

WestminsterResearch

<http://www.westminster.ac.uk/westminsterresearch>

Adaptive Interference Mitigation in GPS Receivers

Arif, Syed

This is a PhD thesis awarded by the University of Westminster.

© Mr Syed Arif, 2022.

<https://doi.org/10.34737/w0zvq>

The WestminsterResearch online digital archive at the University of Westminster aims to make the research output of the University available to a wider audience. Copyright and Moral Rights remain with the authors and/or copyright owners.

UNIVERSITY OF
WESTMINSTER 

Adaptive Interference Mitigation in
GPS Receivers



Syed Waqas Arif

**A thesis submitted in partial fulfilment of the requirements of the
University of Westminster for the degree of Doctor of Philosophy**

2022

I declare that the work presented in this thesis is my own, has not been submitted for any other award, is identical to the content of the electronic submission and that, to the best of my knowledge, it does not contain any material previously created by another person, except where due reference is made.

Syed Waqas Arif

ABSTRACT

Satellite navigation systems (GNSS) are among the most complex radio-navigation systems, providing positioning, navigation, and timing (PNT) information. A growing number of public sector and commercial applications rely on the GNSS PNT service to support business growth, technical development, and the day-to-day operation of technology and socioeconomic systems. As GNSS signals have inherent limitations, they are highly vulnerable to intentional and unintentional interference. GNSS signals have spectral power densities far below ambient thermal noise. Consequently, GNSS receivers must meet high standards of reliability and integrity to be used within a broad spectrum of applications. GNSS receivers must employ effective interference mitigation techniques to ensure robust, accurate, and reliable PNT service.

This research aims to evaluate the effectiveness of the Adaptive Notch Filter (ANF), a pre-correlation mitigation technique that can be used to excise Continuous Wave Interference (CWI), hop-frequency and chirp-type interferences from GPS L1 signals. To mitigate unwanted interference, state-of-the-art ANFs typically adjust a single parameter, the notch centre frequency, and zeros are constrained extremely close to unity. Because of this, the notch centre frequency converges slowly to the target frequency. During this slow converge period, interference leaks into the acquisition block, thus sabotaging the operation of the acquisition block. Furthermore, if the CWI continuously hops within the GPS L1 in-band region, the subsequent interference frequency is locked onto after a delay, which means constant interference occurs in the receiver throughout the delay period.

This research contributes to the field of interference mitigation at GNSS's receiver end using adaptive signal processing, predominately for GPS. This research can be divided into three stages.

I first designed, modelled and developed a Simulink-based GPS L1 signal simulator, providing a homogenous test signal for existing and proposed interference mitigation algorithms. Simulink-based GPS L1 signal simulator provided great flexibility to change various parameters to generate GPS L1 signal under different conditions, e.g. Doppler Shift,

code phase delay and amount of propagation degradation. Furthermore, I modelled three acquisition schemes for GPS signals and tested GPS L1 signals acquisition via coherent and non-coherent integration methods.

As a next step, I modelled different types of interference signals precisely and implemented and evaluated existing adaptive notch filters in MATLAB in terms of Carrier to Noise Density (C/N_0), Signal to Noise Ratio (SNR), Peak Degradation Metric, and Mean Square Error (MSE) at the output of the acquisition module in order to create benchmarks.

Finally, I designed, developed and implemented a novel algorithm that simultaneously adapts both coefficients in lattice-based ANF. Mathematically, I derived the full-gradient term for the notch's bandwidth parameter adaptation and developed a framework for simultaneously adapting both coefficients of a lattice-based adaptive notch filter. I evaluated the performance of existing and proposed interference mitigation techniques under different types of interference signals. Moreover, I critically analysed different internal signals within the ANF structure in order to develop a new threshold parameter that resets the notch bandwidth at the start of each subsequent interference frequency. As a result, I further reduce the complexity of the structural implementation of lattice-based ANF, allowing for efficient hardware realisation and lower computational costs.

It is concluded from extensive simulation results that the proposed fully adaptive lattice-based provides better interference mitigation performance and superior convergence properties to target frequency compared to traditional ANF algorithms. It is demonstrated that by employing the proposed algorithm, a receiver is able to operate with a higher dynamic range of JNR than is possible with existing methods.

This research also presents the design and MATLAB implementation of a parameterisable Complex Adaptive Notch Filer (CANF). Present analysis on higher order CANF for detecting and mitigating various types of interference for complex baseband GPS L1 signals. In the end, further research was conducted to suppress interference in the GPS L1 signal by exploiting autocorrelation properties and discarding some portion of the main lobe of the GPS L1 signal. It is shown that by removing 30% spectrum of the main lobe, either from left, right, or centre, the GPS L1 signal is still acquirable.

ACKNOWLEDGEMENTS

I want to thank everyone who contributed to my research and thesis. I owe my gratitude to my supervisor, Adem Coskun, who has supported, encouraged and guided me throughout the study. He was a constant source of encouragement and motivation, and his academic supervision was ideal. I am incredibly grateful to him, and his moral support throughout this research is unprecedented. Without his advice and knowledge sharing, this research would not reach its level.

I would also like to express my heartfelt gratitude to my second supervisor Professor Izzet Kale, for his expertise in technical discussions, constructive criticism and belief in my capabilities. His great sense of humour, charming character and whole-hearted laughs made our time more memorable and enjoyable. Thank you.

Lastly, my family, parents and wife supported me in every way they could have and especially my four little angels, Haris, Bakhtawar, Hamza and Adeen, who have sacrificed their special childhood moments.

LIST OF ACROYMS

ADC	Analog to Digital Converter
AGC	Automatic Gain Control
ANF	Adaptive Notch Filter
ARMA	Auto-Regressive Moving Average
ASIC	Application Specific Integrated Circuit
AWGN	Additive White Gaussian Noise
BOC	Binary Offset Carrier
BPSK	Binary Phase Shift Keying
BW	Bandwidth
C/A	Coarse Acquisition
CAF	Complex Adaptive Filter
CANF	Complex Adaptive Notch Filter
CDMA	Code Division Multiple Access
CRPA	Controlled Reception Pattern Antennas
C/N_0	Carrier to Noise Density
CWI	Continuous Wave Interference
DFT	Discrete Fourier Transform
DSP	Digital Signal Processing
DSP	Digital Signal Processing

DSSS	Direct Sequence Spread Spectrum
EGNOS	European Geostationary Navigation Overlay Service
FEC	Forwarded Error Correction
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
FPGA	Field Programmable Gate Array
GAGAN	GPS Aided GEO-Augmentation Navigation
GBAS	Ground Based Augmentation System
GEO	Geostationary Earth Orbit
GLONASS	Russian Global Orbiting Navigation Satellite System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IDM	Interference Detector and Monitoring
IF	Intermediate Frequency
IIR	Infinite Impulse Response
JNR	Jamming to Noise Ratio
J/N_0	Jamming to Noise Density
LEO	Low Earth Orbit
LMS	Least Mean Square
LNA	Low Noise Amplifier
MSAS	Michibiki Satellite Augmentation System
MSB	Most Significant Bit
MSE	Mean Square Error

OS	Open Service
P(Y)	Precision Code
PDM	Peak Degradation Metric
PNT	Position Navigation Time
PRN	Pseudo Random Number
PSD	Power Spectral Density
RF	Radio Frequency
RFI	Radio Frequency Interference
RLS	Recursive Least Square
RS	Restricted Service
SBAS	Satellite Based Augmentation System
SINR	Signal-to-Interference-Plus-Noise
SIS	Satellite in Space
SNR	Signal to Noise Ratio
SoL	Safety of Life
STFT	Short Time Fourier Response
STL	Satellite Time Location
TF	Time Frequency
TFD	Time-Frequency Domain
TRF	Time-Frequency Representations
VCO	Voltage Controlled Oscillator
WAAS	Wide Area Augmentation System
ADC	Analog to Digital Converter
AGC	Automatic Gain Control

ANF	Adaptive Notch Filter
ARMA	Auto-Regressive Moving Average
ASIC	Application Specific Integrated Circuit
AWGN	Additive White Gaussian Noise
BOC	Binary Offset Carrier
BPSK	Binary Phase Shift Keying
BW	Bandwidth
C/A	Coarse Acquisition
CAF	Complex Adaptive Filter
CANF	Complex Adaptive Notch Filter
CDMA	Code Division Multiple Access
CRPA	Controlled Reception Pattern Antennas
C/N_0	Carrier to Noise Density
CWI	Continuous Wave Interference
DFT	Discrete Fourier Transform
DSP	Digital Signal Processing
DSP	Digital Signal Processing
DSSS	Direct Sequence Spread Spectrum
EGNOS	European Geostationary Navigation Overlay Service
FEC	Forwarded Error Correction
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
FPGA	Field Programmable Gate Array

GAGAN	GPS Aided GEO-Augmentation Navigation
GBAS	Ground Based Augmentation System
GEO	Geostationary Earth Orbit
GLONASS	Russian Global Orbiting Navigation Satellite System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IDM	Interference Detector and Monitoring
IF	Intermediate Frequency
IIR	Infinite Impulse Response
JNR	Jamming to Noise Ratio
J/N_0	Jamming to Noise Density
LEO	Low Earth Orbit
LMS	Least Mean Square
LNA	Low Noise Amplifier
MSAS	Michibiki Satellite Augmentation System
MSB	Most Significant Bit
MSE	Mean Square Error
OS	Open Service
P(Y)	Precision Code
PDM	Peak Degradation Metric
PNT	Position Navigation Time
PRN	Pseudo Random Number
PSD	Power Spectral Density
RF	Radio Frequency

RFI	Radio Frequency Interference
RLS	Recursive Least Square
RS	Restricted Service
SBAS	Satellite Based Augmentation System
SINR	Signal-to-Interference-Plus-Noise
SIS	Satellite in Space
SNR	Signal to Noise Ratio
SoL	Safety of Life
STFT	Short Time Fourier Response
STL	Satellite Time Location
TF	Time Frequency
TFD	Time-Frequency Domain
TRF	Time-Frequency Representations
VCO	Voltage Controlled Oscillator
WAAS	Wide Area Augmentation System

LIST OF FIGURES

Figure 1.1 Block diagram of generating GPS L1 and L2 signal [24]	18
Figure 1.2 Illustrations of the GPS L1 spectrum and thermal noise level	20
Figure 1.3 (a) Auto-Correlation and (b) Cross-Correlation Peak	21
Figure 1.4 An Example of BPSK Modulation.	23
Figure 1.5 An Example of BOC modulation	24
Figure 1.6 Spectrum of C/A code, P(Y) and BOC (1,1)	25
Figure 2.1 Simulink Model of GPS L1 Signal.....	30
Figure 2.2 C/A code Generator	31
Figure 2.3 Simulink Model Generating Sampled C/A code	32
Figure 2.4 Output of C/A counter (b) Zoomed version of (a)The look-up table is basically.....	33
Figure 2.5 (a) Output signal generated from the C/A Look-Up Table (b) Zoomed version of (a)	34
Figure 2.6 Simulink Model generating the Navigation Data	35
Figure 2.7 Flow Chart of Indexing Navigation Data from Look-up Table.....	36
Figure 2.8 (a) 20 complete cycles of C/A counter, each triangle lasts 1ms (b) Represent single cycle of navigation period counter, equivalent to 20ms, staircase increment from 0 to 20, then reset back to zero after 20 counts (c) The output of the navigation count.....	37
Figure 2.9 Simulink Modelling of P(Y) code	38
Figure 2.10 (a) Spectrum of C/A code, (b) Spectrum of P(Y) code	39
Figure 2.11 Output of GPS L1 Simulator at the carrier frequency of 9.55MHz	40
Figure 3.1 Block Diagram for Serial Search Acquisition Algorithms	44
Figure 3.2 Serial Search Acquisition of PRN 15	45
Figure 3.3 Block Diagram for Parallel Frequency Search Acquisition Algorithms	46
Figure 3.4 Block Diagram for Parallel Frequency Code Phase Search Acquisition Algorithms	47
Figure 3.5 Parallel Frequency Code Phase Search of PRN 17 with the Noise level of 30dB	48
Figure 3.6 Parallel Frequency Code Phase Search of PRN 17 with a Noise level of 45dB	49

Figure 3.7 Search Grid for GPS signal acquisition	50
Figure 3.8 Acquisition scheme with a non-coherent detector.....	53
Figure 3.9 Evolution of correlation peak for weak signal acquisition for GPS L1 for 1ms to 12ms.....	54
Figure 3.10 Evolution of Code Phase delay as the number of non-coherent integration time increases from 1ms to 12ms.....	55
Figure 3.11 Equivalent Input SNR when front end BW=4.76MHz.....	59
Figure 3.12 Input SNR Vs. Output SNR.....	60
Figure 3.13 C/No input Vs. Output SNR.....	61
Figure 4.1 Modelling of continuous wave interference	66
Figure 4.2 Complex Sinusoidal and its projections	67
Figure 4.3 Time-Frequency representation of a linear saw-tooth chirp signal	68
Figure 4.4 Complex chirp signal time-frequency representation of instantaneous frequency as a function of time.....	69
Figure 4.5 Real chirp signal time-frequency representation of instantaneous frequency as a function of time	69
Figure 4.6 Illustration of Normalized PSD of CWI and C/A code	71
Figure 4.7 Response of Notch Filter different values k_{α}	78
Figure 4.8 Complex sinusoidal signal and its removal by the complex notch filter.....	78
Figure 4.9 Distortion of CAF after notch filtering.....	79
Figure 4.10 CAF distortion by second-order filter.....	80
Figure 4.11 System block representation of the calculation of output SNR.....	83
Figure 4.12 Output SNR in dB vs JNR for 1st Order Complex filter with $k_{\alpha}=0.88$ and C/No 39dB	84
Figure 4.13 MSE in dB at the output of 1 st order CNF Vs. JNR	84
Figure 4.14 (a) ρ_{max} for acquired satellite after mitigation (b) None of the satellites is acquired as $\rho_{max}<2$ due presence of interference signal with JNR of 20dB	86
Figure 4.15 (a) Different 3dB bandwidth of three complex notch filter. (b) All three filter with the same 3dB bandwidth.....	89
Figure 4.16 Output SNR Vs. Order of Notch filter with the same z_0^* value for all the notch filter	91
Figure 4.17 Proposed System-Level Model.....	91
Figure 4.18 (a) Order of Notch filter and corresponding frequency response and notch depth but width different 3dB bandwidth. (b) Zoomed in version of (a)....	92

Figure 4.19 Filters Performance with equal 3dB bandwidth	93
Figure 4.20 Effect of coefficients quantization on Output SNR for 1st Order CNF.(b) Effect of coefficients quantization on output SNR 2nd CNF	94
Figure 4.21 Output SNR for complex notch filter up to order 6 with input GPS L1 signal with C/No of 47dB.....	96
Figure 4.22 ARMA Structure of Complex Adaptive Notch Filter.....	98
Figure 4.23 (a) acquisition of GPS L1 signal after tracking and mitigating complex sinusoidal and With JNR of 14dB. (b) STFT of the signal with interference and after mitigation. (c) Convergence of z_0 to unity	102
Figure 4.24 Schematic of GPS signal acquisition after interference mitigation	102
Figure 4.25 Top shows the time-Frequency evolution of the GPS L1 signal with interference. The red dotted line represents the notch's centre frequency, and the bottom plot shows the filtered-out data by the notch filter	104
Figure 4.26 (a) Variation of the magnitude of z_0 over the time. (b) Zoomed version of (a)	105
Figure 4.27 Output of acquisition block after mitigating complex-chirp signal with multiple saw-tooth functions.....	105
Figure 4.28 (a) Zoomed-in version of the evolution of notch centre frequency (red dash line). (b) variation of the magnitude of z_0 between $100\mu\text{s}$ to $150\mu\text{s}$.(c) Evolution of Complex Zero of the CANF filter in between $100\mu\text{s}$ to $150\mu\text{s}$ on z-plane	106
Figure 4.29 Frequency response of second-order adaptive IIR notch filter after it reaches steady state conditions.....	108
Figure 4.30 Convergence of z_0 close to unity.....	109
Figure 4.31 Time-Frequency representation after mitigation of the interference via notch filtering.....	110
Figure 4.32 Output of acquisition block after CWI inference mitigation	110
Figure 4.33 (a) Top plot shows a Time-Frequency illustration of frequency hopping of CWI interference, and the red dot line represents the centre frequency of the notch filter. (b) the Bottom plot filtered out data from the notch filter	111
Figure 4.34 Convergence of magnitude of z_0 for hopping CWI	112
Figure 4.35 Evolution of notch's centre frequency while tracking 1 MHz, 1.25MHz, and 0.75 MHz frequency hopped CWI interference.....	113
Figure 4.36 Convergence of the absolute value of z_0 in the second-order notch filter for value of k_α 0.8 and 0.9.....	114
Figure 4.37 Convergence of magnitude of z_0 for different level of JNR	115
Figure 4.38 (a) and (b) zoomed version of Figure 4.37 for detailed analysis	116

Figure 4.39 Results for setting up the threshold to activate the required filter depending on the variance of the magnitude of z_0	118
Figure 4.40 Proposed System-Level Model.....	119
Figure 4.41 Setting up the threshold to activate the required CANF filter depending on the variance of the magnitude of z_0	119
Figure 5.1 2nd Order Direct Form IIR Notch Filter and its transfer function	125
Figure 5.2 Fully Adaptive Lattice-Based Notch Filter and its transfer function	126
Figure 5.3 The adaptation of ρ via [81] in all-pass lattice notch filter.....	131
Figure 5.4 Tracking performance of all-pass based Notch filter in [81], with a noise variance of 0.25.....	132
Figure 5.5 Tracking performance of all-pass based Notch filter in [81], with a noise variance of 0.04.....	133
Figure 5.6 The adaptation of ρ with a variance of 0.04	134
Figure 5.7 Sign Magnitude LMS for tracking the hop frequency signal [68].....	135
Figure 5.8 Tracking and Convergence performance of Direct Form IIR Notch filter [68]	136
Figure 5.9 (a) 3D Frequency Response of Direct Form IIR Notch Filter for when values of ρ sweep between 0 and 1. (b) Front view of 3D frequency response .	138
Figure 5.10 (a) 3D Frequency response for the value of ρ between 0 to 0.6. (b) The front view of 3D Frequency response in (a)	139
Figure 5.11 (a) More gain scaling if target frequency is closer to Nyquist. (b) Front view of Figure 5.11 (a).....	139
Figure 5.12 3D frequency response when ρ is a sweep from -1 to 1, as ρ approaches -1, a pole is introduced as shown in the figure.....	140
Figure 5.13 (a) 3D Frequency Response of Lattice-based Notch Filter when the values of ρ sweeps from between -1 and 1.(b) top view of 3D frequency response of Figure 5.13 (a).....	141
Figure 5.14 (a) 3D Frequency Response of Lattice-based Notch Filter when target frequency close to Nyquist and ρ sweeps from 0 to 1. (b) Front view of the 3D frequency response of Figure 5.14 (a).....	142
Figure 5.15 Cost Function Plot of Direct Form Transfer Function.....	144
Figure 5.16 Cost Function Plot of All-pass Lattice Transfer Function.....	144
Figure 5.17 Illustration of the plots in Figures 5.15 and 5.16, superimposed on each other.....	145
Figure 5.18 Illustration of the effect of ρ on the deepness of the Cost Function. The deepness of the cost function increases as the ρ approaches zero	146
Figure 5.19 Implementation of full gradient term for updating term ρ	149

Figure 5.20 Full gradient-based adaptation of ρ in Lattice based ANF.....	150
Figure 5.21 The tracking performance of Lattice-based ANF for the hop frequency signal	150
Figure 5.22 Adaptation of parameter ρ via hard reset and wait method.....	153
Figure 5.23 Tracking performance of Lattice-based ANF with hard reset and wait method.....	153
Figure 5.24 Adaptation of ρ with minimum constraints method.....	155
Figure 5.25 (a) The magnitude of the gradient energy shown in red and blue represents the corresponding output of the filter (b). Zoomed version of (a) to illustrate the fact that at one point in time gradient is less than the output of the filter.....	157
Figure 5.26 Corresponding resetting the value of ρ via Method I.....	158
Figure 5.27 Absolute value of the instantaneous gradient of ϕ_β and ϕ_ρ	159
Figure 5.28 Instantaneous Gradient of ϕ_ρ	159
Figure 5.29 (a) Corresponding resetting the value of ρ via Method II, but unable to reset ρ after 2 nd hop frequency. (b) In the second run of the simulation, and this time unable to reset ρ after first hop frequency signal	160
Figure 5.30 Plot of $\psi[n]$ at the output of the ANF	162
Figure 5.31 Corresponding resetting the value of ρ via Method III	163
Figure 5.32 Tracking Performance of Lattice-based ANF when only β is adapted [49]. Blue colour line show tracking of ANF when $\rho=0.95$. Black line show tracking of ANF when ($\rho=0.9$)	165
Figure 5.33 Tracking performance of ANF via Proposed Method. Simultaneously adapting both parameters ρ and β	167
Figure 5.34 Corresponding adaptation of parameter ρ for above diagram	167
Figure 5.35 Frequency tracking comparison between [48] and the Proposed Method.....	168
Figure 5.36 Number of bits required to acquire signal with different levels of JNR.....	169
Figure 5.37 Complete System Modelling for testing of the proposed algorithm	170
Figure 5.38 Evaluation of CAF at the output of acquisition with CWI hop frequency interference.....	172
Figure 5.39 PSD of useful signal with the interference of J/N_0 100dB-Hz.(b) PSD of Interference excision via the proposed method. (c) Interference excision via method published in [51] and [67]	173
Figure 5.40 C/N0 At output of the acquisition module. Red curve represents the Direct form IIR. The black curve represents lattice-base ANF with the proposed algorithm	174
Figure 5.41 Three gradient signal under test (a) represents g_{p3} (b) g_1 (c) g_{p2}	176

Figure 5.42 Frequency Response of $G(z)$	177
Figure 5.43 Frequency response of the gradient signal g_{p3} , g_{p2} and g_1	177
Figure 5.44 The proposed modified gradient signal g_1 for the adaptation of notch bandwidth parameter ρ	179
Figure 5.45 Frequency response of g_p in red and g_1 yellow	179
Figure 5.46 Comparison of tracking of hop frequency CWI with ρ fixed at 0.98 for both the proposed method and [81].....	183
Figure 5.47 Comparison of tracking of hop frequency CWI with ρ fixed at 0.50 for both proposed methods and [81]	184
Figure 5.48 Comparison of tracking of hop frequency CWI with ρ fixed at 0.80 for both proposed methods and [81]	185
Figure 5.49 Comparison of tracking of hop frequency CWI with ρ fixed at 0.995 for both proposed methods and [81]	186
Figure 5.50 Adaptation of ρ with a partial gradient [81], μ_ρ is fixed at 0.003 and utilizing the proposed constraints algorithm	189
Figure 5.51 Adaptation of ρ with partial gradient [81], μ_ρ is fixed at 0.006 and utilizing proposed constraints algorithm	190
Figure 5.52 Adaptation of ρ with partial gradient [81], μ_ρ is fixed at 0.006 & 0.009 and utilizing proposed constraints algorithm.....	190
Figure 5.53 Magnitude response of both ANF in steady state at a normalized frequency of 0.15	192
Figure 5.54 All four notch at target frequencies of 0.103, 0.15, 0.307 and 0.413. (a) Magnitude response of ANF in reference [81]. (b) Magnitude response of the ANF via the proposed method	193
Figure 5.55 C/N0 At the output of the acquisition module. Red curve represents the Direct form IIR. The black curve represents lattice base ANF with the proposed algorithm, and the blue curve represents results attained by reference [81] via constrained ρ algorithm	194
Figure 5.56 Fixed-Pointed Binary Representation.....	197
Figure 5.57 Performance comparison of floating-point and fixed-point	198
Figure 5.58 Illustration of quantization of both coefficients in the proposed filter	200
Figure 5.59 MSE at the output of the filter with different numbers of fractional bits	201
Figure 5.60 MSE of quantization of the coefficient $\beta[n]$ of the ANF filter with different numbers of fractional bits.....	202
Figure 5.61 MSE of quantization of coefficient $\rho[n]$ of the ANF filter with different numbers of fractional bits.....	202
Figure 5.62 Workflow diagram for Practical Implementation via MATLAB/Xilinx platform.....	204

Figure 5.63 Example experiment setup.....	206
Figure 5.64 A typical GNSS receiver architecture.....	207
Figure 6.1 Illustration of the theoretical spectrum of GPS L1 C/A at baseband	215
Figure 6.2 Illustration of the analysis methodology for the acquisition of GPS L1 with a certain percentage of GPS L1 main-lobe discarded.....	217
Figure 6.3 (a) Peak Degradation Metric (PDM) vs Band-stop region. (b) PDM in dB	220
Figure 6.4 SNR at the output of the acquisition module after band-stop filtering with variable width.....	221
Figure 6.5 (a) Peak Degradation Matric (PDM) vs Band-Pass region (b) PDM in dB.....	222
Figure 6.6 SNR at the output of the acquisition module after band-pass filtering with variable width.....	224
Figure 6.7 (a) PSD of C/A code and noise. (b) Combine PSD of C/A code added with noise exact noise level.....	225
Figure 6.8 (a) PDM vs Stop-band region. (b) PDM in dB.....	226
Figure 6.9 SNR at the output of the acquisition module.....	226
Figure 6.10 Illustration of how multi-tone interference signal suppression via band-stop filter	228
Figure 6.11 Shows the peak degradation vs Band-stop region for JNR level 40dB.....	228
Figure 6.12 SNR at the out of the Acquisition module for JNR level of 40dB after the suppression of closely packed multi-tone interference	229
Figure 6.13 Four 3D acquisition plot at 15%, 25% 35% and 45% band-stop region	230
Figure 6.14 Acquisition Peak Ratio for each CAF ranging from 0% to 100%	231
Figure 6.15 Acquisition of GPS L1 via Parallel Frequency Code Phase Search Method .	233
Figure 7.1 Illustration of the proposed method.....	241
Figure 7.2 Frequency Domain interference mitigation method.....	242

LIST OF TABLES

Table 2.1 Global Parameters for GPS L1 Simulator.....	29
Table 2.2 Simulations Parameters.....	39
Table 3.1 Comparison of two detector methods	52
Table 3.2 Simulation Parameters.	54
Table 3.3 Simulation Parameters	58
Table 4.1 Transfer Function of First, Second and Third-Order Notch filter.....	87
Table 4.2 Simulation Set-Up Parameters	89
Table 4.3 Value of $k\alpha$ for corresponding Order of notch filter to attain the same 3dB notch bandwidth for all three filter.....	92
Table 4.4 Simulation Parameters for 1st Order Complex Adaptive Notch Filter.....	101
Table 5.1 Simulation parameters for [81]	131
Table 5.2 Number of samples required to lock on the subsequent target frequency	137
Table 5.3 Simulation parameters used to simulate Figures 5.20 and 5.21.....	151
Table 5.4 Internal Signal used in developing automatic resetting mechanism for ρ	155
Table 5.5 Comparison of the existing method and proposed method.....	181
Table 5.6 Comparison of two filters in terms of transfer functions, gradient signals (for ρ and β) and update equations.....	182
Table 5.7 Simulation parameters for reference [81]	187
Table 5.8 Navigation frequencies processed by typical multi-GNSS receivers	208
Table 6.1 Simulation set parameters	219

Table of Contents

Abstract	i
Acknowledgements	iii
List of Acroyms	iv
List of Figures	x
List of Tables	xvii
Chapter 1 INTRODUCTION	1
1.1 Background and Motivation	2
1.2 Literature Review – <i>mitigation technologies</i>	3
1.2.1 GNSS Augmentation – <i>mitigation via the service provider</i>	3
1.2.2 Interference Mitigation- <i>at receiver-end</i>	6
1.3 Research Aims and Objectives	11
1.4 Thesis Contribution	13
1.5 Characteristics of GPS Signals	16
1.5.1 GPS Signal Structure.....	16
1.6 Spread Spectrum, Ranging Code and CDMA	20
1.7 Modulation Scheme	22
1.8 Outline of the Thesis	25
1.9 Conclusions	27
Chapter 2 Simulink Modelling of GPS L1 Signal	28
2.1 Gold Code and PRN Code Generation	30
2.1.1 C/A Code Generation	32
2.1.2 Navigation Data Generation.....	35
2.1.3 P(Y) Code Generation.....	37
2.2 Simulation Results	38
2.3 Conclusions	40
Chapter 3 Acquisition of GPS L1 Signal	41
3.1 GNSS Signal Acquisition	42
3.1.1 Serial Search Acquisition.....	43
3.1.2 Parallel Frequency Search.....	45
3.1.3 Parallel Code Phase Search.....	46
3.2 FFT-Based Acquisition Algorithm.....	49
3.3 Acquisition of Weak GPS signal	51

3.4 Performance Evaluation Metrics	55
3.5 Modelling of the signals within the GNSS receiver	56
3.5.1 SNR and C/N0	56
3.5.2 Noise power density	57
3.5.3 GPS L1 signal at Receiver Input	57
3.6 Simulation results	58
3.7 Conclusions	61
Chapter 4 Continuous Wave Interference and Complex Adaptive Notch Filters	63
4.1 Continuous Wave Interference	64
4.2 CWI Modelling in MATLAB	65
4.2.1 Continuous Wave Narrowband Interference Real and Complex Sinusoidal	66
4.2.2 Real and Complex Chirp-type Interference	68
4.2.3 Jamming to Noise Ratio (JNR)	70
4.3 Detection and Mitigation	72
4.4 Digital Notch Filters	73
4.4.1 Why IIR complex adaptive notch filters?	75
4.4.2 First-order Notch Filter- <i>A Brief Demo</i>	76
4.5 Performance Analysis Metrics	81
4.6 Complex Notch Filters: Higher Order	87
4.7 Complex Adaptive Notch filter	96
4.7.1 First-order Complex Adaptive Notch Filter	97
4.7.2 Adaptation Algorithm	98
4.7.3 Mitigation of Complex Chirp-Type Interference	103
4.8 Second-order Real IIR Notch Filter	107
4.9 Power of the Adaptation Parameter z_0	114
4.10 Proposed System Model and Algorithm	119
4.11 Conclusions	122
Chapter 5 Adaptive IIR Notch Filters for Interference Mitigation to Improve Signal Acquisition Performance	124
5.1 Introduction	124
5.2 Adaptation of the Notch Bandwidth Parameter	127
5.3 Adaptation of Notch Bandwidth Parameter via partial gradient	129
5.3.1 Partial Gradient Term for Adapting Notch Bandwidth Parameter	129
5.3.2 3D Frequency Response Analysis and the Cost Function	137
5.3.3 2 nd Order Direct Form IIR Notch Filter	138

5.3.4 All-Pass Lattice-based ANF.....	140
5.3.5 Cost Function	143
5.4 Proposed Full Gradient Term for Notch Bandwidth Parameter $\rho[n]$	147
5.4.1 Fully Constraint Adaptation Algorithm of $\rho[n]$	155
5.4.2 Method I Reset Condition: Filter Output and Instantaneous Power (Φ_β).....	156
5.4.3 Method II Reset Condition: Instantaneous Gradient of ϕ_ρ	158
5.4.4 Method III Proposed Method: Instantaneous output power and $V_{\text{threshold}}$	161
5.5 Simulation Results.....	165
5.6 Complete Modelling of the System	168
5.7 Hardware Complexity Reduction of the Proposed Structure	175
5.8 Fixed-Point Simulation of the Proposed Adaptive Notch Filter.....	195
5.8.1 Floating-Point and Fixed-Point Comparison	197
5.8.2 Quantization of the Proposed Filter's Coefficients	199
5.8.3 Implementation, Testing and Transferability.....	203
5.9 Conclusions	209
Chapter 6 Exploiting THE Auto-correlation FUNCTION and THE Spectrum envelope of THE GPS L1 Signal	213
6.1 Acquisition of GPS L1 Signal When Certain Percentage of Main Lobe Discarded.....	217
6.2 CASE I: Filter 'Clean C/A code' through variable width ideal band-stop filter..	219
6.3 CASE II: Filtering 'Clean C/A code' through variable width ideal band-pass filter.....	222
6.4 CASE III: Filtering 'C/A code +Noise' through variable width ideal band-stop filter 224	
6.5 CASE IV: Filtering 'C/A code +Noise + multi-tone interference' through variable width ideal band-stop filter	227
6.6 Concluding Remarks:	231
Chapter 7 Conclusion and Future Work	234
References	244

Chapter 1

INTRODUCTION

The Global Navigation Satellite System (GNSS) constitutes the satellite constellation capable of sending positioning, navigation, and time (PNT) information from space. Different GNSS systems are operating in the world, such as GPS (United States), GALILEO (Europe), GLONASS (Russia) and BEIDOU (China), each having a different number of orbits and satellites. Satellite-based positioning is increasingly becoming a vital part of our lives. Positioning technologies play an important role in supporting various applications with different accuracy requirements. Regardless of current technological advancement, it is challenging for researchers and engineers to acquire positioning signals in the presence of intentional or unintentional interference. The research undertaken in this PhD addresses this challenge. This chapter presents the context of this thesis through a brief background followed by the aim and objectives, highlights the main contributions of this research work, and concludes with a literature review and nomenclature of the GPS signals.

1.1 Background and Motivation

Over the last five decades, location-based services have grown exponentially. It has penetrated everything that moves, travels, or requires timestamping. GNSS-based services are used in surface transportation, marine, mining, tracking of animals and vehicles, police and rescue operations, timestamping transactions, surveying and agriculture, national power grids, air and space applications, and so on. Positioning and navigation-based applications are vastly incorporated in all these sectors for deliverance, optimisation and consumer satisfaction, making it a multi-billion dollar industry. Hence, the reliability and integrity of GNSS signals hold astronomical importance for private or government entities in terms of safety and security for their users.

However, GNSS signals are highly vulnerable to even minute level of in-band interference due fact that by the time it reaches the surface of the Earth; their power is in the range of 10^{-16} watts (-160 dBW), which is extremely weak. GNSS signals are submerged below noise levels by approximately 20dB. Any kind of in-band interference adversely affects the quantisation, acquisition and tracking loop process inside the receiver and largely contributes to the ranging error. These calculations in the receiver become unreliable and inaccurate, thus providing false position, velocity, and time, posing a security and safety risk for the end users.

Modernising GNSS signals by increasing signal power and the number of frequencies and embedding BOC modulation has reduced the effect of interference signals, but it cannot completely eliminate them.

Vulnerabilities faced by GNSS can be divided into three categories. (1) System level related to satellite and receiver errors (2) Propagation medium related to atmospheric and multipath errors. (3) Interference-related that can be deliberated interference such as jammer, spoofing

and meaconing or unintentional interference caused by electronic devices or commercial high power transmitters that emit electromagnetic harmonic in the GNSS frequency band.

Among the different issues faced by GNSS, interference related is particularly challenging for three main reasons. Firstly, it is unannounced, making it difficult to predict the time of occurrence. Secondly, a single piece of hardware cannot eliminate various types of interference signals. Lastly, it is regarded as a two-stage process with a combination of interference detection and classification.

1.2 Literature Review – *mitigation technologies*

Safeguarding GNSS-dependent infrastructure from intentional or unintentional interference is not only the conquest for GNSS researchers or engineers, but it has become the government's policy around the globe. In UK and USA, governments are heavily investing in R&D to ensure the reliability, integrity and accuracy of PNT, hence increasing the resilience of the GNSS receivers. The interference mitigation technologies fall into two categories, non-signal-based design, which exploits existing communication infrastructures such as cellular and wifi networks, eLoran, Low Earth Orbit (LEO) satellites, dedicated terrestrial transmitters and other GNSS augmentation systems. The second category is signal-based design, such as different GNSS signal coding/authentication schemes, adaptive filtering, beamforming antennas, pulse blanking, and time-frequency excision. The following section discusses both categories in detail.

1.2.1 GNSS Augmentation – *mitigation via the service provider*

The principle behind the augmented-GNSS system is that external assistance is provided to make PNT solutions more reliable and accurate. Several countries have developed their

Satellite-Based Augmentation System (SBAS) or Ground-Based Augmentation Systems (GBAS) [1]. For example, in USA Wide Area Augmentation System (WAAS), India has GPS-aided GEO-Augmented Navigation (GAGAN), the Japanese have developed Michibiki Satellite Augmentation System (MSAS), and European use European Geostationary Navigation Overlay Service (EGNOS). All these SBAS systems use a vast network of ground stations that monitors GNSS signals. These ground stations further process PNT signals and apply corrections to propagation delays. Then up-linked corrected data to geostationary satellites again. Hence, it enhances the integrity and accuracy of GNSS-based PNT solutions. SBAS systems currently provide meter-level accuracy. However, all SBAS satellites are situated in MEO and GEO orbits, which means at least 20,000 km above the Earth's surface and by the time signal reaches Earth, the signal is highly attenuated and considered very weak. However, SBAS systems significantly improve GNSS signal propagation vulnerabilities such as satellite clock errors, ephemeris, and ionospheric errors. But errors induced by multipath or interference are not common to the ground station and the user; therefore, they cannot be mitigated by any augmentation systems.

A new concept has emerged in the past few years known as LEO-PNT [2]. With Low Earth Orbit (LEO) satellites, users are provided two positioning methods: dedicated PNT signal and Signal of Opportunity (SOP). Dedicated PNT signals contain navigation messages, whereas SOP does not. LEO satellite constellation such as Iridium hosts Satellite Time + Location (STL), a PNT satellite signal provided by Orolia with global coverage. It transmits an encrypted signal to all users. LEO-PNT operate closer to Earth. Signal strength is a thousand times stronger than the GNSS signal. This means that signals can be received indoors, with a very high jamming tolerance of 40dB, and are impossible to spoof. The positioning accuracy of the STL system is 30m-50m which is low as per current standards.

Two significant factors limit STL's accuracy: geometric dilution due to not enough satellites visible simultaneously and limited signal bandwidth [2].

Assisted-GNSS (A-GNSS) is a method of augmentation for GNSS satellites to be used in conjunction with cellular towers, and it reduces the time to fix position. Cellular towers continuously receive signals from GNSS, and this information is processed by powerful servers. This information is then transmitted to the GNSS receiver via the mobile network. The receiver can use this information to improve positioning accuracy in weak satellite signals. Since the satellite ephemeris data is contained in the assistance data, the receiver does not need to decode the satellite ephemeris data, which can take up to 30 seconds using a GPS L1 C/A signal.

By 2020, only Galileo provide open service with authentication [17]. It is process in which GNSS signal are incorporated with specific features that cannot be replicated or reproduce by malicious user. Authentication can be done either at data level by authenticating navigation message or range level by to authenticating the measured ranges to the satellites. Authentication can protect against spoofing, but not against jamming.

Different coding schemes and modernisation of the GNSS signal have been implemented to make GNSS receivers more interference resilient. Galileo, Modernized GPS, and SBAS signals employ Forwarded Error Correction (FEC) [4]. As navigation systems are one-way systems, FEC techniques encrypt data using an error-correcting code before transmission. In order to recover the original data, the receiver makes use of the additional information provided by the code, also called redundancy. The GPS L2C signal has 2.3 dB less power than the GPS L1 C/A signal; however, it compensates for this loss of power through its modulation. Adding robust coding schemes modernising GNSS signals by increasing signal

power and the number of frequencies and embedding BOC modulation has reduced the effect of interference signals, but it cannot eliminate them when interference power is high.

1.2.2 Interference Mitigation- *at receiver-end*

The GNSS interference can be classified into Narrow Band Interference (NBI) and Wide Band Interference (WBI). NBI affects only a small portion of the GNSS frequency bands, whereas WBI affects almost the entire spectrum. Interference signals can be either intentional or unintentional. There are numerous GNSS jammers available on the Internet that can be used intentionally to create interference. All GNSS jammer operates similarly, emitting a high-power frequency in the band of interest to submerge satellite navigation signal in its vicinity. It depends on the output power of the jammer signal and the area of effect. GNSS jammers can emit a wide range of interference signals such as Continuous Wave Interference (CWI), swept-frequency interference (Linear Chirp Frequency), Quadratic and Cubic Chirp-Interference and CWI hop-frequency interference.

As far as interference is concerned, all satellite-based navigation systems share the same weaknesses. The carrier frequency of GPS is at approximately 1.575 GHz, which has a wavelength of around 19 cm. The high frequency was chosen to enable signals to propagate through the ionosphere. Unlike high-frequency signals, low-frequency signals (below approximately 20 MHz) are attenuated more by the ionosphere so that GPS signals can pass through relatively unaffected. When a high-frequency signal is used, it means that the attenuation between the transmitter and receiver is increased; that is, the signals are strong when they leave the satellite, but after travelling 20,000 km (for GPS), the signal level can be as low as -158.5dBW once they reach the receiver.

Nevertheless, it is possible to use smaller antennas to balance the increased attenuation. This means that receivers can be made smaller, and transmitters such as jammers can be made smaller. Because of their low power and small antenna size, even small battery-powered jammers can easily malfunction nearby receivers. Additionally, they are inexpensive and easy to conceal.

Interference mitigation techniques at the receiver end can be divided into four domains. (1) Time-Domain, using adaptive digital signal processing. (2) Frequency-Domain, employing FFT to analyse the GNSS spectrum and removing interference. (3) Time-Frequency Domain (TFD), such as the spectrogram and the Wigner-Ville distribution. (4) RF antenna based such as Controlled Reception Pattern Antennas (CRPA), beamforming antennas, Spatial-Time and Spatial-Frequency Adaptive Processing (STAP & SFAP).

In the literature, interference detection and mitigation based on spectral analyses using Fast Fourier Transforms (FFT) have been extensively explored [5] [6]. FFT-based interference mitigation techniques aim to differentiate between the harmonics that belong to the interference signal from the GNSS signal and suppress interference in the frequency domain. FFT-based required methods require high resolution to characterise interference signals, which means more FFT points are needed, resulting in high computational demand and power consumption.

Both [7] [8] use Time-Frequency Domain (TFD) method based on Short-Time Fourier Transform (STFT) to excise interference signal. TFD allows detection and eliminates a wide range of disturbing signals with great efficiency. TFD converts a one-dimensional time signal into a two-dimensional function of time and frequency. As a result of the characterisation of the signal on the time-frequency plane can provide a more detailed picture of the spatial location of the signal's spectral components. The method in [8] uses peaks-

interpolation for estimating the instantaneous interference frequency. Using this algorithm, the TFD samples that pass a fixed threshold are selected, and a polynomial curve that better interpolates these TF points is computed as an instantaneous frequency estimate at each of those points. In both [7-8], the practicalities of the implementation were not considered in these papers, and the results were from simulations only. Furthermore, interference mitigation based on TFDs is not a simple procedure in terms of hardware implementation and computational complexity.

Industrial research and products on interference mitigation techniques are dominated by advanced RF antenna technology. Leading industries in the field of GNSS anti-jamming have designed and developed interference mitigation technologies based on CRPA. Algorithm-based CRPA technologies include null generation/null steering, Beamforming/beam steering, STAP and SFAP. CRPA is primarily a market dominated by large defence prime contractors. For instance, 4 –elements GPS Anti-Jam Technology (GAJT)-410ML anti-jammer, a product by NovAtel [9]. It creates null steering in the direction of jammers and provides simultaneous protection to GPS L1/L2, QZSS L1/L2, SBAS L1 and Galileo E1 with a 40dB interference suppression. GPSdome by Orolia, another anti-jamming device based on CPRA [10], with a relatively smaller size that provides only GPS L1 signal protection and does not specify what level of interference it can mitigate. CRPAs are particularly appealing since they do not require modifications to the GPS receiver but are simply designed to replace the existing antenna. The downside is that CRPAs are generally much larger than typical GPS antennas due to the fact that they are composed of many antenna elements. GAJT-410ML anti-jammer weight 1.70 kg and Landshield anti-jammer, developed by Raytheon UK, is the size of a hockey puck with a weight of 1.00 kg.

Compared to defence applications, civilian and critical infrastructure applications have different requirements. One of the most miniature available Anti-Jamming products is GINCAN by Chemring Technology Solutions [11] for civilian use, measuring 48mm by 48mm in size. Its primary purpose is to provide a basic level of anti-jamming protection to vehicle satellite navigation, cellular technology and the protection of critical infrastructure dependent on GPS for PNT.

Both FD and TFD are computational intensive along with complex hardware implementation, whereas antenna-based techniques have certain factors limiting their application to compact, portable and fast-moving devices. Among different pre-correlation interference mitigation techniques, the Adaptive Notch Filter (ANF) is an attractive solution in terms of low complexity and computational cost. Past research has demonstrated its effectiveness in suppressing continuous wave interference [48] and narrowband interference [49][51], including jamming signals with linear chirp [69]. All these works are based on the adaptation of single parameter notch centre frequency and concentrate on specific interference signals rather than testing ANF for CWI, linear chirp and hop-frequency interference simultaneously. A more detailed literature review on Notch Filters and ANF is provided in Chapters 4 and 5.

As part of a government initiative, the UK has run two projects, GAARDIAN (2008-2011) and SENTINEL (2014)[12]. They aim to develop capabilities that detect RF interference within the band associated with satellite-based PNT and provide early warnings to operators of critical infrastructure and law enforcement agencies. It is estimated that the loss of GNSS service can cost the UK approximately £1 billion per day [13].

GAARDIAN stands for ‘GNSS Availability, Accuracy, Reliability and Integrity Assessment for Timing and Navigation’, and this project was led by Chronos Technology

Ltd and partially funded by Innovate UK [13]. It placed a network of Interference Detector and Monitoring (IDM) sensors around the whole UK in the vicinity of critical national infrastructure. IDM sensors are designed to monitor the integrity, reliability, continuity, and accuracy of eLORAN and GPS signals on a round-the-clock basis and report back to a central server acting as a user interface. Any anomalous behaviour in the GPS or eLoran signals was reported to users in real time. The GAARDIAN project evolved into SENTINEL (SErvices Needing Trust in Navigation, Electronics, Location and Timing) in order to provide a more robust solution to deliberate interference incidents on a national level, with real-time information on jamming. A number of key features distinguish SENTINEL from GAARDIAN, including the ability to determine the location of an interference source in real-time, the ability to detect spoofing in advance, and the ability to provide a measure of trust in a PNT system. In 2013-2017, SENTINEL systems [14] detected 6962 instances of GPS jamming at one airport, approximately five incidents per day. Both these projects helped establish the scale of GNSS jamming on the national level and provided policymakers with information to make the right decision to protect the reliability and integrity of GNSS. Two PNT-focused studies were recently commissioned in the UK, one of which is the UK GNSS Vulnerability report [13]. The other is the Blakett Review [15], which provided a comprehensive picture of the UK's PNT-related economic and technical challenges and critical infrastructure dependencies.

1.3 Research Aims and Objectives

In light of the issues outlined above, this thesis aims to develop effective and practical interference mitigation techniques for GPS L1 signals using state-of-the-art digital Adaptive Notch Filter (ANF) algorithms. It primarily focuses on GPS signals, with consideration of how the developed technique can be applied to other GNSS signals as well. To accomplish this aim, the following objectives have been defined:

- Design and develop a Simulink-based GPS L1 signal generator capable of replicating realistic GPS L1 signal characteristics with multiple parameters which can be easily adjusted to match user-desired values for signal power level, Doppler shift, code phase delay, and free path loss.
- Modelled different acquisition schemes for GPS L1 signal in MATLAB, such as serial search, parallel search, and code frequency parallel search. The receiver acquisition process's performance was evaluated in terms of coherent and non-coherent integration time.
- Identify, define and utilise performance metrics and evaluate existing and proposed techniques based on them.
- Analyse and create realistic models of interference signals such as Continuous Wave Interference (CWI), linear chirp and hop-frequency interference.

- Perform a detailed analysis of existing adaptive ANF-based interference mitigation techniques in satellite navigation systems, quantify their performance, and identify their strengths and weaknesses.
- Develop and design a novel adaptive notch filter algorithm that simultaneously adapts both notch centre frequency and notch bandwidth parameters to enhance an ANF's convergence and tracking capabilities.
- Validate the proposed method's performance via a system-level model of the entire receiver channel. Analyse and estimate fixed-point representations of wordlength and coefficients based on coefficient quantisation.
- Exploit the GPS L1 signal spectrum and its autocorrelation property to determine what percentage of the main lobe (2 MHz) of the L1 signal can be discarded and the GPS L1 signal is still acquirable.

1.4 Thesis Contribution

Publications:

S. W. Arif, A. Coskun, and I. Kale, “Multi-Stage Complex Notch Filtering for Interference Detection and Mitigation to Improve the Acquisition Performance of GPS,” in *2018 14th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, 2018

S. W. Arif, A. Coskun, and I. Kale, “A Fully Adaptive Lattice-based Notch Filter for Mitigation of Interference in GPS,” in *2019 15th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, 2019

S. W. Arif, A. Coskun, and I. Kale, “Tracking and Mitigation of Chirp-Type Interference in GPS Receivers Using Adaptive Notch Filter ,” in *2020 IEEE63rd International Midwest Symposium on Circuits and System*, 2020

S. W. Arif, A. Coskun, and I. Kale, “A Novel Optimisation Algorithm for Notch Bandwidth in Lattice-Based Adaptive Filter for the Tracking of Interference in GPS,” in *2020 IEEE International Symposium on Circuits and System*, 2020

Throughout this research work, the results and simulations rely on some basic assumptions to establish and develop the scope of the research work. The basic assumptions and main contribution of this research work are summarised as follows:

- The various noises sources and signal degradation when signal travel between Satellite In Space (SIS) and the portable ground receiver is modelled as Additive White Gaussian Noise (AWGN)
- The correlation between noise samples is assumed to be zero
- The ground-based GPS receiver is assumed to be at low dynamics with a Doppler shift range from -5 kHz to +5 kHz.

- Any non-idealities introduced by Analogue-to-Digital Converter (ADC) are assumed to be negligible
- The results, simulations and validation of proposed algorithms are performed on GPS L1 C/A signal throughout this research. However, with slight modification, the same algorithms can be adapted for the other constellation signals.

Main Contributions

- A novel Fully Adaptive Lattice-based Notch Filter, which can simultaneously adapt both notch bandwidth parameter ρ and notch centre frequency parameter β . A full gradient term was derived to update the notch bandwidth parameter ρ . The proposed method demonstrated excellent performance in terms of convergence speed and tracking capabilities, which enable better retrieval of the useful signal.
- Simulation results have demonstrated that the proposed algorithm is consistently stable and has proven superior performance under different conditions. These advantages are attained due to the constraint imposed on the pole of ANF. If the pole of the ANF moves towards the instability region (unity on unit circle or further away). In that case, it reverts into the stable model set to prevent divergence.
- For the proposed fully Adaptive Lattice-based Notch Filter, various resetting algorithms were designed, modelled, simulated and tested under different resetting criteria. The notch bandwidth parameter ρ has to be sharply reset at the beginning of each hop frequency. Therefore, a novel resetting mechanism was designed and developed with an adaptive algorithm to automatically reset ρ to the desired minimum value just before tracking the subsequent hop frequency.

- A complete system-level model is presented to compare the performance of direct form IIR ANF and proposed lattice-based ANF for GPS-specific applications in terms of C/N_0 measured at the output of the acquisition module.
- Demonstrated excision of four different types of interferences: complex sinusoidal, real sinusoidal, complex chirp-type sinusoidal, and frequency hopped type interferences via first-order complex adaptive notch filter and second-order real adaptive IIR notch filter.
- Proposed a system-level model, which is a parameterisable complex ANF, for efficient excision of CWI interference depending on the JNR of the incoming interference signal. The proposed method uses the same z_0 and k_α to replicate 2nd and 3rd order adaptive notch filters, reducing both hardware implementation and computational complexity.
- Recommended the use of the variance of the adaptive parameter z_0 as a function of JNR for estimating the power of incoming CWI interference. Hence, provide some threshold settings to toggle between different orders of complex adaptive notch filters depending on the strength of interference.
- The analysis in chapter 6 leads to a conclusion that if the 25-30% of the main lobe of the C/A code is discarded from any side of the spectrum, either from the centre, left, or rightmost, the GPS L1 signal is still acquirable

Primary achievements

- Implementation of direct form structure IIR notch filter [48], [68], along with another algorithm employing a lattice-based structure [51] for the purpose of analysis and comparison with newly developed methods and algorithms.

- A complete study of the frequency response of the ANF and its cost function provided the groundwork to understand and develop proposed methods.
- A full description, design, modelling and implementation of GPS L1 C/A signal simulator in MATLAB Simulink
- The study of three acquisition schemes; serial, parallel and double parallel
- Improving weak signal acquisition using longer dwell time employing a non-coherent detector.
- Analysis of complex notch filter of orders 1 to 6 for the mitigation of high-power CWI in terms of SNR, MSE and the acquisition Figure of Merit measured at the output of the acquisition module
- Detailed analysis of complex adaptive parameter z_0 and its multi-dimensional utilisation as a variable
- Efficient excision of the interference without distorting the useful signal.

1.5 Characteristics of GPS Signals

This chapter covers the general characteristics and properties of GPS signals and an overview of different signal processing techniques used in GPS receivers. This chapter and most of its discussions are based on references [23], [24] and [25].

1.5.1 GPS Signal Structure

The GPS signals are transmitted on three different frequency bands; L1, L2 and L5, where the RF carrier frequencies are multiples of 10.23MHz. When multiplied by 154, 120, and 115, this produces L1, L2 and L5 frequencies, as shown below:

$$f_{L1} = 154 \times 10.23MHz = 1575.42$$

$$f_{L2} = 120 \times 10.23\text{MHz} = 1227.60$$

$$f_{L5} = 115 \times 10.23\text{MHz} = 1176.45$$

GPS signal comprises three components

1. Carrier: RF sinusoidal signal with frequencies f_{L1} , f_{L2} , f_{L5}
2. Ranging Codes: Each satellite signal is accompanied by a unique sequence of +1 and -1, making it easier for the receiver to calculate the time the signal takes to reach the receiver. Each satellite has two types of ranging codes: Coarse Acquisition code (C/A code) and precision code P(Y), which is encrypted. C/A is a sequence of 1023 bits, which repeats every millisecond with a chip rate of 1.023MHz. Whereas P(Y) code is much longer (10^{14} bits long), with a chipping rate of 10.23MHz. The higher operating frequency of the P(Y) code corresponds to greater range measurement accuracy than the C/A code.

Navigation Data: The binary coded message contains information regarding satellite orbits, health status, and clock bias parameters. All this information is uploaded to each satellite from the ground control station. Navigation data is transmitted at a slower rate of 50 Hz or 50 bits per second, with a duration of 20ms. Each bit/period of navigation data consists of 20 complete C/A codes.

Each of these three components is derived from a single onboard atomic clock with a frequency of 10.23MHz; to be more precise, it is 10.22999999543MHz [23]. Figure 1.1 shows the block diagram depicting the generation of GPS L1 and L2 signals. Firstly, the C/A and P(Y) codes are combined with the navigation data using modulo-2 addition, as shown in Figure 1.1. This produces the composite binary signal. After the generation of the composite binary sequence, the next step is to modulate the carrier using Binary Phase Shift

Keying (BPSK). In BPSK, 180 degrees change in the carrier phase corresponds to a transition in the code sequence.

In order to transmit both P(Y) and C/A codes on an L1 carrier, both codes are modulated onto the same carrier frequency but with a phase difference of 90 degrees. Hence, generating In-Phase and Quadrature components, where in-phase is modulated by a P(Y) code, and a C/A code modulates the quadrature component.

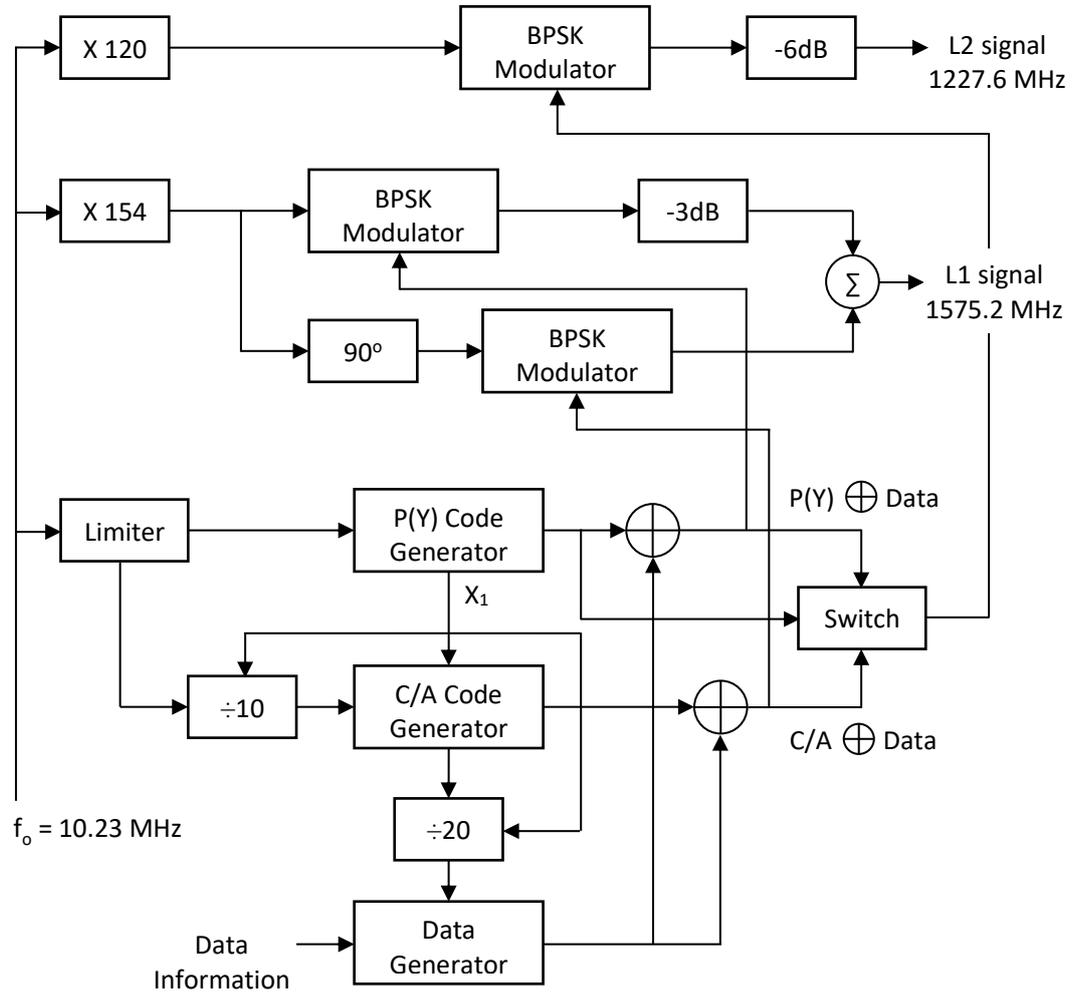


Figure 1.1 Block diagram of generating GPS L1 and L2 signal [23]

Referring to Figure 1.1, the transmitted GPS signal for satellite j can be modelled as follows

$$s^j = \sqrt{2P_c} \left(C^j(t) \oplus D^j(t) \right) \cos(2\pi f_{L1} t) + \sqrt{2P_{PL1}} \left(P^j(t) \oplus D^j(t) \right) \cos(2\pi f_{L1} t)$$

$$+ \sqrt{2P_{PL2}} \left(P^j(t) \oplus D^j(t) \right) \cos(2\pi f_{L2}t) \quad (1.1)$$

where, P_C , P_{PL1} and P_{PL2} represent the signal power level of C/A and P(Y) codes on L1 and L2 frequencies. C^j and P^j are ranging codes of satellite j . D^j is the navigation data for satellite j and f_{L1} and f_{L2} represent the centre frequencies at L1 and L2 bands. C/A code is 1023 chips long with a period of 1ms. Each navigation data bit contains 20 C/A code periods corresponding to the data rate of 50Hz. A ‘chip’ is a single pulse in the spreading code containing no information.

GPS signals operating at the L5 band were explicitly designed for Safety-of-Life (SoL) signals for aviation and maritime navigation. GPS L5 signal has two spreading sequences, one in-phase code (I5) and the other quadrature code (Q5). These spreading codes have the same chipping rate of 10.23Mcps with a code length of 10230. With the higher frequency and longer spreading codes, the precision range is very high compared to C/A code-based GPS signal acquisition, which means ranging code with higher frequency leads to a sharper auto-correlation function. Figure 1.2 illustrates the theoretical spectrum of GPS L1 C/A code and thermal noise level in the red line. C/A code main lobe is centred at ‘0’ with a bandwidth of 2MHz and higher in magnitude, whereas P(Y) code null-to-null bandwidth is -10MHz to 10 MHz (20MHz). C/A code is sitting on top of the P(Y) code and occupying 2MHz of the bandwidth. This Figure also illustrates that the GPS L1 signal is -19dB below the thermal noise level.

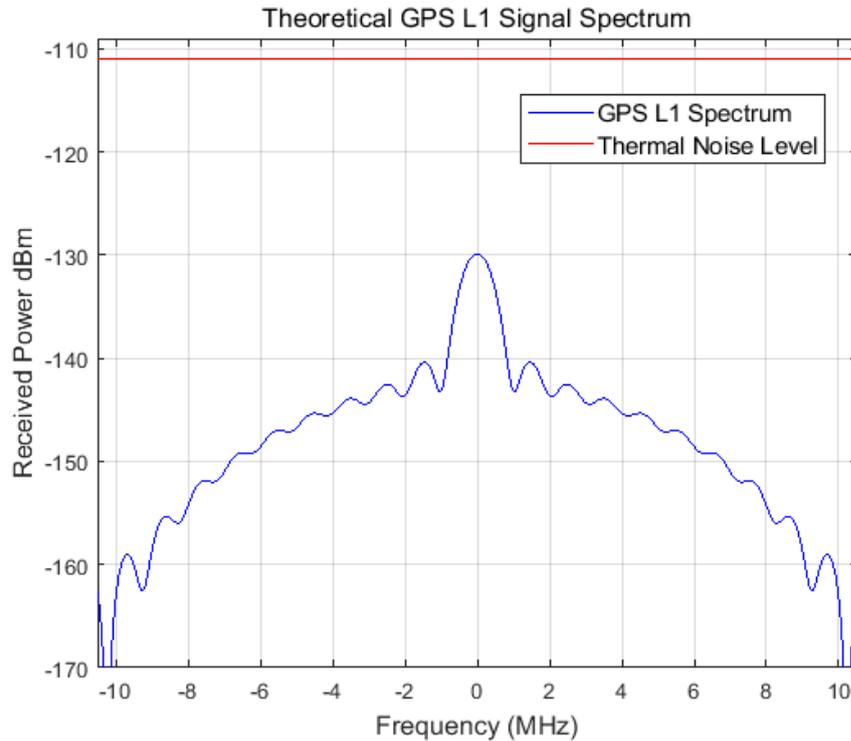


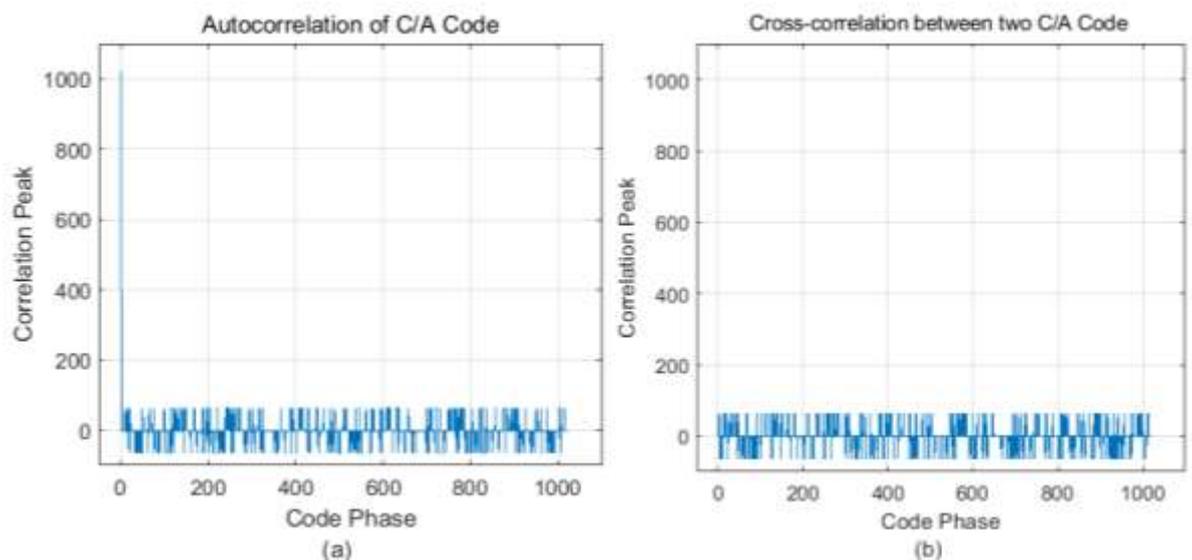
Figure 1.2 Illustrations of the GPS L1 spectrum and thermal noise level

1.6 Spread Spectrum, Ranging Code and CDMA

The navigation message is transmitted with a data rate of 50bps (50Hz), which is a very slow rate, making the navigation message information bandwidth 100Hz, which is a very narrow band. A navigation message is transmitted over a distance of 20,000km. During its journey from satellite to the Earth's surface, it must bear against all sorts of noise and interference, such as ionosphere error, multipath error and receiver clock error. To make the navigation message robust to interference, the navigation message is spread with a pseudo-random sequence. Hence, this spreads the signal across a wider band and appears like a noise-shaped signal. The signal is spread over 100 times the information band. In other words, the spread spectrum uses a unique code with a chipping rate higher than the information bandwidth. This special code is known as spreading code, PRN code, or ranging code.

Higher the rate and longer the length of a PRN code, GPS signals become more resilient to interference. It increases the precision of range measurements but at the expense of longer acquisition time. In the L1 signal, two ranging codes are used one C/A code and the other P(Y) code. The C/A code of each satellite can be transmitted at the same carrier frequency without causing interference with each other. It is known as Code Division Multiple Access (CDMA). The benefits of spreading the spectrum of navigation messages using C/A code are that it reduces interference and provides anti-jamming capabilities.

The C/A code used in each satellite is different from one another. The autocorrelation and cross-correlation properties of the C/A code make the C/A code an attractive and ideal choice for spreading navigation data coming from different satellites. At the same time, it allows CDMA. These codes are almost orthogonal to each other, which results in a sharp peak if they are correlated with itself and no peak (almost zero) if correlated with another code other than itself. Figure 1.3. (a) Shows the autocorrelation with PRN 3 with itself with a lag of 256 chips and Figure 1.3. (b) Shows cross-correlation between PRN 3 and PRN 4.



1.7 Modulation Scheme

This section discusses two modulation schemes: BPSK and Binary Offset Carrier (BOC). Both are used in GNSS signal transmission, where BPSK has been the conventional scheme since the launch of GPS, and BOC is the modernised modulation scheme used in new GNSS signals.

A BPSK modulates a GPS signal that consists of three components: spreading code, binary coded data and RF carrier. BPSK is a simple digital modulation in which the RF carrier phase is shifted instantaneously by 180 degrees when a transition occurs in the spread data. Here, binary-coded data is the navigation message and spreading code is the PRN sequences. The modulo-2 addition between navigation data and PRN code converts unipolar to bipolar ($[0, 1]$ to $[1, -1]$), and then this sum is modulated onto the carrier, as shown in Figure 1.4. A BPSK signal can only hold two distinct phases, with 180 degrees difference.

BOC is another class modulation used to generate the spread spectrum signal, particularly introduced for new-generation GNSS systems. In this scheme, the navigation message is modulated by a binary sub-carrier in addition to a spreading code and RF carrier.

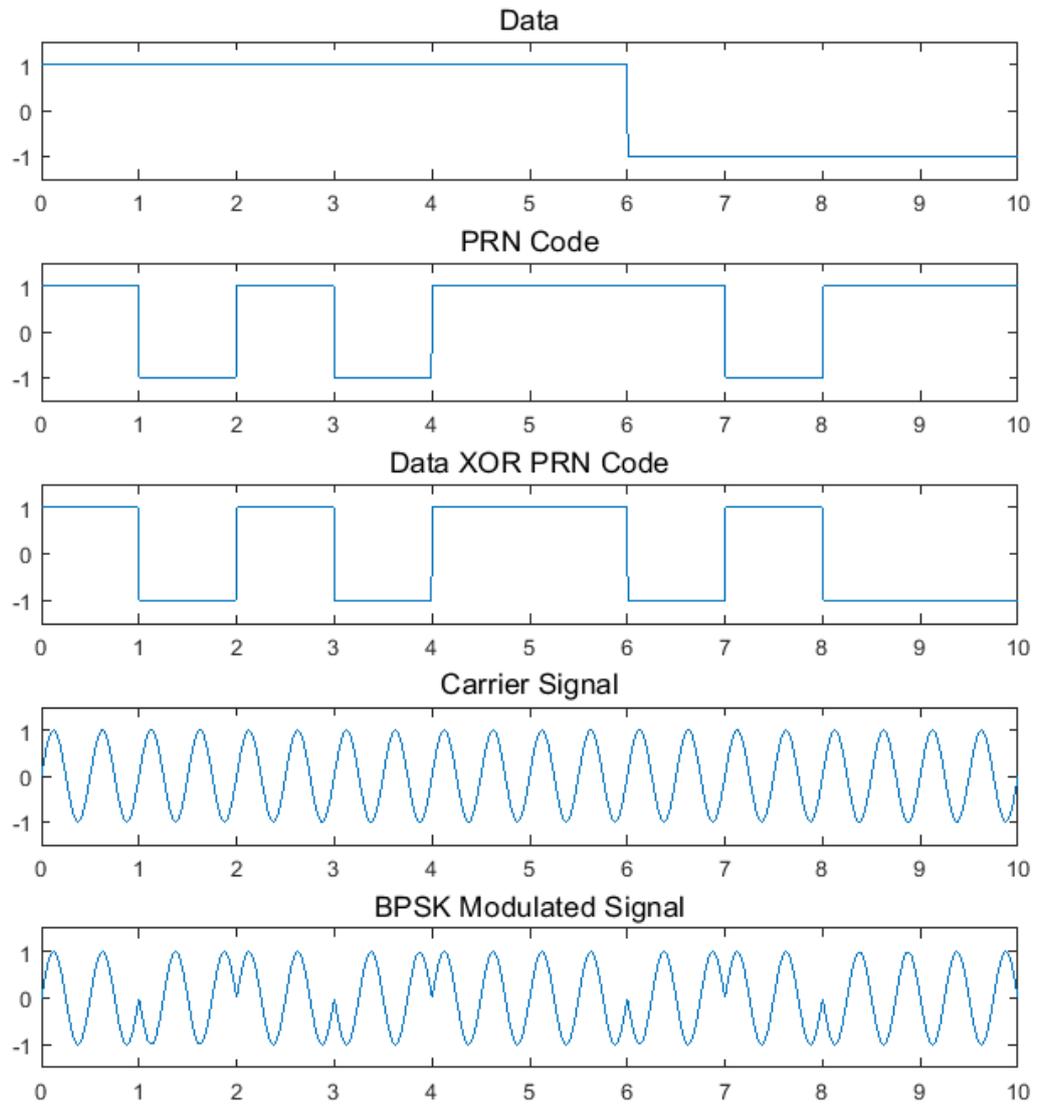


Figure 1.4 An Example of BPSK Modulation.

Figure 1.5 illustrates the BOC modulation. BOC modulation is represented by the notation $\text{BOC}(a, b)$. 'a' and 'b' are the basic factors by which the sub-carrier frequency and chipping rate increase, respectively ($a \times 1.023\text{MHz}$ for subcarrier, $b \times 1.023\text{Mcps}$ for Chipping rate). BOC modulation splits the signal spectrum into two symmetrical lobes with zero power on the carrier frequency [24], as shown in Figure 1.6.

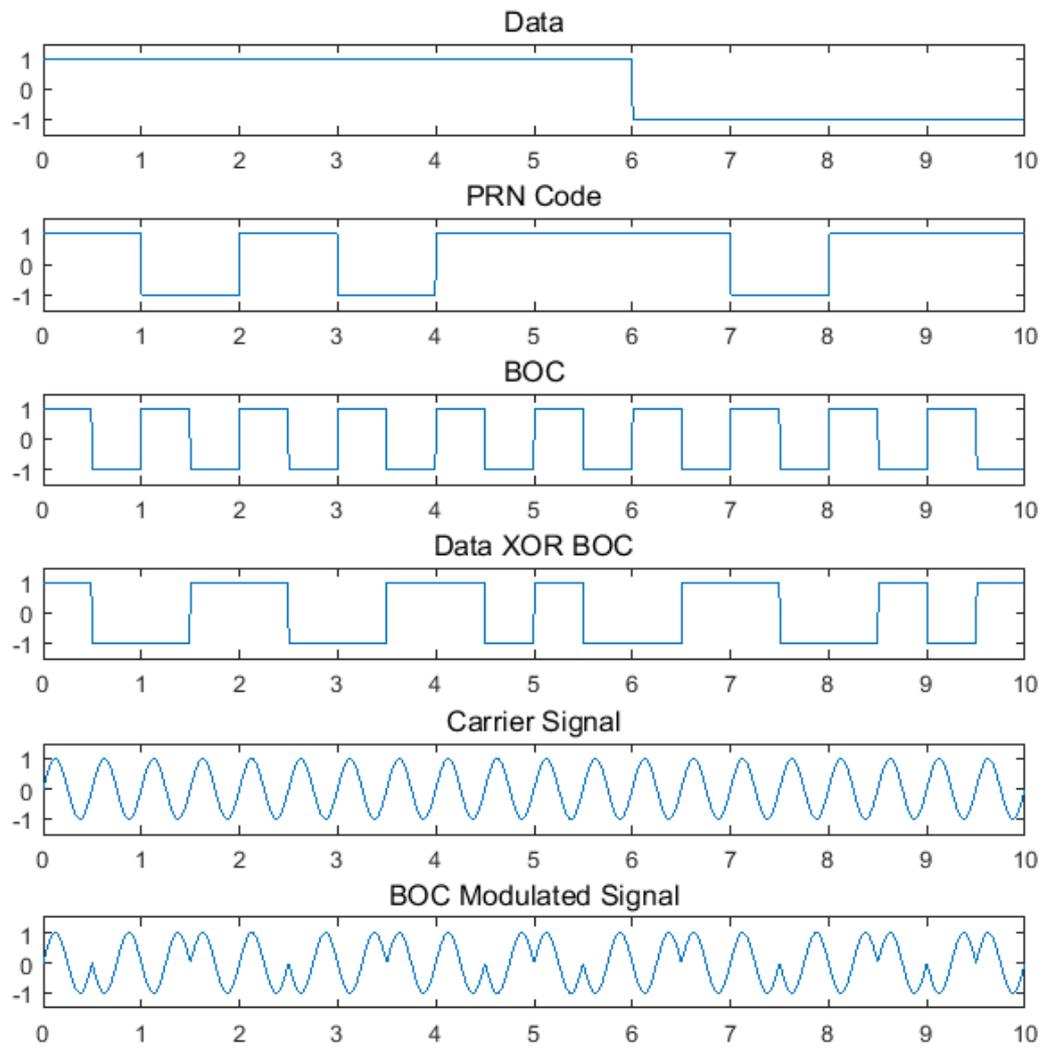


Figure 1.5 An Example of BOC modulation

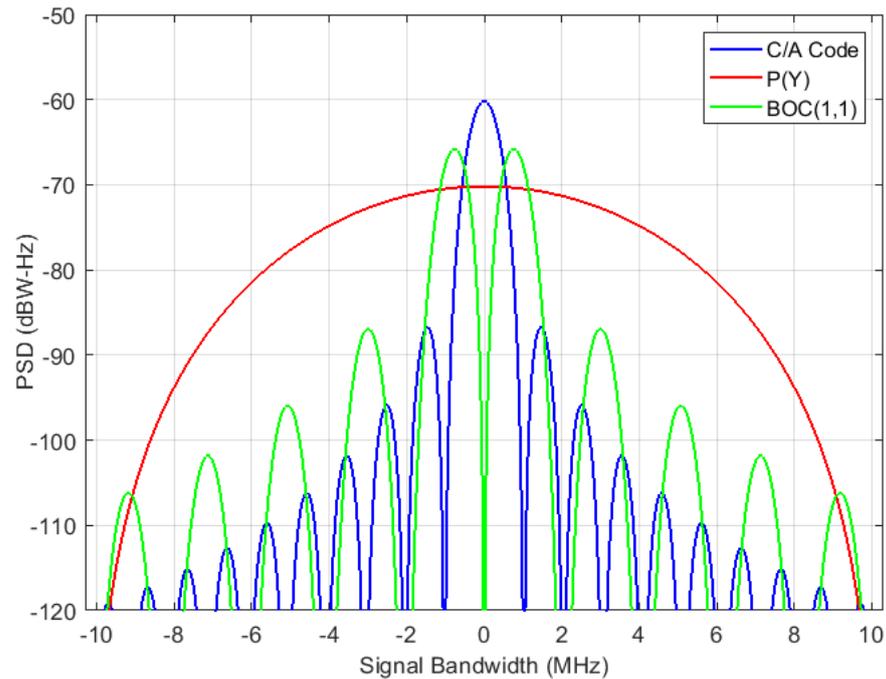


Figure 1.6 Spectrum of C/A code, P(Y) and BOC (1,1)

1.8 Outline of the Thesis

This thesis compromise of seven chapters

Chapter 1: Introduction, literature review, aims and objective, an overview of the nomenclature of GPS signal, modulation scheme and ranging code.

Chapter 2: Presents the complete development of the GPS L1 signal simulator in Simulink, with detailed explanations of each functional block. I revisited the famous Gold Code and the algorithm behind the generation of PRN code for GPS L1 signal.

Chapter 3: An overview of the GPS signal acquisition process is provided in this chapter. A brief exploration of the three most common types of detectors is presented, discussed and implemented in MATLAB. Additionally, this chapter provides insight into the performance evaluation parameters used throughout the research.

Chapter 4: Presents characteristics and modelling of different types of interference signals. The in-depth discussion on Notch Filters is divided into three sections. (1) Static 1st, 2nd and 3rd Complex Notch Filters (CNF). (2) 1st Order Complex Adaptive Notch Filter(CANF). (3) Extended 1st CANF to 2nd order real Adaptive Notch Filter (ANF). Finally, a simple and innovative system-level model is proposed utilising multi-stage CANF efficiently with a threshold setting of JNR estimation.

Chapter 5: This chapter is a complete amalgamation of the previous three chapters, which are L1 Simulator, Interference Signal, Acquisition and Existing/New algorithms comparison via performance metrics determined in chapter 3. A novel lattice-based adaptive IIR notch algorithm is presented, which can simultaneously adapt its notch bandwidth and centre frequency parameters. Presents various other intermediated algorithms and the development of threshold resetting parameters. Throughout the chapter, a larger number of simulation results are presented with an explanation. A complete derivation of the full gradient term is presented with its adaptation. A complete system is modelled to assess the performance of existing and proposed methods. Furthermore, it discusses the practical implementation of the proposed method and its transferability to the other constellation.

Chapter 6: A study focusing on developing an alternative method for mitigating interference signals. A preliminary framework and feasibility assessment are developed based on the acquisition of only the main lobe of GPS L1 signals. This is done by discarding certain portions of the main lobe before acquisition. Large sets of simulations under different scenarios are presented. This chapter concludes with its key finding

Chapter 7: Summarises the key findings of the thesis and presents recommendations for future work.

1.9 Conclusions

This chapter describes the challenges GNSS systems face regarding interference and jamming. Following is a summary of the aims, contributions, and publications, along with the outline of the thesis. This chapter also presents some of the basic structure and characteristics of the GPS L1 C/A signal, which is this research's primary signal of interest. Different modulation schemes are discussed, such as BPSK and BOC. Furthermore, the importance of using PRN code in spread spectrum communication is highlighted, which allows to transmit signals at very low power and enables CDMA. Moreover, this chapter also illustrates the autocorrelation properties of PRN codes. When two PRN codes match, an auto-correlation peak exists at the exact position of the code phase delay value. A complete description of different aspects of navigation data, P(Y) code, C/A code, and carrier frequency and modulation schemes are provided to form a coherent understanding of the signal's characteristics and the mechanism used to generate them.

Chapter 2

SIMULINK MODELLING OF GPS L1 SIGNAL

Based on the fundamentals of GPS signals presented in the previous chapter, this chapter presents a comprehensive model of the GPS L1 signal in Simulink. A virtual GPS L1 signal is modelled and generated in Simulink. This benefits in a way that it makes it possible to control the parameters of the real-data/received signal during the development of the signal processing solutions in the rest of this research work. The received GPS signal has specific parameters that are component driven and cannot be easily altered. On the other hand, generating an actual RF signal requires a hardware GPS simulator, but such a simulator is very expensive. To overcome these constraints at the hardware front-end, the GPS L1 signal is simulated with pre-defined control parameters and can be easily changed. A software-based IF signal simulator is a cost-efficient way to generate a digitized GPS L1 signal. It acts as input to the software-based receiver that the user can test and evaluate various tracking and acquisition algorithms providing enough flexibility to test different acquisition schemes for the simulated GPS L1 signal. Table 2.1 gives the list of parameters of the GPS L1 signal that must be incorporated into a simulator.

Table 2.1 Global Parameters for GPS L1 Simulator

Parameters	Description
PRN ID	Specify the C/A code to be used for the corresponding satellite.
P(Y) Code	L1 signal has P(Y) code, which is modulated onto carrier signal as a quadrature component.
Doppler	The frequency deviation from IF is due to the relative motion between the satellite and receiver.
Code Delay	Corresponds to the amount of delay between the received and local replica C/A code
Data Bits	Navigation data for each satellite with a frequency of 50Hz or 50bps
Signal Power	The GPS L1 signal power should be set accordingly, bearing in mind that the received signal is very weak
Intermediate Frequency	The simulator will generate sampled/digitized data with an IF. This IF frequency act as a reference to the calculated Doppler shift
Sampling Frequency	Enable the sampling of GPS L1 signal at different sampling rates

Simulink provides an excellent platform to design and model such a system in a very intuitive way. Figure 2.1 shows the Simulink model of the L1 simulator and illustrates how P(Y), C/A code and carrier combine to generate the L1 signal. The Simulink model shown in Figure 2.1 can be further modified to add new signal features. For the time being, this model has four main segments C/A Code, Navigation data, P(Y) code and Noise Generator.

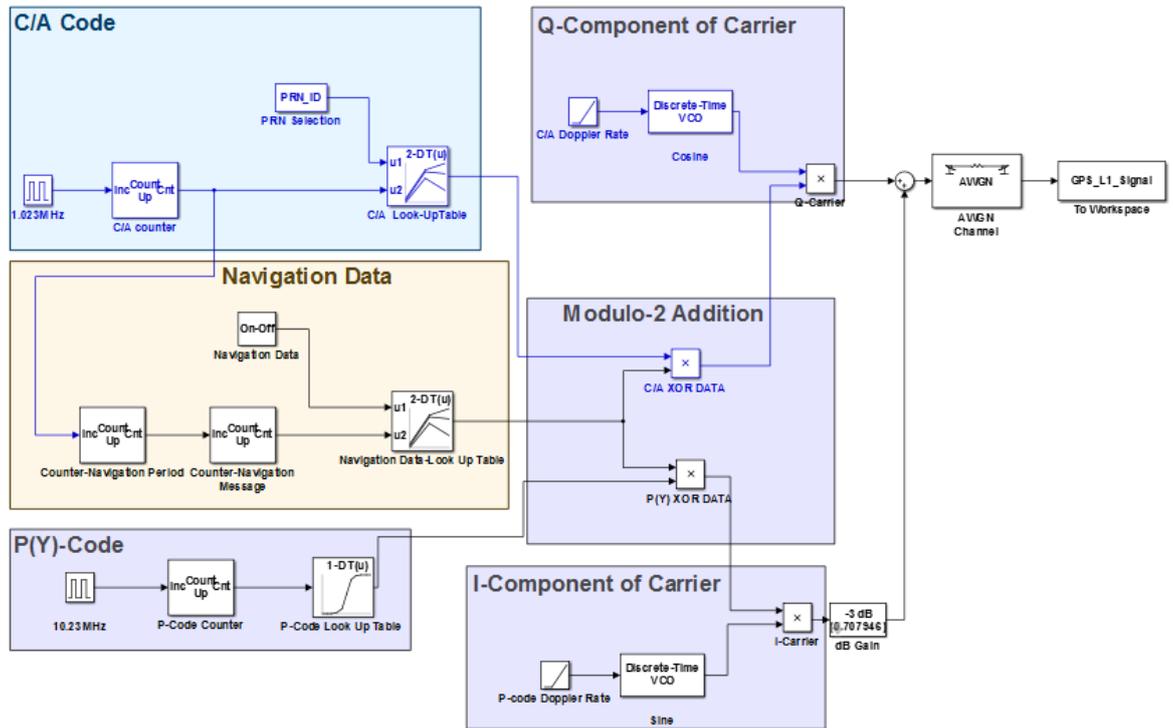


Figure 2.1 Simulink Model of GPS L1 Signal

2.1 Gold Code and PRN Code Generation

As discussed earlier, the navigation message is transmitted at 50Hz, giving a distance ranging accuracy of 300m which is very unsuitable for navigation-based applications. To increase the accuracy of distance measurements, engineers have modulated Navigation Data with a ranging code of a much higher frequency (1.023 MHz). These ranging codes, called PRN or spreading codes, belong to a family of Gold Codes [26].

Gold codes have developed by Robert Gold in 1976 [23]. Two m -length sequences are combined to form the Gold sequence. They appear to have random and noise-like properties but are generated with a carefully designed algorithm for better cross-correlation and autocorrelation properties [26]. C/A can be generated by summing two m -length sequences produced by two LFSR. C/A code used in GPS has a length of $N = 2^n - 1 = 1023$. For the generation of the C/A code, two 10-bit shift registers are used, G1 and G2 (shown in Figure

2.2), to generate the maximum length PRN codes with length 1023. Each shift register is governed by a polynomial given below:

$$G1(x) = 1 + x^3 + x^{10} \quad \&$$

$$G2(x) = 1 + x^2 + x^3 + x^6 + x^8 + x^9 + x^{10} \quad (2.1)$$

Figure 2.2 shows the block diagram of the C/A code generator, which can generate a family of $N + 2$ C/A codes that is equivalent to 1025. Because the GPS constellation has only 37 satellites, only 37 codes are generated. The phase selector block shown in Figure 2.2 determines the amount of shift to be applied to the G2 sequence before modulo-2 is added to the G1 sequence. Hence, a different phase combination corresponds to a unique C/A code for each satellite in the GPS constellation. After every 1023rd period, shift registers are reset with all ones, code starts over again and repeats itself till the end of the next cycle.

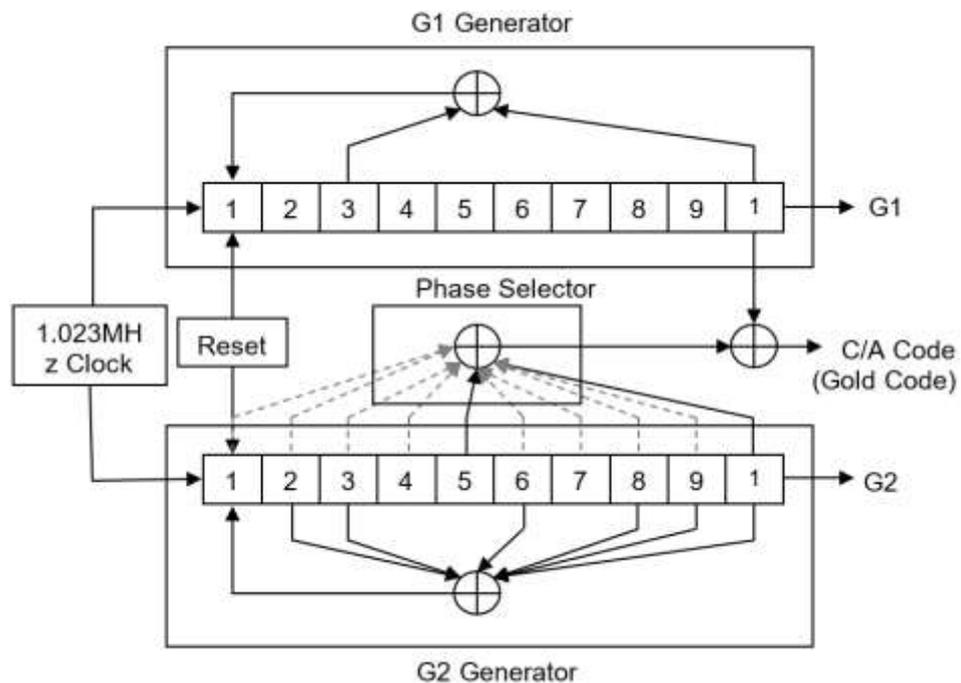


Figure 2.2 C/A code Generator

As a result of this process, a total of 37 C/A codes are generated. Out of which 32 are explicitly used for navigation purposes. The remaining five codes are utilized for control stations and ground transmitters. Similarly, P(Y) code can be generated with LFSR of length $N = 2^{12} - 1 = 4095$. P(Y) code is encrypted and not available for civilian use.

2.1.1 C/A Code Generation

Figure 2.3 illustrates the Simulink model for the generation of C/A code. The input to this block is a pulse wave clock signal with a frequency of 1.023MHz, which is the same as the chipping rate of the C/A code. This clock acts as the up counter input and counts from 0 to 1022, equivalent to 1023 chips in one C/A sequence. There is an increment in the counter's value at every falling edge of the clock pulse. As we are sampling the C/A at a specific sampling frequency, the number of samples of C/A in one period (1ms) changes accordingly. For example, sampling at 6 MHz generates approximately 6000 samples for one period of the C/A code (1ms). Similarly, a sampling rate of 12MHz creates ~12000 samples to last for 1ms. Figure 2.4 shows the output of the counter and takes approximately 6000 samples to count up to 1022, which is one complete period of C/A code.

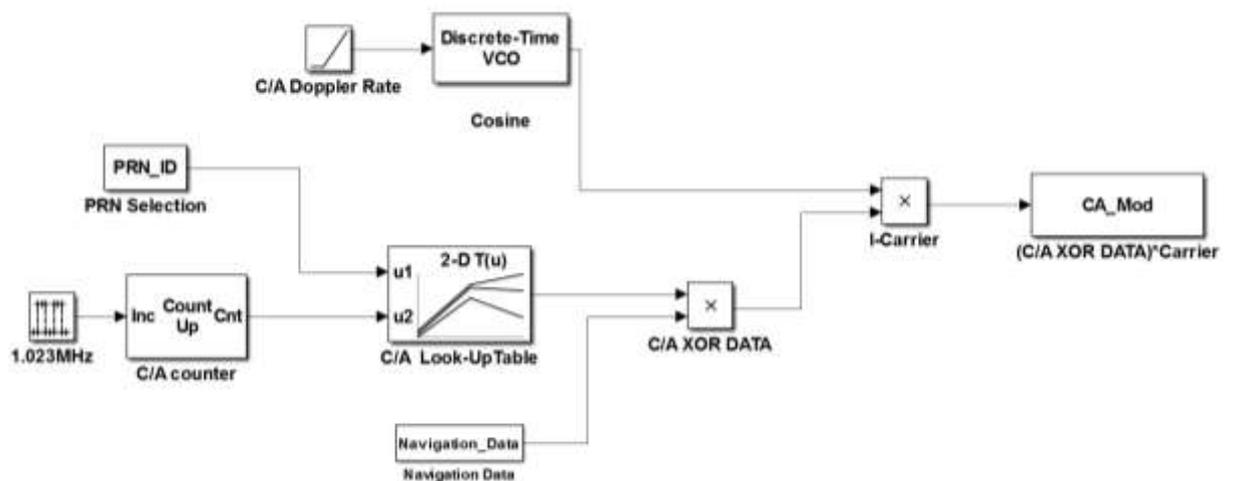


Figure 2.3 Simulink Model Generating Sampled C/A code

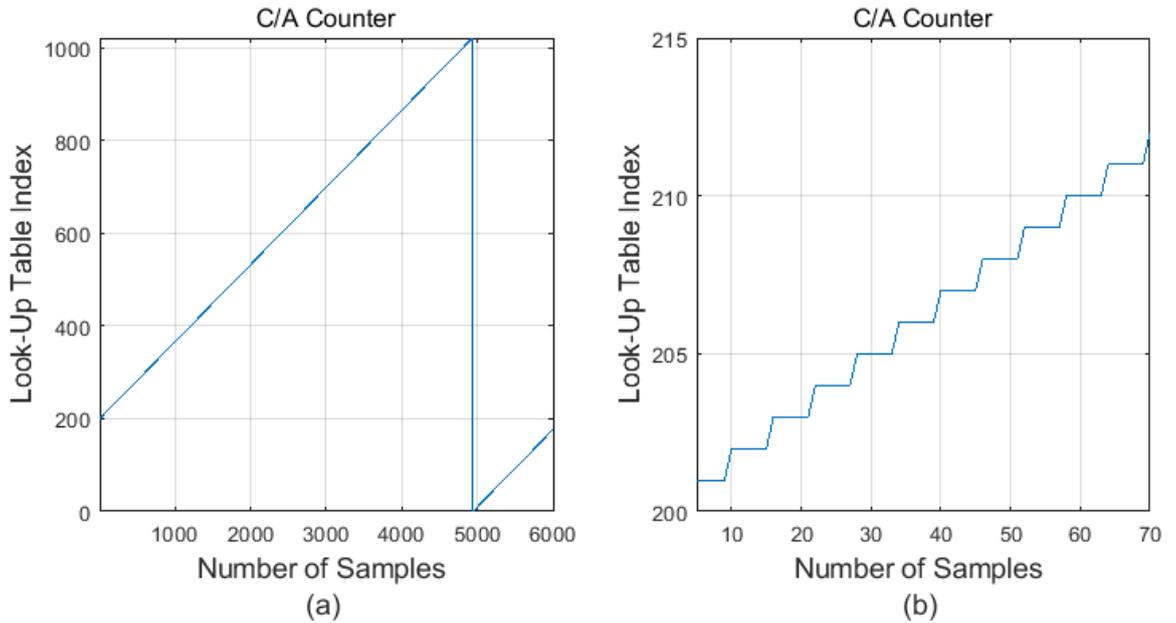


Figure 2.4 Output of C/A counter (b) Zoomed version of (a)The look-up table is basically

With a sampling rate of 6 MHz, the single chip period of 977.5ns lasts for approximately 6 samples. The counter's initial value can be set by 'Code Delay' in the counter setting. As the received C/A code is delayed in time, code delay in simulation represents the time alignment between the C/A code in the received signal and its replica at the receiver end. Figure 2.4 (b) shows a zoomed version of Figure 2.4 (a) with a code delay of 256. This staircase output of the counter is to provide the next block in the form of a look-up table. A two-dimensional array of 32 rows by 1023 columns is generated, which contains a C/A code of length 1023 bits per each of the 32 satellites. PRN ID corresponds to the C/A code of each satellite in the GPS constellation. The output of the look-up table is the sampled version of an actual C/A code. PRN ID determines C/A code generation for a particular satellite and counter-accesses all 1023 bits of that C/A code and sample at the sampling frequency. The output signal from this process is illustrated in Figure 2.5.

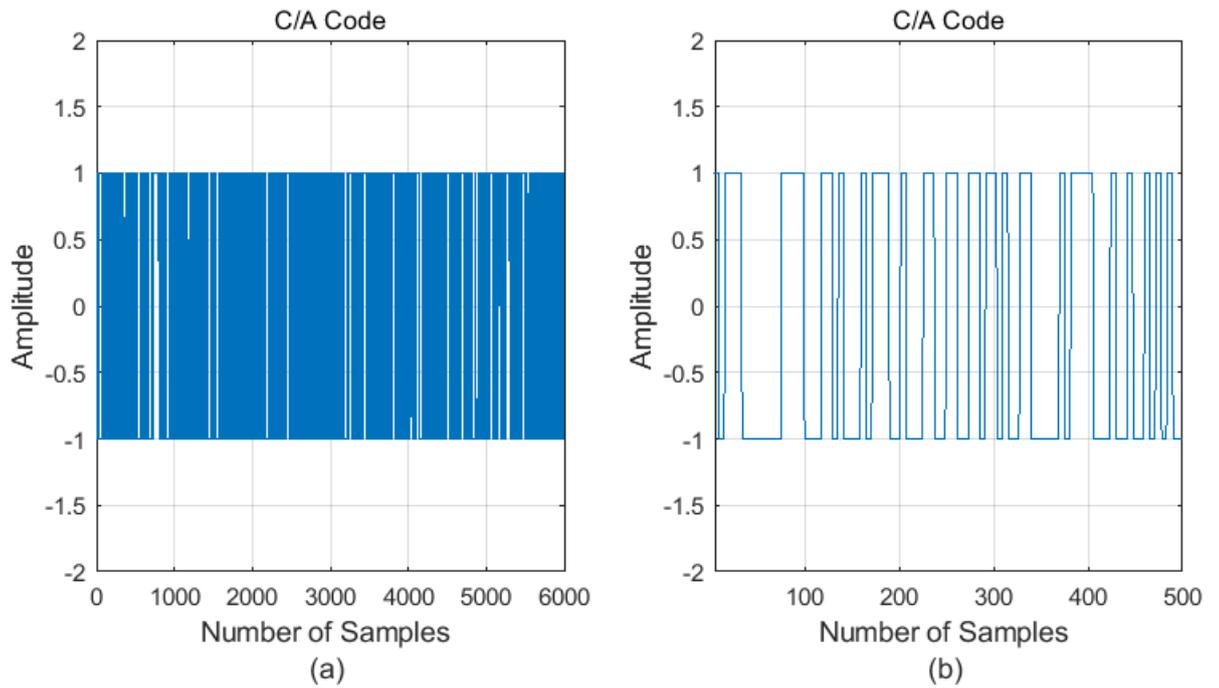


Figure 2.5 (a) Output signal generated from the C/A Look-Up Table (b) Zoomed version of (a)

The sampled C/A code is combined with sampled navigation data and modulated onto the carrier signal. Like the C/A code, navigation data is binary data with '1' and '-1' transmitted at a data rate of 50Hz. The following sections explain the generation of navigation data.

C/A code and the navigation data sequence are modulated in-phase with carrier signal as shown in Figure 2.1, denoted with the label '*I-Component of Carrier*'. The carrier signal is generated with a Voltage Controlled Oscillator (VCO) block in the Simulink model. To model the Doppler shift caused by the relative motion of the satellite and receiver, VCO is fed at the input of the ramp function, shown in Figure 2.1, which acts as a change in the frequency. With zero Doppler rate, the output frequency of the VCO stays the same and causes no drift in the carrier frequency.

2.1.2 Navigation Data Generation

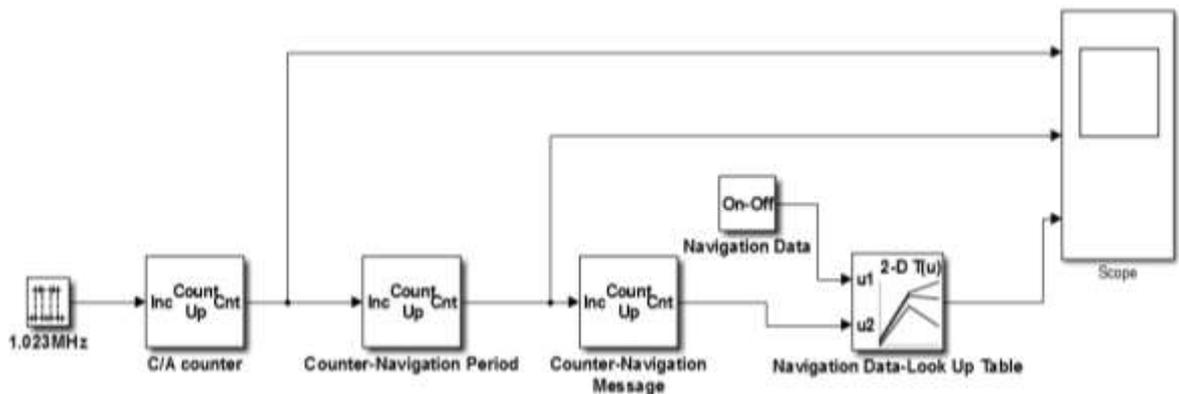


Figure 2.6 Simulink Model generating the Navigation Data

Figure 2.6 shows the Simulink model of a navigation data generator consisting of a navigation counter, a navigation period counter and a two-dimensional look-up table. On/Off option in Figure 2.6 enables the simulation of the GPS L1 signal with navigation data or without it. The navigation data rate of 50Hz corresponds to a period of 20ms to generate the C/A code, and the C/A counter runs for 1ms (at 1 kHz). We can use the output of the C/A counter as an input to the navigation period counter. The navigation period counter counts from 0 to 19. Every time the C/A counter resets or hits zero, as shown in the second plot in Figure 2.8, the navigation period counter increases by one. Hence, if the C/A counter runs for 20ms, the navigation period counter count from 0 to 19, corresponding to a single bit of navigation data, as illustrated in the last plot of Figure 2.8. As the navigation period counter reaches 20, it resets itself. In terms of time, the navigation period counter resets itself every 20ms, and this acts as input to the navigation data counter (second counter in Figure Figure 2.7). Therefore, after every 20ms navigation data counter increases its value by ‘one’, and output from a navigation data counter is input to the look-up table. The output from the navigation data counter acts as an indexed value stored in the look-up table.

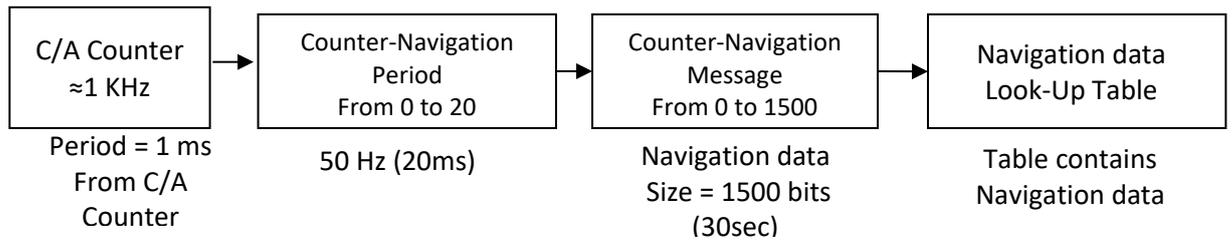


Figure 2.7 Flow Chart of Indexing Navigation Data from Look-up Table

Complete navigation data consist of 1500 bits, so there is 1500 binary value stored in a look-up table. Navigation data counter counts from 0 to 1499. In order to extract all 1500 data bits from the look-up table, the simulation must run for approximately 30 seconds. During this, the navigation data counter gets reset only once, and the navigation period counter resets 1500 times while the C/A counter resets 30,000 times (corresponding to 30 000 complete C/A codes). Figure 2.7 illustrates the flow chart to generate an output at each counter stage. The navigation bit transition can be turned off. Where no navigation data bit is wanted, the output of the lookup table is '1'. Otherwise, when navigation is turned on, the output is '-1' and '1'. Figure 2.8 shows the simulated navigation data spreading the navigation data with two different ranging codes respectively.

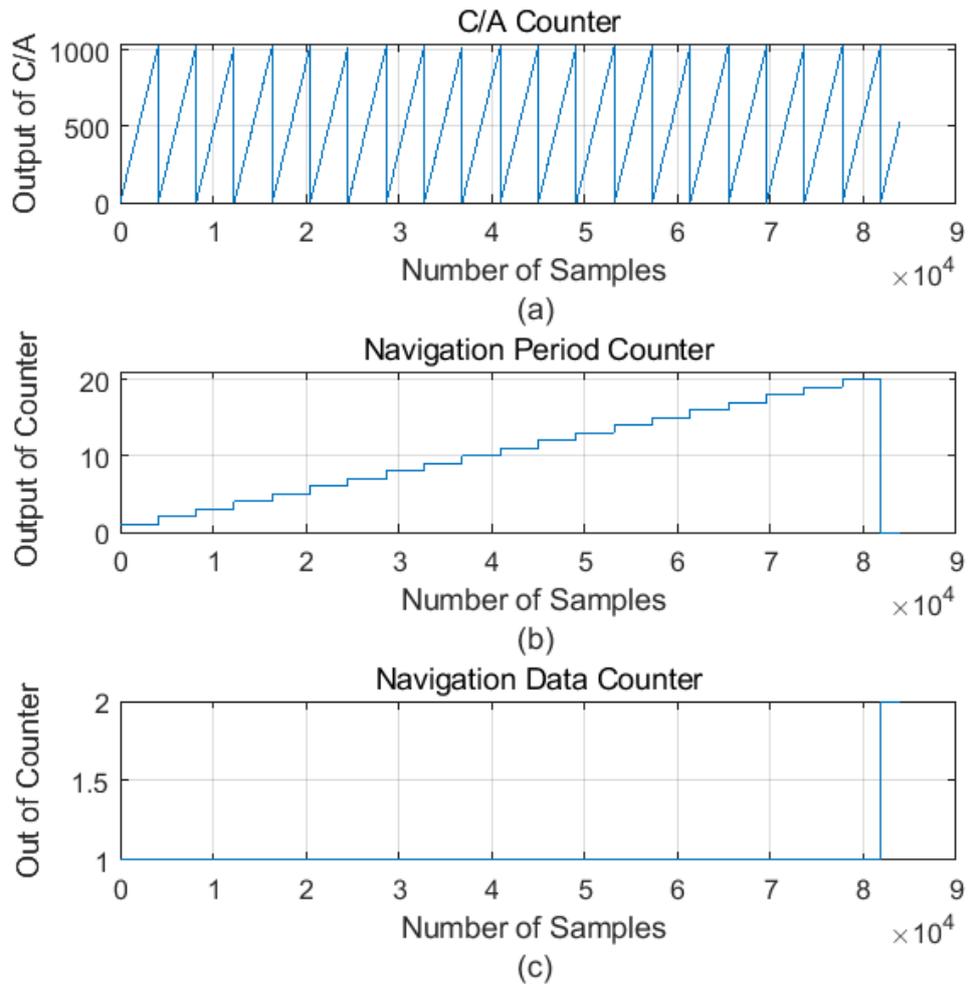


Figure 2.8 (a) 20 complete cycles of C/A counter, each triangle lasts 1ms (b) Represent single cycle of navigation period counter, equivalent to 20ms, staircase increment from 0 to 20, then reset back to zero after 20 counts (c) The output of the navigation count

2.1.3 P(Y) Code Generation

P(Y) code is encrypted and purpose-built for military uses with a chipping rate of 10.23MHz. It is a long and secure code with a period of seven days. As P(Y) is not accessible for civilian use, we simplified the Simulink model, as shown in Figure 2.9. At every falling edge of the clock, the P(Y) counter increases its value by one and indexes the binary data stored in a one-dimensional look-up table. Just like C/A code, P(Y) is binary data consisting of '1' and '-1'. The value stored in the look-up table are random '1' and '-1', and the size of the lookup

table depends on the number of samples generated in 1ms, which depends on the sampling frequency. The clock frequency is 10.23 MHz, corresponding to a P(Y) chipping rate.

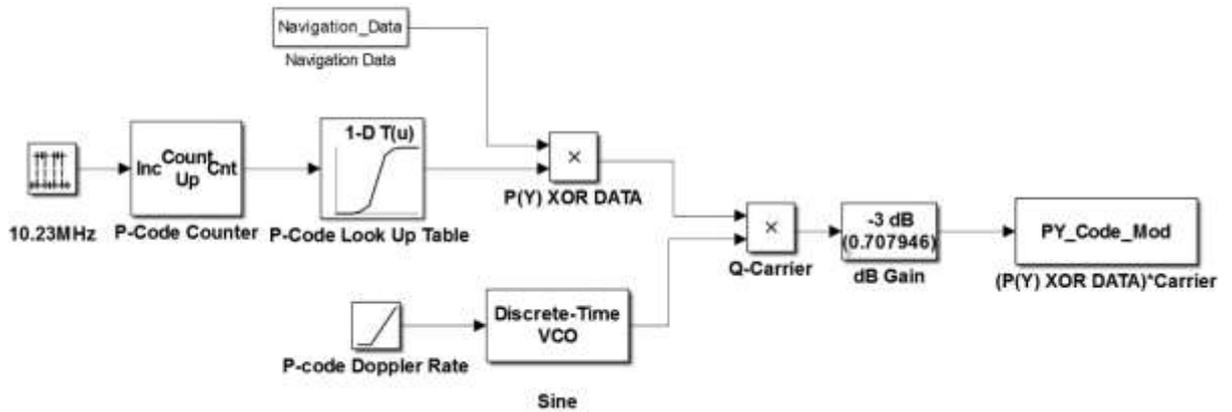


Figure 2.9 Simulink Modelling of P(Y) code

In the next step, P(Y) code is multiplied by the navigation data and spread the navigation data with a bandwidth of 20.46 MHz. The second VCO in the P(Y) model generates a carrier signal, where the only difference is that this VCO generates a carrier signal with a phase shift of 90 degrees, and the output is a sine wave. Then this signal is BPSK modulated on to carrier signal as a quadrature component of the carrier signal. The gain block is the last block in the P(Y) code, model. The function of this block is to decrease the power of modulated P(Y) code by 3dB.

2.2 Simulation Results

As explained in previous sections, each block is modelled individually and tested before joining to form a single large system. This section presents the output results of the GPS L1 signal simulator with various parameters. Figure 2.10 shows the spectrum of the C/A code and P(Y) code obtained from the simulator, one with 2.046MHz bandwidth and the other with 20.46MHz bandwidth, respectively. Figure 2.11 shows the GPS L1 signal simulated by

the Simulink model. This L1 signal is simulated without any noise added to the system.

Simulation Parameters for the Simulink Model are listed in Table 2.2

Table 2.2 Simulations Parameters

Sampling Frequency:	40.92 MHz
IF Frequency	9.540MHz
Number of Samples in 1ms	40920
Power of Received Signal	$1e^{-16}$ Watts

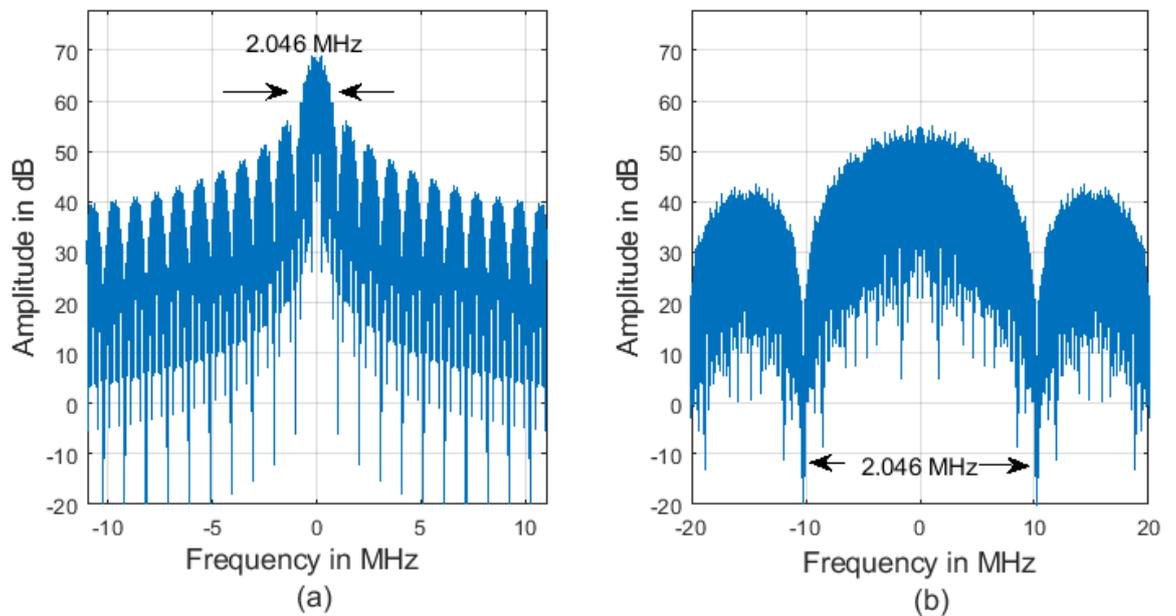


Figure 2.10 (a) Spectrum of C/A code, (b) Spectrum of P(Y) code

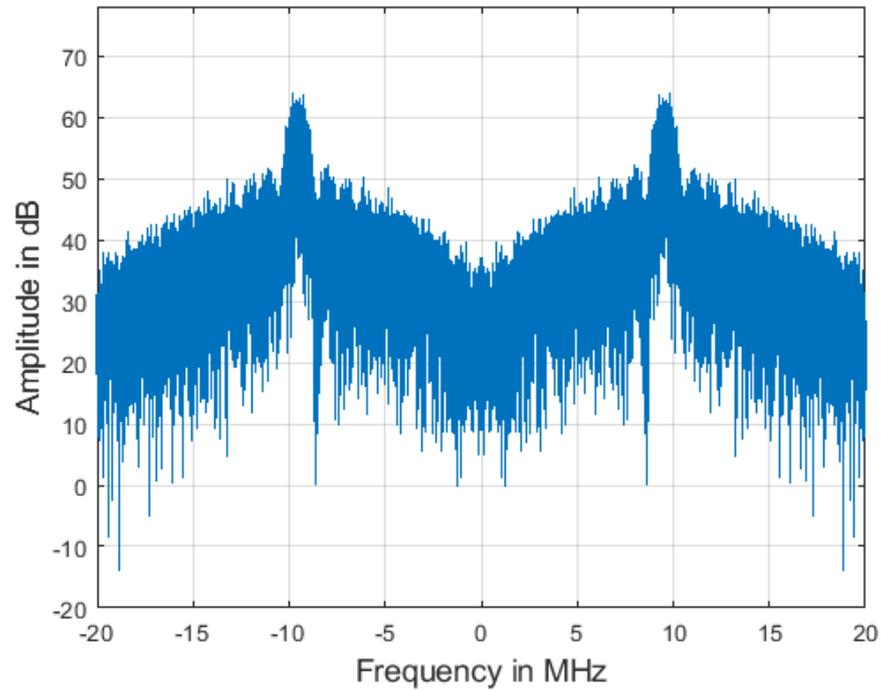


Figure 2.11 Output of GPS L1 Simulator at the carrier frequency of 9.55MHz

2.3 Conclusions

This chapter has outlined the basic operation of the SIMULINK-based GPS L1 signal generator, a toolbox designed and developed in MATLAB for end-to-end modelling, simulation and analysis of GPS L1 signal. A complete step-by-step design and modelling are presented with a set of output waveforms at the end of each block. A software-based GPS L1 signal generator provides more flexibility over controlling different parameters while testing different algorithms. The GPS L1 signal generator comprises three main components; C/A code generator, navigation data, and P(Y) code generator. The SIMULINK-based model also allows altering parameters such as IF frequency, Doppler rate, code phase delay and thermal noise levels per requirements. Furthermore, this chapter also provides insight into Gold Codes and describes the generation of PRN codes via maximum length register.

Chapter 3

ACQUISITION OF GPS L1 SIGNAL

This section overviews the acquisition of the GNSS signal via different detectors/estimators, focusing on the GPS L1 signal acquisition. The signal acquisition process is a coarse alignment process, which gives the estimated value of code phase delay and Doppler frequency of the received code. The two-dimensional search for GPS signal acquisition requires a simultaneous search of all possible code phase delay values against Doppler frequency (evaluated with a granularity of a few hundred Hertz). When both the received and local PRN code synchronizes correctly, a sharp peak is visible at the Cross Ambiguity Function (CAF) output, and its location corresponds to the code phase delay and the carrier Doppler shift.

Under harsh conditions, acquiring a GPS L1 signal becomes even more complex and impossible. The receiver's sensitivity is enhanced simply by using longer integration/dwell time when acquiring the GPS L1 signal. Both coherent and non-coherent detectors can be employed to increase dwell time and are discussed in the following chapter. Furthermore, a complete GPS L1 signal Front-end modelling is presented at this chapter's end, along with the performance evaluation metrics description.

3.1 GNSS Signal Acquisition

Signal acquisition is one of the critical operations within the receiver, also known to be the ‘holiest’ of all processes in a GNSS receiver. The accuracy and precision of the acquisition process affect the later operations in the receiver, such as tracking and receiving the navigation message. Different acquisition schemes are explored, studied, and carried out via MATLAB modelling to get the essence of the acquisition process within the receiver.

During signal acquisition, three main variables are estimated [24]:

- The arrival time (τ): It contains the basic range and clock information required to compute user position and clock offset.
- Doppler Shift (f_D): This provides pseudo-range rate information and calculates user velocity and clock frequency.
- Carrier Phase Shift ($\delta\theta$): For higher precision and accuracy, the carrier phase offset is also estimated and tracked.

The signal acquisition block determines the coarse value of the code phase and carrier frequency offset for all visible satellites with respect to the receiver's position. These values are further used to initialize the tracking of the channels. Acquisition of GPS signal is made in two stages. In the first stage, a coarse estimation of τ and f_D are carried out, and the second stage involves a more accurate calculation of τ and f_D , known as fine acquisition. If the second stage involves the estimation of carrier phase offset, then it is called coherent tracking. If carrier phase offset is ignored, it is called non-coherent tracking [24]. The acquisition process requires replicas of both the ranging code (PRN code) and the Doppler-shifted carrier of the satellite to be tracked. The incoming signal is correlated with each PRN code across all possible combinations of code delay and Doppler frequency bin [27]. The ideal output of acquisition is a very sharp peak that corresponds to τ and f_D . The peak occurs

only when the locally generated frequency matches the carrier frequency and when the locally generated PRN code is perfectly aligned with the received code.

Various algorithms are used to acquire GNSS signals, such as serial search or time domain acquisition, parallel code phase search or FFT-based algorithm, parallel frequency space search acquisition [24] and matched filtering [28]. Each of these algorithms can also be used to acquire BOC-modulated satellite navigation signals by making a slight modification to these algorithms [29]. The following sections briefly explain each acquisition method, with results produced in MATLAB.

3.1.1 Serial Search Acquisition

A serial-based acquisition scheme computes the time domain correlation between received and locally generated signals. The search is carried out in sequential mode, and for each frequency bin, all possible code phase delays are searched before moving on to the next frequency bin. A PRN code is 1023 chips long, and each chip is compared in half-cycle intervals. A total of 2046 half chips need to be compared. The frequency offset due to the Doppler shift is +/- 20 kHz divided into 40 frequency bins, 1 kHz each. The total number of searches thus comes out to be $2046 \times 40 = 81840$ [25]. The frequency setting keeps changing in steps of 1 kHz until the satellite signal is detected. If no satellite signal is detected after searching the whole combination grid, then the PRN code is changed, and the entire searching process starts again. The serial search is thus a long process, and with each search being of 1ms, the whole grid search may take much more time to compute. The hardware required for the serial search is simple but involves more computation paths and resources. These drawbacks make the serial search process inefficient. However, in terms of hardware implementation, it is fairly easily realizable with a simple accumulation of I-component and

Q-component followed by the removal of phase reversal by taking the square of absolute values of I and Q components before adding them, as shown in Figure 3.1.

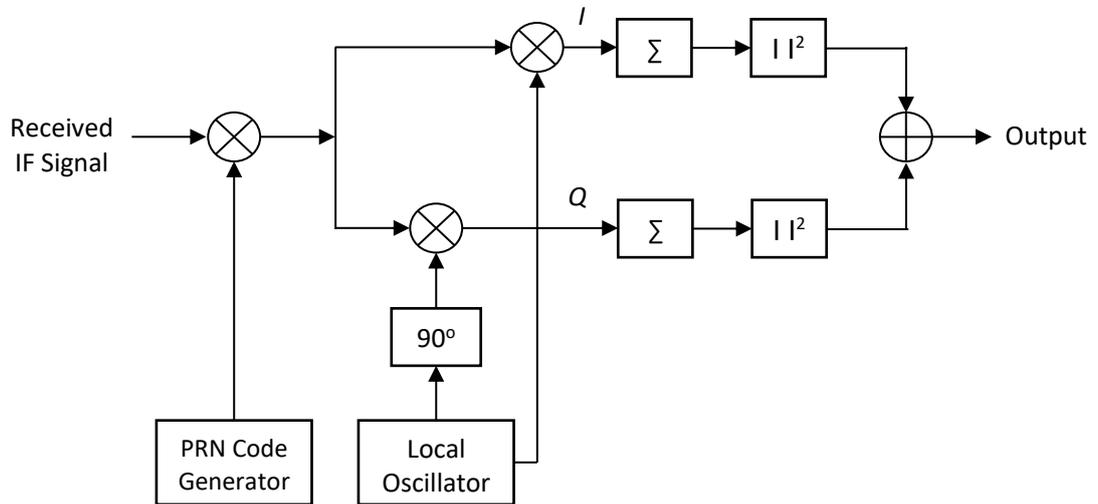


Figure 3.1 Block Diagram for Serial Search Acquisition Algorithms

Figure 3.1 shows the block diagram for the serial search algorithm where a local replica of the PRN code is generated to de-spread the received IF signal. On the other hand, the local oscillator generates a Doppler-shifted carrier signal at specific frequency bins with a range of 20 kHz. Then a phase shift of 90 is applied to the output of the local oscillator to produce a quadrature phase carrier signal from the local oscillator. Once both operations are concluded, the carrier signal's de-spreading and removal are accomplished, and the in-phase and quadrature signals are obtained, referred to as I and Q in Figure 3.1. If there is a satellite signal in the received IF signal and if both locally generated signals, namely, PRN code and carrier signal, are perfectly aligned with the IF signal. The output of the serial search algorithm shows a sharp peak that a simple logic block can determine. Figure 3.2 shows the output of the serial search acquisition algorithm modelled in MATLAB, using the GPS L1 signal described in Chapter 2, which uses the following simulation parameters:

- The sampling frequency of 40.92MHz
- IF signal frequency of 9.540 MHz
- Accumulation of one period of PRN code, 1ms.

3.1.2 Parallel Frequency Search

As discussed in the previous subsection, serial search is a time-consuming process involving a large number of operations but has lower hardware complexity. The parallel frequency search holds an advantage over its serial search counterpart by parallelizing the search process in the frequency domain and reducing the computational complexity requiring a smaller number of operations. Hence, it reduces the time to complete the signal detection search. In contrast to time-domain serial search, this algorithm utilizes a Fast Fourier Transform (FFT) operation to carry out frequency domain processing at the cost of increased complexity in the form of the FFT.

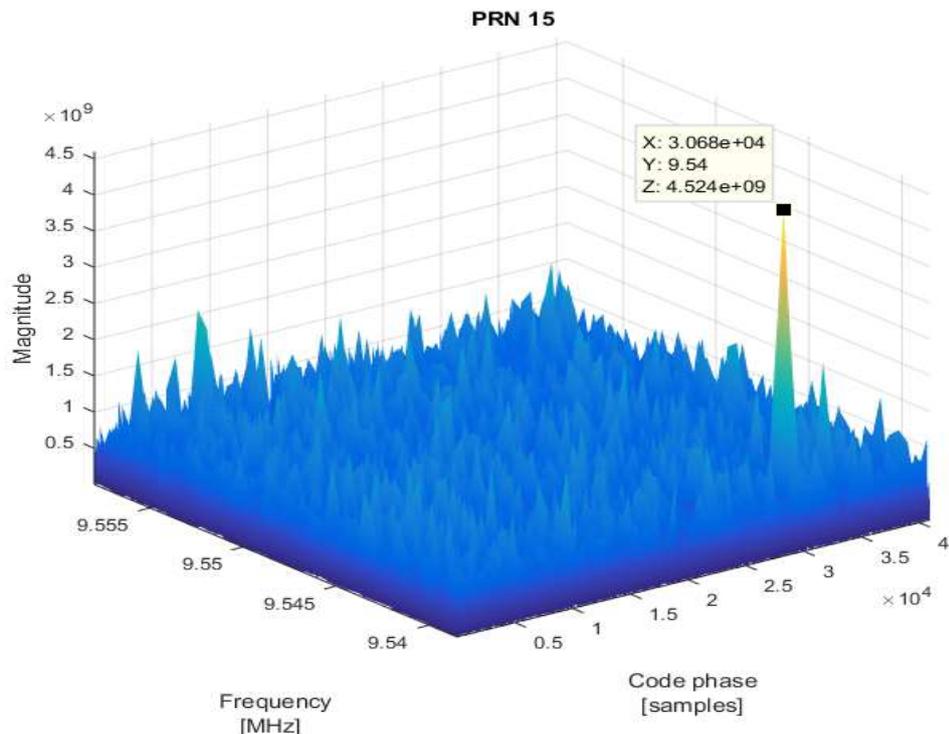


Figure 3.2 Serial Search Acquisition of PRN 15

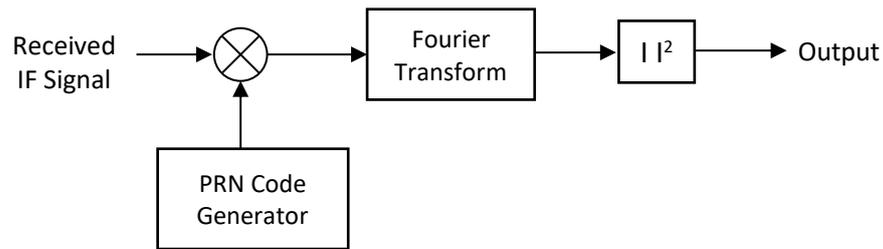


Figure 3.3 Block Diagram for Parallel Frequency Search Acquisition Algorithms

Figure 3.3 illustrates the algorithm for the parallel frequency search acquisition. The received IF signal is multiplied with the local PRN replica code, whose phase changes from 0 to 1022, which is then transformed to the frequency domain via Fourier transform. When the local replica of the PRN code is well aligned with the code in the received IF signal, it simply de-spreads the incoming signal. The result of this operation becomes a continuous-time signal with Doppler-shifted carrier frequency. The Fourier transform of this resulting signal is computed, showing a peak at a specific frequency corresponding to the carrier frequency, including the Doppler offset. Hence, in this method, there is no need to search for each frequency bin to locate the frequency of the carrier signal.

3.1.3 Parallel Code Phase Search

In parallel code search, all the local PRN codes are searched simultaneously. In parallel code search, the Fourier transform of the PRN code is performed, and then the complex conjugate of the Fourier transform of the PRN code is multiplied by the Fourier transform of the received signal. The inverse transform transforms the resultant output back into the time domain. The final output thus represents the correlation between the received signal and the PRN code. The peak value of the output is then compared with the threshold. If it exceeds the threshold in the correlation domain, then that search grid position of the peak indicates the code phase and carrier frequency of the incoming GPS signal. The total number of computations of the parallel code search is limited to 40 [24]. It has been observed from the literature review that the parallel frequency space search algorithm proved to show

measurable and reliable results as compared to the parallel code phase search at the expense of acquisition time. This algorithm can be implemented on a reconfigurable architecture [30].

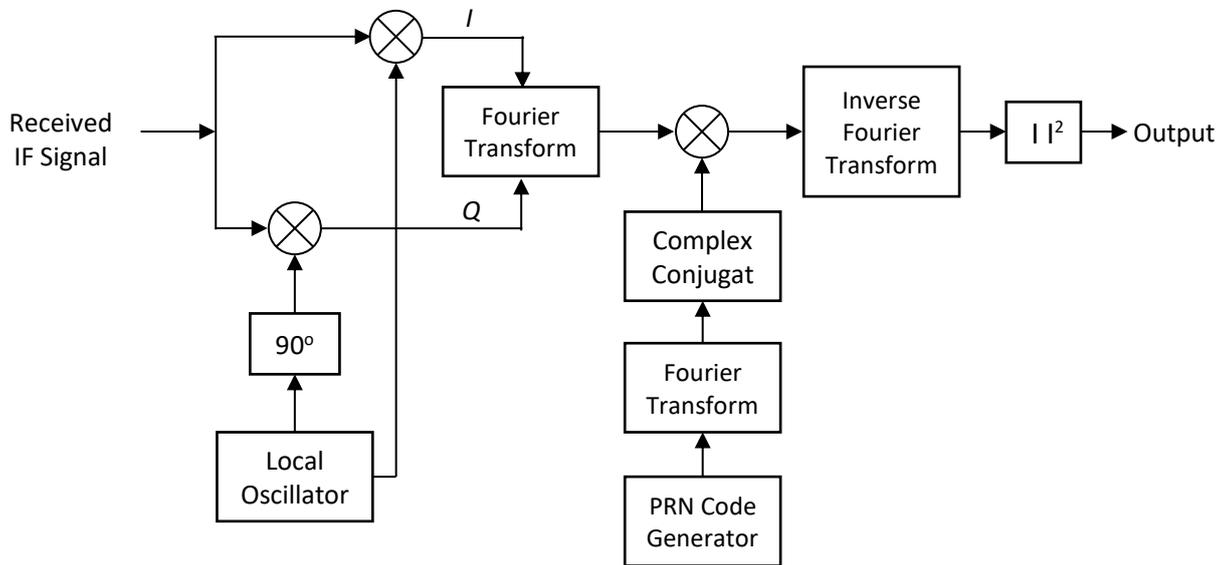


Figure 3.4 Block Diagram for Parallel Frequency Code Phase Search Acquisition Algorithms

Figure 3.4 shows the algorithm for the parallel code phase search. The first step is to remove the carrier signal from the received IF signal by multiplying it by the specific frequency generated by the local oscillator, which creates the I and Q components of the received signal. I and Q components are combined to form a complex signal followed by a Fourier transform operation, converting into the frequency domain. Similarly, a local replica of PRN code is transformed into a frequency domain followed by their complex conjugate. Both signals are multiplied in the frequency domain, which is equivalent to circular correlation in the time domain. This result is transformed back into the time domain by taking its inverse Fourier transform and preceded by squaring operation.

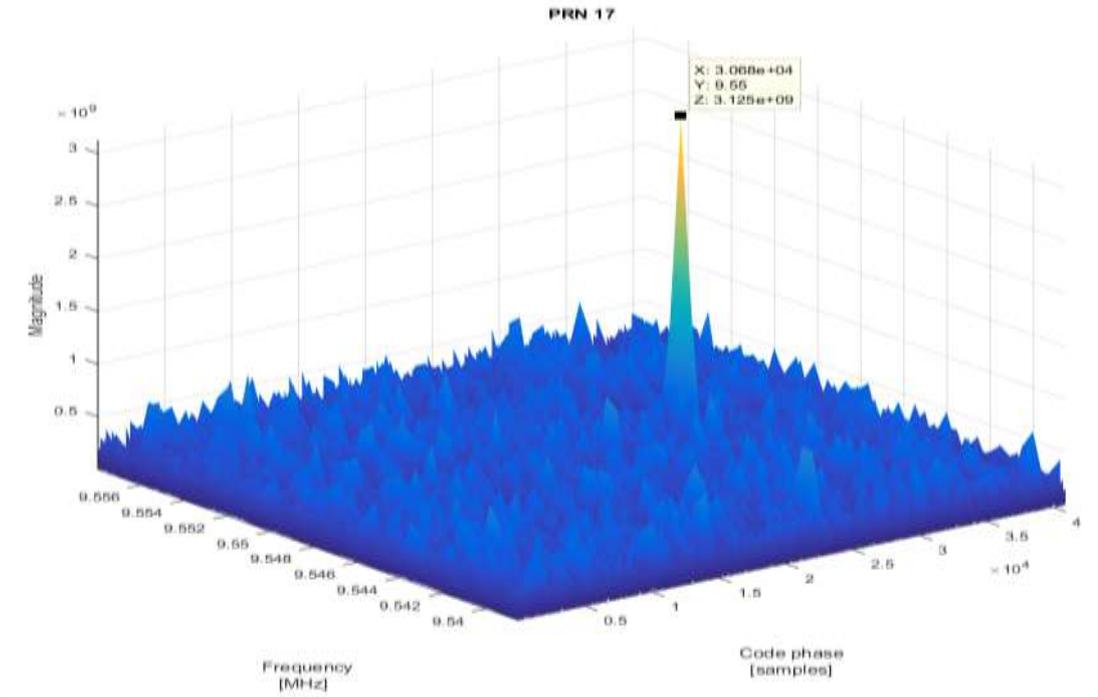


Figure 3.5 Parallel Frequency Code Phase Search of PRN 17 with the Noise level of 30dB

When both received IF signal frequency and code are perfectly aligned with a local replica of PRN code and carrier frequency, the output shows a correlation peak at the corresponding code phase delay and carrier frequency. Both Figures 3.5 and 3.6 are generated with different noise levels. In Figure 3.5 noise level is set to 30dB and the main peak is very prominent. When the noise level is increased, the noise floor around the main peak also gets elevated, as shown in Figure 3.6.

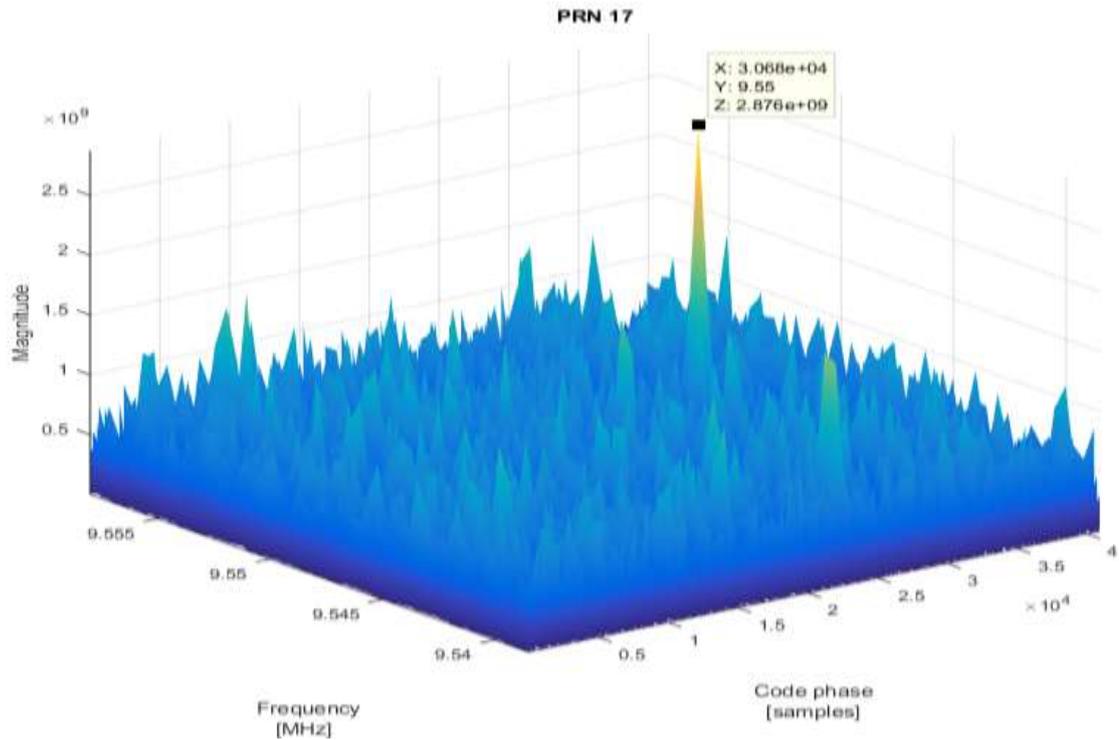


Figure 3.6 Parallel Frequency Code Phase Search of PRN 17 with a Noise level of 45dB

3.2 FFT-Based Acquisition Algorithm

The previous section modelled different acquisition techniques in MATLAB and simulated them. The FFT-based algorithms are far more efficient than time-domain correlation in serial search algorithms. The acquisition algorithms are the most challenging part of a GNSS receiver because the acquisition block has to search for a GNSS signal by scanning an unknown region, which has to be accomplished quickly. In a cold start, the first operation of the receiver is to acquire a signal and test all possible combinations of code phase delay and Doppler frequency shift in two dimensions. Figure 3.7 shows the search grid; each square represents the cell corresponding to a single time-frequency bin. Hence, iterations through each bin become a time-consuming process. In a conventional receiver, to speed up the acquisition process, the receiver is employed with a large number of correlators [28].

Similarly, the acquisition process slows down if a few correlators are used. Reduction of both the number of correlators and the acquisition time in the cold start is not an easy task for a conventional receiver. There are alternative GNSS acquisition methods, such as compressed sensing [31]. In the compressed sensing method, the signal's sparsity is exploited to reduce the number of measurements required.

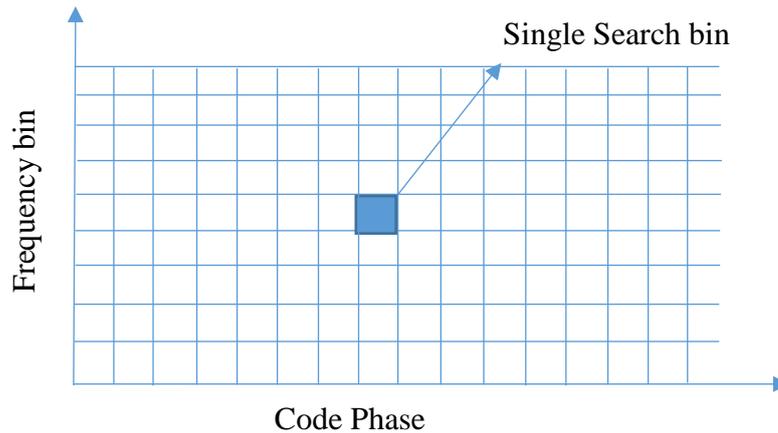


Figure 3.7 Search Grid for GPS signal acquisition

In this method, a local replica of the ranging code is folded, and a correlation is performed between the received signal and the local folded signal. But there is one downside: SNR degrades as the number of folding increases [33]. Recently, FFT-based acquisition algorithms have caught more attention, as they can implement the acquisition process without needing a large amount of hardware (such as massive correlators). The ideal receiver would be one that can acquire signal relatively in less time, with less number of correlators and lower computational power than other conventional receivers. General limitation factors for FFT-based acquisition are summarized below

- Size of the search grid
- Number of samples to compute (depends on the sampling frequency)
- Type of DFT algorithms

3.3 Acquisition of Weak GPS signal

In order to increase the sensitivity of the GNSS receiver, different detection schemes are used, most commonly coherent and non-coherent integration. Both methods differ in how the successive outputs of the correlator are combined and integrated. Coherent integration is the first step to the acquisition of any GNSS signal. In harsh environments or when the GPS signal is very weak, 1 millisecond of data is not enough to detect the signal. Therefore, a longer data set is recorded to process via coherent or non-coherent integration. By increasing the length of the data, more number of operations are needed in the acquisition unit. One way to process a longer set of data is via non-coherent integration. For example, if 5ms of GPS data is recorded, it can be divided into five sets of 1ms and individually processed before performing the square operation and summing them together. The signal is acquired whenever the result of 1ms long non-coherent integration crosses a certain threshold. Squaring block removes the phase information, and successive cross ambiguity function (CAF) can be summed together non-coherently.

On the other hand, coherent integration is performed on a longer set of data with an integration time of more than 1ms. Dwell time of coherent integration is limited by Navigation Bit as the navigation occurs every 20ms. If navigation data cause a phase shift, the acquired data with a phase shift affect the acquisition, leading to degradation of the acquisition unit. In [47], it is shown that the maximum effective length of coherent integration time for GPS C/A code is 16ms. However, extending the coherent integration time to a larger number of operations needs to be done at once, which means more FFT points are needed to compute the acquisition of the signal, hence leading to a higher memory requirement.

Table 3.1 Comparison of two detector methods

Techniques	Pros	Cons
Coherent Integration	Improvement in SNR	More operations/memory required
Non-Coherent Integration	Avoid phase shift caused by Navigation bit	Introduce squaring losses

Extending dwell time to non-coherent integration time leads to squaring losses, which means the main correlation peak emerges from the noise but the noise average around non-zero value. Figure 3.8 shows the acquisition scheme with the no-coherent detector. The following equations can represent coherent and non-coherent integration, assuming the output of Correlation at each 1ms is $[R_1, R_2, \dots, R_m]$. N represent the number of samples in the correlation of 1ms of data.

$$S(\tau, f_D) = \frac{1}{N} \sum_{n=0}^{N-1} r[n]c[n - \tau]e^{-j2\pi f_D n} \text{(Used in chapter 4)} \quad (3.1)$$

$\frac{1}{N} \sum_0^N (\cdot)$ Represents coherent integration before envelope evaluation, τ represents code phase delay and f_D Doppler frequency

$$R_M(\tau, f_D) = \sum_{m=1}^M S_m(\tau, f_D) \quad (3.2)$$

S_m Represents CAF at the output of the Non-coherent detector.

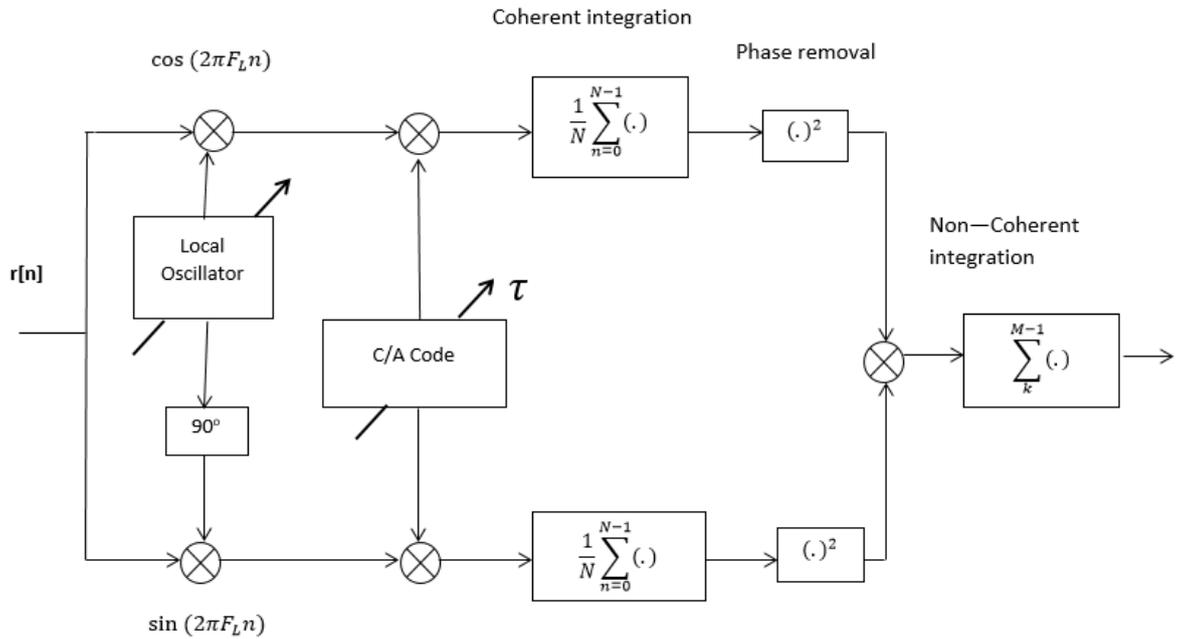


Figure 3.8 Acquisition scheme with a non-coherent detector

A very weak signal was generated using the GPS L1 signal simulator built in chapter two, and non-coherent integration of 12ms was performed to detect the signal. Figure 3.9 shows the evolution of the correlation peak over successive non-coherent integration of 12ms. In Figure 3.9(a), no correlation peak is visible when 1ms of data is used to acquire the GPS L1 signal. While in Figure 3.9 (b), with 5ms of Non-coherent integration time, the main peak slowly emerges from the noise floor and in Figure 3.9 (c), at 7ms, the main peak is more clearly visible. In Figure 3.9 (d), 12ms of non-coherent integration leads to an even more prominent peak and the noise floor around the main peak is lowered further if compared with (a), (b) and (c)

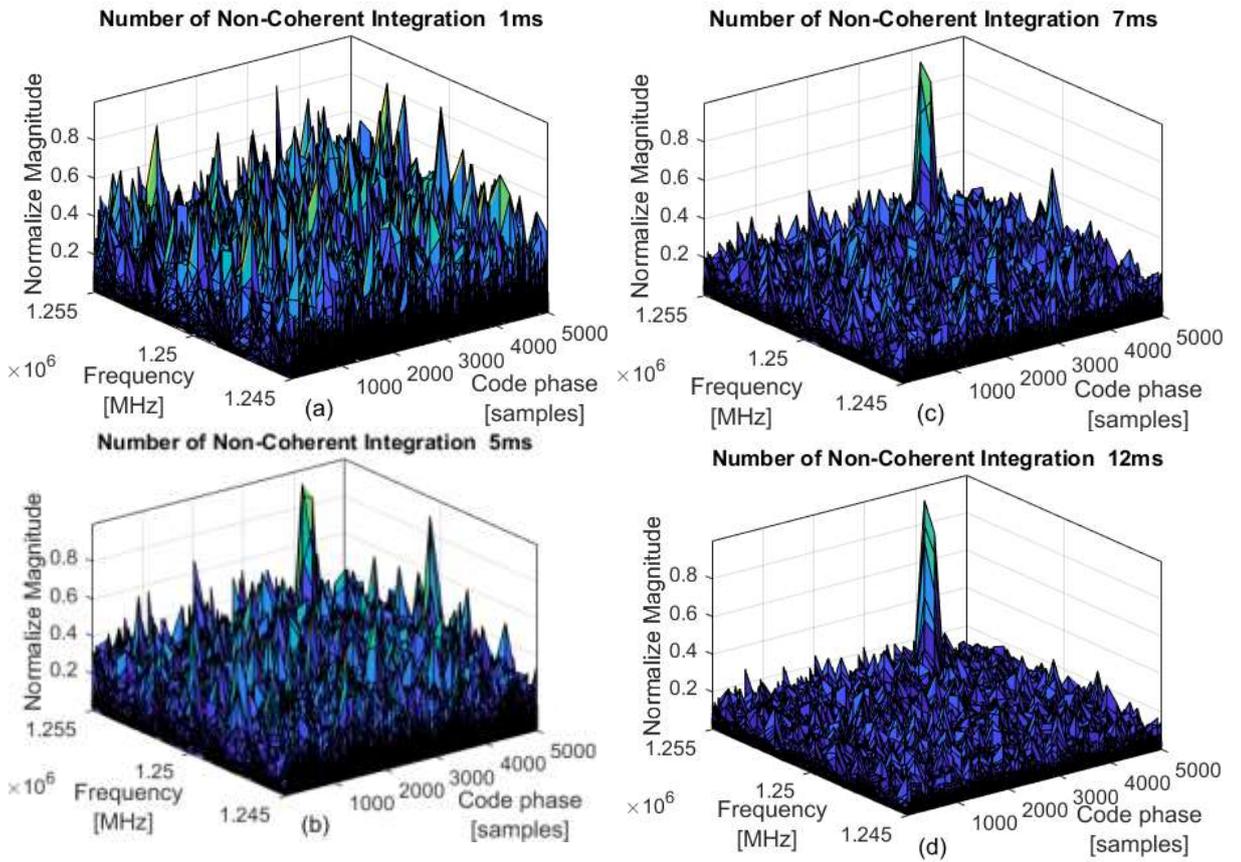


Figure 3.9 Evolution of correlation peak for weak signal acquisition for GPS L1 for 1ms to 12ms

Table 3.2 Simulation Parameters.

Simulation Parameters	
C/N_0	38 dB
Sampling Frequency	5 MHz
Intermediate Frequency	1.25 MHz
Number of Samples	5000

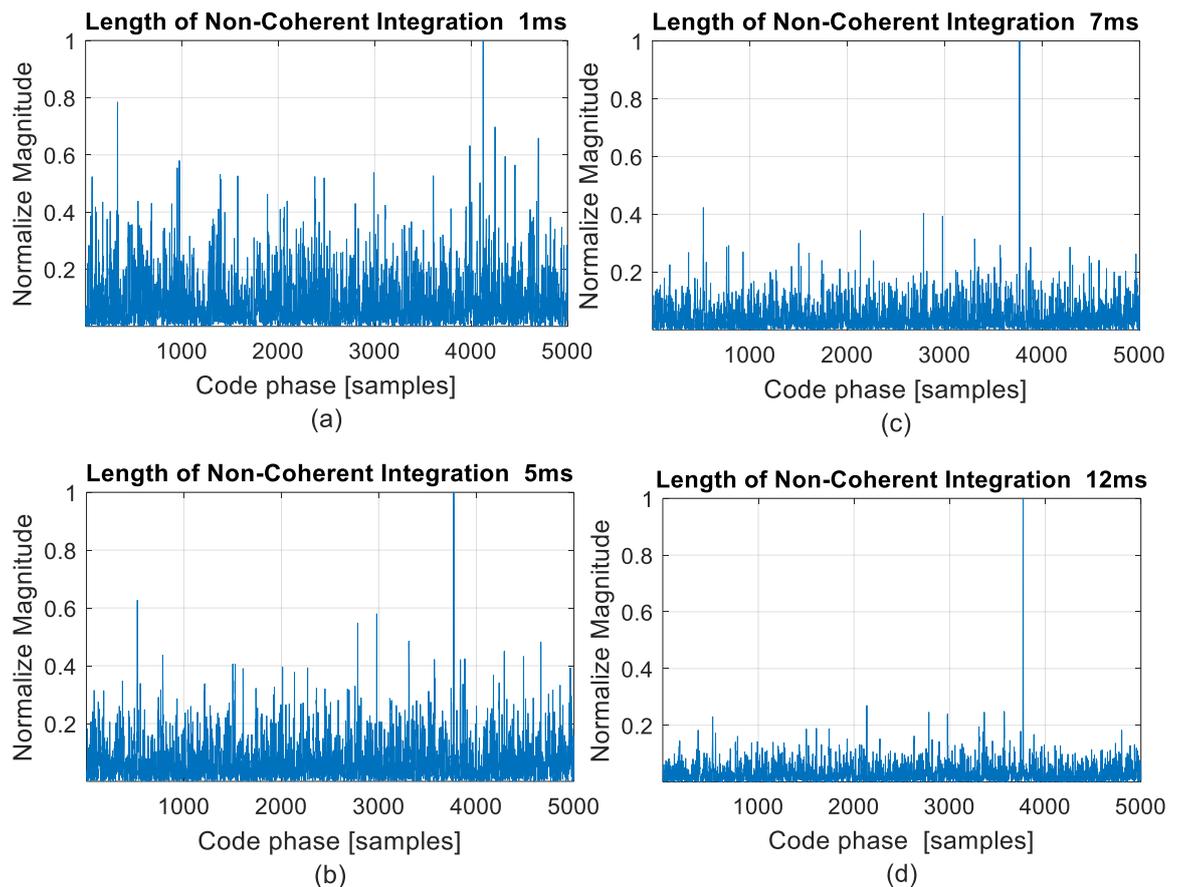


Figure 3.10 Evolution of Code Phase delay as the number of non-coherent integration time increases from 1ms to 12ms

Figure 3.10 shows the four auto-correlation plots between a weak received signal and local C/A code for 1ms, 5ms, 7ms and 12ms. As the non-coherent integration time increases, the auto-correlation peak becomes more visible and prominent. If a signal is weaker, then 1ms of data is not enough to acquire the GPS signal; hence, a weak GNSS signal needs a longer integration time to acquire the weak GNSS signal successfully.

3.4 Performance Evaluation Metrics.

Theoretical assessment of the acquisition search grid is not always viable because it also requires knowledge of the pdf of the decision variable. To run such an assessment, one has to use Monte-Carlo simulation to conduct analysis and evaluation. Such a simulation needed to be run for million iterations to reach conclusive results. Hence, readily realised parameters

are used to evaluate the performance of the acquisition scheme. Equations 3.7 and 3.8 represent peak-to-floor ratios, α_{max} and α_{mean} , where $R_{max\ peak}$ is the maximum of the peak in the acquisition search grid and $R_{noise\ floor}$ is the noise of the acquisition search grid. These metrics are easy to compute and reflect the trend of post-correlation Signal to Noise Ratio (SNR) without carrying out time-consuming simulation and calculation.

$$\alpha_{max} = \frac{|R_{max\ peak}|^2}{\max|R_{noise\ floor}|^2} \quad (3.3)$$

$$\alpha_{mean} = \frac{|R_{max\ peak}|^2}{E[|R_{noise\ floor}|^2]} \quad (3.4)$$

3.5 Modelling of the signals within the GNSS receiver

3.5.1 SNR and C/N_0

SNR is usually expressed in terms of decibels. It refers to the signal and noise power ratio in a given bandwidth.

$$SNR(dB) = S - N \quad (3.5)$$

S is the signal power expressed in units of decibel/milliwatt (dBm) or decibel, and N is the noise power in a given bandwidth in units of dBm . C/N_0 . Whereas C/N_0 is represented in decibel-Hertz ($dB - Hz$) and can be defined as the ratio of the carrier power and the noise power per unit bandwidth [25]. For the GPS L1 signal, one can consider the received signal power as the power of the original unmodulated carrier power (at the point of reception in a receiver) that has been spread by the PRN codes when transmitted from a satellite. We can express C/N_0 as follows:

$$C/N_0(dB - Hz) = C - (N - BW) = C - N_0 = SNR + BW \quad (3.6)$$

Where C is the carrier power in dBW , N is the noise power in dBm or dBW and N_0 is the noise power density in $dBm - Hz$ or $dBW - Hz$. BW is the bandwidth of noise at the receiver end, the equivalent noise bandwidth of the last filter in a receiver's RF front end. Typical values in a GPS L1 C/A code receiver for C/N_0 range in between $35dB-Hz$ to $48dB-Hz$ and receiver front-end bandwidth range from $16MHz$ to $4MHz$, assuming bandwidth is RF front-end is $4MHz$, hence bandwidth in dB equal to $BW = 10 \cdot \log(4,000,000) = 66dB$

$$SNR = C/N_0 - BW \Rightarrow SNR \sim (35-66) \text{ to } (48-66) \Rightarrow SNR \sim -31dB \text{ to } -19dB$$

In order to calculate C/N_0 , then, one must determine the carrier power and noise density at the input to the receiver.

3.5.2 Noise power density

The sources of white noise in a GNSS receiver are usually described by the antenna and the receiver noise temperature [26]. The antenna temperature models the noise entering the antenna from the sky, whereas the receiver noise temperature models the thermal noise due to the motion of electrons within a device such as the ground-based receiver front-end. These noise sources specify the noise density, which can be described as follows [25]:

$$N_0(dBW/Hz) = 10 \log_{10}(kT) \tag{3.7}$$

Where k is the Boltzmann's constant $1.38 \times 10^{-23} J/K$ and T is the typical surrounding temperature of the receiver.

3.5.3 GPS L1 signal at Receiver Input

GPS L1 can be expressed as the following equation at the receiver's input. A_s Is the amplitude of the received signal, C is the C/A, and d is the navigation data.

$$r[n] = A_s C[n - \tau_0] d[n - \tau_0] \cos(2\pi F_L n + \phi_0) \quad (3.8)$$

When code delay and the Doppler shift are accurately recovered, the useful signal can be expressed by equation 3.9, and this equation represents cross ambiguity function (CAF) evaluation.

$$R_y(\tau_0, F_L) = \frac{1}{N} \sum_{n=0}^{N-1} A_s c[n - \tau_0]^2 d[n - \tau_0] \cos(2\pi F_L n + \phi_0) \exp\{-j2\pi F_L n\} \quad (3.9)$$

Where F_L is the $F_{if} + F_D$, intermediate frequency plus Doppler shift.

3.6 Simulation results

Acquisition of the GPS L1 CA code is implemented in MATLAB code, and it was then tested under various C/N_o levels to compare it with the theoretical results. This section presents the graphs obtained after modelling input signal power level, front-end bandwidth and noise level in the receiver front end. The following simulation parameters are used to obtain these graphs:

Table 3.3 Simulation Parameters

Coherent Integration	1 millisecond
Front-end filter bandwidth	4.76 MHz
Sampling Frequency	5MHz
CA code length	5000 samples

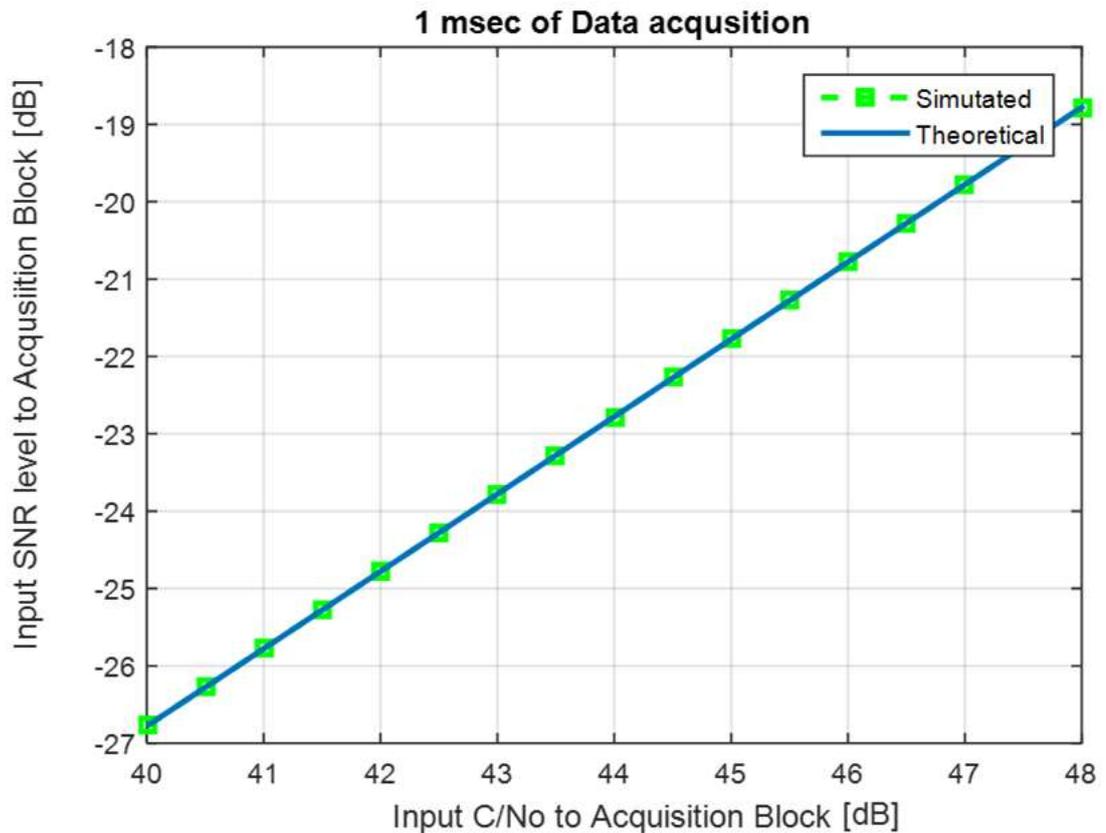


Figure 3.11 Equivalent Input SNR when front end BW=4.76MHz

Input SNR to the receiver front is usually below -19dB and rises above the noise level after acquiring the 1-millisecond signal. Similarly, Figure 3.12 shows these simulation parameters and how an input SNR at the receiver front end and SNR at the acquisition output are related.

This last Figure 3.13 in this section depicts how input C/N_o of GPS L1 signal affect the output SNR at the acquisition unit. This simulation was carried out a few hundred times, and average the result to plot it. The SNR value in for a range of C/N_o (40-43) do not match the theoretical values.

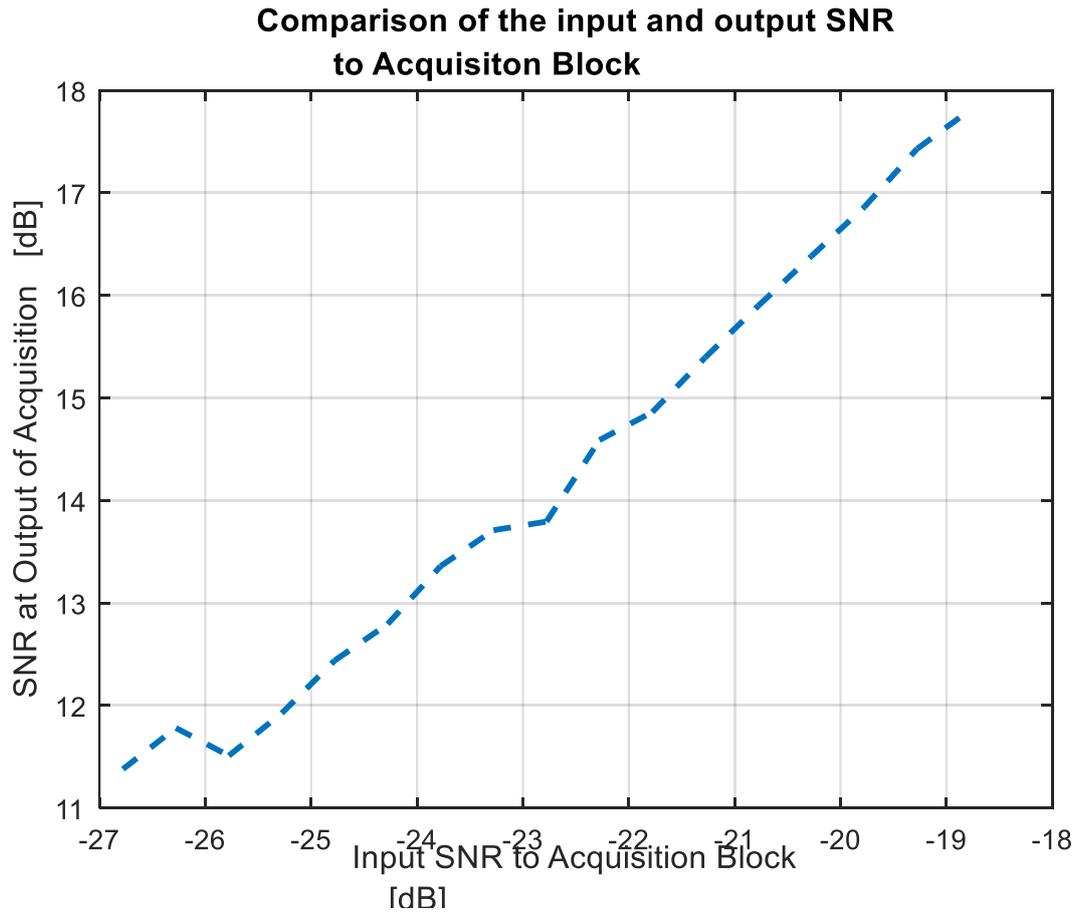


Figure 3.12 Input SNR Vs. Output SNR

The noise level is much higher than the signal; hence, more peaks are generated in the acquisition search grid, leading to wrong calculations. A better method needs to be implemented to model SNR calculation more accurately at the output of the acquisition unit, as described in chapter 4.

The results and simulations in this section were performed to test the functionality of the Simulink-based GPS L1 simulator and perform realistic power level modelling at the receiver front end. Also provide insight into the relationship between SNR, C/N_0 and equivalent noise bandwidth.

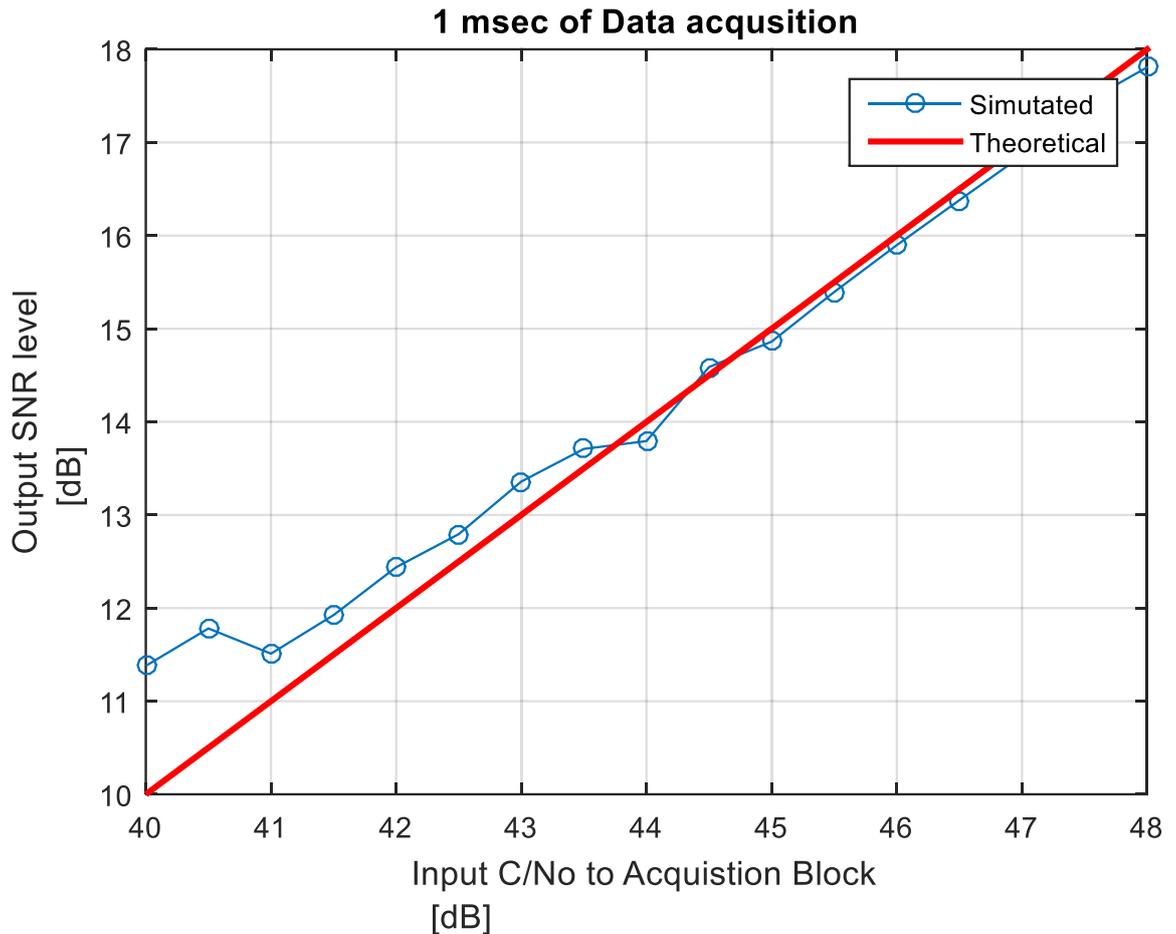


Figure 3.13 C/N_o input Vs. Output SNR

3.7 Conclusions

This chapter provides an outline of the GPS signal acquisition process. A brief overview of the structures of the three most common types of detectors is presented and discussed. The serial search, parallel frequency search and parallel code-phase search algorithms are implemented in MATLAB, and their functionalities were tested. This study led us to conclude that a non-coherent parallel code phase search detector performs better than its counterpart and can acquire a signal in a harsh environment. Unless explicitly stated otherwise, the non-coherent parallel code phase detector is used as the detector/estimator structure for the rest of this thesis.

Furthermore, both coherent and non-coherent detectors are investigated and how they work and are used to increase receivers' sensitivity when the input GPS signal is weak. It was demonstrated that retrieving a weak GPS L1 signal via a non-coherent detector using longer integration/dwell time, the receiver's sensitivity is significantly increased and able to achieve better performance.

This chapter also presented the Front-end GPS L1 signal modelling and demonstrated the receiver end's noise and signal strength level. This chapter also gives insight into the performance evaluation parameters used throughout this research.

Chapter 4

CONTINUOUS WAVE INTERFERENCE AND COMPLEX ADAPTIVE NOTCH FILTERS

The GNSS signal uses a Direct Sequence Spread Spectrum (DS-SS) signal which integrates some level of anti-jamming capability in the structure of the signal itself. However, due to fact, the power level of the GPS signal on the surface of the earth typically ranges from -160dBW to -156dBW for C/A code, which is considered to be a very weak signal, well below the thermal noise floor. Hence, it is easy for the interference signal to overcome the inherent jamming protection of the DS-SS signal. An interference signal in the main lobe of the C/A code makes it difficult for the acquisition module to track the peak of the correlation function.

In this chapter, various kinds of interference signals are modelled, and their effect on GPS signal processing is discussed. Moreover, it reviews different interference mitigation strategies in the literature, and a system-level model is proposed based on multi-stage complex adaptive notch filters.

4.1 Continuous Wave Interference

Jamming and Anti-Jamming of the Global Navigation Satellite System (GNSS) have become a hot research topic due to the fast-evolving GNSS technology and its rapid growth for consumer-based applications. Due to the extremely low power of the GNSS signal received on the surface of the earth, GNSS signal is prone to slight RF interference directed to any GNSS receiver. Its effect on the received GNSS signal is unpredictable and severely degrades the quality of the signal, making it impossible for the receiver to acquire and track it.

Radio Frequency Interference (RFI) can be intentional or unintentional, and its influence on the GNSS system is multi-dimensional. So far, the framework of this research is limited to the analysis of interference caused by the communication system, Continuous Wave Interference (CWI) and chirp-type interference from various digital devices (Jammers).

It is essential to differentiate between types of interference, either wide-band or narrow interference. Interference can be wide-band for the C/A code, but if the same interference is compared with a long P(Y) code is referred to as a narrow-band interference. The wide or narrow band merely depends on the GPS signal's bandwidth. C/A and P(Y) have respective power spectral densities of 2.046 MHz and 20.46 MHz. Intentional jamming is categorized as spoofing and jamming. Spoofing is the transmission of a fake version of a GNSS signal with more strength to deceive the receiver; hence, the GNSS receiver report incorrect position and timing data. On the other hand, jammers use high-power fake data modulated on the carrier wave within the bandwidth of the GNSS signal. The researchers conducted an extensive analysis of jammer signals. From its analysis, any jamming signal can be modelled by linear frequency modulation, meaning that instantaneous frequency sweeps a range of

frequencies in a very small duration, usually microseconds targeting the entire GNSS band to jam signal [48].

As described in the initial chapter, the GNSS signal is transmitted with Direct Sequence Spread Spectrum (DSSS) modulation. DSSS modulation strengthens the GNSS signal against unintentional interference, but its robustness degrades if the power of the interference is high and present for a long time. Any interference signal model as pure sinusoidal is classified as Continuous Wave Interference. These kinds of signals are present in almost proximity of any GNSS device. CWI can be emitted from a wide range of electronic devices that compromise the local oscillator to transmit or receive the data. All these disturbing signals with comparatively high power can affect the performance of the GNSS receivers in terms of the acquisition and tracking of GNSS signals, reducing the precision of the GNSS receivers.

4.2 CWI Modelling in MATLAB

Jammers can radiate a variety of different types of interferences signal. The main focus of this research is on continuous wave interference signals such as the following interference signal:

- Continuous Wave Narrowband Interference
- Complex Sinusoidal Wave Interference
- Real Chirp-Type Interference (Single or Multi Saw-tooth function)
- Complex Chirp-Type Wave Interference (Single or Multi Saw-tooth function)

Above mentioned interference is the most common type of interference emitted by the commercially available jammers in the market. These types of jammers can be easily plugged into the cigarette lighter of the car and generate a variety of narrowband and wideband interference signals.

4.2.1 Continuous Wave Narrowband Interference Real and Complex Sinusoidal

Sections 4.2.1 and 4.2.2 of this report demonstrate the modelling of various interference signals in MATLAB. Figure 4.1 shows the simulation of the instantaneous frequency of narrowband CWI with fixed frequency and can be represented by the following equation.

$$j_{int}(t) = \sqrt{2P_i} \cos(2\pi f_i n T_s + \theta_i) \quad \text{Where amplitude is, } A_i = \sqrt{2P_i} \quad (4.1)$$

Hence,

P_i , is the power of the narrow band CWI

f_i , is the interference frequency.

θ_i , is the initial random phase of the interference

T_s , is the sampling frequency

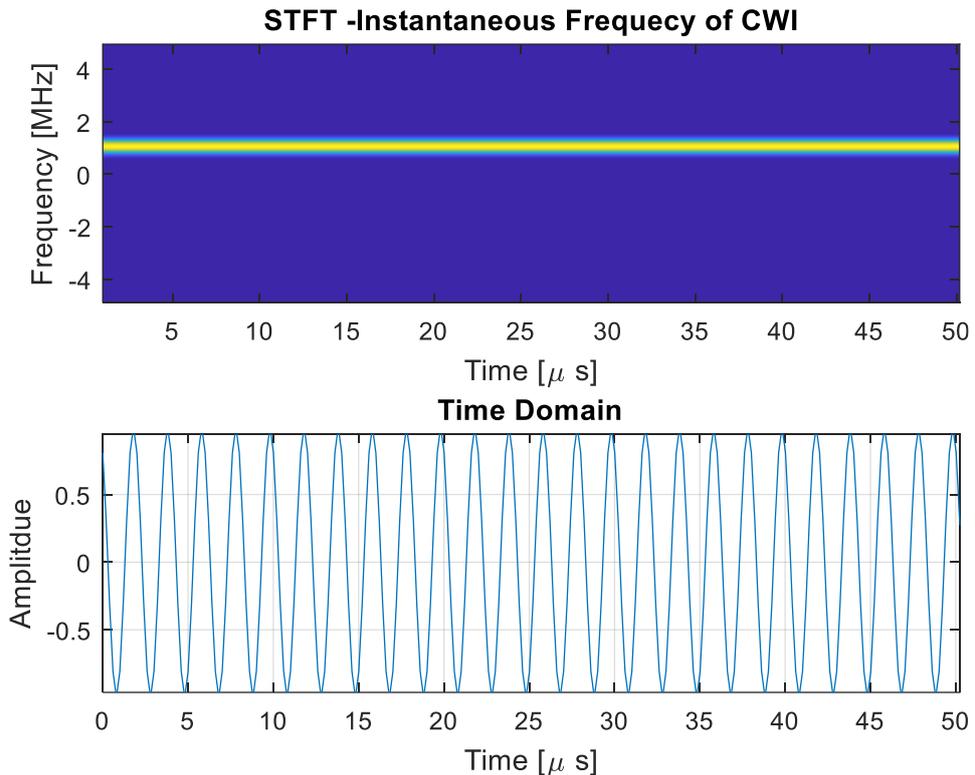


Figure 4.1 Modelling of continuous wave interference

Such a jammer type consists of a voltage control oscillator (VCO) that generates a single sinusoidal tone with constant frequency and costs less than £10 on the market. On the other

hand, some jammers can even generate a sophisticated jamming signal within the frequency spectrum of GNSS signals. Since narrowband and chirp-type of jammers are readily available at low cost, hence pose severe threats to GNSS signals functionality.

Similarly, a complex continuous wave interference can be modelled. A complex sinusoidal signal has both real and imaginary components. It can be expressed as follows.

$$j_{int}[n] = A_i \exp\{j2\pi n f_i T_s + \theta_i\}, \text{ where power is equal to } P_i = \sqrt{A_i} \quad (4.2)$$

Further, equation 4.2 can be decomposed to equation 4.3 using Euler identity. Figure 4.3 shows the 3D plot of the complex sinusoidal versus time and its projections onto the real and imaginary coordinate planes.

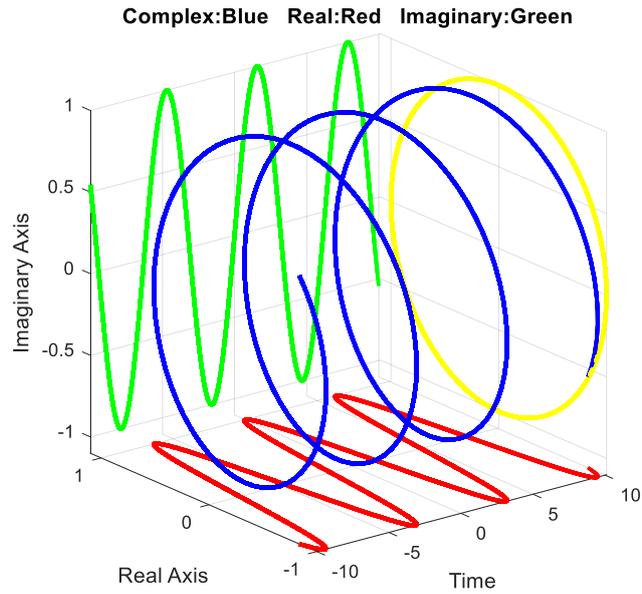


Figure 4.2 Complex Sinusoidal and its projections

$$j_{int}[n] = A_i \cos(2\pi f_i n T_s + \theta_i) + j A_i \sin(2\pi f_i n T_s + \theta_i) \quad (4.3)$$

4.2.2 Real and Complex Chirp-type Interference

Chirp-type signals are the most popular type of jamming signal used by civil jammers. They turn out to be very effective and completely blank out the GNSS receiver, hence unable to acquire the useful signal. Chirp-type interference consists of a sinusoidal signal whose frequency is repetitively swept across a specific bandwidth range. Real and complex chirp signals can be mathematically expressed as equations 4.4 and 4.5.

$$J_{Real-chirp}[n] = \sqrt{2P_i} \cos(2\pi f_{chirp}(t)nT_s + \theta_i) \quad (4.4)$$

$$J_{Complex-chirp}[n] = A_i \exp\{j2\pi n f_{chirp}(t)nT_s + \theta_i\} \quad (4.5)$$

Where $f_{chirp}(t)$ is the instantaneous frequency of the chirp signal at time t and mainly a linear saw-tooth function of the time, as shown in Figure 4.3. This kind of interference signal can be classified as a wideband interference because it sweeps almost the entire GPS L1 frequency band many times during a single coherent acquisition of a millisecond of the GPS L1 signal.

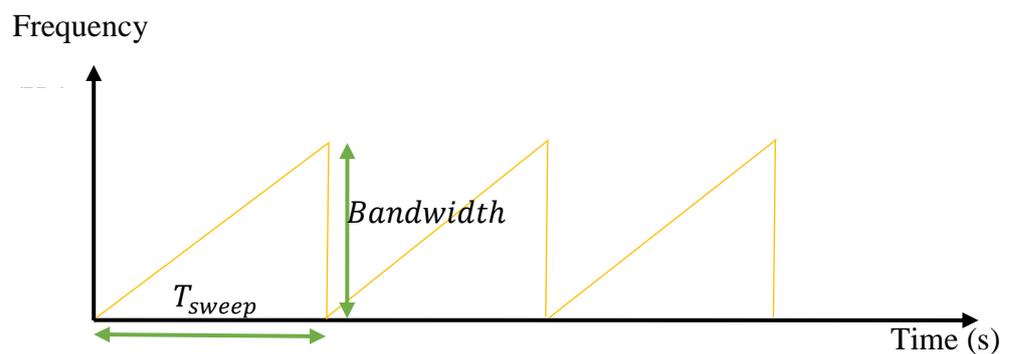


Figure 4.3 Time-Frequency representation of a linear saw-tooth chirp signal

Figure 4.4 and 4.5 show the time-frequency evolution of complex and real chirp interference signal with multiple saw-tooth $f_{chirp}(t)$ function modelled in MATLAB.

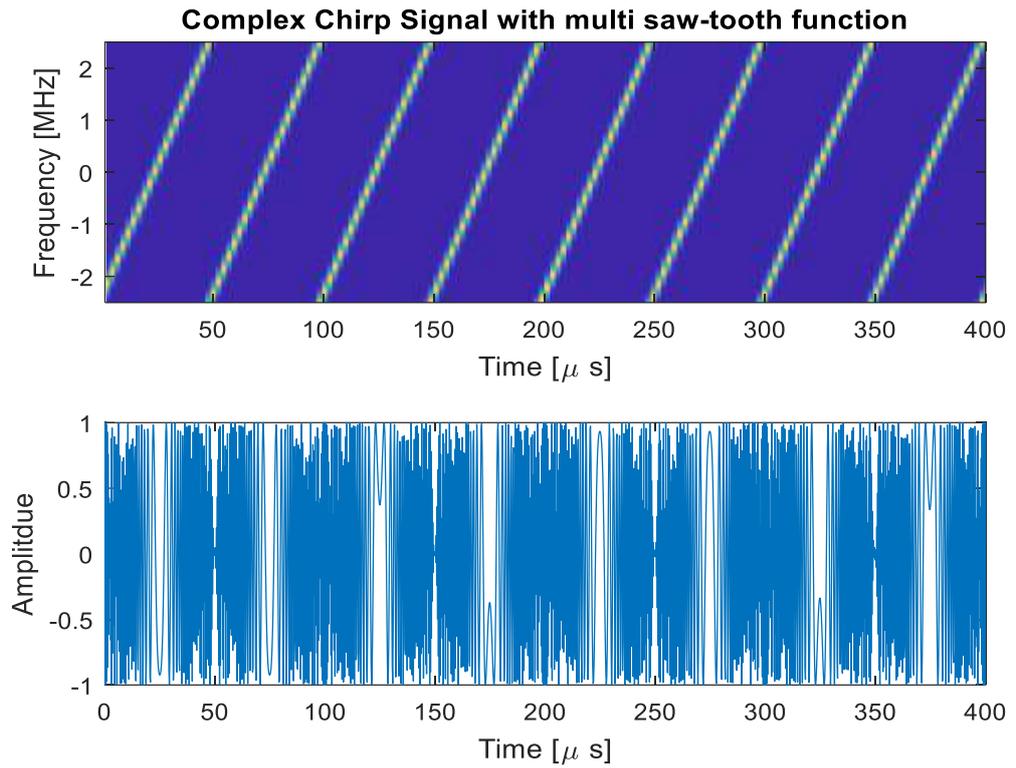


Figure 4.4 Complex chirp signal time-frequency representation of instantaneous frequency as a function of time

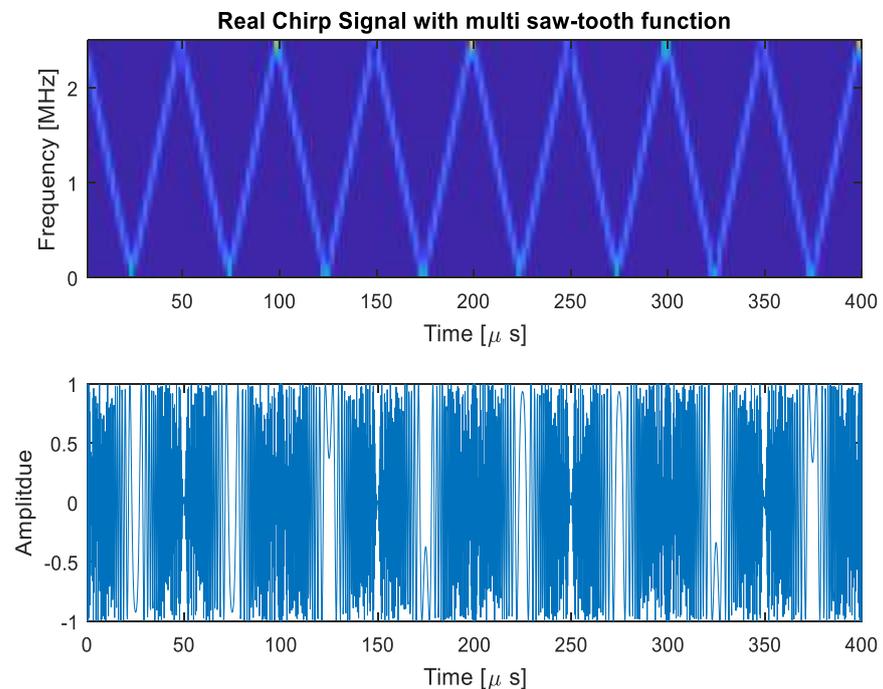


Figure 4.5 Real chirp signal time-frequency representation of instantaneous frequency as a function of time

4.2.3 Jamming to Noise Ratio (JNR)

In order to evaluate the performance of any anti-jamming algorithm, it is important to determine the jamming-to-noise ratio (JNR), which is defined as [49]

$$JNR = 10 \log_{10} \left(\frac{P_i}{P_n} \right) \quad (4.6)$$

P_i and P_n are the respective power of the jamming signal and the noise. The power of the interference with respect to noise variance is defined by the Jammer to Noise ratio, which can be mathematically represented by equation 4.7.

$$JNR = \frac{J}{N} = \frac{\frac{1}{2}A_i^2}{\sigma_n^2} = \frac{A_i^2}{2B_{IF}N_0} = \frac{A_i^2}{F_S N_0} \quad (4.7)$$

Where A_i the amplitude of narrow is band CWI, N_0 is the noise per unit bandwidth, F_S is the sampling frequency and B_{IF} is the bandwidth of the front-end filter. Where σ_n^2 is the variance of the noise and equal to:

$$\sigma_n^2 = B_{IF}N_0 \quad (4.8)$$

This section presents the modelling of narrowband CWI through the acquisition chain and its implications on the acquisition scheme's Cross-Ambiguity Function (CAF). The focus of this chapter is to consider the impact of CWI on CAF with coherent integration of one period of code phase, 1ms of data. When fixed frequency CWI is present as interference, it can be expressed by following Equation 4.9

$$j_{int}(n) = \sqrt{2P_i} \cos(2\pi f_i n t_s + \theta_i) \quad \text{Where amplitude is, } A_i = \sqrt{2P_i} \quad (4.9)$$

Hence,

P_i is the power of the narrow band CWI

f_i is the frequency interference.

θ_i is the initial random phase of the interference

JNR parameter is used to analyse the performance of 1st, 2nd and 3rd order complex notch filters with different Jamming to Noise ratios applied to the GPS L1 signal generator model developed in chapter 2.

Figure 4.6 illustrates the PSDs of both GPS L1 and CWI. PSD of CWI is simply two lines within the main lobe of the GPS L1 signal. Interference is considered NW-CWI for GNSS signal if the frequency is below 1MHz [46].

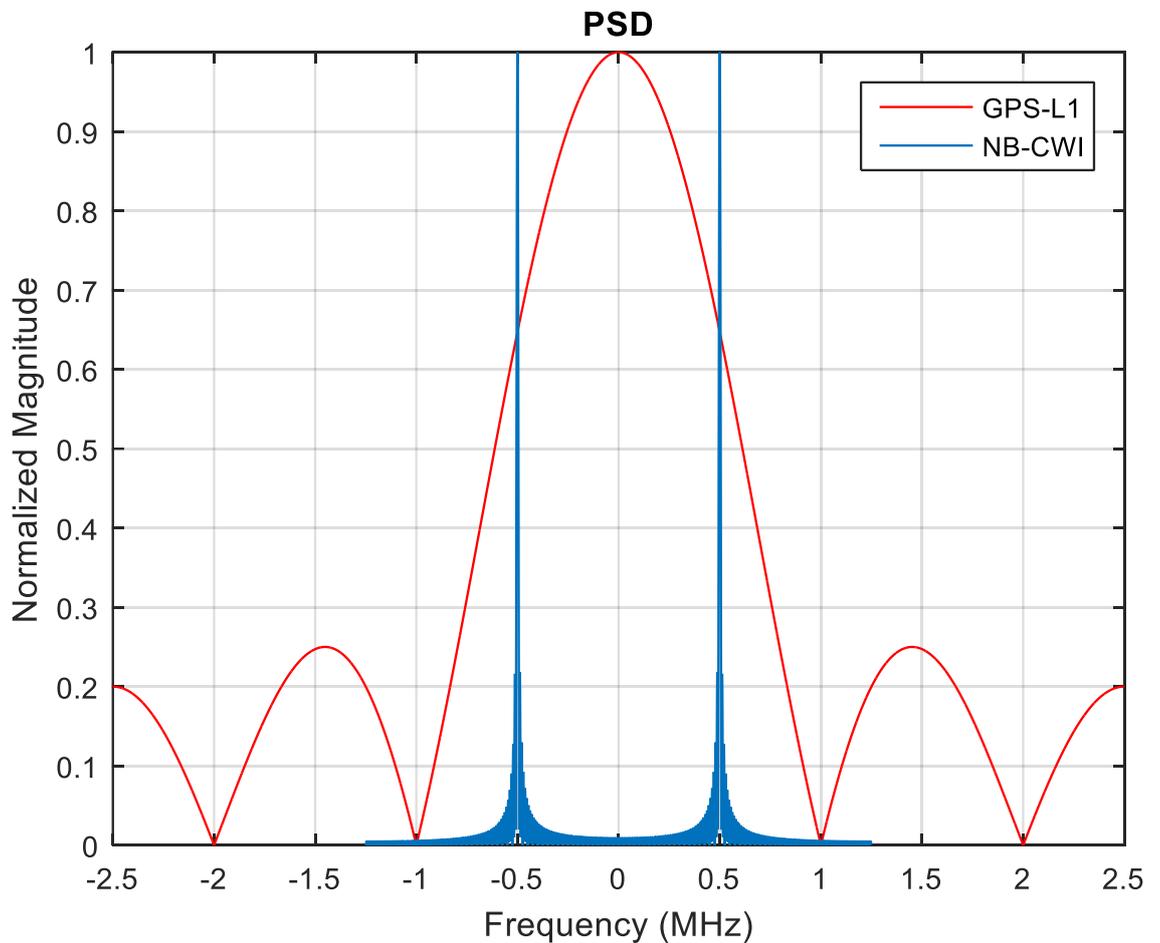


Figure 4.6 Illustration of Normalized PSD of CWI and C/A code

4.3 Detection and Mitigation

There has been extensive literature on the detection and mitigation of CWI, such as [49-53]. Before applying any interference cancellation techniques, it is required to detect the interference and its types. Interference can be of the following four types:

- Narrow-band or Wideband CWI
- Pulsed wave interference
- Narrow-band Gaussian interference
- Time-varying interference

The detection scheme used before correlation uses automatic gain control (AGC) prior to analogue to digital conversion (ADC). The interference detection techniques based on AGC depend on the fact that the AGC is driven by the ambient noise or the interference signal rather than the GNSS signal power. Likewise, the variation of AGC gain can be used to detect interference [53]. If the interference-to-noise ratio can be calculated accurately, as shown in [53], this information can be utilized to adaptively configure the tracking and acquisition units. But the performance of the AGC detection scheme degrades when the interference power is comparably closer to the input noise [53]. Many other detection methods are reported in the literature [54], but their performance is limited by operating location, receiver front-end effects and receiver processing time constraints.

Interference mitigation techniques can be grouped into four main categories, time domain, frequency domain, time-frequency domain and spatial-time domain [54]. Then further classified as pre-correlation and post-correlation mitigation techniques. Techniques such as pulse blanking, adaptive notch filter and null steering are employed before the signal is fed into the acquisition block. And on the other hand, techniques such as vector tracking loops

and expanded adaptive code loops are used post-correlation. The post-correlation techniques can be employed within the acquisition and tracking loop of the GNSS receiver. Post-correlation mitigation techniques detect interference based on estimated C/N_o after the acquisition unit. If the estimated C/N_o have small variance compared to the mean of the estimated C/N_o then interference is considered to be absent; otherwise, if the variance is significant around the mean of C/N_o , then it means interference is present.

In the [51], the authors have extensively analysed different sources of interference and determined the distance between the GNSS satellite signal and the interference. As mentioned so far, the presence of interference in the GNSS frequencies band is highly undesirable because it impairs the performance of the acquisition and tracking unit of the receiver. Various kinds of interference cancellation techniques for GNSS systems were cited and analysed. These techniques have pros and cons; some methods are very effective but highly complex to realize on hardware and with extra cost. On the other hand, the simplest and most cost-effective techniques do not meet modern-day jamming or interference signal devices. Hence they underperform due to the lack of ability to respond promptly to incoming interferences and keep tracking them. Detection and mitigation analysis is made using fixed and complex adaptive notch filters in this research work. It holds a balance between implementation, cost, interference extraction and preserving the rest of the GNSS signal compared to other counterpart techniques when mitigating complex CWI or chirp-based interference [55].

4.4 Digital Notch Filters

This section overviews different classes, types of notch filters, and a brief literature review. Notch filters and adaptive notch filters have been around for many years. Over the years, the structural realization and implementation of a finite response filter (FIR) have been

developed into well-defined and efficient algorithms. Nevertheless, implementing the digital filter as infinite impulse response (IIR) structure is mind-boggling and not fully understood. Like classic leakage integration examples in old-school DSP books, the IIR structure applies feedback from its output, reducing the filter's length and providing an excellent response per computational burden compared to FIR digital filter. Hence, this feedback tends to make the IIR filter unstable if the IIR filter is not properly constrained. On the other hand, FIR filters are incredibly stable and more reliable in the sense of stability.

Similarly, suppose one has to design a filter with a steep cut-off frequency. In that case, FIR filters require far more taps to meet the desired filter specification and response than designing the same filter with an IIR structure. A notch filter must have a very thin stop band or notch region. Therefore, notch filter design and implementation as an IIR structure give an edge over its FIR structures.

- Less computational
- Low cost
- Less number of parameters to adapt (in case of adaptive notch filter)
- Minimal design

An ideal notch filter response can be modelled as in equation 4.10. With a bandwidth of zero and gain of one in the pass-band region [56]

$$H_{NF}(z) = \begin{cases} 0, & f = f_i \\ 1, & \text{elsewhere} \end{cases} \quad (4.10)$$

A notch filter passes all the frequencies except those in the stopband region (the notch). Here in this research, IIR notch filters are the centre of gravity, and IIR notch filters can be implemented in different forms such as lattice structure [56], constrained and unconstrained zeros [40], and bilinear-second-order IIR notch filters [58].

4.4.1 Why IIR complex adaptive notch filters?

This research aims to develop interference mitigation algorithms for GNSS signals for various types of interference signals, such as CWI, single and multiple complex sinusoidal, and complex chirps.

Extensive literature is available on the adaptive notch filter. FIR filter's structure, implementation, and realisation in terms of hardware have matured over the years. However, IIR digital filter's structure is still the centre of research for DSP engineering as they are not fully understood. FIR digital filter has all feedforward coefficients, and it is unconditionally stable. On the other hand, IIR filter structure feedbacks its output into the structure, and this feedback tends to become unstable unless some precautions are taken. However, this feedback reduces the IIR filter's length and produces a more specific response per computation.

Numerous adaptive notch filter structures are reported in the literature [59-62] for different applications. However, very little work has been published on complex adaptive notch filters, most recently wheeler [63], famous regalia's CANF structure [64] and numishra's [62] direct form structure. This motivated and provided groundwork for developing aspects of complex adaptive notch filters capable of tracking CWI, single and multiple complex sinusoidal and complex chirp types of interference, with GNSS receiver-specific applications. The complex adaptive notch filter is a field still unexplored. More work needs to be done to realize them by creating a new type of structure, adapting the notch bandwidth parameter in the structure, improving the performance by implementing stochastic search methods and finally, considering tracking complex chirp-like signals. This makes complex adaptive filters attractive for complex sinusoidal and chirp-like interference.

Why would CANFs be used? This question can be asked; the answer is that CANF employs simple structures that can be implemented efficiently on the hardware providing high performance.

- The designs and structure in the following [59-64] are based on simple adaptation algorithms and have less computational complexity. Thus it is canonic in the number of multipliers and delay elements
- Structures can be easily replicated to facilitate the tracking of multiple interference signals
- As this design is based on a structurally ARMA prototype [64] and an all-pass prototype and this solution provides flexibility from the implementation point of view and gains enhancement; therefore, the signal cannot be detrimentally changed, even if there is an error in the algorithm
- ANFs can mitigate wideband interference and provide Signal-to-Noise Ratio (SNR) performance enhancement, hence improving the performance.
- This technology provides numerically robust solutions

4.4.2 First-order Notch Filter- A Brief Demo

First-order complex notch filter transfer can describe by equation 4.5. Considering expression 4.5 z_0 is the location of zero on the z -plane, and it is required to mitigate the interference and $k_\alpha z_0$ is the pole location and value of k_α have to be between 0 and 1 to guarantee stability. k_α Parameter can be further changed to different values to vary the notch bandwidth. k_α Is also called the pole contraction factor. It controls the bandwidth of the notch and interference mitigation level. Implementation of a notch filter can be further divided into two categories fixed notch filter and adaptive notch filter. Preliminary analysis is made using

fixed notch filters where interference frequency is known before the interference mitigation.

The following transfer function represents the first-order notch filter.

$$H_{NF}(z) = \frac{1 - z_0 z^{-1}}{1 - k_\alpha z_0 z^{-1}} \quad (4.11)$$

- z_0 is the complex zero
- k_α is the parameter to adjust the notch bandwidth

As the first-order notch filter is complex and is designed to filter out complex sinusoidal signal interference, which can be modelled as

$$j_{int}[n] = A_i \exp\{j2\pi n f_i T_s + \theta_i\} \quad (4.12)$$

Figure 4.7 shows the frequency response of the complex notch filter with different values of k_α . For smaller the value of k_α (0.60) the notch width is wider, as shown by the blue dash line when the value of k_α is changed to 0.90, the notch width gets narrower. The narrow notch width is more desirable because it limits its effect on useful signals and preserves the information. If the wider width is used to remove CWI from the signal, it excises useful information and degrades the output SNR and performance of the system.

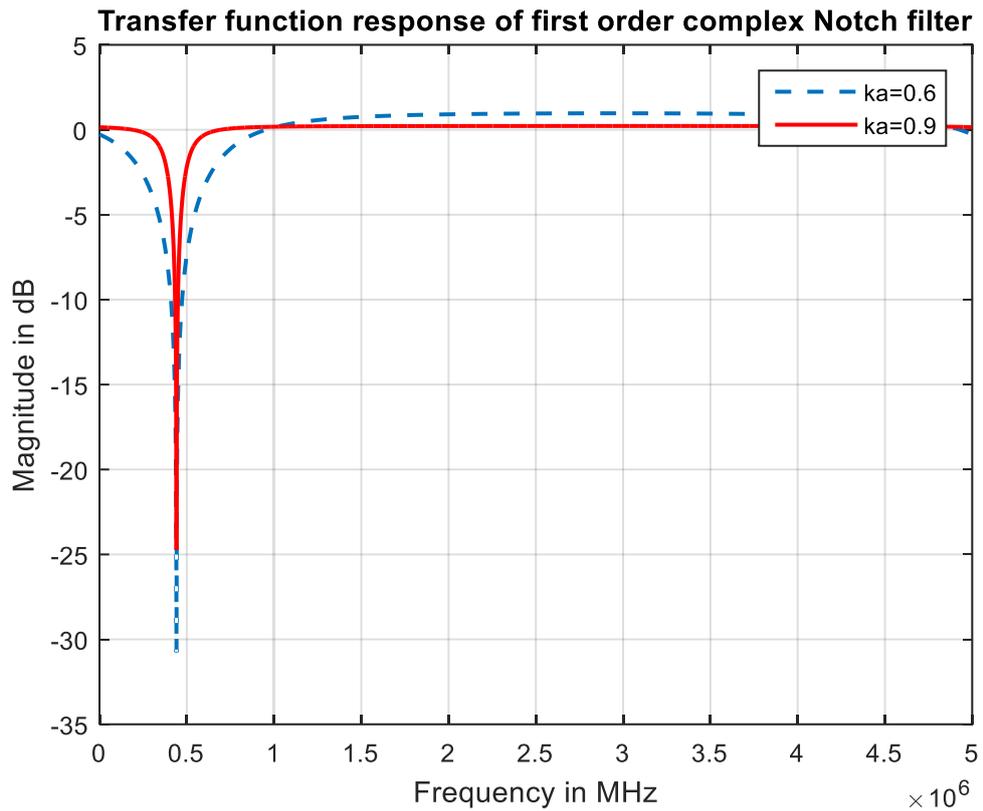


Figure 4.7 Response of Notch Filter different values k_{α}

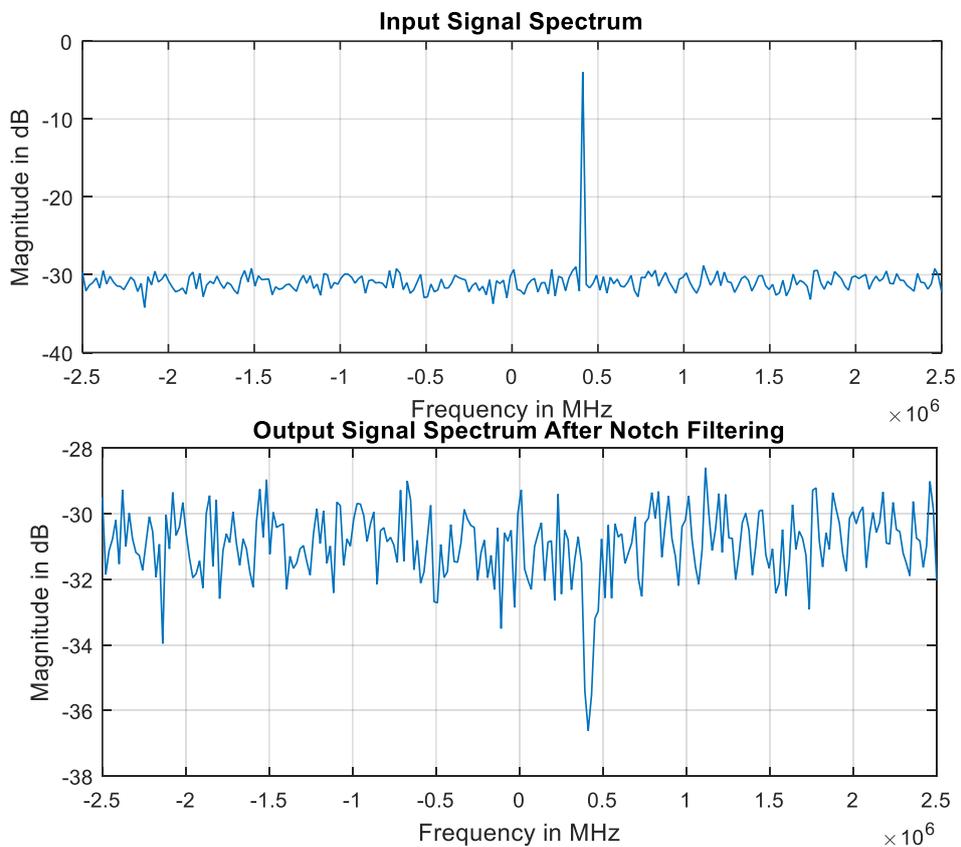


Figure 4.8 Complex sinusoidal signal and its removal by the complex notch filter

IIR Notch filter has a drawback, as it has a non-linear phase it affects the cross-ambiguity function (CAF) of the acquisition search grid. The cross-ambiguity function (CAF) is evaluated in the acquisition module of the receiver to detect the satellite and estimate its parameters, such as Doppler shift and code phase delay. Hence following few simulations shows how CAF get distorted by Notch filters. This distortion can lead to the wrong estimation of the Doppler shift and code phase delay of the C/A code. The correlation between incoming GPS L1 signal and local C/A code is a 2D correlation-based process, and its output function is called the ambiguity function (AF). The AF is calculated for each PRN code across all possible combinations of replica code offset and range of Doppler shift commonly between -5 kHz to 5 kHz. In order to acquire an available satellite signal, the maximum absolute value of the resulting AF is compared with the predefined threshold. As mentioned earlier, the notch filter introduces distortion and de-shapes the CAF, which is no

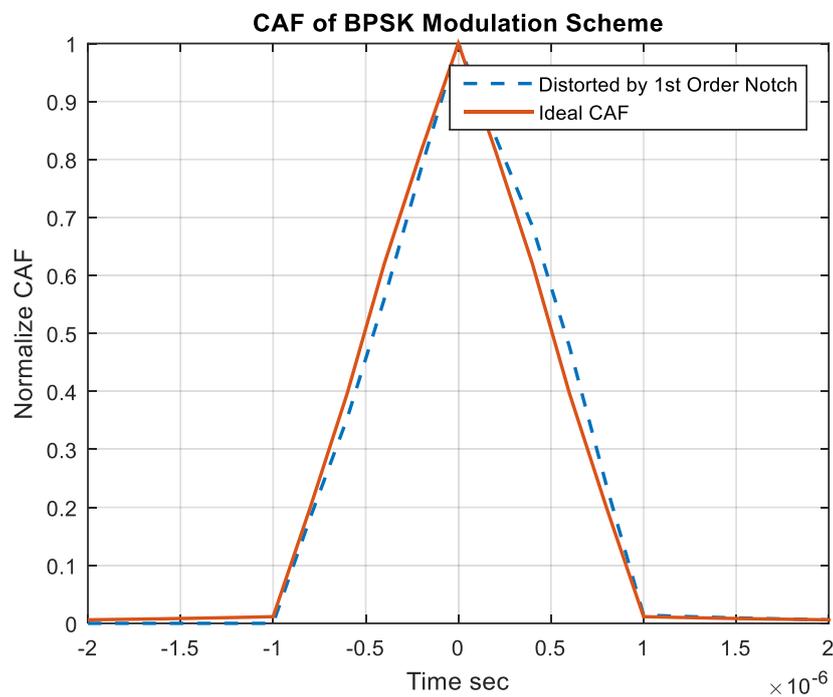


Figure 4.9 Distortion of CAF after notch filtering

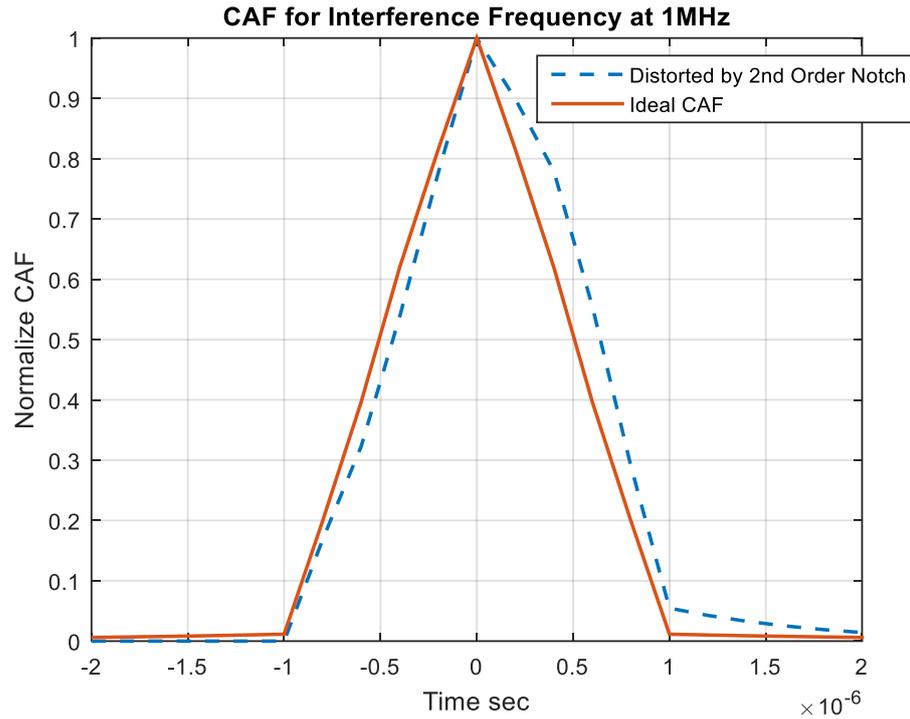


Figure 4.10 CAF distortion by second-order filter

more symmetric, as illustrated in Figures 4.9 and 4.10. Since CAF is no more symmetric, it introduces bias in the code phase delay estimate. This distortion introduced by the notch filter depends on the notch's central frequency and bandwidth. If the centre frequency of the notch filter lies on the main lobe of the GPS L1 signal, it excises a significant chunk of the GPS signal leading to high distortion in CAF function. Incorrect code phase delay estimation produces an error in the tracking loop of the GNSS receiver. Hence post, notch filtering requires extra processing to compensate for bias [71] and minimize these errors.

4.5 Performance Analysis Metrics

In order to analyse and evaluate the performance of the interference mitigation algorithms following metrics were utilized:

1. *SNR of Acquisition Search Grid*: The acquisition of all GNSS systems is based on evaluating and processing the cross-ambiguity function. Mathematically represented as equation 4.13. This is discussed in detail in chapters 2 and 3 of the thesis. It is illustrated in chapters 2 and 3 as plots of the acquisition search grid

$$S(\tau, f_D) = \frac{1}{N} \sum_{n=0}^{N-1} r[n]c[n - \tau]e^{-j2\pi f_D n} \quad (4.13)$$

- $r[n]$ is the received satellite signal
- $c[n - \tau]$ is the local replica of the C/A
- τ and f_D are the code phase delay and the Doppler frequency

$$r[n] = s[n] + j_{int}[n] + \eta[n] \quad (4.14)$$

- $s[n]$ is the GPS L1 signal
- $j_{int}[n]$ is the interference signal with unknown amplitude and frequency
- $\eta[n]$ is white Gaussian noise with zero mean and variance σ_n^2 .

In an ideal scenario, after evaluation of CAF, it should present with a well-defined sharp peak, and the values of the peak corresponding to delay τ and Doppler frequency f_D of the satellite in space. Indeed, due to noise variation and GPS L1 signal degrade further when it travels through space and the earth's atmosphere. Hence the CAF peak is not prominent, or a false peak arises due to an interference signal, then it requires further processing to extract the peaks and their values (τ and f_D), such as non-coherent longer integration as specified in chapter 3. One of the basic and still power metrics that can evaluate the strength of the

signal is the SNR at the output of the acquisition block. SNR in the CAF can be calculated [65] by equation 4.15.

$$SNR(dB) = 10 \log \left[\frac{\max[S(\tau, f_D)] - \text{mean}[S(\tau, f_D)]}{s.t.d[S(\tau, f_D)]} \right] \quad (4.15)$$

2. *Acquisition Figure of Merit ρ_{max}* : In chapter 3 peak-to-floor ratio was introduced, where ρ_{max} is somewhat similar. It is employed in all GNSS receivers' acquisition systems to determine whether the correct peak is evident or not. As the useful signal degrades because of the interference signal, it becomes hard to search or pinpoint the correct peak due to the introduction of many fake peaks while evaluating the CAF. The acquisition Figure of merit is computed simply by taking the ratio of the maximum peak value of CAF $\max\{S(\tau, f_D)\}$ and the second largest peak value. In many literatures [52] it assumed and stated as long as the ratio between two peaks is above two, the value of $\max\{S(\tau, f_D)\}$ is considered as a correct peak.

$$Peak_{ratio} = \frac{|R_{\max peak}|}{\text{secondmaxpeak}(|R_{noise floor}|)} > 2 \quad (4.16)$$

3. *SINR-Improvement after Notch Filtering*: This criterion is used to evaluate the performance of the proposed interference mitigation algorithm. It actually represents the improvement of signal-to-interference-plus-noise ratio (SINR) and is calculated [66] using equation 4.17.

$$SINR_{improvement} = 10 \log \left[\frac{E(|r[n] - s[n]|)^2}{E(|y[n] - s[n]|)^2} \right] \quad (4.17)$$

Where $r[n]$ is the filter(fixed notch or ANF) input, $s[n]$ is the actual baseband GPS L1 signal, $y[n]$ represents the output of the interference mitigation filter and $r[n]$ is received signal with additive Gaussian noise and interference signal.

4. *Mean Square Error*: The usefulness or quality of the retrieved signal is determined by its mean square error [52] and provides insight into the effectiveness of the interference mitigation technique for removing interference from the useful signal. In equation 4.18 $y[n]$ is the output of the notch filter and $s[n]$ is the baseband GPS L1 signal.

$$\text{Mean Square Error} = \frac{1}{N} \sum_{n=1}^N (y[n] - s[n])^2 \quad (4.18)$$

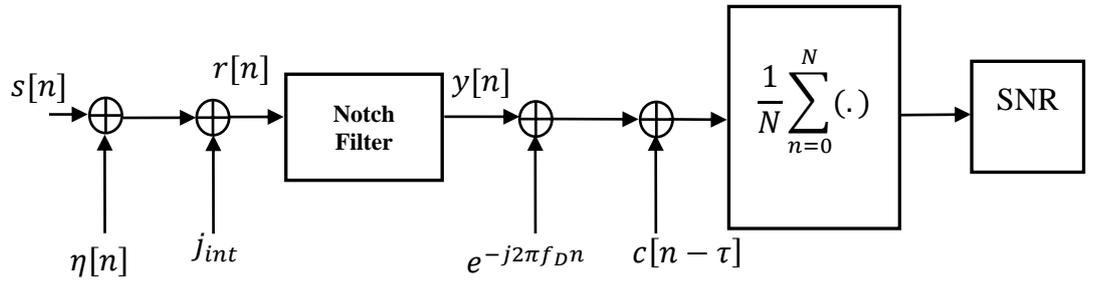


Figure 4.11 System block representation of the calculation of output SNR

Figure 4.12 shows the performance of the first-order complex notch filter with different input interference (CWI with fixed frequency) power levels with JNR from 5dB to 60dB. Each simulated data is run 100 times and with noise power of 110dBm and input, C/N_o was kept constant at 39dB, considered a weak GPS L1 signal. As long as the output SNR of evaluated CAF is above 10dB, it is considered that the received signal is acquirable by the acquisition module [65]. If the output SNR is between 10dB to 11dB, the received signal requires further processing, in this case, longer integration time in the acquisition module as mentioned in chapter 3; in this particular simulation, 4ms of non-coherent acquisition is carried out.

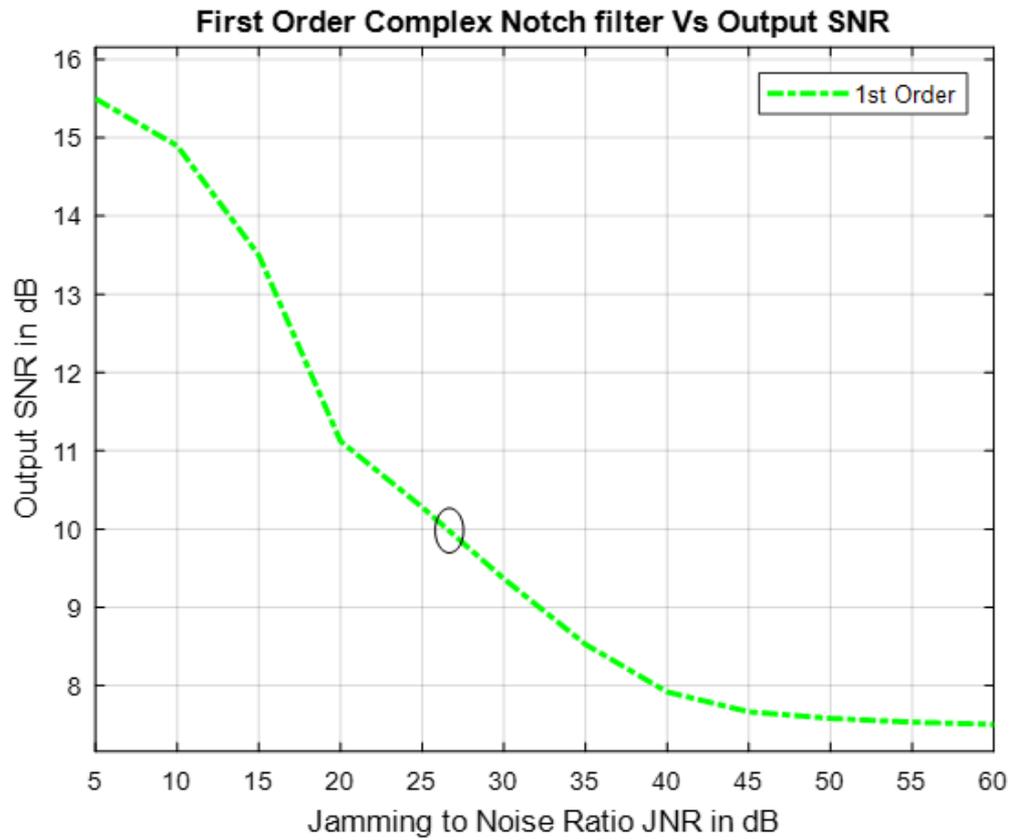


Figure 4.12 Output SNR in dB vs JNR for 1st Order Complex filter with $k_{\alpha}=0.88$ and C/N_o 39dB

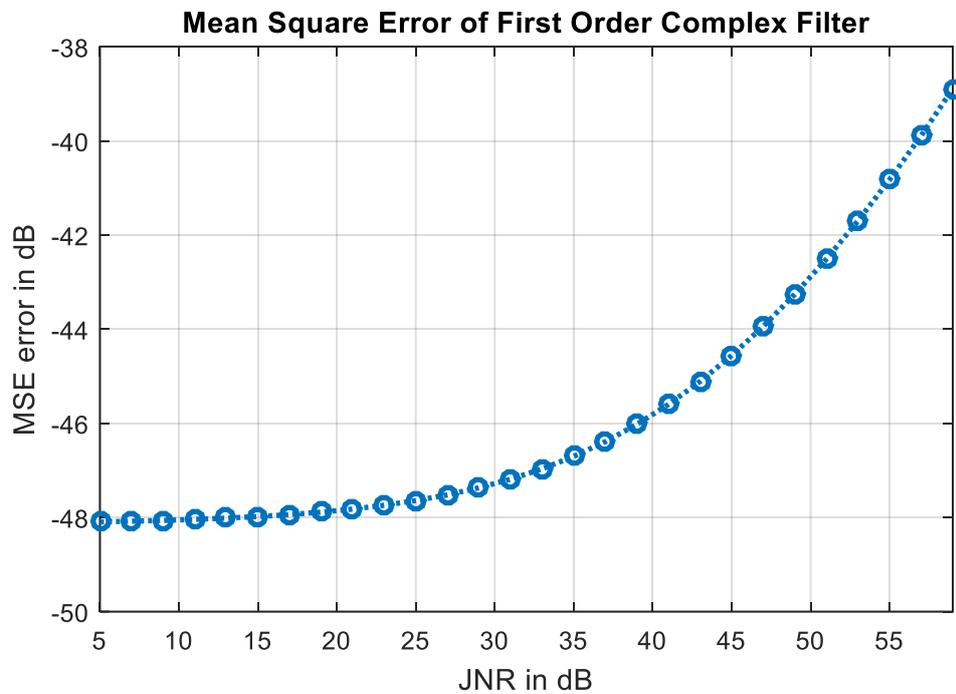


Figure 4.13 MSE in dB at the output of 1st order CNF Vs. JNR

It can be observed from Figure 4.12 that as the JNR increases, the output SNR decreases and degrades the performance of the first-order notch filter. Under this condition, the first-order CANF filter performs well and can attenuate the interference signal as long as the input JNR is below 26dB, shown in Figure 4.12 as a circle marked on the plot.

Figure 4.13 shows the simulation results of the mean square error (MSE) at the output of the CANF. This plot shows that MSE increases rapidly (exponentially) once the input JNR power level is above 26dB and 1st-order CANF underperforms after this point, establishing the relationship between output SNR and MSE. As the MSE gets bigger, the evaluated CAF's output SNR is degraded severely. It also provides evidence that the notch depth (attenuation) is not deep enough to counter high-power CWI interference.

In order to determine the Acquisition Figure of Merit, the GPS L1 signal of eight different satellites (PRN=1,11,14,20,22,23,31, and 32) was generated and combined with CWI interference signal added to it and fed into CANF and then processed by acquisition block. Acquisition block search for each satellite's PRN code one by one and determine the value of ρ_{max} for each of the satellite. If the value of ρ_{max} is above two for any of the satellites, it means the signal of that satellite is visible and can be acquired by acquisition block.

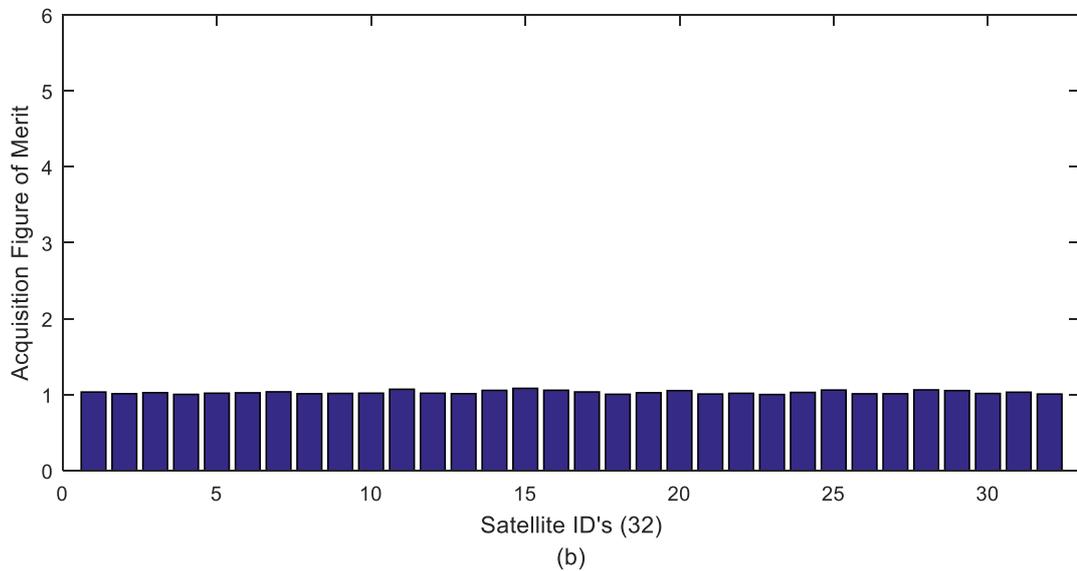
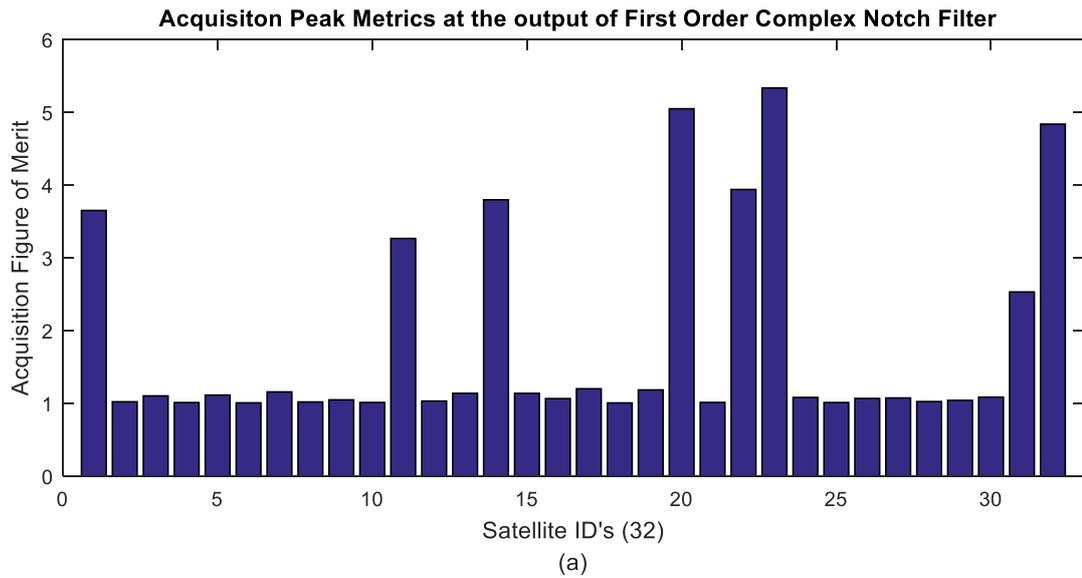


Figure 4.14 (a) ρ_{max} for acquired satellite after mitigation (b) None of the satellites is acquired as $\rho_{max} < 2$ due presence of interference signal with JNR of 20dB

In Figure 4.14 (b) the ρ_{max} values for all 32 satellites is almost one, and none of the satellite's signals can be acquired due presence of CWI interference signal with JNR of 20dB. The same simulated signal with CWI interference is processed through CANF and ρ_{max} values were determined. The ρ_{max} values for PRN, 1, 11, 14, 20, 22, 23, 31, and 32 are well above the threshold ($\rho_{max} > 2$) and satellite with these PRN can be acquired, as shown in 4.14 (b) and tracked by the receiver.

4.6 Complex Notch Filters: Higher Order

Analysis and excision of the CWI and complex sinusoidal signal using 1st order complex notch filter (CNF) leads to the following conclusion

- First, order CNF filter underperformance if the value of JNR is above 25 dB; hence GPS L1 signal is unable to acquire via acquisition block
- To mitigate high power CWI and complex sinusoidal signal, it requires a notch filter with a deeper notch and tighter notch bandwidth
- In literature [48],[49], and [55], the applicability of ANF and validity of its performance for mitigation of CWI is carried out without providing a structural implementation on actual hardware such as FPGA and addressing issues such as quantization of data and round-off noise and quantization of coefficients of ANF
- The performance of first-order CANF further degrades if its coefficient is quantized

These reasoning became the primary motivation for further analysis of higher-order CNF. Incorporate them together so that, depending on the JNR power level system, it can decide which CNF can be used for efficient excision of CWI interference without harming or distorting the actual useful signal. This section of the research focuses on the performance of the Notch filter up to order 6.

Table 4.1 Transfer Function of First, Second and Third-Order Notch filter

First-order	$H_{N1}(z) = \frac{1 - z_0^* z^{-1}}{1 - k_\alpha z_0^* z^{-1}}$
Second-order	$H_{N2}(z) = \frac{1 - z_0^* z^{-1}}{1 - k_\alpha z_0^* z^{-1}} \times \frac{1 - z_0^* z^{-1}}{1 - k_\alpha z_0^* z^{-1}}$
Third Order	$H_{N3}(z) = \frac{1 - z_0^* z^{-1}}{1 - k_\alpha z_0^* z^{-1}} \times \frac{1 - z_0^* z^{-1}}{1 - k_\alpha z_0^* z^{-1}} \times \frac{1 - z_0^* z^{-1}}{1 - k_\alpha z_0^* z^{-1}}$

Mathematically higher order complex notch filter can be represented as the following equation

$$H_{Nth\ order}(z) = \left(\frac{1 - z_0^* z^{-1}}{1 - k_\alpha z_0^* z^{-1}} \right)^N \quad (4.19)$$

where $z_0^* = e^{-j2\pi n f_n}$ a complex zero

- f_n is normalized notch frequency
- N is the order of the notch filter

In the following section, two sets of the simulation were carried out for the first, second and third-order complex filter. In the first case z_0^* and k_α were kept the same for the three filters and these filters have different notch widths, as shown in Figure 4.15 (a). Then SNR versus JNR plot is obtained to find out what is the maximum level interference each of the complex notch filters can attenuate for fixed k_α for all the filters. In the second case k_α is changed such that each of the three filters has the same 3dB bandwidth, as shown in Figure 4.15 (b).

Case I : Different 3dB bandwidth but same z_0^* and k_α for all three filter

Case II : same 3dB bandwidth and different k_α for all three filter

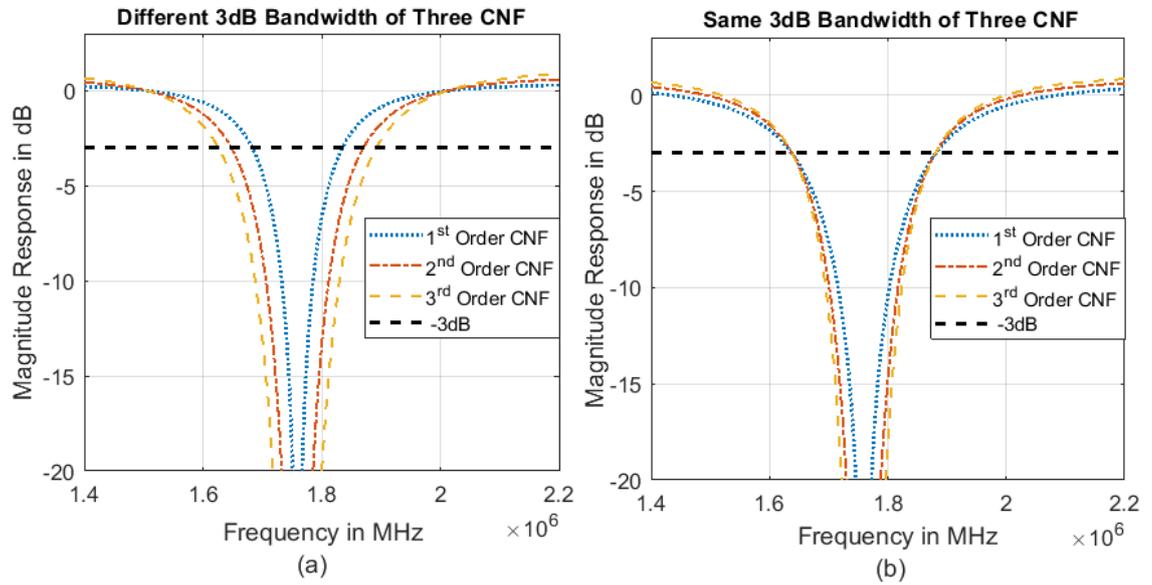


Figure 4.15 (a) Different 3dB bandwidth of three complex notch filter. (b) All three filter with the same 3dB bandwidth

Case I: Simulation Parameters and Results

Table 4.2 Simulation Set-Up Parameters

Parameters	
Fs	5MHz
C/N_o	44dB-Hz
JNR	5-60dB
IF	1.25MHz
Data length in the millisecond	4ms
No. of non-coherent integration	4
Complex Sinusoidal Interference Frequency	1.759MHz
Pole Contraction Factor k_α	0.92

The simulation results in Figure 4.16 shows the performance of the 1st, 2nd and 3rd notch filter in terms of output SNR in dB with simulation parameters in table 4.1. The working threshold for 1st, 2nd and 3rd are 26dB, 46dB and 51dB, respectively, as marked with circles

in Figure 4.16. One of the future research goals is to develop a parameterizable notch filter that can switch from lower to higher order depending on the input JNR level. The proposed system-level model is illustrated in Figure 4.17. The regions marked by three rectangle labels with numbers 1, 2 and 3 in Figure 4.16 indicate respective regions when the 1st, 2nd and 3rd notch filters have to be active to remove the CWI interference. It can be observed from Figure 4.16 first-order complex notch filter has higher SNR compared to that of the 2nd order notch filter for JNR levels up to 11dB and up to 7dB for the 3rd order notch filter.

The notch width of the complex first-order filter is narrower than that of 2nd and 3rd order (refer to Figure 4.15 (a)). Hence narrow bandwidth means a small portion of the useful signal is excised along with interference, and the depth is deep enough to attenuate the interference level of 11dB. But as the interference level increases above 11dB, 1st notch filter cannot sustain its performance. Output SNR steeply descends because the notch depth is not deep enough to attenuate the high-power interference signal completely. Similarly, for the 2nd order complex notch filter, the output SNR start dropping down (in a controlled manner) when the JNR level gets above 30dB, but this drop is not as rapid as for the 1st notch filter. It can be concluded that a higher-order notch filter performs better when the JNR power level is more. As aforementioned, notch bandwidth plays an important role, and there has to be some trade-off between the filter order, notch depth and notch bandwidth. In the simulation, the value of z_0^* and k_α were predetermined. Position of z_0^* on the unit circle corresponding to the frequency of the narrowband CWI interference and the value of the pole contraction factor is $k_\alpha = 0.92$. As the value of k_α is the same for all three notch filters, the corresponding 3dB notch bandwidth is different for all three filters. It is shown in Figure 4.15(a), using the same z_0^* and k_α for each first, second and third notch filter, all three filters have different 3dB bandwidths. As we increase the order while keeping the z_0^* and k_α the same 3dB bandwidth expands for the subsequent order.

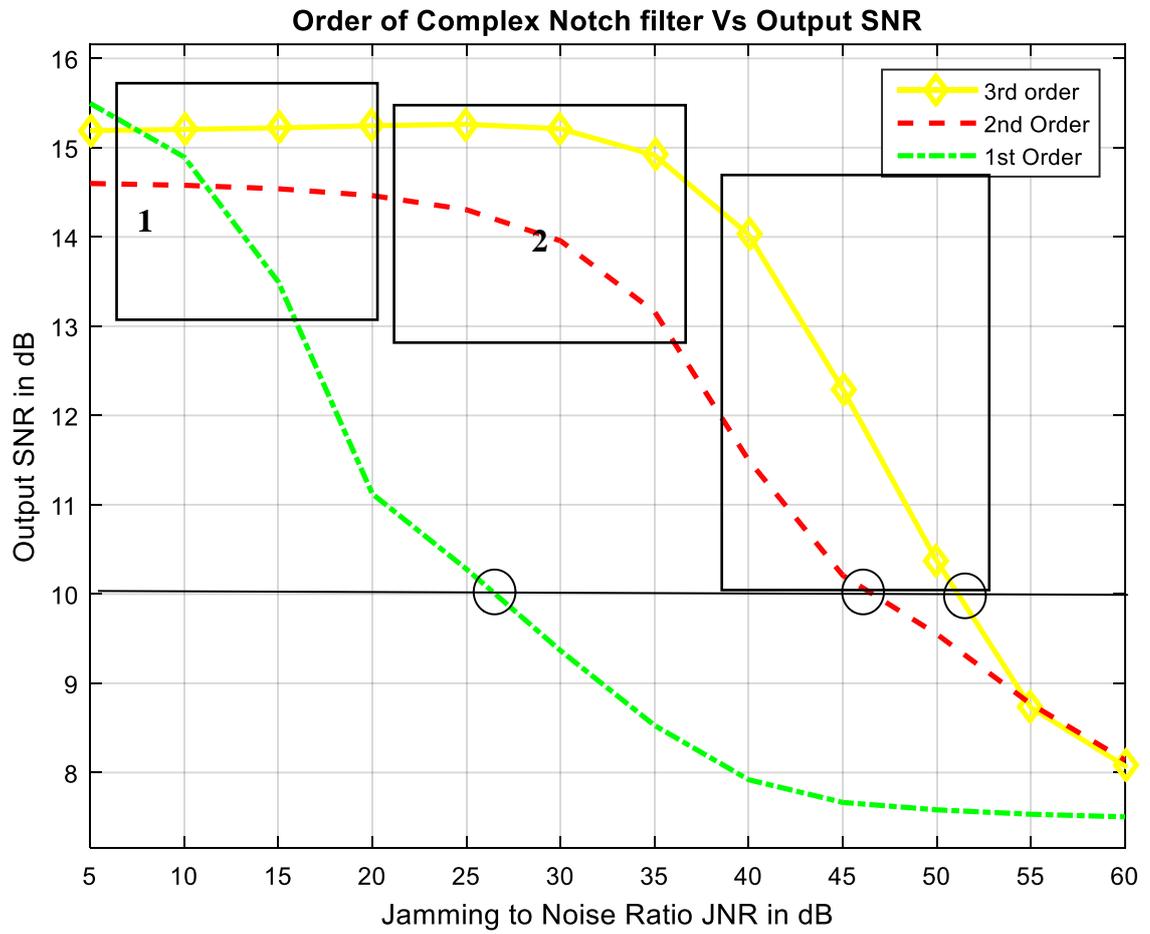


Figure 4.16 Output SNR Vs. Order of Notch filter with the same z_0^* value for all the notch filter

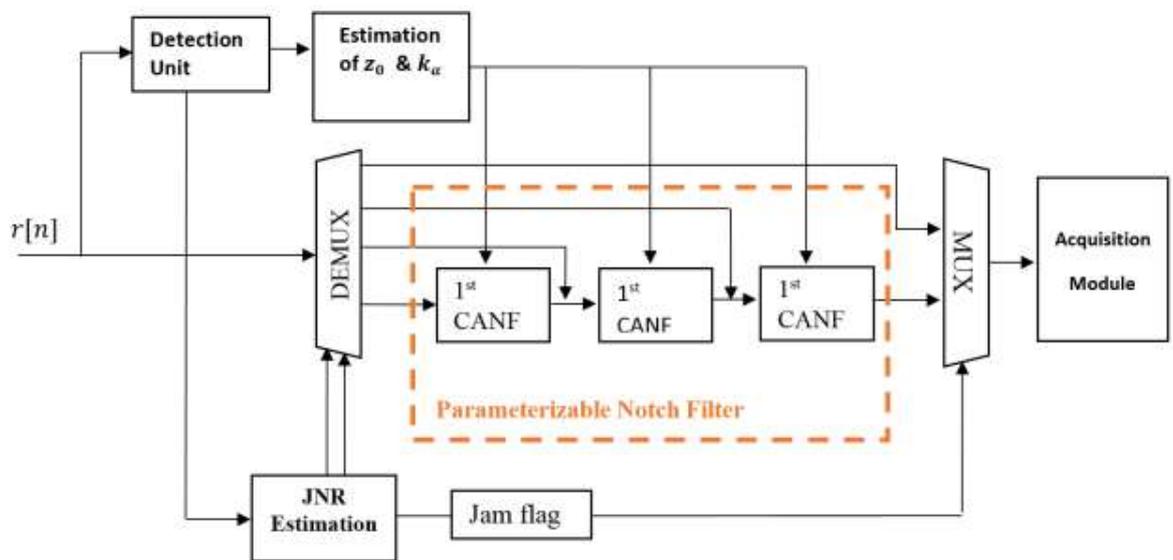


Figure 4.17 Proposed System-Level Model

Reason for keeping the values of z_0^* and k_α same to ease the structural implementation of the parameterizable notch filter. Depending on the incoming interference power, that can easily switch from one order to another. Simply by using the coefficient of a complex first-order filter, the algorithm can be developed using iterative methods to calculate the coefficients of a second and third-order notch filter for mitigating the high-power interference.

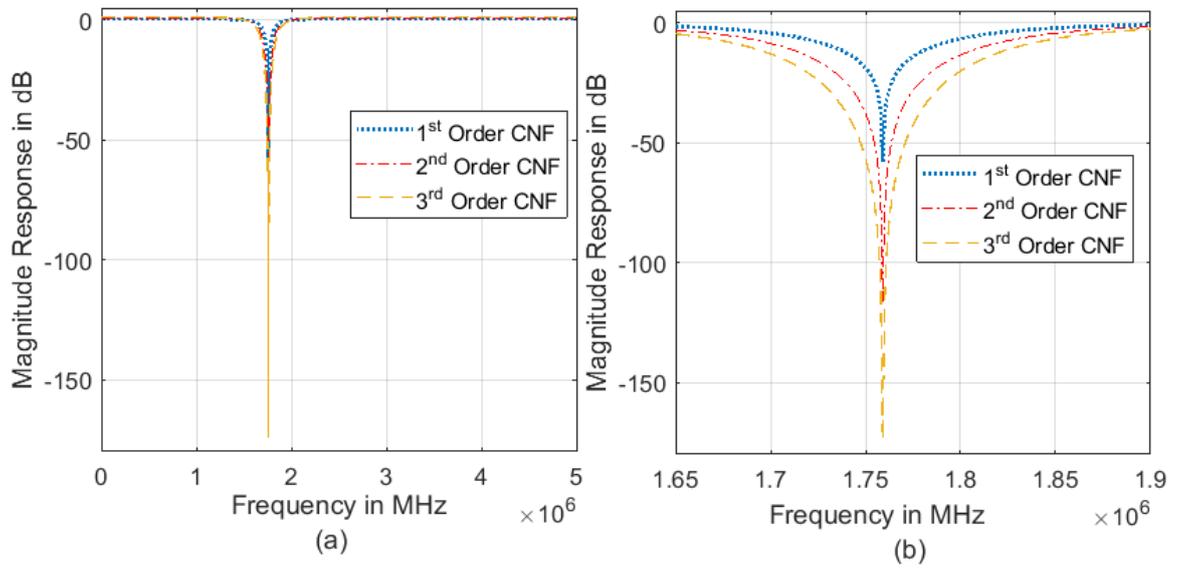


Figure 4.18 (a) Order of Notch filter and corresponding frequency response and notch depth but width different 3dB bandwidth. (b) Zoomed in version of (a).

Simulation Results for Case II: *All three notch filters have equal 3dB notch bandwidth.*

The simulation parameters are the same as in the previous section, except that value of k_α for three complex notch filters are different such that each filter has the same 3dB notch bandwidth, as shown in Figure 4.15(b).

Table 4.3 Value of k_α for corresponding Order of notch filter to attain the same 3dB notch bandwidth for all three filter

k_α	Order of filter
0.84	1
0.89	2
0.92	3

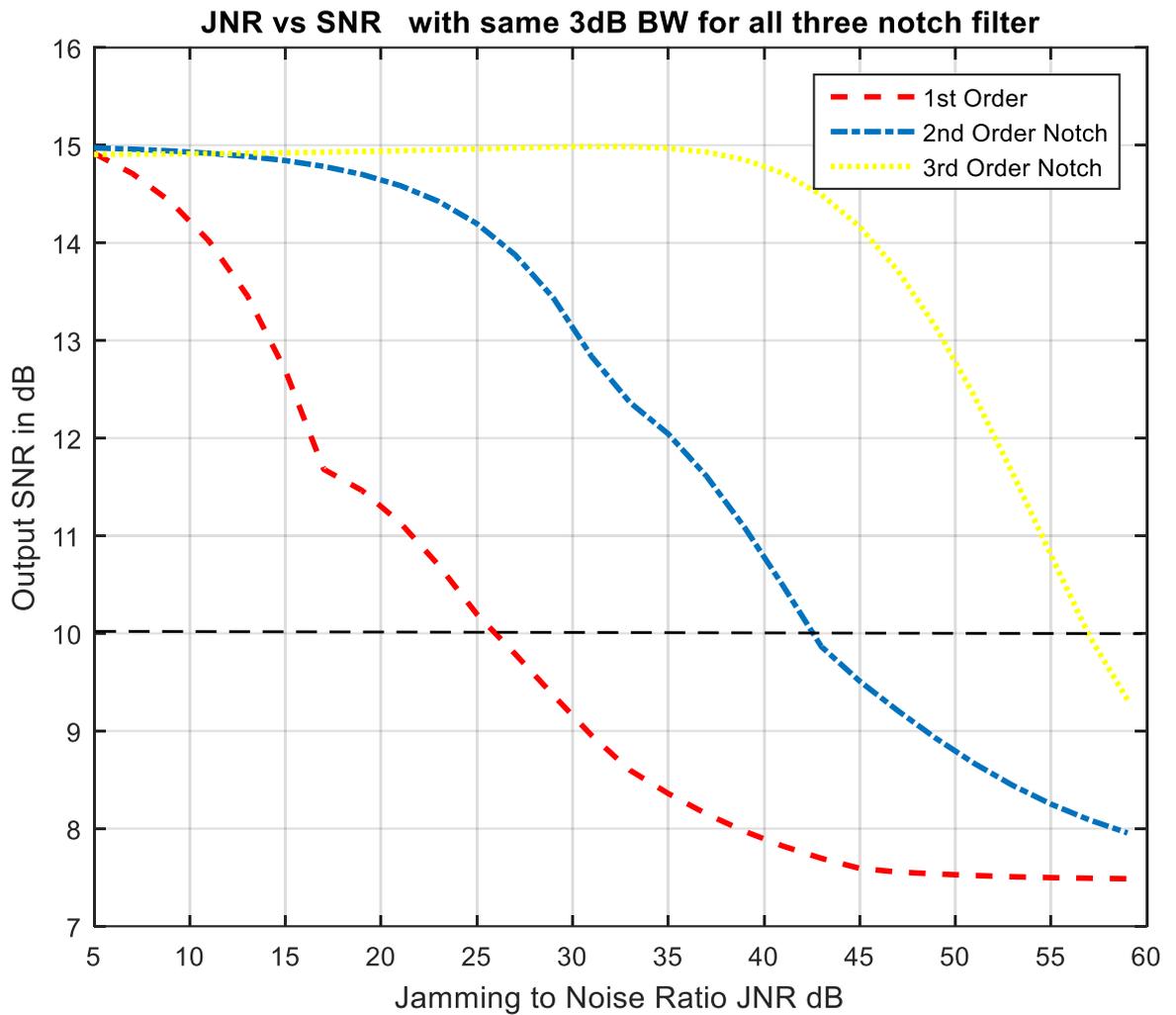


Figure 4.19 Filters Performance with equal 3dB bandwidth

All three notch filters have an output SNR of 15dB for JNR of 5dB, as shown in Figure 4.19. As they have the same notch bandwidth, the equal amount of useful signal along with the interference is excised. All three notches are deep enough to attenuate interference at the JNR of 5dB. Therefore, the output SNR is the same for all of them, which is 15dB. The 2nd and 3rd order notch filter has the same output SNR for JNR up to 13dB; from that point onward, as the interference power increases, the SNR for the 2nd order notch filter decreases, hinting that the notch attenuation capability of 2nd order is not good enough. From this simulation results, it can be concluded that if the notch bandwidth is equal for all three notch filters, the output SNR depends upon the on-attenuation level of the notch depth.

As the three filters have different values of k_α from an implementation point of view, it will require more hardware resources and complicate the hardware realisation. The target is to design and implement a less power-hungry system and reduce hardware structural complexity for the higher-order notch filter.

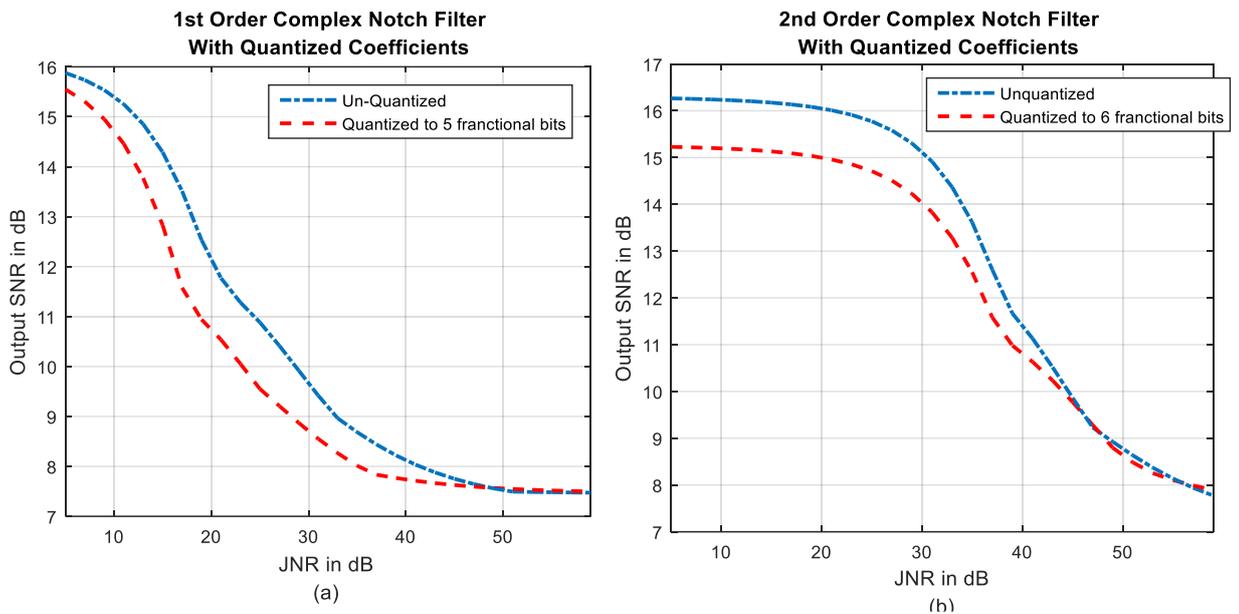


Figure 4.20 Effect of coefficients quantization on Output SNR for 1st Order CNF.(b) Effect of coefficients quantization on output SNR 2nd CNF

The proposed system-level model in Figure 4.17 can further be modified to accommodate an even higher-order notch filter. Hence interest develops to investigate how higher-order notch filters would perform under the same conditions for this set of simulation C/N_o is changed to 47dB-Hz. In the previous simulation, the GPS L1 signal was assumed to be a weak signal with less C/N_o . Therefore, a longer integration dwells time (4ms) was required. To speed up the simulation, it is assumed that the incoming GPS L1 signal is strong and requires 1ms of coherent integration to acquire the signal. The simulations result in Figure 4.21 shows the performance of the notch filter up to order six, with an acquisition time of 1ms, C/N_o of 47dB and k_α equal to 0.88.

The acquisition module's optimal performance requires optimum excision of the interference signal using the right order of the notch filter at a given JNR power level. From the simulation results in Figure 4.20, different thresholds can be set when activating a higher-order filter.

There is no point in using a higher notch filter for a low JNR level which consumes more power. A simple and less power-hungry 1st order notch filter can be used to mitigate interference up to 10dB with reference to Figure 4.20. The second-order notch filter shows exceptional good performance up to JNR of 20dB. The output SNR is even higher than any higher-order notch filter. Using a second-order notch filter to remove interference with a JNR level of 20dB is an ideal choice. The SNR curve for the 3rd and 4th orders is almost similar. There is a slight improvement in the SNR of the 4th order notch filter from JNR 32dB onwards. 5th and 6th order notch filters give superior performance for interference mitigation for the rest of the notch filters when interference power is above 50dB. A decision variable (threshold setting) can be set up using the simulation results of Figure 4.21. Depending on the JNR level, one can decide how to efficiently utilize each notch filter for interference mitigation and reduce the system's power consumption.

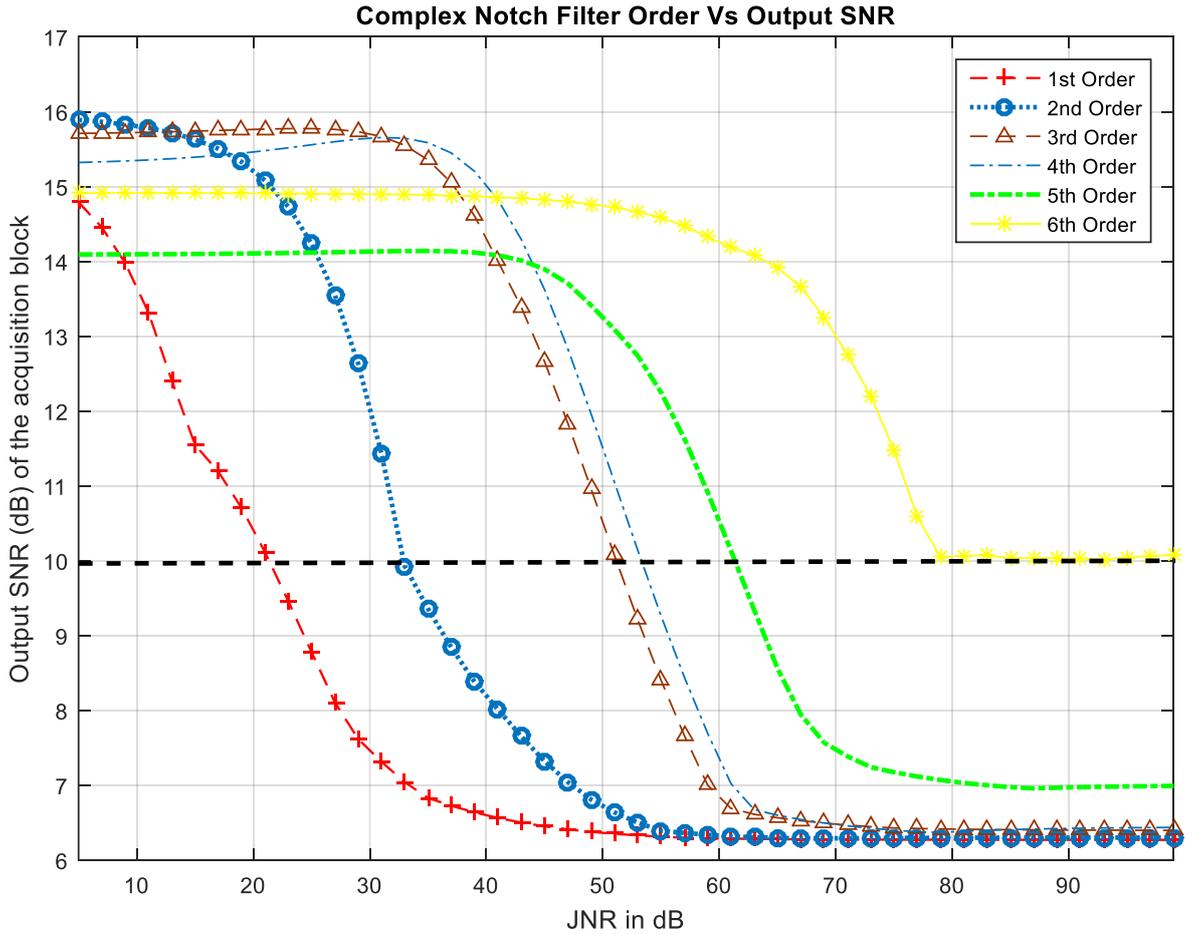


Figure 4.21 Output SNR for complex notch filter up to order 6 with input GPS L1 signal with C/N_0 of 47dB

4.7 Complex Adaptive Notch filter

The previous section analysed fixed complex notch filters up to the 6th order to mitigate complex narrowband CWI interference. Poles and Zeros were predetermined, and the notch's bandwidth was also fixed. The main reason was to develop and establish how higher order can be beneficial for retrieving useful GPS signals and realise them in a simple hardware structure. Then extend the higher-order filter design into an adaptive notch filter for the proposed system-level model, as shown in Figure 4.17. In real-life scenarios, the power, frequency and phase of incoming interference are unknown to the GNSS receiver. Therefore, intelligence (machine learning) needed to be added to a complex notch filter to determine the interference's power level and frequency. Not only that, if the nature of interference

changes something like the chirp signal, the notch filter must also constantly track, follow and mitigate the time-varying frequency of the interference. Most of the research work carried out in this section is based on [48], [63], [64] and [67]. Y.Chien [48] proposed a real second-order adaptive notch filter based on an all-pass structure, with adaptively adjusting the β (determine the frequency location on the unit circle) parameter to track the frequency and power level of the interference. In [68], the author proposed to dynamically adapt both notch centre frequency and the pole contraction factor for first-order complex notch filter and extended the design to track multiple complex sinusoidal signals for general purposes. The goal is to extend these designs and algorithms for GNSS-specific applications for interference mitigation, such as complex sinusoidal and chirp-like signals.

4.7.1 First-order Complex Adaptive Notch Filter

Figure 4.22 shows an implementation of a single pole complex coefficient IIR notch using a normalized Least Mean Square (LMS) algorithm. Discrete input of the CANF is $x_i[n]$, which consists of GPS L1 signal $s[n]$, a complex sinusoidal signal $j_{int}[n]$ with unknown amplitude and frequency and complex white Gaussian noise $\eta[n]$ with zero mean and variance σ_n^2 .

$$x_i[n] = s[n] + j_{int}[n] + \eta[n], \quad \text{where } J_{int}[n] = A_i \exp\{j2\pi n f_i T_s + \theta_i\} \quad (4.20)$$

Incoming GPS L1 signal $x_i[n]$ with interference is filtered out by Autoregressive Moving Average structure, and structure can be represented by the following difference equations:

$$\textbf{Moving Average:} \quad x_o[n] = x_e[n] - z_0 * x_e[n - 1] \quad (4.21)$$

$$\textbf{Autoregressive:} \quad x_e[n] = x_i[n] - k_\alpha * z_0 * x_i[n - 1] \quad (4.22)$$

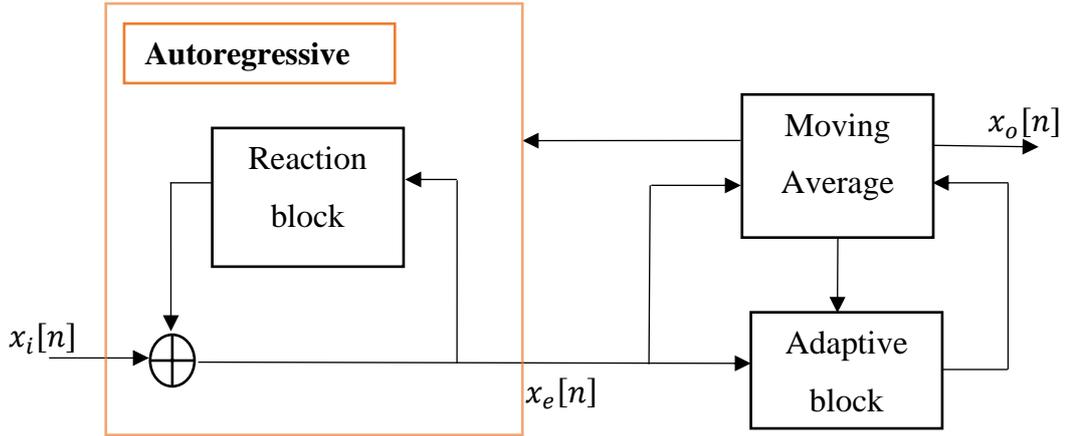


Figure 4.22 ARMA Structure of Complex Adaptive Notch Filter

The equivalent transfer function becomes as follows:

$$H_{notch}(z) = H_{MA}(z) \times H_{AR}(z) = (1 - z_0 z^{-1}) \times \frac{1}{1 - k_\alpha z_0 z^{-1}} \quad (4.23)$$

4.7.2 Adaptation Algorithm

The adaptive block in Figure 4.21 is the main processing block that determines the position of the zero to track and cancels the interference. The adaptive algorithm is used to adjust the position of the zero at each iteration using specific criteria to minimize a specific cost function. Different types of adaptive algorithms are used to minimize the cost function, and two main types are Least Mean Square Algorithms (LMS) [69] and Recursive Least Square Algorithms (RLS) [70]. These algorithms have further variants of each other, such as normalized LMS, variable step size LMS, and RLS-QR. The desired properties of an adaptation algorithm are similar for many applications and are summarized as follows

- *Convergence Rate*: The rate at which or the number of iterations are required to achieve the optimal solution. It has to be as fast as possible means the minimum possible iteration used to reach within the approximation range of the optimal solution

- *Computational and Structure Complexity:* The implementation of algorithm realisation on hardware is simple and power-efficient. There are fewer arithmetical operations, fewer hardware resources and less memory used to implement the algorithm.
- *Mis adjustment:* This is the difference between the filter's performance after it has been allowed to adapt for an infinite time and the optimal filter.
- *Robustness:* The filter should be capable of producing satisfactory performance under ill-conditioned data, such as in a very noisy environment and changes in noise and signal model.

The adaptive algorithms chosen are based on work reported in [67], and the normalized LMS method is used to minimize the cost function [69]. In the basic least mean square algorithm, the filter weights are adjusted one sample at a time to minimise the Mean Square Error (MSE). LMS is based on the steepest descent algorithms where the filter coefficients are updated as follows:

$$W_{n+1} = W_n - \mu \nabla_n \quad (4.24)$$

Where W_n the weight vector (Coefficients) is, ∇_n is the gradient signal which is basically the differentiation of the square of the error signal and μ is the called step size parameter which controls the stability and rate of convergence. The cost function for CANF can be represented by [70]:

$$C[n] = E\{|x_0[n]|^2\} \quad (4.25)$$

Where x_0 is the output of the CANF, and to minimize this cost function $C[n]$, it has to be differentiated with respect to complex zeros z_0 and to find the stochastic gradient of the cost function:

$$\text{grad}(C[n]) = \nabla_{z_0}\{|x_0[n]|^2\} \quad (4.26)$$

z_0 The coefficient of CANF can be updated as follow:

$$z_0[n] = z_0[n - 1] - \mu \times \text{grad}(C[n]) \quad (4.27)$$

μ Is the normalized step size given by:

$$\mu[n] = \frac{\delta}{E_{x_e[n]}} \quad (4.28)$$

Where δ is the non-normalized LMS step-size and $E_{x_e[n]}$ is the squared Euclidean norm of the signal $x_e[n]$ which is the output of the autoregressive part. Step size is normalized to reduce gradient noise amplification when the input signal is very large. δ Has to be chosen carefully and accurately as it controls the convergent properties of the algorithms and reduces misadjustment. Both x_0 and z_0 are complex variables; hence while differentiating the cost function, the complex generalized derivative rules should be used to calculate the gradient of the cost function. As given in [51], cost function differentiation leads to the following equation:

$$\text{grad}(C[n]) = \nabla_{z_0}\{|x_0[n]|^2\} = -4x_0[n](x_e^*[n - 1]) \quad (4.29)$$

Substituting equation 4.23 into equation 4.21 gives the following adaptation equation for the coefficient z_0 of the CANF filter.

$$z_0[n] = z_0[n - 1] - \mu \times 4x_0[n](x_e^*[n - 1]) \quad (4.30)$$

The above normalized LMS is implemented and modelled in MATLAB. The results in Figure 4.23 validate the performance of CANF with normalized LMS algorithms. The bandwidth of the notch can be varied by changing the parameter k_α and rejection band can be narrowed down by keeping the value of k_α close to 1, but it cannot be too close to 1 for

stability and convergence reasons. From Figure 4.23 (a), it can be observed that the peak of the acquisition block is well-defined and distinct. It means the complex adaptive notch filter was able to track and remove undesired complex sinusoidal interference and the GPS L1 signal was able to acquire by the acquisition block.

Plot (c) in Figure 4.23 shows how the magnitude of z_0 converges to unity. The magnitude of the z_0 varies according to the level of JNR. The parameter z_0 is very useful and powerful if its properties are fully utilized. More insight is given into it at the end of this chapter. The first-order notch filter is a complex filter; hence, a complex baseband GPS L1 signal and complex sinusoidal interference are modelled together and processed through the CANF filter. The step size for updating the z_0 is carefully selected and tuned accordingly the type of the interference signal. The plot (b) in Figure 4.23 shows the time-frequency evolution of incoming GPS L1 signal along with fixed frequency complex sinusoidal interference. And the dotted red line represents the estimated centre frequency of the notch. The plot below shows the time-frequency representation of the signal cleaned by the CANF and indicates that interference has been successfully removed. The last and fourth plot illustrates the CAF evaluation of the acquisition block's output after the interference mitigation.

Table 4.4 Simulation Parameters for 1st Order Complex Adaptive Notch Filter

Parameters	
F_s	5MHz
C/N_0	44dB/Hz
JNR	14dB
IF	1.25MHz
Data length	4ms
No. of non-coherent integration	4
Complex Sinusoidal Interference Frequency	1.759MHz
Doppler Shift	2663Hz
Code Phase Delay	3698

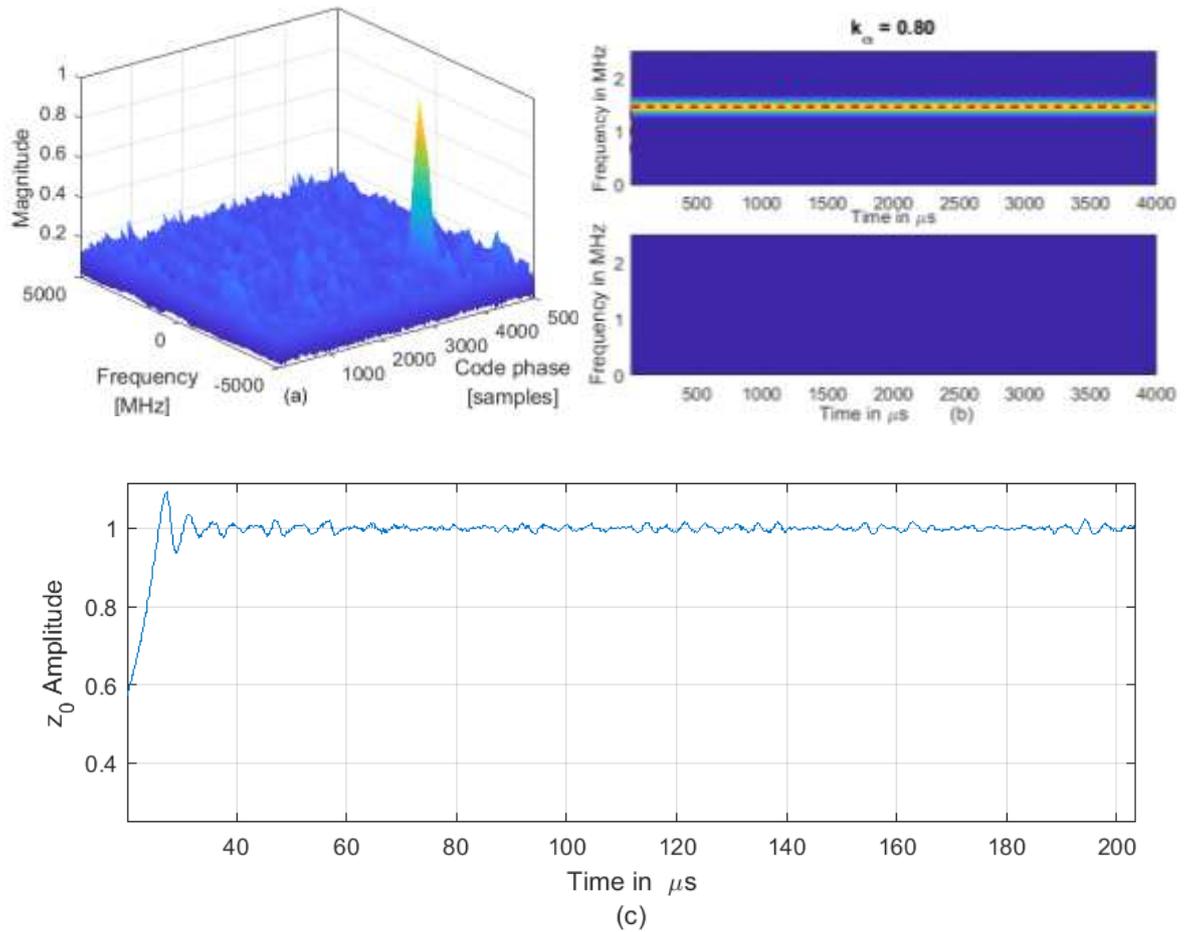


Figure 4.23 (a) acquisition of GPS L1 signal after tracking and mitigating complex sinusoidal and With JNR of 14dB. (b) STFT of the signal with interference and after mitigation. (c) Convergence of z_0 to unity

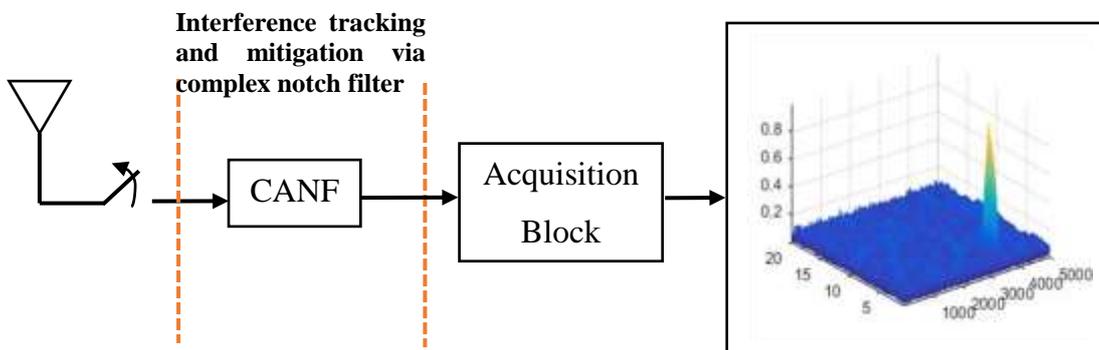


Figure 4.24 Schematic of GPS signal acquisition after interference mitigation

Figure 4.24 illustrates the testing of different adaptation algorithms within an adaptive notch filter in terms of the acquisition. If the peak is prominent and the location of the peak on the grid is at its expected position, it can be presumed adaptive notch filter functioned as desired to.

4.7.3 Mitigation of Complex Chirp-Type Interference

At the start of this chapter, an overview was given on modelling the complex chirp sinusoidal. A first-order complex adaptive notch filter was also tested to mitigate chirp-type interference signal. Plot (a) in Figure 4.25 illustrates a time-frequency representation of a received signal with chirp interference. The red line represents the tracking frequency estimation by the adaptive notch filter. Plot (b) in Figure 4.25 shows the time-frequency representation at the output of the CANF after the mitigation of complex chirp sinusoidal signal with a fast-varying frequency between 2.5MHz to -2.5MHz. By adjusting parameter k_α the performance of the CANF can be further fine-tune and hence it can provide better results. The simulation parameters are the same as given in table 4.4, except for the following changes:

- Doppler Shift = 2620 Hz
- Code Phase Delay = 3505 chips
- Interference = Complex Chirp
- Complex Chirp Interference Frequency = varies between -2.5MHz to 2.5MHz over a period of every 50 μ s

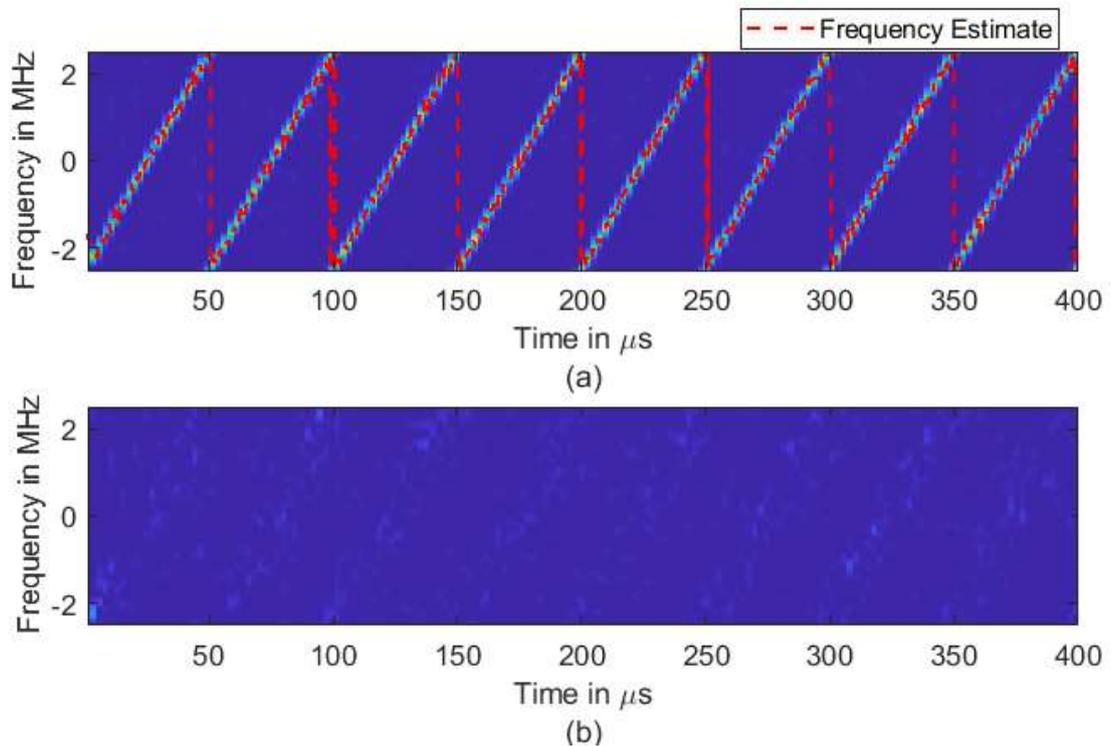


Figure 4.25 Top shows the time-Frequency evolution of the GPS L1 signal with interference. The red dotted line represents the notch's centre frequency, and the bottom plot shows the filtered-out data by the notch filter

The simulation results in the plot (a) of Figure 4.26 illustrates how the magnitude of z_0 varies over the time of 4ms (4000 μs), the total length of the data generated in the simulation. Unlike in the previous section, where the interference considered was a single-tone complex sinusoidal and the magnitude of z_0 tend to converge to unity. Here z_0 is constantly varying over a large range of fluctuations. Plot (b) in Figure 4.26 and 4.28 (b) shows the zoomed-in version of plot (a) of Figure 4.26 and Figure 4.28 and can be closely examined how the magnitude of z_0 varies up and down within a certain range.

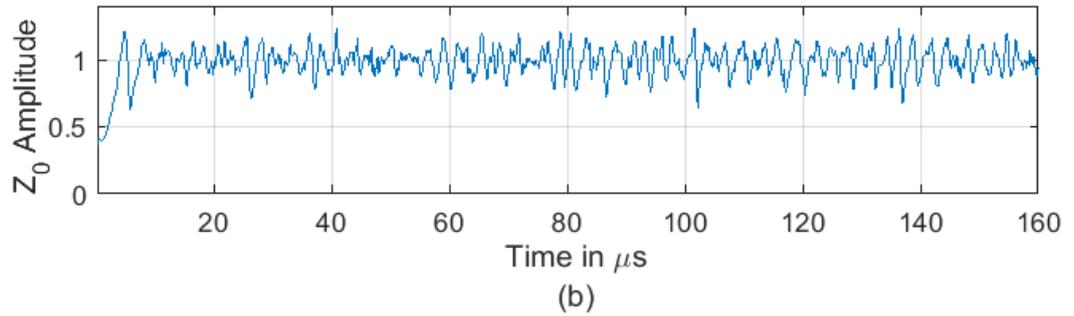
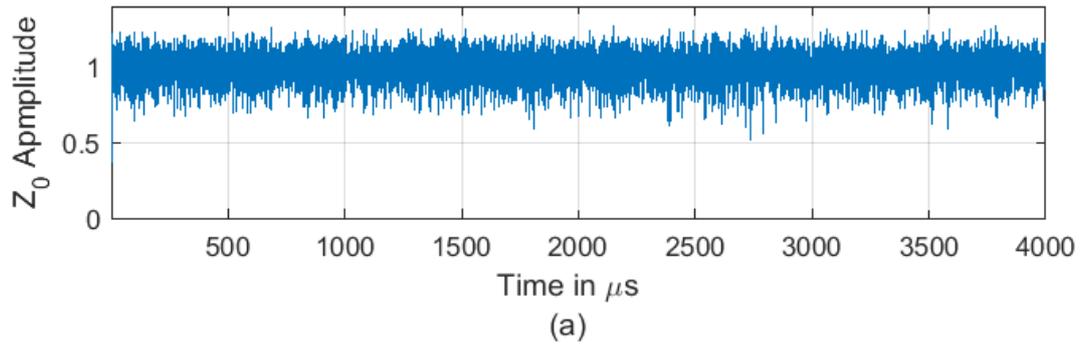


Figure 4.26 (a) Variation of the magnitude of z_0 over the time. (b) Zoomed version of (a)

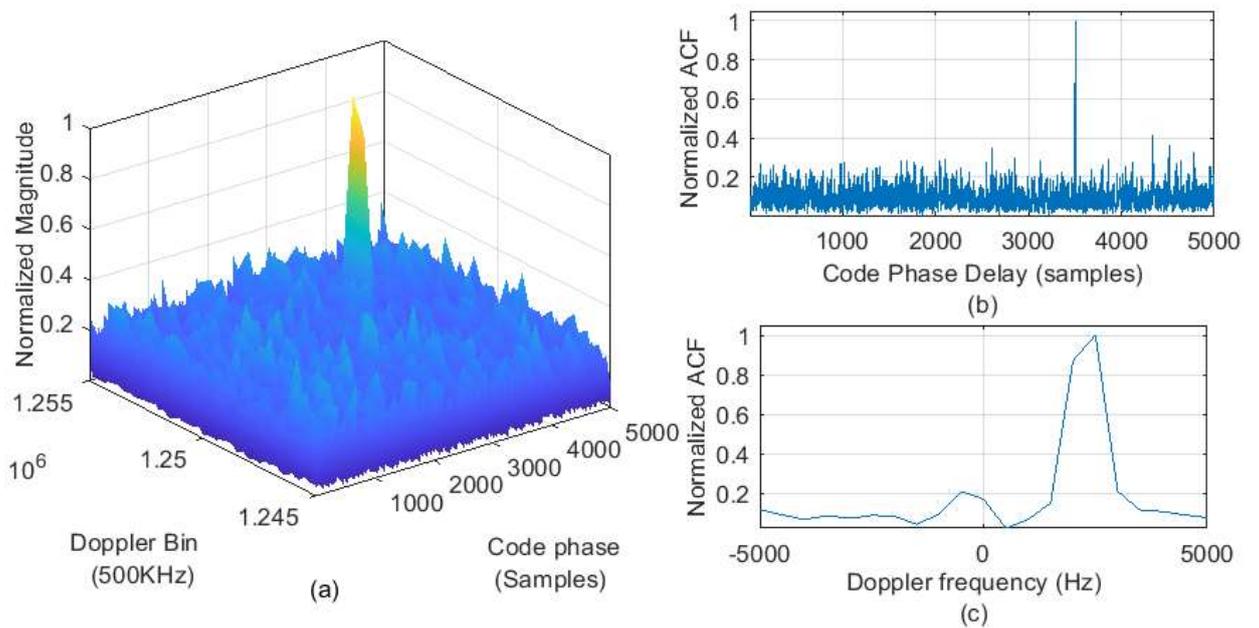
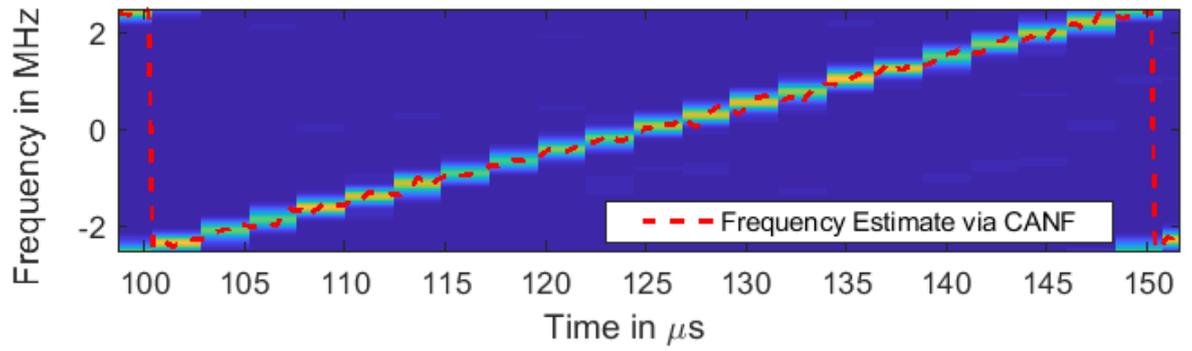
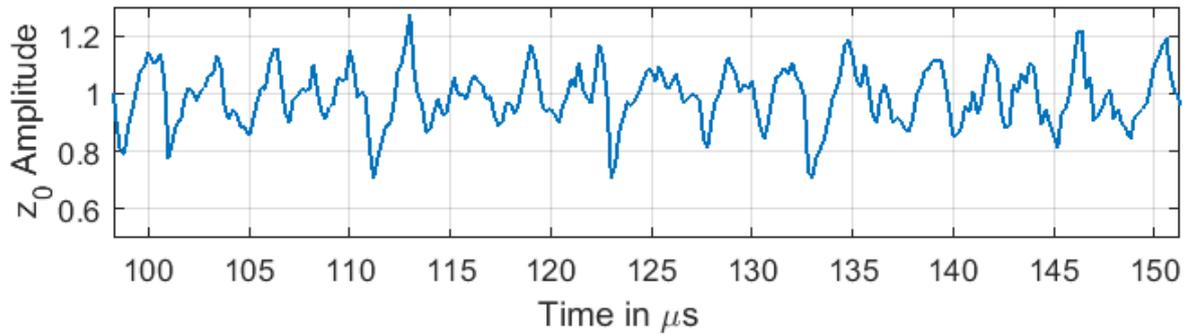


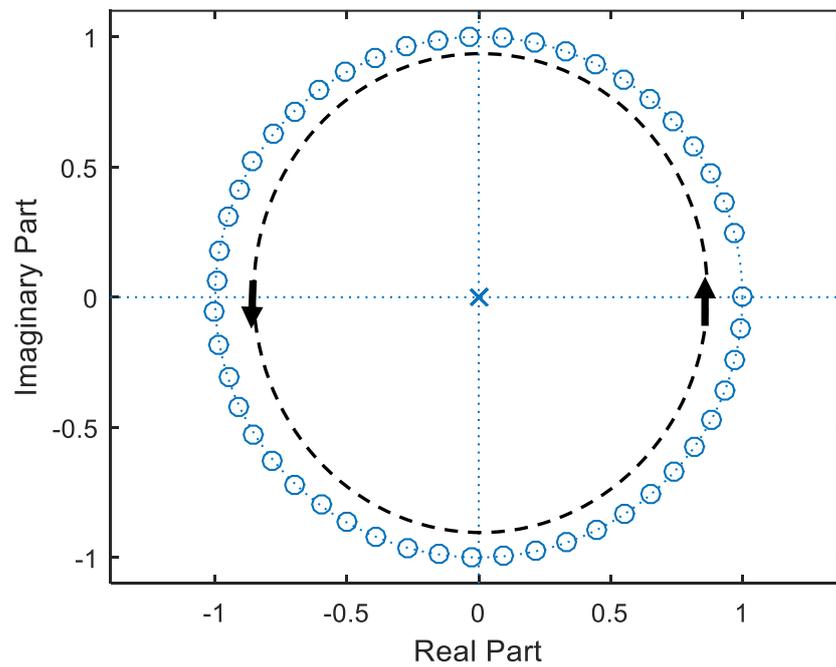
Figure 4.27 Output of acquisition block after mitigating complex-chirp signal with multiple saw-tooth functions



(a)



(b)



(c)

Figure 4.28 (a) Zoomed-in version of the evolution of notch centre frequency (red dash line). (b) variation of the magnitude of z_0 between $100\mu\text{s}$ to $150\mu\text{s}$. (c) Evolution of Complex Zero of the CANF filter in between $100\mu\text{s}$ to $150\mu\text{s}$ on z -plane

4.8 Second-order Real IIR Notch Filter

The design of the complex first-order filter can be modified and extended to mitigate single and multiple tones real narrowband CWI interference. Two spectral lines can represent a real CWI interference in the frequency domain, and the spectral lines are located at f_i and $-f_i$, hence notch filter is required to have two notches to mitigate CWI interference from the positive and negative parts of the spectrum. The structure of the second-order real filter is similar to the one used for the complex first-order filter in section 4.7, as shown in Figure 4.21. The filter is designed to filter the real sinusoidal interference modelled in equation 4.31. All it needs is to introduce conjugate zero and pole in the transfer function of the first complex notch filter and simplify to expression 4.34.

$$j_{int}[n] = A_i \cos(2\pi n f_i T_s + \theta_i) \quad (4.31)$$

- A_i is the interfering signal amplitude,
- f_i And θ_i are the frequency and phase of the continuous wave
- T_s The is the sampling interval,
- n is the time index.

The MA and AR difference equation is then given by

$$\text{MA : } x_o[n] = x_e[n] - 2\text{Real}(z_0)x_e[n-1] + |z_0|^2 x_e[n-1] \quad (4.32)$$

$$\text{AR: } x_e[n] = x_i[n] - 2k_\alpha \text{Real}(z_0)x_i[n-1] + k_\alpha^2 |z_0|^2 x_i[n-1] \quad (4.33)$$

$$\begin{aligned} H_{notch}(z) &= H_{MA}(z) \times H_{AR}(z) \\ &= (1 - 2\text{Real}(z_0)z^{-1} + |z_0|^2 z^{-2}) \times \frac{1}{1 - 2k_\alpha \text{Real}(z_0)z^{-1} + k_\alpha^2 |z_0|^2 z^{-2}} \end{aligned} \quad (4.34)$$

The same normalized LMS algorithm used as employed for the complex notch filter case, the minimization of the cost function $E\{|x_0[n]|^2\}$ and derivative was taken with respect to the complex parameter z_0 . The expression of the stochastic gradient is given by [49]

$$\text{grad}(C[n]) = \nabla_{z_0}\{|x_0[n]|^2\} = -4x_0[n](z_0[n-1]x_e[n-2] - x_e[n-1]) \quad (4.35)$$

And the equation to update the zero of the real notch filter becomes

$$z_0[n] = z_0[n-1] + \mu \times 4x_0[n](z_0[n-1]x_e[n-2] - x_e[n-1]) \quad (4.36)$$

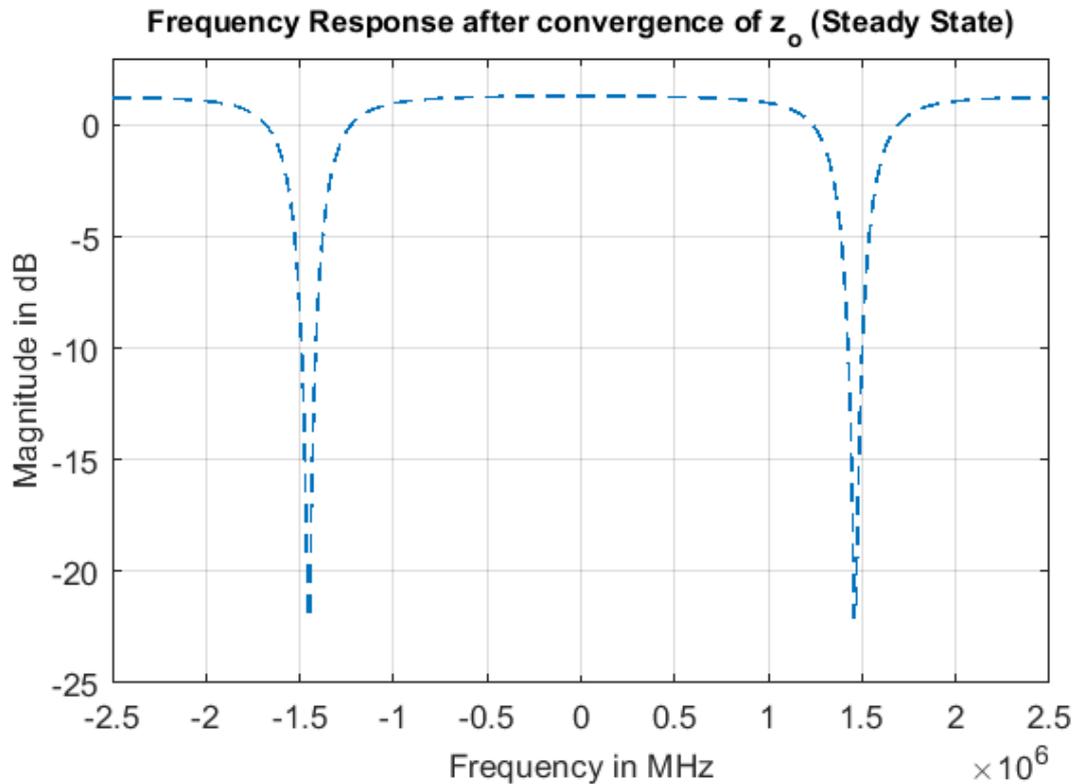


Figure 4.29 Frequency response of second-order adaptive IIR notch filter after it reaches steady state conditions

This normalized LMS algorithm for the second-order IIR notch filter is represented by equation 4.30 and is implemented and verified using MATLAB. The convergence of the complex zero z_0 is shown in Figure 4.30, it settles around the unity. The JNR level for this simulation was 20dB with a pole contraction factor k_a equal to 0.85 and CWI interference with 1.45 MHz frequency. After convergence of the adaptation parameter z_0 the steady-state

response of the second-order filter is illustrated in Figure 4.29. Then more sets of the simulation are obtained at the output of the acquisition block for further verification. The upper plot in Figure 4.31 shows a time-frequency representation of the signal with interference before a notch filter. The lower plot shows the time-frequency representation of the cleaned signal at the output of the second-order notch filter. And the red dot line in Figure 4.31 represents the adaptation of the notch filter's centre frequency, and the thick yellow line represents the CWI interference with a 1.45 MHz frequency.

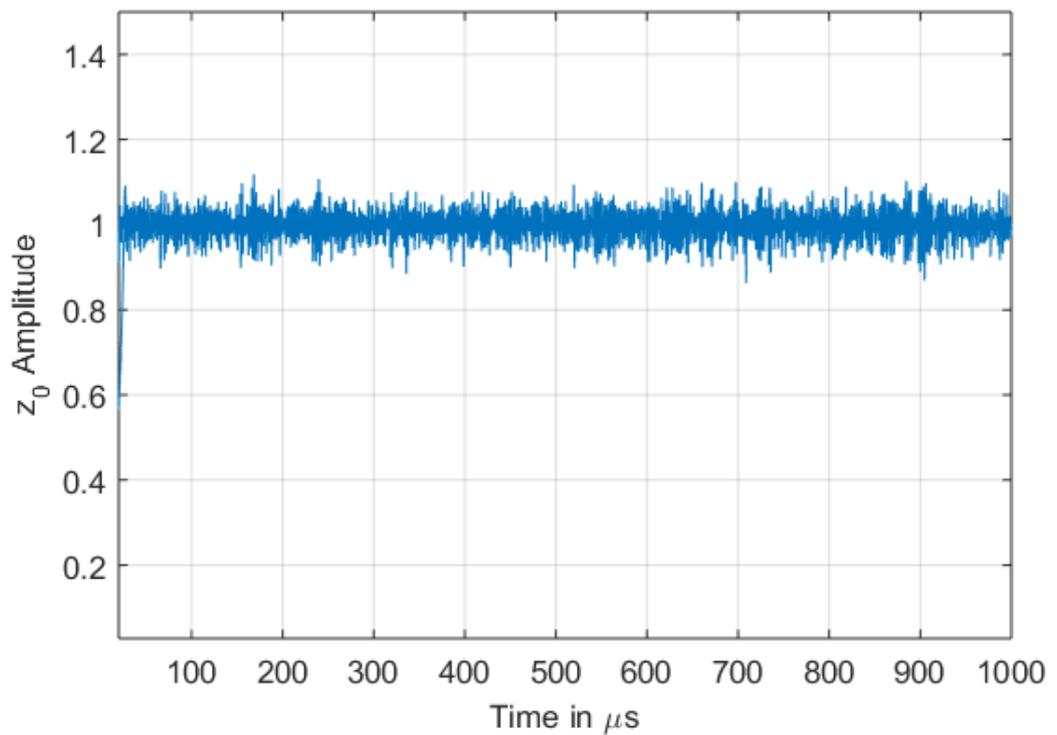


Figure 4.30 Convergence of $|z_0|$ close to unity

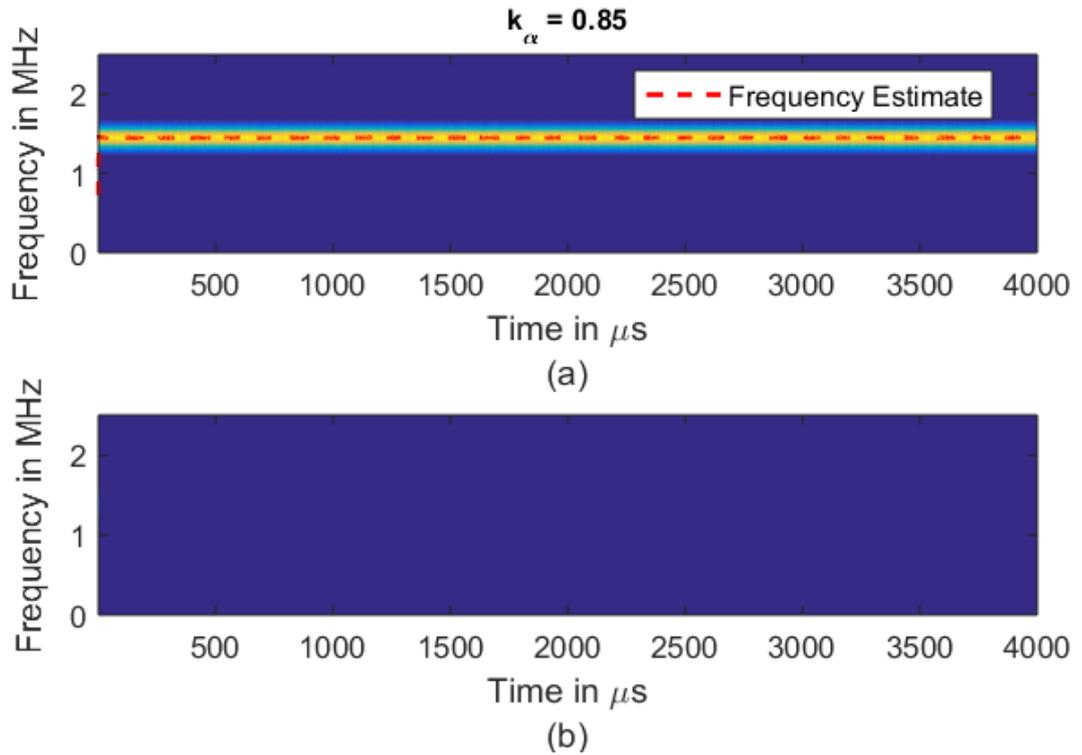


Figure 4.31 Time-Frequency representation after mitigation of the interference via notch filtering

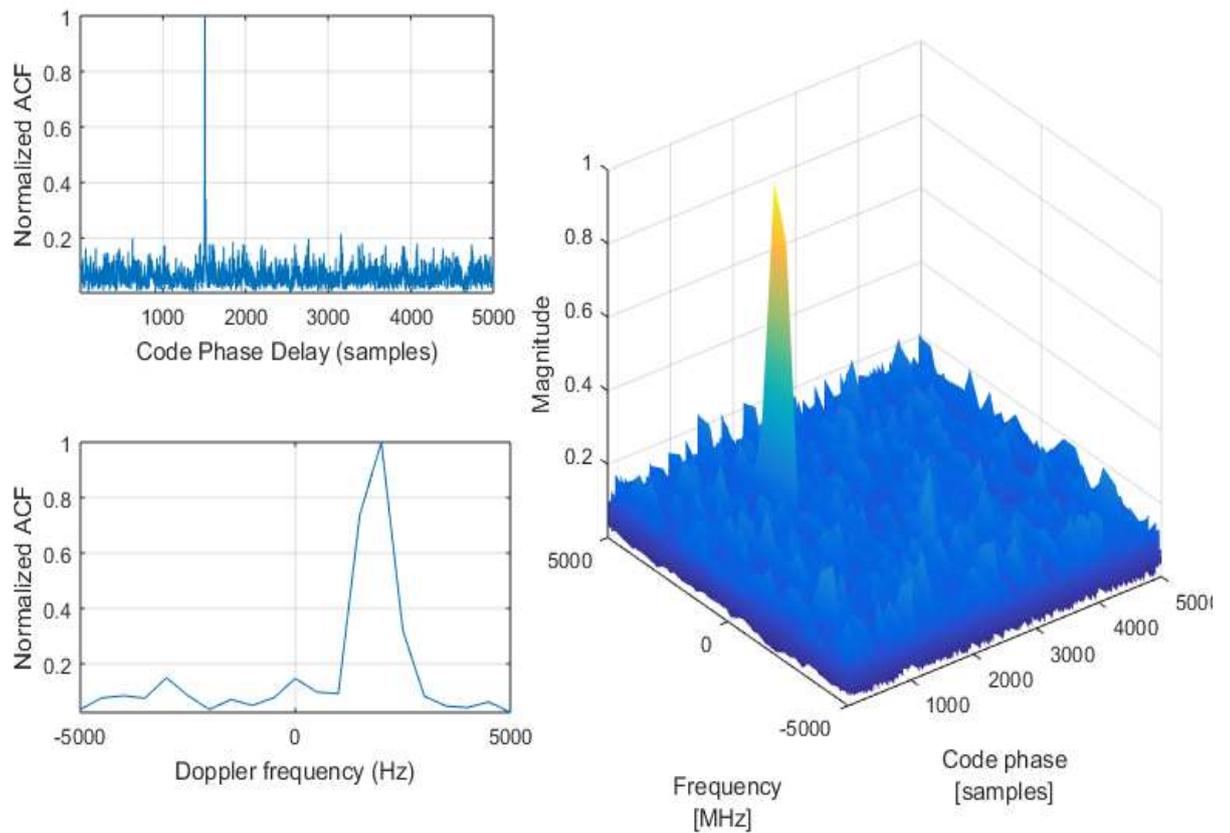


Figure 4.32 Output of acquisition block after CWI inference mitigation

From the simulation results of the convergence of magnitude z_0 it can be analysed that the amplitude of the adaptive parameter strongly depends upon the power of the interference signal. As reported in [64], this parameter can be used to detect interference. If there is no interfering signal present in a useful signal, the magnitude of z_0 will never converge to unity. Hence by further optimizing the algorithm and converging properties of the adaptive variable z_0 with some thresholds that can be set up. If the magnitude of z_0 is equal to or above that predetermined threshold, the proposed system activates the notch filter and processes the received signal through an adaptive notch filter to remove interference.

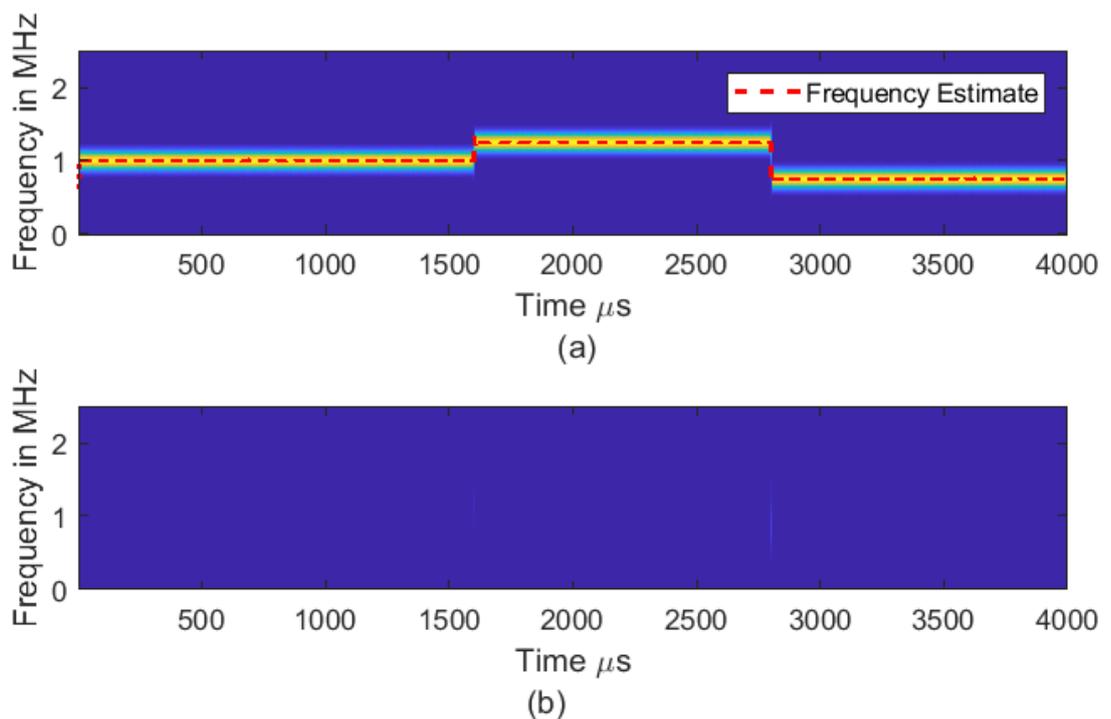


Figure 4.33 (a) Top plot shows a Time-Frequency illustration of frequency hopping of CWI interference, and the red dot line represents the centre frequency of the notch filter. (b) the Bottom plot filtered out data from the notch filter

This section presents more results on the tracking and mitigation capability of second-order IIR notch. In the following experiment, the narrowband interference frequency hopped from one value to another. Starting frequency of CWI was 1 MHz; after 1500 μ s, its frequency switched to 1.25 MHz and then to 0.75MHz, as shown in the upper plot of Figure 4.33.

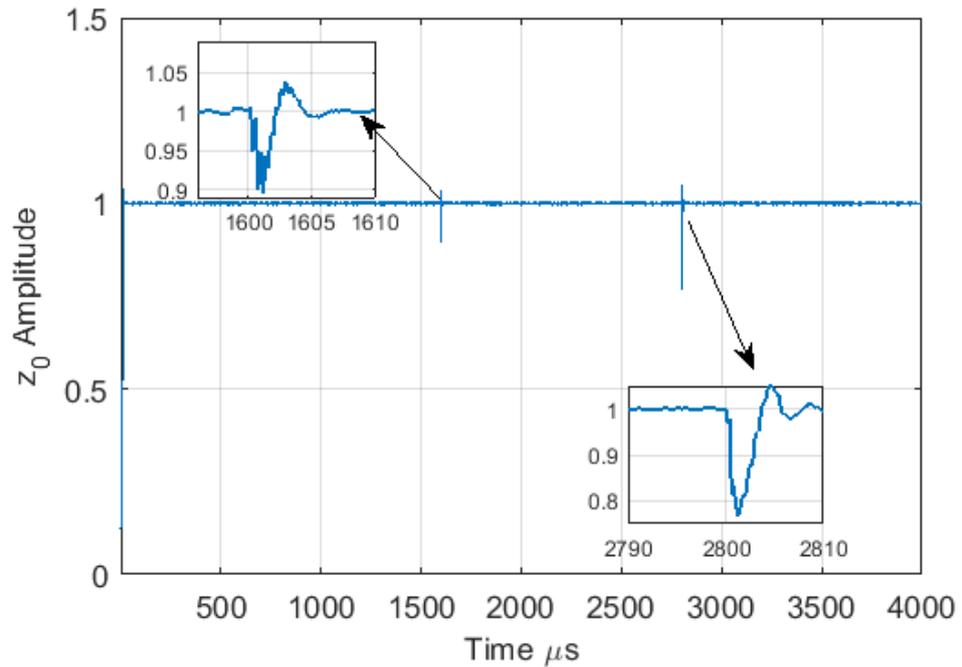


Figure 4.34 Convergence of magnitude of z_0 for hopping CWI

As the second-order filter tracked three frequencies (1 MHz, 1.25MHz, and 0.75 MHz), the simulation shows two glitches, one small and the other large, as reported in Figure 4.34. These are points where the notch's centre frequency switches or adapts to the subsequent frequency. The difference between each successive frequency is 0.25 MHz and 0.5 MHz. Hence it can be safely assumed larger difference between successive frequencies results in a higher amplitude of the glitches. The respective frequency response of the adaptive notch filter before and after the first glitch and after the second glitch is generated from the plot of the magnitude of z_0 and illustrated in Figure 4.35. The Blue, Red and Brown dashed line represents the second-order notch filter response when the adaptive parameter z_0 was at 1MHz, 1.25MHz and 0.75 MHz, respectively. So far, all these results in previous sections are conducted while keeping the pole contraction parameter k_a mainly between 0.8 to 0.9. This parameter controls the distance between the pole and zero on the unit circle, in other words, varying the bandwidth of the notch. Assuming all other conditions are kept the same and the convergence analysis for two different values of k_a is carried out and illustrated in

Figure 4.36. The simulation results in Figure 4.36 are obtained for k_a equal to 0.8 and 0.9, these

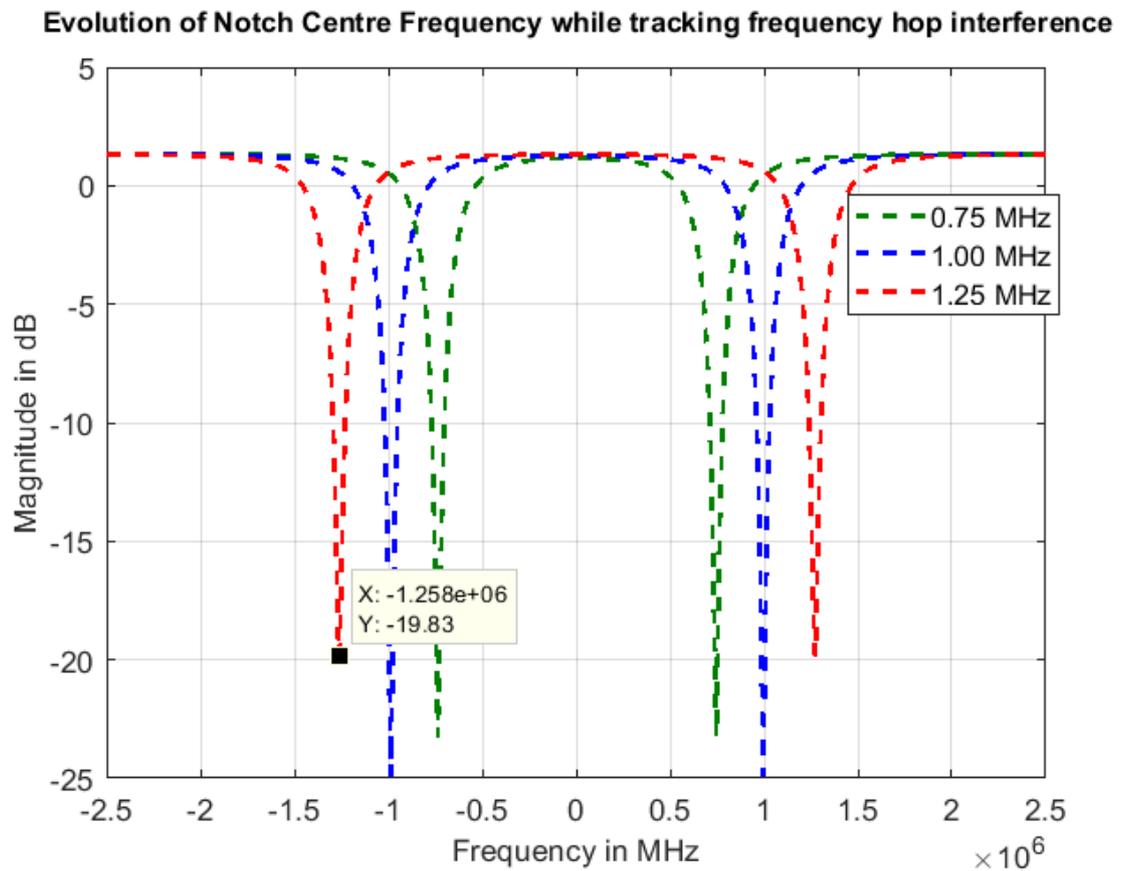


Figure 4.35 Evolution of notch's centre frequency while tracking 1 MHz, 1.25MHz, and 0.75 MHz frequency hopped CWI interference

results indicate that the convergence rate of the $|z_0|$ depends upon the bandwidth of the notch. As the notch bandwidth gets wider, the adapted parameter converges faster; for a narrow notch, convergence is slower. This means a narrow notch is slower to locate narrowband CWI interference. Figure 4.36 clearly shows the time taken by a narrow notch much more than that of a wide notch. Low value of k_a correspond to wide notch and high value of k_a is utilized to implement a narrow notch.

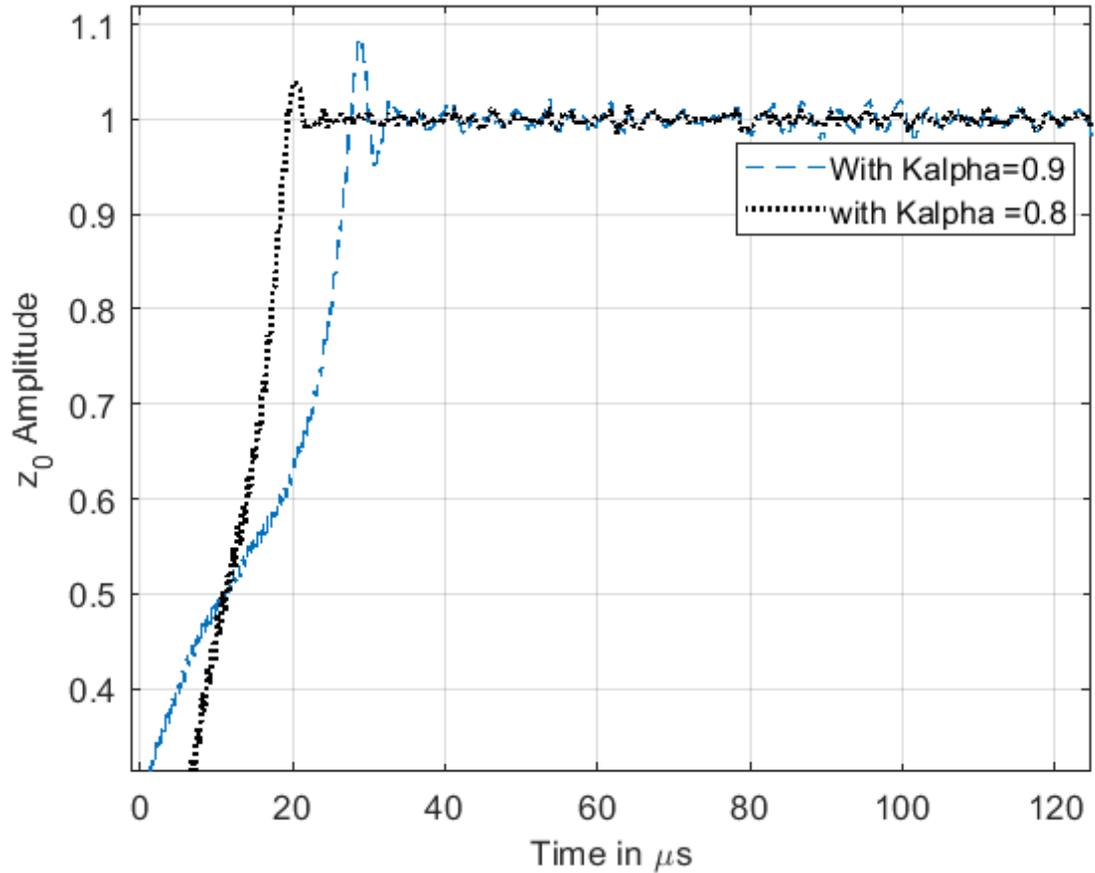


Figure 4.36 Convergence of the absolute value of z_0 in the second-order notch filter for value of k_α 0.8 and 0.9

4.9 Power of the Adaptation Parameter z_0

The parameter z_0 the complex coefficient for both complex first-order and real second-order notch filters can be referred to as the centre of gravity of the adaptive system. Figures 4.30, 4.37 and 4.38 represent the evolution and adaptation of the modulus of z_0 for complex CWI, complex chirp type interference and behaviour of $|z_0|$ for different values of JNR, respectively. It cannot only be used to detect interference and mitigation, as mentioned in [48-49] but also to estimate the power of incoming interference signal JNR. By analysing the simulation result in Figure 4.37, it is observed that while tracking fixed frequency CWI interference, $|z_0|$ tend to converge to unity when fixed CWI interference is present, but it fluctuated around unity. The level or range of fluctuation strongly depends on the level of

JNR. Figures 4.37 and 4.38 illustrate the convergence curves of $|z_0|$ for different values of JNR.

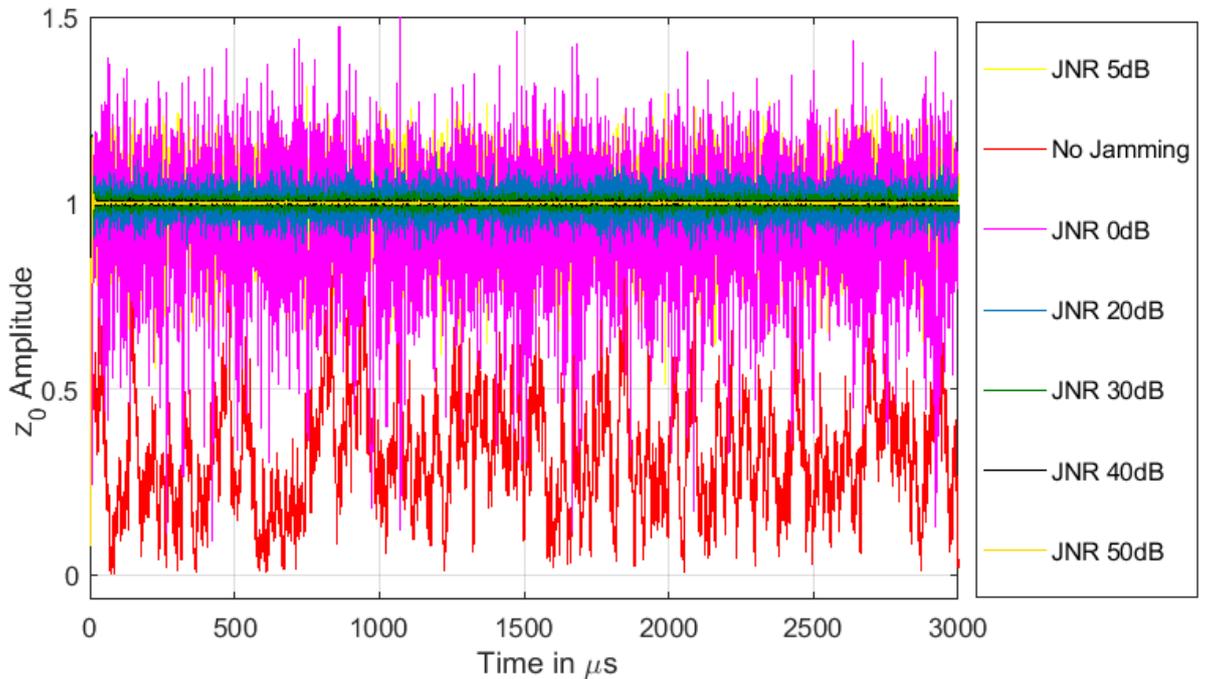


Figure 4.37 Convergence of magnitude of $|z_0|$ for different level of JNR

Different levels of fluctuation of the magnitude of $|z_0|$ is more prominent in Figure 4.38 (a), which shows how $|z_0|$ is eventually converged to unity for different levels of JNR but has different variances about unity. The red curve in Figures 4.37 and 4.38 represents the behaviour of convergence of the modulus of $|z_0|$ when there is no jamming/interference signal present, as shown in the simulation red curve never converges to unity. In other words, it means no CWI interference is present. Another observation from Figure 4.38 (a) is that when the JNR level is 0 dB (the pink curve), the modulus of the parameter z_0 does try to converge to unity, but the level of fluctuation is higher than any of the curves in Figure 4.38(a). As the level of JNR increases, the respective fluctuation for each successive JNR level dampens and decreases in amplitude.

The simulation results in Figure 4.38 (b) indicate that the rate of convergence of the modulus of z_0 is different for different levels of JNR when the rest of the variables in the simulation are kept constant. When interfering power is more, less number of iterations taken by the NLMS algorithm to lock the target frequency and modulus of z_0 converges faster. By keeping the rest of the parameters the same or constant, it can be deduced from the simulation result in this section and Figure 4.30 the convergence of the modulus of z_0 primary depend on the following factors:

- The power of narrowband CWI interference
- The pole contraction parameter k_a

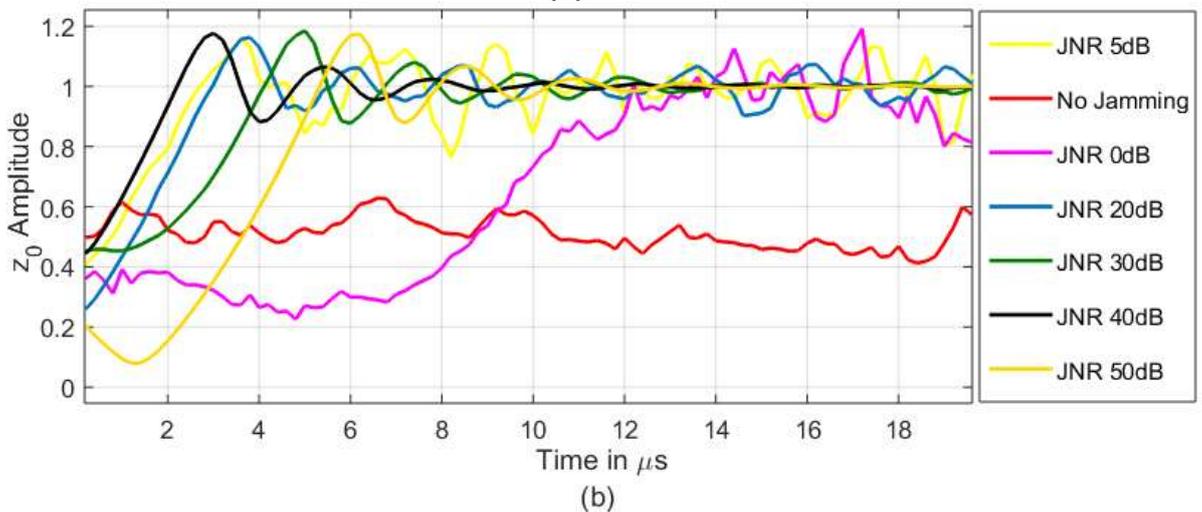
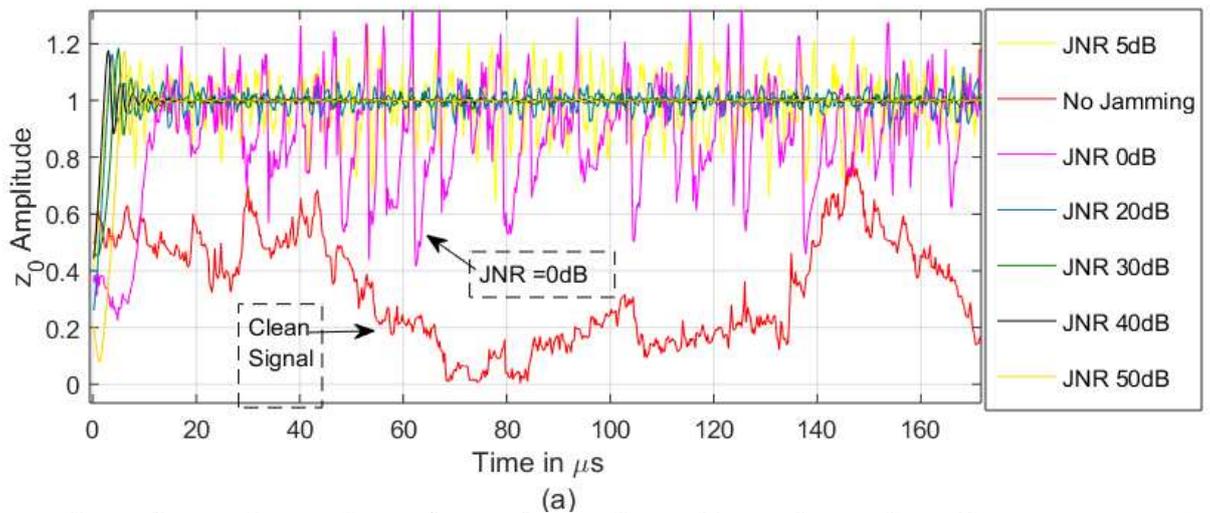


Figure 4.38 (a) and (b) zoomed version of Figure 4.37 for detailed analysis

Summarising the Importance of Parameter $|z_0|$

- *Instantaneous Frequency:* z_0 is the complex number denoted by $x + jy$ the instantaneous frequency for notch centre frequency is given at any point of time by

$$\theta = \tan^{-1} \left(\frac{x}{y} \right) \quad (4.37)$$

- *Detection of Interference:* After the convergence of the magnitude of $|z_0|$ the mean of $|z_0|$ can be used to detect the presence of interference. $|z_0|$ It only converges when interference is present.

$$\text{mean of } |z_0| = \mu_{\text{mean-}|z_0|} = \frac{1}{N} \sum_{n=0}^{N-1} \text{abs}(z_0[n]) \cong 1 \quad (4.38)$$

- *Convergence Analysis of Adaptation Algorithm:* The plot of the magnitude of $|z_0|$ against time or number of iterations gives a good overview of how fast $|z_0|$ is converged when different variables in the algorithm are altered, providing insight into the algorithm's optimisation.
- *Proposed: JNR Level Estimation:* As $|z_0|$ always fluctuate about its mean value whenever CWI interference is present in the useful signal, and these fluctuations differ for different levels of JNR and hence the variance of $|z_0|$ can be used to estimate the power level of JNR.

$$\sigma_{|z_0|}^2 = \frac{1}{N} \sum_{n=0}^{N-1} \text{abs}(z_0[n])^2 - \mu_{\text{mean-}|z_0|} \quad (4.39)$$

Further literature review and different interference power estimation algorithms and techniques are understudies. The proposed interference power estimation is based on the variance of the modulus of z_0 about unity, still in its initial stages. More work needed to be done to develop this concept into a JNR-level estimation properly. Figure 4.39 shows concepts of how the variance of the modulus of the parameter z_0 can be developed as a function of the power of the jamming signal.

$$f(\sigma_{|z_0|}^2) = \text{JNR Level} \quad (4.40)$$

These concepts require further in-depth mathematical modelling, derivation and formulation to set $\sigma_{|z_0|}^2$ as a function of JNR estimation. There are many factors on which the variance of $\sigma_{|z_0|}^2$ might depend. In these simulation results, the majority of the parameters and variables were kept constant, and a single parameter at a time was changed while conducting the experiments.

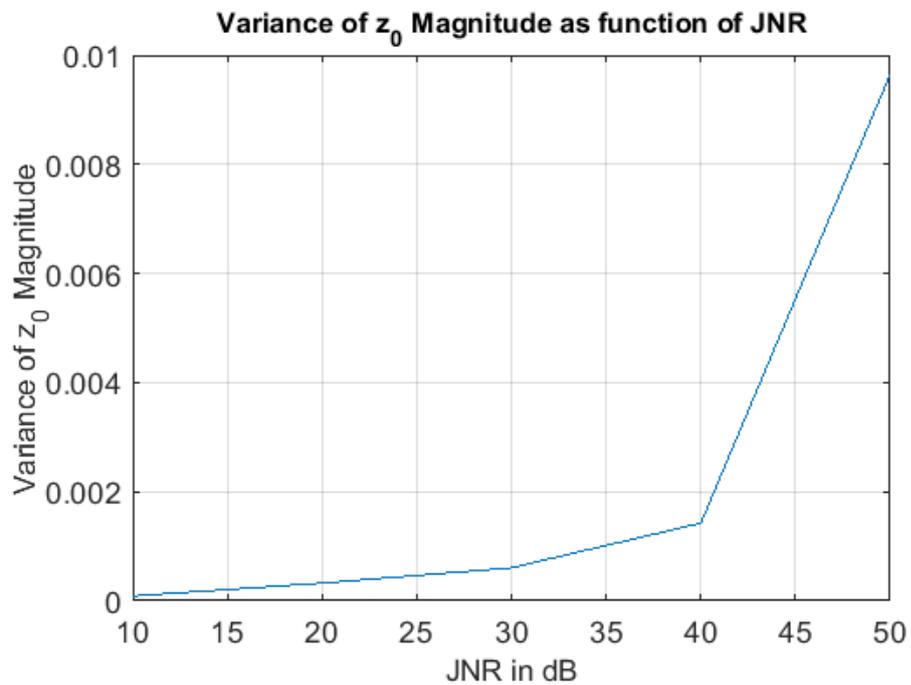


Figure 4.39 Results for setting up the threshold to activate the required filter depending on the variance of the magnitude of z_0

4.10 Proposed System Model and Algorithm

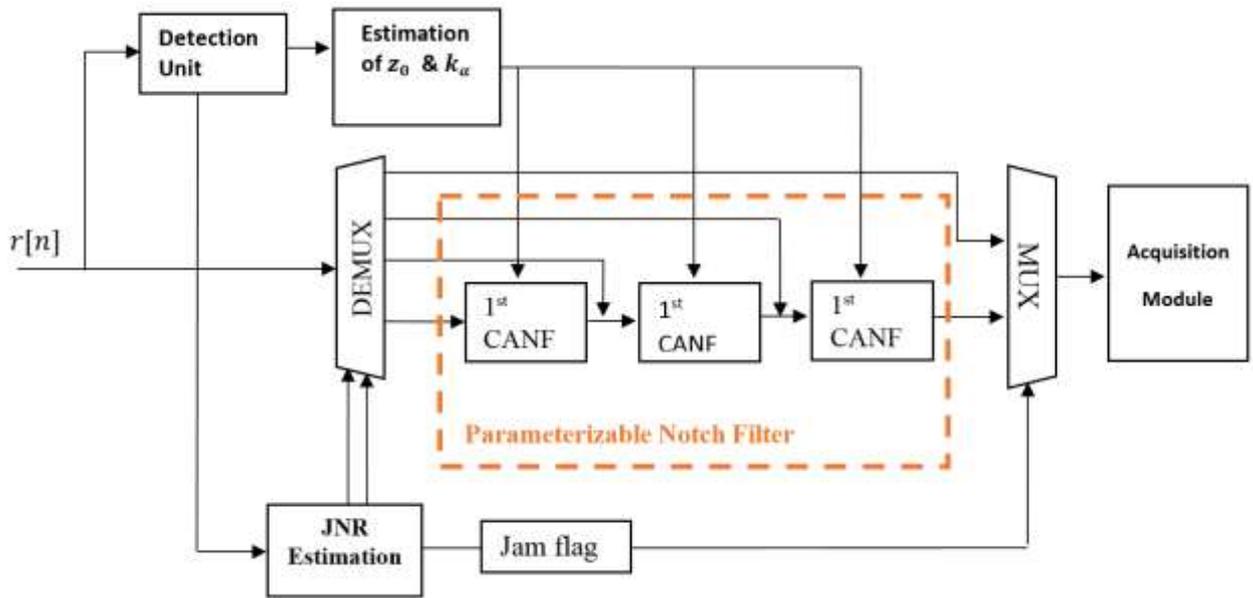


Figure 4.40 Proposed System-Level Model

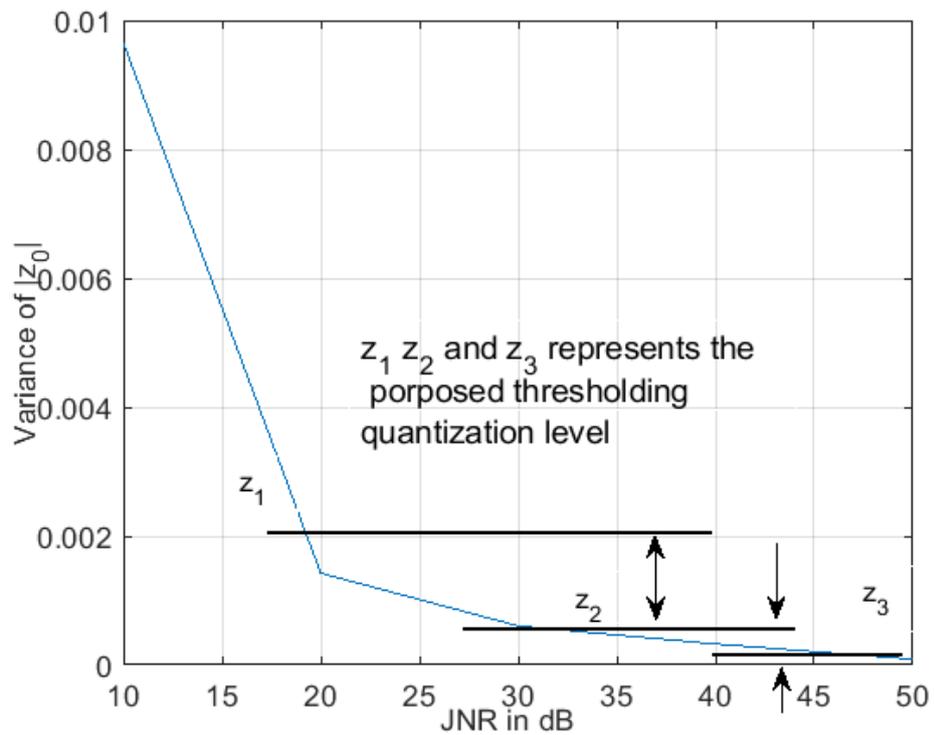


Figure 4.41 Setting up the threshold to activate the required CANF filter depending on the variance of the magnitude of z_0

PROPOSED JNR LEVEL ESTIMATOR

ALGORITHM I: JNR Level Estimation ($N_p, |z_0|, z_1, z_2$ and z_3)

Collect N_p number of samples of adaptive parameter $|b_0|$

Step 1: Compute mean of $|z_0|$ over N_p number of samples

Step 2: Determine $\sigma_{|z_0|}^2$ the variance of $|z_0|$

$$\sigma_{|b_0|}^2 = \frac{1}{N_p} \sum_{n=0}^{N_p-1} \text{abs}(z_0[n])^2 - \text{mean}(|z_0|)$$

Now

If $\sigma_{|b_0|}^2 < z_0$
 then DEMUX_{ctrl} 00 (No CANF filtering)
elseif $z_0 < \sigma_{|z_0|}^2 < z_1$
 then DEMUX_{ctrl} 01 (1st Order)
elseif $z_1 < \sigma_{|b_0|}^2 < z_2$
 then DEMUX_{ctrl} 10 (2st Order)
elseif $\sigma_{|b_0|}^2 > z_3$
 then DEMUX_{ctrl} 11 (3st Order)

The z_1, z_2 and z_3 represent the threshold levels for the DEMUX to pass on the signal either to 1st, 2nd or 3rd CANF. Magnitude of z_0 is the threshold for the detection of incoming interference. If the mean of $|z_0|$ approximate to unity, it indicates the presence of CWI in the useful signal. The absolute value of z_0 , i.e. $|z_0|$, fluctuates about its mean value whenever CWI interference is present in the useful signal, and these fluctuations differ for different levels of JNR and hence the variance of $|z_0|$ can be used to estimate the power level of JNR. The level or range of fluctuation strongly depends on the level of JNR. Figure 4.38 (a) and (b) illustrate the convergence curves of $|z_0|$ for different values of JNR. Different levels of fluctuation of the magnitude of $|z_0|$ are more prominent in Figure 4.38 (a), which shows how $|z_0|$ eventually converges to unity for different levels of JNR have

different variances about unity. Another observation from Figure 4.38 (a) is that when the JNR level is 0 dB (the pink curve), the modulus of the parameter z_0 does try to converge to unity, but the level of fluctuation is higher than any of the curves in Figure 4.39 (a). As the level of JNR increases, the respective fluctuation for each successive JNR level dampens and decreases in amplitude. The simulation results in Figure 4.38 (b) indicate that the rate of convergence of the modulus of z_0 is different for different levels of JNR when the rest of the parameters in the simulation are kept unchanged. When interfering power is more, less number of iterations taken by the NLMS algorithm to lock on the target frequency and the modulus of z_0 converges faster. Keeping the rest of the parameters constant, it can be understood from the simulation results in this section that the convergence of the modulus of z_0 primary depend on two factors, the power of narrowband CWI interference and the pole contraction parameter k_α . As $|z_0|$ fluctuate about its mean value whenever CWI interference is present in the signal, and these fluctuations differ for different levels of JNR; the variance of $|z_0|$ can be used to estimate the power level of JNR. The proposed interference power estimation is based on the variance of the modulus of b_0 about unity. Figure 4.41 shows how the variance of the modulus of the parameter z_0 can be used to estimate the power of the jamming signal and JNR. The labels z_1, z_2 and z_3 in Figure 4.41 represent three different threshold levels for 1st, 2nd and 3rd CANF to be used in the proposed system level model.

4.11 Conclusions

This chapter presents the interference detection and mitigation techniques via first-order complex adaptive notch filters and second-order real notch for GPS-based applications. Different types of jamming signals, such as narrowband CWI, chirp-type interference and frequency hopping signals, are modelled in MATLAB and incorporated with a GPS signal generator designed in Chapter 3 of this research work.

It also provides a detailed analysis of fixed complex notch filters and their performance in terms of interference suppression. Complex notch filters of orders one to six are tested against different levels of JNR. Each filter's performance is evaluated in terms of SNR at the output of the acquisition module. As the order of the CNF increases, the subsequent filter's interference mitigation ability enhances. By doing this, the computational complexity double. The output SNR of the 3rd and 4th order is almost similar, but there is a slight improvement in the SNR of the 4th order notch filter. Complex notch filter of order 5th and 6th gives superior performance for interference mitigation for the rest of the notch filters. 5th Order CNF can mitigate interference up to 60dB of JNR, whereas the 6th order CNF mitigation dynamic range is almost 80dB (JNR). While 1st, 2nd and 3rd order CNF have mitigating dynamic ranges of 21dB, 32dB and 51dB, respectively.

Further, this design was extended to adaptive notch filters, which can autonomously detect and mitigate narrowband CWI and Chirp-type interference. Analysis of the notch bandwidth is conducted as well to determine the optimal notch width (k_a) as the size of the width of the notch control the convergence rate of parameter z_0 and determine how fast target frequency can be located.

There is no point in using a higher notch filter for the low level of JNR, which consumes more power. A simple and less power-hungry 1st order notch filter can be used to mitigate

low-level interference. A simple and innovative system-level model is proposed utilizing multi-stage CANF efficiently with a threshold setting of JNR estimation. The threshold setting parameter provides a trade-off between the effective excision of CWI, the order of the filter, and power consumption. It results in a computationally effective solution for interference mitigation for GNSS-based applications. Different aspects and properties of the parameter z_0 is exploited which could be beneficial for the estimation of JNR levels. The variance of the magnitude of the parameter z_0 is proposed to be set as a threshold setting variable to estimate the level of JNR. It can be employed to toggle between different orders of the filter in the proposed system depending on the level of JNR, as explained in section 4.10 of this thesis.

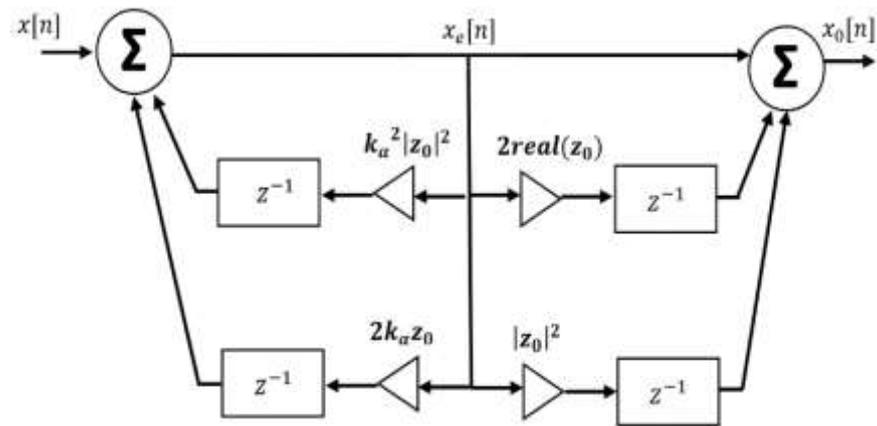
Chapter 5

ADAPTIVE IIR NOTCH FILTERS FOR INTERFERENCE MITIGATION TO IMPROVE SIGNAL ACQUISITION PERFORMANCE

5.1 Introduction

This chapter compares the direct form IIR notch filter and Lattice form IIR notch filter regarding convergence speed and tracking ability of hop frequency interference for GPS-specific applications. It also presents the simulation results of these two adaptive notch filters, the first real 2nd IIR direct form notch filter (Figure 5.1) and the second lattice-based notch filter (Figure 5.2). A modified, fully adaptive lattice-based structure with constraints adaptation of the notch bandwidth parameter is presented in this chapter. A full gradient term is derived from adapting the notch bandwidth parameter ρ . Finally, the proposed algorithm is compared to the existing methods [48] [67] and [68] from the open literature on adaptive notch filtering. The performance of each algorithm is determined under various conditions considering different data quantization levels, Jamming to Noise Density (J/N_0) versus effective Carrier to Noise Density (C/N_0) at the output of the correlator.

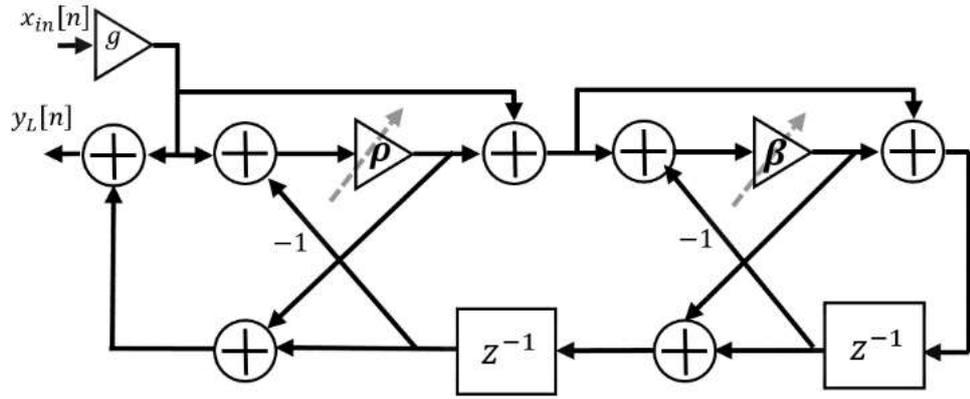
The proposed Fully Adaptive Lattice Notch Filter is based on the simultaneous adaptation of two coefficients to control the notch's centre frequency and bandwidth. It is demonstrated here that the proposed filter has a superior tracking ability and convergence speed against existing adaptive notch filtering methods.



Transfer Function

$$H_{Dnotch}(z) = \frac{(1 - 2\text{Real}(z_0)z^{-1} + |z_0|^2 z^{-2})}{1 - 2k_\alpha \text{Real}(z_0)z^{-1} + k_\alpha^2 |z_0|^2 z^{-2}}$$

Figure 5.1 2nd Order Direct Form IIR Notch Filter and its transfer function



Transfer Function

$$H_L(z) = \frac{1 + \rho}{2} \frac{1 - 2\beta(n)z^{-1} + z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}$$

Figure 5.2 Fully Adaptive Lattice-Based Notch Filter and its transfer function

In this chapter, three different adaptation algorithms, selected from the open literature, are analysed and modelled in MATLAB, which are:

- I. Normalized Least Mean Square (LMS) error based direct form 2nd Order filter Structure (described in Chapter 4) from [51] and [67].
- II. Signed Magnitude Variable step-size LMS based Direct form 2nd Order filter structure from [68] and
- III. Variable step-size LMS-based Lattice-based structure from [48].

Extensive research is available regarding adaptive notch filters for tracking single and multi-tone sinusoidal signals [69-80] and removing these unwanted tones from the useful signal. The research work carried out in [80] [85-86] is most relevant to this work and provides a strong basis and good reference for comparison. But on the other hand, very few papers were published on the utilization of an ANF for the removal of interference from GPS L1 signal or GNSS signal[48-49][51][68], while [83-84] are more concentrated on general DS CDMA type signal. Most of these algorithms focus on single-tone CWI interference. Only a few

have adapted notch bandwidth parameters, such as [74] and [81-82]. Second-order IIR notch filters of various structure types, such as all-pass, lattice and direct form, are extensively used in these research publications for various applications.

5.2 Adaptation of the Notch Bandwidth Parameter

In Chapter 4, only a single parameter is adapted in the CANF, which was the notch centre frequency. It was observed that when the notch is wider, the CANF tracks down the target frequency more quicker. The notch width is narrow, so the notch filter is slower to locate the target frequency. This observation led to the need for a literature review on adaptive the notch bandwidth parameter. It was noted that very few research works were carried out relevant to updating the notch bandwidth parameter in adaptive notch filtering. The following paragraphs present a review of a few selected works.

In [81-82], the authors developed ANF, which adapts the notch bandwidth parameter. The update equation for ρ is presented in this paper as follows.

$$\rho(n) = \rho(n - 1) - \mu_{\rho} y(n) \psi(n) \quad (5.1)$$

Where $\psi(n)$ is the gradient function. The transfer function of the filter, generating this gradient $\psi(n)$, is given as [64].

$$W(z) = \frac{\beta(n)z^{-1} - z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}} \quad (5.2)$$

Both [81] and [82] are published by the same authors and describe how adapting the notch bandwidth parameter via the LMS algorithm improves the SNR. Moreover, in [81], the authors showed the convergence of ρ to 0.76 and 0.88, rather than converging to a value

much closer to the unity (more narrower notch) and provide no explanation for why steepest descent algorithms are used to update the variable $\rho(n)$ as in equation (5.1).

In [80], the steepest descent algorithms are used to update the adaptive parameters of the filter. Still, this publication does not include any result on the enhancements or improvements in the performance of ANF for adapting the ρ , the notch bandwidth parameter.

During the literature review, it was noted that the authors use the all-pass notch transfer function in most of the research work, as shown in (5.3). This transfer function is also utilised in various other research works relevant to adapting bandwidth parameters [48] and [56].

$$H(z) = \frac{1+\rho}{2} \frac{1-2\beta z^{-1}+z^{-2}}{1-\beta(1+\rho)z^{-1}+\rho z^{-2}} \quad (5.3)$$

In [67], the author effectively drives the same update equations as in [81] (given in (5.1)) and limits the value of ρ between 0.70 and 0.96 and but doesn't show what ρ convergences. Furthermore, in chapter 10 of [46], it was shown how the magnitude response of direct form 2ndIIR notch filter alters unevenly when parameter ρ is adapted. It means ρ should only be adapted in an all-pass structure to ensure uniformity of gain in the pass-band region. The author acknowledges the benefits of the adaptation notch bandwidth and systematically narrows down notch bandwidth over time, but such a method is not applicable to hop frequency interference.

As a result of the literature review, it is concluded that

- I. None of these works shows how the notch bandwidth parameter behaves if hop frequency interference is injected into the system.
- II. There is no clarity on how the notch bandwidth parameter is constrained while adapting it.

- III. Narrower notch bandwidth takes longer to adapt to the target frequency, whereas wider notch bandwidth converges to the target frequency at a faster rate.

5.3 Adaptation of Notch Bandwidth Parameter via partial gradient

In this section, an analysis of the partial gradient is conducted, and the proposed full gradient with constraints is presented in section 5.4. The research work shown in [81] and [85] utilized the same partial gradient term to adapt the notch bandwidth parameter. It is also observed that the lattice structure proposed in [48] for anti-jamming of CWI for GPS signal uses the same equation as mentioned in [81] but only adapts a single parameter β (notch centre frequency), while ρ is kept constant. Interestingly [48], [81] and [85] are closely linked together with minor twigs and use a time-varying step size.

5.3.1 Partial Gradient Term for Adapting Notch Bandwidth Parameter

Method I [81]: In this first method, the partial gradient term in [81] and [85] is used to update the parameter ρ , and it is derived from the output of the all-pass-based notch filter. A partial gradient term is derived by assuming ρ is fixed in the numerator of (5.4), and the only denominator is differentiated with respect to ρ as follows

$$H_L(z) = \frac{N(z)}{D(z)} = \frac{1-2\beta(n)z^{-1}+z^{-2}}{1-\beta(n)(1+\rho(n))z^{-1}+\rho(n)z^{-2}} \quad (5.4)$$

where,

$$D(z) = 1 - \beta(1 + \rho)z^{-1} + \rho z^{-2} \quad (5.5)$$

Differentiating $N(z)$ with respect to ρ yields the following terms

$$\frac{\partial}{\partial \rho} \{D(z)\} = -\beta z^{-1} + z^{-2} \quad (5.6)$$

and are the internal state of the all-pass filter.

$$g_\rho = -\beta u[n-1] + u[n-2] \quad (5.7)$$

Where $u[n-1]$ and $u[n-2]$ are the internal state of the all-pass filter. The update equation for ρ , bandwidth parameter becomes

$$\rho[n] = \rho[n-1] - \mu_\rho y_L[n] g_\rho[n] \quad (5.8)$$

Where μ_ρ and y_L are fixed step-size and output of notch filter y_L . On the other hand, β is updated as follows

$$grad(g_\beta[n]) = (1 - \rho(n))u[n-1] \quad (5.9)$$

$$\beta[n] = \beta[n-1] - \mu[n] \cdot y_L[n] grad(J_\beta[n]) \quad (5.10)$$

$$\mu[n] = \frac{\mu_\beta}{\phi_\beta[n]} \quad (5.11)$$

$$\phi_\beta[n] = \gamma \phi_\beta[n-1] + (1 - \gamma) grad(J_\beta[n])^2 \quad (5.12)$$

Where ϕ_β , γ and μ_β are instantaneous power estimation of the gradient signal, forgetting factor (0.9 – 0.99) and fixed step size, respectively.

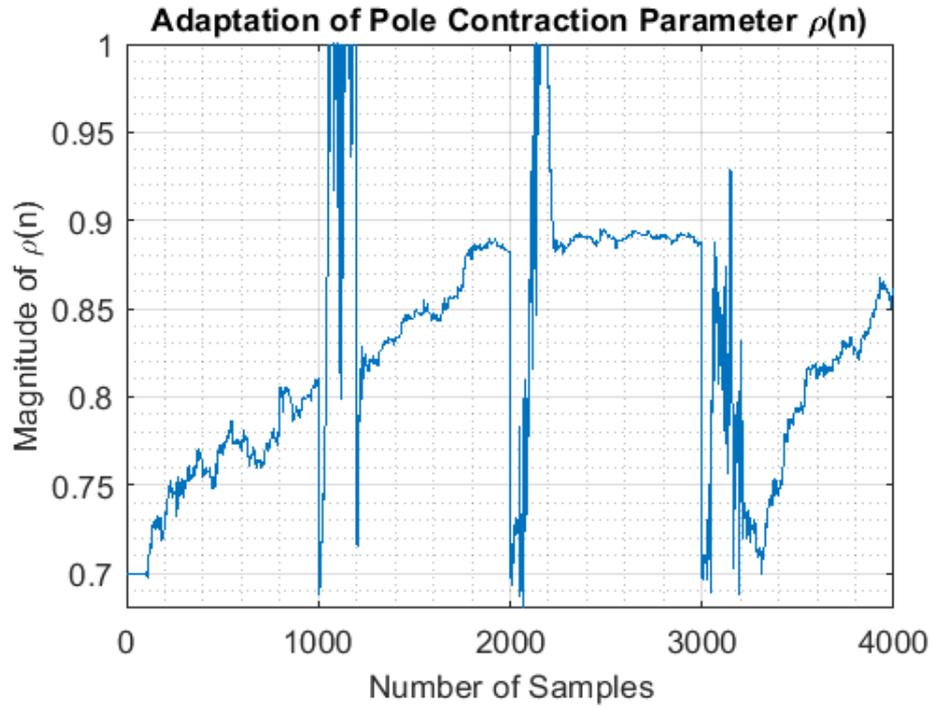


Figure 5.3 The adaptation of ρ via [81] in all-pass lattice notch filter

The above algorithm is modelled in MATLAB and tested for hop frequency interference. The tracking ability and convergence of both β and ρ are analysed using the simulation parameters in Table 5.1. A target signal with four different hop frequencies is generated with different levels of noise variance, as mentioned in Table 5.1. Each hop frequency is 1000 samples in length, as shown in Figure 5.4, represented by the green line.

Table 5.1 Simulation parameters for [81]

Noise variance σ^2	0.25 (6dB) and 0.04 (14dB)
μ_β step-size for β	0.018 (fixed)
μ_ρ step-size for ρ	0.008 (for 6dB) and 0.01 (for 14dB)
'N' no. of samples	4000
Initial value of β	0.25 (normalized frequency)
Initial value of ρ	0.70 (Pole contraction factor)
γ forgetting factor	0.90

From the simulation results in Figures 5.3 and 5.4, it is evident that Method-I performs poorly and ρ fails to converge to an optimum value. The noise variance was set to 0.25, meaning a hop frequency signal is immersed in a noisy environment. As shown in Figure 5.3, at the start, each new target frequency ρ converges extremely slowly, moreover, ρ tries to go beyond unity in between samples 2100-2400; thus, this can create instability. It is also observed that the β is tracking the target frequency with insufficient accuracy, and its ability to track the target frequency degraded, as shown in Figure 5.4. In between region 2100-2400 samples in Figure 5.4, β failed to track the target frequency (green line) as shown by the blue line. In the same region (between 2100 and 2400 samples) ρ try to overshoot as shown in Figure 5.3.

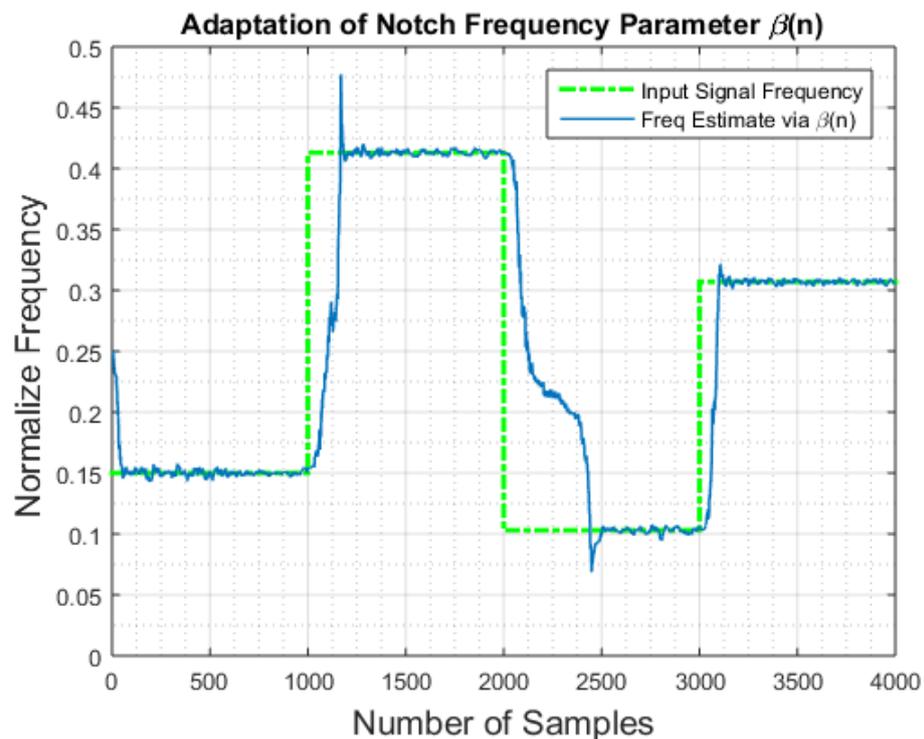


Figure 5.4 Tracking performance of all-pass based Notch filter in [81], with a noise variance of 0.25.

The second simulation set is obtained by further reducing the noise level and making the hop frequency interference more prominent. The noise variance was set to 0.04 this time, equivalent to an SNR of 14dB, as shown in Table 5.1. Also, to obtain these results, the step

size of both β and ρ is adjusted to update them, as mentioned in table 5.1. Again in this set of simulations ρ overshoots in between 2200 and 2300 samples only for a brief period of time as shown in Figure 5.6, but ρ convergence to 0.90 from an initial value of 0.70 is extremely slow which is visible in the first 1000 samples.

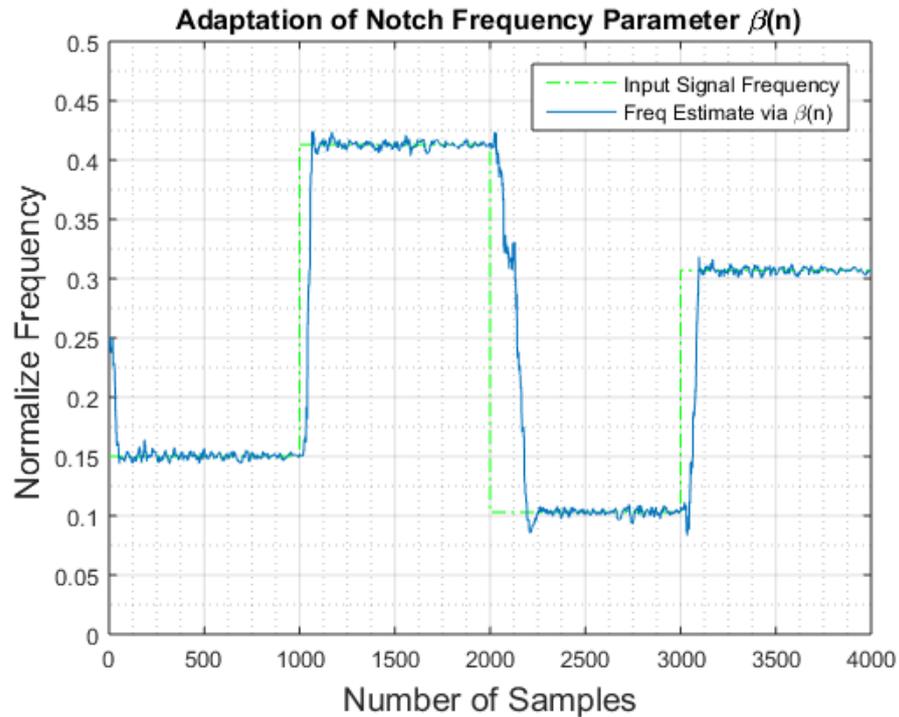


Figure 5.5 Tracking performance of all-pass based Notch filter in [81], with a noise variance of 0.04

Figure 5.5 contains an artefact on the plot, which occurs when the gradient's magnitude approaches zero. At this moment, the output of the adaptive notch filter oscillates between two values.

In both of these sets of simulations with a variance between 0.25 and 0.04, the ρ was hard reset to 0.70 after every 1000 samples. In real-life situations, this has to be done automatically by ANF before tracking of next subsequent frequency to enable fast tracking.

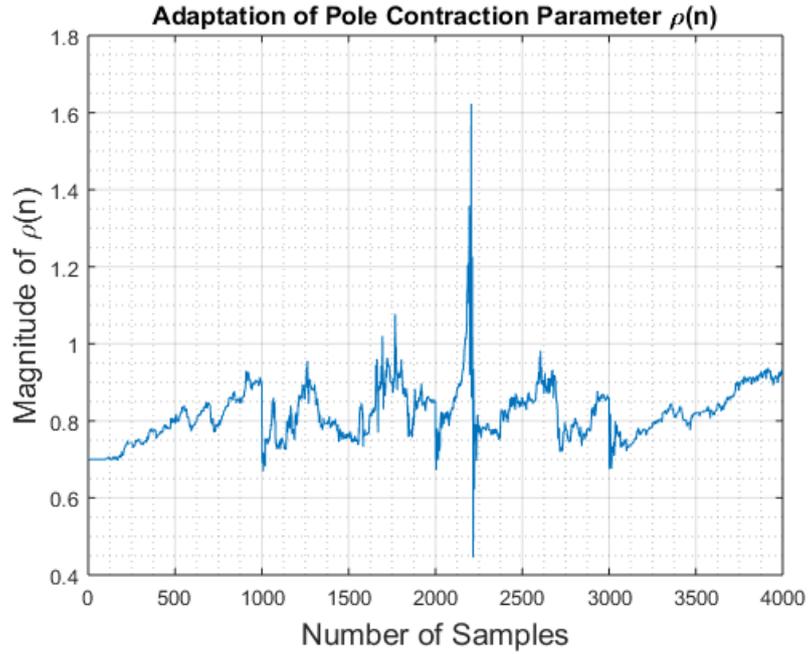


Figure 5.6 The adaptation of ρ with a variance of 0.04

The observations on Method-I can be summarised as follows

- I. The parameter ρ must be constrained in order to obtain a reliable adaptation of the bandwidth of the notch
- II. The bandwidth parameter ρ should not approach unity. Otherwise, the system introduces instability, and vice versa parameter ρ should not be too small (-0.5 to 0.2) not to distort the useful signal.
- III. The partial gradient method shows poor performance whilst adapting ρ and it is too simplistic to track the hop frequency signal.

Method II [68]: Work published in [68] utilizes 2nd Direct Form IIR notch filter structure using a sign-magnitude LMS algorithm which becomes a point of interest due to its simplicity and less computational complexity. In [68], only β is adapted, while notch bandwidth parameter ρ is kept fixed at 0.90. The update equation for β is as follows [68]

$$\beta[n] = \beta[n - 1] - \mu[n] \cdot y_D[n] \cdot \text{sign} \left[\frac{g_1[n]}{1 + |g_1[n]|} \right] \quad (5.13)$$

Where $\mu[n]$ is fixed step-size, y_D is the output of the filter and $g_1[n]$ is gradient signal given as

$$g_1[n] = x[n - 1] - \rho y_D[n - 1] \quad (5.14)$$

Transfer function as mentioned in [68]

$$H_D(z) = \frac{1 - \beta(n)z^{-1} + z^{-2}}{1 - \rho\beta(n)z^{-1} + \rho^2 z^{-2}} \quad (5.15)$$

The same method was tested, and the hop frequency signal was generated with 4000 samples. Every 1000 samples signal is switched to a different frequency. Again, here variance of noise was set to 0.25, the value of μ was set to 0.008 and ρ was set to 0.90. The value of μ is empirically set to achieve the best possible results.

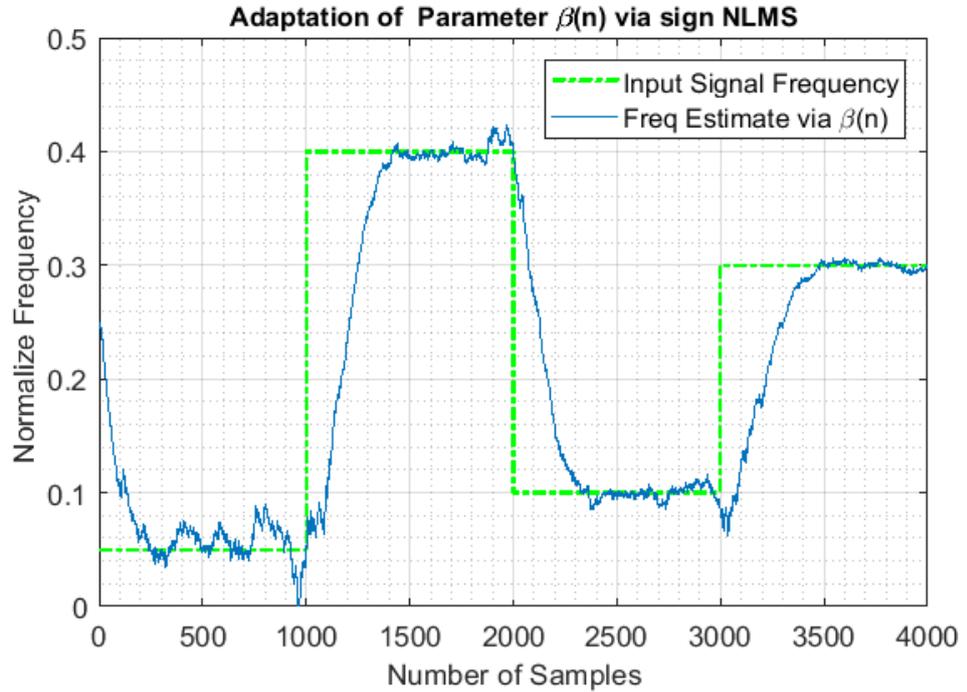


Figure 5.7 Sign Magnitude LMS for tracking the hop frequency signal [68]

Figure 5.7 represent the simulation results of [68] modelled in MATLAB. Due to the algorithm's simplicity, a large amount of performance is lost, failing to track and converge to the target frequency. Hence, removing hop CWI interference from the GPS L1 signal was

not further pursued via [68]. Furthermore, the 2nd Order Direct Form adaptive notch filter in chapter 4 [51] is also tested, and its complete algorithm is described in the previous chapter. Which utilizes complex variable z_0 to adapt target frequency, again in this case, as well as notch bandwidth k_α is fixed, and the step size is normalized.

Figure 5.8 shows the simulation results for adapting the hop frequency signal and the effect of the different values of k_α is demonstrated by observing the red and black curve in Figure 5.8, where the value of k_α is small, meaning a wider notch; ANF with a wide notch quickly locates the target frequency. On the other hand, when k_α is 0.9, the notch narrower, and adapts much slower to locate the target frequency, as presented by the black curve. Table 5.2 compares the number of samples taken to lock onto subsequent hop frequency. From the values in table 5.2, it can be said that the narrower notch track target frequency is 1.7 times slower than the wider notch.

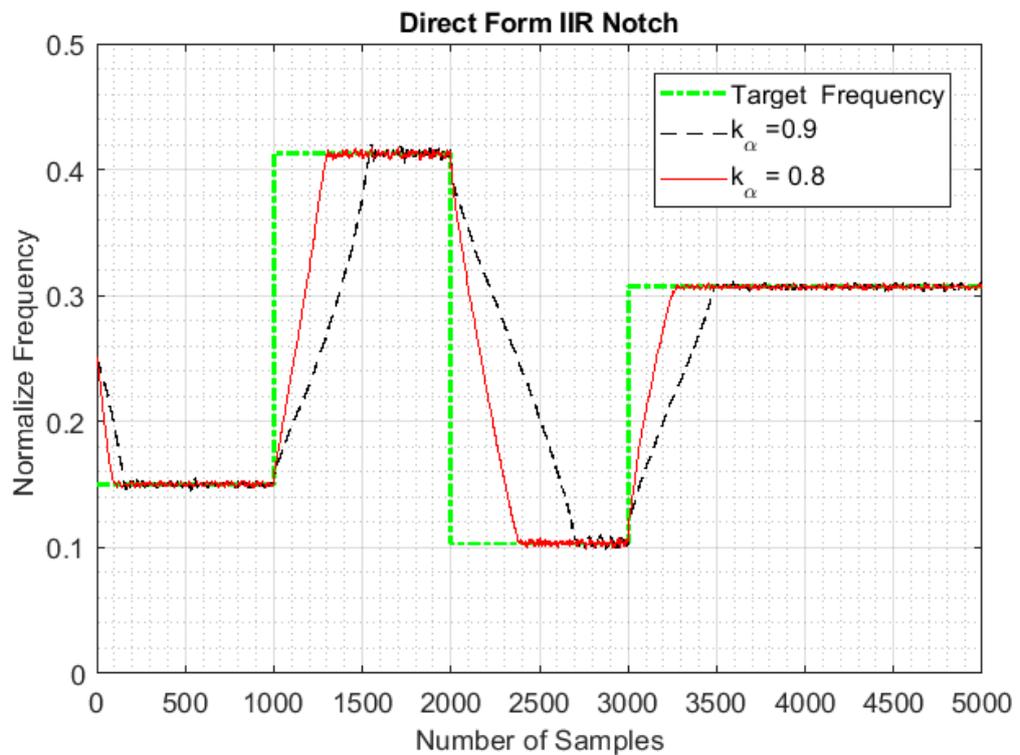


Figure 5.8 Tracking and Convergence performance of Direct Form IIR Notch filter [68]

Table 5.2 Number of samples required to lock on the subsequent target frequency

	f_{1hop}	f_{2hop}	f_{3hop}	f_{4hop}
$k_\alpha = 0.8$ 'Wider Notch'	100	300	400	280
$k_\alpha = 0.9$ 'Narrower Notch'	160	540	700	500

The observations on Method-II can be summarised as follows

- I. Assuming a hopping frequency signal is being tracked without delay or in fewer samples, a stronger performance can be attained by setting the initial bandwidth parameter to a small value for each subsequent hop frequency and then adapting the notch bandwidth parameter to narrow down the notch bandwidth.
- II. Both [34 and 51] 2nd order Direct Form Adaptive IIR Notch Filter algorithms show limitations in terms of performance while tracking hop frequency signal with fixed notch bandwidth parameters.

5.3.2 3D Frequency Response Analysis and the Cost Function

In this section, further 3D frequency domain analysis of both direct form structure and all-pass lattice-based ANF filter is carried out to gain more understanding and provide valuable insight into the limitation and functionality of an ANF. Furthermore, the study is carried out to demonstrate how the frequency response of each filter alters under the following conditions:

- I. Notch bandwidth parameter ' ρ ' for both structures varies from -1 to 1
- II. Frequency response when target frequency is close to DC or Nyquist.
- III. Cost Function of each of the filter

5.3.3 2nd Order Direct Form IIR Notch Filter

Transfer Function of Direct Form IIR notch filter used in [51][67-68]

$$H_D(z) = \frac{1 - \beta(n)z^{-1} + z^{-2}}{1 - \rho\beta(n)z^{-1} + \rho^2z^{-2}} \quad \text{Where } \beta = -2\cos(2\pi f_{CW1}) \quad (5.16)$$

The 3D frequency response is given by taking the FFT of the output of (5.15), which is $FD(\rho)$, where β is kept constant.

$$FD(\rho) = \sum_{\rho=-1}^1 \{ \sum_{n=0}^{N-1} y_D(n, \beta, \rho) \} \quad (5.17)$$

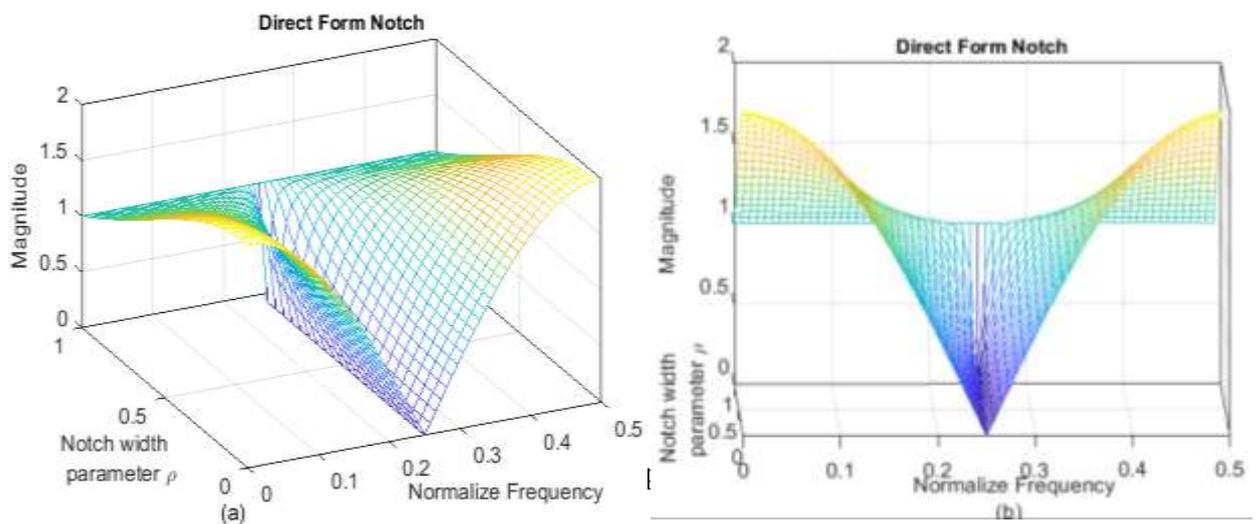


Figure 5.9 (a) 3D Frequency Response of Direct Form IIR Notch Filter for when values of ρ sweep between 0 and 1. (b) Front view of 3D frequency response

One thing common in all these simulations is that, as the value of the notch bandwidth parameter ρ changes from 1 to 0, it alters the gain levels in the pass-band region. When ρ is between 0.60 and 0.80 (Figure 5.10 (a)), the direct form structure alters its magnitude response very slightly in the pass-band region. Though there is also scaling of magnitude response present between 0.80 and 0.90, it is very less in magnitude.

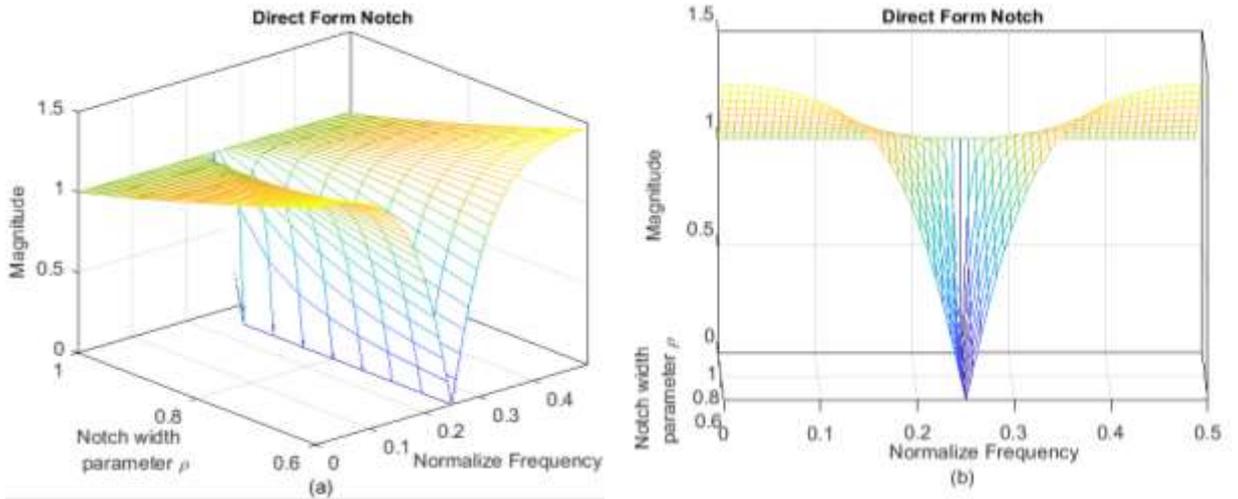


Figure 5.10 (a) 3D Frequency response for the value of ρ between 0 to 0.6. (b) The front view of 3D Frequency response in (a)

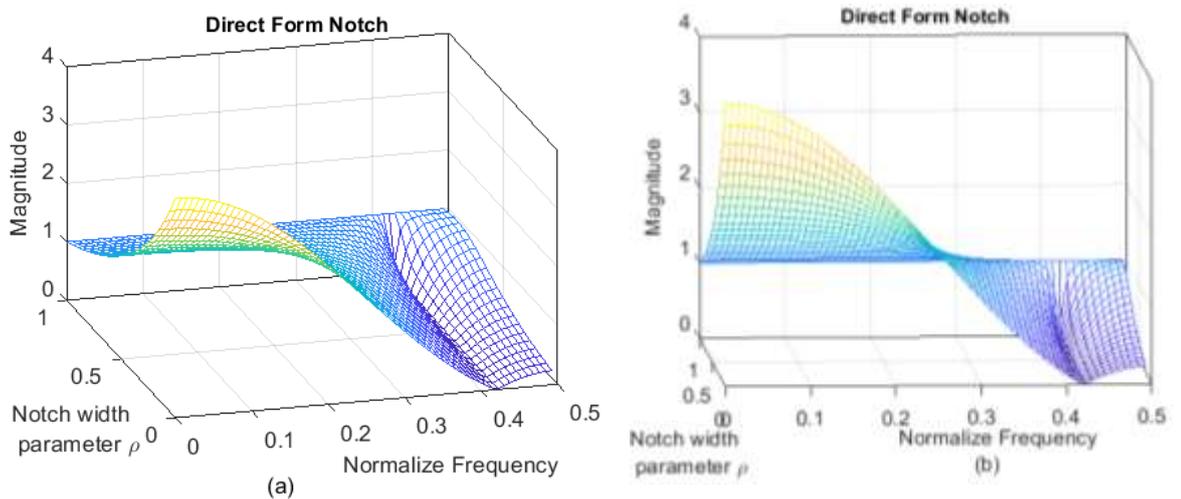


Figure 5.11 (a) More gain scaling if target frequency is closer to Nyquist. (b) Front view of Figure 5.11 (a)

In Figure 5.11(a), it can be observed that as the value of ρ sweeps from 1 to 0, the notch gets wider, as expected. Still, the target frequency close to Nyquist is worse and uneven compared to the target frequency located at half Nyquist, as shown in Figure 5.9(a). Even if the notch bandwidth parameter ρ is adapted in this type of structure, it has to be constrained between 0.80 to close to unity. For example, a target frequency close to Nyquist is being tracked by a direct form ANF filter while tracking. If the value of ρ becomes lower than 0.80, it severely alters the signal in the pass-band region due to a rise in the magnitude of the gain in the passband region, as shown in Figure 5.11(b). Hence algorithm has to be designed and

developed in such a way that it guarantees ρ never becomes smaller than 0.80, which is a very stringent condition for any adaptive algorithm.

Furthermore, in the previous section, it was demonstrated that if the value of ρ is small, it means the notch width is wider, hence locating the target frequency at a faster rate. However, given if one can initialise a notch filter with a value of ρ below 0.80, such as 0.70 or even 0.60, which means more quickly the ANF locate the target frequency. If the value of ρ is set below 0.80, the direct form structure suffers from scaling of the magnitude response in the pass-band region.

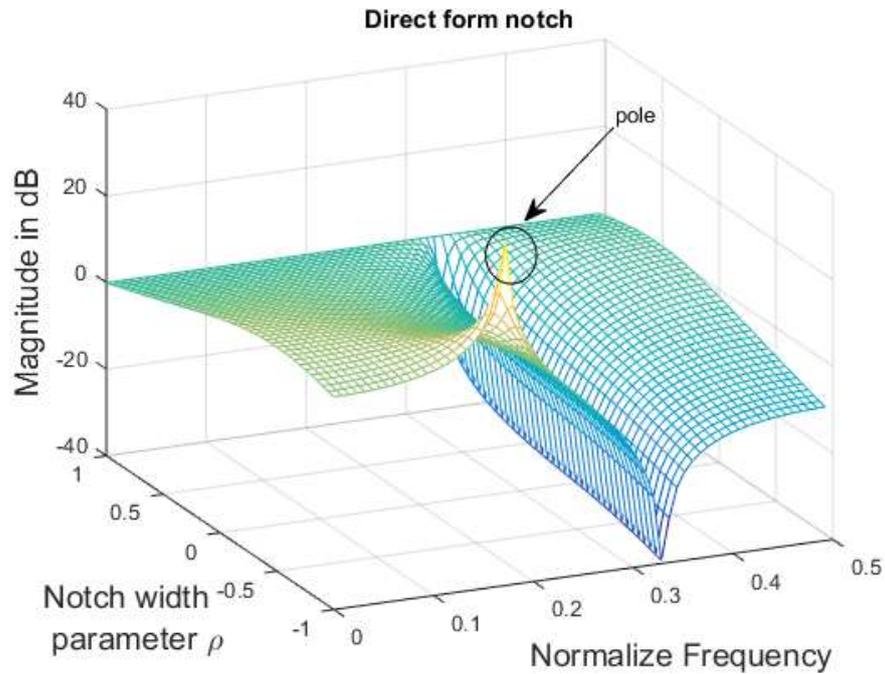


Figure 5.12 3D frequency response when ρ is a sweep from -1 to 1, as ρ approaches -1, a pole is introduced as shown in the figure

5.3.4 All-Pass Lattice-based ANF

The transfer function as given in [48][81] and [85]

$$H_L(z) = \frac{1+\rho}{2} \frac{1-2\beta(n)z^{-1}+z^{-2}}{1-\beta(n)(1+\rho(n))z^{-1}+\rho(n)z^{-2}} \quad \text{Where } \beta = -\cos(2\pi f_{CW1}) \quad (5.18)$$

Similarly, the 3D frequency response of the lattice is obtained by taking the FFT of the output of equation 5.17, which is $FL(\rho)$, where β is kept constant. The input signal to the system is a unit impulse signal with length N.

$$FL(\rho) = \sum_{\rho=-1}^1 \{ \sum_{n=0}^{N-1} y_L(n, \beta, \rho) \} \quad (5.19)$$

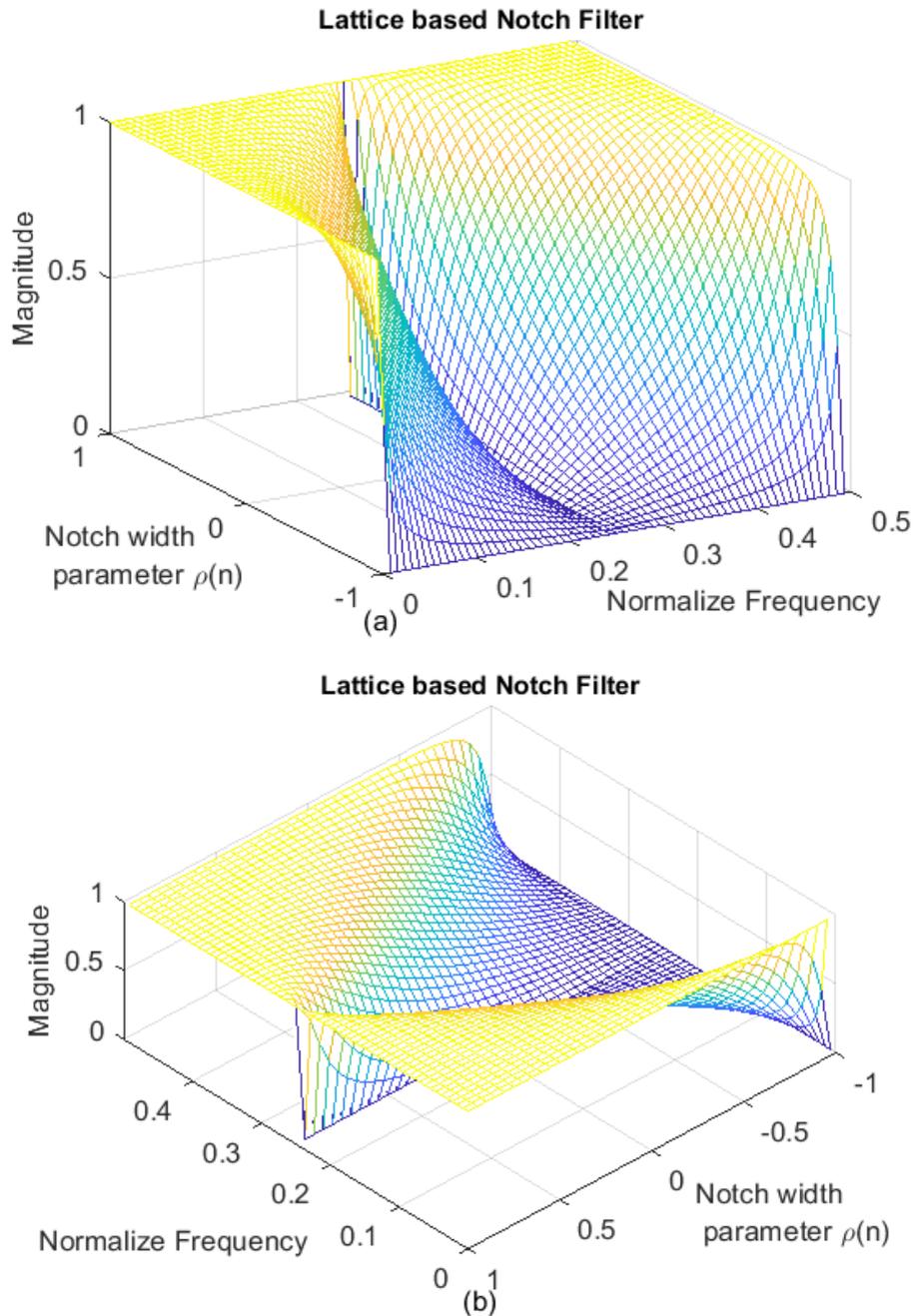


Figure 5.13 (a) 3D Frequency Response of Lattice-based Notch Filter when the values of ρ sweeps from between -1 and 1.(b) top view of 3D frequency response of Figure 5.13 (a)

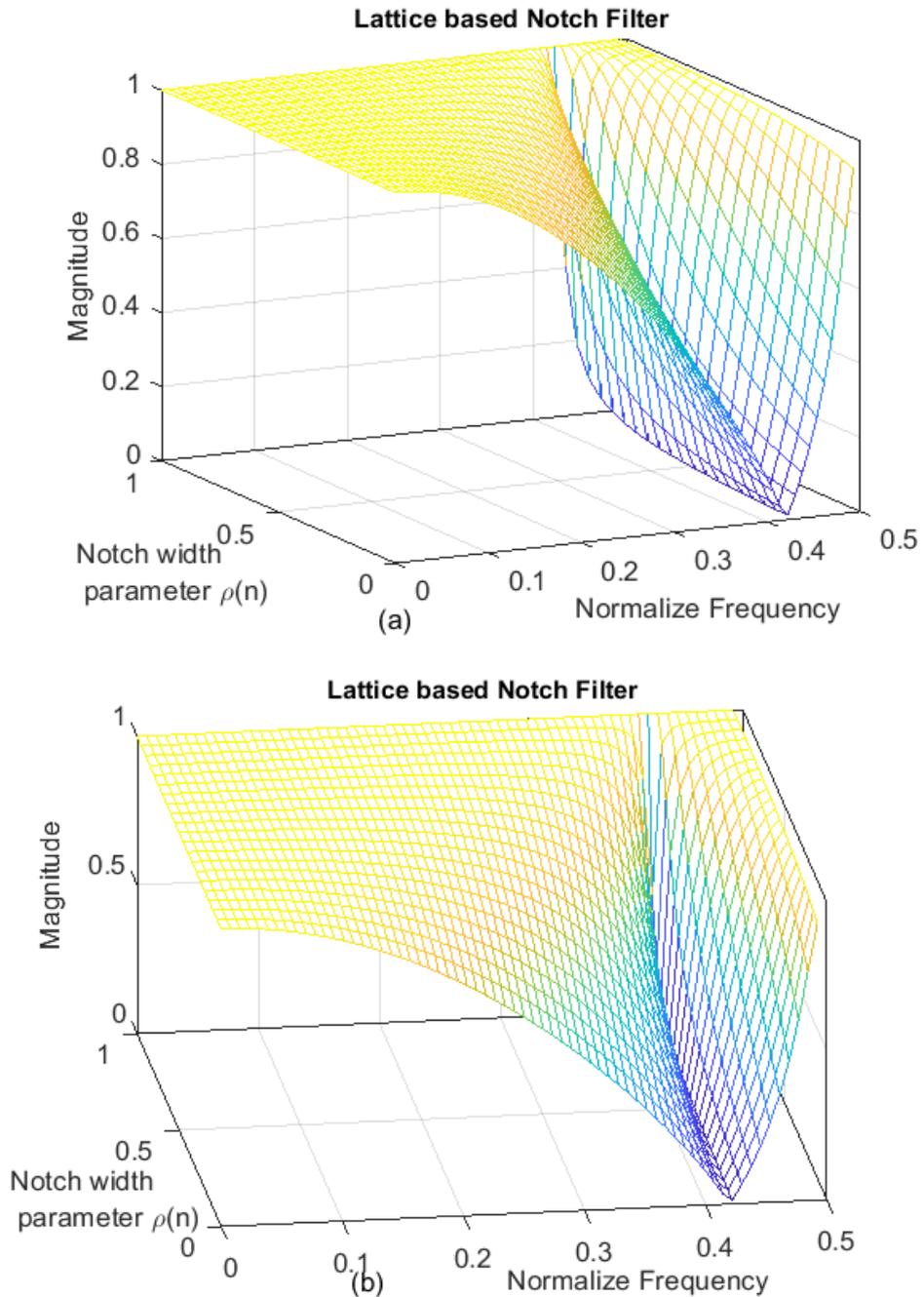


Figure 5.14 (a) 3D Frequency Response of Lattice-based Notch Filter when target frequency close to Nyquist and ρ sweeps from 0 to 1. (b) Front view of the 3D frequency response of Figure 5.14 (a)

As this notch filter design is based on an all-pass transfer function, the gain in the passband region stays constant, which is unity. Hence useful signal in the passband region is not altered or distorted, unlike the direct form transfer function. Figure 5.13 shows the 3D frequency response when the value of ρ sweeps from -1 to 1. As observed from Figure 5.13,

when the value of ρ approaches -1, the notch width becomes wide enough to spread all over the frequency spectrum, hence producing an all-stop kind of response. It implies that while adapting notch bandwidth parameter ρ in an ANF, it should never converge toward minus one.

Suppose the target frequency is close to Nyquist. In that case, the lattice-based filter maintains its intrinsic frequency response characterises, as demonstrated in Figure 5.14, where the target frequency is closer to Nyquist, just above 0.40.

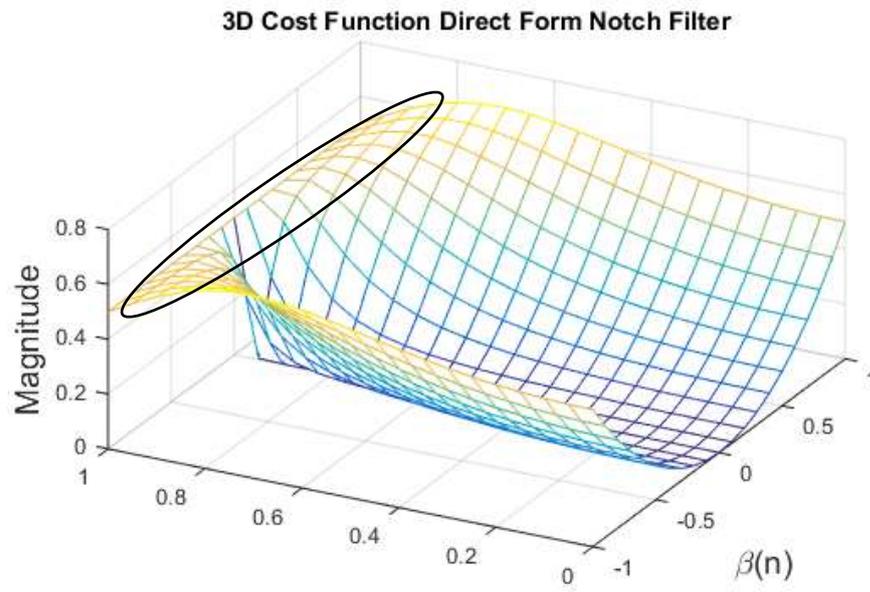
5.3.5 Cost Function

The cost function analysis provides an intuitive understanding of the convergence of both the parameter of ANF β and ρ . The cost function plots in this section offer a different perspective, which helps understand the limitations of the Adaptive IIR notch filter. By considering the cost function $J_D(\beta, \rho)$ (Direct Form) and $J_L(\beta, \rho)$ (Lattice Form) for the transfer function of filters in equations 5.20 and 5.21. It is a function of both variable β and ρ , given by This can be approximate via batch method [85] and can be expressed as follows.

$$J_D(\beta, \rho) = \sum_{\beta=-1}^1 \sum_{\rho=0}^1 \left\{ \frac{1}{N} \sum_{n=0}^{N-1} y_D^2[n] \right\} \quad (5.20)$$

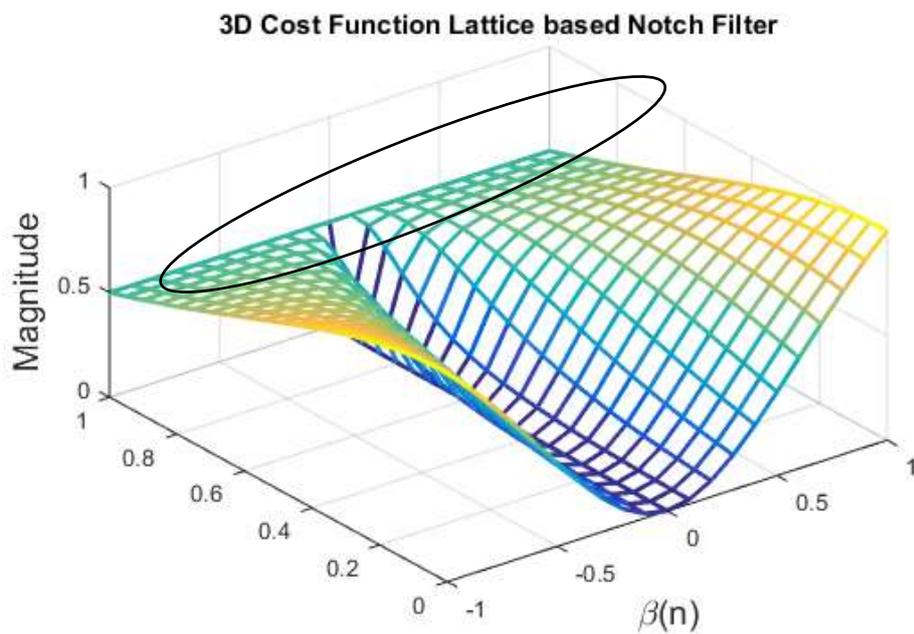
$$J_L(\beta, \rho) = \sum_{\beta=-1}^1 \sum_{\rho=0}^1 \left\{ \frac{1}{N} \sum_{n=0}^{N-1} y_L^2[n] \right\} \quad (5.21)$$

The cost function for both filters is plotted by sweeping the value of β from -1 to 1 and the value of ρ from 0 to 1 and input with the signal of frequency half Nyquist 0.25 and noise variance of 0.04. Figures 5.15 and 5.16 are the respective 3D plot of the cost of two transfer functions.



Notch width
parameter $\rho(n)$

Figure 5.15 Cost Function Plot of Direct Form Transfer Function



Notch width
parameter $\rho(n)$

Figure 5.16 Cost Function Plot of All-pass Lattice Transfer Function

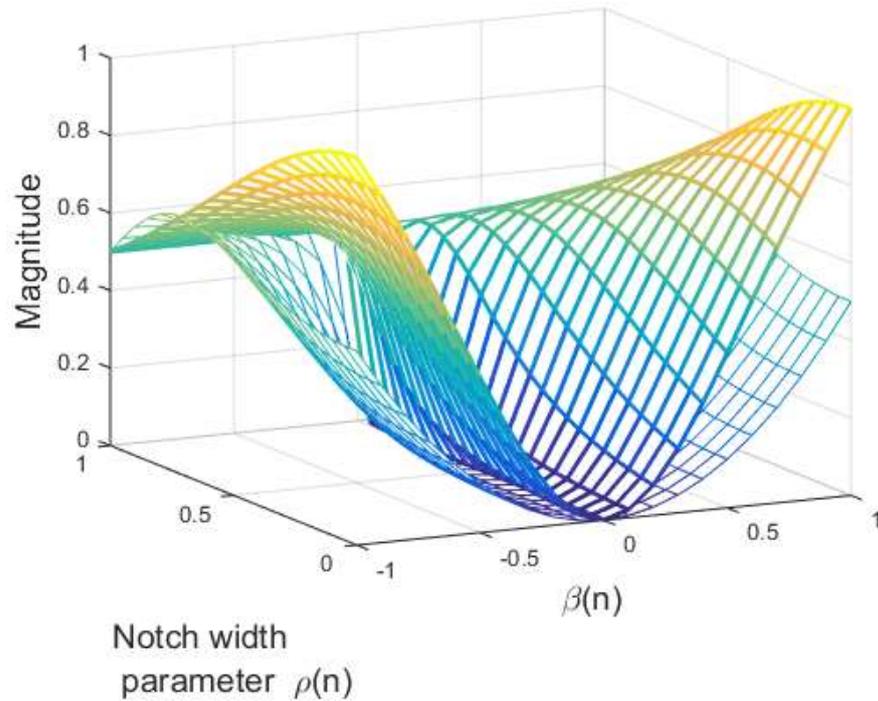


Figure 5.17 Illustration of the plots in Figures 5.15 and 5.16, superimposed on each other

Figures 5.15 and 5.16 show that as notch bandwidth approaches unity, the gradient of the cost function is flat, as shown by two ellipses marked in Figures 5.15 and 5.16. By closely examining the gradient of the cost function for an all-pass based transfer function is flatter for the larger surface area compared to another counterpart.

For both 3D plots of the cost function, it can be observed that the gradient/ steepness of the surface and depth of the cost function are inversely proportional to the notch bandwidth parameter ρ . The effect on the deepness is illustrated in Figure 5.18. In both Figures 5.15 and 5.16, as ρ sweeps from 1 to 0, the valley of both curves becomes steeper, which means the gradient is higher. For much smaller values of ρ (closer to zero), the curve gradient is also steeper. It causes the adaptive algorithm to converge at a faster rate. For ρ close to unity, the cost function has a flat gradient; the adaptive algorithm takes a long time to locate the target frequency. Figure 5.17 shows the difference in the steepness of the slope of surfaces for two curves. Both 3D plots are superimposed onto each other. The curve in the thick line represents an all-pass lattice-based cost function. It is steeper than the other curve, meaning

lattice-based ANF converges quicker to global minima. In practical applications, a very narrow notch is desired to mitigate single-tone sinusoidal signal effectively, which means a higher value of ρ is required. And if the value of ρ is high, close to unity minimization of the cost function yields a flat response, as shown in Figure 5.16, which means the adaptive algorithm adapts at a slower pace. To achieve fast convergence, the initial value of ρ can be set as a small value (let's say 0.70) and then adapted to an optimum value close to unity.

At the start of each subsequent hop signal, ANF has to provide fast convergence of parameter β to the target frequency. To achieve this, the notch bandwidth parameter ρ has to be automatically reset to a lower value (such as 0.60 or 0.70).

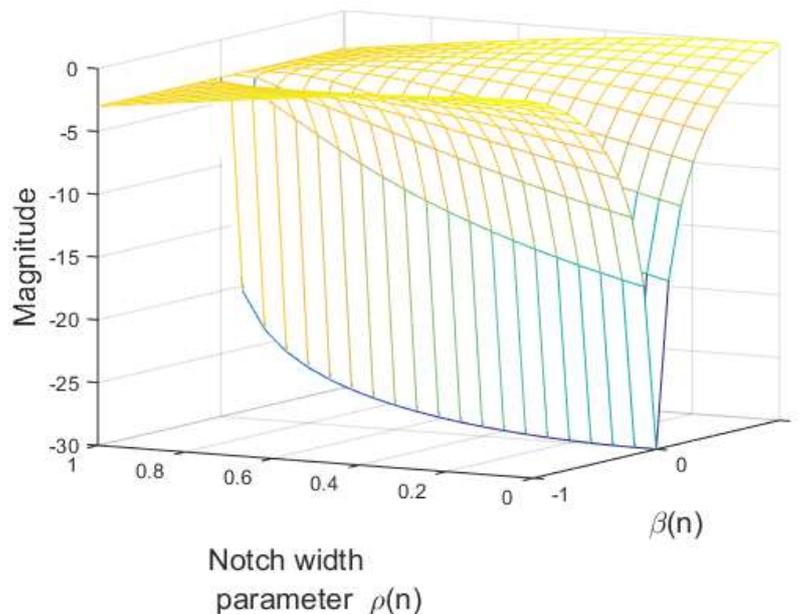


Figure 5.18 Illustration of the effect of ρ on the deepness of the Cost Function. The deepness of the cost function increases as the ρ approaches zero

The observations from the analysis presented in this section are summarised as follows

- I. The parameter ρ has to be constraints between the optimum minimum value and optimum maximum value to develop and design adaptive notch bandwidth in an ANF. If ρ is very small, let's say 0.3 - 0.5, the notch width will be wide enough to

distort the useful signal, and if the value of ρ exceeds unity it will lead to instability or if ρ approaches -1, filter response becomes an all-stop type.

- II. All-pass lattice notch filter can maintain its frequency response characteristics as ρ varies from -1 to 1, providing unity gain in the passband region.
- III. Analysis of the cost function concludes that the initial minimum value ρ has to be less than the value for which the plot of the cost function begins to be non-flat. In other words, the initial or resetting value of ρ cannot be in the flat region of the plot of the cost function. It ensures fast convergence at the start of each subsequent hop frequency.

5.4 Proposed Full Gradient Term for Notch Bandwidth Parameter ρ

In [81], [82] and [85], a partial gradient term is utilized to adapt the notch bandwidth parameter ρ in a lattice-based ANF. The simulation results in section 5.3 show that this method is poor and unreliable for the hop frequency type of interferences. Hence a full gradient term can be derived by assuming ρ is not fixed in the denominator, as previously done in section 5.3.1. It is achieved by differentiating (5.22) with respect to ρ .

$$H_L(z) = \frac{1+\rho}{2} \frac{1-2\beta z^{-1}+z^{-2}}{1-\beta(1+\rho)z^{-1}+\rho z^{-2}} = \frac{U_\rho}{V_\rho} \quad (5.22)$$

Therefore differentiating (5.22) via quotient as follows

$$\frac{\partial}{\partial \rho}(H_L(z)) = \frac{U_\rho' \cdot V_\rho - V_\rho' \cdot U_\rho}{V_\rho^2} \quad (5.23)$$

Which yields

$$\frac{\partial}{\partial \rho} = \frac{0.5 \{(1-2\beta z^{-1}+z^{-2}).(1-\beta(1+\rho)z^{-1}+\rho z^{-2})\} - (1+\rho)(\beta z^{-1}+z^{-2}).(1-2\beta z^{-1}+z^{-2})}{(1-\beta(1+\rho)z^{-1}+\rho z^{-2})^2} \quad (5.24)$$

(5.24) can now be further simplified by rearranging and cancelling the common terms in the numerator.

$$\frac{\partial}{\partial \rho} = \frac{0.5(1-2\beta z^{-1}+z^{-2})(1-z^{-2})}{(1-\beta(1+\rho)z^{-1}+\rho z^{-2})^2} \quad (5.25)$$

Now splitting up the denominator gives the following expression

$$\frac{\partial}{\partial \rho} = \frac{1}{2} \frac{(1-2\beta z^{-1}+z^{-2})}{(1-\beta(1+\rho)z^{-1}+\rho z^{-2})} \times \frac{(1-z^{-2})}{(1-\beta(1+\rho)z^{-1}+\rho z^{-2})} \quad (5.26)$$

Lastly, adding the term $(1 + \rho)$ in the numerator and denominator of equation 5.26 creates the expression as shown in equation 5.27

$$\begin{aligned} & H_L(z) \\ \frac{\partial}{\partial \rho} (H_L) &= \frac{1 + \rho}{2} \frac{1 - 2\beta(n)z^{-1} + z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}} \\ & \quad \times \underbrace{\frac{1 - z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}}_{G_\rho(z)} \times \frac{1}{1 + \rho} \end{aligned} \quad (5.27)$$

To implement (5.27), another filter is required. Two transfer function are being multiplied together in the z -domain, which imply both these filters are needed to cascade in the time domain to generate a full gradient term for ρ . Hence update equation of the notch bandwidth parameter becomes as follows.

$$\rho[n] = \rho[n - 1] + \mu_\rho y_L[n] g_\rho[n] / \phi_\beta[n] \quad (5.28)$$

Where, μ_ρ is step-size, $y_L[n]$ is the output of the filter $H_L(z)$, and $g_\rho[n]$ is the output of the full gradient transfer function $G_\rho(z)$. And β is adapted as done in section 5.3.1 [48] and

This structure is modelled and simulated in MATLAB, and simulation results are shown in Figures 5.20 and 5.21 for tracking the hop frequency signal.

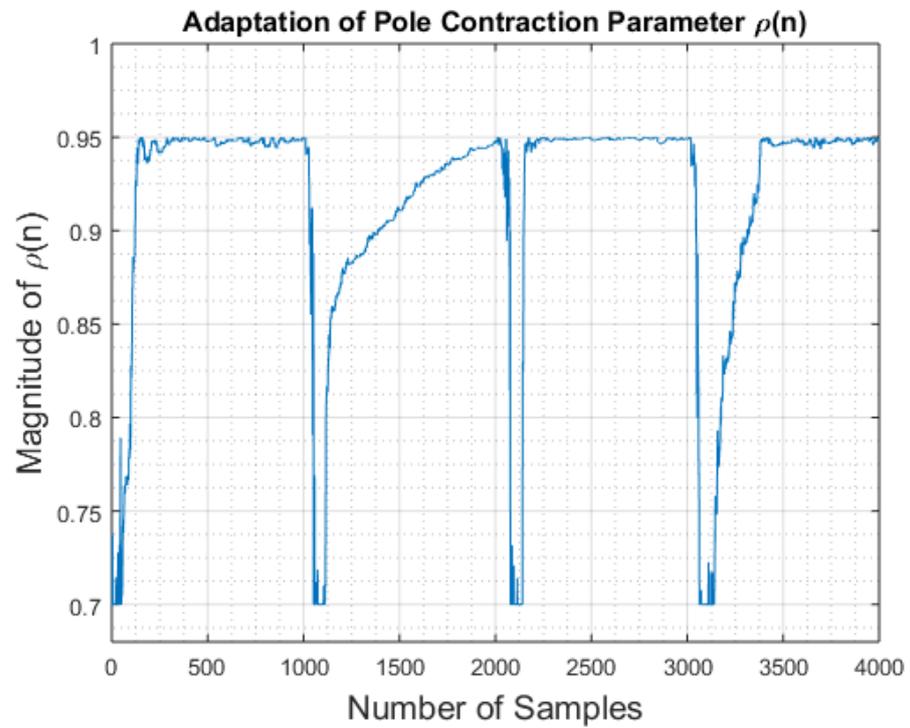


Figure 5.20 Full gradient-based adaptation of ρ in Lattice based ANF

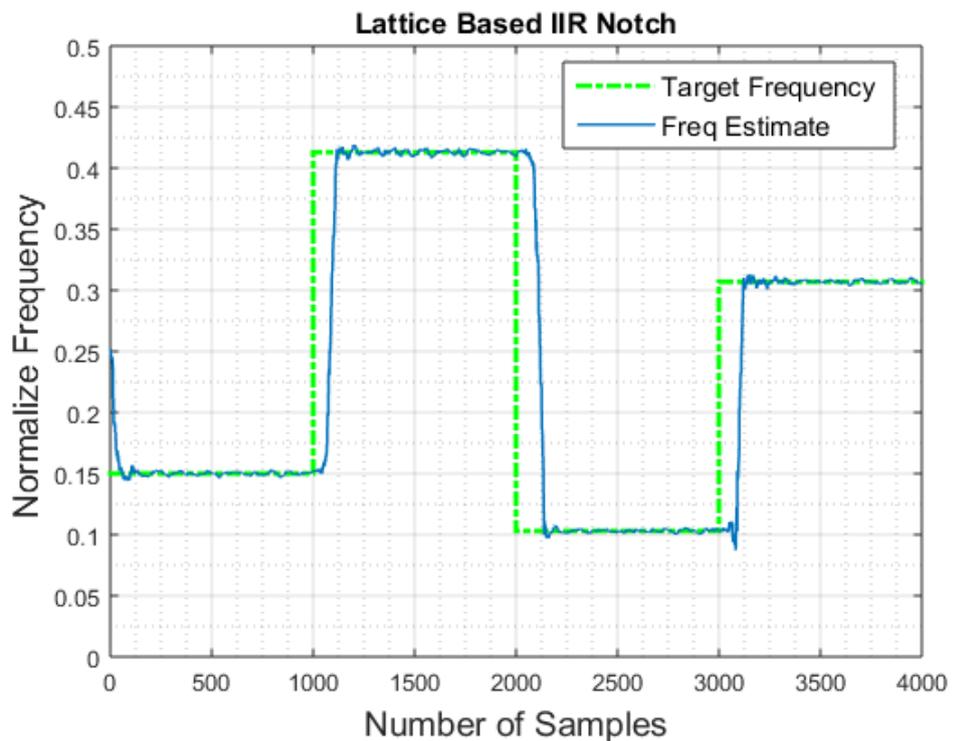


Figure 5.21 The tracking performance of Lattice-based ANF for the hop frequency signal

Table 5.3 Simulation parameters used to simulate Figures 5.20 and 5.21

Noise variance σ^2	0.25
μ_β step-size for β	0.018
μ_ρ step-size for ρ	0.033-0.04
'N' no. of samples	4000
Initial value of β	0.25 (normalized frequency)
Initial value of ρ	0.70 (Pole contraction factor)
γ forgetting factor	0.90

Simulation parameters are shown in table 5.3. Figure 5.20 demonstrate the full gradient approach, which provides encouraging results. The results in Figure 5.20 are produced via constraints adaptation of ρ as follows:

Algorithm II: Adaptation of $\rho(n)$ with constraints

Input: $x[n]$

Output: $y_L[n]$

Initialisation : $\rho[n] = 0.70$; $\mu_\rho = 0.03$; $\gamma = 0.90$

```

1:  For  $n = 1:1:N_{sample}$ 
2:       $y_L[n] = \mathbf{latticefilter}(\beta[n], \rho[n], x[n])$ 
3:  Update  $\rightarrow \beta[n]$ 
4:      If  $0.70 < \rho[n] < 0.95$ 
5:           $\rho[n] = \rho[n - 1] + \mu_\rho \times y[n] \times g[n] / \phi_\beta$   $\triangleright$  Update  $\rightarrow \rho$ 
6:           $\rho[n - 1] = \rho[n]$   $\triangleright$ 
7:      end
8:      If  $0.70 > \rho[n] > 0.95$   $\triangleright$  Retain  $\rightarrow \rho[n - 1]$ 
9:           $\rho[n] = \rho[n - 1]$  previous value of  $\rho$  if it
10:      end exceed 0.95
11:      If  $\phi_\beta < 0.50$   $\triangleright$  Reset condition at start
12:           $\rho[n] = 0.70$  of
13:      end new hop frequency
11: end

```

It is to be noted that these constraints are set empirically to obtain optimum results in terms of convergence speed and robust performance. The parameter ρ only adapts if the current value of ρ is in between set constraints 0.70 and 0.95 and resets itself to 0.70 if the instantaneous power ϕ_β approaches zero (reset if ϕ_β is less than 0.5). In Figure 5.20, it can be noted that ρ does not reset to 0.70 for the subsequent hop frequency exactly at 1000 samples, 2000 samples and 3000 samples mark. This means β starts to lock onto a new target frequency before ρ is reset to 0.70. Moreover, the second hop frequency (1000-2000 samples) ρ converges to 0.95 at a very slow rate. Further constraints are needed to apply, and modification is required in the constraint's algorithm. Another approach is tried and tested, which can be called the '**hard reset and wait method**'. In this method, the value of ρ is reset to 0.70 after every 1000 samples, as we are aware of the fact that each hop frequency is 1000 samples long in 4000 samples in the input signal. After every reset, the value of ρ is kept minimum (0.70) for a certain number of samples (wait for for 100-150 samples), then start updating ρ again. Simulation results for this kind of approach are presented in Figures 5.22 and 5.23.

Comparing the results of Figure 5.21 and 5.23, tracking in Figure 5.21 is offset by more margin, whereas simulation results in Figure 5.23 are more in line and along the target frequency. But in real life, each hopping frequency might have a different duration, size, or number of samples. This implies that ANF required another mechanism within this developed algorithm to automatically set parameter ρ to the desired minimum value just before tracking the next subsequent hop frequency. Figure 5.22 shows the adaptation of parameter ρ , it is well-shaped line graph and, after every reset of parameter ρ , is not updated for 100 samples value is kept at 0.70 and afterwards quickly converges to 0.95 until the next hop signal.

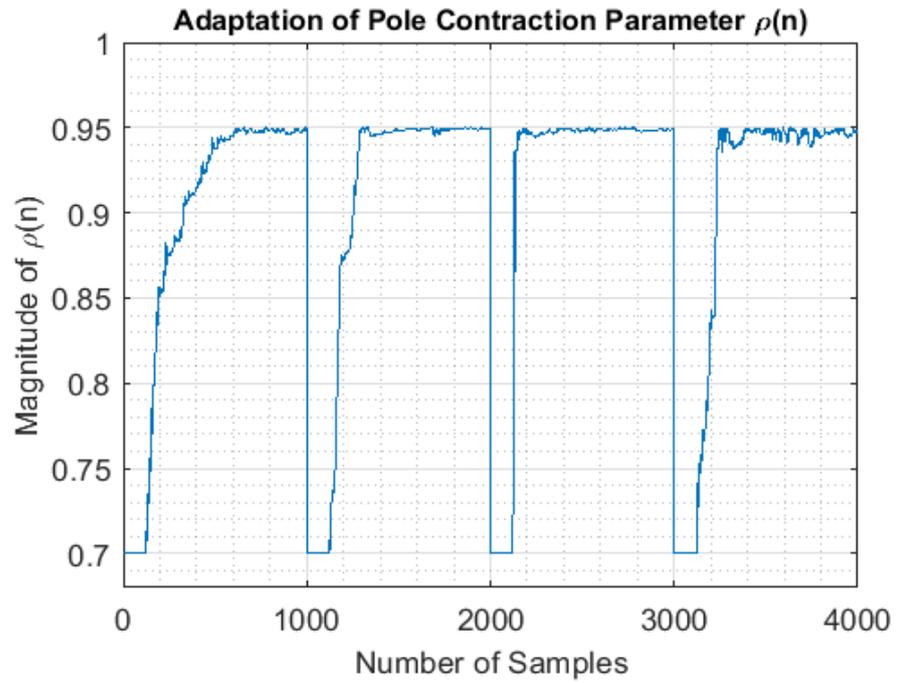


Figure 5.22 Adaptation of parameter ρ via hard reset and wait method

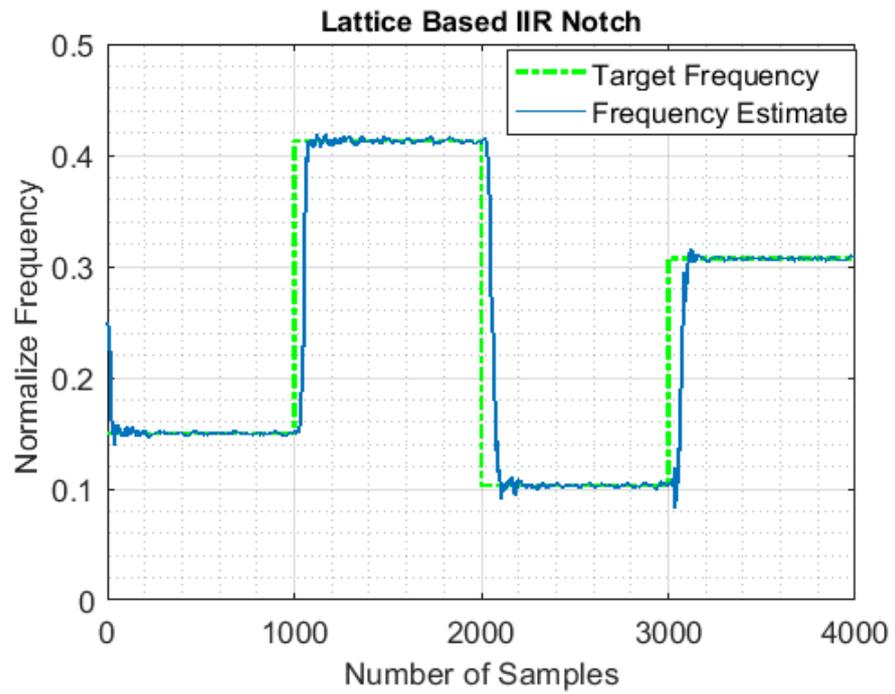


Figure 5.23 Tracking performance of Lattice-based ANF with hard reset and wait method

Algorithm III: Adaptation of $\rho(n)$ with constraints

Input : $x[n]$ **Output :** $y_L[n]$ *Initialisation :* $\rho[n] = 0.70$; $\mu_\rho = 0.03$; $\gamma = 0.90$

```
1:  For  $n = 1:1:N_{sample}$ 
2:       $y_L[n] = \mathbf{latticefilter}(\beta[n], \rho[n], x[n])$ 
3:  Update  $\rightarrow \beta[n]$ 
4:      If  $0.70 < \rho[n] < 0.95$ 
5:           $\rho[n] = \rho[n - 1] + \mu_\rho \times y[n] \times g[n] / \phi_\beta$   $\triangleright$  Update  $\rightarrow \rho$ 
6:           $\rho[n - 1] = \rho[n]$   $\triangleright$ 
7:      end
8:      If  $0.70 > \rho[n] > 0.95$   $\triangleright$  Retain  $\rightarrow \rho[n - 1]$ 
9:           $\rho[n] = \rho[n - 1]$  previous value of  $\rho$  if it
10:         end exceed 0.95
11: end
```

Following simulation results based on the ‘**minimum constraint method**’, in this method ρ initialise again with 0.70 and update the equation 5.26 as long as the value of ρ does not exceed 0.95. If it does exceed 0.95, the previous value of ρ is retained, implemented as follows

The simulation result in Figure 5.24 demonstrates an interesting inherent capability of the derived full gradient function (at the beginning of section 5.4) that at the start of each new hop frequency, the ρ does try to reset to 0.70 without any external constraints. Unfortunately, in between 1000-1100 samples, 2000-2300 samples and 3000-3100 samples, it toggles between 0.95 and 0.75, which is undesirable in a fully adaptive lattice ANF filter.

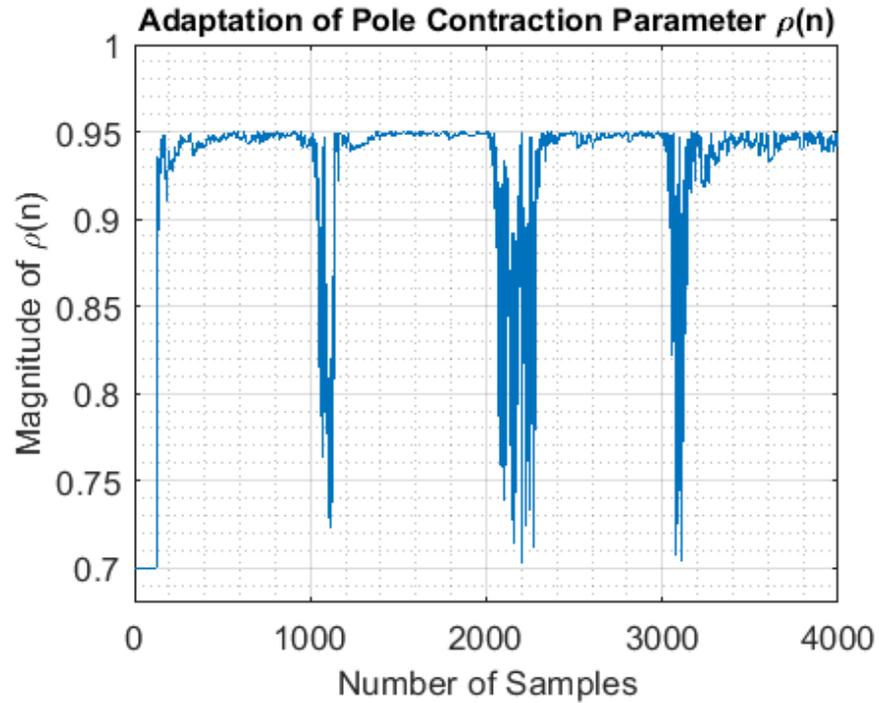


Figure 5.24 Adaptation of ρ with minimum constraints method

5.4.1 Fully Constraint Adaptation Algorithm of ρ

Automatic resetting of notch bandwidth parameter ρ to 0.70 at the start of each subsequent hop frequency became a challenging part of this research. Numerous resetting conditions were set and tested. In this section, only a few of them are discussed. These are all heuristic approaches and are simply designed by analysing the internal signal of the ANF filter, such as ϕ_β , g_β , g_ρ , y_L and ϕ_ρ . These signals are again defined in table 5.4 with their respective names and equations.

Table 5.4 Internal Signal used in developing automatic resetting mechanism for ρ

Signal	Name	Equation
ϕ_β	Instantaneous power of g_β	$\phi_\beta[n] = \gamma\phi_\beta[n - 1] + (1 - \gamma)g_\beta^2$
ϕ_ρ	Instantaneous power of g_ρ	$\phi_\rho[n] = \gamma\phi_\rho[n - 1] + (1 - \gamma)g_\rho^2$
g_β	Gradient signal for updating β	Refer to Figure 5.19
g_ρ	Gradient signal for updating ρ	Refer to Figure 5.19
y_L	Output of ANF	Refer to Figure 5.19

Three methods are proposed in this section which are detailed as follows.

5.4.2 Method I Reset Condition: Filter Output and Instantaneous Power (ϕ_β)

Reset ρ when the output of the filter is more than the gradient energy signal ϕ_β . The signal ϕ_β approaches to the zero whenever ANF have locked on the target frequency, and once the target frequency is located, it grows in magnitude until the subsequent hop frequency. This effect is illustrated in Figure 5.25 (a) with a red colour line at 1000, 2000 and 3000 samples, the ϕ_β drop exponentially and tend to approach zero. At this particular point, the output y_L of the filter is more than the gradient energy signal ϕ_β , which is visible in the zoomed-in version of 5.25(b).

In Figure 5.25 (b), it can be observed that the gradient signal is below the output the y_L for the considerable number of samples (between 1020-1060 samples) in this region, the parameter ρ does not update as well, as shown in Figure 5.26 in between 1000-1100 samples. But again, here, the issue is that the parameter ρ does not reset precisely at the 1000, 2000, and 3000 samples mark. It resets just after 1000, 2000 and 3000 marks, as shown in Figure 5.26. This method failed to reset ρ to 0.70 on time for each subsequent hop signal. One good takeaway from this simulation is that as long as ϕ_β is smaller than the filter's output, the value of ρ is not updated for the small fraction of samples. Hence, there is no need to include a 'wait' condition for certain samples, as demonstrated in the 'hard reset and wait' method.

Reset condition:

if $|y_L| > \phi_\beta$ then reset the value to $\rho(n) = 0.70$

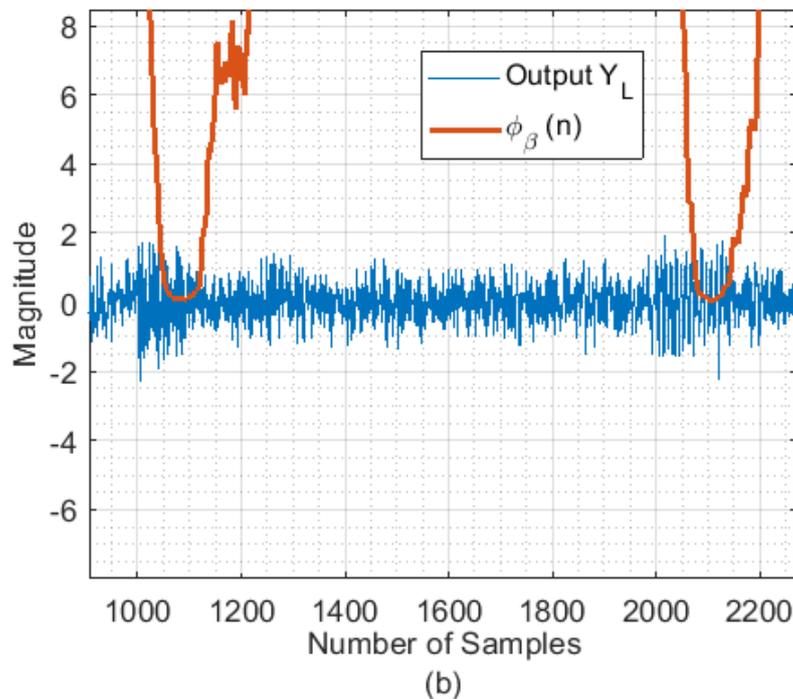
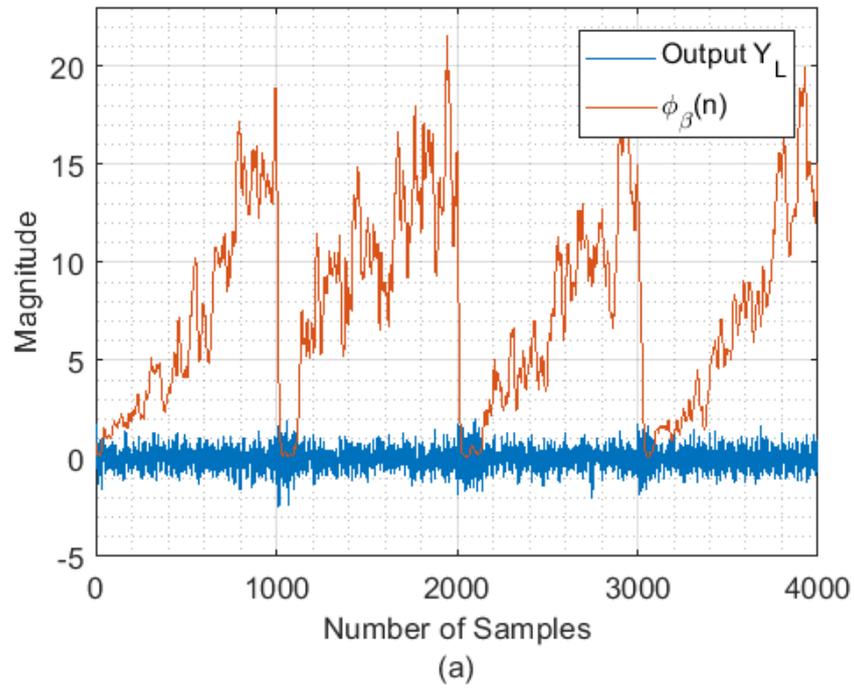


Figure 5.25 (a) The magnitude of the gradient energy shown in red and blue represents the corresponding output of the filter (b). Zoomed version of (a) to illustrate the fact that at one point in time gradient is less than the output of the filter

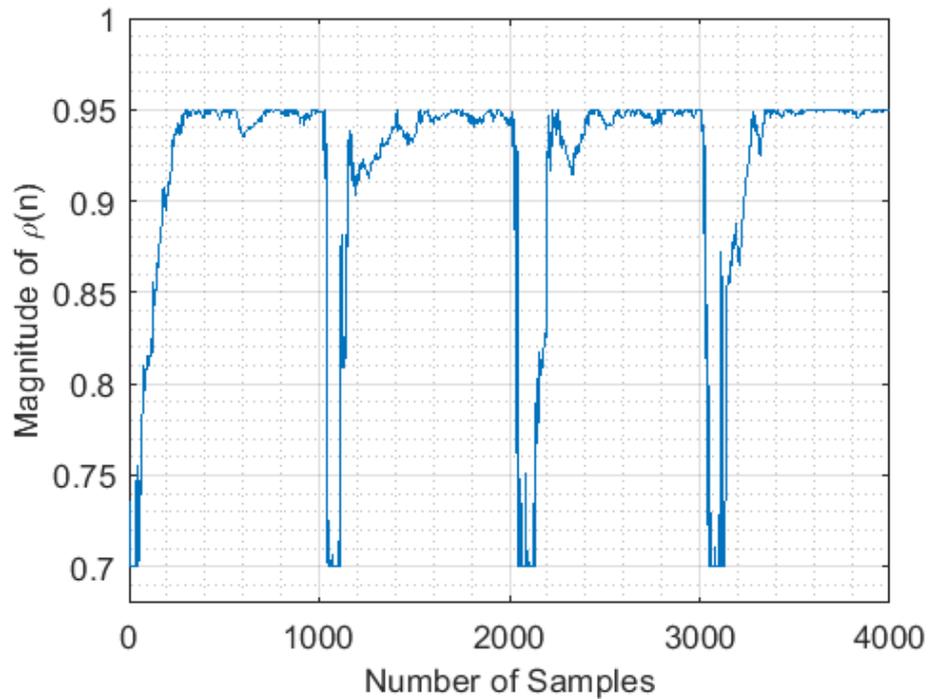


Figure 5.26 Corresponding resetting the value of ρ via Method I

5.4.3 Method II Reset Condition: Instantaneous Gradient of ϕ_ρ .

This method is based on the instantaneous gradient of the signal ϕ_β and ϕ_ρ . During the analysis of different internal signals of the adaptive notch filter, it was observed that at the start of each subsequent hop frequency signal (for example, at 1000, 2000 and 3000 samples), the instantaneous gradient of ϕ_β rapidly decreases and while the instantaneous gradient ϕ_ρ rapidly increases. At this particular point absolute value of the instantaneous gradient of each of ϕ_ρ is of comparable level that of the ϕ_β . Figure 5.27 shows the instantaneous gradient curve for each signal, ϕ_β in blue and ϕ_ρ in red. The instantaneous gradient of the ϕ_ρ (red line) is mostly below 1, between 300-980 samples, 1500 -2000 samples, 2000 - 3000 samples and above 3000 samples. The region of interest is marked in black circles, this peaking of the instantaneous gradient of ϕ_ρ occur exactly around 1000, 2000, and 3000 sample points. These are the positions at which ρ has to reset to 0.70.

Instantaneous gradient of ϕ_ρ can be simply calculated by

$$\Delta\phi_\rho = \phi_\rho[n] - \phi_\rho[n - 1] \tag{5.32}$$

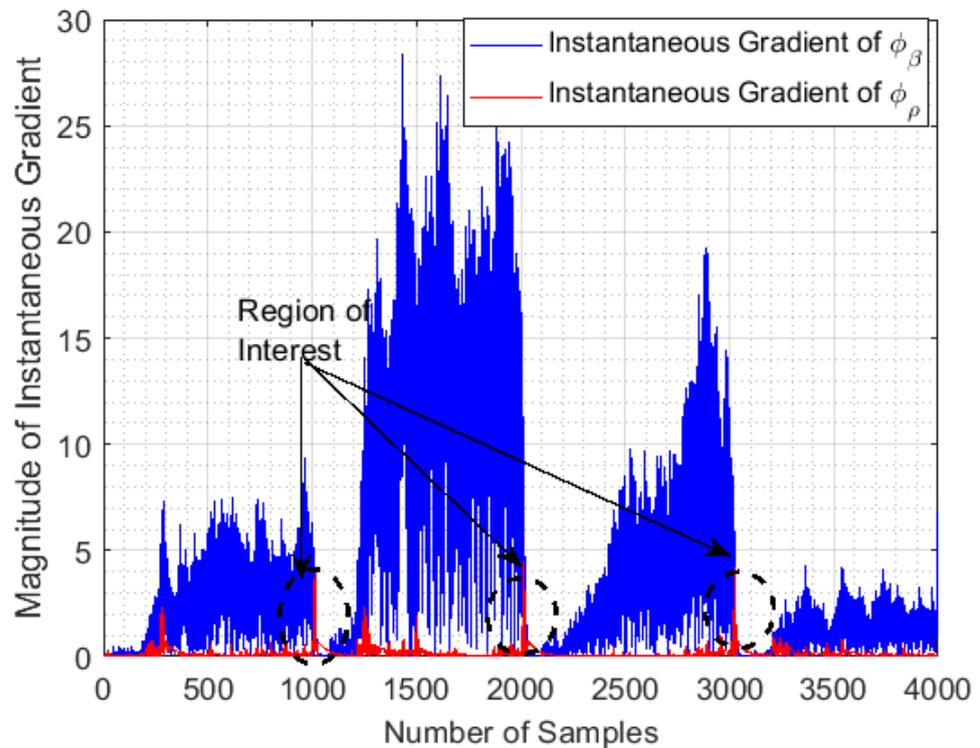


Figure 5.27 Absolute value of the instantaneous gradient of ϕ_β and ϕ_ρ

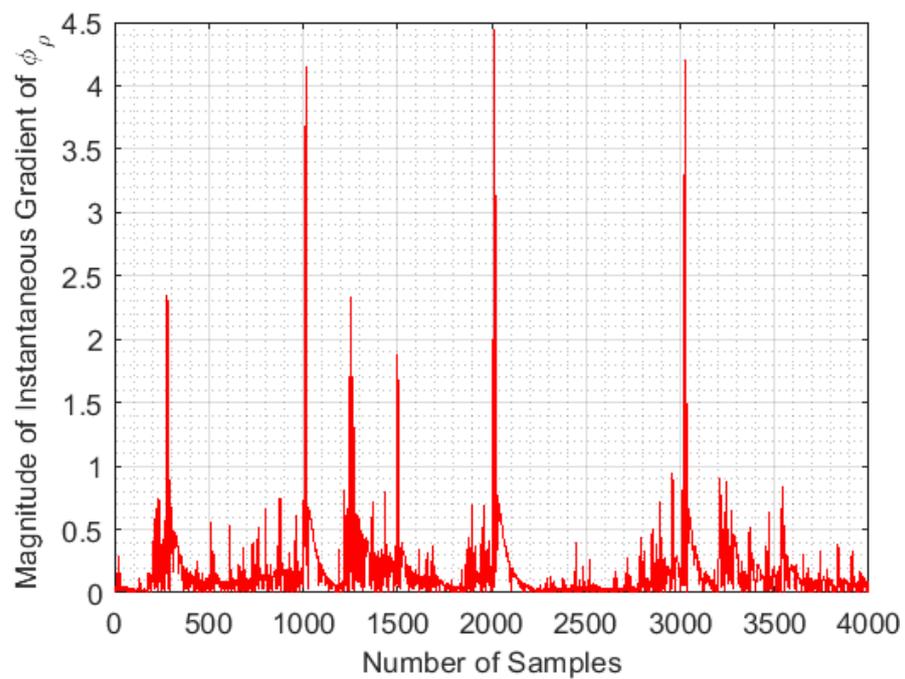


Figure 5.28 Instantaneous Gradient of ϕ_ρ

Reset Condition:

if $abs(\phi_\rho[n] - \phi_\rho[n - 1]) > 3.5$, then reset the value to $\rho(n) = 0.70$
and wait for N number samples before updating ρ again, and $N = 120$

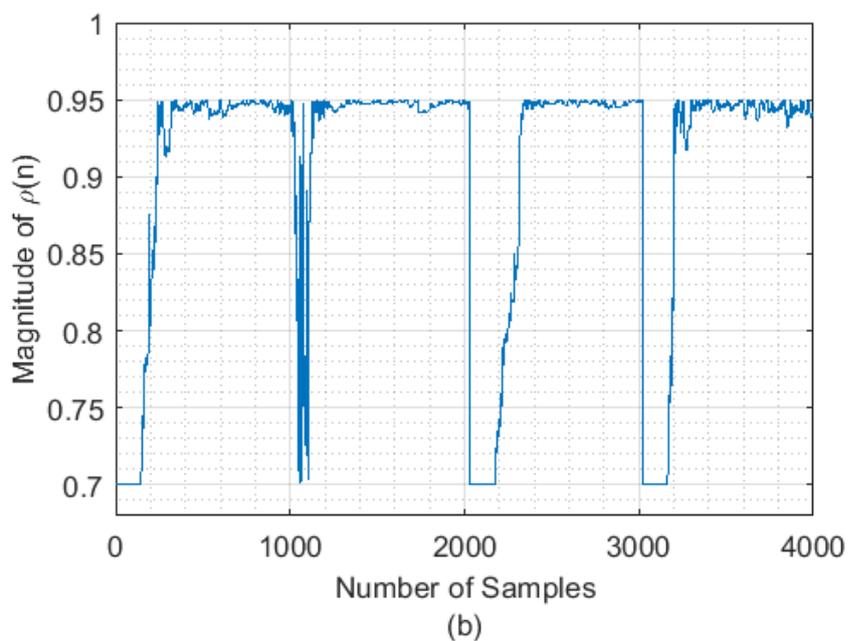
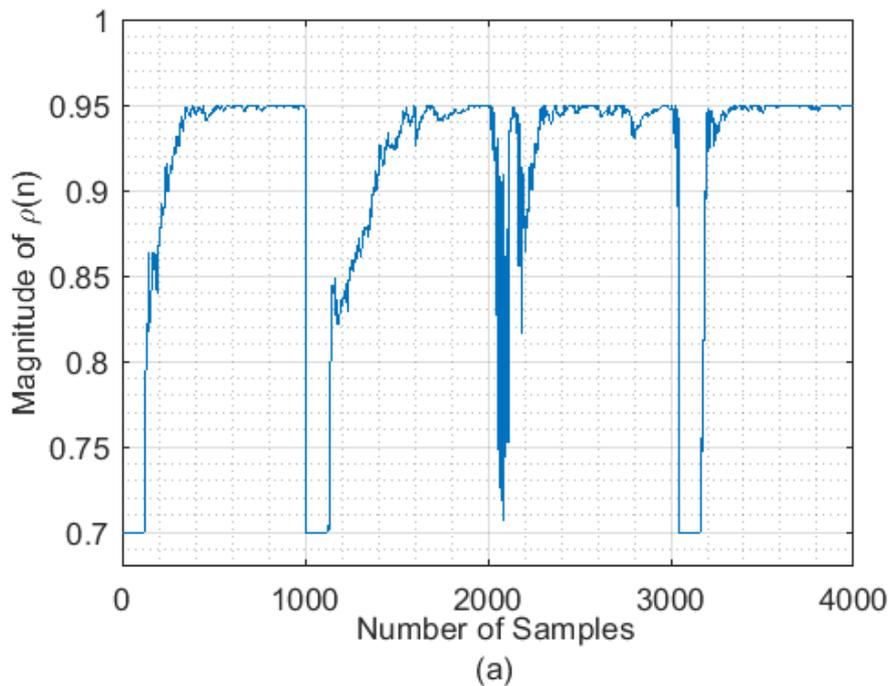


Figure 5.29 (a) Corresponding resetting the value of ρ via Method II, but unable to reset ρ after 2nd hop frequency. (b) In the second run of the simulation, and this time unable to reset ρ after first hop frequency signal

In Figure 5.29 (a), method II is able to reset ρ just after 1000 and 3000 samples, very close to desired positions(1000, 2000,3000 marks). This method delivered better results than method I, where the resetting occurred after a long delay, compared with Figure 5.26. It can be seen in both Figures 5.29 (a) and (b) that this method failed to reset at the 2000 samples mark for (a) and at the 1000 samples mark for (b).

5.4.4 Method III Proposed Method: Instantaneous output power and $V_{threshold}$

After analysis of different signals path and their statistical properties and nature, it is concluded that the best possible method to reset the value of ρ is to observe the output of the ANF. Additionally, resetting ρ to 0.70 just before the new hop frequency is key to successfully adapting both parameters. Therefore, resetting should be applied carefully within the algorithm, and the selection of the constraints must be such that it works universally regardless of the environment. This method utilized the calculations of the gradient energy at the output of the ANF. The output signal of the ANF increases in amplitude just before the next hop frequency signal. If this variation at the output of the filter can be detected just in time (means at the beginning of the new hop frequency), the value of ρ can be reset to 0.70 on time without delay. A new signal was created, the parameter $\psi[n]$ and equals to:

$$\psi[n] = \gamma\psi[n - 1] + (1 - \gamma)|y_L|^2 \quad (5.32)$$

Where $\psi[n]$ is gradient energy at the output of the ANF in Figure 5.19, y_L is the output of the filter, and γ is a forgetting factor close to 0.9-0.99. Figure 5.30 shows the plot of this gradient function at the output of the filter.

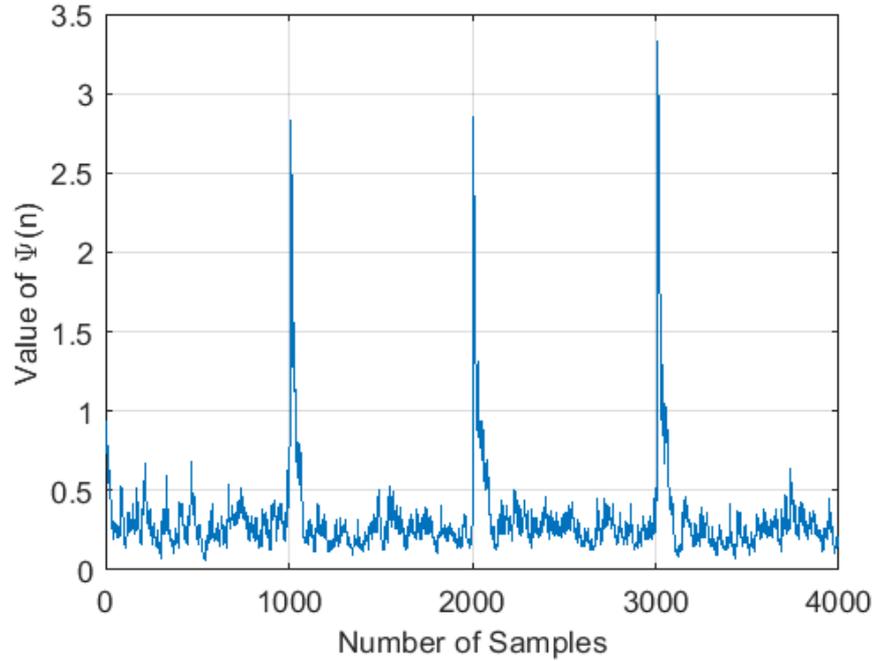


Figure 5.30 Plot of $\psi[n]$ at the output of the ANF

Figure 5.30 illustrates the plot of $\psi[n]$ at the output of the ANF, unlike in Figure 5.28, only three distinct peaks are visible at 1000, 2000 and 3000 samples mark. It provides better resetting criteria than previously tried and tested in Method I and Method II.

$$V_{threshold} = 2E\{|x[n]|\} \quad (5.33)$$

A reset threshold was set, and it is called $V_{threshold}$. It is the equivalent of twice the size of the expected value of the input signal (noisy signal), as shown in equation 5.33. The peaking of the output gradient signal $\psi[n]$ always occurs at the exact position at which parameter ρ needed to be reset, and these peaks are always more than the value of the $V_{threshold}$, hence this provides a resetting condition that can be applied universally regardless of the nature or statistical properties of the input signal.

Reset Condition:

if $\psi[n] > V_{threshold}$, then reset the value to $\rho(n) = 0.70$

and wait for M number samples before updating ρ again , and $M = 120$

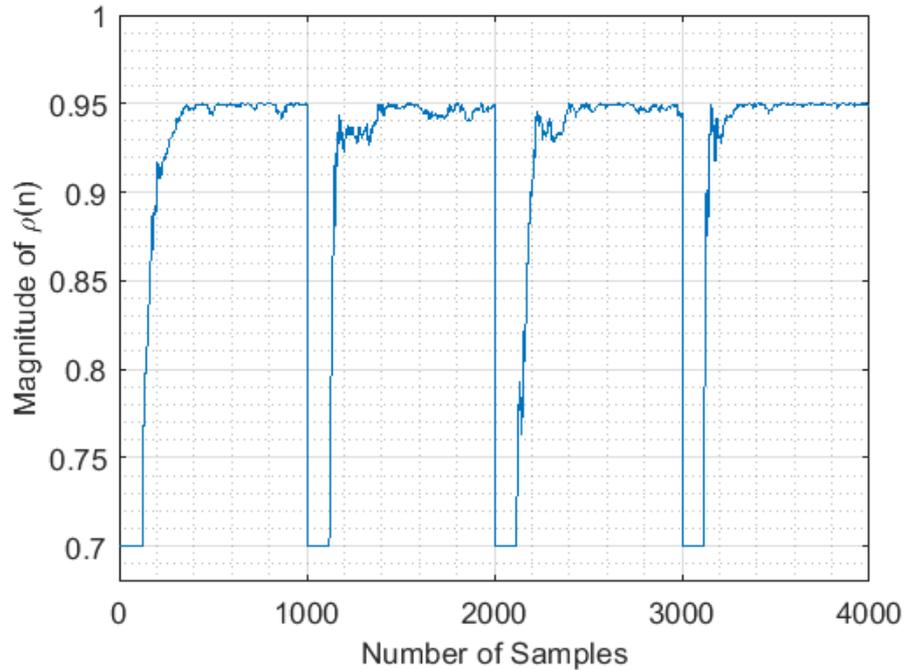


Figure 5.31 Corresponding resetting the value of ρ via Method III

A complete algorithm with constraints is presented on the next page of this thesis. In Figure 5.31, the adaptation of ρ is simulated for the full gradient structure, as shown in Figure 5.19. This simulation demonstrates that the value of ρ is resetting exactly at 1000, 2000, and 3000 samples mark, as desired. To ensure β always adapts before ρ , the adaptation of ρ is always initiated after the ‘M’ number of samples and also whenever ρ reset to value 0.7 as shown in step 4 of the algorithm I (on next page), again it has to wait for ‘M’ number of samples. Hence allowing β to converge to the target frequency at a quicker pace before ρ converges to 0.95, as shown in Figure 5.31. The simulation result in Figure 5.31 was produced with μ_ρ set to 0.03 and μ_β set to 0.018. The forgetting factor γ was fixed at 0.9 for the adaptation of ρ and β . As per proposed algorithm I on the next page, ρ was initialized at 0.70, ρ adapts after ‘M’ of samples, so β start to converge to the target frequency at a faster rate. Step 3 of the algorithm ensures ρ only updates its value when the current value of the ρ is in between the range of 0.70 to 0.95. Otherwise, if ρ tends to be more than 0.95 or below 0.70, the previous value of ρ is retained. Step 4 simply reset the value of ρ back to 0.70 when the

target frequency is about to hop to the subsequent hop frequency. ρ Reset only when the value of $\psi[n]$ is more than $V_{threshold}$. Every time β tries to lock on to the new target frequency, the value ϕ_β tend to approach 0. As β locks on to the target frequency, the recursive calculation (5.27) of ϕ_β keeps on growing until the subsequent target frequency (illustrated in Figure 5.25(a) red colour curve), which control by how much the value of ρ in (5.26) has to increase. Initially ϕ_β is smaller; therefore, ρ converges at a fast rate, but as ϕ_β get larger in size, and the ρ is updated at a very less rate.

Algorithm IV: Adaptation of $\rho(n)$ with constraints

Input: $x[n]$

Output : $y_L[n]$

Initialisation: $\rho[n] = 0.70$; $\mu_\rho = 0.03$; $\gamma = 0.90$

```

1:   For  $n = 1:1:N_{sample}$ 
2:        $y_L[n] = \mathbf{latticfilter}(\beta[n], \rho[n], x[n])$ 
3:        $\psi[n] = \gamma\psi[n - 1] + (1 - \gamma)|y_L|^2$            ▷ start of new hop
4:   Update  $\rightarrow \beta[n]$                                        ▷ frequency
5:   If  $n > j + 120$                                            ▷ detection parameter
6:       If  $0.70 < \rho[n] < 0.95$                                ▷ wait for 120 samples at
7:            $\rho[n] = \rho[n - 1] + \mu_\rho \times y[n] \times g[n]/\phi_\beta$  ▷ start of new hop
8:            $\rho[n - 1] = \rho[n]$                                ▷ frequency
9:       end
10:      If  $0.70 > \rho[n] > 0.95$                                ▷ Retain  $\rightarrow \rho[n - 1]$ 
11:           $\rho[n] = \rho[n - 1]$                                ▷ previous value of  $\rho$  if it
12:      end                                                   ▷ exceed 0.95 or if  $\rho$  is
13:      If  $\psi[n] > V_{threshold}$                                ▷ below 0.70
14:           $\rho[n] = 0.70$                                        ▷ Reset  $\rightarrow \rho$  to minimum
15:           $j = N_{sample}$                                        ▷ value
16:      End                                                   ▷ 0.70 at the start of new
17:  End                                                       ▷ hop
18:  End                                                       ▷ frequency

```

5.5 Simulation Results

Now correct algorithm for resetting ρ has been developed and identified. Along with that, a full gradient approach is used, which is derived at the start of section 5.4. A comparison of simulation results is carried out in this section for tracking hop frequency signal interference. The research work [48] specifically describes the mitigation of CWI interference from GPS signal via lattice-based adaptive notch filter and is considered for comparison. Work in [48] is driven from [81-82], and [81] is already modelled in section 5.3.1. In [48] author only adapts signal parameter β , and with variable step-size.

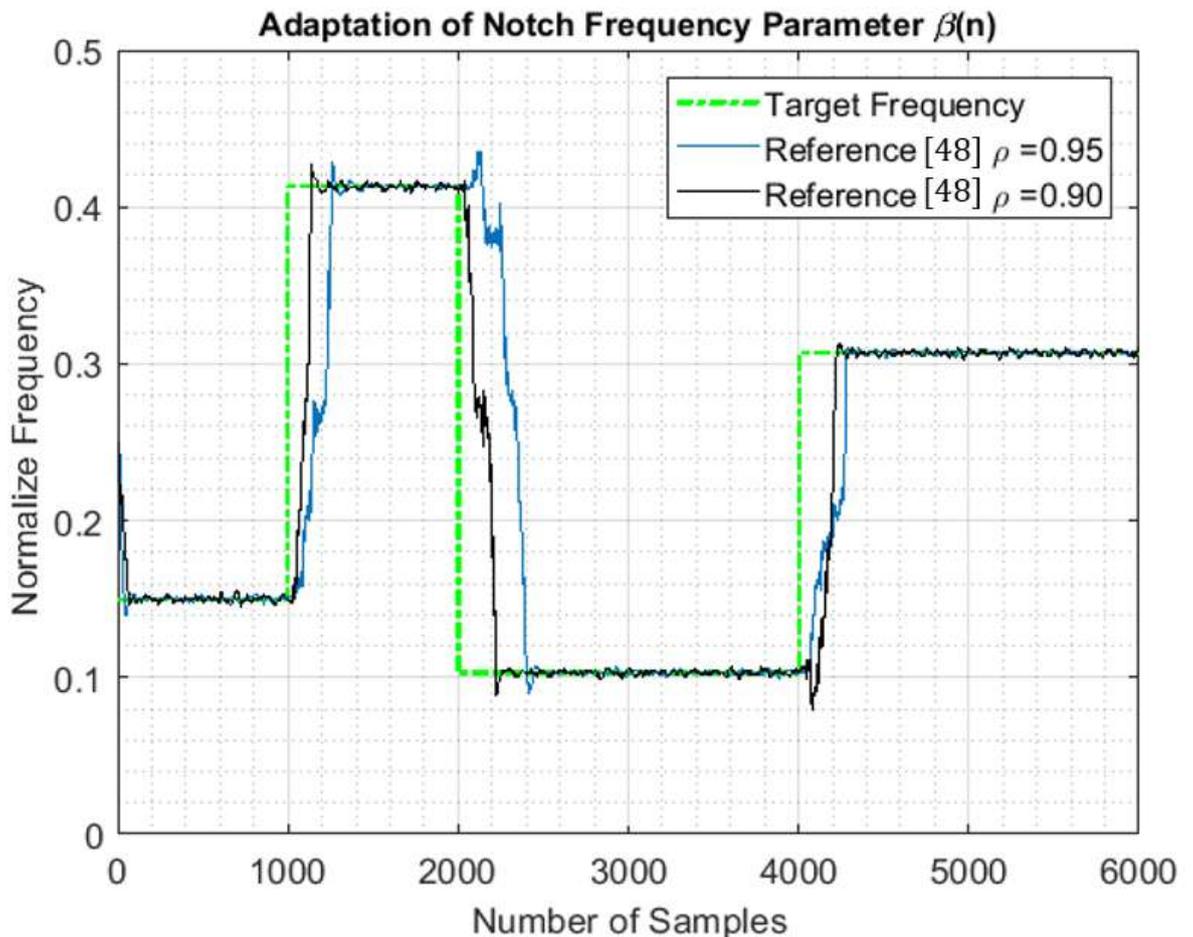


Figure 5.32 Tracking Performance of Lattice-based ANF when only β is adapted [48]. Blue colour line show tracking of ANF when $\rho=0.95$. Black line show tracking of ANF when ($\rho=0.9$)

Figure 5.32 demonstrates the performance of lattice-based when single parameter β is adapted. Two sets of simulation results are produced, one with ρ equal to 0.90 (black curve) and the other with ρ equal to 0.95 (blue curve). It is evident from the simulation results that with a higher value of ρ (approaches to unity), consideration amount of tracking performance of the ANF is lost. Simulation is done with these values as a tighter notch is desired in practical application. Even the black curve with a lower value of ρ is offset by some margin. It takes longer to locate the target frequency because the gradient of the cost function when ρ is closer to unity is flat (as described in section 5.3.2) and takes longer to converge to the target frequency. The same simulation parameters were used to simulate the proposed method, and Figure 5.33 shows the tracking performance of the proposed structure (Figure 5.19) and constraint adaptation algorithm for the updating ρ . And simulation results in Figure 5.33 demonstrate that the proposed method's frequency tracking performance is much superior to the one shown in Figure 5.32. In Figure 5.33, the proposed fully adaptive ANF is able to converge to the next subsequent hop frequency signal very quickly, hence reducing filter transient endure and preserving the useful signal. Figure 5.34 shows the corresponding adaptation of parameter ρ via the proposed algorithm. Figure 5.35 shows a tracking comparison of [48] and the proposed method when the simulated results are superimposed. Both black and red curves struggled to locate the target frequency at the beginning of the subsequent hop frequency. This delay in locating the next subsequent target frequency enables a small interference signal to leak into the system, thus not completely removing the undesired signal. Hence leads to inefficient excision of the interference signals. In contrast, the newly proposed algorithm optimizes the performance of the lattice-based ANF by simultaneously adapting the notch bandwidth parameter ρ and notch centre frequency parameter β . The red curve in Figure 5.35 demonstrates the superior performance of the proposed fully adaptive notch filter in terms of convergence speed and tracking

capabilities. It follows the target frequency (green line dash line) more precisely compared to [48].

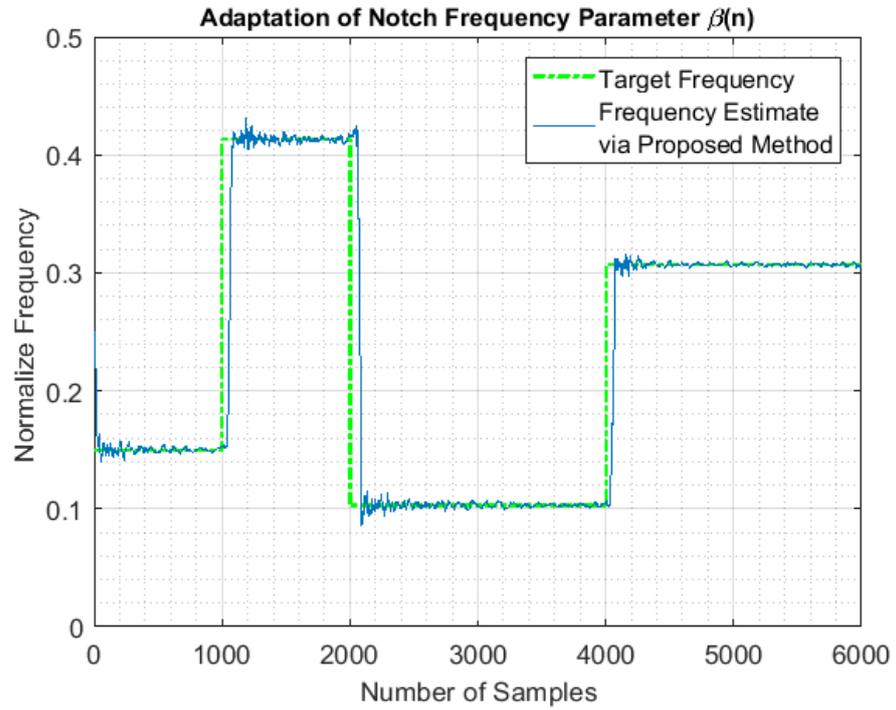


Figure 5.33 Tracking performance of ANF via Proposed Method. Simultaneously adapting both parameters ρ and β

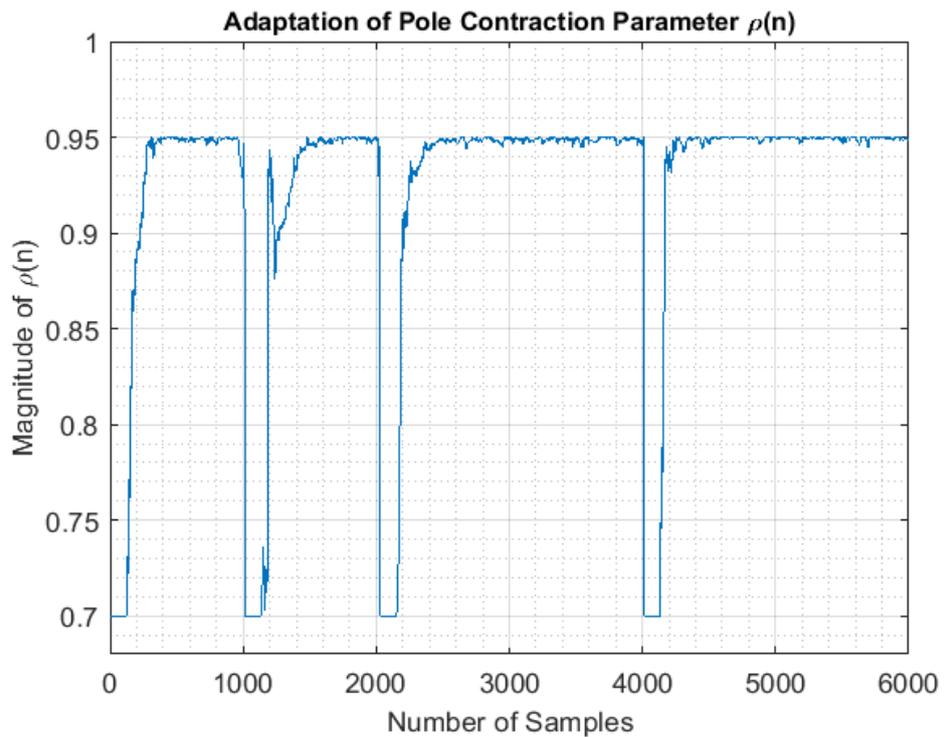


Figure 5.34 Corresponding adaptation of parameter ρ for above diagram

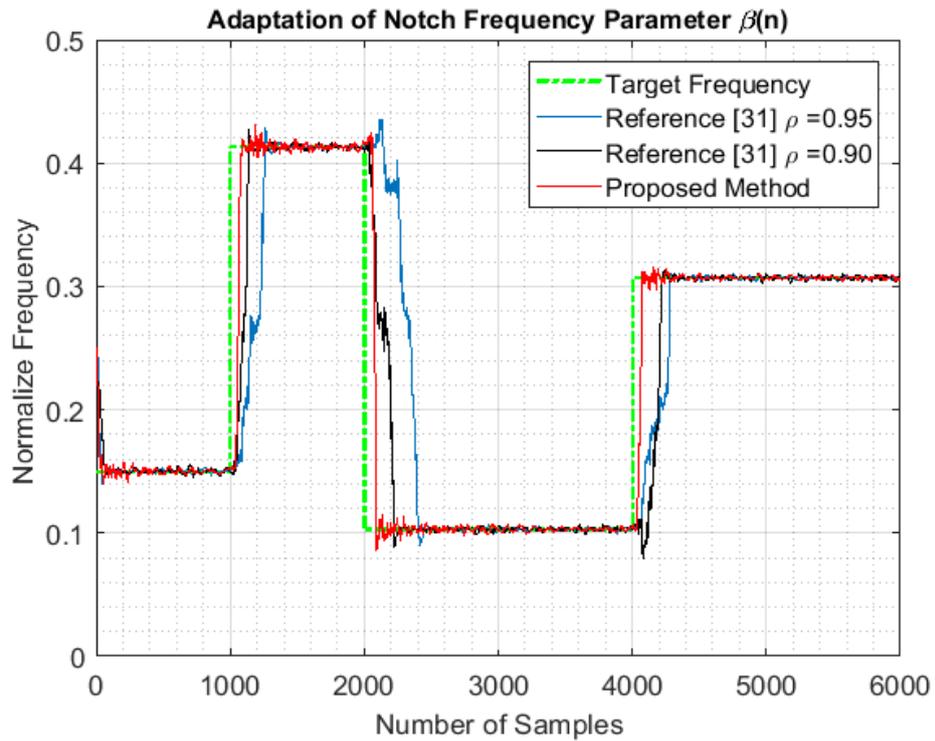


Figure 5.35 Frequency tracking comparison between [48] and the Proposed Method

5.6 Complete Modelling of the System

A complete set-up is modelled in MATLAB and SIMULINK to ensure uniformity throughout both adaptive algorithms ([51] and proposed) under test to perform an analysis of two types of filters. Figure 5.37 shows a typical GPS receiver's front-end and acquisition module. It consists of multistage IF conversion, LNA with a gain of 110 dB [87] and ADC (with 4, 8 and 16 bits). GPS signal Generator is also modelled as described in chapter one of this thesis.

ADC Requirements: An Analog to Digital Converter (ADC) is used to turn live analogue signals into digital values with certain quantization levels. ADCs use a certain number of bits when converting signals, and the number of bits used to digitize the RF signal determines the resolution of the measured signal. For instance, a 16-bit ADC can convert the signal into 65,536 different levels, whereas an 8-bit ADC only converts it into 256 levels. Most low-cost commercial GNSS receiver employs 1-bit to 3-bits. Further increasing the bit depth above 3-bits doesn't contribute to improvement in the C/N_0 value [88].

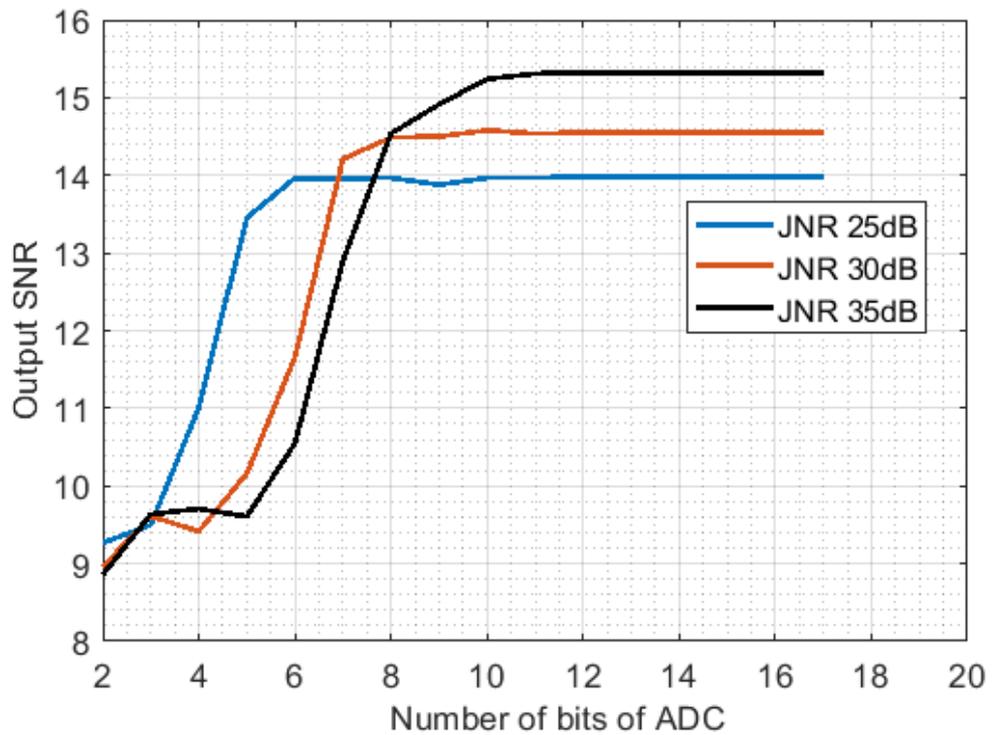


Figure 5.36 Number of bits required to acquire signal with different levels of JNR

In general, the more bits, the better the input signal reproduction. The dynamic range of an ADC is calculated by dividing the maximum and minimum amplitude and converting them to decibels. Each bit in ADC uses increases the dynamic range by 6dB. By applying this criterion, the GPS L1 signal from the output generator and the jamming signal are digitized with 18 bits to accommodate the high power interference and ensure the signal is fully preserved for processing via ANF.

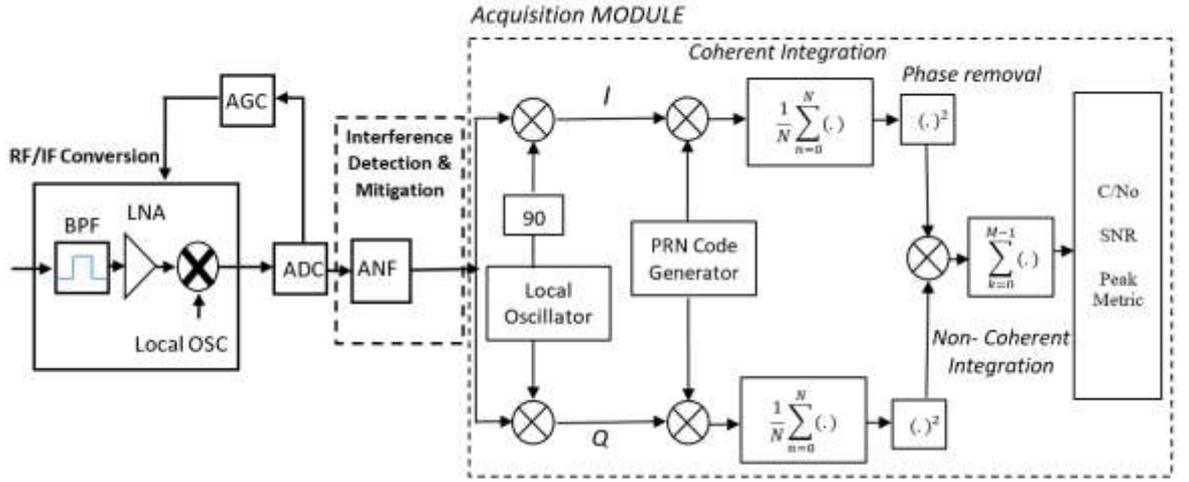


Figure 5.37 Complete System Modelling for testing of the proposed algorithm

Figure 5.36 shows a relationship between level JNR and the number of bits required by ADC to fully preserve the signal at its output. For each JNR level of 25dB, 30dB, and 35dB, approximately 5-bits, 7-bits, and 9-bits of ADC are required, respectively. As observed from the diagram, output SNR improves as the level of JNR increases. It is due to the fact that as the level of interference increases, ANF adaptive more rapidly and with less variation, excise interference more precisely, as explained right at the end of chapter 4.

SNR and C/N₀ In an ideal scenario, after evaluation of the Cross Ambiguity Function (CAF), it should present with well-defined sharp peak and the values of the peak corresponding to the value of τ_0 and F_D which matches the delay and Doppler frequency of the satellite in space. CAF is evaluated via (5.31). One very basic and still power metric that can evaluate the strength of the signal is the SNR at the output of the acquisition block. SNR in the CAF can be calculated [66] by equation (5.34).

$$S(\tau_0, F_D) = \left| \frac{1}{N} \sum_{n=0}^{N-1} r[n] c[n - \tau_0] e^{-j2\pi F_D n} \right|^2 \quad (5.34)$$

$$SNR(dB) = 10 \log \left[\frac{\max[S(\tau_0, F_D)] - \text{mean}[S(\tau, F_D)]}{\text{s.t.d.}[S(\tau_0, F_D)]} \right] \quad (5.35)$$

In GPS signal acquisition, pre-correlation SNR is negative, and post-correlation SNR is positive. Hence it is convenient to normalize the SNR to a 1Hz bandwidth. By doing so, the result is referred to as a carrier to noise density, usually measured in decibels (5.36). B is the bandwidth of that stage of the filter.

$$\frac{c}{N_0} = 10 \log_{10}(SNR \times B) \text{ [dB - Hz]} \quad (5.36)$$

Acquisition Module: Very first operation performed by the GPS receiver is the acquisition of the GPS L1 signal. The acquisition module in the receiver determines whether the satellite signal is present or absent and calculates the coarse value of the code phase delay τ_0 and the Doppler frequency F_D of the incoming signal. Acquisition of GPS signal is carried out by simultaneously searching for Doppler frequency range and all possible code phase offsets. There are various methods to evaluate Cross Ambiguity Function during acquisition, such as i) Serial Search, ii) Parallel Acquisition in the time domain, and iii) Parallel Frequency Code Phase. A later method is used, which provides efficient FFT while evaluating CAF and fast computation compared to i) and ii).

Figure 5.38 shows the CAF evaluation of the jammed signal at the output of the acquisition module. As no clear peak is visible, thus GPS L1 signal is not acquirable. Both the direct form algorithm [51] and the proposed method are compared in terms of output C/N_0 when Jamming to Noise Density (J/N_0) of different levels is applied.

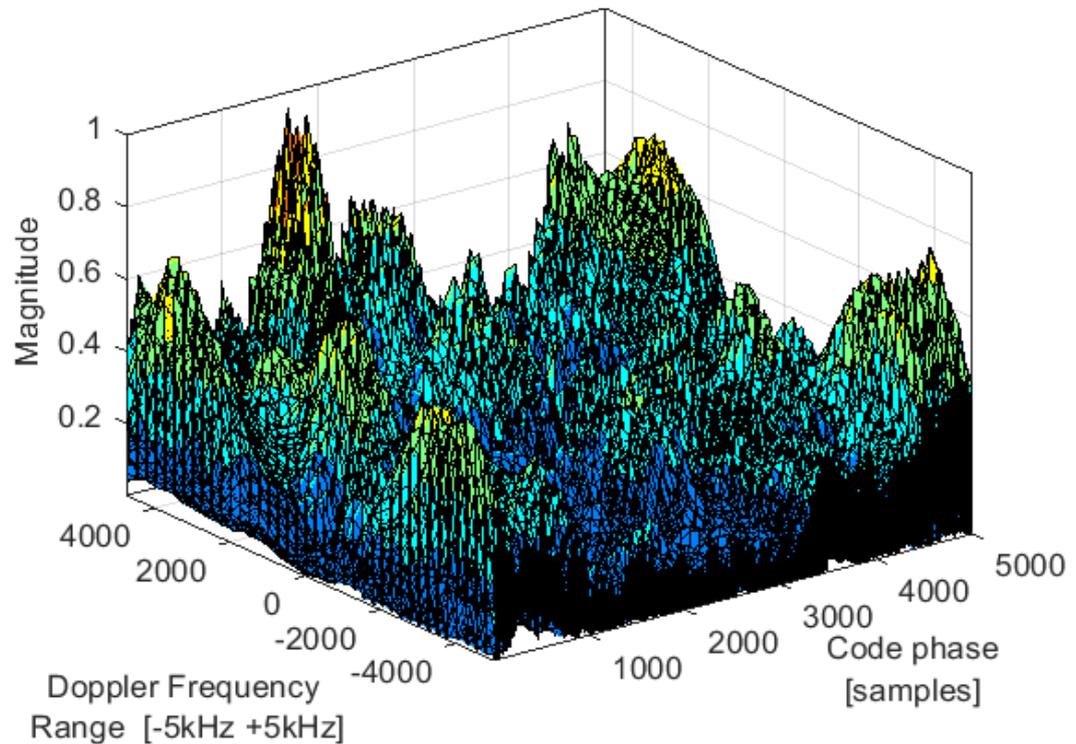


Figure 5.38 Evaluation of CAF at the output of acquisition with CWI hop frequency interference

Keep in mind that JNR is different from J/N_0 the parameter, in J/N_0 jamming power is set against the value N_0 with is specified in chapter 3 equation (3.11). The modelled GPS signal generator in Simulink is used to produce the signal under test with J/N_0 ranging from 0 -160 dB-Hz. Both [51] [67] the direct form-based ANF and the proposed algorithm are employed to remove and excise CWI hop frequency interferences. Figure 5.39 shows each notch filter's performance in terms of preserving useful signals after ANF processing. Figure 5.39 (a) shows the PSD of the jamming signal along with this GPS L1 signal. Figure 5.40 (b) shows the PSD of the signal after mitigation of the hop frequency interference through the proposed

method (fully adaptive notch filter). These blue circles represent the small notches the ANF created with minimal distortion on the useful signal. The same jammed GPS L1 signal was used to test [51], and Figure 5.39(c) shows the PSD of the output signal of direct form ANF [51]. It can be clearly observed by the PSD curve that useful signal is distorted, and Direct Form IIR notch under such conditions is unable to completely mitigate the interference. Furthermore, it introduces the scaling of the signal around the notch centre frequency.

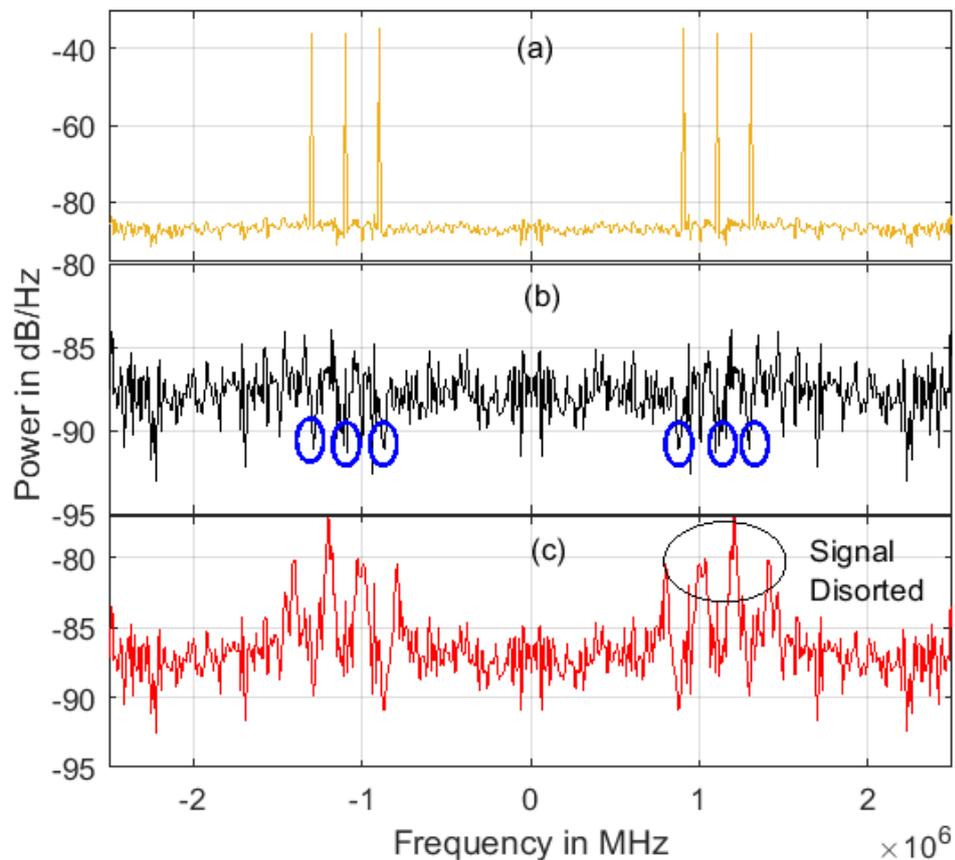


Figure 5.39 PSD of useful signal with the interference of $J/N_0=100\text{dB-Hz}$. (b) PSD of Interference excision via the proposed method. (c) Interference excision via method published in [51] and [67]

Figure 5.40 presents simulation results for C/N_0 at the output of the acquisition block. The brown curve shows the value of output C/N_0 when the jamming signal is present and C/N_0 drops sharply as J/N_0 value gets above 65dB-Hz . At this point (up to $J/N_0=65\text{dB-Hz}$) jamming signal is almost the same size in amplitude as a useful signal. Hence, the signal can be acquired without using ANF because the CDMA signal is jamming resistant for low-

powered interference signals. However, as J/N_0 gets above 80 dB-Hz output C/N_0 level becomes lower than 41 dB-Hz for the brown curve and GPS L1 signal becomes acquirable via acquisition module. The red curve shows simulation results when a direct form structure is used to mitigate the interference [51]. The C/N_0 start to drop even before J/N_0 reaches above 45dB-Hz. This is due to the reason that the adaptive parameter z_0 in [51] is unable to converge when the interference level is smaller than the size of the signal and produces a random notch with mild depth which degrades the useful signal quality hence degrading the C/N_0 value. For J/N_0 between 65-110dB-Hz, the direct form algorithm performs steadily and output C/N_0 value is almost constant and close to 43dB-Hz. As the J/N_0 increases above 120dB-Hz, direct form based ANF's performance rapidly degrades and unable re-attain useful signal, due to slow convergence of parameter z_0 to subsequent hop frequency, as shown in Figure 5.8(section 5.3.1).

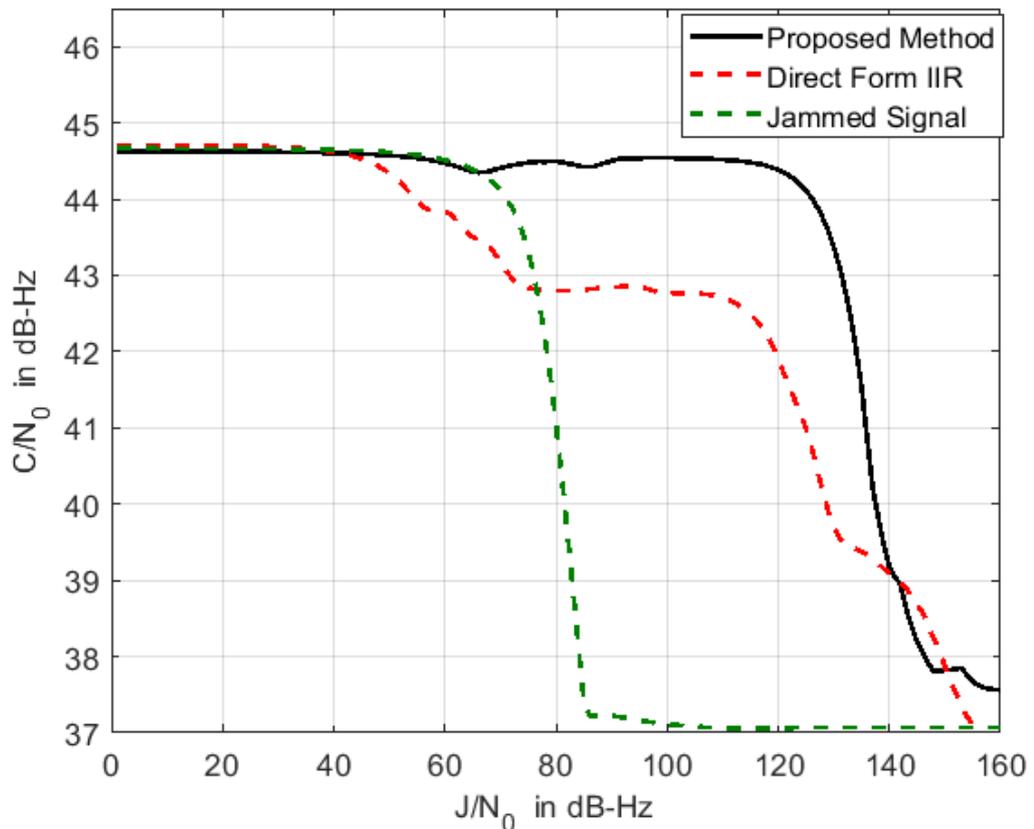


Figure 5.40 C/N_0 At output of the acquisition module. Red curve represents the Direct form IIR. The black curve represents lattice-base ANF with the proposed algorithm

The performance of the lattice-based ANF with the proposed full gradient adaptation of notch bandwidth parameter ρ with constraints is shown by the black curve in Figure 5.40. Its performance is much better than the 2nd order direct-form ANF. It can mitigate the higher level of frequency hop jamming signal due to the fact that by simultaneously adapting both parameters, the transient endurance by ANF at the beginning of each the hop frequency is reduced, hence, preserving the useful signal.

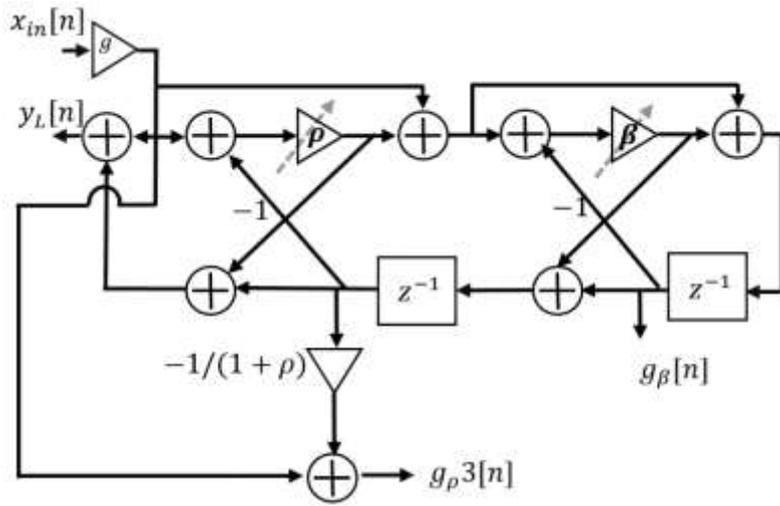
5.7 Hardware Complexity Reduction of the Proposed Structure

The proposed structure to implement the full gradient adaptation of the notch bandwidth parameter is earlier shown in section 5.4, Figure 5.19. An extra filter is required to implement this method, which increases the hardware complexity of the system. In order to simplify hardware and computational complexity, heuristically different gradient signals were generated from the structure. As shown in Figures 5.41(a), (b) and (c).

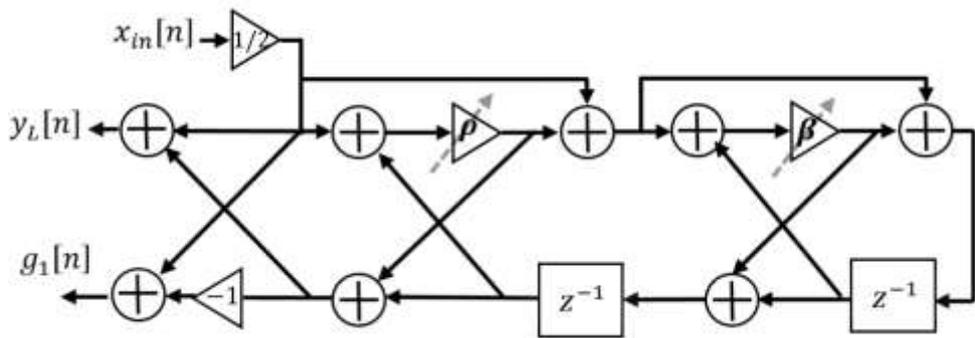
The frequency response of the full gradient signal g_p (Figure 5.19 of section 5.4) is plotted in Figure 5.42. Its transfer function is given as follows

$$G_p(z) = \frac{1-z^{-2}}{1-\beta(n)(1+\rho(n))z^{-1}+\rho(n)z^{-2}} \times \frac{1}{1+\rho} \quad (5.37)$$

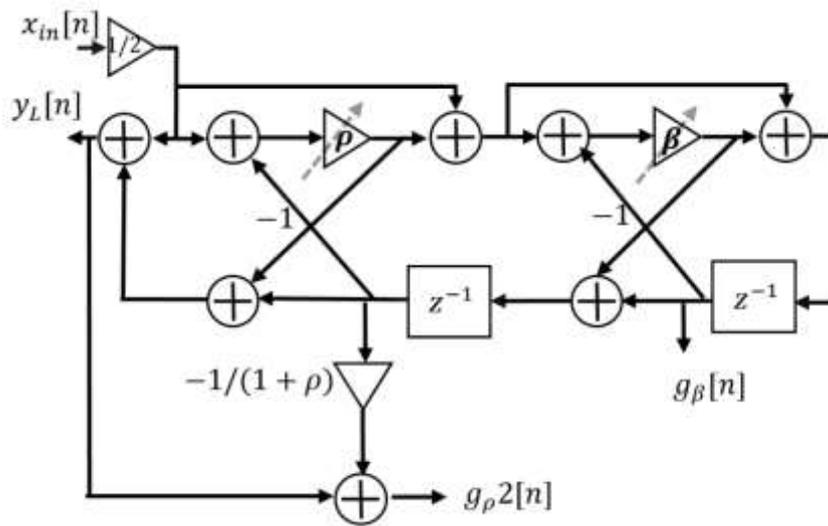
It has a band-pass response, as shown in Figure 5.42, with zeros at DC and Nyquist.



(a)



(b)



(c)

Figure 5.41 Three gradient signal under test (a) represents g_p3 (b) g_1 (c) g_p2

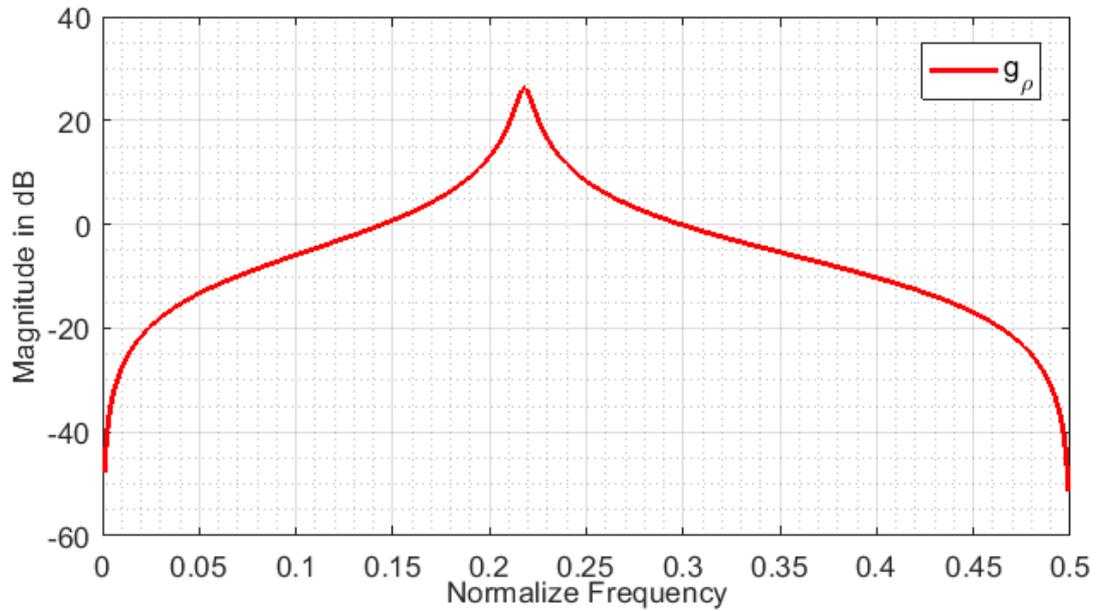


Figure 5.42 Frequency Response of $G(z)$

Now the frequency response of these gradient signals g_{p3} , g_{p2} and g_1 are also plotted to verify whether any of these signals could be adopted as a gradient signal to simplify the structure in Figure 5.19. The figure below shows three curves, yellow, black, and blue, for g_1 , g_{p3} and g_{p2} respectively.

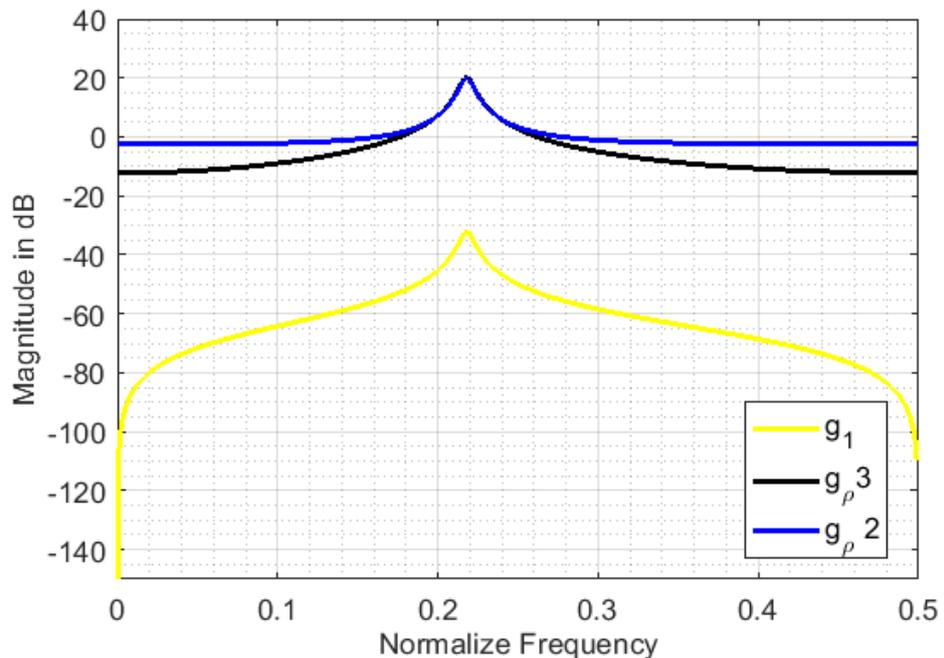


Figure 5.43 Frequency response of the gradient signals g_{p3} , g_{p2} and g_1

The pass-band gain of g_p3 and g_p2 is almost the same in magnitude as for g_ρ in Figure 5.42, but its stop-band rejection is not as good as that of g_ρ . But the g_1 exhibits the same inherent curve shape as that of g_ρ , but with much lower gain levels. Analysing this simple similarity between g_ρ and g_1 , a gain factor can be applied to g_1 to alleviate it to the same level as g_ρ . The transfer function of from $x_{in}[n]$ to output $g_1[n]$ (Figure 5.41c) given in [37], and its z-transform is equal to (5.38)

$$G_1(z) = \frac{1-\rho}{2} \times \frac{1-z^{-2}}{1-\beta(1+\rho)z^{-1}+\rho z^{-2}} \quad (5.38)$$

Now $G_1(z)$ of Figure 5.41 can be converted to $G_\rho(z)$, but simply adding a gain factor

$$G_1(z) = \frac{1-\rho}{2} \times \frac{1-z^{-2}}{1-\beta(1+\rho)z^{-1}+\rho z^{-2}} \times \frac{1}{1+\rho} \times \frac{2}{1-\rho} \quad (5.39)$$

Now multiplying last to gain factors added on the left side of the transfer function yield the following expression

$$G_1(z) = \frac{1-\rho}{2} \times \frac{1-z^{-2}}{1-\beta(n)(1+\rho(n))z^{-1}+\rho(n)z^{-2}} \times \frac{2}{1-\rho^2} \quad (5.40)$$

Which becomes equivalent to the following expression

$$G_\rho(z) = G_1(z) \times \frac{2}{1-\rho^2} \quad (5.41)$$

Hence structure in Figure 5.19 can be modified and simplified as follow.

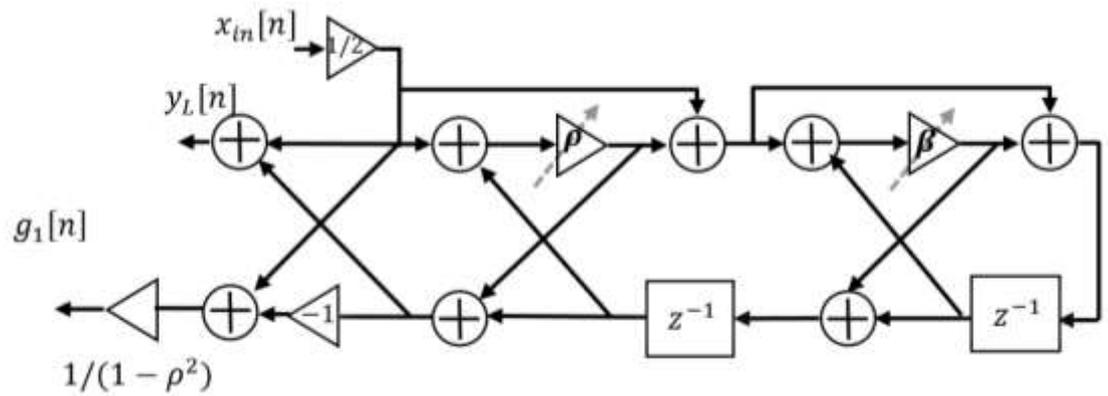


Figure 5.44 The proposed modified gradient signal g_1 for the adaptation of notch bandwidth parameter ρ

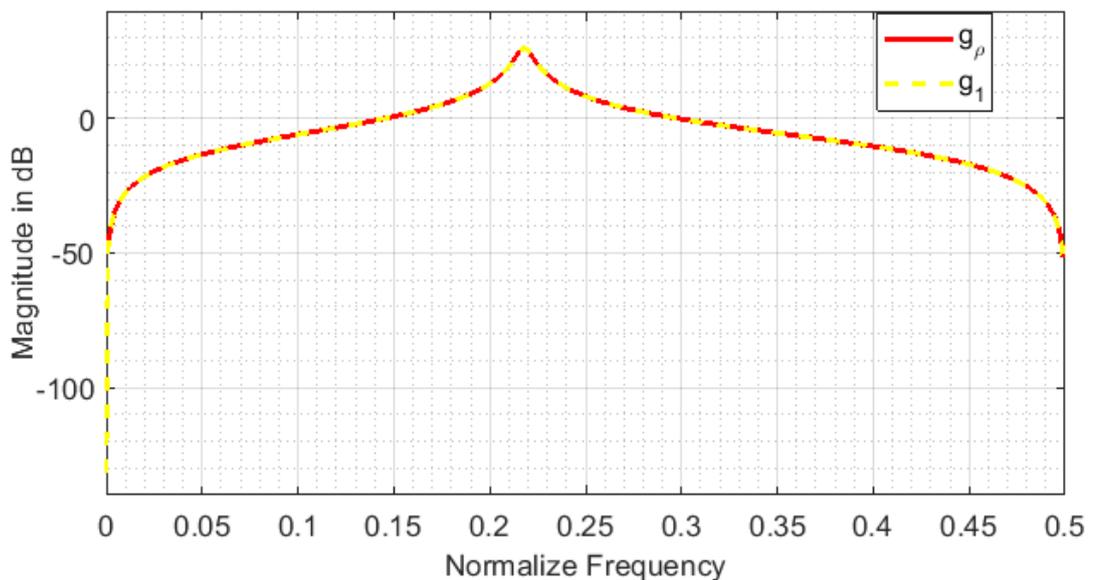


Figure 5.45 Frequency response of g_ρ in red and g_1 yellow

Illustrates the effect of adding this $(\frac{2}{1-\rho^2})$ gain factor and comparison between g_1 and g_ρ .

From this Figure, it can be concluded that this new g_1 gradient signal has the same response as $G_\rho(z)$. It can be generated within the lattice-based adaptive notch filter without cascading an extra filter.

The computational complexity for all methods is compared in table 5.5. As the complexity of ANF increases, there is an improvement in its performance of ANF. As [68] is the least complex in terms of computational burden, it performed miserably whilst tracking hop

frequency interference (as shown in Figure 5.7). And as the complexity increases for both [48] and [51], both performed better than [68] and were able to track hop frequency. Unfortunately, it has to go through a long transient phase at the start of each new hop frequency signal, degrading its performance severely. The computational complexity of the proposed algorithm in section 5.4 is almost double that of the other counterpart. It comes with the advantage that its performance is superior both in terms of output C/N_0 and convergence speed. Moreover, in the proposed algorithm, we are adapting an extra parameter ρ ; hence a number of computations per sample tend to increase. There has to be some kind of trade-off between the quality of interference excision and the computational complexity of the system. GPS-based navigation systems demand high precision and accuracy, and today's high-speed DSP processor can accommodate an increase in computational complexity by two/ three folds. A reduction in the term of computational complexity is achieved by simply generating a gradient signal within the structure itself, as shown in Figure 5.44.

Further simulation results of the proposed structure and its algorithm are compared with research carried out in [81]. In the previous section of this thesis, [81] was modelled in MATLAB without any constraints on the notch bandwidth parameter ρ . Results show that parameter ρ tend to overshoot unity, as presented in Figures 5.3 and 5.6. Hence motivated to develop a full gradient adaptation of parameter ρ with constraints as illustrated in section 5.4 of this thesis.

Now adaptive notch filter in [81] is modelled with constraints on the notch bandwidth parameter ρ . A comparative analysis is made in terms of the convergence of parameters ρ and C/N_0 at the out of the acquisition module between the proposed structure/proposed algorithm II and reference work [81]. Research work in [81] is implemented again in

MATLAB with **proposed algorithm IV** incorporated within the adaptive notch filter of [81].

Table 5.5 Comparison of the existing method and proposed method

	Reference [31]	Reference[34]	Reference [51]	Proposed Method Fully Adaptive Notch Filter – Full gradient term
Structure	Direct form IIR	All-pass Lattice-based structure	Direct Form IIR	All-pass Lattice-based structure
Adaptation of Notch Centre Frequency	Yes	Yes	Yes	Yes
Adaption of Notch Bandwidth Parameter	is Fixed	is Fixed	is Fixed	Yes - adaptive
Update Algorithm	Normalize LMS	Variable Step-Size LMS	Signed Magnitude Variable Step-Size LMS	Variable Step-Size LMS
Computational Complexity per sample	+S × M ÷ D	+S × M ÷ D	+S × M ÷ D	+S × M ÷ D
	8 15 1	7 12 2	7 7 1	14 19 4

Computational Complexity per sample Of Modified Structure (Figure 5.44)	+S	× M	÷ D
Figure 5.44	10	15	3

Both filters have different gradient functions and update equations for notch bandwidth parameter ρ , as shown in Table 5.6. The update equation of ρ for the proposed structure has a variable step-size, while the other one has a fixed step-size. In this section, the goal is to model both of these adaptations of parameter ρ with developed constraints algorithm II in section 5.4.4 of this thesis.

Table 5.6 Comparison of two filters in terms of transfer functions, gradient signals (for ρ and β) and update equations

	Proposed Method	Reference [81]
Transfer Function	$H_L(z) = \frac{1 + \rho}{2} \frac{1 - 2\beta(n)z^{-1} + z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}$	$H_L(z) = \frac{1 - 2\beta(n)z^{-1} + z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}$
Gradient Signal $G_p(z)$	$G_p(z) = \frac{2}{1 - \rho^2} \frac{1 - z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}$	$G_p(z) = \frac{\beta z^{-1} - z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}$
Update Equation for $\rho[n]$	$\rho[n] = \rho[n - 1] + \mu_\rho y_L[n] g_\rho[n] / \phi_\beta[n]$	$\rho[n] = \rho[n - 1] - \mu_\rho y_L[n] g_\rho[n]$
Gradient signal for $\beta[n]$	$G_\beta(z) = \frac{(1 - \rho(n))z^{-1}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}$	$G_\beta(z) = \frac{z^{-1}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}}$

In this section series of simulations is presented under different conditions and hence provide deeper insight into the adaptation of parameter ρ in both these variants. The main aim is to determine each filter's performance and provide a fair comparison by considering relevant research work. The following sets of simulations are obtained while keeping the value of ρ fixed at 0.50, 0.80 and 0.98. As per the ideal notch width value of ρ must be as close as possible to the unity. To provide an extremely narrow notch width to minimise distortion of useful signal in the system for a fixed/static notch filter. But this might not be the case for an adaptive notch filter. An adaptive notch can be characterised by fast locking on to the

target frequency and reducing the transient time for each subsequent hop frequency. By doing so, the performance of the ANF enhances for any given application.

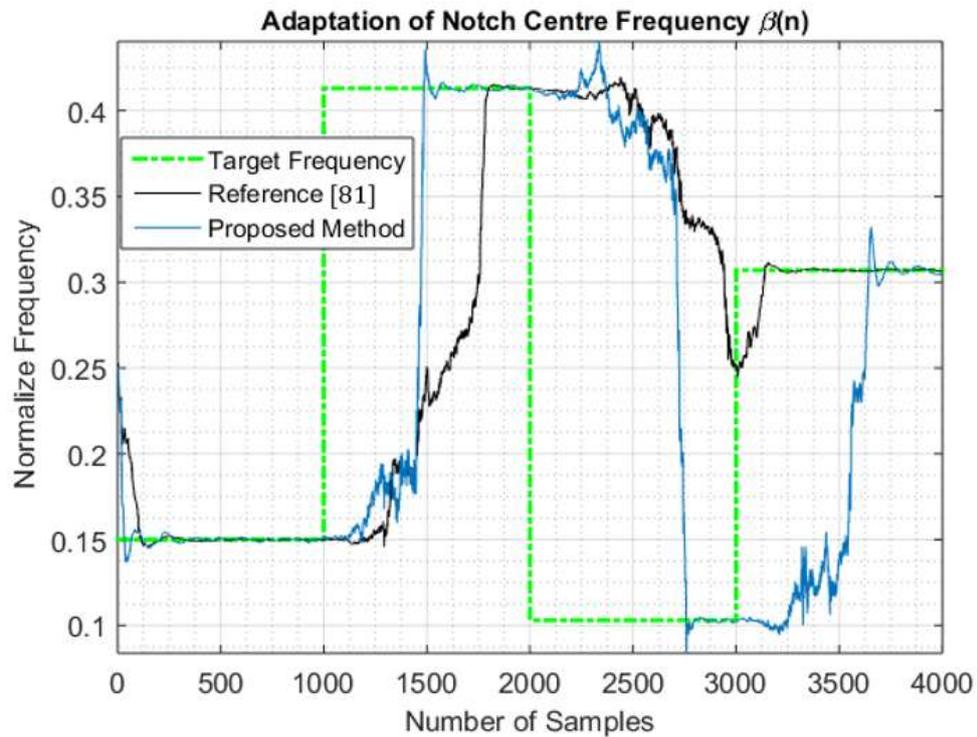


Figure 5.46 Comparison of tracking of hop frequency CWI with ρ fixed at 0.98 for both the proposed method and [81]

Figure 5.46 shows if the value of pole contraction parameter ρ is close to unity, and in this case, is it 0.98, providing an extremely narrow notch width. For both adaptive notch filters, the detection of the next subsequent hop frequency (target frequency) was delayed/lagged severely and unable to lock on to the target frequency on time. This causes the leaking of CWI into useful signal; hence, ANF cannot completely remove hop frequency CWI from the useful signal. In Figure 5.46 blue curve represents the proposed method, and the black curve shows the simulation results for [81] under similar conditions. The variable step-size of both is empirically chosen to provide the best possible results. Under static conditions (ρ is fixed) blue curve shows a reasonable tracking compared to the black curve. But overall,

both adaptive notch filters are unable to track target frequency and converge to the correct hop frequency at the right time.

In the next set of simulation results in Figures 5.47 and 5.48, parameter ρ is fixed at 0.50 and 0.80. Furthermore, the variable step size for the updating notch centre frequency parameter β is kept the same ($\mu_\beta = 0.018$) and compare the ability tracking of two ANF algorithms. In Figure 5.47, the value of the parameter ρ is 0.50, which is much lower and corresponds to a wider notch and fast convergence of β to the target frequency. This effect is illustrated in Figure 5.47 and Figure 5.48. Tracking is more in-line with the prescribed target frequency, represented by the orange dash line.

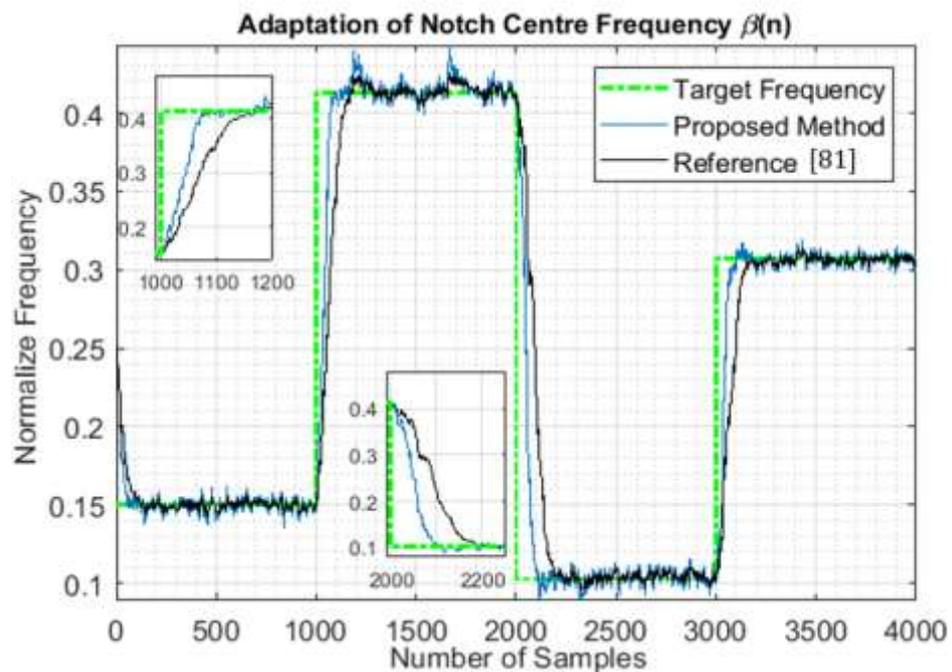


Figure 5.47 Comparison of tracking of hop frequency CWI with ρ fixed at 0.50 for both proposed methods and [81]

The two small zoomed-in graphs in Figure 5.47 clearly show how each of the algorithms converges to the target frequency and the blue curve tracks more precisely, and faster convergence is achieved compared to the black curve [81]. Figure 5.48 demonstrates that as

the value of the ρ is increased to 0.80 (providing a narrower notch width than 0.50), both algorithms lag as compared to the simulation in Figure 5.47. For both Figures 5.47 and 5.48, the value of step-size μ_β are kept the same and the purpose of these simulations is to verify and test the convergence of parameter β to the target frequency. From these simulations, it is concluded that both algorithms provide comparable similar results if the notch bandwidth parameter ρ is kept static (fixed). However, the proposed algorithm converges a lit quick than the other counterpart if the value of step-size μ_β for both is kept the same.

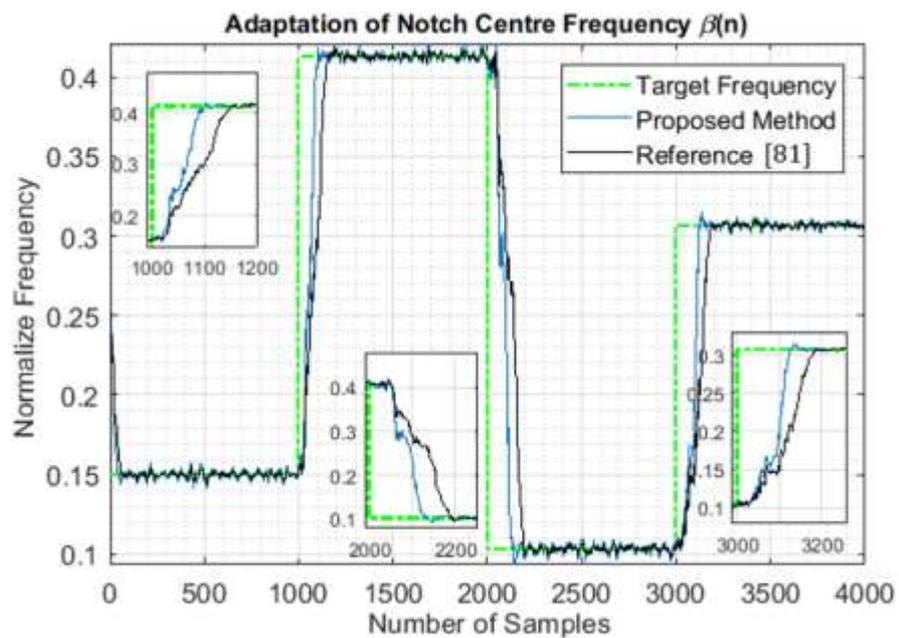


Figure 5.48 Comparison of tracking of hop frequency CWI with ρ fixed at 0.80 for both proposed methods and [81]

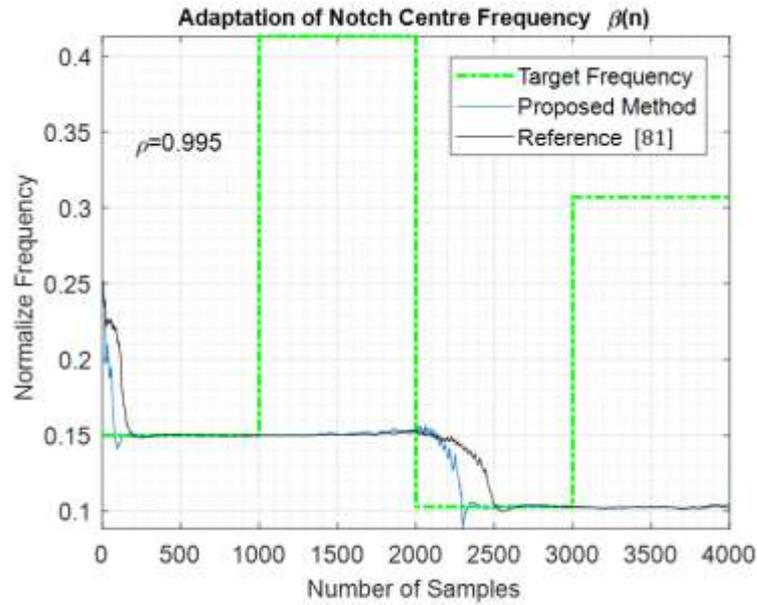


Figure 5.49 Comparison of tracking of hop frequency CWI with ρ fixed at 0.995 for both proposed methods and [81]

Four values of the pole contraction parameter ρ were used in the above simulations.

$$\rho(1) = 0.50 \quad \rho(2) = 0.80 \quad \rho(3) = 0.98 \quad \rho(4) = 0.995$$

The value $\rho(1) = 0.50$ is chosen as the smallest pole contraction parameter, which significantly affects the bandwidth of the notch filter and corresponds to a very wide notch, which is undesirable in practical use. Whereas value $\rho(4) = 0.98$ is extremely close to unity and produces a very narrow notch. Both algorithms' ability to work with poles extremely close to unity diminishes, as shown in Figure 5.49, which means the filter is on the brink of instability. Therefore, the **proposed algorithm IV** monitors the stability of the filter. In the proposed algorithm, if the value of ρ exceeds 0.95, it reverts into the stable model and is reset in order to prevent divergence.

In the next round of the simulations, the algorithm presented in [81] is modelled as it is in MATLAB. But the adaptation of the notch width parameter ρ was carried out as per **proposed algorithm IV** in section 5.4.4 of this thesis.

The motivation here is to thoroughly test, verify and make a fair comparison between the proposed algorithm with full gradient term and relevant work in the field of simultaneous adaptation of both parameters β and ρ in lattice-based ANF filter. Previously comparative analysis of the proposed algorithm was carried out with direct form ANF [51] [67] lattice-based ANF [48] (specific for application GPS). All these algorithms utilized an adaptation of single parameter β , while notch width parameter ρ is kept fixed.

It is clear and evident from simulation results in Figure 5.35 (section 5.4) and Figure 5.40 (section 5.6) that proposed an algorithm with full gradient term and constraints adaptation of parameter ρ shows superior performance in terms of tracking (Figure 5.35) and C/N_0 at the output of the acquisition module (Figure 5.40). Now the task is to compare the proposed algorithm with existing relevant work, which adapts both parameters simultaneously, to ensure the novelty of the proposed method in this research work.

Table 5.7 Simulation parameters for reference [81]

Noise variance σ^2	0.25
μ_β step-size for β	0.018 - 0.022
μ_ρ step-size for ρ	Variable (0.003-0.01) adjusted to get the best possible results
'N' no. of samples	4000
Initial value of β	0.25 (normalized frequency)
Initial value of ρ	0.70 (Pole contraction factor)
γ forgetting factor	0.90

Update equation for the notch width parameter ρ in [81] is as follows.

$$\rho[n] = \rho[n - 1] - \mu_\rho y_L[n]g_\rho[n] \quad (5.42)$$

Where, μ_ρ is the fixed step-size set between 0.003 to 0.01, y_L is the output of the filter and g_ρ is gradient signal for the adaptation of parameter ρ , it is given by following the transfer function as in equation 5.43.

$$G_\rho(z) = \frac{\beta z^{-1} - z^{-2}}{1 - \beta(n)(1 + \rho(n))z^{-1} + \rho(n)z^{-2}} \quad (5.43)$$

The difference between the transfer function of two gradient signals are shown in table 5.5, and the update equation for the parameter $\rho[n]$ in the proposed method is as follows.

$$\rho[n] = \rho[n - 1] + \mu_\rho y_L[n]g_\rho[n]/\phi_\beta[n] \quad (5.44)$$

In the proposed method, variable step-size μ_ρ is used by dividing it by ϕ_β , which is the instantaneous power at the output of the gradient signal of the parameter β . Every time β tries to lock on to the new target frequency, the value ϕ_β tend to approach 0. As β locks on to the target frequency, the recursive calculation (5.27) of ϕ_β keeps on growing until the next target frequency (illustrated in Figure 5.25(a) red colour curve), which control by how much the value of ρ in (5.44) has to increase. Initially, ϕ_β is smaller. Therefore, ρ converges at a fast rate, but as ϕ_β get larger in size, then ρ is updated at a very lower rate. The proposed method is already presented in detail, with the derivation of the full gradient, constraints algorithm, and extensive simulation results in section 5.4 of this chapter.

As the fixed step size is used in [81], hence different values of step size are set up to produce the best possible adaptation of the notch width parameter ρ . The following simulations illustrate the effect of fixed step-size μ_ρ for [81], when proposed algorithm II is incorporated in it. Figure 5.50 presents the simulation results when μ_ρ is fixed at 0.003. It can observe from the simulation result that the notch width parameter ρ converges very slow and does not reach to prescribe upper limit of 0.95 in algorithm II. It means the average notch width

is wide, which might have a significant effect on useful signal and output C/N_0 . Hence the value of the fixed step-size μ_ρ is increased to 0.006 and its simulation results are shown in Figure 5.51. The parameter ρ does converge to upper limit of 0.95 but takes a longer time. Furthermore, the value of ρ does not stabilise around 0.95 and due to fixed step size, the value of ρ tends to spike down to around 1510 samples, 1850 samples, 2300 samples and 2500 samples points. Now step size μ_ρ is further increased from 0.006 to 0.009, and Figure 5.52 shows the comparison between two different step-size used in the simulation.

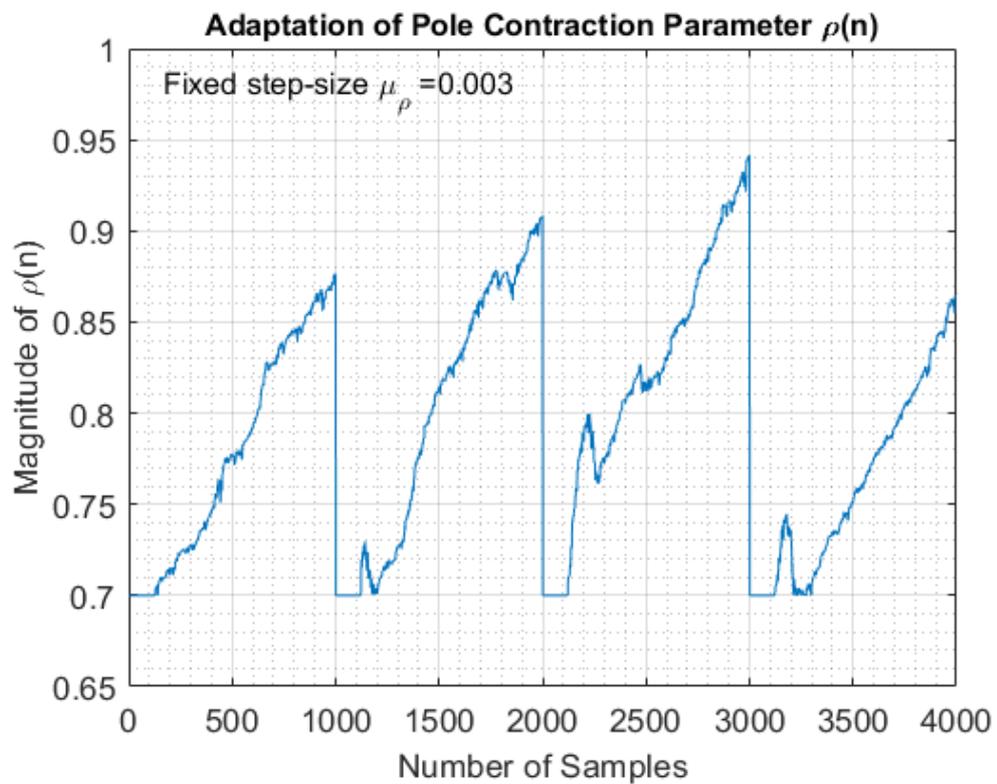


Figure 5.50 Adaptation of ρ with a partial gradient [81], μ_ρ is fixed at 0.003 and utilizing the proposed constraints algorithm

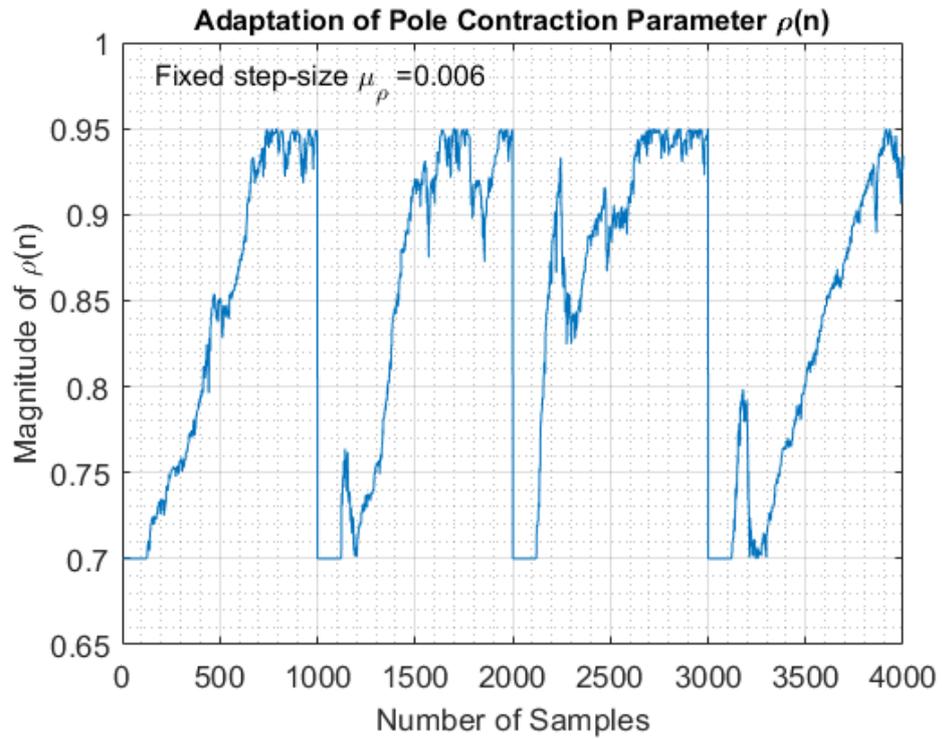


Figure 5.51 Adaptation of ρ with partial gradient [81], μ_ρ is fixed at 0.006 and utilizing proposed constraints algorithm

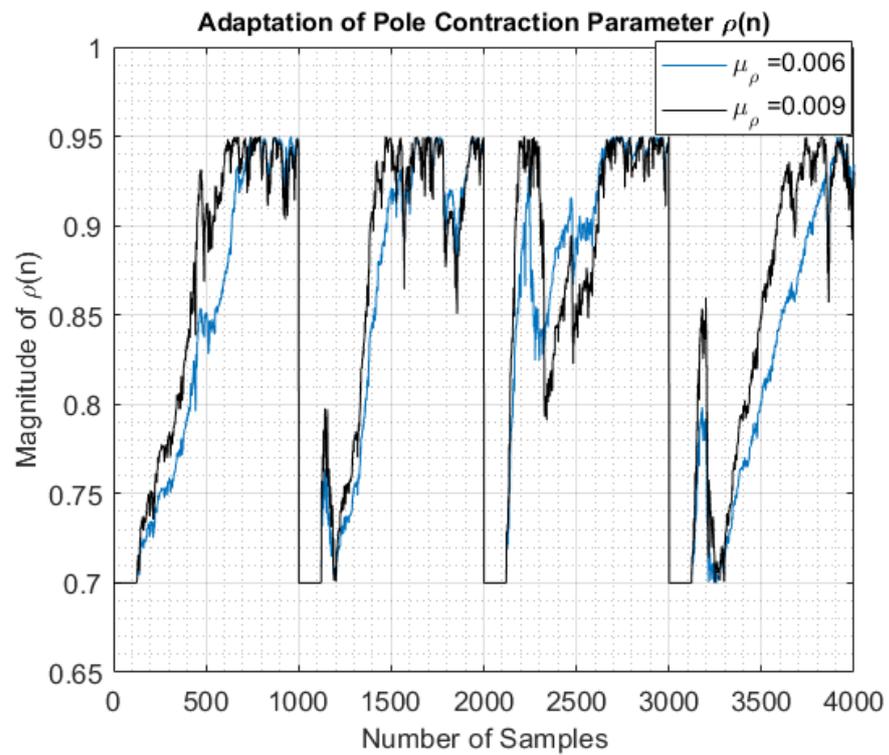


Figure 5.52 Adaptation of ρ with partial gradient [81], μ_ρ is fixed at 0.006 & 0.009 and utilizing proposed constraints algorithm

It is evident from Figure 5.52 as the value of step-size μ_ρ increase to 0.009, the amplitude of the spikes gets larger, and the number of spikes increases, the black curve in Figure 5.52 represent this effect. It means that after the parameter β has locked on to the target frequency, an adaptive notch filter is not in an ideal state (narrower notch width), and the width of the notch keeps on fluctuating. Each time when the parameter ρ spike down, the notch width gets wider, and it simply excises wider content in a spectrum and significantly affects the useful signal. More so over, initial converges of parameter ρ for different values of μ_ρ is also very slow compared to the proposed method. The effect of the slow converges of parameter ρ and intermediate spikes in Figure 5.52 are reflected in the form of the magnitude response of the ANF in the steady state in Figure 5.53. In Figure 5.53, the black curve shows the magnitude response of [81] implemented via the proposed algorithm II. The blue curve shows the simultaneous adaptation of both parameters via the proposed algorithm designed in section 5.4.1 of this chapter. The -3dB bandwidth for the black curve is much wider than the blue curve, which means more along with interference, some portion of the useful signal is being removed as well, degrading the quality of the useful signal.

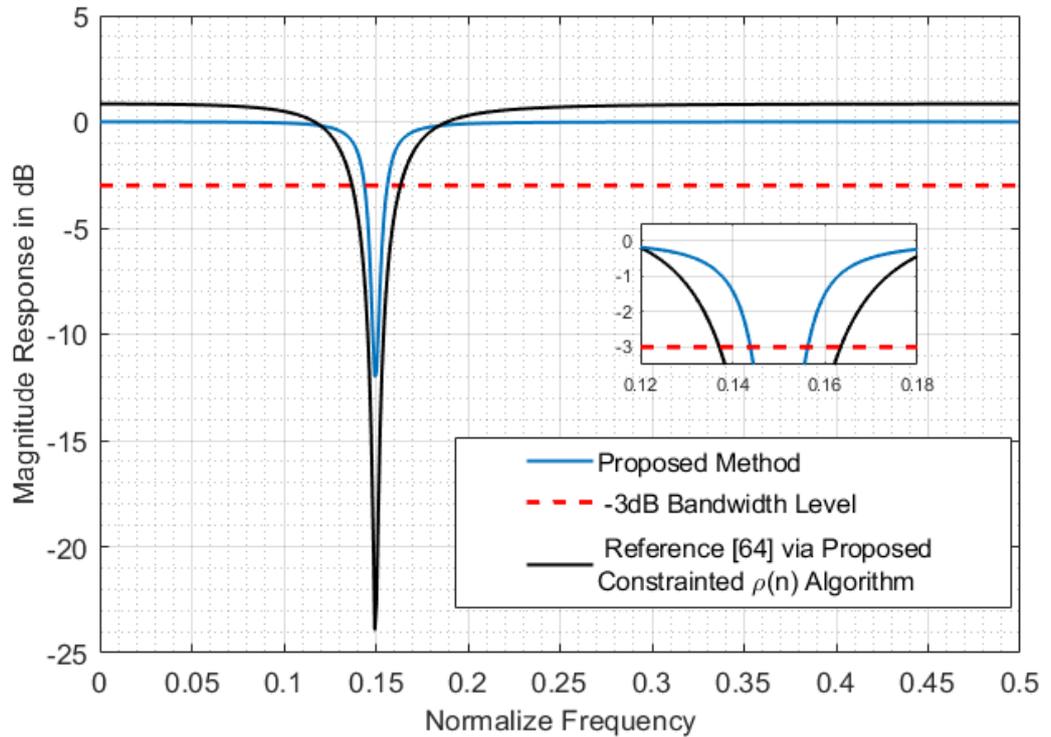


Figure 5.53 Magnitude response of both ANF in steady state at a normalized frequency of 0.15

Figure 5.54 shows this effect in a more elaborated manner. The magnitude response of both ANF is plotted in a steady state. Figure 5.54 (a) shows each of the notches (four notches) produced when the adaptive algorithm in [81] is used along with the proposed constrained algorithm for notch width parameter ρ . The red line represents the -3dB bandwidth level. Figure 5.54 (b) shows the simulation results for the proposed algorithm (full gradient term, variable step-size, and proposed gradient generated within the filter structure) developed in this chapter. Both these simulations were run under the same conditions.

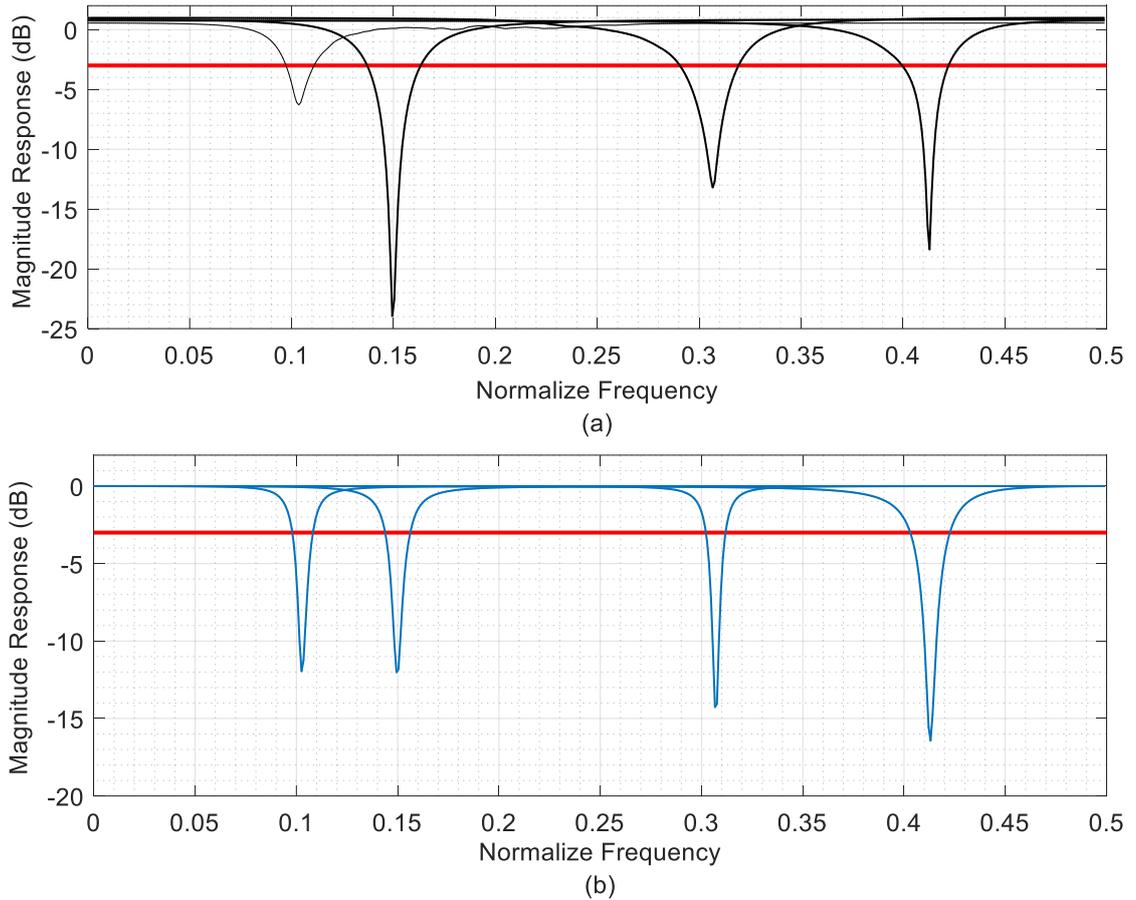


Figure 5.54 All four notch at target frequencies of 0.103, 0.15, 0.307 and 0.413. (a) Magnitude response of ANF in reference [81]. (b) Magnitude response of the ANF via the proposed method

Analysing both Figure 5.54(a) and Figure 5.54(b), it is clearly evident that the proposed method produces an ideal notch filter response. Once the parameter β have locked on the target frequency, all four notches' width is narrower. On the other hand, in Figure 5.54(a), the notches are produced with a wider notch, increasing the area of excision from the spectrum of the GPS L1 signal.

Furthermore, both algorithms were tested to mitigate hop frequency CWI in the GPS L1 signal. Recall, in section 5.6, a completed system was modelled to compare and evaluate the performance of direct form 2nd order ANF and the proposed method in terms of output C/N_0 at acquisition block. The same modelled and test signal is used to mitigate interference using reference [81] with the constrained ρ algorithm.

Figure 5.55 C/N_0 At the output of the acquisition module. Red curve represents the Direct form IIR. The black curve represents lattice base ANF with the proposed algorithm, and the blue curve represents results attained by reference [81] via constrained ρ algorithm

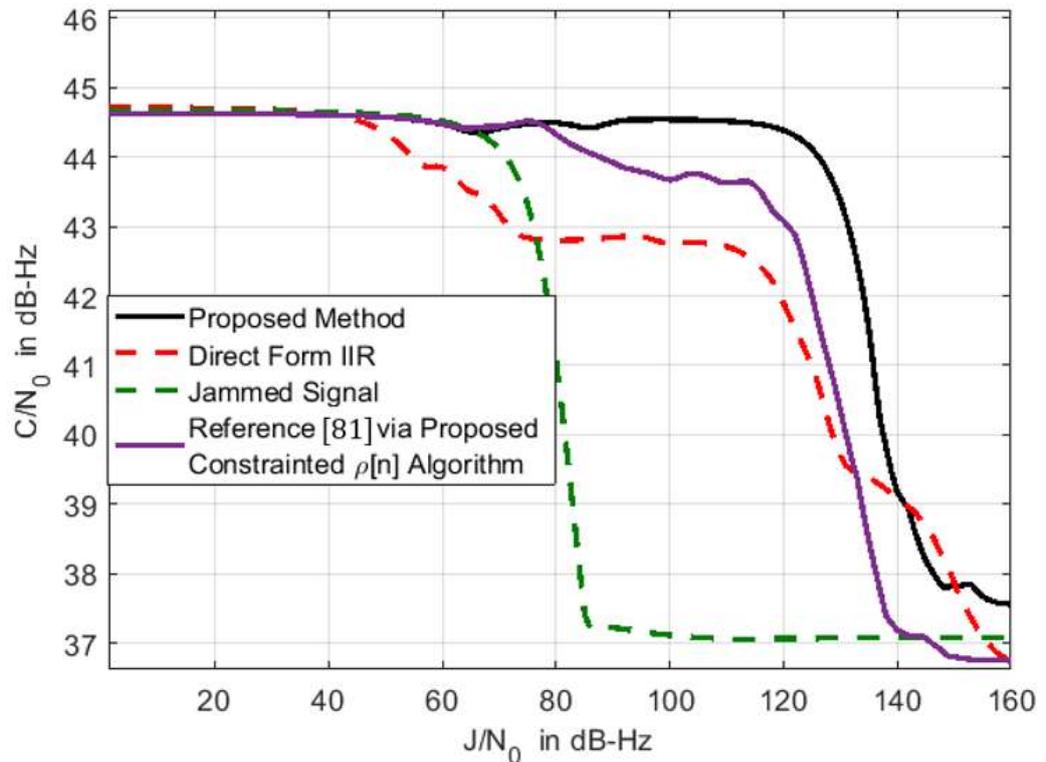


Figure 5.55 shows a comparison of the carrier-to-noise density C/N_0 at the output of the acquisition block for three different types of algorithms.

There is a significant difference between the performance of the direct form 2nd ANF and the proposed method. C/N_0 For the direct form 2nd ANF is lower. This is due to the introduction of the large transient at the start of each subsequent hop frequency because the parameter ρ is fixed at a higher value of 0.90. C/N_0 It starts to degrade rapidly when the value of J/N_0 is just getting above the 110dB mark for the red curve, which presents direct form 2nd ANF.

While if C/N_0 the curve for the reference [81] (blue) and proposed method (black) are compared to the output C/N_0 is same up to J/N_0 level of 78dB. But as levels of J/N_0 exceed

the 80dB mark, the blue curve sinks and lower C/N_0 is achieved. It is due to the fact that as the power of the interference increases, the convergence of parameter ρ to the upper limit is slowed down. Hence producing a wider notch and causing degradation of C/N_0 at the output acquisition block. At one point in the simulation (130dB mark), C/N_0 values for the blue curve get even less than the red curve. Mainly cause of the adaptation of parameter ρ is not consistent or ideal (narrower) for extremely high-power interferences. Simultaneous adaptation of both parameters β and ρ in [81] can be applied where single-tone CWI is available. However, multi-tone hop frequency CWI mitigates interference, but the mitigation dynamic range is less than the proposed method.

Both curves, the black and the blue curve start to rapidly plunge around J/N_0 level of 120dB. Output C/N_0 for both filters start to decrease exponentially. Now keep in mind that JNR is different to J/N_0 parameter, in J/N_0 jamming power is set against the value N_0 with is specified in chapter 3 equation (3.11). To convert from J/N_0 to JNR, simply subtract 60dB from J/N_0 . Now 120dB mark means JNR of 60dB, which means the jamming signal is 1000 times stronger than the thermal noise level in GPS L1. While the actual GPS signal itself is 20dB below the thermal noise level. Hence 120dB mark is a very high-power interference signal, and this point can be considered as C/N_0 degradation point for any second-order ANF. Even direct form 2nd order ANF (red curve) follows a similar trend but slightly earlier than the 120dB mark.

5.8 Fixed-Point Simulation of the Proposed Adaptive Notch Filter

Any real-world number can be represented as a binary digit inside a digital processor. All processors fall into two categories in how they represent numerical values: floating-point and fixed-point. Their representation primarily defers in a way in which they handle the

radix-point or the decimal point. In floating point, radix point is floating and can be moved; hence using the same word size, we can increase both precision and range. Fixed-point, in turn, is characterized by a single word with a fixed radix point, so there is always a trade-off between precision and range for a fixed size. However, the differences between these two processors are significant enough that they require very different approaches for algorithm implementation.

The fixed-point implementation in DSP has fixed word lengths. They are less expensive than their counterparts. They use less power which makes them very ideal for battery-powered applications. They have higher clock speeds. On the other hand, for FPGAs, word length is not restricted by target hardware. The designer can select the optimal word length, but every increase in the wordlength impacts the silicon area and power usage.

Developing an understanding of which applications are appropriate for which processors are worthwhile. The large dynamic range available in floating point design makes them more flexible. They tend to be more expensive because they implement more functionality occupying larger silicon areas and wider buses. Floating point processors tend to be more high-level language friendly, thus can be easier to develop code for; thus, ease of development and schedule advantage can be traded off against higher cost and hardware complexity. Typically, lower cost and higher speed of fixed-point implementation are traded off against added design efforts for algorithm implementation.

Digital designers need to keep track of wordlengths and scaling. They need to handle rounding methods and saturation explicitly. Especially an adaptive algorithm is prone to quantization of the data path and coefficients of the filter because an adaptive algorithm precisely calculates or updates the values of adaptive parameters for every input sample. In this case, we are adapting two coefficients, namely, $\beta[n]$, the notch centre frequency and

$\rho[n]$, the pole contraction parameter. There are many different types of fixed-point arithmetic implementation, but commonly used are sign/magnitude, one's complement and two's complement. Figure 5.56 illustrates fixed-point binary representation; the most Significant Bit (MSB) is the sign bit. If the sign bit is zero, the number is assumed to be positive; if the sign bit is one, then the number is assumed to be negative.

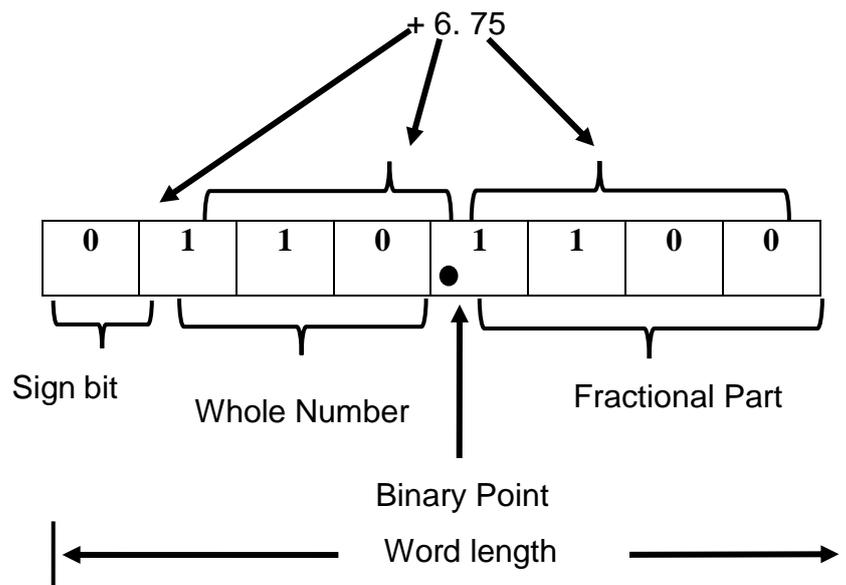


Figure 5.56 Fixed-Pointed Binary Representation

5.8.1 Floating-Point and Fixed-Point Comparison

The received GPS is down-converted to an intermediate frequency (IF), and this signal is sampled at the rate of sampling frequency F_s and discretized the signal in the time domain. The GPS signal is quantized by a B-bit quantizer where the quantizer is defined as in 5.45

$$Q_B^{A_g}[x] = -(2^B - 1) + 2 \sum_{i=-L}^L u(A_g x - i) \quad (5.45)$$

Here A_g is the gain applied to the incoming signal before the quantization process, $u(t)$ is the unit step function, and $L = 2^{(B-1)} - 1$. In the previous section, we have already developed the mathematical model of the GPS receiver, from the RF front end to the out of

the acquisition module. Quantization is a non-linear process which means noise is added to the system. The quantization of the numbers is carried out using rounding or truncation. Various types of rounding techniques are reported in the literature, such as round to zero, convergent round, and round towards minus infinity (truncation). The quantization error and rounding types are beyond this research's scope. In this section, the proposed adaptive filter's coefficients are quantized, and the amplitude of the incoming GPS L1 signal is discretized using 4-8 bits. The quantization of the coefficient of an adaptive filter is a very sensitive process due to the sensitivity of the adaptation algorithm. A minor change can make ANF unstable, leading to divergence instead of converging to an optimal solution or target frequency.

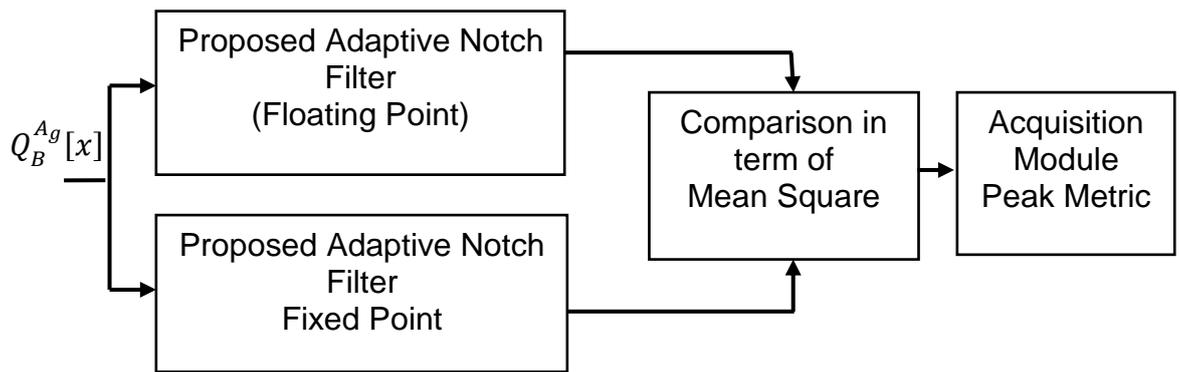


Figure 5.57 Performance comparison of floating-point and fixed-point

Figure 5.57 shows the methodology of comparing a floating-point and a fixed-point proposed adaptive notch filter. Mean Square Error (MSE) is used as a performance metric to assess and optimized the binary representation of the incoming signal and adaptive filter coefficients $\beta[n]$ and $\rho[n]$. Below are three mathematical equations that represent the mean square error between the floating-point values and fixed-point values.

$$MSE \text{ at output} = \frac{1}{N} \sum_{n=1}^N (y_L[n] - y_{qL}[n])^2 \quad (5.46)$$

$$MSE (\beta[n]) = \frac{1}{N} \sum_{n=1}^N (\beta[n] - \beta_q[n])^2 \quad (5.47)$$

$$MSE(\rho[n]) = \frac{1}{N} \sum_{n=1}^N (\rho[n] - \rho_q[n])^2 \quad (5.48)$$

In equations 5.47 and 5.48, the variable $\beta_q[n]$, and $\rho_q[n]$ are the quantized version of the notch frequency and pole contraction parameters, respectively. $y_{qL}[n]$ in equation 5.45 is the output of the filter whose coefficient is quantized to $\beta_q[n]$, and $\rho_q[n]$. $y_L[n]$ is the output of the filter in floating point simulation environment. Mean square error provides insight into how much a floating-point system matches a fixed-point system. Ideally, the error should be zero for the perfect match, but it is not possible in the real world. Therefore, in this case, look at the MSE very close to zero and assume that the signal is an almost perfect match. Furthermore, due to the autocorrelation property of C/A code or PRN code.

5.8.2 Quantization of the Proposed Filter's Coefficients.

It is a lattice-based adaptive notch filter and range of the magnitude of coefficient $\beta[n]$ is between -1 and 1 , and the range is between 0 and 1 for $\rho[n]$. As stated in the previous chapter, to maintain stability, the pole contraction parameter $\rho[n]$ is constrained between 0.70 and 0.95 . If it converges to extremely close to 1 , let's say to 0.9999 , the filter can lead to divergence. Similarly, for notch frequency parameter $\beta[n]$, if the quantization is applied on both coefficients, there should be enough room for the $\rho[n]$ should not be quantized to 1 by default due to the quantization process itself.

The adaptation of both parameters $\beta[n]$ and $\rho[n]$ can be viewed as a quantization process on the micro-level because ANF varies the notch frequency parameter $\beta[n]$ by a very slight amount up or down to find the target frequency. If this change is very small compared to the quantization level, then the algorithm is unable to adapt to the correct target frequency.

Hence, more quantization levels must be considered so that the changes in the adaptation of both $\beta[n]$ and $\rho[n]$ are less visible without distorting the functionality of the adaptive notch filter.

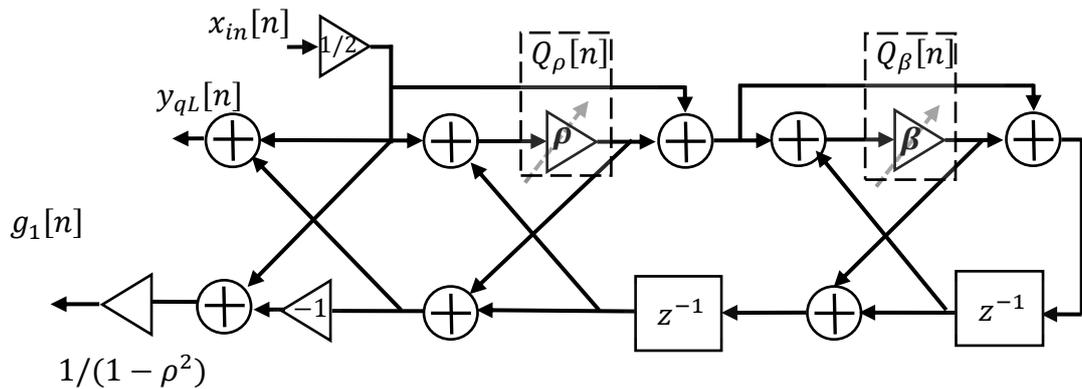


Figure 5.58 Illustration of quantization of both coefficients in the proposed filter

The floating-point implementation of the proposed filter leads to the perfect excision of various types of interference signals in GPS L1, as demonstrated throughout this research work. Therefore, the accuracy of ANF depends on the precise calculation of the adaptive coefficients of the filter. The quantization of the coefficients $\beta[n]$ and $\rho[n]$ leads to non-ideal characteristics in a digital filter, and it deteriorates the adaptability nature of the ANF too. The proposed ANF is simulated both in a floating-point and fixed-point simulation environment to analyse the effect of the quantisation of coefficients. In the quantized version, wordlength varies from 7 bits (1 sign, 1 integer bit, and 5 fractional bit) to 23 bits (1 sign, 1 integer bit and 21 fractional bits). GPS L1 signal is added to the model as was done previously, and the GPS L1 passes through both floating-point and fixed-point models. The MSE between the output of floating-point and fixed-point is measured.

Furthermore, the MSE between estimated coefficients in floating-point and fixed-point is also measured. Figure 5.59 shows the measured error at the output of ANF. The MSE error reduces as the number of fractional bits increases.

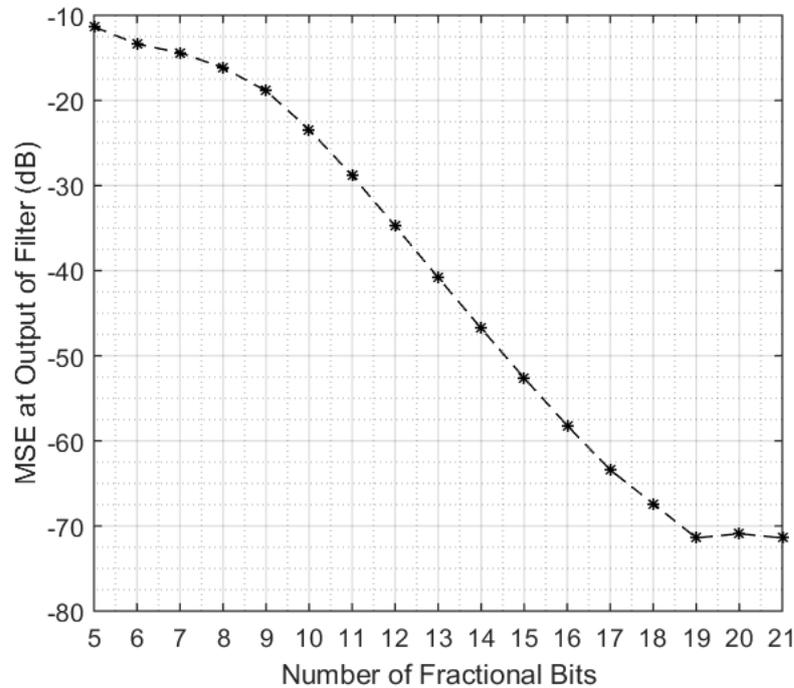


Figure 5.59 MSE at the output of the filter with different numbers of fractional bits

The performance of the fixed-point ANF filter is very low for the fractional bits between 5 to 9. As the fractional bits increase from 10 onwards, there is a fixed amount of improvement per bit, and it has a constant slope till the 19th bit. From there onwards, the MSE levels stay the same, just below -70dB. Due to the fact that the C/A code itself is a pseudo-random noise-like signal with random noise added, simulating the GPS L1 signal every time for a different number of fractional bits produces a different set of random noise, so it is impossible to get MSE of zero. The next two figures demonstrate MSE error between successive unquantized and quantized filter coefficients. The wordlength is varied from 7-bits to 23-bits. The proposed algorithm updates the filter's coefficients after every iteration. Therefore at every iteration, both coefficients are quantized to the required number of fractional bits. Figure 5.61 shows the MSE for quantization of $\beta[n]$. It is a downhill slope

with a constant gradient, which is understandable because once $\beta[n]$ locks the target frequency, there are random changes in its successive values. So fixed-point estimation of $\beta[n]$ gets more accurate as number of fractional bits increases.

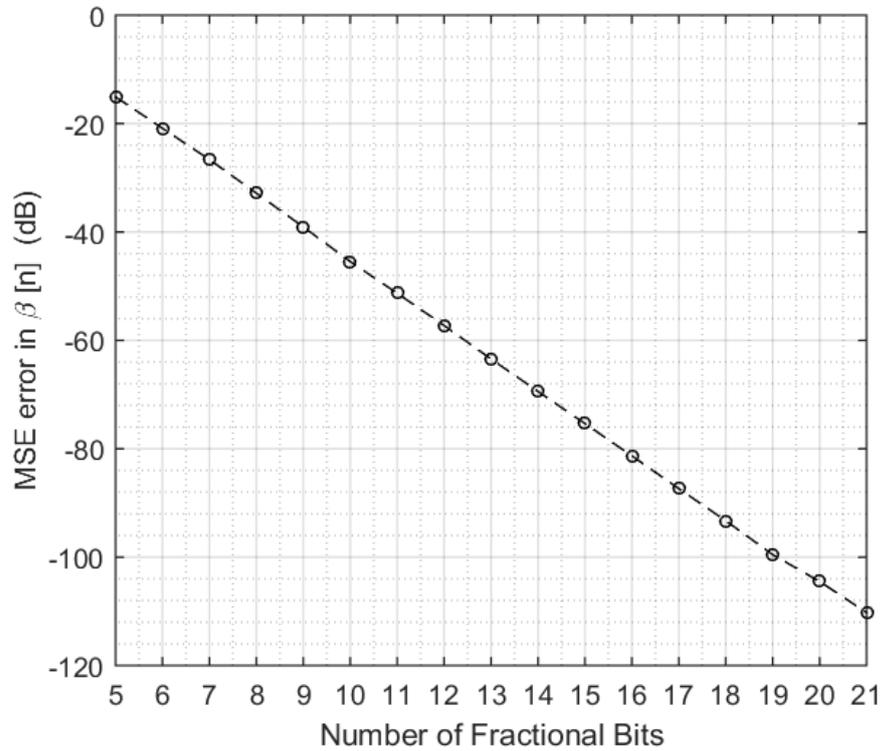


Figure 5.60 MSE of quantization of the coefficient $\beta[n]$ of the ANF filter with different numbers of fractional bits

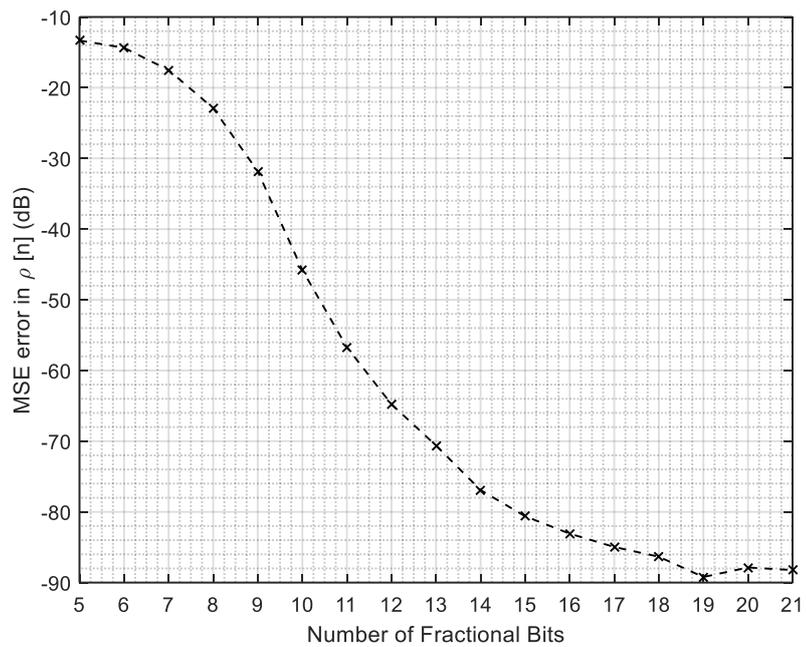


Figure 5.61 MSE of quantization of coefficient $\rho[n]$ of the ANF filter with different numbers of fractional bits

Figure 5.61 demonstrates how MSE error changes when $\rho[n]$ is quantized to a different number of fractional bits. The coefficient $\rho[n]$ adapts between 0.70 and 0.95 and is constrained between these two values. MSE error for $\rho[n]$ decreases as the number of fractional bits increases. The next step is finding a minimum number of fractional bits that the acquisition module can acquire even if the MSE error is high. During simulation, it was observed that with 13 fractional bits, the acquisition module was able to acquire the signal.

5.8.3 Implementation, Testing and Transferability.

Implementation: Adaptive digital signal processing can be synthesized and implemented in hardware. Different platforms are available such as DSP processors, Application Specific Integrated Circuit (ASIC), and Field Programmable Gate Array (FPGA). It is typical for DSP processors to be programmed in C, along with assembly code if needed for performance. It is primarily limited by the clock rate and the number of useful operations per clock that determine performance. ASIC-based special-purpose hardware cannot be altered to suit slightly different needs. As a result, new hardware is required for each new algorithm in such an approach.

ANF have traditionally been built using a digital DSP Processor. When performance is the main requirement, ASICs are another option. It is important to note, however, that both implementations require a trade-off between flexibility and performance. FPGA have recently become very attractive for the implementation of adaptive filters. An FPGA can be reprogrammed to deliver excellent flexibility and performance while enabling parallel computations at short processing times. A modern FPGA also incorporates an embedded microprocessor. Combining hardware and embedded software on a single chip improves the performance of fast filter structures with arithmetic-intensive adaptive algorithms.

Implementing the proposed method on FPGA via the Xilinx system generator for adaptive signal processing requires a definite design methodology, as shown in figure 5.62. By using the System Generator, designers can minimize the time spent describing and simulating circuits. Due to this, the design is flexible, and parameters can be adjusted quickly to see their effect on the architecture and performance. The Xilinx block sets are an add-on toolbox in Simulink to facilitate the development of FPG Applications. Using Xilinx blocks, you can implement designs with pre-existing IPs whose parameters can be altered. Using Simulink simulation or hardware co-simulation, the block-level design is validated once the block-level design is complete. After synthesis or implementation, SysGen can generate the device resource utilization report and the timing report.

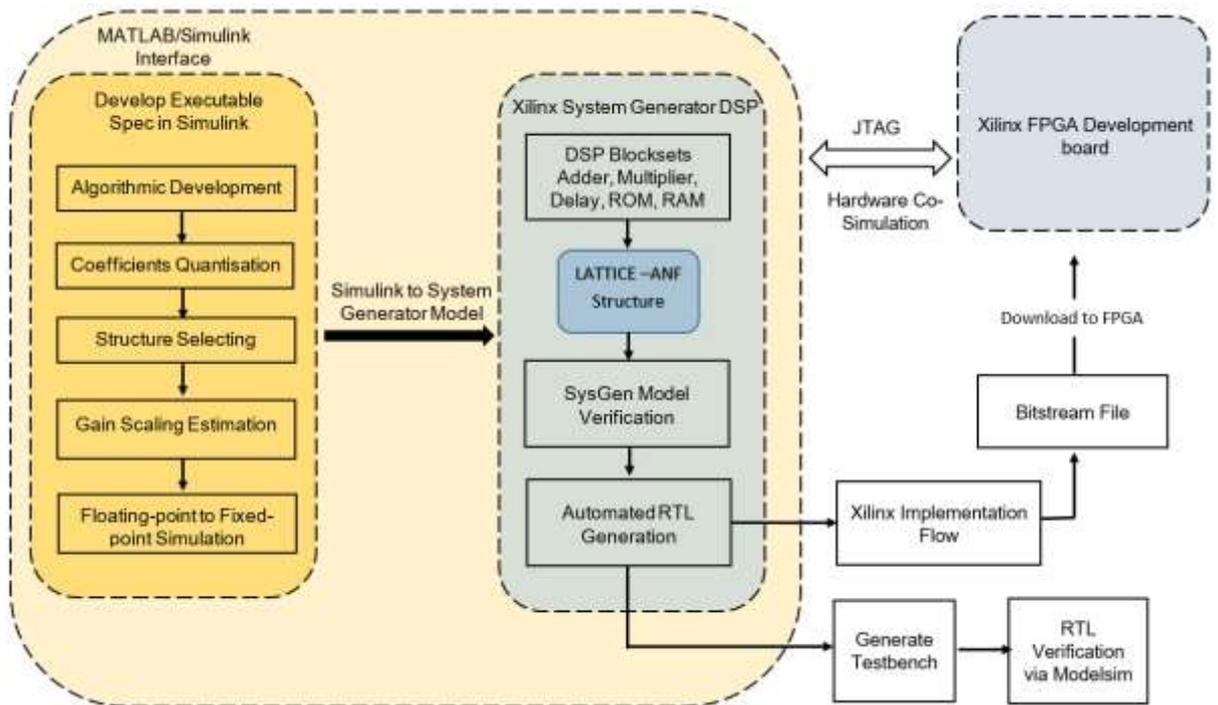


Figure 5.62 Workflow diagram for Practical Implementation via MATLAB/Xilinx platform

Today's FPGA development boards are processing powerhouse for high-end applications, and Xilinx provides various development boards to meet different designer's requirements.

The previous section discussed the effects of finite wordlengths on the proposed adaptive algorithm. ANF performance could be adversely affected by quantization errors, such as acquisition blocks' inability to acquire signals when coefficients were represented with fewer bits. When implementing an adaptive filter, the effects of the quantization errors become even more significant. Due to the fact that updating coefficient values are based on multiplication operation. The critical steps are the two multiplications to calculate the variation of each coefficient ($\beta[n]$ & $\rho[n]$). Both coefficients of the proposed ANF filter change at each iteration, depending on the values of the filter output signal $y_L[n]$, gradient signals ($g_\beta[n]$ & $g_\rho[n]$) and step-sizes $u_\beta[n]$ and $u_\rho[n]$. Whereas signals $y_L[n]$, $g_\beta[n]$ and $g_\rho[n]$ are the signal from the filter itself. Both step-size $u_\beta[n]$ and $u_\rho[n]$ are very small numerical values, thus where small operands can lead to bigger quantization errors than larger operands. Based on the analysis of the proposed algorithm, updating the coefficients is the most problematic step since it has two multiplications, and it is crucial to the algorithm's convergence and stability. Three solutions are considered to overcome these design constraints. (1) Perform quantization after the accumulation with extended resolution: A significant reduction in quantization error can be achieved in the filtering process. (2) Use the power of two factors for both step-size $u_\beta[n]$ and $u_\rho[n]$: By doing so, the first multiplication need not be performed because it can be replaced by an accumulation with a proper left shift operator. (3) Use scaled coefficient: to ensure filter output $y_L[n]$ and filter intermediate signal $g_\beta[n]$ and $g_\rho[n]$ are greater than one.

In addition to this, there are a few other limitations. Firstly, the proposed method is limited to a specific type of interference signal. ANF is ineffective at tracking and mitigating linear chirp interference if the instantaneous frequency changes faster than its convergence rate.

Testing Set-Up: This section briefly describes the proposed method's evaluation, validation, and performance through an example experiment setup. An in-field test can determine how interference signals affect GNSS receivers. There are some limitations to in-field testing:

- Field test equipment and resource transportation costs.
- A lack of control over the testing conditions and interference characteristics.
- Satellites should be tested only under the existing and detected constellations, and no other regional constellations or satellite configurations should be tested.
- There are a variety of interferences that can lead to miscalculations and the inability to generate them all in real-life situations.

The experiment can be set up in a controlled environment, such as inside an anechoic chamber, to overcome these challenges. It provides a completely isolated environment for testing jammers/interference with no interference from any source other than the transmitted interference. High-ended industrial GNSS simulators can generate GPS L1 signal with a realistic test environment and manually configure GPS L1 signal power level, IF frequency, code-phase delay, Doppler frequency and different GNSS signals.

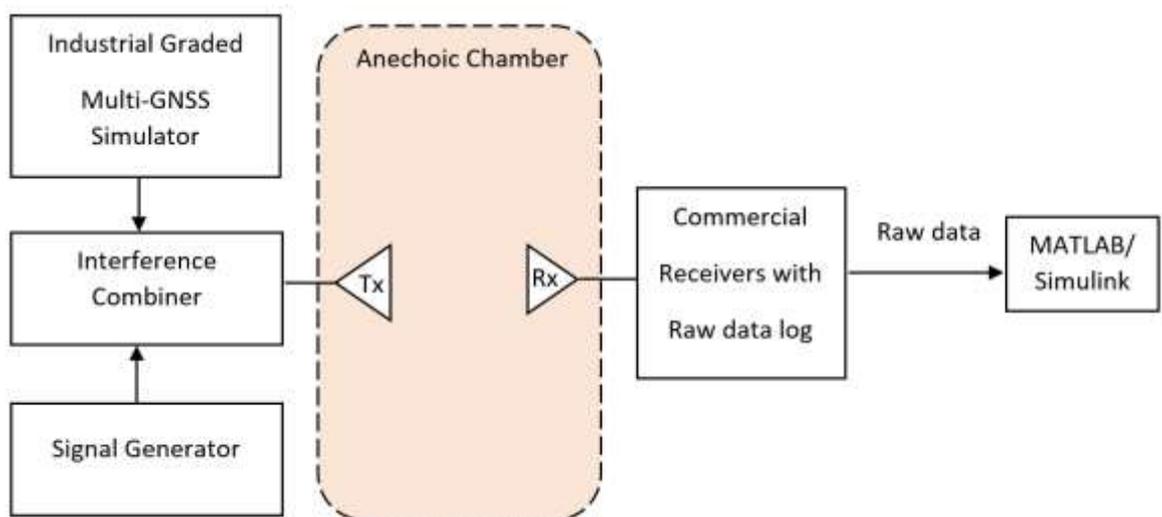


Figure 5.63 Example experiment setup

Similarly, a signal generator gives great flexibility to generate different types of interference, such as CWI, linear chirp, and hop frequency. Signals from the Multi-GNSS simulator and signal generator are combined via an RF combiner and transmitted via an antenna. A high-sensitivity commercial receiver receives the corrupted signal with interference, which can store raw data. Stored raw data post-processing can be carried out in MATLAB and evaluate the performance of the proposed lattice-based ANF against different types of interference based on a fixed-point system-level model.

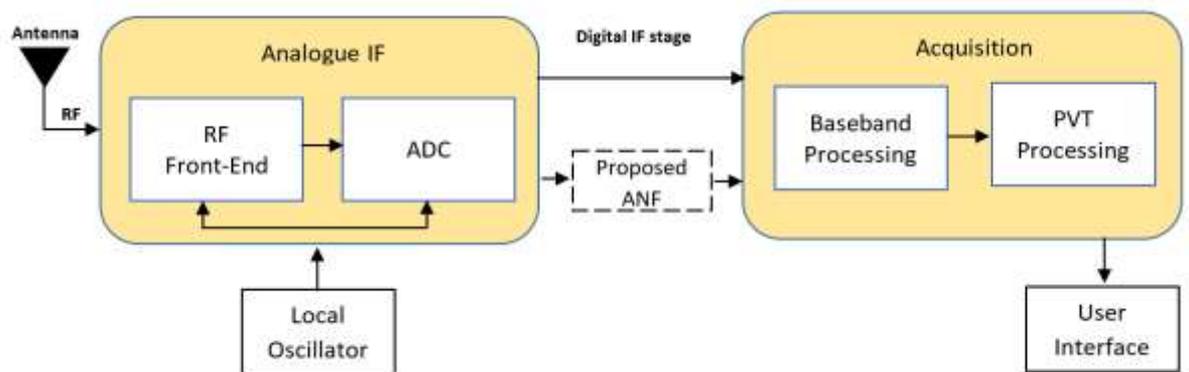


Figure 5.64 A typical GNSS receiver architecture

Transferability: ANF is a very versatile DSP processing block with a wide range of applications, from high-speed applications to superior noise-cancellation technology. Recently advancements in interoperability of open service for true multi-GNSS have created a niche for unified GNSS receivers with multi-frequency dimensions. The navigation signals transmitted by all of these systems are spread across two common frequency ranges (in between 1.1-2.5 GHz). Lower L band with L5, E5, B2 and L3 and upper L band with signal L1, B1 and E1. Future systems will achieve greater positioning accuracy and increase availability by adopting multi-frequency, multi-constellation capabilities.

A typical GNSS receiver architecture is shown in Figure 5.63, and this functional block diagram remains unchanged with the addition of multiple constellation capabilities. But it

does affect some components such as the antenna, RF front-end and baseband processing. For each frequency (Upper L and Lower L band), functional blocks are (approximately) replicated.

Table 5.8 Navigation frequencies processed by typical multi-GNSS receivers

System	Frequency Band	Carrier Frequency (MHz)	Code Modulation	Chip- Rate (MHz)	Length of PRN code (ms)	BW (MHz)	IF (MHz)
GPS	L1	1575.42	BPSK	1.023	1023 (1ms)	2 (OS) 20(RS)	3
	L2	1227.6	BPSK	0.5115	10230 (4ms)	2 (OS) 20(RS)	3
	L5	1176.45	QPSK	10.23	10230 (1ms)	20	20
Galileo	E1	1575.42	CBOC	1.023	4092 (4ms)	4 (OS)	4.5
	E5a	1176.45	AltBOC	10.23	10230 (1ms)	20 (OS)	20
	E5b	1207.14	AltBOC	10.23	10230 (1ms)	20	20
Beidou	B1C	1575.42	BOC	1.023	10230 (10ms)	20 (OS)	20
	B1	1561.098	BPSK	1.023	1023 (1ms)	4 (OS)	4.5
	B2	1207.14	BPSK	2.046	2046 (1ms)	4 (OS)	4.5

Table 5.8 shows the spectrum of GNSS navigation signals processed by a typical multi-GNSS receiver [16]. Yellow colour coded represents Open Service (OS) signals whose bandwidth is from 2-4 MHz except for Beidou III B1C with 20 MHz. In contrast, Restricted Service (RS) on the same frequencies have a signal bandwidth of 20MHz. The red colour code represents the same carrier frequency for OS signal by GPS, Galileo and Beidou to

increase the harmonious development of true multi-GNSS receivers. Finally, green colour coded in table 5.8 represents typical digital IF frequencies for each navigation signal [16], ranging from 2 to 20 MHz. At the RF front-end, the L-band frequency is translated to the IF centre frequency, and then this analogue IF signal is converted to a digital IF signal by ADC; from there onwards, all GNSS processes are digital. At the digital IF stage (as shown in Figure 5.63), all of the GNSS signals in the view are still buried in thermal noise along with CWI or chirp interference if it is present. At this point, the proposed method can be incorporated within the hardware with strong firmware to mitigate interference from the GNSS signal, as shown in Figure 5.63. As the navigation signal is buried in noise, ANF sees different PRN codes from different constellations as random noise and tries to estimate interference signal frequency. In other words, the proposed method tries to extract CWI, chirp type and hop-frequency interference signals embedded in noise.

There are two key features of the proposed method: its superior convergence speed and the ability to set thresholds using internal filter signals. Different GNSS signals received on the surface of the earth have different power levels, such as E1 is -157dBW , E5 is -155dBW , and B1 is -163dBW . Therefore, the proposed method must be modified so that it calculates thresholds adaptively for various levels of signal power received.

5.9 Conclusions

A novel lattice-based adaptive IIR notch algorithm is presented, which can simultaneously adapt its notch bandwidth and centre frequency parameters. The proposed algorithms show favourable convergence properties for their parameters β and ρ . When compared with the relevant competitive designs, the proposed method shows improved tracking performance and higher C/N_0 at the output of the acquisition block.

Simulation results have demonstrated that the proposed algorithm is consistently stable and has also proven its superior performance under different conditions. These advantages are attained due to the constraint imposed on the pole of ANF. If the pole of the ANF moves towards the instability region (unity on unit circle or more), it reverts into the stable model set to prevent divergence.

This chapter provided a detailed analysis of various adaptive algorithms for ANF. Such as two algorithms for direct form structure IIR notch filter were implemented [48] and [68]. Also, one algorithm for the lattice-based structure [51] is also implemented. The existing algorithms were then compared with the proposed fully adaptive filter based on the lattice structure, which can simultaneously adapt its coefficients.

A complete study of 3D frequency response and simulation of the cost function of ANF were generated and plotted, which provided the groundwork to understand and develop proposed methods. It also explained how a direct form structure suffers from gain scaling in the passband region if ρ is adapted; gain scaling gets worse if the target frequency is close to DC or Nyquist.

This chapter proposed a fully adaptive notch filter with full gradient term to update parameter ρ , demonstrating excellent converging speed and better retrieval of the useful signal. Different resetting algorithms for parameter ρ are developed and then proposed a fully constraints algorithm for the adaptation of parameter ρ .

A full gradient term is derived from the filter output in section 5.4, which is then utilized to update the ρ in a fully adaptive notch filter. Also, the modified version of the proposed structure is presented with less computational complexity.

Furthermore, the resetting of the parameter ρ to 0.70 just before the new hop frequency is key to successfully adapting both parameters. Therefore, resetting should be applied carefully within the algorithm, and the selection of the constraints has to be such that it works universally regardless of the environment. Various resetting algorithms were designed, modelled, and tested and presented with simulation results with different resetting criteria. The parameter ρ has to be sharply reset at the beginning of the next hop frequency, hence developed a resetting mechanism within the adaptive algorithm to automatically set parameter ρ to a desired minimum value just before the next subsequent hop frequency tracking.

In section 5.6, two different structures of adaptive notch filter are compared in terms of tracking, convergence speed, and output C/N_0 specifically for GPS-based applications. A complete GPS transmitter, receiver front-end, and acquisition module are modelled in MATLAB and Simulink. Simulation results prove that the lattice structure with the proposed full gradient algorithm for adaptation of notch bandwidth parameter ρ shows superior performance in terms of convergence speed and level of C/N_0 at the output of the acquisition module.

At the end of this chapter proposed method was further compared with [81], as [81] also simultaneously adapted both its coefficient. The simulation results have illustrated that the proposed method has better convergence properties for the parameter ρ and improved C/N_0 was attained. The convergence of ρ and resetting of ρ on time (at the start of new hop frequency) are two key points for successfully mitigating hop frequency CWI.

A constraint adaptation of the notch bandwidth parameter is developed, modelled, and presented. It is shown in the simulation results that by simultaneously adapting both ρ and β , the performance of a lattice-based adaptive notch filter can be enhanced significantly.

Furthermore, the proposed algorithm's tracking ability and convergence speed are compared with the existing techniques to provide an in-depth comparative analysis between existing methods. Existing and proposed methods are tested for hop frequency type of interference.

The trade-off between the quality of interference excision and the computational complexity of the system needs to be addressed, and further work needs to simplify the proposed method and develop it into a computationally efficient system.

Chapter 6

EXPLOITING THE AUTO-CORRELATION FUNCTION AND THE SPECTRUM ENVELOPE OF THE GPS L1 SIGNAL

The circular convolution of any PRN code (C/A code) with itself forms the autocorrelation function. This correlation process is also referred to as matched filtering because a locally stored replica of that signal filters the incoming signal. The incoming GPS L1 signal is matched against different Doppler frequencies and code phase delay, forming the Cross-Ambiguity Function (CAF). GPS signals are designed to give a maximum correlation peak when codes are perfectly aligned and no visible peak (noise-like) when codes are not aligned. The height of the main peak is nominally 1023 samples, and the minimum level is as low as -1. The other side lobes can be as large as +64 or -65. Further details about PRN code, autocorrelation properties, and CAF is already discussed in Chapter 1, 2, and 3.

For a simplified analysis, the focus is on the Power Spectral Density (PSD) and Bandwidth (BW) of the GPS L1 signal. PRN code is a series of -1 and +1 and running at the chip rate $T_c = \frac{1}{F_c}$, where F_c is the chipping rate 1.023MHz. One period of C/A code can be written as [24]

$$x(t) = \sum_{n=0}^{N-1} x_n p\left(\frac{t-nT_c}{T_c}\right) \quad (6.1)$$

In (6.1), $p(t)$ represents chip waveform with unit width and unit height and is centred at the origin. Here in this equation, $p(t)$ is modified to have a pulse duration of T_C and delay of nT_C and x_n represent elements (-1 or 1) of the sequence to be transmitted. The code repeats itself after 1023 chips. Hence N is the number of chips in each code ($N=1023$) and T_C is the duration of a single chip.

Fourier Transform of time domain rectangular pulse $rect(t)$ is *sinc* function in the frequency domain as illustrated in the equation below.

$$p(t) = rect(t) = A \begin{cases} 1, & -\frac{T_C}{2} \leq t \leq \frac{T_C}{2} \\ 0, & otherwise \end{cases} \xrightarrow{FT} P(f) = AT_C \frac{\sin(\pi T_C f)}{\pi T_C f} \quad (6.2)$$

The discussions below give basic definitions of Power Spectral Density (PSD) and signal Bandwidth (BW).

Power Spectral Density: The power spectral density represents the signal power distribution over the frequency interval $(-\infty, +\infty)$ over both positive and negative frequencies. The power of the signal in the frequency band $(-W, W)$ is given by

$$P_{tot} = \frac{1}{T} \int_{-W}^W |P(f)|^2 df \quad (6.3)$$

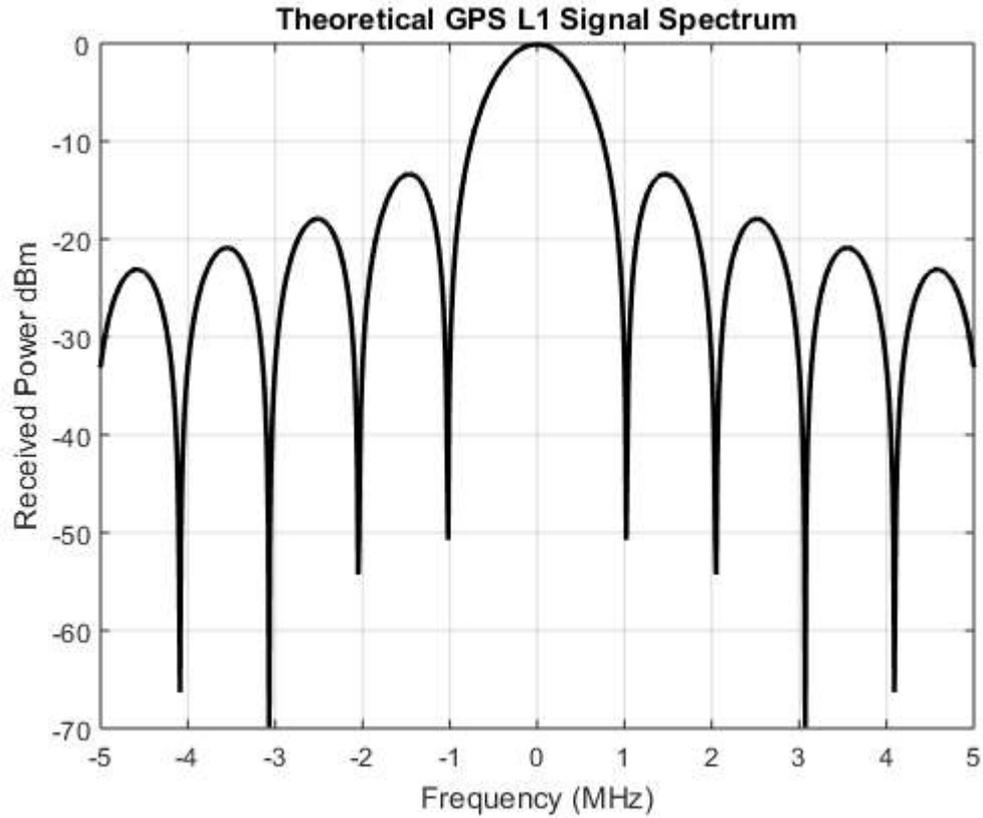


Figure 6.1 Illustration of the theoretical spectrum of GPS L1 C/A at baseband

Hence according to Parseval's relation, the total power of the signal is equal to

$$P_{tot} = \frac{1}{T} \int_0^T p(t)^2 dt \quad (6.4)$$

Bandwidth: The bandwidth (BW) of a signal is the width of the frequency band in which ~95% (usually between 99-90%) of its power is located, which can be formulated as follows

$$\int_{-BW}^{BW} A^2 T_C^2 \text{sinc}^2(\pi T_C f) df \approx 0.9 \int_{-\infty}^{+\infty} A^2 T_C^2 \text{sinc}^2(\pi T_C f) df \quad (6.5)$$

Building onto (6.3), the power spectral density of the C/A code can be approximated using the following equation [24].

$$S(f) = A^2 T_C^2 \text{sinc}^2(\pi T_C f) \quad (6.6)$$

Therefore, the envelope of the spectrum of the C/A code is a 'sinc-like' function, and the main lobe in Figure 6.1 of the spectrum occupies 90% of the signal power and contains 90%

of the signal information. The pre-correlation filter in the GPS RF front-end typically filters this main lobe. As the PSD of the C/A code follows the ‘sinc’ function, the side lobes on the left and right of the main lobe contain minimal signal power. Therefore, filtering outside lobes on both sides causes a small amount of information loss. Only the main lobe can be used to correlate the incoming filtered signals with the local replica of the C/A code for signal acquisition. C/A code is designed to yield the maximum correlation peak when codes are aligned due to this auto-correlation property which makes the satellite navigation signal acquirable even if the main lobe of the signal is filtered out.

After reviewing some basics, now the main goal here in this research is to establish up to what extent or how much bandwidth is required at the minimum to acquire a GPS L1 signal, i.e. how much power loss in the main lobe of C/A code can be compensated in order to acquire the signal fully.

If any type of interference is present in the main lobe of the GPS L1 signal, the GPS receiver cannot acquire and track the satellite. This thesis considers two main types of interference/jamming signals of great interest: single-tone and closely packed multi-tone with higher JNR levels. Two research question arises:

1. Determining the percentage of the main lobe that can be discarded/removed from the received C/A code could still preserve the information for acquisition.
2. Amount of degradation caused by the output of the acquisition module.

Answering these two questions is not that easy. Firstly, because these PRN codes are random, we must convert the received GPS signal from a deterministic signal into a random signal. Secondly, while computing PSD for C/A code, the analysis gets difficult because C/A code is short in length and not well modelled as a random code as it repeats itself 20 times

during a single data bit. While conducting the analysis, numerous digital signal processing techniques are designed, modelled, and simulated in MATLAB.

6.1 Acquisition of GPS L1 Signal When Certain Percentage of Main Lobe Discarded

The term ‘clean C/A code’ refers to when C/A code is correlated with itself only, without any noise being added to it. The test methodology is illustrated in the Figure below.

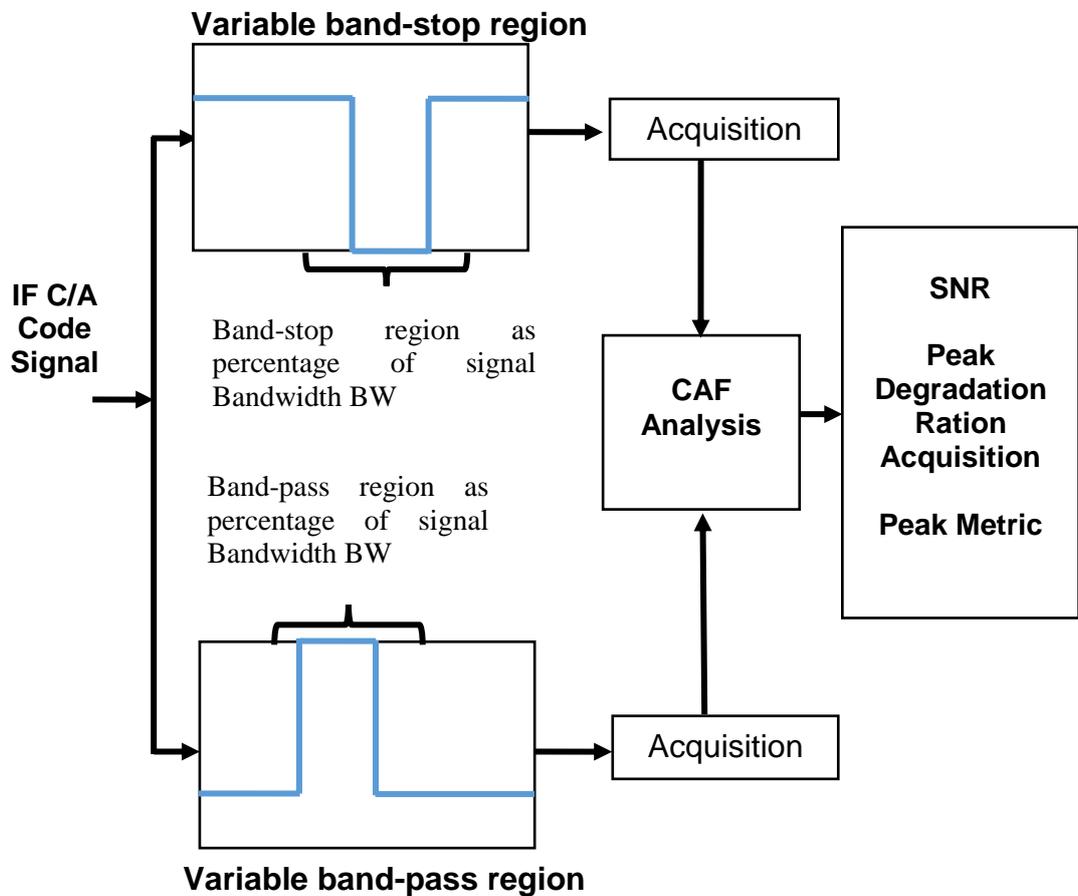


Figure 6.2 Illustration of the analysis methodology for the acquisition of GPS L1 with a certain percentage of GPS L1 main-lobe discarded

A straightforward and intuitive methodology is adopted to ensure the accurate validity of the results. In both cases, band-stop and band-pass, an ideal rectangular filter is employed to analyse optimum achievable performance through the acquisition module. In the simulation results, the bandwidth of region of interest is always represented as the percentage of signal

bandwidth which is 2.046MHz. For the band-stop filter, the band-stop region is set to 10%, 20%, 30% and so on until 100% of the signal bandwidth. After filtering C/A code through an ideal band stop filter of different bandwidths, it is passed on to the acquisition module, which gives the peak at desired Doppler bin and code phase delay. Here maximum correlation peak shortens in height as the stop bandwidth increases because a certain amount of power of the signal from the main lobe is being chopped off. Let's say the maximum attainable correlation peak is 1, without band-stop filtering, but as we invoke the stop-band filtering, the subsequent peaks reduce in height. A new metric is developed to measure the correlation peak's degradation.

$$Peak\ Degradation\ Matrix = \frac{P_{BW}(n)}{P_{max}} \quad (6.7)$$

Where P_{max} represents the maximum correlation peak without filtering, $P_{BW}(n)$ represents the maximum achievable peak, the bandwidth of the band-stop filter is BW, and n represents the percentage of the bandwidth.

As the process is random, to obtain reliable results, each simulation is run multiple times, and output results are averaged and then plotted. Table 6.1 shows specifications for the simulation set-up throughout this section.

6.2 CASE I: Filter ‘Clean C/A code’ through variable width ideal band-stop filter

Table 6.1 Simulation set parameters

Simulation Parameters	
Satellite ID	5
Fs	5 MHz
Intermediate Frequency (IF)	1.25MHz
Number of Samples	5000 samples
C/N0	48 dB
Received Signal Power	-160 dBW
Acquisition Time	1ms of Coherent Integration
Width of Main Lobe in term of number of samples	2046 samples
JNR level	[35dB]
Input SNR	Between -17 dB to -20 dB

The C/A code is sampled at 5 MHz and transmitted at an IF of 1.25 MHz without any noise being added to it. Figure 6.4 shows the PSD of the test signal sampled at 5MHz, and the main lobe is centred at IF frequency with a bandwidth of 2 MHz

Figure 6.3 (a) shows the peak degradation metric, which reduces as the percentage of the band-stop region increases. From the shape of the curve, it can be said that the peak is reducing in the manner of exponential decay function. A constant stop-band region width of different lengths corresponds to the removal of uneven signal power from the main lobe of the C/A code. The centre frequency of the band-stop region is 1.25 MHz. The simulation is run step by step, increasing each band-stop region by 1% at a time. The maximum correlation peak of the subsequent CAF function was observed to decay exponentially.

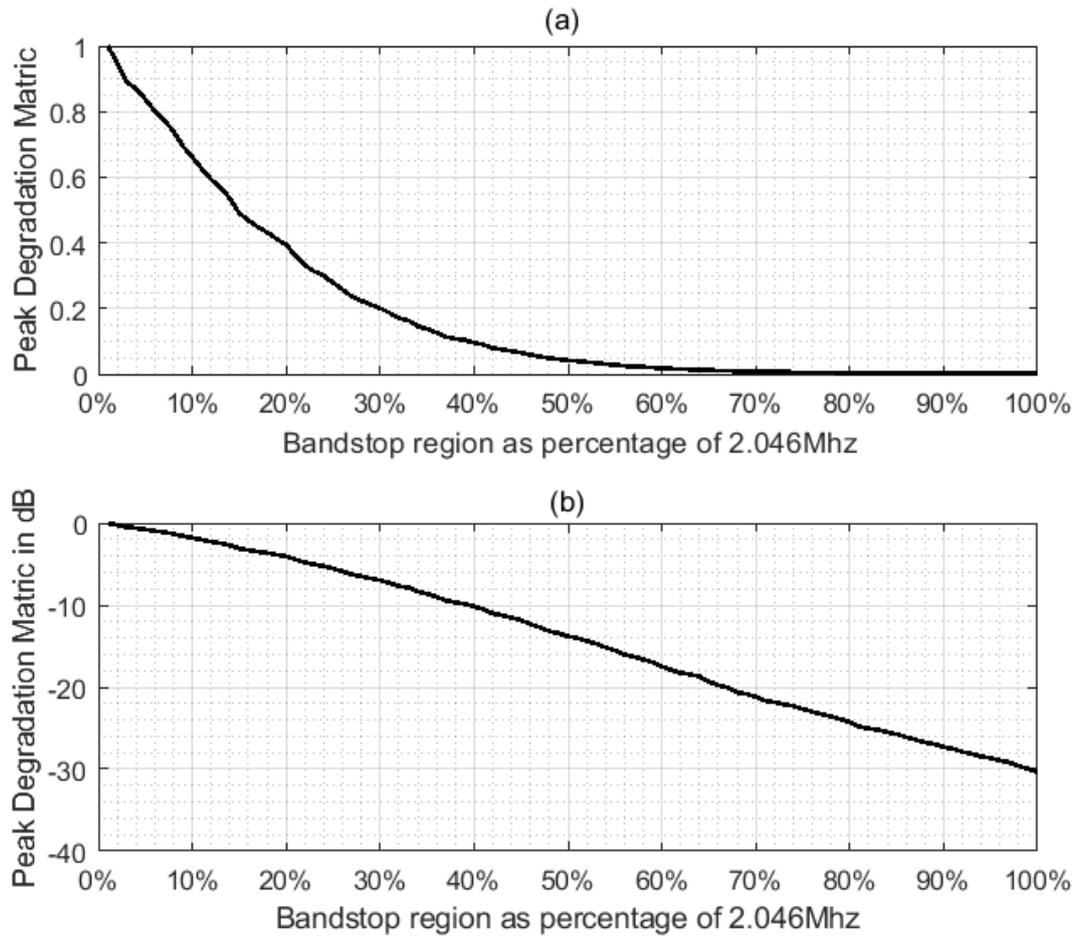


Figure 6.3 (a) Peak Degradation Metric (PDM) vs Band-stop region. (b) PDM in dB

The normalized maximum correlation peak through this simulation is unity. Further analysis plotted in Figure 6.3 (a) shows that when the stop-band region is 10% of the signal bandwidth, the peak degrades by 25% of the maximum correlation peak, while at 10% of the stop-band region, the correlation peak is 0.75 times of the maximum correlation peak. Similarly, for 20% and 30% of the stop-band region, the correlation peak is 0.4 and 0.2 times the maximum correlation peak. By the time the stop-band region reaches 50%, the corresponding correlation is approximately 0.04.

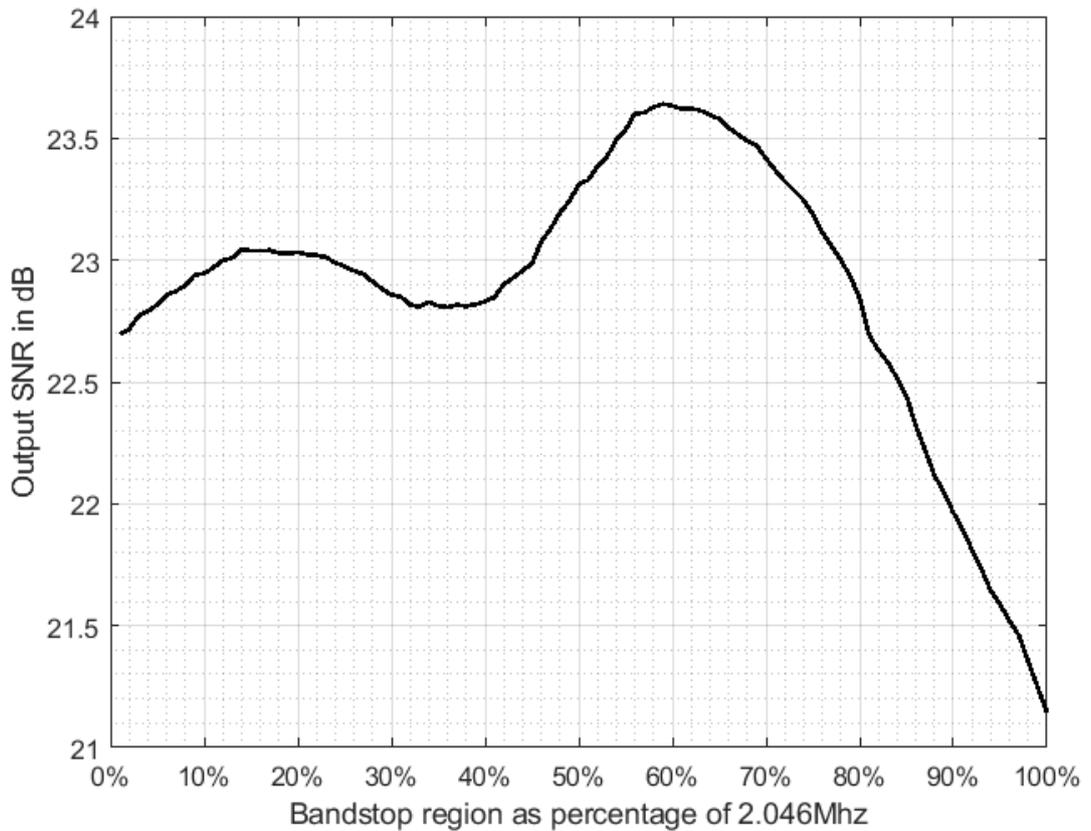


Figure 6.4 SNR at the output of the acquisition module after band-stop filtering with variable width

The maximum achievable output SNR for the ‘clean C/A code’ is above 22.5 dB, as shown in Figure 6.4, when the stop-band region is 0%. Two prominent bumps are visible as the stop-band percentage increases from 0% to 100%. Firstly in between 10% and 20%. Secondly, the larger and wider bumps are between 50% and 70%. These are the two regions where SNR is peaking. Interestingly even a certain percentage of the C/A code spectrum is removed. It can be explained by the fact that auto-correlation of C/A code, when codes are perfectly aligned, produces noise levels at -1, +1 and +64, -65, apart from the main peak of height 1023. Hence while discarding certain regions of the C/A code, the main or maximum peak height reduces along with the noise level. From the definition, SNR is the signal and noise power ratio. Therefore, the noise floor generated by the auto-correlation of two perfectly aligned codes diminishes rapidly in size as the band-stop region increases from 10% to 20% and from 50% to 70%. This is a clean C/A code, and even when the stop-band

region is 100% of the bandwidth of the main lobe, its side lobe in Figure 6.1 is still present for the correlation, and therefore, the signal still has high SNR above 21 dB as shown in Figure 6.4.

6.3 CASE II: Filtering ‘Clean C/A code’ through variable width ideal band-pass filter

In this set of simulations, the same parameters were used, given in Table 6.1. The only difference is that an ideal band-pass filter with a variable pass-band region is used. Performance analysis is carried out in terms of peak degradation metric and SNR at the output of the acquisition module.

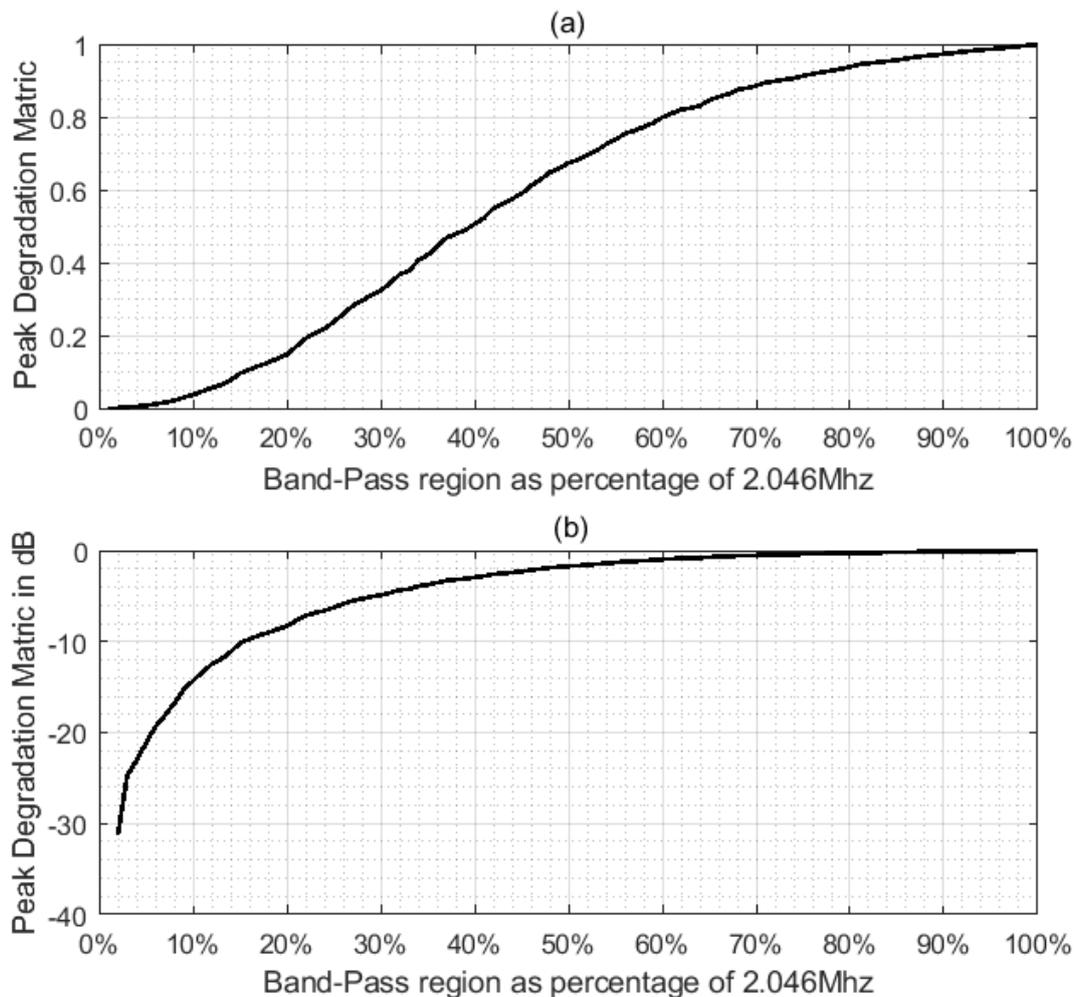


Figure 6.5 (a) Peak Degradation Metric (PDM) vs Band-Pass region (b) PDM in dB

These sets of results and analyses provide information about how much of the main lobe can be filtered using a band-pass filter at the RF front-end. The signals have most of their energy located in a certain frequency lobe. For BPSK modulation, energy is concentrated at around IF frequency, within $\pm F_c$ interval, where F_c is the chip rate. Figure 6.5 (a) shows that the peak at the pass-band region decreases from 100% to 0%. This graph follows the shape of an arctangent function, but the region between 30% to 50% follows an almost linear increase in the PDM value as pass-band width increases. Furthermore, when the pass-band region is 70% of the signal bandwidth, PDM is 0.88 times of maximum correlation peak, corresponding to the loss of approximately 0.53 dB, as shown in Figure 6.6(b). When the pass-band is 40% of the main lobe, the value of PDM is precisely half, corresponding to a -3dB loss in the height of the maximum correlation peak.

Figure 6.6 is more self-explanatory; as the width of the pass-band region increases, the corresponding SNR at the acquisition output also increases. Closely observing this curve, it can be seen that the increase in SNR from 60% to 100% is not as rapid as it increases from 10% to 40%. From 60% to 100%, the increase is very marginal, just by 0.56dB. Hence for an ideal case, for a 'clean C/A code', if the pass-band region is 60% of the main lobe, the reduction in output SNR is lower, which is a very promising result to justify the use of IIR filter bank-based processing of GPS L1 signal.

Both for case I and case II, the centre frequency of the pass-band region (for band-pass filter) and stopband-region (for band-stop filter) is at IF, and the widths of both these regions increase outwards from the centre location of IF.

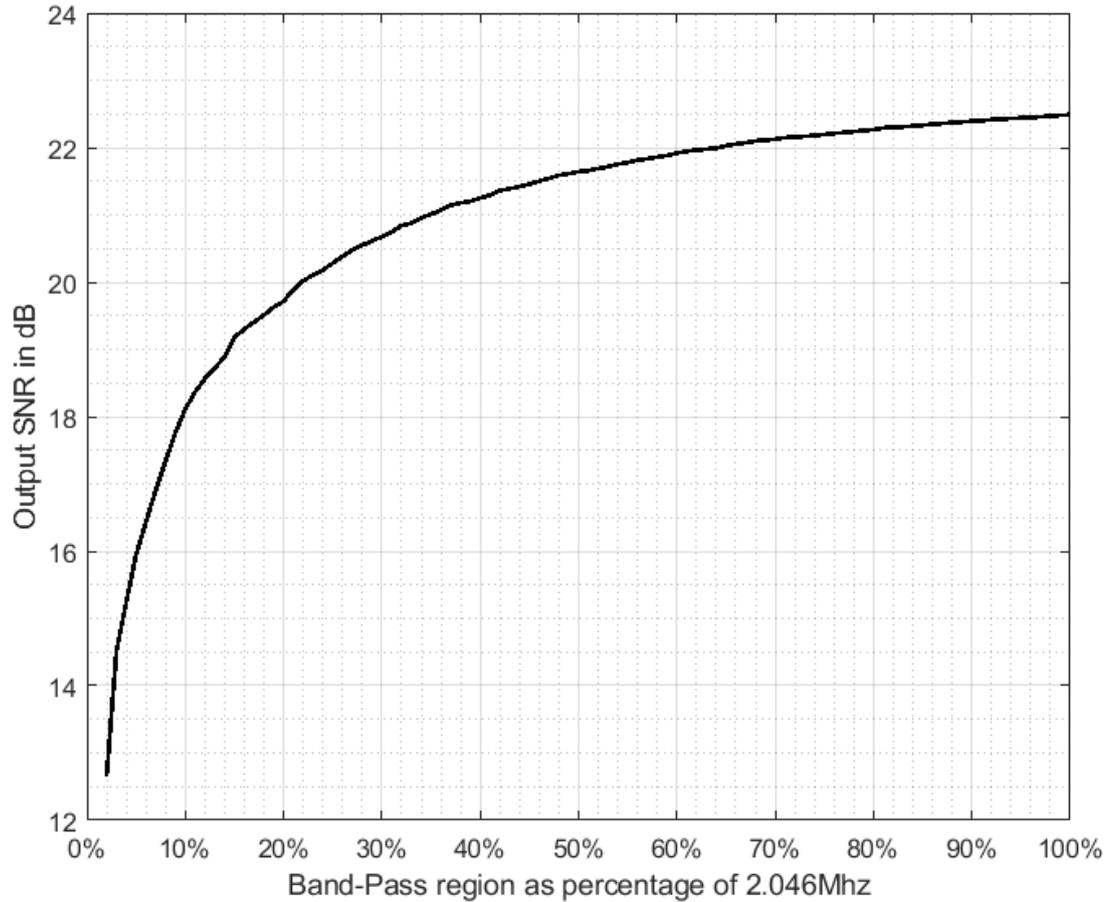


Figure 6.6 SNR at the output of the acquisition module after band-pass filtering with variable width

6.4 CASE III: Filtering ‘C/A code +Noise’ through variable width ideal band-stop filter

The previous two cases assumed an ideal C/A code without any noise. In Case III, higher noise levels and interference are considered. The top graph in Figure 6.7 shows the PSD of the C/A code and noise baseband. The bottom graphs show the GPS L1 signal with noise added to it, which sits around -204dB/Hz, with an input SNR of -19dB, meaning the signal of interest is -16db below the noise floor.

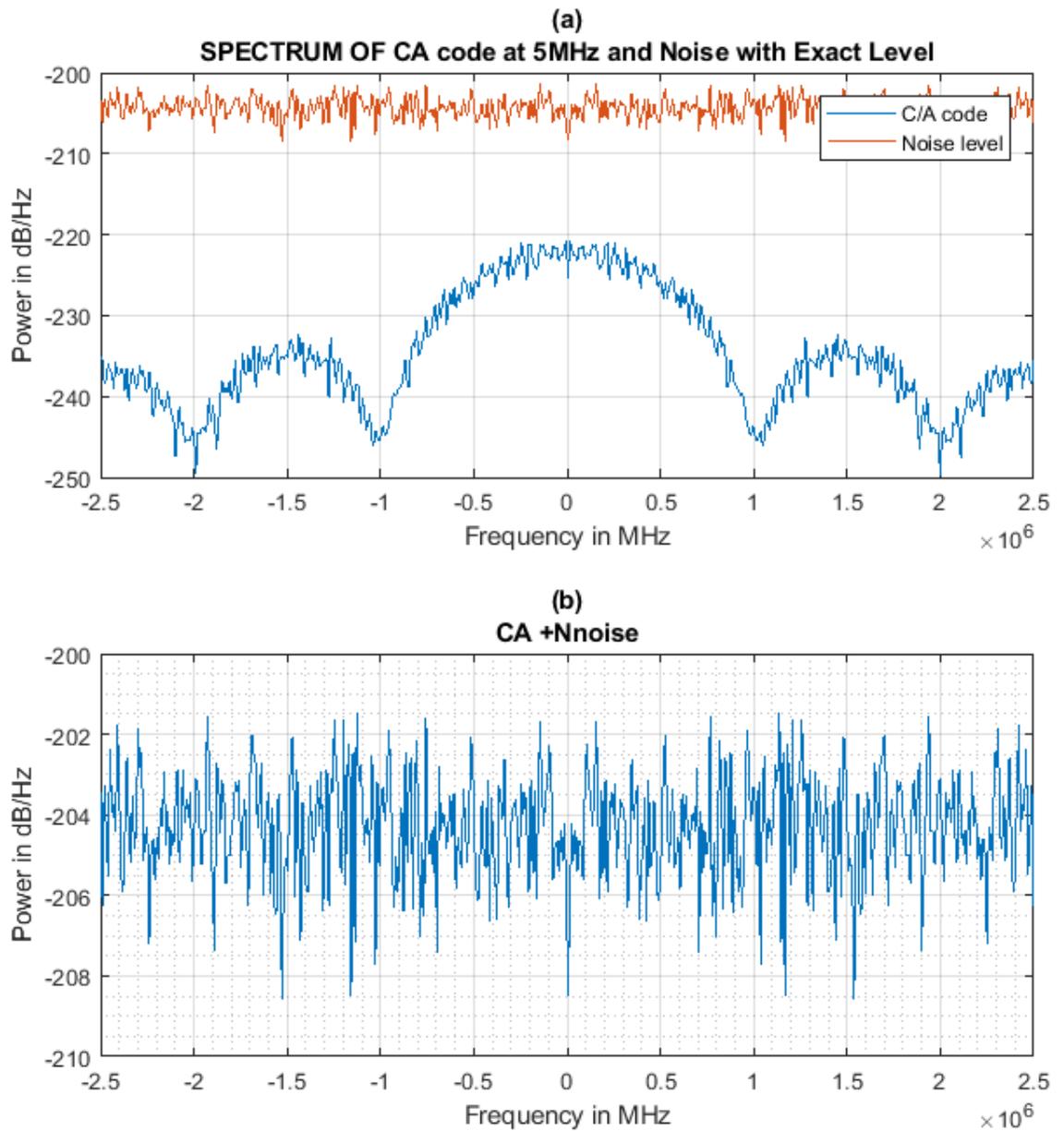


Figure 6.7 (a) PSD of C/A code and noise. (b) Combine PSD of C/A code added with noise exact noise level

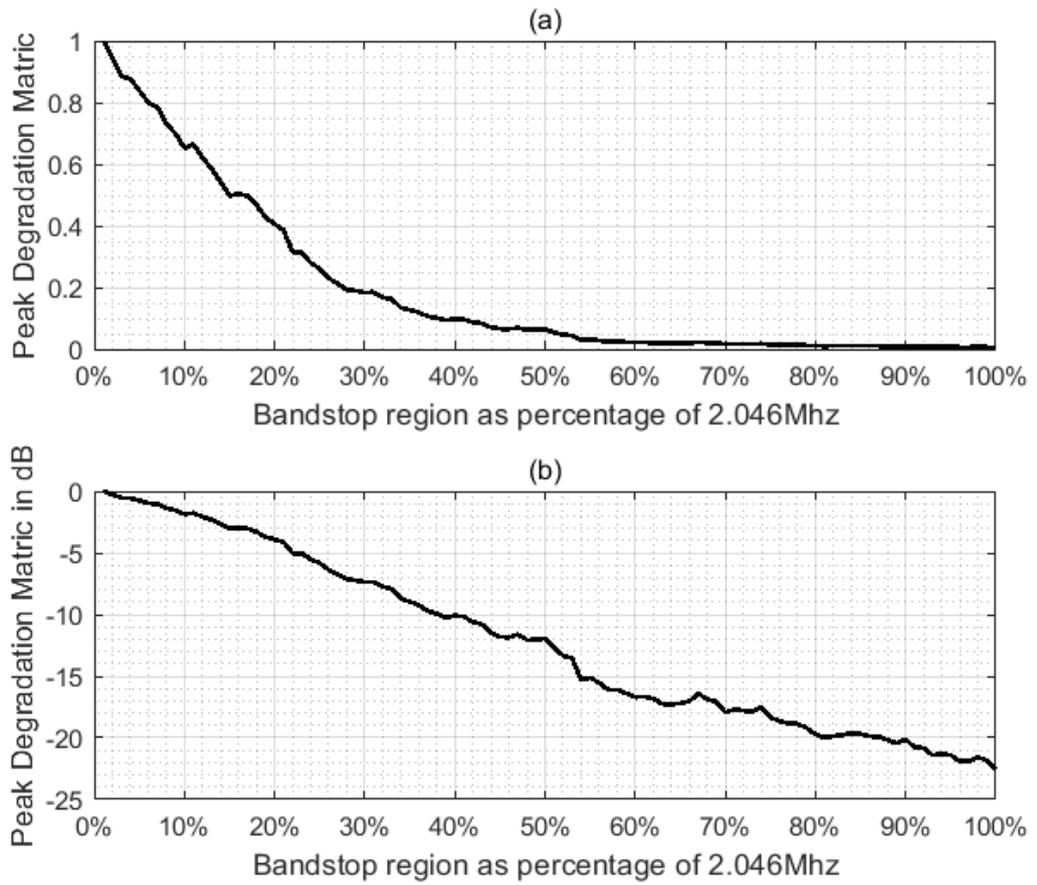


Figure 6.8 (a) PDM vs Stop-band region. (b) PDM in dB

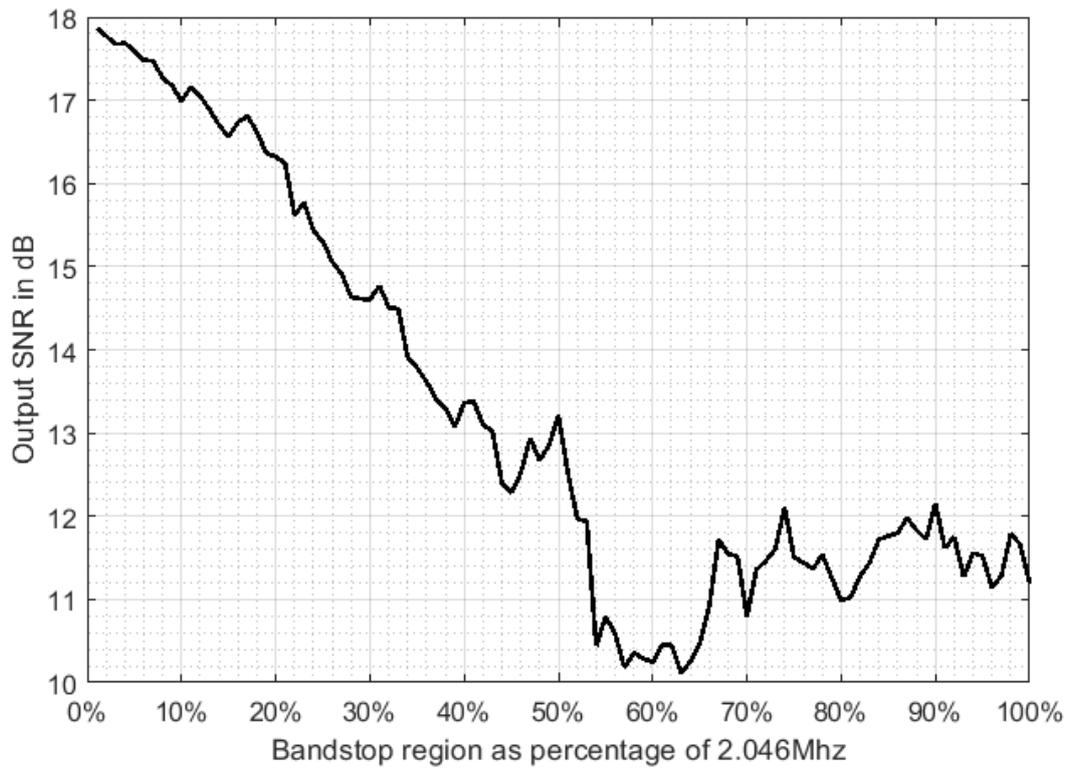


Figure 6.9 SNR at the output of the acquisition module

Figures 6.8 and 6.9 show simulation results for peak degradation metric and output SNR for ideal band-stop filter variable bandwidth. The peak degradation metric follows a similar curve as shown in Figure 6.3 for the case I. On the other hand, the SNR plot in Figure 6.9 is entirely different from Figure 6.4. As we perform coherent integration of 1ms, the output SNR in noisy conditions cannot be close to SNR achieved in an ideal scenario in case I. It can be observed from Figure 6.9 that the output SNR degrades rapidly as the stop-band region is above 40% of the signal bandwidth. The SNR is 13dB or above, between 0% to 40%, which is a good indication for coherent integration of 1ms of data during the acquisition of the signal. It can be concluded from these simulations that if an optimized (close to ideal) IIR band-stop filter is designed with 30% of the stop-band region discarding 30% of the main lobe and still GPS L1 signal is acquirable.

6.5 CASE IV: Filtering ‘C/A code +Noise + multi-tone interference’ through variable width ideal band-stop filter

In chapter five, three different adaptive notch filters were demonstrated. Lattice adaptive notch filter performed and produced more effective results. Later in chapter 5, when a lattice adaptive notch filter is used to suppress a closely packed multi-tone/ hop frequency type of interference signal, its performance is severely degraded, and the acquisition module is unable to acquire the signal. Here, ‘evil waveform’ refers to closely packed multi-tone signals separated by a few frequency bins.

The multiple simulation sets are run, with various parameters being altered. The closely packed tone was placed in such a way that it occupies 5% to 10% of signal bandwidth. Along with that, the JNR level is set to be 40dB. As shown in the last section of chapter 5, when the JNR level was about 40dB, the lattice notch filter could not remove the interference.

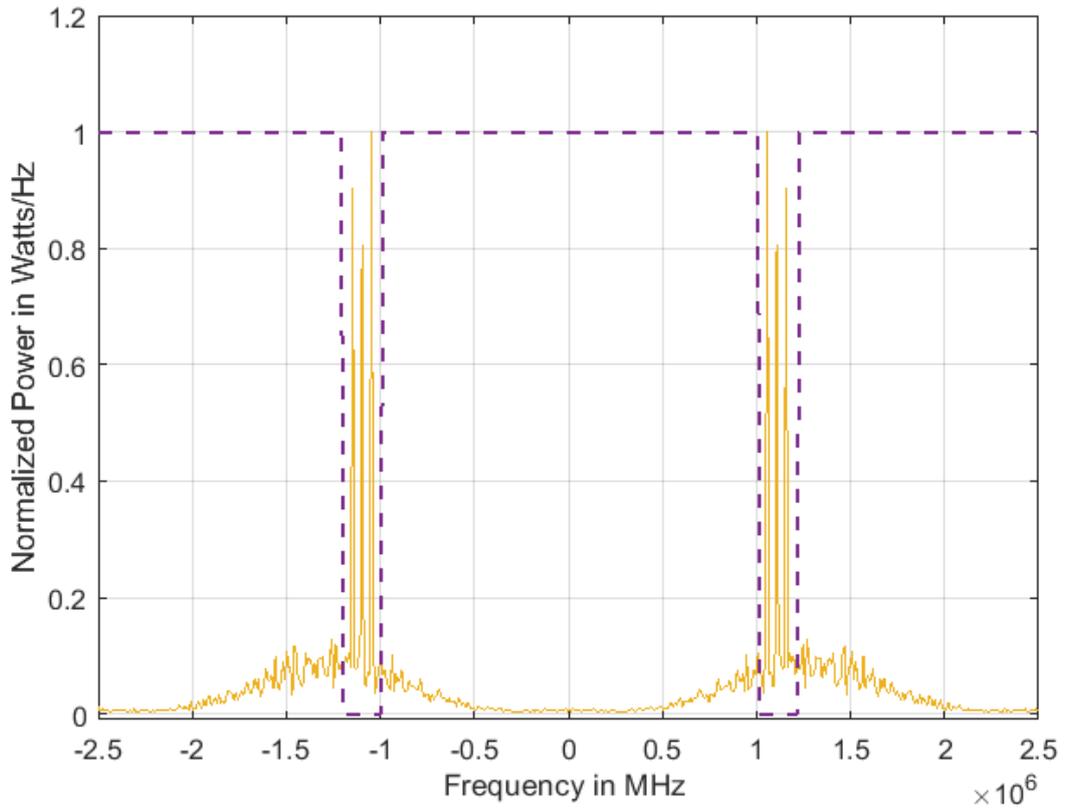


Figure 6.10 Illustration of how multi-tone interference signal suppression via band-stop filter

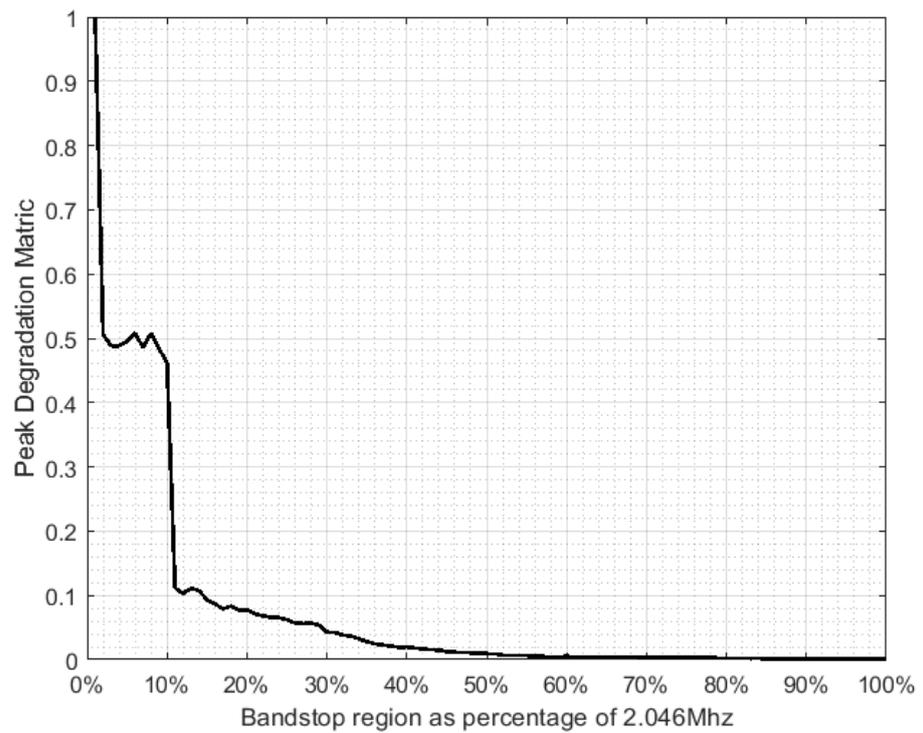


Figure 6.11 Shows the peak degradation vs Band-stop region for JNR level 40dB

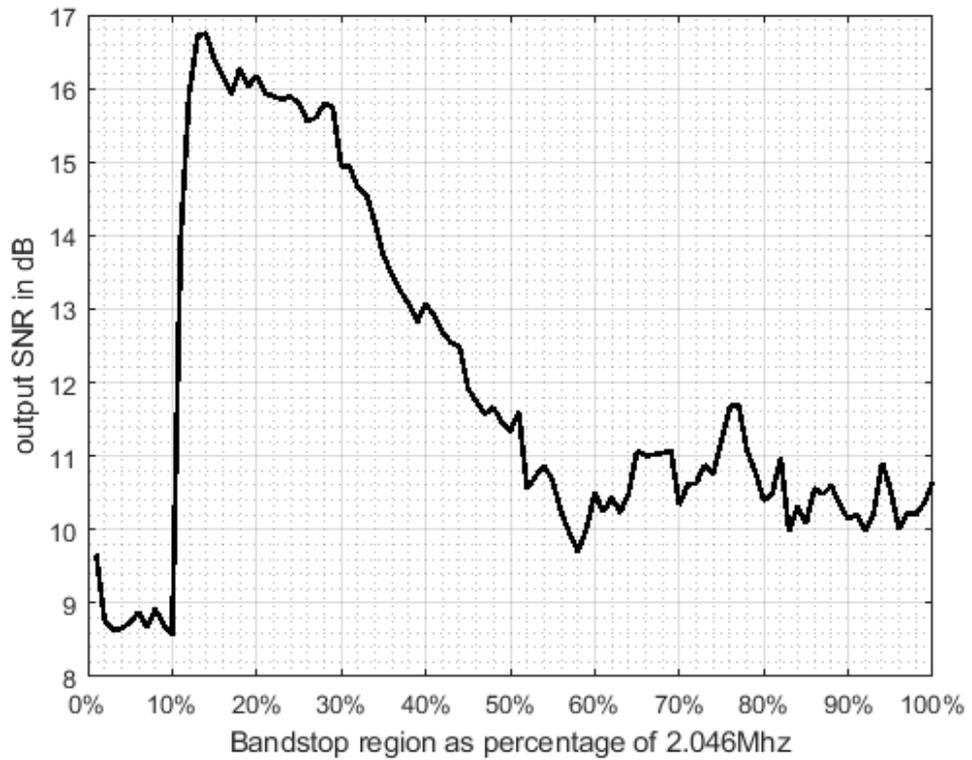


Figure 6.12 SNR at the out of the Acquisition module for JNR level of 40dB after the suppression of closely packed multi-tone interference

Output SNR in Figure 6.12 is very low as long as the stop-region is 10%. This is because the multi-tone interference occupies 10% of the bandwidth. As long as the stopband region is 10 % or less than the interference bandwidth, it is not completely removed, corresponding to a lower SNR. Between 12% and 40%, the output SNR peaks and reaches 17dB. Figure 6.13 shows the corresponding 3D acquisition plots at 15%, 25%,35% and 45% respectively.

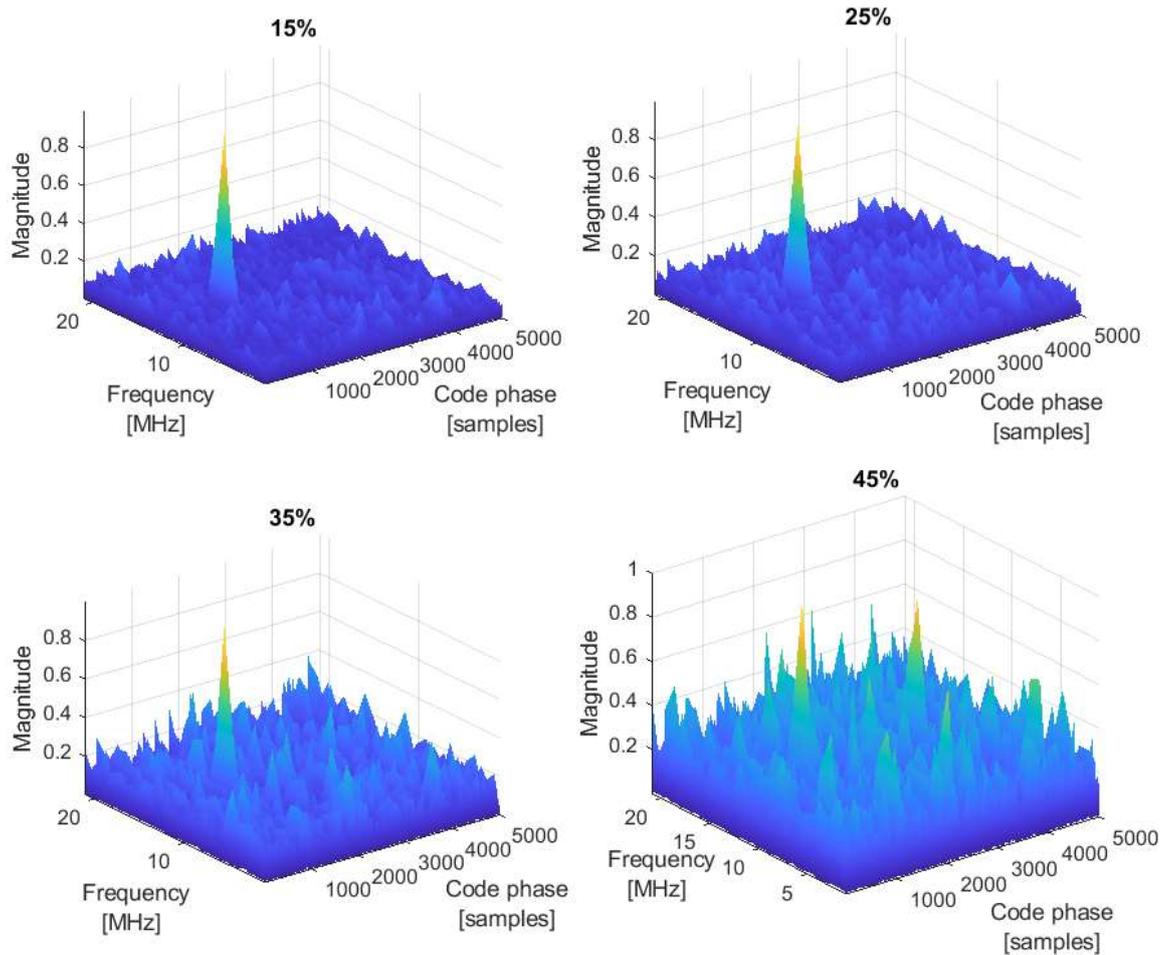


Figure 6.13 Four 3D acquisition plot at 15%, 25% 35% and 45% band-stop region

The noise floor is significantly lower, ranging from 15% to 35%, and the acquisition peak is clearly visible. But at 45%, the useful signal degradation severely produces false peaks in the CAF evaluation function. The acquisition peak ratio metric is utilized for further analysis, which is introduced in chapter 3. The acquisition peak ratio is the ratio between the highest and the second-highest peak while evaluating the CAF. If the ratio is 2.5 or above, the GPS L1 signal can be acquired. While setting variable widths to the stop-band filter, ranging from 1% to 100%, hundreds of such CAF functions are produced at the output of the acquisition module. For each CAF function at 1%, 2%, and so on until 100%, the acquisition peak ratio

is calculated and plotted, providing an alternative way to analyse the performance of band-stop filtering for mitigating interference in the main lobe of the GPS L1 signal.

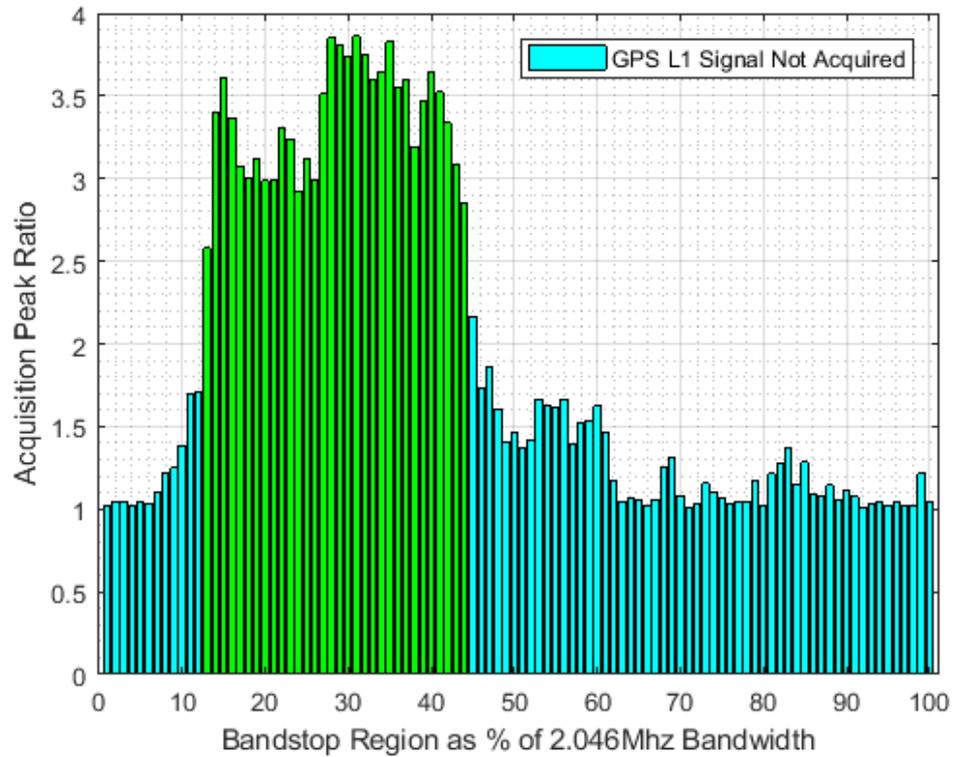


Figure 6.14 Acquisition Peak Ratio for each CAF ranging from 0% to 100%

In Figure 6.14, the green bar represents the region where the peak acquisition ratio is 2.5 or above, meaning it is where a satellite can be acquired. On the same plot, the blue bar represents a region where the satellite signal is not acquirable as the acquisition peak ratio is below 2.5. This metric has provided another performance evaluation parameter to test the proposed method.

6.6 Concluding Remarks:

The interference in the GPS spectrum can be different by type and nature, and the ANF cannot be applied as an effective method to mitigate all interference types. In the earlier chapters of this thesis, only CWI with single-tone, multi-tone and time-varying frequency interference have been analysed and mitigated. These two types of interference represent only a small percent of the interference GPS receiver encounters. The extremely weak level

of GPS signal on the earth's surface makes GPS prone to all different kinds of interference signals. Hence, it is difficult to design and develop an effective interference detection and mitigation module within a GPS receiver for various interference signals.

To tackle other kinds of interference signals, a method is proposed in [89], in which Time-Frequency analysis of incoming DSSS signals with interference signals is carried out to detect and mitigate a broader spectrum of interference signals from the useful signal. Time-Frequency Representations (TFR) are basically mapping the one-dimensional signal in the time domain onto the two-dimensional function of time and frequency. This way signal is represented over a Time-Frequency plane, revealing more information about the localisation of the interference signal component within the useful signal. Furthermore, in [90][91] and [92], TFR-based interference excision methods are used specifically for DSSS types of communication signals. Regarding GNSS signals, TFR analysis is limited by heavy computational requirements due to the length of spreading code (PRN), which is several thousands of symbols [24]. On the other hand, the memory requirement and computational load in a GPS receiver are limited and don't provide extra room for additional modules, such as stand-alone interference detection and mitigation. Therefore, fewer computational techniques, such as adaptive notch filters, have been preferred to TFR analysis. However, these ANF-based techniques are applicable to the specific class of interference signals. Thus it becomes ineffective for those interference signals which are characterized by TFR. TFR separates different desired and undesired signal components and provides a superior view to analyse and monitor different kinds of disturbing signals.

The primary motivation for these experiments on the exploitation of auto-correlation and spectrum envelop of GPS L1 signal was to provide insight into the acquisition of GPS L1 signal when a certain percentage of the information from the main lobe of the GPS L1 signal

is discarded. Then the aim is to utilize the outcome/ findings of the experiments to develop novel methods to mitigate interference in the useful signal.

As the resources available in the consumer-based GPS receiver are limited, the proposed method utilizes the in-built FFT in the acquisition module to develop new mitigation methods based on a frequency domain or time-frequency based excision of interference. The acquisition of the GPS L1 signal via a parallel frequency code phase search (Figure 6.16) is a sort of Time-Frequency analysis [91]. The evaluation of CAF in the acquisition module corresponds to the evaluation of the spectrogram (STFT), where the length of the analysis window depends on the received signal (1ms). But if the acquisition method in Figure 6.16 can be modified slightly, that can incorporate different analysis windows. Hence the acquisition module can act as a detector and mitigate the different kinds of interference by setting up some sort of threshold without any extra resources required.

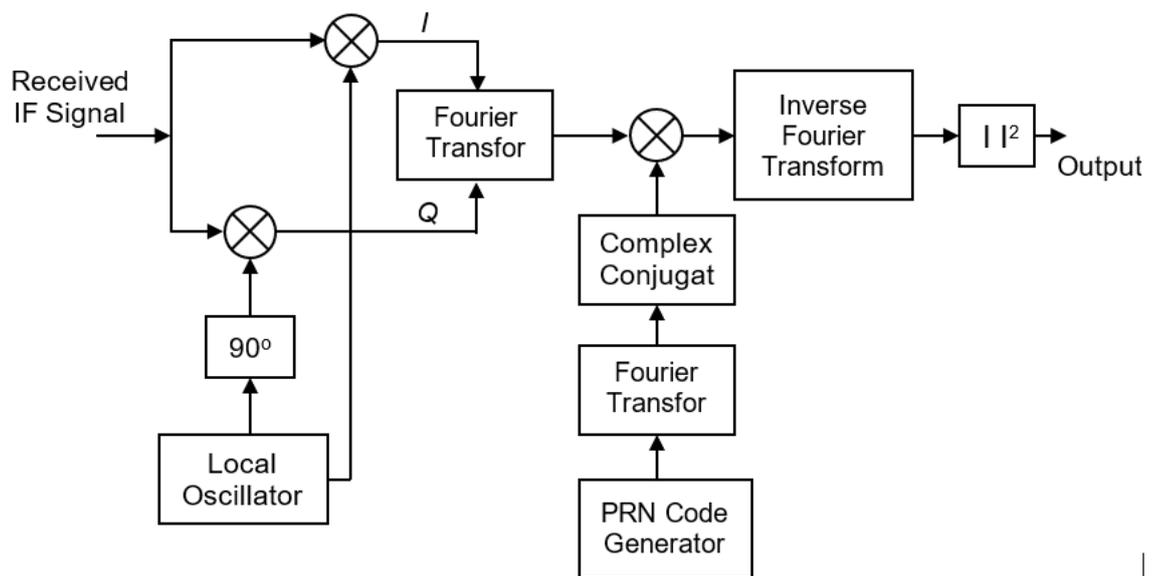


Figure 6.15 Acquisition of GPS L1 via Parallel Frequency Code Phase Search Method

Chapter 7

CONCLUSION AND FUTURE WORK

This section briefly summarizes the work presented in this thesis and outlines each chapter's outcomes.

This research has produced a vast array of simulation results for a fair comparison between existing techniques and the proposed method. Carried out system-level modelling of GPS receivers end in MATLAB to evaluate the performance of the proposed method against other counterparts. SIMULINK was used to design and develop a purpose-built GPS L1 simulator to maintain consistency across different sets of simulation results. GPS L1 simulator can generate a realistic GPS L1 signal with different strengths, code-phase delay and Doppler shift and conditions to provide homogeneous test signals to test both existing techniques and the proposed method. Moreover, simulations based on the GPS L1 simulator's time-series Intermediate Frequency or baseband samples naturally produce high-fidelity results.

Each method, whether existing or proposed, were tested under different scenarios to assess the validity, accuracy and performance in term of Carrier to Noise Density (C/N_0), Signal to Noise Ratio (SNR), Peak Degradation Metric, and Mean Square Error (MSE). Further,

these simulations were initially used to characterise existing techniques and then expanded to incorporate newly proposed methods. These results provide valuable information for designing mitigation units that use adaptive notch filters for a computationally efficient solution for interference mitigation in GPS receivers. For both existing and proposed ANF-based methods, a broad spectrum of configurations was set up within ANFs to robustly test them, run each method at its optimal performance, and finally make a comparison between them.

Chapter one provides an overview of the characteristics of GPS signals and those of fundamental concepts and definitions. A series of signal processing operations such as correlation, modulation, and multiplexing have been explained to get a coherent understanding of the GPS L1 signal to model it in Simulink successfully. Moreover, properties of PRN code are discussed, such as higher chipping rate (for improved precision), orthogonality and correlation peaks. Further insight into the BOC modulation for new-generation GNSS signals is given.

Chapter two outlines the fundamental modelling and simulation of the GPS L1 signal in Simulink. The detail on the generation of PRN code is explained as it was essential to produce the C/A codes for individual satellites in the simulator. Additionally, the P(Y) code and the navigation data are explained with the help of a series of MATLAB plots at the output of each processing stage. It provides the basis for understanding the fundamental concepts presented in chapter one and its implementation to create a software-based GPS L1 signal simulator.

Chapter Three overviews different acquisition algorithms used to acquire GPS L1 signals, such as serial search, parallel frequency search and parallel frequency code phase acquisition algorithms. Later in this chapter, serial and parallel algorithms were modelled in MATLAB.

The GPS L1 signal simulator, designed in chapter two, provided the input to the modelled acquisition loop in MATLAB. The serial search of the C/A code with 40920 sampled data points takes too much time, while the same number of data points in parallel frequency code phase search can be obtained in only two seconds. It was followed by a brief discussion on different techniques in literature to compute DFT efficiently. A literature review of recent research on the fast acquisition of GNSS signals via FFT-based algorithms was also a part of this chapter.

Chapter 3 further investigates coherent and non-coherent detectors and how they work and are used to increase the sensitivity of receivers when the input GNSS signal is weak. Analysis of different acquisition schemes within the GNSS receiver, as well as the details on the development of a MATLAB-based acquisition algorithm, was shown to improve the acquisition performance of the weak GNSS signal.

Chapter four initially gives an overview of CWI interference, its detection mitigation, and implementing a first-order complex notch filter in MATLAB to remove the interference is demonstrated. This chapter presents models of narrowband real CWI, complex CWI, and complex chirp sinusoidal interference in MATLAB.

Furthermore, a detailed analysis of the fixed complex notch filter and their performance in terms of interference suppression is demonstrated. The design of adaptive notch filters, which can autonomously perform detection and mitigation of narrowband CWI and Chirp-type interference.

A brief insight into the effect of the notch filter on the CAF function is presented. The non-linear phase of the IIR notch filter distorts the Cross-Ambiguity Function. Hence needs further processing to compensate for the bias. IIR Notch filter degrades the correlation peak amplitude and translates the autocorrelation function on the time axis. IIR Notch filter-based

schemes to mitigate CWI are less complex and easy to implement but require further research to process the coarse acquisition of GPS L1 signal. Analysis of CWI on the GPS signal is carried out, and different methods in the literature were reviewed to mitigate CWI.

Complex notch filters of orders one to six are employed to remove narrowband complex CWI from the GNSS signal. These complex notch filters' performance was evaluated in terms of output SNR at the acquisition block, mean square error, and acquisition Figure of merit. These simulations are further analysed to give a threshold setting for activating or deactivating a specific adaptive notch filter. There is a limit to the amount of interference at each one of the complex notch filters. Hence, a high-order complex filter is necessary to counter powerful interference signals without degrading the useful signal.

This chapter further presents a detailed analysis of the use of the simple gradient-based algorithm with first-order complex and second-order real IIR notch filters. Finally, the simulation results demonstrate that 1st order CANF and second adaptive IIR notch filter can efficiently suppress the **narrowband CWI, chirp-type interference, and frequency hopping signal**. A complete set of simulation results are produced and presented as follows:

- The plot of convergence of parameter z_0
- The plot of a time-frequency representation of the signal before and after mitigation
- The plot of the CAF after the mitigation of the interference.
- The plot of Doppler shift and code phase delay after mitigation of the interference

Analysis of the notch bandwidth is conducted and needed to determine the optimal notch width (k_a) as a function of the width of notch control the convergence rate of parameter z_0 was established.

A simple and innovative system-level model is proposed, which can utilize each CANF efficiently with a threshold setting of JNR estimation within the adaptation block. The threshold setting parameter provides a trade-off between the effective excision of CWI, the order of the filter, and power consumption. It results in a computationally effective solution for interference mitigation for GPS-based applications.

Different aspects and properties of the parameter z_0 is exploited, which could be beneficial for the estimation of JNR levels. The variance of the magnitude of the parameter z_0 is proposed to be set as a threshold-setting variable for estimating the level of JNR. It can be employed to toggle between different orders of the filter depending on the level of JNR. The effect of coefficient quantization is marginally considered for the first and second-order complex filter

Chapter 5 provides a detailed analysis of various adaptive algorithms for ANF too. Two direct form structure IIR notch filter algorithms were implemented [48] and [68]. Along with that, one algorithm for the lattice-based structure [51] is also implemented. The existing algorithm was then compared with the proposed fully adaptive filter based on the lattice structure, which simultaneously adapts its coefficients and enhances its performance.

A complete study of 3D frequency response and simulation of the cost function of ANF were generated and plotted, which provided the groundwork to understand and develop the proposed methods. This thesis presents a Novel Fully adaptive notch filter with full gradient term to update parameter ρ , it demonstrated excellent performance in terms of convergence speed, providing a better retrieval of the useful signal. Different resetting algorithms for parameter ρ were developed, and then a fully constraints algorithm for the adaptation of parameter ρ was proposed.

A full gradient term is derived from the filter output in section 5.4, which is then utilized to update parameter ρ in a fully adaptive notch filter. The modified version of the proposed structure with less computational complexity was presented. Furthermore, the resetting of the parameter ρ to 0.70 just before the new hop frequency is key to the successful adaptation of the adaptive notch filter. Therefore, resetting of ρ should be applied carefully within the algorithm, and the selection of the constraints has to be such that it works universally regardless of the environment. Various resetting algorithms were designed, modelled, and tested and presented with simulation results with different resetting criteria. The parameter ρ has to be sharply reset at the beginning of the next hop frequency, hence a resetting mechanism within the adaptive algorithm to automatically set parameter ρ to a desired minimum value just before the tracking of the next subsequent hop frequency was developed.

Moreover, the performance of the direct form IIR ANF and proposed lattice form ANF are compared in terms of output C/N_0 at the acquisition module. Complete system-level modelling is carried out to test both ANF filters. The proposed method is able to mitigate desired interference more precisely and enhance the output C/N_0 at the acquisition module.

At the end of this chapter, a novel method was proposed and compared with [81], as [81] also simultaneously adapted its coefficient. The simulation results have shown that the proposed method has better convergence properties for the parameter ρ and improved C/N_0 was attained. The convergence of ρ and resetting of ρ on time (at the start of new hop frequency) are two key points for successfully mitigating hop frequency CWI.

A novel lattice-based adaptive IIR notch algorithm is presented, which can simultaneously adapt its notch bandwidth and centre frequency parameters. The proposed algorithms show favourable convergence properties for both its parameters β and ρ , such as improved

tracking performance and higher C/N_0 at the output of the acquisition block when compared with the relevant competitive designs.

Simulation results have demonstrated that the proposed algorithm is consistently stable and has also proven its superior performance under different conditions. These advantages are attained due to the constraint imposed on the pole of ANF. If the pole of the ANF moves towards the instability region (unity on unit circle or more), it reverts into the stable model set in order to prevent divergence.

Chapter 6 provided a novel analysis, which is a work of the first of its kind for GPS-specific application to the best of my knowledge. The proposed interference scheme consists of three band-stop filters with different stop-band regions along the main lobe of the C/A code. As far as the interference is in the main lobe of the GPS L1 signal, the signal is severely affected. Hence interference excision from the main lobe is of the uttermost importance. On the other hand, if the interference is on the side lobes, it does not really affect the acquisition process because the main lobe is filtered out during the pre-correlation filtering stage. Initial analysis and experiment results show that even if a certain percentage of information is discarded from the main lobe of the GPS L1 signal, the signal is still acquirable as long as only 25-30% of the information is removed from the main lobe of the GPS L1 signal.

One of the main takeaways from the band-stop and band-pass experiment is that the GPS L1 signal is still acquirable even if a certain amount of information is completely removed from the main lobe of the GPS L1 signal. This provides flexibility to design and develop alternative methods and techniques to mitigate interference in GPS, not just CWI interference but other kinds of interference signals.

Such as :

- I. Filter Bank-based techniques
- II. Frequency Domain excision of the interference
- III. Time-Frequency domain suppression of the interference

The following paragraphs address these three techniques, each of which could be considered a potential future work.

I. Filter Bank-based techniques

A Filter Bank Based Method can be constructed to mitigate interference in the main lobe of the C/A code in which a number of band-stop filters with different stop-regions (B_1 , B_2 and B_3 in Figure below) are used to mitigate the interference located anywhere in the main lobe. have shown that if the 30% of the main lobe of C/A code is discarded from any side, either centre or left or right most, GPS L1 signal is still acquirable Figure 7.1 illustrates the main concept of the proposed idea. The analysis from chapter 6

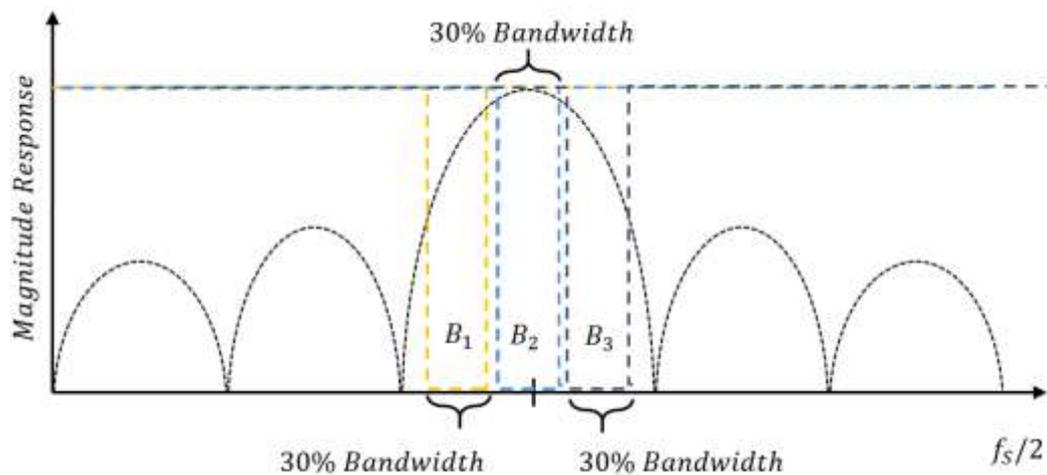


Figure 7.1 Illustration of the proposed method

II. Frequency Domain excision of the interference

The interference suppression techniques based on the frequency domain involve the removal of the interference signal by simply analysing the spectrum of a useful signal with interference. The frequency domain interference suppression technique does not rely on an adaptive mechanism or algorithm; hence its processing speed is much higher than time domain adaptive filtering. The FFT module within the acquisition module can be modified to achieve this. A different threshold setting can be applied to suppress the interference via the frequency domain method. Figure 7.2 represents the system model for implementing frequency domain inference suppression

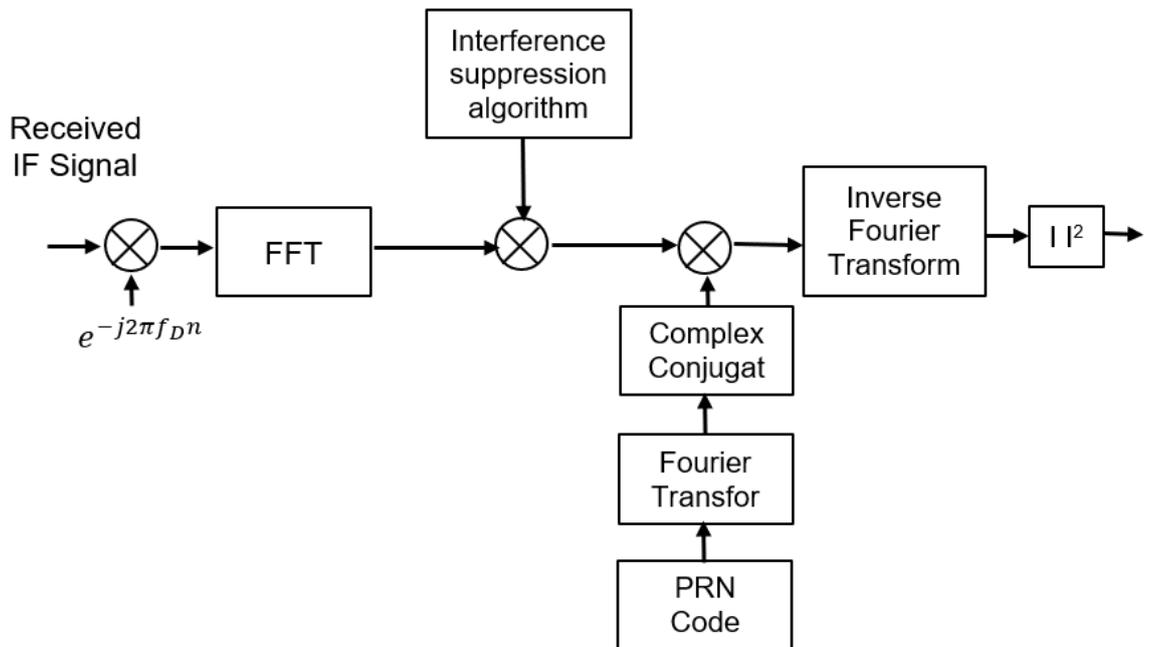


Figure 7.2 Frequency Domain interference mitigation method

III. Time-Frequency domain suppression of the interference

Time-Frequency based method provides better visualization of the interference, and different types of interference can be precisely located within a useful signal. To accomplish this, the structure of the acquisition module is exploited to implement Time-Frequency based excision within the receiver by using already available resources in the acquisition stage. The main objective is to design an efficient Time-Frequency domain suppression of the

interference based on the spectrogram. As mentioned earlier, CAF evaluation corresponds to the spectrogram evaluation. Therefore, an acquisition module within the receiver can be used to design and develop a Time-Frequency based interference excision.

REFERENCES

- [1] J. T. Curran, M. Navarro, M. Anghileri, P. Closas, and S. Pfletschinger, "Coding Aspects of Secure GNSS Receivers," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1271–1287, Jun. 2016,
- [2] F. S. Prol *et al.*, "Position, Navigation, and Timing (PNT) Through Low Earth Orbit (LEO) Satellites: A Survey on Current Status, Challenges, and Opportunities," *IEEE Access*, vol. 10, pp. 83971–84002, 2022,
- [3] D. Borio and C. Gioia, "GNSS interference mitigation: A measurement and position domain assessment," *NAVIGATION*, vol. 68, no. 1, pp. 93–114, 2021
- [4] C. J. Hegarty and E. Chatre, "Evolution of the Global Navigation Satellite System (GNSS)," *Proceedings of the IEEE*, vol. 96, no. 12, pp. 1902–1917, Dec. 2008
- [5] Y. Pan, T. Zhang, G. Zhang, and Z. Luo, "A Narrowband Anti-Jamming Acquisition Algorithm Based on All-Phase Processing for BOC Signals," *IEEE Access*, vol. 7, pp. 41416–41425, 2019,
- [6] A. Tani and R. Fantacci, "Performance Evaluation of a Pre-correlation Interference Detection Algorithm for the GNSS Based on Non parametrical Spectral Estimation," *IEEE Systems Journal*, vol. 2, no. 1, pp. 20–26, Mar. 2008
- [7] P. Wang, E. Cetin, A. G. Dempster, Y. Wang, and S. Wu, "GNSS Interference Detection Using Statistical Analysis in the Time-Frequency Domain," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 1, pp. 416–428, Feb. 2018.
- [8] D. Borio, L. Camoriano, S. Savasta, and L. L. Presti, "Time-Frequency Excision for GNSS Applications," *IEEE Systems Journal*, vol. 2, no. 1, pp. 27–37, Mar. 2008.
- [9] "GAJT-410ML GPS Anti-Jam Technology (GAJT) for smaller platforms," 2022. [online]. Available: <https://hexagondownloads.blob.core.windows.net/public/Novatel/assets/Documents/Papers/GAJT-410ML-Product-Sheet/GAJT-410ML-Product-Sheet.pdf>
- [10] "GPSdome 1.02B – Military systems," 2022. [online]. Available: https://www.oriala.com/wp-content/uploads/2021/07/gpsdome_1.02b_-_military_systems_-_oriala_cobranded_-_8.5x11_10-07-19-1.pdf
- [11] "Enabling the military to get their job done," 2022. [online]. Available: <https://www.satelliteevolutiongroup.com/articles/chemring-Q&A.pdf>
- [12] "Global Navigation Space Systems: reliance and vulnerabilities ," 2011 Royal Academy of Engineering

- [13] “Economic impact to the UK of a disruption to GNSS,” 2017, London Economics and Innovate UK.
- [14] “SENTINEL PROJECT REPORT ON GNSS VULNERABILITIES,” 2014 [online] Available: <https://rntfnd.org/wp-content/uploads/SENTINEL-Project-Report.pdf>
- [15] “Satellite-derived Time and Position: A Study of Critical Dependencies,” 2018, [online]. Available : <https://www.gov.uk/government/publications/satellite-derived-time-and-position-blackett-review>
- [16] V. K. Kanchetla, A. Kharalkar, S.Jain, S.Jose, S.K. Khyalia, J.Joy, R.Zele “A Compact, Reconfigurable CMOS RF Receiver for NavIC/GPS/Galileo/BeiDou,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 70, no. 7, pp. 3361–3373, Jul. 2022.
- [17] “GNSS User Technology Report,” 2020 [online]. Available : https://www.gsc-europa.eu/sites/default/files/sites/all/files/technology_report_2020.pdf
- [18] Misra, P., Hu, W., Jin, Y., Liu, J., de Paula, A. S., Wirström, N., & Voigt, T. “Energy efficient GPS acquisition with Sparse-GPS,” in *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, 2014, pp. 155–166.
- [19] S. U. Qaisar, C. Benson, and M. J. Ryan, “A novel efficient signal processing approach for combined acquisition of GPS L1 and L2 civilian signals,” in *2016 Military Communications and Information Systems Conference (MilCIS)*, 2016, pp. 1–5.
- [20] C. Li and J. Guan, “Code acquisition method of GPS signals based on WFTA, PFA and Cooley-Tukey FFT,” in *2012 IEEE International Conference on Intelligent Control, Automatic Detection and High-End Equipment*, 2012, pp. 119–122.
- [21] S. R. Babu, P. Selvam, G. S. Rao, and J. Wang, “Optimization of GPS L1 acquisition using Radix-4 FFT,” in *2011 International Conference on Recent Trends in Information Technology (ICRTIT)*, 2011, pp. 875–879.
- [22] K. Sun, M. Zhang, and D. Yang, “A New Interference Detection Method Based on Joint Hybrid Time–Frequency Distribution for GNSS Receivers,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 11, pp. 9057–9071, Nov. 2016
- [23] E. D. Kaplan and C. Hegarty, *Understanding GPS: principles and applications*. Artech house, 2005

- [24] P. Misra, *Global Positioning System: Signals, Measurements, and Performance*. Lincoln, Mass. : Ganga-Jamuna Press, 2006.
- [25] J. B. Tsui, *Fundamentals of Global Positioning System Receiver: A Software Approach*. Hoboken, N.J. : Wiley, 2004.
- [26] K. Borre, *A Software-Defined GPS and Galileo Receiver: A Single-Frequency Approach*. Boston, Mass.:Birkhäuser, 2007.
- [27] T. Halder and A. Bhattacharya, “Stockwell transform reduces the Doppler frequency bin size of GPS receiver,” in *International Conference for Convergence for Technology, IEEE*, 2014, pp. 1–6.
- [28] Y.- Lee, C.- Chang, W.- Mao, F.- Chang, and H.- Tsao, “Matched-filter-based low-complexity correlator for simultaneously acquiring global positioning system satellites,” *Sonar Navigation IET Radar*, vol. 4, no. 5, pp. 712–723, Oct. 2010.
- [29] Y. Zhang, W. Luy and D. Yu, “A fast acquisition algorithm based on FFT for BOC modulated signals,” in *TENCON 2015 - IEEE Region 10 Conference*, 2015, pp. 1–6.
- [30] A. H. Aboud, R. Ramadan, and T. AlSharabati, “Software defined radio implementing GPS parallel frequency space search acquisition algorithm in real time environment,” in *2015 International Conference on Information and Communication Technology Research (ICTRC), IEEE*, 2015, pp. 234–237.
- [31] A. Albu-Rghaif and I. A. Lami, “Novel dictionary decomposition to acquire GPS signals using compressed sensing,” in *2014 World Congress on Computer Applications and Information Systems (WCCAIS), IEEE*, 2014, pp. 1–5.
- [32] C. Chang and G. Huang, “Spatial compressive array processing scheme against multiple narrowband interferences for GNSS,” in *2012 IEEE First AESS European Conference on Satellite Telecommunications (ESTEL)*, 2012, pp. 1–6.
- [33] S. Kong, “A Deterministic Compressed GNSS Acquisition Technique,” *IEEE Transactions on Vehicular Technology*, vol. 62, no. 2, pp. 511–521, Feb. 2013.
- [34] R. W. Ramirez, *The FFT: Fundamentals and Concepts*. Englewood Cliffs, N.J. ; London : Prentice-Hall, 1985.
- [35] E. O. Brigham, *The Fast Fourier Transform*. Englewood Cliffs; London : Prentice-Hall, 1974.

- [36] J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics of Computation*, vol. 19, pp. 297-301, 1965.
- [37] S. K. Mitra, *Digital Signal Processing: A Computer-Based Approach*. New York, NY : McGraw-Hill, 2011.
- [38] Z. Wang, "A prime factor fast W transform algorithm," *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2361–2368, Sep. 1992.
- [39] A. Coskun, I. Kale, R. C. S. Morling, R. Hughes, S. Brown, and P. Angeletti, "The design of low complexity low power pipelined short length Winograd Fourier transforms," in *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2014, pp. 2001–2004.
- [40] M. D. Macleod, "Multiplierless Winograd and prime factor FFT implementation," *IEEE Signal Processing Letters*, vol. 11, no. 9, pp. 740–743, Sep. 2004.
- [41] Z. Qian and M. Margala, "Low-Power Split-Radix FFT Processors Using Radix-2 Butterfly Units," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 9, pp. 3008–3012, Sep. 2016.
- [42] T.- Sung, "Memory-efficient and high-speed split-radix FFT/IFFT processor based on pipelined CORDIC rotations," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 153, no. 4, pp. 405–410, Aug. 2006.
- [43] J. Wang, L. Yong, and S. Ni, "Two-stage FFT acquisition method of weak GNSS signals," in *Proceedings of 2012 2nd International Conference on Computer Science and Network Technology*, 2012, pp. 1918–1921.
- [44] J. Chen and J. Ding, "New algorithm for design of low complexity twiddle factor multipliers in radix-2 FFT," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2015, pp. 958–961.
- [45] N. Linty and L. Lo Presti, "Doppler Frequency Estimation in GNSS Receivers Based on Double FFT," *Vehicular Technology, IEEE Transactions on*, vol. 65, pp. 509-524, 2016.
- [46] M. Garrido, "A New Representation of FFT Algorithms Using Triangular Matrices," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 10, pp. 1737–1745, Oct. 2016.
- [47] D. Borio and D. Akos, "Noncoherent Integrations for GNSS Detection: Analysis and Comparisons," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 1, pp. 360–375, Jan. 2009.

- [48] Y. Chien, "Design of GPS Anti-Jamming Systems Using Adaptive Notch Filters," *IEEE Systems Journal*, vol. 9, no. 2, pp. 451–460, Jun. 2015.
- [49] D. Borio, "A multi-state notch filter for GNSS jamming mitigation," in *International Conference on Localization and GNSS 2014 (ICL-GNSS 2014)*, 2014, pp. 1–6.
- [50] Anyaegbu, E., Brodin, G., Cooper, J., Aguado, E., & Boussakta, S, "An Integrated Pulsed Interference Mitigation for GNSS Receivers," *The Journal of Navigation*, vol. 61, no. 2, pp. 239-255, 2008.
- [51] D. Borio, L. Camoriano, and L. L. Presti, "Two-Pole and Multi-Pole Notch Filters: A Computationally Effective Solution for GNSS Interference Detection and Mitigation," *IEEE Systems Journal*, vol. 2, no. 1, pp. 38–47, Mar. 2008.
- [52] M. J. Rezaei, M. Abedi, and M. R. Mosavi, "New GPS anti-jamming system based on multiple short-time Fourier transform," *Sonar Navigation IET Radar*, vol. 10, no. 4, pp. 807–815, 2016.
- [53] R. T. Ioannides, T. Pany, and G. Gibbons, "Known Vulnerabilities of Global Navigation Satellite Systems, Status, and Potential Mitigation Techniques," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1174–1194, Jun. 2016.
- [54] M. G. Amin, D. Borio, Y. D. Zhang, and L. Galleani, "Time-Frequency Analysis for GNSSs: From interference mitigation to system monitoring," *IEEE Signal Processing Magazine*, vol. 34, no. 5, pp. 85–95, Sep. 2017.
- [55] A. Nosan and R. Punalard, "A complex adaptive notch filter using modified gradient algorithm," *Signal Processing*, vol.92, no.6, pp. 1508-1514, 2011.
- [56] P. A. Regalia 1962-, *Adaptive IIR Filtering in Signal Processing and Control*. New York: New York : M. Dekker, 1995.
- [57] Y. Xiao, Y. Takeshita, and K. Shida, "Tracking analysis of a gradient-based adaptive IIR notch filter with constrained poles and zeros," in *2000 10th European Signal Processing Conference*, 2000, pp. 1–4.
- [58] P. S. Nguyen, "A New Formulation for Design of Digital IIR Band Stop Filters," in *2017 International Conference on Vision, Image and Signal Processing (ICVISIP)*, 2017, pp. 27–33.

- [59] C. Benjangkprasert, T. Somkur, and K. Janchitrapongvej, "A novel algorithm for adaptive IIR notch filter based on lattice form structure," in *2008 International Conference on Control, Automation and Systems*, 2008, pp. 2450–2453.
- [60] S. Pei, B. Guo, and W. Lu, "Narrowband Notch Filter Using Feedback Structure Tips & Tricks," *IEEE Signal Processing Magazine*, vol. 33, no. 3, pp. 115–118, May 2016.
- [61] S.-C. Pei, C.-L. Wu, and J.-J. Ding, "Simplified structures for two-dimensional adaptive notch filters," in *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS '03.*, 2003, vol. 4, pp. IV–IV.
- [62] S. Nishimura and M. Aloys, "Steady-state performance of IIR adaptive notch filters. Comparison between direct form and lattice structure implementations," in *IEEE APCCAS 1998. 1998 IEEE Asia-Pacific Conference on Circuits and Systems. Microelectronics and Integrating Systems. Proceedings (Cat. No.98EX242)*, 1998, pp. 339–342.
- [63] P. T. Wheeler and J. A. Chambers, "Complex adaptive notch filter structure for tracking multiple complex sinusoidal signals," *Electronics Letters*, vol. 49, no. 3, pp. 179–181, Jan. 2013.
- [64] P. A. Regalia, "A Complex Adaptive Notch Filter," *IEEE Signal Processing Letters*, vol. 17, no. 11, pp. 937–940, Nov. 2010.
- [65] W. L. Mao, A. B. Chen, Y. F. Tseng, F. R. Chang, H. W. Tsao, and W. S. Huang, "Design of Peak-finding Algorithm on Acquisition of Weak GPS Signals," in *2006 IEEE International Conference on Systems, Man and Cybernetics*, 2006, vol. 3, pp. 1820–1825.
- [66] W.-J. Ma, W.-L. Mao, and F.-R. Chang, "Design of adaptive all-pass based notch filter for narrowband anti-jamming GPS system," in *2005 International Symposium on Intelligent Signal Processing and Communication Systems*, 2005, pp. 305–308.
- [67] L. Musumeci, J. T. Curran, and F. DAVIS, "A Comparative Analysis of Adaptive Notch Filtering and Wavelet Mitigation against Jammers Interference," *Navigation: Journal of The Institute of Navigation*, vol. 63, no. 4, pp. 533–550, Dec. 2016.
- [68] P. Tupchai, C. Benjangkprasert, O. Sangaroon, and K. Janchitrapongvej, "A new algorithm of adaptive IIR notch filter for the detection of sinusoids," in *Asia-Pacific Conference on Circuits and Systems*, 2002, vol. 2, pp. 505–508 vol.2.
- [69] D. Borio, "Loop analysis of adaptive notch filters," *IET Signal Processing*, vol. 10, no. 6, pp. 659–669, 2016.

- [70] P. A. Regalia 1962-, *Adaptive IIR Filtering in Signal Processing and Control*. New York: New York : M. Dekker, 1995.
- [71] S. S. Haykin, *Adaptive Filter Theory*. (3rd edition.) Englewood Cliffs, NJ ; London : Prentice Hall, 1996.
- [72] L. Xianbin, W. Yueke, and C. Jianyun, "Time delay compensation of IIR notch filter for CW interference suppression in GNSS," in *2013 IEEE 11th International Conference on Electronic Measurement Instruments*, 2013, vol. 1, pp. 106–109.
- [73] A. Nosan and R. Puchalard, "Absolute Gradient Algorithm for Adaptive IIR Notch Filter," in *2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 2012, pp. 1–4.
- [74] R. Puchalard, W. Lertvasana, and P. Chumchu, "Convergence speed improvement for a variable step-size plain gradient algorithm by using variable notch bandwidth technique," in *3rd International Symposium on Image and Signal Processing and Analysis, 2003. ISPA 2003. Proceedings of the*, 2003, vol. 2, pp. 788-792 Vol.2.
- [75] U. Biswas and M. Maniruzzaman, "Removing power line interference from ECG signal using adaptive filter and notch filter," in *2014 International Conference on Electrical Engineering and Information Communication Technology*, 2014, pp. 1–4.
- [76] S. Nishimura, A. Mvuma, and T. Hinamoto, "Tracking properties of complex adaptive notch filter for detection of multiple real sinusoids," in *2013 IEEE International Symposium on Circuits and Systems (ISCAS2013)*, 2013, pp. 2928–2931.
- [77] A. Nosan, R. Puchalard, P. Wardkein, and J. Koseeyaporn, "Novel algorithm for sinusoidal interference cancellation," in *2011 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS)*, 2011, pp. 1–4.
- [78] A. Mvuma, S. Nishimura, and T. Hinamoto, "Tracking behavior of an adaptive IIR notch filter for a randomly-varying frequency," in *2008 9th International Conference on Signal Processing*, 2008, pp. 248–251.
- [79] A. Mvuma, S. Nishimura, and T. Hinamoto, "Tracking characteristics of an adaptive IIR notch filter for sinusoidal frequency variation," in *2008 51st Midwest Symposium on Circuits and Systems*, 2008, pp. 858–861.

- [80] K. M. Knill and A. G. Constantinides, “Least-mean square adaptation of the orthogonal block adaptive line enhancer,” in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, 1993, pp. 663–667 vol.1.
- [81] A. Muuma, S. Nishimura, and T. Hinamoto, “Adaptive IIR notch filter with controlled bandwidth for narrow-band interference suppression in DS CDMA system,” in *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS '03.*, 2003, vol. 4, pp. IV–IV.
- [82] A. Mvuma, S. Nishimura, and T. Hinamoto, “Adaptive optimization of notch bandwidth of an IIR filter used to suppress narrow-band interference,” in *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No.02CH37353)*, 2002, vol. 5, pp. V–V.
- [83] S. Nishimura, A. Mruma, and T. Hinamoto, “Performance improvement of DS-SS communication systems using an adaptive IIR notch filter,” in *ISCAS 2001. The 2001 IEEE International Symposium on Circuits and Systems (Cat. No.01CH37196)*, 2001, vol. 2, pp. 813–816 vol. 2.
- [84] R. Punalard, W. Lertvasana, and P. Chumchu, “Convergence speed improvement for a variable step-size plain gradient algorithm by using variable notch bandwidth technique,” in *3rd International Symposium on Image and Signal Processing and Analysis, 2003. ISPA 2003. Proceedings of the*, 2003, vol. 2, pp. 788-792 Vol.2.
- [85] A. Mvuma, T. Hinamoto, and S. Nishimura, “Improved MSE analysis of a gradient algorithm for an adaptive IIR notch filter,” in *2003 46th Midwest Symposium on Circuits and Systems*, 2003, vol. 2, pp. 764-767 Vol. 2.
- [86] R. Punalard, “On adaptive IIR lattice notch filter using a robust variable step-size for the detection of sinusoid,” in *The 8th International Conference on Communication Systems, 2002. ICCS 2002.*, 2002, vol. 2, pp. 800–804 vol.2.
- [87] L. Zhao, S. Gao, P. Li, and X. Han, “The design and validation of RF front-end platform for GPS receiver,” in *The 2010 IEEE International Conference on Information and Automation*, 2010, pp. 519–522.
- [88] A. Ucar, E. Cetin, and I. Kale, “A low complexity DSP driven analog impairment mitigation scheme for Low-IF GNSS receivers,” in *2008 IEEE/ION Position, Location and Navigation Symposium*, 2008, pp. 865–870.
- [89] R. S. Ramineni, M. G. Amin, and A. R. Lindsey, “Performance analysis of subspace projection techniques for interference excision in DSSS communications,” in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, 2000, vol. 5, pp. 2825–2828 vol.5.

- [90] G. X. Gao, M. Sgammini, M. Lu, and N. Kubo, "Protecting GNSS Receivers From Jamming and Interference," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1327–1338, Jun. 2016.
- [91] M. G. Amin, D. Borio, Y. D. Zhang, and L. Galleani, "Time-Frequency Analysis for GNSSs: From interference mitigation to system monitoring," *IEEE Signal Processing Magazine*, vol. 34, no. 5, pp. 85–95, Sep. 2017.
- [92] S. Erkucuk and S. Krishnan, "Time-frequency filtering of interferences in spread spectrum communications," in *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.* 2003, vol. 2, pp. 323–326 vol.2.