 Algorithm

Below is the algorithm that is used in this thesis for the practical implementation of paper [1] to detect the frequency and reconstruction of nonuniformly sampled super high frequencies using sub-Nyquist sampling.

1. Recurrent nonuniform sampling of the input signal.
2. The reference main lobe to side lobe ratio is calculated using equation (31) for the normalized frequencies by computing gains of lobes. Magnitude spectrum of these gains help us to determine the reference ratio. The test ratio will be compared with this reference ratio.
3. The test ratio is calculated for the nonuniformly sampled signal by padding zeros to improve the resolution. Equation (31) is practically implemented in MatLab to automatically detect the test ratio by locating the main lobe and side lobes.
4. The center frequency becomes harder to calculate if the main lobe appears at the one-third and two-third of the sampling frequency. However, it can still be detected by observing the amplitude of the side lobes.
5. The output is then passed through a bandpass filter to remove the side lobes introduced due to nonuniform sampling. This is done by first computing the impulse response of the band pass filter and then convolving the nonuniformly sampled signal with the impulse response.
6. If the signal frequency is lower than half the sampling frequency, the output of the filtering process in step 5 gives a reconstructed signal without any side lobes.

7. If the signal frequency is higher than half the sampling frequency, the bandpass filtering is followed by interpolation for effective reconstruction.
8. The whole process can be verified by drawing comparisons with the outputs from a uniformly sampled signal.

6.2 Frequency detection block diagram

The schematic diagram shown in Figure 13 shows the various functional blocks required in the detection of the center frequency of an input through recurrent nonuniform sampling.

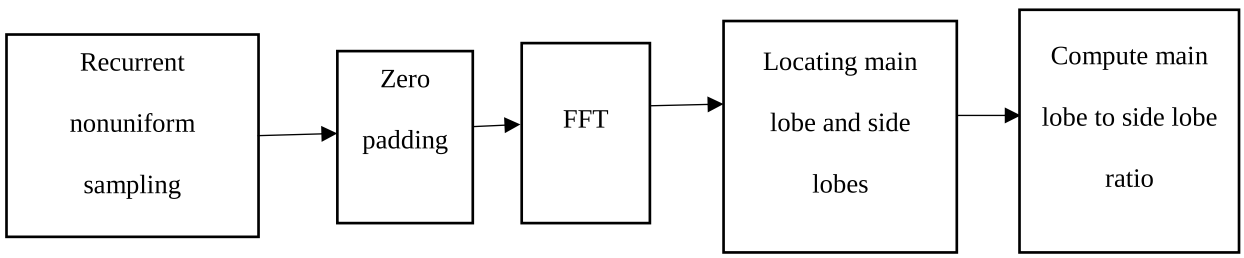


Figure 13. Detection of center frequency of input Test signal

6.3 Results

The above algorithm is the one developed in paper [1]. In this thesis, we implement this algorithm in code. The first step in obtaining the main lobe to side lobe ratio is to first perform nonuniform sampling on the input signal, followed by zero padding and DFT of the result. The zero padding is done to improve the resolution of the result. Next, the main lobe and side lobes are located in the result and their amplitudes are measured. Finally, the test ratio can then be obtained by using the formula in equation (30). This process has been implemented in Matlab, for which the code has been included in the appendix.

Table 2. Practical and theoretical ratios for different frequencies of a pure sinusoidal signal

Center Frequency ($\times 2\pi$)	**Test Ratio**	**Reference Ratio**	**Error Percentage**
0.2	0.020950	0.021665	3.412778
0.4	0.041934	0.042198	0.630106
0.6	0.063132	0.063132	0.229063
0.8	0.084145	0.084362	0.257554
0.9	0.094774	0.094774	0.000000

Table 3. Test and reference ratios for different frequencies of an FM signal

Center Frequency ($\times 2\pi$)	**Test Ratio**	**Reference Ratio**	**Error Percentage**
0.4	0.043146	0.042902	2.890972
0.8	0.086701	0.084145	3.038165
1.4	0.149514	0.148606	0.611488
1.8	0.196598	0.192783	1.978819
2.4	0.262532	0.261683	0.324568
2.8	0.314678	0.309920	1.535035
3.4	0.387961	0.386821	0.294803
3.8	0.447089	0.441870	1.180966
4.4	0.533540	0.531670	0.351715
4.8	0.602610	0.597456	0.862806

The test ratios obtained practically are then compared with the theoretically calculated reference ratios, and the error percentage is calculated which tells us the deviation of the practical ratios from their reference ratios. Tables 2 and 3 show the practical and reference ratios for different frequencies of sinusoidal and FM input signals.

The error percentage has also been calculated in each case. It is seen that for pure sinusoidal input signals, at the normalized frequency of 0.9 and for all frequencies thereafter, the error percentage becomes zero.

The MatLab plots that helped in the calculation of the above-mentioned ratios are shown below for various normalized frequencies and for both sinusoidal and FM input signals. The first part of the plot shows the magnitude spectrum of the nonuniformly sampled input signal and the second part shows the reference main/side lobes spectrum. It can be noted that with an increase in frequency of the input signal, the amplitudes of the lobes also increase.

Purely sinusoidal input signals:

The sinusoidal signal which is sampled nonuniformly with a sampling rate of 1 and normalized frequency f_0 is chosen to be,

$$x(t) = \cos(2\pi * f_0 * t) \tag{32}$$

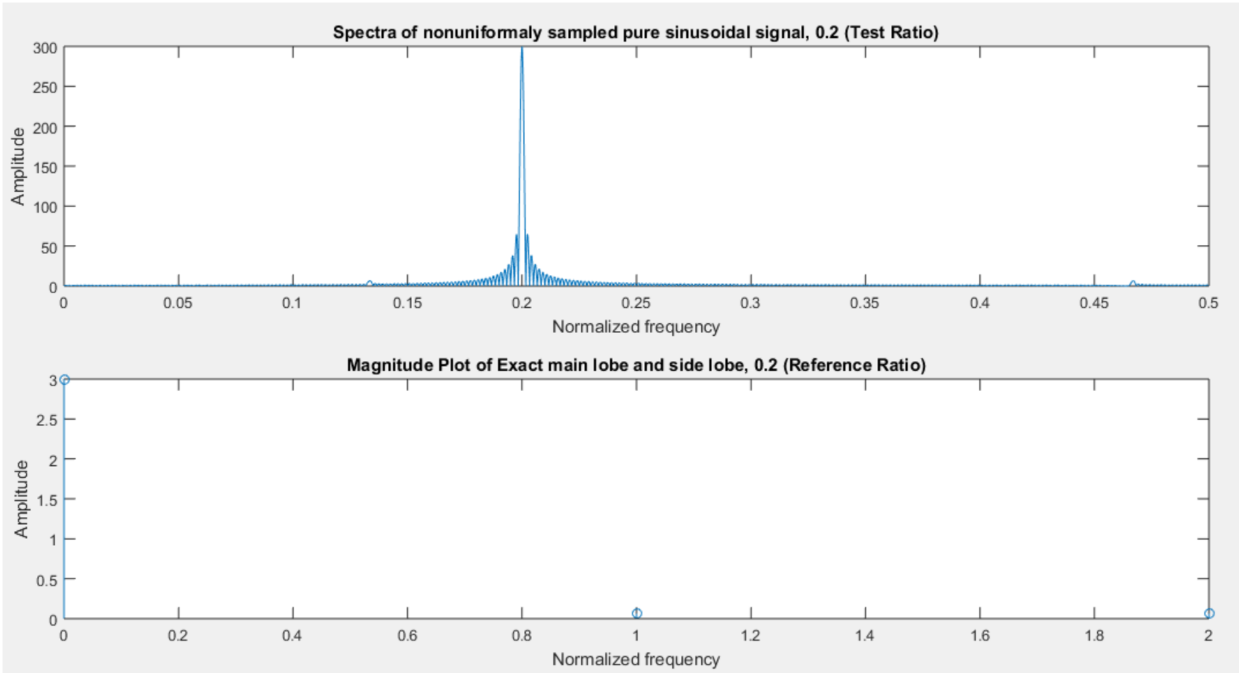


Figure 14. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.2

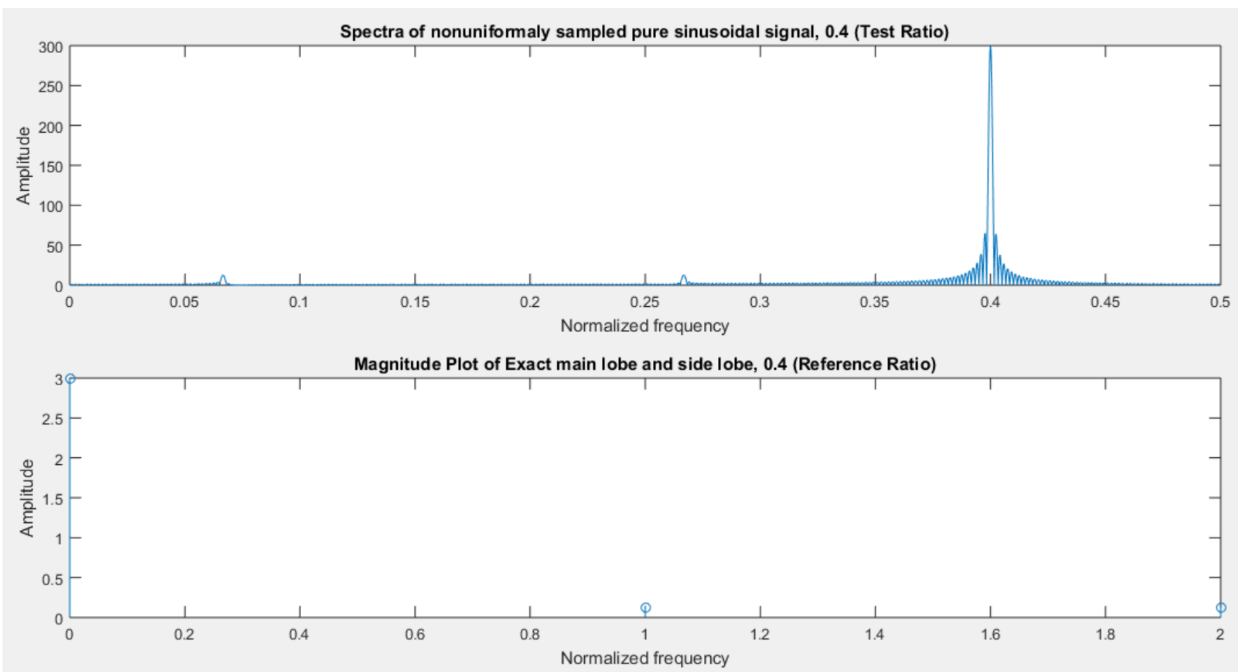


Figure 15. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.4

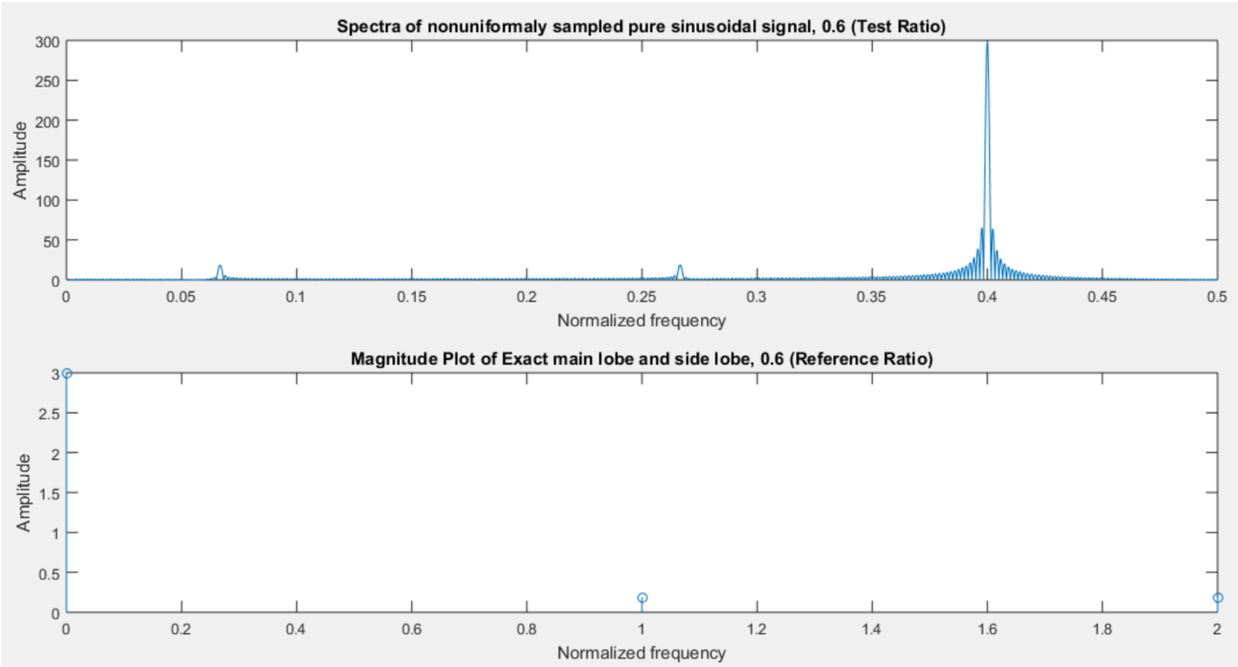


Figure 16. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.6

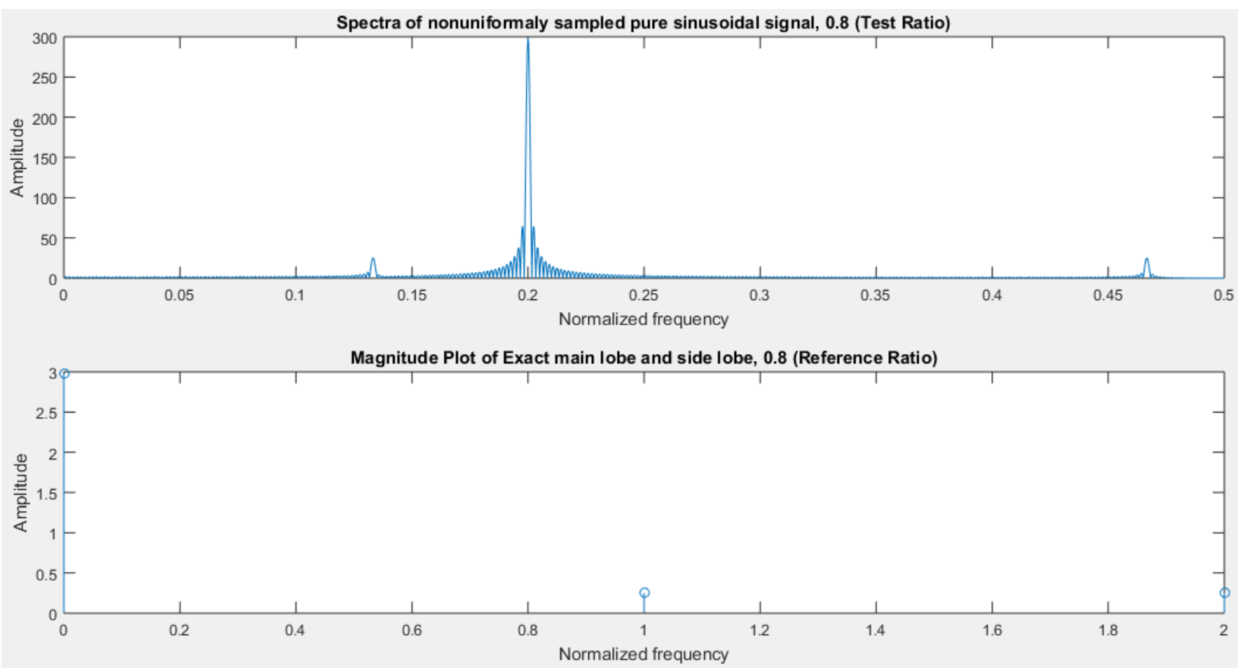


Figure 17. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.8

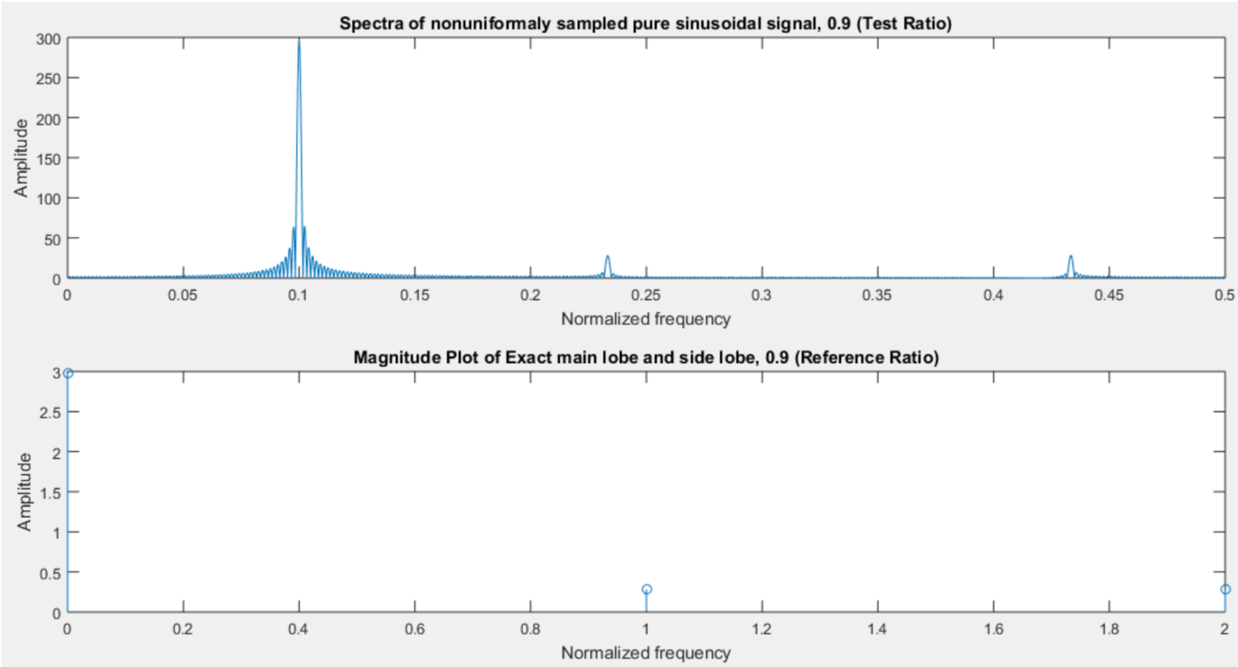


Figure 18. Test and Reference ratio spectra of sinusoidal signal for normalized frequency 0.9

FM input signals:

The FM signal which is sampled nonuniformly with a sampling rate of 1 and normalized frequency f_0 and modulated frequency f_m of 0.001 is chosen to be,

$$x(t) = \cos(2\pi * f_0 * t + \sin(2\pi * f_m * t)) \tag{33}$$

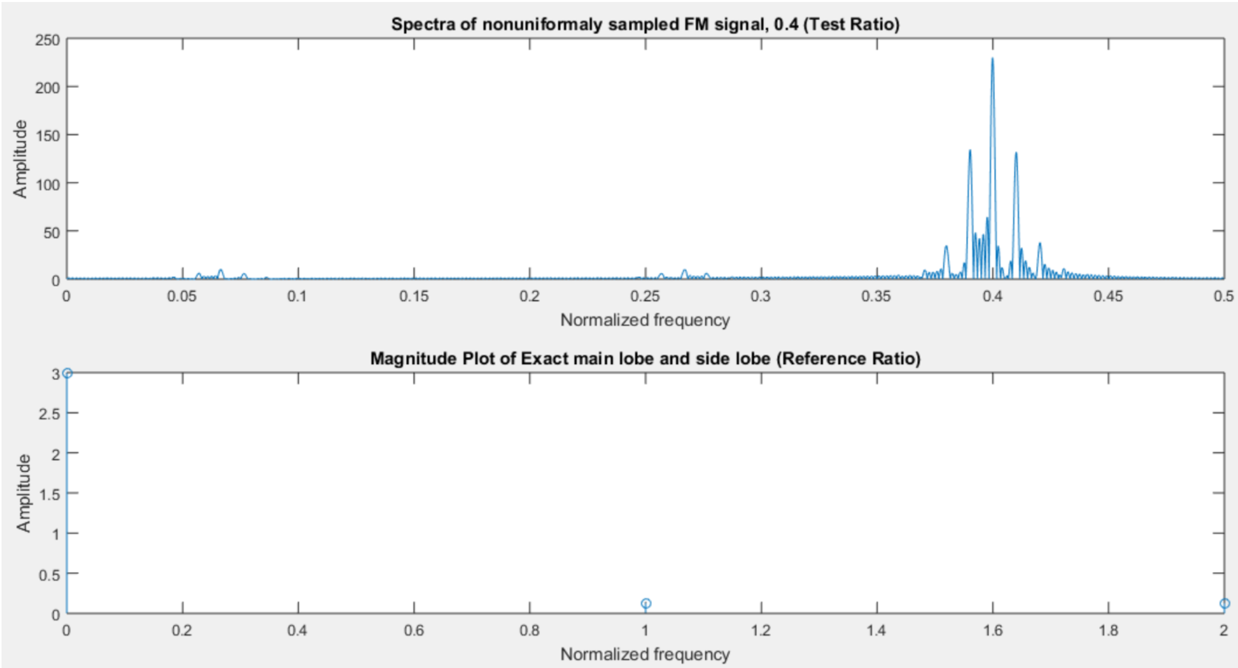


Figure 19. Test and Reference ratio spectra of FM signal for normalized frequency 0.4

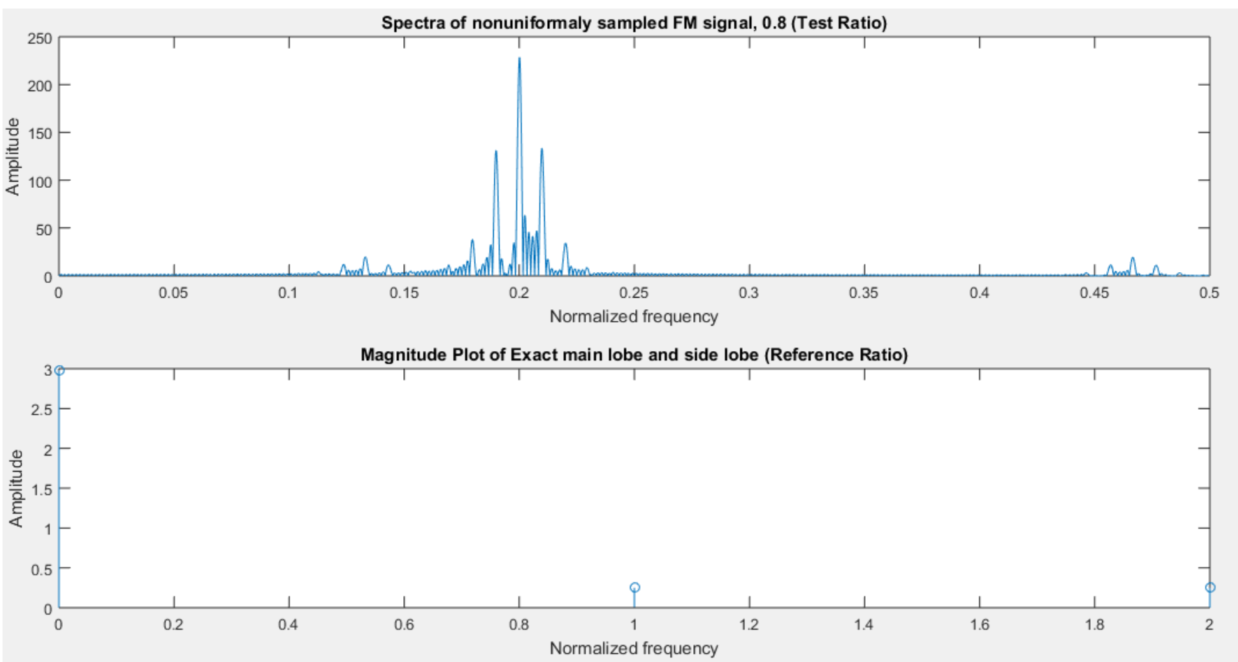


Figure 20. Test and Reference ratio spectra of FM signal for normalized frequency 0.8

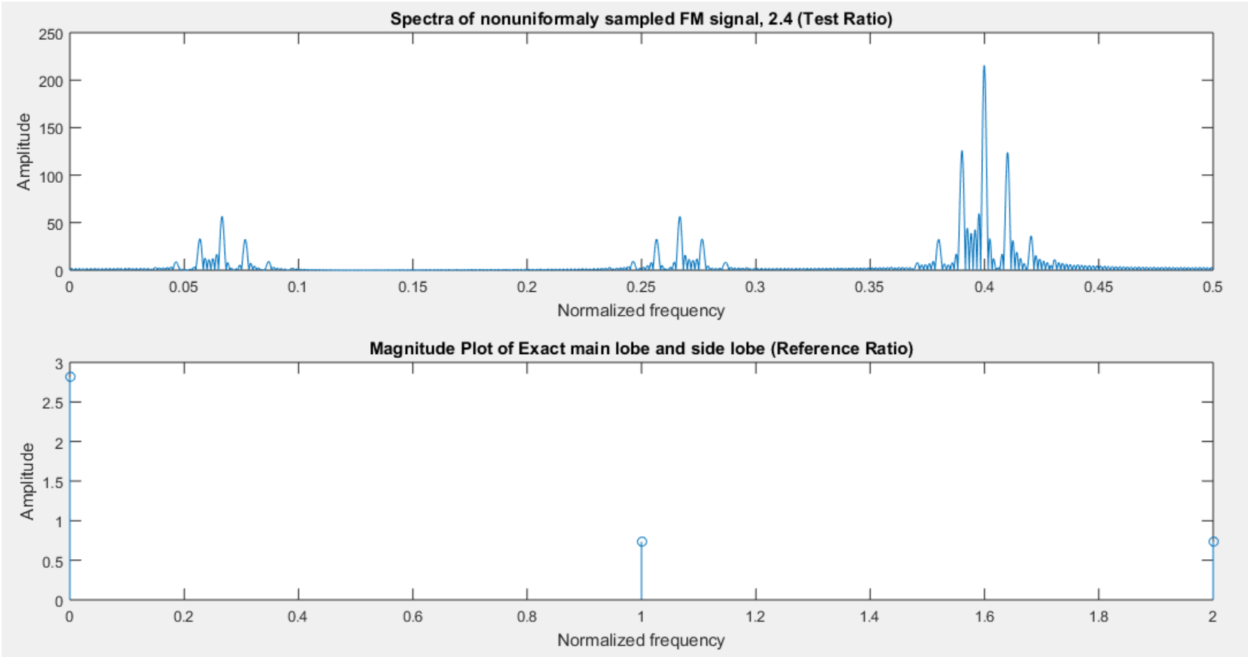


Figure 21. Test and Reference ratio spectra of FM signal for normalized frequency 2.4

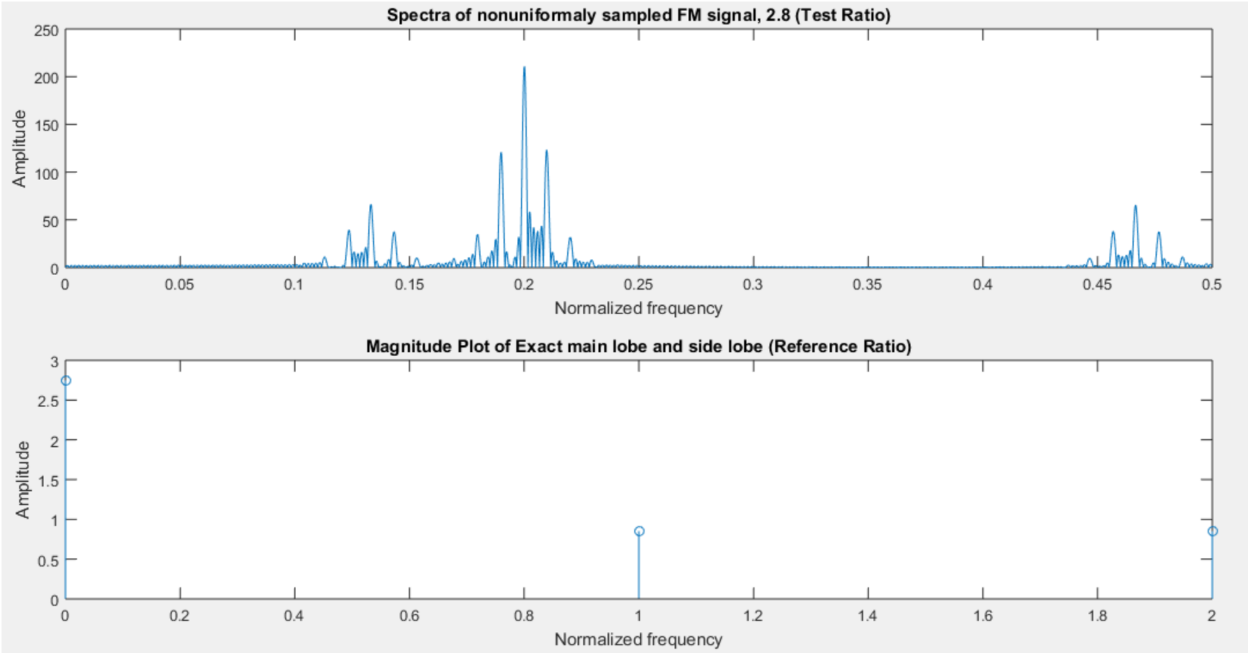


Figure 22. Test and Reference ratio spectra of FM signal for normalized frequency 2.8

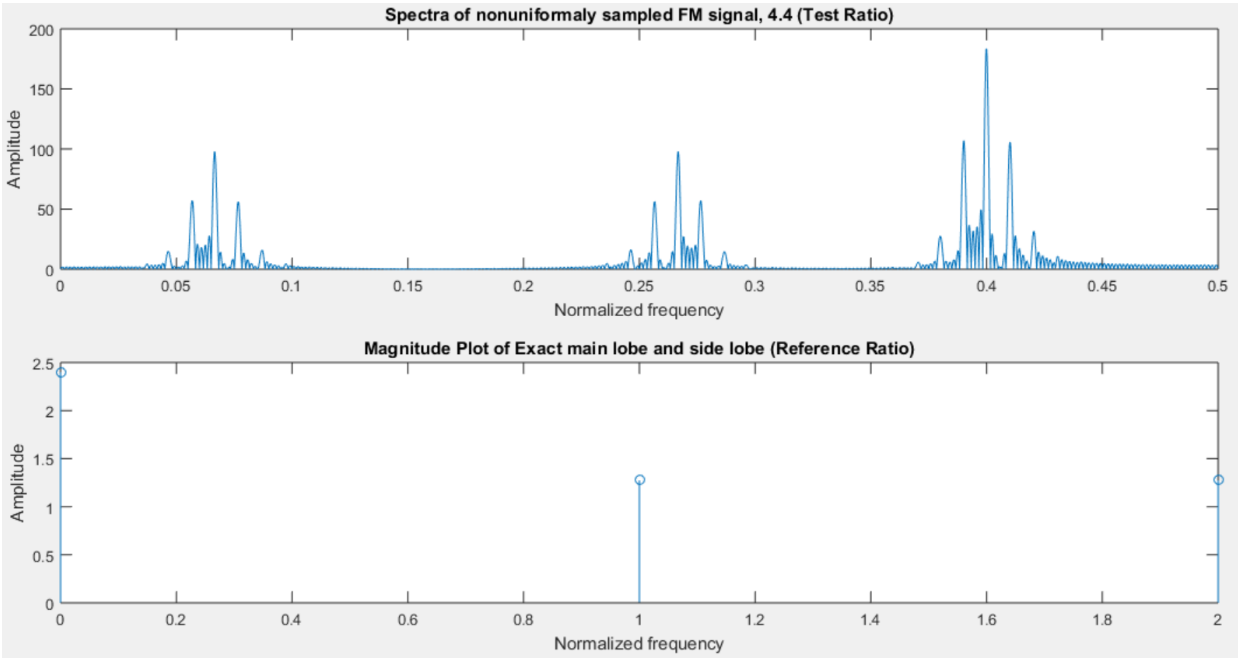


Figure 23. Test and Reference ratio spectra of FM signal for normalized frequency 4.4

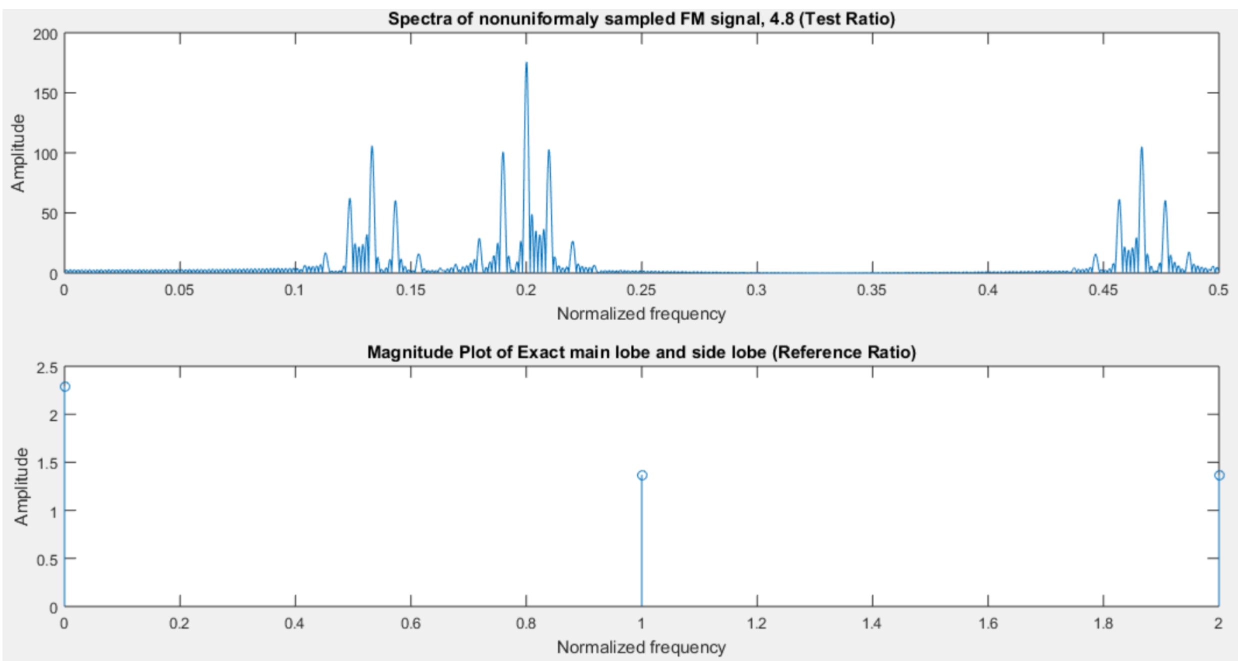


Figure 24. Test and Reference ratio spectra of FM signal for normalized frequency 4.8

From Tables 2 and 3, it is clear that for most values of normalized frequencies, the error percentage does not exceed 10%. This however, doesn't hold true when the main lobe appears at the one-third or two-third the sampling frequency. In these cases, the error percentages tend to be greater than 10%, which is too high. This occurs because we take 3-point nonuniform sampling ratios in our code. At these frequencies we see that the main lobe merges with one of the side lobes resulting in a peak of very high amplitude. It then becomes difficult to find the true amplitude of the main lobe and the side lobes.

That said, on observing the varying amplitudes of the side lobes for a variety of such frequencies, it is seen that a method of estimation can be set up. In Table 4, we observe that the amplitude of the aliasing side lobe varies in a certain range for input frequencies of a certain range. For example, for frequencies from 0.3333 to 0.333333 the side lobe amplitude ranges between 1 and 2. This range of amplitudes is placed in category 0. Similarly, for frequencies from 2.3333 to 2.333333 the side lobe amplitudes range between 9 and 12. This range of amplitudes is placed in category 2. And so, in this way, the range of amplitudes for all such frequencies is calculated and each range is assigned specific categories. Now, whenever an amplitude from one of these ranges is obtained, the center frequency (CF) can simply be calculated using the formula given below:

$$CF = C + MLF \tag{34}$$

where C is the category to which the amplitude range belongs

and, MLF is the main lobe frequency

Table 4. Amplitude of the side lobe for one-third and two-third sampling frequencies.

Center Frequency ($\times 2\pi$) (CF)	**Main Lobe Frequency (MLF)**	**Side Lobe Amplitude**	**Category (C)**
0.333	0.3330	2.6379	0
0.3333	0.3333	1.5200	0
0.33333	0.3334	1.1921	0
0.333333	0.3334	1.2158	0
1.333	0.3330	8.2772	1
1.3333	0.3333	2.7637	1
1.33333	0.3334	3.7113	1
1.333333	0.3334	3.8094	1
2.333	0.3330	41.4342	2
2.3333	0.3333	9.3006	2
2.33333	0.3334	10.9831	2
2.333333	0.3334	11.1482	2
3.333	0.3330	53.7589	3
3.3333	0.3333	19.2738	3
3.33333	0.3334	21.4827	3
3.333333	0.3334	21.6966	3
4.333	0.3330	67.0041	4
4.3333	0.3333	31.6080	4
4.33333	0.3334	34.1399	4
4.333333	0.3334	34.3817	4

For example, let's say we come across a sampled output with the main lobe located at a normalized frequency of 0.3334 with a side lobe of amplitude 10.9. From Table 4, it is clear that this falls in category 2. Therefore, the center frequency (CF) can be calculated as follows:

$$CF = C + MLF = 2 + 0.3334 = 2.3334 \tag{35}$$

Therefore, we can say that if the error percentage is observed to be greater than 10% and if the side lobe amplitude falls within one of the ranges seen in the Table 4, then the center frequency can simply be obtained by using equation (34).

There is still, however, the question of what to do with the outliers. For example, for a center frequency of 2.333 we see that the side lobe amplitude is 41.4, while the other values in the same category, range from 9 to 12. Similarly, for a frequency of 3.333 the side lobe amplitude is found to be 53.7. But the other values in category 3 range from 19 to 22. And even if these values were noted along with the frequencies at which they appeared, the problem doesn't get solved. Because any noise in the signal may change these values and make them fall in one of the other categories. For example, at a frequency of 0.333, the side lobe amplitude is 2.6. This is very close to category 1's range of values that start from 2.7. In this case, even a small amount of noise in the signal could increase the value of 2.6 to something in the range of category 1, thereby leading us to an erroneous situation. Similarly, the side lobe amplitude of 8.2 at a frequency of 1.333 could be increased to fall in the range of values of category 2 (from 9 to 12).

CHAPTER VII

RECONSTRUCTION

Once the center frequency of the signal is determined, the output is passed through a band pass filter to remove the side lobes introduced during nonuniform sampling. Now, if the signal frequency is lower than half the sampling frequency, it means the signal is not folded, and the output of the filtering process gives the reconstructed signal without any side lobes. However, if the signal frequency turns out to be higher than half the sampling frequency, then the signal appears folded and must undergo interpolation before it can be filtered.

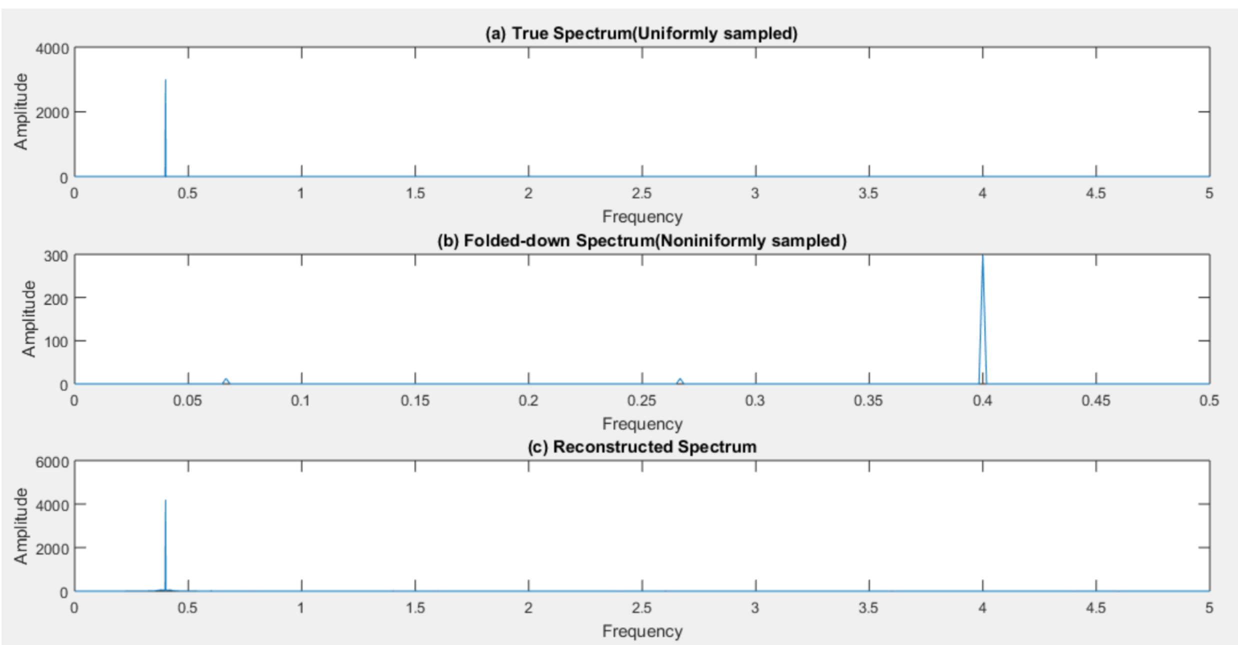


Figure 25. Nonuniform sampling and reconstruction of an unfolded sinusoidal signal (0.4)

Let us consider a few examples for each case. Figures 25, 26 and 27 show the nonuniform sampling and reconstruction of an input sinusoidal signal. In Figure 25, an input with a normalized frequency of 0.4 is used. It is seen that the recurrent nonuniform sampling of this signal yields a main lobe at

0.4 and two small side lobes. Now, since its frequency is lower than half the sampling frequency (0.5), this signal remains unfolded and is reconstructed by simply passing the output of the sampling process through a band pass filter. The reconstructed signal is located at 0.4.

In Figure 26, an input with a normalized frequency of 2.7 is used. The nonuniformly sampled output of this signal is seen to have its main lobe at 0.3. This occurs due to folding of the signal, which takes place because its frequency is higher than half the sampling frequency (0.5). The sampled output signal is a folded mirror image of what it should have been. In this case, the main lobe should have been at 0.2 (2.7 − highest deductible multiple of 0.5), but due to folding, its mirror image places the main lobe at 0.3. The reconstructed signal is located at 2.7.

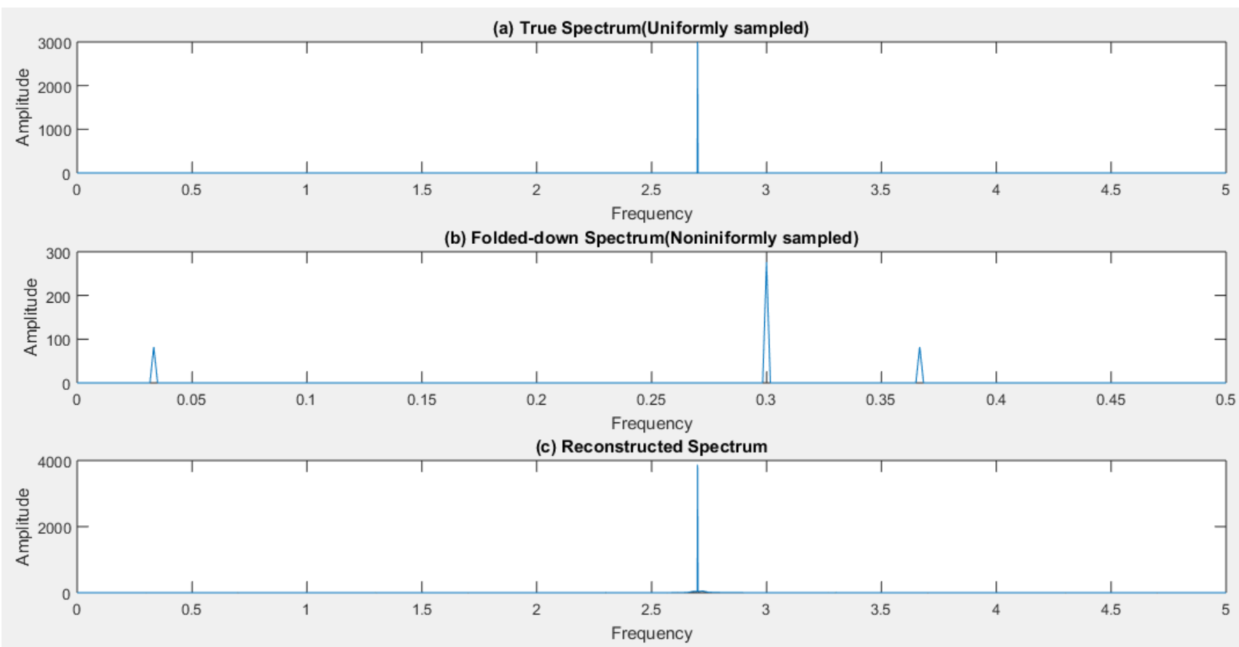


Figure 26. Nonuniform sampling and reconstruction of a folded input sinusoidal signal (2.7)

In Figure 27, an input with a normalized frequency of 4.6 is used. The nonuniformly sampled output of this signal is seen to have its main lobe at 0.4 due to folding. 0.4 is a mirror image of 0.1 (4.6 – highest deductible multiple of 0.5). The reconstructed signal is located at 4.6.

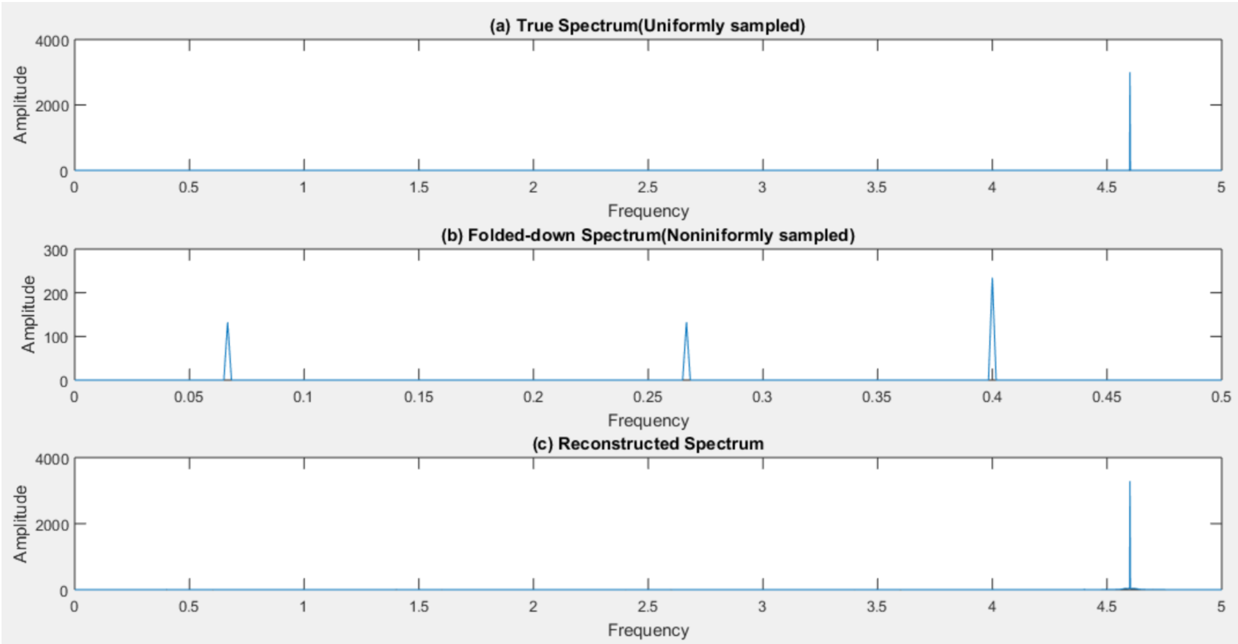


Figure 27. Nonuniform sampling and reconstruction of a folded input sinusoidal signal (4.6)

Figures 28, 29 and 30 show the recurrent nonuniform sampling and reconstruction of FM signals for the same normalized input frequencies as taken above.

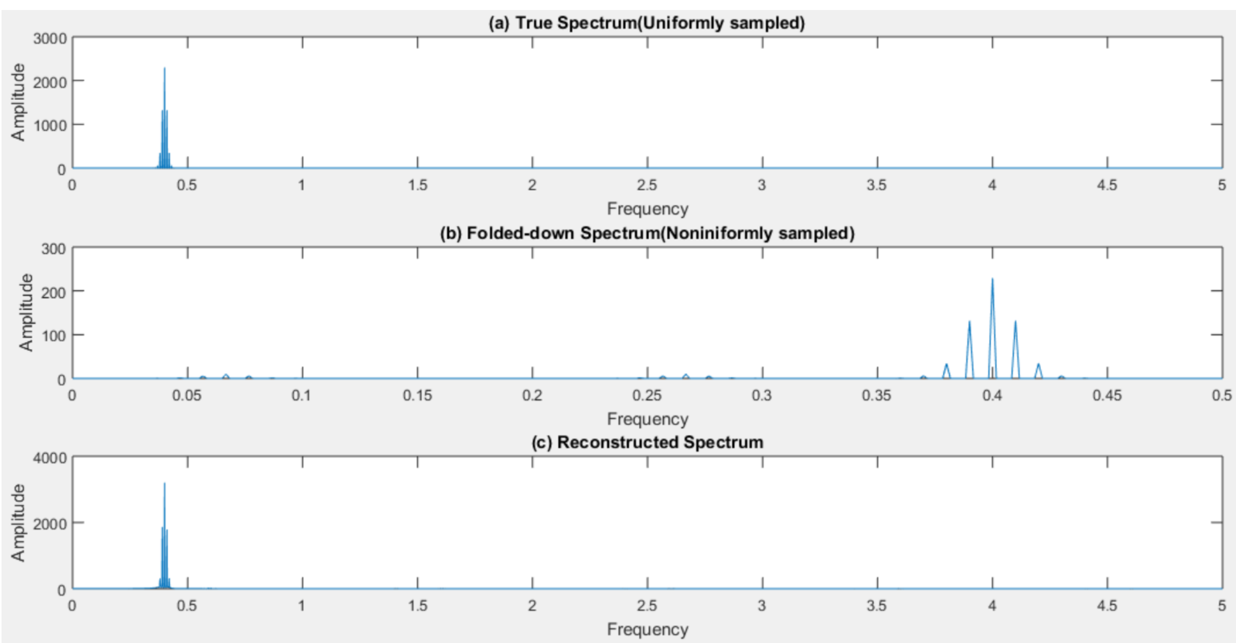


Figure 28. Nonuniform sampling and reconstruction of an unfolded input FM signal (0.4)

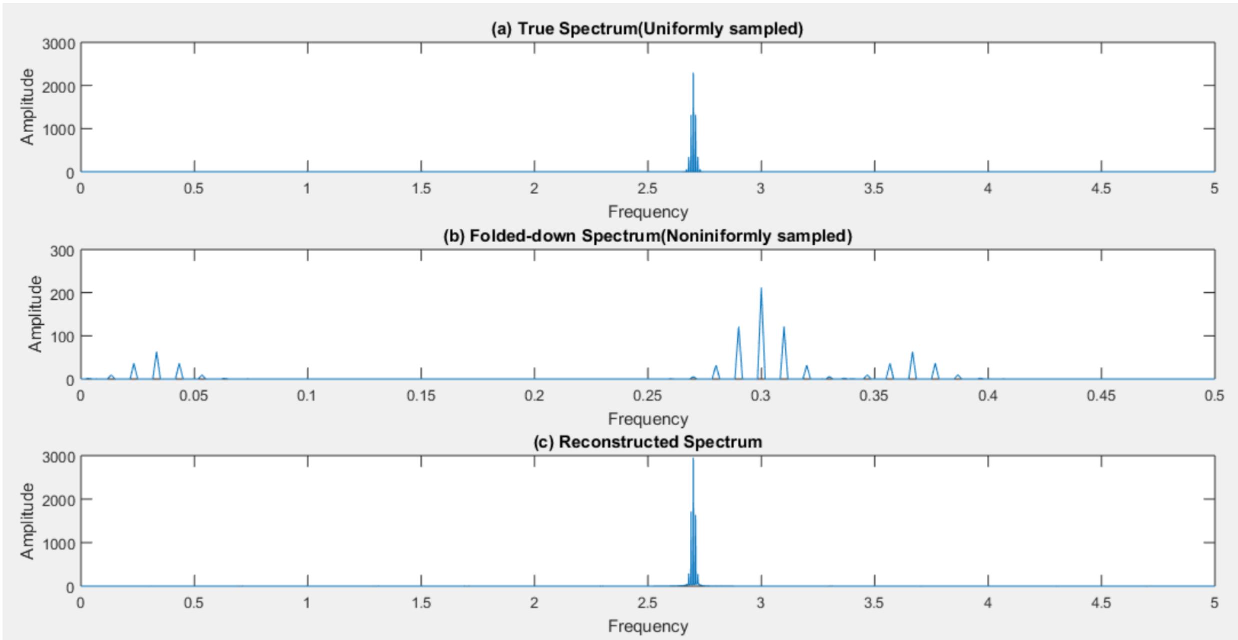


Figure 29. Nonuniform sampling and reconstruction of a folded input FM signal (2.7)

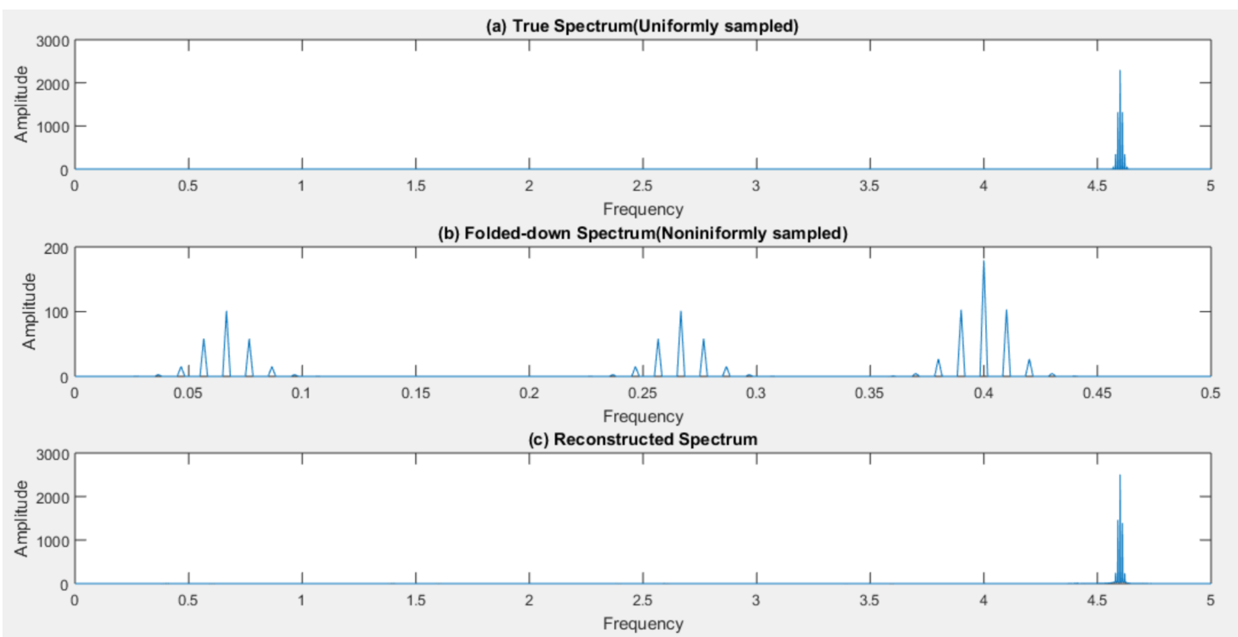


Figure 30. Nonuniform sampling and reconstruction of a folded input FM signal (4.6)

In each case, after the input signal is reconstructed, it is compared with the spectrum of the same signal sampled uniformly so as to observe any inconsistencies. The code for this method has been implemented in Matlab and included in the appendix. The code accepts different values for the number of ADCs, number of samples taken per ADC and order of the bandpass filter.

CHAPTER VIII

APPLICATIONS

The method of nonuniform under-sampling and reconstruction of a signal that has been discussed in this thesis, can be applied to all scenarios where it is not possible to sample at the Nyquist rate. The Nyquist rate in turn depends on the frequency of the signal, and so the Nyquist rate criterion is particularly difficult to meet in applications that deal with very high frequencies. A few such applications have been discussed below:

1. ***Remote Sensing*****:** Remote sensing involves acquiring information about an object or phenomenon without making any physical contact with it. Here, high frequency pulses are transmitted at the foreign objects and the reflected pulses help understand certain characteristics about the object. Remote sensing is used in many applications like military, geology and oceanography.
2. ***Communication***: High frequency radio waves are used for long distance communication because they have larger information carrying capacity and are less susceptible to interference from electronic sources but more susceptible to interference from physical obstacles.
3. ***Radar***: A radar transmits and receives high frequency radio waves so as to measure the time of arrival of the reflected pulses from distant objects. This helps determine the distance to the object. High frequency radars are also used by ocean researchers to measure the surface current velocity near the coast. Besides these, radar systems are used in a wide array of applications from detection of aircrafts and celestial objects to determining weather conditions and many more.

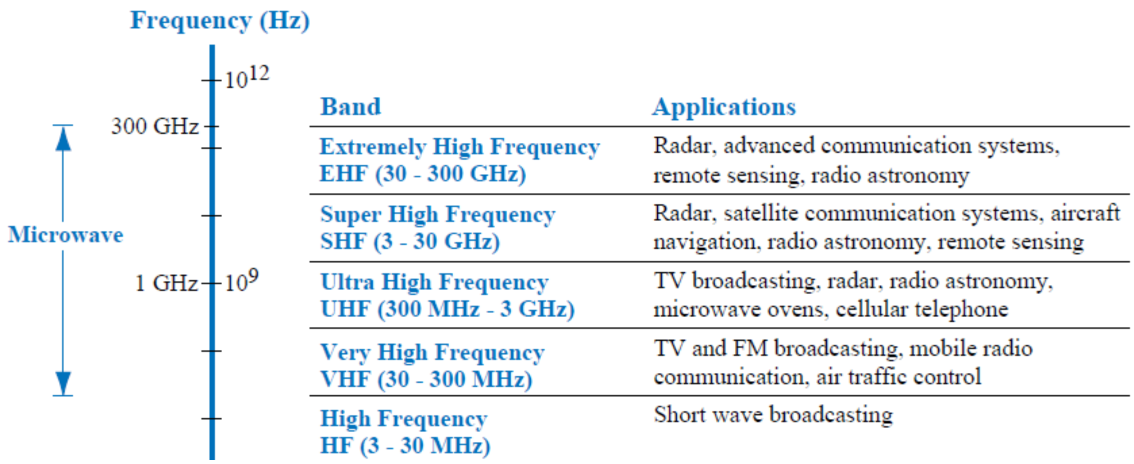


Figure 31. Applications of high frequency signals [22]

CHAPTER IX

CONCLUSION

We use super high frequency radio signals in a large variety of applications today, especially in the communication and radar sectors. The sampling of these high frequency signals at the Nyquist rate puts a lot of stress on present day ADCs and memory locations. However, it has been observed that the level of information contained in a signal is often much lower than the actual bandwidth, which means that better schemes can be used to overcome the stress. One such scheme was developed in paper [1] which put forth the idea of using nonuniform under-sampling to estimate the frequency of the original signal. This thesis practically implements the idea put forth in that paper and also implements a method for reconstruction of those signals. This method employs the use of recurrent nonuniform sampling coupled with interpolation and band pass filtering to effectively sample and reconstruct high frequency signals.

The recurrent nonuniform under-sampling of a signal introduces side lobes. Using these side lobes, a side lobe to main lobe ratio is calculated, which in turn helps estimate the frequency of the input signal. However, when the input signal is nonuniformly sampled at one-third and two-third the sampling frequency, it results in the aliasing of the main lobe with one of the side lobes. This problem is overcome by observing the range of the signal and then calculating the estimated frequency using the derived formula. A narrow bandpass filter is then designed using this estimated frequency as the center frequency, to reconstruct the high frequency input signal. If the frequency of the input signal is higher than half the sampling frequency, it results in the folding down of the sampling output. In this case, the signal has to be bandpass filtered and then interpolated for effective reconstruction.

CHAPTER X

FUTURE WORK

In this thesis, we find a way to determine the frequency estimate of the original signal at one-third and two-third the sampling frequency. However, the issue of how to be able to distinguish the outlying values in the various category ranges, still needs to be addressed. Also, reconstructing such a signal is still a challenge since the main lobe merges with one of the sidelobes. Due to this, the bandpass filtering technique employed in our current method cannot differentiate the main lobe from the side lobe. Therefore, it is not possible to eliminate the side lobes and get back the original signal. This is the challenge that needs to be overcome in future work.

REFERENCES

[1] S. Park and R. Kestur, “Detection of Ultra High Frequency Narrow Band Signal Using Nonuniform Sampling,” *Proceedings of the International Conference on Modeling, Simulation and Visualization Methods*, Las Vegas, Nevada, pp. 82-8, July 2017.

[2] Sung-won Park. *Essence of Digital Signal Processing*, Linus Publications, Inc., New York, 2011.

[3] Website: https://www.diva-portal.org/smash/get/diva2:829911/FULLTEXT01.pdf (last accessed on 11/27/2017) Satish kumar Are, Manoranjan Reddy Thangalla, Saikrishna Gajjala, “*Software Implementation of Digital Filters*”: Electrical Engineering Blekinge Institute of Technology, August 2008.

[4] E. J. Candes and M. B. Wakin, "An Introduction to Compressive Sampling," in *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21-30, March 2008.

[5] G. L. Fudge, R. E. Bland, M. A. Chivers, S. Ravindran, J. Haupt and P. E. Pace, "A Nyquist folding analog-to-information receiver,” *2008 42nd Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, pp. 541-545 CA, 2008.

[6] S. Kirolos *et al*., "Analog-to-Information Conversion via Random Demodulation," *2006 IEEE Dallas/CAS Workshop on Design, Applications, Integration and Software*, Richardson, TX, pp. 71-74, 2006.

[7] R. Maleh, G. L. Fudge, F. A. Boyle and P. E. Pace, "Analog-to-Information and the Nyquist Folding Receiver," in *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, no. 3, pp. 564-578, Sept. 2012.

[8] R. G. Baraniuk, “Compressive Sensing [Lecture Notes],” in *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118-121, July 2007.

[9] S. Park, “Spectrum Reconstruction from Recurrent Nonuniform Sampling with known Uniform Sampling Ratios,” *Proceedings of the IASTED International Conference on Signal and Image Processing (SIP 2013)*, Banff, Canada, pp. 356-360, July 17-19, 2013.

[10] P. Sommen and K. Janse, “On the Relationship Between Uniform and Recurrent Nonuniform Discrete-Time Sampling Schemes,” in *IEEE Transactions on Signal Processing*, vol. 56, no. 10, Oct. 2008.

[11] S. Park, W. Hao and C. S. Leung, “Reconstruction of uniformly sampled sequence from nonuniformly sampled transient sequence using symmetric extension,” *IEEE Trans. on Signal Processing*, vol. 60, no. 3, pp.1498-1451, March 2012.

[12] Junyi Luo and Lin Lei, “Periodic Nonuniform Sampling for Sparse Signals in shiftinvariant Spaces”, *International Journal of Hybrid Information Technology*, vol. 8, no. 12, pp. 421-432, 2015.

[13] Y. C. Jenq, "Perfect reconstruction of digital spectrum from nonuniformly sampled signals," *IEEE Instrumentation and Measurement Technology Conference Sensing, Processing, Networking. IMTC Proceedings*, Ottawa, Ont., vol. 1 pp. 624-627,1997.

[14] J. J. Wojtiuk, “Randomised sampling for radio design,” Ph.D. dissertation, Univ. South Australia, Mawson Lakes, Australia, 2000.

[15] I. Bilinskis and G. Cain, "Digital alias-free signal processing tolerance to data and sensor faults," *IEE Colloquium on Intelligent Sensors (Digest No: 1996/261)*, Leicester, pp. 6/1-6/6,1996.

[16] Eng, F. (2007), “Non-Uniform Sampling in Statistical Signal Processing”. (Doctoral dissertation). Institutionen for systemteknik.

[17] M. Rashidi and S. Mansouri, "Parameter selection in periodic nonuniform sampling of multiband signals," *2010 3rd International Symposium on Electrical and Electronics Engineering (ISEEE)*, Galati, pp. 79-83, 2010.

[18] E. Candès and J. Romberg, “Sparsity and incoherence in compressive sampling,” Inverse Prob., vol. 23, no. 3, pp. 969–985, 2007.

[19] D. Donoho, “Compressed Sensing,” *IEEE Trans. Inf. Theory*, vol. 42, no. 4, pp. 1289–1306, Apr. 2006.

[20] E. Candes, J Romberg and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[21] R. Maleh, G. Fudge, F. Boyle, *Member*, and P. Pace, “Analog-to-Information and the Nyquist Folding Receiver,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, Vol. 2, No. 3, pp. 564-578, September 2012

[22] F. Ulaby and D. B. Lang, "A strategy for active remote sensing amid increased demand for spectrum," *2015 USNC-URSI Radio Science Meeting (Joint with AP-S Symposium)*, Vancouver, BC, page 14, 2015.

APPENDIX

MATLAB SIMULATED CODE

Code for calculating the main lobe to side lobe ratio using theoretical and practical values and calculating error percentage:

1. A pure sinusoidal signal with normalized frequency.

a. <u>0.4:</u>

```
% To calculate the spectrum of the nonuniformly sampled signal
N=600;                                          % Number of samples
freq=0.4;                                       % Frequency 0.3, 0.7, 1.3, 2.7, 3.3, 3.7, etc.
rm=[0 0.05 0.05];                               % Nonuniform sampling ratio
am=[];
% Nonuniform sampling ratio for 1000 samples
for n=1:N/3,
   am=[am,rm];
end
am=[am,0];                                      % Pad zero to the 1000-th sampling
for n=1:N,
   xn(n)=cos(2*pi*freq*(n-1+am(n)));            % Nonuniform sampling
end
pad=6000;
XN=abs(fft(xn,pad));                             % Magnitude spectrum
XN=XN(1:3000);
XN=XN(1:(N*10)/2);
subplot(2,1,1);
plot(0:1/(N*10):((N*10)/2-1)/6000, XN) % Plot of spectrum
title(['Spectra of nonuniformaly sampled pure sinusoidal signal, ',num2str(freq),' (Test Ratio)']);
```

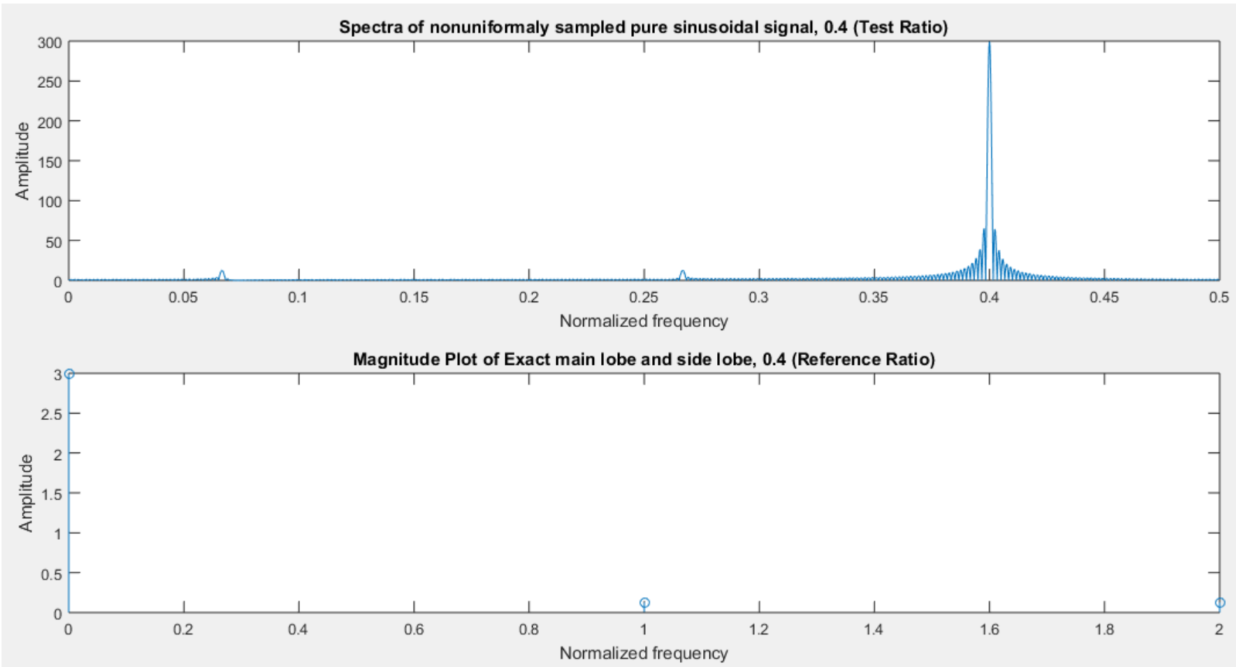
```
%finding peaks and location to calculate approximate ratio
[pks,locs] = findpeaks(abs(XN));
[maxvalue, index_of_max] = max(XN);
A=[pks;locs]
main_lobe=max(XN);                              %Value of main lobe
theta_nod=index_of_max/pad;                     %main lobe frequency

if (freq==0.1);
  SL1=(-theta_nod+1/3)*pad;
  SL2=(theta_nod+1/3)*pad;

elseif (freq==0.2);
  SL1=(-theta_nod+1/3)*pad;
  SL2=(-theta_nod+2/3)*pad;
else
SL1=(-theta_nod+2/3)*pad;             %calculated side lobe location
if(SL1>=(pad/2));
   m=SL1-(pad/2);
   SL1=(pad/2)-m;
end
SL2=(-theta_nod+1/3)*pad;
end


[~,I1] = min(abs(locs-abs(SL1)));     %choosing closest existing value to the calc sidelobe loc
c1 = locs(I1);
[~,I2] = min(abs(locs-abs(SL2)));
c2 = locs(I2);
[row1, column1] = find(A == c1)
[row2, column2] = find(A == c2)
```

```
sidelobe_1=pks(column1);
sidelobe_2=pks(column2);

if(abs((index_of_max) -(c1)) < (0.1*N));
 approximateratio=sidelobe_2/maxvalue;
 disp(approximateratio);
else
approximateratio=((sidelobe_1+sidelobe_2)/2)/maxvalue;
disp(approximateratio);
end

%sidelobe
% This calculates the exact side lobe to main lobe ratio
r=[0, 0.05, 0.05];                          % nonuniform sampling ratios
nf=2*pi*freq;                               % normalized frequency
s=exp(j*nf*r);
S=fft(s);                                   % computation of gains of lobes
Q=abs(S);                                   % magnitude of gains
subplot(2,1,2);
stem(0:2, Q)
title('Magnitude Plot of Exact main lobe and side lobe (Reference ratio)');
exactratio=(Q(2)+Q(3))/Q(1)/2               % ratio of side to main lobes

%error percentahe
ep=((exactratio-approximateratio)/exactratio)*100
disp(sprintf('Error Percentage: %f and Calculated Ratio: %f and Exact Ratio: %f \n Error
Percentage %f', ep, approximateratio, exactratio, abs(ep)))
```

Output:

Error Percentage: -0.630106 and Calculated Ratio: 0.042198 and Exact Ratio: 0.041934

Error Percentage 0.630106

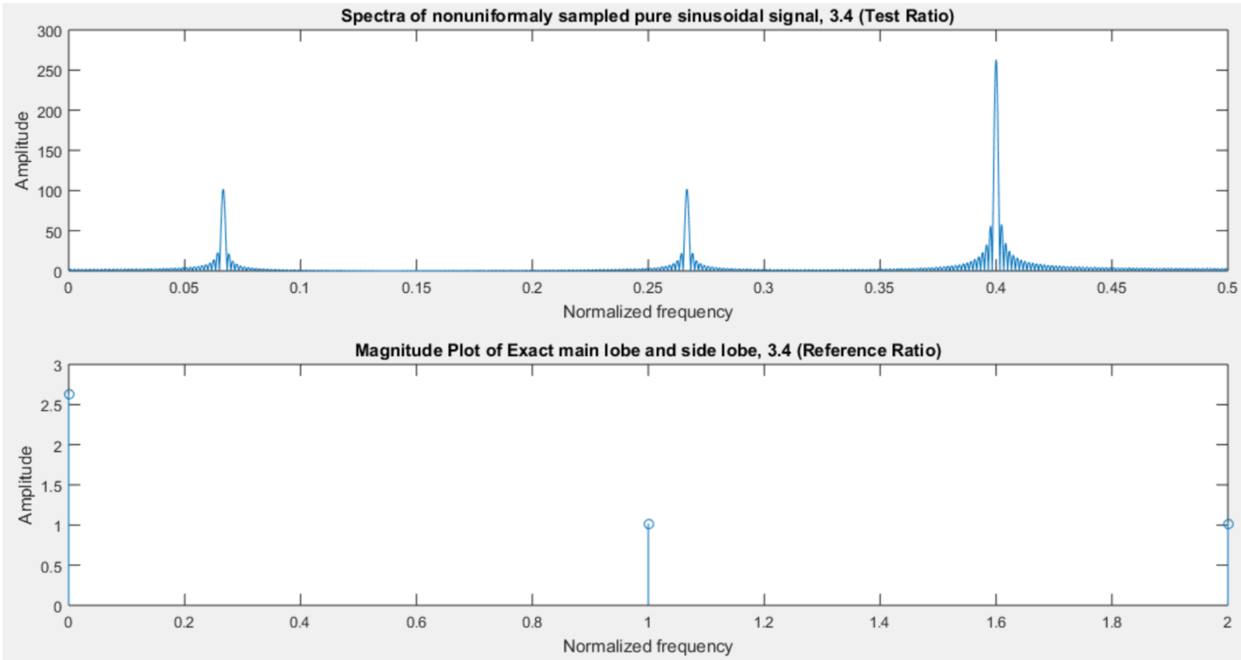


b. 3.4:

Output:

Error Percentage: -0.000000 and Calculated Ratio: 0.386821 and Exact Ratio: 0.386821

Error Percentage 0.000000


Spectra of nonuniformaly sampled pure sinusoidal signal, 3.4 (Test Ratio)
Amplitude
Normalized frequency
Magnitude Plot of Exact main lobe and side lobe, 3.4 (Reference Ratio)
Amplitude
Normalized frequency

2. **An FM signal with normalized frequency**

   a. **<u>0.4:</u>**

```
% To calculate the spectrum of the nonuniformly sampled signal
N=600;                                  % Number of samples
%fm=0.001;
freq=0.4;                               % Frequency 0.3, 0.7, 1.3, 2.7, 3.3, 3.7, etc.
rm=[0 0.05 0.05];                       % Nonuniform sampling ratio
am=[];
% Nonuniform sampling ratio for 1000 samples
for n=1:N/3,
   am=[am,rm];
end
am=[am,0];                              % Pad zero to the 1000-th sampling
for n=1:N,
      xn(n)=cos(2*pi*(n-1+am(n))*freq+sin(2*pi*0.01*(n-1+am(n)))); % Nonuniform sampling
end
XN=abs(fft(xn,6000));                   % Magnitude spectrum
XN=XN(1:3000);
XN=XN(1:(N*10)/2);
subplot(2,1,1);
plot(0:1/(N*10):((N*10)/2-1)/6000, XN)       % Plot of spectrum
title(['Spectra of nonuniformaly sampled pure sinusoidal signal, ',num2str(freq),' (Test Ratio)']);

%finding peaks and location to calculate approximate ratio
[pks,locs] = findpeaks(abs(XN));
[maxvalue, index_of_max] = max(XN);
A=[pks;locs]
main_lobe=max(XN);                      %Value of main lobe
theta_nod=index_of_max;                 %main lobe frequency
SL1=-theta_nod+(2/3*6000);              %calculated side lobe location
```

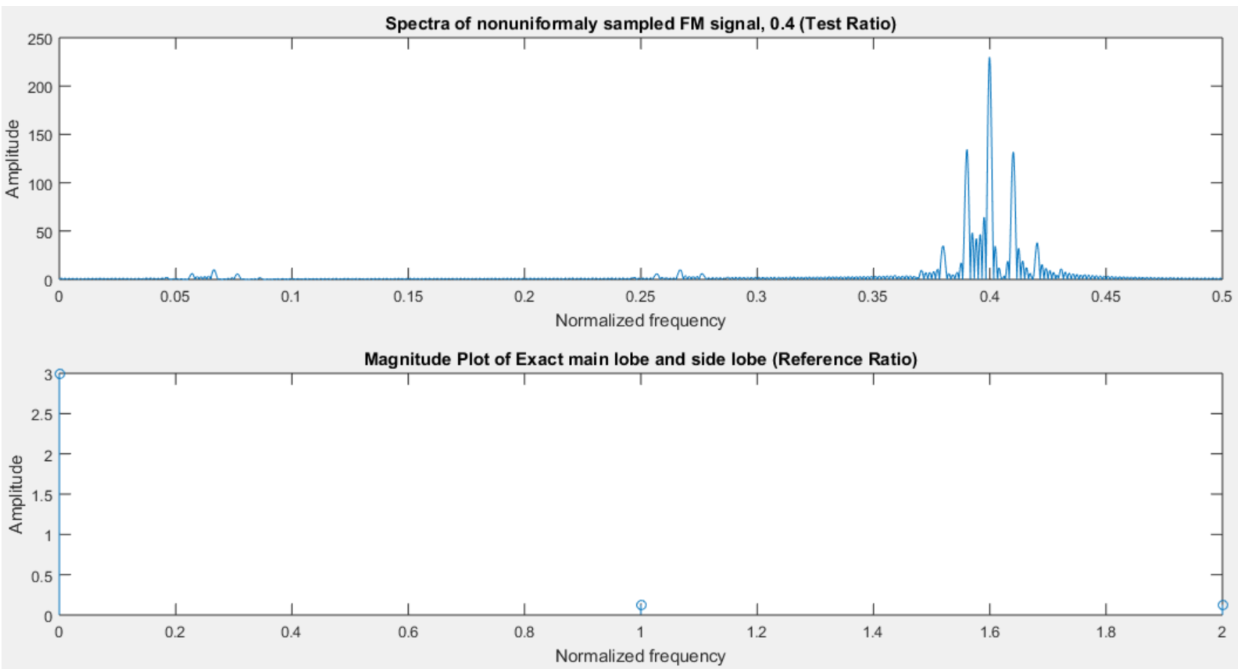
```
SL2=-theta_nod+(1/3*6000);
[~,I1] = min(abs(locs-abs(SL1)));      %choosing closest existing value to the calc sidelobe loc
c1 = locs(I1);
[~,I2] = min(abs(locs-abs(SL2)));
c2 = locs(I2);

[row1, column1] = find(A == c1)
[row2, column2] = find(A == c2)
sidelobe_1=pks(column1);
sidelobe_2=pks(column2);
if(abs(index_of_max)-abs(c1)<=85)
 Testratio=sidelobe_2/maxvalue;
 disp(Testratio);
else
Testratio=((sidelobe_1+sidelobe_2)/2)/maxvalue;
disp(Testratio);
end

%sidelobe
% This calculates the exact side lobe to main lobe ratio
r=[0, 0.05, 0.05];                                % nonuniform sampling ratios
nf=2*pi*freq;                                     % normalized frequency
s=exp(j*nf*r);
S=fft(s);                                 % computation of gains of lobes
Q=abs(S);                                 % magnitude of gains
subplot(2,1,2);
stem(0:2, Q)
title('Magnitude Plot of Exact main lobe and side lobe (Reference ratio)');
referenceratio=(Q(2)+Q(3))/Q(1)/2          % ratio of side to main lobes

%error percentage
```

```
ep=((referenceratio-Testratio)/referenceratio)*100
disp(sprintf('Error Percentage: %f and Test Ratio: %f and Reference Ratio: %f \n Error Percentage %f', ep, Testratio, referenceratio, abs(ep)))
```

Output:

Error Percentage: -2.890972 and Test Ratio: 0.043146 and Reference Ratio: 0.041934

Error Percentage 2.890972

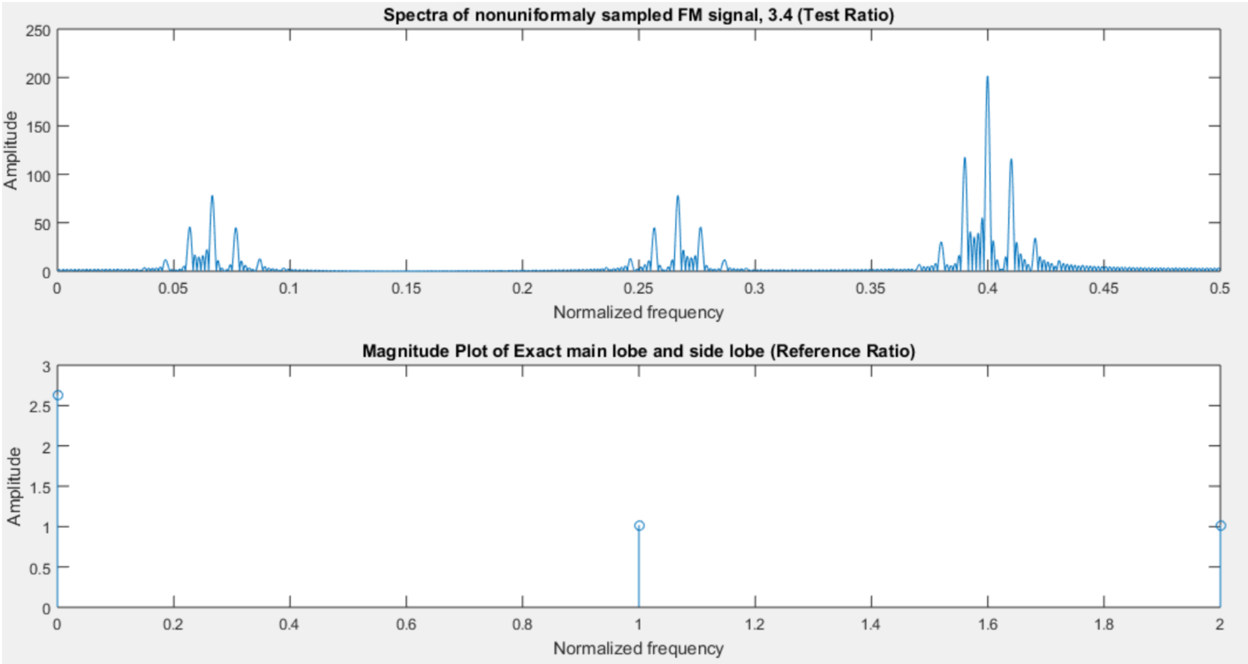


b. 3.4:

Output:

Error Percentage: -0.294803 and Test Ratio: 0.387961 and Reference Ratio: 0.386821

Error Percentage 0.294803



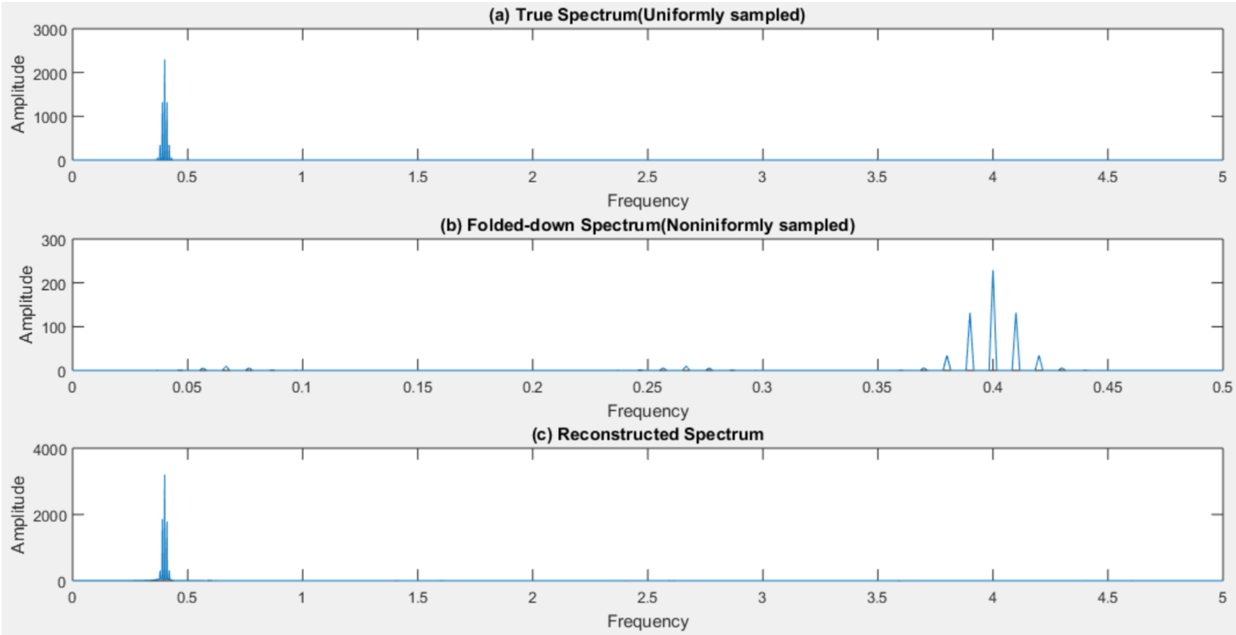
Code for the nonuniform under-sampling and reconstruction of a signal:

```
% Reconstruction from recurrent nonuniform sampling
clear
Order = 100;                  % Bandpass filter order
nseg = 200;                   % Number of samples of each ADC
M = 3;                        % Number of ADCs
N = M*nseg;                   % Total number of samples
NN = N*10;
freq = 3.4;
qfreq = 0.4;                  % folded-down frequency
nfreq = freq/10;              % normalized frequency for the bandpass filter
xorig = cos(2*pi*freq*(0.1)*(0:NN-1)+sin(2*pi*0.001*(0:NN-1)));
                              % Signal sampled at every 0.1 second
rm = [0 0.05 0.05];           % rm is the nonuniform sampling ratio
am =[];
for q = 1:nseg,
```

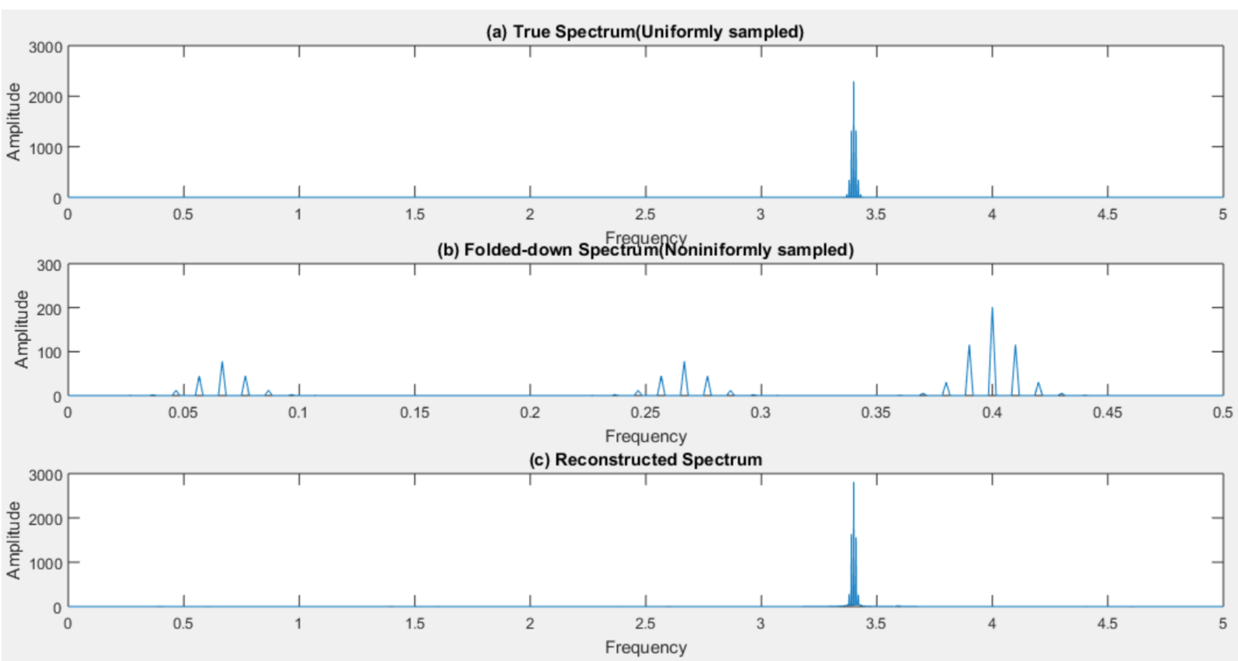
```
   am = [am,rm];
end
% Generate a nonuniformly sampled signal sampled at every 1 second
for n = 1:N,
   y(n) = cos(2*pi*(n-1+am(n))*freq+sin(2*pi*0.01*(n-1+am(n))));
end
h = fir1(Order, [2*qfreq-0.05, 2*qfreq+0.05], 'band');
                                  % Design of the bandpass filter to get rid of side lobes
yr = conv(h,y);                   % Filtering
yr = yr(Order/2+1:N+Order/2);                       % To get rid of transitional samples
yup = upsample(yr,10);                              % Upsample by 10 for interpolation
h = fir1(Order, [2*nfreq-0.005, 2*nfreq+0.005], 'band');    % Design of the banpass filter for
interpolation
yrecon = conv(h,yup);                               % Filtering
yrecon = 10*sqrt(2)*yrecon(Order/2+1:N*10+Order/2); % To get rid of transitioanl samples
XO = abs(fft(xorig));                               % DFT of the true spectrum
subplot(311), plot(0:1/N:5, XO(1:NN/2+1))           % True DFT
title('(a) True Spectrum(Uniformly sampled)')
Y = abs(fft(y));
subplot(312), plot(0:1/N:0.5, Y(1:N/2+1))           % Nonuniform DFT
title('(b) Folded-down Spectrum(Noniniformly sampled)')
YR = abs(fft(yrecon));                              % DFT of the reconstructed signal
subplot(313), plot(0:1/N:5, YR(1:NN/2+1))           % Reconstructed
title('(c) Reconstructed Spectrum')
```

Output:

Freq 0.4:


(a) True Spectrum(Uniformly sampled)
Amplitude
Frequency
(b) Folded-down Spectrum(Noniniformly sampled)
Amplitude
Frequency
(c) Reconstructed Spectrum
Amplitude
Frequency

Freq 3.4


(a) True Spectrum(Uniformly sampled)
Amplitude
Frequency
(b) Folded-down Spectrum(Noniniformly sampled)
Amplitude
Frequency
(c) Reconstructed Spectrum
Amplitude
Frequency

VITA

Name:	Teja Renukaradhya
Education Details:	Master in Science – Electrical Engineering (2016-2018)
	Texas A&M University – Kingsville, Kingsville, Texas, USA
	Bachelor of Engineering – Electronics & Communication (2010-2014)
	Vemana Institute of Technology – Bangalore, India
Professional Experience:	System Engineer (September 2014 – July 2016)
	IBM Pvt Ltd., Bangalore, India
	System Engineer (February 2017 – Present)
Research Interest:	Digital Signal Processing, Embedded Systems, Microprocessors
Contact:	tejur.12@gmail.com
	Teja.renukaradhya@students.tamuk.edu