MULTI-GNSS SIGNALS ACQUISITION TECHNIQUES FOR SOFTWARE DEFINED RECEIVERS

BY

ALI ALBU-RGHAIF

Applied Computing Department

School of Science and Postgraduate Medicine

The University of Buckingham

United Kingdom

A Thesis Submitted for the Degree of Doctor of Philosophy in Computer Science to The University of Buckingham

September 2015

Abstract

Any commercially viable wireless solution onboard Smartphones should resolve the technical issues as well as preserving the limited resources available such as processing and battery. Therefore, integrating/combining the process of more than one function will free up much needed resources that can be then reused to enhance these functions further. This thesis details my innovative solutions that integrate multi-GNSS signals of specific civilian transmission from GPS, Galileo and GLONASS systems, and process them in a single RF front-end channel (detection and acquisition), ideal for GNSS software receiver onboard Smartphones.

During the course of my PhD study, the focus of my work was on improving the reception and processing of localisation techniques based on signals from multi-satellite systems. I have published seven papers on new acquisition solutions for single and multi-GNSS signals based on the bandpass sampling and the compressive sensing techniques. These solutions, when applied onboard Smartphones, shall not only enhance the performance of the GNSS localisation solution but also reduce the implementation complexity (size and processing requirements) and thus save valuable processing time and battery energy.

Firstly, my research has exploited the bandpass sampling technique, if being a good candidate for processing multi-signals at the same time. This portion of the work has produced three methods. The first method is designed to detect the GPS, Galileo and GLONASS-CDMA signals' presence at an early stage before the acquisition process. This is to avoid wasting processing resources that are normally spent on chasing signals not present/non-existent. The second focuses on overcoming the ambiguity when acquiring Galileo-OS signal at a code phase resolution equal to 0.5 Chip or higher and this achieved by multiplying the received signal with the generated sub-carrier frequency. This new conversion saves doing a complete correlation chain processing when compared to conventionally used methods. The third method simplifies the joining implementation of the Galileo-OS data-pilot signal acquisition by constructing an orthogonal signal so as to acquire them in a single correlation chain, yet offering the same performance as using two correlation chains.

Secondly, the compressive sensing technique is used to acquire multi-GNSS signals to achieve computation complexity reduction over correlator based methods, like Matched Filter, while still maintaining acquisition integrity. As a result of this research work, four implementation methods were produced to handle single or multi-GNSS signals. The first of these methods is designed to change dynamically the number and the size of the required channels/correlators according to the received GPS signal-power during the acquisition process. This adaptive solution offers better fix capability when the GPS receiver is located in a harsh signal environment, or it will save valuable processing/decoding time when the receiver is outdoors. The second method enhances the sensing process of the compressive sensing framework by using a deterministic orthogonal waveform such as the Hadamard matrix, which enabled us to sample the signal at the information band and reconstruct it without information loss. This experience in compressive sensing led the research to manage more reduction in terms of computational complexity and memory requirements in the third method that decomposes the dictionary matrix (representing a bank of correlators), saving more than 80% in signal acquisition process without loss of the integration between the code and frequency, irrespective of the signal strength. The decomposition is realised by removing the generated Doppler shifts from the dictionary matrix, while keeping the carrier frequency fixed for all these generated shifted satellites codes. This novelty of the decomposed dictionary implementation enabled other GNSS signals to be combined with the GPS signal without large overhead if the two, or more, signals are folded or down-converted to the same intermediate frequency. The fourth method is, therefore, implemented for the first time, a novel compressive sensing software receiver that acquires both GPS and Galileo signals simultaneously. The performance of this method is as good as that of a Matched Filter implementation performance. However, this implementation achieves a saving of 50% in processing time and produces a fine frequency for the Doppler shift at resolution within 10Hz.

Our experimental results, based on actual RF captured signals and other simulation environments, have proven that all above seven implementation methods produced by this thesis retain much valuable battery energy and processing resources onboard Smartphones.

Acknowledgements

To Almighty Allah, the most Gracious and Merciful, for inspiring me and giving me the energy, health and determination to do this work. In addition, thanks to the Lord for the team of people who supported this study to whom I am dedicating this work, which took a lot of efforts to bring into fruition.

To My sponsors I would like to express my thanks to the Ministry of Higher Education and Scientific Research in Iraq for sponsoring my PhD programme of study. In addition, I would like to thank the European Cooperation Science and Technology (COST) group who graciously offered me a grant in November 2014 to test my proposed methods in the laboratory at Ghent University, Belgium.

To My parents I would like to dedicate the entire fruitful outcome of this work to my parents who have been waiting for so long to see the result of their son's work, to my grandmother, who hasn't stopped praying for this work to be successfully accomplished.

To My Family for all their patience and empathy, I cannot adequately express my thankfulness and appreciation for my wife and lovely sons & daughters in words, for all your support and encouragement you offered me for the duration of this thesis, especially the times when I was abrupt, at times short tempered and overall a difficult temperament to be around with.

Many thanks go to the rest of my extended family, particularly brother and sisters for their support and encouragement.

To My Supervisor Dr. Ihsan Lami with my gratitude for all his patience, viable advice, discussions, convincing arguments and most importantly, his friendship during these years. I wish him all the best for the future.

To My Friends and Colleagues a huge appreciation to all my friends everywhere, my colleagues and staff in the Department of Applied Computing, in the University of Buckingham, for all their encouragement and support, and all the PhD candidates in the department for the atmosphere that was conducive to overall learning and growth.

Abbreviations

ADC	Analogue-to-digital converter
AFE	Analogue front end
AWGN	Additive white Gaussian noise
BOC	Binary offset carrier
BPS	Bandpass sampling
BPSK	Binary phase shift keying
BPSR	Bandpass sampling receiver
C/A	Coarse Acquisition
CBOC	Composite binary offset carrier
CDMA	Code division multiple access
CCF	Cross-correlation function
C/N	Carrier to noise ratio
CMS	Compressive multichannel samplers
CS	Compressive sensing
CSSR	Compressive sensing software receiver
CTF	Continuous to finite
DAC	Digital-to-analogue converter
DC	Dual channel
DCSR	Dynamic compressive sensing receiver
DLL	Delay look loop
DSB	Dual sidebands
DSP	Digital signal processing
ESCE	Enhanced subcarrier conversion elimination
FDMA	Frequency division multiple access
FFT	Fast Fourier transform
FNZ	First Nyquist zone
FPGA	A field-programmable gate array
Galileo	European Union navigation system
GCSR	GPS compressive sensing receiver
GLONASS	Russian navigation system
GNSS	Global navigation satellite system

GPS	American navigation system
HT	Hilbert transform
IFFT	Inverse fast Fourier transform
LFSR	Linear feedback shift register
LOS	Line of sight
MF	Matched filter
NLOS	Non-line of sight
OGSR	Orthogonal Galileo software receiver
OMP	Orthogonal matching pursuit
OS	Open service
PLL	Phase locked loop
PRN	Pseudorandom noise
QPSK	Quadrature phase shift keying
RD	Random demodulation
ReMBo	Reduce and boost
RF	Radio frequency
RMSE	Root-mean-square error
SC	Single channel
SCSSR	Single-GNSS signal compressive sensing software receiver
SDR	Software defined radio
S-GPS	Sparse-GPS
SNR	Signal to noise ratio
SSB	Single sideband
SV	Space vehicle
TMBOC	Time multiplexed binary offset carrier

Contents

List of Figures

List of Tables

Declaration

*I hereby declare that all the work in my thesis entitled "**Multi-GNSS Signals Acquisition Techniques for Software Defined Receivers"** is my own work except where due reference is made within the text of the thesis.*

I also declare that, to the best of my knowledge, none of the material has ever previously been submitted for a degree in this or any other university.

Ali Albu-Rghaif

Chapter 1

Introduction

The Global Navigation Satellite System (GNSS) is a constellation of satellites, transmitting signals from space that offers navigation, positioning and timing data. In fact, there are various satellite systems such as the United States (GPS), the Europe's (Galileo) and the Russian (GLONASS) navigation systems. Each of these GNSS systems has different constellation like a number of orbits, the number of satellite in each orbit, etc.

Practically, in challenging conditions such as urban canyons or harsh environments, the GNSS signals of a single system, like GPS signal, sometimes are not sufficient to provide accurate positioning. Prior to 2012, a GPS receiver was considered as a standard technology for localisation/navigation receiver inside most Smartphones. A growing number of Smartphones, 3.1 billion devices in 2014 and expected to be over 7 billion devices in 2019, as well as the increasing demand on the localisation based services led designers to combine multi-GNSS signals in a single solution based on multi-GNSS receivers implementation, in order to increase the chance of finding localisation in multipath environment. As a result, the multi-GNSS solution is nowadays becoming an essential criterion in most Smartphones designs. However, this extra processing required (in hardware or software processing) has to consider the limited resources such as processing and battery budget as well as cost and size of such multi-GNSS solution.

From a technical point of view, the new Galileo and the GLONASS-CDMA civilian transmission signals are designed to improve the horizontal and vertical localisation. These new technologies will offer better performance than the GPS-C/A signal in mitigating the multipath effect [1]. Therefore, Smartphone based solution with multi-GNSS signal receiving capability will improve the accuracy of localisation by factor 2 in open sky and urban area; thus reducing the time to first fix and increasing the in-view-signal availability to 95% [2]. This is achieved because combining multi-GNSS signals in a single localisation solution enhances the satellite-user geometry and increases the number of satellites in view, as depicted in Figure 1-1.

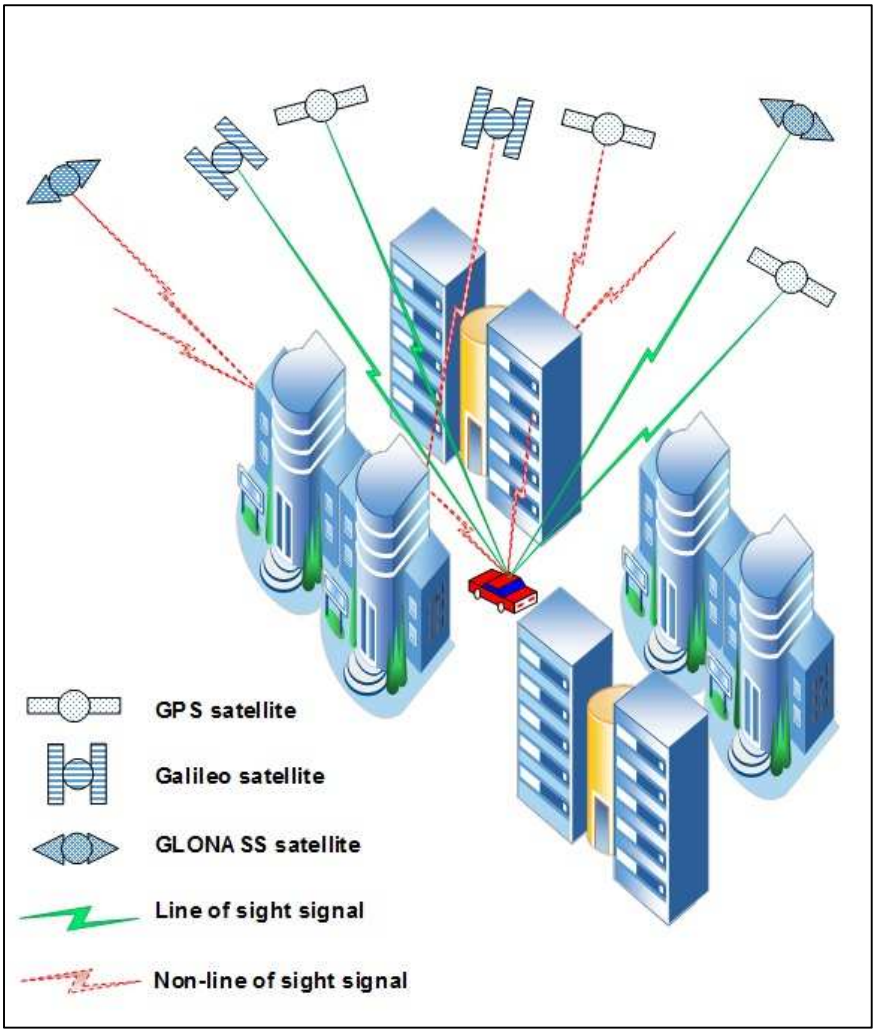


Figure 1-1 Multi-GNSS constellation in a multipath environment

By design, the specification of these Galileo and GLONASS-CDMA signals are interoperable with existing GPS so as to encourage tight integration between these technologies in future solutions [3]. Consequently, most of these modern GNSS signals share the same technical aspects like the modulation techniques, the carrier frequencies, and spreading at the same chipping rate (albeit, these systems have completely different constellation, signal power, spreading codes, etc.). This makes pursuit for a more energy efficient, faster to acquired, more integrated, and high-sensitivity multi-GNSS signal solution very attractive to researchers. As a result, it is estimated that 60% of the current Smartphones are being offered with dual GNSS (GPS-CA and GLONASS-FDMA signals) receivers onboard [4]. Such solutions are typically implemented with minimum integration in signal processing (side-by-side like parallel processing in hardware and/or software) requiring extra power consumption and processing resources. As our literature survey, detailed in Section

2.2 and Section 6.2, shows that several implementations have recently combined the processing from transmissions of either different systems (GPS, Galileo and/or GLONASS), or from the same system (GPS-L1 and GPS-L2). Most of these implementations do improve the localisation issues but result in undesirable processing overhead.

This thesis focuses on addressing combining multi-GNSS signals that are transmitting at same carrier frequency (GPS-L1, Galileo-E1 and GLONASS-L1-CDMA signals) in a single processing chain to reduce the processing resources in a multi-GNSS signal receiver, as well as to enhance user localisation. The combined multi-GNSS signal receivers are designed to detect/acquire the multi-GNSS signals simultaneously in a single receiving/correlation chain.

This thesis also details other implementations that are developed to solve a specific challenge when acquiring single GNSS signal and/or aided to combine multi-GNSS signals. For example, overcoming the ambiguity of acquiring the Galileo signal enables to combine the acquisition of the Galileo signal with the GPS signal in a single CS based process. Also, acquiring the GPS signal based-CS technique helps to understand the matching process in the CS-domain.

1.1 Research challenges and achievements

I completed my MSc degree in Applied Computing from the University of Technology in Baghdad, Iraq. Then I started working at the University of Technology and later at the Engineering College under Dyiala University. During my tenure, I was awarded a scholarship to study Ph.D. in Applied Computing, and my proposal was Air Traffic Management Software Simulator. Upon arrival here at the University of Buckingham, I met the staff members of Applied Computing (Dr. Ihsan Lami and Dr. Sabah Jassim) to discuss my project proposal. After much deliberation, I learnt that there are many subjects and projects under the Air Traffic Management umbrella. Unfortunately, I was not interested in any of them. I also concluded that the work based on the simulator is to serve a purpose rather than be a novelty. They were kind enough to suggest different projects, and I chose this project because it fulfils my aspirations. In addition I wanted to understand the type of the services provided through these signals, which my country can benefit from and capitalise on,

especially given that the infrastructure of this kind of services is immature in my country. On the other hand, admittedly I found the research in this particular field a bit challenging because my academic background is entirely different. Fortunately, I have been encouraged and offered timely guidance by my supervisor in helping me tackle and surmount all my obstacles.

The main challenges were: -

1- Understanding the concept of the GNSS system.

2- Comprehending the signal modulation and the effects on the transmission line from a satellite to the GNSS receiver.

3- Realising the processing of the received GNSS signals such as receiving, acquiring, tracking and demodulation.

4- How to combine these signals into a single efficient chain.

Building a solid background in the GNSS signals formed an active part of my learning process. I would like to attribute my gradual but definite progress to the generous efforts of Dr Ihsan Lami, who was also kind enough to invite me to become a part of the GNSS research team in the Applied Computing Department.

I thoroughly enjoyed acquiring the expertise needed for the project as well as enjoying the work with the GNSS team. The experience and knowledge I have gained during this research helped me to overcome the challenges and difficulties in this particular project. In addition, this work has improved my skills in the academic research. I would most certainly like to continue my work on the GNSS signals in the future.

This thesis offers my achievements and contributions that I have made during my 4 years study, some of which were to combine multi-GNSS signals in single function in hardware/software that can be used to help the multi-GNSS receiver designers.

Research progress

At the beginning of this research, I wanted to build a firm base in this field. My research started by understanding the GPS satellite constellation and the GPS signal modulation, which is based on BPSK modulation. This thread led to investigating the BOC modulation, Galileo signal processing and finding out the difference/overhead between these types of modulations.

The effect of a multipath signal on the GPS signal tracking process and the techniques/algorithms that were proposed to mitigate the multipath effect have been studied and explored in this research. The purpose of this study is to comprehend the signal processing of the tracking signals as well as to understand the multipath effect, especially the function of the early-late correlator inside the DLL.

Then, as shown in Figure 1-2, the following researches investigated the GPS signal acquisition (such as Serial Search Acquisition, Parallel Frequency Space Search Acquisition, Parallel Code Phase/FFT Search Acquisition, Matched Filter Search Acquisition and other researchers' methods). Dilution of Precision (DOP) and the BPS technique have also been investigated. In particular, in the BPS technique the research focused on how to capitalise on the BPS technique to sample multi-GNSS signal and the folded frequency calculation in the FNZ.

Since October 2011 there have been two primary working areas to develop new algorithms or methods, which are compatible in combining multi-GNSS signals, and as follows:

1. Processing GNSS signals based on BPS technique.
2. Processing GNSS signals based on CS technique.

Proposing two approaches concluded the progress based on BPS technique, and these approaches are proposed to detect multi-GNSS signal (GPS-L1, Galileo-E1 and GLONASS-CDMA-L1 signals). The detection in the first approach, which was contributed by my colleague Mr Maher Al-Aboodi (PhD candidate at the University of Buckingham), is based on using Volterra Series, more details in [5]. While, the second approach takes the advantage of the signal modulations, like the BPSK modulation and BOC modulation, to fold the GNSS signals without overlapping between them in the FNZ, by filtering the side lobe of the BOC signals.

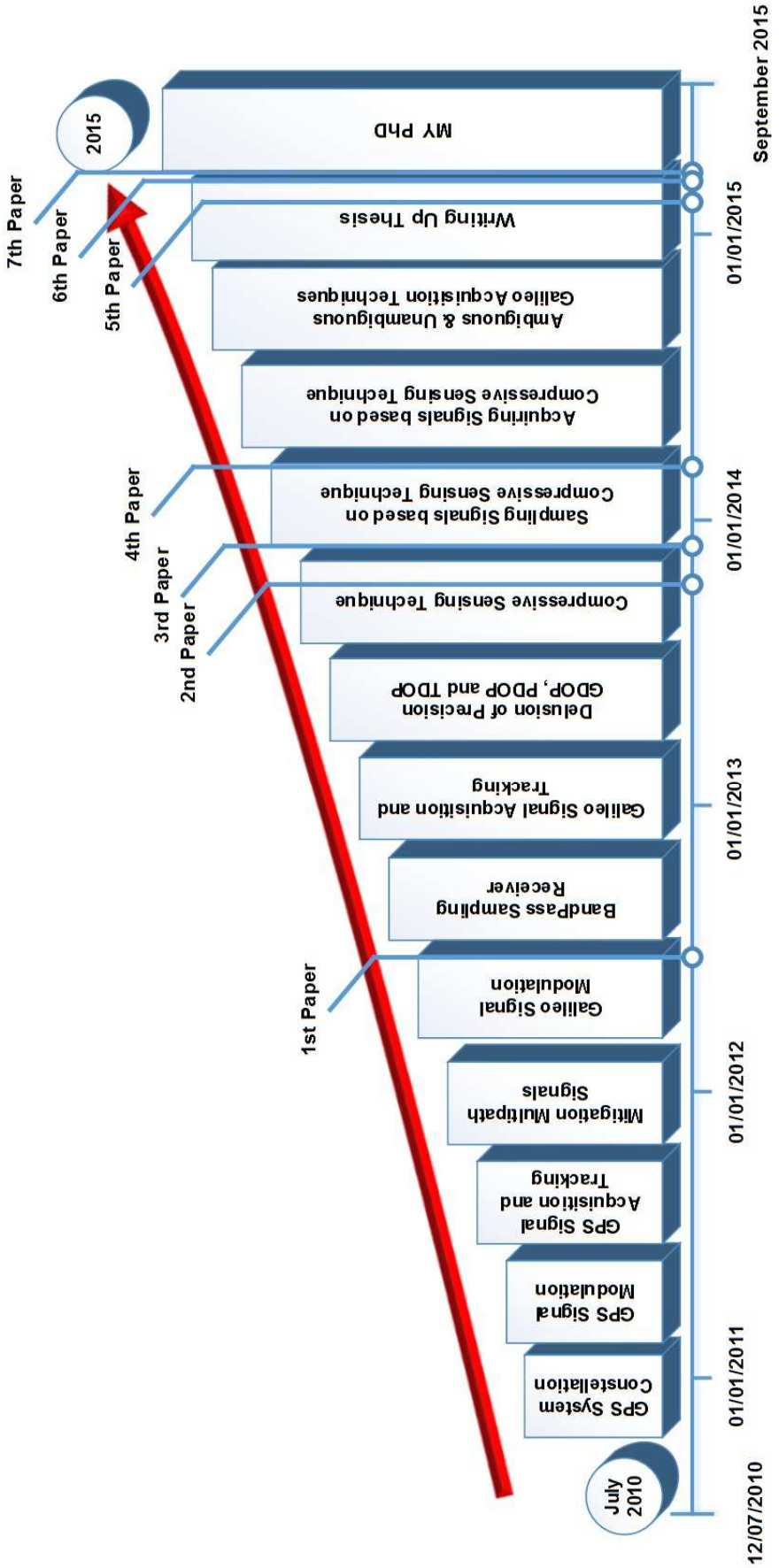


Figure 1-2 Time vs. research progress and publications

Other implementations also utilised BPS technique to receive the Galileo-OS signal. These implementations introduce new methods to acquire this Galileo signal. The first method "ESCE" is designed to:

1. Overcome the ambiguity issue when acquiring Galileo signal at ≥ 0.5 Chip resolution.
2. Enhance the probability of detection.
3. Accelerate the acquisition process.
4. Enable existing GPS receivers to acquire the Galileo signals without large overhead.

The second method "OGSR" focused on joining both data and pilot Galileo signal to acquire them in a single correlation chain. In this method, the data and pilot Galileo signals were combined in the orthogonal format, by shifting the phase of a copy of the received signals by 90-degrees and then adding it to the original received signal. The motivation of having an orthogonal signal is to:

1. Reduce the cost of the acquisition process by saving valuable resources.
2. Maintain the 3dB power of the received two signals.
3. Decrease the acquisition time.
4. Provide a cost effective implementation algorithm for Smartphone's software receiver.

In each method, the Galileo signal has been developed firstly in the simulation environment using MATLAB Simulink-based platform and then in real wireless communication channel using the Signalion HaLo-430 platform.

The second research area is based on the CS technique. The main challenge is how to utilise this technique that has been specifically proposed for image processing, to be used in the signal processing. In this particular research part, DCSR a dynamic GPS signal acquisition based on CS was implemented. The reasons for having a dynamic design are:

1. The dynamic scenario becomes very usual nowadays especially in the Smartphone or in the navigation devices.

2. All of the algorithms in the typical solutions are designed with either fixed number of correlators or fixed length of the data used in the acquisition. Increasing one of these algorithm parameters causes an increase in the processing time.

This designed receiver overcomes these static limitations, where in our DCSR implementation the receiver location determines the number of required channels, i.e. minimum number of channels is adopted when GPS receiver is located outdoors and vice versa. The other implementation “GCSR” proposes to enhance the previous work by utilising a deterministic matrix to improve the orthogonality in the sensing channels. Different deterministic matrices were used such as Hadamard matrix and Jacket matrix, and both of them have the same performance.

The experience of using CS to acquire GPS signal resulted in finding a new way that can offer further savings in terms of processing time and memory requirements. A new implementation was proposed to acquire GPS signal only (SCSSR) by decomposing the dictionary matrix. The dictionary matrix represents a bank of correlators, which are used to determine the satellite number, code phase delay and frequency Doppler shift. The decomposition is achieved by making carrier frequency fixed for all generated PRN codes. For that reason, we have modified the search algorithm, the OMP algorithm that is used in most of the CS-based implementations to search in two dimensions rather than in one dimension. Significant savings were achieved in this work by generating a bank of codes rather than a bank of correlators.

To capitalise on this saving, we then investigated how to use this process for acquiring multi-GNSS signal (CSSR), the GPS-C/A-code and the Galileo-OS-code signals. Even though, the GPS-C\A signal and the Galileo-OS signal have different modulation techniques (BPSK and BOC) but they share the same centre frequency. Nevertheless the SCCSR implementation and the ESCE method make combining two GNSS signals in single CS framework easier. Both of the GPS and Galileo signals have first been generated in the simulation environment using MATLAB Simulink-based platform. Then both signals are transmitted and received in realistic environments using the Signalion HaLo-430 platform.

I would like to acknowledge the combined efforts of the University of Buckingham and Ghent University. The European Cooperation Science and Technology (COST)

group graciously offered me a grant in November 2014 to test my proposed methods in the laboratory at Ghent University. During this visit, I used the Signalion HaLo-430 platform to send and receive real-time GNSS signals in a realistic environment. In point of fact, various scenarios were collected to validate the performance of my proposed methods.

Research novelties and achievements

During this 4-year research study, the following papers and novelties were published with fellow researchers within the Department of Applied Computing at The University of Buckingham:

1. Maher Al-Aboodi, **Ali Albu-Rghaif**, Ihsan Lami, " *GPS, Galileo and GLONASS L1 signal detection algorithms based on bandpass sampling techniques*". Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2012 IEEE 4th International Congress, pp. 255-261.

2. **Ali Albu-Rghaif**, Ihsan Lami, "*DCSR: A dynamic channel and resolution sampling for a Compressive Sensing receiver to acquire GPS signals*". Microwaves, Communications, Antennas and Electronics Systems (COMCAS), 2013 IEEE International Conference, pp. 1-5.

3. Ihsan Lami, **Ali Albu-Rghaif**, Maher Al-Aboodi, "*GCSR: A GPS Acquisition Technique using Compressive Sensing enhanced implementation*". International Journal of Engineering and Innovative Technology (IJEIT), 2013, vol. 3, no. 5, pp. 250-255.

4. **Ali Albu-Rghaif**, Ihsan Lami, "*Novel Dictionary Decomposition to Acquire GPS Signals Using Compressed Sensing*". International Conference on Network Computing and Applications (ICNCA), 2014 IEEE International Conference pp. 1-5.

5. **Ali Albu-Rghaif,** Ihsan Lami, Maher Al-Aboodi, Patrick Van Torre, Hendrik Rogier "*Galileo Signals Acquisition Using Enhanced Subcarrier*

Elimination Conversion and Faster Processing". In the 3rd Computing, Communication and Information Technology (CCIT) conference, 2015.

6. **Ali Albu-Rghaif,** Ihsan Lami, Maher Al-Aboodi "*OGSR: A Low Complexity Galileo Software Receiver using Orthogonal Data and Pilot Channels*". In the 3rd Computing, Communication and Information Technology (CCIT) conference, 2015.
7. **Ali Albu-Rghaif**, Ihsan *Lami "CSSR: a 2FOR1 Compressive Sensing Software Receiver with combined correlation for GPS-CA and Galileo-OS signals"*. In The Institute of Navigation (ION+GNSS) 2015.

8. Maher Al-Aboodi, Ihsan Lami, **Ali Albu-Rghaif,** Patrick Van Torre, Hendrik Rogier *"A Single Acquisition Channel Receiver for GPS L1CA and L2C Signals Based on Orthogonal Signal Processing"*. In The Institute of Navigation (ION+GNSS) 2015.

1.2 Thesis organization

The rest of the thesis is organised as follows:

Chapter two provides the concept of the bandpass sampling receiver and presents our "quick-early GNSS signal detection" implementation. In this chapter we shall explain how we overcome the overlapping amongst the GPS, Galileo and GLONASS L1-signals in the FNZ in order to make each signal have distinct folded frequency, which makes signal detection easier.

Chapter three explains the acquisition ambiguity when acquiring Galileo-OS signal at 0.5 Chip and discusses the mostly used solutions proposed to overcome this issue. In addition, we shall describe our new ECSE unambiguous method. Our analysis focuses on one hand, on the cross-correlation function compared with ambiguous solutions, and on the other hand, on the performance and complexity with commonly used solutions.

Chapter four presents our novel OGSR method that combines the data and the pilot Galileo signals in an orthogonal format. The method structure and the mathematical model are also explained. The assessment in this particular method introduces an

experiment study to determine the appropriate threshold that must be used in the acquisition using frequency-domain search algorithm.

Chapter five describes the main concept of the CS technique and reviews the latest solutions that are proposed based on CS. This chapter comprises two solutions that are designed to acquire the GPS-C/A-code signal. The first solution is the novel DSCR that dynamically resizes both of the CS sensing channels number and the measurement matrix according to the received power of the GPS signal. This dynamic design adds more freedom to manipulate and assign the required resources to be prepared when acquiring GPS signal outdoors and indoors. The second solution is the GCSR that has been designed to improve the sensing matrix by utilising a deterministic matrix in such GPS receiver.

Chapter six details the study of decomposing the CS-dictionary matrix of the GPS signal, followed by reviewing the multi-GNSS signal receivers. Acquiring Multi-GNSS signals and single GNSS signal implementations are explained in this chapter. The evaluation of both implementations focuses on the probability of detection, computational complexity and the frequency resolution.

Chapter seven concludes our work and highlights the significant achievements, as well as pointing out the potential directions for future research.

Chapter 2

Multi-GNSS Signal Detection

The GNSS signals that reach a receiver are weak even when outdoors, which are under the noise level of around -130 dBm [6]. In a typical GNSS receiver, acquiring the signal requires hundreds of correlators to be prepared for hardware implementation or it involves a lot of digital processing in software implementation. In addition, when the GNSS-receiver is located in bad reception area or indoors, the received signal degrades by about (25-30) dB [7]. This causes the receiver to thrash all its available resources to find the signal, based on filtering or guessing algorithms, and dependent on the receiver architecture. As a result, the acquisition process drains the receiver's resources, such as battery energy.

Moreover, multi-signal GNSS (including GPS and GLONASS) solutions are nowadays rolling out in most Smartphones. Solutions that implement these various GNSS-receivers side-by-side will be costly (processing, power, area, etc.) and will still mean that only one signal type is processed at a time. Hence, the key requirements for any GNSS solution on a Smartphone are integration in a small size; take advantage of all the GNSS signals available while using minimum power, and to be low cost. BPS receiver's architecture is a good fit, and so it is more likely to meet these requirements since it is designed to handle multi-signals in a single RF chain [8]. However, most of the proposed implementations are based on combining multi-GNSS signal that transmit at different frequency bands, as detailed in Section 2.2. Resultant, a higher sampling frequency is required to sample these signals, because they occupy different information bandwidths. To clarify that, GPS-L1 signal accommodates 2MHz information bandwidth and GPS-L5 accommodates 24MHz information bandwidth. Therefore, the required processing time for the GPS-L1 signal will be at least 10 times if it is combined with the GPS-L5 signal, since the minimum sampling frequency when combined with these signals based on BPS technique will be at least 52MHz if there is non-overlapping between these signals. While if the GPS-L1 signal combines for example with Galileo-E1 signal the

minimum sampling frequency will be equal to 12MHz, and 4MHz if it is not combined as explained in Section 2.1.

In order to minimise the processing overhead, detecting multi-GNSS signal prior to the DSP, i.e. in RF front-end, would prepare the processing algorithms like acquisition and tracking of the GNSS receiver for dealing with the available GNSS signals only.

2.1 Basic concept of the Bandpass Sampling technique

The BPS is a technique that eliminates the need for analogue mixers, as used in traditional receiver's design [9], by bringing the ADC as close as possible to the antenna as shown below in Figure 2-1. This is achieved by folding the "information band" at the centre frequency of the received signal (or at the "information band" at the centre frequencies of the received signals in the case of a multi-signal BPSR) to the FNZ without any requirement to the down-conversion process. Therefore, it is important to choose a suitable sampling frequency to prevent overlapping of the signal with itself or with other signals in a multi-signal BPSR scenario in the FNZ. Consequently, this makes multi-signal BPSR a good candidate for use in the SDR and cognitive radio [10].

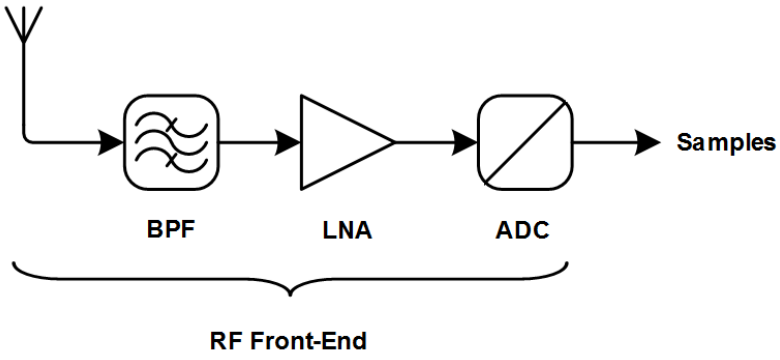


Figure 2-1 BPS receiver

Practically, the minimum sampling frequency based on BPS has to be double the bandwidth of the received signals [11]. This means that the sampling frequency is a fraction of the Nyquist rate and much less than the carrier frequency of the received signal. Equation (2.1) shows the mathematical relationship defining the folding of the carrier frequency to the FNZ.

$$f_{fold} = \begin{cases} rem(f_c, f_s) & if\ fix\left(\frac{f_c}{o.5 * f_s}\right)\ is\ even \\ f_s - rem(f_c, f_s) & if\ fix\left(\frac{f_c}{o.5 * f_s}\right)\ is\ odd \end{cases} \quad \textbf{2. 1}$$

where f_{fold} is the folded frequency, f_s is the sampling frequency, f_c is the carrier frequency, fix(a) is the truncated portion of argument a, and rem(a,b) is the reminder after dividing a by b.

Figure 2-2 and Figure 2-3 show the flowcharts of calculating the appropriate/minimum folding frequency to the FNZ without overlapping for:

a) The single signal with itself.

b) The multi signals with each other or with itself.

The main differences between these two algorithms are:

1- For the single signal it is equal to double the information band, while for multi signals the initial sampling frequency is equal to double the summation of information bandwidths.

2- The last check in the multi signal algorithm is to check if there is any overlapping between folded signals.

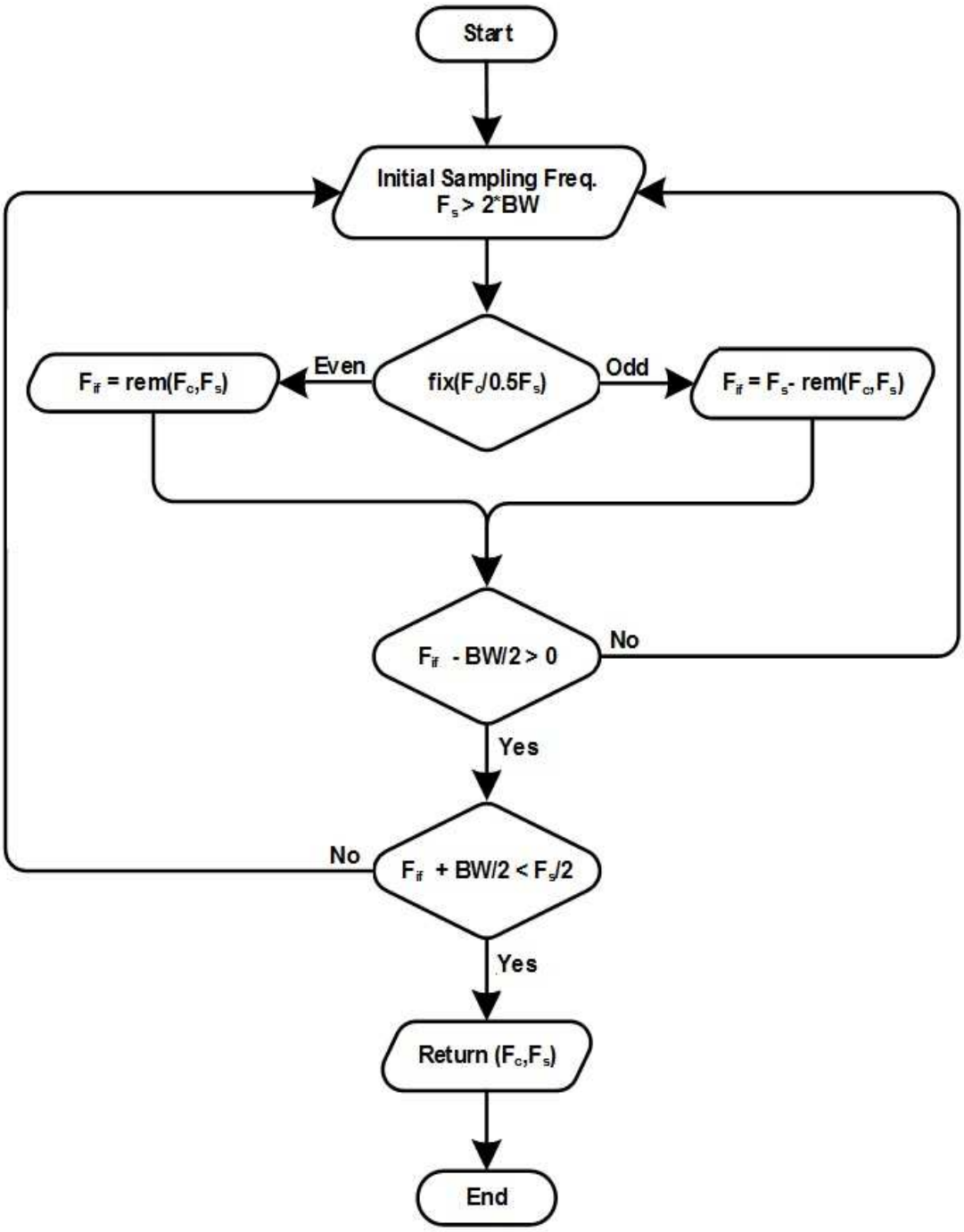


Figure 2-2 Selecting the sampling frequency of a single signal

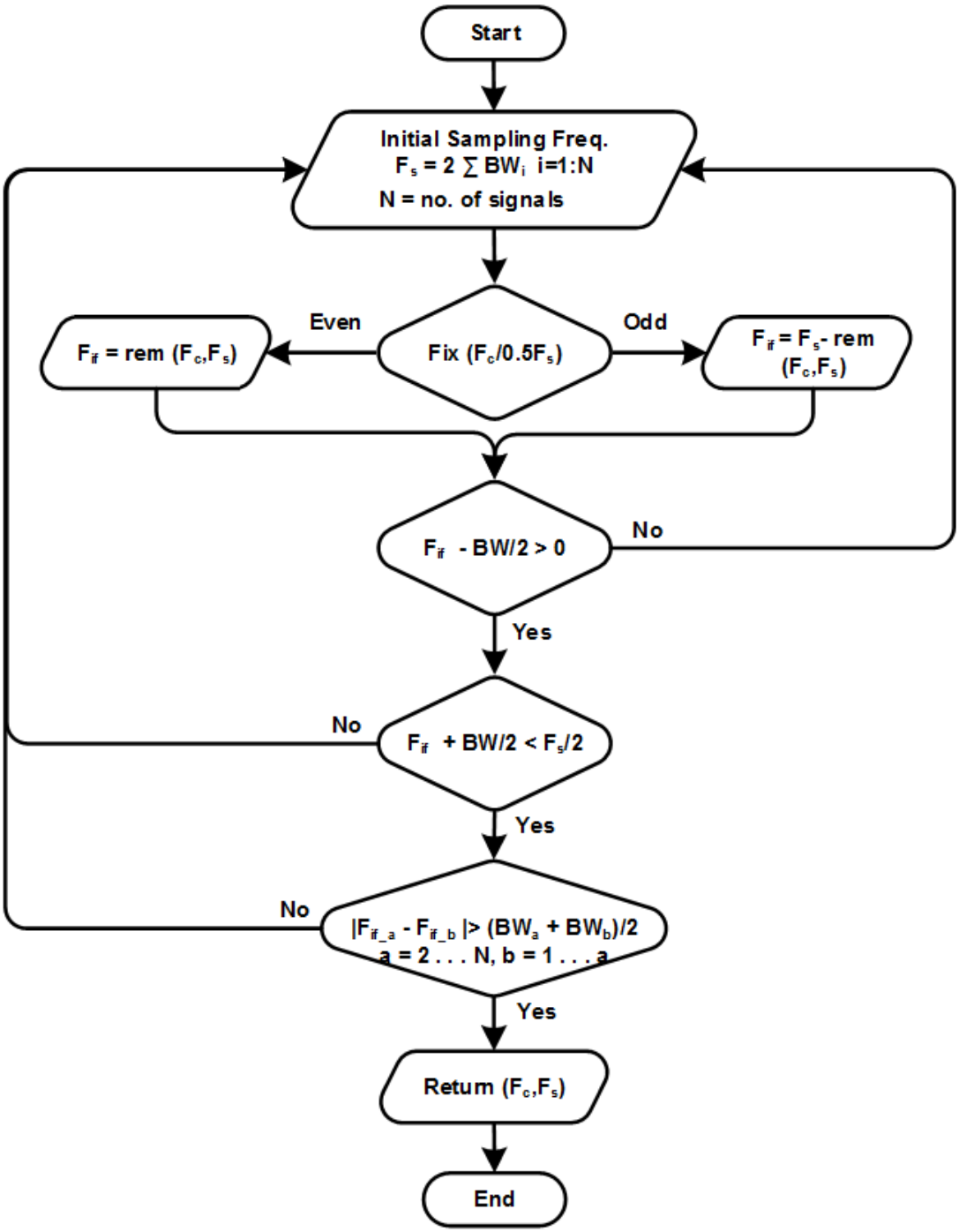


Figure 2-3 Selecting the sampling frequency of multi signals

2.2 Multi-GNSS signals receiver literature survey

Several technical integrations were designed and proposed for the multi-GNSS receiver using single RF front-end and a design for multi-GNSS (GPS-CDMA and GLONASS-FDMA) receiver was finalised [12]. These two signals were received in a single chain, and then fed to the two bandpass filters to isolate their frequencies, as depicted in Figure 2-4. The filter used for the GPS signal is centred on frequency of 1575.42MHz with 3.2MHz bandwidth, while the GLONASS signal's filter was designed to pass 1–12 channels and is therefore it centred at 1605.656MHz with 7.5MHz bandwidth to accommodate the frequencies of these 12 channels. Then the filtered signals were combined and finally sampled by a single ADC at 22MHz sample rate in comparison to the 3.2GHz is required in the traditional sampling. In effect, these signals are ideal for BPS concept based receiver, as their frequencies will not overlap in the FNZ.

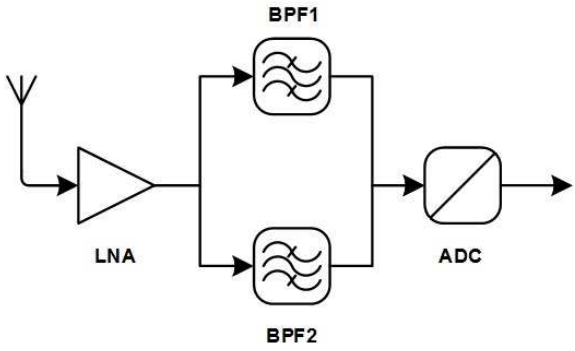


Figure 2-4 BPS receiver of multi-GNSS signals

In the same vein, an L1 (1575.42MHz), L2 (1227.6MHz) and L5 (1176.45MHz) GPS signals are combined in the SDR solutions based BPS receiver and have been successfully implemented in the front-end [13]. Three bandpass filters with 24MHz bandwidth for each filter are used in this designed to filter the signals based on their bands. Then the resultant filtered signals are combined and fed to a single ADC. The minimum sampling frequency is 221MHz, where below this rate there is overlapping between two or three signals' bands. While, 32MHz, 40MHz and 50MHz bandpass filters are used for the Galileo E1 (1575.42MHz), E5 (1191.795MHz) and E6 (1278.75MHZ) signals respectively and the minimum sampling frequency that is found without overlapping between these Galileo signals is 331MHz.

Similarly, GPS L1, L2 and L5 signals and Galileo E1, E5 and E6 signals were combined in a GNSS receiver based SDR solution [14]. A BPS receiver has been exploited to sample all the mentioned bands. The minimum sampling frequency is the key parameter of this implementation, and finding minimum sample rate would reduce the required processing time in the DSP stage. The minimum sample rate in this design is 110MHz to accommodate the information band of the L5-GPS and E5-Galileo signals. Even though, this sample rate is the minimum rate for these bands, it is still wasteful for the L1-GPS signal that needs at least 4MHz sample rate.

In the same way, a multi-GNSS receiver was designed based on BPS receiver to receive and sample GPS and Galileo signals. Three bandpass filters have been used to filter multi-frequencies L1/E1, L2 and L5/E5 in independent channels [15]. The filtered signals are then combined and sampled. The sampling frequency range considered in this design is 158-227MHz. Unlike the previous work, this work focuses on analysing the noise, gain and linearity of the RF components rather than determining minimum sample rate. By contrast, multi-frequency GNSS receiver was proposed to receive GPS and Galileo signals and to reduce the sampling frequency of these GNSS signals, “the L1, L2 and L5 GPS frequencies and E1 and E5 Galileo frequencies” [16]. These signals have been received and passed to multi-bandpass filter and then sampled. This work is based on determining the minimum sampling frequency that satisfies non-overlapping in FNZ for the folded signals, as well as the non-interference errors among these signals. The range of minimum sampling frequency obtained in this work is 111-222.5 MHZ, which is lower than the range of the previous work. In spite of that these two implementations also introduce a complexity to the DSP stage by using high sample rate for the L1/E1 signals.

The drawback in the previous implementations has been overcome in a reconfigurable direct conversion front-end, i.e. choosing sample rate that is proportional to the information band, to handle the GPS (L1, L2 and L5) and the Galileo (E1and E5) signals [17]. This design is able to select these signals based on four operating modes. Switching between these 4-modes is based on changing the rate of the sampling frequency manually, based on the required setup. Modes 1 and 2 handle the GPS signals with 3.125MHz and 6.25MHz sample rate respectively. While, modes 3 and 4 have 12.5MHz and 25MHz sample rate respectively to handle the Galileo signals alone or the GPS+Galileo signals. This solution copes with the

handicap in the previous works by changing the setting/sample rate according to the requirements. Correspondingly, a configurable multi-GNSS receiver was also designed to receive GPS/Galileo/Compass signals in single RF front-end chain. This multi-GNSS receiver can receive L1, E1, B1 (1561.098MHz), B2 (1207.14MHz) and B3 (1268.52MHz) frequencies with filter bandwidth 24MHz to dominate all the bandwidths of these signals. The GNSS signals in this receiver are down-converted to the IF of 46MHz by utilising a reconfigurable local oscillator signal [18].

On the other hand, the L1/L2 GNSS receiver was designed to receive the GPS, GLONASS and Galileo signals using two side-by-side RF front-ends [19]. The first front-end was for L1-GPS/Galileo/GLONASS signals while the second one was for the L2-GPS/GLONASS signals. The signal acquisition, tracking and demodulation are based on SDR implementation, where the FPGA software is used to design a programmable GNSS receiver. Likewise a combining multi-GNSS signal was implemented based on dual software receiver to receive L1 and L2 bands signals, which are GLONASS-FDMA (1602MHz for L1 and 1246MHz for L2) and GPS-CDMA (L1) signals [20]. This implementation was based on using two splitters, i.e. each signal samples and demodulates separately. The use of the frequency splitters makes this design equivalent to using two RF chains process for each GNSS signal. In these two implementations, it can be overcome the side-by-side implementation by employing a BPS technique to handle these multi-GNSS frequency bands.

2.3 Multi-GNSS signal detection setup

The conducted literature review shows that the previous implementations were designed to prevent the overlapping between multi-GNSS signals that transmit at different frequency bands. Actually, these previous works are categorised the signals based on the transmission system, such as GPS group (GPS-L1, GPS-L2 and GPS-L5 signals) or Galileo group (Galileo-E1 and Galileo-E5 signals).

This work is concerned with the GNSS signals that share the same frequency band as well as focuses on detecting multi-GNSS signal by the receiver at an early stage. This will help the DSP to organize the resources according to available GNSS signals only, as shown in Figure 2-5. Therefore, avoid chasing any GNSS signals that do not exist.

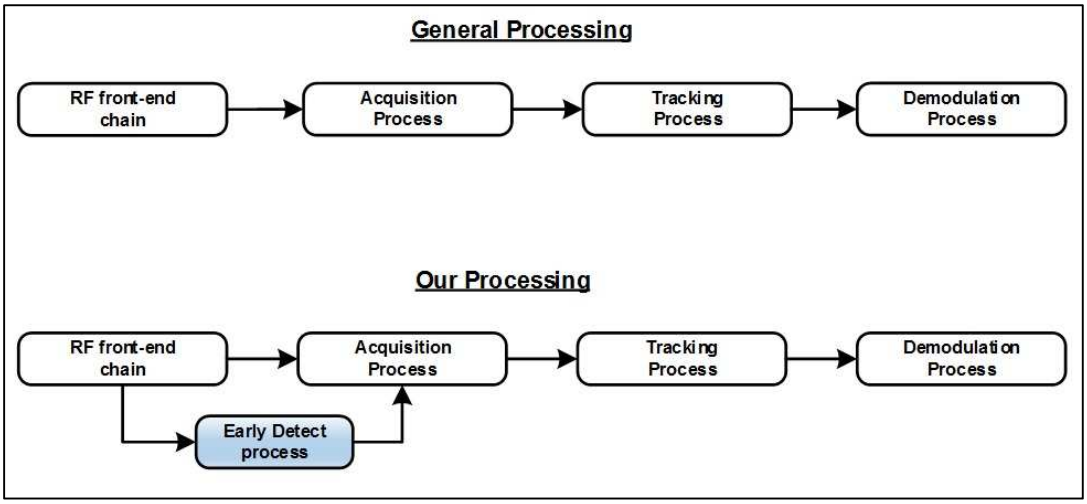


Figure 2-5 Difference between our and general processing

Two quick-early-detection approaches are designed that sense multi-GNSS signals in a single view by measuring the power of all available received signals prior to the acquisition stage and based on the BPS technique. Each approach samples the received GNSS signals at specific sampling frequency and according to the approach setup as follows:

1- 1st Approach: Folding the whole bandwidth of the three GNSS L1- signals: A) the GPS-C/A-BPSK, B) the Galileo-OS-BOC(1,1) and 3) the modernization GLONASS-BOC(2,2) to the FNZ, with isolation between signal frequencies and their harmonics. The appropriate sampling rate chosen is 92.07MHz. This approach is jointly developed with my co-research colleague Mr. Maher Al-Aboodi [5].

2- 2nd Approach: Folding the Galileo and GLONASS BOC signals with the GPS BPSK signal will result in overlapping of these frequencies when excited for a BPSR in the FNZ, with sampling frequency 34.782MHz. This overlapping can be eliminated by filtering out the lower/left-sideband/lobe of the Galileo signal as well as the upper/right-sideband/lobe of the GLONASS signal. Our second approach combines these filtered single-lobe signals with the 3rd harmonic of the GPS signal to avoid overlapping of these signals in FNZ of a BPSR.

The mathematical representation of the GLONASS BOC(2,2) used in this simulation has the same mathematical representation to the formula of the Galileo signal, but

with different values for the bit rate being at 50 bits/s, chipping rate is 2.046 MHz, and the subcarrier frequency at 2.046MHz.

The signal simulations of the approach are implemented by using MATLAB software. Seven scenarios were used to test our approach, and these scenarios are based on satellite transmissions from GPS (C/A-BPSK), Galileo (OS-BOC (1,1)) and GLONASS (BOC (2,2)) using CDMA with a centre frequency of 1575.42 MHz, as shown in Table 2-1.

Note that, in all the following simulation results, the power spectrum density figures are estimated using the Welch algorithm available within MATLAB.

Table 2-1 Scenarios setup

Scenario	GNSS Signals available	CDMA transmission from
1	3	GPS + Galileo + GLONASS
2	2	GPS + GLONASS
3	2	Galileo + GLONASS
4	2	GPS + Galileo
5	1	GLONASS
6	1	Galileo
7	1	GPS

2.3.1 BPSR-Side lobe filtering (BPSR-SLF) approach

1. BPSR-SLF approach setup

The BPSR-SLF approach focuses on detecting the power peaks of all GNSS signals present in the FNZ. This is achieved by removing the overlapping between all the folded GNSS signals in the FNZ so to ensure that the detection of the signals is easier and faster. Therefore, the SSB of the Galileo and the GLONASS BOC signals are used in this approach. The SSB is produced due to using the subcarrier frequency in the BOC modulation. i.e. the subcarrier offers a split the power spectrum of the BOC signal into two symmetrical components around the centre frequency that makes these SSB signals. Furthermore, splitting the DSB into SSBs will remove the effect of

the subcarrier frequency and make each sideband represents a BPSK signal, like the GPS signal [21], or the two SSB can be shifted to the centre frequency, by ± subcarrier frequency, resulting in each sideband of this BOC is like the BPSK signal [22]. Based on that, this approach proposes a way to prevent the overlapping between the chosen GNSS signals. This approach filters out the left-sideband of the Galileo signal and right-sideband of the GLONASS signals, where the reverse of this process is also possible. This filtering must assure the correct choice of the sampling frequency to guarantee there is non-overlapping between these two signals with the 3rd harmonic of the GPS signal. We chose the 3rd GPS harmonic because using GPS signal at the fundamental frequency will produce an overlapping between the three signals (GPS, Galileo, and GLONASS), where the folding frequency will be located at 9.207MHz. However, the power of the 3rd harmonic is lower than the power of the fundamental frequency but it can still be distinguished in the FNZ [23].

The receiver front-end configuration of the BPSR-SLF approach is implemented in MATLAB, as shown in Figure 2-6. The simulated signals are passed through an AWGN channel. The first three BPF are used to obtain right-sideband of the Galileo signal, left-sideband of the GLONASS signal and the GPS signal. Then, the filtered signals are amplified by using an LNA (38dB and 3dB noise figure). A 10-bit ADC converts the amplified signals to their digital form. This configuration uses a sampling frequency of 34.782MHz to ensure non-overlapping between the three GNSS signals in the FNZ, as illustrated in Figure 2-7.

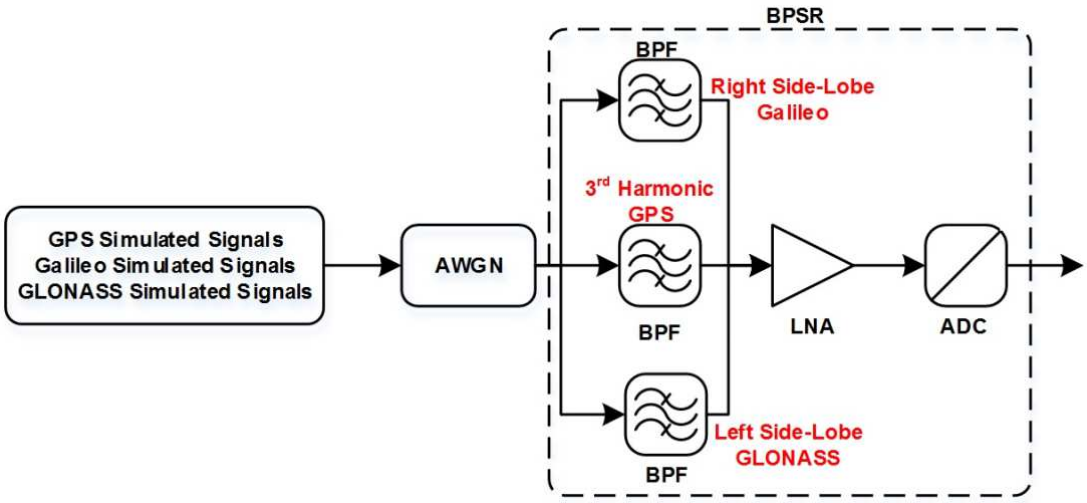


Figure 2-6 BPSR-SLF Multi-GNSS Signals BPS Receiver

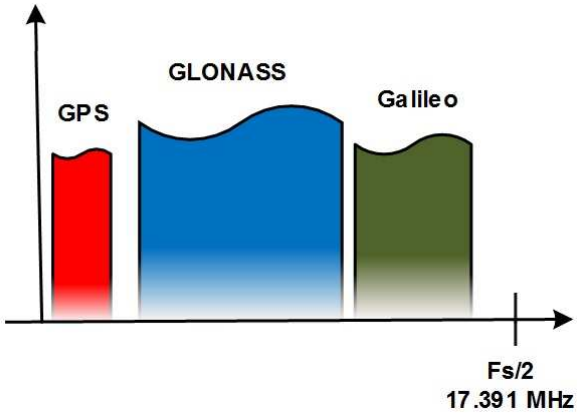


Figure 2-7 BPSR-SLF GNSS folded bands to the FNZ

2. ***BPSR-SLF approach results and discussion***

The seven scenarios, shown above in Table 2-1, are used to test this approach. In these tests, the BPSR will deal with input signals as three distinct GNSS signals i.e. each signal has a separate folded frequency in the FNZ. As shown in Figure 2-8, these signals have three distinct power peaks present in the FNZ, and as follows:

1. The 1st power peak is centred at 4.092MHz (GPS signal) with a bandwidth of 2MHz.
2. The 2nd power peak is at 8.184MHz (GLONASS signal) with a bandwidth of 4MHz.
3. The 3rd power peak is at 11.253MHz (Galileo signal) with a bandwidth of 2MHz.

Also, there is no overlapping between these power peaks and Figure 2-8 proves that three signals are simultaneously excited to our BPSR.

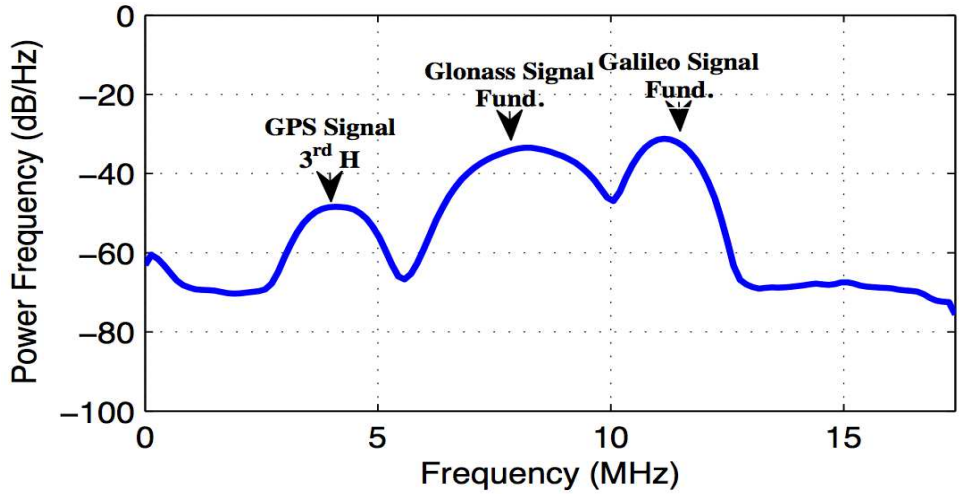


Figure 2-8 Power spectrums of GPS, Galileo and GLONASS signals

The results of the scenarios (2, 3 and 4) from Table 2-1 are illustrated in Figure 2-9, Figure 2-10 and Figure 2-11. In these scenarios the frequency domain proves that there are two separate power peaks existing in the FNZ of any two signals processed by our BPSR.

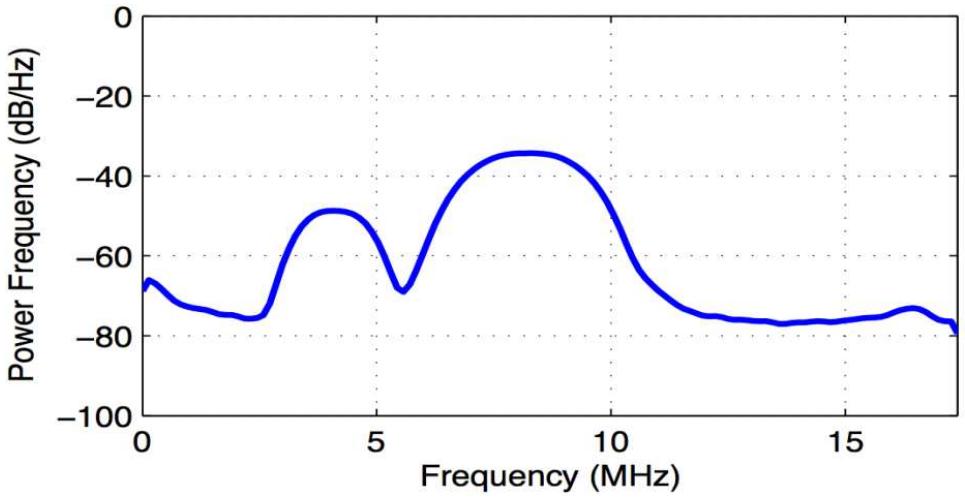


Figure 2-9 Power spectrums of GPS and GLONASS signals

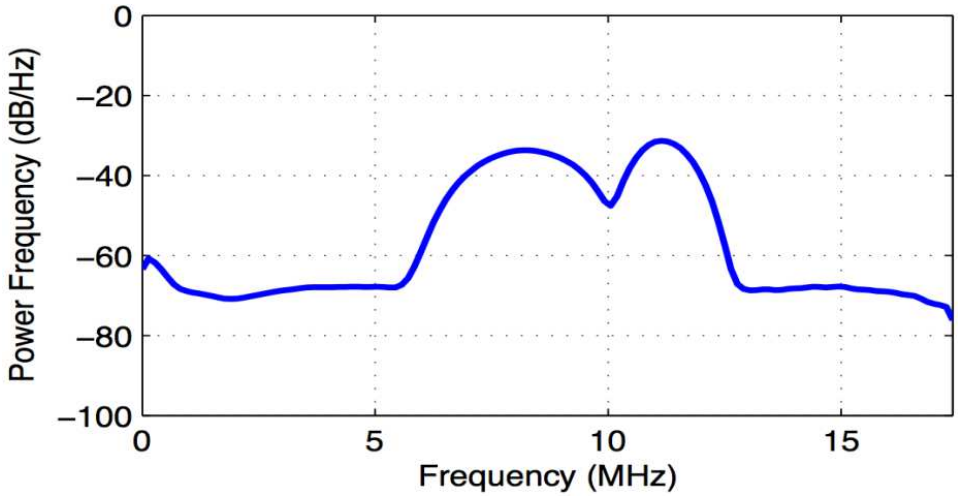


Figure 2-10 Power spectrums of Galileo and GLONASS signals

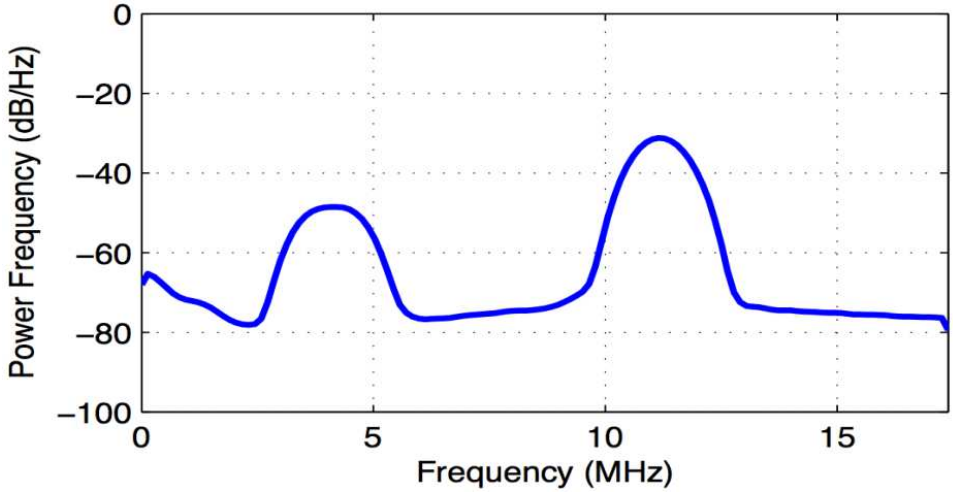


Figure 2-11 Power spectrums of GPS and Galileo signals

The results of the remaining scenarios (5, 6 and 7) are illustrated in Figure 2-12, Figure 2-13 and Figure 2-14. The power distribution of the received signals in these figures proves that there is only single signal power peak present in the FNZ from our BPSR. The position of this power peak determines the type of the received signal since each one has different folded frequency.

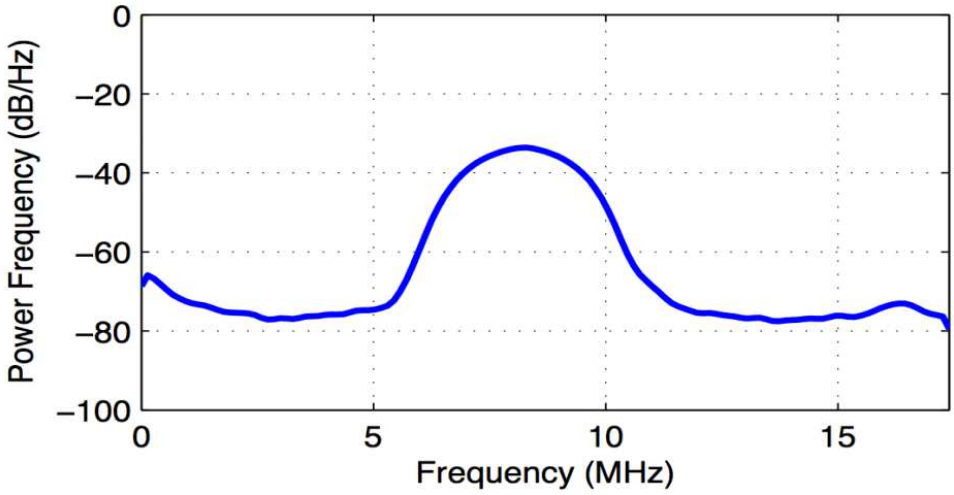


Figure 2-12 Power spectrum of GLONASS signal

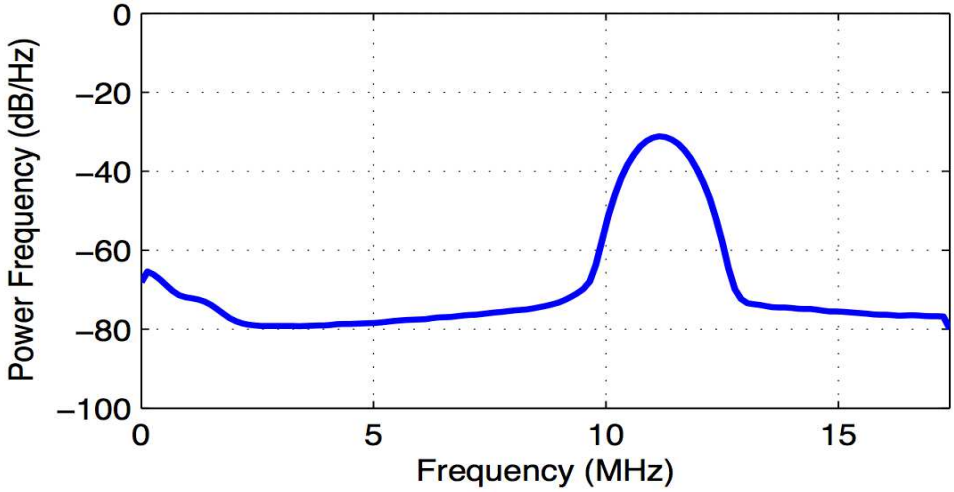


Figure 2-13 Power spectrum of Galileo signal

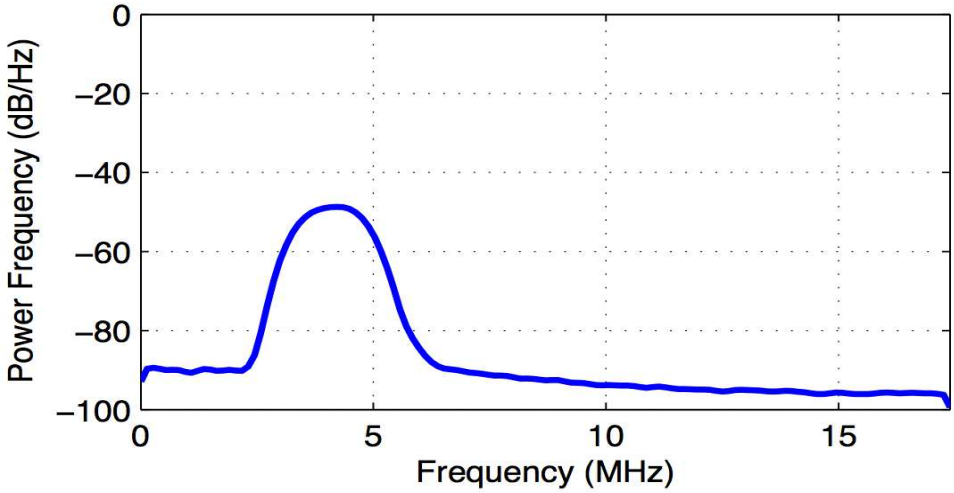


Figure 2-14 Power spectrum of GPS signal

2.4 Concluding remarks on early detection

In this chapter, an early GNSS signals detection was designed, and introduced a rapid early detection of available GNSS signals at the RF front-end. Such implementation filters out the left-sideband of the Galileo signal and the right-sideband of the GLONASS signal. This prevents the overlapping between these two folded signals with the 3rd harmonic of the GPS signal in the FNZ, which easily detects the available GNSS signals. Simulation results show that the proposed approach is a good candidate for GNSS signals detection in the RF front-end. This eliminates the need to search and process signals that are not available at the time, thus saving valuable resources and power.

This work has been presented by my supervisor Dr Ihsan Lami in the Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2012 4th International Congress on Friday, October 5,2012.

Chapter 3

Unambiguous Galileo-OS signal Acquisition

The BOC modulation has been adopted in modern GNSS transmissions such as GPS-M-code and Galileo-OS-code signals. This modulation is designed to help such signal tracking process to mitigate the multipath signal over that of the GPS-C/A code signal. This will therefore enhance the localisation accuracy in a harsh environment. BOC signal is also designed to share an available frequency band with other GNSS signals, like GPS-C/A-code signal and Galileo-OS-code signal.

To generate the BOC modulated signal, the PRN code must be multiplied with a rectangular subcarrier. The resulting BOC signal power spectrum is separated into two symmetric side-lobes placed above and below the centre frequency. For instance, the power spectrum of the BOC Galileo-OS represents two BPSK signals, i.e. two corresponding BPSK GPS-C/A signals, as illustrated in Figure 3-1.

Our research focuses on the BOC Galileo-OS signal. This signal consists of two channels, where each channel has different components. The first one is called “data channel (B)” that comprises the navigation message (D_{E1_B}), data’s primary code (C_{E1_B}) and subcarrier frequency (S_b). The second channel is called “pilot channel (C)” and includes two pilot codes, primary and secondary (C_{E1_c}) and subcarrier frequency (S_c). These channels are then combined and shipped simultaneously at E1 carrier (1575.42MHz), as depicted in Figure 3-2.

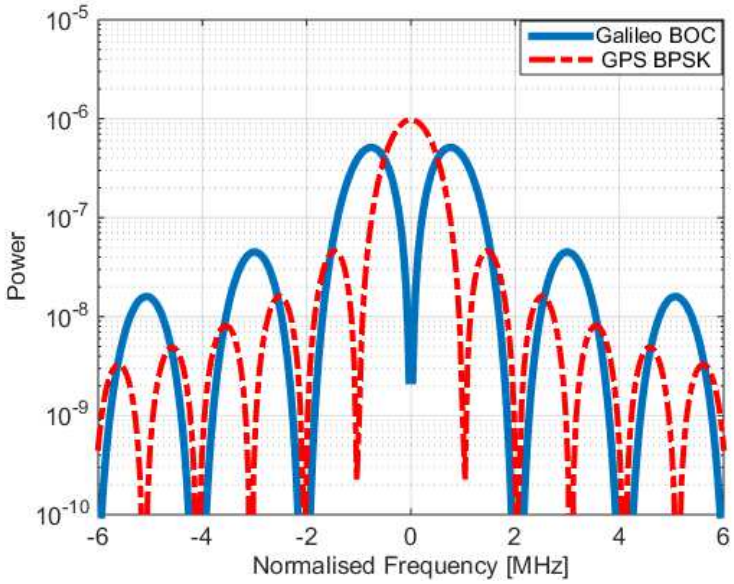


Figure 3-1 Power spectrum of the Galileo-OS and the GPS-C/A signals

The Galileo-OS signal uses the CBOC modulation, which means multi-level spreading symbols formed from the weighted sum of BOC(1,1) and BOC(6,1) as expressed by the following equations [24].

$$g_{BOC(1,1)} = \begin{cases} sign\left(sin\left(\frac{2\pi t}{T_c}\right)\right) & 0 \leq t \geq T_c \\ 0 & elsewhere \end{cases} \quad \textbf{3. 1}$$

and

$$g_{BOC(6,1)} = \begin{cases} sign\left(sin\left(\frac{12\pi t}{T_c}\right)\right) & 0 \leq t \geq T_c \\ 0 & elsewhere \end{cases} \quad \textbf{3. 2}$$

where $g_{BOC(1,1)}$ represents the BOC(1,1) spreading symbols, $g_{BOC(6,1)}$ represents the BOC(6,1) spreading symbols, and T_c is the code chip duration.

Note that, the typical notation for BOC modulation is BOC(m,n), where (m) represents the ratio of the subcarrier frequency (f_{sc}) to 1.023MHz and (n) represents the ratio of the chipping rate (f_c) to 1.023MHz. For example, a BOC(1,1) means both the subcarrier frequency and the chipping rate are equal to 1.023MHz.

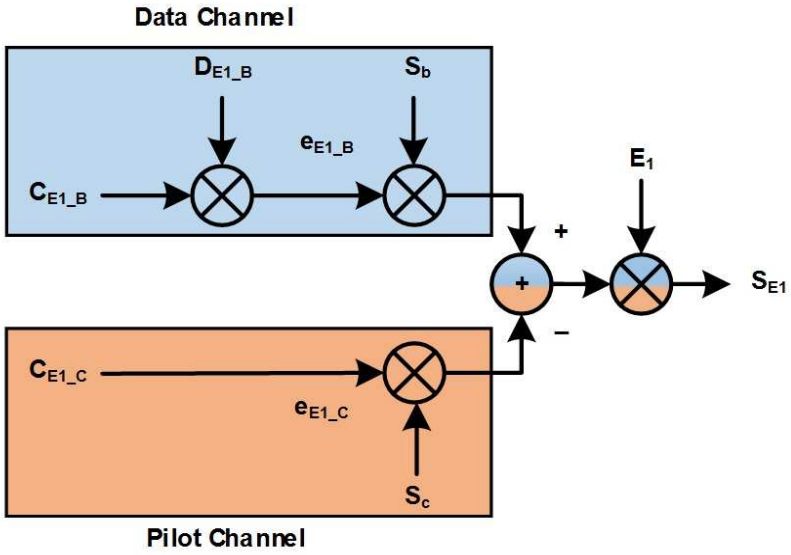


Figure 3-2 Modulation Scheme for the Galileo-OS Signal

These BOCs modulations types are used for both the data channel (S_b) and for the pilot channel (S_c) as expressed in equations (3.3) and (3.4) respectively.

$$S_b = \alpha\, g_{BOC(1,1)} + \beta\ g_{BOC(6,1)} \qquad \textbf{3. 3}$$

$$S_c = \alpha\, g_{BOC(1,1)} - \beta\ g_{BOC(6,1)} \qquad \textbf{3. 4}$$

where α and β are the power parameters to control the combined power of the data and pilot channels and are equal to $\sqrt{10/11}$, $\sqrt{1/11}$ respectively.

The mathematical representation of the transmitted CBOC signal is shown in equation (3.5) and as described in the Galileo Signal-In-Space Interface Control Document (SIS-ICD) [25].

$$X_{E1} = \frac{1}{\sqrt{2}}\left[e_{E1\_B}(t)S_b - e_{E1\_C}(t)S_c\right]\cos(2\pi f_{E1}t) \qquad \textbf{3. 5}$$

where X_{E1} represents the CBOC Galileo E1 signal, the e_{E1_B} and e_{E1_C} are the binary signal components (the navigation message and primary/secondary codes) and f_{E1} is the carrier frequency of the E1 signal at 1.57542GHz.

The CBOC Galileo signal received at the user end is therefore represented as:

$$X_r = A\left[e_{E1\_B}(nT_s - \tau)S_b - e_{E1\_C}(nT_s - \tau)S_c\right] e^{j(2\pi(f_{E1}+f_d)nT_s)} + n_o(nT_s) \quad \textbf{3. 6}$$

where $X_r, A, n, T_s, \tau, f_d$ and n_o represent the received signal, the amplitude of the received signal, number of samples, sampling period, code phase delay, Doppler frequency shift and additive white Gaussian noise respectively.

I have studied this signal in terms of how it is modulated, received at RF front-end, acquired and tracked. My main interest area is the acquisition stage, because understanding the boundaries of the acquisition process and its requirement might enable combining this signal with other GNSS signals, such as GPS-C\A signal.

According to the literature on this particular signal, I have found that the acquisition process comprises two technical aspects:

1. Acquisition becomes ambiguous when the code phase resolution is equal to 0.5 Chip or higher (see Section 3.1). The previous solutions overcome the acquisition ambiguity but at the expense of a complex implementation and/or degrade the power of the received signal by around 3dB (see Section 3.2). To tackle these limitations we have designed a new unambiguous method called "Enhanced Subcarrier Elimination Conversion (ESCE)" to acquire Galileo-OS signal. Our ESCE method eliminates the subcarrier frequency to overcome the acquisition ambiguity.

2. The Galileo signal consists of two channels that comprise data and pilot signals, as illustrated in Figure 3-2, ignoring one of these channels leads to a 3dB power loss that is important when acquiring this signal in multipath environments, such as urban areas. The time-domain solution requires 4-correlation channels to acquire this signal, while the frequency-domain solution needs 2-correlation channels. Chapter four will describe how we overcome this complex implementation by designing an orthogonal joining method that requires only single correlation chain to combine these signals.

3.1 Galileo-OS signal correlation process and ambiguity condition

As described in the previous section, the power spectrum of the resulting BOC Galileo-OS signal is separated around the centre frequency. This separation consequently appears at the receiver end when correlating the received BOC Galileo signal with the generated BOC Galileo signal. As depicted in Figure 3-3, the result in correlation domain will have additional undesired side-peaks besides the main peak that we want. The width of each peak is designed to be equivalent to one-third of the GPS-C/A signal's peak; in order to enhance the signal tracking accuracy [26], i.e. the range of measurement error would be reduced by factor equal to 3.

Note that, the CBOC signals can be processed either with a CBOC generated signal or with a BOC (1,1) generated signal [27].

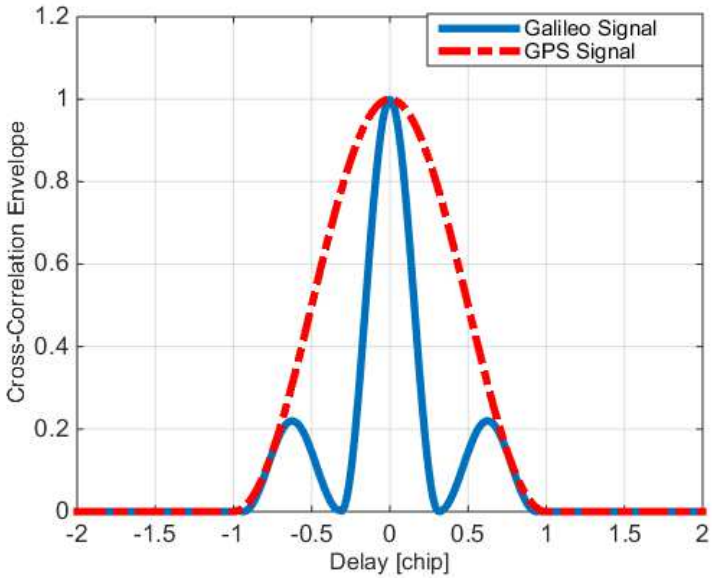


Figure 3-3 The cross-correlation function of the Galileo-BOC and the GPS-BPSK signals

On the other hand, the BOC designers have pointed out that the CCF becomes ambiguous when the received signal is correlated with the reference BOC signal at code phase resolutions of 0.5 Chip or higher. Where, with fewer generated peaks at this resolution, the acquisition process may lock to the wrong peaks or miss the signal detection when the correlated peak (the false peak) does not exceed a certain threshold [28], as depicted in Figure 3-4. Therefore, the receiver should ensure that

the right peak is acquired to reduce the synchronisation time with the received signal in the tracking process.

To overcome this ambiguous correlation problem, the most commonly used solutions recommend that:

1. The code's phase resolution must be divided by 3 to achieve the same correlation results as normally obtained with the GPS-C/A signal, which means more processing time is required.
2. The BOC signal is processed as two BPSK signals. As a result, this approach requires double processing, and that leads to complicating the acquisition implementation.

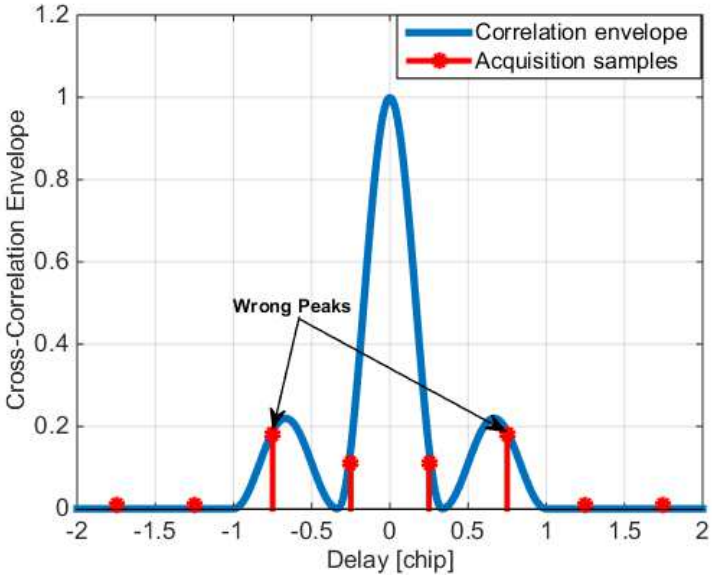


Figure 3-4 Ambiguity problem when chip resolution ≥0.5 Chip

3.2 Common acquisition algorithms for the GNSS signals

Before reviewing the unambiguous methods, this section will give a brief discretion of the commonly used algorithms to acquire the GNSS signals.

Generally, in order to know the presence of the GNSS signals, the signal acquisition must be used to determine the number of visible satellites with respect to the position of the GNSS receiver [29]. A conventional GNSS receiver generates a replica PRN code and carrier frequency within a range of Doppler frequencies (±4KHz) to acquire

the GNSS signal [30]. Signal detection is successful when the generated PRN codes are aligned with received code in the incoming signal, and the locally generated frequency matches the frequency of the received signal.

There are different algorithms to acquire the GNSS signals, such as serial search, parallel frequency search, parallel code phase, matched filter search and differential algorithm. The following sections give a brief explanation of the commonly used algorithms to acquire the GNSS signals.

I. Serial search algorithm

The serial search algorithm relies on a hardware implementation. As shown in Figure 3-5, the received GNSS signal multiplies with the replica PRN code sequence for a particular satellite. Then the output multiplies with the locally generated carriers, which are the in-phase and the quadrature-phase carriers [31]. Both channels are integrated according to the code length of received signal. Finally, each branch is squared separately then they are added together at the end of the acquisition process. The detection is successful when the output exceeds a certain threshold.

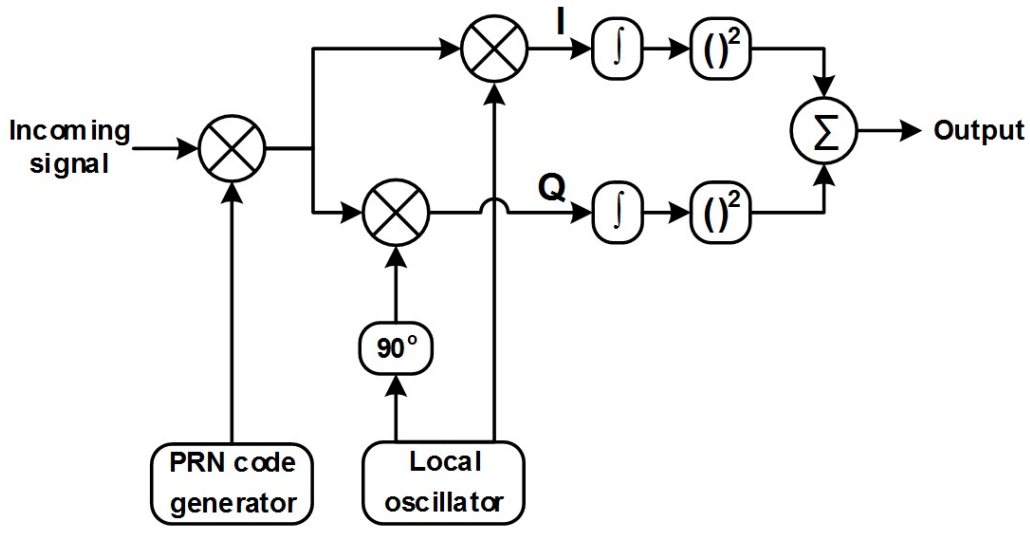


Figure 3-5 Block diagram of the serial search algorithm

II. *Parallel code phase search (FFT-search) algorithm*

The FFT search algorithm is accomplished in a software implementation and the correlation is based on the frequency domain to simplify and reduce the number of combinations [32]. As shown in Figure 3-6, the received GNSS signal multiplies with locally the in-phase (I) and the quadrature-phase (Q) generated carriers. The I and the Q outputs are then combined and transformed to the frequency domain using FFT transform. A replica PRN code sequence for specific satellite is generated, then it is transformed to the frequency domain using FFT transform, and the transformed PRN code is complex conjugated to be multiplied with the transformed signal [33]. The result is inversed to the time domain using IFFT transform, and the absolute output is squared to be compared with a certain threshold.

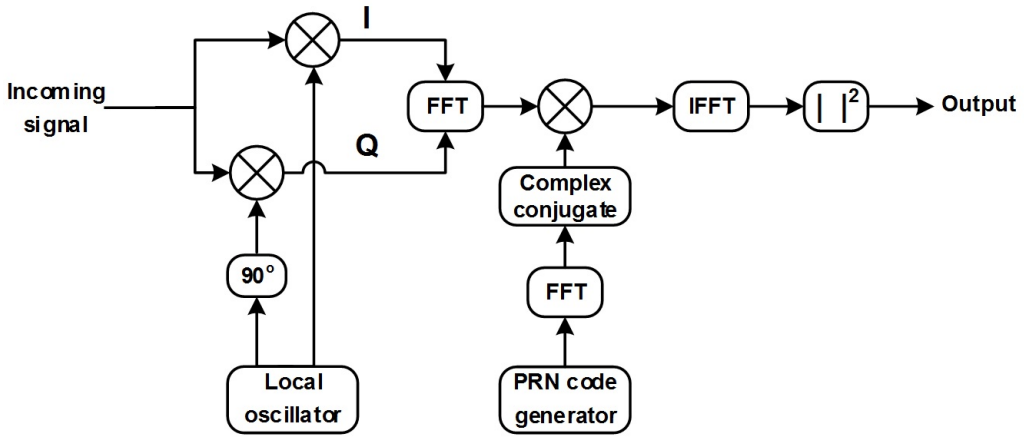


Figure 3-6 Block diagram of parallel code phase (FFT) search algorithm

III. *Matched filter search algorithm*

Matched filter algorithm is more commonly used in the GPS receiver. The received signal firstly gets rid of the carrier frequency by multiplying with a complex local signal with specific Doppler shift, as shown below in Figure 3-7. Then the samples fill the shift register to be correlated with local PRN code that accumulates in the buffer of the matched filter. The correlation output is then compared with a certain threshold to decide whether the signal is acquired or not [34]. The acquisition accuracy based matched filter depends on the space cell in the shift register, which is most often equal to 0.5 chip search step.

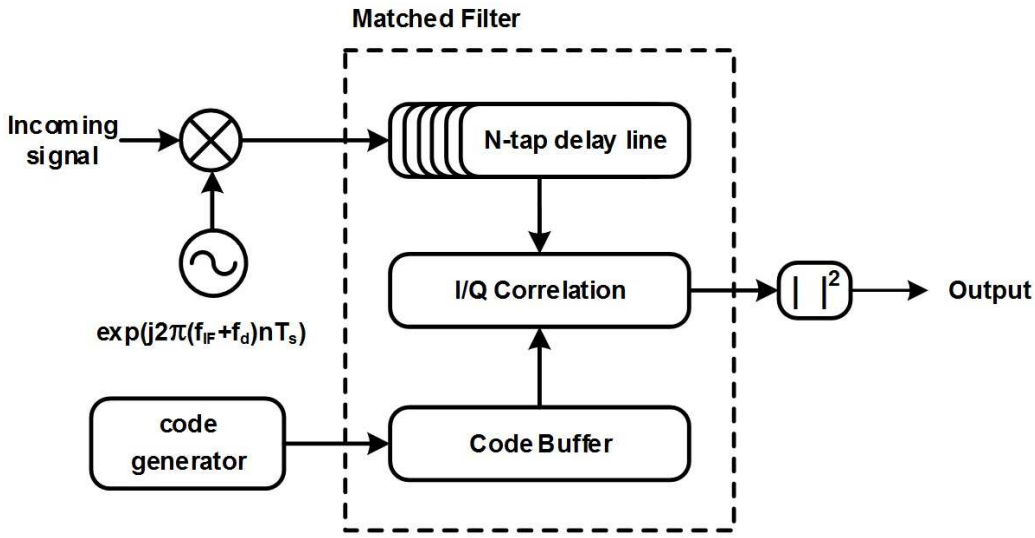


Figure 3-7 Matched filter algorithm

3.3 Related unambiguous contributions literature survey

We have studied the research literature for resolving the ambiguity issue of the BOC signals, focusing on recent and most widely used methods. It is clear that the authors who formulated the BOC specification have actually described this ambiguity clearly and pointed out possible solutions. The most obvious solution is to use chip resolution of less than 0.5 Chip.

In an application attempt to resolve the ambiguity of the BOC(10,5)-GPS-M code signal, the “dual sideband (DSB) method” was developed to acquire this signal as two BPSK signals [21]. As illustrated in Figure 3-8, this method is based on using two filters to filter the upper and the lower sidebands. Thus, each sideband is now repressing a BPSK(5) signal approximately, where number five refers to the chipping rate $(5 * 1.023\text{MHz})$. The acquisition is then accomplished through two distinct correlation channels for the upper and lower sidebands. Each of these channels correlates the filtered received signal with a filtered BOC generated signal, where the generated BOC signal is constructed by multiplying the replica PRN-code with the subcarrier frequency. Finally, after summing the outputs of these channels, the shape of the result CCF is approximately like the shape of the BPSK CCF.

However, this method suffers from the undesirable noise that is introduced by using these filters in the beginning and inside during the acquisition process. For the sake of

comparison, I have implemented this method to highlight the requirements of this method in terms of the processing time and the computational complexity.

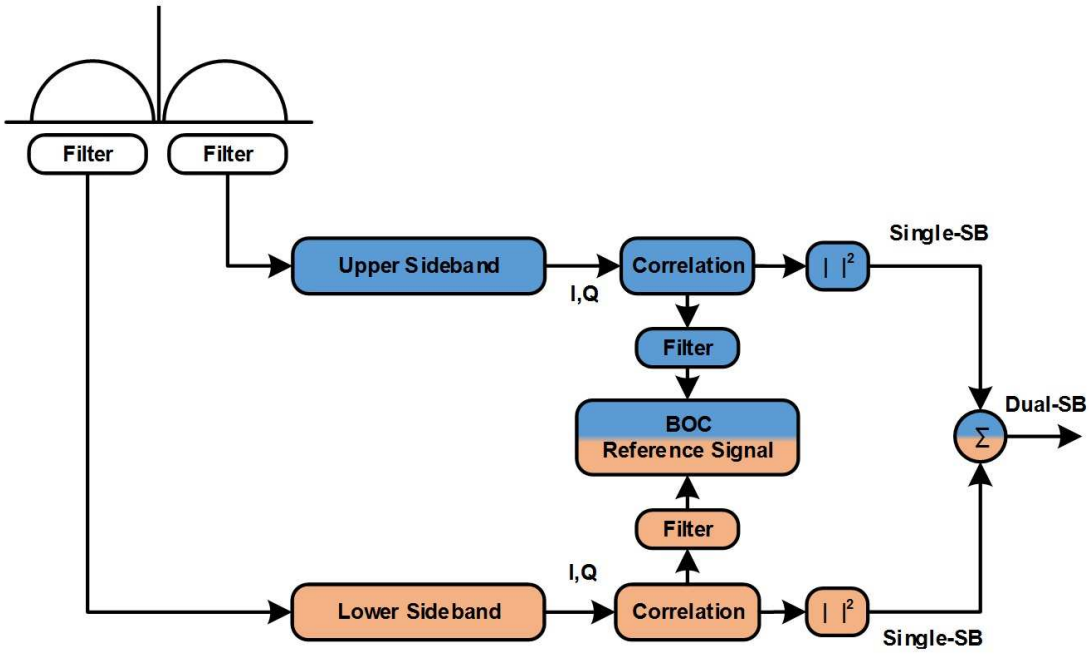


Figure 3-8 Dual sideband method

Similarly, an unambiguous method called "BPSK-Like" was designed to reduce the complexity and to eliminate the effect of the noise caused by using filters in the DSB method [22]. This is achieved by using a single filter rather than two filters, where the bandwidth of this filter accommodates both of these upper and lower sideband signals. As shown in Figure 3-9, these sidebands are then shifted to the centre frequency by the amount of the subcarrier frequency ($\mp f_{sc}$). Then the shifted signals are correlated in two parallel channels with the generated BPSK-modulated code, i.e. only the code without the subcarrier frequency.

However, this method only works with an even BOC modulation order (N_{BOC}, where $N_{BOC} = 2f_{sc}/f_c$). In addition, the use of a single-sideband signal results in 3dB degradation in the SNR of the received signal, but if these sideband correlations are summed then the loss can be partially compensated.

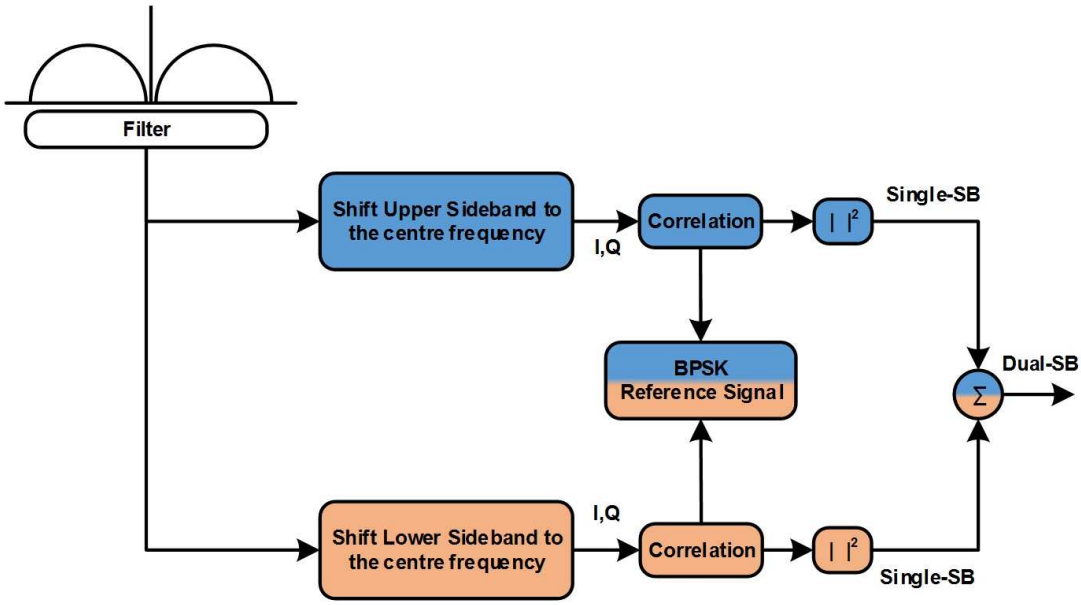


Figure 3-9 BPSK-Like method

To overcome the limitation of the BPSK-Like method, additional conditions are introduced to the BPSK-Like method. Hence, these conditions make the BPSK-Like method work for both even and odd BOC modulation orders [35]. This distinction is realized by determining the amount of the shifting frequency (δf_{sc}), where δ depends on the BOC modulation order and equals to:

$$\delta = \begin{cases} 1 & \text{if } N_{BOC} \text{ even, sin and cos} - \text{BOC} \\ \dfrac{N_{BOC}-1}{N_{BOC}} & \text{if } N_{BOC} \text{ odd, sinBOC} \\ \dfrac{N_{BOC}+1}{N_{BOC}} & \text{if } N_{BOC} \text{ odd, cosBOC} \end{cases}$$

To reduce the complexity of the above DSB and BPSK-Like methods, three proposals have been designed for "Low Complexity (LoCo)" implementations [36]. The concept behind all of these three proposals is based on shifting the received BOC signal to the zero frequency and then generating a BPSK-PRN replica code. What distinguishes the different proposals is the way of using the filters. In the first proposal (modified-DSB), the numbers of filters are reduced by generating BPSK-PRN code rather than generating filtered BOC-PRN code. The second proposal (modified-BPSK-Like) introduces extra filters to overcome the BOC modulation order. The third proposal does not use any filtering. Note that, the number of filters used in both modified-DSB and modified-BPSK-Like are equal for the dual or the

single sideband; also their computational complexities are same and less than the DSB and BPSK-Like methods. Nevertheless, these proposals do not reach the same performance as the DSB method, albeit they reduce the implementation complexity.

Equally, a cyclically shift-and-combine unambiguous acquisition method was designed based on the BPSK-Like method [37]. The concept of this method is to reduce the length of the correlated code via cycling the generated code L times then combining the shifted codes and finally dividing the combined codes into L sections, where the L parameter should be a common divisor of the Galileo length code. The idea behind it is to reduce the dimension of the correlation by a factor equal to L if the signal is detected in the first shift. On the other hand, the process of cycling and combining the code would decrease the level of orthogonality between the Galileo codes because the full code for each satellite is designed to be orthogonal with other satellites full codes or with the same code if the code phase shift is more than 1 Chip.

As is obvious, the previous methods depend on acquiring the BOC signal as a BPSK signal through an early pre-processing. While, the “subcarrier phase cancelation (SCPC)” method was designed to remove the subcarrier frequency effect from the BOC signal in the acquisition process [38]. Figure 3-10 shows the block diagram of SCPC implementation, the received BOC signal multiplied by the in-phase and quadrature-phase carrier frequency to get rid of the Doppler frequency shift. Then the outputs are correlated with the local BOC signal-in-phase subcarrier and the local BOC signal-quadrature-phase subcarrier. Unambiguous CCF can be then obtained when all these in-phases and the quadrature-phases correlation channels are summed, which is same as the CCF of the BPSK signal.

In contrast with the above DSB, BPSK-Like and LoCo methods, this method does not depend on filtering process to correlate the single or double sidebands, but it does cost more correlation channels, i.e. duplicates the numbers of required correlation channels (the in-phases & quadrature-phases of the carrier frequency and the subcarrier frequencies). In addition, the performance of this method does not reach the performance of the BPSK-Like method.

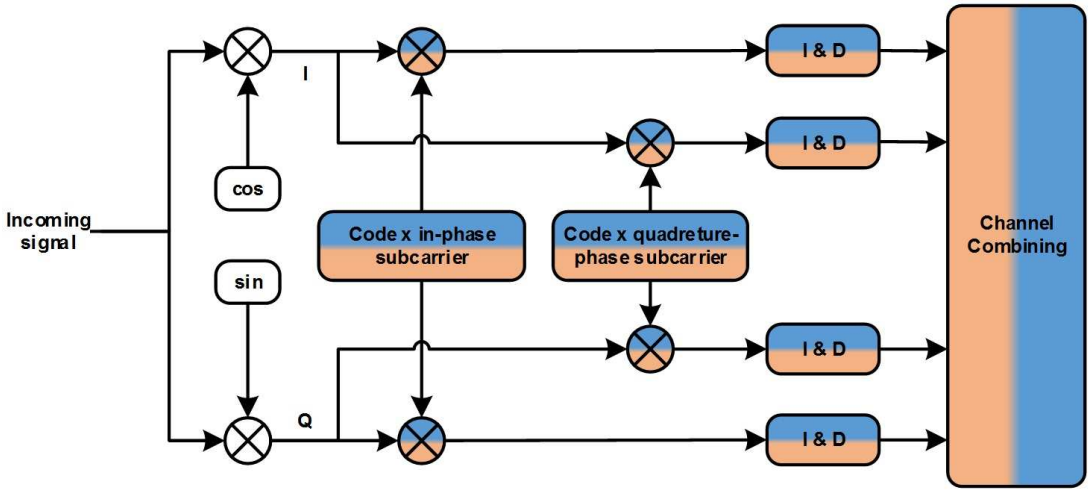


Figure 3-10 SCPC method

An analysis was conducted to evaluate the performance of the BPSK-Like methods, and the SCPC method [39]. The evaluation was based on using same filter bandwidth and the same scenarios. The BOC signals used in this assessment are BOC(1,1), BOC(10,5) and BOC(15,2.5) signals. The PRN code is set to 1023 Chip for all these signals and three integration times have been used which are 5, 10 and 20ms. The results in the aforementioned analysis show that the BPSK-like method has performances better than the SCPC method over all these signals and as follows:

1. The BOC(1,1) signal performance : BPSK-like method 1dB > than SCPC method.
2. The BOC(10,5) signal performance: BPSK-like method 0.5dB >than SCPC method.
3. The BOC(15,2.5) signal performance: BPSK-like method ≈ SCPC method.

To construct CCF equivalent to the BPSK's CCF, the side-peaks effects have been removed by using multi-stages of matched filter process [40]. The process of this solution started by correlating the received BOC signal with the PRN code using matched filter to produce BOC-CCF (three peaks). Then the BOC-CCF is squared; this makes the main peak, i.e. the peak in the middle, centred at the zero frequency and the two unwanted replicas are far from the main lobe by $\mp 2f_{sc}$. Then the squared outputs are passed through another filter that composes three nonzero samples. This stage will again generate seven nonzero samples, where the main sample/peak is also centred at zero frequency, and on each side (right and left) there are three samples.

These seven nonzero samples construct the unambiguous CCF that is equivalent to the BPSK's CCF.

Similarly, a side-peak cancelation method was suggested to eliminate the subcarrier effect [41]. The method is based on correlating the received BOC signal with the divided generated subcarrier signals in multi-stages. In each stage the received signal is correlated with partial subcarrier signals. Note that, each chip is represented by the combined data and pilot subcarrier frequencies. The number of stages is equal to (v-1) where "v" represents the ratio of the second subcarrier frequency to the 1.023MHz, i.e. in case of Galileo-OS signal there are 5 stages to eliminate the subcarrier frequency effect. The unambiguous CCF was achieved when the whole correlations between received and the divided generated signals are combined. Nevertheless, this method requires more correlation due to a number of correlation stages that will increase if the "v" value increased. In a different manner, side peaks were cancelled by combining the correlation of two formats [42]. Practically, after removing the carrier frequency, this method takes the summations of the correlations between the received BOC signal with the generated BOC signal (PRN code x subcarrier) and the generated PRN code (non-subcarrier). Then by taking the absolute value of the above correlation this will eliminate the side peaks. The result of this process creates sharper CCF than the GPS's CCF. For the sake of comparison between this method and the SCPC method, both methods have a same number of correlation channels but this method needs more mixers than the SCPC method since it generates the subcarrier and non-subcarrier signals.

3.4 Methodology of the ESCE method

As detailed in my literature survey (Section 3.3), the previous methods can be divided into two groups. The first group deals with the BOC signal as a BPSK signal via filtering or shifting processes such as DSB and LoCo methods. While, the second group focuses on removing the subcarrier frequency inside the acquisition process like SCPC method.

This section describes the methodology of our new ESCE method that eliminates the subcarrier frequency for the BOC Galileo-OS signal by using the whole subcarrier frequency. ESCE process tackles three technical aspects and as follows:

I. Resolving the ambiguity

As shown in Figure 3-11, the process of the ESCE method starts by multiplying the received Galileo signal by either the generated data channel's subcarrier (3.3) or by the generated pilot channel's subcarrier (3.4). This process will convert the BOC modulation to a BPSK like modulation, because we partially remove the subcarrier frequency effect from the received data and the pilot Galileo signal. Practically, the subcarrier removing process is equivalent to shifting of the two side lobes to the centre frequency simultaneously. As a result, this process shapes the envelope of the CCF from multiple peaks to a single peak, as illustrated in Section (3.5.1).

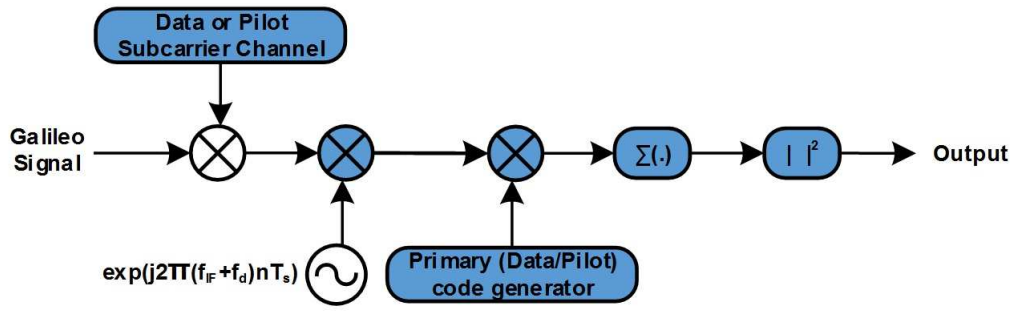


Figure 3-11 The ESCE method

Therefore, the processing requirements of the converted signal are same as the BPSK signal's requirements, such as generating the carrier frequency and the PRN-code only without the necessity of generating a subcarrier frequency.

Note that, the required code in this implementation is therefore either the primary code of the data channel if the generated data channel's subcarrier is employed to eliminate the subcarrier frequency effect, or the primary code of the pilot channel if the generated pilot channel's subcarrier is used.

II. Enhancing the power of the received Galileo signal

The conducted literature review showed that the current unambiguous methods, such as BPSK-Like and LoCo methods suffer from 3dB power degradation when the shifting process applies to the received Galileo signal, as shown in Figure 3-12. While, the other method that is based on filtering like DSB method might lose 3dB if a single sideband is used to acquire the Galileo signal.

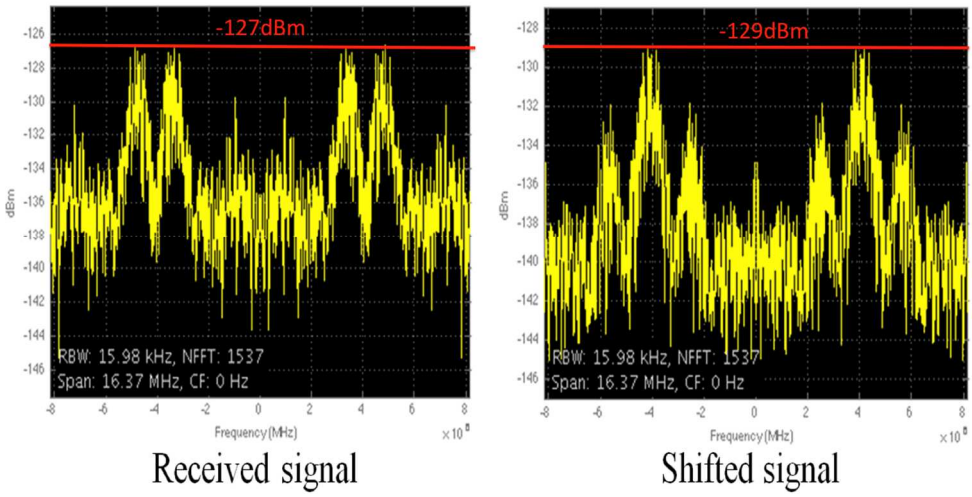


Figure 3-12 Received vs. shifted Galileo signal

Therefore, to overcome the 3dB loss the two sidebands should be used to compensate this power degradation. However, the overhead is a more complex implementation because it needs two distinct correlation channels to handle each sideband separately. In contrast, we have found that multiplying the subcarrier frequency with the received Galileo-BOC signal not only removes the subcarrier frequency effect but it enhances the power of the received Galileo signal because the powers of the two shifted sidebands are also added together. Therefore, the resulting output signal gained is at least 2dB, compared to the actual received signal, as depicted in Figure 3-13. To illustrate that, let us assume the normalised power of the left sideband equals to (1pows) and the normalised power of right sideband equals (1pows) then the gain of the combined sidebands equal to 3dB (G =10log10(2pows)). Consequently, this 2dB power improvement will directly enhance the probability of the signal detection.

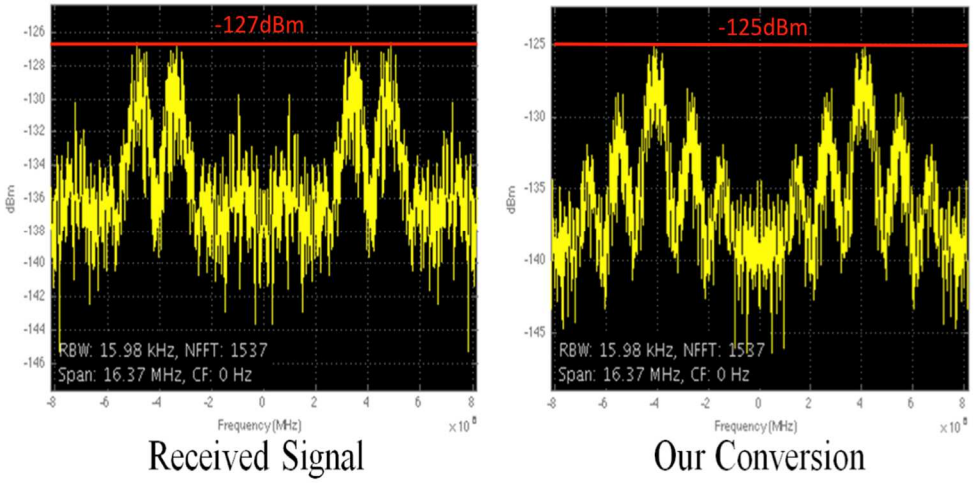


Figure 3-13 Received vs. our conversion Galileo signal

III. Enabling GPS receiver to acquire Galileo signal

The GPS-CA code signal and Galileo-OS code signals share the same centre frequency and they have the same chipping rate but enabling the GPS receiver to acquire the Galileo signal requires an additional resource. For example, in time-domain implementation (unambiguous solution) the requirement is another correlation channel to process the other sideband signal. While, in frequency domain implementation the requirements are a subcarrier frequency generation for the Galileo signal as well as another correlation channel if the implementation does the joining between the data and the pilot channels.

Our ESCE method enables the GPS receiver to acquire the Galileo signal and reduce the overhead because the subcarrier frequency elimination is achieved in an early stage before the acquisition process. This elimination makes the existing resources of the GPS receiver to be used for both signals, as depicted in Figure 3-14, which means only single correlation channel is required because:

A. In time-domain implementation, same frequency generation and the same buffer can be used.

B. In frequency-domain implementation, same frequency generation and same transformation can be used.

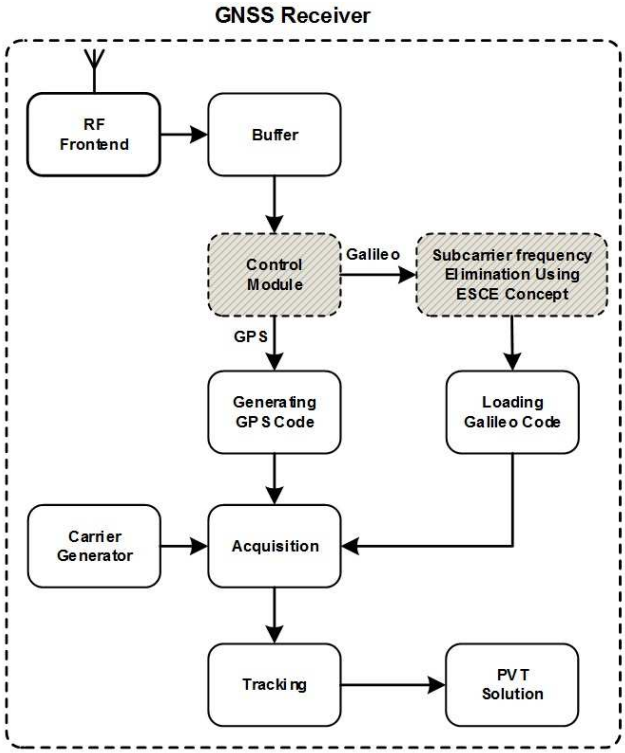


Figure 3-14 Enabling GPS receiver to acquire Galileo signals

3.5 ESCE experimental setup

It was important to devise realistic signal environment, where ESCE can be tested with multipath and harsh scenarios. The ESCE experiments were based on using HaLo-430 platform that were performed within the Short-Term Scientific Mission (STSM) at Ghent University during November 2014. In these experiments, two HaLo-430 platforms were used, as shown in Figure 3-15, to transmit and receive Galileo signals in the real communication channel.

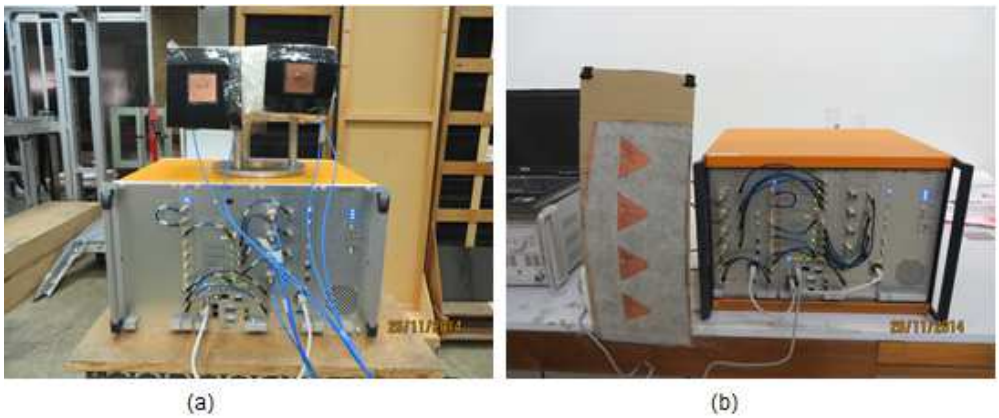


Figure 3-15 HaLo-430 platform (a) Transmitter (b) Receiver

This wireless testbed is fully controlled by MATLAB and up to four signals can be transmitted and received simultaneously, but the transmission signals should be at the same carrier frequency.

The transmitting process is as follows:

1. Generate Galileo baseband signals using MATLAB. For each Galileo baseband signal, navigation data and secondary pilot code are spread in two separate channels using two codes (primary data code and primary pilot code) and two subcarrier frequencies. Then the two channels are combined to construct CBOC baseband signal, as shown in Figure 3-16.
2. After that, the baseband signals are uploaded to the HaLo-430 platform from the PC via USB.
3. The uploaded baseband signals are then converted to analogue using DAC and the signals are transmitted repetitively by RF-frontend on the selected transmission band.

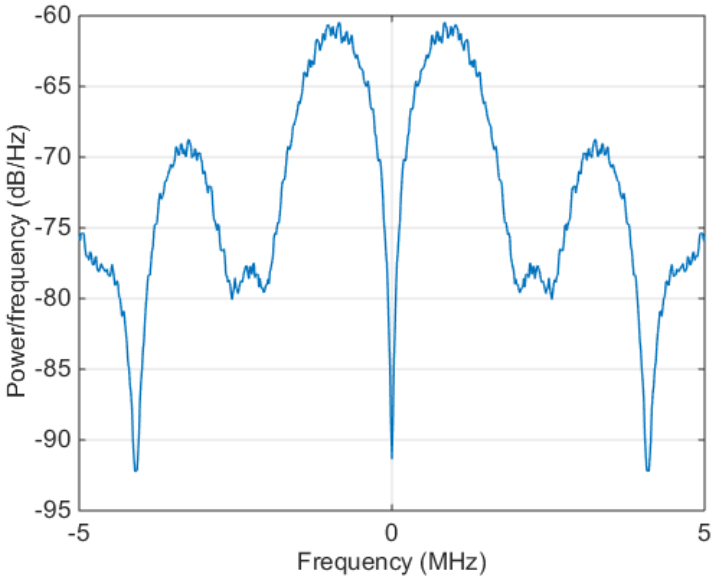


Figure 3-16 Uploaded Baseband Galileo-OS signal

In each scenario, we used all the four channels of the HaLo-430 platform to transmit four SVs Galileo signals. This will help to analyse the performance of the detection probability in terms of number of acquired signals, Doppler frequency shift and code phase delay. Also, I used all the four receiver channels via four antennae to obtain different signal receptions.

On the reception side, the receiving process is as follows:

1. The received signals are down-converted to baseband and sampled by the ADC and stored in real time into the memory.
2. Finally, the stored data are downloaded to the PC via USB.

The Galileo-OS signal samples at rate 10MHz and the IF is 0Hz. The general settings of these scenarios are as follows:

1. Software setup-MATLAB: the number of Galileo signals is 4 SVs that are uploaded simultaneously, and the length of the tested signal is 10ms.
2. Hardware setup-HaLo-430: The powers of the transmitted signals are set to 0 dBm; the frame length of each baseband signal is 81,920 samples and the number of pause samples after transmitting frame is 12800 samples. The local oscillator of the receiver device is equal to the carrier frequency, which down-converts the received signal to 0Hz. The frame length of the received signal is 200,064 samples, which represents double the (length of transmitting frame+ pause frame), to ensure receiving at least one full frame of the transmit signal. This frame length depends on the length of transmitting signal and the sample rate.

Several experiments were carried out to obtain various signals receptions, such as LOS signals, multipath or NLOS signals, high and low SNR values. This is accomplished by:

1. Rotating the transmission antenna by 0^o, 90^o, 180^o and 270^o to make LOS and NLOS signals.
2. Fixing and moving transmission antenna, to produce Doppler shift.
3. Blocking the transmitting and receiving antennae by different objects, such as glass, metal, a human body, water and wood, to create NLOS signals as well as to control the power of the received signal, as depicted in Figure 3-17.

However, we believe that this testbed signal still does not meet the actual Galileo signal received on ground after passing through 23Km of space (troposphere and ionosphere) and has attenuated considerably especially once it is mixed with current surrounding wireless technologies (Wi-Fi, etc.)

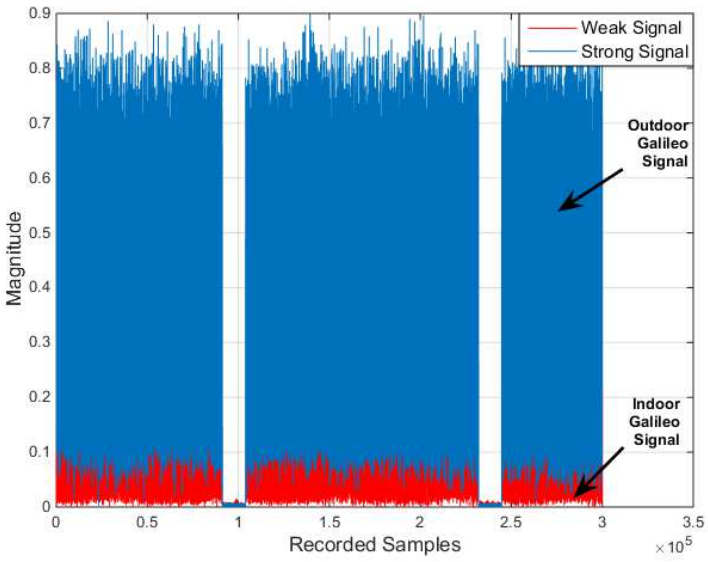


Figure 3-17 Received real-time signals

3.6 Measurements results of the ESCE method

3.6.1 Unambiguous cross-correlation function result

The multiplication of the generated subcarrier channels with the received Galileo signal shapes the envelope of the Galileo-OS signal from multiple peaks to a single peak. In order to assure the subcarrier elimination, we compared the CCF of our ESCE method with the CCF of an ambiguous method that shown in Figure 3-18.

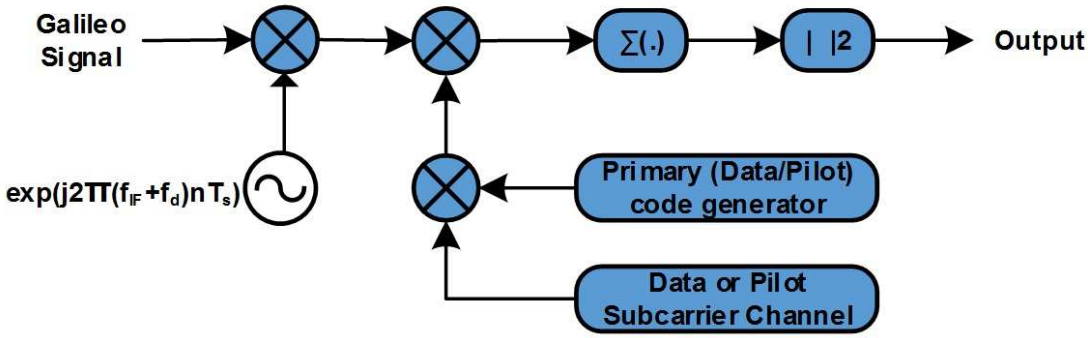


Figure 3-18 Ambiguous method

The result in Figure 3-19 shows that the CCF of a time-domain implementation of ESCE has a single peak in comparison with an ambiguous method time-domain implementation that produces three narrow peaks. Also, it is obvious that the magnitude of the CCF based ESCE method differs from the magnitude of the CCF based ambiguous method by around 0.5dB. This means that removing the subcarrier frequency in our implementation does not affect or reduce the performance of the detection probability because the gain obtained from acquiring 4ms at 10MHz sample rate equals to 46dB.

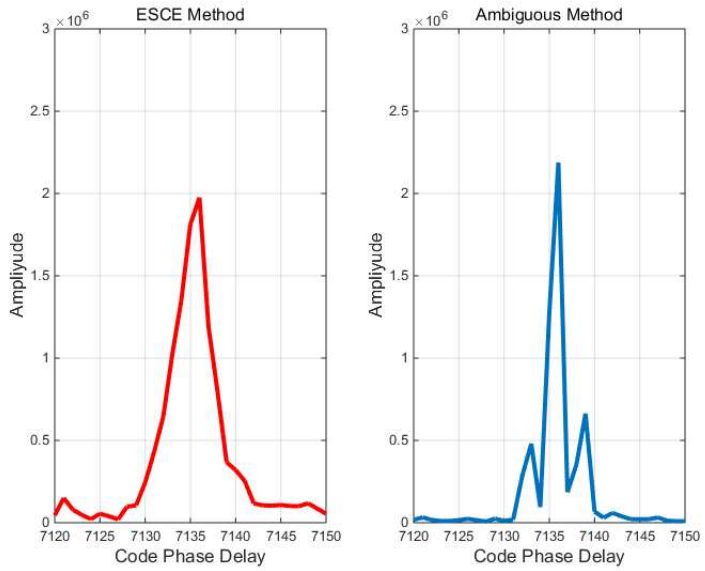


Figure 3-19 Cross-Correlation Function of ESCE and Ambiguous methods

3.6.2 Galileo power enhancement results and analysis

Apart from the ESCE implementation, we have implemented three other unambiguous methods. These are the DSB method [21], the BPSK-Like method [22] and the LoCo method [36]. This enabled us to compare ESCE to popular existing methods in similar conditions.

A probability of detection is important to determine the receiver sensitivity, because it represents a function of the carrier to noise ratio (C/N). Therefore, we have performed this comparison to; on one hand to assess the ESCE performance and on the other hand to show the enhancement made to the received Galileo signal with the mostly used methods.

The performance comparison is based on frequency-domain using FFT-search acquisition and realistic channel. The nominal received power of Galileo signal is set to -127dBm that equals 50dB-Hz, which represents the amount of C/N. The performance comparison between all methods is conducted with 4092 Chips PRN code length and Doppler frequency bin equal to 500Hz.

Figure 3-20 illustrates the detection probabilities of ESCE compared with the BPSK-Like method and LoCo method. ESCE process improves the power of the received Galileo signal by around 2dB. This power improvement leads to enhancing the probability of detection, where the ESCE method has better performance than BPSK-Like and LoCo methods by 1 and 2 dB respectively. As seen the LoCo method has lower performance than the BPSK-Like method due to applying other filters to the correlation process in each sideband. Also, because the intermediate frequency of the received Galileo signal is centred on 0Hz, this makes the performance of the LoCo method close to the performance of the BPSK-Like method. In other words, if the intermediate frequency is far from 0Hz by XMHz then the shifting of each sideband will be equal to ($X \mp f_{sc}$).

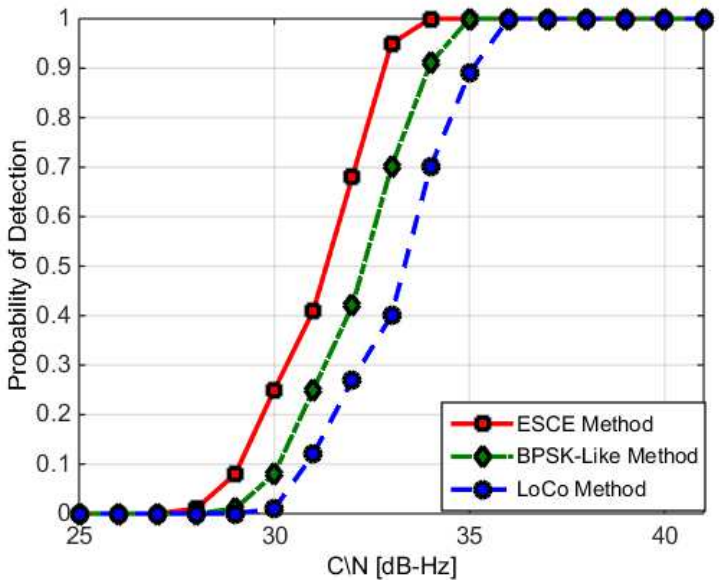


Figure 3-20 ESCE probability of detection vs. C\N

3.6.3 ESCE computational complexity considerations

The ESCE correlations can be implemented either in time-domain or in frequency-domain. The main difference in using these methods is that the real and imaginary parts are combined in the frequency-domain before being correlated with the reference code, while in time domain each part is correlated separately with the reference code. Therefore, the method that uses the two sidebands in time-domain will require four correlation channels, two channels for each sideband. The methods involved in this comparison are the DSB and LoCo methods, and the correlation is done in frequency-domain. Note that, the complexity rate of the time and the frequency domains correlation is same and based on [43] calculation.

This comparison demonstrates the correlation complexity versus different sampling frequency, which means the signal length (N_s) depends on the sample rate and equal to code length (4092) multiplied by the number of samples per code. For example, if the sampling frequency is 4.092MHz, then the sample per code is 1. In this comparison the addition operation is denoted by (N_{add}), multiplication is represented by (N_{mu}) and the Fourier transform operation is (N_{fft}) and calculated by ($N_s \log(N_s)$). The complexity representation for each method is divided into three sections, the first section comprises the shifting or filtering operation, the second section represents the multiplication of the received signal with the locally generated carrier signal operations and the last section includes the rest of the correlation process (i.e. from the generated code to the final correlation stage). For simplicity we represented the filtering process by a single multiplication.

The DSB's computational complexity (DSB_{comp}) is equal to:

$$DSB_{comp} = [(2N_{mu}) + (6N_{mu} + 2N_{add} + 2N_{fft}) + (7N_{mu} + 4N_{fft} + N_{add})] \quad \textbf{3. 7}$$

It can be seen clearly in equation (3.7) that the second stage and the third stage have a large number of multiplications because in the second stage the multiplications are the shifting of the locally generated carrier by the amount of the subcarrier frequency for each sideband and the multiplication of the locally generated carrier. While, in the third stage it requires generating BOC signal, i.e. the generated code multiplied by

the generated subcarrier frequency, then the generated BOC signal is filtered using another two filters.

The computational complexity of the LoCo method ($\mathrm{LoCo_{comp}}$) is formulated as shown in equation (3.8).

$$LoCo_{comp} = [(2\mathrm{N_{mu}}) + (4\mathrm{N_{mu}} + 2\mathrm{N_{add}} + 2\mathrm{N_{fft}}) + (4\mathrm{N_{mu}} + 4\mathrm{N_{fft}} + \mathrm{N_{add}})] \quad \textbf{3. 8}$$

Finally, the computational complexity of our ESCE method ($\mathrm{ESCE_{comp}}$) is illustrated in equation (3.9).

$$ESCE_{comp} = [(\mathrm{N_{mu}}) + (2\mathrm{N_{mu}} + \mathrm{N_{add}} + \mathrm{N_{fft}}) + (\mathrm{N_{mu}} + 2 * \mathrm{N_{fft}} + \mathrm{N_{add}})] \quad \textbf{3. 9}$$

To illustrate the calculation of the computational complexity for each method, let us assume that the sampling frequency is 10MHz, then the $\mathrm{N_{add}} = 40000$, the $\mathrm{N_{mu}} = 1.6 * 10^9$ and the $\mathrm{N_{fft}} = 1.84 * 10^5$. Consequently, our ESCE is about 70% less computationally expensive than the DSB method and 50% less than the LoCo method, , as shown Figure 3-21; this is achieved by saving complete correlation chain without affecting the acquisition process.

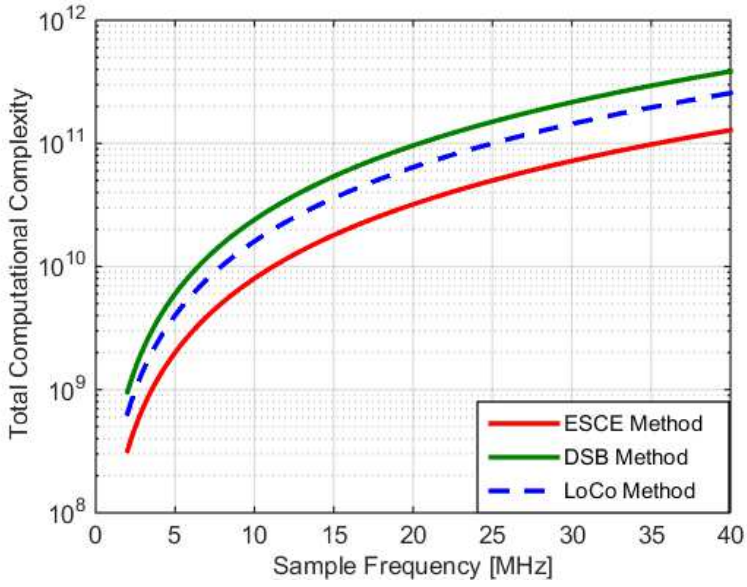


Figure 3-21 ESCE Total computational complexity

For processing time comparison, a Monte Carlo simulation was performed with 100 runs to calculate the average acquisition time. In this comparison, all the methods are

run with the same realistic signal-processing scenario. As shown in Table 3-1, the processing time achieved by ESCE is nearly half of the time required by the other methods. This proves that ESCE implementation is not only simple, but also faster. However, the only limitation of our ESCE method is the requirement of generating a primary code based on the locally generated subcarrier frequency, which is either data or pilot codes.

Table 3-1 ESCE processing time

Method	Processing Time	
	Average	**Standard deviation**
ESCE Method	2.84 sec.	0.0231
DSB Method	5.53 sec.	0.0548
BPSK-Like Method	4.35 sec.	0.0278
LoCo Method	5.26 sec.	0.0474

3.7 Concluding remarks on the ESCE method

In this chapter, the ESCE method was proposed to overcome the ambiguity and to enhance the Galileo-OS signal acquisition. The implementation of ESCE has eliminated the subcarrier frequency effect and simplified the acquisition process. The implementation requirements and detection performance of ESCE are analysed and compared to other widely used solutions, such as DSB, BPSK-Like and LoCo methods. The results showed ESCE's advantages in terms of reducing the complexity, improving the performance of the Galileo-OS signal acquisition and accelerating the acquisition process. In addition, the simulation shows that ESCE elimination of the subcarrier frequency effect offers around 2dB gain to the received signal power. Moreover, the literature survey has illustrated that unambiguous methods have been successful in acquiring and tracking BOC signals. However, most of them suffer from having complicated implementations using double sideband processing, or they suffer from a signal-to-noise deterioration, of around 3dB power in case of single sideband processing.

Our solution of eliminating the subcarrier frequency produces better results than the previous work in terms of performance, saving processing time, implementation

complexity and shapes the CCF to have a single peak like the CCF of the GPS-BPSK signal. Furthermore, our acquisition method can be implemented in the time-domain or the frequency-domain.

Therefore, we capitalised on this subcarrier elimination to combine the acquisition of the Galileo signal with the GPS signal in single process based on compressive sensing technique, as will be detailed in chapter 6.

I gave an engaging presentation on the ESCE method implementation in the 3rd Computing, Communication and Information Technology-CCIT conference on Wednesday, May 27, 2015 at Birmingham City University, UK.

Chapter 4

Orthogonal Joining Data and Pilot Galileo-OS Signals Acquisition

The Galileo-OS signal offers an efficient power distribution technique between the two navigation components/channels. The power distribution has been designed with 50/50 power split between the pilot and data channels [25]. Hence, it is possible to shorten the receiver's acquisition process by acquiring only the pilot or only the data channel using a single correlation chain. However, this leads to a 3dB power loss that is important when being acquired in harsh environments. Consequently, dual-channel (DC) acquisition is preferable to enhance the probability of detection of weaker signals. Obviously, the hardware implementation of the DC acquisition will require double the size/resources, while the software implementation will require more processing time as well as the resources overheads.

In this Chapter, we shall explain our novel implementation "OGSR" to acquire the Galileo-OS signal by joining these channels in a single correlation chain. This is accomplished by shifting the phase by 90-degrees to the copy of the received signals and then adding the shifted signal to the original received signal. The motivations of having an orthogonal signal are:

1. Both the data and the pilot signals are received with the same code phase delay and Doppler frequency shift (see section 4.2).
2. To be able to perform the acquisition in a single correlation chain, yet offering the same performance as using two correlation chains (see section 4.4).

OGSR implementation will require, as an overhead, having an orthogonally generated signal instead of having data and pilot generated signals separately. Note that the newly formed orthogonal signal will have the same power as the received signal.

4.1 Previous joining Galileo signal acquisition methods literature survey

A joint-data-pilot signal acquisition method is necessary to acquire Galileo-OS signal so as to gain the benefits of this transmission. Therefore, Galileo-OS signal acquisition needs at least two correlation channels to perform the joining between the received data and the pilot signals. A typical time-domain acquisition process, the received Galileo signal is multiplied by two orthogonal carrier frequencies, and then the output of the in-phase and the quadrature-phase components will go through two branches, as shown in Figure 4-1. The two branches are then correlated with the data channel components in the first branch and the pilot channels components in the second branch [44]. The detecting stage is accomplished by joining the output from these two branches, resulting in a 3dB gain as well as overcoming the ambiguity between the navigation message in the data channel and the secondary code in the pilot channel. However, this implementation requires four correlation channels.

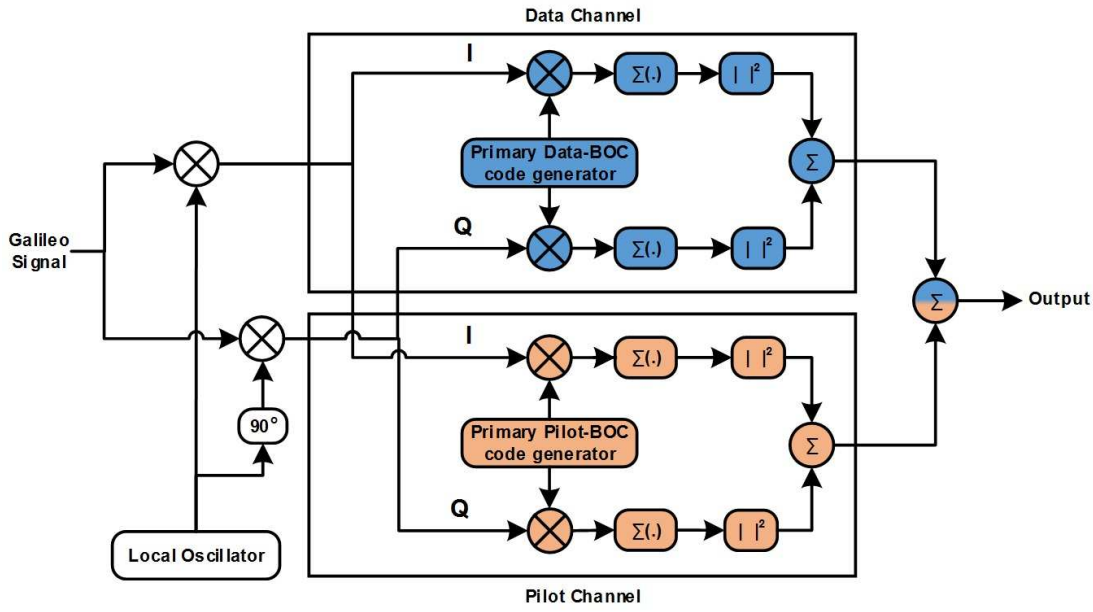


Figure 4-1 DC acquisition method based serial search

To implement a frequency-domain acquisition (FFT-search algorithm) for the Galileo-OS signal, solutions either acquire the data or the pilot signal alone, i.e. SC acquisition, or still join the data and the pilot signals, but using less correlation channels than a time-domain implementation. Figure 4-2, shows a joint data-pilot channels implementation [45]. This solution claims that combining the powers of the

two data and pilot signals provide a 2.8dB improvement in the signal detection performance over the acquisition of the data or pilot signal alone. However, this implementation does require two correlation channels.

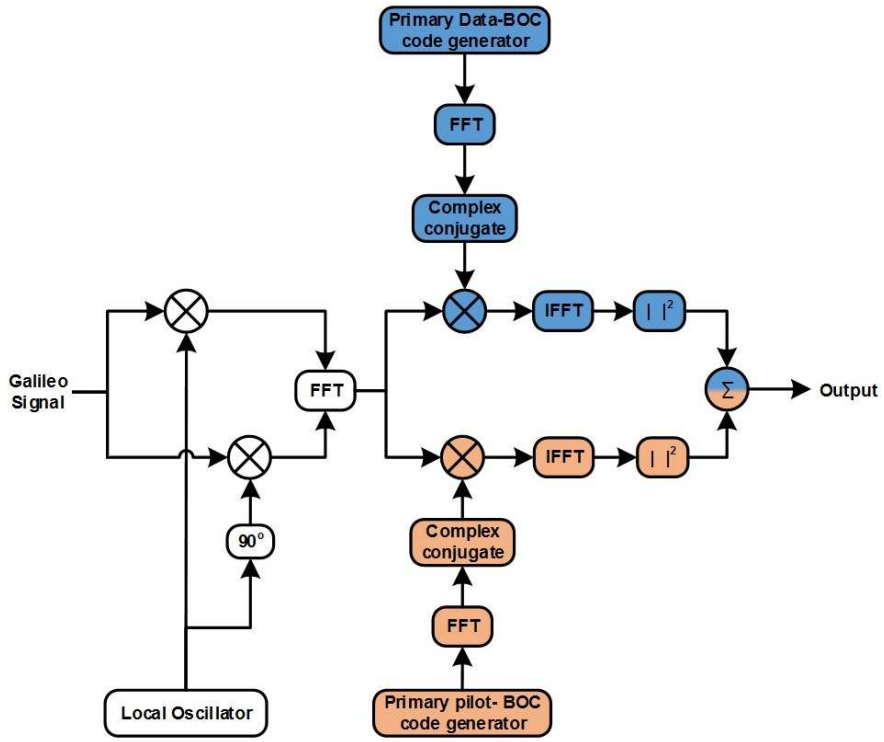


Figure 4-2 DC acquisition method based FFT search

A combined time-domain (serial-search) and frequency-domain (parallel/FFT-search) solution was designed to acquire weak Galileo signals [46]. The primary code is searched serially and the secondary code is searched in parallel. The gain obtained by combining these two engines is equal to:

$$\text{Total Gain} = G_{primary} + G_{secondary}$$

$$\text{Total Gain} = 10\log(N_s) + 10\log(N_{sp})$$

where N_s represents the number of samples in serial search and N_{sp} represents the number of samples in serial and parallel search.

Thus, the gain obtained from the signal that has 20 Chip code lengths is 13dB while for the code lengths that are equal to 100 Chip is equal to 20 dB. Nevertheless, this design has a good sensitivity, i.e. the processing gain will increase because is directly proportional to the number of the performed accumulation powers of the samples, but

it involves complex implementation because it correlates the received signal with two different engines, the serial and the parallel search. Also, the secondary code of the Galileo-OS signal is same for all 32 satellites, which is not like the other Galileo signals that have an own secondary code for each satellite. In fact, this gain can be partially obtained by using a single engine (time or frequency) for example differential acquisition or by increasing the integration time.

A further enhancement has been achieved by a differential joint-data-pilot acquisition solution (DfDC) [47]. In this implementation, the complex correlator output of each branch is multiplied by the delayed copy of itself, as shown in Figure 4-3. Where, the two branches are the correlation of the power-difference between data and pilot signals (B-C) and the power-sum of the data and pilot signals (B+C). These multiplication outputs are then summed up to complete this differential joining process. This solution claims that this post-processing would help to improve the acquisition of the low power signals that has C/N < 27 dB-Hz by 2dB. The performance of this work was compared with other SC and DC acquisition methods. The results showed that the differential acquisition has better performance, but obviously at the expense of higher complexity.

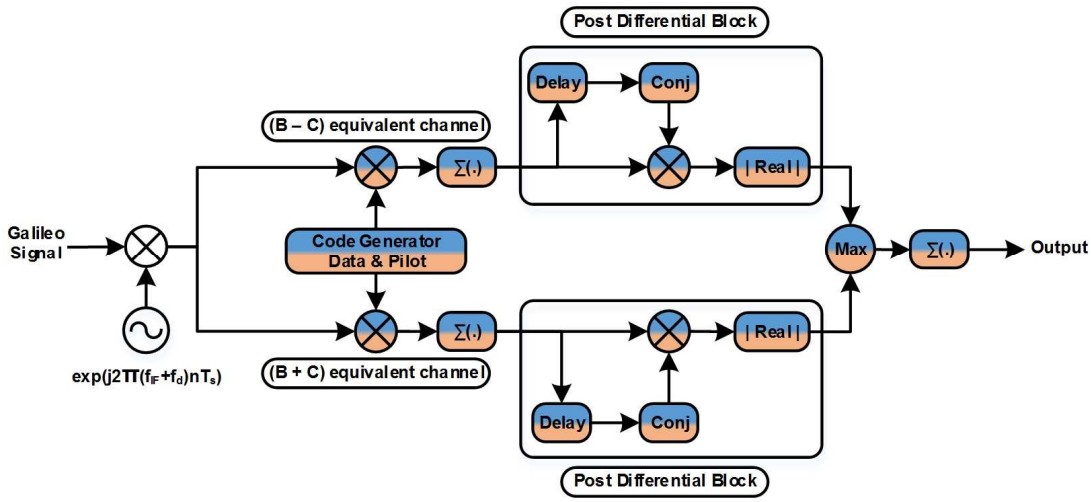


Figure 4-3 DfDC acquisition Method

In the same vein, a space differential acquisition (SDfC) solution claims to save half of the memory requirements that are used in the DfDC solution. This is achieved by using the phases of both data and pilot signals at the same time (rather than one at a time as used in DfDC) because the data and the pilot signals are transmitted

simultaneously from the same satellite, which means the Doppler and phase shifts of both signals are the same [48]. As depicted in Figure 4-4, the complex correlator's outputs are separated into the In-phase and the Quadrature-phase parts. Then the acquisition process is accomplished by summing the multiplication outputs of the real parts (I_data x I_pilot) and the imaginary parts (Q_data x Q_pilot). This solution, therefore, has better detection probability than the DC solution by around 2dB, and it is better than the SC solution by 5dB.

However, it requires more computational operations to split the real and imaginary components of both signals, as well as requiring more resources, such as mixers in the time domain, to multiply the complex correlator outputs.

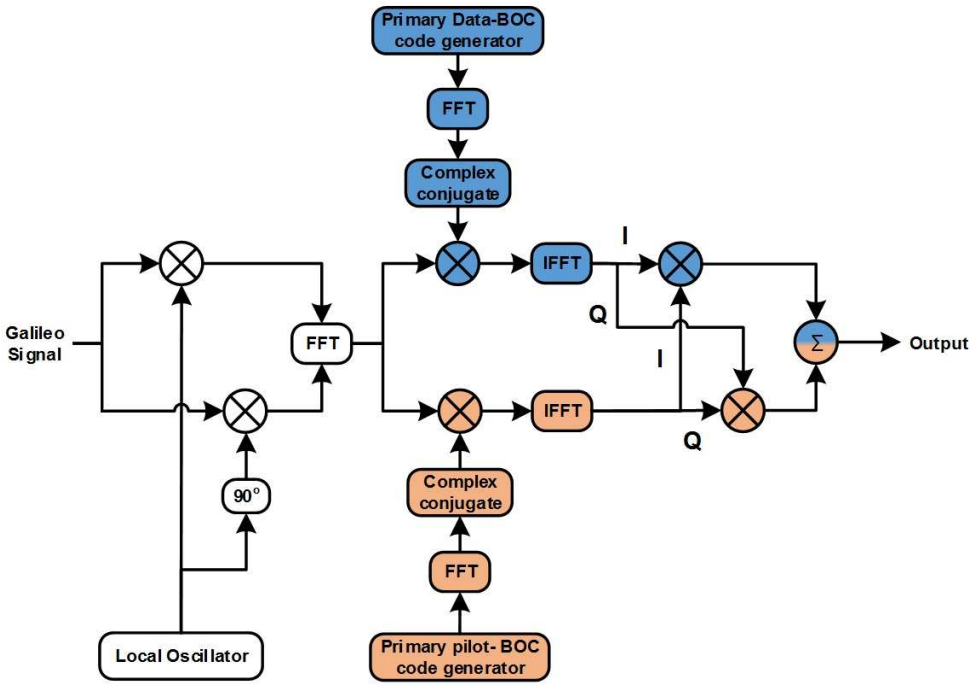


Figure 4-4 SDfC acquisition Method

A study was made to compare the performance of several joint strategies that have been designed to acquire Galileo-OS signal [49]. These strategies are SC, DC, multiplying strategy (B x C), assisted (B - C), summing combination (SB-C + SB+C) and comparing combination (CC), as shown in Figure 4-5. The comparison showed that there was around 2.8dB improvement to the acquisition performance for the CC, (B x C) and DC strategies than using conventional SC when there is no assistance available.

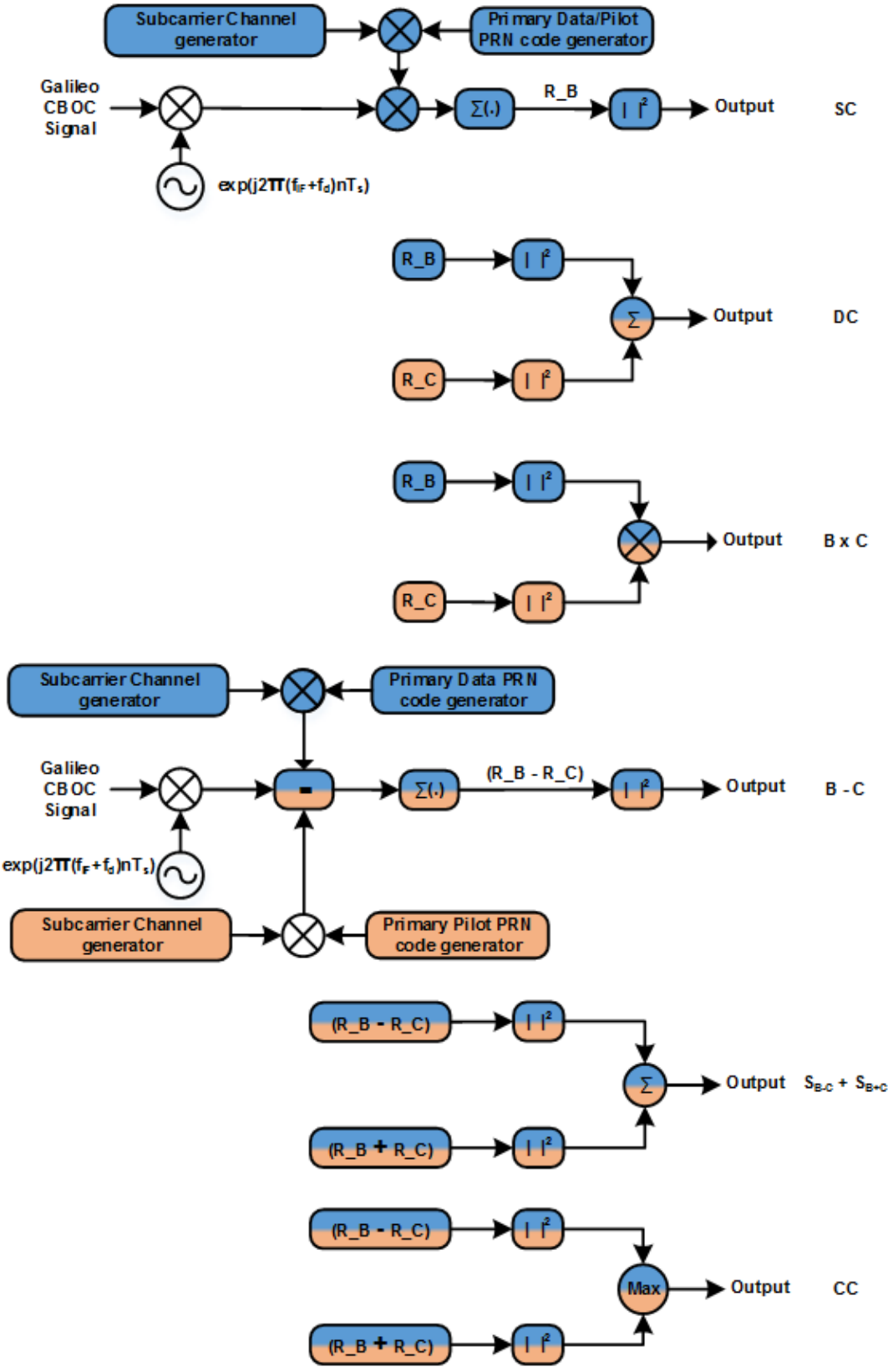


Figure 4-5 Joint strategies to acquire Galileo-E1 signal

While, if there is assistance, like which satellites are in view or the Doppler shifts for the present satellites, the (B - C) strategy is the best choice to acquire Galileo signals.

Moreover, in the low carrier to noise ratio C/N < 24 dB-Hz, the performance of the DC strategy is better than the CC strategy. Because in this region, low C\N, the Galileo signal is still in the noise floor and the CC strategy makes the search space suffers from the noise that coming from the two components channels (R_B - R_C) and (R_B + R_C).

To save valuable correlation resources, orthogonalising any two BPSK signals in a bandpass sampling receiver have been successfully implemented to track two signals simultaneously [50]. In this implementation, one of the received signals was passed through an HT and then combined with the second signal to construct an orthogonal signal. The orthogonal signal is then folded to the same reference frequency in the FNZ by using a single ADC. It is worthwhile to mention that this orthogonality has been implemented in the RF front-end to simplify the digital processing.

Our proposed OGSR capitalises on this saving to design an orthogonal acquisition chain for the Galileo-OS data and pilot signals, thus saving valuable resources and processing time.

4.2 The OGSR method structure

OGSR is designed to overcome the complexity of other DC-receivers implementations by half, or in other words, to reduce the overall acquisition/processing time. As shown in Figure 4-6, this is achieved by making the received Galileo signal orthogonal with a 90-degrees phase-shifted copy of itself, using an HT to do the phase-shift. OGSR is possible because the Galileo data and pilot signals are transmitted simultaneously, and therefore they have the same Doppler and phase shift, as illustrated in equation (3.2). Consequently, the 90-degrees shift can be applied to any of the locally generated channels. In this particular implementation, we have chosen the pilot signal to be the phase-shifted signal.

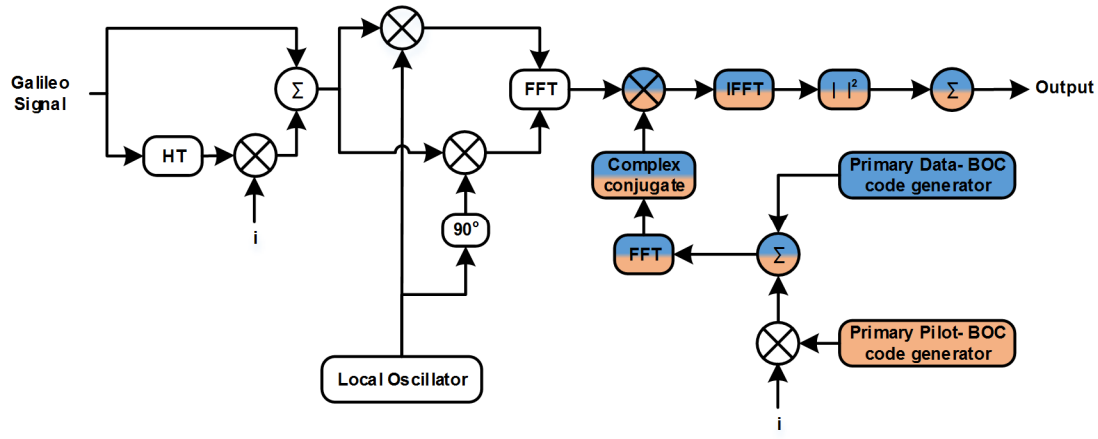


Figure 4-6 OGSR acquisition method

The following steps illustrate the design process of the OGSR acquisition block. As shown in Figure 4-6 the resultant combined orthogonal signal is represented by equation (4.1).

$$X_C[nT_s] = X_r[nT_s] - j\,X_r[nT_s] \qquad \textbf{4.1}$$

where X_C represents the complex received signal.

By substituting equation (3.9) in equation (4.1), then this orthogonal signal is:

$$\begin{aligned} X_r = {} & \Big(A\left[e_{E1\_B}(nT_s-\tau)S_b - e_{E1\_C}(nT_s-\tau)S_c\right] e^{j(2\pi(f_{E1}+f_d)nT_s)} \\ & + n_o(nT_s)\Big) \\ & - j\Big(A\left[e_{E1\_B}(nT_s-\tau)S_b \right. \\ & \left. - e_{E1\_C}(nT_s-\tau)S_c\right] e^{j(2\pi(f_{E1}+f_d)nT_s)} + n_o(nT_s)\Big) \end{aligned} \qquad \textbf{4.2}$$

After removing the carrier frequency with Doppler shift, i.e. when the locally generated frequency matches the frequency of the orthogonal signal, then the exponential term in equation (4.2) becomes equal to 1. The noise component n_o is considered uncorrelated, and for the sake of simplicity will be ignored, then the matched signal is shown in equation (4.3), which contains the codes and the navigation message only, but in a complex format X_{Co}.

$$\begin{aligned} X_{Co}[nT_s] = {} & \left[e_{E1\_B}(nT_s-\tau)S_b - e_{E1\_C}(nT_s-\tau)S_c\right] \\ & - j\left[e_{E1\_B}(nT_s-\tau)S_b - e_{E1\_C}(nT_s-\tau)(nT_s-\tau)S_c\right] \end{aligned} \qquad \textbf{4.3}$$

At this stage, we orthogonalise the replica BOC-codes of the data and the pilot channels. Therefore, the primary BOC-code of the data channel will be located in the real part of the equation (4.4) while the primary BOC-code of the pilot channel will be located in its imaginary part. This represents the complex generated code $\mathrm{C_C}$ and as follows:

$$\mathrm{C_C}[nT_s] = \left[\mathrm{e_{E1\_B}}(\mathrm{n}T_s - \tau)\mathrm{S_b}\right] + \mathrm{j}\left[\mathrm{e_{E1\_C}}(\mathrm{n}T_s - \tau)\mathrm{S_c}\right] \quad \textbf{4.4}$$

We now multiply this complex code with the matched signal of equation (4.5) after transforming them to the frequency domain:

$$Y_{BC} = IFFT[FFT(X_{Co}[nT_s]).FFT(C_C[nT_s])^*] \quad \textbf{4.5}$$

where $IFFT$ represent the inverse Fourier transform FFT is the Fourier transform and $(C_C[nT_s])^*$ is the complex conjugate of the complex generated code.

The aligned complex output of this stage is shown in equation (4.6).

$$Y_{BC} = N\,D_{E1\_B} - j\,N\,D_{E1_B} - N\,D_{E1_C} - j\,N\,D_{E1\_C} \quad \textbf{4.6}$$

where D_{E1_B} is the navigation message, D_{E1_c} is the secondary code and N is the total number of samples.

Finally, we square this complex output to represent the correlation output, $\mathrm{S_{BC-OGSR}}$ as follows:

$$S_{BC-OGSR}(\tau, f_d) = 2\,N^2\,D_{E1_B} + 2\,N^2\,D_{E1_C} \quad \textbf{4.7}$$

4.3 OGSR experiments setup

In order to verify the orthogonality of the designed acquisition process, OGSR method has been performed in two environments. The setups of these environments are:

1. Realistic environments: a same setup that was made in the ESCE method is applied in this OGSR implementation, see Section 3.5 for more details.

2. Simulink/MATLAB is used to simulate the OGCR implementation, with the sampling frequency of the Galileo signal is set at 16.368MHz and the folding frequency (IF) at 4.092MHZ. Figure 4-7 shows a snapshot of the Galileo-OS signal Simulink block diagram and Figure 4-8 depicts the Galileo-OS signal generation block that is implemented based on equation (3.5).

   Various scenarios were chosen to validate the performance of our OGSR implementation, as follows:

   A. (Tx→Channel→Rx) Block: the range power of the simulated Galileo signal is from -127dBm to -150dBm.

   B. Rayleigh Fading Block: two types of Doppler spectrum are used “Flat and Gaussian” to create multipath signals, which are up to 2 multipath signals in each generated signal.

   C. AWGN Block: in this particular block the chosen Mode was SNR-mode to control the amount of the additive white Gaussian noise.

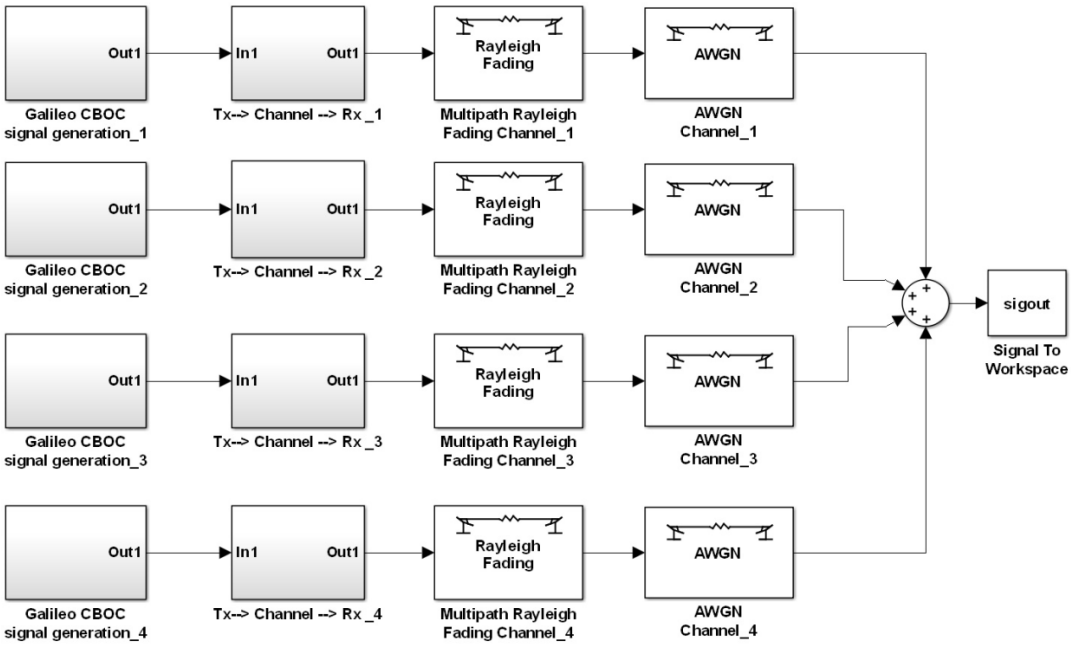


Figure 4-7 Galileo CBOC signals transmitter channels (upper-level block diagram)

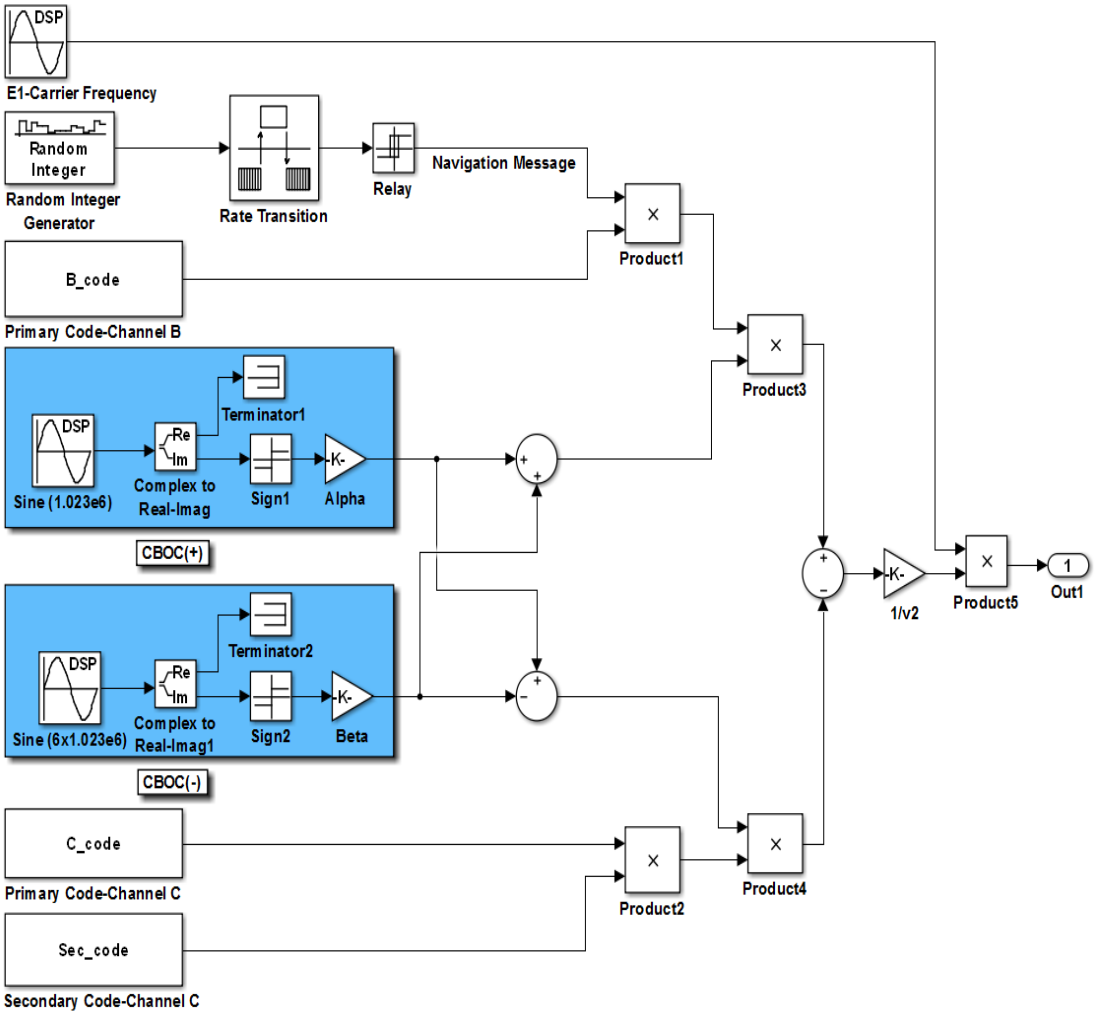


Figure 4-8 Galileo CBOC signal generation

4.4 Measurements validation of the OGSR method

This section highlights the achievements in terms of saving processing time, maintaining the acquisition performance, reducing the acquisition requirements and determining a suitable threshold for ambiguous Galileo signal acquisition.

4.4.1 OGCR performance and processing time results

A fair comparison is conducted by implementing the DC method shown in Figure 4-2, which is technically recommended in high and low C/N, as discussed in our literature survey.

For the processing time comparison, we performed Monte Carlo simulations with 100 runs to calculate the average processing time. In this comparison, all the methods are run with the same simulation scenario. Table 4-1 shows the processing time comparison, where our OGSR involves less processing time than the DC method by 35% and is only 1% greater than the SC method.

Table 4-1 OGSR Processing Time

Method	Processing Time	
	Average	**Standard deviation**
OGSR Method	6.63 sec.	0.0525
SC Method	6.62 sec.	0.1084
DC Method	10.16 sec.	0.3766

To ensure that OGSR maintains the acquisition performance as the DC method and realises the joint acquisition gain, a probability of detection comparison is carried out between our OGSR method with the DC and SC methods. The detection setup is as follows:

A. The acquisition search algorithm is the FFT-search.

B. The transmission channel path loss is calculated by using Rayleigh Fading channel, where the nominal received power is -127dBm

C. The number of multipath signals in each scenario is equal to; 1-signal in high C/N (>34 dB-Hz) and 2-signals in low C/N (<34 dB-Hz).

D. The Doppler frequency shifts are set to 1250 for scenarios (>34 dB-Hz) and -1500 for scenarios (<34 dB-Hz), where the bin frequency step is 500Hz.

E. The dwell time is equal to 4ms length of the received Galileo signal, i.e. full bit of the navigation message and represented by 4092 Chips PRN code length.

Figure 4-9, shows the result of the signal detection, where OGSR and DC perform better than using SC by 2.8dB. This is achieved as a result of combining the accumulated power of the data and the pilot signals. The results

also show that the OGSR's performance is as good as the DC's performance. This is realised because the correlation output of our OGSR ($S_{BC-OGSR}$) is equivalent to the correlation output of the DC method, as described in Section 4.2.

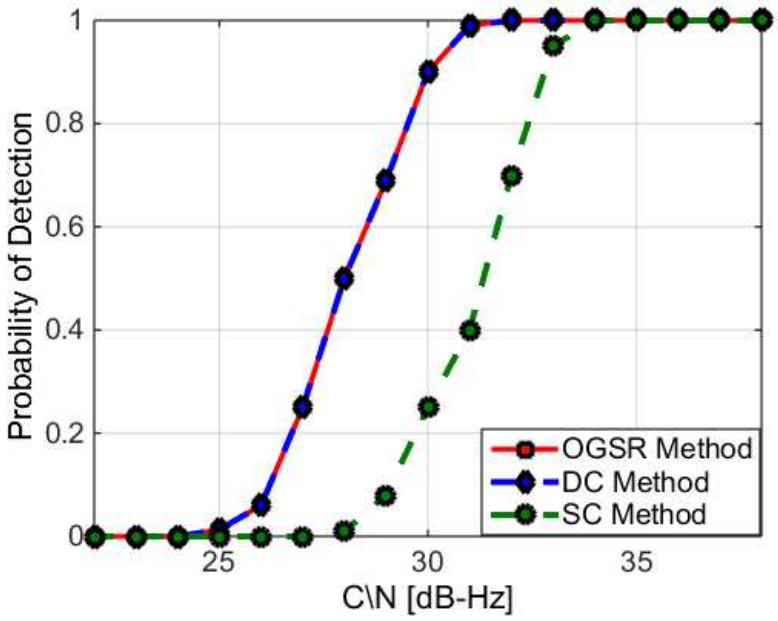


Figure 4-9 OGSR Probability of detection vs. C\N

4.4.2 OGSR computational complexity

In order to highlight the amount of reduction that is achieved in terms of the computational complexity, we have compared our OGSR with DC implementation. The same consideration to calculate the operation of addition and multiplication used in Section 3.6.3 is employed in this section. The computational complexity of the DC method (DC_{comp}) is calculated based on equation (4.8).

$$DC_{comp} = \left[\left(2N_{mu} + N_{add} + N_{fft}\right) + \left(4N_{mu} + 4N_{fft} + 2N_{mu} + N_{add}\right)\right] \quad \textbf{4.8}$$

While, the complexity of our OGSR method ($OGSR_{comp}$) is represented by equation (4.9).

$$OGSR_{comp} = \left[(N_{add}) + \left(2N_{mu} + N_{add} + N_{fft}\right) + \left(3N_{mu} + 2N_{fft} + N_{add}\right)\right] \quad \textbf{4.9}$$

As represented in equation (4.8) the DC method does not require a pre-processing stage but it costs twice more than the OGSR complexity when generating BOC signal in the third stage. While, the overhead of OGSR is only when adding the shifted signal and the locally generated BOC signal in the first and third stages. Figure 4-10 shows that our OGSR is less by 49% than the DC implementation because it saves a whole correlation processing chain, such as transformation to the frequency domain for both the received signal and the generated codes, inverse transformation to the time domain, and also the multiplication process.

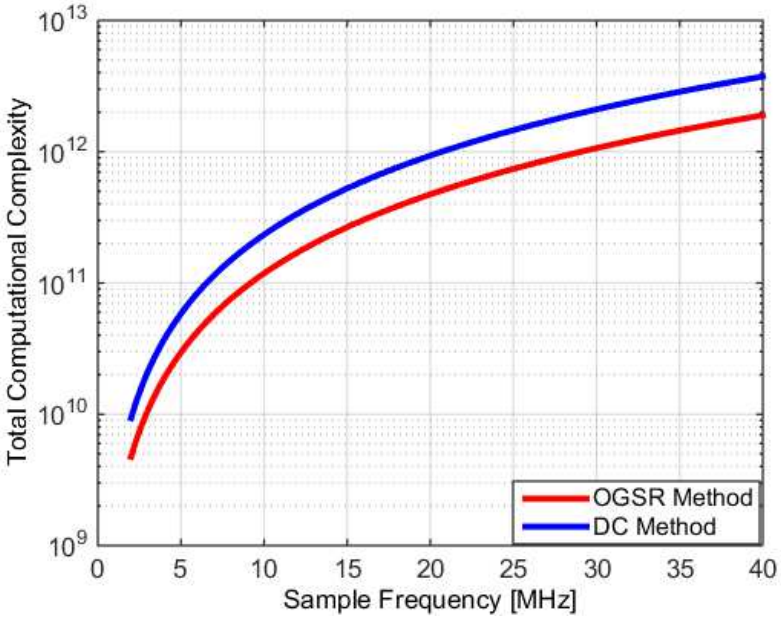


Figure 4-10 OGSR Total computational complexity

4.4.3 OGSR's threshold determination

Acquisition threshold is important to determine the detection probability and the false alarm probability; where decreasing or increasing the amount of the threshold will directly affect these probabilities. Typically, the threshold can be determined by two ways, either by capturing real signal under various environments to determine the threshold experimentally or by using simulated signal where the Gaussian noise is considered equivalent to the actual noise [51].

This measurement is valid when the acquired signal has a single cross correlation peak. In Section 3.1, we have explained the effect of the side peaks on the acquisition process, and as illustrated in Figure 3-4, the power ratio of the main peak to the side

peaks is equal to 4. This ratio represents the correlation between local BOC signal and outdoors received signal, i.e. just LOS signal without any multipath or power loss. While, in harsh environments the power of the received Galileo signal is weak and below -150dBm. Consequently, the power ratio between the mean and the side peaks decreases, because in this environment the power of noise is increased. Thus, the noise and multipath effect might increase the power of the side peaks to be equal or higher than the main peak. Therefore, the power ratio (4) is not valid in this situation, and we cannot acquire any present Galileo signal. So, in order to understand what the right threshold at low C/N is, this section shall explain the suitable threshold that must be used for the Galileo-OS signal.

To do so, we have performed our OGSR method and DC method to find a proper power ratio (threshold) that can allow acquiring Galileo signal with minimum false alarm probability. In this particular experiment the setup was as follows:

1. The received Galileo signal's power is set to -145dBm to -150dBm.
2. The fading channel used is Rayleigh Fading as illustrated in Figure 4-7
3. The Doppler frequency shift is equal to 500Hz.
4. Two multipath signals with code phase delay equal to (1ms and 0.1ms)
5. The power degradations for these multipath signals are set to -2dB and -3dB respectively.

As illustrated in Table 4-2, the ratio of the main peak to the side peak is equal to (3) at received power equal to -145dBm and when we minimise the power of the Galileo signal to -150dBm the ratio is reduced to (2) and so on and when the power goes down the ratio also decreases.

Table 4-2 Main peak to side peak and noise ratio

Method	OGSR Method	DC Method

	-145dBm	-150dBm	-145dBm	-150dBm
Correlation Power (dB-unit)	77.06	75.39	72.59	70.71
Highest peak to the second peak	3.3	2.4	3.4	2.2
Highest peak to the noise level	5.1	3.5	7.05	5.59

Where, decibels (dB) express the magnitude measurements specified in y. The relationship between magnitude and decibels is ydb = 20 log10(y).

Therefore, we have performed the false alarm test as a function of threshold with fixed C/N (-145dBm) to determine the proper threshold for both OGSR and DC methods. As depicted in Figure 4-11, when the threshold is below (1.8) the false alarm is increased for both methods. Practically, when we set the threshold to (2.5) it lead to reducing both of the probability detection and the false alarm probability, which means the receiver cannot acquire the available Galileo signal. Consequently, choosing a threshold equal to (2) will allow acquiring present Galileo signal and also at this value the false alarm is equal to zero. The test also showed that the false alarm of the DC method is only better than OGSR when the threshold is below (1.5) and that proves the competence of the OGSR method.

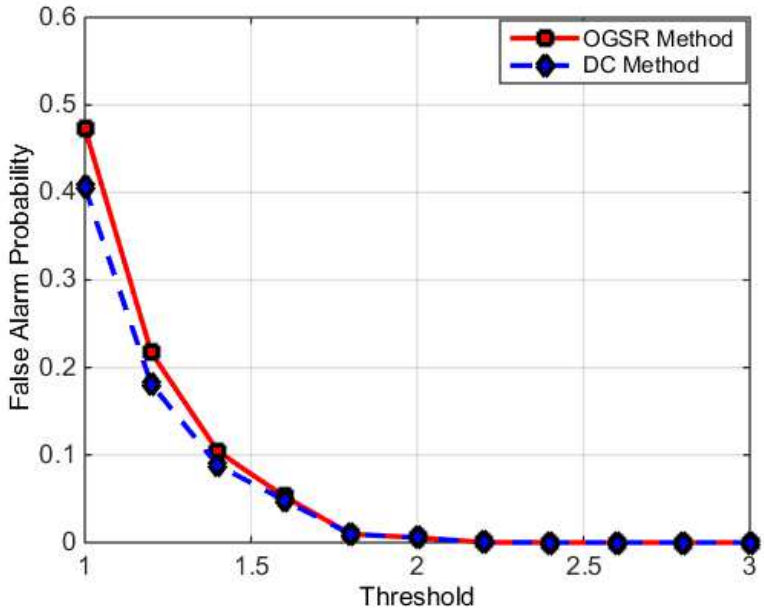


Figure 4-11 False alarm probability vs. Threshold

On the other hand, it is obvious that the correlation power (acquisition gain) based on our implementation is greater than the DC's correlation power by 5dB, as illustrated in Table 4-2. This is because in our implementation the codes (primary of the data and the pilot channels) are correlated twice in the real and the imaginary parts, while in the DC method each code is correlated separately, as illustrated in Figure 4-12. On the other hand, the noise level in our OGSR is 28% higher than in the DC method. However, with a threshold equal to (2) this will not affect the detection probability performance, as shown in Figure 4-11.

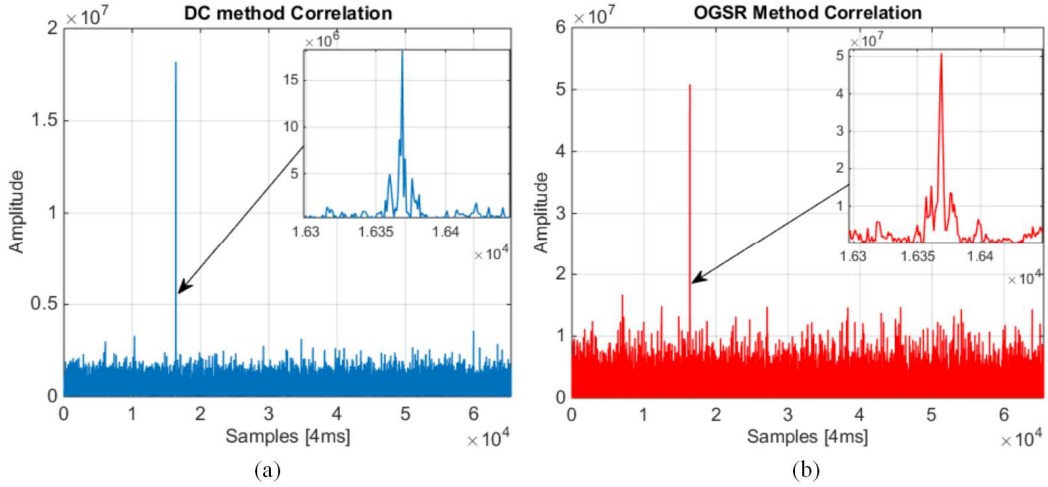


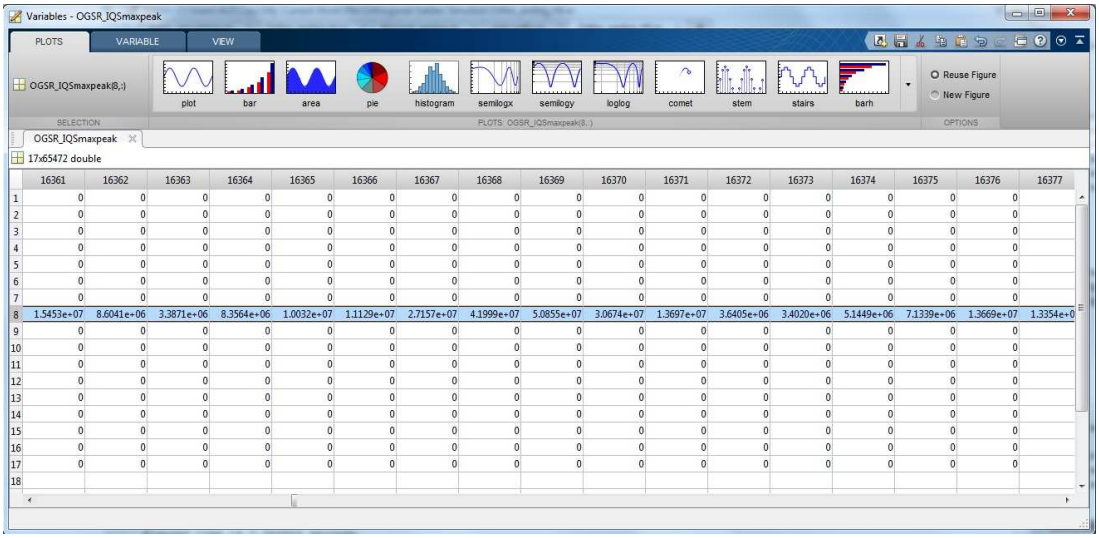
Figure 4-12 Highest correlation peak outputs (a) DC method and (b) OGSR method

The main interesting results that have been obtained by using OGSR method are that the correlation has occurred only when both the code phase delay and the frequency shift (carrier frequency + Doppler shift) of the orthogonal signal and the generated signal are matched, as shown in Figure 4-13-a, and as compared with the DC method that shown in Figure 4-13-b.

This is achieved because there is strong association between:

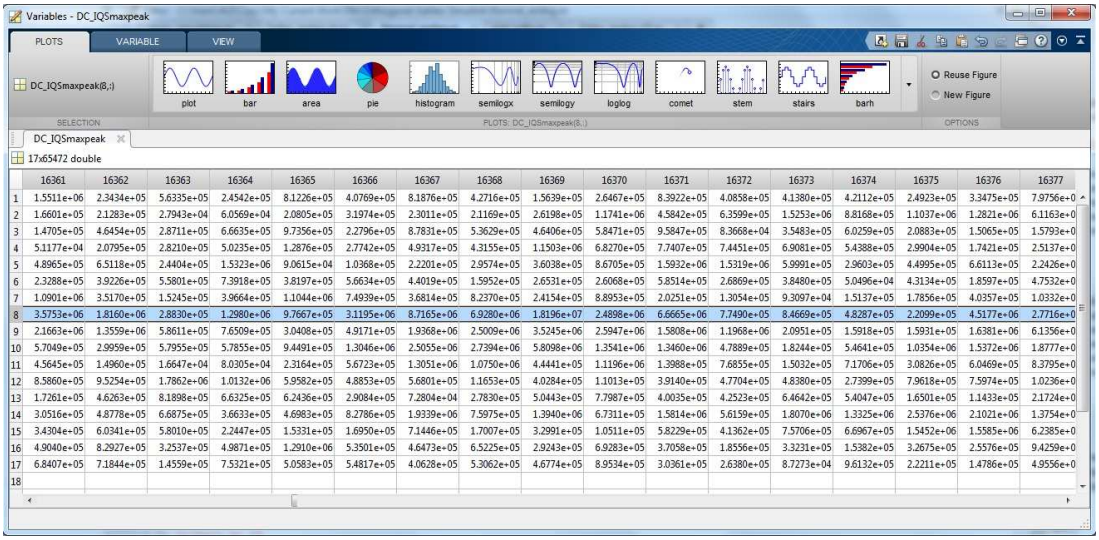
1. The integration of the codes and frequency in the orthogonal received signal and the orthogonal generated signal.
2. The correlation of the orthogonal codes in the received and in the generated orthogonal codes.

This means that acquiring Galileo-OS signal based on our OGSR is easily achieved because we have only one correlation if the code phase delay and Doppler frequency shift are same in the received and generated signals.



	16361	16362	16363	16364	16365	16366	16367	16368	16369	16370	16371	16372	16373	16374	16375	16376	16377
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	1.5453e+07	8.6041e+06	3.3871e+06	8.3564e+06	1.0032e+07	1.1129e+07	2.7157e+07	4.1999e+07	5.0855e+07	3.0674e+07	1.3697e+07	3.6405e+06	3.4020e+06	5.1449e+06	7.1339e+06	1.3669e+07	1.3354e+0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
18																	

(a)



	16361	16362	16363	16364	16365	16366	16367	16368	16369	16370	16371	16372	16373	16374	16375	16376	16377
1	1.5511e+06	2.3434e+05	5.6335e+05	2.4542e+05	8.1226e+05	4.0769e+05	8.1876e+05	4.2716e+05	1.5639e+05	2.6467e+05	8.3922e+05	4.0858e+05	4.1380e+05	4.2112e+05	2.4923e+05	3.3475e+05	7.9756e+0
2	1.6601e+05	2.1283e+05	2.7943e+04	6.0569e+04	2.0805e+05	3.1974e+05	2.3011e+05	2.1169e+05	2.6198e+05	1.1741e+06	4.5842e+05	6.3599e+05	1.5253e+06	8.8168e+05	1.1037e+06	1.2821e+06	6.1163e+0
3	1.4705e+05	4.6454e+05	2.8711e+05	6.6635e+05	9.7356e+05	2.2796e+05	8.7831e+05	5.3629e+05	4.6406e+05	5.8471e+05	9.5847e+05	8.3668e+04	3.5483e+05	6.0259e+05	2.0883e+05	1.5065e+05	1.5793e+0
4	5.1177e+04	2.0795e+05	2.8210e+05	5.0235e+05	1.2876e+05	2.7742e+05	4.9317e+05	4.3155e+05	1.1503e+06	6.8270e+05	7.7407e+05	7.4451e+05	6.9081e+05	5.4388e+05	2.9904e+05	1.7421e+05	2.5137e+0
5	4.8965e+05	6.5118e+05	2.4404e+05	1.5323e+06	9.0615e+04	1.0368e+05	2.2201e+05	2.9574e+05	3.6038e+05	8.6705e+05	1.5932e+06	1.5319e+06	5.9991e+05	2.9603e+05	4.4995e+05	6.6113e+05	2.2426e+0
6	2.3288e+05	3.9226e+05	5.5801e+05	7.3918e+05	3.8197e+05	5.6634e+05	4.4019e+05	1.5952e+05	2.6531e+05	2.6068e+05	5.8514e+05	2.6869e+05	3.8480e+05	5.0496e+04	4.3134e+05	1.8597e+05	4.7532e+0
7	1.0901e+06	3.5170e+05	1.5245e+05	3.9664e+05	1.1044e+06	7.4939e+05	3.6814e+05	8.2370e+05	2.4154e+05	8.8953e+05	2.0251e+05	1.3054e+05	9.3097e+04	1.5137e+05	1.7856e+05	4.0357e+05	1.0332e+0
8	3.5753e+06	1.8160e+06	2.8830e+05	1.2980e+06	9.7667e+05	3.1195e+06	8.7165e+06	6.9280e+06	1.8196e+07	2.4898e+06	6.6665e+06	7.7490e+05	8.4669e+05	4.8287e+05	2.2099e+05	4.5177e+06	2.7716e+0
9	2.1663e+06	1.3559e+06	5.8611e+05	7.6509e+05	3.0408e+05	4.9171e+05	1.9368e+06	2.5009e+06	3.5245e+06	2.5947e+06	1.5808e+06	1.1968e+06	2.0951e+05	1.5918e+05	1.5931e+05	1.6381e+06	6.1356e+0
10	5.7049e+05	2.9959e+05	5.7955e+05	5.7855e+05	9.4491e+05	1.3046e+06	2.5055e+06	2.7394e+06	5.8098e+06	1.3541e+06	1.3460e+06	4.7889e+05	1.8244e+05	5.4641e+05	1.0354e+06	1.5372e+06	1.8777e+0
11	4.5645e+05	1.4960e+05	1.6647e+04	8.0305e+04	2.3164e+05	5.6723e+05	1.3051e+06	1.0750e+06	4.4441e+05	1.1196e+06	1.3988e+05	7.6855e+05	1.5032e+05	7.1706e+05	3.0826e+05	6.0469e+05	8.3795e+0
12	8.5860e+05	9.5254e+05	1.7862e+06	1.0132e+06	5.9582e+05	4.8853e+05	5.6801e+05	1.1653e+05	4.0284e+05	1.1013e+05	3.9140e+05	4.7704e+05	4.8380e+05	2.7399e+05	7.9618e+05	7.5974e+05	1.0236e+0
13	1.7261e+05	4.6263e+05	8.1898e+05	6.6325e+05	6.2436e+05	2.9084e+05	7.2804e+04	2.7830e+05	5.0443e+05	7.7987e+05	4.0035e+05	4.2523e+05	6.4642e+05	5.4047e+05	1.6501e+05	1.1433e+05	2.1724e+0
14	3.0516e+05	4.8778e+05	6.6875e+05	3.6633e+05	4.6983e+05	8.2786e+05	1.9339e+06	7.5975e+05	1.3940e+06	6.7311e+05	1.5814e+06	5.6159e+05	1.8070e+06	1.3325e+06	2.5376e+06	2.1021e+06	1.3754e+0
15	3.4304e+05	6.0341e+05	5.8010e+05	2.2447e+05	1.5331e+05	1.6950e+05	7.1446e+05	1.7007e+05	3.2991e+05	1.0511e+05	5.8229e+05	4.1362e+05	7.5706e+05	6.6967e+05	1.5452e+06	1.5585e+06	6.2385e+0
16	4.9040e+05	8.2927e+05	3.2537e+05	4.9871e+05	1.2910e+06	5.3501e+05	4.6473e+05	6.5225e+05	2.9243e+05	9.9283e+05	3.7058e+05	1.8556e+05	3.3231e+05	1.5382e+05	3.2675e+05	2.5576e+05	9.4259e+0
17	6.8407e+05	7.1844e+05	1.4559e+05	7.5321e+05	5.0583e+05	5.4817e+05	4.0628e+05	5.3062e+05	4.6774e+05	8.9534e+05	3.0361e+05	2.6380e+05	8.7273e+04	9.6132e+05	2.2211e+05	1.4786e+05	4.9556e+0
18																	

(b)

Figure 4-13 Doppler frequency bin steps correlation (a) OGSR method (b) DC method

4.5 Concluding remarks on the OGSR method

In this work a novel joint-data-pilot signals acquisition method for Galileo-OS signal is designed. The novelty of this work focuses on joining and acquiring these data and pilot signals in a single correlation chain by forming them in an orthogonal format. So

this process saves valuable resources and accelerates the acquisition time as compared with conventional joining methods, such as time-domain implementation “4-correlation channels” and frequency-domain implementation “2-correlation channels”.

The implementation requirements and detection performance are compared and analysed with the DC acquisition method. The results show that our OGSR performance is as good as DC method because our OGSR combines the Galileo-OS data and pilot signals’ powers. In fact the significant reductions are achieved in terms of the computational complexity (49%) and the processing time (35%); these make our OGSR a good candidate for the Smartphone’s software receiver.

I have presented this OGSR method in the 3rd Computing, Communication and Information Technology-CCIT conference on Wednesday, May 27, 2015 at Birmingham City University, UK. The audience were very attractive of this novel idea for constructing an orthogonal Galileo signal.

Chapter 5

GPS Signals Acquisitions Based-Compressive Sensing

The GPS/GNSS signal acquisition process is the most important and power/processing intensive in any GNSS solution to achieve short time to first fix. In time domain solutions, typical receivers have hundreds of hardware correlator engines, while in the frequency domain a high performance processor is needed to perform FFT/IFFT processes. Either way, the acquisition process requires a large overhead from the Smartphone's CPU and battery power.

The acquisition in such GPS conventional receivers generates replica SV signals PRN codes and carrier frequency with a range of Doppler frequencies (±4KHz) to acquire the GPS signals. i.e. signal acquisition is to find the correct code delay and frequency in the received signal. The search process is conducted in two dimensions; the code phase search and the Doppler frequency shift search as shown in Figure 5-1, i.e. each cell comprises a replica PRN code and locally generated frequency. The signal detection is achieved when the two parameters of the code and the Doppler have high correlation value.

On the other hand, the CS technique aims to recover the full signal band by using below-Nyquist rate sampling if the signal has a sparse representation or is nearly sparse [52]. Fortunately, the GNSS signals, like any wireless RF signal, are relatively/nearly sparse. Thus, the GNSS-based-CS receiver solutions can achieve faster acquisition process and low power consumption; which can be performed with fewer measurements than a conventional solution.

This chapter will introduce the main concept of CS as details in Section 5.1, and then demonstrate the implementations of our CS-based methods in Sections 5.3 and 5.6 to acquire the GPS signal.

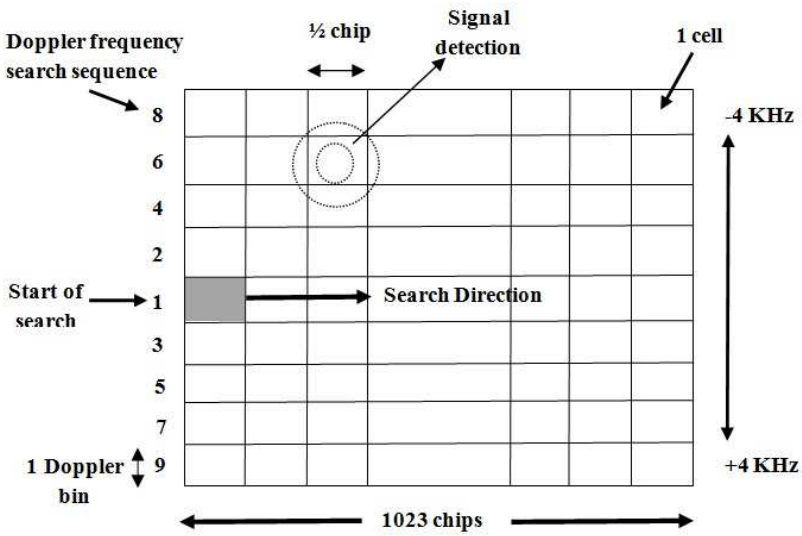


Figure 5-1 Two-dimension signal detection

5.1 Basic CS concept

In typical signal acquisition processing, the signal is sampled based on Nyquist rate to recover the full information of the signal, if present. While, CS asserts that the received signal can be sampled at below Nyquist rate sampling without information loss if the signal has a sparse representation [52]. Also, it is guaranteed to acquire and reconstruct any sparse signal with much fewer measurements, by exploiting the sparsity in the signal [53]. Practically, CS samples the received signal according to either the "occupation information band" or the "information rate", which is less than the minimum Nyquist sampling rate of double the carrier frequency. This is why CS is very attractive in applications such as image and signal processing [54].

As depicted in Figure 5-2, for any k sparse received signal x(t), in some "sparsity basis" $\psi \in \mathbb{C}^{N\times N}$, then, in CS theory this signal vector $x \in \mathbb{R}^{N\times 1}$ can be recovered from m linear measurements or compressed measurements $y \in \mathbb{R}^{M\times 1}$, where ($K < M << N$) [55].

$$y = \phi\, \psi\, \alpha \qquad \textbf{5.1}$$

where $\phi \in \mathbb{R}^{M\times N}$ *is the sensing/transform matrix and* $\alpha \in \mathbb{R}^{N\times 1}$ *is recovery signal*

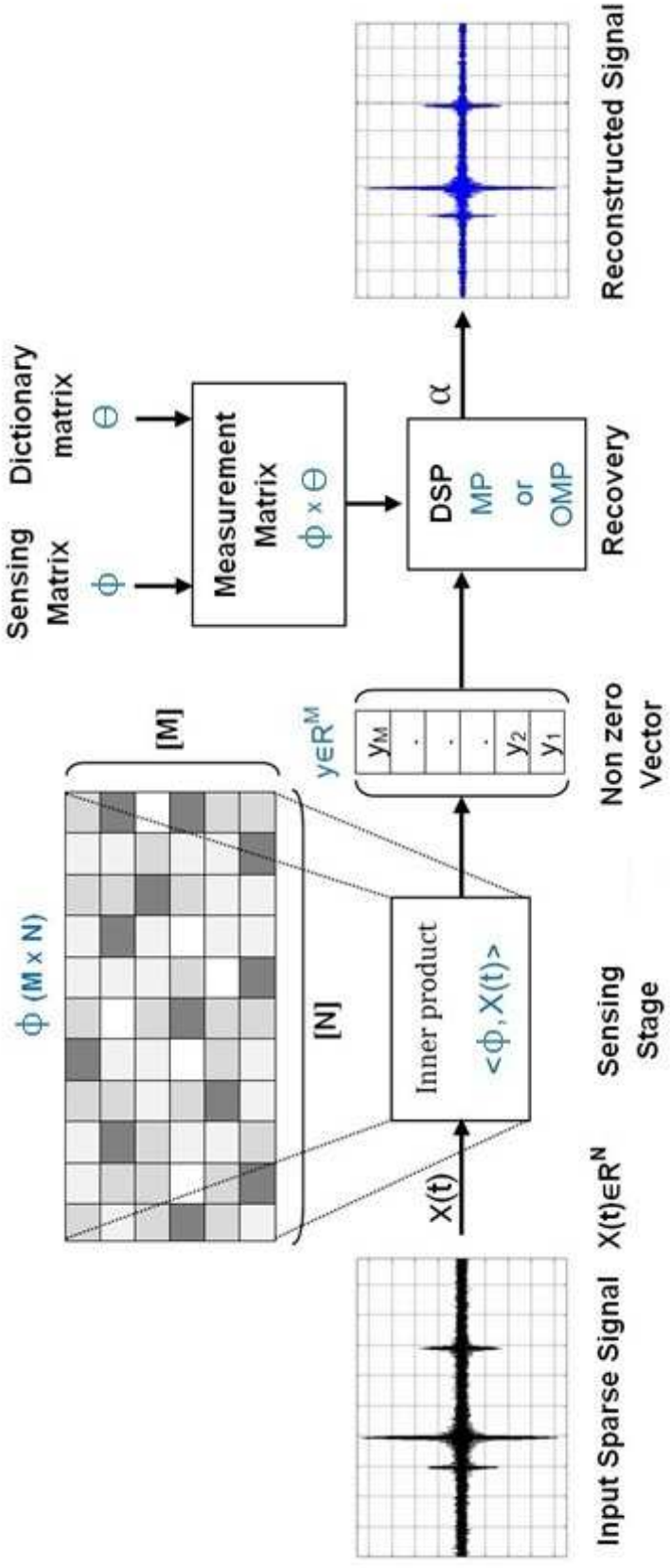


Figure 5-2 CS process

$$x(t) = \sum_{n=1}^{N} \alpha_n \psi_n(t) \quad \textbf{5.2}$$

To achieve this measurement's reduction, the following conditions have to be met [52]:

1) The sensing/transform matrix ϕ should satisfy the Restricted Isometry Property (RIP), where the RIP depends on the orthogonality of the arbitrary subsets of the column vectors of ϕ, more details in [56].

2) There should be low coherence between the transform matrix ϕ and the basis matrix ψ. This low coherence, $\mu(\phi, \psi)$, is expressed by:

$$\mu(\phi,\psi) = \sqrt{n} \max_{1 \le k,m \le n} \left| \langle \phi_k, \psi_m \rangle \right| \in \left[1, \sqrt{n}\right] \quad \textbf{5.3}$$

Furthermore, the deterministic sensing matrix will downsize the number of the required measurements. Nonetheless, random sensing matrix such as Gaussian matrix, binary matrix [57] or Bernoulli matrix [58] can also be used, as they would satisfy the RIP.

5.2 CS-based solutions literature survey

A high sample rate produces a huge number of samples that are necessary for the correlation (acquisition and tracking) process. Location-based services/applications on Smartphones require fast GNSS acquisition with low power consumption. On the other hand, the principle of CS is to permit the sampling of sparse signal below Nyquist rate and reconstruct the signal without information loss.

Therefore, an RD technique was designed to sample the wireless signal below Nyquist rate by exploiting the sparsity in the received signal [59]. In this technique, the received signal is typically mixed with a square waveform (±1), as shown in Figure 5-3, that are generated by an LFSR at or greater than Nyquist rate. This mixing process makes each received signal have a distinct signature. The mixed signal is then applied to the low-pass filter and then sampled below the Nyquist rate. However, the sampling based RD technique has to be performed in the AFE,

requiring synchronisation with the signal during the reconstruction, which necessitates the addition of more header bits to the spread signal.

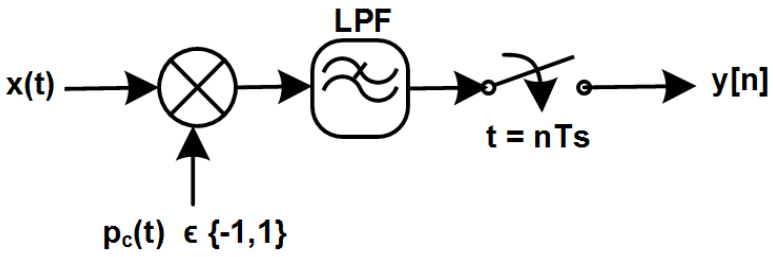


Figure 5-3 RD technique

Aimed at recent wireless communication technology signals, an RD technique has been successfully applied to demodulate the direct sequence spread spectrum IEEE 802.15.4 standard technology signals, where the input signal is mixed with PRN sequences for a period equal to the Nyquist rate [60]. The sampling rate used in this work is half the Nyquist rate, which leads to reducing the power consumption.

An effective technique was proposed, called Xampling technique, to improve on the RD technique by [61]:

1. Improving the sparse representation of the received signal by aliasing it with a locally generated arbitrary periodic waveforms using multichannel, as shown in Figure 5-4. The mixing rate of these channel waveforms is below the Nyquist rate (equal to the sampling rate, which is the maximum information band of the received signals).

2. This multi-channel aliasing arrangement helps direct signal reconstruction without the need for code synchronization [62].

The outputs from each channel represent baseband signals with a distinct sensing, i.e. each channel has its own signature, because in each channel the received signal multiplies with different periodic waveform. As shown in Figure 5- 5 (a, b and c), both signals occupy the same bandwidth (2MHz) but each one of them has different sensing, these differences will help easily reconstruct/acquire the received signal by utilising the same periodic waveforms (sensing matrix).

The signal reconstruction is then accomplished by:

1. Converting the sparse/samples vector to a frame using CTF block [63], which converts the computational problem from infinite-measurement vectors to multi-measurement vectors.
2. Converting these multi-measurement vectors to a single-measurement vector using the ReMBo algorithm [64].
3. Applying the OMP algorithm [65] to find the support values, this can be used to reconstruct the signal.

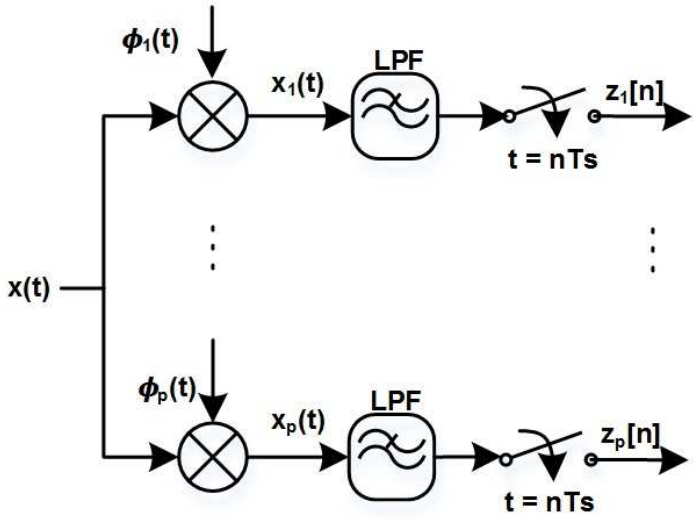


Figure 5-4 Xampling technique

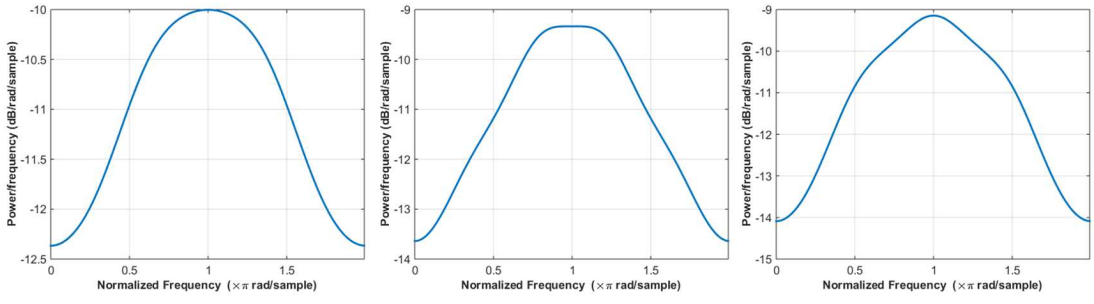


Figure 5- 5 Power spectrum outputs from different sampling channels

Furthermore, the resolution of the RD and the Xampling techniques is less than the traditional techniques such as BPS. To illustrate that the GPS code resolution based on Xampling technique will be half a chip because each chip is represented by two samples since the Xampling sampled the GPS signal at the information band

(2.046MHz). While the code resolution based on BPS will be at least double the Xampling resolution because the received signal is sampled at rate equal to double the information bandwidth, as detailed in Section 2.1. Therefore, the acquisition gain or the SNR of the acquired signal base on BPS will be greater than the Xampling as it accommodates the power of more samples but at the expense of higher digital processing.

A compressive multiplexer can be used to sample the RF signal below the Nyquist rate [66]. The main differences between this algorithm and the RD technique are:

1. This algorithm uses a multichannel to sample the received signal.
2. Each of the channels will down convert the received signal to baseband, and then mix the baseband signal with a square wave at a period equal to the Nyquist rate.
3. The output from the mixing process is summed once per chip before being sampled at a low rate.

In the same way, a combined RD and Xampling techniques are therefore more efficient in reconstructing the signal [67]. Such implementation takes the advantages of Xampling by using multichannel to alias the received signal with the random square waveform by using shift register at Nyquist rate. Then the alias signals are sampled below Nyquist rate. Note that increasing the number of channels would be allowed to reduce the sample rate, but not below the signal information rate. However it can apply the concept of the Xampling technique to reduce the rate of the random square waveform.

In order to have objective comparison, the rest survey specifically summarises the most recent CS-based solutions that are designed to acquire the GPS signal.

A CS-based solution would be to generate sparse vectors of the received GPS signals using random compressive multichannel samplers, CMS [57]. As shown in Figure 5-6, these random channels sample the GPS signals at the bandwidth information rate (2.046MHz) to have half chip resolution. The sparsed signals are then acquired through the same Xampling in three steps:

1. Convert the sparse vector to a frame using CTF block, to convert the computational problem from infinite-measurement vectors to multi-measurement vectors.
2. Convert these multi-measurement vectors to a single-measurement vector using the ReMBo algorithm.
3. Apply the OMP algorithm to find the support values of the acquired signals, where the recovery support represents the code phase-delay and the Doppler-shift of each acquired satellite signal.

This solution allows for accumulating the received power for acquiring 20ms length with the same number of channels at once. However, it necessitates a complex front-end hardware to constructing the random CMS. Moreover, this solution is designed based on static acquisition, and making it dynamic or doing any modification like reducing the samplers' length or improving these multichannel will need very complicated hardware implementation.

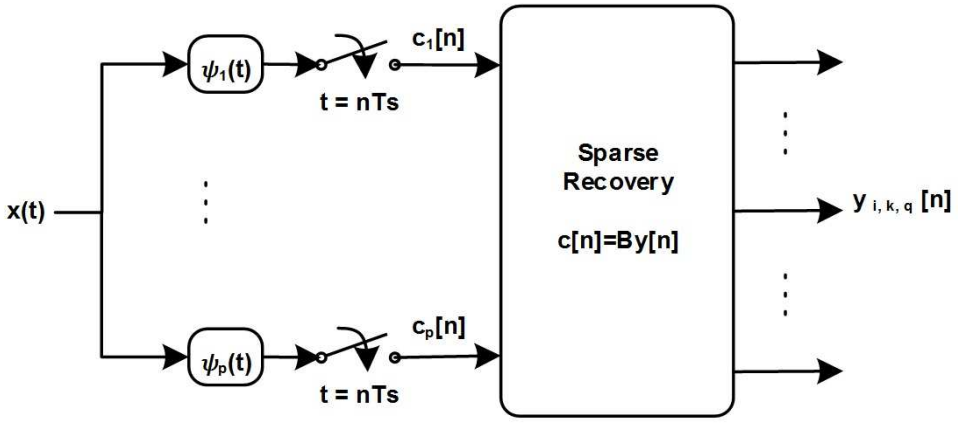


Figure 5-6 CMS acquisition solution

On the other hand, a CS solution that targets the GPS signal acquisition, we called (GCS-1) uses a low sampling frequency (2.046MHz) to have half chip resolution [68]. The sampled signal then correlates with banks of correlators or parallel-correlators. The resultant sparse vector (y_o) represents the matching powers between the correlation process, and these powers contain at least three high values due to half chip spacing in the generating correlators. Identifying the code phase delay and

Doppler shift of the GPS signal is then obtained by two stages using deterministic sensing matrix (Walsh-Hadamard matrix) and the two measurement matrix ψ_1 & ψ_2 that dividing the output into two sub-vectors.

$$\mathrm{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

$$\mathrm{y} = \begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix} y_o \qquad \textbf{5.4}$$

The first stage of detection, as expressed in equation (5.5), is to find the peak index from y_1 and the magnitude of this peak should be more than a certain threshold. The next stage is to find the correct index, as expressed in equation (5.6), by taking the maximum value of the inner product between y_2 with each row of the second measurement matrix ψ_2:

$$y_1 = \psi_1 y_o \qquad \textbf{5.5}$$

$$\hat{l}_{max} = \arg \max_{l} \left(\langle y_2 (m), \psi_{2\, m,l}(m) \rangle \right) \qquad \textbf{5.6}$$

These operations of producing sparse vector and the two stages are repeated in the same sequence to acquire the rest of the GPS signals. However, it overcomes the computational complexity of the CMS solution it is at the expense of large memory storage. Where the dimension of each bank is (2046 x 2046) and each row comprises single code phase delay and each bank includes only single Doppler frequency shift. And therefore, there are 41 banks of correlators for each GPS satellite and so on. On the other hand, this solution is designed with a fixed signal length dictionary that equals to 1ms, and changing the length will directly affect the required memory storage.

In the same vein, the measurement matrix can be made to acquire either BPSK modulation signal or BOC modulation signal, and we named as GCS-2 [69]. This distinction in signal modulation is realized by adding an indicator (“0” for BPSK and “1” for BOC) inside the measurement matrix. This method, however, focused on the chipping rate to acquire the signals that use BPSK(1) and BOC(1,1) but without consideration of the codes’ length of these signals. For example:

1. Both the GPS C/A-BPSK(1) code signal and the Galileo-OS BOC(1,1) signals have the same chipping rate but they don't have same code length.
2. Both GPS L1-C, and Galileo E1-OS use BOC(1,1) modulation but they have different code length, where 10ms for L1-C and 4ms for E1-OS.

Moreover, the measurement matrix was designed to handle one type of signal at a time, i.e. either BPSK signal or BOC signal.

A C/A code folding GPS signal acquisition was designed based on CS to accelerate the CS processing time (FCSG) [70]. Basically, the C/A codes for each satellite is folded "M" times then the folded codes are combined to reduce the computation complexity in the code phase delay search process. This implementation, firstly, transforms the received GPS signal to the CS-domain by multiplying the signal with a sensing matrix. Then the outputs are correlated with folded codes. The final process is achieved by applying the correlated results with the dictionary matrix to the search algorithm. In this implementation, a new search algorithm is proposed that is called "Projection Elimination Recovery Algorithm (PE)". The PE algorithm is developed the OMP algorithm, where finding the supports value is achieved by determining the highest column sum, while the PE determines the support by taking the highest projection. This implementation, however, reduces the cost of the computational complexity from two aspects

1. Reducing the size of codes and dictionary matrix
2. Minimise the search calculations in the OMP algorithm

Nevertheless, the folding process was firstly proposed for the P(Y)-code GPS signal acquisition because the length of P(Y)-code is (6.1871 x 1012) Chip while the C/A-code is only (1023) Chip. Therefore, folding the C/A code generates a low orthogonal code, because the C/A code is designed to be orthogonal to itself at or more than 1 Chip phase delay and with other GPS signals' codes at code length (1023) Chip and otherwise this orthogonality will decay. On the other hand, this implementation neglects the Doppler frequency shift and considers the shift as constant. Incidentally, we had this experience before and we have found that if the code phase resolution is 0.5, which is same as the resolution in this implementation, and there is at least 500Hz Doppler frequency shift that makes the correlation between the received and

locally generated code equal to “0”, more details in Section 6.1. Consequently, regarding this implementation if the Doppler frequency shift is considered then the maximum fold will be equal to one.

Three types of dictionary matrices are adopted in the CS-based GPS signal acquisition implementation, which is called “Sparse-GPS (S-GPS)” to find shortest searching time versus these types [71]. These dictionaries are:

1. Multi-channel stacked: the dictionary of each satellite breaks down into small dictionaries (code delay and Doppler shift). The number of dictionaries for each satellite therefore is equal to the number of Doppler bins (41 dictionaries) and the total number of channels is equal to (41 dictionaries x 24 = 984 channels).

2. Multi-channel flattened: This dictionary type combines the representation of the code and frequency in a single dictionary to reduce the number of channels to make it only 24 channels. This type will reduce the computational complexity in the search algorithm.

3. Single-channel flattened: this type joins all previous 24 channels in a single channel to construct a full dictionary matrix for all GPS satellites. Therefore, this enables the S-GPS implementation to acquire the GPS signals at once.

Note that all these dictionaries consist of multiplying C/A code with in-phase (I) and quadrature-phase (Q) of Doppler frequency shifts and then multiplied by the sensing matrix, as illustrated in Figure 5-7. Moreover, these dictionaries are generated once and stored in the memory to reduce the locally generated signal overhead.

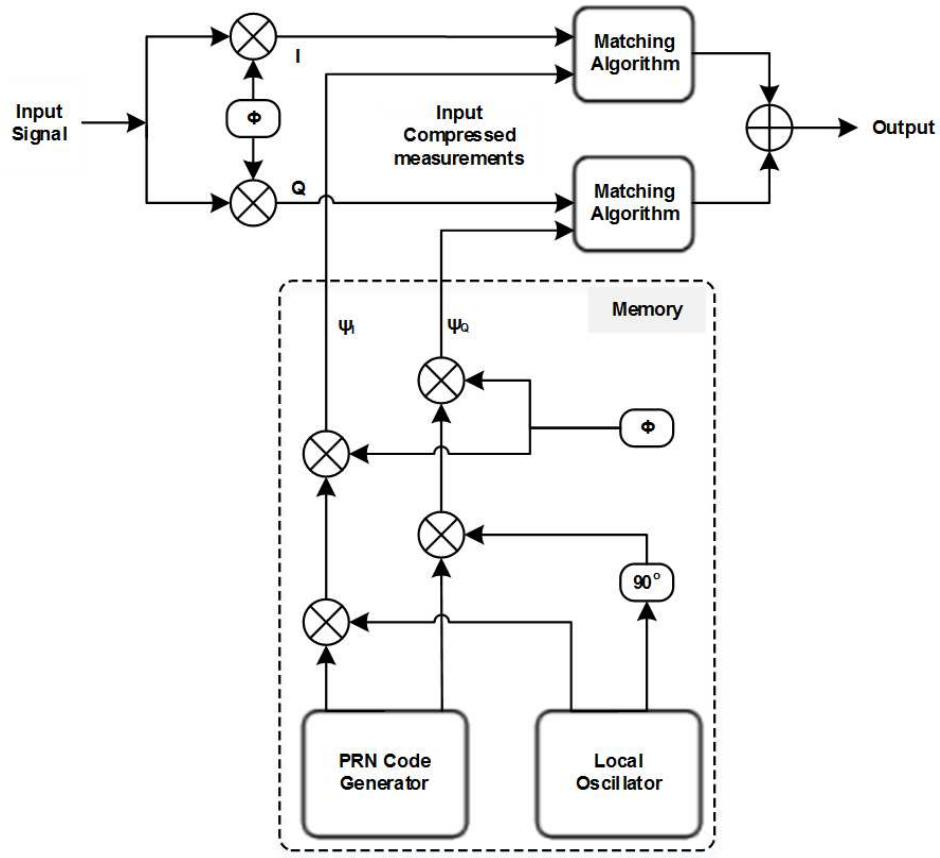


Figure 5-7 S-GPS solution

The S-GPS acquisition process is accomplished by multiplying the received GPS signal with the Gaussian sensing matrix to compress the signal and transformed to CS-domain. Then the compressed outputs and the I and the Q dictionaries are solved based on Second-Order-Cone-Program. The last stage takes the absolute value of the resultant I and Q and then summed to find the highest correlation. In this implementation decreasing the compression factor, i.e. reducing the row number in the sensing matrix, will reduce the detection probability. Even though the S-GPS has a good acquisition quality it is at the expense of large memory requirements, I and Q dictionaries, and costly searching process that can be overcome by combining these dictionaries to construct a single complex dictionary and this leads to reducing the cost of searching process.

Regarding the compression concept, the GPS signals are acquired based on the traditional implementation using compress samples in order to reduce the acquisition time [72]. The algorithm depends on the average samples or down samples. The sampling frequency that is employed in this implementation is 5.115 MHz, which

means each 1ms represents 5115 samples or points. These points are compressed or down-sampled to 1024 points for each 1ms. The algorithm uses 2ms to acquire the GPS signal and that leads to having 2048 points after down-sampled. These 2ms are converted to the frequency domain using FFT transform. At the same time, the locally 1ms generated signal compresses to 1024 points and then padding 1024 zeroes to create 2ms length as the received signal. After that, the generated 2ms is converted to the frequency domain, and the complex conjugate is used to multiply with the compressed received points. As a standard process of the FFT-search algorithm, the IFFT is placed after this multiplication. Then the output points are divided into two parts, and the second part is neglected while the first is checked with a certain threshold. In this algorithm the authors pointed out that if there is a bit transition in the selected/tested 2ms then the probability of detection will be reduced.

5.3 The DCSR structure

The use of CS technique to acquire GPS signals saves processing time and memory resources when compared with hardware or software receivers. The DCSR proposes; on one hand, a dynamic CS-based acquisition for GPS signals, and on the other hand to reduce the hardware complexity of the CMS solution and the software complexity for the GCS-1 solution.

The main idea of having a dynamic implementation instead of using a fixed size sensing channels and fixed number of correlators in the measurement matrix, is that DCSR dynamically changes the number and the size of the required Channels/Correlators according to the received GPS signal power during acquisition process. This adaptive solution offers better fix capability when the GPS receiver is located in harsh signal environment, or it will save valuable processing/decoding time (battery power, especially for Smartphones) when the receiver is outdoors. A feedback loop is devised to control the sensing channel number and resize the measurement matrix. Furthermore, such CS solution uses a fixed size measurement matrix chosen to offer a compromise between the processing overhead and signals acquisition success level.

In this chapter, the mathematical model of the transmitted and received GPS signal are presented in equations (5.7) and (5.8) represent the received GPS signal.

$$y_k(t) = \sum \sqrt{2P}\,[C_k(t)D_k(t)\cos(2\pi F_{L1}t)] \qquad \textbf{5.7}$$

where the $y_k(t)$ represent the signals transmitted from k^{th} satellites, , P is the power of the signal, $C_k(t)$ is the C\A assigned to this satellite, $D_k(t)$ is the navigation data sequence and F_{L1} is the carrier frequency of L1.

$$x_k(t) = AC_k(t-\tau)D_k(t-\tau)e^{j(2\pi(F_{L1}-f_{d1})t+\theta)} + n(t) \qquad \textbf{5.8}$$

where the $x_k(t)$ represent the received signals, A represents the received power, τ is the code phase delay, f_d is the Doppler frequency shift, θ represents the carrier phase and $n_x(t)$ represent a complex AWGN.

The rest of the sections shall describe how we overcome the hardware complexity of the CMS solution by using Xampling technique rather than complex multi-channels samplers in Section 5.3.1, Section 5.3.2 illustrates the acquisition process, which overcomes the software overhead in the GCS solution, and Section 5.3.3 shows the dynamism process that makes this solution adaptive in different environments.

5.3.1 DCSR's sampling and sensing procedure

In effect, the Xampling technique is proven to be more reliable for sampling signals below Nyquist rate, as well as being easy to implement [61]. This is accomplished because it produces a compressed signal that is suitable to be used with the CS technique. Therefore, to achieve the desired acquisition accuracy, this technique has been adopted in our DCSR solution to sample and sense the received GPS signal.

As shown in Figure 5-8, the received GPS signals are multiplied by the square periodic waveform $\phi_k(t)$ using a number of channels simultaneously; these waveforms are equivalent to the binary sensing matrix that will be used, together with the bank-correlators/dictionary matrix Θ, to acquire the GPS signal. The rate of the square wave (F_p) is equal to the sampling rate (F_s), which corresponds to the bandwidth of the received signal, where ($F_p = F_s \approx$ Bandwidth) [62]. The Xampling samples the received signals according to the maximum bandwidth of the multiband signals. In our case, only the GPS signals are used, and so the sample rate is equal to

2.046MHz. The multiplication outputs represent a linear combination of the frequency shift copies of F_p, then low-pass filters are used to filter the baseband signals, and the output is sampled at a low rate corresponding to the signal bandwidth.

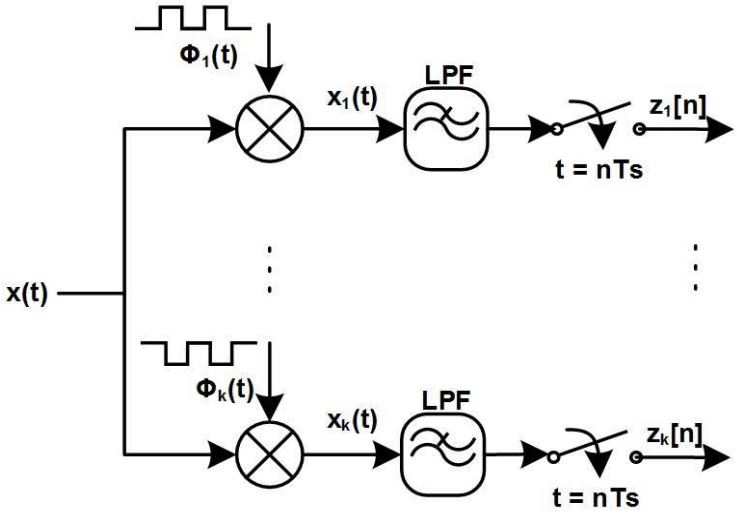


Figure 5-8 Multi-channel sampling

$$Z_k(\omega) = \sum_{k\epsilon K} \phi_k(k\omega_p)\, x_k(\omega), \quad \omega\epsilon\omega_s \qquad \textbf{5.9}$$

5.3.2 DCSR's acquisition process

In order to acquire the GPS signals, we first construct the frame V from the jointly sparse over time vectors z[n], as illustrated in equations (5.10) and (5.11), by using the CTF block, as shown in Figure 5-9.

$$z[n]=[z_1[n], \ldots, z_k[n]]^T \qquad \textbf{5.10}$$

$$V=\sum_n z[n]z^H[n] \qquad \textbf{5.11}$$

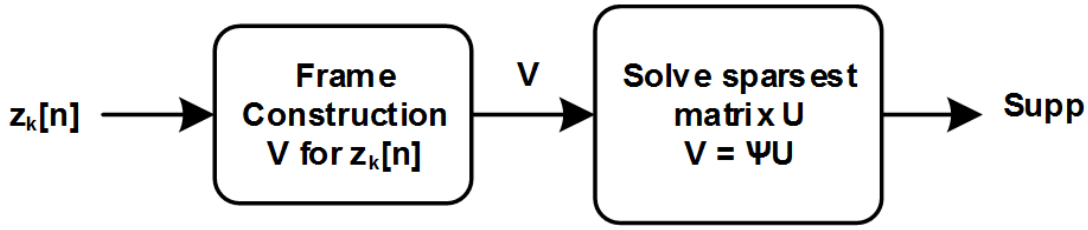


Figure 5-9 Continuous to finite block

The sparse matrix U in equation (5.12) can be solved using the "pursuit algorithm" such as the OMP algorithm to find the support values of the sparse matrix [64].

$$V=\psi U \qquad \textbf{5.12}$$

where ψ *is the measurement matrix* $\psi(\omega) \triangleq [\psi_1(\omega), \dots, \psi_k(\omega)]^T$*, which and consists of the sensing matrix* $\phi(\omega)$ *and the bank-correlators matrix* $\Theta(\omega)$.

The measurement matrix can be formulated as:

$$\psi_{k,[l,p,q]}(t) = \sum_{l=1}^{L} \sum_{p\in P} \sum_{q\in Q} \phi_k(t)\Theta_{l,p,q}(t) \qquad \textbf{5.13}$$

where L, P and Q are the numbers of satellites, "search step of code phase delay" (at half chip resolution) and the "Doppler frequency shift" at 500 Hz steps respectively.

and

$$[\Theta]_{l,p,q}=C_l(t-pT_c)e^{j(2\pi(F_d+q)T_c} \qquad \textbf{5.14}$$

The bank-correlators matrix is the (L x P x Q) columns vectors, where each column represents an expected shift of the code phase delay with the Doppler frequency shift of each GPS satellite signal. After the completion of the support recovery of the sparse matrix U, the right code phase delay and the Doppler frequency shift of the acquired satellites ($supp_{l,p,q}$) are determined by calculating the maximum values of the column vectors of U (S = supp(U)).

5.3.3 DCSR's dynamic function

Acquiring GPS signals based CMS solution suffers from high processing requirements if acquiring GPS signals in the harsh environment, else the GPS signals lock is lost. In other words, using larger number of channels will produce higher rate of acquired satellite signals, but that means these channels will have to be used in good signal areas too, which is a waste of efforts. Our DCSR is designed to overcome this drawback of using a fixed number of channels by controlling the deployed number of channels and the size of used correlations on the go as needed determined by the actual signal strength.

In fact, the DCSR will resize both sensing and measurement matrices by using the feedback control as shown in Figure 5-10. The feedback-controller continuously measures the power of the received signal to determine the signal complexity level. The received power is calculated based on equation (5.15), more details in [6]:

$$P_r = P_t G_t G_a / 4\pi\rho^2 \qquad \textbf{5.15}$$

where P_r , P_t , G_t , G_a, ρ are the power of the received GPS signal, transmit power, satellite antenna gain (10.2 to 12.3 dB), the effective area of the receiver antenna, see equation (5.16) and measured pseudoranges respectively.

$$G_a = \lambda^2 / 4\pi \quad \lambda \text{ is the wavelength} = 0.1903 \text{ m} \qquad \textbf{5.16}$$

Once the power of the GPS signals is calculated, we can easily determine the carrier to noise ratio (C/N) or SNR [73].

$$C/N_o = P_{r(dBW)} - N_{TH(dBW)} \qquad \textbf{5.17}$$

where C/N_o is the nominal carrier to noise ratio in dBW, $P_{r(dBW)}$ is the received power in dBW ($10log(P_r \text{ in } W)$) and N_{TH} is the thermal noise power (-204 dBW).

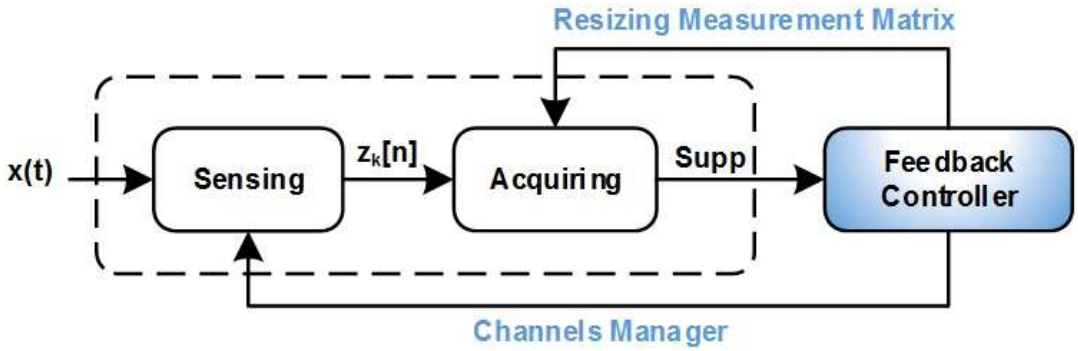


Figure 5-10 Dynamic GPS signal acquisition

From the nominal carrier to noise ratio we can determine the actual C/N, as illustrated in equation (5.18):

$$C/N = C/N_o - N_F \qquad \textbf{5.18}$$

where N_F is the cascade Noise Figure of this receiver.

The range of the SNR values of the received signals depends on the front-end bandwidth BW (dB-Hz), which is equal to (BW= 10log (F_B)), and the F_B is a filter bandwidth:

$$SNR = C/N - BW \qquad \textbf{5. 19}$$

More precisely, our dynamic design is based on three levels of measurement complexity, as illustrated in Figure 5-11. High measurements levels are used in bad reception areas and so 600 channels are selected to compensate for the sensing and measurements of these weak signals. While when outdoors, our simulations show that using only 240 channels is sufficient to acquire the signals (represents a low measurements level). Finally, 480 channels are chosen as a middle case to help compare the DCSR and CMS algorithms performance. Our simulations show that these three channel-number selection-levels make the DSCR much more efficient than the CMS by using only necessary resources depending on the reception environment.

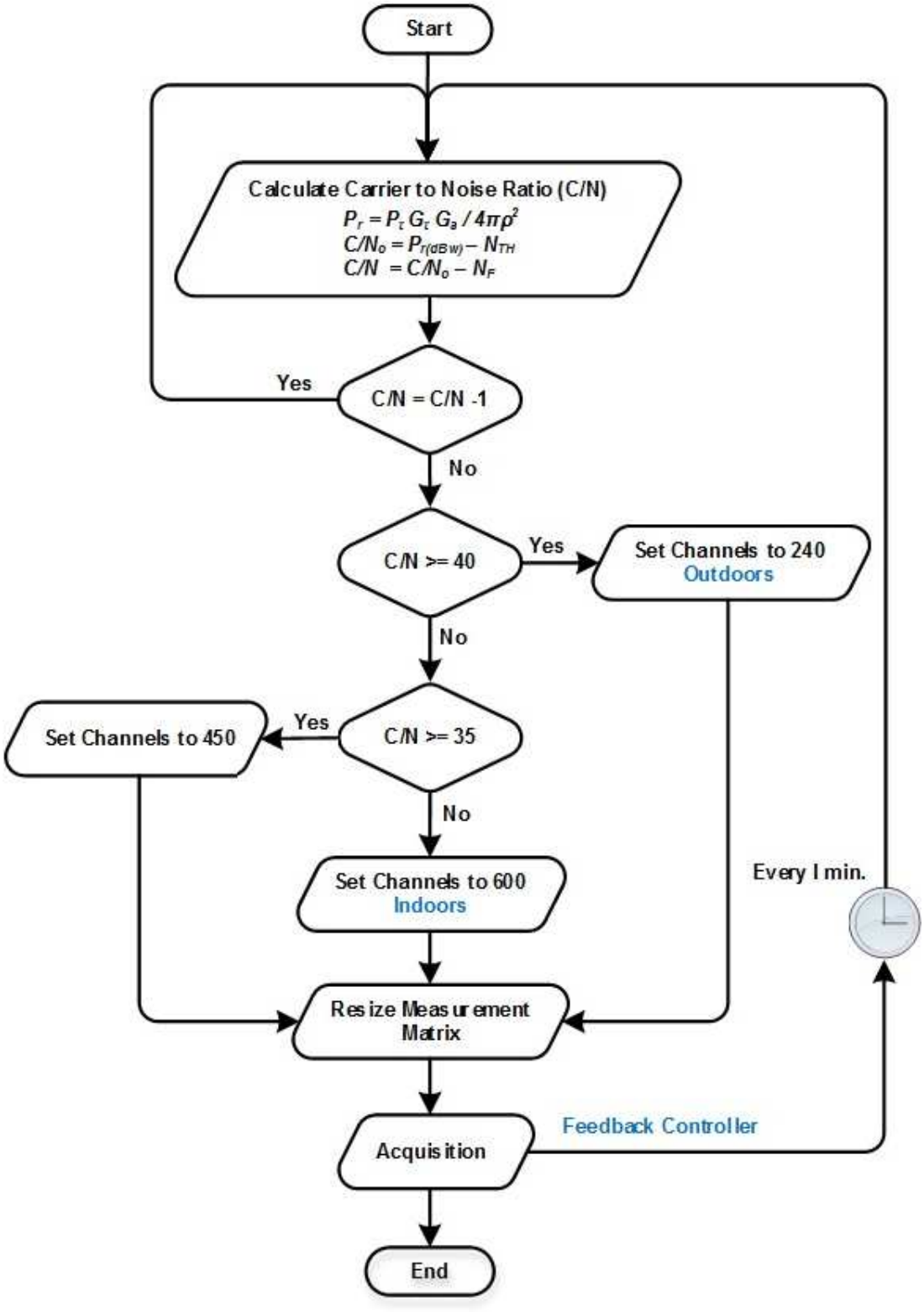


Figure 5-11 Determining number of channels flowchart

Furthermore, the design of the measurement matrix $\psi_{k,[l,p,q]}(t)$ makes the algorithm more powerful because each element in the measurement matrix contains all information of a single satellite, see equation (5.13), in contrast with the design of the measurement matrix in CMS solution. So, when the number of channels is increased or decreased, the information of the satellite in the measurement matrix does not get affected, and only the sensing property of the measurement matrix will be changed.

5.4 DCSR simulation setup and results

The DCSR algorithm performance has been compared with that used by the CMS solution. Our dynamic scenario for the GPS signal, which has C/N value varies according to the signal environment, is listed in Table 5-1. Note that the C/N is ranged from 50dB-Hz to 25dB-Hz, the bandwidth of the low-pass filter is set at 2 MHz with a 3dB cascade noise figure.

Table 5-1 Open-Sky and Multipath Scenarios

C\N dB-Hz	**LOS & Multipath signals Scenarios**	
	Number of received signals	***Scenario name***
50	5 LOS signals	S1
45 to 40	5 LOS signals and one multipath for each signal	S2
35	5 LOS signals and two multipath for each signal	SM
30 to 25	15 multipath signals from 5 satellites	M

The scenarios in Table 5-1 start with an open sky reception area having LOS signals only (S1). These signals are then degraded, and multipath signals are added to the existing LOS signals. We consider two signal degradations as shown in (S2) & (SM). The urban area scenario is represented by only multipath signals (M). Since the algorithms are based on acquiring weak GPS signal, the length of tested signals is chosen to be 20ms. This will increase the acquisition sensitivity by around 13 dB

more than the normal methods (1ms length), where the gain obtained by the 20ms acquisition is around 46 dB compared with the 33 dB gain obtained from the 1ms processing ($GdB = 20\log\sqrt{n}$).

The bank-correlators matrix has a half chip resolution for code phase delay; 500 Hz frequency search steps with Doppler frequency range ±4 KHz, and the maximum channel delay distribution of the C/A code is 30Tc to reduce the implementation complexity. Additionally, to reduce the computational complexity as much as possible, the "Approximate Conjugate Direction Gradient Pursuit (ADGP)" algorithm [74] is used rather than the OMP algorithm. Our simulations show that both algorithms have the same performance overall signal conditions.

The simulated dynamic scenario is illustrated in Figure 5-12-a, where the C/N values represent the various received signal conditions. The dynamic scenario is changed gradually to simulate Smartphone movements. The results show that the number of acquired satellites signals by our DCSR is almost constant at all signal strength cases, while the CMS solution has failed to acquire signals in low signal conditions as shown in Figure 5-12-b.

Figure 5-12-c shows how the DCSR has dynamically switched the number of channels as dictated by the received signal condition. So, in order to acquire the GPS signals in harsh environment, the DCSR switches to the maximum number of channels, while in ideal environment "open-sky", the DCSR uses the minimum number of channels to save power and reduce processing time. The results, therefore, show the adaptability of the DCSR to work in various environments to maintain the performance of the GPS receiver while saving processing time and battery power.

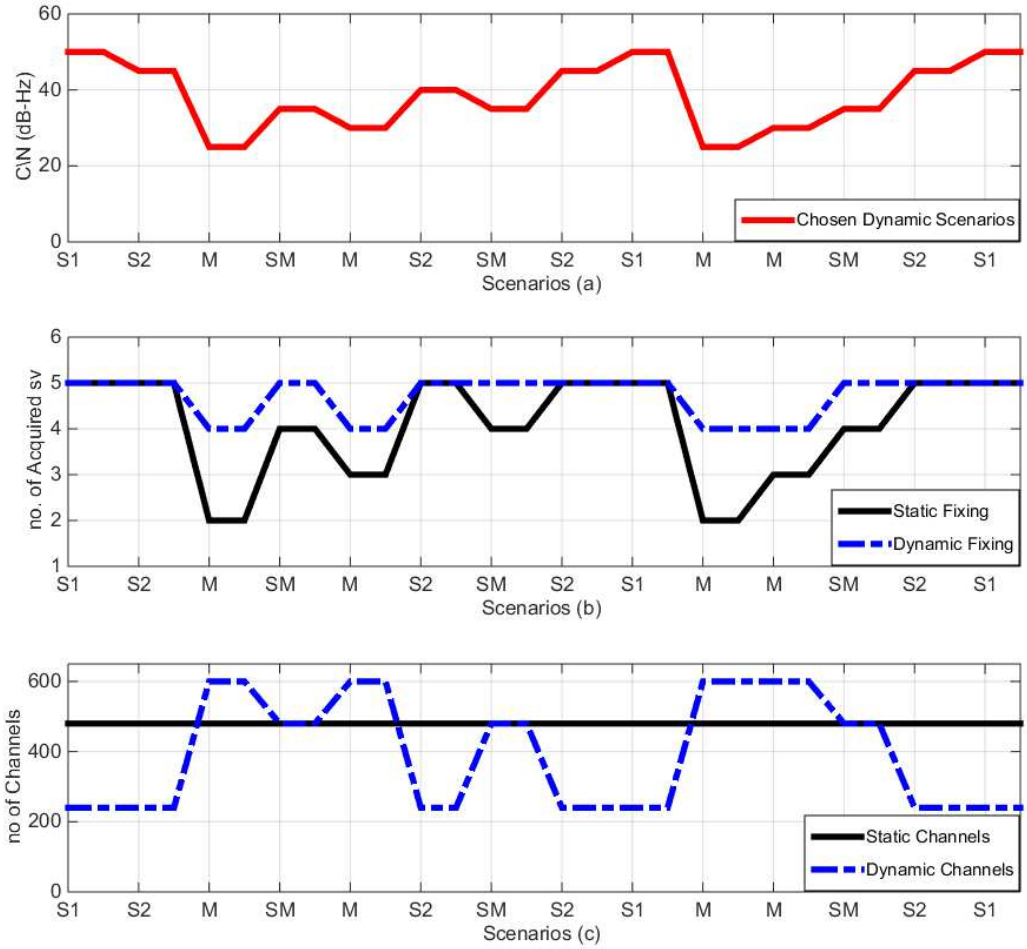


Figure 5-12 Dynamic channels vs. fix number of channel performance, (a) the dynamic scenario, (b) number of satellite signals acquired from CMS & DCSR algorithms and (c) number of channels of CMS & DCSR algorithms

5.5 Conclusion on the DCSR

In this DCSR solution a novel dynamic acquisition is implemented based on CS technique. The DCSR solution can reduce processing time and so minimise the power consumption required by a GPS receiver to acquire signals in outdoors. This is accomplished by dynamically altering the number of measurements and the required number of sparse channels to suit the actual signal strength. Also, the DCSR maintains lock of the available signals in difficult signal conditions by using an extra number of channels to compensate the measurements. In addition, the design of the dictionary matrix enables the measurement matrix to change its size without affecting

the signal compression and integrity. Moreover, applying the ADGP algorithm to recover the sparse signal will also reduce the computational complexity.

My supervisor Dr Ihsan Lami has presented the DCSR implementation in the Microwaves, Communications, Antennas and Electronics Systems (COMCAS), IEEE International Conference, on Wednesday, October 23, 2013.

5.6 The GCSR structure

The GPS correlation process becomes very intensive when the signal power is degraded, depending on the environment wherein the clear outdoors signal power is around (-125 to -130) dBm, and it can be around -155dBm in light indoors. If sampling is performed at the Nyquist recommended rates, then a weak signal will make the receiver thrash for all the 20ms available to try to find the signal. The CS is adopted to either use the 20ms more efficiently by giving the Doppler/Code matching process a better chance to find the signal and/or to do this matching at much less sampling and correlations without impacting the resolution/sensitivity of the acquired signals [57].

Our GCSR focuses; on one hand on acquiring the GPS signal since it becomes a drain especially during cold start and in harsh signal environments, and on other hand to enhance the sparseness of the GPS received signal by using deterministic waveforms such the Hadamard matrix or Jacket matrix in the sensing stage rather than any periodic waveform.

In this section we shall illustrate the enhancing that is made to the Xampling technique, and then we shall explain the acquisition process. After that in Section 5.7 we shall demonstrate the experiment result.

I. GCSR sensing enhancement

One of the GCSR aims is to enhance the sensing by using the same resources that are used in this technique, i.e. there is no resource overhead. The sampling process is started by multiplying the received analogue signal x(t) (see equation 5.8) by a periodic waveform. The number of channels “m” is pre-determined to generate z[n] baseband vectors that represent unique values of the information band at that particular time. To check the sensing in our GCSR, number of periodic waveforms such as square, saw-tooth (Ramp) and sinusoidal waveforms were tested. However, we concluded that deterministic waveforms such as the Hadamard or the Jacket blocks offered better performance when we construct the signals. The Jacket block [75] is an extension of the Hadamard matrix, and it is a centre-weighted matrix, where the inverse matrix can be obtained from closed form entity as shown below.

i.e. either block inverse or element-wise inverse are possible as in equation (5.20) and (5.21).

$$[J]_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -c & c & -1 \\ 1 & c & -c & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad \textbf{5.20}$$

and the inverse is:

$$[J]_4^{-1} = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1/c & 1/c & -1 \\ 1 & 1/c & -1/c & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad \textbf{5.21}$$

where [J]4 a 4x4 Jacket matrix, and c is the non-zero arbitrary number, which represents a weighted factor.

For "m", the larger the number of channels selected, the more this will influence the resolution of the sensing. Therefore, based on our experiments, we chose m=480 for normal GPS signal and m=600 for acquiring low sensitivity signals.

The main differences between GCSR and DCSR implementations are:

1. The periodic waveform used in our GCSR is a deterministic waveform rather than square waveform that is employed in our DCSR.
2. The number of sensing channels is fixed and not dynamic.

II. GCSR acquisition process

The acquisition process based GCSR implementation is same as DCSR acquisition process (see cp. Section 5.3.2). Where, the resultant sampled vectors are then converted to a frame "V" by using CTF block as illustrated in equations (5.10 and 5.11) and as shown in Figure 5-9. In our GCSR, we have used the OMP algorithm to find the dictionary elements (support values) of the measurement matrix ψ. After the completion of the support recovery of the sparse matrix U, the right code phase delay and Doppler frequency shift of acquired satellites ($supp_{l,p,q}$) are determined by calculating the maximum values of the column vectors of U matrix ($supp_{l,p,q}$ = supp(U)) as adopted in our DCSR.

5.7 GCSR experimental results

In this particular GCSR method, the results are divided into two parts. The first part shows the performance of using deterministic waveform against the random periodic waveforms. The second part of the result is focused on the effect of the CS processing on the signal reconstruction.

I. GCSR acquisition performance

The GPS signals are simulated using MATLAB. The front-end is designed to have 2 MHz bandwidth of the low-pass filter, 3 dB cascade noise figure and the nominal power of the received signal is set to -125 dBm. DCSR's scenarios have been adopted to prove the performance of GCSR implementation, and as illustrated in Table 5-1.

Considering the acquisition and the computational performance, we have compared our receiver with the CMS solution. Choosing to simulate a 20ms length signal will add around 13dB gain to the C/N of the received signal at the sample rate of 2.046MHz. The acquisition rate can be improved by increasing the number of channels representing the rows in the sensing matrix. This will improve the reliability of the sampling signal, which results in detecting weak signals. To illustrate this, two types of channels are used; 480-channels for comparison with CMS method and 600 channels to demonstrate the performance of our GCSR, as shown in Figure 5-13.

The deterministic orthogonal waveforms "Hadamard or Jacket matrices" used to sample the received signal and to construct the measurement matrix increases the acquisition rate by 20% more than when using a square waveform. The gain obtained is equal to 3dB-Hz that is clearly noticeable in high noise situations. This is achieved because of the low coherence between our GPS dictionary matrix and the Jacket/Hadamard matrix, as a result of the perfect orthogonality of these matrices.

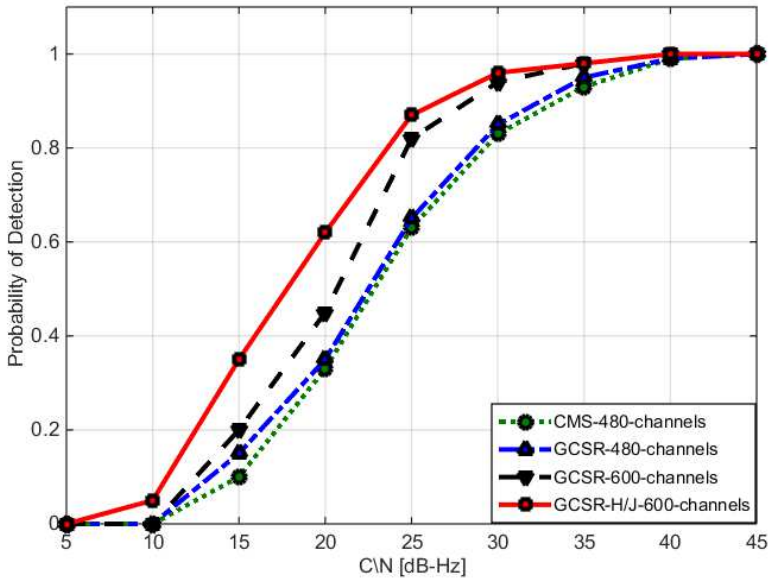


Figure 5-13 GCSR Acquisition rate for different channels and waveforms

To ensure performance stability of the GCSR's selected waveforms, these seven different C/Ns' scenarios were performed based on Monte Carlo simulations with 100 runs for each waveform, i.e. (7 scenarios x 100 times x 5 waveforms) as shown in Figure 5-14. As shown, the use of the Jacket/Hadamard matrix is more stable than using others periodic waveforms versus different scenarios.

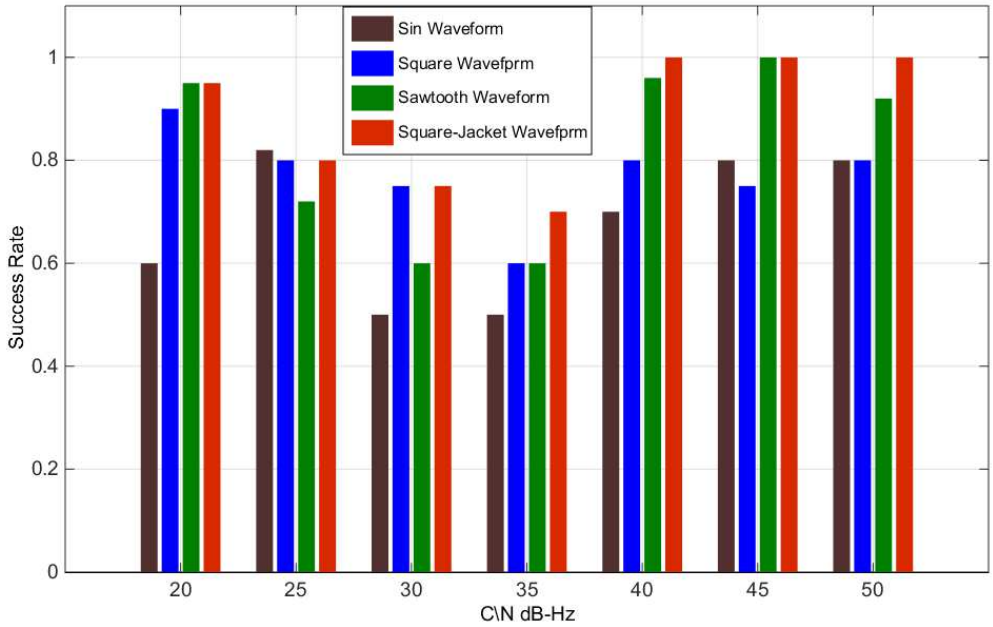


Figure 5-14 Success rate with 100 runs for each of 5 waveforms and for each of 7 scenarios

To analyse the implementation and the performance of our GCSR and CMS implementations, the following points summarise this analysis:

1- The GCSR has a simple implementation as the CMS method uses multichannel random samplers to sample the GPS signal; using the measurement matrix, which means complex hardware and pre-processing to construct the measurement matrix.

2- GCSR achieves better signal matching. This is because finding the dictionary elements in the CMS method depends on the matching between the sensing matrix and the frame V, whereas, our GCSR method depends on matching between the measurement matrix and the frame V. This matching can produce more reliable correlation since the sensing matrix multiplies by both the received signal and the generated dictionary.

II. GCSR tracking performance

For a further check on the quality of acquisition, our GCSR has been also compared with a traditional GPS receiver as shown in Figure 5-15. To do this, the resultant GCSR signals are fed into a GPS decoder to recover the actual acquired SV navigation message.

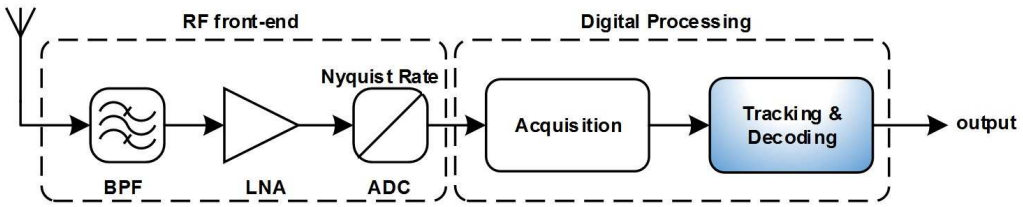


Figure 5-15 Traditional GPS receiver

The bit error rate (BER) and the error vector magnitude (EVM) are used for evaluating and analysing the effect of GCSR acquired signals' decoded messages. The BER performance of both the GCSR receiver and the traditional GPS receiver, as a function of the normalize C/N, are shown in Figure 5-16. The results illustrate that our GCSR has a small degradation of BER of about 10% because the bit representation in our GCSR has less number of samples than the traditional receiver. To clarify that, the number of samples in each bit in our GCSR equal to (2 x 1023 x 20) while in the traditional receiver equal to (4 x

1023 x 20), where the 1st number refers to the number of samples in each chip, the 2nd number is the length of the GPS-C/A code and the 3rd number represents the number of the GPS-C/A code in each bit. In fact, this little degradation is expected as an overhead of using the CS technique [76].

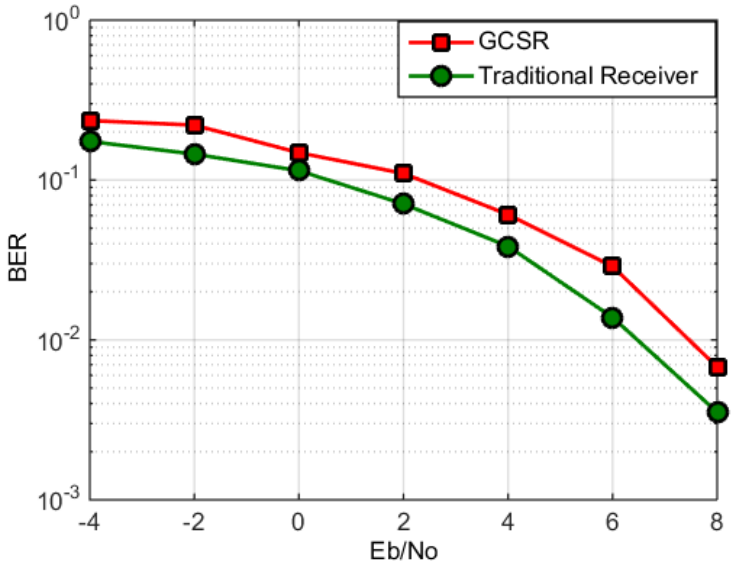


Figure 5-16 Bit error rate versus energy per bit to noise power spectral density

The EVM analysis defines the difference between the estimated phase and amplitude values of the demodulated/decoded symbol with the values of the actual received symbol. This will show whether the GCSR has preserved the distance between any pair of samples (phases and amplitudes) during compressing and reconstruction the GPS signals. Figure 5-17 shows the values of EVM of the estimated phase and amplitude of CS against the traditional GPS receiver. It displays a bit of distortion in estimating the phases & amplitudes in the GCSR. However, this distortion is acceptable in the application of the CS technique [76].

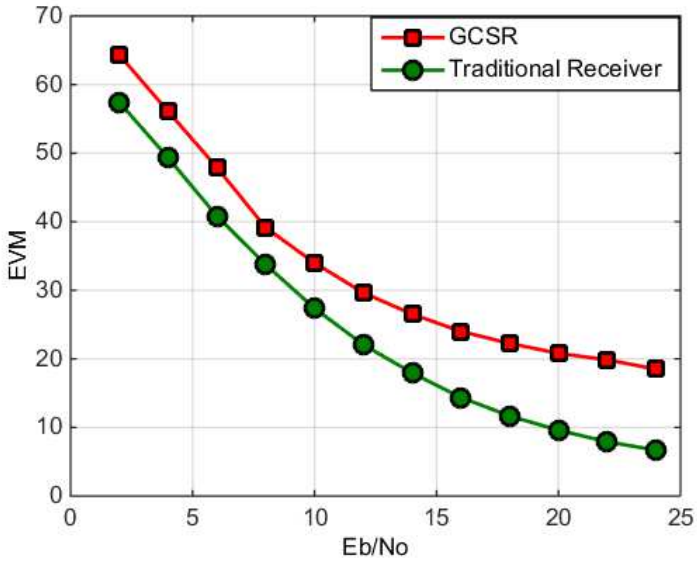


Figure 5-17 Error vector magnitude curves (RMS)

The final proof of the GCSR implementation is to measure the stability of the PLL discriminator in the decoder part. The basic work of the PLL is to recover/track the actual phase, amplitude and frequency of the received signal. The traditional receiver and our GCSR have almost the same steady-state values while running the simulation of tracking one-second of GPS data, as shown in Figure 5-18.

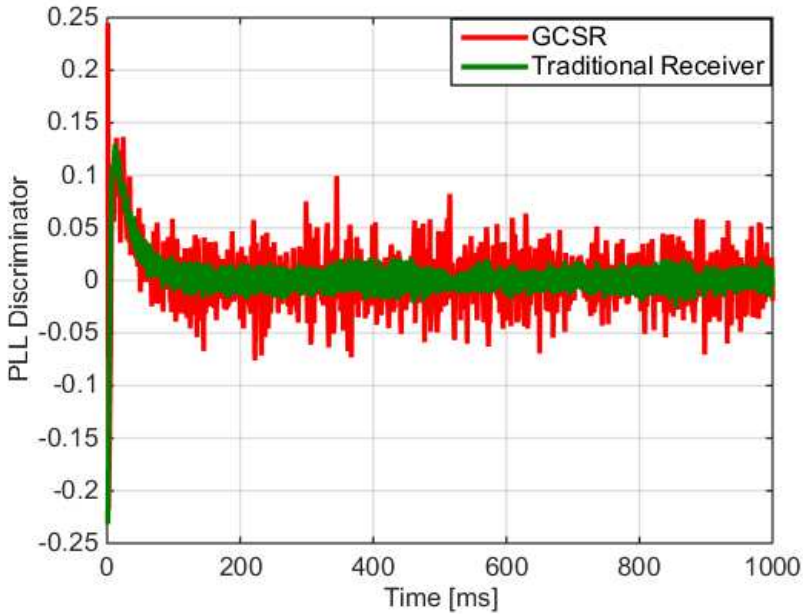


Figure 5-18 The steady state of PLL discriminator

5.8 Conclusion on the GCSR

The enhancements adopted in the GCSR implementation do simplify the front-end design as well as introduce better performance when compared with other CS-based or traditional GPS receiver solutions.

The higher acquisition rate is achieved by using a deterministic waveform generator to sparse the received signal such as the Hadamard or the Jacket matrices. This is because these matrices have the best orthogonality. As a result, the sensing process has been enhanced as well as improving the signal sampling.

The GPS signals have been sampled below the Nyquist rate and equal to the information band. Acquiring GPS signals therefore can be accomplished with fewer correlations as the CS process is now transferred from matching the length of the whole signal samples number to matching whole rows/channels of the sensing matrix. Increasing the number of rows/channels will increase the acquisition rate proportionately.

Reconstructing the signal based on our implementation is simpler than others because we have moved the measurement process to the DSP side while others process it on the Analogue side. Our test scenarios and analysis showed a slight phase distortion and amplitude degradation of the decoded signal; however the integrity of the received signal was maintained.

Chapter 6

Multi-GNSS Signals Acquisition using Decomposed Dictionary Matrix

We have pointed out in chapter 5 that the CS technique can be used for a software receiver to acquire GPS signal with less computational complexity than traditional implementations. Also, the experience gained thus far has motivated our research to capitalise on this saving in computational complexity to design a dual mode GNSS receiver that is capable of acquiring both the GPS-C/A-code and the Galileo-OS-code signals simultaneously. Initially, the requirement was to combine their dictionary matrices ($\Theta_{GPS} + \Theta_{Galileo}$) in a single combined matrix (Θ). This meant that the size of this combined matrix is too big, because the size of the $\Theta_{Galileo}$ matrix is four times the Θ_{GPS} matrix (the OS code length is equal to 4092 chips while the C/A code length is equal to 1023 chips), i.e. the size of the Θ matrix is equal to 10×10^6.

To explain how the size of the dictionary matrix is determined, we shall calculate the size of the Θ_{GPS} matrix, as an example. This matrix consists of the code-phase delay and the Doppler-frequency shift for each of these 24 SV (= SV x Code phase delay (P) x Doppler frequency shift (Q) = $24 \times 2046 \times 41 \approx 2 \times 10^6$), where the code phase is half chip resolution and the frequency search step is 500 Hz with range (±10 KHz). Consequently, combining the two matrices leads to an increase in the time of searching for each of the dictionary elements. Furthermore, the Θ_{GPS} is considered as an “overcomplete dictionary” or fat dictionary, as the number of columns basis (C) is much greater than the length of the tested GNSS signal (N) [77]. Therefore, it has become important to decompose the Θ matrix so as to make it manageable by reducing the search process.

6.1 Decomposition study

This section shall detail the simulation trails that led to decomposing of the Θ matrix. At the beginning of this study, we have tried with a single GNSS signal (GPS signal only). The idea was to exploit the GPS signal parameters and use them as a sensing matrix, as they satisfied the RIP condition (see cp. Section 5.1), and so to generate sparse vector. According to that we had two options, which are:

1. C/A codes sensing matrix with 0.5 Chip code phase resolution
2. Doppler frequency sensing matrix with 500Hz frequency resolution

Each one of these matrices (codes or frequencies) represents a deterministic matrix and they are orthogonal. To emphasize the orthogonality of both proposals equation (6.1) shows the normalised correlation or the inner product of the C/A codes ($\mathrm{C/A_{NC}}$).

$$\mathrm{C/A_{NC}} = \begin{cases} 1 & if\ the\ phase\ shift\ = 0\ Chip \\ 0.5 & if\ the\ phase\ shift\ = 0.5\ Chip \\ \approx 0 & if\ the\ phase\ shift\ \geq 1\ Chip \\ \approx 0 & if\ correlated\ with\ other\ SV's\ codes \end{cases} \quad \textbf{6.1}$$

Equation (6.2) illustrates the normalised correlation of the Doppler frequency shifts ($\mathrm{D_{NC}}$).

$$\mathrm{D_{NC}} = \begin{cases} 1 & if\ the\ frequency\ shift\ = 0Hz \\ 0 & otherwise \end{cases} \quad \textbf{6.2}$$

Firstly, we started with the C/A codes sensing matrix to test the new CS-framework.

1. The first stage of the CS acquisition takes the inner product between the received GPS signal and the C/A codes matrix to produce a sparse/compressed vector. According to equation (6.1), the output therefore includes at least 3 peaks/values if the received signal comprises only one GPS signal. Then it was found that this scenario is only applicable if the Doppler frequency shifts equal to 0Hz.
2. While, when we add Doppler frequency shift, for example equal 250Hz to the GPS signal, then these peaks are decreased in terms of numbers and amplitude.

3. For further testing, we increased the Doppler frequency shift to be 500Hz and that led to fading of all the peaks.

4. Then we realised that in each chip there are 1540 cycles of carrier frequency (1575.42MHz/1.023MHz), i.e. if the code phase resolution equals to 0.5 and if the Doppler frequency shift equals or is greater than 750Hz then theoretically the resultant correlation is zero.

In addition, using the C/A codes as a sensing matrix has results in a large numbers of rows, which is equal to (2 x 1023 x 24 = 49104 rows). Whereas, this number is equal to double the signal length when (4ms length is used at 6.138MHz sampling rate), and this is inconsistent with the CS concept. While, using Doppler matrix produces very good compression, however it too did not work because of the resultant weak correlation when there is a code or Doppler shift.

This strong association between the integration of the code and frequency in the GPS signal led to conducting the research in this particular area to devise new decomposed dictionary (Θ_D) matrix without affecting the signal integrity. The Θ_D matrix is designed by making the Doppler frequency shifts fixed for generating codes of all GPS+Galileo signals. Thus, the Θ_D matrix is represented as a bank of codes rather than a bank of correlators. In addition, using Θ_D matrix requires having input signal to the CS framework without Doppler frequency shift and two dimensions searching algorithm because Θ_D matrix contains only GNSS codes. Consequently, we have used Doppler channels as a pre-processing to strip the Doppler frequency shift, where these channels provide high frequency resolution, as explained in Section 6.3.2. Also we have modified the OMP algorithm to search for two dimensions by advising “2D-OMP” algorithm, as detailed in Section 6.3.3.2.

In this chapter the Θ_D matrix is can be applied for dual GNSS signals receiver like (GPS+Galileo), as presented in Section 6.3, and as described in Section 6.7, the Θ_D matrix is can be also utilised in single GNSS signal receiver, such as GPS signal. To illustrate the combined solution, the next section shall detail the multi-GNSS signal receiver implementations.

6.2 Combined multi-GNSS signals methods literature survey

Using multi-GNSS signals in a localisation algorithm can help mitigate the multipath effects in urban canyons, as more SVs comes in view, leading to improved availability, time to first fix and location accuracy especially in cold start scenarios. Therefore, integrating multi-GNSS signals in a single efficient acquisition/tracking process is valuable in saving processing time and power on Smartphones. Several solutions have been published for integration of the multi-GNSS signals, most of which need undesirable complexity (abundant number of correlators) and/or overhead (high sampling rate).

An example of these solutions is a hardware unit that has been designed to combine the acquisition of the GPS-C/A-code signal and the Galileo-OS-code signal [78]. As depicted in Figure 6-1 acquiring signals is accomplished by utilizing MF (coherent integration) and FFT search (non-coherent integration). This design provides considerable sensitivity for the weak signals by combining the coherent and non-coherent integrations.

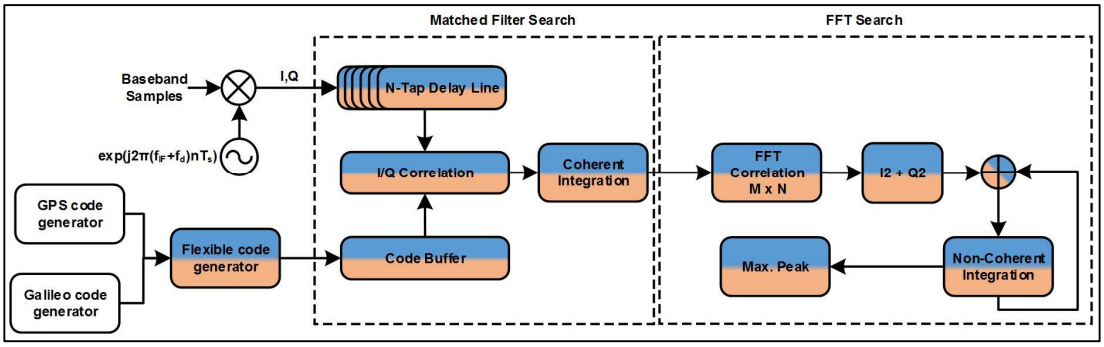


Figure 6-1 Hardware unit acquisition for GPS and Galileo signals

However, it acquires one GNSS signal at a time and the overheads of this unit are:

1. Highly parallel scanning for the cells area (code and frequency) in the first MF stage and long integration time in the FFT stage, where the dwell time is 4 times longer than the matched filter's time.
2. As regards acquiring Galileo signal, the acquisition is ambiguous because the chip search space is half chip resolution. And therefore, to overcome the ambiguity issue and to have the same correlation gain as the GPS signal, the

chip search space of the Galileo signal must be divided by 3. Consequently, a high sampling rate is required to satisfy proper chip search space.

3. This design becomes more complex due to combining two different engines the MF and the FFT search algorithms.

A serial search algorithm (time-domain implementation), as illustrated in Figure 6-2, was adopted to acquire the GPS-C/A code signal and the Galileo-OS code signal using side-by-side implementation [79]. This implementation was proposed to acquire weak GNSS signals and by relying on aiding the network that enables this implementation to determine all possible visible satellites with their estimation Doppler frequency shifts. This aiding, however, reduces the size of the frequency bin search but:

1. It still needs an abundant number of correlators "highly parallel processing" that accommodates all codes of the two GNSS signals.
2. Long integration time, where 30ms are used for the GPS signal and 200ms are employed for the Galileo signal.

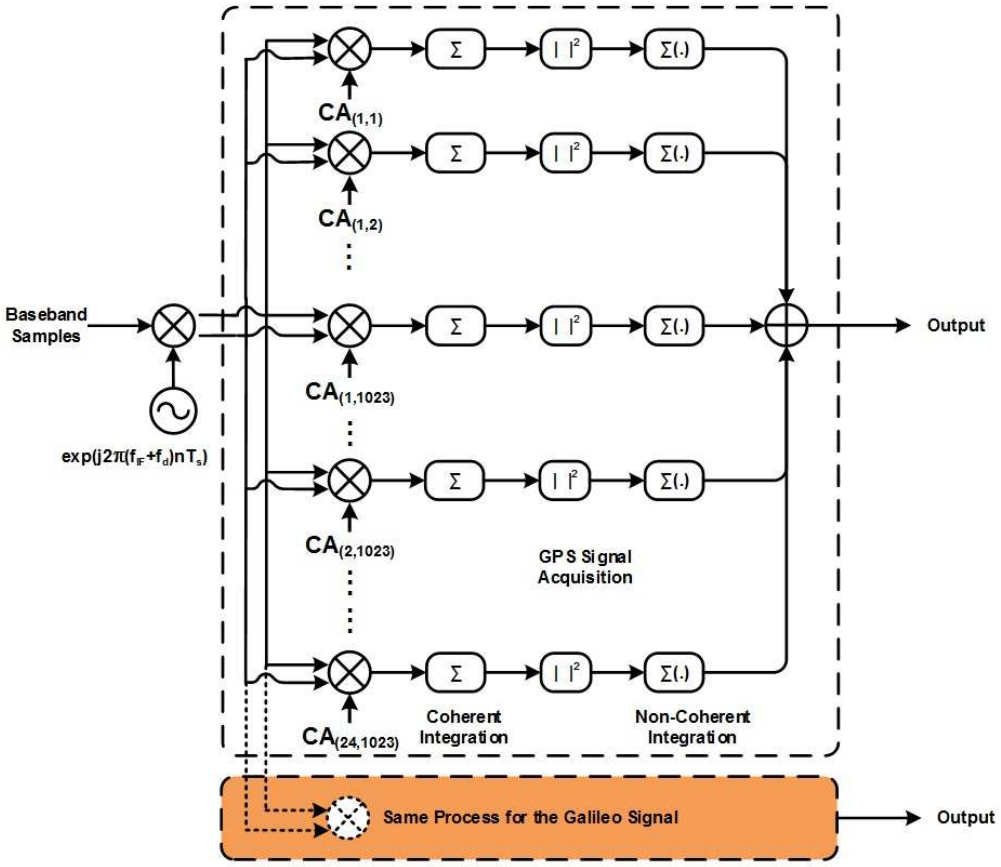


Figure 6-2 Side-by-side GPS and Galileo Signals acquisition

To understand the processing overhead and complexity of doing combined multi-GNSS signal implementation the rest of survey focuses on software GNSS receiver implementations rather than on hardware-correlator based solutions.

A reconfigurable multi-GNSS signal receiver that can acquire and track any (BPSK, BOC (m,n), CBOC, or TMBOC) signal at a time by means of software control was designed [80]. In this software receiver only the GNSS signals that broadcast in the two bands of frequencies 1150-1310 MHz (for E5, L5, L2 and E6 GNSS signals) and 1550-1610MHz (for E1 and L1 GNSS signals) were acquired based on 137.5MHz sampling rate. Depending on the chosen signal to be processed at the time, various number of correlation channels or various kinds of search algorithms are deployed such as using a single correlation channel for the BPSK signals or the Data or the Pilot signals, while another configuration might use two correlation channels for the Data & Pilot signals and unambiguous methods like DSB or BPSK-Like method. Also, different discriminators can be reconfigured or selected depending on the target use of the receiver. Again, this receiver does process more than one signal concurrently. In addition, it needs massive processing especially with that sampling frequency. It would be remarkable if there was such a range of sampling rate at least to be worked per requirement, because it is unprofitable to sample signal that has 2MHz bandwidth according to other GNSS signals that have 24MHz or 50MHz bandwidth.

The limitation of acquiring a single GNSS signal at a time has been overcome by combining the codes of multi-GNSS signals in a dual GPS-C/A-code signal and the Galileo-OS-code signal acquisition solution. This is achieved by generating a PRN code that contains the sum of two or more GNSS codes if these codes have the same synchronization property (chipping rate) [81]. The resultant code is then multiplied by the local carrier frequency with a specific Doppler frequency shift to construct a bank of correlators. The acquisition is then accomplished in two stages. The first stage determines the highest correlation between the generated bank and the received signal using a serial search engine. The second stage firstly identifies the satellite ID then it performs parallel correlation (parallel search engine) to estimate the code phase delay and Doppler frequency shifts for the identified satellite. This method does acquire multi-GNSS signal simultaneously but it also complicates the acquisition process by joining two search engines, the serial search and parallel search.

Another GNSS receiver implementation was proposed to take advantage of the interoperable properties between the GPS, GLONASS and Galileo signals, such as sharing the same bands, i.e. L1-E1 and L5-E5 bands [82]. The number of required correlators to track the BOC signals that have subcarrier frequency equal to the chipping rate is same as the number of correlators used in the BPSK signal. For example, if the number of correlators to track the GPS-BPSK signal is 3 (early, prompt and late) then the same number can be deployed for the Galileo-CBOC signal because the rates of chipping to the subcarrier frequency are equal. While for the other BOC signals such as GPS-M code signal that has high subcarrier frequency, then the required correlators are proportional to the BOC order. This means, if the BOC order is 2 then the required correlators are equal to twice the BPSK's correlators. Moreover, for the other modulation techniques like QPSK modulation, the required correlators are double the BPSK signal because it is processed as two decomposed BPSK signals. However, to combine all these GNSS signals in a single tracking unit requires a high sampling rate (40MHz) to accommodate the wideband signals such as GPS-L5 and Galile-E5 signals. In other words, the processing time to track the GPS-L1 signal is now 10 times longer than if it is sampled at 4MHz.

The combined solution that targets signals broadcasting from the same system like GPS signals has some advantages, such as compensating the delay in the code phase estimates the Doppler shift of the other signal. Nevertheless, there is no increase in the number of satellites or any chance to have different distributions by integrating these signals because they have the same constellation. A combined GPS L1-C/A and L2-C signals was proposed to enhance error correction in the entire system [83]. This work was developed to acquire and track both signals. Various acquisition implementations were designed and all of them relied on correlating the signals codes separately and combining them at the end of the correlation. For the tracking implementation, Kalman filter was used to estimate the error caused by the ionosphere, troposphere and the time delay of the instruments biases.

Regarding the CS solutions, the GCS-2 solution is the only CS-based solution that considers the BOC signal acquisition. Nevertheless, it cannot be applied to the current BOC(1,1) signals like Galileo-OS or L1C signals, unless it modifies the size of the banks of correlators and yet it only acquires one signal at a time [69].

On the other hand, determining the fine frequency is essential in the conventional GNSS receiver to reduce the phase error between the carrier frequency of the received and the locally generated signals, and practically it is located after acquisition process. Our implementation provides more accurate frequency estimation due to using the high resolution of the Doppler channels that can overcome this processing stage.

A fine-frequency acquisition method based on circular correlation and the FFT-search was proposed to prepare a 1Hz fine frequency error to the tracking stage [84]. This 1Hz fine frequency is accomplished when the C/N of the received signal is above 32dB-Hz. In a typical solution, if the coherent integration time is equal to 1ms, then the frequency bin search step is 333Hz [29], i.e. when the 1ms is used to acquire GPS signal the frequency error is around ±333Hz. The process of the fine frequency calculation is mostly similar to the early-late tracking process. Where in this method three channels are constructed, one in the middle/prompt “P” and the other two are located in the left “L” and in the right “R” with frequency space equal to 666Hz. This construction is more likely the space between the early and late correlators in the tracking code phase delay (DLL). These channels are used in three discriminators to find the fine frequency; the best discriminator adopted in this method is:

Fine frequency = G_L-G_R/G_P, where G represents the amplitude of the power of non-coherent integration.

6.3 The CSSR implementation

CSSR is designed to overcome the reviewed undesirable complexity and the processing overhead, as well as acquiring two GNSS signals simultaneously. Figure 6-3, shows the block diagram of our CSSR implementation, and the following sections are a detailed explanation of the 4 CSSR implementation stages; receiving and sampling stage, removing the subcarrier frequency effect stage to convert the BOC signal to the BPSK like signal, generating non-Doppler shift vectors stage to compensate the measurement in our CS framework and acquisition stage in CS domain.

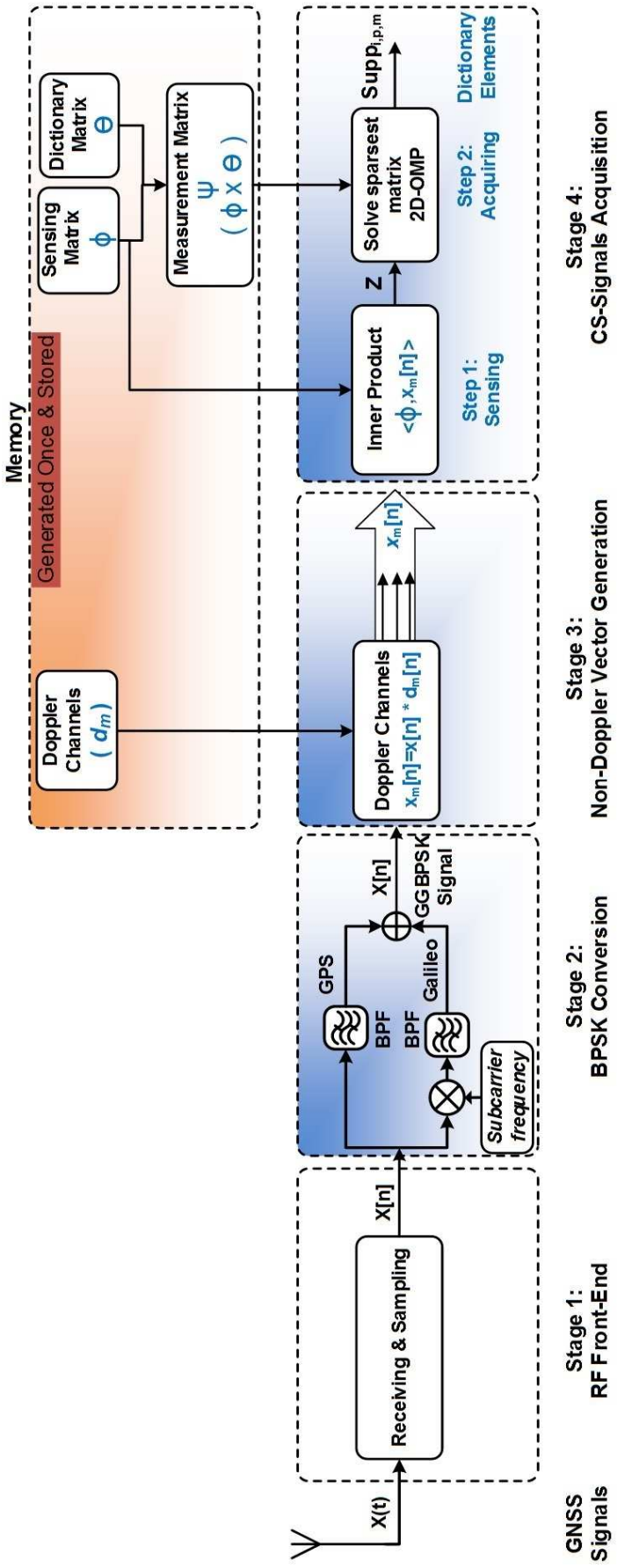


Figure 6-3 CSSR block diagram

6.3.1 CSSR-Receiving, sampling and removing the subcarrier effect

The CSSR can be implemented with any RF front-end. In this implementation we have used BPS receiver, where the received GPS and Galileo signals are sampled at the rate equal to the summation of information bandwidths, which is equal to 6.138MHz and therefore the folded frequency is located at 2.046MHz.

The challenges for achieving this CSSR implementation were; on one hand, we needed to overcome the Galileo signal acquisition ambiguity because the CS dictionary matrix is based on half chip resolution. On the other hand, we needed to figure out how to reduce the complexity of the CS framework, because our chosen Galileo OS and GPS-C/A signals use different modulation techniques (BPSK and CBOC modulations). The solution to both of these challenges is to have both signals as BPSK modulated signals. To achieve this conversion without loss of signal power, the best conversion to realize this without loss of signal power is to use the DSB [21] or BPSK-Like [22] methods. However, implementation of these two methods will result in a complex CS framework. Therefore, to further reduce the computational complexity and adding more freedom to manipulate with the CS frameworks, we have capitalised on our ESCE method (see cp. Section 3.4). This will eliminate the subcarrier frequency effect and overcome the ambiguity.

As illustrated in Figure 6-4, the received GNSS signals go through two channels simultaneously, the first channel filters out the Galileo signal to obtain GPS signal only, while the second channel is responsible for converting the BOC modulation signal to the BPSK like modulation signal. This removing of the subcarrier frequency effect is achieved by multiplying the Galileo signal by the subcarrier channel, which is either the data's subcarrier channel S_b (3.3) or the pilot's subcarrier channels S_c (3.4). The resultant filtered and converted signals are then combined to construct a GGBPSK signal, which is then passed to the next processing stage.

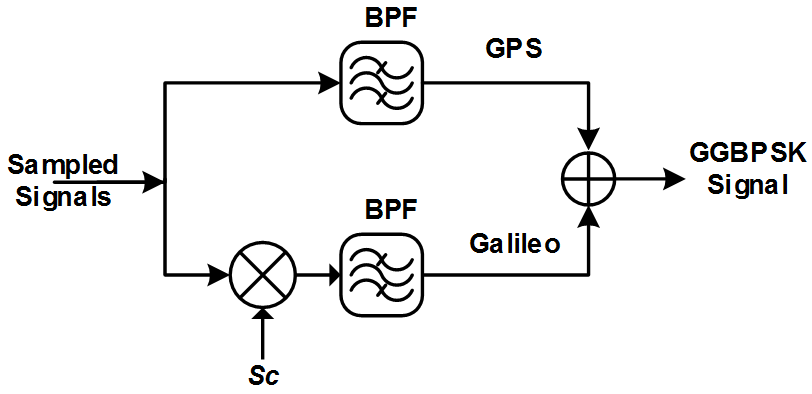


Figure 6-4 GGBPSK signal conversion

Note that, the required Galileo code in the Θ_D matrix is therefore either the primary code of the data channel if S_b is employed to eliminate the subcarrier frequency effect, or the primary code of the pilot channel if S_c is used. Where, using one of these channels (data or pilot) would simplify the construction of the Θ_D. Furthermore, according to the experiments, our recommendation is to use zero phase shift of the generated subcarrier frequency. We found at this phase shift that the elimination gives a better matching performance with the actual code-phase delay in the received Galileo signal than the others shifts.

6.3.2 CSSR-Non-Doppler shift vectors generation

Converting BOC modulation signal to a BPSK modulation signal will significantly reduce the correlator complexity, i.e. from **"code + subcarrier + frequency"** to **"code + frequency"**. Consequently, minimizing these numbers of correlators would accelerate the acquisition process. Thus, the pre-processing, non-Doppler channels, is adopted here to generate non-Doppler shift vectors.

The Θ_D matrix is implemented without any Doppler shift, as will be explained in Section 6.3.4.1. In order to aid our CS framework for finding the right code, it necessitates having a signal without Doppler shift. For that reason, the sampled signal passes through "m" Doppler channels $d_m(t)$ simultaneously that is expressed in equation (6.3). This process will generate non-Doppler shift signals/vectors that will compensate the measurements in the Θ_D matrix. Therefore, only channels that have zero frequency shifts will be selected for our CS framework. Note that, the number

"m" will control the acquisition complexity as it determines the number of rows vectors in the sensing matrix, which will be used in the next stage.

$$d_m(t) = e^{j2\pi m\Delta ft} \qquad \textbf{6.3}$$

As shown in Figure 6-5, these "m" Doppler channels contain a range of all possible Doppler shifts (Δf), where "m" value for a normal signal environment is 321 and can be increased to 401 for high-resolution acquisition in harsh environments. So, when the Doppler shift in the received signal matches the generating frequency of the channel, then the output will contain only the GGBPSK signal without Doppler shift that can be easily acquired in CS process, as illustrated in equation (6.4).

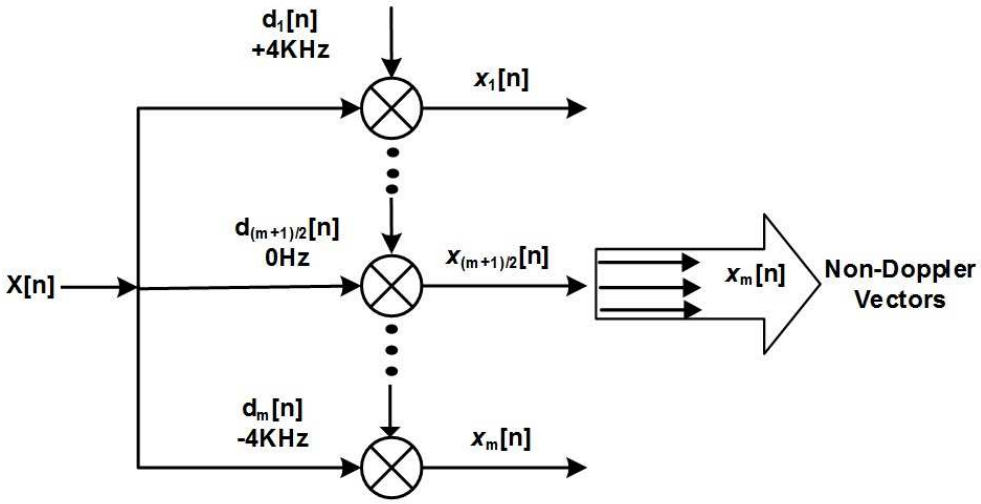


Figure 6-5 Multi-Doppler channels

$$x_m(t) = X(t)d_m(t)$$

$$x_m(t) = \mathcal{S}(t-\tau)e^{j2\pi(f_{L1}-f_d)t}\, e^{j2\pi m\Delta ft}$$

$$x_m(t) = \mathcal{S}(t-\tau)e^{j2\pi f_{L1}t}\; e^{j2\pi(m\Delta f-f_d)t}$$

$$x_m(t) = X_o(t-\tau)\, e^{j2\pi(m\Delta f-f_d)t}$$

$$x_m(t) = X_o(t-\tau), \quad \text{when } m\Delta f = f_d \qquad \textbf{6.4}$$

where $\mathcal{S}$ is the navigation component and X_o represents the received signal without Doppler shift.

6.3.3 CSSR-Signals acquisition procedure

6.3.3.1 *Combined decomposed dictionaries*

In fact, scaling down the number of required correlators and its requirements, i.e. frequency, are highly desired and will directly reduce the searching time and power consumption. As a result, the processing in stages (2&3) relaxes the construction of the Θ_D matrix, whereas both the GPS+Galileo signals are presented now as a BPSK signal and without Doppler shift.

Practically, the design of any GNSS dictionary matrix based on CS technique should include all the GNSS signal code shifts and Doppler frequency shifts; otherwise the signal cannot be acquired. In Section 6.1, we have pointed out that the current dictionary matrix used to acquire GPS signals is considered as a fat dictionary. In order to overcome this issue, we have effectively resolved that by decomposing the dictionary matrix. This decomposition is achieved by generating a bank of codes multiplied by a fixed carrier frequency, i.e. without any Doppler shifts. Thus the size of the dictionary will reduce from (**satellites number x codes shifts x Doppler frequencies shifts**) to (**satellites number x codes shifts**), i.e. huge dimension reduction can be achieved.

To realize this achievement the number of column vectors in our Θ_D matrix is equal to:

$$I \times (P_1+P_2) = 24 \times ((1023+4092) \times 2) \approx 2.5 \times 10^5$$

Where, this size is much less than the previous CS-based GPS dictionary dimension (2×10^6), i.e. it is only 12%, where P_1 represents the code phase of the GPS-C/A-code and P_2 represents the code phase of the Galileo-OS-code as expressed in equation (6.5). Furthermore, our Θ_D matrix does maintain the signal integration between the codes and frequency, irrespective of the signal strength.

$$\Theta_D(t) = \begin{bmatrix} C_{CA_{1,1}} e^{jw_{L1}T_s} \\ C_{CA_{1,2}} e^{jw_{L1}T_s} \\ \vdots \\ C_{CA_{1,2046}} e^{jw_{L1}T_s} \\ \vdots \\ C_{CA_{24,2046}} e^{jw_{L1}T_s} \\ C_{OS_{1,1}} e^{jw_{L1}T_s} \\ \vdots \\ C_{OS_{1,8184}} e^{jw_{L1}T_s} \\ \vdots \\ C_{OS_{24,8184}} e^{jw_{L1}T_s} \end{bmatrix}^T \quad \textbf{6.5}$$

where C_{CA} refers to the GPS C/A code and C_{OS} represent the Galileo OS primary (data or pilot code)

The typical design of the Θ_D matrix makes acquiring both GPS+Galileo signals in a single process possible, rather than one at a time as the work proposed in [69]. It also saves valuable processing time via minimizing the number of required correlators.

Moreover, the generality of the Θ_D matrix enables any GNSS signals or other wireless signals to be combined their dictionary with the GPS dictionary if these signals are folded or down-converted to the same IF frequency. Fortunately, both signals of interest share same frequency band and therefore their dictionaries can be easily combined.

6.3.3.2 *Finding the dictionary elements*

By observing the output from Doppler channels, it can be seen that only a few vectors are useful and have zero Doppler shift. Our idea is to exploit these numbers of channels to determine the amount of Doppler frequency shift. In other words, the channel that generates zero Doppler shifts will be chosen in our CS framework. As shown in Figure 6-3, the CS acquisition is accomplished in two steps:

1. The first step in the acquisition process is transforming the time-domain non-Doppler vectors to the CS-domain. This is accomplished by multiplying (inner product) the non-Doppler vectors outputs with a known orthogonal transform "the sensing matrix ϕ", and the total multiplication will produce a compressed matrix "Z" ensemble with (m x m) dimension, this Z matrix compresses the

necessary information to be simply acquired in CS domain by linear measurements.

$$\begin{bmatrix} z_{1,1} & \cdots & z_{1,m} \\ \vdots & \ddots & \vdots \\ z_{m,1} & \cdots & z_{m,m} \end{bmatrix} \triangleq \begin{bmatrix} \phi_{1,1} & \cdots\cdots & \phi_{1,n} \\ \vdots & \ddots & \vdots \\ \phi_{m,1} & \cdots\cdots & \phi_{m,n} \end{bmatrix} \begin{bmatrix} x_1^1 & \cdots\cdots & x_n^1 \\ & \vdots & \\ x_1^m & \cdots\cdots & x_n^m \end{bmatrix}^T$$

$$Z \triangleq < \phi, x_m > \qquad \textbf{6.6}$$

As shown in equation (6.6) the dimension of ϕ matrix is equal to (m x n) dimension where m is equal to the number of Doppler channels, and n is equal to the number of samples.

2. The next step then is to acquire both signals, where the acquisition is accomplished by solving matrix V in equation (6.7).

$$Z = \psi V \qquad \textbf{6.7}$$

$$\psi = \phi \Theta_D \qquad \textbf{6.8}$$

where $\psi(\omega)$ is the measurement matrix and constructs by multiplying the transform matrix $\phi(\omega)$ with the Θ_D matrix.

To solve sparse matrix V it requires a two-dimensional search algorithm. While, the frequently used CS-based algorithms are designed to search in one dimension such as the Matching Pursuit (MP) algorithm [85], or the OMP algorithm [65]. Practically, the main difference between these algorithms is the MP algorithm that is based on finding the better approximation, which represents the support value, while the OMP algorithm is based on finding the better approximation but it updates this approximation each time or each iteration. The approximation is can be realised by matching the compressed signal (inner product between received signal and sensing matrix) with measurement matrix ψ. Consequently, these algorithms suffer when the problem has a fat dictionary, because it takes a long time to calculate the support value (highest matching).

In this CSSR implementation we have modified the OMP algorithm to search in two dimensions rather than the one dimension. This was managed by adding an extra step to the OMP algorithm. Typically, the one dimension OMP determines the matching from the highest sum. Our additional step is determined the highest peak inside the predetermined highest sum, as illustrated in Algorithm 6-1.

Practically, the 2D-OMP algorithm selects one item at a time; this item is the support value of the Θ_D matrix, which represents the highest inner product between the ψ and the residual, where the initial value of the residual is the Z matrix. So, the right code phase delay $S_{i,c} = \mathrm{supp}(V)$ is determined by the number of columns that represents the highest sum. While, calculating the Doppler frequency shift cannot be obtained from the Θ_D matrix because it does not have any Doppler shifts. Our modification to the OMP algorithm is realised by determining the maximum value inside the highest sum $S_d = \max(S_{i,c})$, which represents the highest inner product value with the zero Doppler shift vector.

Algorithm 6-1 2D-OMP Algorithm

1. Initialise $r^0 = Z, y^0 = 0, \Gamma^0 = \varnothing$
2. for n = 1;n := n+1 till stopping criterion is met
   (a) $\nabla^n = \psi^T r^n$
   (b) $p^n = arg_k \, max|\nabla_p|$
   (c) $q^n = arg_k \, max|\nabla_p^n|$ (update step)
   (d) $\Gamma^n = \Gamma^{n-1} \cup p^n$
   (e) $y^n = \psi_{\Gamma^n}^{\dagger} Z$
   (f) $r^n = Z - \psi y^n$
3. Output: p^n, q^n, r^n & y^n

Here the stopping criterion is the number of iteration (n), which equals to twice of the satellites number, and the dagger † indicates the Moore-Penrose pseudoinverse [74].

For this particular modification we take a simple example to demonstrate how the 2D-OMP determines the highest correlation or matching process. In this example, we picked three matching outputs for each GNSS signal. The setup of this example is as follows:

1. The tested signals are the GPS-SV2 signal and the Galileo-SV2.
2. The maximum channel delay distribution of the C/A and the OS codes are set to 10Tc, i.e. each satellite has 20 code shifts because it is designed with half chip resolution. Therefore the total number of column vectors in the Θ_D matrix is equal to (20 x (24+24) = 980), as illustrated in Figure 6-6.
3. The code phase delay for the GPS signal is set 1½ Chip and 1 Chip for the Galileo signal.
4. The Doppler frequency shift of GPS signal is equal to 500Hz and of the Galileo signal is set to -500Hz. Where the row number 1 means +4KHz Doppler frequency shift and row number 401 refers to -4KHz Doppler frequency shift, as shown in Figure 6-7.

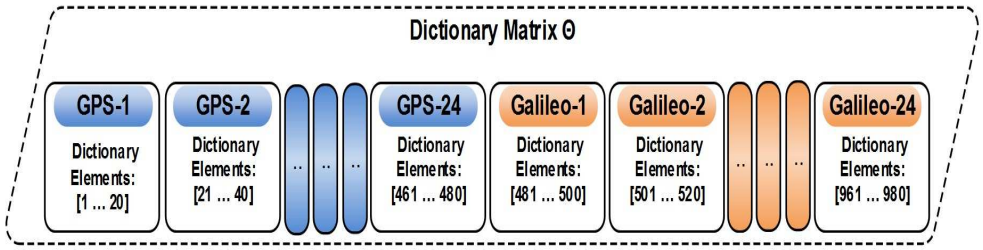


Figure 6-6 Dictionary matrix with maximum phase delay 10Tc

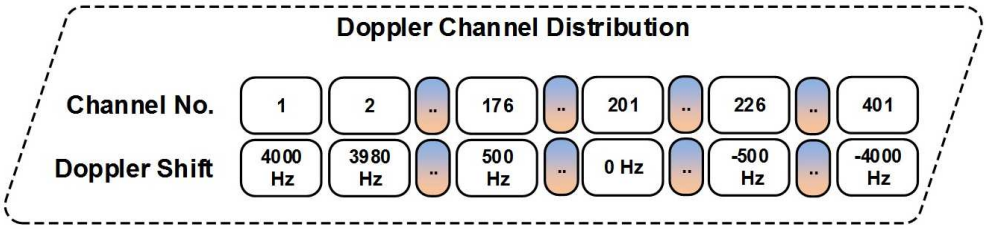
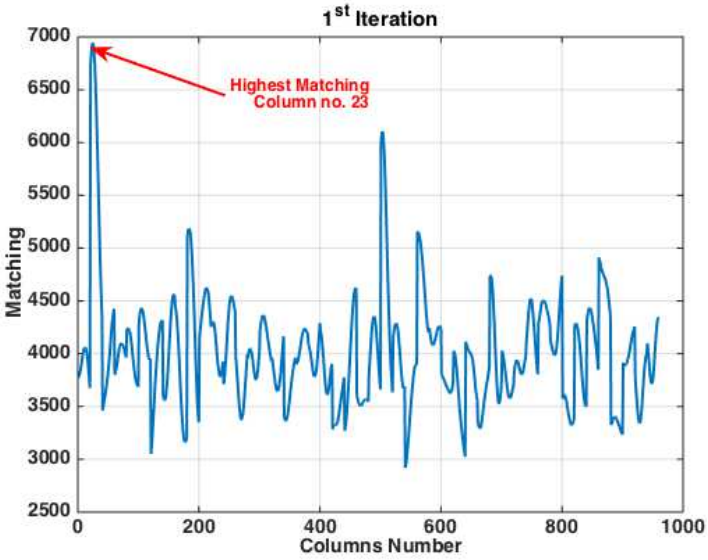


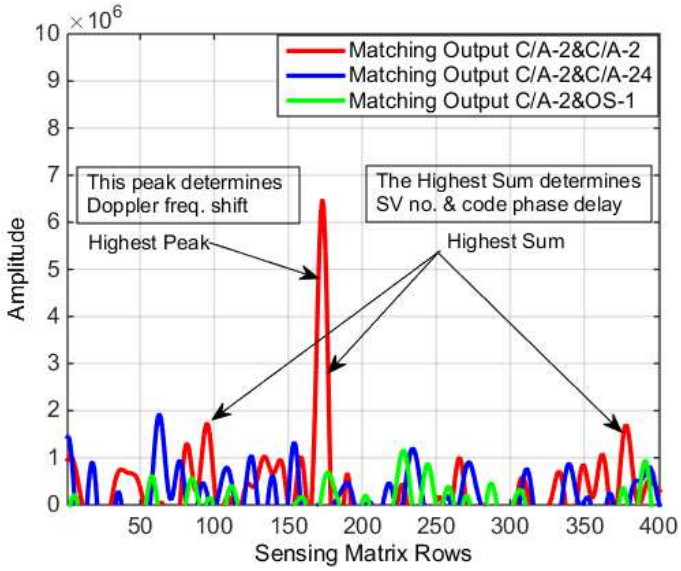
Figure 6-7 Doppler channel distribution

In the first iteration when we matched the output from sensing-step with the measurement matrix, the highest sum is obtained at column number 23, as shown in Figure 6-8-a. This column number belongs to the GPS-SV2 and code phase delay is equal to 1½ Chip. The next step in this algorithm is to find the highest peak inside the

highest column sum. As shown in Figure 6-8-b, the highest peak is located at row number 176, the red line and this point out that the Doppler frequency shift is 500Hz. The matching amplitude of the GPS-SV2 is three times the matching amplitude of the GPS-SV24 and of the Galileo-SV1 (blue and green lines respectively). Then the algorithm replaces the obtained highest peak with zeros to perform the second iteration.



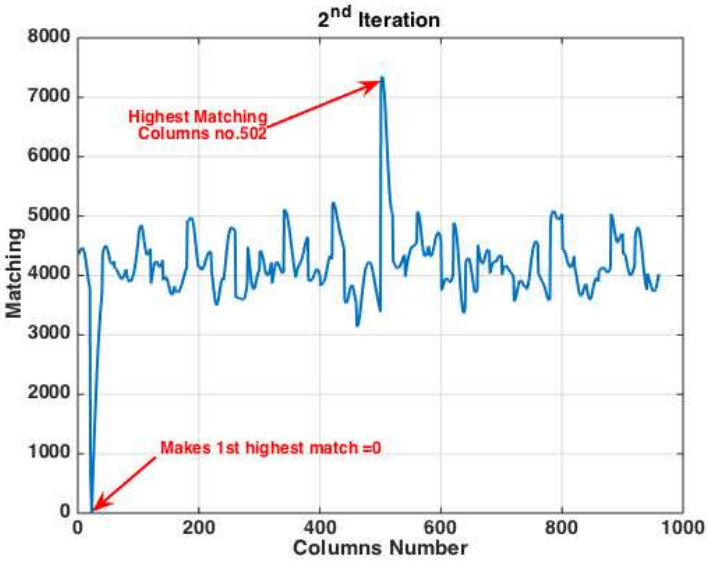
(a)



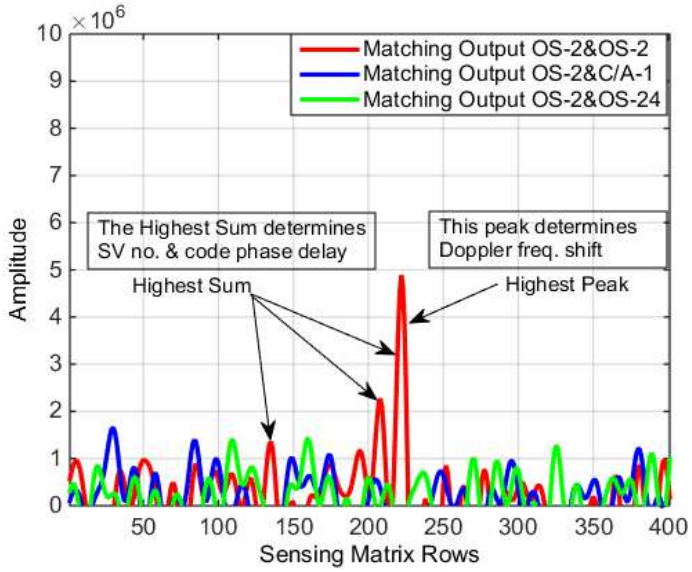
(b)

Figure 6-8 GPS matching results using 2D-OMP a) determining code phase delay b) determining Doppler frequency shift

In the second iteration, the highest sum is obtained at column number 502, as shown in Figure 6-9-a. This column number refers to the Galileo-SV2 with a code phase delay equal to 1 Chip. Then finding the highest peak inside the highest sum is performed and as depicted in Figure 6-9-b the highest peak is located at row number 226, the red line, and that denotes the Doppler frequency shift is equal to -500Hz. In addition, the result shows that the matching amplitude of the Galileo-SV2 is three times the matching amplitude of the GPS-SV1 and of the Galileo-SV24 (blue and green lines respectively).



(a)



(b)

Figure 6-9 Galileo matching results using 2D-OMP determining code phase delay b) determining Doppler frequency shift

6.4 CSSR experiments setup

Several experiments were carried out to evaluate the performance of our CSSR with various signal conditions. We have captured signals from actual wireless communication channel using Signalion HaLo-430 platform. Also we have implemented both GPS and Galileo signals in simulation environment using MATLAB-Simulink platform. The setups of both environments are:

1. The use of the realistic HaLo-430 platform testbed enables us to assess the performance of our CSSR versus other traditional implementations. The setups of these scenarios are same as the ESCE experiments setup (see cp. Section 3.5) except the following parameters:
    a) The sampling frequency is equal to 6.25MHz.
    b) The number of GNSS signals is four (two GPS signals and two Galileo signals).
    c) The length of tested signals is 20ms.
    d) The frame length of each baseband signal is 125,000 samples and the number of pause samples after transmitting a frame is 12800 samples.
    e) The frame length of the received signal is 300,000 samples.

    Also, in each scenario we have used all the 4 channels of the HaLo-430 platform in the transmitter and receiver sides.
2. The simulation environments are used to highlight the high frequency resolution obtained based on CSSR implementation under control scenarios. In this environment two GPS signals and two Galileo signals are simulated in each scenario. The same simulation conditions that were used to simulate Galileo signals in our OGSR experimental setup (see cp. Section 4.3) have been applied in this Simulink environment, such as (T_x→Channel→R_x) block, Rayleigh Fading block and AWGN block. The setting of the experiments is as follows:
    a) The sampling frequency is equal to 6.138MHz.
    b) Two GPS signals, as shown in Figure 6-10, and two Galileo signals, as depicted in Figure 4-8, are simulated in each scenario.

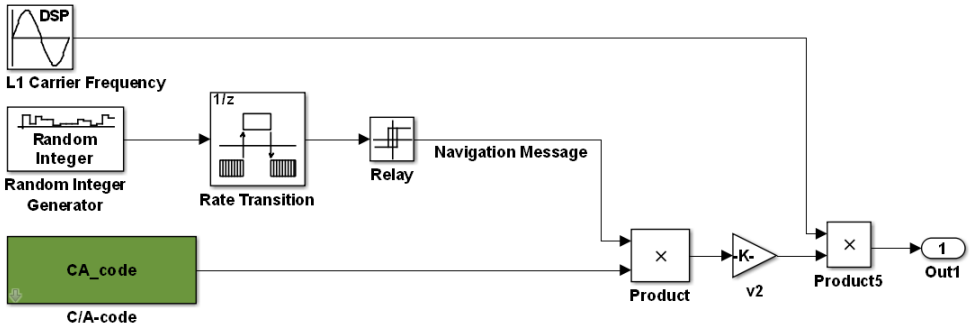


Figure 6-10 GPS C/A signal generation

6.5 CSSR results and performance

We worked on the GPS alone based on CS and we had some impressive results, this work, when including a Galileo signal, we are still in the borderline in that other solution based CS.

For the GPS signal acquisition, we have found that the suitable range of Doppler channels is 81 channels as will be illustrated in Section 6.4, So, when combining two GNSS signals these numbers of channels certainly must be increased. The Galileo OS code length is 4 times longer than the GPS C/A code; therefore the minimum number of Doppler channels is then 4 times the number used for only GPS signal, i.e. 321 channels. Note that we cannot use 324 due to a range of frequency bin step. So, in order to improve and ensure correctness of our CSSR, two ranges of the Doppler channels are used to evaluate the CSSR performance, which are 321 and 401 channels. These Doppler channels produce very high frequency resolution and equal to 25Hz when using 321 channels and 20Hz if 401 channels are deployed.

Our assessment is divided into two parts; the first part compares CSSR with traditional implementation in terms of detection performance, the acquisition time, the acquisition frequency resolution and computational complexity. The second comparison compares CSSR (GPS+Galileo) with the CMS solution [57] in terms of the computational complexity and the memory requirements, where this solution is designed for acquiring only GPS signal.

I. CSSR versus MF implementations

Apart from the CSSR implementation, we have implemented three other combined implementations. These methods are the MF-D (GPS+Galileo dual-sideband), the MF-S (GPS+Galileo single-sideband) and the MF-BS (GPS+Galileo single-BPSK-Like), as illustrated in Figure 6-11. This enabled us to compare CSSR to the others under similar conditions.

Figure 6-12, shows the acquisition rate of our CSSR using various numbers of Doppler channels, actually CSSR-H refers to using 401 channels and CSSR-L refers to using 321 channels, compared with MF implementations. In this particular test, the subcarrier-data channel is used to remove the subcarrier frequency effect and then the primary data code is used in the Θ_D matrix.

In this comparison, it is necessary to point out that the CSSR signals acquisition performance of the 401 channels is better than the performance using 321 channels by 2dB. This increasing of the channel number improves the measurements in the CS framework. For the performance of MF implementations, it is clearly shown that the MF-D is better than the MF-S by 2dB and 4dB outperforms the MF-BS, due to combined powers of the dual sidebands when acquiring Galileo signal. The result also shows that the performance of CSSR-H is as good as MF-D in high and low C/N.

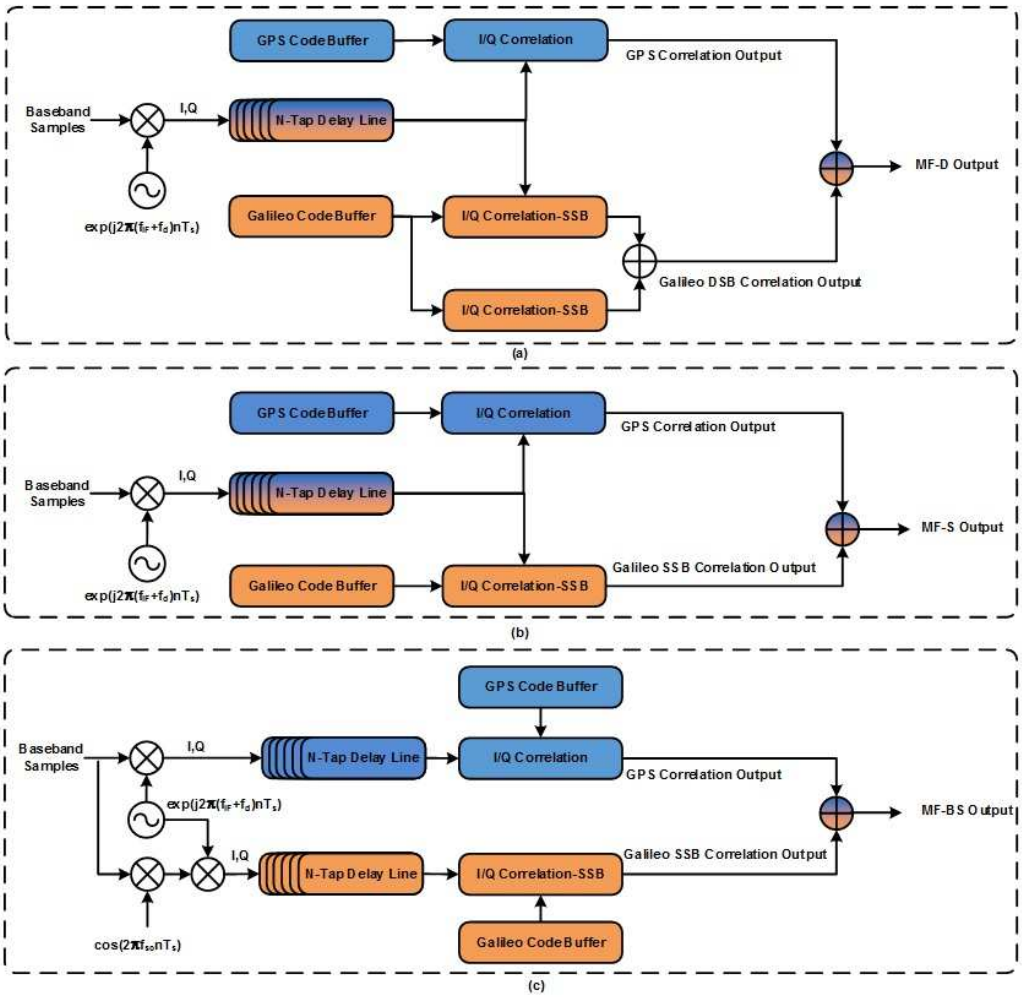


Figure 6-11 Matched filter implementations a) MF-D, b) MF-S and c) MF-BS

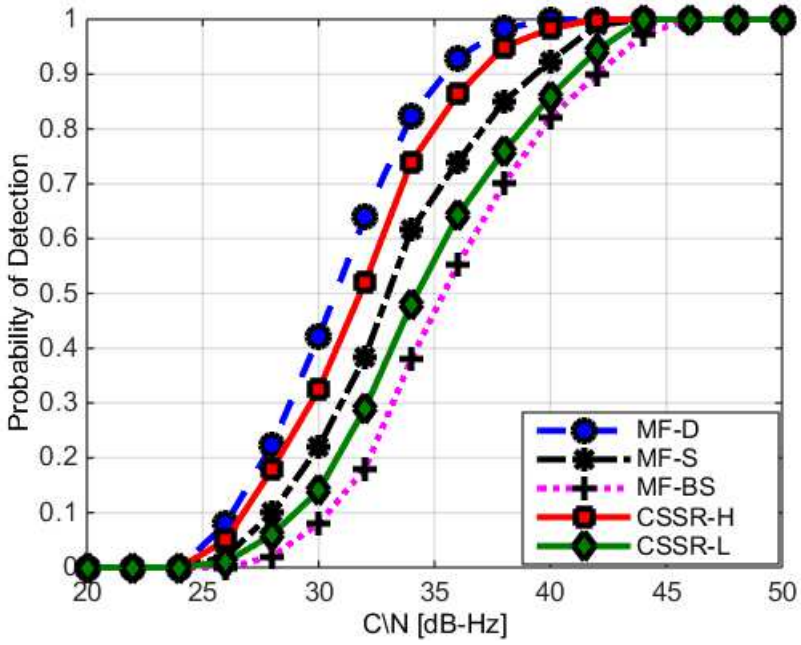


Figure 6-12 CSSR probability of detection vs. C\N

For the acquisition time comparison, it is necessary to point that in traditional (time-domain or frequency-domain) implementation, the acquisition time depends on two factors, which are the signal length (dwell time) and the sampling frequency. So, long dwell time and high sampling rate leads to long processing/searching time and vice versa. While, the acquisition/searching time in our CSSR implementation depends on the number of row vectors in the sensing matrix and the number of column vectors in the Θ_D matrix. Therefore, increasing the dwell time, for example from 4ms to 8ms or 20ms to acquire low sensitivity signals will result in the same cost of the processing 4ms. The only overhead takes place in the second stage, i.e. when generating the non-Doppler shift vectors, while in the rest CS-process the time is almost constant. To count that:

1. The first stage complexity is the inner product between the output from the Doppler channels X_m and the sensing matrix ϕ and the computational is highly dependent on the number of Doppler channels and equal to $\boldsymbol{O}(N_d M^2)$.

   where Nd represents the number of samples and M is the Doppler channels.

2. The complexity of the second stage, i.e. finding the support elements is same for all signal length because the inputs to the 2D-OMP algorithm rely on the block Z (m x m) and the measurement matrix ψ (m x (IP_d)).

   where "m" is the number of the Doppler channels, "I" represents 24 satellites and "P_d" is the code phases for GPS+Galileo signals with half chip resolution (10230).

As shown in Figure 6-13, increasing the dwell time makes the acquisition time increase linearly in the MF implementation. While in our CSSR the acquisition time is constant and when counting the non-Doppler vector generation the processing time is still much less than the time required in the MF implementation. Note that, in this comparison the sampling frequency is equal to 6.138MHz.

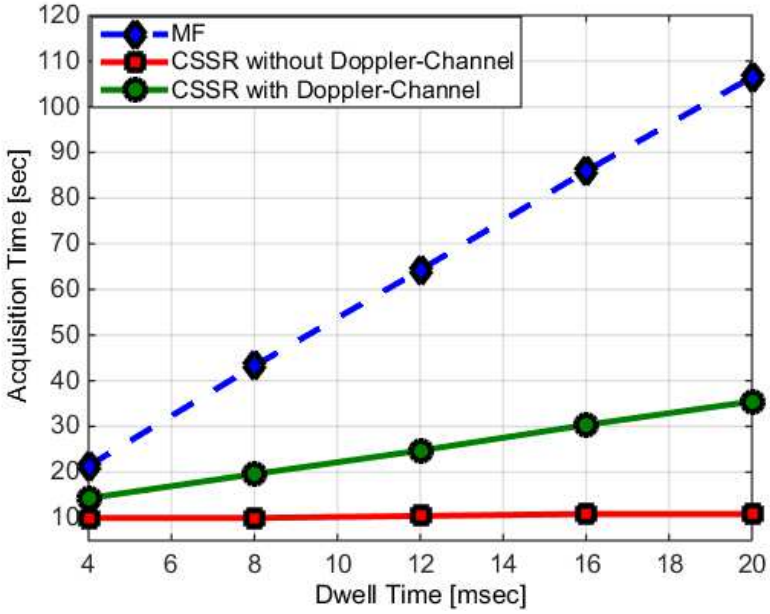


Figure 6-13 Processing time and first stage complexity vs. signal length

The third comparison focused on the frequency resolution obtained by our CSSR, which is equivalent to the fine frequency. In typical GNSS receiver, the fine frequency process is used to increase the frequency resolution of the acquired signal, for example from 1KHz or 500Hz to 10s of Hz. This process would accelerate the lock of the local carrier frequency with the frequency of the received signal in the tracking stage. This process is located after the acquisition process, and that leads to make the transition time from the acquisition stage to tracking stage longer. Furthermore, in time-domain or frequency-domain signal acquisition, the frequency resolution (frequency search step) depends on the signal length. This is because if the local carrier is off by one cycle it means there is no correlation and when it is less than one cycle it leads to partial correlation. Thus, in 1ms signal length then 1KHz will change one cycle because the frequency bin step represents the ratio between the sampling frequency to the number of samples. For example, the sampling frequency in our simulation setup is 6.138MHz and the number of samples of 1ms is 6138 samples, then the ratio is equal to 1KHz. Consequently, when the tested signal length is 4ms then the frequency bin step is equal to 250Hz and so on, more details in [33]. While, in CS-domain the acquisition resolution depends on the compression factor, i.e. the numbers of rows in the sensing matrix. In our CSSR implementation the number of Doppler channels controls these row numbers. In fact, the frequency

resolution that is deployed in our CSSR produces high frequency resolution and is equal to 20Hz. This resolution enables tracking signal without any need for the fine-frequency process and therefore our CSSR would reduce the transition time.

Figure 6-14, shows the RMSE frequency resolution of our CSSR versus MF-S implementation. Whereas, the typical use of the 401 Doppler channels (20Hz frequency resolution) makes the estimation of the Doppler frequency shifts close to the actual frequency of the received signals, and the accuracy are around 10Hz and 40Hz in high C\N and low C\N respectively. While reducing these channels to 321 (25Hz frequency resolution) will reduce the accuracy of calculating the Doppler by 10Hz in the high and low C\N. However, the Doppler frequency shift calculated by our CSSR is much better than the MF implementation that based on 250Hz frequency resolution. Where, the RMSE Doppler frequency shifts of the MF-S implementation vary between 60Hz-150Hz in high C\N and low C\N respectively. Note that increasing the frequency bin step in MF will increase the acquisition time.

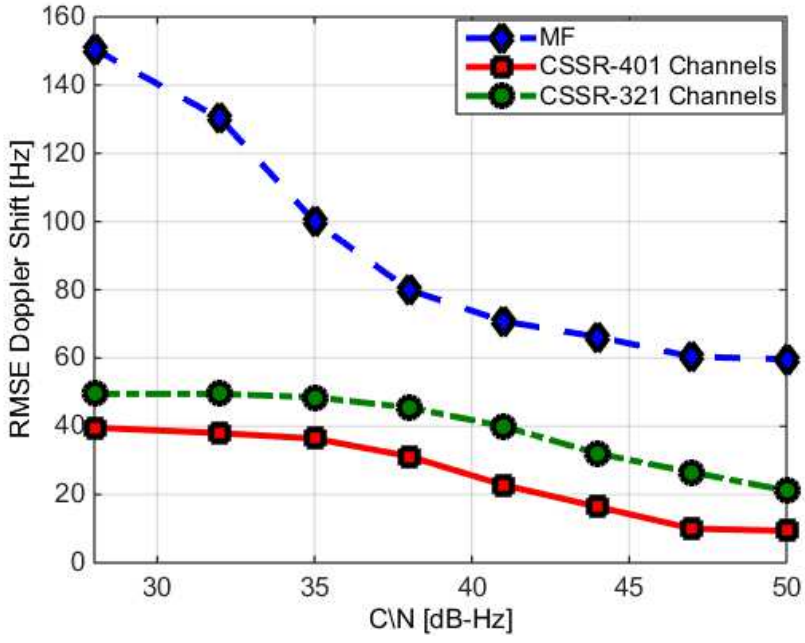


Figure 6-14 RMSE Doppler frequency shifts

The last comparison with the MF implementation is the computational complexity. In this comparison we have compared CSSR using 401-channels "CSSR-H" and the MF-S implementations, and as illustrated in Table 6-1.

Table 6-1 CSSR vs. MF Computational Complexity

CSSR-H	MF-S
Generate non-Doppler Vectors: ***O***(N M)	Removing carrier frequency ***O***(N Q)
Inner Product: ***O***(N M^2)	Correlating GPS codes ***O***(I C P_s N)
Inner Projection ***O***(M I P_d S)	Correlating Galileo codes ***O***(I C P_s N)
Find Code Phase Delay ***O***(S log(I P_d))	GPS accumulation ***O***(I P_s N)
Find Doppler Shift ***O***(S log(M))	Galileo accumulation ***O***(I P_s N)
Stopping Criterion ***O***(M S)	Threshold comparison ***O***(I log(P_s))

where "Q" is 33 frequency bin step, "C" is 2 correlation channels (the in-phase and quadrature-phase). "P_s" is 8184 code phases for GPS or Galileo signals with half chip resolution based on 4ms signal length and "S" is 50 the numbers of iterations setting.

Figure 6-15 depicts the total computational complexity versus increasing the sampling rate, from 2MHz to 16MHz. The comparison showed that more than 50% reduction has been achieved in acquiring both GPS and Galileo signals using our CS framework.

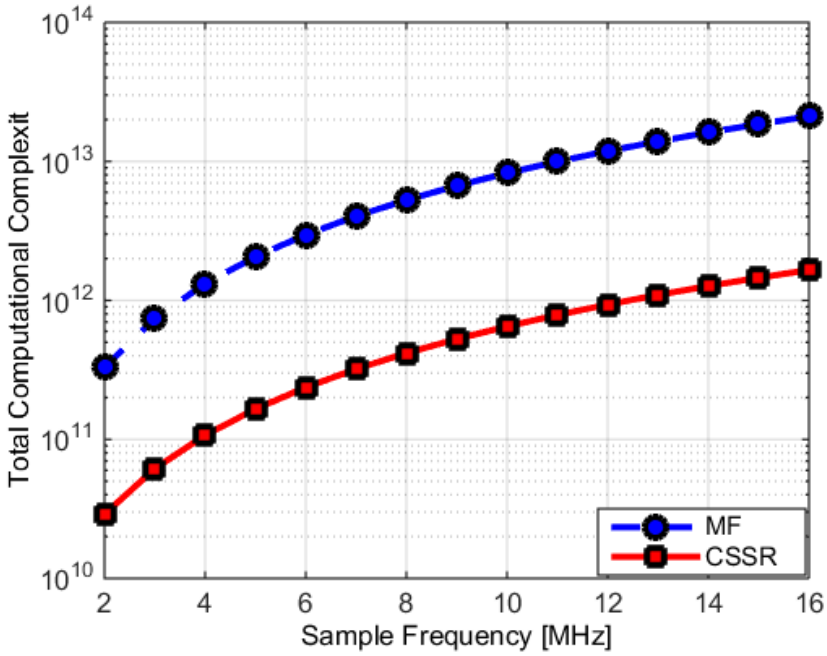


Figure 6-15 Computational complexity vs. sampling frequency

II. CSSR (GPS+Galileo) implementation versus CMS (GPS) solution

The achievements with regards to memory requirements are illustrated in Table 6-2. The CSSR implementation again proves the saving of the required memory storage, where our Θ_D matrix is 73% of the CMS's dictionary matrix that is based on GPS only.

Table 6-2 CSSR-Memory Requirements

Matrix	CSSR	CMS
Sensing Matrix	$M\ N_d$	$C\ I\ P_s\ Q$
Dictionary Matrix	$I\ P_d\ N_d$	$I\ N_s\ P_s\ Q$
Measurement Matrix	$M\ I\ P_d$	$N_s\ C$

*where "N_d" represents the 24552 samples based on 4ms signal length at 6.138MHz sampling frequency, "*Ns*" represents the 8,184 samples for the tested 4ms length signal at 2.046MHz sampling frequency, "I" is the 24 SV satellites, "C" represents the chosen 600 number of channels in CMS, "Q" is the 41 frequency search steps and "Ps" is the 2046 code phase resolution for GPS signals.*

Moreover, to discuss the dwell time effect on the CS-based acquisition solution, we also compare CSSR with CMS and GCS-1 solutions because they have different dictionary matrix implementation. We found that these CS-based solutions were also directly proportional to the dwell time, as detailed below:

1. In CMS solution the sparse vector generation is directly proportional to the dwell time because the row length depends on the signal length. Therefore increasing the dwell time will increase the time to generate sparse vector [57].
2. In GCS-1 solution, the acquisition time is also directly proportional to the dwell time due to the bank of correlators that multiplied by the received signal must have the same length. So, increasing or decreasing the dwell time will directly effect on the matching time that located at the beginning of the CS acquisition [58].

6.6 Conclusion on the CSSR

We called a 2for1 receiver because it acquires both GPS and Galileo signals at less than the complexity and processing time required by an MF acquisition process.

The CSSR implementation combines the acquisition of the GPS+Galileo signals, for the first time, in single search process based CS technique. Acquiring GPS+Galileo signals is accomplished with fewer correlators/measurements as the CS process transfers the correlation/matching from the whole length of the signal to the number of rows/channels in the sensing matrix. CSSR combines the dictionaries of these signals in a single combined dictionary, by capitalising on our previous ESCE method that eliminates the subcarrier frequency effect that converts the BOC signal to BPSK signal. This eliminates the repetition to find the supports values.

The implementation requirements and detection performance of our CSSR are analysed and compared with other MF implementations that are based on ambiguous and unambiguous Galileo signal acquisition. Also our CSSR is compared with the CS-based solution that designed to acquire GPS signal only. The results based on simulation and realistic environments of our CSSR implementation indicate that, in one hand, the acquisition time and complexity are less by 50% than the conventional

MF implementations, as well as CSSR achieves high frequency resolution. On the other hand, CSSR reduces the memory storage requirements and computational complexity 73% and 21% respectively in comparison to the CS-based solutions like CMS solution. In addition, to help acquiring low sensitivity signals, the process of increasing the dwell time from 4ms to 8ms or 20ms will cost the same as processing 4ms dwell in CSSR implementation.

The analysis of the CSSR implementation revealed that the control parameter is the range of the Doppler channels, where increasing or decreasing channels will directly effect on the CS measurements, the acquisition rate and the resolution of estimating both of the Doppler frequency shift and the code phase delay. Thereby, in order to obtain desired acquisition rate, frequency resolution and accuracy, increasing the number of Doppler channels/sensing matrix's rows is required.

The other contribution in this implementation is the 2D-OMP algorithm that can be used to solve CS problem in one or two dimensions. The computational cost of this modification is much less than the cost when we solve CS problem that has a fat dictionary.

In conclusion, unlike other GNSS receivers, our CSSR implementation achieves significant saving, in terms of reducing the complexity and accelerating the acquisition process, as well as achieving higher frequency resolution acquisition that is equivalent to the fine frequency.

I gave an attractive presentation on the CSSR implementation in the Institute of Navigation (ION GNSS+ 2015) on Friday, September 18, 2015 at Tampa, Florida in the USA.

6.7 The SCSSR implementation

As illustrated in Section 6.3.3 the Θ_D matrix can be used to acquire either a single signal or combined signals, like combine GPS with other GNSS or any wireless signals if their frequencies are folded or down-converted to the same IF frequency. In this section we shall demonstrate the performance of acquiring GPS signal only using Θ_D matrix.

Figure 6-16, shows the block diagram of our SCSSR implementation. The SCSSR process is less than the CSSR process by one stage (the conversion stage) and this makes acquiring GPS signal being accomplished in only 3 stages:

1. The GPS signal is sampled at information rate (chipping rate = 2.046MHz) to have half chip resolution.
2. Then the samples pass through “m” Doppler channels simultaneously to generate non-Doppler shift vectors.
3. Acquiring GPS signal based on our CS framework also consists of two steps:
    A. Sensing step
    B. Acquisition step

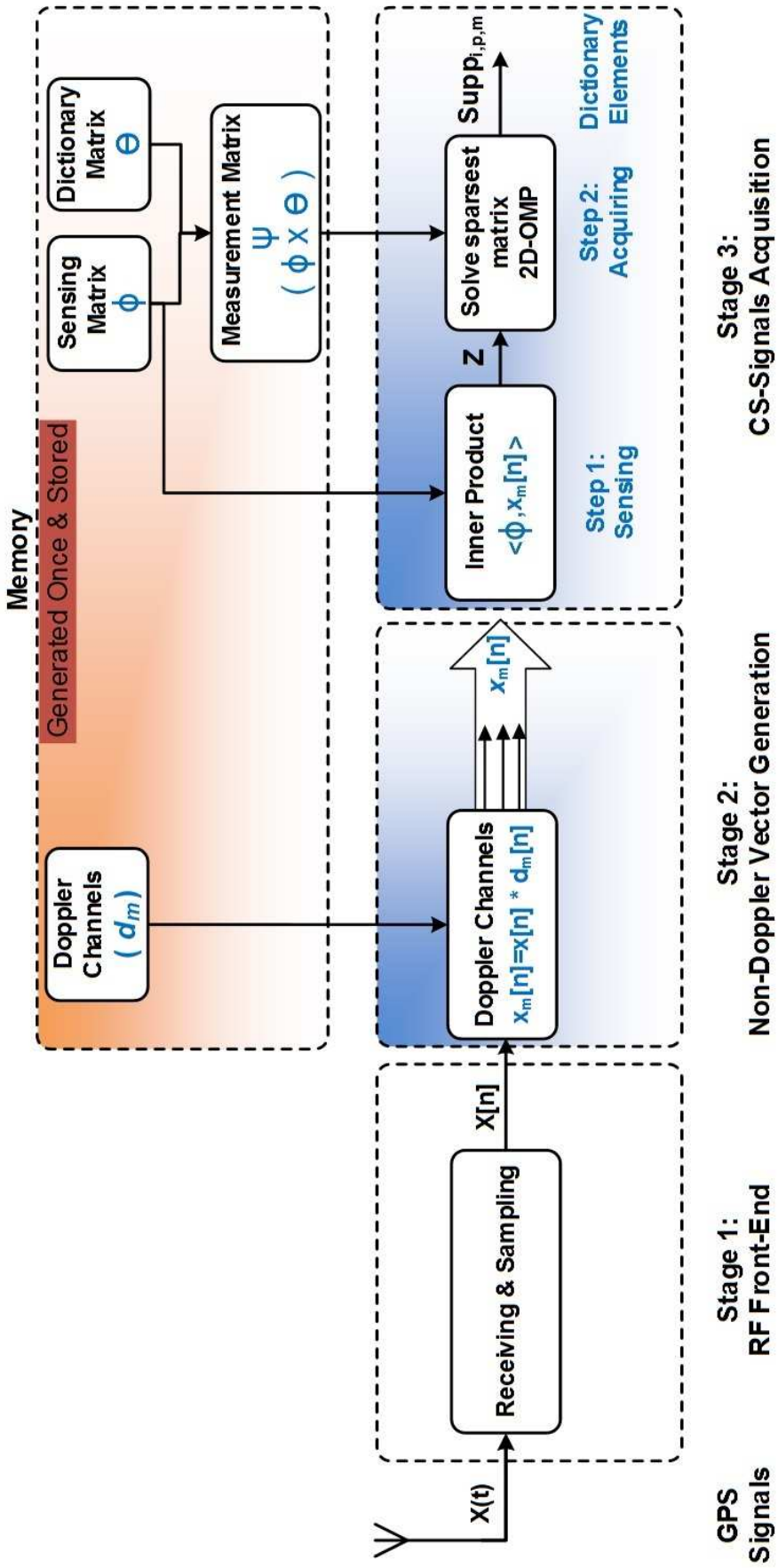


Figure 6-16 SCSSR block diagram

6.7.1 SCCSR-GPS dictionary matrix structure

As detailed in Section 6.3.3.1, the Θ_D matrix is represented as a bank of codes rather than a bank of correlators. Consequently, the Θ_{DS} matrix of the GPS signal is equal to (I.P) multiplied by the same carrier frequency, where (P) is the "search step of code phase delay". As a result, our Θ_{DS} matrix implementation achieves a massive dimension reduction in comparison to the previous implementations and as explained in Section 6.1, where the number of columns vectors in the previous CS-based dictionaries is:

$IPQ = 24 \times (1023x2) \times 41 \approx 2x10^6$

While in our SCSSR now equal to:

$IP = 24 \times (1023x2) \approx 4.9x10^4$

$$\Theta_{DS}(t) = \begin{bmatrix} C_{1,1}\, e^{jw_{L1}T_s} \\ C_{1,2}\, e^{jw_{L1}T_s} \\ \vdots \\ C_{1,2046}\, e^{jw_{L1}T_s} \\ \vdots \\ C_{24,2046}\, e^{jw_{L1}T_s} \end{bmatrix}^T \qquad \textbf{6.9}$$

It is worthwhile to mention that our Θ_{DS} matrix is same as a Toeplitz matrix since all descend diagonally from left to right be constant. Therefore, the solution will be easier to find the right code. In other words, if we assume that the dimension of the dictionary matrix is (c x c) then the computational complexity of the solution will be reduced from $O(c^2)$ to $O(2c-1)$.

6.7.2 SCSSR-Non-Doppler shift vectors generation

The process of generating non-Doppler vectors is the same process that is used for our CSSR implementation. As explained in Section 6.3.2 the length of the C/A code is a quarter of the OS and the required number of Doppler channel then will be 81 channels. So, as expressed in equation (6.4) there are few vectors, which do not have Doppler frequency that will be selected in our CS framework. Furthermore, the frequency resolution is now equal to 100Hz and it is still less than the frequency resolution that is used in the traditional implementations.

6.7.3 SCSSR-Signal acquisition process

The same process that was deployed in our CSSR implementation is used here. The main differences in the size of matrices between SCSSR and CSSR are:

1. The number of rows vectors “m” in the sensing matrix is less than that used in the CSSR.

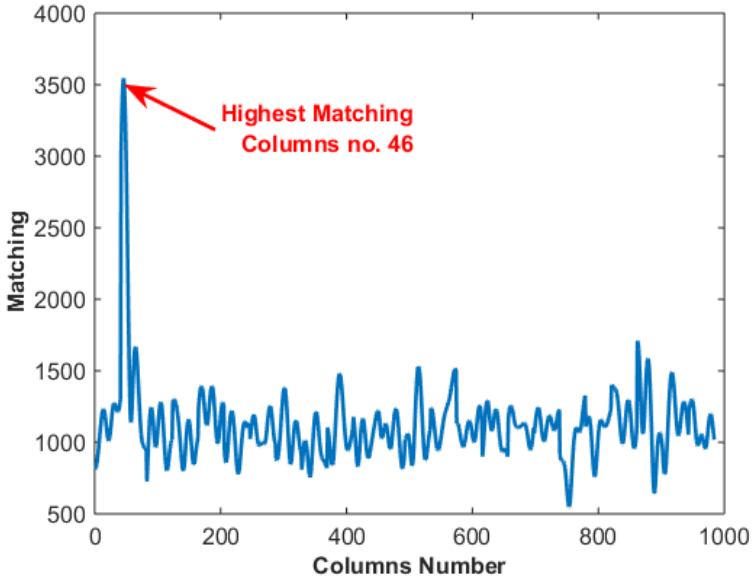
2. The number of column vectors of the measurement matrix ψ is one-fifth than the measurement matrix in the CSSR.

The process is accomplished by taking the inner product between the non-Doppler vectors output with the sensing matrix ϕ, to construct (m x m) Z block (see equation 6.6). Then we used the 2D-OMP to solve matrix V in equation (6.7) and to determine the dictionary elements.

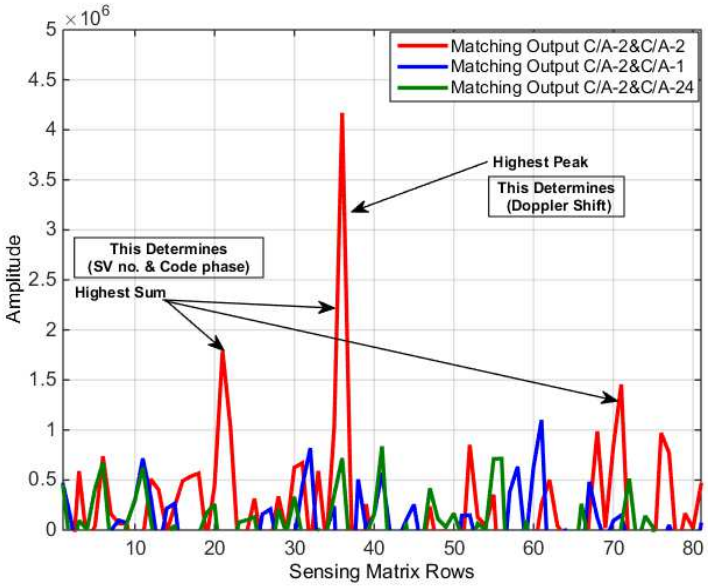
In our SCSSR implementation we also demonstrate how the 2D-OMP matching process. As shown in Figure 6-17, three matching outputs are selected to describe the performance of our 2D-OMP algorithm and in this example:

1. The simulated signal is a GPS-SV2 signal.
2. The maximum channel delay distribution of the C/A code is set to $20T_c$, i.e. each satellite has 40 code shifts, and therefore the total number of column vectors in the measurement matrix is (40 x 24 = 960).
3. The code phase delay for the GPS signal is set 3 Chip.
4. The Doppler frequency shift of a simulated signal is equal to 500Hz, note that row number 1 means +4KHz Doppler frequency shift and row number 81 refers to -4KHz Doppler frequency shift.

As shown in Figure 6-17-a the highest sum is achieved at column number 46 and this number belongs to the GPS-SV2 with code phase delay equal to 3 Chip, because each SV is represented by 40 code shifts. Then, as depicted in Figure 6-17-b, the highest peak inside the highest sum is located at row number 36 (the red line) and that means the Doppler frequency shift is 500Hz. While the matching amplitude between the generated GPS-SV-1 and generated GPS-SV24 (blue and green lines respectively) with the simulated SV-2 is a quarter of the right match.



(a)



(b)

Figure 6-17 GPS-only Matching results using 2D-OMP a) determining code phase delay b) determining Doppler frequency shift

6.8 SCSSR experimental results and performance

In order to highlight the reduction obtained for both computational complexity and memory requirements, our SCSSR was compared with the MCS [57] and our GCSR [86]. Table 6-3 illustrates the breakdown of the computational complexity. While, Table 6-4 shows the memory storage requirements that are needed for the three

matrices. In these tables, "N_s" represents the 8184 samples for the tested 4ms length signal at 2.046MHz sampling frequency, "I" is the 24 GPS satellites, "M" is the 81 Doppler channels used in our SCSSR, "C" represents the chosen 480 number of channels, "Q" is the 17 frequency search steps as used in CMS and GCSR methods, "S" is the 24 number of iterations setting, and "P" is the 2046 code phase resolution.

This comparison shows that our computational complexity and memory requirements are less by 80% than CMS and GCSR solutions. Also, our SCSSR satisfies acquiring signals at higher frequency resolution up to 100 Hz, which is almost equivalent to the "fine frequency" stage of acquisition (reduced search space integration deployed after acquiring the signal) in the traditional receivers.

Table 6-3 SCSSR Breakdown Computational Complexity

Steps	Our SCSSR	CMS & GCSR
	Generate Vectors ➔ ***O***(N_sM)	Digital Compression ➔ ***O***(N_sC)
	Inner Product ➔ ***O***(M^2)	CTF Block ➔ ***O***(C^2)
Residual Update	***O***(S^2)	
Inner Projection	***O***(MIPS)	***O***(CIPQS)
	Find Code Phase Delay ➔ ***O***(S log(IP))	Find Dictionary Element ➔ ***O***(S log(IPQ))
	Find Doppler Shift ➔ ***O***(S log(M))	
Stopping Criterion	***O***(MS)	***O***(CS)
Note that shaded cells represent the lowest computational		

Table 6-4 SCSSR Memory Requirements

Matrix type	Our SCSSR	CMS	GCSR
Sensing Matrix	M N_s	C IPQ	C N_s
Dictionary Matrix	N_s IP	N_s IPQ	N_s IPQ
Measurement Matrix	M IP	C N_s	C IPQ

To evaluate the performance of our SCSSR, four scenarios of various GPS signal conditions, as shown in Table 6-5, are simulated using MATLAB. Firstly, the simulated signals are fed to AWGN and the received power is set to -125dBm. Then it is sampled at a rate equal to the information bandwidth, which is 2.046MHz and the total cascade noise figure is 3dB.

To assess the performance of our SCSSR, we have performed several experiments that use a different number of Doppler channels to illustrate the effect of increasing these channels. Experimentally, we have found that the minimum number of channels that can be used to acquire GPS signal is 33 channels. This number is controlled by the frequency bin step, where the 33 frequency bin steps refers to 500Hz frequency resolution with ±4KHz Doppler frequency range. Hence, increasing this number will enhance the acquisition rate and increase the frequency resolution. Therefore, two numbers of Doppler channels have been chosen (33 & 81) for these simulations. Also, to overcome the dwell time ambiguity we have used (1ms and 4ms), which are equivalent to 2046 and 8184 samples respectively.

Table 6-5 GPS Signals Scenarios

C\N dB-Hz	**LOS & Multipath signals Scenarios**
	Number of received signals
50-46	5 LOS signals
45-41	5 LOS signals and one multipath for each signal
40-36	5 LOS signals and two multipath for each signal
35-30	5 LOS signals and three multipath for each signal

As shown in Figure 6-18, SCSSR achieves low acquisition rate if the number of channels (33-channels) is used and at lowest dwell time 1ms. Increasing the time of tested signal to 4ms improves the acquisition rate by 10%. While, when the number of Doppler channels is risen from 33 to 81 channels, the probability of acquiring the 1ms signal will increase to 40%. And when increasing the signal length to 4ms, the fixed rate is more than 20% overall better performance.

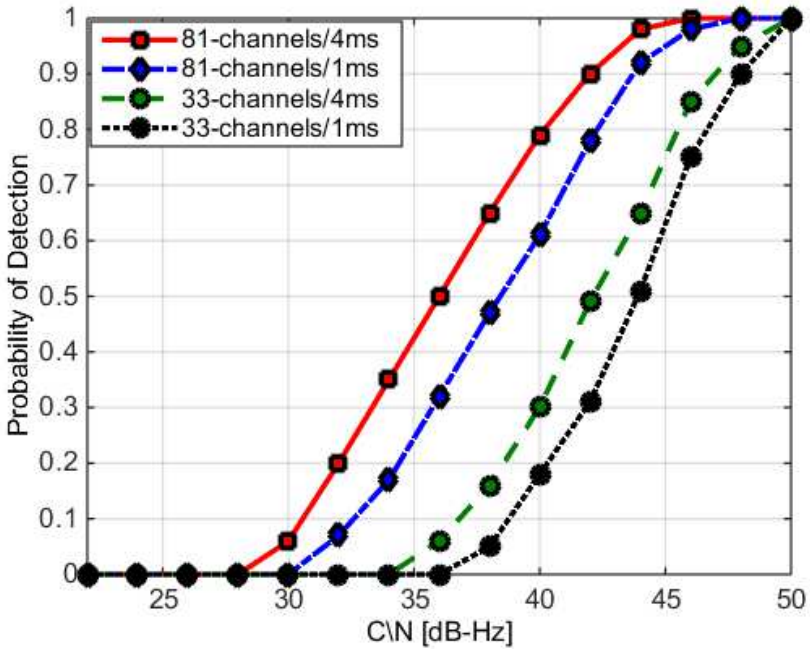


Figure 6-18 SCSSR probability of detection vs. C\N

Additionally, to emphasise the high frequency resolution achievement, Figure 6-19 shows the performance of using two types of Doppler channels, which are 33 and 81. These Doppler channels will control the size or the number of the rows in the sensing matrix. i.e. the high number means high sensing and of course more accurate correlation or matching. The results show that the RMSE is less than 100Hz when 81 Doppler channels are deployed, and can be assumed to be equivalent to the fine frequency. While the use of 33 Doppler channels the RMSE increases to 150Hz, which still is highly accurate than time and frequency implementations that are based on 500Hz frequency resolution.

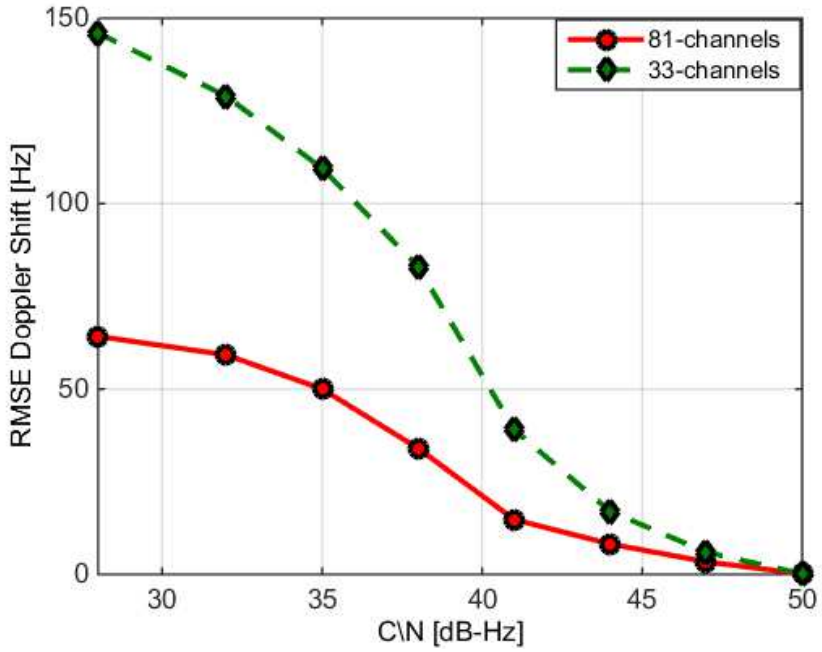


Figure 6-19 RMSE frequency vs. C\N

6.9 Conclusion on the SCSSR

Our SCSSR implementation achieves better GPS signals acquisition at much reduced computational processing by up to 80% and in dictionary matrix size than other CS-based solutions. This reduction makes our implementation method faster and, therefore, consumes less battery power than other methods. The control parameter of our method is the number of the chosen Doppler channels, i.e. increasing or decreasing these channels will directly effect on the acquisition performance. In other words, increasing the Doppler channels will increase the chance of acquiring available GPS signal; through increasing number of rows in the transform matrix and that leads to increasing the chance of sensing the samples. This will also increase the high frequency resolution that will be close to the “fine frequency” acquisition.

This SCSSR work was presented in the Computer Applications and Information Systems (WCCAIS), IEEE International Conference on Sunday, January 19, 2014.

Chapter 7

Conclusion and Future work

7.1 Conclusion

This research has focused on enhancing the acquisition of multi-GNSS signals in Software receivers to reduce the processing overhead. With the rolling out of various GNSS systems, our methods will make it more desirable for localisation providers to deploy such solutions to enhance the localisation accuracy. The technical achievements of our methods have proven that architectural approaches are aplenty for solving and enhancing multi-signal acquisition as follows:

1. Most acquisition algorithms try to match the code phase delay and Doppler frequency shift of the received signal simultaneously irrespective of it being done in the time or frequency domains. With the use of the Compressive Sensing technique, it allowed us to recover the code phase delay first before the Doppler frequency shift without loss of signal integrity or correlation quality, but with 50% saving in processing time and acquisition complexity. This accomplished by combining the codes of the GPS and the Galileo signals in a single bank of codes multiplied by a fixed carrier frequency. So the matching in our CS framework calculates the code phase of the GNSS signal from the matching in CS domain then it determines the Doppler frequency shift based on the highest code matching, as detailed in Section 6.3.3.

2. Down the processing chain thrashing (correlation and integration) can be saved by an early detection mechanism if the signal actually exists or not, before demodulating the received GNSS signal. The reasoning about this issue has led us to the use of a simple bandpass sampling receiver designed for this purpose. This is achieved by folding the GNSS signals to the FNZ with non-overlapping between them, and then examine the power at a specific IF frequency, as detailed in Section 2.3. It was a big lesson for that slicing a process over several parts can actually save processing and time.

3. As explained in Section 3.3, solving the ambiguity issue when acquiring Galileo-OS signal has been successfully overcome but at the expense of having a complex implementation or suffering from a signal-to-noise deterioration. However, we found that the subcarrier frequency can be partially removed if the received Galileo-OS signal multiplies with one of the subcarrier data or pilot channels before acquiring the signal. This multiplication converts the BOC signal to BPSK like signal and shapes the cross correlation function to have only single peak in the correlation domain.

4. The Galileo-OS signal is constructed from combining two signals of data and pilot channels in a single transmission, with the code phase delay and the Doppler frequency shift are the same in both of these channels, as detailed in Section 4.2. Therefore, for a receiver, in the acquisition process, ignoring any one of these channels means that it is losing half the power of the received signal. Our determination to process both channels without doubling the efforts has led to designing our orthogonal acquisition chain. This design can provide the same performance with half resources and processing time of a parallel/multi acquisition chains. The orthogonality is achieved by making the received Galileo signal orthogonal with a 90-degrees phase-shifted copy of itself. The only overhead of the orthogonal acquisition chain is to have both generated data and pilot codes in orthogonal format.

5. Studying the literature of receiver architectures shows that they are static receivers irrespective of how genius or optimum the implementation/algorithms are. When we achieved our saving with the CS technique, we found that CS also allows us to react to various SNR conditions, which the receiver is in (not possible with other architectures). The sensitivity of the signal acquisition is determined by the size of the sensing matrix in a CS scheme. Therefore, instead of fixing this at high sensitivity acquisition for all SNR conditions, we have developed a dynamic sensing algorithm that adjusts the acquisition channel resource allocation depending on the receiver location. i.e. it senses the SNR and adapts fewer number of measurements in open-sky environments and large number of measurements in harsh environments to keep the lock for the present signals. Full explanation of this algorithm is in section 5.3.

6. All our methods were successfully performed and assessed in realistic simulation environments. Effectively, the implementation and the performance of these methods clearly show the amount of reduction achieved in terms of the processing time and the resources requirements, which make most of these methods good candidates to be implemented in the current Smartphones.

7. Our experiments and the results therein have potential to improve the usage of Smartphones for the end user. The methodologies implemented can offer a more efficient battery life, faster and accurate positioning for multi-GNSS signal reception.

In order to highlight the outcomes of our multi and single GNSS signals implementations, Section 7.1.1 and Section 7.1.2 summarise the entire achievements.

7.1.1 My multi-GNSS research achievements

Under this particular area a novel CSSR implementation was designed to combine the acquisition of both the GPS-C/A-code signal and the Galileo-OS-code signal. Where, this implementation represents the main target of this research. Our CSSR is a 2for1 receiver because it acquires both GNSS signals at half the complexity and processing time that required by a MF acquisition process. The CSSR implementation was capitalised on the concept of the CS process that transferred the matching from the whole length of the signal to the number of rows in the sensing matrix. Our CSSR is based on 4 stages: receiving-sampling, converting the Galileo-BOC signal to the Galileo-BPSK like signal by exploiting on our previous ESCE method that eliminates the subcarrier frequency effect, generating non-Doppler shift vectors to compensate the measurement in our CS framework and finally the acquisition stage accomplished in CS domain. In our CS framework, we have combined the dictionaries of these GNSS signals in a single dictionary matrix. The typical design of the combined dictionary matrix makes acquiring both GPS+Galileo signals in single process possible rather than managing one at a time and thereby eliminating the repetition of finding the dictionary elements that represent the satellite ID, code phase delay and Doppler frequency shift.

Furthermore, CSSR implementation overcomes the fat dictionary problem by decomposing the dictionary matrix that is achieved by generating a bank of codes multiplied by a fixed carrier frequency. Consequently, the generality of such decomposition enables any GNSS signals or other wireless signals to combine their dictionary if these signals are folded or down-converted to the same IF frequency.

Our analysis showed that increasing or decreasing the range of the Doppler channels will directly affect the CS measurements, the acquisition rate and the resolution of estimating both of the Doppler frequency shift and the code phase delay.

The comparison between our CSSR implementation and other MF implementations (GPS + ambiguous and unambiguous Galileo signals acquisition), in terms of implementation requirements and detection performance, showed that:

1) The CSSR performed as good as the MF.
2) The CSSR accelerated the acquisition process by 60% than the MF.
3) The CSSR was less complex than MF by 50%.
4) The CSSR achieved high frequency resolution 10Hz-40Hz, which equivalents to the fine frequency.
5) The processing of different dwell time based on our CSSR, such as 4ms or 8ms/20ms has the same processing cost as using 4ms.

The second comparison was carried out between our CSSR implementation (GPS+Galileo) and other CS-based solutions (GPS only), such as CMS solution, denoted that:

1) The computational complexity of our CSSR was about 21%.
2) The overall memory requirement was less by 73%.

In this particular implementation there was another contribution, which was developing a 2D-OMP algorithm. This algorithm introduced a new way to solve any CS problem that has one or two dimensions. The computational complexity of this modification costs much less than when solving a CS problem that has a fat dictionary.

The second achievement is a quick-early detection algorithm that was developed to combine the GPS-L1, Galileo-E1 and GLONASS-L1-CDMA signals in single RF front-end. In this algorithm, the left-sideband of the Galileo signal and the right-sideband of the GLONASS signal were filtered out and combined with the 3rd harmonic of the GPS signal to be sampled using single ADC. The benefits of this combination are to:

1) Prevent the overlapping between these GNSS signals in the FNZ.
2) Detect quickly multi-GNSS signals in a single view by measuring the powers of the available received signals prior to the acquisition stage.
3) Stop chasing signals that are not available at the time, thus saving processing time and power.

7.1.2 My single-GNSS research achievements

In this research area, Galileo-OS-code signal acquisition and GPS-C/A-code signal acquisition methods were designed.

For the Galileo signal we have tackled the acquisition process from two aspects, ambiguity and data-pilot joining, and via two methods, which are ESCE and OGSR methods.

The ESCE method was designed to overcome the ambiguity in acquiring Galileo signal at code phase resolution ≥ 0.5 Chip, as well as to enhance the signal acquisition performance. This was obtained by eliminating the subcarrier frequency effect from the received signal, so as to simplify the acquisition process. The ESCE implementation were analysed and compared to other widely used methods, such as DSB, BPSK-Like and LoCo methods in terms of probability of detection, complexity and processing time. The assessment based on actual wireless channel experiments showed that:

1) The ESCE method effectively overcame the ambiguity from the acquisition process, which shaped the CCF of the converted Galileo-BPSK like signal to have a single peak like the CCF of the GPS-BPSK signal.
2) The conversion result showed that the ESCE elimination offers around 2dB gain to the received signal power.

3) The ESCE method had better performance than BPSK-Like and LoCo methods by 1 and 2 dB respectively.

4) The acquisition time of the ESCE was half the time required by the DSB, the BPSK-Like and the LoCo methods.

5) The computational complexity of the ESCE was about 70% less than the DSB method and the LoCo method.

6) ESCE method can be implemented in the time-domain or the frequency-domain.

The second method to acquire the Galileo-OS-code signal was the OGSR method. The novelty of this method was concentrated on joining the data and pilot signals in a single correlation chain by forming these signals in an orthogonal format. Therefore, this process when compared with the traditional time-domain or the frequency-domain joining methods definitely saves valuable resources. The implementation requirements and detection performance were compared and analysed with the DC acquisition method, and the results showed that:

1) The OGSR performed as good as DC method due to the OGSR combined the Galileo-OS data and pilot signals' powers as the DC method.

2) The computational complexity of the OGSR was 49% of the DC method.

3) The OGSR required only 35% of the time required for the DC method.

4) To allow acquiring more Galileo signals and to reduce the false alarm detection the acquisition threshold must be set to 2.

Acquiring GPS-C/A-code signal was accomplished based on the CS technique. Three methods were proposed and each one of them solved certain drawback of the previous CS-based solutions.

A novel dynamic acquisition was implemented based on CS technique denoted as DCSR. The novelty of such implementation was obtained by designing a feedback controller that determines the position of the GPS receiver, i.e. outdoors or indoors, via calculating the power of the received signal. The DCSR was designed to:

1) Overcome the hardware complexity in the CMS by replacing the complex multichannel sampler with shift registers to generate square waveforms. Thus simplifying the receiver front-end and allowing sampling the GPS signals at a low rate.

2) Change dynamically the required sensing channels and resizing the measurement matrix. This was achieved because the design of the dictionary matrix enables the measurement matrix to change its size without affecting signal compression and integrity.

3) Reduce the processing time and so minimise the power consumption required by a GPS receiver to acquire signals in outdoors. This was accomplished by dynamically altering the number of measurements and the required number of sparse channels to fit the actual signal strength.

4) Maintain the lock of the available signals in difficult signal conditions by using an extra number of channels to compensate the measurements.

To enhance the measurement in our DCSR implementation, GCSR implementation was employed a deterministic waveform such as the Hadamard or the Jacket matrices/ waveforms instead of using any square or saw-tooth periodic waveforms. The use of these deterministic waveforms produced better orthogonality than the random square waveform. The GCSR implementation pointed out that:

1) Using these deterministic orthogonal waveforms to construct the measurement matrix increased the acquisition rate by 20% more than using a random square or saw-tooth waveforms.

2) Reconstructing the signal based on GCSR implementation is simpler than the CMS solution because we have moved the measurement process to the DSP side while the other processed it on the Analogue side.

3) Tracking reconstructed signal showed a slight phase distortion and amplitude degradation of the decoded signal; however the integrity of the received signal was maintained.

Finally the decomposed dictionary matrix design was applied to acquire a single GNSS signal, the GPS-C/A-code signal that called SCSSR. The analysis revealed that:

1) Both of the computational processing and memory requirements were less by 80% than other CS-based solutions and also less than both our DCSR and GCSR methods. Consequently, this reduction makes our implementation method faster and therefore consumes less battery power than other methods.
2) Increasing the dwell time from 1ms to 4ms improved the acquisition rate by 10%. While, when increasing the number of non-Doppler channels from 33 to 81 introduced 40% improvement in the acquisition performance.
3) The RMSE of Doppler frequency was around 50Hz when deploying 81 Doppler channels, while it rose to 150Hz if 33 Doppler channels were used. However, it is still highly accurate than time and frequency implementations that are based 500Hz frequency resolution.

7.2 Future work

My future work shall continue in the area of the GNSS signals. Various schemes will be addressed; some of them will represent a development of the current achievements while others will focus on designing new methods and as follows:

1. Generalising the decomposed dictionary design by applying to other GNSS signals, for example the Galileo-E1-OS-code and the GPS-L1-C-code signal as long as they are employing the same BOC modulation technique. The challenge in such proposed designs will be how to overcome the variety of the code length, as it is known that the Galileo-E1-OS-code is 4ms length and the GPS-L1-C-code signal is 10ms length. Furthermore, combine one of the GNSS signals, such as GPS with other wireless signals that are currently used in the Smartphones devices as they fold or down-convert to the same IF frequency.
2. Despite that those non-Doppler channels have specific Doppler frequency distribution (for the 401 channels the frequency resolution is 20Hz and the 321 channels the frequency resolution is 25Hz), we shall improve either our CSSR or SCSSR implementation to devise another dynamic CS-based acquisition. The appropriate solution that can overcome this limitation is by

designing a very high frequency resolution of the non-Doppler channels up to 10Hz or 5Hz to control various types of dynamic range.

3. Capitalising on the saved dictionary and the measurement matrices to design a new tracking method for the GPS signal only. This implementation will incur less processing cost than the traditional implementation because the early and the late correlators are already generated and saved in the memory, as well as the code phase resolution is designed based on 0.5 Chip like the space between the traditional correlators. This tracking engine is not applicable for the Galileo signal unless we have high code phase resolution, i.e. less than 0.5 Chip to overcome the ambiguity, or continuing with our conversion using ECSE method.
4. Exploiting our OGSR method and applying it to the ECSE method to propose, for the first time, unambiguous-joint-data-pilot Galileo signal acquisition. Moreover, according to our previous evaluation of the OGSR and the ECSE methods, we expect that the performance of the new method will be less than the normal ambiguous-joint by 1dB and the complexity will be quarter of the mostly used unambiguous methods if they are designed for joining purpose.
5. To enhance the detection probability of the joint-data-pilot Galileo signal acquisition, i.e. the OGSR method, we shall implement a differential OGSR acquisition. This implementation will take advantage of the orthogonal format representation to have also a single correlation engine to acquire Galileo signal.
6. Finalising the proposed dynamic early-late correlator. The experiments in this particular design would be based on dynamic scenario, i.e. from outdoors to indoors and vice versa. Also, the assessment would include the proposed discriminator to validate both of the proposals. In addition, determining the threshold of the maximum space between the early and late correlators would be incorporated.

References

[1] A. Simsky, D. Mertens, J.-M. Sleewaegen, M. Hollreiser and M. Crisci, “Experimental results for the multipath performance of Galileo signals transmitted by GIOVE-A satellite,” *International Journal of Navigation and Observation*, vol. 2008, 2008.

[2] Y. Feng, “Combined Galileo and GPS: a technical perspective,” *Journal of Global Positioning Systems*, vol. 2, no. 1, pp. 67-72, 2003.

[3] B. Hofmann-Wellenhof, H. Lichtenegger and E. Wasle, GNSS--Global Navigation Satellite Systems: GPS, GLONASS, Galileo \& more, Springer, 2007.

[4] T. E. G. A. (GSA), March 2015. [Online]. Available: www.gsa.europa.eu/2015-gnss-market-report. [Accessed May 2015].

[5] M. Al-Aboodi, A. Albu-Rghaif and I. A. Lami, “GPS, Galileo and Glonass L1 signal detection algorithms based on bandpass sampling techniques,” in *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2012 4th International Congress on*, 2012.

[6] F. S. T. V. Diggelen, A-GPS: Assisted GPS, GNSS, and SBAS, Artech House Publishers, 2009.

[7] J. Chandrasekhar and C. R. Murthy, “GNSS Signal Detection Under Noise Uncertainty,” in *Communications (ICC), 2010 IEEE International Conference on*, 2010.

[8] D. Akos, A software radio approach to global navigation satellite system receiver design, Ohio University, 1997.

[9] R. Qi, F. Coakley and B. Evans, “Practical consideration for bandpass sampling,” *Electronics Letters*, vol. 32, no. 20, pp. 1861-1862, sep 1996.

[10] P. M. Cruz and N. B. Carvalho, “Modeling band-pass sampling receivers nonlinear behavior in different Nyquist zones,” in *Microwave Symposium Digest (MTT), 2010 IEEE MTT-S International*, 2010.

[11] R. Vaughan, N. Scott and D. White, “The theory of bandpass sampling,” *Signal Processing, IEEE Transactions on,* vol. 39, no. 9, pp. 1973-1984, sep 1991.

[12] D. M. Akos, M. Stockmaster, J. B. Tsui and J. Caschera, “Direct bandpass sampling of multiple distinct RF signals,” *Communications, IEEE Transactions on,* vol. 47, no. 7, pp. 983-988, 1999.

[13] D. M. Akos, A. Ene and J. Thor, “A Prototyping Platform for Multi-Frequency GNSS Receivers,” in *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003),* 2001.

[14] A. Alonso, J. Perre and I. Arizaga, “SDR direct-sampling device for multistandard GNSS signals,” in *Communication Systems, Networks and Digital Signal Processing, 2008. CNSDSP 2008. 6th International Symposium on*, 2008.

[15] E. R. Parada, F. Chastellain, C. Botteron, Y. Tawk and P.-A. Farine, “Design of a GPS and Galileo Multi-Frequency Front-End,” in *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, 2009.

[16] A. Attaby, K. M. Sharaf and M. M. Ibrahim, “Frequency planning for multi-frequency sub-sampling receiver and its application for UMTS/WLAN and GNSS receivers,” in *Microelectronics (ICM), 2011 International Conference on*, 2011.

[17] A. Noroozi, C. Verhoeven, G. Monna and E. Gill, “A direct conversion GPS/Galileo receiver front-end for space applications,” in *Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2010 5th ESA Workshop on*, 2010.

[18] Z. Zhang, W. Li, W. Wen, W. Wu and Y. Li, “A configurable multi-band GNSS receiver for Compass/GPS/Galileo applications,” in *Circuits and Systems*

(ISCAS), 2013 IEEE International Symposium on, 2013.

[19] R. Lopez La Valle, J. G. Garcia, P. A. Roncagliolo and C. H. Muravchik, “An Experimental L1/L2 GNSS Receiver for High Precision Applications,” *Latin America Transactions, IEEE (Revista IEEE America Latina),* vol. 11, no. 1, pp. 48-53, 2013.

[20] S. A. Nik and M. Petovello, “Implementation of a Dual-frequency GLONASS and GPS L1 C/A Software Receiver,” *Journal of Navigation,* vol. 63, no. 02, pp. 269-287, 2010.

[21] P. M. Fishman and J. W. Betz, “Predicting performance of direct acquisition for the M-code signal,” in *Proceedings of the 2000 National Technical Meeting of The Institute of Navigation*, 2000.

[22] N. Martin, V. Leblond, G. Guillotel and V. Heiries, “BOC (x, y) signal acquisition techniques and performances,” in *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*, 2001.

[23] J. Ray, Z. Altamimi, X. Collilieux and T. van Dam, “Anomalous harmonics in the spectra of GPS position estimates,” *GPS Solutions*, vol. 12, no. 1, pp. 55-64, 2008.

[24] G. W. Hein, J.-A. Avila-Rodriguez, S. Wallner, A. R. Pratt, J. Owen, J.-L. Issler, J. W. Betz, C. J. Hegarty, L. S. Lenahan, J. J. Rushanan and others, “MBOC: the new optimized spreading modulation recommended for GALILEO L1 OS and GPS L1C,” in *Proceedings of the IEEE/ION position, location, and navigation symposium*, 2006.

[25] O. ICD, *European GNSS (Galileo) open service. Signal in space. Interface control document. Issue 1.1,* 2010.

[26] F. Xiangning and H. Shuancheng, “Unambiguous tracking techniques for Galileo BOC (1, 1) signals,” in *Circuits and Systems, 2008. APCCAS 2008. IEEE Asia Pacific Conference on*, 2008.

[27] E. S. Lohan, “Analytical performance of CBOC-modulated Galileo E1 signal using sine BOC (1, 1) receiver for mass-market applications,” in *Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, 2010.

[28] L. Elena Simona, “Limited bandwidths and correlation ambiguities: Do they co-exist in Galileo receivers,” *Positioning*, vol. 2011, 2011.

[29] E. D. Kaplan and C. J. Hegarty, Understanding GPS: principles and applications, Artech House Publishers, 2006.

[30] K. Borre, D. M. Akos, N. Bertelsen, P. Rinder and S. H. Jensen, A software-defined GPS and Galileo receiver, Birkhhauser, 2007.

[31] J. Wu and Y. Hu, “The study on GPS signal acquisition algorithm in time domain,” in *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference on*, 2008.

[32] Q. Lei and L. Lei, “GPS signal acquisition based on FFT,” in *Information Technology and Computer Science (ITCS), 2010 Second International Conference on*, 2010.

[33] J. BAO and Y. TSUI, Fundamentals of Global Positioning System Receivers: A Software Approach, A JOHN WILEY & SONS, INC. Publishers, 2005.

[34] T. H. Ta, N. C. Shivaramaiah and A. G. Dempster, “Significance of Cell Correlations in Matched Filter GPS Acquisition Engines,” 2009.

[35] A. Burian, E. S. Lohan and M. Renfors, “BPSK-like methods for hybrid-search acquisition of Galileo signals,” in *Communications, 2006. ICC'06. IEEE International Conference on*, 2006.

[36] E. S. Lohan, A. Burian and M. Renfors, “Low-complexity unambiguous acquisition methods for BOC-modulated CDMA signals,” *International Journal of Satellite Communications and Networking*, vol. 26, no. 6, pp. 503-522, 2008.

[37] W.-L. Mao, C.-S. Hwang, C.-W. Hung, J. Sheen and P.-H. Chen, “Unambiguous BPSK-like CSC method for Galileo acquisition,” in *Methods and Models in*

Automation and Robotics (MMAR), 2013 18th International Conference on, 2013.

[38] V. Heiries, D. Roviras, L. Ries and V. Calmettes, “Analysis of non ambiguous BOC signal acquisition performance Acquisition,” 2004.

[39] Z. Yang, Z. Huang and S. Geng, “unambiguous acquisition performance analysis of boc (m, n) signal,” in *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*, 2009.

[40] F. Benedetto, G. Giunta, E. S. Lohan and M. Renfors, “A Fast Unambiguous Acquisition Algorithm for BOC-Modulated Signals,” *Vehicular Technology, IEEE Transactions on,* vol. 62, no. 3, pp. 1350-1355, 2013.

[41] Y. Lee and S. Yoon, “A Side-Peak Cancellation Scheme for CBOC Code Acquisition,” *World Academy of Science, Engineering and Technology,* vol. 69, 2012.

[42] H. Pham-Viet, C. Dao-Ngoc and K. Nguyen-Van, “A scheme of cosine-BOC (n, n) side peaks cancellation for navigation applications,” in *Advanced Technologies for Communications (ATC), 2013 International Conference on*, 2013.

[43] S. Arora and B. Barak, Computational complexity: a modern approach, Cambridge University Press, 2009.

[44] D. Borio and L. Lo Presti, “Data and pilot combining for composite GNSS signal acquisition,” *International Journal of Navigation and Observation*, vol. 2008, 2008.

[45] B. A. Siddiqui, J. Zhang, M. Z. H. Bhuiyan and E. S. Lohan, “Joint data-pilot acquisition and tracking of Galileo E1 open service signal,” in *Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), 2010*, 2010.

[46] Y. Tawk, A. Jovanovic, J. Leclere, C. Botteron and P.-A. Farine, “A new FFT-

based algorithm for secondary code acquisition for Galileo signals," in *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, 2011.

[47] T. H. Ta, L. L. Presti, F. Dovis, D. Margaria and R. Lesca, "Differential Joint Data/Pilot Acquisition Strategies for Indoor Galileo E1 Signal," in *proceedings of ION GNSS 21st International Technical Meeting of the Satellite Division, Savannah GA*, 2008.

[48] Q. Wan and W. Meng, "Space-Differential Cooperative acquisition for Galileo E1 OS signals," in *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, 2010.

[49] T. H. Ta, F. Dovis, D. Margaria and L. L. Presti, "Comparative study on joint data/pilot strategies for high sensitivity Galileo E1 open service signal acquisition," *IET radar, sonar \& navigation*, vol. 4, no. 6, pp. 764-779, 2010.

[50] I. A. Lami and M. Ai-Aboodi, "OBPSR: A multi-signal receiver based on the orthogonal and bandpass sampling techniques," in *Computer Applications Technology (ICCAT), 2013 International Conference on*, 2013.

[51] J. B.-Y. Tsui, Fundamentals of global positioning system receivers, Wiley-Interscience, 2000.

[52] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21-30, 2008.

[53] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289-1306, 2006.

[54] Y. C. Eldar and G. Kutyniok, Compressed sensing: theory and applications, Cambridge University Press, 2012.

[55] S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud and R. Baraniuk, "Analog-to-information conversion via random demodulation," in *Design, Applications, Integration and Software, 2006 IEEE Dallas/CAS Workshop on*, 2006.

[56] E. J. Candes, “The restricted isometry property and its implications for compressed sensing,” *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589-592, 2008.

[57] X. Li, Y. C. Eldar and A. Scaglione, “Low complexity acquisition of GPS signals,” in *Signal Processing Advances in Wireless Communications (SPAWC), 2011 IEEE 12th International Workshop on*, 2011.

[58] S. Kong, “A Compressed Sensing Technique for GPS Signal Acquisition,” in *Proceedings of the 2012 International Technical Meeting of The Institute of Navigation*, 2012.

[59] J. A. Tropp, J. N. Laska, M. F. Duarte, J. K. Romberg and R. G. Baraniuk, “Beyond Nyquist: Efficient sampling of sparse bandlimited signals,” *Information Theory, IEEE Transactions on*, vol. 56, no. 1, pp. 520-544, 2010.

[60] K. Fyhn, T. Arildsen, T. Larsen and S. H. Jensen, “Demodulating subsampled direct sequence spread spectrum signals using compressive signal processing,” in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, 2012.

[61] M. Mishali, Y. C. Eldar, O. Dounaevsky and E. Shoshan, “Xampling: Analog to digital at sub-Nyquist rates,” *IET circuits, devices \& systems*, vol. 5, no. 1, pp. 8-20, 2011.

[62] M. Mishali and Y. C. Eldar, “From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 375-391, 2010.

[63] M. Mishali and Y. C. Eldar, “Blind multiband signal reconstruction: Compressed sensing for analog signals,” *Signal Processing, IEEE Transactions on*, vol. 57, no. 3, pp. 993-1009, 2009.

[64] M. Mishali and Y. C. Eldar, “Reduce and boost: Recovering arbitrary sets of jointly sparse vectors,” *Signal Processing, IEEE Transactions on*, vol. 56, no. 10, pp. 4692-4702, 2008.

[65] Y. C. Pati, R. Rezaiifar and P. Krishnaprasad, “Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition,” in *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, 1993.

[66] J. P. Slavinsky, J. N. Laska, M. A. Davenport and R. G. Baraniuk, “The compressive multiplexer for multi-channel compressive sensing,” in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011.

[67] J. Yoo, S. Becker, M. Monge, M. Loh, E. Candes and A. Emami-Neyestanak, “Design and implementation of a fully integrated compressed-sensing signal acquisition system,” in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012.

[68] E.-U. C. o. S. Navigation, “Combined Performances for Open GPS/GALILEO Receivers,” 2010.

[69] S.-H. Kong, “A Deterministic Compressed GNSS Acquisition Technique,” 2013.

[70] Y. Ji, X. Wu, J. Yan and W. Zhu, “GPS signal acquisition via folding and compressive sensing,” in *Communication Technology (ICCT), 2013 15th IEEE International Conference on*, 2013.

[71] P. K. Misra, W. Hu, Y. Jin, J. Liu, A. Souza de Paula, N. Wirstrom and T. Voigt, “Energy efficient GPS acquisition with sparse-gps,” in *Proceedings of the 13th international symposium on Information processing in sensor networks*, 2014.

[72] K. Wei, W. Zhiping, Z. Yongxue and B. Bo, “New compress sampling algorithm for FFT-based GPS signal acquisition,” in *Convergence Information Technology, 2007. International Conference on*, 2007.

[73] A. Joseph, “GNSS Solutions: Measuring GNSS Signal Strength,” *Inside GNSS-Engineering Solutions for the Global Navigation Satellite System Community,* vol. 5, no. 8, pp. 20-25, 2010.

[74] T. Blumensath and M. E. Davies, “Gradient pursuits,” *Signal Processing, IEEE Transactions on*, vol. 56, no. 6, pp. 2370-2382, 2008.

[75] M. H. Lee, “Jacket Matrix and Its Applications to Signal Processing,” in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2011 IEEE 10th International Conference on*, 2011.

[76] D. Mesecher, L. Carin, I. Kadar and R. Pirich, “Exploiting signal sparseness for reduced-rate sampling,” in *Systems, Applications and Technology Conference, 2009. LISAT'09. IEEE Long Island*, 2009.

[77] T. L. Hansen, D. Johansen, P. Jorgensen, K. F. Trillingsgaard, T. Arildsen, K. Fyhn and T. Larsen, “Compressed sensing with rank deficient dictionaries,” in *Global Communications Conference (GLOBECOM), 2012 IEEE*, 2012.

[78] W. De Wilde, J.-M. Sleewaegen, A. Simsky, J. Van Hees, C. Vandewiele, E. Peeters, J. Grauwen and F. Boon, “Fast signal acquisition technology for new GPS/Galileo receivers,” *Proc. of IEEE/ION PLANS’06*, pp. 1074-1079, 2006.

[79] G. Heinrichs, N. Lemke, A. Schmid, A. Neubauer, R. Kronberger, G. Rohmer, F. Forster, J. A. Avila-Rodriguez, T. Pany, B. Eissfeller and others, “Galileo/GPS receiver architecture for high sensitivity acquisition,” in *Proc. of the International Symposium on Signals, Systems, and Electronics (ISSSE'04)*, 2004.

[80] G. Artaud, L. Ries, H. Al-Bitar, M. Monnerat, F. Legrand and M. Weyer, “Development of a flexible real time GNSS software receiver,” in *Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2010 5th ESA Workshop on*, 2010.

[81] J. Arribas, P. Closas and C. F. Prades, “Joint acquisition strategy of GNSS satellites for computational cost reduction,” in *Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2010 5th ESA Workshop on*, 2010.

[82] P. Kovavr, P. Kavcmavrik and F. Vejravzka, “Interoperable GPS, GLONASS

and Galileo software receiver," *Gyroscopy and Navigation*, vol. 2, no. 2, pp. 69-74, 2011.

[83] C. Gernot, "Development of combined GPS L1/L2C acquisition and tracking methods for weak signals environments," *Department of Geomatics Engineering*, 2009.

[84] D. Zeng and J. Li, "GPS signal fine acquisition algorithm," in *Information Science and Engineering (ICISE), 2010 2nd International Conference on*, 2010.

[85] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, pp. 3397-3415, 1993.

[86] I. A. Lami, A. Albu-Rghaif and M. Al-Aboodi, *GCSR: A GPS acquisition technique using compressive sensing enhanced implementation*, IJEIT, 2013.